

SIEMENS

SINUMERIK

SINUMERIK 840D sl / 828D Przygotowanie do pracy

Podręcznik programowania

Obowiązuje dla

Sterowania
SINUMERIK 840D sl / 840DE sl
SINUMERIK 828D

Oprogramowanie
oprogramowania CNC


Wersja
4.5 SP2


Słowo wstępne	
Elastyczne programowanie NC	1
Zarządzanie plikami i programami	2
Obszary ochrony	3
Specjalne polecenia wykonania ruchu	4
Transformacje współrzędnych (Frame)	5
Transformacje	6
Łańcuchy kinematyczne	7
Uniknięcie kolizji za pomocą łańcuchów kinematycznych	8
Korekcje narzędzi	9
Zachowanie się w ruchu po torze	10
Sprzężenia osi	11
Akcje synchroniczne	12
Ruch wahliwy	13
Tłoczenie i cięcie	14
Szlifowanie	15
Dalsze funkcje	16
Własne programy obróbki	17
Zewnętrzne programowanie cykli	18
Tablice	19
Aneks	A


Wskazówki prawne

Koncepcja wskazówek ostrzeżeń

Podręcznik zawiera wskazówki, które należy bezwzględnie przestrzegać dla zachowania bezpieczeństwa oraz w celu uniknięcia szkód materialnych. Wskazówki dot. bezpieczeństwa oznaczone trójkątnym symbolem, ostrzeżenia o możliwości wystąpienia szkód materialnych nie posiadają trójkątnego symbolu ostrzegawczego. W zależności od opisywanego stopnia zagrożenia, wskazówki ostrzegawcze podzielono w następujący sposób.

 NIEBEZPIECZEŃSTWO
oznacza, że nieprzestrzeganie tego typu wskazówek ostrzegawczych grozi śmiercią lub odniesieniem ciężkich obrażeń ciała.

 OSTRZEŻENIE
oznacza, że nieprzestrzeganie tego typu wskazówek ostrzegawczych może grozić śmiercią lub odniesieniem ciężkich obrażeń ciała.

 OSTROŻNIE
oznacza, że nieprzestrzeganie tego typu wskazówek ostrzegawczych może spowodować lekkie obrażenia ciała.

UWAGA
oznacza, że nieprzestrzeganie tego typu wskazówek ostrzegawczych może spowodować szkody materialne.


W wypadku możliwości wystąpienia kilku stopni zagrożenia, wskazówkę ostrzegawczą oznaczono symbolem najwyższego z możliwych stopnia zagrożenia. Wskazówka oznaczona symbolem ostrzegawczym w postaci trójkąta, informująca o istniejącym zagrożeniu dla osób, może być również wykorzystana do ostrzeżenia przed możliwością wystąpienia szkód materialnych.

Wykwalifikowany personel

Produkt /system przynależny do niniejszej dokumentacji może być obsługiwany wyłącznie przez **personel wykwalifikowany** do wykonywania danych zadań z uwzględnieniem stosownej dokumentacji, a zwłaszcza zawartych w niej wskazówek dotyczących bezpieczeństwa i ostrzegawczych. Z uwagi na swoje wykształcenie i doświadczenie wykwalifikowany personel potrafi podczas pracy z tymi produktami / systemami rozpoznać ryzyka i unikać możliwych zagrożeń.

Zgodne z przeznaczeniem używanie produktów firmy Siemens

Przestrzegać następujących wskazówek:

 OSTRZEŻENIE
Produkty firmy Siemens mogą być stosowane wyłącznie w celach, które zostały opisane w katalogu oraz w załączonej dokumentacji technicznej. Polecenie lub zalecenie firmy Siemens jest warunkiem użycia produktów bądź komponentów innych producentów. Warunkiem niezawodnego i bezpiecznego działania tych produktów są prawidłowe transport, przechowywanie, ustawienie, montaż, instalacja, uruchomienie, obsługa i konserwacja. Należy przestrzegać dopuszczalnych warunków otoczenia. Należy przestrzegać wskazówek zawartych w przynależnej dokumentacji.

Znaki towarowe

Wszystkie produkty oznaczone symbolem ® są zarejestrowanymi znakami towarowymi firmy Siemens AG. Pozostałe produkty posiadające również ten symbol mogą być znakami towarowymi, których wykorzystywanie przez osoby trzecie dla własnych celów może naruszać prawa autorskie właściciela danego znaku towarowego.

Wykluczenie od odpowiedzialności

Treść drukowanej dokumentacji została sprawdzona pod kątem zgodności z opisywanym w niej sprzętem i oprogramowaniem. Nie można jednak wykluczyć pewnych rozbieżności i dlatego producent nie jest w stanie zagwarantować całkowitej zgodności. Informacje i dane w niniejszej dokumentacji poddawane są ciągłej kontroli. Poprawki i aktualizacje ukazują się zawsze w kolejnych wydaniach.

Słowo wstępne

Dokumentacja SINUMERIK

Dokumentacja SINUMERIK jest podzielona na następujące kategorie:

- Dokumentacja ogólna
- Dokumentacja użytkownika
- Dokumentacja producenta/serwisowa

Bardziej szczegółowa informacja

Pod linkiem www.siemens.com/motioncontrol/docu można znaleźć informacje na następujące tematy:

- Zamawianie dokumentacji / przegląd druków
- Bardziej szczegółowe linki do download dokumentacji
- Korzystanie z dokumentacji online (szukanie i przeglądanie podręczników/informacji)

W przypadku pytań dot. dokumentacji technicznej (np. propozycje, korekty) proszę wysłać e-mail na następujący adres:

docu.motioncontrol@siemens.com

My Documentation Manager (MDM)

Pod poniższym linkiem można znaleźć informacje potrzebne do tego, by na bazie treści Siemens indywidualnie zestawić specyficzną dla OEM dokumentację maszyny:

www.siemens.com/mdm

Szkolenie

Informacje dot. oferty szkoleniowej można znaleźć pod:

- www.siemens.com/sitrain
SITRAIN - prowadzine przez Siemens szkolenie dot. produktów, systemów i rozwiązań techniki automatyzacji
- www.siemens.com/sinustrain
SinuTrain - oprogramowanie szkoleniowe dla SINUMERIK

FAQ

Frequently Asked Questions można znaleźć na stronach Service&Support pod Produkt Support. <http://support.automation.siemens.com>

SINUMERIK

Informacje dot. SINUMERIK można znaleźć pod następującym linkiem:
www.siemens.com/sinumerik

Adresaci

Niniejszy druk jest przeznaczony dla:

- programistów
- projektantów

Korzyści

Podręcznik programowania umożliwia adresatom projektowanie, pisanie i testowanie programów i softwareowych interfejsów graficznych oraz usuwanie błędów.

Zakres standardowy

W niniejszej instrukcji programowania opisano funkcje zakresu standardowego. Uzupełnienia albo zmiany, które zostały dokonane przez producenta maszyny, są przez niego dokumentowane.

W sterowaniu mogą być możliwe do realizacji dalsze funkcje, nie opisane w niniejszej dokumentacji. Nie ma jednak roszczenia do tych funkcji w przypadku dostawy nowego sterowania albo wykonania usługi serwisowej.

Ze względu na przejrzystość, dokumentacja nie zawiera również wszystkich informacji szczegółowych dot. wszystkich typów produktu i może nie uwzględniać każdego przypadku ustawienia, pracy i utrzymania.

Wsparcie techniczne

Specyficzne dla kraju numery telefonów doradztwa technicznego można znaleźć w internecie pod <http://www.siemens.com/automation/service&support>

Informacje odnośnie struktury i treści

Podręcznik programowania "Podstawy" i "Przygotowanie pracy"

Opisy do programowania NC są podzielone na dwa podręczniki:

1. Podstawy

Podręcznik programowania "Podstawy" służy fachowemu operatorowi przy maszynie i zakłada posiadanie odpowiedniej wiedzy w zakresie obróbki wiertarskiej, frezarskiej i tokarskiej. Na prostych przykładach programowania zostaną objaśnione polecenia i instrukcje znane również z DIN66025.

2. Przygotowanie do pracy

Podręcznik programowania "Przygotowanie do pracy" służy technologowi znającemu wszystkie możliwości programowania. Sterowanie SINUMERIK umożliwia przy pomocy specjalnego języka programowania sporządzenie skomplikowanego programu obróbki (np. powierzchnie swobodne, koordynacja kanałów, ...) i ułatwia technologom pracochłonne programowanie.

Dostępność opisanych elementów językowych NC

Wszystkie elementy językowe opisane w niniejszym podręczniku są dostępne dla SINUMERIK 840D sl. Dostępność odnośnie SINUMERIK 828D należy przeczytać w tablicy "Instrukcje: Dostępność w przypadku SINUMERIK 828D (Strona 797)".

Spis treści

	Słowo wstępne	3
1	Elastyczne programowanie NC.....	17
1.1	Zmienne	17
1.1.1	Zmienna systemowa	18
1.1.2	Predefiniowane zmienne użytkownika: parametry obliczeniowe (R).....	20
1.1.3	Predefiniowane zmienne użytkownika: zmienne link.....	22
1.1.4	Definicja zmiennych użytkownika (DEF).....	24
1.1.5	Redefinicja zmiennych systemowych, zmiennych użytkownika i poleceń językowych NC (REDEF).....	30
1.1.6	Atrybut: wartość inicjalizacyjna	33
1.1.7	Atrybut: wartości graniczne (LLI, ULI).....	36
1.1.8	Atrybut: jednostka fizyczna (PHU)	38
1.1.9	Atrybut: prawa dostępu (APR, APW, APRP, APWP, APRB, APWB).....	40
1.1.10	Przegląd atrybutów definiowalnych i redefiniowalnych.....	45
1.1.11	Definicja i inicjalizacja zmiennych tablicowych (DEF, SET, REP)	46
1.1.12	Definicja i inicjalizacja zmiennych tablicowych (DEF, SET, REP): Dalsze informacje	50
1.1.13	Typy danych.....	53
1.1.14	Konwersje explicite typu danych (AXTOINT, INTTOAX).....	54
1.1.15	Sprawdzenie istnienia zmiennej (ISVAR)	55
1.1.16	Odczyt wartości absolutnych/typu danych (GETVARPHU, GETVARAP, GETVARLIM, GETVARDFT, GETVARTYP)	57
1.2	Programowanie pośrednie	62
1.2.1	Pośrednie programowanie adresów	62
1.2.2	Pośrednie programowanie G-Code	65
1.2.3	Programowanie pośrednie atrybutów pozycji (GP).....	66
1.2.4	Pośrednie programowanie wierszy programu obróbki (EXECSTRING).....	69
1.3	Funkcje arytmetyczne	70
1.4	Operacje porównania i operacje logiczne.....	73
1.5	Korekcja dokładności w przypadku błędów porównania (TRUNC)	75
1.6	Minimum, maksimum i zakres zmiennych (MINVAL, MAXVAL i BOUND).....	77
1.7	Priorytet operacji	79
1.8	Możliwe konwersje typu	80
1.9	Operacja na łańcuchach znaków	81
1.9.1	Konwersja typu na STRING (AXSTRING).....	81
1.9.2	Konwersja typu z STRING (NUMBER, ISNUMBER, AXNAME)	82
1.9.3	Powiązanie łańcuchów znaków (<<).....	84
1.9.4	Zmiana na litery małe/duże (TOLOWER, TOUPPER).....	85
1.9.5	Określenie długości łańcucha znaków (STRLEN)	86
1.9.6	Szukanie znaku/łańcucha znaków w łańcuchu znaków (INDEX, RINDEX, MINDEX, MATCH)	86
1.9.7	Wybór częściowego łańcucha znaków (SUBSTR)	88

1.9.8	Odczyt i zapis pojedynczych znaków.....	89
1.9.9	Formatowanie łańcucha znaków (SPRINT).....	90
1.10	Skoki i rozgałęzienia w programie	98
1.10.1	Skok do początku programu (GOTOS).....	98
1.10.2	Skoki w programie do znaczników skoku (GOTOB, GOTOF, GOTO, GOTOC).....	99
1.10.3	Rozgałęzienie programu (CASE ... OF ... DEFAULT ...)	102
1.11	Powtórzenie programu (REPEAT, REPEATB, ENDLABEL, P).....	104
1.12	Struktury kontrolne	110
1.12.1	Warunkowa instrukcja i rozgałęzienie (IF, ELSE, ENDIF)	112
1.12.2	Pętla programowa bez końca (LOOP, ENDLOOP)	113
1.12.3	Pętla FOR (FOR ... TO ..., ENDFOR)	114
1.12.4	Pętla programowa z warunkiem na początku pętli (WHILE, ENDWHILE)	116
1.12.5	Pętla programowa z warunkiem na końcu (REPEAT, UNTIL).....	117
1.12.6	Przykład programu z kaskadowanymi strukturami kontrolnymi	117
1.13	Koordinacja programu (INIT, START, WAITM, WAITMC, WAITE, SETM, CLEARM)	118
1.14	Procedura przerwania (ASUP).....	123
1.14.1	Funkcja procedury przerwania	123
1.14.2	Sporządzenie procedury przerwania	125
1.14.3	Przyporządkowanie i uruchomienie procedury przerwania (SETINT, PRIO, BLSYNC).....	126
1.14.4	Wyłączenie aktywności / reaktywowanie przyporządkowania procedury przerwania (DISABLE, ENABLE)	127
1.14.5	Skasowanie przyporządkowania procedury przerwania (CLRINT)	128
1.14.6	Szybkie cofnięcie od konturu (SETINT LIFTFAST, ALF).....	129
1.14.7	Kierunek ruchu przy szybkim cofnięciu od konturu.....	131
1.14.8	Przebieg ruchów w przypadku procedur przerwania	135
1.15	Zamiana osi, zamiana wrzecion (RELEASE, GET, GETD).....	136
1.16	Przekazanie osi do innego kanału (AXTOCHAN).....	140
1.17	Ustawienie działania danych maszynowych (NEWCONF).....	142
1.18	Zapisanie pliku (WRITE)	143
1.19	Skasowanie pliku (DELETE).....	148
1.20	Odczyt wierszy w pliku (READ)	150
1.21	Sprawdzenie istnienia pliku (ISFILE)	153
1.22	Odczyt informacji o pliku (FILEDATE, FILETIME, FILESIZE, FILESTAT, FILEINFO)	155
1.23	Zaokrąglenie do góry (ROUNDUP).....	158
1.24	Technika podprogramów.....	159
1.24.1	Informacje ogólnie	159
1.24.1.1	Podprogram	159
1.24.1.2	Nazwy podprogramów	160
1.24.1.3	Kaskadowanie podprogramów.....	161
1.24.1.4	Ścieżka szukania	162
1.24.1.5	Parametry formalne i aktualne	162
1.24.1.6	Przekazanie parametrów	163
1.24.2	Definicja podprogramu	165
1.24.2.1	Podprogram bez przekazania parametrów	165
1.24.2.2	Podprogram z przekazaniem parametrów Call-by-Value (PROC)	166

1.24.2.3	Podprogram z przekazaniem parametrów Call-by-Reference (PROC, VAR)	167
1.24.2.4	Zapisywanie modalnych funkcji G (SAVE)	170
1.24.2.5	Blokowanie wykonywania pojedynczymi blokami (SBLOF, SBLON)	171
1.24.2.6	Blokowanie wyświetlania aktualnego bloku (DISPLOF, DISPLON, ACTBLOCNO).....	177
1.24.2.7	Oznakowanie podprogramów z przygotowaniem (PREPRO)	180
1.24.2.8	Powrót z podprogramu M17.....	181
1.24.2.9	Skok powrotny z podprogramu RET	182
1.24.2.10	Parametryzowalny powrót z podprogramu (RET ...)	183
1.24.3	Wywołanie podprogramu	189
1.24.3.1	Wywołanie podprogramu bez przekazania parametrów.....	189
1.24.3.2	Wywołanie podprogramu z przekazaniem parametrów (EXTERN).....	191
1.24.3.3	Liczba powtórzeń programu (P).....	193
1.24.3.4	Modalne wywołanie podprogramu (MCALL).....	195
1.24.3.5	Pośrednie wywołanie podprogramu (CALL)	197
1.24.3.6	Pośrednie wywołanie podprogramu z podaniem części programu do wykonania (CALL BLOCK ... TO ...).....	198
1.24.3.7	Pośrednie wywołanie programu programowanego w języku ISO (ISOCALL).....	199
1.24.3.8	Wywołanie podprogramu z podaniem ścieżki i parametrami (PCALL).....	200
1.24.3.9	Rozszerzenie ścieżki szukania przy wywołaniach podprogramu (CALLPATH)	201
1.24.3.10	Wykonywanie podprogramu zewnętrznego (840D sl) (EXTCALL)	202
1.24.3.11	Wykonywanie podprogramu zewnętrznego (828D) (EXTCALL)	206
1.25	Technika makr (DEFINE ... AS).....	210
2	Zarządzanie plikami i programami	213
2.1	Pamięć programów	213
2.2	Pamięć robocza (CHANDATA, COMPLETE, INITIAL).....	217
3	Obszary ochrony.....	221
3.1	Ustalenie obszarów ochrony (CPROTDEF, NPROTDEF)	221
3.2	Uaktywnienie / wyłączenie aktywności obszarów ochrony (CPROT, NPROT)	224
3.3	Sprawdzenie naruszenia obszaru ochrony, ograniczenia obszaru pracy oraz wyłączników krańcowych oprogramowania (CALCPOSI)	228
4	Specjalne polecenia wykonania ruchu	233
4.1	Ruch do pozycji kodowanych (CAC, CIC, CDC, CACP, CACN).....	233
4.2	Interpolacja Spline (ASPLINE, BSPLINE, CSPLINE, BAUTO, BNAT, BTAN, EAUTO, ENAT, ETAN, PW, SD, PL)	234
4.3	Zespół spline (SPLINEPATH).....	244
4.4	Kompresja bloków NC (COMPON, COMPCURV, COMPCAD, COMPOF).....	245
4.5	Interpolacja wielomianowa (POLY, POLYPATH, PO, PL).....	248
4.6	Ustawiane odniesienie do toru (SPATH, UPATH).....	254
4.7	Pomiar przy pomocy sondy przełączającej, MEAS, MEAW	257
4.8	Pomiar osiowy (MEASA, MEAWA, MEAC) (Opcja).....	260
4.9	Funkcje specjalne dla użytkownika OEM (OMA1 ... OMA5, OEMIPO1, OEMIPO2, G810 ... G829)	271
4.10	Zmniejszenie posuwu ze zwłoką na narożach (FENDNORM, G62, G621)	272

4.11	Programowane kryterium końca ruchu (FINEA, COARSEA, IPOENDA, IPOBRKA, ADISPOSA).....	273
5	Transformacje współrzędnych (Frame).....	277
5.1	Transformacja współrzędnych przez zmienną frame	277
5.1.1	Predefiniowana zmienna frame (\$P_BFRAME, \$P_IFRAME, \$P_PFRAME, \$P_ACTFRAME).....	279
5.2	Przyporządkowanie wartości zmiennym frame / frame	284
5.2.1	Bezpośrednie przyporządkowanie wartości (wartość osi, kąt, skala).....	284
5.2.2	Odczyt i zmiana komponentów frame (TR, FI, RT, SC, MI)	287
5.2.3	Powiązanie kompletnych frame	288
5.2.4	Definicja nowych frame (DEF FRAME).....	289
5.3	Przesunięcie zgrubne i dokładne (CFINE, CTRANS).....	290
5.4	Zewnętrzne przesunięcie punktu zerowego.....	292
5.5	Przesunięcie preset przy pomocy PRESETON	293
5.6	Obliczenie frame z 3 punktów pomiarowych w przestrzeni (MEAFRAME)	294
5.7	Frame globalne dla NCU.....	298
5.7.1	Frame kanałowe (\$P_CHBFR, \$P_UBFR)	299
5.7.2	Frame działające w kanale.....	300
6	Transformacje.....	305
6.1	Ogólne programowanie rodzajów transformacji.....	305
6.1.1	Ruchy orientacji przy transformacjach	307
6.1.2	Przegląd transformacji orientacji TRAORI	311
6.2	Transformacje trzy-, cztero- i pięcioosiowe (TRAORI)	313
6.2.1	Zależności ogólne głowicy narzędziowej Kardana	313
6.2.2	Transformacje trzy-, cztero- i pięcioosiowe (TRAORI)	316
6.2.3	Warianty programowania orientacji i położenie podstawowe (OTIRESET).....	317
6.2.4	Programowanie orientacji narzędzia (A..., B..., C..., LEAD, TILT).....	319
6.2.5	Frezowanie czołowe (A4, B4, C4, A5, B5, C5)	325
6.2.6	Odniesienie osi orientacji (ORIWKS, ORIMKS).....	327
6.2.7	Programowanie osi orientacji (ORIXES, ORIVECT, ORIEULER, ORIRPY, ORIRPY2, ORIVIRT1, ORIVIRT2).....	329
6.2.8	Programowanie orientacji wzdłuż pobocznic stożka (ORIPLANE, ORICONCW, ORICONCCW, ORICONTO, ORICONIO)	332
6.2.9	Zadanie orientacji dwóch punktów kontaktowych (ORICURVE, PO[XH]=, PO[YH]=, PO[ZH]=)	335
6.3	Wielomiany orientacji (PO[kąt], PO[współrzędna])	337
6.4	Obroty orientacji narzędzia (ORIROTA, ORIROTR, ORIROTT, ORIROTC, THETA).....	339
6.5	Orientacje w stosunku do toru	341
6.5.1	Rodzaje orientacji w stosunku do toru	341
6.5.2	Odniesiony do toru obrót orientacji narzędzia (ORIPATH, ORIPATHS, kąt obrotu)	343
6.5.3	Interpolacja obrotu narzędzia względem toru (ORIROTC, THETA)	344
6.5.4	Wygładzanie przebiegu orientacji (ORIPATHS A8=, B8=, C8=)	346
6.6	Kompresja orientacji (COMPON, COMPCURV, COMPCAD)	348
6.7	Wygładzanie przebiegu orientacji (ORISON, ORISOF).....	351

6.8	Transformacja kinematyczna	353
6.8.1	Obróbka frezarska na częściach toczonych (TRANSMIT)	353
6.8.2	Transformacja pobocznic walca (TRACYL)	355
6.8.3	Oś skośna (TRAANG).....	364
6.8.4	Programowanie osi skośnej (G5, G7)	367
6.9	Ruch kartezjański PTP	369
6.9.1	PTP przy TRANSMIT	373
6.10	Warunki brzegowe przy wyborze transformacji	377
6.11	Cofnięcie wyboru transformacji (TRAFOOF)	378
6.12	Transformacje powiązane (TRACON, TRAFOOF).....	378
7	Łańcuchy kinematyczne	381
7.1	Usunięcie komponentów (DELOBJ)	381
7.2	Określanie indeksu poprzez nazwy (NAMETOINT).....	384
8	Uniknięcie kolizji za pomocą łańcuchów kinematycznych	385
8.1	Sprawdzenie pod kątem pary kolizyjnej (COLLPAIR)	385
8.2	Żądanie nowego obliczenia modelu kolizji (PROTA).....	386
8.3	Ustawienie stanu obszaru ochrony (PROTS)	388
8.4	Określenie odległości dwóch obszarów ochrony (PROTD).....	389
9	Korekcje narzędzi	391
9.1	Pamięć korekcji	391
9.2	Korekcje addytywne.....	394
9.2.1	Wybranie korekcji addytywnych (DL).....	394
9.2.2	Ustalenie zużycia i wartości ustawczych (\$TC_SCPxy[t,d], \$TC_ECPxy[t,d])	395
9.2.3	Skasowanie korekcji addytywnych (DELDEL)	396
9.3	Traktowanie specjalne korekcji narzędzia	397
9.3.1	Lustrzane odbicie długości narzędzia.....	399
9.3.2	Reakcja na znak zużycia	400
9.3.3	Układ współrzędnych aktywnej obróbki (TOWSTD, TOWMCS, TOWWCS, TOWBCS, TOWTCS, TOWKCS)	401
9.3.4	Długość narzędzia i zmiana płaszczyzny	404
9.4	Korekcja narzędzia online (PUTFTOCF, FCTDEF, PUTFTOC, FTOCON, FTOCOF)	405
9.5	Uaktywnienie korekcji narzędzia 3D (CUT3DC..., CUT3DF...).....	410
9.5.1	Uaktywnienie korekcji narzędzia 3D (CUT3DC, CUT3DF, CUT3DFS, CUT3DFF, ISD)	410
9.5.2	Korekcja narzędzia 3D: frezowanie obwodowe, frezowanie czołowe	412
9.5.3	Korekcja narzędzia 3D: kształty narzędzi i dane narzędzi dla frezowania czołowego	414
9.5.4	Korekcja narzędzia 3D: Korekcja po torze, zakrzywienie toru, głębokość wgłębienia (CUT3DC, ISD)	415
9.5.5	Korekcja narzędzia 3D: naroża wewnętrzne/zewnętrzne i metoda punktu przecięcia (G450/G451)	418
9.5.6	Korekcja narzędzia 3D: frezowanie obwodowe 3D z płaszczyznami ograniczającymi	419
9.5.7	Korekcja narzędzia 3D: Uwzględnienie powierzchni ograniczającej (CUT3DCC, CUT3DCCD)	420
9.6	Orientacja narzędzia (ORIC, ORID, OSOF, OSC, OSS, OSSE, ORIS, OSD, OST)	424

9.7	Dowolne nadawanie numerów D, numer ostrza	430
9.7.1	Dowolne nadawanie numerów D, numer ostrza (adres CE).....	430
9.7.2	Dowolne nadawanie numerów D: sprawdzenie numerów D (CHKDNO)	430
9.7.3	Dowolne nadawanie numerów D: Zmiana nazw numerów D (GETDNO, SETDNO)	431
9.7.4	Dowolne nadawanie numerów D: Określenie numeru T do zadanego numeru D (GETACTTD)	432
9.7.5	Dowolne nadawanie numerów D: ustawienie nie obowiązywania numerów D (DZERO)	433
9.8	Kinematyka nośnika narzędzi	433
9.9	Korekcja długości narzędzia dla orientowalnych nośników narzędzi (TCARR, TCOABS, TCOFR, TCOFRX, TCOFRY, TCOFRZ)	439
9.10	Korekcja długości narzędzia online (TOFFON, TOFFOF).....	442
9.11	Modyfikacja danych skrawania w przypadku narzędzi obrotowych (CUTMOD)	445
10	Zachowanie się w ruchu po torze	451
10.1	Sterowanie styczne (TANG, TANGON, TANGOF, TLIFT, TANGDEL)	451
10.2	Przebieg posuwu (FNORM, FLIN, FCUB, FPO).....	458
10.3	Przyśpieszenie	463
10.3.1	Tryb przyśpieszenia (BRISK, BRISKA, SOFT, SOFTA, DRIVE, DRIVEA)	463
10.3.2	Sterowanie przyśpieszeniem w przypadku osi nadążnych (VELOLIMA, ACCLIMA, JERKLIMA).....	465
10.3.3	Uaktywnienie specyficznych dla technologii wartości dynamiki (DYNNORM, DYNPOS, DYNROUGH, DYNSEMIFIN, DYNFINISH)	467
10.4	Ruch ze sterowaniem wyprzedzającym (FFWON, FFWOF)	469
10.5	Programowalna dokładność konturu (CPRECON, CPRECOF)	470
10.6	Przebieg programu z pamięcią przebiegu wyprzedzającego (STOPFIFO, STARTFIFO, FIFOCTRL, STOPRE).....	472
10.7	Warunkowo przerywalne segmenty programu (DELAYFSTON, DELAYFSTOF)	475
10.8	Zablokowanie miejsca w programie dla SERUPRO (IPTRLOCK, IPTRUNLOCK)	480
10.9	Dosunięcie przywracające do konturu (REPOSA, REPOSL, REPOSQ, REPOSQA, REPOSH, REPOSHA, DISR, DISPR, RMIBL, RMBBL, RMEBL, RMNBL).....	483
10.10	Sterowanie prowadzeniem ruchu.....	492
10.10.1	Procentowa korekcja przyśpieszenia drugiego stopnia (JERKLIM)	492
10.10.2	Procentowa korekcja prędkości (VELOLIM)	493
10.10.3	Przykład programu dla JERKLIM i VELOLIM	495
10.11	Programowana tolerancja konturu/orientacji (CTOL, OTOL, ATOL)	496
10.12	Tolerancja przy ruchach G0 (STOLF).....	500
10.13	Zmiana bloku przy aktywnym sprzężeniu (CPBC).....	502
11	Sprzężenia osi	503
11.1	Nadążanie (TRAILON, TRAILOF).....	503
11.2	Tablice krzywych (CTAB).....	508
11.2.1	Definiowanie tablic krzywych (CTABDEF, CTABEND).....	509
11.2.2	Sprawdzenie istnienia pliku tablicy krzywych (CTABEXISTS).....	515
11.2.3	Kasowanie tablic krzywych (CTABDEL)	515

11.2.4	Zablokowanie tablic krzywych przed skasowaniem i zastąpieniem (CTABLOCK, CTABUNLOCK)	517
11.2.5	Tablice krzywych: określenie właściwości tablicy (CTABID, CTABISLOCK, CTABMEMTYP, CTABPERIOD)	518
11.2.6	Odczyt wartości tablicy krzywych (CTABTSV, CTABTEV, CTABTSP, CTABTEP, CTABSSV, CTABSEV, CTAB, CTABINV, CTABTMIN, CTABTMAX).....	520
11.2.7	Tablice krzywych: sprawdzić wykorzystanie zasobów (CTABNO, CTABNOMEM, CTABFNO, CTABSEGID, CTABSEG, CTABFSEG, CTABMSEG, CTABPOLID, CTABPOL, CTABFPOL, CTABMPOL)	525
11.3	Osiowe sprzężenie wartości wiodącej (LEADON, LEADOF)	526
11.4	Przekładnia elektroniczna (EG)	531
11.4.1	Zdefiniowanie przekładni elektronicznej (EGDEF)	532
11.4.2	Włączenie przekładni elektronicznej (EGON, EGONSYN, EGONSYNE)	533
11.4.3	Wyłączenie przekładni elektronicznej (EGOFS, EGOFC)	537
11.4.4	Skasowanie definicji przekładni elektronicznej (EGDEL)	538
11.4.5	Posuw na obrót (G95) / przekładnia elektroniczna (FPR)	538
11.5	Wrzeciono synchroniczne	539
11.5.1	Wrzeciono synchroniczne: Programowanie (COUPDEF, COUPDEL, COUPON, COUPONC, COUPOF, COUPOFS, COUPRES, WAITC)	539
11.6	Sprzężenie rodzajowe (CP...)	550
11.7	Sprzężenie Master/Slave (MASLDEF, MASLDEL, MASLON, MASLOF, MASLOFS)	557
12	Akcje synchroniczne	561
12.1	Definicja akcji synchronicznej	561
13	Ruch wahliwy.....	563
13.1	Ruch wahliwy asynchroniczny (OS, OSP1, OSP2, OST1, OST2, OSCTRL, OSNSC, OSE, OSB).....	563
13.2	Ruch wahliwy sterowany przez akcje synchroniczne (OSCILL).....	568
14	Tłoczenie i cięcie	577
14.1	Uaktywnienie, wyłączenie aktywności	577
14.1.1	Tłoczenie i cięcie wł. lub wył. (SPOF, SON, PON, SONS, PONS, PDELAYON, PDELAYOF, PUNCHACC)	577
14.2	Automatyczny podział drogi.....	582
14.2.1	Podział drogi w przypadku osi uczestniczących w tworzeniu konturu.....	585
14.2.2	Podział drogi w przypadku pojedynczych osi	587
15	Szlifowanie	589
15.1	Specyficzny dla szlifowania nadzór narzędzia w programie obróbki (TMON, TMOF)	589
16	Dalsze funkcje	591
16.1	Funkcje osi (AXNAME, AX, SPI, AXTOSPI, ISAXIS, AXSTRING, MODAXVAL)	591
16.2	Przełączalne osie geometryczne (GEOAX).....	593
16.3	Pojemnik osi (AXCTSWE, AXCTSWED, AXCTSWEC)	598
16.4	Czekanie na obowiązującą pozycję osi (WAITENC)	600
16.5	Programowalne przełączenie zestawu parametrów (SCPARA).....	601

16.6	Sprawdzenie występującego zakresu językowego NC (STRINGIS)	602
16.7	Interaktywne wywołanie okna z programu obróbki (MMC)	607
16.8	Czas przebiegu programu / licznik obrabianych przedmiotów	608
16.8.1	Czas przebiegu programu / licznik obrabianych przedmiotów (przeгляд)	608
16.8.2	Czas przebiegu programu	609
16.8.3	Licznik obrabianych przedmiotów	612
16.9	Process DataShare - Wyprowadzenie do zewnętrznego urządzenia/pliku (EXTOPEN, WRITE, EXTCLOSE)	614
16.10	Alarmy (SETAL)	622
16.11	Rozszerzone zatrzymanie i wycofanie (ESR)	624
16.11.1	ESR prowadzone przez NC	625
16.11.1.1	Wycofanie prowadzone przez NC (POLF, POLFA, POLFMASK, POLFMLIN)	625
16.11.1.2	Zatrzymanie prowadzone przez NC	629
16.11.2	ESR wystarczające dla napędu	630
16.11.2.1	Zaprojektowanie zatrzymania niezależnego dla napędu (ESRS)	630
16.11.2.2	Zaprojektowanie wycofania niezależnego dla napędu (ESRR)	631
17	Własne programy obróbki	633
17.1	Funkcje wspierające skrawanie	633
17.2	Sporządzenie tablicy konturu (CONTPRON)	634
17.3	Sporządzenie kodowanej tablicy konturu (CONTDCON)	640
17.4	Obliczenie punktu przecięcia między dwoma elementami konturu (INTERSEC)	643
17.5	Przejście pojedynczymi blokami elementów konturu w tablicy (EXECTAB)	645
17.6	Obliczenie danych okręgu (CALCDAT)	646
17.7	Wyłączenie przygotowania konturu (EXECUTE)	648
18	Zewnętrzne programowanie cykli	649
18.1	Cykle technologiczne	649
18.1.1	Wprowadzenie	649
18.1.2	Wiercenie, nawiercanie - CYCLE81	651
18.1.3	Wiercenie, pogłębianie czołowe - CYCLE82	652
18.1.4	Rozwiercanie - CYCLE85	653
18.1.5	Wiercenie głębokich otworów - CYCLE83	654
18.1.6	Wytaczanie - CYCLE86	656
18.1.7	Gwintowanie otworu bez oprawki kompensacyjnej - CYCLE84	657
18.1.8	Gwintowanie otworu z oprawką kompensacyjną - CYCLE840	660
18.1.9	Wiercenie otworu z frezowaniem gwintu - CYCLE78	662
18.1.10	Dowolne pozycje - CYCLE802	665
18.1.11	Szereg otworów - HOLES1	666
18.1.12	Siatka albo ramka - CYCLE801	667
18.1.13	Okrąg otworów - HOLES2	668
18.1.14	Frezowanie płaszczyzny - CYCLE61	669
18.1.15	Frezowanie kieszeni prostokątnej - POCKET3	671
18.1.16	Frezowanie kieszeni kołowej - POCKET4	674
18.1.17	Frezowanie czopa prostokątnego - CYCLE76	676
18.1.18	Frezowanie czopa kołowego - CYCLE77	678
18.1.19	Wielobok - CYCLE79	680

18.1.20	Rowek podłużny - SLOT1	682
18.1.21	Rowek kołowy - SLOT2	684
18.1.22	Frezowanie rowka otwartego - CYCLE899	687
18.1.23	Otwór podłużny - LONGHOLE	689
18.1.24	Frezowanie gwintu - CYCLE70	691
18.1.25	Cykl grawerowania - CYCLE60	693
18.1.26	Wywołanie konturu - CYCLE62	695
18.1.27	Frezowanie konturu - CYCLE72	696
18.1.28	Wiercenie wstępne kieszeni konturowej - CYCLE64	699
18.1.29	Frezowanie kieszeni konturowej - CYCLE63	700
18.1.30	Skrawanie warstwowe - CYCLE951	703
18.1.31	Wytoczenie - CYCLE930	705
18.1.32	Podcięcie kształtowe - CYCLE940	708
18.1.33	Toczenie gwintu - CYCLE99	710
18.1.34	Łańcuch gwintów - CYCLE98	713
18.1.35	Przecinanie - CYCLE92	717
18.1.36	Skrawanie konturu - CYCLE95	718
18.1.37	Toczenie wcinające konturu - CYCLE952	720
18.1.38	Skręt - CYCLE800	724
18.1.39	"Obróbka szybkościowa" - CYCLE832	727
19	Tablice	731
19.1	Instrukcje	731
19.2	Instrukcje: Dostępność w przypadku SINUMERIK 828D	797
19.3	Aktualny język w HMI	821
A	Aneks	823
A.1	Lista skrótów	823
A.2	Przegląd dokumentacji	832
	Glosariusz	833
	Indeks	857

Elastyczne programowanie NC

1.1 Zmienne

Dzięki zastosowaniu zmiennych, w szczególności w połączeniu z funkcjami obliczeniowymi i strukturami kontrolnymi, można ekstremalnie elastycznie kształtować programy obróbki i cykle. W tym celu system udostępnia trzy różne rodzaje zmiennych:

- Zmienne systemowe

Zmienne systemowe są zdefiniowanymi w systemie i udostępnionymi użytkownikowi zmiennymi o stałym znaczeniu. Są one też czytane i zapisywane przez oprogramowanie systemowe. Przykład: Dane maszynowe

Znaczenie zmiennej systemowej jest na stałe zadane przez system. Właściwości mogą jednak w małym zakresie zostać jeszcze dopasowane przez użytkownika w drodze redefinicji. Patrz "Redefinicja zmiennych systemowych, zmiennych użytkownika i poleceń językowych NC (REDEF) (Strona 30)"

- Zmienne użytkownika

Zmiennymi użytkownika są zmienne, których znaczenie nie jest znane systemowi i na które system też nie reaguje. Znaczenie jest ustalane wyłącznie przez użytkownika.

Zmienne użytkownika są podzielone na:

- Predefiniowane zmienne użytkownika

Predefiniowanymi zmiennymi użytkownika są zmienne już zdefiniowane w systemie, których liczbę użytkownik musi już tylko sparametryzować przez specyficzne dane maszynowe. Właściwości tych zmiennych użytkownik może w najwyższym stopniu dopasować. Patrz "Redefinicja zmiennych systemowych, zmiennych użytkownika i poleceń językowych NC (REDEF) (Strona 30)".

- Zmienne definiowane przez użytkownika

Zmienne definiowane przez użytkownika są zmiennymi, które są definiowane wyłącznie przez użytkownika i tworzone przez system dopiero w czasie przebiegu. Ich liczbę, typ danych, widoczność i wszystkie dalsze właściwości ustala wyłącznie użytkownik.

Patrz "Definicja zmiennych użytkownika (DEF) (Strona 24)"

1.1.1 Zmienna systemowa

Zmienne systemowe są zmiennymi predefiniowanymi w systemie, które w programach obróbki i cyklach zapewniają dostęp do aktualnej parametryzacji sterowania, jak też do stanów maszyny, sterowania i procesu.

Zmienne przebiegu wyprzedzającego

Jako zmienne przebiegu wyprzedzającego są określane zmienne systemowe, które są czytane i zapisywane w kontekście przebiegu wyprzedzającego, np. w chwili interpretacji bloku programu obróbki, w którym jest zaprogramowana zmienna systemowa. Zmienne przebiegu wyprzedzającego nie wyzwalają zatrzymania tego przebiegu.

Zmienne przebiegu głównego

Jako zmienne przebiegu głównego są określane zmienne systemowe, które są czytane i zapisywane w kontekście przebiegu głównego, np. w chwili bloku programu obróbki, w którym jest zaprogramowana zmienna systemowa. Zmiennymi przebiegu głównego są:

- Zmienne systemowe, które mogą być programowane w akcjach synchronicznych (odczyt/zapis)
- Zmienne systemowe, które mogą być programowane w programie obróbki i wyzwalają zatrzymanie przebiegu wyprzedzającego (odczyt/zapis)
- Zmienne systemowe, które mogą być programowane w programie obróbki, a których wartość jest określana w przebiegu wyprzedzającym ale zapisywana dopiero w przebiegu głównym (synchronicznie z przebiegiem głównym: tylko zapis)

Usystematyzowanie prefiksów

W celu specjalnego oznaczenia zmiennych systemowych nazwa jest w normalnym przypadku poprzedzona prefiksem, który składa się ze znaku \$, po którym następuje jedna lub dwie litery i podkreślenie:

\$ + 1. Litera	Znaczenie: Rodzaj danych
Zmienne systemowe, które są czytane / zapisywane w przebiegu wyprzedzającym	
\$M	Dane maszynowe ¹⁾
\$S	Dane ustawcze, obszary ochrony ¹⁾
\$T	Dane zarządzania narzędziami
\$P	Wartości programowane
\$C	Zmienne cykli ISO
\$O	Dane opcji
R	Parametry R (parametry obliczeniowe) ²⁾
Zmienne systemowe, które są czytane / zapisywane w przebiegu głównym	
\$\$M	Dane maszynowe ¹⁾
\$\$\$	Dane ustawcze ¹⁾
\$A	Aktualne dane przebiegu głównego
\$V	Dane serwo

\$ + 1. Litera	Znaczenie: Rodzaj danych
\$R	Parametry R (parametry obliczeniowe) ²⁾
<p>1) To czy dane maszynowe i ustawcze są traktowane, jako dane przebiegu wyprzedzającego, czy jako dane przebiegu głównego zależy od tego, czy są pisane z jednym czy z dwoma znakami \$. Sposób zapisu można wybrać odpowiednio do zastosowania.</p> <p>2) Przy zastosowaniu parametru R w programie obróbki / cyklu jako zmienna przebiegu wyprzedzającego prefiks nie jest pisany, np. R10. Przy zastosowaniu w akcji synchronicznej jako zmienna przebiegu głównego jest jako prefiks pisany znak \$, np. \$R10.</p>	

2. Litera	Znaczenie: widoczność
N	Zmienna globalna dla NCK (NCK)
C	Zmienna kanałowa (Channel)
A	Zmienna osiowa (Axis)

Warunki brzegowe

Wyjątki w usystematyzowaniu prefiksów

Następujące zmienne systemowe odbiegają od podanego usystematyzania prefiksów:

- \$TC_...: 2. litera C wskazuje tutaj nie na kanałowe lecz na specyficzne dla uchwytu narzędzia zmienne systemowe (TC = Tool Carrier)
- \$P_ ...: Kanałowe zmienne systemowe

Zastosowanie danych maszynowych i ustawczych w akcjach synchronicznych

Przy zastosowaniu danych maszynowych i ustawczych w akcjach synchronicznych można określić poprzez prefiks, czy dane te są czytane/zapisywane synchronicznie z przebiegiem wyprzedzającym czy głównym.

Gdy dana pozostaje podczas obróbki niezmienną, odczyt może następować synchronicznie z przebiegiem wyprzedzającym. Prefiks danej maszynowej lub ustawczej jest w tym celu pisany ze znakiem \$:

```
ID=1 WHENEVER $AA_IM[z] < $SA_OSCILL_REVERSE_POS2[Z]-6 DO $AA_OVR[X]=0
```

Gdy dana jest podczas obróbki zmieniana, odczyt / zapis musi następować synchronicznie z przebiegiem głównym. Prefiks danej maszynowej lub ustawczej jest w tym celu pisany z dwoma znakami \$:

```
ID=1 WHENEVER $AA_IM[z] < $$SA_OSCILL_REVERSE_POS2[Z]-6 DO $AA_OVR[X]=0
```

Wskazówka

Zapis danych maszynowych

Przy zapisie danej maszynowej lub ustawczej należy zwracać uwagę, by aktywny poziom dostępu przy wykonywaniu programu obróbki / cyklu pozwalał na dostęp w celu zapisu i by działanie danej było "IMMEDIATE".

1.1 Zmienne

Literatura

Wyszczególnienie właściwości wszystkich zmiennych systemowych znajduje się w:
Podręcznik Lista zmiennych systemowych

Patrz również

Zmienne (Strona 17)

1.1.2 Predefiniowane zmienne użytkownika: parametry obliczeniowe (R)

Funkcja

Parametry obliczeniowe albo parametry R są predefiniowanymi zmiennymi użytkownika o określeniu R, definiowanymi jako tablica typu danych REAL. Ze względów historycznych dla parametrów R jest oprócz pisowni z indeksem tablicy np. $R_{[10]}$ również dozwolona pisownia bez indeksu tablicy np. R_{10} .

Przy zastosowaniu w akcjach synchronicznych musi nastąpić poprzedzenie literą \$ np. $\$R_{10}$.

Składnia

Przy zastosowaniu jako zmienna przebiegu wyprzedzającego:

$R_{<n>}$

$R_{[<wyrażenie>]}$

Przy zastosowaniu jako zmienna przebiegu głównego:

$\$R_{<n>}$

$\$R_{[<wyrażenie>]}$

Znaczenie

R: Identyfikator przy zastosowaniu, jako zmienna przebiegu wyprzedzającego
np. w programie obróbki

\$R: Identyfikator przy zastosowaniu, jako zmienna przebiegu głównego, np. w
akcjach synchronicznych

Typ: REAL

Zakres wartości: Przy zapisie nie wykładniczym:
 $\pm (0.000\ 0001 \dots 9999\ 9999)$

Wskazówka:

Jest dozwolonych maksymalnie 8 miejsc dziesiętnych

Przy zapisie wykładniczym:

$\pm (1*10^{-300} \dots 1*10^{+300})$

Wskazówka:

- Notacja: <mantysa>EX<wykładnik> np. 8.2EX-3
- Jest dozwolonych maksymalnie 10 znaków łącznie ze znakiem liczby i kropką dziesiętną.

<n>: Numer parametru R
 Typ: INT
 Zakres wartości: 0 - MAX_INDEX
Wskazówka
 MAX_INDEX wynika ze sparametryzowanej liczby parametrów R:
 MAX_INDEX = (MD28050 \$MN_MM_NUM_R_PARAM) - 1

<wyrażenie>: Indeks tablicy
 Jako indeks tablicy można podawać dowolne wyrażenie, jak długo wynik wyrażenia można zamienić na typ danych INT (INT, REAL, BOOL, CHAR)

Przykład

Przyporządkowania do parametrów R i zastosowanie parametrów R w funkcjach matematycznych:

Kod programu	Komentarz
R0=3.5678	; Przyporządkowanie w przebiegu wyprzedzającym
R[1]=-37.3	; Przyporządkowanie w przebiegu wyprzedzającym
R3=-7	; Przyporządkowanie w przebiegu wyprzedzającym
\$R4=-0.1EX-5	; Przyporządkowanie w przebiegu głównym: $R4 = -0.1 * 10^{-5}$
\$R[6]=1.874EX8	; Przyporządkowanie w przebiegu głównym: $R6 = 1.874 * 10^8$
R7=SIN(25.3)	; Przyporządkowanie w przebiegu wyprzedzającym
R[R2]=R10	; Adresowanie pośrednie przez parametry R
R[(R1+R2)*R3]=5	; Adresowanie pośrednie przez wyrażenie matematyczne
X=(R1+R2)	; Wykonaj ruch w osi X do pozycji wynikającej z sumy R1 i R2
Z=SQRT(R1*R1+R2*R2)	; Wykonaj ruch w osi Z do pozycji pierwiastek kwadratowy z $(R1^2 + R2^2)$

Patrz również

Zmienne (Strona 17)

1.1.3 Predefiniowane zmienne użytkownika: zmienne link

Funkcja

Poprzez zmienne link mogą w ramach funkcji "NCU-Link" być wymieniane dane cykliczne między NCU, które są ze sobą połączone w sieci. Umożliwiają one przy tym specyficzny dla formatu danych dostęp do pamięci zmiennych link. Pamięć zmiennych link jest odnośnie zarówno wielkości jak też struktury danych ustalana przez użytkownika / producenta maszyny specyficznie dla urządzenia.

Zmienne link są globalnymi dla systemu zmiennymi użytkownika, które przy zaprojektowanej komunikacji link mogą być czytane i zapisywane w programach obróbki i cyklach przez wszystkie NCU zespołu link. W przeciwieństwie do globalnych zmiennych użytkownika (GUD) zmienne link mogą być stosowane również w akcjach synchronicznych.

W przypadku urządzeń bez aktywnej NCU-Link, zmienne Link mogą lokalnie w sterowaniu być stosowane oprócz globalnych zmiennych użytkownika (GUD), jako dodatkowe globalne zmienne użytkownika.

Składnia

```
$A_DLB [<indeks>]  
$A_DLW [<indeks>]  
$A_DLB [<indeks>]  
$A_DLW [<indeks>]
```

Znaczenie

\$A_DLB:	Zmienna link dla formatu danych BYTE (1 bajt) Typ danych: UINT Zakres wartości: 0 ... 255
\$A_DLW:	Zmienna link dla formatu danych WORD (2 bajty) Typ danych: INT Zakres wartości: -32768 ... 32767
\$A_DLD:	Zmienna link dla formatu danych DWORD (4 bajty) Typ danych: INT Zakres wartości: -2147483648 ... 2147483647
\$A_DLR:	Zmienna link dla formatu danych REAL (8 bajtów) Typ danych: REAL Zakres wartości: $\pm(2,2 \cdot 10^{-308} \dots 1,8 \cdot 10^{308})$

<indeks>: Indeks adresowy w bajtach, licząc od początku pamięci zmiennych link
 Typ danych: INT
 Zakres wartości: 0 - MAX_INDEX

Wskazówka

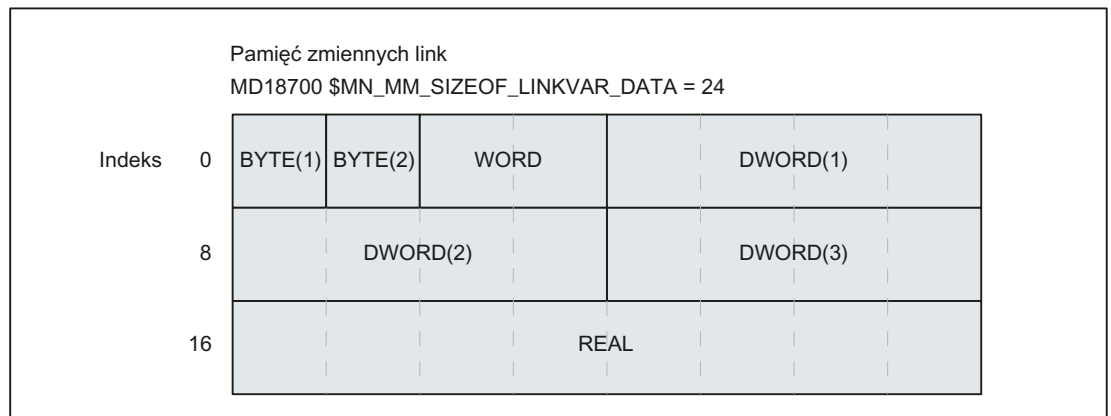
- MAX_INDEX wynika ze sparametryzowanej wielkości pamięci zmiennych link: $MAX_INDEX = (MD18700 \$MN_MM_SIZEOF_LINKVAR_DATA) - 1$
- Wolno programować indeksy tylko tak, by bajty adresowane w pamięci zmiennych link leżały na granicy formatu danych \Rightarrow
 $indeks = n * \text{bajty}$, gdzie $n = 0, 1, 2, \dots$
 - $\$A_DLB[i]: i = 0, 1, 2, \dots$
 - $\$A_DLW[i]: i = 0, 2, 4, \dots$
 - $\$A_DLD[i]: i = 0, 4, 8, \dots$
 - $\$A_DLR[i]: i = 0, 8, 16, \dots$

Przykład

W urządzeniu automatycznym są 2 NCU: NCU1 i NCU2. Do NCU1 jest przyłączona oś maszyny AX2, ruch w której jest wykonywany z NCU2, jako w osi link.

NCU1 zapisuje cyklicznie wartość zadaną prądu ($\$VA_CURR$) osi AX2 w pamięci zmiennych link. NCU2 czyta cyklicznie wartość rzeczywistą prądu przesyłaną przez komunikację link i przy przekroczeniu wartości granicznej wyświetla alarm 61000.

Struktura danych w pamięci zmiennych link jest przedstawiona na następującym rysunku. Wartość rzeczywista prądu jest przenoszona przez wartość REAL.

**NCU1**

NCU1 zapisuje w statycznej akcji synchronicznej, cyklicznie w taktie IPO, wartość rzeczywistą prądu osi AX2 przez zmienną link $\$A_DLR[16]$ w pamięci zmiennych link.

Kod programu

```
N111 IDS=1 WHENEVER TRUE DO $A_DLR[16]=$VA_CURR[AX2]
```

1.1 Zmienne

NCU2

NCU2 czyta w statycznej akcji synchronicznej, cyklicznie w takcie IPO, wartość rzeczywistą prądu osi AX2 przez zmienną link \$A_DLR[16] z pamięci zmiennych link. Jeżeli wartość rzeczywista prądu jest większa niż 23.0 A, jest wyświetlany alarm 61000.

Kod programu
N222 IDS=1 WHEN \$A_DLR[16] > 23.0 DO SETAL(61000)

Patrz również

Zmienne (Strona 17)

1.1.4 Definicja zmiennych użytkownika (DEF)

Funkcja

Przy pomocy polecenia `DEF` można definiować własne zmienne i wyposażać je w wartości. W odróżnieniu od zmiennych systemowych są one określane jako zmienne definiowane przez użytkownika lub zmienne użytkownika (User Data).

Odpowiednio do zakresu obowiązywania, tzn. zakresu w którym zmienna jest widoczna, są następujące kategorie zmiennych użytkownika:

- Lokalne zmienne użytkownika (LUD)

Lokalnymi zmiennymi użytkownika (LUD) są zmienne, które są zdefiniowane w programie obróbki, który w chwili wykonywania nie jest programem głównym. Są one tworzone przy wywołaniu programu obróbki i kasowane z końcem tego programu lub przy pomocy NC-Reset. Do LUD można mieć dostęp tylko w ramach tego programu obróbki, w którym są zdefiniowane.

- Dane użytkownika globalne dla programu (PUD)

Globalne dla programu zmienne użytkownika (PUD) są zmiennymi, które są definiowane w programie obróbki, będącym jako program główny. Są one tworzone ze startem programu obróbki i kasowane z jego końcem przy pomocy NC-Reset. Do PUD można mieć dostęp w programie głównym i we wszystkich podprogramach.

- Globalne zmienne użytkownika (GUD)

Globalne zmienne użytkownika (GUD) są zmiennymi globalnymi dla NC lub kanału, które są zdefiniowane w module danych (SGUD, MGUD, UGUD, GUD4 ... GUD9) i pozostają zachowane po power on. Do GUD można sięgać we wszystkich programach obróbki.

Zmienne użytkownika muszą być zdefiniowane przed ich zastosowaniem (odczyt / zapis). Należy przy tym przestrzegać następujących zasad:

- GUD muszą być zdefiniowane w pliku definicji, np. `_N_DEF_DIR/_M_SGUD_DEF`.
- PUD i LUD muszą być zdefiniowane w części definicyjnej programu obróbki.
- Definicja danych musi nastąpić we oddzielnym bloku.
- W definicji danych wolno zastosować tylko jeden typ danych.
- W definicji można zdefiniować wiele zmiennych o takim samym typie danych.

Składnia

LUD i PUD

```
DEF <typ> <jednostka_fizyczna> <wartości graniczne>
<nazwa>[<wartość_1>, <wartość_2>,
<wartość_3>]=<wartość_inicjalizacyjna>
```

GUD

```
DEF <zakres> <stop_przebiegu_wyprzedz.> <prawa dostępu> <typ>
<jednostka_fizyczna> <wartości graniczne> <nazwa>[<wartość_1>,
<wartość_2>, <wartość_3>]=<wartość_inicjalizacyjna>
```

Znaczenie

DEF:	Polecenie do definicji zmiennych użytkownika GUD, PUD, LUD
<zakres>:	Zakres obowiązywania, ma znaczenie tylko dla GUD:
	NCK: Zmienna użytkownika globalna dla NC
	CHAN: Zmienna użytkownika globalna dla kanału
<stop_przebiegu_wyprzedz.>:	Zatrzymanie przebiegu wyprzedzającego, ma znaczenie tylko dla GUD (opcjonalnie)
	SYNR: Zatrzymanie przebiegu wyprzedzającego przy odczycie
	SYNW: Zatrzymanie przebiegu wyprzedzającego przy zapisie
	SYNRW: Zatrzymanie przebiegu wyprzedzającego przy odczycie/zapisie
<prawa dostępu>:	Prawa dostępu dla odczytu / zapisu GUD przez program obróbki lub BTSS (opcjonalnie)
	APRP <poziom dostępu>: Odczyt: Program obróbki
	APWP <poziom dostępu>: Zapis: Program obróbki
	APRB <poziom dostępu>: Odczyt: BTSS
	APWB <poziom dostępu>: Zapis: BTSS
	<poziom ochrony>: Zakres wartości: 0 ... 7
	Patrz "Atrybut: prawa dostępu (APR, APW, APRP, APWP, APRB, APWB) (Strona 40)"
<Typ>:	Typ danych:
	INT: Wartość całkowitoliczbowa ze znakiem
	REAL: Liczba rzeczywista (LONG REAL według IEEE)
	BOOL: Wartość logiczna TRUE (1) / FALSE (0)
	CHAR: Znak ASCII
	STRING[<max długość>]: Łańcuch znaków o zdefiniowanej długości
	AXIS: Identyfikator osi/wrzeciona

1.1 Zmienne

	FRAME:	Dane geometryczne dla statycznej transformacji współrzędnych
		Patrz "Typy danych (Strona 53)"
<jednostka_fizyczna>:		Jednostka fizyczna (opcjonalnie)
	PHU <jednostka>:	jednostka fizyczna
		Patrz "Atrybut: jednostka fizyczna (PHU) (Strona 38)"
<wartości_graniczne>:		dolna / górna wartość graniczna (opcjonalnie)
	LLI <wartość graniczna>:	Dolna wartość graniczna (lower limit)
	ULI <wartość graniczna>:	Górna wartość graniczna (upper limit)
		Patrz "Atrybut: wartości graniczne (LLI, ULI) (Strona 36)"
<nazwa>:		Nazwa zmiennej
		Wskazówka
		<ul style="list-style-type: none"> • Maksymalnie 31 znaków • Obydwa pierwsze znaki muszą być literą i/lub podkreśleniem. • Znak "\$" jest zarezerwowany dla zmiennych systemowych i nie wolno go stosować.
[<wartość_1>, <wartość_2>, <wartość_3>]:		Podanie wielkości tablic dla zmiennych tablicowych 1- do max 3-wymiarowych (opcjonalnie)
		Odnosnie inicjalizacji zmiennych tablicowych patrz "Definicja i inicjalizacja zmiennych tablicowych (DEF, SET, REP) (Strona 46)"
<wartość_inicjalizacyjna>:		Wartość inicjalizacyjna (opcjonalnie)
		Patrz "Atrybut: wartość inicjalizacyjna (Strona 33)"
		Odnosnie inicjalizacji zmiennych tablicowych patrz "Definicja i inicjalizacja zmiennych tablicowych (DEF, SET, REP) (Strona 46)"

Przykłady

Przykład 1: Definicje zmiennych użytkownika w module danych dla producenta maszyny

Kod programu	Komentarz
%_N_MGUD_DEF	; Moduł GUD: Producent maszyny
\$PATH=/_N_DEF_DIR	
DEF CHAN REAL PHU 24 LLI 0 ULI 10 STROM_1, STROM_2	
;Opis	
;Definicja dwóch GUD: STROM_1, STROM_2	
;Zakres obowiązywania: W kanale	
;Typ danych: REAL	
;VL-Stop: nie zaprogramowano => wartość domyślna = brak VL-Stop	
;Jednostka fizyczna: 24 = [A]	

Kod programu	Komentarz
;Wartości graniczne: Low = 0.0, High = 10.0	
;Prawa dostępu: nie zaprogramowane => wartość domyślna = 7 = położenie 0 przełącznika z kluczykiem	
;Wartość inicjalizacyjna: nie zaprogramowana => wartość domyślna = 0.0	
DEF NCK REAL PHU 13 LLI 10 APWP 3 APRP 3 APWB 0 APRB 2 CZAS_1=12, CZAS_2=45	
;Opis	
;Definicja dwóch GUD: CZAS_1, CZAS_2	
;Zakres obowiązywania: w NCK	
;Typ danych: REAL	
;VL-Stop: nie zaprogramowano => wartość domyślna = brak VL-Stop	
;Jednostka fizyczna: 13 = [s]	
;Wartości graniczne: Low = 10.0, High = nie zaprogramowano => górna granica zakresu definicji	
;Prawa dostępu:	
;Program obróbki: zapis/odczyt = 3 = użytkownik końcowy	
;BTSS: zapis = 0 = Siemens, odczyt = 3 = użytkownik końcowy	
;Wartość inicjalizacyjna: CZAS_1 = 12.0, CZAS_2 = 45.0	
DEF NCK APWP 3 APRP 3 APWB 0 APRB 3 STRING[5] GUD5_NAME = "COUNTER"	
;Opis	
;Definicja GUD: GUD5_NAME	
;Zakres obowiązywania: w NCK	
;Typ danych: STRING, max 5 znaków	
;VL-Stop: nie zaprogramowano => wartość domyślna = brak VL-Stop	
;Jednostka fizyczna: nie zaprogramowano => wartość domyślna = 0 = bez jednostki fizycznej	
;Wartości graniczne: nie zaprogramowano => granice zakresu definicji: Low = 0, High = 255	
;Prawa dostępu:	
;Program obróbki: zapis/odczyt = 3 = użytkownik końcowy	
;BTSS: zapis = 0 = Siemens, odczyt = 3 = użytkownik końcowy	
;Wartość inicjalizacyjna: "COUNTER"	
M30	

Przykład 2: Globalne i lokalne zmienne użytkownika w programie (PUD / LUD)

Kod programu	Komentarz
PROC MAIN	;Program główny
DEF INT VAR1	;Definicja PUD
...	
SUB2	; wywołanie podprogramu
...	
M30	

Kod programu	Komentarz
PROC SUB2	;Podprogram SUB2
DEF INT VAR2	;DEFINICJA LUD
...	
IF (VAR1==1)	;Odczyt PUD
VAR1=VAR1+1	;Odczyt i zapis PUD
VAR2=1	;Zapis LUD
ENDIF	
SUB3	; wywołanie podprogramu
...	
M17	

Kod programu	Komentarz
PROC SUB3	;Podprogram SUB3
...	
IF (VAR1==1)	;Odczyt PUD
VAR1=VAR1+1	;Odczyt i zapis PUD
VAR2=1	;Błąd: LUD z SUB2 nie jest znana
ENDIF	
...	
M17	

Przykład 3: Definicja i zastosowanie zmiennych użytkownika o typie danych AXIS

Kod programu	Komentarz
DEF AXIS ODCIETA	;1. Oś geometryczna
DEF AXIS WRZECIONO	;Wrzeciono
...	
IF ISAXIS(1) == FALSE GOTOF DALEJ	
ODCIETA = \$P_AXN1	
DALEJ:	
...	
WRZECIONO=(S1)	;1. Wrzeciono
OVRA[WRZECIONO]=80	;Override wrzeciona = 80%
WRZECIONO=(S3)	;3. Wrzeciono

Warunki brzegowe

Globalne zmienne użytkownika (GUD)

W ramach definicji globalnych zmiennych użytkownika (GUD) należy uwzględnić następujące dane maszynowe:

Nr	Identyfikator: \$MN_	Znaczenie
11140	GUD_AREA_SAVE_TAB	Dodatkowe zabezpieczenie dla modułów GUD
18118 ¹⁾	MM_NUM_GUD_MODULES	Liczba plików GUD w aktywnym systemie plików
18120 ¹⁾	MM_NUM_GUD_NAMES_NCK	Liczba globalnych nazw GUD
18130 ¹⁾	MM_NUM_GUD_NAMES_CHAN	Liczba specyficznych dla kanału nazw GUD
18140 ¹⁾	MM_NUM_GUD_NAMES_AXIS	Liczba specyficznych dla osi nazw GUD
18150 ¹⁾	MM_GUD_VALUES_MEM	Miejsce w pamięci dla globalnych wartości GUD
18660 ¹⁾	MM_NUM_SYNACT_GUD_REAL	Liczba projektowanych GUD typ danych REAL
18661 ¹⁾	MM_NUM_SYNACT_GUD_INT	Liczba projektowanych GUD typ danych INT
18662 ¹⁾	MM_NUM_SYNACT_GUD_BOOL	Liczba projektowanych GUD typ danych BOOL
18663 ¹⁾	MM_NUM_SYNACT_GUD_AXIS	Liczba projektowanych GUD typ danych AXIS
18664 ¹⁾	MM_NUM_SYNACT_GUD_CHAR	Liczba projektowanych GUD typ danych CHAR
18665 ¹⁾	MM_NUM_SYNACT_GUD_STRING	Liczba projektowanych GUD typ danych STRING

¹⁾ MD w przypadku SINUMERIK 828D tylko do odczytu!

Dane użytkownika globalne dla programu (PUD)

Wskazówka

Widoczność globalnych w programie zmiennych użytkownika (PUD)

Zdefiniowane w programie głównym globalnie w programie zmienne użytkownika (PUD) są tylko wtedy widoczne również w podprogramach, gdy jest ustawiona następująca dana maszynowa:

MD11120 \$MN_LUD_EXTENDED_SCOPE = 1

Za pomocą MD11120 = 0 zdefiniowane w programie głównym globalne w programie zmienne użytkownika są widoczne tylko w programie głównym.

Wykraczające poza kanał zastosowanie globalnej w NCK zmiennej użytkownika o typie danych AXIS

Globalna w NCK zmienna użytkownika o typie danych `AXIS`, która przy definicji w module danych została zainicjalizowana z identyfikatorem osi, może być tylko wtedy stosowana w różnych kanałach NC, gdy oś w tych kanałach ma taki sam numer osi kanałowej.

Jeżeli tak nie jest, zmienna musi zostać załadowana na początku programu obróbki albo, jak w poniższym przykładzie, musi zostać zastosowana funkcja AXNAME(...).

Kod programu	Komentarz
DEF NCK STRING[5] OS="X"	;Definicja w module danych
...	
N100 AX[AXNAME(OS)]=111 G00	;Zastosowanie w programie obróbki

1.1.5 Redefinicja zmiennych systemowych, zmiennych użytkownika i poleceń językowych NC (REDEF)

Funkcja

Przy pomocy polecenia `REDEF` można zmienić atrybuty zmiennych systemowych, zmiennych użytkownika i polecenia językowe NC. Podstawowym warunkiem redefinicji jest, by pod względem czasu została wykonana po odpowiedniej definicji.

Przy redefinicji nie można równocześnie zmienić wielu atrybutów. Dla każdego zmienianego atrybutu musi zostać zaprogramowana oddzielna instrukcja `REDEF`.

Jeżeli zostanie zaprogramowanych wiele konkurujących zmian atrybutów, wówczas jest zawsze aktywna ostatnia zmiana.

Cofnięcie wartości atrybutów

Zmienione za pomocą `REDEF` atrybuty dla praw dostępu oraz chwili inicjalizacji mogą zostać cofnięte do ich wartości domyślnej poprzez ponowne zaprogramowanie polecenia `REDEF` poprzedzonego nazwą zmiennej bądź polecenia językowego NC.

- Prawa dostępu: Poziom dostępu 7
- Chwila inicjalizacji: brak instalacji lub też zachowania aktualnej wartości

Atrybuty redefiniowalne

Patrz "Przegląd atrybutów definiowalnych i redefiniowalnych (Strona 45)"

Lokalne zmienne użytkownika (PUD / LUD)

Dla lokalnych zmiennych użytkownika (PUD / LUD) nie wolno dokonywać żadnych redefinicji.

Składnia

```
REDEF <nazwa> <VL_Stop>
REDEF <nazwa> <jednostka_fizyczna>
REDEF <nazwa> <wartości_graniczne>
REDEF <nazwa> <prawa_dostępu>
REDEF <nazwa> <chwila_inicjalizacji>
REDEF <nazwa> <chwila_inicjalizacji> <wartość_inicjalizacyjna>
REDEF <nazwa>
```

Znaczenie

REDEF:	Rozkaz przedefiniowania danego atrybutu lub też cofnięcia atrybutów "Prawo dostępu" i/lub "Chwila inicjalizacji" zmiennych systemu, użytkownika oraz rozkazów językowych NC.
<nazwa>:	Nazwa już zdefiniowanej zmiennej albo polecenia języka NC
<Zatrzymanie przebiegu wyprzedzającego>:	Zatrzymanie przebiegu wyprzedzającego
	SYNR: Zatrzymanie przebiegu wyprzedzającego przy odczycie
	SYNW: Zatrzymanie przebiegu wyprzedzającego przy zapisie
	SYNRW: Zatrzymanie przebiegu wyprzedzającego przy odczycie/zapisie
<jednostka_fizyczna>:	Jednostka fizyczna
	PHU <jednostka>: jednostka fizyczna
	Patrz "Atrybut: jednostka fizyczna (PHU) (Strona 38)"
	Wskazówka
	Nie redefiniowalna przy:
	<ul style="list-style-type: none"> Zmienne systemowe Globalne dane użytkownika (GUD) dot. typów danych: <code>BOOL</code>, <code>AXIS</code>, <code>STRING</code>, <code>FRAME</code>
<wartości graniczne>:	dolna / górna wartość graniczna
	LLI <wartość graniczna>: Dolna wartość graniczna (lower limit)
	ULI <wartość graniczna>: Górna wartość graniczna (upper limit)
	Patrz "Atrybut: wartości graniczne (LLI, ULI) (Strona 36)"
	Wskazówka
	Nie redefiniowalna przy:
	<ul style="list-style-type: none"> Zmienne systemowe Globalne dane użytkownika (GUD) dot. typów danych: <code>BOOL</code>, <code>AXIS</code>, <code>STRING</code>, <code>FRAME</code>
<prawa dostępu>:	Prawa dostępu dla odczytu / zapisu przez program obróbki lub BTSS
	APRP <poziom dostępu>: Wykonanie: Element języka NC
	APRP <poziom dostępu>: Odczyt: Program obróbki
	APWP <poziom dostępu>: Zapis: Program obróbki
	APRB <poziom dostępu>: Odczyt: BTSS
	APWB <poziom dostępu>: Zapis: BTSS
	<poziom ochrony>: Zakres wartości: 0 ... 7
	Patrz "Atrybut: prawa dostępu (APR, APW, APRP, APWP, APRB, APWB) (Strona 40)"

1.1 Zmienne

<chwila_inicjalizacji>: Chwila, w której zmienna jest inicjalizowana

INIPO: Power On

INIRE: Koniec programu głównego, NC-Reset lub Power On

INICF: NewConfig lub koniec programu głównego, NC-Reset lub Power On

PRLOC: Koniec programu głównego, NC-Reset po lokalnej zmianie lub Power On

Patrz "Atrybut: wartość inicjalizacyjna (Strona 33)"

<wartość_inicjalizacyjna>: Wartość inicjalizacyjna

Przy redefinicji wartości inicjalizacyjnej musi zawsze zostać podana również chwila inicjalizacji (patrz <chwila_inicjalizacji>.

Patrz "Atrybut: wartość inicjalizacyjna (Strona 33)"

Odnosnie inicjalizacji zmiennych tablicowych patrz "Definicja i inicjalizacja zmiennych tablicowych (DEF, SET, REP) (Strona 46)"

Uwaga
Nieredefiniowalna dla zmiennych systemowych oprócz danych ustawczch.

Przykład

Redefinicje zmiennej systemowej \$TC_DPC1 w module danych dla producenta maszyny

```

Kod programu
%_N_MGUD_DEF ; Moduł GUD: Producent maszyny
N100 REDEF $TC_DPC1 APWB 2 APWP 3
N200 REDEF $TC_DPC2 PHU 21
N300 REDEF $TC_DPC3 LLI 0 ULI 200
N400 REDEF $TC_DPC4 INIPO (100, 101, 102, 103)
; N100: Prawo dostępu w celu zapisu: BTSS = stopień ochrony 2, program obróbki = stopień ochrony 3
; Uwaga
; Przy zastosowaniu plików ACCESS redefinicja praw dostępu
; musi zostać przeniesiona z _N_MGUD_DEF do _N_MACCESS_DEF
; N200: Jednostka fizyczna = [% ]
; N300: Wartości graniczne: dolna wartość graniczna = 0, górna wartość graniczna = 200
; N400: Zmienna tablicowa jest przy PowerOn inicjalizowana z czteroma wartościami
: Cofnięcie wartości atrybutów "Prawo dostępu" i/lub "Chwila inicjalizacji"
N800 REDEF $TC_DPC1
N900 REDEF $TC_DPC4
M30
    
```


Warunki brzegowe

Podzielność

Redefinicja odnosi się zawsze do całej zmiennej, jednoznacznie określonej przez nazwę. Nie jest możliwe np. w przypadku zmiennych tablicowych, przyporządkowanie różnych wartości atrybutów poszczególnym elementom tablicy.

1.1.6 Atrybut: wartość inicjalizacyjna

Definicja (_{DEF}) zmiennych użytkownika

Przy definicji można dla następujących zmiennych użytkownika zadać wartość inicjalizacyjną:

- Globalne zmienne użytkownika (GUD)
- Dane użytkownika globalne dla programu (PUD)
- Lokalne zmienne użytkownika (LUD)

Redefinicja (_{REDEF}) zmiennych systemowych i zmiennych użytkownika

Przy redefinicji można dla następujących zmiennych zadać wartość inicjalizacyjną:

- Dane systemowe
 - Dane ustawcze
- Dane użytkownika
 - Parametry R
 - Zmienna akcji synchronicznej (\$AC_MARKER, \$AC_PARAM, \$AC_TIMER)
 - GUD akcji synchronicznej (SYG_xy[], gdzie x=R, I, B, A, C, S, a y=S, M, U, 4, ..., 9)
 - Parametry EPS
 - Dane narzędzia-OEM
 - Dane magazynu-OEM
 - Globalne zmienne użytkownika (GUD)

Chwila reinicjalizacji

Przy redefinicji może zostać podana chwila, w której zmienna ma zostać zainicjalizowana, tzn. ponownie ustawiona na wartość inicjalizacyjną:

- `INIPO` (Power On)
Zmienna jest reinicjalizowana przy PowerOn.
- `INIRE` (Reset)
Zmienna jest reinicjalizowana przy NC-Reset, BAG-Reset, końcu programu obróbki (`M02 / M30`) lub przy PowerOn.

1.1 Zmienne

- INICF (NewConfig)

Zmienna jest reinicjalizowana przy zażądaniu NewConf przez HMI, poleceniu programu obróbki `NEWCONFIG` lub przy NC-Reset, BAG-Reset, końcu programu obróbki (M02 / M30) lub PowerOn.

- PRLOC (zmiana lokalna w programie)

Zmienna jest tylko wtedy reinicjalizowana przy NC-Reset, BAG-Reset lub końcu programu obróbki (M02 / M30), gdy została zmieniona w ramach aktualnego programu obróbki.

Atrybut `PRLOC` wolno stosować tylko w związku z programowalnymi danymi ustawczymi (patrz poniższa tablica).

Tabela 1- 1 Programowane dane ustawcze

Numer	Identyfikator	Polecenie G ¹⁾
42000	<code>\$SC_THREAD_START_ANGLE</code>	SF
42010	<code>\$SC_THREAD_RAMP_DISP</code>	DITS / DITE
42400	<code>\$SA_PUNCH_DWELLTIME</code>	PDELAYON
42800	<code>\$SA_SPIND_ASSIGN_TAB</code>	SETMS
43210	<code>\$SA_SPIND_MIN_VELO_G25</code>	G25
43220	<code>\$SA_SPIND_MAX_VELO_G26</code>	G26
43230	<code>\$SA_SPIND_MAX_VELO_LIMS</code>	LIMS
43300	<code>\$SA_ASSIGN_FEED_PER_REV_SOURCE</code>	FPRAON
43420	<code>\$SA_WORKAREA_LIMIT_PLUS</code>	G26
43430	<code>\$SA_WORKAREA_LIMIT_MINUS</code>	G25
43510	<code>\$SA_FIXED_STOP_TORQUE</code>	FXST
43520	<code>\$SA_FIXED_STOP_WINDOW</code>	FXSW
43700	<code>\$SA_OSCILL_REVERSE_POS1</code>	OSP1
43710	<code>\$SA_OSCILL_REVERSE_POS2</code>	OSP2
43720	<code>\$SA_OSCILL_DWELL_TIME1</code>	OST1
43730	<code>\$SA_OSCILL_DWELL_TIME2</code>	OST2
43740	<code>\$SA_OSCILL_VELO</code>	FA
43750	<code>\$SA_OSCILL_NUM_SPARK_CYCLES</code>	OSNSC
43760	<code>\$SA_OSCILL_END_POS</code>	OSE
43770	<code>\$SA_OSCILL_CTRL_MASK</code>	OSCTRL
43780	<code>\$SA_OSCILL_IS_ACTIVE</code>	OS
43790	<code>\$SA_OSCILL_START_POS</code>	OSB

1) Przy pomocy tego polecenia G następuje dostęp do danej ustawczej

Warunki brzegowe

Wartość inicjalizacyjna: globalne zmienne użytkownika (GUD)

- Dla globalnej zmiennej użytkownika (GUD) o zakresie obowiązywania `NCK`, można jako chwilę inicjalizacji zadać tylko `INIPO` (Power On).
- Dla globalnych zmiennych użytkownika (GUD) o zakresie obowiązywania `CHAN` można jako chwilę inicjalizacji oprócz `INIPO` (Power On) zadać również `INIRE` (Reset) lub `INICF` (NewConfig).
- W przypadku globalnych zmiennych użytkownika (GUD) o zakresie obowiązywania `CHAN` i chwili inicjalizacji `INIRE` (Reset) lub `INICF` (NewConfig), przy NC-Reset, BAG-Reset i NewConfig zmienne są ponownie inicjalizowane tylko w kanałach, w których podane zdarzenia zostały wyzwolone.

Wartość inicjalizacyjna: typ danych FRAME

Dla zmiennych o typie danych `FRAME` nie wolno podawać wartości inicjalizacyjnej. Zmienne o typie danych `FRAME` są implicite inicjalizowane z frame domyślnym.

Wartość inicjalizacyjna: typ danych CHAR

Dla zmiennych o typie danych `CHAR` można zamiast kodu ASCII (0...255) programować również odpowiednie znaki ASCII w cudzysłowie, np. "A"

Wartość inicjalizacyjna: typ danych STRING

W przypadku zmiennych o typie danych `STRING` łańcuch znaków musi zostać umieszczony w cudzysłowie np.: ...= "MASZYNA_1"

Wartość inicjalizacyjna: typ danych AXIS

Dla zmiennych o typie danych `AXIS` przy rozszerzonym sposobie pisania adresów identyfikator osi musi zostać umieszczony w nawiasach, np.: ...=(X3)

Wartość inicjalizacyjna: zmienna systemowa

Dla zmiennej systemowej nie można przez redefinicję zadać wartości inicjalizacyjnych specyficznych dla użytkownika. Wartości inicjalizacyjne zmiennych systemowych są na stałe zadane przez system. Przez redefinicję można jednak zmienić chwilę (`INIRE`, `INICF`) w której zmienna systemowa jest reinicjalizowana.

Wartość inicjalizacyjna implicite: typ danych AXIS

Dla zmiennych o typie danych `AXIS` jest implicite stosowana następująca wartość inicjalizacyjna:

- Dane systemowe: "pierwsza oś geometryczna"
- GUD akcji synchronicznej (określenie: SYG_A*), PUD, LUD:
Identyfikator osi z danej maszynowej: MD20082
\$MC_AXCONF_CHANAX_DEFAULT_NAME

Wartość inicjalizacyjna implicite: dane narzędzi i magazynu

Dla danych narzędzi i magazynu wartości inicjalizacyjne mogą być zadawane przez następującą daną maszynową: MD17520 \$MN_TOOL_DEFAULT_DATA_MASK

Wskazówka

Synchronizacja

Synchronizacja zdarzeń, które wyzwalają reinicjalizację zmiennej globalnej z odczytem tej zmiennej w innym miejscu, pozostaje wyłącznie w zakresie odpowiedzialności użytkownika / producenta maszyny.

Patrz również

Zmienne (Strona 17)

1.1.7 Atrybut: wartości graniczne (LLI, ULI)

Górną i dolną wartość graniczną zakresu definicji można zadać tylko dla następujących typów danych:

- INT
- REAL
- CHAR

Definicja (DEF) zmiennych użytkownika: wartości graniczne i implicite wartości inicjalizacyjne

Jeżeli przy definicji zmiennej użytkownika w jednym z wyżej wymienionych typów danych nie zostanie explicite zdefiniowana wartość inicjalizacyjna, zmienna zostanie implicite ustawiona na wartość inicjalizacyjną danego typu danych:

- INT: 0
- REAL: 0.0
- CHAR: 0

Jeżeli wartość graniczna implicite leży poza zakresem definicji ustalonym przez zaprogramowane wartości graniczne, zmienna jest inicjalizowana z wartością graniczną, która jest najbliższa wartości inicjalizacyjnej implicite:

- Wartość inicjalizacyjna implicite < dolna wartość graniczna (LLI) ⇒
wartość inicjalizacyjna = dolna wartość graniczna
- Wartość inicjalizacyjna implicite > górna wartość graniczna (ULI) ⇒
wartość inicjalizacyjna = górna wartość graniczna

Przykłady:

Kod programu	Komentarz
DEF REAL GUD1	; Dolna wartość graniczna = granica zakresu definicji ; górna wartość graniczna = granica zakresu definicji ; Nie zaprogramowano wartości inicjalizacyjnej ; => wartość inicjalizacyjna implicity = 0.0
DEF REAL LLI 5.0 GUD2	; Dolna wartość graniczna = 5.0 ; górna wartość graniczna = granica zakresu definicji ; => wartość inicjalizacyjna = 5.0
DEF REAL ULI -5 GUD3	; dolna wartość graniczna = granica zakresu definicji ; Górna wartość graniczna = -5.0 ; => wartość inicjalizacyjna = -5,0

Redefinicja (REDEF) zmiennych użytkownika: wartości graniczne i aktualne wartości rzeczywiste

Jeżeli przy redefinicji wartości granicznych zmiennej użytkownika zostaną one tak zmienione, że aktualna wartość rzeczywista znajdzie się poza nowym zakresem definicji, następuje alarm i wartości graniczne nie są przejmowane.

Wskazówka

Redefinicja (REDEF) zmiennych użytkownika

Przy redefinicji wartości granicznych zmiennej użytkownika należy zwracać uwagę na spójną zmianę następujących wartości:

- Wartości graniczne
- Wartość rzeczywista
- Wartość inicjalizacyjna przy redefinicji, a przy automatycznej reinicjalizacji na podstawie INIPO, INIRE lub INICF

Patrz również

Zmienne (Strona 17)

1.1.8 Atrybut: jednostka fizyczna (PHU)

Jednostka fizyczna może zostać zadana tylko dla zmiennych o następujących typach danych:

- INT
- REAL

Programowane jednostki fizyczne (PHU)

Podanie jednostki fizycznej następuje jako liczba stałoprzecinkowa: PHU <jednostka>

Można programować następujące jednostki fizyczne:

<jednostka >	Znaczenie	Jednostka fizyczna
0	Nie jednostka fizyczna	-
1	Pozycja liniowa lub kątowa ¹⁾²⁾	[mm], [cali], [stopień]
2	Pozycja liniowa ²⁾	[mm], [cali]
3	Pozycja kątowa	[stopień]
4	Prędkość liniowa lub kątowa ¹⁾²⁾	[mm/min], [cali/min], [obr/min]
5	Prędkość liniowa ²⁾	[mm/min]
6	Prędkość kątowa	[obr/min]
7	Przyśpieszenie liniowe lub kątowe ¹⁾²⁾	[m/s ²], [cali/s ²], [obr/s ²]
8	Przyśpieszenie liniowe ²⁾	[m/s ²], [cali/s ²]
9	Przyśpieszenie kątowe	[obr/s ²]
10	Przyśpieszenie drugiego stopnia liniowe lub kątowe ¹⁾²⁾	[m/s ³], [cali/s ³], [obr/s ³]
11	Przyśpieszenie drugiego stopnia liniowe ²⁾	[m/s ³], [cali/s ³]
12	Przyśpieszenie drugiego stopnia kątowe	[obr/s ³]
13	Czas	[s]
14	Wzmocnienie regulatora położenia	[16.667/s]
15	Posuw na obrót ²⁾	[mm/obr], [cali/obr]
16	Kompensacja temperatury ¹⁾²⁾	[mm], [cali]
18	Siła	[N]
19	Masa	[kg]
20	Moment bezwładności ³⁾	[kgm ²]
21	Procent	[%]
22	Częstotliwość	[Hz]
23	Napięcie	[V]
24	Prąd	[A]
25	Temperatura	[°C]
26	Kąt	[stopień]
27	KV	[1000/min]
28	Pozycja liniowa lub kątowa ³⁾	[mm], [cali], [stopień]
29	Prędkość skrawania ²⁾	[m/min], [stóp/min]

<jednostka >	Znaczenie	Jednostka fizyczna
30	Prędkość obwodowa ²⁾	[m/s], [stóp/s]
31	Oporność	[Ohm]
32	Indukcyjność	[mH]
33	Moment obrotowy ³⁾	[Nm]
34	Stała momentu obrotowego ³⁾	[Nm/A]
35	Wzmocnienie regulatora prądu	[V/A]
36	Wzmocnienie regulatora prędkości obrotowej ³⁾	[Nm/(rad*s)]
37	Prędkość obrotowa	[obr/min]
42	Moc	[kW]
43	Prąd, mały	[μ A]
46	Moment obrotowy, mały ³⁾	[μ Nm]
48	Promil	-
49	-	[Hz/s]
65	Przepływ	[l/min]
66	Ciśnienie	[bar]
67	Objętość ³⁾	[cm ³]
68	Wzmocnienie obiektowe ³⁾	[mm/(V*min)]
69	Wzmocnienie obiektowe regulator siły	[N/V]
155	Skok gwintu ³⁾	[mm/obr], [cali/obr]
156	Zmiana skoku gwintu ³⁾	[mm/obr / obr], [cali/obr / obr]
1) Jednostka fizyczna jest zależna od typu osi: liniowa lub obrotowa		
2) Przełączenie systemu miar G70/G71(calowy/metryczny) Po przełączeniu systemu podstawowego (MD10240 \$MN_SCALING_SYSTEM_IS_METRIC) przy pomocy G70/G71 przy dostępie w celu zapisu/odczytu do związanych z długością zmiennych systemowych i zmiennych użytkownika nie następuje przeliczenie wartości (wartość rzeczywista, wartość domyślna i wartości graniczne) G700/G710(calowy/metryczny) Po przełączeniu systemu podstawowego (MD10240 \$MN_SCALING_SYSTEM_IS_METRIC) przy pomocy G700/G710 przy dostępie w celu zapisu/odczytu do związanych z długością zmiennych systemowych i zmiennych użytkownika następuje przeliczenie wartości (wartość rzeczywista, wartość domyślna i wartości graniczne)		
3) Zmienna nie jest automatycznie przeliczana na aktualny system miar NC (calowy/metryczny). Przeliczenie leży wyłącznie w zakresie odpowiedzialności użytkownika / producenta maszyny.		

Wskazówka

Przekroczenie poziomu w wyniku przeliczenia formatu

Wewnętrzny format zapisu wszystkich zmiennych użytkownika (GUD / PUD / LUD) o związanych z długością jednostkach fizycznych jest formatem metrycznym. Nadmierne stosowanie tego rodzaju zmiennych w przebiegu głównym NCK, np. w akcjach synchronicznych, może przy przełączeniu systemu miar prowadzić do przekroczenia czasu obliczeniowego poziomu interpolatora, alarm 4240.

Wskazówka

Kompatybilność jednostek

Przy stosowaniu zmiennych (przyporządkowanie, porównanie itd.) nie następuje kontrola na kompatybilność jednostek. Ewentualnie wymagane przeliczenie leży wyłącznie w zakresie odpowiedzialności użytkownika / producenta maszyny.

Patrz również

Zmienne (Strona 17)

1.1.9 Atrybut: prawa dostępu (APR, APW, APRP, APWP, APRB, APWB)

Prawom dostępu odpowiadają następujące poziomy dostępu podawane przy programowaniu:

Prawo dostępu	Poziom dostępu
Hasło "System"	0
Hasło "Maschinenhersteller"	1
Hasło "Service"	2
Hasło "Endanwender"	3
Położenie 3 przełącznika z kluczykiem	4
Położenie 2 przełącznika z kluczykiem	5
Położenie 1 przełącznika z kluczykiem	6
Położenie 0 przełącznika z kluczykiem	7

Definicja (DEF) zmiennych użytkownika

Prawa dostępu (APR... / APW...) mogą być definiowane dla następujących zmiennych:

- Globalne dane użytkownika (GUD)

Redefinicja (REDEF) zmiennych systemowych i zmiennych użytkownika

Prawa dostępu (APR... / APW...) mogą być definiowane dla następujących zmiennych:

- Dane systemowe
 - Dane maszynowe
 - Dane ustawcze
 - FRAME
 - Dane procesowe

- Kompensacja błędu skoku śruby pociągowej (EEC)
- Kompensacja zwisu (CEC)
- Kompensacja błędu ćwiartki koła (QEC)
- Dane magazynu
- Dane narzędzi
- Obszary ochrony
- Orientowane nośniki narzędzi
- Łańcuchy kinematyczne
- Obszary ochrony 3D
- Ograniczenie obszaru pracy
- Dane narzędzi ISO
- Dane użytkownika
 - Parametry R
 - Zmienna akcji synchronicznej (\$AC_MARKER, \$AC_PARAM, \$AC_TIMER)
 - GUD akcji synchronicznej (SYG_xy[], gdzie x=R, I, B, A, C, S, a y=S, M, U, 4, ..., 9)
 - Parametry EPS
 - Dane narzędzia-OEM
 - Dane magazynu-OEM
 - Globalne zmienne użytkownika (GUD)

Wskazówka

Przy redefinicji prawo dostępu do zmiennej może zostać dowolnie nadane między najniższym poziomem dostępu 7 i własnym poziomem dostępu, np. 1 (producent maszyny).

Redefinicja (`REDEF`) poleceń językowych NC

Prawo dostępu wzgl. wykonania (`APX`) można redefiniować dla następujących poleceń językowych NC:

- Funkcje G / warunki drogowe

Literatura:

Podręcznik programowania Podstawy; Rozdział: Funkcje G / warunki drogowe

- Funkcje predefiniowane

Literatura:

Podręcznik programowania Podstawy; Rozdział: Funkcje predefiniowane

- Predefiniowane wywołania podprogramów

Literatura:

Podręcznik programowania Podstawy; Rozdział: Predefiniowane wywołania podprogramów

- Instrukcja DO w przypadku akcji synchronicznych
- Identyfikatory programowe cykli
Cykl musi być zapisany w katalogu cykli i zawierać instrukcję PROC.

Prawa dostępu odnośnie programów obróbki i cykli (APRP, APWP)

Różne prawa dostępu mają następujące działanie dla dostępu w programie obróbki lub cyklu:

- APRP 0 / APWP 0
 - Przy wykonywaniu programu obróbki musi być ustawione hasło systemowe
 - Cykl musi być zapisany w katalogu _N_CST_DIR (System)
 - Dla katalogu _N_CST_DIR prawo wykonania musi w MD11160 \$MN_ACCESS_EXEC_CST być ustawione na system
- APRP 1 / APWP 1 lub APRP 2 / APWP 2
 - Przy wykonywaniu programu obróbki musi być ustawione hasło "Maschinenhersteller" lub "Service"
 - Cykl musi być zapisany w katalogu _N_CMA_DIR (producent maszyny) lub _N_CST_DIR
 - Dla katalogów _N_CMA_DIR lub _N_CST_DIR w danych maszynowych MD11161 \$MN_ACCESS_EXEC_CMA lub MD11160 \$MN_ACCESS_EXEC_CST prawa wykonania muszą być ustawione co najmniej na "Maschinenhersteller"
- APRP 3 / APWP 3
 - Przy wykonywaniu programu obróbki musi być ustawione hasło "Endanwender"
 - Cykl musi być zapisany w katalogu _N_CUS_DIR (użytkownik), _N_CMA_DIR lub _N_CST_DIR
 - Dla katalogów _N_CUS_DIR, _N_CMA_DIR lub _N_CST_DIR w danych maszynowych MD11162 \$MN_ACCESS_EXEC_CUS, MD11161 \$MN_ACCESS_EXEC_CMA lub MD11160 \$MN_ACCESS_EXEC_CST prawa wykonania muszą być ustawione co najmniej na "Endanwender"
- APRP 4...7 / APWP 4...7
 - Przy wykonywaniu programu obróbki zamek z kluczykiem musi być ustawiony w położeniu 3 ... 0
 - Cykl musi być zapisany w katalogu _N_CUS_DIR, _N_CMA_DIR lub _N_CST_DIR
 - Dla katalogów _N_CUS_DIR, _N_CMA_DIR lub _N_CST_DIR w danych maszynowych MD11162 \$MN_ACCESS_EXEC_CUS, MD11161 \$MN_ACCESS_EXEC_CMA lub MD11160 \$MN_ACCESS_EXEC_CST prawa wykonania muszą być ustawione co najmniej na odpowiednie położenie przełącznika z kluczykiem

Prawa dostępu odnośnie BTSS ($APRB$, $APWB$)

Prawa dostępu ($APRB$, $APWB$) ograniczają dostęp do zmiennych systemowych i zmiennych użytkownika przez BTSS dla wszystkich komponentów systemowych (HMI, PLC, komputer zewnętrzny, usługi EPS, itd.) w tym samym stopniu.

Wskazówka

Prawa dostępu lokalne w HMI

Przy zmianie praw dostępu do danych systemowych jest konieczne zwrócenie uwagi, by nastąpiło to spójnie z prawami dostępu ustalonymi przez mechanizmy HMI.

Atrybuty dostępu APR / APW

Z powodu kompatybilności atrybuty APR i APW są implícite odwzorowywane na atrybuty $APRP$ / $APRB$ i $APWP$ / $APWB$:

- $APR\ x \Rightarrow APRP\ x\ APRB\ x$
- $APW\ y \Rightarrow APWP\ y\ APWB\ y$

Ustawienie praw dostępu przez pliki ACCESS

Przy stosowaniu plików ACCESS do nadawania praw dostępu, redefinicje praw dostępu dla danych systemowych, danych użytkownika i poleceń językowych NC wolno programować tylko w tych plikach ACCESS. Wyjątek stanowią globalne dane użytkownika (GUD). Dla nich, jeżeli okaże się to konieczne, redefinicja praw dostępu musi być nadal programowana w odpowiednich plikach definicji.

Dla wolnej od luk ochrony przed dostępem dane maszynowe praw wykonywania i ochrona przed dostępem do odpowiednich katalogów musi zostać spójnie dopasowana.

Jest następujący zasadniczy sposób postępowania:

- Sporządzenie potrzebnych plików definicji:
 - $_N_DEF_DIR/_N_SACCESS_DEF$
 - $_N_DEF_DIR/_N_MACCESS_DEF$
 - $_N_DEF_DIR/_N_UACCESS_DEF$
- Parametryzacja prawa zapisu dla plików definicji na wartość wymaganą dla redefinicji:
 - MD11170 \$MN_ACCESS_WRITE_SACCESS
 - MD11171 \$MN_ACCESS_WRITE_MACCESS
 - MD11172 \$MN_ACCESS_WRITE_UACCESS

1.1 Zmienne

- Dla dostępów do chronionych elementów z cykli muszą zostać dopasowane prawa wykonania i zapisu katalogów cykli `_N_CST_DIR`, `_N_CMA_DIR` i `_N_CST_DIR`:

Prawa wykonania

- MD11160 \$MN_ACCESS_EXEC_CST
- MD11161 \$MN_ACCESS_EXEC_CMA
- MD11162 \$MN_ACCESS_EXEC_CUS

Prawa zapisu

- MD11165 \$MN_ACCESS_WRITE_CST
- MD11166 \$MN_ACCESS_WRITE_CMA
- MD11167 MN_ACCESS_WRITE_CUS

Prawo wykonania musi zostać ustawione na co najmniej taki sam poziom ochrony, co najwyższy poziom ochrony zastosowanego elementu.

Prawo zapisu musi zostać ustawione co najmniej na taki sam poziom dostępu co prawo wykonania.

- Prawa zapisu lokalnych w HMI katalogów cykli muszą zostać ustawione na taki sam poziom dostępu co lokalne w NC katalogi cykli.

Literatura:

Podręcznik obsługi

Wywołania podprogramów w plikach ACCESS

Dla dalszej strukturyzacji ochrony przed dostępem mogą w plikach ACCESS być również wywoływane podprogramy (rozszerzenie SPF lub MPF). Podprogramy dziedziczą przy tym prawa wykonywania wywołującego pliku ACCESS.

Wskazówka

W plikach ACCESS mogą być redefiniowane tylko prawa dostępu. Wszystkie inne atrybuty muszą nadal być programowane lub redefiniowane w odpowiednich plikach definicji.

Patrz również

Zmienne (Strona 17)

1.1.10 Przegląd atrybutów definiowalnych i redefiniowalnych

Poniższe tablice pokazują w przypadku których rodzajów danych które atrybuty mogą być definiowane (DEF) i/lub redefiniowane (REDEF).

Dane systemowe

Rodzaj danych	Wartość inicjalizacyjna	Wartości graniczne	Jednostka fizyczna	Prawa dostępu
Dane maszynowe	---	---	---	REDEF
Dane ustawcze	REDEF	---	---	REDEF
Dane FRAME	---	---	---	REDEF
Dane procesowe	---	---	---	REDEF
Kompensacja błędu skoku śruby pociągowej (EEC)	---	---	---	REDEF
Kompensacja zwisu (CEC)	---	---	---	REDEF
Kompensacja błędu ćwiartki koła (QEC)	---	---	---	REDEF
Dane magazynu	---	---	---	REDEF
Dane narzędzi	---	---	---	REDEF
Obszary ochrony	---	---	---	REDEF
Orientowane nośniki narzędzi	---	---	---	REDEF
Łańcuchy kinematyczne	---	---	---	REDEF
Obszary ochrony 3D	---	---	---	REDEF
Ograniczenie obszaru pracy	---	---	---	REDEF
Dane narzędzi ISO	---	---	---	REDEF

Dane użytkownika

Rodzaj danych	Wartość inicjalizacyjna	Wartości graniczne	Jednostka fizyczna	Prawa dostępu
Parametry R	REDEF	REDEF	REDEF	REDEF
Zmienna akcji synchronicznej (\$AC_...)	REDEF	REDEF	REDEF	REDEF
GUD akcji synchronicznej (SYG_...)	REDEF	REDEF	REDEF	REDEF
Parametry EPS	REDEF	REDEF	REDEF	REDEF
Dane narzędzia OEM	REDEF	REDEF	REDEF	REDEF
Dane magazynu OEM	REDEF	REDEF	REDEF	REDEF
Globalne zmienne użytkownika (GUD)	DEF / REDEF	DEF	DEF	DEF / REDEF
Lokalne zmienne użytkownika (PUD / LUD)	DEF	DEF	DEF	---

Patrz również

Zmienne (Strona 17)

1.1.11 Definicja i inicjalizacja zmiennych tablicowych (DEF, SET, REP)

Funkcja

Zmienna użytkownika może zostać zdefiniowana jako 1- do maksymalnie 3-wymiarowa tablica (array):

- 1-wymiarowa: DEF <typ danych> <nazwa zmiennej> [<n>]
- 2-wymiarowa: DEF <typ danych> <nazwa zmiennej> [<n>, <m>]
- 3-wymiarowa: DEF <typ danych> <nazwa zmiennej> [<n>, <m>, <o>]

Wskazówka

Zmienne użytkownika o typie danych STRING mogą być definiowane maksymalnie jako tablica 2-wymiarowa.

Typy danych

Zmienne użytkownika mogą być definiowane jako tablice dla następujących typów danych: BOOL, CHAR, INT, REAL, STRING, AXIS, FRAME

Przyporządkowanie wartości do elementów tablicy

Przyporządkowania wartości do elementów tablicy mogą być dokonywane w następującym czasie:

- Przy definicji tablicy (wartości inicjalizacyjne)
- Podczas przebiegu programu

Przyporządkowanie wartości może przy tym nastąpić przez:

- podanie explicite elementu tablicy
- podanie explicite elementu tablicy jako elementu startowego i podanie listy wartości (SET)
- podanie explicite elementu tablicy jako elementu startowego i podanie wartości i częstości jej powtarzania (REP)

Wskazówka

Zmiennym użytkownika o typie danych FRAME nie można przyporządkować wartości inicjalizacyjnych.

Składnia (DEF)

```
DEF <typ danych> <nazwa zmiennej> [<n>, <m>, <o>]
DEF STRING [<długość łańcucha znaków>] <nazwa zmiennej> [<n>, <m>]
```

Składnia (DEF...=SET...)

Zastosowanie listy wartości:

- Przy definicji:

```
DEF <typ danych> <nazwa zmiennej>[<n>,<m>,<o>] = SET(<wartość1>,<wartość2>,...)
```

Równoznaczne z:

```
DEF <typ danych> <nazwa zmiennej>[<n>,<m>,<o>] = (<wartość1>,<wartość2>,...)
```

Wskazówka

Przy inicjalizacji przez listę wartości podanie SET jest opcjonalne.

- Przy przyporządkowaniu wartości:

```
<nazwa zmiennej>[<n>,<m>,<o>] = SET(<wartość1>,<wartość2>,...)
```

Składnia (DEF...=REP...)

Zastosowanie wartości z powtórzeniem

- Przy definicji:

```
DEF <typ danych> <nazwa zmiennej>[<n>,<m>,<o>] = REP(<wartość>)
```

```
DEF <typ danych> <nazwa zmiennej>[<n>,<m>,<o>] =  
REP(<wartość>,<liczba_elementów_tablicy>)
```

- Przy przyporządkowaniu wartości:

```
<nazwa zmiennej>[<n>,<m>,<o>]=REP(<wartość>)
```

```
<nazwa zmiennej>[<n>,<m>,<o>]=REP(<wartość>,<liczba_elementów_tablicy>)
```

Znaczenie

DEF:

<typ danych>:

<długość łańcucha znaków>:

<nazwa zmiennej>:

Polecenie dot. definicji zmiennych

Typ danych zmiennej

Zakres wartości:

- W przypadku zmiennych systemowych:
BOOL, CHAR, INT, REAL, STRING, AXIS
- W przypadku zmiennych GUD lub LUD:
BOOL, CHAR, INT, REAL, STRING, AXIS, FRAME

Maksymalna liczba znaków w przypadku typu danych
STRING

Nazwa zmiennej

<p>[<n>, <m>, <o>]:</p>	<p>Wielkości lub indeksy tablicy</p>
<p><n>:</p>	<p>Wielkość lub indeks tablicy dla 1. wymiaru</p> <p>Typ: INT (w przypadku zmiennej systemowej również AXIS)</p> <p>Zakres wartości: Maks. wielkość tablicy: 65535 Indeks tablicy: $0 \leq n \leq 65534$</p>
<p><m>:</p>	<p>Wielkość lub indeks tablicy dla 2. wymiaru</p> <p>Typ: INT (w przypadku zmiennej systemowej również AXIS)</p> <p>Zakres wartości: Maks. wielkość tablicy: 65535 Indeks tablicy: $0 \leq m \leq 65534$</p>
<p><o>:</p>	<p>Wielkość lub indeks tablicy dla 3. wymiaru</p> <p>Typ: INT (w przypadku zmiennej systemowej również AXIS)</p> <p>Zakres wartości: Maks. wielkość tablicy: 65535 Indeks tablicy: $0 \leq o \leq 65534$</p>
<p>SET:</p>	<p>Przyporządkowanie wartości przez podaną listę wartości</p>
<p>(<wartość1>, <wartość2>, ...):</p>	<p>Lista wartości</p>
<p>REP:</p>	<p>Przyporządkowanie wartości przez podaną <wartość></p>
<p><wartość>:</p>	<p>Wartość, którą mają zostać zapisane elementy tablicy przy inicjalizacji z REP.</p>
<p><Liczba_elementów_tablicy>:</p>	<p>Liczba elementów tablicy, które mają zostać zapisane podaną <wartościami>. Dla pozostałych elementów tablicy obowiązuje zależnie od czasu:</p> <ul style="list-style-type: none"> • Inicjalizacja przy definicji tablicy: <ul style="list-style-type: none"> → Pozostałe elementy tablicy są zapisywane wartością zero • Przyporządkowanie podczas przebiegu programu: <ul style="list-style-type: none"> → Aktualne wartości elementów tablicy pozostają bez zmian. <p>Jeżeli parametr nie jest zaprogramowany, wszystkie elementy tablicy są zapisywane wartością <wartość>.</p> <p>Jeżeli parametr jest równy zero, obowiązuje zależnie od czasu:</p> <ul style="list-style-type: none"> • Inicjalizacja przy definicji tablicy: <ul style="list-style-type: none"> → Wszystkie elementy są wyposażane w wartość domyślną zero • Przyporządkowanie podczas przebiegu programu: <ul style="list-style-type: none"> → Aktualne wartości elementów tablicy pozostają bez zmian.

Indeks tablicy

Implicite kolejność elementów tablicy np. przy przyporządkowaniu wartości przez SET lub REP następuje przez iterację indeksów tablicy od prawej do lewej.

Przykład: Inicjalizacja tablicy 3-wymiarowej o 24 elementach:

```
DEF INT FELD[2,3,4] = REP(1,24)
  FELD[0,0,0] = 1           1. Element tablicy
  FELD[0,0,1] = 1           2. Element tablicy
  FELD[0,0,2] = 1           3. Element tablicy
  FELD[0,0,3] = 1           4. Element tablicy
  ...
  FELD[0,1,0] = 1           5. Element tablicy
  FELD[0,1,1] = 1           6. Element tablicy
  ...
  FELD[0,2,3] = 1           12. Element tablicy
  FELD[1,0,0] = 1           13. Element tablicy
  FELD[1,0,1] = 1           14. Element tablicy
  ...
  FELD[1,2,3] = 1           24. Element tablicy
```

Odpowiednio:

```
FOR n=0 TO 1
  FOR m=0 TO 2
    FOR o=0 TO 3
      FELD[n,m,o] = 1
    ENDFOR
  ENDFOR
ENDFOR
```

Przykład: Inicjalizacja kompletnych tablic zmiennych

Aktualna zajętość patrz rysunek.

Kod programu
N10 DEF REAL FELD1[10,3]=SET(0,0,0,10,11,12,20,20,20,30,30,30,40,40,40,)
N20 FELD1[0,0]=REP(100)
N30 FELD1[5,0]=REP(-100)
N40 FELD1[0,0]=SET(0,1,2,-10,-11,-12,-20,-20,-20,-30, , , , -40,-40,-50,-60,-70)
N50 FELD1[8,1]=SET(8.1,8.2,9.0,9.1,9.2)

Indeks tablicy

	[1,2]	N10: Inicjalizacja przy definicji			N20/N30: Inicjalizacja z identyczną wartością			N40/N50: Inicjalizacja z różnymi wartościami		
		0	1	2	0	1	2	0	1	2
0	0	0	0	0	100	100	100	0	1	2
1	10	11	12	100	100	100	-10	-11	-12	
2	20	20	20	100	100	100	-20	-20	-20	
3	30	30	30	100	100	100	-30	0	0	
4	40	40	40	100	100	100	0	-40	-40	
5	0	0	0	-100	-100	-100	-50	-60	-70	
6	0	0	0	-100	-100	-100	-100	-100	-100	
7	0	0	0	-100	-100	-100	-100	-100	-100	
8	0	0	0	-100	-100	-100	-100	8.1	8.2	
9	0	0	0	-100	-100	-100	9.0	9.1	9.2	
	Elementy tablicy [5,0] do [9,2] zostały zainicjalizowane z wartością domyślną (0.0).						Elementy tablicy [3,1] do [4,0] zostały zainicjalizowane z wartością domyślną (0.0). Elementy tablicy [6,0] do [8,0] nie zostały zmienione.			

Patrz również

Definicja i inicjalizacja zmiennych tablicowych (DEF, SET, REP): Dalsze informacje (Strona 50)

Zmienne (Strona 17)

1.1.12 Definicja i inicjalizacja zmiennych tablicowych (DEF, SET, REP): Dalsze informacje

Dalsze informacje (SET)

Inicjalizacja przy definicji

- Jest, rozpoczynając od 1. elementu tablicy, inicjalizowanych tyle elementów z wartościami z listy wartości, ile elementów jest zaprogramowanych na liście wartości.
- Elementy tablicy bez explicite podanych wartości na liście wartości (luki na liście) otrzymują wartość 0.
- W przypadku zmiennych o typie danych AXIS luki na liście wartości są niedopuszczalne.
- Jeżeli lista wartości zawiera więcej wartości, niż zdefiniowano elementów tablicy, zostanie wyświetlony alarm.

Przyporządkowanie wartości w przebiegu programu

Przy przyporządkowywaniu wartości w przebiegu programu obowiązują zasady opisane wyżej przy definicji. Dodatkowo są następujące możliwości:

- Jako elementy na liście wartości są dozwolone również wyrażenia.
- Przyporządkowanie wartości rozpoczyna się od zaprogramowanego indeksu tablicy. Przez to można w sposób celowy wyposażyć w wartości tablice częściowe.

Przykład:

Kod programu	Komentarz
DEF INT FELD[5,5]	; Definicja tablicy
FELD[0,0]=SET(1,2,3,4,5)	; Przyporządkowanie wartości do pierwszych 5 elementów tablicy [0,0] - [0,4]
FELD[0,0]=SET(1,2, , ,5)	; Przyporządkowanie wartości z luką do pierwszych 5 elementów tablicy [0,0] - [0,4], elementy tablicy [0,2] i [0,3] = 0
FELD[2,3]=SET(VARIABLE,4*5.6)	; Przyporządkowanie wartości ze zmienną i wyrażeniem od indeksu tablicy [2,3]: [2,3] = VARIABLE [2,4] = 4 * 5.6 = 22.4

Dalsze informacje (REP)

Inicjalizacja przy definicji

- Wszystkie lub opcjonalnie podana liczba elementów tablicy są inicjalizowane z podaną wartością (stała).
- Zmienne o typie danych FRAME nie mogą być inicjalizowane.

Przykład:

Kod programu	Komentarz
DEF REAL varName[10]=REP(3.5,4)	; Inicjalizacja definicji tablicy i elementów tablicy [0] do [3] o wartości 3,5

Przyporządkowanie wartości w przebiegu programu

Przy przyporządkowywaniu wartości w przebiegu programu obowiązują zasady opisane wyżej przy definicji. Dodatkowo są następujące możliwości:

- Jako elementy na liście wartości są dozwolone również wyrażenia.
- Przyporządkowanie wartości rozpoczyna się od zaprogramowanego indeksu tablicy. Przez to można w sposób celowy wyposażyć w wartości tablice częściowe.

Przykłady:

Kod programu	Komentarz
DEF REAL varName[10]	; Definicja tablicy
varName[5]=REP(4.5,3)	; Elementy tablicy [5] do [7] = 4,5
R10=REP(2.4,3)	; Parametry R10 do R12 = 2,4
DEF FRAME FRM[10]	; Definicja tablicy
FRM[5]=REP(CTTRANS(X,5))	; Elementy tablicy [5] do [9] = CTTRANS(X,5)

Dalsze informacje (ogólne)

Przyporządkowania wartości do osiowych danych maszynowych

Osiowe dane maszynowe mają zasadniczo indeks tablicy o typie danych `AXIS`. Przy przyporządkowaniach wartości do osiowej danej maszynowej przy pomocy `SET` lub `REP` ten indeks tablicy jest ignorowany lub nie realizowany.

Przykład: Przyporządkowanie wartości do danej maszynowej MD36200
`$MA_AX_VELO_LIMIT`

```
$MA_AX_VELO_LIMIT[1,AX1]=SET(1.1, 2.2, 3.3)
```

Odpowiada:

```
$MA_AX_VELO_LIMIT[1,AX1]=1.1
```

```
$MA_AX_VELO_LIMIT[2,AX1]=2.2
```

```
$MA_AX_VELO_LIMIT[3,AX1]=3.3
```

Wskazówka

Przyporządkowania wartości do osiowych danych maszynowych

Przy przyporządkowaniach wartości do osiowych danych maszynowych przy pomocy `SET` lub `REP` indeks tablicy o typie danych `AXIS` jest ignorowany lub nie realizowany.

Zapotrzebowanie na pamięć

Typ danych	Zapotrzebowanie na pamięć na element
BOOL	1 bajt
CHAR	1 bajt
INT	4 bajty
REAL	8 bajtów
STRING	(długość łańcucha znaków + 1) bajtów
FRAME	~ 400 bajtów, zależnie od liczby osi
AXIS	4 bajty

Patrz również

Definicja i inicjalizacja zmiennych tablicowych (DEF, SET, REP) (Strona 46)

1.1.13 Typy danych

Następujące typy danych są do dyspozycji w NC:

Typ danych	Znaczenie	Zakres wartości
INT	Wartość całkowitoliczbowa ze znakiem	-2147483648 ... +2147483647
REAL	Liczba rzeczywista (LONG REAL według IEEE)	$\pm(\sim 2,2 \cdot 10^{-308} \dots \sim 1,8 \cdot 10^{+308})$
BOOL	Wartość logiczna TRUE (1) i FALSE (0)	1, 0
CHAR	Znak ASCII	Kod ASCII 0 ... 255
STRING	Łańcuch znaków o zdefiniowanej długości	Maksymalnie 200 znaków (bez znaków specjalnych)
AXIS	Identyfikator osi/wrzeciona	Identyfikator kanału
FRAME	Dane geometryczne dla statycznej transformacji współrzędnych (przesunięcie, obrót, skalowanie, lustrzane odbicie)	---

Implicite zmiany typów danych

Następujące zmiany typów danych są możliwe i są implicite dokonywane przy przyporządkowaniach i przekazaniach parametrów:

Z +/- na →	REAL	INT	BOOL
REAL	x	o	&
INT	x	x	&
BOOL	x	x	x

x: Możliwe bez ograniczeń
o: Możliwa utrata danych w wyniku przekroczenia zakresu wartości ⇒ alarm;
Zaokrąglenie: wartość po przecinku $\geq 0,5$ ⇒ zaokrąglenie do góry, wartość po przecinku $< 0,5$ ⇒ zaokrąglenie do dołu
&: Wartość $\neq 0$ ⇒ TRUE, wartość $= 0$ ⇒ FALSE

Patrz również

Zmienne (Strona 17)

1.1.14 Konwersje explicite typu danych (AXTOINT, INTTOAX)

Funkcja

Przy pomocy predefiniowanych funkcji AXTOINT i INTTOAX można explicite konwertować typ danych zmiennej osiowej.

Konwersja typu AXIS → INT

Składnia:

<wynik>=AXTOINT (<wartość>)

Znaczenie:

<wynik>:	Forma INT zmiennej osiowej (≙ indeks osi <n>)	
	W przypadku błędu:	
	= 7	NO_AXIS, tzn. <wartość> zawiera wartość "nie oś"
	= -1	<wartość> nie jest nazwą osi typu AXIS
AXTOINT:	AXTOINT zmienia typ danych zmiennej osiowej z AXIS na INT	
<wartość>:	nazwa osi geometrycznej (MD20060 \$MC_AXCONF_GEOAX_NAME_TAB[<n>]) lub nazwa osi kanałowej (MD20080 \$MC_AXCONF_CHANAX_NAME_TAB[<n>]) lub nazwa osi maszynowej (MD10000 \$MN_AXCONF_MACHAX_NAME_TAB[<n>])	
	Typ danych:	AXIS

Konwersja typu INT → AXIS

Składnia:

<wynik>=INTTOAX (<wartość>)

<wynik>:	Forma AXIS zmiennej osiowej (≙ nazwa osi)	
	W przypadku błędu:	
	= NO_AXIS	<wartość> zawiera wartość "nie oś"?
	= -1	<wartość> jest wartością INT, do której nie ma nazwy osi typu AXIS?
INTTOAX:	INTTOAX zmienia typ danych zmiennej osiowej z INT na AXIS	
<wartość>:	Wartość INT zmiennej osiowej	
	Zakres wartości:	0 - 32

Przykład

Patrz przykład do GETVARDFT pod "Odczyt wartości absolutnych/typu danych (GETVARPHU, GETVARAP, GETVARLIM, GETVARDFT, GETVARTYP) (Strona 57)".

1.1.15 Sprawdzenie istnienia zmiennej (ISVAR)

Funkcja

Przy pomocy predefiniowanej funkcji ISVAR można sprawdzić, czy zmienna systemowa/użytkownika (np. dana maszynowa, dana ustawcza, zmienna systemowa, zmienna ogólna jak GUD) jest znana w NCK.

Składnia

`<wynik>=ISVAR(<wartość>)`

Przekazywany parametr `<zmienna>` można zbudować następująco:

- bezwymiarowa zmienna \$: `$<identyfikator>`
- jednowymiarowa zmienna \$ bez indeksu tablicy: `$<identyfikator>[]`
- jednowymiarowa zmienna \$ z indeksem tablicy: `$<identyfikator><n>`
- dwuwymiarowa zmienna \$ bez indeksu tablicy: `$<identyfikator>[,]`
- dwuwymiarowa zmienna \$ z indeksem tablicy: `$<identyfikator><n>,<m>`

Znaczenie

<code><wynik></code> :	Wartość zwrotna	
	Typ danych:	BOOL
	Zakres wartości:	1 Zmienna istnieje 0 Zmienna nieznana
ISVAR:	Sprawdza, czy zmienna systemowa/użytkownika jest znana w NCK	
<code><identyfikator></code> :	Nazwa zmiennej systemowej/użytkownika	
	Typ danych:	STRING
<code><n></code> :	Indeks tablicy dla pierwszego wymiaru	
	Typ danych:	INT
<code><m></code> :	Indeks tablicy dla drugiego wymiaru	
	Typ danych:	INT

Odpowiednio do przekazywanego parametru są przeprowadzane następujące kontrole:

- Czy identyfikator istnieje
- Czy jest to tablica jedno- czy dwuwymiarowa
- Czy indeks tablicy jest dozwolony

Tylko gdy wszystkie te kontrole mają wynik pozytywny, jest zwracane TRUE (1). Jeżeli tylko jedna kontrola ma wynik negatywny, albo gdy wystąpił błąd składni, wówczas jest to kwitowane przez FALSE (0).

Przykłady

Kod programu	Komentarz
DEF INT VAR1	
DEF BOOL IS_VAR=FALSE	
N10 IS_VAR=ISVAR("VAR1")	; IS_VAR jest w tym przypadku TRUE.

Kod programu	Komentarz
DEF REAL VARARRAY[10,10]	
DEF BOOL IS_VAR=FALSE	
N10 IS_VAR=ISVAR("VARARRAY[,]")	; IS_VAR jest w tym przypadku TRUE, tablica jest dwuwymiarowa.
N20 IS_VAR=ISVAR("VARARRAY")	; IS_VAR jest TRUE, zmienna istnieje.
N30 IS_VAR=ISVAR("VARARRAY[8,11]")	; IS_VAR jest FALSE, indeks tablicy niedozwolony.
N40 IS_VAR=ISVAR("VARARRAY[8,8]")	; IS_VAR jest FALSE, brakuje "]" (błąd składni).
N50 IS_VAR=ISVAR("VARARRAY[,8]")	; IS_VAR jest TRUE, indeks tablicy jest dozwolony.
N60 IS_VAR=ISVAR("VARARRAY[8,]")	; IS_VAR jest TRUE, indeks tablicy jest dozwolony.

Kod programu	Komentarz
DEF BOOL IS_VAR=FALSE	
N100 IS_VAR=ISVAR("\$MC_GCODE_RESET_VALUES[1]")	; Przekazywany parametr jest daną maszynową, IS_VAR jest TRUE.

Kod programu	Komentarz
DEF BOOL IS_VAR=FALSE	
N10 IS_VAR=ISVAR("\$P_EP")	; IS_VAR jest w tym przypadku TRUE.
N20 IS_VAR=ISVAR("\$P_EP[X]")	; IS_VAR jest w tym przypadku TRUE.

1.1.16 Odczyt wartości absolutnych/typu danych (GETVARPHU, GETVARAP, GETVARLIM, GETVARDFT, GETVARTYP)

Przy pomocy funkcji predefiniowanych GETVARPHU, GETVARAP, GETVARLIM i GETVARDFT można czytać wartości atrybutów zmiennych systemowych/użytkownika, przy pomocy GETVARTYP typ danych zmiennej systemowej/użytkownika.

Odczyt jednostki fizycznej

Składnia:

<wynik>=GETVARPHU (<nazwa>)

Znaczenie:

<wynik>:	Wartość liczbowa jednostki fizycznej	
	Typ danych:	INT
	Zakres wartości:	patrz tablica w "Atrybut: jednostka fizyczna (PHU) (Strona 38)"
		W przypadku błędu: - 2 Podana <nazwa> nie jest przyporządkowana do żadnego parametru systemowego i żadnej zmiennej użytkownika.
GETVARPHU:	Odczyt jednostki fizycznej zmiennej systemowej/użytkownika	
<nazwa>:	Nazwa zmiennej systemowej/użytkownika	
	Typ danych:	STRING

Przykład:

Niech NCK zawiera następującą zmienną GUD:

```
DEF CHAN REAL PHU 42 LLI 0 ULI 10000 electric
```

Kod programu	Komentarz
DEF INT result=0	
result=GETVARPHU("electric")	; Określ jednostkę fizyczną zmiennej GUD.
IF (result < 0) GOTOF error	

Jako wynik jest zwracana wartość 42. Odpowiada to jednostce fizycznej [kW].

Wskazówka

Przy pomocy GETVARPHU można np. sprawdzić, czy dla przyporządkowania zmiennej a = b obydwie zmienne mają oczekiwane jednostki fizyczne.

Odczyt prawa dostępu

Składnia:

<wynik>=GETVARAP (<nazwa>, <dostęp>)

Znaczenie:

<wynik>:	Poziom ochrony dla podanego <dostępu>		
	Typ danych:	INT	
	Zakres wartości:	0 ... 7	Patrz "Atrybut: prawa dostępu (APR, APW, APRP, APWP, APRB, APWB) (Strona 40)".
		W przypadku błędu:	
		- 2	Podana <nazwa> nie jest przyporządkowana do żadnego parametru systemowego i żadnej zmiennej użytkownika.
- 3	nieprawidłowa wartość dla <dostęp>		
GETVARAP:	Odczyt prawa dostępu do zmiennej systemowej/użytkownika		
<nazwa>:	Nazwa zmiennej systemowej/użytkownika		
	Typ danych:	STRING	
<dostęp>:	Rodzaj dostępu		
	Typ danych:	STRING	
	Zakres wartości:	"RP":	Odczyt poprzez program obróbki
		"WP":	Zapis poprzez program obróbki
		"RB":	Odczyt poprzez BTSS
"WB":		Zapis poprzez BTSS	

Przykład:

Kod programu	Komentarz
DEF INT result=0 result=GETVARAP("\$TC_MAP8", "WB") IF (result < 0) GOTO error	 ; Określ ochronę przed dostępem dla parametru systemowego "pozycja magazynu" odnośnie zapisu poprzez BTSS.

Jako wynik jest zwracana wartość 7. Odpowiada to pozycji 0 przełącznika z kluczykiem (= brak ochrony przed dostępem).

Wskazówka

Przy pomocy GETVARAP można np. realizować program sprawdzania, który sprawdza prawa dostępu oczekiwane przez aplikację.

Odczyt wartości granicznych

Składnia:

<status>=GETVARLIM(<nazwa>,<wartość graniczna>,<wynik>)

Znaczenie:

<status>:	Status funkcjonalny		
	Typ danych:	INT	
	Zakres wartości:	1	o. k.
		-1	żadna wartość graniczna nie jest zdefiniowana (w przypadku zmiennych typu AXIS, STRING, FRAME)
		-2	Podana <nazwa> nie jest przyporządkowana do żadnego parametru systemowego i żadnej zmiennej użytkownika.
-3		nieprawidłowa wartość dla <wartość graniczna>	
GETVARLIM:	Odczyt dolnej/górnej wartości granicznej zmiennej systemowej/użytkownika		
<nazwa>:	Nazwa zmiennej systemowej/użytkownika		
	Typ danych:	STRING	
<wartość graniczna>:	Podaje, która wartość graniczna ma zostać odczytana		
	Typ danych:	CHAR	
	Zakres wartości:	"L":	= dolna wartość graniczna
		"U":	= górna wartość graniczna
<wynik>:	Zwrot wartości granicznej		
	Typ danych:	VAR REAL	

Przykład:

Kod programu	Komentarz
DEF INT state=0	
DEF REAL result=0	
state=GETVARLIM("\$MA_MAX_AX_VELO","L",result)	; Określ dolną wartość graniczną dla MD32000 \$MA_MAX_AX_VELO.
IF (result < 0) GOTO error	

Odczyt wartości standardowej

Składnia:

```
<status>=GETVARDFT(<nazwa>,<wynik>[,<indeks_1>,<indeks_2>,<indeks_3>])
```

Znaczenie:

<status>:	Status funkcjonalny			
	Typ danych:	INT		
	Zakres wartości:	1	o. k.	
		-1	Wartość standardowa nie jest dostępna (np. ponieważ <wynik> ma nieprawidłowy typ dla <nazwa>)	
		-2	Podana <nazwa> nie jest przyporządkowana do żadnego parametru systemowego i żadnej zmiennej użytkownika.	
		-3	Nieprawidłowa wartość dla <indeks_1>, wymiar mniejszy od jedności (= nie tablica = skalar)	
-4		Nieprawidłowa wartość dla <indeks_2>		
-5	Nieprawidłowa wartość dla <indeks_3>			
GETVARDFT:	Odczyt wartości standardowej zmiennej systemowej/użytkownika			
<nazwa>:	Nazwa zmiennej systemowej/użytkownika			
	Typ danych:	STRING		
<wynik>:	Zwrot wartości standardowej			
	Typ danych:	VAR REAL (przy odczycie wartości standardowej zmiennych typów INT, REAL, BOOL, AXIS)		
		VAR STRING (przy odczycie wartości standardowej zmiennych typów STRING i CHAR)		
		VAR FRAME (przy odczycie wartości standardowej zmiennych typu FRAME)		
<indeks_1>:	Indeks dla pierwszego wymiaru (opcjonalnie)			
	Typ danych:	INT		
	Nie zaprogramowanie oznacza = 0			
<indeks_2>:	Indeks dla drugiego wymiaru (opcjonalnie)			
	Typ danych:	INT		
	Nie zaprogramowanie oznacza = 0			
<indeks_3>:	Indeks dla trzeciego wymiaru (opcjonalnie)			
	Typ danych:	INT		
	Nie zaprogramowanie oznacza = 0			

Przykład:

Kod programu	Komentarz
DEF INT state=0	
DEF REAL resultR=0	; Zmienna dla ujęcia wartości standardowych typów INT, REAL, BOOL, AXIS.
DEF FRAME resultF=0	; Zmienna dla ujęcia wartości standardowych typu FRAME
IF (GETVARTYP("\$MA_MAX_AX_VELO") <> 4) GOTO error	
state=GETVARDFT("\$MA_MAX_AX_VELO", resultR, AXTOINT(X))	; Określ wartość standardową osi "X". ; AXTOINT konwertuje nazwę osi "X" na odpowiedni indeks dostępu.
IF (resultR < 0) GOTO error	
IF (GETVARTYP("\$TC_TP8") <> 3) GOTO error	
state=GETVARDFT("\$TC_TP8", resultR)	
IF (GETVARTYP("\$P_UBFR") <> 7) GOTO error	
state=GETVARDFT("\$P_UBFR", resultF)	

Odczyt typu danych

Składnia:

<wynik>=GETVARTYP (<nazwa>)

Znaczenie:

<wynik>:	Typ danych podanej zmiennej systemowej/użytkownika		
	Typ danych:	INT	
	Zakres wartości:	1	= BOOL
		2	= CHAR
		3	= INT
		4	= REAL
		5	= STRING
		6	= AXIS
		7	= FRAME
W przypadku błędu:			
< 0	Podana <nazwa> nie jest przyporządkowana do żadnego parametru systemowego i żadnej zmiennej użytkownika.		
GETVARTYP:	Odczyt typu danych zmiennej systemowej/użytkownika		
<nazwa>:	Nazwa zmiennej systemowej/użytkownika		
	Typ danych:	STRING	

Przykład:

Kod programu	Komentarz
DEF INT result=0	
DEF STRING name="R"	
result=GETVARTYP(name)	; Określ typ parametru R.
IF (result < 0) GOTO error	

Jako wynik jest zwracana wartość 4. Odpowiada to typowi danych REAL.

1.2 Programowanie pośrednie

1.2.1 Pośrednie programowanie adresów

Funkcja

Przy pośrednim programowaniu adresów adres rozszerzony (<indeks>) jest zastępowany przez zmienną odpowiedniego typu.

Wskazówka

Programowanie pośrednie adresów jest niemożliwe przy:

- N (numer bloku)
- L (podprogram)
- Adresy ustawiane
(np. X[1] zamiast X1 jest niedopuszczalne)

Składnia

<ADRES> [<indeks>]

Znaczenie

Element	Opis
<ADRES> [...]:	Adres stały z rozszerzeniem (indeks)
<indeks>:	Zmienna np. dla numeru wrzeciona, osi, ...

Przykłady

Przykład 1: Pośrednie programowanie numeru wrzeciona

Programowanie bezpośrednie:

Kod programu	Komentarz
S1=300	; Prędkość obrotowa 300 obr/min dla wrzeciona o numerze 1.

Programowanie pośrednie:

Kod programu	Komentarz
DEF INT SPINU=1	; Definicja zmiennej typu INT i przyporządkowanie wartości.
S[SPINU]=300	; Prędkość obrotowa 300 obr/min dla wrzeciona, którego numer jest zapisany w zmiennej SPINU (w tym przykładzie wrzeciono o numerze 1).

Przykład 2: Programowanie pośrednie jednej osi

Programowanie bezpośrednie:

Kod programu	Komentarz
FA[U]=300	; Posuw 300 dla osi "U".

Programowanie pośrednie:

Kod programu	Komentarz
DEF AXIS AXVAR2=U	; Definicja jednej zmiennej typu AXIS i przyporządkowanie wartości.
FA[AXVAR2]=300	; Posuw 300 dla osi, której nazwa adresowa jest zapisana w zmiennej o nazwie AXVAR2.

Przykład 3: Programowanie pośrednie jednej osi

Programowanie bezpośrednie:

Kod programu	Komentarz
\$AA_MM[X]	; Odczyt wartości zmierzonej przez sondę pomiarową (MKS) osi "X".

Programowanie pośrednie:

Kod programu	Komentarz
DEF AXIS AXVAR3=X	; Definicja jednej zmiennej typu AXIS i przyporządkowanie wartości.
\$AA_MM[AXVAR3]	; Odczyt wartości zmierzonej przez sondę pomiarową (MKS) dla osi, której nazwa jest zapisana w zmiennej AXVAR3.

Przykład 4: Pośrednie programowanie jednej osi

Programowanie bezpośrednie:

Kod programu
X1=100 X2=200

Programowanie pośrednie:

Kod programu	Komentarz
DEF AXIS AXVAR1 AXVAR2	; Definicja dwóch zmiennych typu AXIS.
AXVAR1=(X1) AXVAR2=(X2)	; Przyporządkowanie nazw osi.
AX[AXVAR1]=100 AX[AXVAR2]=200	; Ruch w osiach, których nazwy adresowe są zapisane w zmiennych o nazwach AXVAR1 i AXVAR2.

Przykład 5: Programowanie pośrednie jednej osi

Programowanie bezpośrednie:

Kod programu
G2 X100 I20

Programowanie pośrednie:

Kod programu	Komentarz
DEF AXIS AXVAR1=X	; Definicja jednej zmiennej typu AXIS i przyporządkowanie wartości.
G2 X100 IP[AXVAR1]=20	; Programowanie pośrednie podania punktu środkowego dla osi, której nazwa adresowa jest zapisana w zmiennej o nazwie AXVAR1

Przykład 6: Programowanie pośrednie elementów tablicy

Programowanie bezpośrednie:

Kod programu	Komentarz
DEF INT FELD1[4,5]	; Definicja pola 1.

Programowanie pośrednie:

Kod programu	Komentarz
DEFINE DIM1 AS 4	; Wymiary tablicy muszą zostać podane jako wartości stałe.
DEFINE DIM2 AS 5	
DEF INT FELD[DIM1,DIM2]	
FELD[DIM1-1,DIM2-1]=5	

Przykład 7: Pośrednie wywołanie podprogramu

Kod programu	Komentarz
CALL "L" << R10	; Wywołanie programu, którego numer znajduje się w R10 (powiązanie łańcuchów znaków).

1.2.2 Pośrednie programowanie G-Code**Funkcja**

Programowanie pośrednie G-Code umożliwia efektywne programowanie cykli.

Składnia

G [<grupa>] = <numer>

Znaczenie

G [...]: Polecenie G z rozszerzeniem (indeks)
<grupa>: Parametr indeksu: grupa funkcji G
Typ: INT
<numer>: Zmienna dla numeru G-Code
Typ: INT lub REAL

Wskazówka

Z reguły można pośrednio programować tylko G-Code nie określających składni.

Z G-Code określających składnię są możliwe tylko te z grupy 1 funkcji G.
Określające składnię G-Code grup 2, 3 i 4 funkcji G są niemożliwe.

Wskazówka

W pośrednim programowaniu G-Code nie są dozwolone żadne funkcje arytmetyczne.
Konieczne obliczenie numeru G-Code musi nastąpić w oddzielnym wierszu programu obróbki przed pośrednim zaprogramowaniem G-Code.

Przykłady

Przykład 1: Ustawiane przesunięcie punktu zerowego (grupa 8 funkcji G)

Kod programu	Komentarz
N1010 DEF INT INT_VAR	
N1020 INT_VAR=2	
...	
N1090 G[8]=INT_VAR G1 X0 Y0	; G54
N1100 INT_VAR=INT_VAR+1	; Obliczenie G-Code
N1110 G[8]=INT_VAR G1 X0 Y0	; G55

Przykład 2: wybór płaszczyzny (grupa 6 funkcji G)

Kod programu	Komentarz
N2010 R10=\$P_GG[6]	; Odczyt aktywnej funkcji G z grupy 6
...	
N2090 G[6]=R10	

Literatura

Informacje dot. grup funkcji G patrz:
Podręcznik programowania Podstawy; punkt "Grupy funkcji G"

1.2.3 Programowanie pośrednie atrybutów pozycji (GP)

Funkcja

Atrybuty pozycji, jak np. przyrostowe albo absolutne programowanie pozycji osi, mogą być programowane w połączeniu ze słowem kluczowym GP pośrednio jako zmienne.

Zastosowanie

Programowanie pośrednie atrybutów pozycji znajduje zastosowanie w **cyklach podstawienia**, ponieważ ma tu miejsce następująca zaleta w stosunku do programowania atrybutów pozycji jako słowo kluczowe (np. IC, AC, ...):

Dzięki programowaniu pośredniemu jako zmienne **nie** jest potrzebna instrukcja CASE, która rozgałęzia na wszystkie możliwe atrybuty pozycji.

Składnia

```
<POLECENIE POZYCJONOWANIA> [<oś/wrzeciono>] =
GP (<pozycja>, <atrybut pozycji>)
<oś/wrzeciono>=GP (<pozycja>, <atrybut pozycji>)
```

Znaczenie

<POLECENIE POZYCJONOWANIA> []:

Następujące polecenia pozycjonowania mogą być programowane razem ze słowem kluczowym GP:

POS, POSA, SPOS, SPOSA

Poza tym możliwe:

- Wszystkie występujące w kanale identyfikatory osi/wrzeciona

<oś/wrzeciono>

- Zmienny identyfikator osi/wrzeciona AX

Oś/wrzeciono, które ma być pozycjonowane

Słowo kluczowe do pozycjonowania

Parametr 1

Pozycja osi/wrzeciona jako stała lub zmienna

Parametr 2

Atrybut pozycji (np. tryb ruchu do pozycji) jako zmienna (np. \$P_SUB_SPOSMODE) lub jako słowo kluczowe (IC, AC, ...)

<oś/wrzeciono>:

GP():

<pozycja>:

<atrybut pozycji>:

Wartości dawane przez zmienne mają następujące znaczenie:

Wartość	Znaczenie	Dopuszczalny przy:
0	Bez zmiany atrybutu pozycji	
1	AC	POS, POSA, SPOS, SPOSA, AX, Adres osi
2	IC	POS, POSA, SPOS, SPOSA, AX, Adres osi
3	DC	POS, POSA, SPOS, SPOSA, AX, Adres osi
4	ACP	POS, POSA, SPOS, SPOSA, AX, Adres osi
5	ACN	POS, POSA, SPOS, SPOSA, AX, Adres osi
6	OC	-
7	PC	-
8	DAC	POS, POSA, AX, Adres osi
9	DIC	POS, POSA, AX, Adres osi
10	RAC	POS, POSA, AX, Adres osi
11	RIC	POS, POSA, AX, Adres osi
12	CAC	POS, POSA
13	CIC	POS, POSA
14	CDC	POS, POSA
15	CACP	POS, POSA
16	CACN	POS, POSA

Przykład

Przy aktywnym sprzężeniu wrzeciona synchronicznego między wrzecionem wiodącym S1 i wrzecionem nadążnym S2 jest przez polecenie SPOS w programie głównym wywoływany następujący cykl podstawienia do pozycjonowania wrzecion.

Pozycjonowanie następuje poprzez instrukcję w N2230:
 SPOS [1] =GP (\$P_SUB_SPOSIT, \$P_SUB_SPOSMODE)
 SPOS [2] =GP (\$P_SUB_SPOSIT, \$P_SUB_SPOSMODE)

Pozycja, do której ma nastąpić ruch, jest odczytywana ze zmiennej systemowej \$P_SUB_SPOSIT, tryb ruchu - ze zmiennej systemowej \$P_SUB_SPOSMODE.

Kod programu	Komentarz
N1000 PROC LANG_SUB DISPLOF SBLOF	
...	
N2100 IF (\$P_SUB_AXFCT==2)	
N2110	; Zastąpienie polecenia SPOS / SPOSA / M19 przy aktywnym sprzężeniu wrzecion synchronicznych
N2185 DELAYFSTON	; Początek zakresu stopp-delay
N2190 COUPOF (S2, S1)	; Wyłączenie aktywności sprzężenia wrzecion synchronicznych
N2200	; Pozycjonowanie wrzeciona wiodącego i nadążnego
N2210 IF (\$P_SUB_SPOS==TRUE) OR (\$P_SUB_SPOSA==TRUE)	
N2220	; Pozycjonowanie wrzeciona przy pomocy SPOS:
N2230 SPOS [1] =GP (\$P_SUB_SPOSIT, \$P_SUB_SPOSMODE)	
SPOS [2] =GP (\$P_SUB_SPOSIT, \$P_SUB_SPOSMODE)	
N2250 ELSE	
N2260	; Pozycjonowanie wrzeciona przy pomocy M19:
N2270 M1=19 M2=19	; Pozycjonowanie wrzeciona wiodącego i nadążnego
N2280 ENDIF	
N2285 DELAYFSTOF	; Koniec zakresu stopp-delay
N2290 COUPON (S2, S1)	; Uaktywnienie sprzężenia wrzeciona synchronicznego
N2410 ELSE	
N2420	; Odpytanie na dalsze zastąpienia
...	
N3300 ENDIF	
...	
N9999 RET	

Warunki brzegowe

- W akcjach synchronicznych pośrednie programowanie atrybutów pozycji jest niemożliwe.

Literatura

Podręcznik działania Funkcje podstawowe; BAG, kanał, praca programowa, zachowanie się przy zresetowaniu (K1),
Rozdział: Zastępowanie funkcji NC przez podprogramy

1.2.4 Pośrednie programowanie wierszy programu obróbki (EXECSTRING)

Funkcja

Przy pomocy polecenia programu obróbki `EXECSTRING` jest możliwe wykonanie przedtem utworzonej zmiennej string, jako wiersza programu obróbki.

Składnia

`EXECSTRING` jest programowany we własnym wierszu programu obróbki:
`EXECSTRING (<zmienna string>)`

Znaczenie

<code>EXECSTRING:</code>	Polecenie do wykonania zmiennej string, jako wiersza programu obróbki
<code><zmienna string>:</code>	Zmienna typu STRING, która zawiera będący właściwie do wykonania wiersz programu obróbki

Wskazówka

Przy pomocy `EXECSTRING` mogą być osadzone wszystkie układy, które mogą być programowane w części programowej programu obróbki. Wykluczone są przez to instrukcje `PROC` i `DEF`, jak też generalnie stosowanie w plikach INI i DEF.

Przykład

Kod programu	Komentarz
N100 DEF STRING[100] BLOCK	; Definicja zmiennej string do ujęcia będącego do wykonania wiersza programu obróbki.
N110 DEF STRING[10] MFCT1="M7"	
...	
N200 EXECSTRING(MFCT1 << "M4711")	; Wykonanie wiersza programu obróbki "M7 M4711".
...	
N300 R10=1	
N310 BLOCK="M3"	
N320 IF(R10)	
N330 BLOCK = BLOCK << MFCT1	
N340 ENDIF	
N350 EXECSTRING(BLOCK)	; Wykonanie wiersza programu obróbki "M3 M7".

1.3 Funkcje arytmetyczne

Funkcja

Funkcje obliczeniowe dają się stosować przede wszystkim dla parametrów R i zmiennych (albo stałych i funkcji) typu REAL. Dopuszczalne są również typy INT i CHAR.

Operator / funkcja obliczeniowa	Znaczenie
+	Dodawanie
-	Odejmowanie
*	Mnożenie
/	Dzielenie
	Uwaga: (typ INT)/(typ INT)=(typ REAL); Przykład: 3/4 = 0.75
DIV	Dzielenie, dla typu zmiennej INT i REAL
	Uwaga: (typ INT)DIV(typ INT)=(typ INT); Przykład: 3 DIV 4 = 0
MOD	Dzielenie modulo (tylko typ INT) daje resztę z dzielenia INT Przykład: 3 MOD 4 = 3
:	Operator powiązania (w przypadku zmiennych FRAME)
SIN()	Sinus

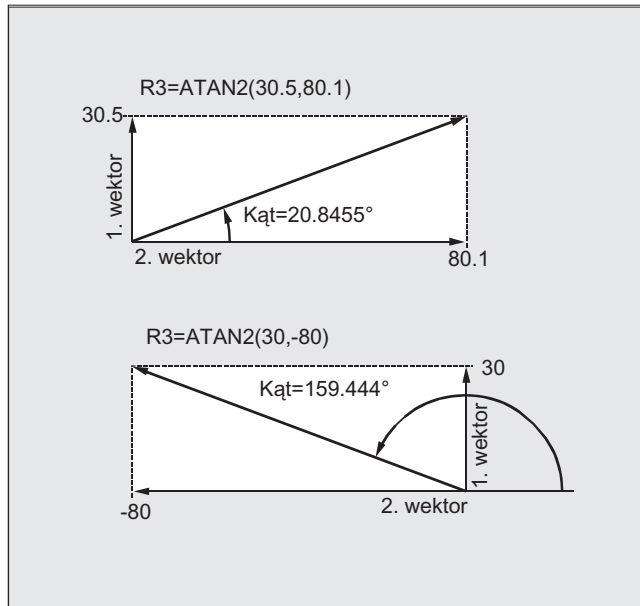
COS ()	Cosinus
TAN ()	Tangens
ASIN ()	Arcus sinus
ACOS ()	Arcus cosinus
ATAN2 (,)	Arcus tangens2
SQRT ()	Pierwiastek kwadratowy
ABS ()	Wartość bezwzględna
POT ()	2. potęga (kwadrat)
TRUNC ()	Część całkowitoliczbowa Dokładności w przypadku poleceń porównań, ustawiane przy pomocy TRUNC(patrz "Korekcja dokładności w przypadku błędów porównania (TRUNC) (Strona 75)")
ROUND ()	Zaokrąglenie do liczby całkowitej
LN ()	Logarytm naturalny
EXP ()	Funkcja wykładnicza
MINVAL ()	Mniejsza wartość z dwóch zmiennych (patrz "Minimum, maksimum i zakres zmiennych (MINVAL, MAXVAL i BOUND) (Strona 77)")
MAXVAL ()	Większa wartość z dwóch zmiennych (patrz "Minimum, maksimum i zakres zmiennych (MINVAL, MAXVAL i BOUND) (Strona 77)")
BOUND ()	Wartość zmiennej, która leży w zdefiniowanym zakresie (patrz "Minimum, maksimum i zakres zmiennych (MINVAL, MAXVAL i BOUND) (Strona 77)")
CTRANS ()	Przesunięcie
CROT ()	Obrót
CSCALE ()	Zmiana skali
CMIRROR ()	Lustrzane odbicie

Programowanie

W przypadku funkcji obliczeniowych obowiązuje zwykły matematyczny sposób zapisu. Priorytety wykonywania są ustawiane przez nawiasy okrągłe. Dla funkcji trygonometrycznych i ich funkcji odwrotnych obowiązuje podawanie w stopniach (prawy kąt = 90°).

Przykłady

Przykład 1: ATAN2



Z dwóch prostopadłych do siebie wektorów funkcja obliczeniowa ATAN2 oblicza kąt wektora sumarycznego.

Wynik leży w zakresie czterech ćwiartek koła (-180° < 0 < +180°).

Bazą dla odniesienia kąta jest zawsze 2. wartość w kierunku dodatnim.

Przykład 2: Inicjalizacja kompletnych tablic zmiennych

Kod programu	Komentarz
R1=R1+1	; Nowe R1 = stare R1 +1
R1=R2+R3 R4=R5-R6 R7=R8*R9	
R10=R11/R12 R13=SIN(25.3)	
R14=R1*R2+R3	; "Kropka przed kreską".
R14=(R1+R2)*R3	; Najpierw obliczenie w nawiasach.
R15=SQRT(POT(R1)+POT(R2))	; Najpierw następuje obliczenie w nawiasach wewnętrznych: R15 = pierwiastek kwadratowy z (R1+R2)
RESFRAME=FRAME1:FRAME2	; Przy pomocy operatora powiązania frame są łączone w
FRAME3=CTRANS(...):CROT(...)	jeden frame wynikowy albo komponentom frame są przyporządkowywane wartości.

1.4 Operacje porównania i operacje logiczne

Funkcja

Operacje porównania mogą np. być używane do formułowania warunku skoku. Porównywane mogą być przy tym również skomplikowane wyrażenia.

Operacje porównania można stosować do zmiennych typu `CHAR`, `INT`, `REAL` i `BOOL`. W przypadku typu `CHAR` jest porównywana wartość kodowa.

W przypadku typów `STRING`, `AXIS` i `FRAME` są możliwe: `==` i `<>`, które dla operacji typu `STRING` mogą być stosowane również w akcjach synchronicznych.

Wynik operacji porównania jest zawsze typu `BOOL`.

Operatory logiczne służą do powiązania wartości logicznych.

Operacje logiczne dają się stosować tylko do zmiennych typu `BOOL`. Przez wewnętrzną konwersję typu dają się one stosować również do typów danych `CHAR`, `INT` i `REAL`.

W przypadku operacji logicznych (boolowskich) obowiązuje dla typów danych `BOOL`, `CHAR`, `INT` i `REAL`:

- 0 odpowiada: `FALSE`
- Nierówne 0 odpowiada: `TRUE`

Bitowe operatory logiczne

Na zmiennych typu `CHAR` i `INT` mogą również pojedynczymi bitami być wykonywane operacje logiczne. Ewentualnie konwersja typu następuje automatycznie.

Programowanie

Operator porównania	Znaczenie
<code>==</code>	Równe
<code><></code>	Nierówne
<code>></code>	Większe niż
<code><</code>	Mniejsze niż
<code>>=</code>	Większe lub równe
<code><=</code>	Mniejsze lub równe

Operator logiczny	Znaczenie
<code>AND</code>	<code> </code>
<code>OR</code>	<code>LUB</code>
<code>NOT</code>	Negacja
<code>XOR</code>	<code>ALBO</code>

Bitowy operator logiczny	Znaczenie
B_AND	Bitowe I
B_OR	Bitowe LUB
B_NOT	Bitowa negacja
B_XOR	Bitowe ALBO

Wskazówka

W wyrażeniach arytmetycznych można przy pomocy nawiasów okrągłych ustalić kolejność wykonywania wszystkich operatorów, a przez to czynić odstępstwa od normalnych zasad pierwszeństwa.

Wskazówka

Między BOOLOWSKIMI argumentami i operatorami muszą być pisane spacje.

Wskazówka

Operator B_NOT odnosi się do tylko jednego argumentu. Znajduje się on za operatorem.

Przykłady

Przykład 1: Operatory porównania

```
IF R10>=100 GOTOF CEL
```

lub

```
R11=R10>=100
```

```
IF R11 GOTOF CEL
```

Wynik porównania `R10>=100` jest najpierw poddawany pośredniemu zapisaniu w `R11`.

Przykład 2: Operatory logiczne

```
IF (R10<50) AND ($AA_IM[X]>=17.5) GOTOF CEL
```

lub

```
IF NOT R10 GOTOB START
```

NOT odnosi się tylko do jednego argumentu.

Przykład 3: Bitowe operatory logiczne

```
IF $MC_RESET_MODE_MASK B_AND 'B10000' GOTOF ACT_PLANE
```

1.5 Korekcja dokładności w przypadku błędów porównania (TRUNC)

Funkcja

Polecenie TRUNC odcina argument pomnożony przez współczynnik dokładności.

Ustawiana dokładność w przypadku błędów porównania

Dane programu obróbki typu REAL są wewnątrznie przedstawiane w formacie IEEE przy pomocy 64 bitów. Ze względu na tę formę przedstawienia liczby dziesiętne mogą być odwzorowywane niedokładnie, co przy porównywaniu z idealnie obliczonymi wartościami może prowadzić do nieoczekiwanych wyników.

Równość względna

Aby niedokładności wywołane przez formę przedstawienia nie zniekształcały przebiegu programu, w przypadku poleceń porównania jest sprawdzana nie równość absolutna lecz równość względna.

Składnia

Korekcja dokładności w przypadku błędów porównania

```
TRUNC (R1*1000)
```

Znaczenie

TRUNC: Odcięcie miejsc po przecinku

Uwzględniona równość względna 10^{-12} przy

- Równość: (==)
- Równość: (<>)
- Większe lub równe: (>=)
- Mniejsze lub równe: (<=)
- Większe niż/mniejsze niż: (><) z równością absolutną
- Większe od: (>)
- Mniejsze od: (<)

Kompatybilność

Z powodów kompatybilności można wyłączyć aktywność kontroli na równość względną przy (>) i (<) przez ustawienie danej maszynowej MD10280 \$MN_PROG_FUNCTION_MASK Bit0 = 1.

Wskazówka

Porównania z danymi typu REAL są z wymienionych powodów generalnie obciążone pewną niedokładnością. Przy nie akceptowalnych odchyleniach konieczne jest przejście na rachunek INTEGER przez mnożenie argumentów przez współczynnik dokładności, a następnie obcinanie przy pomocy TRUNC.

Akcje synchroniczne

Opisane zachowanie się poleceń porównania obowiązuje również w przypadku akcji synchronicznych.

Przykłady**Przykład 1: Zagadnienie dokładności**

Kod programu	Komentarz
N40 R1=61.01 R2=61.02 R3=0.01	; Przyporządkowanie wartości początkowych
N41 IF ABS(R2-R1) > R3 GOTOF BLAD	; Skok zostałby dotychczas wykonany
N42 M30	; Koniec programu
N43 BLAD: SETAL(66000)	;
R1=61.01 R2=61.02 R3=0.01	; Przyporządkowanie wartości początkowych
R11=TRUNC(R1*1000) R12=TRUNC(R2*1000) R13=TRUNC(R3*1000)	; Korekcja dokładności
IF ABS(R12-R11) > R13 GOTOF BLAD	; Skok nie zostanie już wykonany
M30	; Koniec programu
BLAD: SETAL(66000)	;

Przykład 2: utworzenie i ewaluacja ilorazu obydwu argumentów

Kod programu	Komentarz
R1=61.01 R2=61.02 R3=0.01	; Przyporządkowanie wartości początkowych
IF ABS((R2-R1)/R3) - 1 > 10EX-5 GOTOF BLAD	; Skok nie zostanie wykonany
M30	; Koniec programu
BLAD: SETAL(66000)	;

1.6 Minimum, maksimum i zakres zmiennych (MINVAL, MAXVAL i BOUND)

Funkcja

Przy pomocy poleceń `MINVAL` i `MAXVAL` można porównać ze sobą wartości dwóch zmiennych. Jako wynik jest zwracana mniejsza (w przypadku `MINVAL`) lub większa wartość (w przypadku `MAXVAL`).

Przy pomocy polecenia `BOUND` można sprawdzić, czy wartość sprawdzanej zmiennej leży w zdefiniowanym zakresie.

Składnia

```
<mniejsza wartość>=MINVAL(<zmienna1>,<zmienna2>)
<większa wartość>=MAXVAL(<zmienna1>,<zmienna2>)
<zwracana wartość>=<BOUND>(<minimum>,<maksimum>,<sprawdzana
zmienna>)
```

Znaczenie

<code>MINVAL:</code>	Określa mniejszą wartość z dwóch zmiennych (<code><zmienna1></code> , <code><zmienna2></code>)
<code><mniejsza wartość>:</code>	Zmienna wynikowa dla polecenia <code>MINVAL</code> Jest ustawiana na mniejszą wartość zmiennej.
<code>MAXVAL:</code>	Określa większą wartość z dwóch zmiennych (<code><zmienna1></code> , <code><zmienna2></code>)
<code><większa wartość>:</code>	Zmienna wynikowa dla polecenia <code>MAXVAL</code> Jest ustawiana na większą wartość zmiennej.
<code>BOUND:</code>	Sprawdza, czy zmienna (<code><zmienna sprawdzana></code>) leży w zdefiniowanym zakresie wartości.
<code><minimum>:</code>	Zmienna, która definiuje wartość minimalną zakresu
<code><maksimum>:</code>	Zmienna, która definiuje wartość maksymalną zakresu
<code><zwracana wartość>:</code>	Zmienna wynikowa dla polecenia <code>BOUND</code> Jeżeli wartość sprawdzanej zmiennej leży w zdefiniowanym zakresie wartości, wówczas zmienna wynikowa jest ustawiana na wartość tej zmiennej. Jeżeli wartość sprawdzanej zmiennej jest większa, niż wartość maksymalna, wówczas zmienna wynikowa jest ustawiana na maksymalną wartość zdefiniowanego zakresu. Jeżeli wartość sprawdzanej zmiennej jest mniejsza, niż wartość minimalna, wówczas zmienna wynikowa jest ustawiana na minimalną wartość zdefiniowanego zakresu.

Wskazówka

MINVAL, MAXVAL i BOUND można programować również w akcjach synchronicznych.

Wskazówka**Zachowanie się w przypadku równości**

W przypadku równości jest przy MINVAL/MAXVAL dawana jest równa wartość. W przypadku BOUND jest zwracana wartość sprawdzanej zmiennej.

Przykład

Kod programu	Komentarz
DEF REAL rVar1=10.5, rVar2=33.7, rVar3, rVar4, rVar5, rValMin, rValMax, rRetVar	
rValMin=MINVAL(rVar1,rVar2)	; rValMin jest ustawiana na wartość 10.5.
rValMax=MAXVAL(rVar1,rVar2)	; rValMax jest ustawiana na wartość 33.7.
rVar3=19.7	
rRetVar=BOUND(rVar1,rVar2,rVar3)	; rVar3 leży w ramach granic, rRetVar jest ustawiana na 19.7.
rVar3=1.8	
rRetVar=BOUND(rVar1,rVar2,rVar3)	; rVar3 leży poniżej granicy minimum, rRetVar jest ustawiana na 10,5.
rVar3=45.2	
rRetVar=BOUND(rVar1,rVar2,rVar3)	; rVar3 leży powyżej granicy maksimum, rRetVar jest ustawiana na 33,7.

1.7 Priorytet operacji

Funkcja

Do każdego operatora jest przyporządkowany priorytet. Przy ewaluacji wyrażenia są zawsze najpierw stosowane operatory o wyższym priorytecie. W przypadku operatorów o takiej samej randze ewaluacja następuje od lewej do prawej.

W wyrażeniach arytmetycznych można przy pomocy nawiasów okrągłych ustalić kolejność wykonywania wszystkich operatorów, a przez to czynić odstępstwa od normalnych zasad pierwszeństwa.

Kolejność operatorów

Od najwyższego do najniższego priorytetu

1.	NOT, B_NOT	Negacja, negacja bitowa
2.	*, /, DIV, MOD	Mnożenie, dzielenie
3.	+, -	Dodawanie, odejmowanie
4.	B_AND	Bitowe I
5.	B_XOR	Bitowe ALBO
6.	B_OR	Bitowe LUB
7.	AND	I
8.	XOR	ALBO
9.	OR	LUB
10.	<<	Powiązanie łańcuchów znaków, typ wyniku STRING
11.	==, <>, >, <, >=, <=	Operatory porównania

Wskazówka

Operator powiązania ":" dla Frame nie może występować w jednym wyrażeniu z innymi operatorami. Dlatego dla tego operatora nie wymaga się ustalenia priorytetu.

Przykład instrukcji If

```
If (otto==10) and (anna==20) gotof end
```

1.8 Możliwe konwersje typu

Funkcja

Konwersja typu przy przyporządkowaniu

Stała wartość liczbowa, zmienna albo wyrażenie, które jest przyporządkowywane do zmiennej, musi tolerować się z typem tej zmiennej. Jeżeli tak jest, wówczas przy przyporządkowywaniu typ jest automatycznie zmieniany.

Możliwe konwersje typu

na	REAL	INT	BOOL	CHAR	STRING	AXIS	FRAME
z							
REAL	tak	tak*	tak ¹⁾	tak*	–	–	–
INT	tak	tak	tak ¹⁾	tak ²⁾	–	–	–
BOOL	tak	tak	tak	tak	tak	–	–
CHAR	tak	tak	tak ¹⁾	tak	tak	–	–
STRING	–	–	tak ⁴⁾	tak ³⁾	tak	–	–
AXIS	–	–	–	–	–	tak	–
FRAME	–	–	–	–	–	–	tak

Objaśnienia

- * Przy zmianie typu z REAL na INT następuje w przypadku ułamka ≥ 0.5 zaokrąglenie do góry, w przeciwnym przypadku następuje zaokrąglenie do dołu (por. funkcja ROUND)
- 1) Wartość $< > 0$ odpowiada TRUE, wartość $== 0$ odpowiada FALSE
- 2) Gdy wartość leży w dopuszczalnym zakresie liczbowym
- 3) Gdy tylko 1 znak
- 4) Długość łańcucha znaków $0 \Rightarrow$ FALSE, w innym przypadku TRUE

Wskazówka

Jeżeli przy konwersji wartość jest większa, niż zakres docelowy, następuje komunikat błędu.

Jeżeli w wyrażeniu występują mieszane typy, wówczas ich dopasowanie następuje automatycznie. Konwersje typu są możliwe również w akcjach synchronicznych, patrz punkt "Akcje synchroniczne ruchu, implicite konwersja typu".

1.9 Operacja na łańcuchach znaków

Operacje na łańcuchach znaków

Oprócz klasycznych operacji "przyporządkowanie" i "porównanie" są możliwe następujące operacje na łańcuchach znaków:

- Konwersja typu na STRING (AXSTRING) (Strona 81)
- Konwersja typu z STRING (NUMBER, ISNUMBER, AXNAME) (Strona 82)
- Powiązanie łańcuchów znaków (<<) (Strona 84)
- Zmiana na litery małe/duże (TOLOWER, TOUPPER) (Strona 85)
- Określenie długości łańcucha znaków (STRLEN) (Strona 86)
- Szukanie znaku/łańcucha znaków w łańcuchu znaków (INDEX, RINDEX, MINDEX, MATCH) (Strona 86)
- Wybór częściowego łańcucha znaków (SUBSTR) (Strona 88)
- Odczyt i zapis pojedynczych znaków (Strona 89)
- Formatowanie łańcucha znaków (SPRINT) (Strona 90)

Znaczenie specjalne znaku 0

Znak 0 jest wewnętrznie interpretowany jako oznaczenie końca łańcucha znaków. Jeżeli znak zostanie zastąpiony przez znak 0, łańcuch znaków zostanie przez to skrócony.

Przykład:

Kod programu	Komentarz
DEF STRING[20] STRG="Oś . stoi"	
STRG[6]="X"	
MSG(STRG)	; Daje komunikat "Oś X stoi".
STRG[6]=0	
MSG(STRG)	; Daje komunikat "Oś".

1.9.1 Konwersja typu na STRING (AXSTRING)

Funkcja

Dzięki funkcji "konwersja typu na STRING" można wykorzystywać zmienne różnego typu, jako część składową komunikatu (MSG).

Następuje przy stosowaniu operatora << implicite dla typów danych INT, REAL, CHAR i BOOL (patrz "Powiązanie łańcuchów znaków (<<) (Strona 84)").

Wartość INT jest zmieniana w normalnie czytelną formę. W przypadku wartości REAL jest podawanych do 10 miejsc po przecinku.

Przy pomocy polecenia `AXSTRING` zmienne typu `AXIS` mogą być konwertowane na `STRING`.

Składnia

```
<STRING_ERG> = << <dowolny_typ>
<STRING_ERG> = AXSTRING(<identyfikator osi>)
```

Znaczenie

<STRING_ERG>:	Zmienna dla wyniku konwersji typu Typ: STRING
<dowolny_typ>:	Typy zmiennych INT, REAL, CHAR, STRING i BOOL
AXSTRING:	Polecenie AXSTRING daje podany identyfikator osi jako łańcuch znaków.
<identyfikator osi>:	Zmienna dla identyfikatora osi Typ: AXIS

Wskazówka

Zmienne FRAME nie mogą być konwertowane.

Przykłady**Przykład 1:**

```
MSG ("Pozycja: "<<$AA_IM[X] )
```

Przykład 2: AXSTRING

Kod programu	Komentarz
DEF STRING[32] STRING_ERG	
STRING_ERG=AXSTRING(X)	; STRING_ERG == "X"

1.9.2 Konwersja typu z STRING (NUMBER, ISNUMBER, AXNAME)**Funkcja**

Przy pomocy polecenia NUMBER następuje konwersja ze STRING na REAL. Konwertowalność można sprawdzić poleceniem ISNUMBER.

Przy pomocy polecenia AXNAME łańcuch znaków jest konwertowany na typ danych AXIS.

Składnia

```
<REAL_ERG>=NUMBER("<łańcuch znaków>")
<BOOL_ERG>=ISNUMBER("<łańcuch znaków>")
<AXIS_ERG>=AXNAME("<łańcuch znaków>")
```

Znaczenie

NUMBER:	Polecenie NUMBER zwraca liczbę przedstawioną przez <łańcuch znaków> jako wartość REAL.	
<łańcuch znaków>:	Konwertowana zmienna typu STRING	
<REAL_ERG>:	Zmienna dla wyniku konwersji typu przy pomocy NUMBER Typ: REAL	
ISNUMBER:	Przy pomocy polecenia ISNUMBER można sprawdzić, czy <łańcuch znaków> można przekonwertować na poprawną liczbę.	
<BOOL_ERG>:	Zmienna dla wyniku odpytania przy pomocy ISNUMBER Typ: BOOL Wartość TRUE ć:	
	FALSE	ISNUMBER daje wartość TRUE, gdy <łańcuch znaków> przedstawia liczbę REAL poprawną według zasad języka. Jeżeli ISNUMBER da wartość FALSE, jest przy wywołaniu NUMBER z takim samym <łańcuchem znaków> wyzwalany alarm.
AXNAME:	Polecenie AXNAME konwertuje podany <łańcuch znaków> na identyfikator osi. Wskazówka: Jeżeli <łańcuch znaków> nie może zostać przyporządkowany do żadnego z zaprojektowanych identyfikatorów osi, jest wyzwalany alarm.	
<AXIS_ERG>:	Zmienna dla wyniku konwersji typu przy pomocy AXNAME Typ: AXIS	

Przykład

Kod programu	Komentarz
DEF BOOL BOOL_ERG	
DEF REAL REAL_ERG	
DEF AXIS AXIS_ERG	
BOOL_ERG=ISNUMBER("1234.9876Ex-7")	; BOOL_ERG == TRUE
BOOL_ERG=ISNUMBER("1234XYZ")	; BOOL_ERG == FALSE
REAL_ERG=NUMBER("1234.9876Ex-7")	; REAL_ERG == 1234.9876Ex-7
AXIS_ERG=AXNAME("X")	; AXIS_ERG == X

1.9.3 Powiązanie łańcuchów znaków (<<)

Funkcja

Funkcja "Powiązanie łańcuchów znaków" stwarza możliwość składania łańcucha znaków z poszczególnych części składowych.

Powiązanie jest realizowane przez operator "<<". Ten operator ma typ docelowy STRING dla wszystkich kombinacji typów bazowych CHAR, BOOL, INT, REAL i STRING. Ewentualnie niezbędna konwersja jest dokonywana według istniejących zasad.

Składnia

```
<dowolny_typ> << <dowolny_typ>
```

Znaczenie

<dowolny_typ>: Zmienna typu CHAR, BOOL, INT, REAL lub STRING

<< : Operator do powiązania zmiennych (<dow._typ>) w złożony łańcuch znaków (typ STRING).

Ten operator jest dostępny również jako jedyny tzw. wariant "unarny". W ten sposób jest możliwe wykonanie explicite konwersji typu na STRING (nie dla FRAME i AXIS):

```
<< <dow._typ>
```

Na przykład można w ten sposób złożyć komunikat albo polecenie z list tekstowych i wstawić parametr (na przykład nazwę modułu):

```
MSG (STRG_TAB [LOAD_IDX] <<MODUŁ_NAZWA)
```

Wskazówka

Wyniki pośrednie przy wiązaniu łańcuchów znaków nie mogą przekraczać maksymalnej długości łańcucha znaków.

Wskazówka

Typy FRAME i AXIS nie mogą być stosowane razem z operatorem "<<".

Przykłady

Przykład 1: Powiązanie łańcuchów znaków

Kod programu	Komentarz
DEF INT IDX=2	
DEF REAL VALUE=9.654	
DEF STRING [20] STRG="INDEX:2"	
IF STRG=="Index:"<<IDX GOTOF NO_MSG	
MSG ("indeks:" <<IDX <<"/wartość:" <<VALUE)	; Wyświetlenie:
	"indeks: 2/wartość: 9.654"
NO_MSG:	

Przykład 2: Dokładny typ konwersji za pomocą <<

Kod programu	Komentarz
DEF REAL VALUE=3.5	
<<VALUE	; Podana zmienna typu REAL jest konwertowana na typ STRING.

1.9.4 Zmiana na litery małe/duże (TOLOWER, TOUPPER)

Funkcja

Funkcja "zmiana na litery małe/duże" pozwala zmienić wszystkie litery łańcucha znaków na jednolite ich przedstawienie.

Składnia

```
<STRING_ERG>=TOUPPER("<łańcuch znaków>")  
<STRING_ERG>=TOLOWER("<łańcuch znaków>")
```

Znaczenie

TOUPPER:	Przy pomocy polecenia <code>TOUPPER</code> wszystkie litery łańcucha znaków są zamieniane na litery duże .
TOLOWER:	Przy pomocy polecenia <code>TOLOWER</code> wszystkie litery łańcucha znaków są zamieniane na litery małe .
<łańcuch znaków>:	Łańcuch znaków, który ma zostać przekonwertowany Typ: STRING
<STRING_ERG>:	Zmienna dla wyniku konwersji Typ: STRING

Przykład

Ponieważ jest również możliwe zainicjalizowanie wprowadzeń użytkownika na interfejsie graficznym, można uzyskać jednolitą prezentację z literami małymi lub dużymi:

```
Kod programu  
DEF STRING [29] STRG  
...  
IF "LEARN.CNC"==TOUPPER(STRG) GOTOF LOAD_LEARN
```

1.9.5 Określenie długości łańcucha znaków (STRLEN)

Funkcja

Przy pomocy polecenia `STRLEN` jest możliwe określenie długości łańcucha znaków.

Składnia

```
<INT_ERG>=STRLEN ("<STRING>")
```

Znaczenie

`STRLEN`: Przy pomocy polecenia `STRLEN` jest określana długość podanego łańcucha znaków.

Jest zwracana liczba znaków, które - licząc od początku łańcucha znaków - nie są znakiem 0.

<łańcuch znaków>: Łańcuch znaków, którego długość ma zostać określona

Typ: STRING

<INT_ERG>: Zmienna dla wyniku określenia

Typ: INT

Przykład

Funkcja w związku z dostępem do pojedynczych znaków umożliwia określenie końca łańcucha znaków:

Kod programu

```
IF (STRLEN (MODUŁ_NAZWA) > 10) GOTO BŁĄD
```

1.9.6 Szukanie znaku/łańcucha znaków w łańcuchu znaków (INDEX, RINDEX, MINDEX, MATCH)

Funkcja

Funkcja ta pozwala na poszukiwanie pojedynczych znaków wzgl. ich łańcucha w innym łańcuchu. Wyniki funkcji podają, w jakiej pozycji łańcucha znaleziono znak/łańcuch.

Składnia

```
INT_ERG=INDEX (STRING, CHAR) ; Typ wyniku: INT
```

```
INT_ERG=RINDEX (STRING, CHAR) ; Typ wyniku: INT
```

```
INT_ERG=MINDEX (STRING, STRING) ; Typ wyniku: INT
```

```
INT_ERG=MATCH (STRING, STRING) ; Typ wyniku: INT
```

Semantyka

Funkcje szukania: Zwracają one pozycję w łańcuchu znaków (pierwszy parametr), w którym poszukiwanie zakończyło się wynikiem pozytywnym. Jeżeli znaku/łańcucha nie można znaleźć, jest zwracana wartość -1. Pierwszy znak ma przy tym pozycję 0.

Znaczenie

INDEX:	Poszukuje w pierwszym parametrze znaku podanego jako drugi parametr (od przodu).
RINDEX:	Poszukuje w pierwszym parametrze znaku podanego jako drugi parametr (od tyłu).
MINDEX:	Odpowiada funkcji INDEX, oprócz tego, że jest przekazywana lista znaków (jako łańcuch znaków), z których jest zwracany indeks pierwszego znalezionej znaku.
MATCH:	Szuka łańcucha znaków w łańcuchu znaków.

W ten sposób łańcuchy znaków dają się dzielić według określonych kryteriów, na przykład w pozycjach ze spacją albo znakiem rozdzielającym przy podawaniu ścieżki ("/").

Przykład

Wprowadzenie do podziału na nazwy ścieżek i modułów

Kod programu	Komentarz
DEF INT PFADIDX, PROGIDX	
DEF STRING[26] WPROWADZENIE	
DEF INT LISTIDX	
WPROWADZENIE = "/_N_MPF_DIR/_N_EXECUTE_MPF"	
LISTIDX = MINDEX (WPROWADZENIE, "M,N,O,P") + 1	; Jako wartość LISTIDX jest zwracane 3; ponieważ "N" jest w parametrze WPROWADZENIE pierwszym od przodu znakiem z listy wyboru.
PFADIDX = INDEX (WPROWADZENIE, "/") + 1	; przez to obowiązuje: PFADIDX = 1
PROGIDX = RINDEX (WPROWADZENIE, "/") + 1	; przez to obowiązuje: PROGIDX = 12
	Przy pomocy wprowadzonej w następnym punkcie funkcji SUBSTR zmienna WPROWADZENIE daje się rozłożyć na komponenty "ścieżka" i "moduł":
VARIABLE = SUBSTR (WPROWADZENIE, PFADIDX, PROGIDX-PFADIDX-1)	; daje wówczas "_N_MPF_DIR"
ZMIENNA = SUBSTR (WPROWADZENIE, PROGIDX)	; daje wówczas "_N_EXECUTE_MPF"

1.9.7 Wybór częściowego łańcucha znaków (SUBSTR)

Funkcja

Przy pomocy funkcji SUBSTRING można czytać dowolne części w ramach łańcucha znaków.

Składnia

```
<STRING_ERG>=SUBSTR(<łańcuch_znaków>,<indeks>,<długość>)
```

```
<STRING_ERG>=SUBSTR(<łańcuch_znaków>,<indeks>)
```

Znaczenie

SUBSTR: Funkcja daje z <łańcucha_znaków> łańcuch częściowy, rozpoczynając od <indeksu> o podanej <długości>.

Gdy parametr <długości> nie jest podany, funkcja daje częściowy łańcuch znaków rozpoczynając od <indeksu> do końca łańcucha znaków.

<indeks>: Pozycja początkowa częściowego łańcucha znaków w ramach łańcucha znaków. Jeżeli pozycja początkowa znajduje się za końcem łańcucha znaków, jest zwracany łańcuch pusty (""). Pierwszy znak łańcucha znaków: indeks = 0

Zakres wartości: 0 ... (długość łańcucha znaków - 1)

<długość>: Długość częściowego łańcucha znaków. Gdy zostanie podana za duża długość, jest zwracany tylko częściowy łańcuch znaków do końca łańcucha znaków.

Zakres wartości: 1 ... (długość łańcucha znaków - 1)

Przykład

Kod programu	Komentarz
DEF STRING[29] ERG	
; 1	
; 0123456789012345678	
ERG = SUBSTR ("POKWITOWANIE: 10 do 99", 10, 2)	; ERG == "10"
ERG = SUBSTR("POKWITOWANIE: 10 do 99", 10)	; ERG == "10 do 99"

1.9.8 Odczyt i zapis pojedynczych znaków

Funkcja

W ramach łańcucha znaków można czytać i zapisywać poszczególne znaki.

Należy przy tym przestrzegać następujących warunków brzegowych:

- możliwe tylko w przypadku zmiennych definiowanych przez użytkownika, nie w przypadku zmiennych systemowych
- poszczególne znaki łańcucha są przy wywołaniach podprogramów przekazywane tylko jako "call by value"

Składnia

```
<znak>=<łańcuch znaków>[<indeks>]  
<znak>=<łańcuch_znaków_tablica>[<tablica_indeks>,<indeks>]  
<łańcuch_znaków>[<indeks>]=<znak>  
<łańcuch_znaków_tablica>[<tablica_indeks>,<indeks>]=<znak>
```

Znaczenie

<łańcuch znaków>: Dowolny łańcuch znaków
<znak>: Zmienna typu CHAR
<indeks>: Pozycja znaku w ramach łańcucha znaków.
Pierwszy znak łańcucha znaków: indeks = 0
Zakres wartości: 0 ... (długość łańcucha znaków -1)

Przykłady

Przykład 1: Zmienna komunikat

Kod programu	Komentarz
; 0123456789	
DEF STRING[50] KOMUNIKAT = "Os n doszła do pozycji"	
KOMUNIKAT[6] = "X"	
MSG(KOMUNIKAT)	; "Os X doszła do pozycji"

Przykład 2: Ewaluacja zmiennej systemowej

Kod programu	Komentarz
DEF STRING[50] STRG	; Pamięć pośrednia dla zmiennej systemowej
...	
STRG = \$P_MMCA	; Załadowanie zmiennej systemowej
IF STRG[0] == "E" GOTO ...	; Ewaluacja zmiennej systemowej

Przykład 3: Przekazanie parametrów "call by value" i "call by reference"

Kod programu	Komentarz
; 0123456	
DEF STRING[50] STRG = "Achse X"	
DEF CHAR CHR	
...	
EXTERN UP_VAL(ACHSE)	; Definicja podprogramu z parametrem "call by value"
EXTERN UP_REF(VAR ACHSE)	; Definicja podprogramu z parametrem "call by reference"
...	
UP_VAL(STRG[6])	; Przekazanie parametrów "by value"
...	
CHR = STRG[6]	; Zapisywanie tymczasowe
UP_REF(CHR)	; Przekazanie parametrów "by reference"

1.9.9 Formatowanie łańcucha znaków (SPRINT)

Funkcja

Przy pomocy predefiniowanej funkcji SPRINT mogą być formatowane łańcuchy znaków i np. przygotowywane do wyprowadzenia do urządzeń zewnętrznych (patrz też "Process DataShare - Wyprowadzenie do zewnętrznego urządzenia/pliku (EXTOPEN, WRITE, EXTCLOSE) (Strona 614)").

Składnia

```
"<Wynik_String>"=SPRINT("<Format_String>", <Wartość_1>, <Wartość_2>, ..
., <Wartość_n>)
```

Znaczenie

SPRINT:	Identyfikator funkcji predefiniowanej, która daje wartość typu STRING.
"<Format_String>":	Łańcuch znaków, który zawiera części stałe i zmienne. Części zmienne są ustalane poprzez formatyzator % i następujący po nim opis formatu.
< Wartość_1>, < Wartość_2>, ..., < Wartość_n>:	Wartość w formie stałej albo zmiennej NC, która w miejscu, w którym znajduje się n-ty formatyzator %, jest wstawiana do <Format_String> odpowiednio do opisu formatu.
"<Wynik_String>":	Formatowany łańcuch znaków (maksymalnie 400 bajtów)

Dostępne opisy formatów

%B:	<p>Konwersja na łańcuch znaków "TRUE", gdy konwertowana wartość:</p> <ul style="list-style-type: none">• jest nierówna 0.• nie jest pustym łańcuchem znaków (w przypadku wartości string). <p>Konwersja na łańcuch znaków "FALSE", gdy konwertowana wartość:</p> <ul style="list-style-type: none">• jest równa 0.• jest pustym łańcuchem znaków. <p>Przykład:</p> <pre>N10 DEF BOOL BOOL_VAR=1 N20 DEF STRING[80] RESULT N30 RESULT=SPRINT("CONTENT OF BOOL_VAR:%B", BOOL_VAR)</pre> <p>Wynik: Zmienna string RESULT jest opisywana przy pomocy łańcucha znaków "CONTENT OF BOOL_VAR:TRUE".</p>
%C:	<p>Konwersja na znak ASCII.</p> <p>Przykład:</p> <pre>N10 DEF CHAR CHAR_VAR="X" N20 DEF STRING[80] RESULT N30 RESULT=SPRINT("CONTENT OF CHAR_VAR:%C", CHAR_VAR)</pre> <p>Wynik: Zmienna string RESULT jest opisywana przy pomocy łańcucha znaków "CONTENT OF CHAR_VAR:X".</p>
%D:	<p>Konwersja na łańcuch znaków wartości całkowitoliczbowej (INTEGER).</p> <p>Przykład:</p> <pre>N10 DEF INT INT_VAR=123 N20 DEF STRING[80] RESULT N30 RESULT=SPRINT("CONTENT OF INT_VAR:%D", INT_VAR)</pre> <p>Wynik: Zmienna string RESULT jest opisywana przy pomocy łańcucha znaków "CONTENT OF INT_VAR:123".</p>
%<m>D:	<p>Konwersja na łańcuch znaków wartości całkowitoliczbowej (INTEGER). Łańcuch znaków ma długość minimalna <m> znaków. Brakujące miejsca są wypełniane do lewej spacjami.</p> <p>Przykład:</p> <pre>N10 DEF INT INT_VAR=-123 N20 DEF STRING[80] RESULT N30 RESULT=SPRINT("CONTENT OF INT_VAR:%6D", INT_VAR)</pre> <p>Wynik: Zmienna string RESULT jest opisywana przez łańcuch znaków "CONTENT OF INT_VAR:xx-123" ("x" w przykładzie zastępuje spację).</p>
%F:	<p>Konwersja na łańcuch znaków liczby dziesiętnej z 6 miejscami po przecinku. Miejsca po przecinku są ew. zaokrąglane albo wypełniane zerami.</p> <p>Przykład:</p> <pre>N10 DEF REAL REAL_VAR=-1.2341234EX+03 N20 DEF STRING[80] RESULT N30 RESULT=SPRINT("CONTENT OF REAL_VAR:%F", REAL_VAR)</pre> <p>Wynik: Zmienna RESULT typu string jest opisywana przy pomocy łańcucha znaków "CONTENT OF REAL_VAR: -1234.123400".</p>

%<m>F:	<p>Konwersja na łańcuch znaków liczby dziesiętnej o 6 miejscach po przecinku i długości całkowitej co najmniej <m> znaków. Miejsca po przecinku są ew. zaokrąglane albo wypełniane zerami. Brakujące znaki są wypełniane spacjami do długości całkowitej <m> z wyrównaniem do lewej.</p> <p>Przykład:</p> <pre>N10 DEF REAL REAL_VAR=-1.23412345678EX+03 N20 DEF STRING[80] RESULT N30 RESULT=SPRINT("CONTENT OF REAL_VAR:%15F", REAL_VAR)</pre> <p>Wynik: Zmienna string RESULT jest opisywana przez łańcuch znaków "CONTENT OF REAL_VAR: xxx-1234.123457" ("x" w przykładzie zastępuje spację).</p>
%.<n>F:	<p>Konwersja na łańcuch znaków liczby dziesiętnej z <n> miejscami po przecinku. Miejsca po przecinku są ew. zaokrąglane albo wypełniane zerami.</p> <p>Przykład:</p> <pre>N10 DEF REAL REAL_VAR=-1.2345678EX+03 N20 DEF STRING[80] RESULT N30 RESULT=SPRINT("CONTENT OF REAL_VAR:%.3F", REAL_VAR)</pre> <p>Wynik: Zmienna string RESULT jest opisywana przy pomocy łańcucha znaków "CONTENT OF REAL_VAR: -1234.568".</p>
%<m>.<n>F:	<p>Konwersja na łańcuch znaków liczby dziesiętnej o <n> miejscach po przecinku i długości całkowitej co najmniej <m> znaków. Miejsca po przecinku są ew. zaokrąglane albo wypełniane zerami. Brakujące znaki są wypełniane spacjami do długości całkowitej <m> z wyrównaniem do lewej.</p> <p>Przykład:</p> <pre>N10 DEF REAL REAL_VAR=-1.2341234567890EX+03 N20 DEF STRING[80] RESULT N30 RESULT=SPRINT("CONTENT OF REAL_VAR:%10.2F", REAL_VAR)</pre> <p>Wynik: Zmienna string RESULT jest opisywana przez łańcuch znaków "CONTENT OF REAL_VAR:xx-1234.12" ("x" w przykładzie zastępuje spację).</p>
%E:	<p>Konwersja na łańcuch znaków liczby dziesiętnej w formie wykładniczej. Mantysa jest zapisywana w znormalizowaniu z jednym miejscem przed przecinkiem i 6 miejscach po przecinku. Miejsca po przecinku są ew. zaokrąglane albo wypełniane zerami. Wykładnik rozpoczyna się od słowa kluczowego "EX". Dalej następuje znak liczby ("+" albo "-") i liczba dwu- lub trzycyfrowa.</p> <p>Przykład:</p> <pre>N10 DEF REAL REAL_VAR=-1234.567890 N20 DEF STRING[80] RESULT N30 RESULT=SPRINT("CONTENT OF REAL_VAR:%E", REAL_VAR)</pre> <p>Wynik: Zmienna string RESULT jest opisywana przy pomocy łańcucha znaków "CONTENT OF REAL_VAR:-1.234568EX+03".</p>

%<m>E:	<p>Konwersja na łańcuch znaków liczby dziesiętnej w formie wykładniczej o długości całkowitej co najmniej <m> znaków. Brakujące znaki są wypełniane spacjami z wyrównaniem do lewej. Mantysa jest zapisywana w znormalizowaniu z jednym miejscem przed przecinkiem i 6 miejscach po przecinku. Miejsca po przecinku są ew. zaokrąglane albo wypełniane zerami. Wykładnik rozpoczyna się od słowa kluczowego "EX". Dalej następuje znak liczby ("+" albo "-") i liczba dwu- lub trzycyfrowa.</p> <p>Przykład:</p> <pre>N10 DEF REAL REAL_VAR=-1234.5 N20 DEF STRING[80] RESULT N30 RESULT=SPRINT("CONTENT OF REAL_VAR:%20E",REAL_VAR)</pre> <p>Wynik: Zmienna string RESULT jest opisywana przez łańcuch znaków "CONTENT OF REAL_VAR:xxxxxx-1.234500EX+03" ("x" w przykładzie zastępuje spację)</p>
%.<n>E:	<p>Konwersja na łańcuch znaków liczby dziesiętnej w formie wykładniczej. Mantysa jest zapisywana w znormalizowaniu z jednym miejscem przed przecinkiem i <n> miejscach po przecinku. Miejsca po przecinku są ew. zaokrąglane albo wypełniane zerami. Wykładnik rozpoczyna się od słowa kluczowego "EX". Dalej następuje znak liczby ("+" albo "-") i liczba dwu- lub trzycyfrowa.</p> <p>Przykład:</p> <pre>N10 DEF REAL REAL_VAR=-1234.5678 N20 DEF STRING[80] RESULT N30 RESULT=SPRINT("CONTENT OF REAL_VAR:%.2E",REAL_VAR)</pre> <p>Wynik: Zmienna string RESULT jest opisywana przy pomocy łańcucha znaków "CONTENT OF REAL_VAR:-1.23EX+03".</p>
%<m>.<n>E:	<p>Konwersja na łańcuch znaków liczby dziesiętnej w formie wykładniczej o długości całkowitej co najmniej <m> znaków. Brakujące znaki są wypełniane spacjami z wyrównaniem do lewej. Mantysa jest zapisywana w znormalizowaniu z jednym miejscem przed przecinkiem i <n> miejscach po przecinku. Miejsca po przecinku są ew. zaokrąglane albo wypełniane zerami. Wykładnik rozpoczyna się od słowa kluczowego "EX". Dalej następuje znak liczby ("+" albo "-") i liczba dwu- lub trzycyfrowa.</p> <p>Przykład:</p> <pre>N10 DEF REAL REAL_VAR=-1234.5678 N20 DEF STRING[80] RESULT N30 RESULT=SPRINT("CONTENT OF REAL_VAR:%12.2E",REAL_VAR)</pre> <p>Wynik: Zmienna string RESULT jest opisywana przez łańcuch znaków "CONTENT OF REAL_VAR:xx-1.23EX+03" ("x" w przykładzie zastępuje spację).</p>

%G:	<p>Konwersja na łańcuch znaków liczby dziesiętnej zależnie od zakresu wartości w formie dziesiętnej albo wykładniczej: jeżeli przedstawiana wartość ma wartość bezwzględną mniejszą, niż 1.0EX-04 albo większą/równą 1.0EX+06, jest wybierana forma wykładnicza, w przeciwnym przypadku forma dziesiętna. Jest wyświetlanych maksymalnie sześć miejsc znaczących, ew. następuje zaokrąglenie.</p> <p>Przykład z przedstawieniem dziesiętnym:</p> <pre>N10 DEF REAL REAL_VAR=1.234567890123456EX-04 N20 DEF STRING[80] RESULT N30 RESULT=SPRINT("CONTENT OF REAL_VAR:%G",REAL_VAR)</pre> <p>Wynik: Zmienna string RESULT jest opisywana przy pomocy łańcucha znaków "CONTENT OF REAL_VAR: 0.000123457".</p> <p>Przykład z przedstawieniem wykładniczym:</p> <pre>N10 DEF REAL REAL_VAR=1.234567890123456EX+06 N20 DEF STRING[80] RESULT N30 RESULT=SPRINT("CONTENT OF REAL_VAR:%G",REAL_VAR)</pre> <p>Wynik: Zmienna string RESULT jest opisywana przy pomocy łańcucha znaków "CONTENT OF REAL_VAR:1.23457EX+06".</p>
%<m>G:	<p>Konwersja na łańcuch znaków liczby dziesiętnej zależnie od zakresu wartości w formie dziesiętnej albo wykładniczej (jak %G). Łańcuch znaków ma długość całkowitą co najmniej <m> znaków. Brakujące znaki są wypełniane spacjami z wyrównaniem do lewej.</p> <p>Przykład z przedstawieniem dziesiętnym:</p> <pre>N10 DEF REAL REAL_VAR=1.234567890123456EX-04 N20 DEF STRING[80] RESULT N30 RESULT=SPRINT("CONTENT OF REAL_VAR:%15G",REAL_VAR)</pre> <p>Wynik: Zmienna string RESULT jest opisywana przez łańcuch znaków "CONTENT OF REAL_VAR:xxxx0.000123457" ("x" w przykładzie zastępuje spację).</p> <p>Przykład z przedstawieniem wykładniczym:</p> <pre>N10 DEF REAL REAL_VAR=1.234567890123456EX+06 N20 DEF STRING[80] RESULT N30 RESULT=SPRINT("CONTENT OF REAL_VAR:%15G",REAL_VAR)</pre> <p>Wynik: Zmienna string RESULT jest opisywana przez łańcuch znaków "CONTENT OF REAL_VAR:xxx1.23457EX+06" ("x" w przykładzie zastępuje spację).</p>
%.<n>G:	<p>Konwersja na łańcuch znaków liczby dziesiętnej zależnie od zakresu wartości w formie dziesiętnej albo wykładniczej. Jest wyświetlanych maksymalnie <n> miejsc znaczących, ew. następuje zaokrąglenie. Jeżeli wartość bezwzględna przedstawianej wartości jest mniejsza, niż 1.0EX-04 albo większa/równa 1.0EX(+<n>) jest wybierana forma wykładnicza, w przeciwnym przypadku forma dziesiętna.</p> <p>Przykład z przedstawieniem dziesiętnym:</p> <pre>N10 DEF REAL REAL_VAR=1.234567890123456EX-04 N20 DEF STRING[80] RESULT N30 RESULT=SPRINT("CONTENT OF REAL_VAR:%.3G",REAL_VAR)</pre> <p>Wynik: Zmienna string RESULT jest opisywana przy pomocy łańcucha znaków "CONTENT OF REAL_VAR: 0.000123".</p> <p>Przykład z przedstawieniem wykładniczym:</p> <pre>N10 DEF REAL REAL_VAR=1.234567890123456EX+03 N20 DEF STRING[80] RESULT N30 RESULT = SPRINT("CONTENT OF REAL_VAR:%.3G",REAL_VAR)</pre> <p>Wynik: Zmienna string RESULT jest opisywana przy pomocy łańcucha znaków "CONTENT OF REAL_VAR:1.23EX+03".</p>

<p><code>%<m> . <n>G:</code></p>	<p>Konwersja na łańcuch znaków liczby dziesiętnej zależnie od zakresu wartości w formie dziesiętnej albo wykładniczej (jak <code>% . <n>G</code>). Łańcuch znaków ma długość całkowitą co najmniej <code><m></code> znaków. Brakujące znaki są wypełniane spacjami z wyrównaniem do lewej.</p> <p>Przykład z przedstawieniem dziesiętnym:</p> <pre>N10 DEF REAL REAL_VAR=1.234567890123456EX-04 N20 DEF STRING[80] RESULT N30 RESULT=SPRINT("CONTENT OF REAL_VAR:%12.4G",REAL_VAR)</pre> <p>Wynik: Zmienna string RESULT jest opisywana przez łańcuch znaków "CONTENT OF REAL_VAR:xxx0.0001235" ("x" w przykładzie zastępuje spację).</p> <p>Przykład z przedstawieniem wykładniczym:</p> <pre>N10 DEF REAL REAL_VAR=1.234567890123456EX+04 N20 DEF STRING[80] RESULT N30 RESULT=SPRINT("CONTENT OF REAL_VAR:%12.4G",REAL_VAR)</pre> <p>Wynik: Zmienna string RESULT jest opisywana przez łańcuch znaków "CONTENT OF REAL_VAR:xx1.235EX+06" ("x" w przykładzie zastępuje spację).</p>
<p><code>% . <n>P:</code></p>	<p>Konwersja wartości REAL na wartość INTEGER przy uwzględnieniu <code><n></code> miejsc po przecinku. Wartość INTEGER jest wyprowadzana, jako liczba binarna 32-bitowa. Jeżeli konwertowanej wartości nie można przedstawić przy pomocy 32 bitów, wykonywanie jest przerwane z alarmem.</p> <p>Ponieważ ciąg bajtów utworzony przy pomocy instrukcji formatu <code>% . <n>P</code> może zawierać również zera binarne, tak utworzony całkowity łańcuch znaków nie odpowiada już konwersjom typu danych NC STRING. Dlatego, ani nie można go zapisać w zmiennej typu STRING, ani dalej przetwarzać przy użyciu poleceń string języka NC. Jedynym możliwym zastosowaniem jest przekazanie parametrów do polecenia <code>WRITE</code> z wyprowadzeniem do odpowiedniego urządzenia zewnętrznego (patrz poniższy przykład).</p> <p>Gdy tylko <code><Format_String></code> zawiera opis formatu typu <code>%P</code>, cały łańcuch znaków, za wyjątkiem liczby binarnej wygenerowanej przy pomocy <code>% . <n>P</code>, jest odpowiednio do MD10750 \$MN_SPRINT_FORMAT_P_CODE wyprowadzany w kodzie ASCII, ISO (DIN6024) albo EIA (RS244). Gdy zostanie zaprogramowany znak niekonwertowalny, wykonywanie jest przerywane z alarmem.</p> <p>Przykład:</p> <pre>N10 DEF REAL REAL_VAR=123.45 N20 DEF INT ERROR N30 DEF STRING[20] EXT_DEVICE="/ext/dev/1" ... N100 EXTOPEN(ERROR,EXT_DEVICE) N110 IF ERROR <> 0 ... ; Postępowanie z błędami N200 WRITE(ERROR,EXT_DEVICE,SPRINT("INTEGER BINARY CODED:%.3P",REAL_VAR)) N210 IF ERROR <> 0 ... ; Postępowanie z błędami</pre> <p>Wynik: Łańcuch znaków "INTEGER BINARY CODED: 'H0001E23A'" jest przesyłany na urządzenie wyjścia /ext/dev/1. Wartość szesnastkowa 0x0001E23A odpowiada wartości dziesiętnej 123450.</p>

%<m> . <n>P:	<p>Łańcuch znaków z:</p> <ul style="list-style-type: none"> • jedną liczbą całkowitą <m> + <n> miejsc albo • liczbą dziesiętną z maksymalnie <m> miejscami przed przecinkiem i dokładnie <n> miejscami po przecinku. <p>Jak w przypadku opisu formatu % . <n>P cały łańcuch znaków jest zapisywany w kodzie znaków ustalonym przez MD10750 \$MN_SPRINT_FORMAT_P_CODE.</p> <p>Konwersja przy MD10751 = 0:</p> <p>Wartość REAL jest konwertowana na łańcuch znaków z liczbą całkowitą o <m> + <n> miejscach. Ew. miejsca po przecinku są zaokrąglane do <n> miejsc albo wypełniane przez 0. Brakujące miejsca przed przecinkiem są wypełniane spacjami. Znak minus jest dołączany z wyrównaniem do lewej, w miejscu znaku plus jest umieszczana spacja.</p> <p>Przykład:</p> <pre>N10 DEF REAL REAL_VAR=-123.45 N20 DEF STRING[80] RESULT N30 RESULT=SPRINT("PUNCHED TAPE FORMAT:%5.3P",REAL_VAR)</pre> <p>Wynik: Zmienna string RESULT jest opisywana przez łańcuch znaków "PUNCHED TAPE FORMAT:-xx123450" ("x" w przykładzie zastępuje spację).</p> <p>Konwersja przy MD10751 = 1:</p> <p>Wartość REAL jest konwertowana na łańcuch znaków z liczbą dziesiętną o maksymalnie <m> miejscach przed przecinkiem i dokładnie <n> miejscach po przecinku. Ew. miejsca przed przecinkiem są obcinane, a miejsca po przecinku zaokrąglane albo wypełniane przez 0. Jeżeli <n> jest równe 0, odpada również kropka dziesiętna.</p> <p>Przykład:</p> <pre>N10 DEF REAL REAL_VAR1=-123.45 N20 DEF REAL REAL_VAR2=123.45 N30 DEF STRING[80] RESULT N40 RESULT=SPRINT("PUNCHED TAPE FORMAT:%5.3P VAR2:%2.0P", REAL_VAR1,REAL_VAR2)</pre> <p>Wynik: Zmienna string RESULT jest opisywana przy pomocy łańcucha znaków "PUNCHED TAPE FORMAT:-123.450 VAR2:23".</p>
%S:	<p>Wstawienie łańcucha znaków.</p> <p>Przykład:</p> <pre>N10 DEF STRING[16] STRING_VAR="ABCDEFGH" N20 DEF STRING[80] RESULT N30 RESULT=SPRINT("CONTENT OF STRING_VAR:%S",STRING_VAR)</pre> <p>Wynik: Zmienna string RESULT jest opisywana przy pomocy łańcucha znaków "CONTENT OF STRING_VAR:ABCDEFGH".</p>
%<m>S:	<p>Wstawienie łańcucha znaków o co najmniej <m> znakach. Brakujące miejsca są wypełniane spacjami.</p> <p>Przykład:</p> <pre>N10 DEF STRING[16] STRING_VAR="ABCDEFGH" N20 DEF STRING[80] RESULT N30 RESULT=SPRINT("CONTENT OF STRING_VAR:%10S",STRING_VAR)</pre> <p>Wynik: Zmienna string RESULT jest opisywana przez łańcuch znaków "CONTENT OF STRING_VAR:xxxABCDEFGH" ("x" w przykładzie zastępuje spację).</p>

%.<n>S:	<p>Wstawienie <n> znaków łańcucha znaków (rozpoczynając od pierwszego znaku).</p> <p>Przykład:</p> <pre>N10 DEF STRING[16] STRING_VAR="ABCDEFGH" N20 DEF STRING[80] RESULT N30 RESULT=SPRINT("CONTENT OF STRING_VAR:%.3S", STRING_VAR)</pre> <p>Wynik: Zmienna string RESULT jest opisywana przy pomocy łańcucha znaków "CONTENT OF STRING_VAR:ABC".</p>
%<m>.<n>S:	<p>Wstawienie <n> znaków łańcucha znaków (rozpoczynając od pierwszego znaku). Utworzony łańcuch znaków ma długość całkowitą co najmniej <m> znaków. Brakujące miejsca są wypełniane spacjami.</p> <p>Przykład:</p> <pre>N10 DEF STRING[16] STRING_VAR="ABCDEFGH" N20 DEF STRING[80] RESULT N30 RESULT=SPRINT("CONTENT OF STRING_VAR:%10.5S", STRING_VAR)</pre> <p>Wynik: Zmienna string RESULT jest opisywana przez łańcuch znaków "CONTENT OF STRING_VAR:xxxxxABCDE" ("x" w przykładzie zastępuje spację).</p>
%X:	<p>Konwersja wartości INTEGER na łańcuch znaków w formie szesnastkowej.</p> <p>Przykład:</p> <pre>N10 DEF INT INT_VAR='HA5B8' N20 DEF STRING[80] RESULT N30 RESULT=SPRINT("INTEGER HEXADECIMAL:%X", INT_VAR)</pre> <p>Wynik: Zmienna string RESULT jest opisywana przy pomocy łańcucha znaków "INTEGER HEXADECIMAL:A5B8".</p>

Wskazówka

Właściwość języka NC, nie rozróżniania dużych i małych liter odnośnie identyfikatorów i słów kluczowych, obowiązuje również dla opisów formatu. Dlatego można je bez funkcjonalnej różnicy programować małymi albo dużymi literami.

Możliwości kombinacji

Poniższa tablica informuje o tym, które typy danych NC z którym opisem formatu można zestawiać. Obowiązują reguły konwersji implicite typu danych (patrz "Typy danych (Strona 53)").

	Typy danych NC						
	BOOL	CHAR	INT	REAL	STRING	AXIS	FRAME
%B	+	+	+	+	+	-	-
%C	-	+	-	-	+	-	-
%D	+	+	+	+	-	-	-
%F	-	-	+	+	-	-	-
%E	-	-	+	+	-	-	-
%G	-	-	+	+	-	-	-
%S	-	+	-	-	+	-	-

	Typy danych NC						
%X	+	+	+	-	-	-	-
%P	-	-	+	+	-	-	-

Wskazówka

Tablica pokazuje, że typów danych NC AXIS i FRAME nie można bezpośrednio stosować w funkcji SPRINT. Jest jednak możliwe:

- przekonwertowanie typu danych AXIS funkcją AXSTRING na łańcuch znaków, który następnie można przetwarzać przy pomocy SPRINT.
- czytanie poszczególnych wartości typu danych FRAME poprzez dostęp do komponentów frame. Przez to uzyskuje się daną typu REAL, którą można przetwarzać dalej przy pomocy SPRINT.

1.10 Skoki i rozgałęzienia w programie

1.10.1 Skok do początku programu (GOTOS)

Funkcja

Przy pomocy polecenia GOTOS jest możliwe, w celu powtórzenia programu, przeskoczenie z powrotem na początek programu głównego albo podprogramu.

Poprzez dane maszynowe można ustawić, że przy każdym skoku powrotnym na początek programu:

- czas przebiegu programu zostanie ustawiony na "0".
- stan licznik obrobionych przedmiotów jest zwiększany o wartość "1".

Składnia

GOTOS

Znaczenie

GOTOS: Instrukcja skoku z celem początek programu.
 Wykonanie jest sterowane przez sygnał interfejsowy NC/PLC:
 DB21, ... DBX384.0 (sterowanie rozgałęzieniem programu)
 Wartość: Znaczenie:
 0 Bez skoku powrotnego do początku programu. Wykonywanie programu jest kontynuowane od następnego bloku programu obróbki po GOTOS.
 1 Skok powrotny do początku programu. Program obróbki jest powtarzany.

Warunki brzegowe

- `GOTOS` wyzwała wewnętrznie `STOPRE` (zatrzymanie przebiegu wyprzedzającego).
- W przypadku programu obróbki z definicjami danych (zmienne LUD) następuje przy pomocy `GOTOS` skok do pierwszego bloku programu za segmentem definicji, tzn. definicje danych nie są ponownie wykonywane. Definiowane zmienne zachowują dlatego wartość uzyskaną w bloku `GOTOS` i nie są cofane do wartości standardowych zaprogramowanych w segmencie definicji.
- W akcjach synchronicznych i cyklach technologicznych polecenie `GOTOS` nie jest dostępne.

Przykład

Kod programu	Komentarz
N10 ...	; Początek programu.
...	
N90 GOTOS	; Skok do początku programu.
...	

1.10.2 Skoki w programie do znaczników skoku (GOTOB, GOTOF, GOTO, GOTOC)

Funkcja

W programie mogą być ustawiane znaczniki skoku (etykiety), do których można wykonywać skoki z innych miejsc tego samego programu przy pomocy poleceń `GOTOF`, `GOTOB`, `GOTO` lub `GOTOC`. Wykonywanie programu jest wówczas kontynuowane od instrukcji, która następuje bezpośrednio po znaczniku skoku. Przez to mogą być realizowane rozgałęzienia w ramach programu.

Oprócz znaczników skoku są jako cele skoku możliwe również numery bloków głównych i pomocniczych.

Gdy przed instrukcją skoku jest sformułowany warunek skoku (`IF ...`), wówczas skok w programie następuje tylko wtedy, gdy ten warunek jest spełniony.

Składnia

```
GOTOB <cel skoku>
IF <warunek skoku> = TRUE GOTOB <cel skoku>
GOTOF <cel skoku>
IF <warunek skoku> = TRUE GOTOF <cel skoku>
GOTO <cel skoku>
IF <warunek skoku> = TRUE GOTO <cel skoku>
GOTOC <cel skoku>
IF <warunek skoku> = TRUE GOTOC <cel skoku>
```

Znaczenie

GOTOB:	Instrukcja skoku z celem w kierunku początku programu.
GOTOF:	Instrukcja skoku z celem w kierunku końca programu.
GOTO:	Instrukcja skoku z szukaniem celu. Szukanie następuje w kierunku końca programu, następnie w kierunku początku programu.
GOTOC:	Działanie GOTO z tą różnicą, że alarm 14080 "Cel skoku nie został znaleziony" jest ukrywany. Oznacza to, że w przypadku bezowocnego szukania celu skoku wykonywanie programu nie jest przerywane, lecz jest kontynuowane od wiersza programu następującego po poleceniu GOTOC.
<cel skoku>:	Parametr celu skoku Możliwymi danymi są: <Znacznik skoku>: Celem skoku jest umieszczony w programie znacznik skoku o nazwie zdefiniowanej przez użytkownika: <Znacznik skoku>: <Numer bloku>: Celem skoku jest numer bloku głównego lub pomocniczego (np. 200, N300) Zmienna typu STRING: Zmienny cel skoku. Zmienna oznacza znacznik skoku albo numer bloku.
IF:	Słowo kluczowe do sformułowania warunku skoku. Warunek skoku dopuszcza wszystkie operacje porównywania i logiczne (wynik: TRUE lub FALSE) Skok w programie jest wykonywany, gdy wynik tej operacji jest TRUE.

Wskazówka

Znaczniki skoku (etykiety)

Znaczniki skoku zawsze znajdują się na początku bloku. Gdy jest numer bloku, znacznik skoku znajduje się bezpośrednio za tym numerem.

Dla nazywania znaczników skoku obowiązują następujące zasady:

- Maksymalna liczba znaków:
 - Co najmniej 2
 - Co najwyżej 32
 - Dopuszczalnymi znakami są:
 - Litery
 - Cyfry
 - Podkreślenia
 - Pierwsze dwa znaki muszą być literami albo podkreśleniami.
 - Po nazwie znacznika skoku następuje dwukropek (":").
-

Warunki brzegowe

- Celem skoku może być tylko blok ze znacznikiem skoku albo numer bloku, który leży w ramach programu.
- Instrukcja skoku bez warunku skoku musi zostać zaprogramowana w oddzielnym bloku. W przypadku instrukcji skoku z warunkami skoku to ograniczenie nie obowiązuje. Tutaj można sformułować wiele instrukcji skoku w jednym bloku.
- W przypadku programów z instrukcjami skoku bez warunków skoku koniec programu M2/M30 nie musi koniecznie znajdować się na końcu programu.

Przykłady

Przykład 1: Skoki do znaczników skoku

Kod programu	Komentarz
N10 ...	
N20 GOTOF Etykieta_1	; Skok w kierunku końca programu do znacznika "Etykieta_1".
N30 ...	
N40 Etykieta_0: R1=R2+R3	; Znacznik skoku "Etykieta_0" ustawiony.
N50 ...	
N60 Etykieta_1:	; Znacznik skoku "Etykieta_1" ustawiony.
N70 ...	
N80 GOTOB Etykieta_0	; Skok w kierunku początku programu do znacznika "Etykieta_0".
N90 ...	

Przykład 2: Skok pośredni do numeru bloku

Kod programu	Komentarz
N5 R10=100	
N10 GOTOF "N"<<R10	; Skok do bloku, którego numer znajduje się w R10.
...	
N90 ...	
N100 ...	; Cel skoku
N110 ...	
...	

Przykład 3: Skok do zmiennego celu skoku

Kod programu	Komentarz
DEF STRING[20] CEL	
CEL = "Znacznik2"	
GOTOF CEL	; Skok w kierunku końca programu do zmiennego celu skoku CEL.

Kod programu	Komentarz
Znacznik1: T="Wiertlo1"	
...	
Znacznik2: T="Wiertlo2"	; Cel skoku
...	

Przykład 4: Skok z warunkiem skoku

Kod programu	Komentarz
N40 R1=30 R2=60 R3=10 R4=11 R5=50 R6=20	; Przyporządkowanie wartości początkowych.
N41 LA1: G0 X=R2*COS(R1)+R5 Y=R2*SIN(R1)+R6	; Znacznik skoku LA1 ustawiony.
N42 R1=R1+R3 R4=R4-1	
N43 IF R4>0 GOTOB LA1	; Gdy warunek skoku jest spełniony, wówczas skok w kierunku początku programu do znacznika skoku LA1.
N44 M30	; Koniec programu

1.10.3 Rozgałęzienie programu (CASE ... OF ... DEFAULT ...)

Funkcja

Funkcja CASE stwarza możliwość sprawdzenia aktualnej wartości (typ: INT) zmiennej lub funkcji obliczeniowej i wykonania skoku do różnych miejsc w programie zależności od wartości wyniku.

Składnia

CASE(<wyrażenie>) OF <stała_1> GOTOF <cel_skoku_1> <stała_2> GOTOF <cel_skoku_2> ... DEFAULT GOTOF <cel skoku_n>

Znaczenie

CASE:	Instrukcja skoku
<wyrażenie>:	Zmienna lub funkcja obliczeniowa
OF:	Słowo kluczowe do formułowania warunkowych rozgałęzień programu
<stała_1>:	Pierwsza podana stała wartość dla zmiennej lub funkcji obliczeniowej Typ: INT
<stała_2>:	Druga podana stała wartość dla zmiennej lub funkcji obliczeniowej Typ: INT

DEFAULT:	Dla przypadków, w których zmienna albo funkcja obliczeniowa nie przyjmuje żadnej z podanych stałych wartości, można przy pomocy instrukcji DEFAULT określić cel skoku. Wskazówka: W przypadku gdy instrukcja DEFAULT nie jest zaprogramowana, celem skoku jest blok następujący po instrukcji CASE.
GOTOF:	Instrukcja skoku z celem w kierunku końca programu. Zamiast GOTOF można programować również wszystkie inne polecenia GOTO (patrz temat "Skoki w programie do znaczników skoku").
<cel_skoku_1>:	Na ten cel skoku następuje rozgałęzienie, gdy wartość zmiennej albo funkcji obliczeniowej odpowiada pierwszej podanej stałej. Cel skoku może zostać podany następująco: <Znacznik skoku>: Celem skoku jest umieszczony w programie znacznik skoku o nazwie zdefiniowanej przez użytkownika: <Znacznik skoku>: <Numer bloku>: Celem skoku jest numer bloku głównego lub pomocniczego (np. 200, N300) Zmienna typu STRING: Zmienny cel skoku. Zmienna oznacza znacznik skoku albo numer bloku.
<cel_skoku_2>:	Na ten cel skoku następuje rozgałęzienie, gdy wartość zmiennej albo funkcji obliczeniowej odpowiada drugiej podanej stałej.
<cel_skoku_n>:	Na ten cel skoku następuje rozgałęzienie, gdy wartość zmiennej nie odpowiada żadnej z podanych wartości stałych.

Przykład

Kod programu

```
...  
N20 DEF INT VAR1 VAR2 VAR3  
N30 CASE (VAR1+VAR2-VAR3) OF 7 GOTOF Etykieta_1 9 GOTOF Etykieta_2 DEFAULT GOTOF Etykieta_3  
N40 Etykieta_1: G0 X1 Y1  
N50 Etykieta_2: G0 X2 Y2  
N60 Etykieta_3: G0 X3 Y3  
...
```

Instrukcja CASE z N30 definiuje następujące możliwości rozgałęziania programów:

1. Gdy wartość funkcji obliczeniowej $VAR1+VAR2-VAR3 = 7$, wówczas przeskocz do bloku z definicją znacznika skoku "Etykieta_1" (→ N40).
2. Gdy wartość funkcji obliczeniowej $VAR1+VAR2-VAR3 = 9$, wówczas przeskocz do bloku z definicją znacznika skoku "Etykieta_2" (→ N50).
3. Gdy wartość funkcji obliczeniowej $VAR1+VAR2-VAR3$ nie wynosi ani 7 ani 9, wówczas przeskocz do bloku z definicją znacznika skoku "Etykieta_3" (→ N60).

1.11 Powtórzenie programu (REPEAT, REPEATB, ENDLABEL, P)

Funkcja

Powtarzanie części programu umożliwia powtórzenie już napisanych części programu w ramach jednego programu w dowolnym układzie.

Będące do powtórzenia wiersze wzgl. części programu są zaznaczane znacznikami skoku (etykietami).

Wskazówka

Znaczniki skoku (etykiety)

Znaczniki skoku zawsze znajdują się na początku bloku. Gdy jest numer bloku, znacznik skoku znajduje się bezpośrednio za tym numerem.

Dla nazywania znaczników skoku obowiązują następujące zasady:

- Maksymalna liczba znaków:
 - Co najmniej 2
 - Co najwyżej 32
 - Dopuszczalnymi znakami są:
 - Litery
 - Cyfry
 - Podkreślenia
 - Pierwsze dwa znaki muszą być literami albo podkreśleniami.
 - Po nazwie znacznika skoku następuje dwukropek (":").
-

Składnia

1. Powtórzenie pojedynczego wiersza programu:

```
<Znacznik skoku>: ...  
...  
REPEATB <znacznik skoku> P=<n>  
...
```

2. Powtórzenie części programu między znacznikiem skoku i instrukcją REPEAT:

```
<Znacznik skoku>: ...  
...  
REPEAT <znacznik skoku> P=<n>  
...
```


3. Powtórzenie części między dwoma znacznikami skoku:

```

<Znacznik skoku>: ...
...
<Końcowy znacznik skoku>: ...
...
REPEAT <startowy znacznik skoku> <końcowy znacznik skoku> P=<n>
...

```

Wskazówka

Wzięcie w nawias instrukcji REPEAT z obydwojema znacznikami skoku jest niemożliwe. Jeżeli <startowy znacznik skoku> zostanie znaleziony przed instrukcją REPEAT, a dojdzie do <końcowego znacznika skoku> nie nastąpi przed instrukcją REPEAT, wówczas jest przeprowadzane powtórzenie między <startowym znacznikiem skoku> i instrukcją REPEAT.

4. Powtórzenie części między znacznikiem skoku i ETYKIETĄ KOŃCOWĄ:

```

<Znacznik skoku>: ...
...
ENDLABEL: ...
...
REPEAT <znacznik skoku> P=<n>
...

```

Wskazówka

Wzięcie w nawias instrukcji REPEAT ze <znacznikiem skoku> i ETYKIETĄ KOŃCOWĄ jest niemożliwe. Jeżeli <startowy znacznik skoku> zostanie znaleziony przed instrukcją REPEAT, a dojdzie do <ETYKIETY KOŃCOWEJ> nie nastąpi przed instrukcją REPEAT, wówczas jest przeprowadzane powtórzenie między <znacznikiem skoku> i instrukcją REPEAT.

Znaczenie

REPEATB:	Polecenie do powtórzenia wiersza programu
REPEAT:	Polecenie do powtórzenia części programu
<Znacznik skoku>:	<Znacznik skoku> oznacza: <ul style="list-style-type: none"> • będący do powtórzenia wiersz programu (w przypadku REPEATB) wzgl. • początek będącej do powtórzenia części programu (w przypadku REPEAT)

1.11 Powtórzenie programu (REPEAT, REPEATB, ENDLABEL, P)

Wiersz programu oznaczony przy pomocy <znacznika skoku> może znajdować się przed lub za instrukcją REPEAT/REPEATB. Szukanie następuje najpierw w kierunku początku programu. Gdy etykieta nie zostanie w tym kierunku znaleziona, wówczas szukanie następuje w kierunku końca programu

Wyjątek:

Gdy ma zostać powtórzona część programu między znacznikiem skoku i instrukcją REPEAT (patrz 2. pod "Składnia"), wówczas wiersz programu oznaczony przez <znacznik skoku> musi znajdować się **przed** instrukcją REPEAT, ponieważ w tym przypadku szukanie odbywa się **tylko** w kierunku do początku programu.

Jeżeli blok ze <znacznikiem skoku> zawiera dalsze instrukcje, wówczas są one ponownie wykonywane przy każdym powtórzeniu.

ENDLABEL:

Słowo kluczowe, które zaznacza koniec będącej do powtórzenia części programu

Jeżeli blok z ENDLABEL zawiera dalsze instrukcje, wówczas są one ponownie wykonywane przy każdym powtórzeniu.

ENDLABEL może być wielokrotnie powtarzana w programie.

P:

Adres do podania liczby powtórzeń

<n>:

Liczba powtórzeń części programu

Typ: INT

Będąca do powtarzania część programu jest powtarzana <n> razy. Po ostatnim powtórzeniu program jest kontynuowany od wiersza następującego po wierszu z REPEAT/REPEATB.

Wskazówka:

Jeżeli nie podano P=<n>, będąca do powtórzenia część programu jest powtarzana dokładnie jeden raz.

Przykłady

Przykład 1: Powtórzenie pojedynczego wiersza programu

Kod programu	Komentarz
N10 POSITION1: X10 Y20	
N20 POSITION2: CYCLE(0,,9,8)	; Cykl pozycji
N30 ...	
N40 REPEATB POSITION1 P=5	; Wykonaj BLOK N10 pięć razy.
N50 REPEATB POSITION2	; Wykonaj blok N20 jeden raz.
N60 ...	
N70 M30	

Przykład 2: Powtórzenie części programu między znacznikiem skoku i instrukcją REPEAT

Kod programu	Komentarz
N5 R10=15	
N10 Begin: R10=R10+1	; Szerokość
N20 Z=10-R10	
N30 G1 X=R10 F200	
N40 Y=R10	
N50 X=-R10	
N60 Y=-R10	
N70 Z=10+R10	
N80 REPEAT BEGIN P=4	; Wykonaj cztery razy zakres N10 do N70.
N90 Z10	
N100 M30	

Przykład 3: Powtórzenie części między dwoma znacznikami skoku

Kod programu	Komentarz
N5 R10=15	
N10 Begin: R10=R10+1	; Szerokość
N20 Z=10-R10	
N30 G1 X=R10 F200	
N40 Y=R10	
N50 X=-R10	
N60 Y=-R10	
N70 END: Z=10	
N80 Z10	
N90 CYCLE(10,20,30)	
N100 REPEAT BEGIN END P=3	; Wykonaj trzy razy zakres N10 do N70.
N110 Z10	
N120 M30	

Przykład 4: Powtórzenie części między znacznikiem skoku i ETYKIETĄ KOŃCOWĄ

Kod programu	Komentarz
N10 G1 F300 Z-10	
N20 BEGIN1:	
N30 X10	
N40 Y10	
N50 BEGIN2:	
N60 X20	
N70 Y30	

Kod programu	Komentarz
N80 ETYKIETA KONCOWA:Z10	
N90 X0 Y0 Z0	
N100 Z-10	
N110 BEGIN3: X20	
N120 Y30	
N130 REPEAT BEGIN3 P=3	; Wykonaj trzy razy zakres N110 do N120.
N140 REPEAT BEGIN2 P=2	; Wykonaj dwa razy zakres N50 do N80.
N150 M100	
N160 REPEAT BEGIN1 P=2	; Wykonaj dwa razy zakres N20 do N80.
N170 Z10	
N180 X0 Y0	
N190 M30	

Przykład 5: Obróbka frezarska, obróbka w pozycjach wiercenia przy pomocy różnych technologii

Kod programu	Komentarz
N10 NAWIERTAK()	; Wprowadzenie nawiertaka do pozycji roboczej.
N20 POS_1:	; Pozycje wiercenia 1
N30 X1 Y1	
N40 X2	
N50 Y2	
N60 X3 Y3	
N70 ENDLABEL:	
N80 POS_2:	; Pozycje wiercenia 2
N90 X10 Y5	
N100 X9 Y-5	
N110 X3 Y3	
N120 ENDLABEL:	
N130 WIERTLO()	; Wprowadzenie wiertła do pozycji roboczej i cykl wiercenia.
N140 GWINT(6)	; Wprowadzenie do pozycji roboczej gwintownika M6 i cykl gwintowania.
N150 REPEAT POS_1	; Powtórz segment programu od POS_1 jeden raz do ENDLABEL.
N160 WIERTLO()	; Wprowadzenie wiertła do pozycji roboczej i cykl wiercenia.
N170 GWINT(8)	; Wprowadzenie do pozycji roboczej gwintownika M8 i cykl gwintowania.
N180 REPEAT POS_2	; Powtórz segment programu od POS_2 jeden raz do ENDLABEL.
N190 M30	

Dalsze informacje

- Powtarzanie części programu można wywoływać w sposób kaskadowy. Każde wywołanie zajmuje jeden poziom podprogramu.
- Jeżeli podczas wykonywania powtarzania części programu jest zaprogramowane M17 lub RET, wówczas powtarzanie części programu jest przerywane. Program jest kontynuowany od bloku następującym po wierszu REPEAT.
- W aktualnym wyświetleniu programu powtórzenie części programu jest wyświetlane jako oddzielny poziom podprogramu.
- Jeżeli podczas wykonywania części programu zostanie wyzwolone anulowanie poziomu, wówczas program jest kontynuowany po wywołaniu wykonywania części programu.

Przykład:

Kod programu	Komentarz
N5 R10=15	
N10 BEGIN: R10=R10+1	; Szerokość
N20 Z=10-R10	
N30 G1 X=R10 F200	
N40 Y=R10	; Anulowanie poziomu
N50 X=-R10	
N60 Y=-R10	
N70 END: Z10	
N80 Z10	
N90 CYCLE(10,20,30)	
N100 REPEAT BEGIN END P=3	
N120 Z10	; Kontynuowanie wykonywania programu.
N130 M30	

- Struktur kontrolnych i powtarzania części programu można używać w sposób kombinowany. Nie powinno być jednak krzyżowania się. Powtórzenie części programu powinno leżeć w ramach jednej gałęzi struktury kontrolnej wzgl. jedna struktura kontrolna w ramach jednego powtórzenia części programu.
- Przy pomieszczeniu skoków i powtórzenia części programu bloki są wykonywane czysto sekwencyjnie. Jeżeli np. skok następuje z powtórzenia programu, wówczas wykonywanie następuje tak długo, aż zostanie znaleziony koniec części programu.

Przykład:

Kod programu
N10 G1 F300 Z-10
N20 BEGIN1:
N30 X=10
N40 Y=10
N50 GOTOF BEGIN2
N60 ENDLABEL:
N70 BEGIN2:

```
Kod programu
N80 X20
N90 Y30
N100 ENDLABEL: Z10
N110 X0 Y0 Z0
N120 Z-10
N130 REPEAT BEGIN1 P=2
N140 Z10
N150 X0 Y0
N160 M30
```

Wskazówka

Instrukcja REPEAT powinna znajdować się za blokami ruchu.

1.12 Struktury kontrolne

Funkcja

Sterowanie standardowo wykonuje bloki NC w zaprogramowanej kolejności.

Ta kolejność może być zmieniana przez programowanie alternatywnych bloków programu i pętli programowych. Programowanie struktur kontroli następuje za pomocą słów kluczowych IF, ELSE, ENDIF, LOOP, FOR, WHILE oraz REPEAT.

UWAGA

Błąd programowania

Struktury kontrolne są możliwe tylko w ramach części instrukcyjnej programu. Definicje w nagłówku programu nie mogą być powtarzane warunkowo albo powtarzalnie.

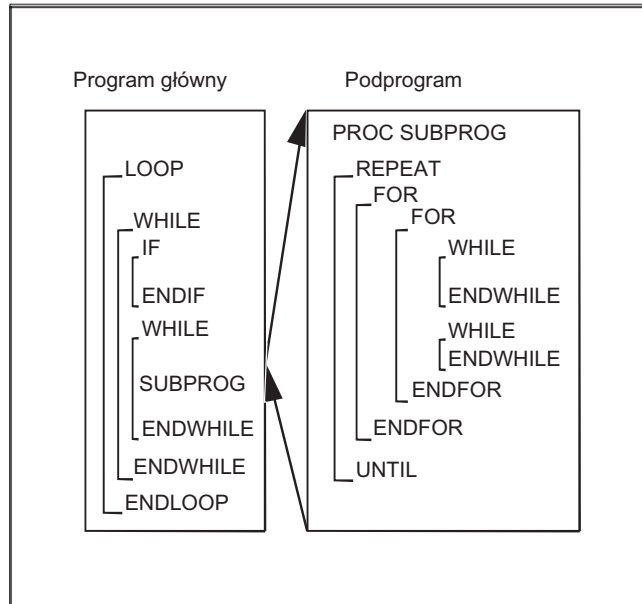
Na słowa kluczowe dla struktur kontrolnych nie wolno, tak samo jak celów skoku, nakładać makropoleci. Sprawdzenie w przypadku definicji makra nie następuje.

Działanie

Struktura kontroli nie może być stosowana w rozszerzeniu na inne programy.

Głębokość kaskadowania

W ramach każdej płaszczyzny podprogramu jest możliwa głębokość kaskadowania wynosząca do 16 struktur kontrolnych.



Zachowanie się pod względem czasu przebiegu

W standardowo aktywnej pracy interpretera można poprzez zastosowanie skoków w programie uzyskać szybszy przebieg programu, niż ze strukturami kontrolnymi.

W cyklach prekompilowanych nie ma żadnej różnicy między skokami w programie i strukturami kontrolnymi.

Aktualne wyświetlenie bloku przy pętlach w programie

Jeżeli podczas pętli w programie wykonywane są tylko bloki przebiegu wyprzedzającego, na wyświetlaczu bloku wyświetli się ostatni **przed** pętlą w programie blok przebiegu głównego.

Aby np. celu diagnozy na wyświetlaczu bloku widoczne były również wykonywane bloki przebiegu wyprzedzającego, musi zistać aktywowany dekodowany pojedynczy blok SBL2.

Literatura

Podręcznik działania Funkcje podstawowe, Rozdział: BAG, kanał, tryb pracy programu, zachowanie się przy zresetowaniu (K1) > blok pojedynczy > dekodowany blok pojedynczy SBL2 z domyślnym zatrzymaniem przebiegu wyprzedzającego

Warunki brzegowe

- Bloki z elementami struktur kontrolnych nie mogą zostać ukryte.
- Znaczniki skoku (etykiety) są niedozwolone w blokach z elementami struktur kontrolnych.
- Struktury kontrolne są wykonywane interpretacyjnie. Przy rozpoznaniu końca pętli jest przy uwzględnieniu znalezionych przy tym struktur kontrolnych szukany początek pętli. Dlatego w pracy interpretera struktura bloków programu nie jest kompletnie sprawdzana.
- Zasadniczo zaleca się nie stosowanie w sposób mieszany struktur kontrolnych i skoków w programie.
- Przy wyprzedzającym przetwarzaniu cykli może zostać sprawdzone prawidłowe kaskadowanie struktur kontrolnych.

1.12.1 Warunkowa instrukcja i rozgałęzienie (IF, ELSE, ENDIF)

Funkcja

Instrukcja warunkowa: IF - blok programu - ENDIF

W przypadku instrukcji warunkowej, jeśli warunek jest spełniony, zostanie wykonany blok programu znajdujący się pomiędzy `IF` oraz `ENDIF`.

Rozgałęzienie: IF - blok programu_1 - ELSE - blok programu_2 - ENDIF

Przy rozgałęzieniu wuy

Jeżeli warunek jest spełniony, blok programu_1 znajdujący się pomiędzy `IF` und `ELSE` zostanie wykonany.

Jeżeli warunek **nie** jest spełniony, zostanie wykonany blok programu_2 znajdujący się pomiędzy `ELSE` und `ENDIF`.

Składnia

Instrukcja warunkowa

```
IF <warunek>
Blok programu                ; Wykonanie przy: <warunek> == TRUE
ENDIF
```

Rozgałęzienie

```
IF <warunek>
  Blok programu_1            ; Wykonanie przy: <warunek> == TRUE
ELSE
  Blok programu_2            ; Wykonanie przy: <warunek> == FALSE
ENDIF
```


Znaczenie

IF:	Rozpoczyna instrukcję warunkową lub rozgałęzienie.
ELSE:	Rozpoczyna alternatywny blok programu.
ENDIF:	Zaznacza koniec instrukcji warunkowej lub rozgałęzienia.
<warunek>:	Wyrażenie logiczne, którego ewaluacja wykazuje TRUE lub FALSE.

Przykład: Podprogram zmiany narzędzia

Kod programu	Komentarz
PROC L6	Procedura zmiany narzędzia
N500 DEF INT TNR_AKTUALNE	Zmienna dla aktywnego numeru T
N510 DEF INT TNR_PRESELEKCJA	Zmienna dla wstępnie wybranego numeru T
	Określenie aktualnego narzędzia
N520 STOPRE	
N530 IF \$P_ISTEST	W trybie testu programu ...
N540 TNR_AKTUALNE = \$P_TOOLNO	... jest czytane "aktualne" narzędzie.
N550 ELSE	W przeciwnym przypadku ...
N560 TNR_AKTUALNE = \$TC_MPP6[9998,1]	... jest odczytywane narzędzie wrzeczona.
N570 ENDIF	
N580 GETSELT(TNR_PRESELEKCJA)	Odczyt numeru T wstępnie wybranego narzędzia we wrzeczonie.
N590 IF TNR_AKTUALNY <> TNR_PRESELEKCJA	Gdy wstępnie wybrane narzędzie nie jest jeszcze aktualnym narzędziem, wówczas ...
N600 G0 G40 G60 G90 SUPA X450 Y300 Z300 D0	... ruch do punktu zmiany narzędzia ...
N610 M206	... i wykonanie wymiany narzędzia.
N620 ENDIF	
N630 M17	

1.12.2 Pętla programowa bez końca (LOOP, ENDLOOP)

Funkcja

Pętla bez końca znajduje zastosowanie w programach bez końca. Na końcu pętli następuje zawsze przeskok do jej początku.

Składnia

```
LOOP  
...  
ENDLOOP
```

Znaczenie

LOOP: Rozpoczyna pętlę bez końca.
ENDLOOP: Zaznacza koniec pętli i powoduje powrót do jej początku.

Przykład

```
Kod programu  
...  
LOOP  
MSG ("Żadne ostrze narzędzia nie jest aktywne")  
M0  
STOPRE  
ENDLOOP  
...
```

1.12.3 Pętla FOR (FOR ... TO ..., ENDFOR)

Funkcja

Pętla FOR jest stosowana, gdy przebieg pracy ma być powtarzany ze stałą liczbą przebiegów.

Składnia

```
FOR <zmienna> = <wartość początkowa> TO <wartość końcowa>  
...  
ENDFOR
```

Znaczenie

FOR:	Rozpoczyna pętlę FOR.
ENDFOR:	Zaznacza koniec pętli i powoduje skok powrotny do jej początku, jak długo wartość końcowa liczenia nie jest jeszcze osiągnięta.
<zmienna>:	Zmienna sterująca, która jest zwiększana od wartości początkowej do końcowej i przy każdym przebiegu zwiększa się o wartość "1". Typ INT lub REAL Wskazówka: Typ REAL jest wybierany, gdy np. są programowane parametry R dla pętli FOR. Gdy zmienna sterująca jest typu REAL, jej wartość jest zaokrąglana do wartości całkowitoliczbowej.
<wartość początkowa>:	Wartość początkowa liczenia Warunek: Wartość początkowa musi być mniejsza, niż wartość końcowa.
<Wartość końcowa>:	Wartość końcowa liczenia

Przykłady

Przykład 1: Zmienna INTEGER lub parametr R, jako zmienna sterująca

Zmienna INTEGER jako zmienna sterująca:

Kod programu	Komentarz
DEF INT iVARIABLE1	
R10=R12-R20*R1 R11=6	
FOR iVARIABLE1= R10 TO R11	; Zmienna sterująca = zmienna INTEGER
R20=R21*R22+R33	
ENDFOR	
M30	

Parametr R jako zmienna sterująca:

Kod programu	Komentarz
R11=6	
FOR R10=R12-R20*R1 TO R11	; Zmienna sterująca = parametr R (zmienna real)
R20=R21*R22+R33	
ENDFOR	
M30	

Przykład 2: Wykonanie stałej liczby części

Kod programu	Komentarz
DEF INT LICZBA_SZTUK	; Definiuje zmienną typu INT o nazwie "LICZBA_SZTUK"
FOR LICZBA_SZTUK = 0 TO 100	; Rozpoczyna pętlę FOR. Zmienna "LICZBA_SZTUK" jest inkrementowana od wartości początkowej "0" do wartości końcowej "100".
G01 ...	
ENDFOR	; Koniec pętli FOR.
M30	

1.12.4 Pętla programowa z warunkiem na początku pętli (WHILE, ENDWHILE)

Funkcja

W przypadku pętli WHILE warunek znajduje się na jej początku. Jak długo warunek jest spełniony, pętla WHILE będzie wykonywana.

Składnia

```
WHILE <warunek>
...
ENDWHILE
```

Znaczenie

WHILE: Rozpoczyna pętlę programową.
 ENDWHILE: Zaznacza koniec pętli i powoduje powrót do jej początku.
 <warunek>: Warunek, który musi być spełniony, aby pętla WHILE była wykonywana.

Przykład

Kod programu	Komentarz
...	
WHILE \$AA_IW[OŚ WIERCENIA] > -10	; Wywołanie pętli WHILE pod następującym warunkiem: aktualna wartość zadana w WKS dla osi wiercenia musi być większa, niż -10.
G1 G91 F250 AX[OŚ WIERCENIA] = -1	
ENDWHILE	
...	

1.12.5 Pętla programowa z warunkiem na końcu (REPEAT, UNTIL)

Funkcja

W przypadku pętli REPEAT warunek znajduje się na jej końcu. Pętla REPEAT jest wykonywana jeden raz i powtarzana tak długo, aż warunek zostanie spełniony.

Składnia

```
REPEAT
...
UNTIL <warunek>
```

Znaczenie

REPEAT:	Rozpoczyna pętlę programową.
UNTIL:	Zaznacza koniec pętli i powoduje powrót do jej początku.
<warunek>:	Warunek, który musi być spełniony, aby pętla REPEAT nie była już wykonywana.

Przykład

Kod programu	Komentarz
...	
REPEAT	; Wywołanie pętli REPEAT.
...	
UNTIL ...	; Sprawdzenie, czy warunek jest spełniony.
...	

1.12.6 Przykład programu z kaskadowanymi strukturami kontrolnymi

Kod programu	Komentarz
LOOP	
IF NOT \$P_SEARCH	; IF bez szukania bloku
G1 G90 X0 Z10 F1000	
WHILE \$AA_IM[X] <= 100	; WHILE (wartość zadana osi X<=100)
G1 G91 X10 F500	; Układ otworów
Z-5 F100	
Z5	
ENDWHILE	

Kod programu	Komentarz
ELSE	; ELSE Szukanie bloku
MSG ("W trakcie szukania wiercenie nie następuje")	
ENDIF	; ENDIF
\$A_OUT[1] = 1	; następna płyta wiertarska
G4 F2	
ENDLOOP	
M30	

1.13 Koordynacja programu (INIT, START, WAITM, WAITMC, WAITE, SETM, CLEARM)

Funkcja

Kanał może wykonywać własny program, niezależnie od innych kanałów. Dzięki temu przyporządkowane mu czasowo osie i wrzeciona mogą być sterowane przez program.

Jeżeli w obróbce przedmiotu uczestniczy wiele kanałów, wówczas może być wymagana synchronizacja przebiegów programów. Dla tej koordynacji programów są specjalne instrukcje (polecenia).

Wskazówka

Koordynacja programu jest możliwa również we własnym kanale.

Warunek

Uczestniczące kanały muszą należeć do **tej samej grupy trybów pracy (BAG)**.

Składnia

```
INIT(<nr kanału>,<ścieżka>,<tryb kwitowania>)
START(<nr kanału>,<nr kanału>,...)
WAITM(<nr znacznika>,<nr kanału>,<nr kanału>,...)
WAITMC(<nr znacznika>,<nr kanału>,<nr kanału>,...)
WAITE(<nr kanału>,<nr kanału>,...)
SETM(<nr znacznika>,<nr znacznika>,...)
SETM(<nr znacznika>,<nr znacznika>,...)
```

Wskazówka

Instrukcje do koordynacji programu muszą znajdować się w oddzielnych blokach.

Znaczenie

INIT:	Predefiniowana procedura do wyboru programu NC, który ma zostać wykonany w podanym kanale
<nr kanału>:	Numer kanału Typ: INT
<ścieżka>:	Absolutna lub względna ścieżka do programu NC Typ: STRING Absolutne podanie ścieżki: Ścieżka absolutna musi zostać utworzona według następującego wzorca: aktualny katalog/_N_<nazwa>_MPF <ul style="list-style-type: none"> "aktualny katalog" oznacza wybrany katalog obrabianego przedmiotu albo katalog standardowy /_N_MPF_DIR. Należy podać kompletną nazwę programu. Względne podanie ścieżki: Przy względnym podaniu ścieżki obowiązują te same zasady jak dla wywołań podprogramów.
<tryb kwitowania>:	Parametr typu CHAR Wartości: "N" bez kwitowania Wykonywanie programu jest kontynuowane po wysłaniu rozkazu. Nadawca nie jest powiadamiany, gdy rozkazu nie można pomyślnie wykonać. "S" kwitowanie synchroniczne Wykonywanie programu jest tak długo zatrzymywane, aż komponent odbierający pokwituje rozkaz. Przy pozytywnym pokwitowaniu jest wykonywane następne polecenie. Przy ujemnym pokwitowaniu jest wprowadzany komunikat błędu. Wskazówka: Gdy tryb kwitowania nie zostanie podany, wówczas następuje kwitowanie synchroniczne.
START:	Predefiniowana procedura do wystartowania wybranych programów w innych kanałach
<nr kanału>, ...:	Wykaz numerów kanałów Typ: INT

WAITM:	<p>Predefiniowana procedura dla czekania na dojście do znacznika w podanych kanałach</p> <p>We własnym kanale podany znacznik jest ustawiany przez WAITM. Blok poprzedzający ulega zakończeniu z zatrzymaniem dokładnym. Znacznik jest kasowany po synchronizacji.</p> <p>Równocześnie można ustawić maks. 10 znaczników na kanał.</p> <p><nr znacznika>: Numer znacznika Typ: INT</p> <p><nr kanału>, ...: Wykaz numerów kanałów (własny kanał nie musi zostać podany) Typ: INT</p>
WAITE:	<p>Predefiniowana procedura dla czekania na koniec programu w jednym albo wielu innych kanałach</p> <p><nr kanału>, ...: Wykaz numerów kanałów Typ: INT</p>
WAITMC:	<p>Predefiniowana procedura dla czekania na dojście do znacznika w podanych kanałach</p> <p>W przeciwieństwie do WAITM dokładne zatrzymanie następuje tylko wtedy, gdy inne kanały jeszcze nie doszły do znacznika.</p> <p>Parametr jak w przypadku WAITM.</p>
SETM:	<p>Predefiniowana procedura dla ustawienia jednego lub wielu znaczników do koordynacji kanałów</p> <p>Nie wpływa to na wykonywanie we własnym kanale.</p> <p>SETM zachowuje obowiązywanie po Reset i NC-Start.</p> <p><nr znacznika>, ...: Wykaz numerów znaczników</p>
CLEARM:	<p>Predefiniowana procedura do kasowania jednego lub wielu znaczników do koordynacji kanałów</p> <p>Nie wpływa to na wykonywanie we własnym kanale.</p> <p>Wszystkie znaczniki w kanale można kasować przy pomocy CLEARM().</p> <p>CLEARM (0) kasuje znacznik "0".</p> <p>CLEARM zachowuje obowiązywanie po Reset i NC-Start.</p> <p><nr znacznika>, ...: Wykaz numerów znaczników</p>

Wskazówka

Numer kanału

Nazwy kanałów muszą poprzez zmienne być konwertowane na numery.

 **OSTROŻNIE**

Numer kanału

Przyporządkowanie numerów należy zabezpieczyć przed lekkomyślną zmianą.

Wskazówka**Nazwa kanału**

Zamiast numerów kanałów można również programować nazwy kanałów (identyfikator albo słowo kluczowe, typ: STRING) zdefiniowane poprzez \$MC_CHAN_NAME).

! OSTROŻNIE**Nazwa kanału**

Nazw nie może być już w NC w takim znaczeniu, jak np. słowo kluczowe, polecenie językowe, nazwa osi itd.

Wskazówka

Do wymiany danych między programami mogą być używane zmienne, którymi kanały wspólnie dysponują (zmienne globalne specyficzne dla NCK). Poza tym sporządzanie programów następuje oddzielnie dla każdego z kanałów.

Przykłady**Przykład 1: Nadanie numerów kanałów**

Kanał o nazwie "MASZYNA" ma otrzymać numer 1, kanał o nazwie "MANIPULATOR" ma otrzymać numer 2. Zmienne otrzymają tą samą nazwę co kanały:

```
DEF INT MASZYNA=1, MANIPULATOR=2
```

Dzięki temu na przykład instrukcja `START` brzmi:
`START (MASZYNA)`

Przykład 2: Możliwości podawania kanałów

```
$MC_CHAN_NAME[0]= "CHAN_X" ; nazwa 1. kanału
```

```
$MC_CHAN_NAME[1]= "CHAN_Y" ; nazwa 2. kanału
```

Programowanie przy pomocy:

- Numery kanałów:

Kod programu	Komentarz
START(1,2)	; Wykonanie startu w 1. i 2. kanale.

- Identyfikatory kanałów:

Kod programu	Komentarz
START(CHAN_X, CHAN_Y)	; Wykonanie startu w 1. i 2. kanale. ; Identyfikatory "kanal_X" i "kanal_Y" reprezentują na podstawie danej maszynowej \$MC_CHAN_NAME wewnętrznie numer kanału 1 i 2. Dlatego wykonują one również start

Kod programu	Komentarz
	w 1. i 2. kanale

- Zmienne integer:

Kod programu	Komentarz
DEF INT chanNo1, chanNo2	; Definicja zmiennych.
chanNo1=CHAN_X chanNo2=CHAN_Y	
START(chanNo1, chanNo2)	; Wykonanie startu w 1. i 2. kanale.

Przykład 3: Polecenie INIT z absolutnym podaniem ścieżki

Kod programu	Komentarz
N10 INIT(2, "/_N_WKS_DIR/_N_WALEK1_WPD/_N_ABSPAN1_MPF")	

Przykład 4: polecenie INIT ze względnym podaniem ścieżki

Kod programu	Komentarz
N10 INIT(2, "MYPROG")	; Wybór programu /_N_MPF_DIR/_N_MYPROG_MPF w kanale 2.

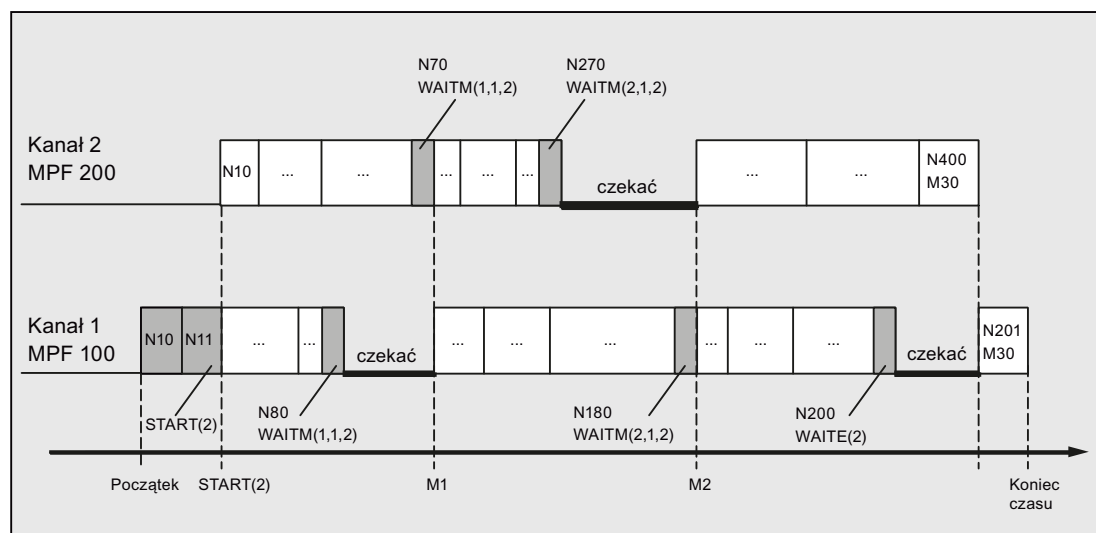
Przykład 5: Koordynowanie programu przy pomocy WAITM

Kanał 1: Program /_N_MPF_DIR/_N_MPF100_MPF jest wybrany.

Kod programu	Komentarz
N10 INIT(2, "MPF200", "N")	
N11 START(2)	
...	; Obróbka w kanale 1
N80 WAITM(1,1,2)	; Czekanie na dojście do znacznika czekania 1 w kanałach 1 i 2.
...	; Dalsze wykonywanie w kanale 1.
N180 WAITM(2,1,2)	; Czekanie na dojście do znacznika czekania 2 w kanałach 1 i 2.
...	; Dalsze wykonywanie w kanale 1.
N200 WAITE(2)	; Czekanie na koniec programu w kanale 2.
N201 M30	; Koniec programu kanał 1, koniec całości.

Kanał 2: Przy pomocy polecenia INIT (patrz N10 w _N_MPF100_MPF) program _N_MPF200_MPF jest wybierany do wykonania w kanale 2.

Kod programu	Komentarz
;\$PATH=/_N_MPF_DIR	
...	; Obróbka w kanale 2
N70 WAITM(1,1,2)	; Czekanie na dojście do znacznika czekania 1 w kanałach 1 i 2.
...	; Dalsze wykonywanie w kanale 2.
N270 WAITM(2,1,2)	; Czekanie na dojście do znacznika czekania 2 w kanałach 1 i 2.
...	; Dalsze wykonywanie w kanale 2.
N400 M30	; Koniec programu kanału 2.



1.14 Procedura przerwania (ASUP)

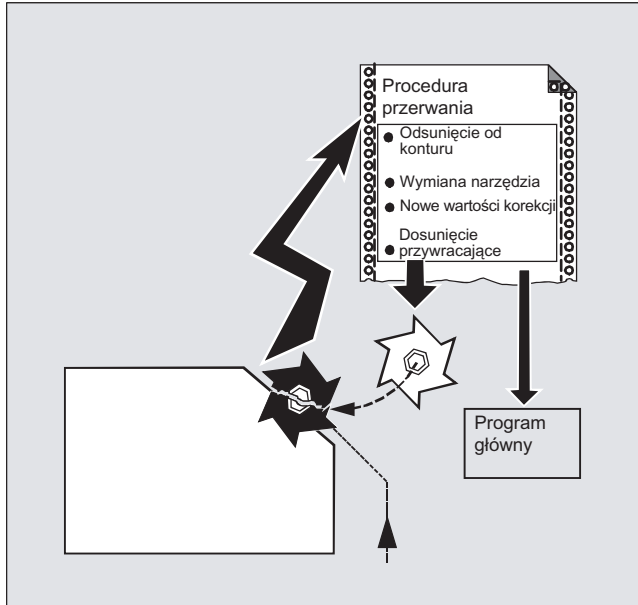
1.14.1 Funkcja procedury przerwania

Wskazówka

Występujące w poniższym opisie na przemian pojęcia "Podprogram asynchroniczny (ASUP)" i "Procedura przerwania" oznaczają tą samą funkcjonalność.

Funkcja

Wyłumaczymy funkcję procedury przerwania na podstawie typowego przykładu:



Podczas obróbki następuje pęknięcie narzędzia. Przez to jest wyzwalany sygnał, który zatrzymuje bieżący przebieg obróbki i równocześnie uruchamia podprogram - tak zwaną procedurę przerwania. W tym podprogramie znajdują się wszystkie instrukcje, które w takim przypadku mają zostać wykonane.

Gdy podprogram jest wykonany (i przez to jest przywrócona gotowość do pracy), sterowanie przeskakuje z powrotem do programu głównego i kontynuuje obróbkę od miejsca przerwania - zależnie od polecenia `REPOS` - patrz "Dosunięcie przywracające do konturu (Strona 483)".

OSTROŻNIE

Niebezpieczeństwo kolizji

Gdy w podprogramie nie jest zaprogramowane polecenie `REPOS`, wówczas pozycjonowanie następuje na koniec bloku, po którym następuje blok przerwany.

Literatura

Podręcznik działania Funkcje podstawowe; BAG, Kanał, Praca programowa, Zachowanie się przy zresetowaniu (K1), punkt: "Asynchroniczne podprogramy (ASUPs), rutyny przerwania"

1.14.2 Sporządzenie procedury przerwania

Sporządzenie procedury przerwania jako podprogramu

Procedura przerwania jest oznakowywana przy definicji jak podprogram.

Przykład:

Kod programu	Komentarz
PROC ABHEB_Z	; Nazwa programu "ABHEB_Z"
N10 ...	; Następnie następują bloki NC.
...	
N50 M17	; Na zakończenie koniec programu i powrót do programu głównego.

Zapisywanie modalnych funkcji G (SAVE)

Procedura przerwania może w definicji zostać oznaczona przez `SAVE`.

Atrybut `SAVE` powoduje, że aktywne przed wywołaniem procedury przerwania modalne funkcje G są zapisywane, a po zakończeniu procedury przerwania ponownie reaktywowane (patrz "Podprogramy z mechanizmem `SAVE (SAVE)` (Strona 170)").

Przez to jest możliwe kontynuowanie obróbki w miejscu przerwania po upływie procedury przerwania.

Przykład:

Kod programu
PROC ABHEB_Z SAVE
N10 ...
...
N50 M17

Przyporządkowanie dalszych procedur przerwania (SETINT)

W ramach procedury przerwania mogą być programowane i uaktywniane przez dalsze procedury przerwania instrukcje `SETINT` (patrz "Przyporządkowanie i uruchomienie procedury przerwania (SETINT)" (Strona 126)). Wyzwolenie następuje dopiero przez wejście.

Literatura

Odnosnie dalszych informacji dot. sporządzania podprogramów patrz punkt "Technika podprogramów, technika makr".

1.14.3 Przyporządkowanie i uruchomienie procedury przerwania (SETINT, PRIO, BLSYNC)

Funkcja

Sterowanie dysponuje sygnałami (wejście 1...8), które wyzwalają przerwanie bieżącego programu i mogą wystartować odpowiednią procedurę przerwania.

Przyporządkowanie, które wejście który program uruchamia, następuje w programie obróbki przy pomocy polecenia SETINT.

W przypadku gdy w programie obróbki jest wiele instrukcji SETINT i przez to może nastąpić jednoczesny wpływ wielu sygnałów, przyporządkowanym procedurom przerwania muszą zostać przyporządkowane wartości priorytetu, które ustalają kolejność wykonywania:

PRIO=<wartość>

Jeżeli podczas wykonywania procedury dotrą nowe sygnały, wówczas procedura o wyższym priorytecie przerywa procedurę aktualną.

Składnia

```
SETINT (<n>) PRIO=<wartość> <NAZWA>
SETINT (<n>) PRIO=<wartość> <NAZWA> BLSYNC
SETINT (<n>) PRIO=<wartość> <NAZWA> LIFTFAST
```

Znaczenie

SETINT (<n>): Polecenie: Wejście <n> przyporządkować do procedury przerwania. Przyporządkowana procedura przerwania ulega uruchomieniu, gdy przełączy się wejście <n>.

Wskazówka:

Jeżeli do zajętego wejścia zostanie przyporządkowana nowa procedura, stare przyporządkowanie automatycznie przestaje obowiązywać.

<n>: Parametr: Numer wejścia
 Typ: INT
 Zakres wartości: 1 ... 8

PRIO= : Polecenie: Ustalenie priorytetu

<wartość>: Wartość priorytetu
 Typ: INT
 Zakres wartości: 1 ... 128

Priorytet 1 odpowiada najwyższemu priorytetowi.

<NAZWA>: Nazwa podprogramu (procedury przerwania), który ma zostać wykonany.

BLSYNC: Gdy instrukcja SETINT zostanie zaprogramowana razem z BLSYNC, wówczas przy przybyciu sygnału przerwania bieżący blok programu zostanie jeszcze wykonany, a dopiero potem zostanie uruchomiona procedura przerwania.

LIFTFAST: Gdy instrukcja SETINT zostanie zaprogramowana razem z LIFTFAST, wówczas przy przybyciu sygnału przerwania zostanie przed uruchomieniem procedury przerwania przeprowadzone "szybkie cofnięcie narzędzia od konturu" (patrz " Szybkie cofnięcie od konturu (SETINT LIFTFAST, ALF) (Strona 129)").

Przykłady

Przykład 1: Przyporządkowanie procedur przerywania i ustalenie priorytetu

Kod programu	Komentarz
...	
N20 SETINT(3) PRIO=1 ABHEB_Z	; Gdy wejście 3 przełączy, wówczas powinna zostać uruchomiona procedura przerywania "ABHEB_Z".
N30 SETINT(2) PRIO=2 ABHEB_X	; Gdy wejście 2 przełączy, wówczas powinna wystartować procedura przerywania "ABHEB_X".
...	

Procedury przerywania wykonywane są w kolejności wartości priorytetów, gdy sygnał na wejściach występuje równocześnie: najpierw "ABHEB_Z", następnie "ABHEB_X".

Przykład 2: Ponowne przyporządkowanie procedury przerywania

Kod programu	Komentarz
...	
N20 SETINT(3) PRIO=2 ABHEB_Z	; Gdy wejście 3 przełączy, wówczas powinna zostać uruchomiona procedura przerywania "ABHEB_Z".
...	
N120 SETINT(3) PRIO=1 ABHEB_X	; Wejściu 3 jest przyporządkowywana nowa procedura przerywania: zamiast "ABHEB_Z" powinna zostać uruchomiona "ABHEB_X", gdy wejście 3 przełączy.

1.14.4 Wyłączenie aktywności / reaktywowanie przyporządkowania procedury przerywania (DISABLE, ENABLE)

Funkcja

Aktywność instrukcji SETINT może zostać wyłączona przy pomocy DISABLE i ponownie włączona przy pomocy ENABLE, bez utraty przyporządkowania wejście → procedura przerywania.

Składnia

DISABLE (<n>
ENABLE (<n>

Znaczenie

DISABLE (<n>):	Polecenie: Wyłączenie aktywności przyporządkowania procedury przerwania od wejścia <n>
ENABLE (<n>):	Polecenie: Reaktywowanie przyporządkowania procedury przerwania od wejścia <n>
<n>:	Parametr: Numer wejścia
	Typ: INT
	Zakres wartości: 1 ... 8

Przykład

Kod programu	Komentarz
...	
N20 SETINT(3) PRIO=1 ABHEB_Z	; Gdy wejście 3 przełączy, wówczas powinna zostać uruchomiona procedura przerwania "ABHEB_Z".
...	
N90 DISABLE(3)	; Aktywność instrukcji SETINT z N20 jest wyłączana.
...	
N130 ENABLE(3)	; Instrukcja SETINT z N20 jest ponownie uaktywniana.
...	

1.14.5 Skasowanie przyporządkowania procedury przerwania (CLRINT)

Funkcja

Zdefiniowane przy pomocy SETINT przyporządkowanie wejście → procedura przerwania może zostać skasowane przy pomocy CLRINT.

Składnia

CLRINT (<n>)

Znaczenie

CLRINT (<n>):	Polecenie: Skasowanie przyporządkowania procedury przerwania od wejścia <n>
<n>:	Parametr: Numer wejścia
	Typ: INT
	Zakres wartości: 1 ... 8

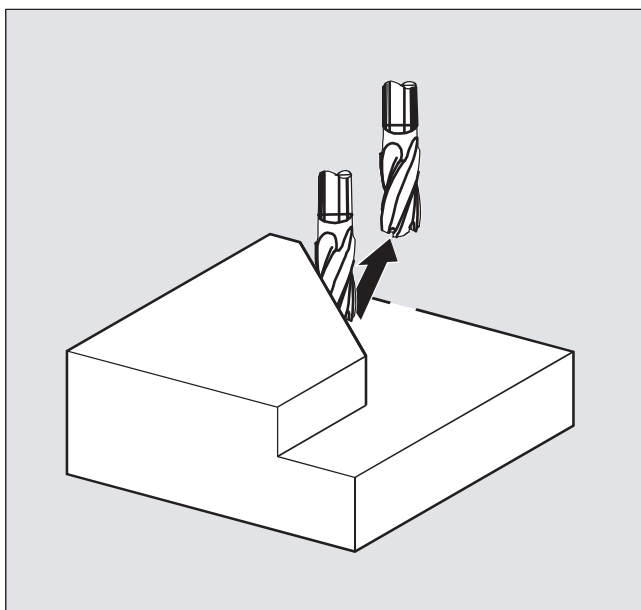
Przykład

Kod programu	Komentarz
...	
N20 SETINT(3) PRIO=2 ABHEB_Z	;
...	
N50 CLRINT(3)	; Przyporządkowanie między wejściem "3" i procedurą przerwania "ABHEB_Z" jest skasowane.
...	

1.14.6 Szybkie cofnięcie od konturu (SETINT LIFTFAST, ALF)

Funkcja

W przypadku instrukcji `SETINT` z `LIFTFAST` przy przełączeniu wejścia narzędzie jest odsuwane od konturu przez szybkie cofnięcie.



Dalszy przebieg jest zależny od tego, czy instrukcja `SETINT` oprócz `LIFTFAST` zawiera procedurę przerwania:

Z procedurą przerwania: **Po** szybkim cofnięciu jest wykonywana procedura przerwania.
Bez procedury przerwania: Po szybkim cofnięciu obróbka jest zatrzymywana z alarmem.

Składnia

```
SETINT(<n>) PRIO=1 LIFTFAST  
SETINT(<n>) PRIO=1 <NAZWA> LIFTFAST
```

Znaczenie

SETINT (<n>):	Polecenie: Wejście <n> należy przyporządkować do procedury przerwania. Przyporządkowana procedura przerwania ulega uruchomieniu, gdy przełączy się wejście <n>.
<n>:	Parametr: Numer wejścia Typ: INT Zakres wartości: 1 ... 8
PRIO= :	Ustalenie priorytetu
<wartość>:	Wartość priorytetu Zakres wartości: 1 ... 128 Priorytet 1 odpowiada najwyższemu priorytetowi.
<NAZWA>:	Nazwa podprogramu (procedury przerwania), który ma zostać wykonany.
LIFTFAST:	Polecenie: Szybkie cofnięcie od konturu
ALF=... :	Polecenie: Programowany kierunek ruchu (znajduje się w bloku ruchu) Odnosnie możliwości programowania z ALF patrz temat "Kierunek ruchu przy szybkim cofnięciu od konturu (Strona 131)".

Warunki brzegowe

Zachowanie się przy aktywnym frame z lustrzanym odbiciem

Przy określaniu kierunku odsunięcia następuje sprawdzenie, czy jest aktywny frame zawierający lustrzane odbicie. W tym przypadku przy kierunku cofnięcia odniesionym do kierunku stycznej następuje zamiana prawej i lewej strony. Składowe kierunków w kierunku narzędzia nie są poddawane lustrzanemu odbiciu. Takie zachowanie jest uaktywniane przez ustawienie MD:

```
MD21202 $MC_LIFTFAST_WITH_MIRROR = TRUE
```

Przykład

Wyłamane narzędzie ma zostać automatycznie zastąpione przez narzędzie siostrzane. Obróbka jest następnie kontynuowana przy użyciu nowego narzędzia.

Program główny:

Program główny	Komentarz
N10 SETINT(1) PRIO=1 W_WECHS LIFTFAST	; Gdy zastąpi zmiana sygnału na wejściu 1, narzędzie jest natychmiast w drodze cofnięcia szybkiego (kod nr 7 dla korekcji promienia narzędzia G41) odsuwane od konturu. Następnie jest wykonywana procedura przerwania "W_WECHS".
N20 G0 Z100 G17 T1 ALF=7 D1	
N30 G0 X-5 Y-22 Z2 M3 S300	
N40 Z-7	

Program główny	Komentarz
N50 G41 G1 X16 Y16 F200	
N60 Y35	
N70 X53 Y65	
N90 X71.5 Y16	
N100 X16	
N110 G40 G0 Z100 M30	

Podprogram:

Podprogram	Komentarz
PROC W_WECHS SAVE	; Podprogram z zapisaniem w pamięci aktualnego stanu roboczego
N10 G0 Z100 M5	; Pozycja wymiany narzędzia, stop wrzecziona
N20 T11 M6 D1 G41	; Wymiana narzędzia
N30 REPOSL RMBBL M3	; Ruch przywracający do konturu o powrót do programu głównego (programowanie następuje w jednym bloku)

1.14.7 Kierunek ruchu przy szybkim cofnięciu od konturu

Ruch wycofania

Kierunek ruchu wycofania jest określany przez następujące G-Code:

- LFTXT

Płaszczyzna ruchu wycofania jest określana ze stycznej do toru i kierunku narzędzia (ustawienie standardowe).

- LFWP

Płaszczyzna ruchu wycofania jest aktywną płaszczyzną roboczą, która jest wybierana przy pomocy G-Code G17, G18 albo G19. Kierunek ruchu wycofania jest niezależny od stycznej do toru. Przez to jest możliwe programowanie szybkiego cofnięcia równoległego do osi.

- LFPOS

Wycofanie osi podanej przy pomocy POLFMASK / POLFMLIN do absolutnej pozycji osi zaprogramowanej przy pomocy POLF.

ALF nie ma wpływu na kierunek cofnięcia dla wielu osi, jak też dla wielu osi w zależności liniowej.

Literatura:

Podręcznik programowania Podstawy; Rozdział: "Szybkie wycofanie się podczas nacinania gwintu"

Programowany kierunek ruchu (ALF=...)

W płaszczyźnie ruchu wycofania jest przy pomocy `ALF` programowany kierunek w krokach dyskretnych co 45 stopni.

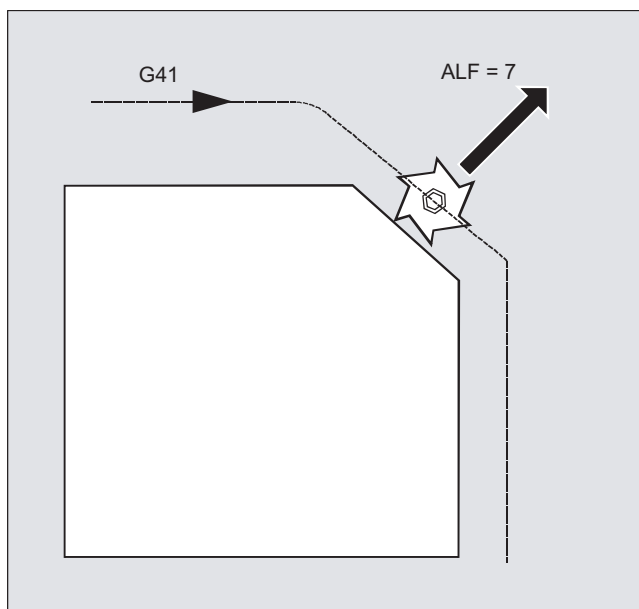
Możliwe kierunki ruchu są zapisane w sterowaniu pod specjalnymi numerami kodowymi i mogą być pod tymi numerami wywoływane.

Przykład:

Kod programu

```
N10 SETINT(2) PRIO=1 ABHEB_Z LIFTFAST  
ALF=7
```

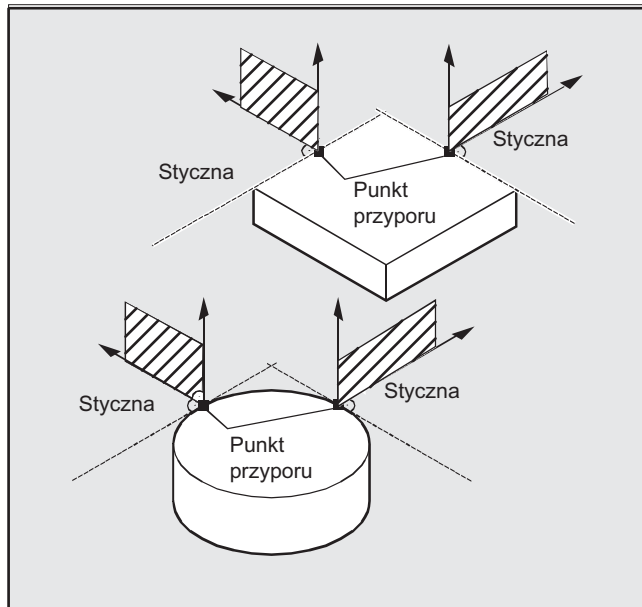
Narzędzie przy włączonym `G41` (kierunek obróbki na lewo od konturu) odsuwa się prostopadle od konturu.



Płaszczyzna odniesienia dla opisu kierunków ruchu w przypadku LFTXT

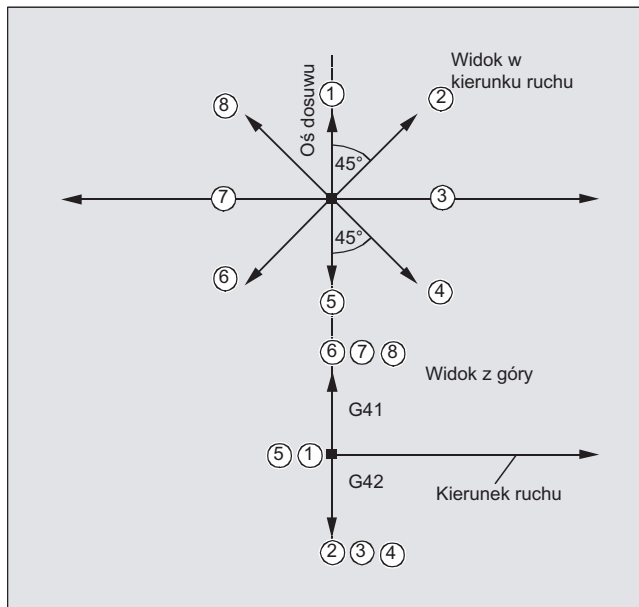
W punkcie przyporu narzędzia na zaprogramowanym konturze jest tworzona płaszczyna, która służy jako odniesienie dla podania ruchu cofnięcia przy pomocy odpowiedniego numeru kodowego.

Płaszczyzna odniesienia jest tworzona z osi wzdłużnej narzędzia (kierunek dosuwu) i wektora, który stoi na konturze prostopadle do niej i do stycznej w punkcie przyporu narzędzia.



Numery kodowe z kierunkami ruchu w przypadku LFTXT

Wychodząc od płaszczyzny odniesienia można na poniższym rysunku znaleźć numery kodowe z kierunkami ruchu.



Dla $ALF=1$ jest ustalone wycofanie w kierunku narzędzia.

Przy pomocy $ALF=0$ funkcja "szybkie cofnięcie" jest wyłączona.

OSTROŻNIE

Niebezpieczeństwo kolizji

Przy włączonej korekcji promienia narzędzia:

- przy $G41$ kodowania 2, 3, 4
- przy $G42$ kodowania 6, 7, 8

nie powinny być stosowane, gdyż w tych przypadkach narzędzie wykonałoby ruch w kierunku konturu i nastąpiłaby kolizja z obrabianym przedmiotem.

Numery kodowe z kierunkami ruchu w przypadku LFWP

W przypadku LFWP kierunek na płaszczyźnie roboczej wynika z następującego przyporządkowania:

- G17: Płaszczyzna X/Y
ALF=1: Wycofanie w kierunku X
ALF=3: wycofanie w kierunku Y
- G18: płaszczyzna Z/X
ALF=1: Wycofanie w kierunku Z
ALF=3: Wycofanie w kierunku X
- G19: Płaszczyzna Y/Z
ALF=1: wycofanie w kierunku Y
ALF=3: Wycofanie w kierunku Z

1.14.8 Przebieg ruchów w przypadku procedur przerwania

Procedura przerwania bez LIFTFAST

Ruchy w osiach są hamowane na torze, aż do stanu zatrzymanego. Następnie jest uruchamiana procedura przerwania.

Pozycja stanu zatrzymanego jest zapisywana jako pozycja przerwania i dosunięcie do niej w przypadku REPOS następuje za pomocą RMI na końcu procedury przerwania.

Procedura przerwania z LIFTFAST

Ruchy w osiach są hamowane na torze. Równocześnie jest wykonywany ruch LIFTFAST, jako ruch nałożony. Gdy ruch po torze i ruch LIFTFAST zatrzymały się, następuje uruchomienie procedury przerwania.

Jako pozycja przerwania jest zapisywana pozycja na konturze, w której jest uruchamiany ruch LIFTFAST, a przez to nastąpiło wyjście z toru.

Procedura przerwania z LIFTFAST i ALF=0 zachowuje się identycznie jak procedura przerwania bez LIFTFAST.

Wskazówka

Wielkość bezwzględna, o którą osie geometryczne odsuwają się od konturu przy szybkim cofnięciu, jest ustawiana przez daną maszynową.

1.15 Zamiana osi, zamiana wrzecion (RELEASE, GET, GETD)

Funkcja

Jedna albo wiele osi lub wrzecion może zawsze interpolować tylko w jednym kanale. Jeżeli oś musi pracować na przemian w dwóch różnych kanałach (np. w wymienniczu palet), wówczas musi ona najpierw zostać zwolniona w aktualnym kanale, a następnie przejęta w innym kanale. Oś jest zamieniana między kanałami.

Rozszerzenia zamiany osi

Oś/wrzeciono może być poddawane zamianie z zatrzymaniem przebiegu wyprzedzającego i synchronizacją między przebiegiem wyprzedzającym i przebiegiem głównym lub również bez zatrzymania przebiegu wyprzedzającego. Poza tym zamiana osi jest również możliwa przez

- Obrót pojemnika osi AXCTSWE lub AXCTWED przy pomocy implicite GET/GETD.
- Frame z rotacją, gdy ta oś jest przez to powiązana z innymi osiami.
- Akcje synchroniczne, patrz akcje synchroniczne ruchu, "zamiana osi RELEASE, GET".

Producent maszyny

Proszę przestrzegać danych producenta maszyny. Poprzez projektowane dane maszynowe oś musi być w celu zamiany osi jednoznacznie zdefiniowana we wszystkich kanałach, a zachowanie się pod względem zamiany osi jest również przez dane maszynowe ustawiane zmiennie.

Składnia

RELEASE(nazwa osi, nazwa osi, ...) lub RELEASE(S1)

GET(nazwa osi, nazwa osi, ...) lub GET(S2)

GETD(nazwa osi, nazwa osi, ...) lub GETD(S3)

Przy pomocy GETD (GET Directly) oś jest pobierana bezpośrednio z innego kanału. Oznacza to, że do tego GETD nie musi być programowany pasujący RELEASE w innym kanale. Oznacza to jednak również, że teraz musi zostać zbudowana inna komunikacja kanałowa (np. znaczniki Wait).

Znaczenie

RELEASE (nazwa osi, nazwa osi, ...):

GET (nazwa osi, nazwa osi, ...):

GETD (nazwa osi, nazwa osi, ...):

Nazwa osi:

RELEASE(S1) :

GET(S2) :

GETD(S3) :

Zwolnienie osi

Przejęcie osi

Bezpośrednie przejęcie osi

Przyporządkowanie osi w systemie: AX1, AX2, ... lub podanie nazw osi maszyny

Zwolnienie wrzeciona S1, S2, ...

Przejęcie wrzeciona S1, S2, ...

Bezpośrednie przejęcie wrzeciona S1, S2, ...

Zażądanie GET bez zatrzymania przebiegu wyprzedzającego

Jeżeli po zażądaniu GET bez zatrzymania przebiegu wyprzedzającego oś zostanie ponownie zwolniona przy pomocy RELEASE (oś) lub WAITP (oś), wówczas następne GET prowadzi do GET z zatrzymaniem przebiegu wyprzedzającego.

! OSTROŻNIE**Zmiana przyporządkowania osi**

Oś wzgl. wrzeciono przejęte przy pomocy GET pozostaje przyporządkowane do tego kanału również po naciśnięciu RESET albo z programu.

Przy ponownym uruchomieniu programu przyporządkowanie zamienionych osi lub wrzecion musi nastąpić programowo, w przypadku gdy oś jest potrzebna w swoim podstawowym kanale.

W przypadku POWER ON zostaje ona przyporządkowana do kanału zapisanego w danej maszynowej.

Przykłady**Przykład 1: Wymiana osi pomiędzy kanałami**

6 osi kanału 1 używa się do obróbki: 1., 2., 3. i 4 oś.

Osie 5. i 6. są używane w kanale 2 do wymiany obrabianego przedmiotu.

Oś 2 ma być możliwa do zamiany między obydwoma kanałami, a po załączeniu zasilania być przyporządkowana do kanału 1.

Program „MAIN“ w kanale 1:

Kod programu	Komentarz
INIT (2, "TAUSCH2")	; Należy wybrać program TAUSCH2 w kanale 2.
N... START (2)	; Start programu w kanale 2.
N... GET (AX2)	; Przejęcie osi AX2.
...	
N... RELEASE (AX2)	; Zwolnienie osi AX2.
N... WAITM (1,1,2)	; Czekanie na znacznik WAIT w kanale 1 i 2 w celu synchronizacji w obydwu kanałach.
...	; Dalszy przebieg po zamianie osi.
N... M30	

Program "TAUSCH2" w kanale 2:

Programowanie	Komentarz
N... RELEASE (AX2)	
N160 WAITM(1,1,2)	; Czekanie na znacznik WAIT w kanale 1 i 2 w celu synchronizacji w obydwu kanałach.
N150 GET (AX2)	; Przejęcie osi AX2.
...	; Dalszy przebieg po zamianie osi.
N... M30	

Przykład 2: Zamiana osi bez synchronizacji

Gdy oś nie musi być synchronizowana, GET nie wytwarza zatrzymania przebiegu wyprzedzającego.

Programowanie	Komentarz
N01 G0 X0	
N02 RELEASE (AX5)	
N03 G64 X10	
N04 X20	
N05 GET (AX5)	; Gdy synchronizacja nie jest konieczna, nie jest to wykonywalny blok.
N06 G01 F5000	; Nie jest to wykonywalny blok.
N07 X20	; Nie jest to wykonywalny blok, ponieważ pozycja X jest taka jak w N04.
N08 X30	; Pierwszy wykonywalny blok po N05.
...	

Przykład 3: Uaktywnienie zamiany osi bez zatrzymania przebiegu wyprzedzającego

Warunek: Zamiana osi bez zatrzymania przebiegu wyprzedzającego musi zostać zaprogramowana przez daną maszynową.

Programowanie	Komentarz
N010 M4 S100	
N011 G4 F2	
N020 M5	
N021 SPOS=0	
N022 POS[B]=1	
N023 WAITP(B)	; Oś B staje się osią neutralną.
N030 X1 F10	
N031 X100 F500	
N032 X200	
N040 M3 S500	; Oś nie wyzwala zatrzymania przebiegu wyprzedzającego/REORG.
N041 G4 F2	
N050 M5	
N099 M30	

Jeżeli jest wykonywany ruch wrzeciona lub osi B bezpośrednio po bloku N023 jako **osi PLC**, np. na 180 stopni i z powrotem na 1 stopień, wówczas ta oś staje się z powrotem osią neutralną i w bloku N40 nie wyzwala zatrzymania przebiegu wyprzedzającego.

Warunek

Warunki zamiany osi

- Oś musi być przez dane maszynowe zdefiniowana we wszystkich kanałach, które chcą tej osi używać.
- Poprzez specyficzną dla osi daną maszynową musi być ustalone, któremu kanałowi ma być przyporządkowana oś po załączeniu zasilania.

Opis

Zwolnienie osi: RELEASE

Przy zwolnieniu osi należy przestrzegać:

1. Oś nie może uczestniczyć w żadnej transformacji.
2. Przy sprzężeniach osi (sterowanie styczne), muszą zostać zwolnione wszystkie osie zespołu.
3. Konkurującej osi pozycjonowania nie wolno w tym stanie zamienić.
4. W przypadku osi wiodącej gantry są zamieniane również wszystkie osie nadążne.
5. W przypadku sprzężeń osi (nadażanie, sprzężenie wartości wiodącej, przekładnia elektroniczna) można zwolnić tylko oś wiodącą systemu.

Przejęcie osi: GET

Przy pomocy tego polecenia przeprowadza się właściwą zamianę osi. Odpowiedzialność za oś leży całkowicie w tym kanale, w którym zaprogramowano polecenie.

Skutki GET:

Zamiana osi z synchronizacją:

Oś musi być synchronizowana zawsze wtedy, gdy w międzyczasie była przyporządkowana do innego kanału albo do PLC, a przed GET nie nastąpiła synchronizacja przez "WAITP", G74 albo skasowanie pozostałej drogi.

- Zatrzymanie przebiegu wyprzedzającego następuje (jak przy STOPRE).
- Obróbka zostaje przerwana na tak długo, aż zamiana zostanie całkowicie wykonana.

Automatyczne "GET"

Jeżeli oś jest zasadniczo dostępna w kanale, ale obecnie nie występuje jako "oś kanału", następuje automatyczne wykonanie GET. Gdy oś(sie) jest(sa) już zsynchronizowana(e), zatrzymanie przebiegu wyprzedzającego nie jest wytwarzane.

Ustawienie zmiennego zachowania się przy zamianie osi

Moment oddania osi daje się ustawić poprzez daną maszynową jak następuje:

- Automatyczna zamiana odbywa się między dwoma kanałami również wtedy, gdy oś została przez WAITP doprowadzona do stanu neutralnego (zachowanie się jak dotychczas)
- Przy zażądaniu obrotu pojemnika osi wszystkie przyporządkowalne do kanału wykonującego osie pojemnika są przy pomocy następującego implicite GET lub GETD pobierane do kanału. Następną zamiana osi jest dozwolona dopiero po zakończeniu obrotu pojemnika osi.
- Po wstawionym bloku pośrednim w przebiegu głównym następuje sprawdzenie, czy reorganizacja jest konieczna czy nie. Tylko gdy stany osi tego bloku **nie** są zgodne z aktualnymi stanami osi, jest wymagana reorganizacja.
- Zamiast bloku GET z zatrzymaniem przebiegu wyprzedzającego i synchronizacją między przebiegiem wyprzedzającym i głównym zamiana osi może nastąpić również bez zatrzymania przebiegu wyprzedzającego. Jest wówczas tworzony tylko blok pośredni z zażądaniem GET. W przebiegu głównym przy wykonywaniu tego bloku następuje sprawdzenie, czy stany osi w bloku są zgodne z aktualnymi stanami osi.

Dalsze informacje dot. działania zamiany osi i wrzecion patrz

Podręcznik działania Funkcje rozszerzające; BAG, kanały, zamiana osi (K5).

1.16 Przekazanie osi do innego kanału (AXTOCHAN)

Funkcja

Przy pomocy polecenia językowego `AXTOCHAN` może zostać zażądana oś przekazana do innego kanału. Oś może zostać pobrana do odpowiedniego kanału zarówno z programu obróbki NC, jak też z akcji synchronicznej.

Składnia

`AXTOCHAN(nazwa osi,numer kanału[,nazwa osi,numer kanału[,...]])`

Znaczenie

Element	Opis
<code>AXTOCHAN:</code>	Zażądanie osi dla określonego kanału
Nazwa osi:	Przyporządkowanie osi w systemie: X, Y, ... lub podanie nazw uczestniczących osi maszyny. Wykonującym kanałem nie musi być własny kanał i nie musi to być też ten kanał, który posiada aktualnie prawo interpolacji dla osi
numer kanału:	Numer kanału, któremu oś ma zostać przyporządkowana

Wskazówka**Konkurująca oś pozycjonowania i oś kontrolowana wyłącznie przez PLC**

Oś PLC nie może jako konkurująca oś pozycjonowania zmienić kanału. Oś kontrolowana wyłącznie przez PLC nie może zostać przyporządkowana do programu NC.

Literatura:

Podręcznik działania Funkcje rozszerzające; Osie pozycjonowania (P2)

Przykład**AXTOCHAN w programie NC**

Osie X i Y są znane w 1. kanale i w 2. kanale. Aktualnie kanał 1 ma prawo interpolacji i w kanale 1 jest uruchamiany następujący program:

Kod programu	Komentarz
N110 AXTOCHAN(Y,2)	; Przesunięcie osi Y do 2. kanału.
N111 M0	
N120 AXTOCHAN(Y,1)	; Sprowadzenie osi Y z powrotem (neutralna).
N121 M0	
N130 AXTOCHAN(Y,2,X,2)	; Przesunięcie osi Y i osi X do 2. kanału (osie neutralne).
N131 M0	
N140 AXTOCHAN(Y,2)	; Przesunięcie osi Y do 2. kanału (program NC).
N141 M0	

Dalsze informacje**AXTOCHAN w programie NC**

Przy tym jest tylko przy zażądaniu osi dla programu NC we własnym kanale przeprowadzane GET, a przez to również następuje czekanie na rzeczywistą zmianę stanu. Jeżeli oś zostanie zażądana dla innego kanału albo ma stać się osią neutralną we własnym kanale, wówczas jest tylko odpowiednio generowane żądanie.

AXTOCHAN z akcji synchronicznej

Gdy oś zostanie zażądana dla własnego kanału, wówczas AXTOCHAN z akcji synchronicznej zostanie odwzorowane na GET z akcji synchronicznej. W tym przypadku oś staje się neutralna przy pierwszym zażądaniu dla własnego kanału. Przy drugim zażądaniu oś jest przyporządkowywana do programu NC analogicznie do zażądania GET w programie NC. Odnośnie zażądania GET z akcji synchronicznej patrz punkt "Akcje synchroniczne ruchu".

1.17 Ustawienie działania danych maszynowych (NEWCONF)

Funkcja

Przy pomocy polecenia `NEWCONF` jest ustawiane działanie wszystkich danych maszynowych stopnia działania "NEW_CONFIG". Funkcja może zostać również uaktywniona na interfejsie graficznym HMI przez naciśnięcie przycisku programowego "Ustaw działanie MD".

Przy wykonywaniu funkcji `NEWCONF` następuje implicite zatrzymanie przebiegu wyprzedzającego, tzn. ruch po torze jest przerywany.

Składnia

`NEWCONF`

Znaczenie

`NEWCONF`: Polecenie do ustawienia działania wszystkich danych maszynowych stopnia działania "NEW_CONFIG"

Wykonanie NEWCONF z programu obróbki poza granice kanału

Jeżeli osiowe dane maszynowe zostaną zmienione z programu obróbki, a następnie uaktywnione przy pomocy `NEWCONF`, wówczas polecenie `NEWCONF` uaktywnia tylko te dane maszynowe, które powodują zmiany dla kanału programu obróbki.

Wskazówka

Aby spowodować niezawodne działanie wszystkich zmian, polecenie `NEWCONF` musi zostać wykonane w każdym kanale, w którym są aktualnie liczone odnośne osie albo funkcje zmienione przez dane maszynowe.

Przy `NEWCONF` nie są ustawiane żadne osiowe dane maszynowe.

Dla osi kontrolowanych przez PLC musi zostać wykonany RESET osiowy.

Przykład

Obróbka frezarska: obróbka w pozycjach wiercenia przy pomocy różnych technologii

Kod programu	Komentarz
N10 \$MA_CONTOUR_TOL[AX]=1.0	; Zmiana danej maszynowej
N20 NEWCONF	; Ustawienie działania danych maszynowych.
...	

1.18 Zapisanie pliku (WRITE)

Funkcja

Przy pomocy polecenia `WRITE` bloki/dane z programu NC można zapisać na końcu podanego pliku w pasywnym systemie plików (plik protokołu). Może to być również właśnie wykonywany program.

Wskazówka

Plik, w którym ma nastąpić zapisanie przez polecenie `WRITE`, ulega utworzeniu, jeżeli nie istnieje w NC.

Miejszem zapisania jest statyczna pamięć NC. W przypadku SINUMERIK 840D si jest to CompactFlash Card. W porównaniu z SINUMERIK 840D czas wykonywania polecenia `WRITE` zwiększa się przez to o ok. 75 ms.

Jeżeli na dysku twardym istnieje plik o takiej samej nazwie, jest on zastępowany po zamknięciu pliku (w NC). Pomoc: Poprzez interfejs graficzny zmienić nazwę pliku w NC.

Ponadto jest przy pomocy polecenia `WRITE` również możliwe zapisanie danych z programu NC na urządzeniu zewnętrznym / w pliku zewnętrznym (patrz też "Process DataShare - Wyprowadzenie do zewnętrznego urządzenia/pliku (EXTOPEN, WRITE, EXTCLOSE) (Strona 614)").

Warunek

Aktualnie ustawiony poziom dostępu musi być równy albo większy od prawa `WRITE` pliku. Jeżeli tak nie jest, dostęp jest odrzucany z komunikatem błędu (zwracana wartość zmiennej błędu = 13).

Składnia

```
DEF INT <błąd>  
...  
WRITE(<błąd>,"<nazwa pliku>"/<urządzenie zewnętrzne>",<blok/dane>")
```

Znaczenie

`WRITE:` Polecenie do dołączenia bloku lub danych na końcu podanego pliku

`<błąd>:` **Parametr 1:** Zmienna do zwrotu wartości błędu

Typ: INT

Wartość: 0 Nie ma błędu
1 Ścieżka niedozwolona
2 Ścieżka nie znaleziona
3 Plik nie znaleziony

- 4 Nieprawidłowy typ pliku
- 10 Plik jest zapełniony
- 11 Plik jest używany
- 12 Brak wolnych zasobów
- 13 Brak praw dostępu
- 14 Brakujące lub nieudane EXTOPEN dla urządzenia wyjściowego
- 15 Błąd przy zapisie na urządzeniu zewnętrznym
- 16 Zaprogramowano niepoprawną ścieżkę zewnętrzną

<nazwa pliku>:

Parametr 2: Nazwa pliku w pasywnym systemie plików, w którym podany blok wzgl. podane dane mają zostać dołączone

Typ: STRING

Przy podawaniu nazwy pliku należy przestrzegać następujących punktów:

- Podana nazwa pliku nie może zawierać żadnych spacji ani znaków sterujących (znaki w kodzie ASCII ≤ 32), gdyż w przeciwnym przypadku polecenie WRITE zostanie anulowane z identyfikatorem błędu 1 "Niedozwolona ścieżka".
- Nazwę pliku można podać ze ścieżką i rozszerzeniem.

- Podanie ścieżki

Nazwy ścieżek muszą być absolutne, tzn. rozpoczynają się one od "/".

Bez podania ścieżki plik jest zapisywany w aktualnym katalogu (= katalogu wybranego programu).

- Rozszerzenie pliku

Jeżeli nazwa pliku nie zawiera identyfikatora domeny ("_N_"), jest on odpowiednio uzupełniany.

Jeżeli nazwa pliku zawiera podkreślenie "_" jako czwarty znak od końca, wówczas następne trzy znaki są interpretowane jako rozszerzenie pliku. Aby w przypadku wszystkich poleceń plikowych móc stosować tę samą nazwę pliku, np. przez zmienną typu STRING, wolno stosować tylko rozszerzenia _SPF i _MPF.

Jeżeli nie jest podane rozszerzenie "_MPF" lub "_SPF", jest automatycznie uzupełniane _MPF.

- Długość nazwy pliku może wynosić maksymalnie 32 bajty, długość ścieżki maksymalnie 128 bajtów.

Przykład:

```
"PROFILE"  
"_N_PROFILE"  
"_N_PROFILE_MPF"  
"/_N_MPF_DIR/_N_PROFILE_MPF/"
```

<urządzenie zewnątrzne>:	Jeżeli dane mają zostać wyprowadzone na urządzenie zewnętrzne / do pliku zewnętrznego, musi zamiast nazwy pliku zostać podany symboliczny identyfikator otwieranego zewnętrznego urządzenia/pliku. Typ: STRING Dalsze informacje patrz "Process DataShare - Wyprowadzenie do zewnętrznego urządzenia/pliku (EXTOPEN, WRITE, EXTCLOSE) (Strona 614)". Wskazówka: Identyfikator musi być identyczny z identyfikatorem podanym w poleceniu EXTOPEN.
<blok/dane>:	Plik lub dane, które mają zostać dołączone w podanym pliku. Typ: STRING

Wskazówka

Przy zapisie do pasywnego systemu plików NCK polecenie `WRITE` dołącza implicite znak "LF" (LINE FEED = nowy wiersz) na końcu wyprowadzanego łańcucha znaków.

Dla wyprowadzania na urządzenie zewnętrzne / do pliku zewnętrznego nie jest wprowadzany znak LF. Jeżeli równocześnie ma być wprowadzane "LF", musi zostać explicite podane w wyprowadzonym łańcuchu znaków.

→ Patrz na niniejszy przykład 3: Implicite/explicite "LF"!

Warunki brzegowe

- **Maksymalna wielkość pliku (→ producent maszyny!)**

Maksymalna możliwa wielkość plików protokołowych w pasywnym systemie plików jest ustawiana w danej maszynowej:

MD11420 \$MN_LEN_PROTOCOL_FILE

Maksymalna wielkość pliku obowiązuje dla wszystkich plików, które zostaną utworzone w pasywnym systemie plików przy pomocy polecenia `WRITE`. Przy przekroczeniu jest wyprowadzany komunikat błędu i blok wzgl. dane nie są zapisywane. O ile pamięć jest wystarczająca, można utworzyć nowy plik.

Przykłady

Przykład 1: Polecenie WRITE do pasywnego systemu plików bez absolutnego podania ścieżki

Kod programu	Komentarz
N10 DEF INT ERROR	; Definicja zmiennej błędu.
N20 WRITE(ERROR,"PROT","PROTOKOL Z 7.2.97")	; Zapisz tekst z "PROTOKOL Z 7.2.97" do pliku _N_PROT_MPF.
N30 IF ERROR	; Ewaluacja błędu.
N40 MSG ("Błąd przy poleceniu WRITE:" <<ERROR)	
N50 M0	
N60 ENDIF	
...	

Przykład 2: Polecenie WRITE do pasywnego systemu plików z absolutnym podaniem ścieżki

Kod programu
...
WRITE(ERROR,"/_N_WKS_DIR/_N_PROT_WPD/_N_PROT_MPF","PROTOKOL Z 7.2.97")
...

Przykład 3: Implicite/explicite "LF"

a, zapis do pasywnego systemu plików z tworzonym implicite "LF"

Kod programu
...
N110 DEF INT ERROR
N120 WRITE(ERROR,"/_N_MPF_DIR/_N_MYPROTFILE_MPF","MY_STRING")
N130 WRITE(ERROR,"/_N_MPF_DIR/_N_MYPROTFILE_MPF","MY_STRING")
N140 M30

Wyprowadzany wynik:

MY_STRING

MY_STRING

b, zapis do pliku zewnętrznego bez tworzonego implicity "LF"

```
Kod programu
...
N200 DEF STRING[30] DEV_1
N210 DEF INT ERROR
N220 DEV_1="LOCAL_DRIVE/myprotfile.mpf"
N230 EXTOPEN(ERROR,DEV_1)
N240 WRITE(ERROR,DEV_1,"MY_STRING")
N250 WRITE(ERROR,DEV_1,"MY_STRING")
N260 EXTCLOSE(ERROR,DEV_1)
N270 M30
```

Wyprowadzany wynik:

MY_STRINGMY_STRING

c, zapis w urządzeniu zewnętrznym z zaprogramowanym explicite "LF"

Aby uzyskać ten sam wynik co przy a, musi zostać zaprogramowane co następuje:

```
Kod programu
...
N200 DEF STRING[30] DEV_1
N210 DEF INT ERROR
N220 DEV_1="LOCAL_DRIVE/myprotfile.mpf"
N230 EXTOPEN(ERROR,DEV_1)
N240 WRITE(ERROR,DEV_1,"MY_STRING'H0A'")
N250 WRITE(ERROR,DEV_1,"MY_STRING'H0A'")
N260 EXTCLOSE(ERROR,DEV_1)
N270 M30
```

Wyprowadzany wynik:

MY_STRING

MY_STRING

Patrz również

Process DataShare - Wyprowadzenie do zewnętrznego urządzenia/pliku (EXTOPEN, WRITE, EXTCLOSE) (Strona 614)

1.19 Skasowanie pliku (DELETE)

Funkcja

Przy pomocy polecenia `DELETE` mogą być kasowane wszystkie pliki, wszystko jedno czy powstały poprzez polecenie `WRITE` czy nie. Również pliki, które zostały sporządzone pod najwyższym poziomem dostępu, można skasować przy pomocy `DELETE`.

Składnia

```
DEF INT <błąd>  
DELETE (<błąd>, "<nazwa pliku>")
```

Znaczenie

<code>DELETE:</code>	Polecenie do skasowania podanego pliku
<code><błąd>:</code>	Zmienna do zwrotu wartości błędu
Typ.	INT
Wartość:	0 Nie ma błędu
	1 Ścieżka niedozwolona
	2 Ścieżka nie znaleziona
	3 Plik nie znaleziony
	4 Nieprawidłowy typ pliku
	11 Plik jest używany
	12 Brak wolnych zasobów
	20 Inny błąd

- <nazwa pliku>: Nazwa pliku do skasowania
Typ: STRING
Przy podawaniu nazwy pliku należy przestrzegać następujących punktów:
- Podana nazwa pliku nie może zawierać żadnych spacji ani znaków sterujących (znaki w kodzie ASCII ≤ 32), gdyż w przeciwnym przypadku polecenie DELETE zostanie anulowane z identyfikatorem błędu 1 "Niedozwolona ścieżka".
 - Nazwę pliku można podać ze ścieżką i rozszerzeniem.
 - Podanie ścieżki
Nazwy ścieżek muszą być absolutne, tzn. rozpoczynają się one od "/".
Bez podania ścieżki plik jest szukany w aktualnym katalogu (= katalogu wybranego programu).
 - Rozszerzenie pliku
Jeżeli nazwa pliku nie zawiera identyfikatora domeny ("_N_"), jest on odpowiednio uzupełniany.
Jeżeli nazwa pliku zawiera podkreślenie "_" jako czwarty znak od końca, wówczas następne trzy znaki są interpretowane jako rozszerzenie pliku. Aby w przypadku wszystkich poleceń plikowych móc stosować tę samą nazwę pliku, np. przez zmienną typu STRING, wolno stosować tylko rozszerzenia _SPF i _MPF.
Jeżeli nie jest podane rozszerzenie "_MPF" lub "_SPF", jest automatycznie uzupełniane _MPF.
 - Długość nazwy pliku może wynosić maksymalnie 32 bajty, długość ścieżki maksymalnie 128 bajtów.

Przykład:

```
"PROTFILE"  
"_N_PROTFILE"  
"_N_PROTFILE_MPF"  
"/_N_MPF_DIR/_N_PROTFILE_MPF/"
```

Przykład

Kod programu	Komentarz
N10 DEF INT ERROR	; Definicja zmiennej błędu.
N15 STOPRE	; Zatrzymanie przebiegu wyprzedzającego.
N20 DELETE(ERROR, "/_N_SPF_DIR/_N_TEST1_SPF")	; Skasuj plik TEST1 w katalogu podprogramu
N30 IF ERROR	; Ewaluacja błędu.
N40 MSG ("Błąd przy poleceniu DELETE:" <<ERROR)	
N50 M0	
N60 ENDIF	

1.20 Odczyt wierszy w pliku (READ)

Funkcja

Polecenie `READ` czyta w podanym pliku jeden albo wiele wierszy i zapisuje przeczytane informacje w tablicy typu `STRING`. Każdy przeczytany wiersz zajmuje w tej tablicy jeden element.

Wskazówka

Plik musi się znajdować w statycznej pamięci użytkownika NCK (pasywny system plików).

Warunek

Aktualnie ustawiony poziom dostępu musi być równy albo większy od uprawnienia `READ` pliku. Jeżeli tak nie jest, dostęp jest odrzucany z komunikatem błędu (zwracana wartość zmiennej błędu = 13).

Składnia

```
DEF INT <błąd>
DEF STRING [<długość łańcucha znaków>] <wynik> [<n>, <m>]
READ (<błąd>, "<nazwa pliku>", <wiersz początkowy>, <liczba
wierszy>, <wynik>)
```

Znaczenie

<code>READ:</code>	Polecenie do odczytu wierszy podanego pliku i do zapisania tych wierszy w tablicy zmiennych.
<code><błąd>:</code>	Zmienna do zwrotu wartości błędu (parametr Call-By-Reference)
Typ.	INT
Wartość:	0 Nie ma błędu
	1 Ścieżka niedozwolona
	2 Ścieżka nie znaleziona
	3 Plik nie znaleziony
	4 Nieprawidłowy typ pliku
	11 Plik jest używany
	13 Prawa dostępu niewystarczające
	21 Wiersza nie ma (parametr <code><wiersz początkowy></code> lub <code><liczba wierszy></code> większy, niż liczba wierszy w podanym pliku).
	22 Długość tablicy zmiennych wynikowych (<code><wynik></code>) jest za mała.
	23 Zakres wiersza za duży (parametr <code><liczba wierszy></code> został wybrany tak duży, że odczyt następuje poza końcem pliku).

<nazwa pliku>:	Nazwa czytanego pliku (parametr Call-By-Value) Typ: STRING Przy podawaniu nazwy pliku należy przestrzegać następujących punktów: <ul style="list-style-type: none">• Podana nazwa pliku nie może zawierać żadnych spacji ani znaków sterujących (znaki w kodzie ASCII ≤ 32), gdyż w przeciwnym przypadku polecenie <code>READ</code> zostanie anulowane z identyfikatorem błędu 1 "Niedozwolona ścieżka".• Nazwę pliku można podać ze ścieżką i rozszerzeniem.<ul style="list-style-type: none">– Podanie ścieżki Nazwy ścieżek muszą być absolutne, tzn. rozpoczynają się one od "/". Bez podania ścieżki plik jest szukany w aktualnym katalogu (= katalogu wybranego programu).– Rozszerzenie pliku Jeżeli nazwa pliku nie zawiera identyfikatora domeny ("_N_"), jest on odpowiednio uzupełniany. Jeżeli nazwa pliku zawiera podkreślenie "_" jako czwarty znak od końca, wówczas następne trzy znaki są interpretowane jako rozszerzenie pliku. Aby w przypadku wszystkich poleceń plikowych móc stosować tę samą nazwę pliku, np. przez zmienną typu STRING, wolno stosować tylko rozszerzenia <code>_SPF</code> i <code>_MPF</code>. Jeżeli nie jest podane rozszerzenie <code>"_MPF"</code> lub <code>"_SPF"</code>, jest automatycznie uzupełniane <code>_MPF</code>.• Długość nazwy pliku może wynosić maksymalnie 32 bajty, długość ścieżki maksymalnie 128 bajtów.
	Przykład: "PROTFILE" "_N_PROTFILE" "_N_PROTFILE_MPF" "/_N_MPF_DIR/_N_PROTFILE_MPF/"
<wiersz początkowy>:	Wiersz początkowy czytanej części pliku (parametr Call-By-Value) Typ: INT Wartość: 0 Jest czytana liczba wierszy przed końcem pliku podana przy pomocy parametru <Liczba wierszy>. 1 ... n Numer pierwszego wiersza do odczytania.
<liczba wierszy>:	Liczba czytanych wierszy (parametr Call-By-Value) Typ: INT

<wynik>: Zmienna wynikowa (parametr Call-By-Reference)
 Tablica zmiennych, w której jest zapisywany czytany tekst.
 Typ: STRING (maks. długość: 255)
 Jeżeli w parametrze <liczba wierszy> jest podanych mniej wierszy, niż wynosi wielkość tablicy zmiennych wynikowych [<n>, <m>], wówczas pozostałe elementy tablicy nie są zmieniane.
 Zakończenie wiersza przez znak sterujący "LF" (line feed) albo "CR LF" (Carrige Return Line Feed) **nie** jest zapisywane w zmiennej wynikowej.
 Przeczytane wiersze są obcinane, gdy wiersz jest dłuższy, niż zdefiniowana długość łańcucha znaków. Nie następuje komunikat błędu.

Wskazówka

Pliki binarne nie mogą być wczytywane. Jest wyprawdzany błąd "Nieprawidłowy typ pliku" (zwracana wartość zmiennej błędu = 4). Następujące typy plików nie dają się czytać: _BIN, _EXE, _OBJ, _LIB, _BOT, _TRC, _ACC, _CYC, _NCK.

Przykład

Kod programu	Komentarz
N10 DEF INT ERROR	;Definicja zmiennej błędu.
N20 DEF STRING[255] RESULT[5]	;Definicja zmiennej wynikowej.
N30 READ(ERROR, "_N_CST_DIR/_N_TESTFILE_MPF", 1, 5, RESULT)	;Nazwa pliku z identyfikatorem domeny, rozszerzeniem pliku ; i podaniem ścieżki.
N40 IF ERROR <>0	;Ewaluacja błędu.
N50 MSG("BŁĄD"<<ERROR<<"W PRZYPADKU POLECENIA READ")	
N60 M0	
N70 ENDIF	
...	

1.21 Sprawdzenie istnienia pliku (ISFILE)

Funkcja

Przy pomocy polecenia `ISFILE` można sprawdzić, czy plik istnieje w statycznej pamięci użytkownika w NCK (pasywny system plików).

Składnia

```
<wynik>=ISFILE("<nazwa pliku>")
```

Znaczenie

- `ISFILE`: Polecenie do sprawdzenia, czy podany plik istnieje w pasywnym systemie plików.
- `<nazwa pliku>`: Nazwa pliku, którego istnienie w pasywnym systemie plików ma zostać sprawdzone.
- Typ: `STRING`
- Przy podawaniu nazwy pliku należy przestrzegać następujących punktów:
- Podawana nazwa pliku nie może zawierać spacji ani znaków sterujących (znaki w kodzie ASCII ≤ 32).
 - Nazwę pliku można podać ze ścieżką i rozszerzeniem.
 - Podanie ścieżki
Nazwy ścieżek muszą być absolutne, tzn. rozpoczynają się one od `"/`.
 - Rozszerzenie pliku
Bez podania ścieżki plik jest szukany w aktualnym katalogu (= katalogu wybranego programu).
- Jeżeli nazwa pliku nie zawiera identyfikatora domeny (`"_N_"`), jest on odpowiednio uzupełniany.
- Jeżeli nazwa pliku zawiera podkreślenie `"_"` jako czwarty znak od końca, wówczas następne trzy znaki są interpretowane jako rozszerzenie pliku. Aby w przypadku wszystkich poleceń plikowych móc stosować tę samą nazwę pliku, np. przez zmienną typu `STRING`, wolno stosować tylko rozszerzenia `_SPF` i `_MPF`.
- Jeżeli nie jest podane rozszerzenie `"_MPF"` lub `"_SPF"`, jest automatycznie uzupełniane `_MPF`.
- Długość nazwy pliku może wynosić maksymalnie 32 bajty, długość ścieżki maksymalnie 128 bajtów.

Przykład:

```
"PROTFILE"  
"_N_PROTFILE"  
"_N_PROTFILE_MPF"  
"/_N_MPF_DIR/_N_PROTFILE_MPF/"
```

<wynik>: Zmienna wynikowa do ujęcia wyniku kontroli
Typ. BOOL
Wartość: TRUE Plik istnieje
 FALSE Plik nie istnieje

Przykład

Kod programu	Komentarz
N10 DEF BOOL RESULT	; Definicja zmiennej wynikowej.
N20 RESULT=ISFILE("TESTFILE")	
N30 IF (RESULT==FALSE)	
N40 MSG("NIE MA PLIKU")	
N50 M0	
N60 ENDIF	
...	

lub:

Kod programu	Komentarz
N10 DEF BOOL RESULT	; Definicja zmiennej wynikowej.
N20 RESULT=ISFILE("TESTFILE")	
N30 IF (NOT ISFILE("TESTFILE"))	
N40 MSG("NIE MA PLIKU")	
N50 M0	
N60 ENDIF	
...	

1.22 Odczyt informacji o pliku (FILEDATE, FILETIME, FILESIZE, FILESTAT, FILEINFO)

Funkcja

Przez polecenia FILEDATE, FILETIME, FILESIZE, FILESTAT i FILEINFO można odczytać określone informacje o pliku, jak data / czas zegarowy ostatniego dostępu zapisującego, aktualna wielkość pliku, status pliku lub suma tych informacji.

Wskazówka

Plik musi się znajdować w statycznej pamięci użytkownika NCK (pasywny system plików).

Warunek

Aktualnie ustawiony poziom dostępu musi być równy albo większy od praw dostępu katalogu nadrzędnego. Jeżeli tak nie jest, dostęp jest odrzucany z komunikatem błędu (zwracana wartość zmiennej błędu = 13).

Składnia

FILE... (<błąd>, "<nazwa pliku>", <wynik>)

Znaczenie

FILEDATE:	Podaje datę ostatniego dostępu do pliku w celu zapisu												
FILETIME:	Podaje czas zegarowy ostatniego dostępu do pliku w celu zapisu												
FILESIZE:	Podaje aktualną wielkość pliku												
FILESTAT:	Podaje dla statusu pliku odnośnie następujących praw : <ul style="list-style-type: none"> • odczyt (r: read) • zapis (w: write) • wykonanie (x: execute) • wyświetlenie (s: show) • skasowanie (d: delete) 												
FILEINFO:	Podaje dla pliku sumę informacji , które można odczytać przez FILEDATE, FILETIME, FILESIZE i FILESTAT												
<błąd>:	Zmienna do zwrotu wartości błędu (parametr Call-By-Reference) <table> <thead> <tr> <th>Typ.</th> <th>VAR INT</th> </tr> </thead> <tbody> <tr> <td>Wartość:</td> <td>0 Nie ma błędu</td> </tr> <tr> <td></td> <td>1 Ścieżka niedozwolona</td> </tr> <tr> <td></td> <td>2 Ścieżka nie znaleziona</td> </tr> <tr> <td></td> <td>3 Plik nie znaleziony</td> </tr> <tr> <td></td> <td>4 Nieprawidłowy typ pliku</td> </tr> </tbody> </table>	Typ.	VAR INT	Wartość:	0 Nie ma błędu		1 Ścieżka niedozwolona		2 Ścieżka nie znaleziona		3 Plik nie znaleziony		4 Nieprawidłowy typ pliku
Typ.	VAR INT												
Wartość:	0 Nie ma błędu												
	1 Ścieżka niedozwolona												
	2 Ścieżka nie znaleziona												
	3 Plik nie znaleziony												
	4 Nieprawidłowy typ pliku												

13	Prawa dostępu niewystarczające												
22	Długość łańcucha znaków zmiennej wynikowej (<wynik>) jest za mała.												
<nazwa pliku>:	<p>Nazwa pliku, z którego mają zostać odczytane informacje o pliku</p> <p>Typ: CHAR[160]</p> <p>Przy podawaniu nazwy pliku należy przestrzegać następujących punktów:</p> <ul style="list-style-type: none"> • Podana nazwa pliku nie może zawierać żadnych spacji ani znaków sterujących (znaki w kodzie ASCII ≤ 32), gdyż w przeciwnym przypadku polecenie FILE... zostanie anulowane z identyfikatorem błędu 1 "Niedozwolona ścieżka". • Nazwę pliku można podać ze ścieżką i rozszerzeniem. <ul style="list-style-type: none"> – Podanie ścieżki <p>Nazwy ścieżek muszą być absolutne, tzn. rozpoczynają się one od "/".</p> <p>Bez podania ścieżki plik jest szukany w aktualnym katalogu (= katalogu wybranego programu).</p> – Rozszerzenie pliku <p>Jeżeli nazwa pliku nie zawiera identyfikatora domeny ("_N_"), jest on odpowiednio uzupełniany.</p> <p>Jeżeli nazwa pliku zawiera podkreślenie "_" jako czwarty znak od końca, wówczas następne trzy znaki są interpretowane jako rozszerzenie pliku. Aby w przypadku wszystkich poleceń plikowych móc stosować tę samą nazwę pliku, np. przez zmienną typu STRING, wolno stosować tylko rozszerzenia _SPF i _MPF.</p> <p>Jeżeli nie jest podane rozszerzenie "_MPF" lub "_SPF", jest automatycznie uzupełniane _MPF.</p> • Długość nazwy pliku może wynosić maksymalnie 32 bajty, długość ścieżki maksymalnie 128 bajtów. <p>Przykład:</p> <pre>"PROFILE" "_N_PROFILE" "_N_PROFILE_MPF" "/_N_MPF_DIR/_N_PROFILE_MPF/"</pre>												
<wynik>:	<p>Zmienna wynikowa (parametr Call-By-Reference)</p> <p>Zmienna, w której jest zapisywana zażądana informacja o pliku.</p> <table border="0" style="width: 100%;"> <tr> <td style="width: 150px;">Typ: VAR CHAR[8]</td> <td style="width: 50px;">przy</td> <td>FILEDATE</td> <td>Format: "dd.mm.yy"</td> </tr> <tr> <td>VAR CHAR[8]</td> <td>przy</td> <td>FILETIME</td> <td>Format: "hh:mm:ss"</td> </tr> <tr> <td>VAR INT</td> <td>przy</td> <td>FILESIZE</td> <td>Wielkość pliku jest wyprowadzana w bajtach.</td> </tr> </table>	Typ: VAR CHAR[8]	przy	FILEDATE	Format: "dd.mm.yy"	VAR CHAR[8]	przy	FILETIME	Format: "hh:mm:ss"	VAR INT	przy	FILESIZE	Wielkość pliku jest wyprowadzana w bajtach.
Typ: VAR CHAR[8]	przy	FILEDATE	Format: "dd.mm.yy"										
VAR CHAR[8]	przy	FILETIME	Format: "hh:mm:ss"										
VAR INT	przy	FILESIZE	Wielkość pliku jest wyprowadzana w bajtach.										

VAR CHAR[5]	przy	FILESTAT Format: "rwxsd" (r: read, w: write, x: execute, s: show, d: delete)
VAR CHAR[32]	przy	FILEINFO Format: "rwxsd nnnnnnnn dd.mm.yy hh:mm:ss"

Przykład

Kod programu	Komentarz
N10 DEF INT ERROR	; Definicja zmiennej błędu.
N20 STRING[32] RESULT	; Definicja zmiennej wynikowej.
N30 FILEINFO(ERROR, "/_N_MPF_DIR/_N_TESTFILE_MPF", RESULT)	; Nazwa pliku z identyfikatorem domeny, rozszerzeniem pliku i podaniem ścieżki.
N40 IF ERROR <>0	; Ewaluacja błędu
N50 MSG("BŁĄD"<<ERROR<<"W PRZYPADKU POLECENIA FILEINFO")	
N60 M0	
N70 ENDIF	
...	

Przykład mógłby w zmiennej wynikowej RESULT dać np. następujący wynik:
"77777 12345678 26.05.00 13:51:30"

1.23 Zaokrąglenie do góry (ROUNDUP)

Funkcja

Przy pomocy funkcji "ROUNDUP" wprowadzane wartości typu REAL (liczby ułamkowe z kropką dziesiętną) mogą być zaokrąglane do góry do najbliższej liczby całkowitej.

Składnia

ROUNDUP (<wartość>)

Znaczenie

ROUNDUP: Polecenie zaokrąglenia wprowadzonej wartości do góry
<wartość>: Wprowadzona wartość typu REAL

Wskazówka

Wprowadzane wartości typu INTEGER (liczba całkowita) są zwracane bez zmiany.

Przykłady

Przykład 1: Różne wprowadzane wartości i wyniki ich zaokrąglenia

Przykład	Wynik zaokrąglenia
ROUNDUP (3.1)	4.0
ROUNDUP (3.6)	4.0
ROUNDUP (-3.1)	-3.0
ROUNDUP (-3.6)	-3.0
ROUNDUP (3.0)	3.0
ROUNDUP (3)	3.0

Przykład 2: ROUNDUP w programie NC

Kod programu

```
N10 X=ROUNDUP (3.5) Y=ROUNDUP (R2+2)
N15 R2=ROUNDUP ($AA_IM [Y])
N20 WHEN X=100 DO Y=ROUNDUP ($AA_IM [X])
...
```

1.24 Technika podprogramów

1.24.1 Informacje ogólne

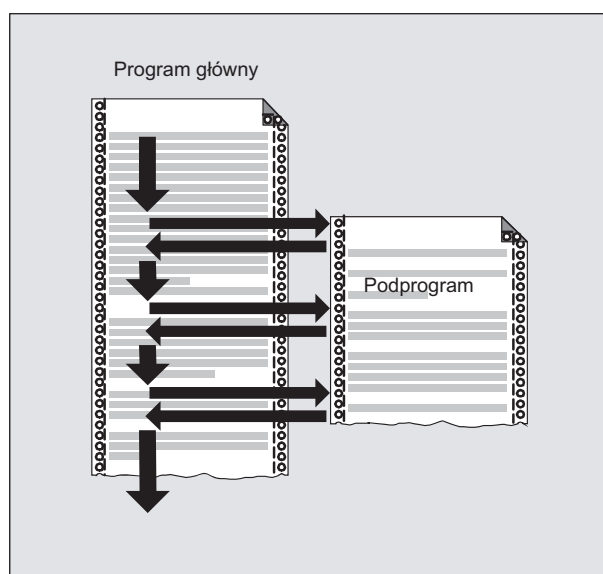
1.24.1.1 Podprogram

Funkcja

Określenie "podprogram" pochodzi jeszcze z okresu, gdy programy obróbki były na stałe podzielone na programy główne i podprogramy. Programami głównymi były przy tym programy obróbki, które były w sterowaniu wybierane do wykonania i następnie uruchamiane. Podprogramami były programy obróbki, które były wywoływane z programu głównego.

Tego stałego podziału już nie ma w dzisiejszym języku SINUMERIK NC. Każdy program obróbki może zasadniczo zostać wybrany i uruchomiony jako program główny, albo zostać wywołany z innego programu obróbki jako podprogram.

Stąd dalej jako podprogram jest określany program obróbki, który jest wywoływany z innego programu obróbki.



Zastosowanie

Tak, jak we wszystkich językach programowania wyższego poziomu, również w języku NC podprogramy są stosowane do tego, by części programu, które są wykorzystywane wielokrotnie, wyłączyć jako programy stanowiące oddzielną całość.

Podprogramy mają następujące zalety:

- Zwiększają przejrzystość i czytelność programów
- Zwiększają jakość dzięki ponownemu stosowaniu przetestowanych części programów
- Umożliwiają tworzenie specyficznych bibliotek obróbkowych
- Oszczędzają miejsce w pamięci

1.24.1.2 Nazwy podprogramów

Zasady nazewnictwa

Przy nadawaniu nazw podprogramom należy przestrzegać następujących zasad:

- Dwa pierwsze znaki muszą być literami (A - Z, a - z).
- Następnymi znakami mogą być w dowolnej kombinacji litery, cyfry (0 - 9) i podkreślenie ("_").
- Wolno użyć maksymalnie 31 znaków.

Wskazówka

W języku SINUMERIK NC **nie** rozróżnia się pisania dużymi i małymi literami.

Rozszerzenia nazwy programu

Nazwa programu nadana przy sporządzaniu programu jest wewnętrznie w sterowaniu rozszerzana o prefiks i postfiks:

- Prefiks: `_N_`
- Postfiks:
 - Programy główne: `_MPF`
 - Podprogramy: `_SPF`

Zastosowanie nazwy programu

Przy stosowaniu nazwy programu, np. przy wywołaniu podprogramu, są możliwe wszystkie kombinacje prefiksu, nazwy programu i postfiksu.

Przykład:

Podprogram o nazwie "SUB_PROG" można uruchomić przez następujące wywołania:

1. `SUB_PROG`
2. `_N_SUB_PROG`
3. `SUB_PROG_SPF`
4. `_N_SUB_PROG_SPF`

Wskazówka

Taka sama nazwa programu głównego i podprogramu

Jeżeli istnieją programy główne (.MPF) i podprogramy (.SPF) o takiej samej nazwie, musi przy stosowaniu nazwy programu w programie obróbki być każdorazowo podawany prefiks, aby jednoznacznie określić program.

1.24.1.3 Kaskadowanie podprogramów

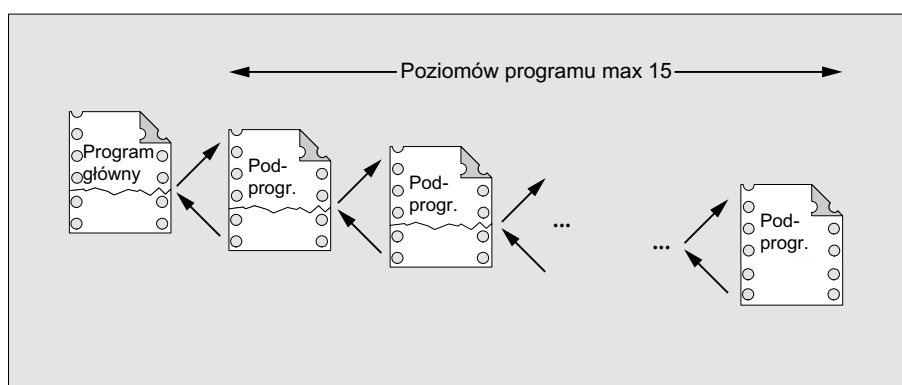
Program główny może wywoływać podprogramy, które wywołują kolejne podprogramy. Przebiegi programów są przez to w sobie zagnieżdżane. Każdy program jest przy tym wykonywany na własnym poziomie programu.

Głębokość kaskadowania

Język NC udostępnia aktualnie 16 poziomów programu. Program główny jest wykonywany zawsze na najwyższym poziomie programu 0. Podprogram jest zawsze wykonywany na kolejnym niższym poziomie programu. Poziom programu 1 jest przez to pierwszym poziomem podprogramu.

Podział poziomów programu:

- Poziom programu 0: Poziom programu głównego
- Poziom programu 1 - 15: poziom podprogramu 1 - 15



Procedury przerwania (ASUP)

Jeżeli w ramach procedury przerwania zostanie wywołany podprogram, jest on wykonywany nie na poziomie programu aktualnie aktywnym w kanale, lecz również na następnym niższym poziomie ($n + 1$). Aby było to możliwe również na najniższym poziomie programu, są w związku z procedurami przerwania do dyspozycji 2 dodatkowe poziomy (16 i 17).

Jeżeli będą potrzebne więcej, niż 2 poziomy programu, musi to zostać explicite uwzględnione w strukturyzacji programu obróbki wykonywanego w kanale. Tzn. wolno wówczas wykorzystać maksymalnie tylko tyle poziomów programu, by pozostały jeszcze wystarczające poziomy do wykonywania przerwania.

Jeżeli wykonanie przerwania wymaga np. 4 poziomów programu, wówczas program obróbki musi być tak strukturyzowany, by zajmował maksymalnie 13. poziom programu. Jeżeli wówczas nastąpi przerwanie, będą do dyspozycji 4 poziomy programu (14 do 17).

Cykle Siemens

Cykle Siemens wymagają 3 poziomów programu. Wywołanie cyklu Siemens musi dlatego nastąpić co najpóźniej w:

- Wykonywanie programu obróbki: poziom programu 12
- Procedura przerwania: poziom programu 14

1.24.1.4 Ścieżka szukania

Przy wywołaniu podprogramu bez ścieżki szukania sterowanie szuka w podanej kolejności w następujących katalogach:

Kolejność	Katalog	Opis
1.	Aktualny katalog	Katalog programu wywołującego
2.	/_N_SPF_DIR /	Globalny katalog podprogramów
3.	/_N_CUS_DIR /	Cykle użytkownika
4.	/_N_CMA_DIR /	Cykle producenta
5.	/_N_CST_DIR /	Cykle standardowe

1.24.1.5 Parametry formalne i aktualne

O parametrach formalnych i aktualnych mówimy w związku z definicją i wywoływaniem podprogramów z przekazaniem parametrów.

Parametry formalne

Przy definicji podprogramu muszą zostać zdefiniowane parametry będące do przekazania podprogramowi, tak zwane parametry formalne, z typem i nazwą parametru.

Parametry formalne definiują przez to interfejs podprogramu.

Przykład:

Kod programu	Komentarz
PROC KONTUR (REAL X, REAL Y)	; Parametry formalne: X i Y obydwu typu REAL
N20 X1=X Y1=Y	; Ruch osi X1 do pozycji X i osi Y1 do pozycji Y
...	
N100 RET	

Parametry aktualne

Przy wywołaniu podprogramu muszą zostać przekazane do podprogramu absolutne wartości lub zmienne, tak zwane parametry aktualne.

Przez to aktualne parametry przy wywołaniu wyposażają interfejs podprogramu w aktualne wartości.

Przykład:

Kod programu	Komentarz
N10 DEF REAL SZEROKOSC	; Definicja zmiennej
N20 SZEROKOSC=20.0	; Przyporządkowanie zmiennych
N30 KONTUR(5.5, SZEROKOSC)	; Wywołanie podprogramu z aktualnymi parametrami: 5.5 i SZEROKOSC
...	
N100 M30	

1.24.1.6 Przekazanie parametrów

Definicja podprogramu z przekazaniem parametrów

Definicja podprogramu z przekazaniem parametrów następuje przy pomocy słowa kluczowego `PROC` i kompletnego wyszczególnienia wszystkich parametrów oczekiwanych od podprogramu.

Niekompletne przekazanie parametrów

Przy wywołaniu podprogramu nie zawsze wszystkie parametry zdefiniowane w interfejsie podprogramu muszą zostać *explicite* przekazane. Gdy parametr zostanie pominięty, jest dla tego parametru przekazywana wartość standardowa "0".

Do jednoznacznego oznaczenia kolejności parametrów muszą w każdym razie zawsze być podawane przecinki jako znaki rozdzielające parametry. Wyjątek stanowi ostatni parametr. Gdy zostanie on przy wywołaniu pominięty, można również pominąć ostatni przecinek.

Przykład:

Podprogram:

Kod programu	Komentarz
<code>PROC SUB_PROG (REAL X, REAL Y, REAL Z)</code>	<code>; Parametry formalne: X, Y i Z</code>
<code>...</code>	
<code>N100 RET</code>	

Program główny:

Kod programu	Komentarz
<code>PROC MAIN_PROG</code>	
<code>...</code>	
<code>N30 SUB_PROG(1.0,2.0,3.0)</code>	<code>; Wywołanie podprogramu z kompletnym przekazaniem parametrów:</code>
	<code>X=1.0, Y=2.0, Z=3.0</code>
<code>...</code>	
<code>N100 M30</code>	

Przykłady wywołania podprogramu w `N30` z niekompletnym przekazaniem parametrów:

<code>N30 SUB_PROG(,2.0,3.0)</code>	<code>; X=0.0, Y=2.0, Z=3.0</code>
<code>N30 SUB_PROG(1.0, ,3.0)</code>	<code>; X=1.0, Y=0.0, Z=3.0</code>
<code>N30 SUB_PROG(1.0,2.0)</code>	<code>; X=1.0, Y=2.0, Z=0.0</code>
<code>N30 SUB_PROG(, ,3.0)</code>	<code>; X=0.0, Y=0.0, Z=3.0</code>
<code>N30 SUB_PROG(, ,)</code>	<code>; X=0.0, Y=0.0, Z=0.0</code>

UWAGA**Przekazanie parametrów Call-by-Reference**

Parametrów, które są przekazywane przez Call-by-Reference, nie wolno pomijać przy wywołaniu podprogramu.

UWAGA**Typ danych AXIS**

Parametrów o typie danych AXIS nie wolno pomijać przy wywołaniu podprogramu.

Sprawdzenie przekazywanych parametrów

Przez zmienną systemową \$P_SUBPAR [n] z $n = 1, 2, \dots$ można sprawdzić w podprogramie, czy parametr został *explicite* przekazany czy pominięty. Indeks n odnosi się do kolejności parametrów formalnych. Indeks $n = 1$ odnosi się do 1. parametru formalnego, indeks $n = 2$ do 2. parametru formalnego, itd.

Poniższy wycinek programu pokazuje na przykład dla 1. parametru formalnego, jak można realizować sprawdzenie:

Programowanie	Komentarz
PROC SUB_PROG (REAL X, REAL Y, REAL Z)	; Parametry formalne: X, Y i Z
N20 IF \$P_SUBPAR[1]==TRUE	; Sprawdzenie 1. parametru formalnego X.
...	; Te akcje są przeprowadzane, gdy parametr formalny X został <i>explicite</i> przekazany.
N40 ELSE	
...	; Te akcje są przeprowadzane, gdy parametr formalny X nie został przekazany.
N60 ENDIF	
...	; Akcje ogólne
N100 RET	

1.24.2 Definicja podprogramu

1.24.2.1 Podprogram bez przekazania parametrów

Funkcja

Przy definicji podprogramów bez przekazania parametrów można pominąć wiersz definicji na początku programu.

Składnia

```
[PROC <nazwa programu>]  
...
```

Znaczenie

PROC:	Polecenie definiujące na początku programu
<nazwa programu>:	Nazwa programu

Przykład

Przykład 1: Podprogram z instrukcją PROC

Kod programu	Komentarz
PROC SUB_PROG	; Wiersz definicji
N10 G01 G90 G64 F1000	
N20 X10 Y20	
...	
N100 RET	; Powrót z podprogramu

Przykład 2: Podprogram bez instrukcji PROC

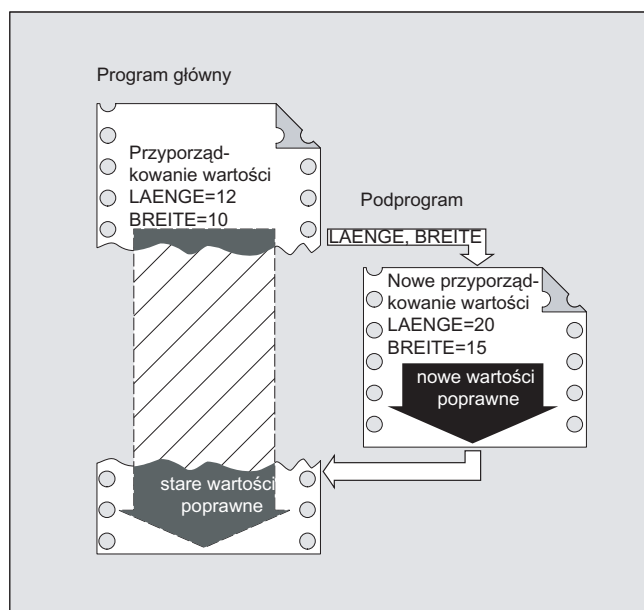
Kod programu	Komentarz
N10 G01 G90 G64 F1000	
N20 X10 Y20	
...	
N100 RET	; Powrót z podprogramu

1.24.2.2 Podprogram z przekazaniem parametrów Call-by-Value (PROC)

Funkcja

Definicja podprogramu z przekazaniem parametrów Call By Value następuje przy pomocy słowa kluczowego `PROC`, po którym następuje nazwa programu i kompletne wyszczególnienie wszystkich oczekiwanych przez podprogram parametrów z typem i nazwą. Polecenie definiujące musi znajdować się w pierwszym wierszu programu.

Przekazanie parametrów Call-by-Value nie ma odwrotnego wpływu na program wywołujący. Program wywołujący przekazuje podprogramowi tylko wartości aktualnych parametrów.



Wskazówka

Może zostać przekazanych maksymalnie 127 parametrów.

Składnia

```
PROC <nazwa_prog> (<typ_par> <par_1> [=<wartość_inicj_1>] {,
par_2 [=<wartość_inicj_1>] })
```

Znaczenie

<code>PROC:</code>	Polecenie definiujące na początku programu
<code><nazwa_prog>:</code>	Nazwa programu
<code><typ_par>:</code>	Typ danych parametru (np. REAL, INT, BOOL)

<par_n>:	Nazwa parametru
<wartość_inicj>:	Wartość opcjonalna do inicjalizacji parametru
	Jeżeli przy wywołaniu podprogramu parametr nie zostanie podany, zostanie mu przyporządkowana wartość inicjalizacyjna.

Przykład

Definicja podprogramu SUB_PROG z trzema parametrami typu REAL z wartościami domyślnymi:

```

Kod programu
PROC SUB_PROG(REAL LENGTH=10.0, REAL WIDTH=20.0, REAL HIGHT=30.0)
...
N100 RET

```

Różne warianty wywołania:

```

Kod programu
PROC MAIN_PROG
  REAL PAR_1 = 100
  REAL PAR_2 = 200
  REAL PAR_3 = 300
  ; warianty wywołania
  SUB_PROG
  SUB_PROG(PAR_1, PAR_2, PAR_3)
  SUB_PROG(PAR_1)
  SUB_PROG(PAR_1, , PAR_3)
  SUB_PROG( , , PAR_3)
N100 RET

```

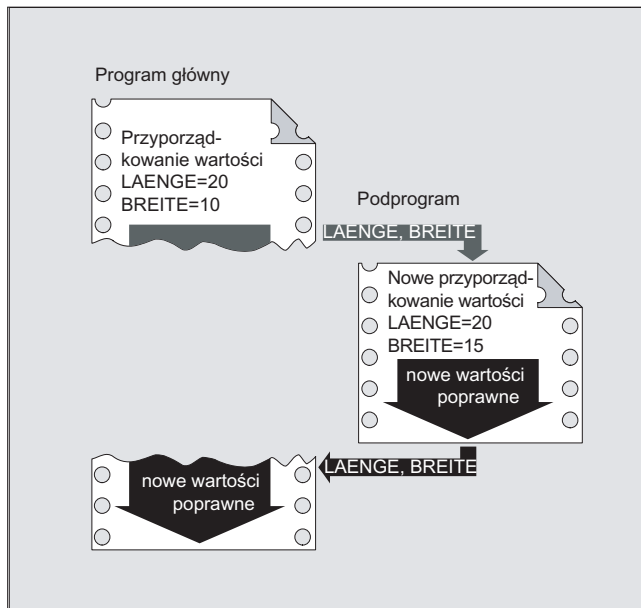
1.24.2.3 Podprogram z przekazaniem parametrów Call-by-Reference (PROC, VAR)

Funkcja

Definicja podprogramu z przekazaniem parametrów Call-by-Reference następuje przy pomocy słowa kluczowego `PROC`, po którym następuje nazwa programu i kompletne wyszczególnienie wszystkich oczekiwanych przez podprogram parametrów ze słowem kluczowym `VAR`, typem i nazwą. Polecenie definiujące musi znajdować się w pierwszym wierszu programu.

Przy przekazaniu parametrów Call-by-Reference mogą być również przekazywane odniesienia do tablic.

Przekazanie parametrów Call-by-Reference nie ma odwrotnego wpływu na program wywołujący. Program wywołujący przekazuje podprogramowi odniesienie do aktualnego parametru i umożliwia mu przez to bezpośredni dostęp do odpowiedniej zmiennej.



Wskazówka

Może zostać przekazanych maksymalnie 127 parametrów.

Wskazówka

Przekazanie parametrów Call-by-Reference jest wymagane tylko wtedy, gdy przekazana zmienna została zdefiniowana w programie wywołującym (LUD). Zmienne globalne dla kanału lub globalne dla NC nie muszą być przekazywane, gdyż dostęp do nich jest możliwy bezpośrednio z podprogramu.

Składnia

```
PROC <nazwa programu> (VAR <typ parametru> <nazwa parametru>, ...)  
PROC <nazwa programu> (VAR <typ tablicy> <nazwa tablicy>  
[<m>,<n>,<o>], ...)
```

Znaczenie

PROC:	Polecenie definiujące na początku programu
VAR:	Słowo kluczowe dla przekazania parametrów przez odniesienie
<nazwa programu>:	Nazwa programu
<typ parametru>:	Typ danych parametru (np. REAL, INT, BOOL)
<nazwa parametru>:	Nazwa parametru
<typ tablicy>:	Typ danych elementów tablicy (np. REAL, INT, BOOL)

<nazwa tablicy>:	Nazwa tablicy
[<m>, <n>, <o>]:	Wielkość tablicy
	Aktualnie są możliwe tablice maksymalnie 3-wymiarowe:
<m>:	Wielkość tablicy dla 1. wymiaru
<n>:	Wielkość tablicy dla 2. wymiaru
<o>:	Wielkość tablicy dla 3. wymiaru

Wskazówka

Nazwa programu podana po słowie kluczowym PROC musi być zgodna z nazwą programu nadaną na interfejsie graficznym.

Wskazówka

Dzięki tablicom o nieokreślonej długości jako parametrom formalnym podprogramy mogą wykonywać tablice o zmiennej długości. W tym celu przy definicji np. tablicy dwuwymiarowej jako parametru formalnego, długość 1. wymiaru nie jest podawana. Przecinek musi być jednak pisany.

Przykład: PROC <nazwa_programu> (VAR REAL TABLICA[,5])

Przykład

Definicja podprogramu z 2 parametrami jako odniesienie do typu REAL:

Kod programu	Komentarz
PROC SUB_PROG(VAR REAL DLUGOSC, VAR REAL SZEROKOSC)	; Parametr 1: odniesienie do typu: REAL, nazwa: DLUGOSC
...	Parametr 2: odniesienie do typu: REAL, nazwa: SZEROKOSC
N100 RET	

1.24.2.4 Zapisywanie modalnych funkcji G (SAVE)

Funkcja

Atrybut `SAVE` powoduje, że aktywne przed wywołaniem podprogramu modalne funkcje G są zapisywane i ponownie uaktywniane po zakończeniu podprogramu.

UWAGA**Przerwanie trybu przechodzenia płynnego**

Jeżeli przy aktywnym trybie przechodzenia płynnego zostanie wywołany podprogram z atrybutem `SAVE`, tryb przechodzenia płynnego jest przerywany na końcu podprogramu (skok powrotny).

Składnia

```
PROC <nazwa podprogramu> SAVE
```

Znaczenie

`SAVE:` Zapisanie modalnych funkcji G przed wywołaniem podprogramu i odtworzenie po zakończeniu podprogramu

Przykład

W podprogramie KONTUR działa modalna funkcja `G91` (wymiar przyrostowy). W programie głównym działa modalna funkcja `G90` (wymiar absolutny). Przez definicję podprogramu z `SAVE` po zakończeniu podprogramu działa ponownie `G90`.

Definicja podprogramu:

Kod programu	Komentarz
PROC KONTUR (REAL WARTOSC1) SAVE	; Definicja podprogramu z parametrem SAVE
N10 G91 ...	; Modalna funkcja G91: Wymiar przyrostowy
N100 M17	; Koniec podprogramu

Program główny:

Kod programu	Komentarz
N10 G0 X... Y... G90	; Modalna funkcja G90: Wymiar absolutny
N20 ...	
...	
N50 KONTUR (12.4)	; Wywołanie podprogramu
N60 X... Y...	; Modalna funkcja G90 reaktywowana przez SAVE

Warunki brzegowe

Frame

Zachowanie się frame odnośnie podprogramów z atrybutem `SAVE` jest zależne od typu frame i może zostać ustawione przez dane maszynowe.

Literatura

Podręcznik działania Funkcje podstawowe; Osie, układy współrzędnych, Frame (K2), Rozdział: "Skok powrotny z podprogramu z `SAVE`"

1.24.2.5 Blokowanie wykonywania pojedynczymi blokami (SBLOF, SBLON)

Funkcja

Blokowanie wykonywania pojedynczymi blokami dla całego programu

Programy oznaczone przez `SBLOF` są przy aktywnym wykonywaniu pojedynczymi blokami wykonywane jak jeden kompletny blok, tzn. dla całego programu wykonywanie pojedynczymi blokami jest zablokowane.

`SBLOF` znajduje się w wierszu `PROC` i działa, aż do końca albo anulowania podprogramu. Poleceniem powrotu następuje decyzja, czy na końcu podprogramu następuje zatrzymanie czy nie:

Powrót z <code>M17</code> :	Stop na końcu podprogramu
Powrót z <code>RET</code> :	Bez zatrzymania na końcu podprogramu

Blokowanie wykonywania pojedynczymi blokami w ramach programu

`SBLOF` musi znajdować się w oddzielnym bloku. Od tego bloku wykonywanie pojedynczymi blokami jest wyłączane do:

- następnego `SBLON`
lub
- do końca aktywnego poziomu podprogramu

Składnia

Blokowanie wykonywania pojedynczymi blokami dla całego programu:

```
PROC ... SBLOF
```

Blokowanie wykonywania pojedynczymi blokami w ramach programu:

```
SBLOF
...
SBLON
```

Znaczenie

PROC:	Pierwsza instrukcja programu
SBLOF:	Polecenie wyłączenia wykonywania pojedynczymi blokami SBLOF może znajdować się w bloku PROC albo w oddzielnym bloku.
SBLON:	Polecenie włączenia wykonywania pojedynczymi blokami SBLON musi znajdować się w oddzielnym bloku.

Warunki brzegowe

- **Blokowanie wykonywania pojedynczymi blokami i wyświetlenie bloku**

Aktualne wyświetlanie bloku może w cyklach/podprogramach być blokowane przy pomocy DISPL0F. Jeżeli DISPL0F zostanie zaprogramowane razem z SBLOF, wówczas przy zatrzymaniu wykonywania pojedynczymi blokami w ramach cyklu/podprogramu wyświetlanie następuje, jak przed wywołaniem cyklu/podprogramu.

- **Blokowanie wykonywania pojedynczymi blokami w ASUP systemowym i ASUP użytkownika**

Gdy zatrzymanie w wykonywaniu pojedynczymi blokami jest blokowane w ASUP systemowym lub ASUP użytkownika przez ustawienia w danej maszynowej MD10702 \$MN_IGNORE_SINGLEBLOCK_MASK (Bit0 = 1 wzgl. Bit1 = 1), wówczas może ono zostać ponownie uaktywnione przez zaprogramowanie SBLON w ASUP.

Gdy zatrzymywanie w wykonywaniu pojedynczymi blokami jest blokowane w ASUP użytkownika przez ustawienie w danej maszynowej MD20117 \$MC_IGNORE_SINGLEBLOCK_ASUP, nie można ponownie uaktywnić zatrzymywania przez zaprogramowanie SBLON w ASUP.

- **Cechy szczególne blokowania wykonywania pojedynczymi blokami przy różnych typach wykonywania pojedynczymi blokami**

Przy aktywnym wykonywaniu pojedynczymi blokami SBL2 (stop po każdym bloku programu obróbki) **nie** następuje zatrzymanie w bloku SBLON, gdy w MD10702 \$MN_IGNORE_SINGLEBLOCK_MASK (uniemożliwienie zatrzymywania w wykonywaniu pojedynczymi blokami) bit 12 jest ustawiony na "1".

Przy aktywnym wykonywaniu pojedynczymi blokami SBL3 (stop po każdym bloku programu obróbki również w cyklu) polecenie SBLOF jest blokowane.

Przykłady

Przykład 1: Blokowanie wykonywania pojedynczymi blokami w ramach programu

Kod programu	Komentarz
N10 G1 X100 F1000	
N20 SBLOF	; Wyłączenie wykonywania pojedynczymi blokami
N30 Y20	
N40 M100	
N50 R10=90	
N60 SBLON	; Ponowne włączenie wykonywania pojedynczymi blokami
N70 M110	
N80 ...	

Zakres między N20 i N60 jest przy wykonywaniu pojedynczymi blokami wykonywany jako jeden krok.

Przykład 2: Cykl powinien dla użytkownika działać jak polecenie

Program główny:

Kod programu
N10 G1 X10 G90 F200
N20 X-4 Y6
N30 CYCLE1
N40 G1 X0
N50 M30

Cykl CYCLE1:

Kod programu	Komentarz
N100 PROC CYCLE1 DISPLOF SBLOF	; Blokowanie wykonywania pojedynczymi blokami
N110 R10=3*SIN(R20)+5	
N120 IF (R11 <= 0)	
N130 SETAL(61000)	
N140 ENDIF	
N150 G1 G91 Z=R10 F=R11	
N160 M17	

Cykl CYCLE1 jest wykonywany przy aktywnym wykonywaniu pojedynczymi blokami, tzn. w celu wykonania CYCLE1 musi zostać naciśnięty przycisk Start.

Przykład 3:

ASUP uruchomiony z PLC w celu uaktywnienia zmienionego przesunięcia punktu zerowego i korekcyj narzędzia powinien być niewidoczny.

Kod programu

```

N100 PROC NV SBLOF DISPLOF
N110 CASE $P_UIFRNUM OF      0 GOTOF _G500
                              1 GOTOF _G54
                              2 GOTOF _G55
                              3 GOTOF _G56
                              4 GOTOF _G57
                              DEFAULT GOTOF END
N120 _G54: G54 D=$P_TOOL T=$P_TOOLNO
N130 RET
N140 _G54: G55 D=$P_TOOL T=$P_TOOLNO
N150 RET
N160 _G56: G56 D=$P_TOOL T=$P_TOOLNO
N170 RET
N180 _G57: G57 D=$P_TOOL T=$P_TOOLNO
N190 RET
N200 END: D=$P_TOOL T=$P_TOOLNO
N210 RET

```

Przykład 4: Przy pomocy MD10702 Bit 12 = 1 nie następuje zatrzymanie**Sytuacja wyjściowa:**

- Wykonywanie pojedynczymi blokami jest aktywne.
- MD10702 \$MN_IGNORE_SINGLEBLOCK_MASK Bit12 = 1

Program główny:

Kod programu	Komentarz
N10 G0 X0	; Zatrzymać w tym wierszu programu obróbki.
N20 X10	; Zatrzymać w tym wierszu programu obróbki.
N30 CYCLE	; Blok ruchu wygenerowany przez cykl.
N50 G90 X20	; Zatrzymać w tym wierszu programu obróbki.
M30	

Cykl CYCLE:

Kod programu	Komentarz
PROC CYCLE SBLOF	; Blokowanie zatrzymywania w wykonywaniu pojedynczymi blokami
N100 R0 = 1	
N110 SBLON	; Z powodu MD10702 bit12=1 w tym wierszu programu obróbki nie następuje zatrzymanie.
N120 X1	; W tym wierszu programu obróbki następuje zatrzymanie.
N140 SBLOF	
N150 R0 = 2	
RET	

Przykład 5: Blokowanie wykonywania pojedynczymi blokami przy kaskadowaniu programu**Sytuacja wyjściowa:**

Wykonywanie pojedynczymi blokami jest aktywne.

Zagnieżdżenie programu:

Kod programu	Komentarz
N10 X0 F1000	; W tym bloku następuje zatrzymanie.
N20 UP1(0)	
PROC UP1(INT _NR) SBLOF	; Blokowanie zatrzymywania w wykonywaniu pojedynczymi blokami.
N100 X10	
N110 UP2(0)	
PROC UP2(INT _NR)	
N200 X20	
N210 SBLON	; Włączenie zatrzymywania w wykonywaniu pojedynczymi blokami.
N220 X22	; W tym bloku następuje zatrzymanie.
N230 UP3(0)	
PROC UP3(INT _NR)	
N300 SBLOF	; Blokowanie zatrzymywania w wykonywaniu pojedynczymi blokami.
N305 X30	
N310 SBLON	; Włączenie zatrzymywania w wykonywaniu pojedynczymi blokami.
N320 X32	; W tym bloku następuje zatrzymanie.
N330 SBLOF	; Blokowanie zatrzymywania w wykonywaniu pojedynczymi blokami.
N340 X34	
N350 M17	; SBLOF jest aktywne.
N240 X24	; W tym bloku następuje zatrzymanie. SBLON jest aktywne.

Kod programu	Komentarz
N250 M17	; W tym bloku następuje zatrzymanie. SBLON jest aktywne.
N120 X12	
N130 M17	; W tym bloku powrotu następuje zatrzymanie. SBLOF instrukcji PROC jest aktywne.
N30 X0	; W tym bloku następuje zatrzymanie.
N40 M30	; W tym bloku następuje zatrzymanie.

Dalsze informacje

Blokada wykonywania pojedynczymi blokami dla podprogramów asynchronicznych

Aby wykonać ASUP w jednym kroku przy wykonywaniu pojedynczymi blokami, musi w ASUP zostać zaprogramowana instrukcja `PROC Z SBLOF`. Obowiązuje to również dla funkcji "edytowalny ASUP systemowy" (MD11610 \$MN_ASUP_EDITABLE).

Przykład dla edytowalnego ASUP systemowego:

Kod programu	Komentarz
N10 PROC ASUP1 SBLOF DISPLOF	
N20 IF \$AC_ASUP=='H200'	
N30 RET	; Bez REPOS przy zmianie trybu pracy
N40 ELSE	
N50 REPOSA	; REPOS we wszystkich pozostałych przypadkach.
N60 ENDIF	

Sterowanie programem w wykonywaniu pojedynczymi blokami

W trybie wykonywania pojedynczymi blokami użytkownik może wykonywać program obróbki pojedynczymi blokami. Istnieją następujące rodzaje ustawienia:

- SBL1: IPO pojedynczymi blokami z zatrzymaniem po każdym bloku zawierającym funkcje maszynowe.
- SBL2: Wykonywanie pojedynczymi blokami z zatrzymaniem po każdym bloku.
- SBL3: Zatrzymanie w cyklu (przez wybór SBL3 jest blokowane polecenie `SBLOF`).

Blokowanie wykonywania pojedynczymi blokami przy kaskadowaniu programu

Jeżeli w podprogramie zaprogramowano `SBLOF` w instrukcji `PROC`, wówczas następuje zatrzymanie do powrotu z podprogramu przy pomocy `M17`. Zapobiega to temu, że w programie wywołującym będzie już wykonywany następny blok. Jeżeli w podprogramie z `SBLOF`, bez `SBLOF` w instrukcji `PROC`, zostanie uaktywnione blokowanie wykonywania pojedynczymi blokami, zatrzymanie następuje dopiero po następnym bloku zawierającym funkcję maszynową w programie wywołującym. Jeżeli jest to niepożądane, musi w podprogramie jeszcze przed powrotem (`M17`) ponownie zostać zaprogramowane `SBLON`. Przy powrocie z `RET` do programu nadrzędnego nie następuje zatrzymanie.

1.24.2.6 Blokowanie wyświetlania aktualnego bloku (DISPLOF, DISPLON, ACTBLOCNO)

Funkcja

Na wyświetlaniu bloku jest standardowo wyświetlany aktualny blok programu. W cyklach wzgl. podprogramach może przy pomocy polecenia `DISPLOF` być blokowane wyświetlanie aktualnego bloku. Zamiast aktualnego bloku jest wówczas wyświetlane wywołanie cyklu lub podprogramu. Poleceniem `DISPLON` można usunąć blokowanie wyświetlania bloku.

`DISPLOF` wzgl. `DISPLON` jest programowane w wierszu programu z instrukcją `PROC` i działa dla całego podprogramu i implicite dla wszystkich podprogramów wywoływanych z tego podprogramu, które nie zawierają polecenia `DISPLON` wzgl. `DISPLOF`. To zachowanie się działa również dla `ASUP`.

Składnia

```
PROC ... DISPLOF
PROC ... DISPLOF ACTBLOCNO
PROC ... DISPLON
```

Znaczenie

<code>DISPLOF:</code>	<p>Polecenie blokowania aktualnego wyświetlania bloku.</p> <p>Usytuowanie: Na końcu wiersza programu z instrukcją <code>PROC</code></p> <p>Działanie: Aż do powrotu z podprogramu albo do końca programu.</p> <p>Wskazówka:</p> <p>Gdy z podprogramu z poleceniem <code>DISPLOF</code> są wywoływane dalsze podprogramy, wówczas również w tych podprogramach jest blokowane wyświetlanie aktualnego bloku, o ile nie jest w nich explicite zaprogramowane <code>DISPLON</code>.</p>
<code>DISPLON:</code>	<p>Polecenie do wyłączenia blokowania wyświetlania aktualnego bloku.</p> <p>Usytuowanie: Na końcu wiersza programu z instrukcją <code>PROC</code></p> <p>Działanie: Aż do powrotu z podprogramu albo do końca programu.</p> <p>Wskazówka:</p> <p>Gdy z podprogramu z poleceniem <code>DISPLON</code> są wywoływane dalsze podprogramy, wówczas również w tych podprogramach jest wyświetlany aktualny blok, o ile nie jest w nich explicite zaprogramowane <code>DISPLOF</code>.</p>
<code>ACTBLOCNO:</code>	<p><code>DISPLOF</code> razem z atrybutem <code>ACTBLOCNO</code> powoduje, że w przypadku alarmu jest wyprowadzany numer aktualnego bloku, w którym wystąpił alarm. Obowiązuje to również wtedy, gdy na niższym poziomie programu jest zaprogramowane tylko <code>DISPLOF</code>.</p> <p>W przypadku <code>DISPLOF</code> bez <code>ACTBLOCNO</code> jest natomiast wyświetlany numer bloku wywołania cyklu wzgl. podprogramu z ostatniego poziomu programu nie oznaczonego przez <code>DISPLOF</code>.</p>

Przykłady

Przykład 1: Blokowanie aktualnego wyświetlania bloku w cyklu

Kod programu	Komentarz
PROC CYCLE (AXIS TOMOV, REAL POSITION) SAVE DISPLOF	; Blokowanie wyświetlania aktualnego bloku. Zamiast tego ma być wyświetlane wywołanie cyklu, np.: CYCLE (X,100.0)
DEF REAL DIFF	; Treść cykli
G01 ...	
...	
RET	; Powrót z podprogramu. Na wyświetleniu bloku jest wyświetlany blok następujący po wywołaniu cyklu.

Przykład 2: Wyświetlanie bloku przy wyprowadzeniu alarmu

Podprogram SUBPROG1 (z ACTBLOCNO):

Kod programu	Komentarz
PROC SUBPROG1 DISPLOF ACTBLOCNO	
N8000 R10 = R33 + R44	
...	
N9040 R10 = 66 X100	; Wyzwolenie alarmu 12080
...	
N10000 M17	

Podprogram SUBPROG2 (bez ACTBLOCNO):

Kod programu	Komentarz
PROC SUBPROG2 DISPLOF	
N5000 R10 = R33 + R44	
...	
N6040 R10 = 66 X100	; Wyzwolenie alarmu 12080
...	
N7000 M17	

Program główny:

Kod programu	Komentarz
N1000 G0 X0 Y0 Z0	
N1010 ...	
...	
N2050 SUBPROG1	; Wyprowadzenie alarmu = "12080 kanał K1 blok N9040 błąd składni w tekście R10="
N2060 ...	
N2350 SUBPROG2	; Wyprowadzenie alarmu = "12080 kanał K1 blok N2350 błąd składni w tekście R10="
...	
N3000 M30	

Przykład 3: Wyłączenie blokowania wyświetlania aktualnego bloku

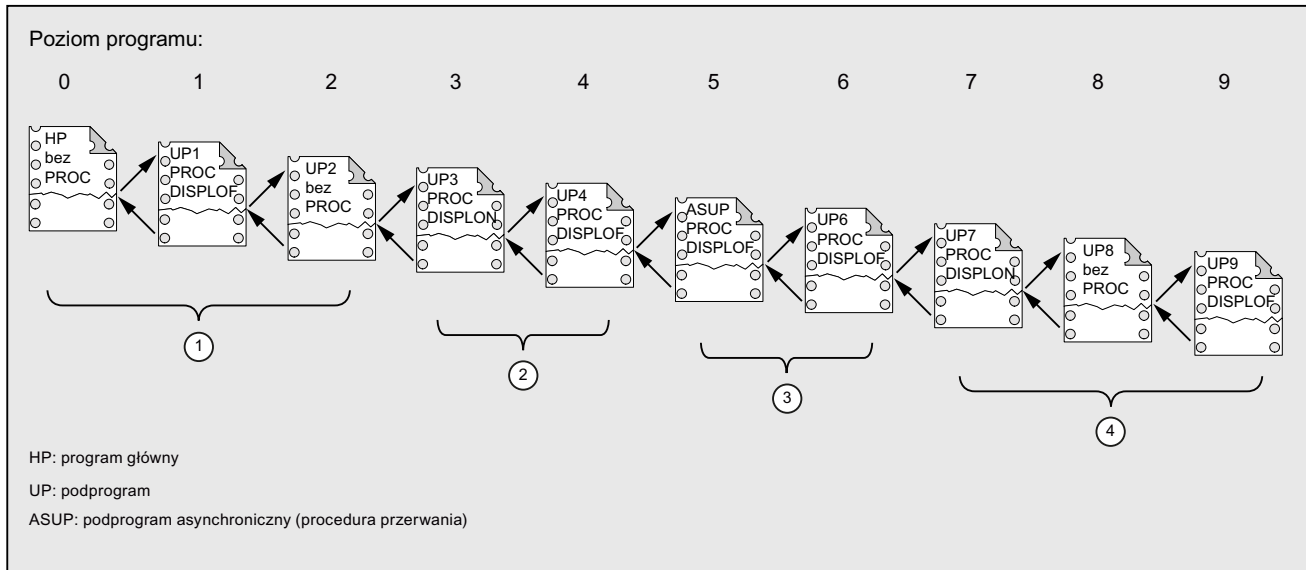
Podprogram SUB1 z blokowaniem:

Kod programu	Komentarz
PROC SUB1 DISPLOF	; Blokowanie wyświetlania aktualnego bloku w podprogramie SUB1. Zamiast tego ma być wyświetlany blok z wywołaniem SUB1.
...	
N300 SUB2	; Wywołanie podprogramu SUB2.
...	
N500 M17	

Podprogram SUB2 bez blokowania:

Kod programu	Komentarz
PROC SUB2 DISPLON	; Zniesienie blokowania wyświetlania aktualnego bloku w podprogramie SUB2.
...	
N200 M17	; Powrót do podprogramu SUB1. W SUB1 wyświetlanie aktualnego bloku jest ponownie blokowane.

Przykład 4: Zachowanie się wyświetlania przy różnych kombinacjach DISPLON/DISPLOF



- ① Na wyświetlaniu aktualnego bloku są wyświetlane wiersze programu obróbki z poziomu programu 0.
- ② Na wyświetlaniu aktualnego bloku są wyświetlane wiersze programu obróbki z poziomu programu 3.
- ③ Na wyświetlaniu aktualnego bloku są wyświetlane wiersze programu obróbki z poziomu programu 3.
- ④ Na wyświetlaniu aktualnego bloku są wyświetlane wiersze programu obróbki z poziomu programu 7/8.

1.24.2.7 Oznakowanie podprogramów z przygotowaniem (PREPRO)

Funkcja

Przy pomocy słowa kluczowego `PREPRO` mogą w rozruchu na końcu wiersza instrukcji `PROC` zostać oznakowane wszystkie pliki.

Wskazówka

Ten rodzaj przygotowania programu jest zależny od odpowiednio ustawionej danej maszynowej. Proszę przestrzegać danych producenta maszyny.

Literatura:

Podręcznik działania Funkcje specjalne; Przygotowanie (V2)

Składnia

`PROC ... PREPRO`

Znaczenie

PREPRO: Słowo kluczowe do oznakowania wszystkich plików przygotowanych w rozruchu, programów NC zapisanych w katalogach cykli

Wczytanie podprogramów z przygotowaniem i wywołanie podprogramu

Zarówno w rozruchu przygotowanych podprogramów z parametrami, jak też przy wywołaniu podprogramu katalogi cykli są traktowane w tej samej kolejności:

1. `_N_CUS_DIR` cykle użytkownika
2. `_N_CMA_DIR` cykle producenta
3. `_N_CST_DIR` cykle standardowe

W przypadku programów NC o tej samej nazwie, a różnym wykonaniu jest uaktywniana instrukcja `PROC` znaleziona jako pierwsza, a inna instrukcja `PROC` jest pomijana bez komunikatu alarmowego.

1.24.2.8 Powrót z podprogramu M17

Funkcja

Na końcu podprogramu znajduje się polecenie powrotu `M17` (wzgl. polecenie programu obróbki `M30`). Powoduje ono skok powrotny do programu wywołującego do bloku następującego po wywołaniu podprogramu.

Wskazówka

`M17` i `M30` są w języku NC traktowane tak samo.

Składnia

```
PROC <nazwa programu>  
...  
M17/M30
```

Warunki brzegowe

Wpływ powrotu z podprogramu na tryb przechodzenia płynnego

Jeżeli `M17` (wzgl. `M30`) jest jedynym poleceniem w bloku programu obróbki, przerywa to aktywny w kanale tryb przechodzenia płynnego..

Aby uniknąć przerwania trybu przechodzenia płynnego, należy napisać `M17` (wzgl. `M30`) w ostatnim bloku ruchu postępowego. Dodatkowo następująca dana maszynowa musi być ustawiona na "0":

`MD20800 $MC_SPF_END_TO_VDI = 0` (bez wyprowadzenia M30/M17 do interfejsu NC/PLC)

Przykład

1. Podprogram z M17 w oddzielnym bloku

Kod programu	Komentarz
N10 G64 F2000 G91 X10 Y10	
N20 X10 Z10	
N30 M17	; Powrót z przerwaniem trybu przechodzenia płynnego.

2. Podprogram z M17 w ostatnim bloku ruchu postępowego

Kod programu	Komentarz
N10 G64 F2000 G91 X10 Y10	
N20 X10 Z10 M17	; Powrót bez przerywania trybu przechodzenia płynnego.

1.24.2.9 Skok powrotny z podprogramu RET**Funkcja**

Zamiast polecenia powrotu M17 można w podprogramie stosować również polecenie RET. RET musi być programowane w oddzielnym bloku programu obróbki. Tak jak M17 RET powoduje skok powrotny do programu wywołującego do bloku następującym po wywołaniu podprogramu.

Wskazówka

Przez zaprogramowanie parametrów można zmienić zachowanie się RET pod względem skoku powrotnego (patrz "Parametryzowalny powrót z podprogramu (RET ...) (Strona 183)").

Zastosowanie

Instrukcji RET należy użyć wtedy, gdy praca z przechodzeniem płynnym G64 (G641 ..., G645) nie powinna być przerywana przez powrót.

Warunek

Polecenie RET można stosować tylko w podprogramach, które nie zostały zdefiniowane z atrybutem SAVE.

Składnia

```
PROC <nazwa programu>
...
RET
```

Przykład

Program główny:

Kod programu	Komentarz
PROC MAIN_PROGRAM	; Początek programu
...	
N50 SUB_PROG	; Wywołanie podprogramu:SUB_PROG
N60 ...	
...	
N100 M30	; Koniec programu

Podprogram:

Kod programu	Komentarz
PROC SUB_PROG	
...	
N100 RET	; Powrót następuje do bloku N60 w programie głównym.

1.24.2.10 Parametryzowany powrót z podprogramu (RET ...)

Funkcja

Powszechnie skok powrotny z podprogramu następuje z końcem podprogramu `RET` lub `M17` do programu, z którego podprogram został wywołany i obróbka jest kontynuowana od wiersza następującego po wywołaniu podprogramu.

Oprócz tego są jednak również przypadki zastosowania, gdzie wykonywanie programu ma być kontynuowane w innym miejscu, np.:

- Kontynuacja wykonywania programu po wywołaniu cykli skrawania w trybie dialektu ISO (według opisu konturu).
- Powrót do programu głównego z dowolnej płaszczyzny podprogramu (również po ASUP) w przypadku reakcji na błąd.
- Skok powrotny przez wiele poziomów programu dla zastosowań specjalnych w cyklach kompilacyjnych i w trybie dialektu ISO.

W takich przypadkach jest programowane polecenie `RET` razem z "parametrami powrotu".

Składnia

```
RET("<blok docelowy>")
RET("<blok docelowy>",<blok po bloku docelowym>)
RET("<blok docelowy>",<blok po bloku docelowym>,<liczba poziomów
powrotu>)
RET("<blok docelowy>",<liczba poziomów powrotu>)
RET("<blok docelowy>",<blok po bloku docelowym>,<liczba poziomów
powrotu>,<powrót do początku programu>)
RET( , ,<liczba płaszczyzn powrotu>,<powrót do początku programu>)
```

Znaczenie

<p>RET:</p> <p><blok docelowy>:</p>	<p>Koniec podprogramu (zastosowanie zamiast M17)</p> <p>Parametr powrotu 1</p> <p>Podaje jako cel skoku ten blok, od którego ma być kontynuowane wykonywanie programu.</p> <p>Gdy parametr powrotu 3 nie jest zaprogramowany, wówczas cel skoku znajduje się w tym programie, z którego został wywołany aktualny podprogram.</p> <p>Możliwymi danymi są:</p> <p>"<Numer bloku>" Numer bloku docelowego</p> <p>"<Znacznik skoku>" Znacznik skoku, który musi być umieszczony w bloku docelowym.</p> <p>"<Łańcuch znaków>" Łańcuch znaków, który musi być znany w programie (np. nazwa programu albo zmiennej).</p> <p>Dla programowania łańcucha znaków w bloku docelowym obowiązują następujące zasady:</p> <ul style="list-style-type: none"> • Spacja na końcu (w odróżnieniu od znacznika skoku, który jest oznaczony przez ":" na końcu). • Przed łańcuchem znaków wolno umieścić tylko jeden numer bloku i/albo jeden znacznik skoku, nie polecenia programowe.
<p><blok po bloku docelowym>:</p>	<p>Parametr powrotu 2</p> <p>Odnosi się do parametru powrotu 1.</p> <p>Typ: INT</p> <p>Wartość: 0 Skok powrotny następuje do bloku, który został podany parametrem powrotu 1.</p> <p style="padding-left: 40px;">> 0 Skok powrotny następuje do bloku, który następuje po bloku podanego parametrem powrotu 1.</p>
<p><liczba płaszczyzn powrotu>:</p>	<p>Parametr powrotu 3</p> <p>Podaje liczbę płaszczyzn, o które ma nastąpić skok powrotny, aby dotrzeć do płaszczyzny programowej, w której ma być kontynuowane wykonywanie programu.</p> <p>Typ: INT</p> <p>Wartość: 1 Program jest kontynuowany na "aktualnym poziomie programu - 1" (jak RET bez parametrów).</p> <p style="padding-left: 40px;">2 Program jest kontynuowany na "aktualnym poziomie programu - 2", tzn. jedna płaszczyzna jest przeskakiwana.</p>

	3	Program jest kontynuowany na "aktualnym poziomie programu - 3", tzn. są przeskakiwane dwie płaszczyzny.
	...	
	Zakres	
	wartości: 1 ... 15	
	Parametr powrotu 4	
	Typ: BOOL	
<skok powrotny do początku programu>:	Wartość: 1	Gdy skok powrotny następuje do programu głównego i tam jest aktywny tryb dialektu ISO , następuje rozgałęzienie na początek programu.

Wskazówka

Przy powrocie z podprogramu z łańcuchem znaków jako informacją dla szukania bloku docelowego następuje w programie wywołującym zawsze najpierw szukanie znacznika skoku.

Gdy cel skoku ma być jednoznacznie zdefiniowany przez łańcuch znaków, łańcuch znaków nie może być zgodny z nazwą znacznika skoku, gdyż w przeciwnym przypadku powrót z podprogramu będzie zawsze wykonywany na znacznik skoku, a nie na łańcuch znaków (patrz przykład 2).

Warunki brzegowe

W przypadku powrotu przez wiele poziomów programu są uwzględniane instrukcje `SAVE` poszczególnych poziomów programu.

Jeżeli przy powrocie przez wiele poziomów programu jest aktywny podprogram modalny i w jednym z przeskakiwanych podprogramów jest zaprogramowane polecenie `MCALL` cofnięcia podprogramu modalnego, wówczas modalny podprogram pozostaje nadal aktywny.

UWAGA

Błąd programowania

Programista musi zwracać uwagę, by przy powrocie przez wiele płaszczyzn programowych kontynuować z prawidłowymi ustawieniami modalnymi. Jest to uzyskiwane np. przez zaprogramowanie odpowiedniego bloku głównego.

Przykłady

Przykład 1: Kontynuacja w programie głównym po wykonaniu ASUP

Programowanie	Komentarz
N10010 CALL "UP1"	; Poziom programu 0 (program główny)
N11000 PROC UP1	; Poziom programu 1
N11010 CALL "UP2"	
N12000 PROC UP2	; Poziom programu 2
...	
N19000 PROC ASUP	; Poziom programu 3 (obróbka ASUP)
...	
N19100 RET("N10900", , \$P_STACK)	; Powrót z podprogramu
N10900	; Kontynuacja w programie głównym
N10910 MCALL	; Wyłączenie podprogramu modalnego.
N10920 G0 G60 G40 M5	; Skorygowanie dalszych ustawień modalnych.

Przykład 2: Łańcuch znaków (<String>) jako informacja dla szukania bloku docelowego

Program główny:

Kod programu	Komentarz
PROC MAIN_PROGRAM	
N1000 DEF INT iVar1=1, iVar2=4	
N1010 ...	
N1200 subProg1	; Wywołanie podprogramu "subProg1"
N1210 M2 S1000 X10 F1000	
N1220	
N1400 subProg2	; Wywołanie podprogramu "subProg2"
N1410 M3 S500 Y20	
N1420 ..	
N1500 lab1: iVar1=R10*44	
N1510 F500 X5	
N1520 ...	
N1550 subprog1: G1 X30	; "subProg1" jest tutaj definiowany jako znacznik skoku.
N1560 ...	
N1600 subProg3	; Wywołanie podprogramu "subProg3"
N1610 ...	
N1900 M30	

Podprogram subProg1:

Kod programu	Komentarz
PROC subProg1	
N2000 R10=R20+100	
N2010 ...	
N2200 RET("subProg2")	; Powrót do programu głównego do bloku N1400

Podprogram subProg2:

Kod programu	Komentarz
PROC subProg2	
N2000 R10=R20+100	
N2010 ...	
N2200 RET("iVar1")	; Powrót do programu głównego do bloku N1500

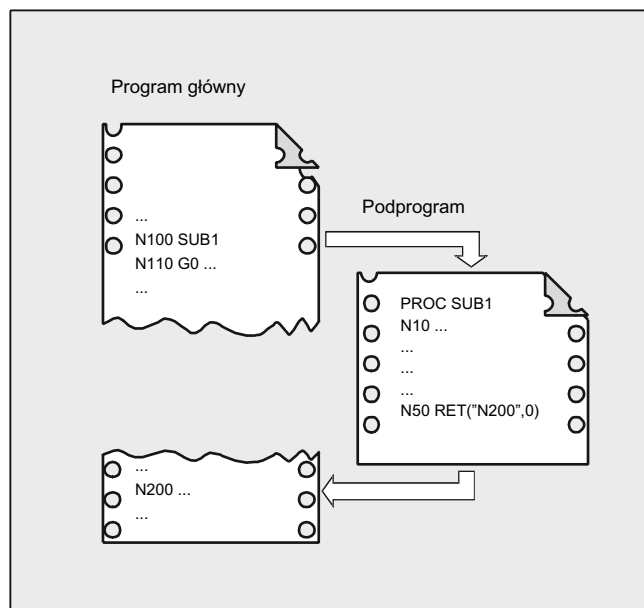
Podprogram subProg3:

Kod programu	Komentarz
PROC subProg3	
N2000 R10=R20+100	
N2010 ...	
N2200 RET("subProg1")	; Powrót do programu głównego do bloku N1550

Dalsze informacje

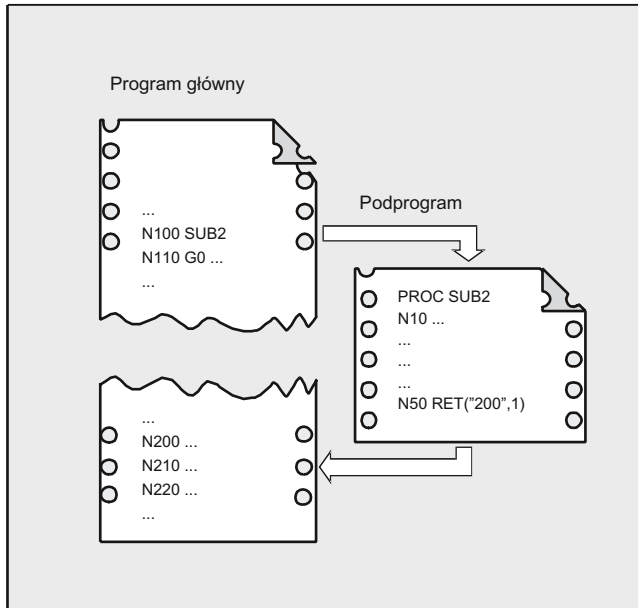
Poniższe grafiki powinny unaocznic różne działania parametrów powrotu 1 do 3.

1. Parametr powrotu 1 = "N200", parametr powrotu 2 = 0



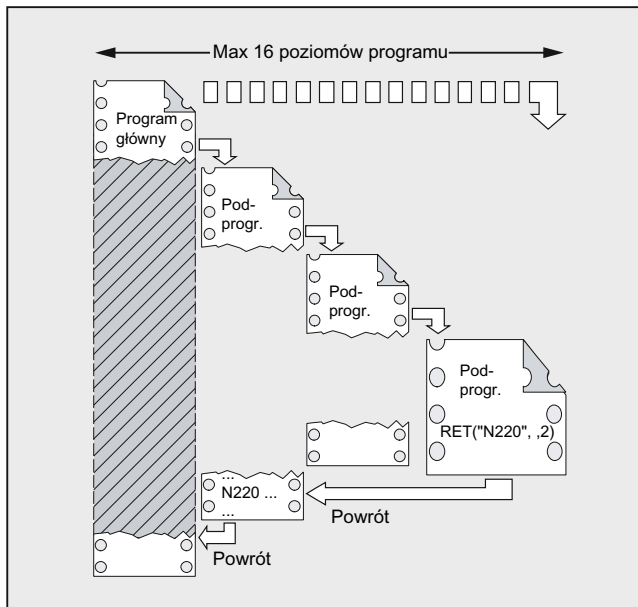
Po poleceniu RET wykonywanie jest kontynuowane od bloku N200 w programie głównym.

2. Parametr powrotu 1 = "N200", parametr powrotu 2 = 1



Po poleceniu `RET` wykonywanie programu jest kontynuowane od bloku `N210`, który w programie głównym następuje po bloku `N200`.

3. Parametr powrotu 1 = "N220", parametr powrotu 3 = 2



Po poleceniu `RET` następuje skok powrotny o dwa poziomy programu i wykonywanie programu jest kontynuowane od bloku `N220`.

1.24.3 Wywołanie podprogramu

1.24.3.1 Wywołanie podprogramu bez przekazania parametrów

Funkcja

Wywołanie podprogramu następuje albo z adresem L i numerem podprogramu albo przez podanie nazwy programu.

Również program główny może zostać wywołany jako podprogram. Wstawiony do programu głównego koniec programu M2 albo M30 jest w tym przypadku traktowany jak M17 (koniec programu z powrotem do programu wywołującego).

Wskazówka

Odpowiednio podprogram może również zostać uruchomiony jako program główny

Strategia szukania przez sterowanie:

Czy istnieje *_MPF?

Czy istnieje *_SPF?

Z tego wynika: Jeżeli nazwa podprogramu wywoływanego jest identyczna z nazwą programu głównego, wówczas jest ponownie wywoływany program wywołujący. Tego z reguły nie pożądanego zjawiska należy uniknąć przez jednoznaczny wybór nazw programów i podprogramów.

Wskazówka

Podprogramy, które nie wymagają przekazania parametrów, mogą też być wywoływane z pliku inicjalizacyjnego.

Składnia

L<numer>/<nazwa programu>

Wskazówka

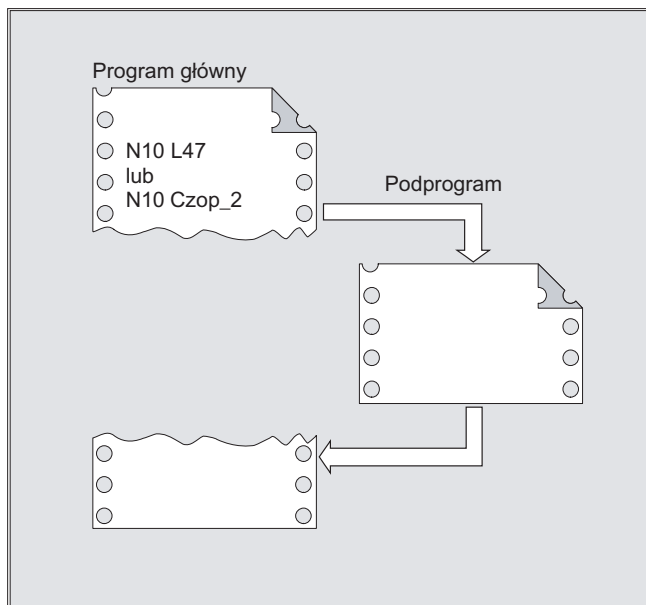
Wywołanie podprogramu musi zawsze być programowane w oddzielnym bloku NC.

Znaczenie

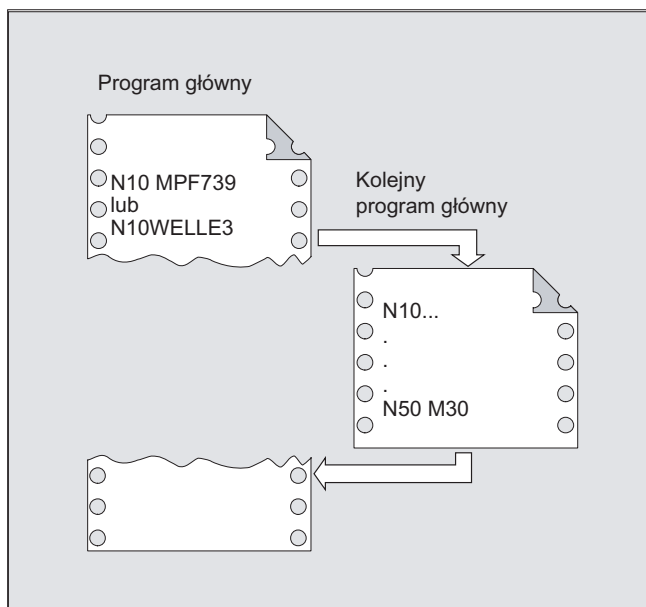
L:	Adres dla wywołania podprogramu
<numer>:	Numer podprogramu
Typ:	INT
Wartość:	Maksymalnie 7 miejsc dziesiętnych
	Uwaga:
	Zera na początku mają znaczenie przy nazewnictwie (⇒ L123, L0123 i L00123 są trzema różnymi podprogramami)
<nazwa programu>:	Nazwa podprogramu (lub programu głównego)

Przykłady

Przykład 1: Wywołanie podprogramu bez przekazania parametrów



Przykład 2: Wywołanie programu głównego jako podprogramu



1.24.3.2 Wywołanie podprogramu z przekazaniem parametrów (EXTERN)

Funkcja

Przy wywołaniu podprogramu z przekazaniem parametrów można bezpośrednio przekazywać zmienne albo wartości (nie w przypadku parametrów `VAR`).

Podprogramy z przekazaniem parametrów muszą przed wywołaniem zostać podane w programie głównym przez `EXTERN` (np. na początku programu). Podawane są przy tym nazwa podprogramu i typy zmiennych w kolejności ich przekazania.

UWAGA

Niebezpieczeństwo pomyłki

Zarówno typy zmiennych jak i kolejność przekazania muszą być zgodne z definicjami, które zostały uzgodnione w nazwie podprogramu pod `PROC`. Nazwy parametrów mogą być różne w programie głównym i podprogramie.

Składnia

```
EXTERN <nazwa programu>(<typ_par1>,<typ_par2>,<typ_par3>)
...
<nazwa_programu>(<wartość_par1>,<wartość_par2>,<wartość_par3>)
```

Wskazówka

Podprogram musi zawsze być programowany w oddzielnym bloku NC.

Znaczenie

<nazwa programu>:

EXTERN:

<typ_par1>,<typ_par2>,<typ_par3>:

<wartość_par1>,<wartość_par2>,<wartość_par3>:

Nazwa podprogramu

Słowo kluczowe do podania podprogramu z przekazaniem parametrów.

Uwaga:

`EXTERN` musi zostać podane tylko wtedy, gdy podprogram znajduje się w katalogu obrabianego przedmiotu albo w globalnym katalogu podprogramów. Cykle nie muszą być deklarowane jako `EXTERN`.

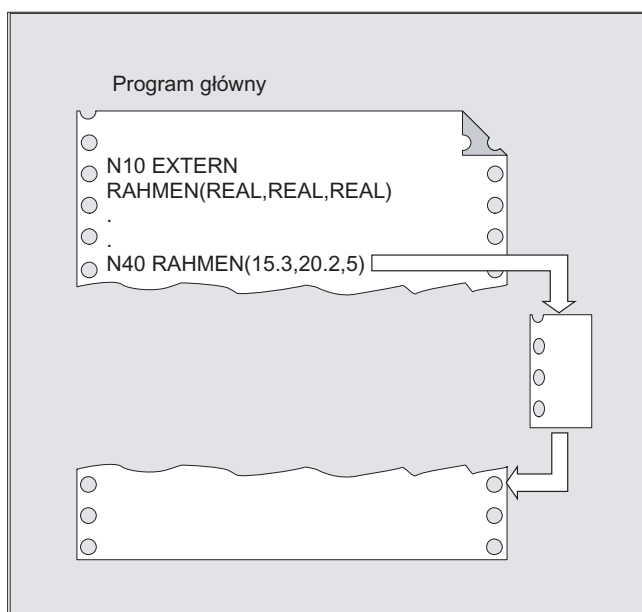
Typy zmiennych przekazywanych parametrów w kolejności przekazania

Wartości zmiennych dla przekazywanych parametrów

Przykłady

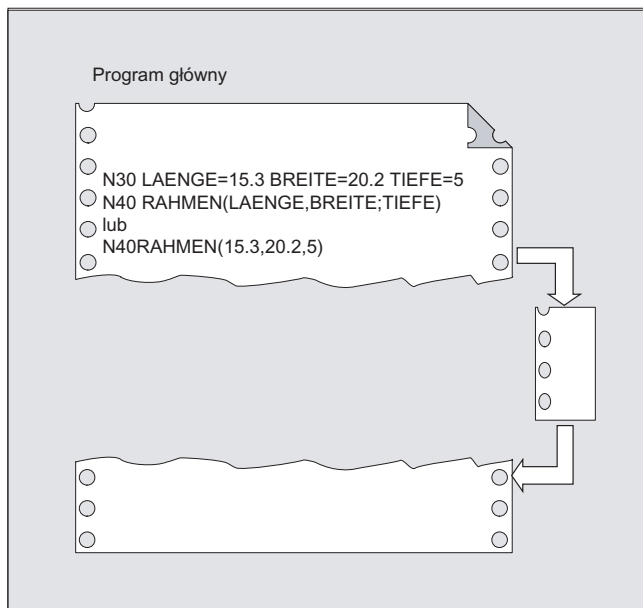
Przykład 1: Wywołanie podprogramu za pomocą poprzedzającego komunikatu

Kod programu	Komentarz
N10 EXTERN RAMKA (REAL, REAL, REAL)	; Podanie podprogramu.
...	
N40 RAMKA (15.3, 20.2, 5)	; Wywołanie podprogramu z przekazaniem parametrów



Przykład 2: Wywołanie podprogramu bez komunikatu

Kod programu	Komentarz
N10 DEF REAL DLUGOSC, SZEROKOSC, GŁĘBOKOSC	
N20 ...	
N30 DŁUGOSC=15.3 SZEROKOSC=20.2 GŁĘBOKOSC=5	
N40 RAMKA (DŁUGOSC, SZEROKOSC, GŁĘBOKOSC)	; lub: N40 RAMKA (15.3, 20.2, 5)



1.24.3.3 Liczba powtórzeń programu (P)

Funkcja

Jeżeli podprogram ma być wykonywany kolejno wiele razy, można w bloku z wywołaniem podprogramu pod adresem P zaprogramować żądaną liczbę powtórzeń programu.

! OSTROŻNIE

Wywołanie podprogramu z powtórzeniem programu i przekazaniem parametrów

Parametry są przekazywane tylko przy wywołaniu programu wzgl. pierwszym przebiegu. Dla dalszych powtórzeń parametry pozostają bez zmian. W przypadku, gdy przy powtórzeniach programu chcemy zmieniać parametry, należy w podprogramie ustalić odpowiednie uzgodnienia.

Składnia

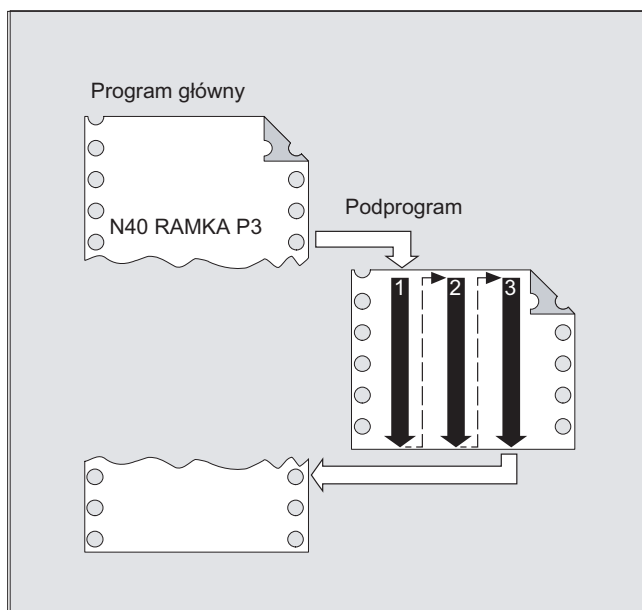
<nazwa programu> P<wartość>

Znaczenie

<nazwa programu>:	Wywołanie podprogramu
P:	Adres dla programowania powtórzeń programu
<wartość>:	Liczba powtórzeń programu
Typ:	INT
Zakres wartości:	1 ... 9999 (bez znaku liczby)

Przykład

Kod programu	Komentarz
...	
N40 RAMKA P3	; Podprogram RAMKA ma być wykonywany kolejno 3 razy.
...	



1.24.3.4 Modalne wywołanie podprogramu (MCALL)

Funkcja

W przypadku modalnego wywołania podprogramu przy pomocy `MCALL` podprogram jest automatycznie wywoływany i wykonywany po każdym bloku z ruchem po torze. Można przez to zautomatyzować wywoływanie podprogramów, które powinny być wykonywane w różnych pozycjach obrabianego przedmiotu, na przykład wykonywanie układów wierconych otworów.

Wyłączenie funkcji następuje przy pomocy `MCALL` bez wywołania podprogramu albo przez zaprogramowanie nowego modalnego wywołania nowego podprogramu.

Wskazówka

W jednym przebiegu programu może jednocześnie działać tylko jedno wywołanie `MCALL`. Parametry są przekazywane tylko jeden raz przy wywołaniu `MCALL`.

Podprogram modalny jest wywoływany w następujących sytuacjach, również bez programowania ruchu:

- Przy programowaniu adresów `S` i `F` gdy jest aktywne `G0` lub `G1`.
- Gdy `G0/G1` zostało zaprogramowane samo w bloku albo z dalszymi `G-Code`.

Wskazówka

W ASUPach, przez które zostało przerwane wykonywanie programu (patrz "Procedura przerwania (ASUP) (Strona 123)"), **nie** są wykonywane modalne wywołania podprogramu.

Natomiast ASUPy, które zostały uruchomione ze stanu Reset, zachowują się odnośnie modalnych wywołań podprogramów, jak normalne programy obróbki.

Składnia

```
MCALL <nazwa programu>
```

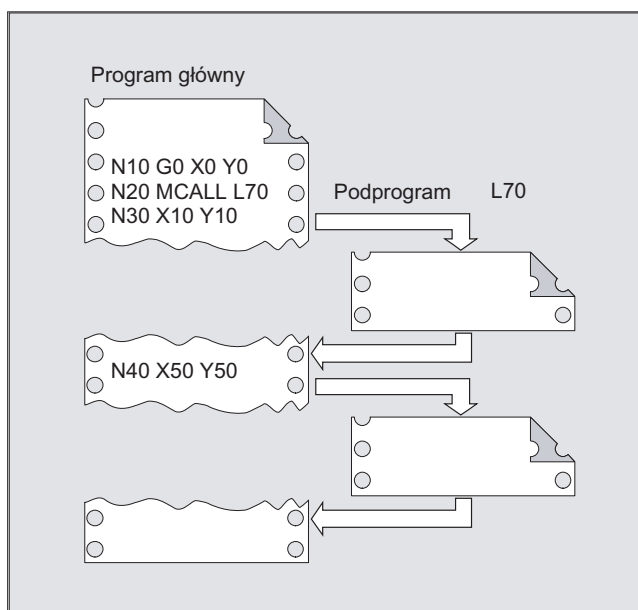
Znaczenie

MCALL:	Polecenie modalnego wywołania podprogramu
<nazwa programu>:	Nazwa podprogramu

Przykłady

Przykład 1:

Kod programu	Komentarz
N10 G0 X0 Y0	
N20 MCALL L70	; Modalne wywołanie podprogramu.
N30 X10 Y10	; Następuje ruch do zaprogramowanej pozycji, a następnie wykonanie podprogramu L70.
N40 X50 Y50	; Następuje ruch do zaprogramowanej pozycji, a następnie wykonanie podprogramu L70.



Przykład 2:

Kod programu
N10 G0 X0 Y0
N20 MCALL L70
N30 L80

W tym przykładzie następne bloki NC z zaprogramowanymi osiami biorącymi udział w tworzeniu konturu znajdują się w podprogramie L80. L70 jest wywoływany przez L80.

1.24.3.5 Pośrednie wywołanie podprogramu (CALL)

Funkcja

W zależności od danych warunków mogą w jednym miejscu być wywoływane różne podprogramy. W tym celu nazwa podprogramu jest zapisywana w zmiennej typu STRING. Wywołanie podprogramu następuje przy pomocy `CALL` i nazwy zmiennej.

Wskazówka

Pośrednie wywołanie podprogramu jest możliwe tylko dla podprogramów bez przekazania parametrów. W celu bezpośredniego wywołania podprogramu należy zapisać nazwę w stałej STRING.

Składnia

`CALL <nazwa programu>`

Znaczenie

`CALL:` Polecenie pośredniego wywołania podprogramu
`<nazwa programu>:` Nazwa podprogramu (zmienna albo stała)
 Typ: STRING

Przykład

Wywołanie bezpośrednio ze stałą STRING:

Kod programu	Komentarz
...	
<code>CALL "/_N_WKS_DIR/_N_SUBPROG_WPD/_N_CZESC1_SPF"</code>	; Bezpośrednie wywołanie podprogramu TEIL1 za pomocą CALL.
...	

Wywołanie pośrednie przez zmienną:

Kod programu	Komentarz
...	
<code>DEF STRING[100] PROGNAME</code>	; Definicja zmiennej.
<code>PROGNAME="/_N_WKS_DIR/_N_SUBPROG_WPD/_N_CZESC1_SPF"</code>	; Przyporządkowanie podprogramu TEIL1 do zmiennej PROGNAME.
<code>CALL PROGNAME</code>	; Pośrednie wywołanie podprogramu TEIL1 poprzez CALL i zmienną PROGNAME.
...	

1.24.3.6 Pośrednie wywołanie podprogramu z podaniem części programu do wykonania (CALL BLOCK ... TO ...)

Funkcja

Przy pomocy `CALL` i kombinacji słów kluczowych `BLOCK... TO` podprogram jest pośrednio wywoływany i wykonywana jego część oznaczona znacznikiem startowym i znacznikiem końcowym.

Składnia

```
CALL <nazwa programu> BLOCK <znacznik startowy> TO <znacznik
końcowy>
CALL BLOCK <znacznik startowy> TO <znacznik końcowy>
```

Znaczenie

<code>CALL:</code>	Polecenie pośredniego wywołania podprogramu
<code><nazwa programu>:</code>	Nazwa podprogramu (zmienna lub stała), która zawiera będącą do wykonania część programu (podanie opcjonalne). Typ: STRING
	Wskazówka: Gdy <code><nazwa programu></code> nie jest zaprogramowana, jest w aktualnym programie szukana i wykonywana jego część oznaczona <code><znacznikiem startowym></code> i <code><znacznikiem końcowym></code> .
<code>BLOCK ... TO ... :</code>	Kombinacja słów kluczowych dla pośredniego wykonania części programu
<code><znacznik startowy>:</code>	Zmienna, która odsyła do początku będącej do wykonania części programu. Typ: STRING
<code><znacznik końcowy>:</code>	Zmienna, która odsyła do końca będącej do wykonania części programu. Typ: STRING

Przykład

Program główny:

Kod programu	Komentarz
...	
DEF STRING[20] ETYKIETA_STARTOWA, ETYKIETA_KONCOWA	; Definicja zmiennych dla znacznika startowego i znacznika końcowego.
ETYKIETA_STARTOWA="ETYKIETA_1"	
ETYKIETA_KONCOWA="ETYKIETA_2"	
...	

Kod programu	Komentarz
CALL "CONTUR_1" SEGMENT ETYKIETA_STARTOWA DO ETYKIETA_KONCOWA	; Pośrednie wywołanie podprogramu i oznaczenie będącej do wykonania części programu.
...	

Podprogram:

Kod programu	Komentarz
PROC CONTUR_1 ...	
ETYKIETA_1	; Znacznik startowy: początek wykonywania części programu
N1000 G1 ...	
...	
ETYKIETA_2	; Znacznik końcowy: koniec wykonywania części programu
...	

1.24.3.7 Pośrednie wywołanie programu programowanego w języku ISO (ISOCALL)

Funkcja

Przy pomocy pośredniego wywołania programu `ISOCALL` może zostać wywołany program zaprogramowany w języku ISO. Jest przy tym uaktywniany tryb ISO ustawiony w danych maszynowych. Na końcu programu działa ponownie pierwotny tryb obróbki. Jeżeli w danych maszynowych nie jest ustawiony tryb ISO, wywołanie podprogramu następuje w trybie Siemens.

Dalsze informacje dot. trybu ISO patrz:

Literatura:

Opis działania Dialekty ISO

Składnia

`ISOCALL <nazwa programu>`

Znaczenie

<code>ISOCALL:</code>	Słowo kluczowe do pośredniego wywołania podprogramu, przy pomocy którego jest uaktywniany tryb ISO ustawiony w danych maszynowych
<code><nazwa programu>:</code>	Nazwa programu zaprogramowanego w języku ISO (zmienna lub stała typu STRING)

Przykład: Wywołanie konturu z zaprogramowaniem cykli z trybu ISO

Kod programu	Komentarz
0122_SPF	; Opis konturu w trybie ISO
N1010 G1 X10 Z20	
N1020 X30 R5	
N1030 Z50 C10	
N1040 X50	
N1050 M99	
N0010 DEF STRING[5] PROGNAME = "0122"	; Program obróbki (cykl) Siemens
...	
N2000 R11 = \$AA_IW[X]	
N2010 ISOCALL PROGNAME	
N2020 R10 = R10+1	; Wykonanie programu 0122.spf w trybie ISO
...	
N2400 M30	

1.24.3.8 Wywołanie podprogramu z podaniem ścieżki i parametrami (PCALL)

Funkcja

Przy pomocy `PCALL` można wywoływać podprogramy z absolutnym podaniem ścieżki i przekazaniem parametrów.

Składnia

`PCALL <ścieżka/nazwa_programu>(<parametr 1>, ..., <parametr n>)`

Znaczenie

<code>PCALL:</code>	Słowo kluczowe do wywołania podprogramu z absolutnym podaniem ścieżki.
<code><ścieżka/nazwa programu>:</code>	Absolutne podanie ścieżki rozpoczynające się od "/", łącznie z nazwą podprogramu. Jeżeli nie podano ścieżki absolutnej, <code>PCALL</code> zachowuje się jak standardowe wywołanie podprogramu z identyfikatorem programu. Identyfikator programu jest podawany bez przedrostka <code>_N_</code> i bez rozszerzenia. Jeżeli nazwa programu ma być programowana z przedrostkiem i rozszerzeniem, wówczas musi ona <i>explicitie</i> zostać zadeklarowana z przedrostkiem i rozszerzeniem przy pomocy polecenia <code>EXTERN</code> .
<code><parametr 1>, ...:</code>	Aktualny parametr odpowiednio do instrukcji <code>PROC</code> podprogramu.

Przykład

Kod programu

```
PCALL /_N_WKS_DIR/_N_WALEK_WPD/WALEK(parametr1,parametr2,...)
```

1.24.3.9 Rozszerzenie ścieżki szukania przy wywołaniach podprogramu (CALLPATH)

Funkcja

Przy pomocy polecenia `CALLPATH` można rozszerzyć ścieżkę szukania dla wywołań podprogramów.

Przez to mogą być wywoływane również podprogramy z nie wybranego katalogu obrabianego przedmiotu, bez podania kompletnej, absolutnej nazwy ścieżki podprogramu.

Rozszerzenie ścieżki szukania następuje przed wpisem dla cykli użytkownika (`_N_CUS_DIR`).

Następujące zdarzenia ponownie cofają wybór rozszerzenia ścieżki szukania:

- `CALLPATH` ze spacją
- `CALLPATH` bez parametrów
- Koniec programu obróbki
- Reset

Składnia

```
CALLPATH("<nazwa ścieżki>")
```

Znaczenie

`CALLPATH`: Słowo kluczowe do programowanego rozszerzenia ścieżki szukania. Jest programowane we oddzielnym wierszu programu obróbki.

`<nazwa ścieżki>`: Stała albo zmienna typu `STRING`. Zawiera absolutne podanie ścieżki katalogu, o który ścieżka szukania ma zostać rozszerzona. Podanie ścieżki rozpoczyna się od `/`. Ścieżka musi być podana kompletnie z przedrostkami i przyrostkami. Maksymalna długość ścieżki wynosi 128 bajtów.

Jeżeli `<nazwa ścieżki>` zawiera spację albo zostanie wywołane `CALLPATH` bez parametrów, instrukcja ścieżki szukania jest cofana.

Przykład

Kod programu

CALLPATH ("/_N_WKS_DIR/_N_MYWPD_WPD")

Jest przez to ustawiana następująca ścieżka szukania (pozycja 5. jest nowa):

1. Aktualny katalog/identyfikator podprogramu
2. Aktualny katalog/identyfikator podprogramu_SPF
3. Aktualny katalog/identyfikator podprogramu_MPF
4. /_N_SPF_DIR/identyfikator podprogramu_SPF
5. **/_N_WKS_DIR/_N_MYWPD_WPD/identyfikator podprogramu_SPF**
6. /_N_CUS_DIR/identyfikator podprogramu_SPF
7. /_N_CMA_DIR/identyfikator podprogramu_SPF
8. /_N_CST_DIR/identyfikator podprogramu_SPF

Warunki brzegowe

- CALLPATH sprawdza, czy zaprogramowana nazwa ścieżki rzeczywiście istnieje. W przypadku błędu wykonywanie programu obróbki jest przerywane z alarmem bloku korekcyjnego 14009.
- CALLPATH można również programować w plikach INI. Działa ono wówczas przez okres wykonywania pliku INI (plik WPD-INI albo program inicjalizacyjny dla aktywnych danych NC np. frame w 1. kanale _N_CH1_UFR_INI). Następnie ścieżka szukania jest ponownie cofana.

1.24.3.10 Wykonywanie podprogramu zewnętrznego (840D sl) (EXTCALL)

Funkcja

Przy pomocy polecenia `EXTCALL` można wykonywać program obróbki doładowując go z pamięci zewnętrznej.

Jako pamięć zewnętrzną są do dyspozycji:

- Stacja lokalna
- Stacja sieciowa
- Stacja USB

Wskazówka

Stacja USB

Jako interfejs do wykonywania programu zewnętrznego znajdującego się w stacji USB wolno stosować tylko interfejsy USB na pulpicie obsługi wzgl. TCU.

Wskazówka**Nie stosować USB-FlashDrive**

Jest zalecane, by do wykonywania podprogramu zewnętrznego nie stosować USB-FlashDrive. Przerwanie komunikacji z USB-FlashDrive podczas wykonywania programu obróbki w wyniku problemów na stykach, wypadnięcia, przerwanie w wyniku uderzenia albo pomyłkowe wyciągnięcie, prowadzi do natychmiastowego zatrzymania obróbki. Może przy tym nastąpić uszkodzenie narzędzia i/albo obrabianego przedmiotu.

Ustawienie domyślne ścieżki programu zewnętrznego

Ścieżka do zewnętrznego katalogu programów może zostać domyślnie ustawiona przy pomocy danej ustawczej:

SD42700 \$SC_EXT_PROG_PATH

Ze ścieżki i identyfikatora programu, podanego przy wywołaniu `EXTCALL` wynika ścieżka całkowita wywoływanego programu.

Wskazówka**Parametry**

Przy wywoływaniu programu zewnętrznego nie można mu przekazać żadnych parametrów.

Składnia

```
EXTCALL("<ścieżka>:<nazwa programu>")
```

Znaczenie

EXTCALL:

```
"<ścieżka><nazwa programu>":
```

Polecenie do wywołania podprogramu zewnętrznego
Stała/Zmienna typu STRING

```
<ścieżka>:
```

Absolutne lub względne podanie ścieżki (**opcjonalnie**)

```
<nazwa programu>:
```

Nazwa programu jest podawana bez przedrostka "_N_".
Rozszerzenie pliku ("MPF", "SPF") można dołączyć do nazwy programu przy pomocy znaku "_" lub "." (**opcjonalnie**).

Przykład:

```
"WALEK"  
"WALEK_SPF"  
"WALEK.SPFF"
```

Podanie ścieżki: Skróty

Następujące oznaczenia skrótowe można stosować przy podaniu ścieżki:

- Stacja lokalna: "LOCAL_DRIVE:"
- Karta CF: "CF_CARD:"
- Stacja USB (pulpit obsługi): "USB:"

Skróty "CF_CARD:" i "LOCAL_DRIVE": są stosowane alternatywnie.

Przykład

Wykonywanie ze stacji lokalnej

Program główny "MAIN.MPF" znajduje się w pamięci NC i jest wybrany do wykonania.

Podprogram "SP_1"

Podprogram zewnętrzny "SP_1.SPF" wzgl. "SP_1.MPF" znajduje się w stacji lokalnej w katalogu "/user/sinumerik/data/prog/WKS.DIR/WST1.WPD".

Ścieżkę do zewnętrznego katalogu programów należy ustawić przy pomocy:

```
SD42700 $SC_EXT_PROG_PATH = LOCAL_DRIVE:WKS.DIR/WST1.WPD
```

Wskazówka

Podanie ścieżki do wywołania podprogramu zewnętrznego:

- Bez zastosowania ustawienia domyślnego: "LOCAL_DRIVE:WKS.DIR/WST1.WPD/SP_1"
 - Z zastosowaniem ustawienia domyślnego: "SP_1"
-

Podprogram "SP_2"

Podprogram zewnętrzny "SP_2.SPF" wzgl. "SP_2.MPF" znajduje się w katalogu WKS.DIR/WST1.WPD stacji USB. Ustawienie domyślne ścieżki do zewnętrznego katalogu programów jest stosowane dla ścieżki podprogramu "SP_1" i w programie głównym również nie jest przepisywane. Dlatego przy wywołaniu podprogramu "SP_2" musi zostać podana kompletna ścieżka.

Program główny "MAIN"

Kod programu

```
N010 PROC MAIN
N020 ...
N030 EXTCALL("SP_1")
N030 EXTCALL("USB:WKS.DIR/WST1.WPD/SP_2")
N050 ...
N060 M30
```

Dalsze informacje

Wywołanie EXTCALL z absolutnym podaniem ścieżki

Gdy podprogram pod podaną ścieżką istnieje, wówczas jest wykonywany po wywołaniu EXTCALL. Gdy podprogram pod podaną ścieżką nie istnieje, wówczas po wywołaniu EXTCALL wykonywanie programu jest przerywane.

Wywołanie EXTCALL ze względnym podaniem ścieżki / bez podania ścieżki

W przypadku wywołania EXTCALL ze względnym podaniem ścieżki lub bez jej podania istniejące pamięci programów są przeszukiwane według następującego wzorca:

1. Gdy w SD42700 \$SC_EXT_PROG_PATH jest domyślnie ustawiona ścieżka, następuje najpierw szukanie rozpoczynając od tej ścieżki według danej w wywołaniu EXTCALL (nazwa programu ew. ze względnym podaniem ścieżki). Ścieżka absolutna wynika wówczas przez powiązanie znaków:
 - z domyślnie ustawionego podania ścieżki w SD42700 \$SC_EXT_PROG_PATH
 - Znak rozdzielający "/"
 - Podanie ścieżki i nazwa podprogramu w poleceniu EXTCALL
2. Gdy podprogram nie został znaleziony pod 1., są przeszukiwane katalogi pamięci użytkownika.

Suchanie kończy się, gdy podprogram został po raz pierwszy znaleziony. Jeżeli podprogram nie zostanie znaleziony, wykonywanie programu zostanie przez wywołanie EXTCALL przerwane.

Ustawiana pamięć doładowcza (bufor FIFO)

Do wykonywania podprogramu zewnętrznego jest potrzebna pamięć doładowania. Wielkość pamięci doładowania jest domyślnie ustawiona na 30 kByte i może zostać zmieniona tylko przez producenta maszyny (za pomocą MD18360 MM_EXT_PROG_BUFFER_SIZE).

Wskazówka

Podprogramy z instrukcjami skoku

W przypadku podprogramów zewnętrznych, które zawierają instrukcje skoku (GOTO, GOTOB, CASE, FOR, LOOP, WHILE, REPEAT, IF, ELSE, ENDF itd.), cele skoku muszą być położone w ramach pamięci doładowania.

Wskazówka

Programy ShopMill/ShopTurn

Programy ShopMill i ShopTurn z powodu dołączonych na końcu pliku opisów konturu muszą być zapisywane w pamięci doładowania w sposób kompletny.

Dla równoległe wykonywanych podprogramów zewnętrznych jest potrzebna własna pamięć doładowania.

Reset / koniec programu / POWER ON

Przez Reset i POWER ON zewnętrzne wywołania podprogramów są przerywane, a poszczególne pamięci doładowania kasowane.

Podprogram wybrany do "Wykonywania ze źródła zewnętrznego" pozostaje po Reset / końcu podprogramu nadal wybrany dla "Wykonywania ze źródła zewnętrznego". W wyniku POWER ON wybór ulega utraceniu.

Literatura

Dalsze informacje dot. "Wykonywania ze źródła zewnętrznego" patrz:

Podręcznik działania Funkcje podstawowe; BAG, Kanał, Praca programowa, Zachowanie się przy zresetowaniu (K1)

1.24.3.11 Wykonywanie podprogramu zewnętrznego (828D) (EXTCALL)

Funkcja

Przy pomocy polecenia `EXTCALL` można wykonywać program obróbki doładowując go z pamięci zewnętrznej.

Jako pamięć zewnętrzną są do dyspozycji:

- Karta CF użytkownika
- Stacja sieciowa
- Stacja USB

Wskazówka

Stacja USB

Jako interfejs do wykonywania programu zewnętrznego znajdującego się na stacji USB wolno stosować tylko interfejs USB na pulpicie obsługi (PPU).

Wskazówka

Nie stosować USB-FlashDrive

Jest zalecane, by do wykonywania podprogramu zewnętrznego nie stosować USB-FlashDrive. Przerwanie komunikacji z USB-FlashDrive podczas wykonywania programu obróbki w wyniku problemów na stykach, wypadnięcia, przerwanie w wyniku uderzenia albo pomyłkowe wyciągnięcie, prowadzi do natychmiastowego zatrzymania obróbki. Może przy tym nastąpić uszkodzenie narzędzia i/albo obrabianego przedmiotu.

Ustawienie domyślne ścieżki programu zewnętrznego

Ścieżka do zewnętrznego katalogu programów może zostać domyślnie ustawiona przy pomocy danej ustawczej:

```
SD42700 $SC_EXT_PROG_PATH
```

Ze ścieżki i identyfikatora programu, podanego przy wywołaniu `EXTCALL` wynika ścieżka całkowita wywoływanego programu.

Wskazówka**Parametry**

Przy wywoływaniu programu zewnętrznego nie można mu przekazać żadnych parametrów.

Składnia

```
EXTCALL("<ścieżka>:<nazwa programu>")
```

Znaczenie

```
EXTCALL:
"<ścieżka><nazwa
programu>":
```

Polecenie do wywołania podprogramu zewnętrznego

Stała/Zmienna typu STRING

<ścieżka>:

Absolutne lub względne podanie ścieżki
(**opcjonalnie**)

<nazwa programu>:

Nazwa programu jest podawana bez
przedrostka "_N_".

Rozszerzenie pliku ("MPF", "SPF") można
dołączyć do nazwy programu przy pomocy
znaku "_" lub "." (**opcjonalnie**).

Przykład:

```
"WALEK"
"WALEK_SPF"
"WALEK.SP"
```

Podanie ścieżki: Skróty

Następujące oznaczenia skrótowe można stosować przy podaniu ścieżki:

- Karta CF użytkownika: "CF_CARD:"
- Stacja USB (pulpit obsługi): "USB:"

Przykład

Program główny "MAIN.MPF" znajduje się w pamięci NC i jest wybrany do wykonania.

Podprogram "SP_1"

Podprogram zewnętrzny "SP_1.SPF" wzgl. "SP_1.MPF" znajduje się na karcie CF użytkownika w katalogu "/WKS.DIR/WST1.WPD".

Ścieżkę do zewnętrznego katalogu programów należy ustawić przy pomocy:

```
SD42700 $SC_EXT_PROG_PATH = CF_CARD:WKS.DIR/WST1.WPD
```

Wskazówka

Podanie ścieżki do wywołania podprogramu zewnętrznego:

- Bez zastosowania ustawienia domyślnego: "CF_CARD:WKS.DIR/WST1.WPD/SP_1"
 - Z zastosowaniem ustawienia domyślnego: "SP_1"
-

Podprogram "SP_2"

Podprogram zewnętrzny "SP_2.SPF" wzgl. "SP_2.MPF" znajduje się w katalogu WKS.DIR/WST1.WPD stacji USB. Ustawienie domyślne ścieżki do zewnętrznego katalogu programów jest stosowane dla ścieżki podprogramu "SP_1" i w programie głównym również nie jest przepisywane. Dlatego przy wywołaniu podprogramu "SP_2" musi zostać podana kompletna ścieżka.

Program główny "MAIN"

```
Kod programu
N010 PROC MAIN
N020 ...
N030 EXTCALL("SP_1")
N030 EXTCALL("USB:WKS.DIR/WST1.WPD/SP_2")
N050 ...
N060 M30
```

Dalsze informacje

Wywołanie EXTCALL z absolutnym podaniem ścieżki

Gdy podprogram pod podaną ścieżką istnieje, wówczas jest wykonywany po wywołaniu EXTCALL. Gdy podprogram pod podaną ścieżką nie istnieje, wówczas po wywołaniu EXTCALL wykonywanie programu jest przerywane.

Wywołanie EXTCALL ze względnym podaniem ścieżki / bez podania ścieżki

W przypadku wywołania EXTCALL ze względnym podaniem ścieżki lub bez jej podania istniejące pamięci programów są przeszukiwane według następującego wzorca:

1. Gdy w SD42700 \$SC_EXT_PROG_PATH jest domyślnie ustawiona ścieżka, następuje najpierw szukanie rozpoczynając od tej ścieżki według danej w wywołaniu EXTCALL (nazwa programu ew. ze względnym podaniem ścieżki). Ścieżka absolutna wynika wówczas przez powiązanie znaków:
 - z domyślnie ustawionego podania ścieżki w SD42700 \$SC_EXT_PROG_PATH
 - Znak rozdzielający "/"
 - Podanie ścieżki i nazwa podprogramu w poleceniu EXTCALL
2. Gdy podprogram nie został znaleziony pod 1., są przeszukiwane katalogi pamięci użytkownika.

Szukanie kończy się, gdy podprogram został po raz pierwszy znaleziony. Jeżeli podprogram nie zostanie znaleziony, wykonywanie programu zostanie przez wywołanie EXTCALL przerwane.

Ustawiana pamięć doładowcza (bufor FIFO)

Do wykonywania podprogramu zewnętrznego jest potrzebna pamięć doładowania. Wielkość pamięci doładowania jest ustawiona domyślnie (patrz MD18360 MM_EXT_PROG_BUFFER_SIZE).

Wskazówka

Podprogramy z instrukcjami skoku

W przypadku podprogramów zewnętrznych, które zawierają instrukcje skoku (GOTO, GOTOB, CASE, FOR, LOOP, WHILE, REPEAT, IF, ELSE, ENDF itd.), cele skoku muszą być położone w ramach pamięci doładowania.

Wskazówka

Programy ShopMill/ShopTurn

Programy ShopMill i ShopTurn z powodu dołączonych na końcu pliku opisów konturu muszą być zapisywane w pamięci doładowania w sposób kompletny.

Dla równoległe wykonywanych podprogramów zewnętrznych jest potrzebna własna pamięć doładowania.

Reset / koniec programu / POWER ON

Przez Reset i POWER ON zewnętrzne wywołania podprogramów są przerywane, a poszczególne pamięci doładowania kasowane.

Podprogram wybrany do "Wykonywania ze źródła zewnętrznego" pozostaje po Reset / końcu podprogramu nadal wybrany dla "Wykonywania ze źródła zewnętrznego". W wyniku POWER ON wybór ulega utraceniu.

Literatura

Dalsze informacje dot. "Wykonywania ze źródła zewnętrznego" patrz:

Podręcznik działania Funkcje podstawowe; BAG, Kanał, Praca programowa, Zachowanie się przy zresetowaniu (K1)

1.25 Technika makr (DEFINE ... AS)

UWAGA

Utrudnione programowanie

Przy zastosowaniu techniki makr język programowania sterowania może zostać bardzo zmieniony. Technika makr powinna być stosowana z dużą starannością.

Funkcja

Jako makro określa się połączenie poszczególnych instrukcji w nową instrukcję łączną z własną nazwą. Również funkcje G, M i H albo podprogramy L mogą być tworzone jako makra. Przy wywołaniu makra w przebiegu programu instrukcje zaprogramowane pod nazwą makra są kolejno wykonywane.

Zastosowanie

Ciągi instrukcji, które się powtarzają, programuje się tylko jeden raz jako makro we własnym module makr (plik makr) albo jeden raz na początku programu. Makro można następnie wywołać i wykonać w dowolnym programie głównym albo podprogramie.

Uaktywnienie

Aby móc stosować makra pliku makr w programie NC, plik makr musi zostać załadowany do NC.

Składnia

Definicja makra:

```
DEFINE <nazwa makra> AS <instrukcja 1> <instrukcja 2>
```

Wywołanie w programie NC:

```
<nazwa makra>
```

Znaczenie

DEFINE ... AS

Kombinacja słów kluczowych do definicji makra

<nazwa makra>:

Nazwa makra

Jako nazwy makr są dopuszczalne tylko identyfikatory.

Przy pomocy nazwy makra makro jest wywoływane z programu NC.

<instrukcja>:

Instrukcja programowa, która ma być zawarta w makropoleceniu.

Zasady definicji makr

- W makropoleceniu mogą być definiowane dowolne identyfikatory, funkcje G, M, H i nazwy programów L.
- Makra mogą być definiowane również w programie NC.
- Makra funkcji G mogą być definiowane tylko globalnie dla sterowania w module makr.
- Funkcje H i L mogą być programowane dwumiejscowo.
- Funkcje M i G mogą być programowane 3-miejscowo.

Wskazówka

Słów kluczowych i zarezerwowanych nazw nie wolno przedefiniowywać przy pomocy makr.

Warunki brzegowe

Kaskadowanie makr jest niemożliwe.

Przykłady

Przykład 1: Definicja makra na początku programu

Kod programu	Komentarz
DEFINE LINIA AS G1 G94 F300	; Definicja makra
...	
...	
N70 LINIA X10 Y20	; Wywołanie makra
...	

Przykład 2: Definicje makr w pliku makr

Kod programu	Komentarz
DEFINE M6 AS L6	; Przy wymianie narzędzia wywoływany jest podprogram, który przejmuje niezbędną transmisję danych. W podprogramie jest wyprowadzana właściwa funkcja M wymiany narzędzia (np. M106).
DEFINE G81 AS DRILL(81)	; Odwzorowanie funkcji G DIN
DEFINE G33 AS M333 G333	; Przy nacinaniu gwintu następuje żądanie synchronizacji z PLC. Pierwotna funkcja G33 została przez MD przemianowana na G333, programowanie pozostaje dla użytkownika takie samo.

Przykład 3: Zewnętrzny plik makro

Po wczytaniu zewnętrznego pliku makr do sterowania, plik makr musi zostać załadowany do NC. Dopiero wówczas makra mogą być stosowane w programie NC.

Kod programu	Komentarz
%_N_UMAC_DEF	
;\$PATH=/_N_DEF_DIR	; Makra specyficzne dla klienta
DEFINE PI AS 3.14	
DEFINE TC1 AS M3 S1000	
DEFINE M13 AS M3 M7	; Wrzeczono w kierunku prawym, włączenie chłodziwa
DEFINE M14 AS M4 M7	; Wrzeczono w kierunku lewym, włączenie chłodziwa
DEFINE M15 AS M5 M9	; Zatrzymanie wrzeciona, wyłączenie chłodziwa
DEFINE M6 AS L6	; Wywołanie programu wymiany narzędzia
DEFINE G80 AS MCALL	; Cofnięcie wyboru cyklu wiercenia
M30	

Zarządzanie plikami i programami

2.1 Pamięć programów

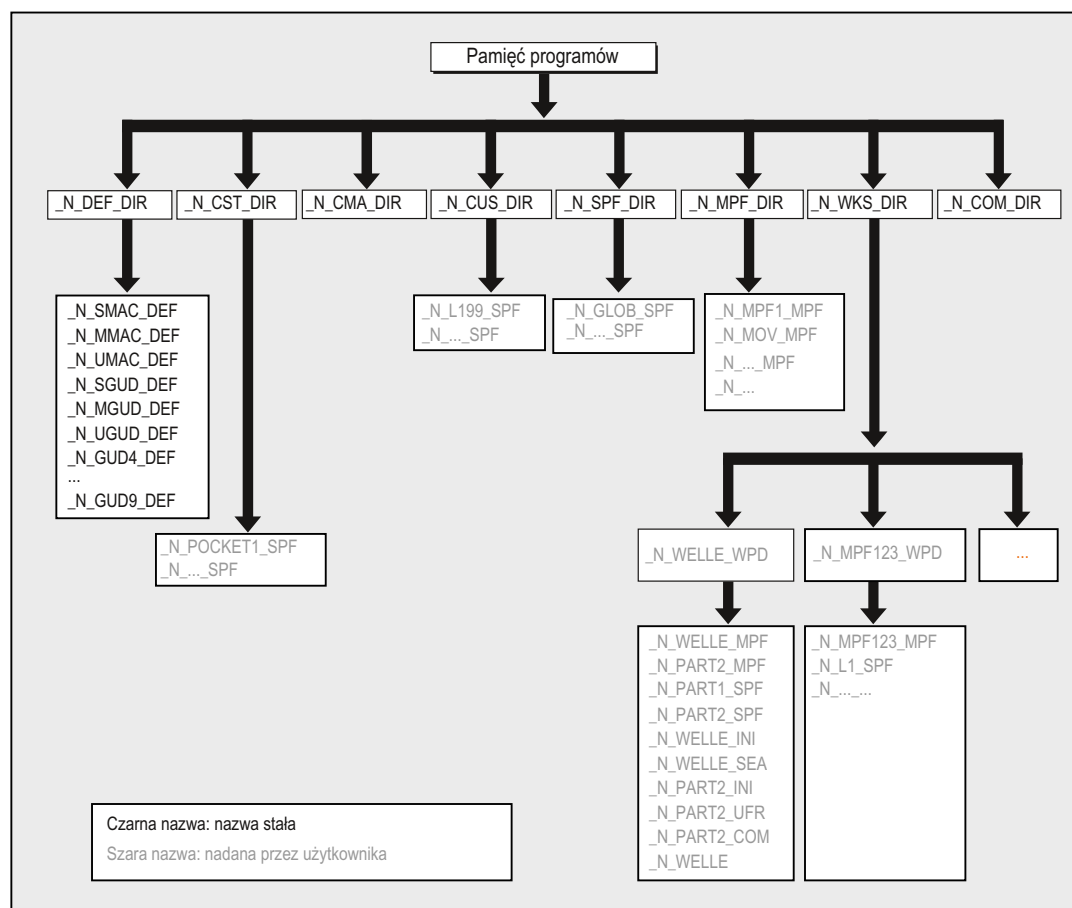
Funkcja

W pamięci programów są trwale zapisywane pliki i programy (np. programy główne i podprogramy, definicje makr) (→ pasywny system plików).

Literatura:

Podręcznik działania Funkcje rozszerzone; Konfiguracja pamięci (S7)

Oprócz tego jest pewna liczba typów plików, które mogą być tutaj zapisywane tymczasowo i w razie potrzeby (np. przy obróbce określonego przedmiotu) powinny być przenoszone do pamięci roboczej (np. do celów inicjalizacyjnych).



Katalogi standardowe

Standardowo są zawarte następujące katalogi:

Wykaz	Treść
_N_DEF_DIR	Moduły danych i moduły makr
_N_CST_DIR	Cykle standardowe
_N_CMA_DIR	Cykle producenta
_N_CUS_DIR	Cykle użytkownika
_N_WKS_DIR	Obrabiane przedmioty
_N_SPF_DIR	Podprogramy globalne
_N_MPF_DIR	Programy główne
_N_COM_DIR	Komentarze

Typy plików

W pamięci programów mogą być umieszczane następujące typy plików:

Typ pliku	Opis
<Nazwa>_MPF	Program główny
<Nazwa>_SPF	Podprogram
<Nazwa>_TEA	Dane maszynowe
<Nazwa>_SEA	Dane ustawcze
<Nazwa>_TOA	Korekcje narzędzi
<Nazwa>_UFR	Przesunięcia punktu zerowego/frame
<Nazwa>_INI	Plik inicjalizacyjny
<Nazwa>_GUD	Globalne dane użytkownika
<Nazwa>_RPA	Parametry R
<Nazwa>_COM	Komentarz
<Nazwa>_DEF	Definicje globalnych danych użytkownika i makr

Katalog główny obrabianych przedmiotów (_N_WKS_DIR)

Katalog główny obrabianych przedmiotów jest standardowo utworzony w pamięci programów pod nazwą _N_WKS_DIR. Katalog główny obrabianych przedmiotów zawiera dla wszystkich obrabianych przedmiotów, które zostały zaprogramowane, odpowiednie katalogi obrabianego przedmiotu.

Katalogi obrabianego przedmiotu (...WPD)

W celu bardziej elastycznego posługiwania się danymi i programami określone dane i programy mogą być ze sobą wiązane albo zapisywane w poszczególnych katalogach obrabianego przedmiotu.

Katalog obrabianego przedmiotu zawiera wszystkie pliki, które są konieczne do obrobienia przedmiotu. Mogą to być programy główne, podprogramy, dowolne programy inicjalizacyjne i pliki komentarzowe.

Programy inicjalizacyjne są jednorazowo wykonywane z pierwszym startem programu po jego wybraniu (odpowiednio do danej maszynowej MD11280 \$MN_WPD_INI_MODE).

Przykład:

Katalog obrabianego przedmiotu `_N_WALEK_WPD`, który został utworzony dla obrabianego przedmiotu WALEK, zawiera następujące pliki:

Plik	Opis
<code>_N_WALEK_MPF</code>	Program główny
<code>_N_PART2_MPF</code>	Program główny
<code>_N_PART1_SPF</code>	Podprogram
<code>_N_PART2_SPF</code>	Podprogram
<code>_N_WALEK_INI</code>	Ogólny program inicjalizacyjny danych obrabianego przedmiotu
<code>_N_WALEK_SEA</code>	Program inicjalizacyjny dane ustawcze
<code>_N_PART2_INI</code>	Ogólny program inicjalizacyjny danych dla programu Part 2
<code>_N_PART2_UFR</code>	Program inicjalizacyjny dla danych frame dla programu Part 2
<code>_N_WALEK_COM</code>	Plik komentarzy

Tworzenie katalogów obrabianych przedmiotów na zewnętrznym PC

Poniżej opisany sposób postępowania jest przeprowadzany na zewnętrznej stacji danych. Informacje odnośnie zarządzania plikami i programami (od PC do sterowania), bezpośrednio w sterowaniu, można znaleźć w instrukcji obsługi.

Utworzenie katalogu obrabianego przedmiotu z podaniem ścieżki (\$PATH=...)

W drugim wierszu pliku jest podawana ścieżka celu przy pomocy `$PATH=...`. Plik jest wówczas zapisywany pod podaną ścieżką.

Przykład:

Kod programu
<pre>%_N_WALEK_MPF ; \$PATH=/_N_WKS_DIR/_N_WALEK_WPD N10 G0 X... Z... ... M2</pre>

Plik `_N_WALEK_MPF` jest zapisywany w katalogu `/_N_WKS_DIR/_N_WALEK_WPD`.

Utworzenie katalogu obrabianego przedmiotu bez podania ścieżki

Jeżeli brak jest podania ścieżki, wówczas pliki o rozszerzeniu _SPF są zapisywane w katalogu /_N_SPF_DIR, pliki o rozszerzeniu _INI w pamięci roboczej, a wszystkie pozostałe pliki w katalogu _N_MPF_DIR.

Przykład:

```
Kod programu
-----
%_N_WALEK_SPF
...
M17
```

Plik _N_WALEK_SPF jest zapisywany w katalogu /_N_SPF_DIR.

Wybranie obrabianego przedmiotu do obróbki

Katalog obrabianego przedmiotu może zostać wybrany w celu wykonania w kanale. Jeżeli w tym katalogu znajduje się program główny **o tej samej nazwie** albo tylko jeden jedyny program główny (_MPF), wówczas jest on automatycznie wybierany do wykonania.

Literatura:

Podręcznik obsługi

Ścieżki szukania przy wywołaniu podprogramu

Jeżeli ścieżka wywołania nie zostanie explicite podana w programie obróbki przy wywołaniu podprogramu (albo też plik inicjalizacyjny), wówczas wywoływany program jest określany na stałej ścieżce szukania.

Wywołanie podprogramu z absolutnym podaniem ścieżki

Przykład:

```
Kod programu
-----
...
CALL"/_N_CST_DIR/_N_CYCLE1_SPF"
...
```

Wywołanie podprogramu bez absolutnego podania ścieżki

Z reguły programy są wywoływane bez podania ścieżki.

Przykład:

```
Kod programu
-----
...
CYCLE1
...
```


Katalogi są przeszukiwane w poszukiwaniu wywołanego programu w następującej kolejności:

Nr	Wykaz	Opis
1	Aktualny katalog / <i>nazwa</i>	Katalog główny obrabianych przedmiotów albo katalog standardowy <i>_N_MPF_DIR</i>
2	Aktualny katalog / <i>nazwa_SPF</i>	
3	Aktualny katalog / <i>nazwa_MPF</i>	
4	<i>/_N_SPF_DIR / nazwa_SPF</i>	Podprogramy globalne
5	<i>/_N_CUS_DIR / nazwa_SPF</i>	Cykle użytkownika
6	<i>/_N_CMA_DIR / nazwa_SPF</i>	Cykle producenta
7	<i>/_N_CST_DIR / nazwa_SPF</i>	Cykle standardowe

Programowanie ścieżek szukania przy wywołaniu podprogramu (CALLPATH)

Przy wywołaniu podprogramu można rozszerzyć ścieżkę szukania przy pomocy polecenia programu obróbki `CALLPATH`.

Przykład:

```
Kod programu
-----
CALLPATH ("/_N_WKS_DIR/_N_MYWPD_WPD")
...
```

Ścieżka szukania jest zapisywana przed pozycją 5. (cykl użytkownika) odpowiednio do podanego zaprogramowania.

Dalsze informacje dot. programowalnej ścieżki szukania przy wywołaniach podprogramów przy pomocy `CALLPATH` patrz "Rozszerzenie ścieżki szukania przy wywołaniach podprogramu (`CALLPATH`) (Strona 201)".

2.2 Pamięć robocza (CHANDATA, COMPLETE, INITIAL)

Funkcja

Pamięć robocza zawiera aktualne dane systemowe i użytkownika, z którymi sterowanie jest eksploatowane (aktywny system plików), np.:

- Aktywne dane maszynowe
- Dane korekcyjne narzędzia
- Przesunięcia punktu zerowego
- ...

Programy inicjalizacyjne

Chodzi tutaj o programy, przy pomocy których dane pamięci roboczej są domyślnie ustawiane (inicjalizowane). Mogą w tym celu być stosowane następujące typy plików:

Typ pliku	Opis
nazwa_TEA	Dane maszynowe
nazwa_SEA	Dane ustawcze
nazwa_TOA	Korekcje narzędzi
nazwa_UFR	Przesunięcia punktu zerowego/frame
nazwa_INI	Plik inicjalizacyjny
nazwa_GUD	Globalne dane użytkownika
nazwa_RPA	Parametry R

Zakresy danych

Dane mogą być włączone do różnych zakresów, w których mają obowiązywać. Na przykład sterowanie może dysponować wieloma kanałami albo jak zazwyczaj wieloma osiami.

Są:

Oznaczenie	Zakresy danych
NCK	Dane specyficzne dla NCK
CH<n>	Dane specyficzne dla kanału (<n> podaje numer kanału)
AX<n>	Dane specyficzne dla osi (n podaje numer osi maszyny)
TO	Dane narzędzi
COMPLETE	Wszystkie dane

Utworzenie programu inicjalizacyjnego na zewnętrznym PC

Przy pomocy identyfikacji zakresów danych i identyfikacji typów danych mogą zostać określone obszary, które przy zapisywaniu danych są traktowane jako jednostka:

_N_AX5_TEA_INI	Dane maszynowe dla osi 5
_N_CH2_UFR_INI	Frame kanału 2
_N_COMPLETE_TEA_INI	Wszystkie dane maszynowe

Po uruchomieniu sterowania w pamięci roboczej jest zestaw danych, który gwarantuje jego należyłą pracę.

Sposób postępowania w przypadku sterowań wielokanałowych (CHANDATA)

CHANDATA (<numer kanału>) dla wielu kanałów jest dopuszczalne tylko w pliku `_N_INITIAL_INI`. Jest to plik uruchomieniowy, przy pomocy którego są inicjalizowane wszystkie dane sterowania.

Kod programu	Komentarz
<code>%_N_INITIAL_INI</code>	
<code>CHANDATA(1)</code>	
	; Przyporządkowanie osi maszyny kanał 1:
<code>\$MC_AXCONF_MACHAX_USED[0]=1</code>	
<code>\$MC_AXCONF_MACHAX_USED[1]=2</code>	
<code>\$MC_AXCONF_MACHAX_USED[2]=3</code>	
<code>CHANDATA(2)</code>	
	; Przyporządkowanie osi maszyny kanał 2:
<code>\$MC_AXCONF_MACHAX_USED[0]=4</code>	
<code>\$MC_AXCONF_MACHAX_USED[1]=5</code>	
<code>CHANDATA(1)</code>	
	; Osiowe dane maszynowe:
	; okno zatrzymania dokładnego zgrubnie:
<code>\$MA_STOP_LIMIT_COARSE[AX1]=0.2</code>	; Oś 1
<code>\$MA_STOP_LIMIT_COARSE[AX2]=0.2</code>	; Oś 2
	; okno zatrzymania dokładnego dokładnie:
<code>\$MA_STOP_LIMIT_FINE[AX1]=0.01</code>	; Oś 1
<code>\$MA_STOP_LIMIT_FINE[AX2]=0.01</code>	; Oś 2

UWAGA

Instrukcja CHANDATA

W programie obróbki wolno umieścić instrukcję `CHANDATA` tylko dla tego kanału, w którym jest wykonywany program NC, tzn. instrukcja może zostać użyta do tego, by chronić programy przed wykonywaniem w nieprzewidzianym kanale.

W przypadku błędu wykonywanie programu jest przerywane.

Wskazówka

Pliki INI na listach zadań nie zawierają instrukcji `CHANDATA`.

Wykonanie kopii zapasowej programów inicjalizacyjnych (COMPLETE, INITIAL)

Pliki pamięci roboczej mogą zostać zapisane na zewnętrznym PC i być stamtąd wczytywane.

- Pliki są zapisywane przy pomocy COMPLETE.
- Przy pomocy INITIAL jest dla wszystkich obszarów tworzony plik INI (_N_INITIAL_INI).

Wczytanie programów inicjalizacyjnych

UWAGA

Utrata danych

Gdy zostanie wczytany plik o nazwie "INITIAL_INI", wówczas wszystkie dane, które nie są dostarczane w tym pliku, są inicjalizowane jako dane standardowe. Wyłączone z tego są tylko dane maszynowe. Jako standardowe są więc dostarczane **dane ustawcze, dane narzędzi, PPZ, wartości GUD, ...** (zazwyczaj "ZERO").

Do wczytywania poszczególnych danych maszynowych nadaje się np. plik COMPLETE_TEA_INI. W tym pliku sterowanie oczekuje tylko danych maszynowych. Inne obszary danych pozostają w tym przypadku nietknięte.

Ładowanie programów inicjalizacyjnych

Programy INI mogą być również wybierane i wywoływane jako programy obróbki, gdy stosują tylko dane jednego kanału. W ten sposób jest również możliwe inicjalizowanie danych sterowanych programem.

Obszary ochrony

3.1 Ustalenie obszarów ochrony (CPROTDEF, NPROTDEF)

Funkcja

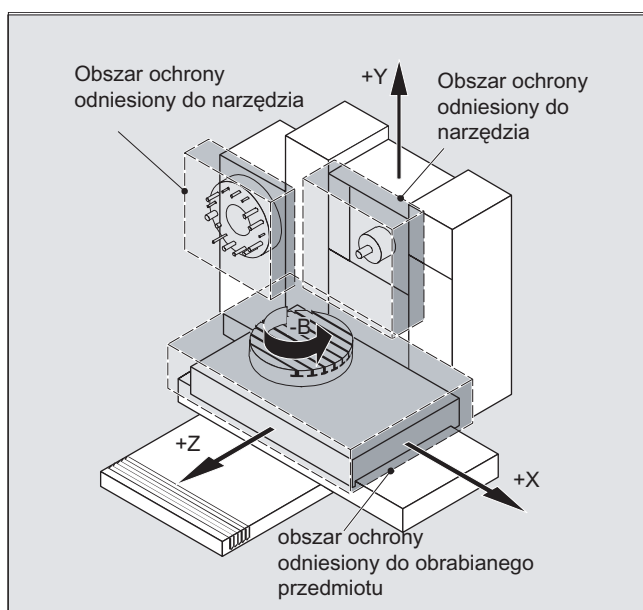
Przy pomocy obszarów ochrony można chronić przed nieprawidłowymi ruchami różne elementy w maszynie, wyposażenie, jak też obrabiany przedmiot.

Obszary ochrony odniesione do narzędzia:

Dla części, które należą do narzędzia (np. narzędzie, nośnik narzędzi).

Obszary ochrony odniesione do obrabianego przedmiotu:

Dla części, które należą do obrabianego przedmiotu (np. części obrabianego przedmiotu, stół, łapy mocujące, uchwyt tokarski, konik).



Składnia

```
DEF INT NOT_USED
G17/G18/G19
CPROTDEF/NPROTDEF (<n>, <t>, <applim>, <applus>, <appminus>)
G0/G1/... X/Y/Z...
...
EXECUTE (NOT_USED)
```

Znaczenie

DEF INT NOT_USED:	Zdefiniowanie zmiennej lokalnej o typie danych INTEGER
G17/G18/G19:	Żądana płaszczyzna jest wybierana przed CPROTDEF lub NPROTDEF przy pomocy G17/G18/G19 i nie wolno jej zmienić przed EXECUTE. Programowanie aplikacji między CPROTDEF lub NPROTDEF i EXECUTE jest niedopuszczalne.
CPROTDEF:	Definiowanie kanałowych obszarów ochrony
NPROTDEF:	Definicja obszarów ochrony specyficznych dla maszyny
G0/G1/... X/Y/Z... ... :	Kontur obszarów ochrony jest podawany przy pomocy maksymalnie 11 ruchów postępowych w wybranej płaszczyźnie. Pierwszym ruchem jest przy tym ruch do konturu. Jako obszar ochrony obowiązuje przy tym obszar na lewo od konturu. Wskazówka: Ruchy znajdujące się między CPROTDEF lub NPROTDEF i EXECUTE nie są wykonywane, lecz definiują obszar ochrony.
EXECUTE:	Zakończenie definicji
<n>:	Numer zdefiniowanego obszaru ochrony
<t>:	Typ obszaru ochrony:
	TRUE: Obszar ochrony odniesiony do narzędzia
	FALSE: Obszar ochrony odniesiony do obrabanego przedmiotu
<aplim>:	Rodzaj ograniczenia w 3. wymiarze
	0: Nie ma ograniczenia
	1: Ograniczenie w kierunku dodatnim
	2: Ograniczenie w kierunku ujemnym
	3: Ograniczenie w kierunku dodatnim i ujemnym
<applus>:	Wartość ograniczenia w kierunku dodatnim 3. wymiaru
<appminus>:	Wartość ograniczenia w kierunku ujemnym 3. wymiaru
NOT_USED:	Zmienna błędu nie działa w przypadku obszarów ochrony z EXECUTE

Warunki brzegowe

Podczas definicji obszarów ochrony nie może:

- być aktywna korekcja promienia frezu lub promienia ostrza.
- być aktywna transformacja.
- być aktywny frame.

Nie może być też zaprogramowane bazowanie do punktu odniesienia (G74), dosunięcie do punktu stałego (G75), zatrzymanie przebiegu wyprzedzającego, ani koniec programu.

Dalsze informacje

Definicja obszarów ochrony

Do definicji obszarów ochrony należą:

- CPROTDEF dla kanałowych obszarów ochrony
- NPROTDEF dla obszarów ochrony specyficznych dla maszyny
- Opis konturu obszaru ochrony
- Zakończenie definicji przy pomocy EXECUTE

Przy uaktywnieniu obszaru ochrony w programie obróbki NC można relatywnie przesunąć jego punkt odniesienia.

Punkt odniesienia opisu konturu

Obszary ochrony odniesione do obrabianego przedmiotu są definiowane w bazowym układzie współrzędnych.

Obszary ochrony odniesione do narzędzia są podawane w odniesieniu do punktu odniesienia nośnika narzędzi F.

Dopuszczalne elementy konturu

Dla opisu konturu obszaru ochrony są dopuszczalne:

- G0, G1 dla prostych elementów konturu
- G2 dla segmentów kołowych w kierunku ruchu wskazówek zegara (tylko dla obszarów ochrony odniesionych do obrabianego przedmiotu)
- G3 dla segmentów kołowych przeciwnie do ruchu wskazówek zegara

Wskazówka

Jeżeli pełny okrąg ma opisywać obszar ochrony, wówczas należy podzielić go na 2 części. Sekwencja G2, G3 wzgl. G3, G2 jest niedopuszczalna. Tutaj należy ew. wstawić krótki blok G1.

Ostatni punkt opisu konturu musi pokrywać się z pierwszym punktem.

Zewnętrzne obszary ochrony

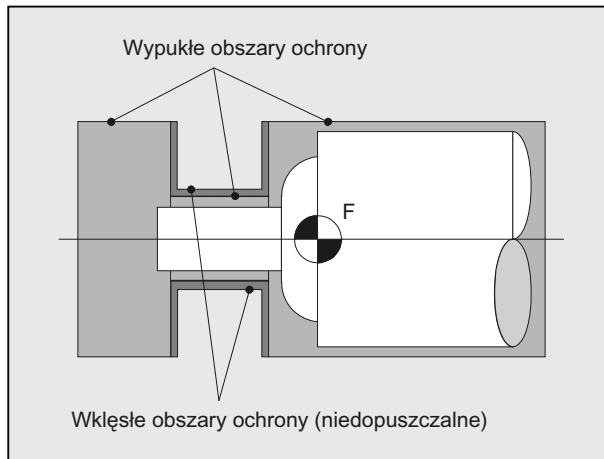
Zewnętrzne obszary ochrony (możliwe tylko w przypadku obszarów ochrony odniesionych do obrabianego przedmiotu) należy definiować w kierunku ruchu wskazówek zegara.

Obszary ochrony symetryczne obrotowo

W przypadku obrotowo-symetrycznych obszarów ochrony (np. uchwyt tokarski) należy opisać cały kontur (nie tylko do osi toczenia!).

Obszary ochrony odniesione do narzędzia:

Obszary ochrony odniesione do narzędzia muszą być zawsze wypukłe. Jeżeli jest żądany wklęsły obszar ochrony, należy go podzielić na wiele obszarów wypukłych.



3.2 Uaktywnienie / wyłączenie aktywności obszarów ochrony (CPROT, NPROT)

Funkcja

Uaktywnienie, wstępne uaktywnienie wcześniej zdefiniowanych obszarów ochrony dla nadzoru na kolizję lub wyłączenie aktywności aktywnych obszarów ochrony.

Maksymalna liczba obszarów ochrony równocześnie aktywnych w kanale jest ustalana przez daną maszynową.

Jeżeli żaden obszar ochrony odniesiony do narzędzia nie jest aktywny, wówczas tor ruchu narzędzia jest sprawdzany w stosunku do obszarów ochrony odniesionych do obrabianego przedmiotu.

Wskazówka

Jeżeli żaden obszar ochrony odniesiony do obrabianego przedmiotu nie jest aktywny, nie następuje nadzór obszarów ochrony.

Składnia

CPROT (<n>, <state>, <xMov>, <yMov>, <zMov>)
NPROT (<n>, <state>, <xMov>, <yMov>, <zMov>)

Znaczenie

CPROT:	Wywołanie obszaru ochrony specyficznego dla kanału
NPROT:	Wywołanie obszaru ochrony specyficznego dla maszyny
<n>:	Numer obszaru ochrony

<state>:	Podanie statusu
	0: Wyłączenie aktywności obszaru ochrony
	1: Wstępne uaktywnienie obszaru ochrony
	2: Uaktywnienie obszaru ochrony
	3: Wstępne uaktywnienie obszaru ochrony z warunkowym zatrzymaniem
<xMov>, <yMov>, <zMov>:	Przesunięcie w osiach geometrycznych już zdefiniowanego obszaru ochrony

Warunki brzegowe

Nadzór obszaru ochrony przy aktywnej korekcji promienia narzędzia

Przy aktywnej korekcji promienia narzędzia działanie nadzoru obszaru ochrony jest możliwe tylko wtedy, gdy płaszczyzna korekcji promienia narzędzia jest identyczna z płaszczyzną definicji obszaru ochrony.

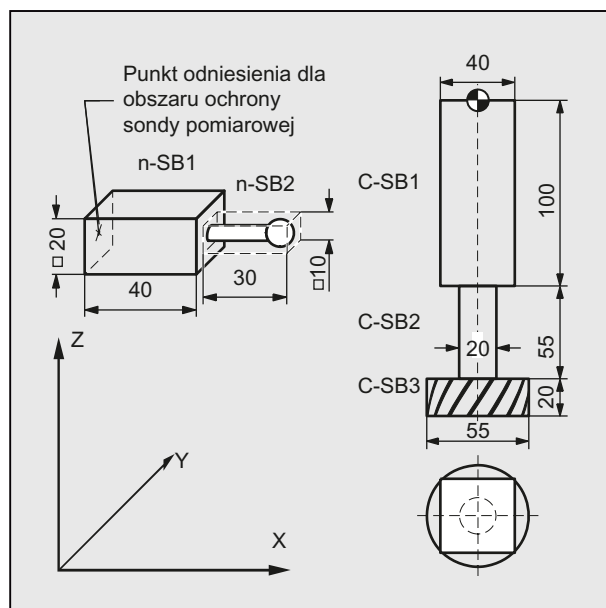
Przykład

Dla frezarki ma być prowadzony nadzór na możliwą kolizję frezu z sondą pomiarową. Położenie sondy pomiarowej powinno przy uaktywnieniu zostać podane przez przesunięcie. Są w tym celu definiowane następujące obszary ochrony:

- Każdorazowo jeden specyficzny dla maszyny i odniesiony do obrabianego przedmiotu obszar ochrony dla uchwytu sondy pomiarowej (n-SB1) i dla samej sondy pomiarowej (n-SB2).
- Każdorazowo specyficzny dla kanału i odniesiony do narzędzia obszar ochrony dla uchwytu frezu (c-SB1), chwytu frezu (c-SB2) i dla samego frezu (c-SB3).

Wszystkie obszary ochrony są zorientowane w kierunku Z.

Położenie punktu odniesienia czujnika pomiarowego przy uaktywnieniu niech wynosi $X = -120$, $Y = 60$ i $Z = 80$.



3.2 Uaktywnienie / wyłączenie aktywności obszarów ochrony (CROT, NROT)

Kod programu	Komentarz
DEF INT SCHUTZB	; Definicja zmiennej pomocniczej
Definicja obszarów ochrony G17	; Ustawienie orientacji
NPROTDEF(1,FALSE,3,10,-10)G01 X0 Y-10	; Obszar ochrony n-SB1
X40	
Y10	
X0	
Y-10	
EXECUTE(SCHUTZB)	
NPROTDEF(2,FALSE,3,5,-5)	; Obszar ochrony n-SB2
G01 X40 Y-5	
X70	
Y5	
X40	
Y-5	
EXECUTE(SCHUTZB)	
CPROTDEF(1,TRUE,3,0,-100)	; Obszar ochrony c-SB1
G01 X-20 Y-20	
X20	
Y20	
X-20	
Y-20	
EXECUTE(SCHUTZB)	
CPROTDEF(2,TRUE,3,-100,-150)	; Obszar ochrony c-SB2
G01 X0 Y-10	
G03 X0 Y10 J10	
X0 Y-10 J-10	
EXECUTE(SCHUTZB)	
CPROTDEF(3,TRUE,3,-150,-170)	; Obszar ochrony c-SB3
G01 X0 Y-27,5	
G03 X0 Y27,5 J27,5	
X0 Y27,5 J-27,5	
EXECUTE(SCHUTZB)	
Uaktywnienie obszarów ochrony:	
NPROT(1,2,-120,60,80)	; Uaktywnienie obszaru ochrony n-SB1 z przes.
NPROT(2,2,-120,60,80)	; Uaktywnienie obsz. ochr. n-SB2 z przes.
CPROT(1,2,0,0,0)	; Uaktywnienie obsz. ochr. c-SB1 z przes.
CPROT(2,2,0,0,0)	; Uaktywnienie obsz. ochr. c-SB2 z przes.
CPROT(3,2,0,0,0)	; Uaktywnienie obsz. ochr. c-SB3 z przes.

Dalsze informacje

Status uaktywnienia (<state>)

- <state>=2

Obszar ochrony jest zazwyczaj uaktywniany w programie obróbki ze statusem = 2.

Status jest zawsze specyficzny dla kanału, również w przypadku obszarów ochrony odniesionych do maszyny.

- <state>=2

Jeżeli przez program użytkownika PLC jest przewidziane, że działanie obszaru ochrony może zostać ustawione przez program użytkownika PLC, wówczas przewidziane w tym celu wstępne uaktywnienie następuje przez status = 1.

- <state>=3

Przy wstępnym uaktywnieniu z warunkowym zatrzymaniem zasadniczo nie następuje zatrzymanie przed naruszonym, wstępnie uaktywnionym obszarem ochrony. Zatrzymanie następuje tylko wtedy, gdy jest ustawione działanie obszaru ochrony. Umożliwia to nieprzerwaną obróbkę, gdy działanie obszarów ochrony jest ustawiane tylko w szczególnych przypadkach. Należy uwzględnić, że w wyniku charakterystyki hamowania następuje ew. wejście w obszar ochrony, gdy działanie obszaru ochrony zostało ustawione dopiero bezpośrednio przed wejściem w ten obszar.

Wstępne uaktywnienie z zatrzymaniem warunkowym następuje przez status = 3.

- <state>=0

Wyłączenie aktywności, a przez to wyłączenie obszarów ochrony następuje przez status = 0. Nie jest przy tym konieczne żadne przesunięcie.

Przesunięcie obszarów ochrony przy (wstępnym) uaktywnieniu

Przesunięcie może nastąpić w 1, 2 albo 3 wymiarach. Podanie przesunięcia odnosi się do:

- punktu zerowego maszyny w przypadku obszarów ochrony specyficznych dla obrabianego przedmiotu.
- punktu odniesienia nośnika narzędzi F w przypadku obszarów ochrony specyficznych dla narzędzia

Status po rozruchu

Obszary ochrony mogą być uaktywnione już po rozruchu i następnie przeprowadzonym bazowaniu do punktu odniesienia. Musi być w tym celu ustawiona na TRUE zmienna systemowa \$SN_PA_ACTIV_IMMED[<n>] lub \$SC_PA_ACTIV_IMMED[<n>]. Są one zawsze uaktywniane ze statusem = 2 i nie mają przesunięcia.

Wielokrotne uaktywnienie obszarów ochrony

Obszar ochrony może działać równocześnie w wielu kanałach (np. tuleja wrzecionowa w przypadku dwóch naprzeciwległych sań). Nadzór obszarów ochrony następuje tylko wtedy, gdy nastąpiło bazowanie wszystkich osi geometrycznych do punktu odniesienia.

Przy tym obowiązuje:

- Obszar ochrony nie da się uaktywnić w tym samym czasie wielokrotnie w jednym kanale z różnymi przesunięciami.
- Obszary ochrony odniesione do maszyny muszą w obydwu kanałach być tak samo zorientowane.

3.3 Sprawdzenie naruszenia obszaru ochrony, ograniczenia obszaru pracy oraz wyłączników krańcowych oprogramowania (CALCPOSI)

Funkcja

Funkcja `CALCPOSI()` sprawdza układ współrzędnych (UW) pod względem **osi geometrycznych** i czy aktywne ograniczenia wychodząc od pozycji startowej wzdłuż drogi ruchu nie zostały naruszone.

Jeżeli nie można całkowicie przebyć drogi ruchu, zwracana jest dodatnia, dziesiętnie kodowana wartość statusu oraz maksymalnie możliwa droga ruchu.

Składnia

`<Status> = CALCPOSI (VAR <Start>, VAR <Dist>, VAR <Limit>, VAR <MaxDist>, <System>, <TestLim>)`

Znaczenie

CALCPOSI:	Test na naruszenia ograniczeń odnośnie osi geometrycznych.	
	Zatrzymanie przebiegu wyprzedzającego:	nie
	W oddzielnym bloku:	tak
<status>: (Część 1)	Zwracana wartość funkcji. Wartości ujemne wskazują stany błędów.	
	Typ danych:	INT
	Zakres wartości:	$-8 \leq x \leq 100000$
	Wartości	Znaczenie
	0	Można całkowicie przebyć drogę ruchu
	-1	W <Limit> przynajmniej jedna składowa jest ujemna
	-2	Błędy przy obliczaniu transformacji. Przykład: Droga ruchu prowadzi przez położenie osobliwe, tak że pozycje osi nie są zdefiniowane.
	-3	Podana droga ruchu <odległość> i maksymalnie możliwa droga ruchu do przebycia <maks.odległość> są zależne liniowo. Uwaga Może wystąpić tylko w związku z <TestLim>, Bit 4 == 1.
	-4	Projekcję zawartego w <odległość> kierunku ruchu na powierzchni ograniczającej określa wektor zerowy lub też kierunek ruchu znajduje się w pozycji prostopadłej w stosunku do naruszonej powierzchni ograniczającej. Uwaga Może wystąpić tylko w związku z <TestLim>, Bit 5 == 1.
-5	W <TestLim> są Bit 4 == 1 UND Bit 5 == 1	

3.3 Sprawdzenie naruszenia obszaru ochrony, ograniczenia obszaru pracy oraz wyłączników krańcowych oprogramowania

	-6	Minimum jedna oś maszyny, która służy do kontroli granic ruchu, nie jest zbazowana.
	-7	Funkcja Unikanie kolizji: Nieważna definicja łańcucha kinematycznego bądź obszarów ochronnych.
	-8	Funkcja Unikanie kolizji: Funkcja może nie zostać zrealizowana z powodu braku pamięci.
<status>: (Część 2)	Miejsce jedności:	
	Uwaga Jeśli równocześnie naruszono kilka granic, zgłoszona zostanie ta, która prowadzi do najsilniejszego ograniczenia wyznaczonej drogi ruchu.	
	1	Krańcowe wyłączniki softwareowe ograniczają drogę ruchu.
	2	Ograniczenie pola roboczego narusza drogę ruchu
	3	Obszary ochrony ograniczają drogę ruchu.
	Miejsce dziesiątek	
	1x	Wartość początkowa narusza granicę
	2x	Zadana prosta narusza granicę. Ta wartość jest zwracana również wtedy, gdy sam punkt końcowy nie narusza granicy, ale na drodze od punktu startowego do końcowego wystąpiłoby naruszenie wartości granicznej (np. przejście przez obszar ochrony, zakrzywione krańcowe wyłączniki softwareowe w WKS przy transformacjach nieliniowych, np. Transmit).
<status>: (Część 3)	Miejsce setek	
	1xx	I miejsce jedności == 1 lub 2: Naruszono dodatnią wartość graniczną.
		I miejsce jedności == 3 ¹⁾ : Naruszono specyficzny dla NC obszar ochrony.
	2xx	I miejsce jedności == 1 lub 2: Naruszono ujemną wartość graniczną.
		I miejsce jedności == 3 ¹⁾ : Naruszono specyficzny dla kanału obszar ochrony.
<status>: (Część 4)	Miejsce tysięcy	
	1xxx	I miejsce jedności == 1 lub 2: Czynnik, przez który jest pomnożony numer osi, która narusza granicę. Liczenie osi zaczyna się od 1. Odniesienie: <ul style="list-style-type: none"> • Softwareowy wyłącznik krańcowy Osie maszyny • Ograniczenie obszaru pracy: Osie geometryczne I miejsce jedności == 3 ¹⁾ : Czynnik, przez który pomnożony jest numer naruszonego obszaru ochrony.

3.3 Sprawdzenie naruszenia obszaru ochrony, ograniczenia obszaru pracy oraz wyłączników krańcowych oprogramowania (CALCPOSI)

<status>: (Część 5)	Pozycja stutysięczna	
	0xxxxx	Pozycja stutysięczna == 0: <odległość> pozostaje niezmienną
	1xxxxx	<p>W <odległość> cofnięty zostaje jeden wektor kierunku, który definiuje dalszy kierunek ruchu na powierzchni ograniczającej. Może wystąpić przy następujących warunkach brzegowych:</p> <ul style="list-style-type: none"> Naruszenie softwaerowych wyłączników krańcowych lub ograniczenia obszaru roboczego (nie w punkcie startowym) Transformacja nie jest aktywna. <TestID>, Bit 4 lub Bit 5 == 1
1) Jeśli naruszonych jest wiele obszarów ochrony, zgłoszony zostanie obszar ochrony, który prowadzi do największego ograniczenia zadanej drogi ruchu.		
<Start>:	Odniesienie do wektora z pozycjami startowymi:	
	<ul style="list-style-type: none"> <Start> [0]: Odcięta <Start> [1]: Rzędna <Start> [2]: Aplikata 	
	Typ parametru:	Wejście
	Typ danych:	VAR REAL [3]:
	Zakres wartości:	-maks. wartość REAL ≤ x[n] ≤ + maks. wartość REAL
<odległość>:	Odniesienie do wektora z przyrostową drogą ruchu:	
	<ul style="list-style-type: none"> <odległość> [0]: Odcięta <odległość> [1]: Rzędna <odległość> [2]: Aplikata 	
	Przy ustalonej pozycji stutysięcznej w <status>:	
	<odległość> zawiera wektor jednostkowy v jako wartość wyjściową, który definiuje dalszy kierunek ruchu w WCS.	
	<p>Przypadek 1: Tworzenie wektora v przy <TestID>, Bit 4 == 1</p> <p>Wektory sterowania <odległość> i <maks.odległość> mocują płaszczyznę ruchu. Płaszczyzna ta zostanie nacięta uszkodzoną powierzchnią ograniczającą. Prosta przecięcia obu płaszczyzn definiuje kierunek wektora v. Orientacja (znak) jest przy tym tak wybierana, że kąt pomiędzy wektorem sterowania <maks.odległość> i v nie jest większy niż 90 stopni.</p>	
	<p>Przypadek 2: Tworzenie wektora v przy <TestID>, Bit 5 == 1</p> <p>Wektor v stanowi wektor jednostkowy w kierunku projekcji zawartego w <odległość> wektora ruchu na powierzchni ograniczającej. Jeżeli projekcję wektora ruchu na powierzchni ograniczającej stanowi wektor zerowy, błąd zostanie zwrócony.</p>	
	Typ parametru:	Wejście/wyjście
Typ danych:	VAR REAL [3]:	
	Zakres wartości:	-maks. wartość REAL ≤ x[n] ≤ + maks. wartość REAL

<Limit>:	<p>Odniesienie do tablicy o długości 5.</p> <ul style="list-style-type: none"> • <Limit> [0 - 2]: Minimalny odstęp osi geometrycznych, odciętych, rzędnych, aplikat Pierwsze trzy elementy zawierają minimalne odstęp osi geometrycznych, które w przeciwieństwie do nadzorowanych granic muszą być zachowane. Są one zawsze stosowane w odniesieniu do ograniczenia obszaru roboczego oraz softwaerowych wyłączników krańcowych, gdy albo brak jest transformacji lub transformacja jest aktywna, w przypadku której możliwe jest przyporządkowanie osi geometrycznych do liniowych osi maszyny (np. transformacje 5-osiowe). • <Limit> [3]: Zawiera minimalny odstęp dla liniowych osi maszyny, które np. z powodu transformacji nieliniowej nie mogą zostać przyporządkowane do żadnej osi geometrycznej. Poza tym wartość ta jest stosowana jako wartość graniczna przy nadzorze tradycyjnych obszarów ochrony oraz obszarów ochrony służących do unikania kolizji. • <Limit> [4]: Zawiera minimalny odstęp dla obrotowych osi maszyny, które np. z powodu transformacji nieliniowej nie mogą zostać przyporządkowane żadnej osi geometrycznej. <p>Uwaga Wartość ta działa tylko w przypadku nadzoru softwaerowych włączników krańcowych transformacji specjalnych.</p> <table border="1" data-bbox="571 974 1481 1086"> <tr> <td>Typ parametru:</td> <td>Wejście</td> </tr> <tr> <td>Typ danych:</td> <td>VAR REAL [5]:</td> </tr> <tr> <td>Zakres wartości:</td> <td>-maks. wartość REAL ≤ x[n] ≤ + maks. wartość REAL</td> </tr> </table>	Typ parametru:	Wejście	Typ danych:	VAR REAL [5]:	Zakres wartości:	-maks. wartość REAL ≤ x[n] ≤ + maks. wartość REAL
Typ parametru:	Wejście						
Typ danych:	VAR REAL [5]:						
Zakres wartości:	-maks. wartość REAL ≤ x[n] ≤ + maks. wartość REAL						
<maks.odległość>:	<p>Odniesienie do wektora z przyrostową drogą ruchu, w przypadku którego zadany minimalny odstęp od granicy osi wszystkich uczestniczących osi maszyny nie został osiągnięty:</p> <ul style="list-style-type: none"> • <odległość> [0]: Odcięta • <odległość> [1]: Rzędna • <odległość> [2]: Aplikata <p>Jeżeli droga ruchu nie jest ograniczona, treść tego parametru zwrotnego jest równa treści <odległość>.</p> <p>Przy <TestID>, Bit 4 == 1: <odległość> i <maks.odległość> <maks.odległość> i <odległość> jako wartości wejściowe muszą zawierać wektory, które definiują płaszczyznę ruchu. Oba wektory muszą być liniowo niezależne od siebie. Wartość bezwzględna <maks.odległość> jest dowolna. W celu obliczenia kierunku ruchu patrz opis dla <odległość>.</p> <table border="1" data-bbox="571 1534 1481 1653"> <tr> <td>Typ parametru:</td> <td>Wyjście</td> </tr> <tr> <td>Typ danych:</td> <td>VAR REAL [3]:</td> </tr> <tr> <td>Zakres wartości:</td> <td>-maks. wartość REAL ≤ x[n] ≤ + maks. wartość REAL</td> </tr> </table>	Typ parametru:	Wyjście	Typ danych:	VAR REAL [3]:	Zakres wartości:	-maks. wartość REAL ≤ x[n] ≤ + maks. wartość REAL
Typ parametru:	Wyjście						
Typ danych:	VAR REAL [3]:						
Zakres wartości:	-maks. wartość REAL ≤ x[n] ≤ + maks. wartość REAL						

3.3 Sprawdzenie naruszenia obszaru ochrony, ograniczenia obszaru pracy oraz wyłączników krańcowych oprogramowania (CALCPOSI)

<System>:	System miar (całowy/metryczny) dla danych pozycji i długości (opcjonalnie)		
	Typ danych:	BOOL	
	Wartość domyślna:	FALSE	
	Wartość	Znaczenie	
	FALSE	System miar odpowiednio do aktualnie aktywnych funkcji G z grupy G 13 (G70, G71, G700, G710). Uwaga Przy aktywnej G70 i metrycznym systemie podstawowym lub aktywnej G71 i całowym systemie podstawowym zmienne systemowe \$AA_IW i \$AA_MW są dawane w systemie podstawowym i w celu zastosowania przez funkcję CALCPOSI () muszą zostać przeliczone.	
TRUE	System miar odpowiednio do ustawionego systemu podstawowego: MD52806 \$MN_ISO_SCALING_SYSTEM		
<TestLim>:	Kodowany bitowo wybór nadzorowanych ograniczeń (opcjonalnie)		
	Typ danych:	INT	
	Wartość domyślna:	Bit 0, 1, 2, 3 == 1 (15)	
	Bit	Dziesiętnie	Znaczenie
	0	1	Softwareowy wyłącznik krańcowy
	1	2	Ograniczenie obszaru pracy
	2	4	Uaktywnione tradycyjne obszary ochronne
	3	8	Wstępnie aktywowane tradycyjne obszary ochrony
	4	16	Przy uszkodzonych softwaerowych wyłącznikach krańcowych lub ograniczeniach obszaru roboczego w <odległość> kierunku ruchu należy zwrócić odpowiednio do Fall 1 .
	5	32	Przy uszkodzonych softwaerowych wyłącznikach krańcowych lub ograniczeniach obszaru roboczego w <odległość> kierunku ruchu należy zwrócić odpowiednio do Fall 2 .
	6	64	Uaktywnione obszary ochronne unikania kolizji
	7	128	Wstępnie aktywowane obszary ochronne służące do unikania kolizji
8	256	Pary uaktywnionych oraz wstępnie aktywowanych obszarów ochrony służących do unikania kolizji	

Literatura

Podręcznik działania Funkcje podstawowe, (A3) nadzór osi, obszary ochrony, rozdział "Obszary ochrony"

Specjalne polecenia wykonania ruchu

4.1 Ruch do pozycji kodowanych (CAC, CIC, CDC, CACP, CACN)

Funkcja

Przez poniższe polecenia można wykonywać ruchy w osiach liniowych i obrotowych do stałych pozycji w osiach, zapisanych w tablicach danych maszynowych. Ten rodzaj programowania jest określany jako "ruchy do pozycji kodowanych".

Składnia

```
CAC (<n>)
CIC (<n>)
CACP (<n>)
CACN (<n>)
```

Znaczenie

CAC (<n>)	Ruch do pozycji kodowanej o numerze n
CIC (<n>)	Ruch do pozycji kodowanej, wychodząc od aktualnego numeru pozycji, o n miejsc do przodu (+n) albo do tyłu (-n)
CDC (<n>)	Ruch do pozycji kodowanej o numerze n po najkrótszej drodze (tylko dla osi obrotowych)
CACP (<n>)	Ruch do pozycji kodowanej o numerze n w kierunku dodatnim (tylko dla osi obrotowych)
CACN (<n>)	Dosunięcie do pozycji kodowanej o numerze n w kierunku ujemnym (tylko dla osi obrotowych)
<n>	Numer pozycji w ramach tablicy danych maszynowych Zakres wartości: 0, 1, ... (maks. liczba miejsc w tablicy - 1)

Przykład: Ruch do kodowanych pozycji osi pozycjonowania

Kod programu	Komentarz
N10 FA[B]=300	; Posuw dla osi pozycjonowania B
N20 POS[B]=CAC(10)	; Ruch do pozycji kodowanej o numerze 10
N30 POS[B]=CIC(-4)	; Ruch do pozycji kodowanej od "aktualnego numeru pozycji" - 4

Literatura

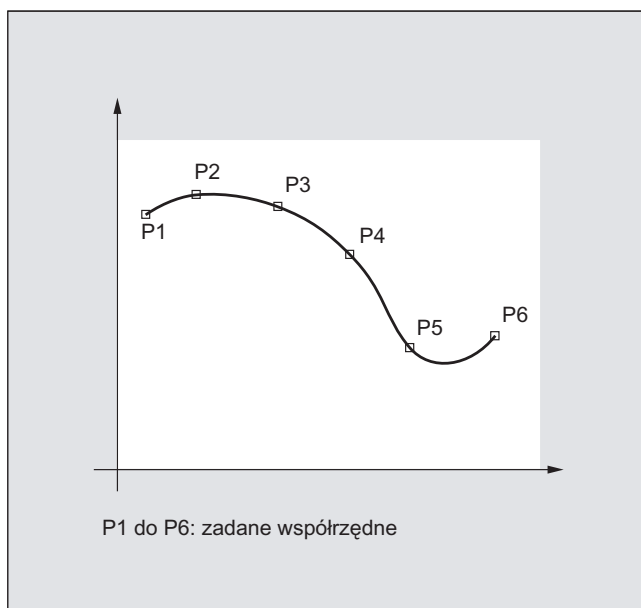
- Podręcznik działania, Funkcje rozszerzające; Osie podziałowe (T1)
- Podręcznik działania Akcje synchroniczne

4.2 Interpolacja Spline (ASPLINE, BSPLINE, CSPLINE, BAUTO, BNAT, BTAN, EAUTO, ENAT, ETAN, PW, SD, PL)

Funkcja

Dowolnie zakrzywione kontury na obrabianych przedmiotach nie mogą zostać dokładnie opisane analitycznie. Tego rodzaju kontury są dlatego przybliżane przez ograniczoną liczbę punktów oparcia, np. przy digitalizacji powierzchni. W celu utworzenia powierzchni digitalizowanej na obrabianym przedmiocie punkty oparcia muszą zostać połączone w opis konturu. Umożliwia to interpolacja Spline.

Spline definiuje krzywą, która składa się z wielomianów 2. albo 3. stopnia. Właściwości w punktach oparcia spline są definiowane **zależnie od zastosowanego typu spline**.



W przypadku SINUMERIK solution line są do dyspozycji następujące typy spline:

- A-Spline
- B-Spline
- C-Spline

Składnia

Informacje ogólne:

```
ASPLINE X... Y... Z... A... B... C...
BSPLINE X... Y... Z... A... B... C...
CSPLINE X... Y... Z... A... B... C...
```

Dodatkowo programowane w przypadku B-Spline:

```
PW=<n>
SD=2
PL=<wartość>
```

Dodatkowo programowane w przypadku A-Spline i C-Spline:

```
BAUTO / BNAT / BTAN
EAUTO / ENAT / ETAN
```

Znaczenie

Typ interpolacji spline:

ASPLINE	Polecenie do włączenia interpolacji A-Spline
BSPLINE	Polecenie do włączenia interpolacji B-Spline
CSPLINE	Polecenie do włączenia interpolacji C-Spline

Polecenia ASPLINE, BSPLINE i CSPLINE działają modalnie i należą do grupy poleceń drogowych.

Punkty oparcia wzgl. punkty kontrolne:

X... Y... Z... A... B... C...	Pozycje we współrzędnych kartezjańskich
----------------------------------	---

Waga punktu (tylko B-Spline):

PW	Przy pomocy polecenia PW jest dla każdego punktu oparcia możliwe zaprogramowanie tak zwanej "wagi punktu".				
<n>	"Waga punktu"				
Zakres wartości:	$0 \leq n \leq 3$				
Długość kroku:	0.0001				
Działanie:	<table> <tr> <td>$n > 1$</td> <td>Krzywa jest silniej przyciągana przez punkt kontrolny.</td> </tr> <tr> <td>$n < 1$</td> <td>Krzywa jest słabiej przyciągana przez punkt kontrolny.</td> </tr> </table>	$n > 1$	Krzywa jest silniej przyciągana przez punkt kontrolny.	$n < 1$	Krzywa jest słabiej przyciągana przez punkt kontrolny.
$n > 1$	Krzywa jest silniej przyciągana przez punkt kontrolny.				
$n < 1$	Krzywa jest słabiej przyciągana przez punkt kontrolny.				

Stopień spline (tylko B-Spline):

SD	Standardowo jest stosowany wielokąt 3. stopnia. Przez zaprogramowanie SD=2 można jednak użyć również wielokąt 2. stopnia.
----	---

Odstęp węzłów (tylko B-Spline):

PL	Odstępy węzłów są wewnętrznie odpowiednio obliczane. Sterowanie może jednak realizować również zadane odstępy węzłów, które przy pomocy polecenia PL są podawane jako tzw. długość przedziału między parametrami.
----	---

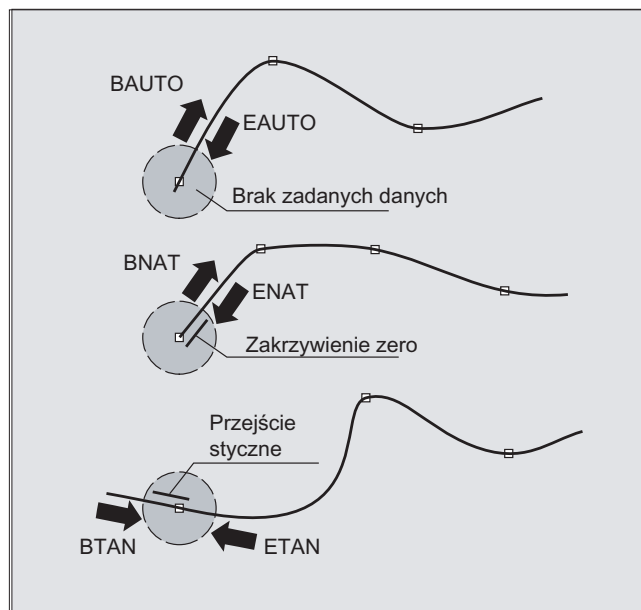
<wartość> Parametr - przedział - długość
Zakres wartości: jak wymiar drogi

Zachowanie się na przejściu na początku krzywej spline (tylko A-Spline lub C-Spline):

BAUTO Brak zadania zachowania się na przejściu. Początek wynika z położenia pierwszego punktu.
BNAT Zakrzywienie zero
BTAN Przejście styczne do poprzedniego bloku (położenie kasowania)

Zachowanie się na przejściu na końcu krzywej spline (tylko A-Spline lub C-Spline):

EAUTO Brak zadania zachowania się na przejściu. Koniec wynika z położenia ostatniego punktu.
ENAT Zakrzywienie zero
ETAN Przejście styczne do poprzedniego bloku (położenie kasowania)



Wskazówka

Programowane zachowanie się na przejściu nie ma wpływu na B-Spline. B-Spline jest w punkcie startowym i końcowym zawsze styczny do wielokąta kontrolnego.

Warunki brzegowe

- Korekcję promienia narzędzia można stosować.
- Nadzór na kolizję następuje w rzucie na płaszczyznę.

Przykłady

Przykład 1: B-Spline

Kod programu 1 (wszystkie wagi 1)

```

N10 G1 X0 Y0 F300 G64
N20 BSPLINE
N30 X10 Y20
N40 X20 Y40
N50 X30 Y30
N60 X40 Y45
N70 X50 Y0

```

Kod programu 2 (różne wagi)

```

N10 G1 X0 Y0 F300 G64
N20 BSPLINE
N30 X10 Y20 PW=2
N40 X20 Y40
N50 X30 Y30 PW=0.5
N60 X40 Y45
N70 X50 Y0

```

Kod programu 3 (wielokąt kontrolny)

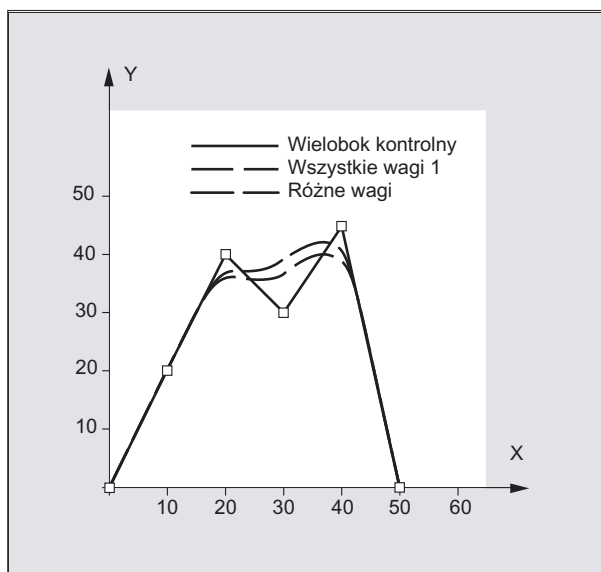
```

N10 G1 X0 Y0 F300 G64
N20
N30 X10 Y20
N40 X20 Y40
N50 X30 Y30
N60 X40 Y45
N70 X50 Y0

```

Komentarz

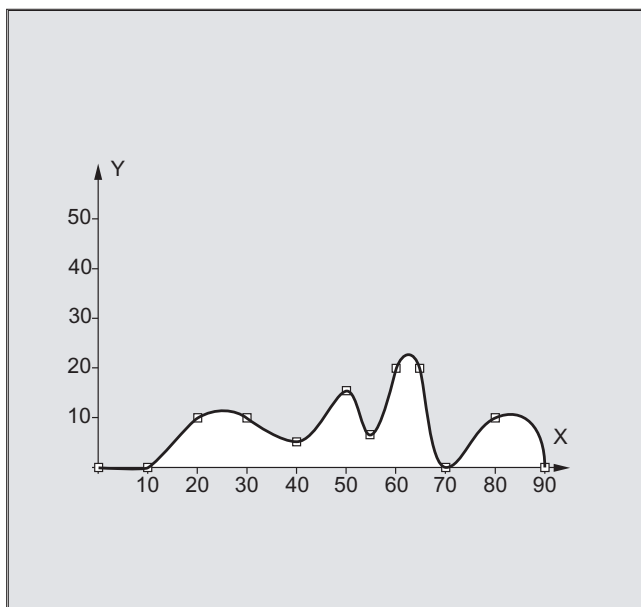
; Odpada



Przykład 2: C-Spline, na początku i na końcu zakrzywienie zero

Kod programu

```
N10 G1 X0 Y0 F300
N15 X10
N20 BNAT ENAT
N30 CSPLINE X20 Y10
N40 X30
N50 X40 Y5
N60 X50 Y15
N70 X55 Y7
N80 X60 Y20
N90 X65 Y20
N100 X70 Y0
N110 X80 Y10
N120 X90 Y0
N130 M30
```



Przykład 3: Interpolacja spline (A-Spline) i transformacja współrzędnych (ROT)

Program główny:

Kod programu	Komentarz
N10 G00 X20 Y18 F300 G64	; Ruch do punktu startowego.
N20 ASPLINE	; Uaktywnienie typu interpolacji A-Spline.
N30 KONTUR	; Pierwsze wywołanie podprogramu.
N40 ROT Z-45	; Transformacja współrzędnych: obrót WKS o -45° wokół osi Z.

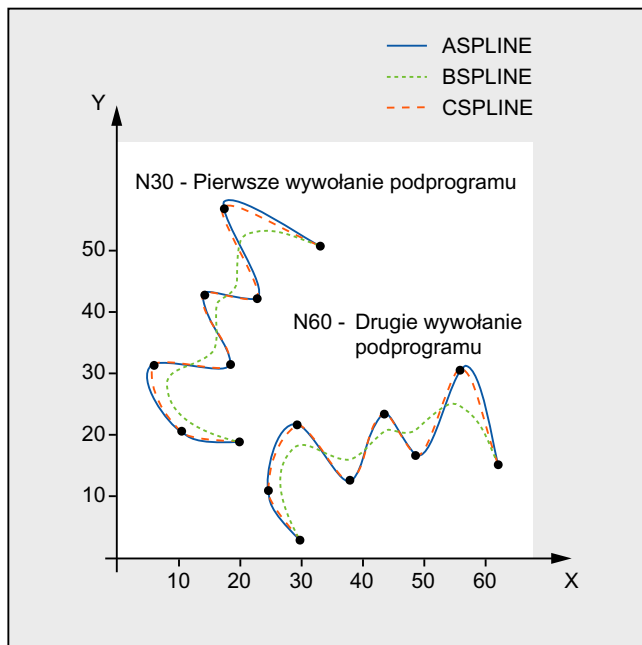
4.2 Interpolacja Spline (ASPLINE, BSPLINE, CSPLINE, BAUTO, BNAT, BTAN, EAUTO, ENAT, ETAN, PW, SD, PL)

Kod programu	Komentarz
N50 G00 X20 Y18	; Ruch do punktu początkowego konturu.
N60 KONTUR	; Drugie wywołanie podprogramu.
N70 M30	; Koniec programu

Podprogram "Kontur" (zawiera współrzędne punktu oparcia):

Kod programu
N10 X20 Y18
N20 X10 Y21
N30 X6 Y31
N40 X18 Y31
N50 X13 Y43
N60 X22 Y42
N70 X16 Y58
N80 X33 Y51
N90 M1

Na poniższym rysunku są oprócz krzywej spline, która wynika z przykładu programu (ASPLINE), zawarte również krzywe spline, które wynikłyby przy uaktywnieniu interpolacji B-Spline albo C-Spline



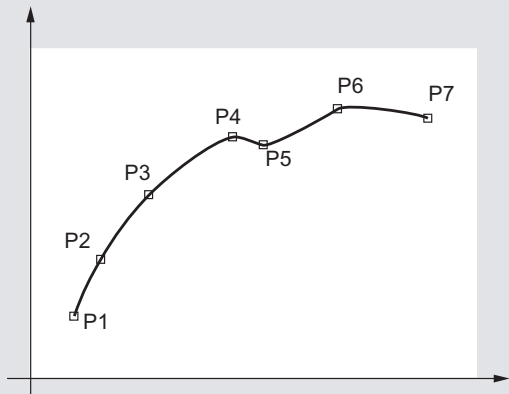
Dalsze informacje

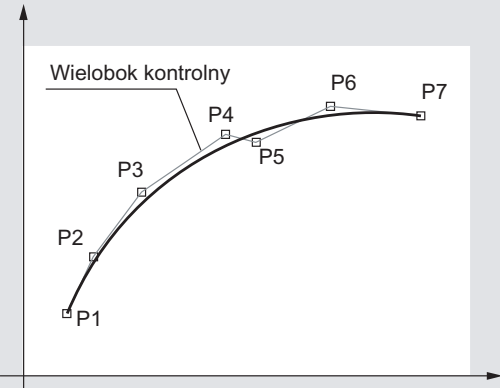
Zalety interpolacji spline

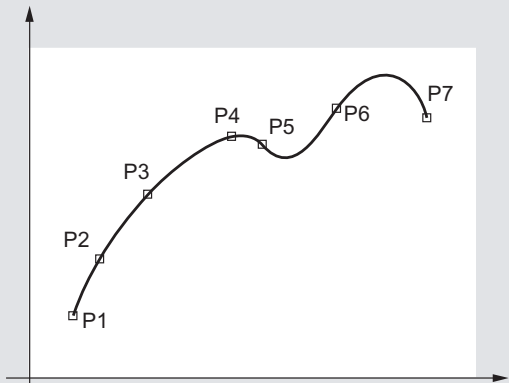
Poprzez zastosowanie interpolacji spline można, w przeciwieństwie do zastosowania bloków prostoliniowych G01, uzyskać następujące zalety:

- Zmniejszenie liczby potrzebnych bloków programu obróbki do opisanego konturu
- Miękki, oszczędzający mechanikę, przebieg krzywej przy przejściach między blokami programu obróbki

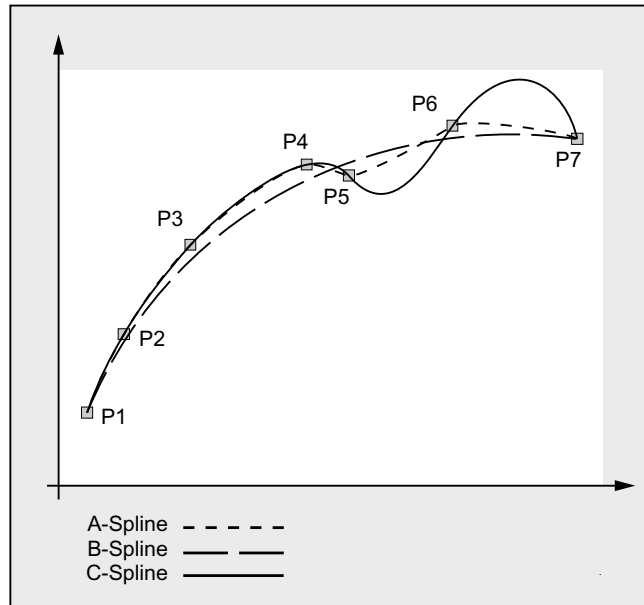
Właściwości i zastosowanie różnych typów spline

Typ spline	Właściwości i zastosowanie
<p>A-Spline</p>	<div data-bbox="574 817 1220 1422" style="border: 1px solid black; padding: 10px; margin-bottom: 10px;"> <p style="text-align: center;">A-Spline (spline Akima)</p>  <p style="text-align: center;">P1 do P7: zadane współrzędne</p> </div> <p>Właściwości:</p> <ul style="list-style-type: none"> • Przebiega ściśle przez zadane punkty oparcia. • Przebieg krzywej ma ciągłą pochodną, ale nie stałe zakrzywienie. • Prawie nie wytwarza niepotrzebnych drgań. • Obszar wpływu zmian punktów oparcia jest lokalny, tzn. zmiana punktu oparcia ma wpływ na max 6 sąsiednich punktów oparcia. <p>Zastosowanie:</p> <p>A-Spline nadaje się przede wszystkim do interpolacji przebiegów krzywych i dużych zmian nachylenia (np. przebiegi schodkowe).</p>

Typ spline	Właściwości i zastosowanie
B-Spline	<div data-bbox="611 398 1257 1003" style="border: 1px solid black; padding: 10px; margin-bottom: 10px;"> <p style="text-align: center;">B-Spline</p>  <p style="text-align: center;">P1 do P7: zadane współrzędne</p> </div> <p>Właściwości:</p> <ul style="list-style-type: none"> • Przebiega nie przez zadane punkty oparcia lecz tylko w ich pobliżu. Krzywa jest przyciągana przez punkty oparcia. Poprzez ważenie punktów oparcia za pomocą współczynnika można dodatkowo wpływać na przebieg krzywej. • Przebieg krzywej ma stałą pochodną i stałe zakrzywienie. • Nie wytwarza niepotrzebnych drgań. • Obszar wpływu zmian punktów oparcia jest lokalny, tzn. zmiana punktu oparcia ma wpływ na max 6 sąsiednich punktów oparcia. <p>Zastosowanie:</p> <p>B-Spline pierwotnie pomyślano, jako interfejs do systemów CAD.</p>

Typ spline	Właściwości i zastosowanie
<p>C-Spline</p>	<div data-bbox="571 427 1219 1032" style="border: 1px solid black; padding: 10px; margin-bottom: 10px;"> <p style="text-align: center;">C-Spline (spline sześcienny)</p>  <p style="text-align: center;">P1 do P7: zadane współrzędne</p> </div> <p>Właściwości:</p> <ul style="list-style-type: none"> • Przebiega ściśle przez zadane punkty oparcia. • Przebieg krzywej ma stałą pochodną i stałe zakrzywienie. • Wytwarza często niepotrzebne drgania, w szczególności w miejscach o dużych zmianach nachylenia. • Zakres wpływu zmian punktów oparcia jest globalny, tzn. zmiana jednego punktu oparcia ma wpływ na cały przebieg krzywej. <p>Zastosowanie:</p> <p>C-Spline można tylko wtedy dobrze stosować, gdy punkty oparcia leżą na analitycznie znanej krzywej (okrąg, parabola, hiperbola)</p>

Porównanie trzech typów spline przy takich samych punktach oparcia



Liczba minimalna bloków spline

G-Code `ASPLINE`, `BSPLINE` i `CSPLINE` łączą punkty końcowe bloków przy pomocy spline. W tym celu musi w przebiegu wyprzedzającym być równocześnie obliczanych szereg bloków (punktów końcowych). Wielkość bufora do obliczania wynosi standardowo 10 bloków. Nie każda informacja zawarta w bloku jest punktem końcowym spline. Sterowanie potrzebuje jednak z 10 bloków określonej liczby bloków z punktem końcowym spline.

Typ spline	Liczba minimalna bloków spline
A-Spline:	Z każdych 10 bloków co najmniej 4 muszą być blokami spline. Nie wlicza się do tego bloków komentarzowych i obliczeń parametrów.
B-Spline:	Z każdych 10 bloków co najmniej 6 muszą być blokami spline. Nie wlicza się do tego bloków komentarzowych i obliczeń parametrów.
C-Spline:	Potrzebna minimalna liczba bloków spline wynika z następującej sumy: Wartość z MD20160 \$MC_CUBIC_SPLINE_BLOCKS + 1 W MD20160 jest wpisywana liczba punktów, z których jest obliczany segment spline. Ustawienie standardowe wynosi 8. Z 10 bloków musi dlatego w przypadku standardowym być 9 bloków spline.

Wskazówka

Przy zejściu poniżej tolerowalnej wartości jest wyprowadzany alarm, tak samo gdy oś uczestnicząca w spline zostanie zaprogramowana jako oś pozycjonowania.

Połączenie krótkich bloków Spline

W przypadku interpolacji spline mogą powstawać krótkie bloki spline, które prowadzą do niepotrzebnego zmniejszenia prędkości ruchu po torze. Przy pomocy funkcji "połączenie krótkich bloków spline" bloki te mogą zostać tak połączone, by wynikowa długość bloku była wystarczająco duża i nie prowadziła do zmniejszenia prędkości ruchu po torze.

Funkcja jest uaktywniana poprzez kanałową daną maszynową:

MD20488 \$MC_SPLINE_MODE (ustawienie dla interpolacji spline)

Literatura:

Podręcznik działania Funkcje podstawowe; Tryb przechodzenia płynnego, Zatrzymanie dokładne, LookAhead (B1),
punkt: Połączenie krótkich bloków spline

4.3 Zespół spline (SPLINEPATH)

Funkcja

Osie, które mają interpolować w zespole spline, są wybierane poleceniem `SPLINEPATH`. Przy interpolacji spline jest możliwych do ośmiu osi uczestniczących w tworzeniu konturu.

Wskazówka

Jeżeli `SPLINEPATH` nie zostanie explicite zaprogramowane, wówczas jako zespół spline wykonują ruch pierwsze trzy osie kanału.

Składnia

Ustalenie zespołu spline następuje w oddzielnym bloku:

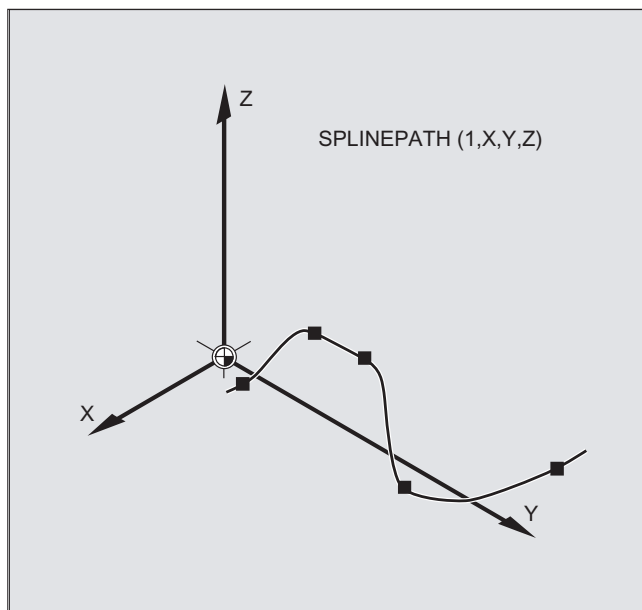
```
SPLINEPATH (n, X, Y, Z, ...)
```

Znaczenie

<code>SPLINEPATH</code>	Polecenie do ustalenia zespołu spline
<code>n</code>	=1 (stała wartość)
<code>X, Y, Z, ...</code>	Identyfikatory osi uczestniczących w tworzeniu konturu, które mają interpolować w zespole spline

Przykład: zespół Spline z trzema osiami uczestniczącymi w tworzeniu konturu

Kod programu	Komentarz
N10 G1 X10 Y20 Z30 A40 B50 F350	
N11 SPLINEPATH(1,X,Y,Z)	; Zespół spline
N13 CSPLINE BAUTO EAUTO X20 Y30 Z40 A50 B60	; C-Spline
N14 X30 Y40 Z50 A60 B70	; Punkty oparcia
...	
N100 G1 X... Y...	; Cofnięcie wyboru interpolacji spline



4.4 Kompresja bloków NC (COMPON, COMPCURV, COMPCAD, COMPOF)

Funkcja

Systemy CAD/CAM dają z reguły bloki liniowe, które dotrzymują sparametryzowanej dokładności. W przypadku skomplikowanych konturów prowadzi to do znacznej ilości danych i do ewentualnie krótkich odcinków torów. Te krótkie odcinki ograniczają prędkość wykonywania.

Przez zastosowanie funkcji kompresora następuje przybliżenie do konturu zadanego przez bloki liniowe poprzez użycie bloków wielomianowych. Wynikają z tego następujące zalety:

- Zmniejszenie liczby bloków programu obróbki potrzebnych do opisanego konturu obrabianego przedmiotu
- Stałe przejścia między blokami
- Zwiększenie maksymalnie możliwych prędkości ruchu po torze

4.4 Kompresja bloków NC (COMPON, COMPCURV, COMPCAD, COMPOF)

Są do dyspozycji następujące funkcje kompresora:

- COMPON

Przejścia między blokami są stałe tylko pod względem **prędkości**, podczas gdy przyspieszenie uczestniczących osi może na przejściach między blokami wykonywać skoki.

- COMPCURV

Przejścia między blokami mają **stałe przyspieszenie**. Przez to jest zagwarantowany gładki przebieg zarówno prędkości, jak i przyspieszenia wszystkich osi na przejściach między blokami.

- COMPCAD

Kompresja wymagająca długiego czasu obliczania i dużo miejsca w pamięci, zoptymalizowana pod względem jakości powierzchni i prędkości. COMPCAD należy stosować tylko wtedy, gdy środków do polepszenia powierzchni nie można z góry użyć z programu CAD/CAM.

Funkcja kompresora ulega zakończeniu przy pomocy COMPOF.

Składnia

COMPON
COMPCURV
COMPCAD
COMPOF

Znaczenie

COMPON:	Polecenie do włączenia funkcji kompresora COMPON. Działanie: Modalnie
COMPCURV:	Polecenie do włączenia funkcji kompresora COMPCURV. Działanie: Modalnie
COMPCAD:	Polecenie do włączenia funkcji kompresora COMPCAD. Działanie: Modalnie
COMPOF:	Polecenie do wyłączenia aktualnie aktywnej funkcji kompresora.

Wskazówka

W celu dodatkowego polepszenia jakości powierzchni można użyć funkcji ścinania narożników G64.2 i ograniczenia przyspieszenia drugiego stopnia SOFT. Te polecenia należy pisać na początku programu.

Warunki brzegowe

- Kompresja bloków NC jest z reguły przeprowadzana dla bloków liniowych (G1).
- Są kompresowane tylko bloki, które spełniają wymóg prostej składni:
N... G1X... Y... Z... F... ;Komentarz
Wszystkie inne bloki są wykonywane bez zmian (bez kompresji).
- Bloki ruchu z adresami rozszerzonymi, jak C=100 albo A=AC(100) są również kompresowane.
- Wartości pozycji nie muszą być bezpośrednio programowane, lecz mogą być podawane również pośrednio poprzez przyporządkowania parametrów, np. X=R1*(R2+R3).
- Gdy jest do dyspozycji opcja "Transformacja orientacji", wówczas mogą być kompresowane również bloki NC, w których jest zaprogramowana orientacja narzędzia (i ew. również obrót narzędzia) przy pomocy wektorów kierunkowych (patrz "Kompresowanie orientacji (Strona 348)").
- Proces kompresji jest przerywany przez każdą inną instrukcję NC, np. wyprowadzenie funkcji pomocniczej.

Przykłady

Przykład 1: COMPON

Kod programu	Komentarz
N10 COMPON	; Funkcja kompresora COMPON wł.
N11 G1 X0.37 Y2.9 F600	; G1 przed punkt końcowy i posuw.
N12 X16.87 Y-.698	
N13 X16.865 Y-.72	
N14 X16.91 Y-.799	
...	
N1037 COMPOF	; Funkcja kompresora wył.
...	

Przykład 2: COMPCAD

Kod programu	Komentarz
G00 X30 Y6 Z40	
G1 F10000 G642	; Funkcja ścinania narożników G642 wł.
SOFT	; Ograniczenie przyśpieszenia drugiego stopnia SOFT wł.
COMPCAD	; Funkcja kompresora COMPCAD wł.
STOPFIFO	
N24050 Z32.499	
N24051 X41.365 Z32.500	
N24052 X43.115 Z32.497	

4.5 Interpolacja wielomianowa (POLY, POLYPATH, PO, PL)

Kod programu	Komentarz
N24053 X43.365 Z32.477	
N24054 X43.556 Z32.449	
N24055 X43.818 Z32.387	
N24056 X44.076 Z32.300	
...	
COMPOF	; Funkcja kompresora wyłącz.
G00 Z50	
M30	

Literatura

Podręcznik działania Funkcje podstawowe; Tryb przechodzenia płynnego, Zatrzymanie dokładne, LookAhead (B1),
Rozdział: "Kompresja bloków NC"

4.5 Interpolacja wielomianowa (POLY, POLYPATH, PO, PL)

Funkcja

W przypadku interpolacji wielomianowej (POLY) chodzi właściwie nie o rodzaj interpolacji spline. Jest ona w pierwszym rzędzie pomyślana jako interfejs do programowania zewnętrznie wytwarzanych krzywych spline. Przy tym odcinki spline mogą być programowane bezpośrednio.

Ten rodzaj interpolacji odciąża NC od obliczania współczynników wielomianu. Daje się ona optymalnie stosować wtedy, gdy współczynniki przychodzą bezpośrednio od systemu CAD albo postprocesora.

Składnia

Wielomian 3. stopnia:

POLY PO [X] = (xe, a2, a3) PO [Y] = (ye, b2, b3) PO [Z] = (ze, c2, c3) PL=n

Wielomian 5. stopnia i nowa składnia wielomianu:

POLY X=PO (xe, a2, a3, a4, a5) Y=PO (ye, b2, b3, b4, b5) Z=PO (ze, c2, c3, c4, c5)

PL=n

POLYPATH ("AXES", "VECT")

Wskazówka

Suma zaprogramowanych w jednym bloku współczynników wielomianu i osi nie może przekraczać maksymalnej dozwolonej w bloku liczby osi.

Znaczenie

POLY :	Włączenie interpolacji wielomianowej przy pomocy bloku z POLY.
POLYPATH :	Interpolacja wielomianowa wybierana dla obydwu grup osi AXIS lub VECT
PO[identyfikator osi/zmienna] :	Punkty końcowe i współczynniki wielomianu
X, Y, Z :	Identyfikatory osi
x e , ye, ze :	Podanie pozycji końcowej dla każdorazowej osi; zakres wartości jak miara drogi
a2, a3, a4, a5 :	Współczynniki a ₂ , a ₃ , a ₄ , i a ₅ są pisane z ich wartościami; zakres wartości jak miara drogi. Każdorazowo ostatni współczynnik można pominąć, gdy ma on wartość zero.
PL :	Długość przedziału parametru, na którym wielomiany są zdefiniowane (zakres definicji funkcji f(p)). Przedział rozpoczyna się zawsze od 0, p może przyjmować wartości od 0 do PL. Teoretyczny zakres wartości dla PL: 0,0001 ... 99 999,9999
	Wskazówka: Wartość PL obowiązuje dla bloku, w którym się znajduje. Gdy PL nie zaprogramowano, działa PL=1.

Włączenie/wyłączenie interpolacji wielomianowej

Interpolacja wielomianowa jest w programie obróbki włączana poleceniem G POLY.

Polecenie G POLY należy razem z G0, G1, G2, G3, ASPLINE, BSPLINE i CSPLINE do 1. grupy G.

Osie, które są zaprogramowane przez podanie tylko nazwy i punktu końcowego (np. X10), wykonują ruch liniowy. Gdy wszystkie osie bloku NC są tak zaprogramowane, sterowanie zachowuje się tak, jak w przypadku G1.

Interpolacja wielomianowa jest przez zaprogramowanie innego polecenia z 1. grupy G (np. G0, G1) implicite ponownie wyłączana.

Współczynnik wielomianu

Wartość PO (PO[]=) lub ...=PO(...) podaje wszystkie współczynniki wielomianu dla osi. Odpowiednio do stopnia wielomianu podaje się wiele wartości rozdzielonych przecinkiem. W ramach jednego bloku są możliwe różne stopnie wielomianów dla różnych osi.

Wywołanie podprogramu POLYPATH

Przy pomocy POLYPATH(...) może zostać udostępniona interpolacja wielomianowa selektywnie dla określonych grup osi:

Tylko osie uczestniczące w tworzeniu konturu i osie dodatkowe: POLYPATH("AXES")
 Tylko osie orientacji: POLYPATH("VECT")
 (przy wykonywaniu ruchu z transformacją orientacji)

Każdorazowo nie udostępnione osie wykonują ruch liniowy.

Standardowo jest udostępniona interpolacja wielomianowa dla obydwu grup osi.

Przez zaprogramowanie bez parametru POLYPATH() jest wyłączana interpolacja wielomianowa dla wszystkich osi.

Przykład

Kod programu	Komentarz
N10 G1 X... Y... Z... F600	
N11 POLY PO[X]=(1,2.5,0.7) PO[Y]=(0.3,1,3.2) PL=1.5	; Interpolacja wielomianowa wł.
N12 PO[X]=(0,2.5,1.7) PO[Y]=(2.3,1.7) PL=3	
...	
N20 M8 H126 ...	
N25 X70 PO[Y]=(9.3,1,7.67) PL=5	; Dane mieszane dla osi
N27 PO[X]=(10,2.5) PO[Y]=(2.3)	; PL nie zaprogramowano; działa PL=1
N30 G1 X... Y... Z.	; Interpolacja wielomianowa wyłą.
...	

Przykład: Nowa składnia wielomianu

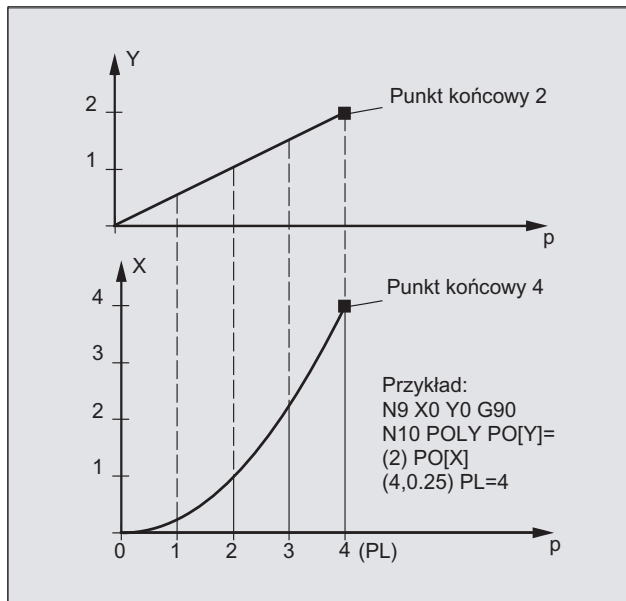
Dalej obowiązująca składnia wielomianowa	Nowa składnia wielomianu
PO[identyfikator osi]=(.. , ..)	Identyfikator osi=PO(.. , ..)
PO[PHI]=(.. , ..)	PHI=PO(.. , ..)
PO[PSI]=(.. , ..)	PSI=PO(.. , ..)
PO[THT]=(.. , ..)	THT=PO(.. , ..)
PO[]=(.. , ..)	PO(.. , ..)
PO[zmienna]=IC(.. , ..)	zmienna=PO IC(.. , ..)

Przykład: Krzywa na płaszczyźnie X/Y

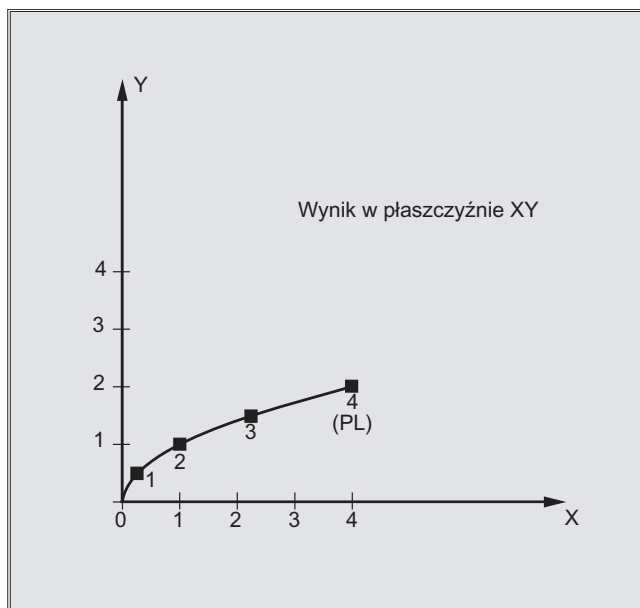
Programowanie

Kod programu
N9 X0 Y0 G90 F100
N10 POLY PO[Y]=(2) PO[X]=(4,0.25) PL=4

Przebieg krzywych X(p) i Y(p)



Przebieg krzywej w płaszczyźnie XY



Opis

Ogólna postać funkcji wielomianowej brzmi:

$$f(p) = a_0 + a_1p + a_2p^2 + \dots + a_np^n$$

gdzie: a_n : stałe współczynniki
 p : Parametry

W sterowaniu można programować maksymalnie wielomiany 5. stopnia:

$$f(p) = a_0 + a_1p + a_2p^2 + a_3p^3 + a_4p^4 + a_5p^5$$

Przez wyposażanie współczynników w konkretne wartości można tworzyć różne przebiegi krzywych, jak funkcje prostoliniowe, paraboliczne, potęgowe.

Prosta jest tworzona przez $a_2 = a_3 = a_4 = a_5 = 0$:

$$f(p) = a_0 + a_1p$$

Dalej obowiązuje:

a_0 : Pozycja osi na końcu bloku poprzedzającego
 $p = PL$
 $a_1 = (x_E - a_0 - a_2 \cdot p^2 - a_3 \cdot p^3) / p$

Jest możliwe programowanie wielomianów, **bez** bez uaktywnienia interpolacji wielomianowej poleceniem G_{POLY} . W tym przypadku nie są interpolowane zaprogramowane wielomiany, lecz następuje ruch liniowy do zaprogramowanych punktów końcowych osi (G_1). Dopiero po uaktywnieniu *explicite* interpolacji wielomianowej w programie obróbki ($POLY$) ruch dla zaprogramowanych wielomianów jest wykonywany, jak dla wielomianów.

Cecha szczególna: wielomian mianownikowy

Dla osi geometrycznych można przy pomocy $PO[] = (...)$ bez podania nazwy osi zaprogramować również wspólny wielomian mianownikowy, tzn. ruch w osiach geometrycznych jest wykonywany jako iloraz dwóch wielomianów.

W ten sposób dają się dokładnie przedstawiać np. krzywe stożkowe (okrąg, elipsa, parabola, hiperbola).

Przykład:

Kod programu	Komentarz
POLY G90 X10 Y0 F100	; Osie geometryczne wykonują ruch liniowy do pozycji X10, Y0.
PO[X]=(0,-10) PO[Y]=(10) PO[]=(2,1)	; Osie geometryczne wykonują ruch po ćwierćokręgu do X0, Y10.

Stały współczynnik (a_0) wielomianu mianownikowego jest stale przyjmowany o wielkości 1. Zaprogramowany punkt końcowy jest niezależny od G_{90} / G_{91} .

Z zaprogramowanych wartości następuje obliczenie $X(p)$ i $Y(p)$:

$$X(p) = (10 - 10 * p^2) / (1 + p^2)$$

$$Y(p) = 20 * p / (1 + p^2)$$

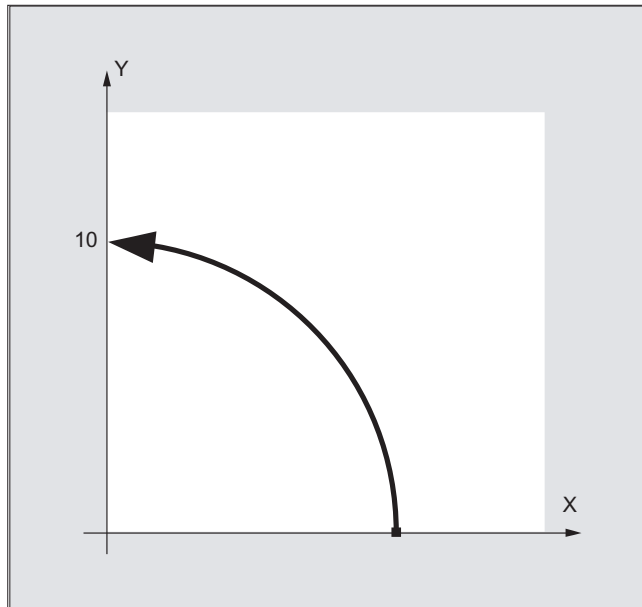
gdzie $0 \leq p \leq 1$

Z zaprogramowanych punktów początkowych, punktów końcowych, współczynnika a_2 i $PL=1$ wynikają następujące wartości pośrednie:

$$\text{Licznik (X)} = 10 + 0 * p - 10 * p^2$$

$$\text{Licznik (Y)} = 0 + 20 * p + 0 * p^2$$

$$\text{Mianownik} = 1 + p^2$$



Przy włączonej interpolacji wielomianowej zaprogramowanie wielomianu mianownikowego z miejscami zerowymi w ramach przedziału $[0, PL]$ jest odrzucone z alarmem. Na ruch osi dodatkowych wielomian mianownikowy nie ma wpływu.

Wskazówka

Korekcja promienia narzędzia daje się przy interpolacji wielomianowej włączyć przy pomocy G41, G42 i stosować jak dla interpolacji prostoliniowej albo kołowej.

4.6 Ustawiane odniesienie do toru (SPATH, UPATH)

Funkcja

Podczas interpolacji wielomianowej użytkownik może życzyć sobie dwóch różnych zależności między określającymi prędkość osiami FGROUP i pozostałymi osiami uczestniczącymi w tworzeniu konturu: Te ostatnie powinny być prowadzone albo synchronicznie do drogi po torze S albo synchronicznie do parametru U krzywej osi FGROUP.

Obydwa rodzaje interpolacji ruchu po torze są potrzebne w różnych aplikacjach i mogą być ustawiane/przełączane przez obydwie zawarte w 45. grupie G-Code, modalnie działające polecenia językowe `SPATH` i `UPATH`.

Składnia

```
SPATH  
UPATH
```

Znaczenie

`SPATH`: Odniesieniem toru dla osi FGROUP jest długość łuku
`UPATH`: Odniesieniem toru dla osi FGROUP jest parametr krzywej

Wskazówka

`UPATH` i `SPATH` określają również współzależność wielomianu słowa F (`FPOLY`, `FCUB`, `FLIN`) z ruchem po torze.

Warunki brzegowe

Ustawione odniesienie do toru nie ma znaczenia przy:

- przy interpolacji liniowej i kołowej
- w blokach gwintowania
- gdy wszystkie osie uczestniczące w tworzeniu konturu są zawarte w FGROUP.

Przykłady

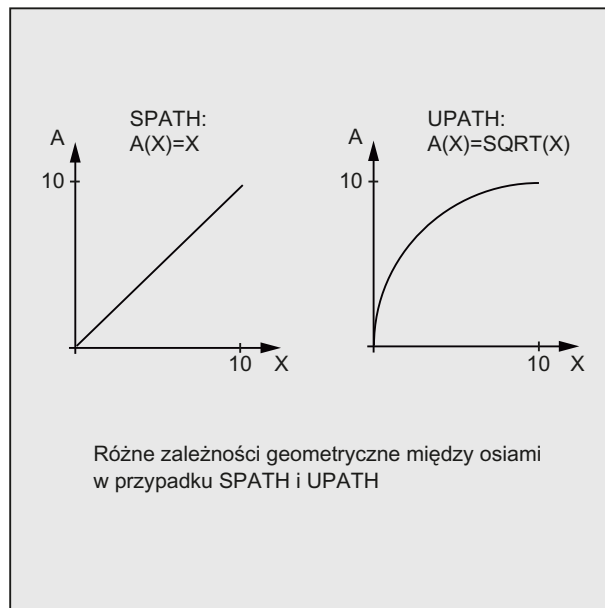
Przykład 1:

W poniższym przykładzie zostaną przy pomocy G643 ścięte naroża kwadratu o długości krawędzi 20 mm. Maksymalne odchylenia od dokładnego konturu są przy tym ustalane przez dane maszynowe MD33100 \$MA_COMPRESS_POS_TOL[...] dla każdej osi.

Kod programu	Komentarz
N10 G1 X... Y... Z... F500	
N20 G643	; Wewnętrzne w bloku ścięcie naroża przy pomocy G643
N30 X0 Y0	
N40 X20 Y0	; Długość krawędzi (mm) dla osi
N50 X20 Y20	
N60 X0 Y20	
N70 X0 Y0	
N100 M30	

Przykład 2:

Poniższy przykład ilustruje różnicę między obydwoimi rodzajami prowadzenia ruchu. Niech w obydwu przypadkach jest aktywne ustawienie domyślne FGROU(X,Y,Z).



Kod programu
N10 G1 X0 A0 F1000 SPATH
N20 POLY PO[X] = (10,10) A10

lub:

4.6 Ustawiane odniesienie do toru (SPATH, UPATH)

Kod programu

```
N10 G1 X0 F1000 UPATH  
N20 POLY PO[X]=(10,10) A10
```

W bloku N20 droga S osi FGROU**P** zależy od kwadratu parametru krzywej U. Dlatego wzdłuż drogi X wynikają różne pozycje osi synchronicznej A, w zależności od tego, czy jest aktywne SPATH czy UPATH.

Dalsze informacje

Podczas interpolacji wielomianowej - niech przez to będzie zawsze rozumiana interpolacja wielomianowa w węższym sensie (POLY), wszystkie rodzaje interpolacji spline (ASPLINE, BSPLINE, CSPLINE) i interpolacja liniowa z funkcją kompresora (COMPON, COMPCURV) - pozycje wszystkich osi uczestniczących w tworzeniu konturu są zadane przez wielomiany pi(U). Parametr krzywej U porusza się przy tym w ramach bloku NC od 0 do 1, jest więc znormalizowany.

Poprzez polecenie językowe FGROU**P** można w ramach osi uczestniczących w tworzeniu konturu wybrać te osie, do których ma się odnosić zaprogramowany posuw po torze. Interpolacja ze stałą prędkością na drodze S tych osi oznacza podczas interpolacji wielomianowej jednak z reguły zmienność parametru krzywej U.

Zachowanie się sterowania w przypadku reset i danych maszynowych / danych opcji

Po zresetowaniu działa określony przez MD20150: \$MC_GCODE_RESET_VALUES[44] G-Code (45. grupa G-Code). Aby pozostała kompatybilność z istniejącymi urządzeniami, jest tutaj jako wartość standardowa domyślnie ustawiane SPATH.

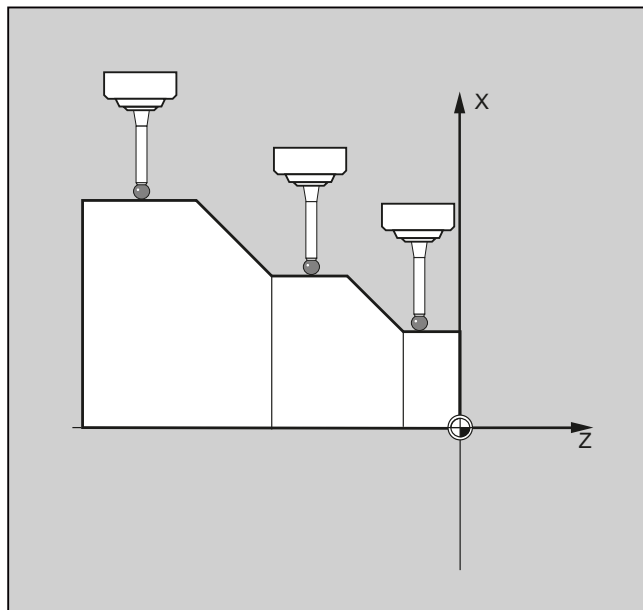
Wartość ustawienia podstawowego dla rodzaju ścięcia jest ustalana przy pomocy MD20150 \$MC_GCODE_RESET_VALUES[9] (10. grupa G-Code).

Osiowa dana maszynowa MD33100 \$MA_COMPRESS_POS_TOL[<n>] ma rozszerzone znaczenie: zawiera tolerancje dla funkcji kompresora i dla ścinania naroży przy pomocy G642.

4.7 Pomiar przy pomocy sondy przełączającej, MEAS, MEAW

Funkcja

Przy pomocy funkcji "pomiar sondą przełączającą" są wykonywane ruchy do pozycji rzeczywistych na obrabianym przedmiocie i przy zboczu przełączenia sondy są mierzone pozycje wszystkich osi zaprogramowanych w bloku pomiarowym i dla każdej osi zapisywane w odpowiedniej komórce pamięci.



Dla programowania funkcji są do dyspozycji obydwie następujące adresy stałe:

- MEAS

Przy pomocy `MEAS` jest kasowana pozostała droga między pozycją rzeczywistą i zadaną.

- MEAW

`MEAW` jest stosowane do zadań pomiarowych, przy których w każdym przypadku powinno nastąpić dosunięcie do zaprogramowanej pozycji.

`MEAS` i `MEAW` działają pojedynczymi blokami i są programowane razem z instrukcjami ruchu. Posuw i rodzaj interpolacji (G0, G1, ...), tak samo jak liczba osi, muszą przy tym zostać dopasowane do zadania pomiarowego.

Składnia

```
MEAS=<TE> G... X... Y... Z...
MEAW=<TE> G... X... Y... Z...
```

Znaczenie

MEAS:	Polecenie: Pomiar z kasowaniem pozostałej drogi Działanie: Pojedynczymi blokami
MEAW:	Polecenie: Pomiar bez kasowania pozostałej drogi Działanie: pojedynczymi blokami
<TE>:	Zdarzenie przerzutnikowe do wyzwolenia pomiaru Typ: INT Zakres wartości: -2, -1, 1, 2 Znaczenie: (+)1 Zbocze rosnące od sondy pomiarowej 1 (na wejście pomiarowe 1) -1 Zbocze opadające od sondy pomiarowej 1 (na wejście pomiarowe 1) (+)2 Zbocze rosnące od sondy pomiarowej 2 (na wejście pomiarowe 2) -2 Zbocze opadające od sondy pomiarowej 2 (na wejście pomiarowe 2) Wskazówka: Istnieją maksymalnie 2 sondy pomiarowe (zależnie od stopnia rozbudowy).
G...:	Rodzaj interpolacji, np. G0, G1, G2 lub G3
X... Y... Z...:	Punkty końcowe we współrzędnych kartezjańskich

Przykład

Kod programu	Komentarz
N10 MEAS=1 G1 F1000 X100 Y730 Z40	; Blok pomiarowy z sonda pomiarową pierwszego wejścia pomiarowego i interpolacją prostoliniową. Zatrzymanie przebiegu wyprzedzającego jest wytwarzane automatycznie.
...	

Dalsze informacje

Status zlecenia pomiarowego

Jeżeli w programie jest wymagana ewaluacja, czy sonda pomiarowa przełączyła czy nie, można odczytać zmienną stanu \$AC_MEA[<n>] (<n> = numer sondy pomiarowej):

Wartość	Znaczenie
0	Zlecenie pomiarowe nie wykonane
1	Zlecenie pomiarowe zakończone pomyślnie (sonda pomiarowa przełączyła)

Wskazówka

Jeżeli sonda pomiarowa jest wychylana w programie, zmienna jest ustawiana na 1. Przy starcie bloku pomiarowego zmienna jest automatycznie ustawiana na stan początkowy.

Odczyt zmierzonej wartości

Są rejestrowane pozycje wszystkich osi uczestniczących w tworzeniu konturu i osi pozycjonowania (maksymalna liczba osi zależy od konfiguracji sterowania). W przypadku MEAS ruch jest po przełączeniu sondy w sposób zdefiniowany hamowany.

Wskazówka

Gdy w bloku pomiarowym jest zaprogramowana oś geometryczna, zostaną zapisane wartości pomiarowe dla wszystkich aktualnych osi geometrycznych.

Jeżeli w bloku pomiaru jest zaprogramowana oś uczestnicząca w transformacji, zostaną zapisane wartości pomiarowe wszystkich osi uczestniczących w tej transformacji

Odczyt wyników pomiarów

Wyniki pomiaru dla osi zmierzonych przy pomocy sondy pomiarowej można przeczytać poprzez następujące zmienne systemowe:

- \$AA_MM[<oś>]

Wyniki pomiaru w układzie współrzędnych maszyny

- \$AA_MW[<oś>]

Wyniki pomiaru w układzie współrzędnych obrabianego przedmiotu

Przy odczycie tych zmiennych nie jest wewnętrznie wytwarzane zatrzymanie przebiegu wyprzedzającego.

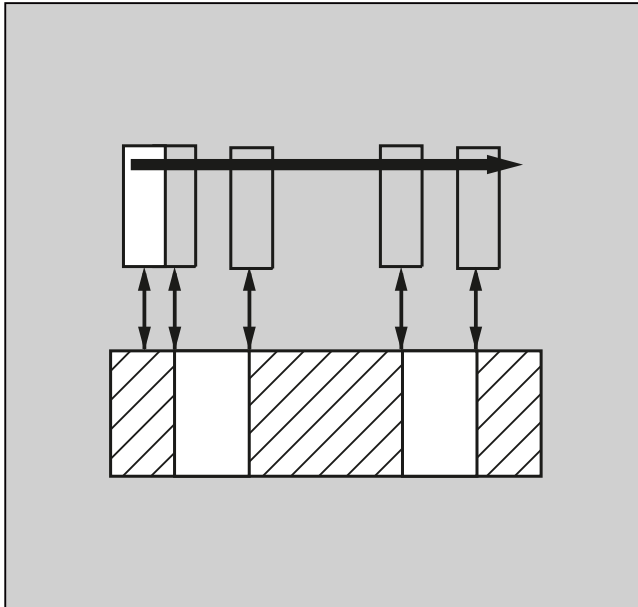
Wskazówka

Przy pomocy `STOPRE` musi w programie NC w odpowiednim miejscu zostać zaprogramowane zatrzymanie przebiegu wyprzedzającego. W przeciwnym przypadku będą czytane nieprawidłowe wartości.

4.8 Pomiar osiowy (MEASA, MEAWA, MEAC) (Opcja)

Funkcja

Przy pomiarze osiowym można używać wielu sond pomiarowych i wielu systemów pomiarowych.



Dla programowania funkcji są do dyspozycji słowa kluczowe MEASA, MEAWA i MEAC.

Przy pomocy MEASA lub MEAWA jest dla każdorazowo zaprogramowanej osi odczytywanych do czterech wartości na pomiar i zapisywanych w zmiennych systemowych odpowiednio do zdarzenia przerzutnikowego.

Ciągle zlecenia pomiaru mogą być przeprowadzane przy pomocy MEAC. W tym przypadku wyniki pomiaru są zapisywane w zmiennych FIFO.

Składnia

MEASA [<os>] = (<tryb>, <TE1>, ..., <TE4>)

MEAWA [<os>] = (<tryb>, <TE1>, ..., <TE4>)

MEAC [<os>] = (<tryb>, <pamięć pomiarowa>, <TE1>, ..., <TE4>)

Wskazówka

MEASA i MEAWA działają pojedynczymi blokami i można je programować w jednym bloku. Jeżeli natomiast MEASA/MEAWA zostanie zaprogramowane w jednym bloku z MEAS/MEAW, nastąpi komunikat błędu.

Znaczenie

MEASA:	Słowo kluczowe: Pomiar w osi z kasowaniem pozostałej drogi Działanie: pojedynczymi blokami
MEAWA:	Słowo kluczowe: Pomiar w osi bez skasowania pozostałej drogi Działanie: pojedynczymi blokami
MEAC:	Słowo kluczowe: Ciągły pomiar w osi bez skasowania pozostałej drogi Działanie: pojedynczymi blokami
<os>:	Nazwa osi kanałowej użytej do pomiaru
<tryb>:	Liczba dwucyfrowa do podania trybu pracy (tryb i system pomiaru) Dekada jednostek (tryb pomiaru): 0 Anulowanie zlecenia pomiarowego. 1 Maks. 4 różne równocześnie uaktywniane zdarzenia przerzutnikowe. 2 Maks. 4 kolejno uaktywniane zdarzenia przerzutnikowe. 3 Maks. 4 kolejno uaktywnialne zdarzenia przerzutnikowe, ale nie ma nadzoru zdarzenia przerzutnikowego 1 przy starcie (alarmy 21700/21703 są ukrywane). Wskazówka: Ten tryb jest przy MEAC niemożliwy. Dekada dziesiątek (system pomiarowy): 0 (albo brak Aktywny system pomiarowy informacji) 1 System pomiarowy 1 2 System pomiarowy 2 3 Obydwa systemy pomiarowe
<TE>:	Zdarzenie przerzutnikowe do wyzwolenia pomiaru Typ: INT Zakres wartości: -2, -1, 1, 2 Znaczenie: (+)1 Zbocze rosnące od sondy pomiarowej 1 -1 Zbocze opadające od sondy pomiarowej 1 (+)2 Zbocze rosnące od sondy pomiarowej 2 -2 Zbocze opadające od sondy pomiarowej 2
<pamięć pomiarowa>:	Numer pamięci FIFO

Przykłady

Przykład 1: Pomiar osiowy z kasowaniem pozostałej drogi 1 (ewaluacja w kolejności czasowej)**a) z 1 systemem pomiarowym**

Kod programu	Komentarz
...	
N100 MEASA[X]=(1,1,-1) G01 X100 F100	; Pomiar w trybie 1 z aktywnym systemem pomiarowym. Czekanie na sygnał pomiarowy o zboczu rosnącym/opadającym od sondy pomiarowej 1 na drodze ruchu do X=100.
N110 STOPRE	; Zatrzymanie przebiegu wyprzedzającego
N120 IF \$AC_MEA[1]==FALSE GOTOF ENDE	; Skontrolowanie pomyślnego zakończenia pomiaru.
N130 R10=\$AA_MM1[X]	; Zapisanie wartości pomiarowej należącej do pierwszego zaprogramowanego zdarzenia przerzutnikowego (zbocze rosnące).
N140 R11=\$AA_MM2[X]	; Zapisanie wartości pomiarowej należącej do drugiego zaprogramowanego zdarzenia przerzutnikowego (zbocze opadające).
N150 KONIEC:	

b) z 2 systemami pomiarowymi

Kod programu	Komentarz
...	
N200 MEASA[X]=(31,1,-1) G01 X100 F100	; Pomiar w trybie 1 z obydwojma systemami pomiarowymi. Czekanie na sygnał pomiarowy o zboczu rosnącym/opadającym od sondy pomiarowej 1 na drodze ruchu do X=100.
N210 STOPRE	; Zatrzymanie przebiegu wyprzedzającego
N220 IF \$AC_MEA[1]==FALSE GOTOF ENDE	; Skontrolowanie pomyślnego zakończenia pomiaru.
N230 R10=\$AA_MM1[X]	; Zapisanie wartości pomiarowej systemu pomiarowego 1 przy zboczu rosnącym.
N240 R11=\$AA_MM2[X]	; Zapisanie wartości pomiarowej systemu pomiarowego 2 przy zboczu rosnącym.
N250 R12=\$AA_MM3[X]	; Zapisanie wartości pomiarowej systemu pomiarowego 1 przy zboczu opadającym.
N260 R13=\$AA_MM4[X]	; Zapisanie wartości pomiarowej systemu pomiarowego 2 przy zboczu opadającym.
N270 KONIEC:	

Przykład 2: Pomiar osiowy z kasowaniem pozostałej drogi w trybie 2 (ewaluacja w kolejności zaprogramowanej)

Kod programu	Komentarz
...	
N100 MEASA[X]=(2,1,-1,2,-2) G01 X100 F100	; Pomiar w trybie 2 z aktywnym systemem pomiarowym. Czekanie na sygnał pomiarowy w kolejności zbocze rosnące od sondy pomiarowej 1, zbocze opadające sonda 1, zbocze rosnące od sondy 2, zbocze opadające sonda 2 na drodze ruchu do X = 100.
N110 STOPRE	; Zatrzymanie przebiegu wyprzedzającego
N120 IF \$AC_MEA[1]==FALSE GOTOF MESSTASTER2	; Skontrolowanie pomyślnego pomiaru sondą pomiarową 1.
N130 R10=\$AA_MM1[X]	; Zapisanie wartości pomiarowej należącej do pierwszego zaprogramowanego zdarzenia przerzutnikowego (zbocze rosnące sonda pomiarowa 1).
N140 R11=\$AA_MM2[X]	; Zapisanie wartości pomiarowej należącej do drugiego zaprogramowanego zdarzenia przerzutnikowego (zbocze rosnące sonda pomiarowa 1).
N150 SONDA_POMIAROWA2:	
N160 IF \$AC_MEA[2]==FALSE GOTOF KONIEC	; Skontrolowanie pomyślnego pomiaru sondą pomiarową 2.
N170 R12=\$AA_MM3[X]	; Zapisanie wartości pomiarowej należącej do trzeciego zaprogramowanego zdarzenia przerzutnikowego (zbocze rosnące sonda pomiarowa 2).
N180 R13=\$AA_MM4[X]	; Zapisanie wartości pomiarowej należącej do czwartego zaprogramowanego zdarzenia przerzutnikowego (zbocze rosnące sonda pomiarowa 2).
N190 KONIEC:	

Przykład 3: Pomiar osiowy ciągły w trybie 1 (ewaluacja w kolejności czasowej)

a) Pomiar do 100 wartości

Kod programu	Komentarz
...	
N110 DEF REAL WARTOŚĆ_POMIAROWA[100]	
N120 DEF INT petla=0	
N130 MEAC[X]=(1,1,-1) G01 X1000 F100	; Pomiar w trybie 1 z aktywnym systemem pomiarowym, zapisanie wartości pomiarowych pod \$AC_FIFO1, czekanie na sygnał pomiarowy o zboczu opadającym od sondy pomiarowej 1 na drodze ruchu do X=1000.
N135 STOPRE	
N140 MEAC[X]=(0)	; Przerwanie pomiaru po osiągnięciu pozycji osi.
N150 R1=\$AC_FIFO1[4]	; Zapisanie liczby zebranych wartości pomiarowych w parametrze R1.
N160 FOR petla=0 TO R1-1	
N170 WARTOSC_POMIAROWA[petla] = \$AC_FIFO1[0]	; Odczytanie i zapisanie wartości pomiarowych z \$AC_FIFO1.
N180 ENDFOR	

b) Pomiar z kasowaniem pozostałej drogi po 10 wartościach pomiarowych

Kod programu	Komentarz
...	
N10 WHEN \$AC_FIFO1[4] >=10 DO MEAC[x]=(0) DELDTG(x)	; Skasowanie pozostałej drogi
N20 MEAC[x]=(1,1,1,-1) G01 X100 F500	
N30 MEAC[X]=(0)	
N40 R1=\$AC_FIFO1[4]	; Liczba wartości pomiarowych.
...	

c) Pomiar "opadającego/rosnącego" boku zęba przy użyciu 2 sond pomiarowych

Kod programu	Komentarz
...	
N110 DEF REAL WARTOŚĆ_POMIAROWA[16]	
N120 DEF INT Petla=0	
N130 MEAC[X]=(1,1,-1,2) G01 X100 F100	; Pomiar w trybie 1 z aktywnym systemem pomiarowym, zapisanie wartości pomiarowych pod \$AC_FIFO1, oczekiwanie na sygnał pomiarowy w kolejności: zbocze opadające od sondy pomiarowej 1, zbocze rosnące sondy pomiarowej 2, na drodze ruchu do X=100.
N140 STOPRE	; Zatrzymanie przebiegu wyprzedzającego
N150 MEAC[X]=(0)	; Przerwanie pomiaru po osiągnięciu pozycji osi.
N160 R1=\$AC_FIFO1[4]	; Zapisanie liczby zebranych wartości pomiarowych w parametrze R1.
N170 FOF Petla=0 TO R1-1	
N180 WARTOSC_POMIAROWA[petla] = \$AC_FIFO1[0]	; Odczytanie i zapisanie wartości pomiarowych z \$AC_FIFO1.
N190 ENDFOR	

Dalsze informacje

Zlecenie pomiarowe

Zaprogramowanie zlecenia pomiarowego może nastąpić w programie obróbki albo z akcji synchronicznej (patrz rozdział "Akcje synchroniczne (Strona 561)"). Na oś może przy tym w jednej i tej samej chwili być aktywne tylko jedno zlecenie pomiarowe.

Wskazówka

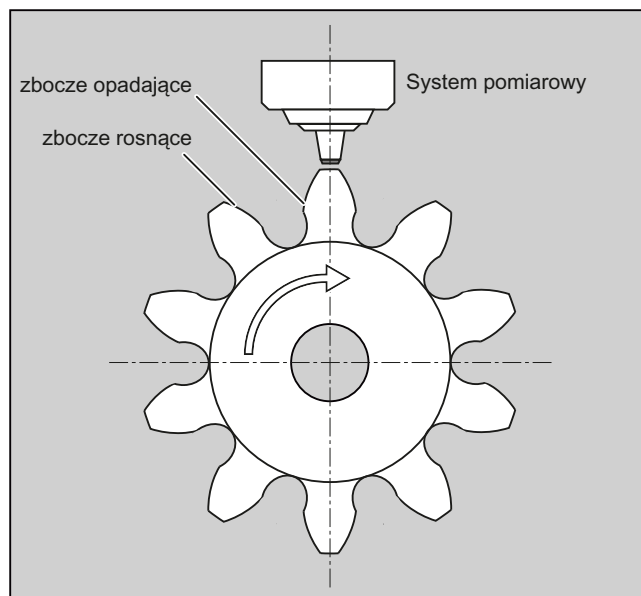
Posuw należy dopasować do każdorazowego zadania pomiarowego.

W przypadku MEASA i MEAWA prawidłowe wyniki mogą zostać zagwarantowane tylko w przypadku posuwów, przy których następuje nie więcej, niż jedno takie samo i nie więcej, niż 4 różne zdarzenia przerzutnikowe na takt regulacji położenia.

Przy pomiarze ciągłym przy pomocy MEAC stosunek między taktem interpolacji i taktem regulacji położenia nie może być większy, niż 1:8.

Zdarzenie przerzutnikowe

Zdarzenie przerzutnikowe składa się z numeru sondy pomiarowej i kryterium wyzwalania (zbocze rosnące albo malejące) sygnału pomiarowego.



Dla każdego pomiaru można przetwarzać każdorazowo do 4 zdarzeń przerzutnikowych uaktywnionej sondy pomiarowej, a więc do dwóch sond pomiarowych po dwa zbocza pomiarowe. Kolejność przetwarzania jak też maksymalna liczba zdarzeń przerzutnikowych, są przy tym zależne od wybranego trybu.

Wskazówka

Dla trybu pomiaru 1 obowiązuje: Takie samo zdarzenie przerzutnikowe wolno tylko jeden raz zaprogramować w zleceniu pomiarowym!

4.8 Pomiar osiowy (MEASA, MEAWA, MEAC) (Opcja)

W przypadku MEAC liczba wartości pomiarowych na takt regulatora położenia może zostać zwiększona do 8 od zbrocza rosnącego i 8 od zbrocza opadającego. Dzięki temu można realizować większe posuwy i prędkości obrotowe.

Literatura:

Podręcznik działania, Funkcje rozszerzające; Pomiar (M5), Rozdział: Pomiar osiowy

Tryb pracy

Przy pomocy pierwszej cyfry (dziesiątki) trybu pracy jest wybierany żądany system pomiarowy. Gdy jest tylko jeden system pomiarowy, a został jednak zaprogramowany, drugi, zostaje automatycznie zastosowany system istniejący.

Przy pomocy drugiej cyfry (jednostki) jest wybierany żądany tryb pomiaru. Przez to proces pomiaru jest dopasowywany do możliwości każdorazowego sterowania:

- **Tryb 1**

Ewaluacja zdarzeń przerzutnikowych następuje w czasowej kolejności ich wystąpienia. W tym trybie przy zastosowaniu sześcioksiowych zespołów konstrukcyjnych możliwe jest zaprogramowanie tylko jednego zdarzenia przerzutnikowego wzgl. przy podaniu wielu zdarzeń przerzutnikowych następuje automatyczne przejście na tryb 2 (bez komunikatu).

- **Tryb 2**

Ewaluacja zdarzeń przerzutnikowych następuje w kolejności zaprogramowanej.

- **Tryb 3**

Ewaluacja zdarzeń przerzutnikowych następuje w kolejności zaprogramowanej, ale nie ma nadzoru zdarzenia przerzutnikowego 1 przy STARCIE.

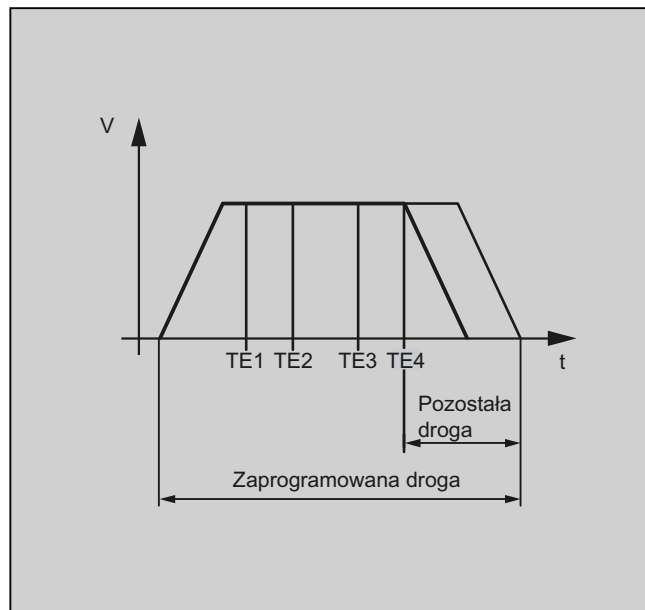
Wskazówka

Przy zastosowaniu 2 systemów pomiarowych dają się zaprogramować tylko dwa zdarzenia przerzutnikowe.

Pomiar z i bez kasowania pozostałej drogi

Przy zaprogramowaniu `MEASA` kasowanie pozostałej drogi jest przeprowadzane dopiero po zarejestrowaniu wszystkich wymaganych wartości pomiarowych.

Dla specjalnych zadań pomiarowych, w przypadku których w każdym przypadku ma zostać dokonane dosunięcie do zaprogramowanej pozycji, stosuje się `MEAWA`.



Wskazówka

`MEASA` nie daje się programować w akcjach synchronicznych. Zastępczo można zaprogramować `MEAWA` plus skasowanie pozostałej drogi jako akcja synchroniczna.

Jeżeli zlecenie pomiaru jest uruchamiane przy pomocy `MEAWA` z akcji synchronicznych, wartości pomiarowe są dostępne tylko w układzie współrzędnych maszyny.

Wyniki pomiaru dla MEASA, MEAWA

Wyniki pomiaru są do dyspozycji pod następującymi zmiennymi systemowymi:

- W układzie współrzędnych maszyny:

`$AA_MM1 [<oś>]`

Wartość pomiarowa zaprogramowanego systemu pomiarowego przy zdarzeniu przerzutnikowym 1

...

...

`$AA_MM4 [<oś>]`

Wartość pomiarowa zaprogramowanego systemu pomiarowego przy zdarzeniu przerzutnikowym 4

- W układzie współrzędnych obrabianego przedmiotu:

\$AA_WM1 [<oś>]	Wartość pomiarowa zaprogramowanego systemu pomiarowego przy zdarzeniu przerzutnikowym 1
...	...
\$AA_WM4 [<oś>]	Wartość pomiarowa zaprogramowanego systemu pomiarowego przy zdarzeniu przerzutnikowym 4

Wskazówka

Przy odczycie tych zmiennych nie jest wewnętrznie wytwarzane zatrzymanie przebiegu wyprzedzającego. Przy pomocy `STOPRE` musi w odpowiednim miejscu zostać zaprogramowane zatrzymanie przebiegu wyprzedzającego. W przeciwnym przypadku będą wczytywane nieprawidłowe wartości.

Osie geometryczne / transformacje

Jeżeli ma zostać uruchomiony pomiar osiowy dla osi geometrycznej, to samo zlecenie pomiaru musi zostać explicite zaprogramowane dla wszystkich pozostałych osi geometrycznych. To samo dotyczy osi, które uczestniczą w transformacji.

Przykłady:

```
N10 MEASA [Z] = (1, 1) MEASA [Y] = (1, 1) MEASA [X] = (1, 1) G0 Z100
```

lub

```
N10 MEASA [Z] = (1, 1) POS [Z] = 100
```

Zlecenie pomiarowe z 2 systemami pomiarowymi

Jeżeli jest przeprowadzane zlecenie pomiaru z dwoma systemami pomiarowymi, każde z obydwu możliwych zdarzeń przerzutnikowych jest rejestrowane przez obydwa systemy każdej osi. Zajętość zarezerwowanych zmiennych jest przez to zadana:

\$AA_MM1 [<oś>]	wzgl.	\$AA_MW1 [<oś>]	Wartość pomiarowa od systemu pomiarowego 1 przy zdarzeniu przerzutnikowym 1
\$AA_MM2 [<oś>]	wzgl.	\$AA_MW2 [<oś>]	Wartość pomiarowa od systemu pomiarowego 2 przy zdarzeniu przerzutnikowym 1
\$AA_MM3 [<oś>]	wzgl.	\$AA_MW3 [<oś>]	Wartość pomiarowa od systemu pomiarowego 1 przy zdarzeniu przerzutnikowym 2
\$AA_MM4 [<oś>]	wzgl.	\$AA_MW4 [<oś>]	Wartość pomiarowa od systemu pomiarowego 2 przy zdarzeniu przerzutnikowym 2

Zmienne systemowe

Status sondy pomiarowej jest do dyspozycji pod następującymi zmiennymi systemowymi:

\$A_PROBE[<n>]

Wartość	Znaczenie
1	Sonda pomiarowa wychylona
0	Sonda pomiarowa nie wychylona

Ograniczenie sondy pomiarowej jest do określenia pod następującymi zmiennymi systemowymi:

\$A_PROBE_LIMITED[<n>]

Wartość	Znaczenie
1	Ograniczenie sondy pomiarowej aktywne
0	Ograniczenie sondy pomiarowej nieaktywne

<n> = sonda pomiarowa

Literatura:

Podręcznik Lista zmiennych systemowych

Status zlecenia pomiarowego w przypadku MEASA, MEAWA

Jeżeli w programie jest wymagana ewaluacja, wówczas można dokonać odczytania statusu zlecenia pomiarowego przez \$AC_MEA[<n>], gdzie n = numer sondy pomiarowej. Gdy wszystkie zaprogramowane w bloku zdarzenia przełączające sond pomiarowych <n> nastąpiły, zmienna ta daje wartość 1. W przeciwnym przypadku wartość wynosi 0.

Wskazówka

Jeżeli pomiar jest uruchamiany z akcji synchronicznych, \$AC_MEA nie jest już aktualizowane. W tym przypadku należy odczytać nowy sygnał interfejsu DB31, ... DBX62.3 wzgl. równoważną zmienną \$AA_MEA[<oś>].

Znaczenie:

\$AA_MEA==1: pomiar aktywny

\$AA_MEA==0: pomiar nie aktywny

Pomiar ciągły (MEAC)

Wartości pomiarowe są w przypadku MEAC w układzie współrzędnych maszyny i są zapisywane w podanej pamięci FIFO[n]. Jeżeli dla pomiarów są zaprojektowane dwie sondy pomiarowe, wartości pomiarowe drugiej sondy są zapisywane oddzielnie w zaprojektowanej w tym celu (ustawianej przez MD) pamięci FIFO[n+1].

Pamięć FIFO jest pamięcią obiegową, do której wartości pomiarowe są wpisywane na zasadzie obiegu w zmiennych \$AC_FIFO, patrz rozdział "Akcje synchroniczne (Strona 561)".

Wskazówka

Treść pamięci FIFO może być odczytana tylko jeden raz. W celu wielokrotnego użycia danych pomiarowych muszą one być poddane zapisaniu pośredniemu w danych użytkownika.

Gdy liczba wartości pomiarowych przekracza liczbę maksymalną ustaloną w danej maszynie dla pamięci FIFO, wówczas pomiar ulega automatycznemu zakończeniu.

Pomiar bez końca daje się zrealizować przez cykliczny odczyt wartości pomiarowych. Odczyt musi przy tym następować co najmniej z taką samą częstotliwością co wpływ nowych wartości pomiarowych.

Literatura:

- Podręcznik działania, Akcje synchroniczne; Opis szczegółowy, Rozdział: Parametry (\$AC_FIFO)
- Podręcznik działania, Funkcje rozszerzające; Pomiar (M5), Rozdział: Pomiar osiowy

Ochrona przed błędnym programowaniem

Następujące błędne zaprogramowania są rozpoznawane i wyświetlane jako błąd:

- MEASA/MEAWA zaprogramowano w jednym bloku razem z MEAS/MEAW

Przykład:

```
N01 MEAS=1 MEASA[X]=(1,1) G01 F100 POS[X]=100
```

- MEASA/MEAWA z liczbą parametrów <2 albo >5

Przykład:

```
N01 MEAWA[X]=(1) G01 F100 POS[X]=100
```

- MEASA/MEAWA ze zdarzeniem przerzutnikowym nierównym 1/ -1/ 2/ -2

Przykład:

```
N01 MEASA[B]=(1,1,3) B100
```

- MEASA/MEAWA z nieprawidłowym trybem

Przykład:

```
N01 MEAWA[B]=(4,1) B100
```

- MEASA/MEAWA z podwójnie zaprogramowanym zdarzeniem przerzutnikowym

Przykład:

```
N01 MEASA[B]=(1,1,-1,2,-1) B100
```

- MEASA/MEAWA i brakująca oś geometryczna

Przykład:

```
N01 MEASA[X]=(1,1) MEASA[Y]=(1,1) G01 X50 Y50 Z50 F100 ;oś GEO X/Y/Z
```

- Niejednolite zadanie pomiarowe w przypadku osi geometrycznych

Przykład:

```
N01 MEASA[X]=(1,1) MEASA[Y]=(1,1) MEASA[Z]=(1,1,2) G01 X50 Y50 Z50 F100
```

4.9 Funkcje specjalne dla użytkownika OEM (OMA1 ... OMA5, OEMIPO1, OEMIPO2, G810 ... G829)

Adresy OEM

Znaczenie adresów OEM określa użytkownik OEM. Działanie jest określane przez cykle kompilacji. 5 adresów OEM jest zarezerwowanych (OMA1 ... OMA5). Identyfikatory adresów można ustawiać. Adresy OEM są dopuszczalne w każdym bloku.

Zarezerwowane wywołania funkcji G

Dla użytkownika OEM są zarezerwowane następujące wywołania funkcji G:

- OEMIPO1, OEMIPO2 (z grupy funkcji G 1)
- G810 ... G819 (grupa funkcji G 31)
- G820 ... G829 (grupa funkcji G 32)

Działanie jest określane przez cykle kompilacji.

Funkcje i podprogramy

Dodatkowo użytkownicy OEM mogą również tworzyć predefiniowane funkcje i podprogramy z przekazaniem parametrów.

Wskazówka

Symulacja obrabianego przedmiotu

Do w. opr. 4.4 nie są w symulacji obrabianego przedmiotu obsługiwane żadne cykle, a od w. opr. 4.4 tylko niektóre cykle kompilacyjne (CC).

Polecenia językowe w programie obróbki nie obsługiwanych cykli kompilacyjnych (OMA1 ... OMA5, OEMIPO1/2, G810 ... G829, własne procedury i funkcje) prowadzą do braku indywidualnego ich traktowania oraz do komunikatu alarmowego i do anulowania symulacji.

Rozwiązanie: Należy indywidualnie traktować specyficzne dla CC elementy językowe w programie obróbki (odpytanie \$P_SIM).

Przykład:

```
N1 G01 X200 F500
IF (1== $P_SIM)
N5 X300 ;podczas symulacji CC nie aktywne
ELSE
N5 X300 OMA1=10
ENDIF
```

4.10 Zmniejszenie posuwu ze zwłoką na narożach (FENDNORM, G62, G621)

Funkcja

Przy automatycznej zwłoce na narożach posuw jest zmniejszany (kształt dzwona) na krótko przed odnośnym narożem. Poza tym mające znaczenie dla obróbki zachowanie się narzędzia może być parametryzowane przez dane ustawcze. Są to:

- Początek i koniec zmniejszenia posuwu
- Korektor, z którym posuw jest zmniejszany
- Rozpoznanie znaczącego naroża

Jako znaczące naroża są uwzględniane te naroża, których kąt wewnętrzny jest mniejszy niż naroże sparametryzowane przez daną ustawczą.

Przy pomocy wartości domyślnej `FENDNORM` funkcja automatycznego korektora na narożach jest wyłączana.

Literatura:

/FBFA/ Opis działania Dialekty ISO

Składnia

`FENDNORM`

`G62 G41`

`G621`

Znaczenie

`FENDNORM` Automatyczna zwłoka na narożach wył.

`G62` Zwłoka w narożach wewnętrznych przy aktywnej korekcji promienia narzędzia

`G621` Zwłoka na wszystkich narożach przy aktywnej korekcji promienia narzędzia

G62 działa tylko na narożach wewnętrznych z

- aktywną korekcją promienia narzędzia `G41`, `G42` i
- aktywnym przechodzeniu płynnym `G64`, `G641`

Ruch do odpowiedniego naroża następuje ze zmniejszonym posuwem, który wynika z:

$F * (\text{korektor do redukcji posuwu}) * \text{korektor posuwu}$

Maksymalnie możliwe obniżenie posuwu jest uzyskiwane dokładnie wtedy, gdy narzędzie, w odniesieniu do toru punktu środkowego, ma dokonać zmiany kierunku na odnośnym narożu.

G621 działa analogicznie do G62 na każdym narożu, osi ustalonych przez `FGROUP`.

4.11 Programowane kryterium końca ruchu (FINEA, COARSEA, IPOENDA, IPOBRKA, ADISPOSA)

Funkcja

Podobnie do kryterium zmiany bloku przy interpolacji ruchu po torze (G601, G602 i G603), kryterium końca ruchu przy interpolacji pojedynczej osi może zostać zaprogramowane w programie obróbki wzgl. w akcjach synchronicznych dla osi rozkazowych/PLC.

Zależnie od tego, które kryterium końca ruchu jest ustawione, bloki programu obróbki wzgl. bloki cyklu technologicznego ulegają w różny sposób szybkiemu zakończeniu. To samo obowiązuje dla PLC przez FC15/16/18.

Składnia

```

FINEA [<oś>]
COARSEA [<oś>]
IPOENDA [<oś>]
IPOBRKA (<oś> [, <chwila>])
ADISPOSA (<oś> [, <tryb>, <wielkość okna>])

```

Znaczenie

FINEA:	Kryterium końca ruchu: "Zatrzymanie dokładne dokładnie" Działanie: Modalnie
COARSEA:	Kryterium końca ruchu: "Zatrzymanie dokładne zgrubnie" Działanie: Modalnie
IPOENDA:	Kryterium końca ruchu: "Interpolator-Stop" Działanie: Modalnie
IPOBRKA:	Kryterium zmiany bloku: Charakterystyka hamowania Działanie: Modalnie
ADISPOSA:	Okno tolerancji dla kryterium końca ruchu Działanie: Modalnie
<oś>:	Nazwa osi kanału (X, Y,)
<chwila>:	Chwila zmiany bloku, odniesiona do charakterystyki hamowania w %: <ul style="list-style-type: none"> • 100% = początek charakterystyki hamowania • 0% = koniec charakterystyki hamowania, równoznaczny z
	IPOENDA
	Typ: REAL

4.11 Programowane kryterium końca ruchu (FINEA, COARSEA, IPOENDA, IPOBRKA, ADISPOSA)

<tryb>:	Odniesienie okna tolerancji
	Zakres wartości: 0 Okno tolerancji nie aktywne
	1 Okno tolerancji odnośnie pozycji zadanej
	2 Okno tolerancji odnośnie pozycji rzeczywistej
	Typ: INT
<wielkość okna>:	Wielkość okna tolerancji
	Typ: REAL

Przykłady

Przykład 1: Kryterium końca ruchu: "Zatrzymanie interpolatora"

Kod programu

```

; Wykonaj ruch w osi pozycjonowania X na 100, prędkość 200 m/min, przyśpieszenie 90%,
; Kryterium końca ruchu: Zatrzymanie interpolatora
N110 G01 POS[X]=100 FA[X]=200 ACC[X]=90 IPOENDA[X]

; Akcja synchroniczna:
; ZAWSZE GDY: jest ustawione wejście 1
; WÓWCZAS Wykonaj ruch w osi pozycjonowania X na 50, prędkość 200 m/min, przyśpieszenie 140%,
; Kryterium końca ruchu: Zatrzymanie interpolatora
N120 EVERY $A_IN[1] DO POS[X]=50 FA[X]=200 ACC[X]=140 IPOENDA[X]
    
```

Przykład 2: Kryterium zmiany bloku: "Charakterystyka hamowania"

Kod programu

Komentarz

N40 POS[X]=100	; Ustawienie domyślne działa ; Pozycjonowanie w X na pozycję 100
N20 IPOBRKA(X,100)	; Kryterium zmiany bloku: zatrzymanie dokładne dokładnie ; 100% = początek charakterystyki hamowania
N30 POS[X]=200	; Zmiana bloku następuje, gdy tylko oś X zaczyna hamować
N40 POS[X]=250	; Oś X nie hamuje na pozycji 200, lecz wykonuje nadal ruch ; dalej na pozycję 250.
N50 POS[X]=0	; Gdy tylko oś rozpocznie hamowanie, następuje zmiana bloku ; Oś X hamuje i wykonuje ruch z powrotem do pozycji 0
N60 X10 F100	; Zmiana bloku następuje na pozycji 0 i "zatrzymaniu dokładnym dokładnie" ; Oś X wykonuje ruch do pozycji 10, jako oś uczestnicząca w tworzeniu konturu

Dalsze informacje

Zmienna systemowa dla kryterium końca ruchu

Działające kryterium końca ruchu można odczytać z zmiennej systemowej \$AA_MOTEND.

Literatura: /LIS2s/ Podręcznik lista, podręcznik 2

Kryterium zmiany bloku: "Charakterystyka hamowania" (IPOBRKA)

Jeżeli przy uaktywnieniu kryterium zmiany bloku "charakterystyka hamowania" jest dla opcji: czas zmiany bloku zaprogramowana wartość, działa ona dla następnego ruchu pozycjonowania i zostanie synchronicznie z przebiegiem głównym zapisana w danej ustawczej. Jeżeli nie podano wartości czasu zmiany bloku, działa aktualna wartość danej ustawczej.

SD43600 \$SA_IPOBRAKE_BLOCK_EXCHANGE

Z następnym zaprogramowaniem osiowego kryterium końca ruchu (FINEA, COARSEA, IPOENDA) następuje wyłączenie aktywności IPOBRKA dla odpowiedniej osi.

Kryterium zmiany bloku: "okno tolerancji" (ADISPOSA)

Przy pomocy ADISPOSA można, jako dodatkowe kryterium zmiany bloku, zdefiniować okno tolerancji wokół punktu końcowego bloku (do wyboru pozycja rzeczywista albo zadana). Dla zmiany bloku muszą być wówczas spełnione obydwa warunki:

- Kryterium zmiany bloku: "Charakterystyka hamowania"
- Kryterium zmiany bloku: "Okno tolerancji"

Literatura

Dalsze informacje dot. kryterium zmiany bloku osi pozycjonowania patrz:

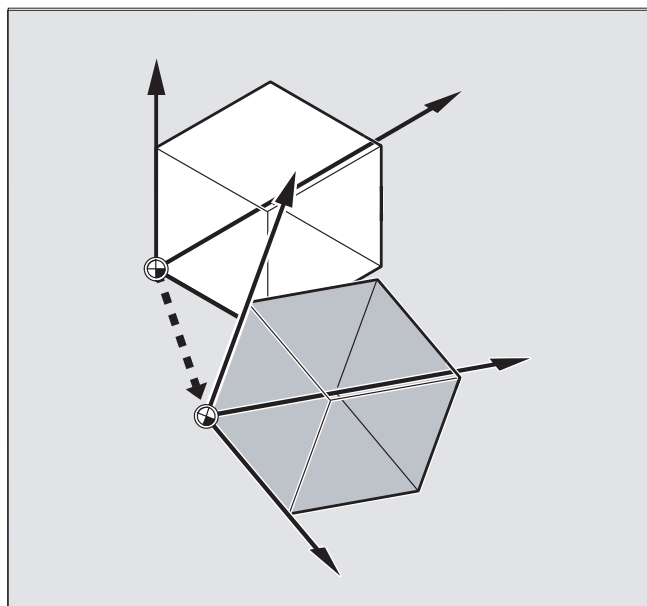
- Podręcznik działania Funkcje rozszerzające; Osie pozycjonowania (P2)
- Podręcznik programowania Podstawy; punkt "Regulacja posuwu"

Transformacje współrzędnych (Frame)

5.1 Transformacja współrzędnych przez zmienną frame

Funkcja

Oprócz możliwości programowania już opisanych w podręczniku programowania "Podstawy", można ustalać układy współrzędnych również przy pomocy predefiniowanych zmiennych frame.



Są zdefiniowane następujące układy współrzędnych:

MKS: Układ współrzędnych maszyny

BKS: Bazowy układ współrzędnych

BNS: Układ współrzędnych bazowego przesunięcia punktu zerowego

ENS: Układ współrzędnych ustawianego przesunięcia punktu zerowego

WKS: Układ współrzędnych obrabianego przedmiotu

Co to jest predefiniowana zmienna frame?

Domyślnie zdefiniowane zmienne frame są to słowa kluczowe, które są już ustalone w języku sterowania razem z ich odpowiednim działaniem i mogą być przetwarzane w programie NC.

Możliwe zmienne frame:

- Frame bazowy (przesunięcie bazowe)
- Frame ustawiany:
- Frame programowany

Odczyt przyporządkowań wartości i wartości rzeczywistych

Zależność zmienna frame/frame

Transformację współrzędnych można uaktywnić przez przyporządkowanie wartości frame do zmiennej frame.

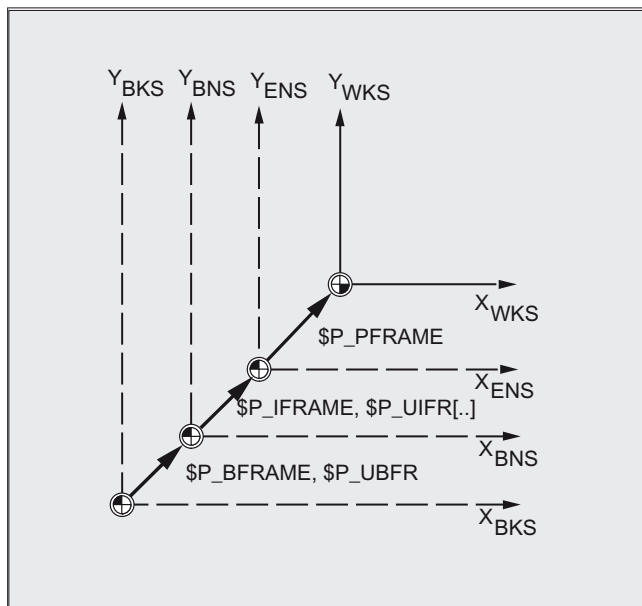
Przykład: `$P_PFRAME=CTTRANS (X, 10)`

Zmienna frame:

`$P_PFRAME` oznacza: aktualny programowany frame.

Frame:

`CTTRANS (X, 10)` oznacza: programowane przesunięcie punktu zerowego osi X o 10 mm.



Odczyt wartości rzeczywistych

Poprzez predefiniowane zmienne w programie obróbki mogą być odczytywane aktualne wartości rzeczywiste układów współrzędnych:

\$AA_IM[oś]: odczyt wartości rzeczywistej w MKS

\$AA_IB[oś]: odczyt wartości rzeczywistej w BKS

\$AA_IBN[oś]: odczyt wartości rzeczywistej w BNS

\$AA_IEN[oś]: odczyt wartości rzeczywistej w ENS

\$AA_IW[oś]: odczyt wartości rzeczywistej w WKS

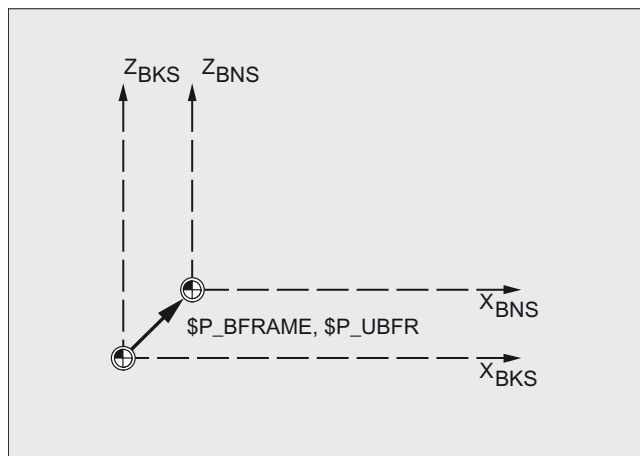
5.1.1 Predefiniowana zmienna frame (\$P_BFRAME, \$P_IFRAME, \$P_PFRAME, \$P_ACTFRAME)

\$P_BFRAME

Aktualna zmienna frame bazowego, która tworzy odniesienie między bazowym układem współrzędnych (BKS) i układem współrzędnych bazowego przesunięcia punktu zerowego (BNS).

Jeżeli frame bazowy opisany przez \$P_UBFR ma natychmiast działać w programie, musi albo

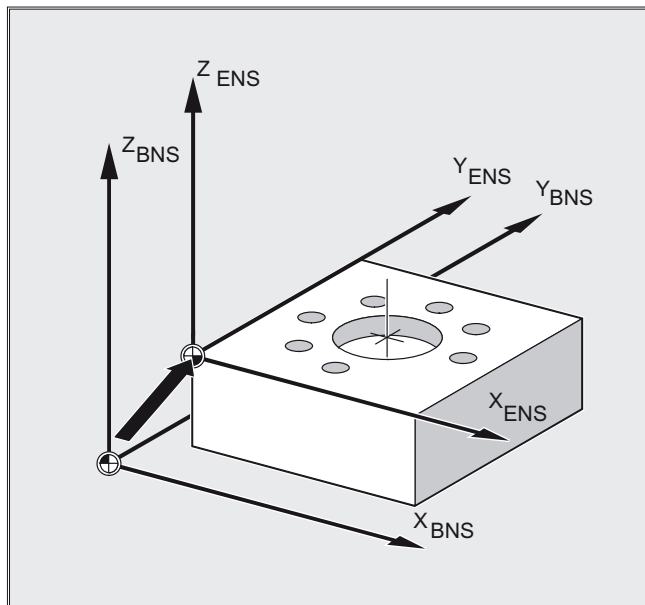
- nastąpić zaprogramowanie G500, G54...G599 albo
- \$P_BFRAME zostać opisana przez \$ \$P_UBFR .



\$P_IFRAME

Aktualna, ustawiana zmienna frame, która tworzy odniesienie między układem współrzędnych bazowego przesunięcia punktu zerowego (BNS) i układem ustawianego przesunięcia punktu zerowego (ENS).

- \$P_IFRAME odpowiada \$P_UIFR[\$P_IFRNUM]
- \$P_IFRAME zawiera po zaprogramowaniu np. G54 zdefiniowane przez G54 przesunięcie, obrót, skalowanie i lustrzane odbicie.

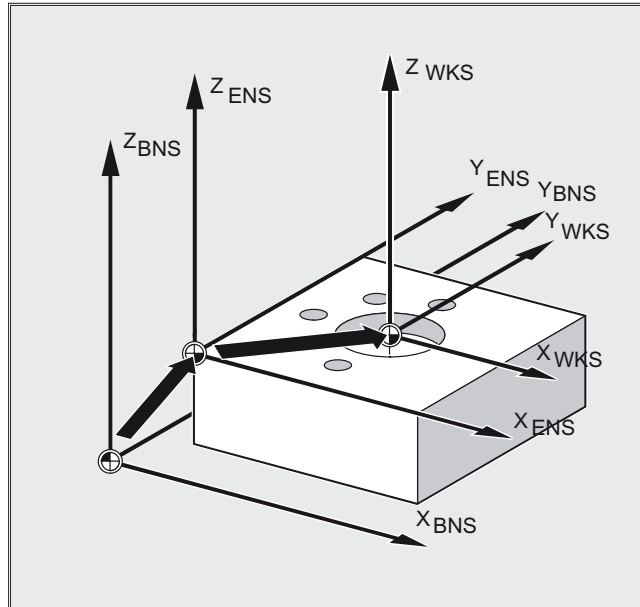


\$P_PFRAME

Aktualna, programowana zmienna frame, która tworzy odniesienie między układem ustawianego przesunięcia punktu zerowego (ENS) i układem współrzędnych obrabianego przedmiotu (WKS).

\$P_PFRAME zawiera wynikający frame, który wynika

- **z zaprogramowania** TRANS/ATRANS, ROT/AROT, SCALE/ASCALE, MIRROR/AMIRROR wzgl.
- **z przyporządkowania** CTRANS, CROT, CMIRROR, CSCALE do FRAME programowanego



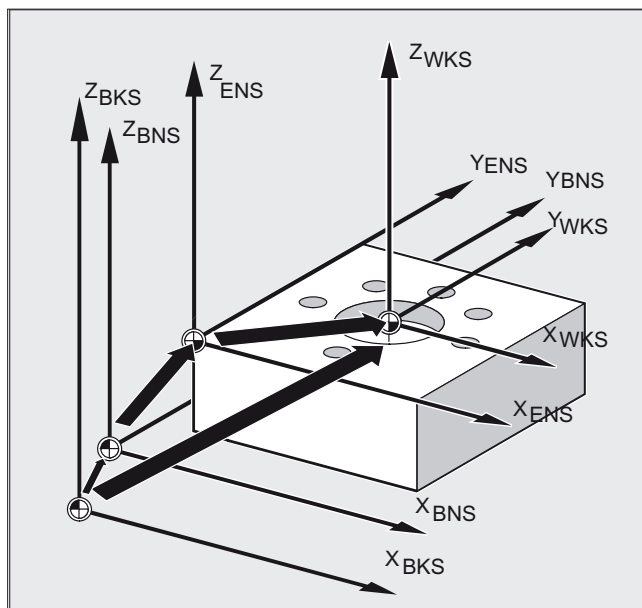
\$P_ACTFRAME

Aktualny, wynikający frame całkowity, który poprzez powiązanie wynika z

- aktualnej zmiennej frame bazowego \$P_BFRAME,
 - aktualnej ustawianej zmiennej frame \$P_IFRAME z frame systemowymi i
 - aktualnej programowanej zmiennej frame \$P_PFRAME z frame systemowymi
- wynika. Frame systemowe, patrz punkt "Frame działające w kanale"

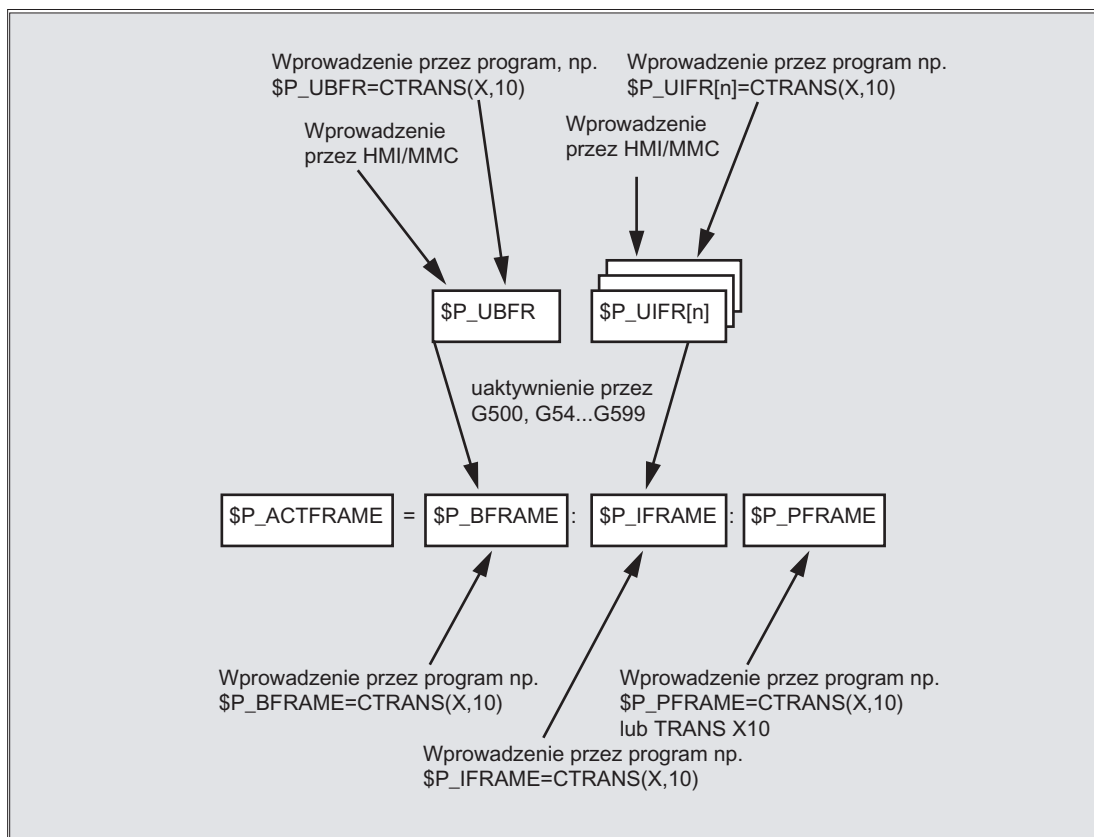
\$P_ACTFRAME opisuje aktualnie obowiązujący punkt zerowy obrabianego przedmiotu.

5.1 Transformacja współrzędnych przez zmienną frame



W przypadku zmiany \$P_BFRAME, \$P_IFRAME lub \$P_PFRAME , następuje ponowne obliczenie \$P_ACTFRAME .

\$P_ACTFRAME odpowiada \$P_BFRAME:\$P_IFRAME:\$P_PFRAME



Frame bazowy i frame ustawiany działają po reset, gdy MD 20110 RESET_MODE_MASK jest ustawiona następująco:

Bit0=1, Bit14=1 --> \$P_UBFR (frame bazowy) działa

Bit0=1, Bit5=1 --> \$P_UIFR[\$P_UIFRNUM] (frame ustawiany) działa

Predefiniowane frame ustawiane \$P_UBFR

Przy pomocy \$P_UBFR jest programowany frame bazowy, nie jest on jednak równocześnie aktywny w programie obróbki. Napisany z \$P_UBFR frame bazowy jest wliczany, gdy

- został wykonany reset i są ustawione bity 0 i 14 danej MD RESET_MODE_MASK,
- zostały wykonane instrukcje G500, G54...G599 .

Predefiniowane frame ustawiane \$P_UIFR[n]

Przez predefiniowaną zmienną frame \$P_UIFR[n] mogą być czytane albo zapisywane z programu obróbki ustawiane przesunięcia punktu zerowego G54 do G599 .

Pod względem budowy te zmienne stanowią jednowymiarową tablicę typu FRAME o nazwie \$P_UIFR[n] .

przyporządkowanie do poleceń G

Standardowo jest 5 ustawianych frame \$P_UIFR[0]...\$P_UIFR[4] wzgl. 5 poleceń G o takim samym znaczeniu – G500 i G54 do G57 , pod których adresami mogą być zapisywane wartości.

Wskazówka

Programowanie zmiennych frame i frame wymaga w programie NC własnego bloku NC.

Wyjątek: Programowanie ustawianego Frame przez G54, G55, ...

\$P_IFRAME=\$P_UIFR[0] odpowiada G500

\$P_IFRAME=\$P_UIFR[1] odpowiada G54

\$P_IFRAME=\$P_UIFR[2] odpowiada G55

\$P_IFRAME=\$P_UIFR[3] odpowiada G56

\$P_IFRAME=\$P_UIFR[4] odpowiada G57

Poprzez daną maszynową można zmienić liczbę frame:

\$P_IFRAME=\$P_UIFR[5] odpowiada G505

... ..

\$P_IFRAME=\$P_UIFR[99] odpowiada G599

Wskazówka

Przez to można utworzyć w sumie 100 układów współrzędnych, które np. jako punkt zerowy dla różnych elementów mogą być wywoływane w różnych programach.

5.2 Przyporządkowanie wartości zmiennym frame / frame

5.2.1 Bezpośrednie przyporządkowanie wartości (wartość osi, kąt, skala)

Funkcja

W programie NC można frame albo zmienne frame bezpośrednio wyposażać w wartości.

Składnia

```
$P_PFRAME=CTRANS (X, wartość osi, Y, wartość osi, Z, wartość osi, ...)
```

```
$P_PFRAME=CROT (X, kąt, Y, kąt, Z, kąt, ...)
```

```
$P_UIFR[..]=CROT (X, kąt, Y, kąt, Z, kąt, ...)
```

```
$P_PFRAME=CSCALE (X, skala, Y, skala, Z, skala, ...)
```

```
$P_PFRAME=CMIRROR (X, Y, Z)
```

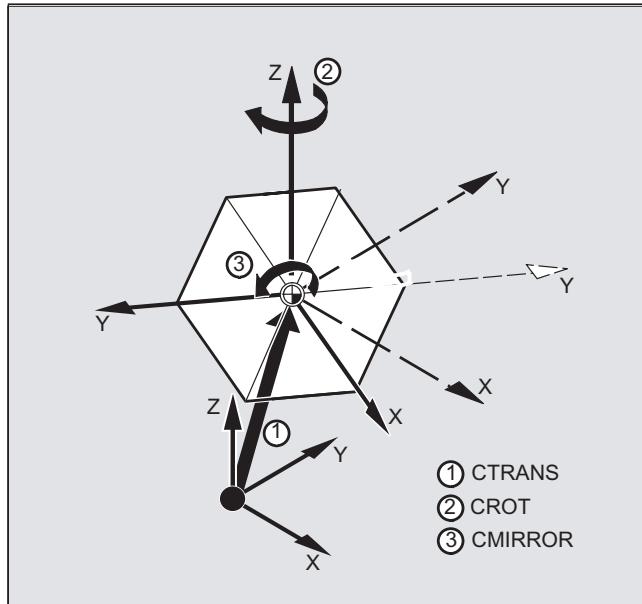
Programowanie \$P_BFRAME następuje analogicznie do \$P_PFRAME.

Znaczenie

CTRANS	Przesunięcie w podanych osiach
CROT	Obrót wokół podanych osi
CSCALE	Zmiana skali w podanych osiach
CMIRROR	Odwroćenie kierunku podanej osi
X Y Z	Wartość przesunięcia w kierunku podanej osi geometrycznej
Wartość osi	Przyporządkowanie wartości osi do przesunięcia
Kąt	Przyporządkowanie kąta obrotu wokół podanych osi
Skala	Zmiana skali

Przykład

Przez przyporządkowanie wartości na aktualnym zaprogramowanym frame są uaktywniane translacja, obrót i lustrzane odbicie.



```
N10 $P_PFRAME=CTRANS(X,10,Y,20,Z,5):CROT(Z,45):CMIRROR(Y)
```

Wstępne wyposażenie komponentów frame-rot w inne wartości

Wstępne wyposażenie w wartości wszystkich trzech komponentów UIFR przy pomocy CROT

Kod programu	Komentarz
\$P_UIFR[5] = CROT(X, 0, Y, 0, Z, 0)	
N100 \$P_UIFR[5, y, rt]=0	
N100 \$P_UIFR[5, x, rt]=0	
N100 \$P_UIFR[5, z, rt]=0	

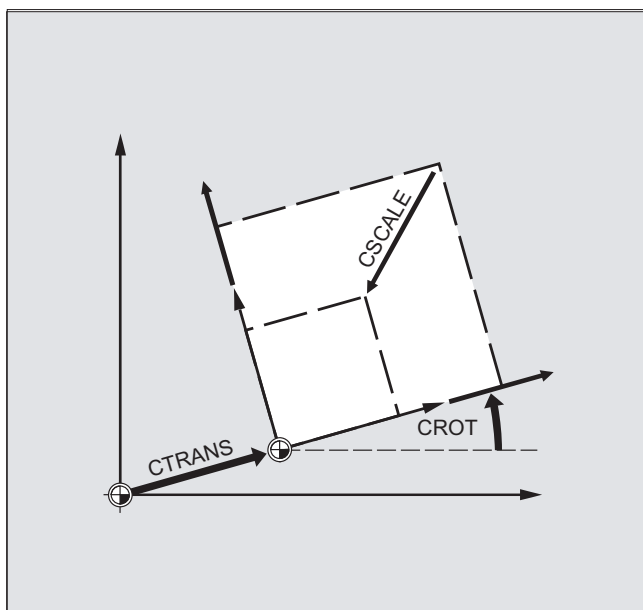
Opis

Można programować wiele kolejnych instrukcji obliczeniowych.

Przykład:

```
$P_PFRAME=CTRANS(...):CROT(...):CSCALE...
```

Należy pamiętać, że polecenia muszą zostać ze sobą powiązane operatorem powiązania w postaci dwukropka (...):(...). Przez to polecenia zostają po pierwsze powiązane ze sobą, a po drugie addytywnie wykonane w zaprogramowanej kolejności.



Wskazówka

Wartości zaprogramowane przy pomocy wymienionych poleceń są przyporządkowywane do frame i zapisywane.

Wartości stają się aktywne dopiero wtedy, gdy zostaną przyporządkowane do frame aktywnej zmiennej frame `$P_BFRAME` wzgl. `$P_PFRAME`.

5.2.2 Odczyt i zmiana komponentów frame (TR, FI, RT, SC, MI)

Funkcja

Istnieje możliwość sięgania do **pojedynczych** danych danego frame, np. do określonej wartości przesunięcia albo kąta obrotu. Te wartości można zmienić albo przyporządkować innej zmiennej.

Składnia

<code>R10=\$P_UIFR[\$P_UIFNUM,X,RT]</code>	Kąt obrotu RT wokół osi X z aktualnie obowiązującego ustawianego przesunięcia punktu zerowego \$P_UIFRNUM ma zostać przyporządkowany do zmiennej R10.
<code>R12=\$P_UIFR[25,Z,TR]</code>	Wartość przesunięcia TR w Z z zestawu danych ustawionego frame nr 25 ma zostać przyporządkowana do zmiennej R12.
<code>R15=\$P_PFRAME[Y,TR]</code>	Wartość przesunięcia TR w Y aktualnego programowanego frame ma zostać przyporządkowana do zmiennej R15.
<code>\$P_PFRAME[X,TR]=25</code>	Wartość przesunięcia TR w X aktualnego programowanego frame ma zostać zmieniona. Od teraz obowiązuje X25.

Znaczenie

\$P_UIFRNUM	Przy pomocy tej zmiennej jest automatycznie tworzone odniesienie do aktualnie obowiązującego ustawianego przesunięcia punktu zerowego.
P_UIFR[n,...,...]	Poprzez podanie numeru frame n sięgamy do ustawianego frame numer n. Podanie składowej, który ma być czytana albo zmieniona:
TR	TR translacja
FI	FI translacja dokładnie
RT	RT obrót
SC	SC Scale zmiana skali
MI	MI lustrzane odbicie
X Y Z	Dodatkowo (patrz przykłady) jest podawana odpowiednia oś X, Y, Z.

Zakres wartości dla obrotu RT

Obrót wokół 1. osi geometrycznej:	-180° do +180°
Obrót wokół 2. osi geometrycznej:	-90° bis +90°
Obrót wokół 3. osi geometrycznej:	-180° do +180°

Opis

Wywołanie frame

Przez podanie zmiennej systemowej \$P_UIFRNUM można uzyskać dostęp bezpośrednio do przesunięcia punktu zerowego aktualnie ustawionego przy pomocy \$P_UIFR wzgl. G54, G55, ...

(\$P_UIFRNUM zawiera numer aktualnie ustawionego frame).

Wszystkie inne zapisane w pamięci ustawiane frame \$P_UIFR wywołujemy przez podanie odpowiedniego numeru \$P_UIFR[n].

Dla predefiniowanej zmiennej frame z frame zdefiniowanych przez siebie podajemy nazwę, np. \$P_IFRAME.

Wywołanie danych

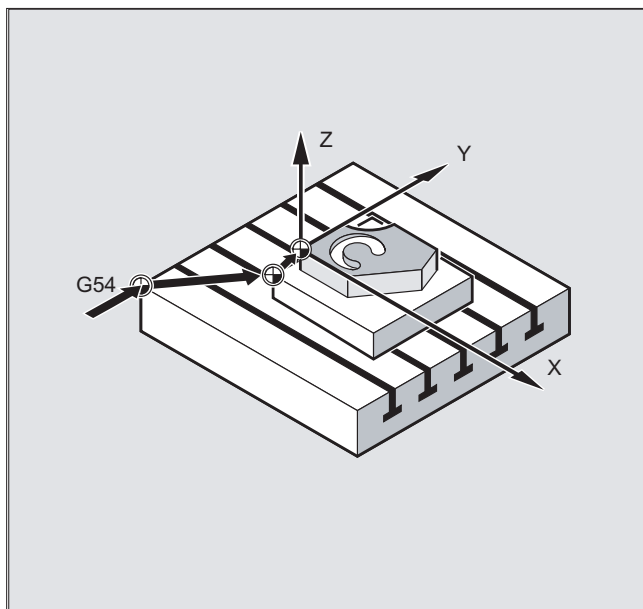
W nawiasach kwadratowych znajduje się nazwa osi i składowa frame wartości, do której chcemy sięgnąć lub którą chcemy zmienić, np. [X, RT] albo [Z, MI].

5.2.3 Powiązanie kompletnych frame

Funkcja

W programie NC kompletny frame może zostać przyporządkowany do innego frame albo frame mogą zostać ze sobą powiązane.

Powiązania frame nadają się np. do opisu wielu obrabianych przedmiotów, które są umieszczone na paletach i mają być obrabiane w jednym przebiegu.



Dla opisu zadań obróbki na paletach składowe frame mogą zawierać tylko określone wartości częściowe, przez których powiązanie są generowane różne punkty zerowe obrabianego przedmiotu.

Składnia

Przyporządkowanie frame

```
DEF FRAME USTAWIENIE1
USTAWIENIE1=CTRANS (X,10)
$P_PFRAME=USTAWIENIE1
DEF FRAME USTAWIENIE4
USTAWIENIE4=$P_PFRAME
$P_PFRAME=USTAWIENIE4
```

Do aktualnego programowanego frame są przyporządkowywane wartości zdefiniowanego przez siebie frame USTAWIENIE1.

Aktualny programowany frame jest poddawany pośredniemu zapisaniu w pamięci i następnie w razie potrzeby zapisywany z powrotem.

Łańcuchy frame

Frame są ze sobą wiązane w zaprogramowanej kolejności, składowe frame jak np. przesunięcia, obroty itd., są wykonywane kolejno addytywnie.

```
$P_IFRAME=$P_UIFR[15]:$P_UIFR[16]
```

\$P_UIFR[15] zawiera np. dane dla przesunięć punktu zerowego. Następnie są – budując na tym – przetwarzane dane z \$P_UIFR[16] np. dane dla obrotu.

```
$P_UIFR[3]=$P_UIFR[4]:$P_UIFR[5]
```

Ustawiany frame 3 jest wytwarzany przez powiązanie ustawianych frame 4 i 5.

Wskazówka

Należy pamiętać, że frame muszą być ze sobą połączone operatorem powiązania w postaci dwukropka :.

5.2.4 Definicja nowych frame (DEF FRAME)

Funkcja

Obok dotychczas opisanych, domyślnie zdefiniowanych, ustawianych frame, jest również możliwość tworzenia nowych frame. Chodzi przy tym o zmienne typu FRAME, które definiujemy przy dowolnym nadaniu nazwy.

Przy pomocy funkcji CTRANS, CROT, CSCALE i CMIRROR można w programie NC nadawać wartości swoim frame.

Składnia

```
DEF FRAME PALETA1
PALETA1=CTRANS (...) :CROT (...) ...
```

Znaczenie

DEF FRAME	Utworzenie nowych frame.
PALETA1	Nazwa nowego frame
=CTRANS (...):	Przyporządkowanie wartości do możliwych funkcji
CROT (...)	

5.3 Przesunięcie zgrubne i dokładne (CFINE, CTRANS)

Funkcja

Przesunięcie dokładne

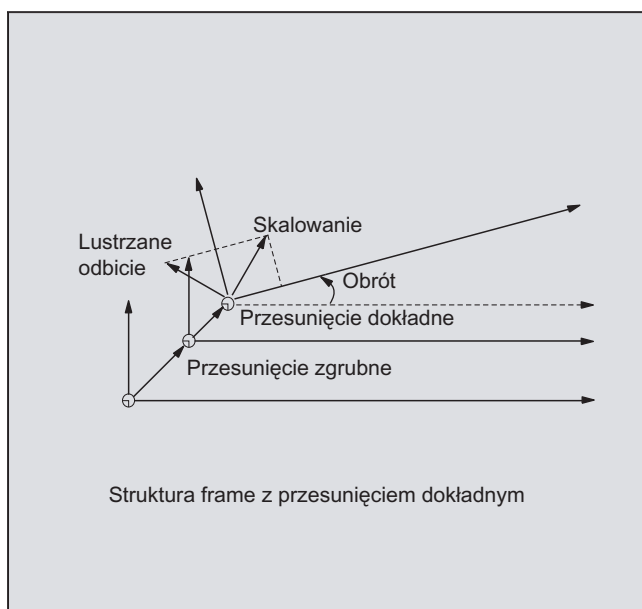
Przesunięcie dokładne Frame jest programowane przy pomocy polecenia `CFINE(...)`.

Wskazówka

Zwolnienie przesunięcia dokładnego przez MD18600 `$MN_MM_FRAME_FINE_TRANS = 1`

Przesunięcie zgrubne

Przesunięcie zgrubne Frame jest programowane przy pomocy polecenia `CTRANS(...)`.



Przesunięcie zgrubne i przesunięcie dokładne dodają się dając przesunięcie łączne.

Składnia

Przesunięcie dokładne

Na przykładzie Frame zarządzania danymi \$P_UIFR:

- Frame łączny
 - \$P_UIFR [<n>] = CFINE (<K1>, <wartość>)
 - \$P_UIFR [<n>] = CFINE (<K1>, <wartość>, <K2>, <wartość>)
 - \$P_UIFR [<n>] = CFINE (<K1>, <wartość>, <K2>, <wartość>, <K3>, <wartość>)
- Komponent Frame
 - \$P_UIFR [<n>, <współrzędna>, FI] = <wartość>

Przesunięcie zgrubne

Na przykładzie Frame zarządzania danymi \$P_UIFR:

- Frame łączny
 - \$P_UIFR [<n>] = CTRANS (<K1>, <wartość>)
 - \$P_UIFR [<n>] = CTRANS (<K1>, <wartość>, <K2>, <wartość>)
 - \$P_UIFR [<n>] = CTRANS (<K1>, <wartość>, <K2>, <wartość>, <K3>, <wartość>)
- Komponent Frame
 - \$P_UIFR [<n>, <współrzędna>, TR] = <wartość>

Frame programowany \$P_PFRAME również:

- TRANS <K1> <wartość>
- TRANS <K1> <wartość> <K2> <wartość>
- TRANS <K1> <wartość> <K2> <wartość> <K3> <wartość>

Znaczenie

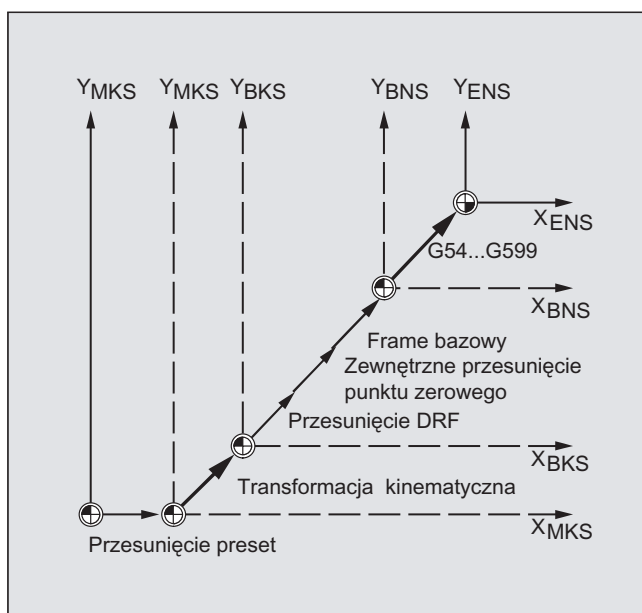
CFINE:	Przesunięcie dokładne, addytywne (translacja).
CTrans:	Przesunięcie zgrubne, absolutne (translacja).
TRANS:	Frame programowany: Przesunięcie zgrubne, absolutne (translacja).
X, Y, Z:	Współrzędne
<Kn>:	Współrzędna X, Y lub Z
<wartość>:	Wartość przesunięcia

5.4 Zewnętrzne przesunięcie punktu zerowego

Funkcja

Stwarza to kolejną możliwość przesunięcia punktu zerowego między bazowym układem współrzędnych i układem współrzędnych obrabianego przedmiotu.

Przy zewnętrznym przesunięciu punktu zerowego mogą być programowane tylko przesunięcia liniowe.



Programowanie

Programowanie wartości przesunięcia, \$AA_ETRANS następuje przez ustawienie zmiennych systemowych specyficznych dla osi.

Przyporządkowanie wartości przesunięcia

`$AA_ETRANS [oś] = RI`

RI jest zmienną obliczeniową typu REAL, która zawiera nową wartość.

Przesunięcie zewnętrzne z reguły nie jest podawane w programie obróbki, lecz ustawiane przez PLC.

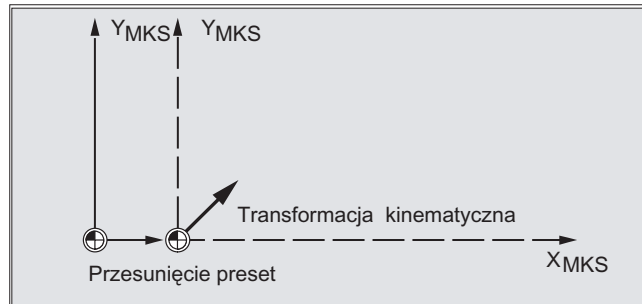
Wskazówka

Wartość napisana w programie obróbki działa dopiero wtedy, gdy na interfejsie VDI (interfejs NCU-PLC) jest ustawiony odpowiedni sygnał.

5.5 Przesunięcie preset przy pomocy PRESETON

Funkcja

Dla specjalnych zastosowań może być wymagane przyporządkowanie do wybazowanej już osi maszyny nowej wartości rzeczywistej przy pomocy `PRESETON`. Odpowiada to przesunięciu punktu zerowego w układzie współrzędnych maszyny.



⚠ OSTROŻNIE

Brak współrzędnych odniesienia

Po `PRESETON` oś maszyny znajduje się w statusie "nie wybazowana". Dlatego zaleca się stosowanie funkcji tylko dla osi maszyny, dla których nie ma obowiązku bazowania. W celu przywrócenia pierwotnego układu współrzędnych maszyny oś maszyny musi być ponownie bazowana, np. przy pomocy `G74` (bazowanie do punktu odniesienia).

Literatura:

Podręcznik programowania Podstawy

Składnia

```
PRESETON (<oś_1>, <wartość_1>, <oś_2>, <wartość_2>, ...)
```

Wskazówka

Przy pomocy `PRESETON` można zaprogramować przesunięcia Preset dla maks. 8 osi.

Znaczenie

<code>PRESETON:</code>	predefiniowana procedura do ustawienia wartości rzeczywistej
<code><oś_...>:</code>	Identyfikator osi maszyny, której punkt zerowy sterowania ma zostać zmieniony
	Zakres wartości: wszystkie identyfikatory osi zdefiniowane w kanale
<code><wartość_...>:</code>	nowa wartość rzeczywista osi maszyny w układzie współrzędnych maszyny

Przykład

Oś geometryczna: A

Przynależna oś maszyny: X1

Kod programu	Komentarz
N10 G0 A100	; Oś A wykonuje ruch do pozycji 100.
N20 PRESETON(X1,50)	; Oś maszyny X1 otrzymuje w pozycji 100 nową wartość rzeczywistą 50 => nowe wyświetlenie wartości rzeczywistej: - oś X1, MKS: 50 - oś A, WKS: 50
N30 A100	; Oś A wykonuje ruch 50 mm do pozycji 100.

Literatura

Odnosnie programowania przesunięcia preset w akcjach synchronicznych patrz:

Podręcznik działania Akcje synchroniczne; Rozdział: "Akcje w akcjach synchronicznych" > "Ustawienie wartości rzeczywistej (PRESETON)"

5.6 Obliczenie frame z 3 punktów pomiarowych w przestrzeni (MEAFRAME)

Funkcja

Funkcja MEAFRAME jest stosowana do wspierania cykli pomiarowych. Oblicza ona Frame z trzech idealnych i korespondujących zmierzonych punktów.

Gdy obrabiany przedmiot jest pozycjonowany w celu obróbki, jego rzeczywista pozycja jest względem do kartezjańskiego układu współrzędnych maszyny (pozycja idealna) zazwyczaj zarówno przesunięta, jak też obrócona. W celu dokładnej obróbki albo pomiaru jest konieczne albo kosztowne ustawienie fizyczne albo zmiana ruchów w programie obróbki.

Frame można ustalić przez odczyt trzech punktów w przestrzeni, których idealne pozycje są znane. Odczyt następuje przy pomocy sondy dotykowej lub optycznej, która dotyka specjalnych otworów, precyzyjnie ustalonych na płycie nośnej, albo kulek pomiarowych.

Składnia

MEAFRAME(<punkty idealne>,<punkty pomiarowe>,<jakość>)

Znaczenie

MEAFRAME:	Wywołanie funkcji
<punkty idealne>:	2-wymiarowa tablica REAL, zawierająca trzy współrzędne punktów idealnych
<punkty pomiarowe>:	2-wymiarowa tablica REAL, zawierająca trzy współrzędne punktów zmierzonych
<jakość>:	Zmienna, przy pomocy której są zwracane informacje dot. jakości obliczenia FRAME
	Typ: VAR REAL
	Wartość: -1 Punkty idealne leżą prawie na jednej prostej: Frame nie można było obliczyć. Zwracana zmienna FRAME zawiera neutralny Frame.
	-2 Punkty pomiarowe leżą prawie na jednej prostej: Frame nie można było obliczyć. Zwracana zmienna FRAME zawiera neutralny Frame.
	-4 Obliczenie macierzy obrotu nie udaje się z innego powodu.
	≥ 0.0 Suma zniekształceń (odstępny między punktami), które są potrzebne do zamiany zmierzonego trójkąta na trójkąt zgodny z trójkątem idealnym.

Wskazówka**Jakość pomiaru**

Aby współrzędne zmierzone mogły zostać przyporządkowane do współrzędnych idealnych przy pomocy kombinacji obrót/translacja, trójkąt wyznaczony przez zmierzone punkty musi być zgodny z trójkątem idealnym. Jest to realizowane przy pomocy algorytmu kompensacji, który minimalizuje sumę kwadratów odchyłań między trójkątem zmierzonym i idealnym, zmieniających trójkąt zmierzony na idealny.

Efektywnie zniekształcenie punktów pomiarowych może służyć, jako wskaźnik jakości pomiaru i jest dlatego wyprowadzane przez MEAFRAME, jako dodatkowa zmienna.

Wskazówka

Frame utworzony przez MEAFRAME może poprzez funkcję ADDFRAME zostać przetransformowany na inny Frame w łańcuchu Frame (patrz przykład "Powiązanie przy pomocy ADDFRAME").

Przykłady

Przykład 1:

Program obróbki 1:

Kod programu
...
DEF FRAME CORR_FRAME

Ustawienie punktów pomiarowych:

Kod programu	Komentarz
DEF REAL IDEAL_POINT[3,3]=SET(10.0,0.0,0.0,0.0,10.0,0.0,0.0,0.0,10.0)	
DEF REAL MEAS_POINT[3,3]=SET(10.1,0.2,-0.2,-0.2,10.2,0.1,-0.2,0.2,9.8)	; Dla testu.
DEF REAL FIT_QUALITY=0	
DEF REAL ROT_FRAME_LIMIT=5	; Pozwala na obrót pozycji przedmiotu o maks. 5 stopni.
DEF REAL FIT_QUALITY_LIMIT=3	; Pozwala na przesunięcie o maks. 3 mm między trójkątem idealnym i trójkątem zmierzonym.
DEF REAL SHOW_MCS_POS1[3]	
DEF REAL SHOW_MCS_POS2[3]	
DEF REAL SHOW_MCS_POS3[3]	

Kod programu	Komentarz
N100 G01 G90 F5000	
N110 X0 Y0 Z0	
N200 CORR_FRAME=MEAFRAME(IDEAL_POINT,MEAS_POINT,FIT_QUALITY)	
N230 IF FIT_QUALITY < 0	
SETAL(65000)	
GOTOF NO_FRAME	
ENDIF	
N240 IF FIT_QUALITY > FIT_QUALITY_LIMIT	
SETAL(65010)	
GOTOF NO_FRAME	
ENDIF	
N250 IF CORR_FRAME[X,RT] > ROT_FRAME_LIMIT	; Ograniczenie 1. kąta RPY
SETAL(65020)	
GOTOF NO_FRAME	
ENDIF	
N260 IF CORR_FRAME[Y,RT] > ROT_FRAME_LIMIT	; Ograniczenie 2. kąta RPY.
SETAL(65021)	
GOTOF NO_FRAME	
ENDIF	

5.6 Obliczenie frame z 3 punktów pomiarowych w przestrzeni (MEAFRAME)

Kod programu	Komentarz
N270 IF CORR_FRAME[Z,RT] > ROT_FRAME_LIMIT	; Ograniczenie 3. kąta RPY.
SETAL(65022)	
GOTOF NO_FRAME	
ENDIF	
N300 \$P_IFRAME=CORR_FRAME	; Uaktywnienie Frame próbkowania z Frame ustawianym. ; Sprawdzenie Frame przez pozycjonowanie osi geometrycznych na punkty idealne.
N400 X=IDEAL_POINT[0,0] Y=IDEAL_POINT[0,1] Z=IDEAL_POINT[0,2]	
N410 SHOW_MCS_POS1[0]=\$AA_IM[X]	
N420 SHOW_MCS_POS1[1]=\$AA_IM[Y]	
N430 SHOW_MCS_POS1[2]=\$AA_IM[Z]	
N500 X=IDEAL_POINT[1,0] Y=IDEAL_POINT[1,1] Z=IDEAL_POINT[1,2]	
N510 SHOW_MCS_POS2[0]=\$AA_IM[X]	
N520 SHOW_MCS_POS2[1]=\$AA_IM[Y]	
N530 SHOW_MCS_POS2[2]=\$AA_IM[Z]	
N600 X=IDEAL_POINT[2,0] Y=IDEAL_POINT[2,1] Z=IDEAL_POINT[2,2]	
N610 SHOW_MCS_POS3[0]=\$AA_IM[X]	
N620 SHOW_MCS_POS3[1]=\$AA_IM[Y]	
N630 SHOW_MCS_POS3[2]=\$AA_IM[Z]	
N700 G500	; Wyłączenie aktywności frame ustawianego, ponieważ z frame zerowym (ustawienie wstępne braku wpisu wartości).
No_FRAME	; Wyłączenie Frame ustawianego, ponieważ jest wstępnie ustawiony Frame zerowy (wartość nie jest wpisana).
M0	
M30	

Przykład 2: powiązanie Frame

Powiązanie MEAFRAME dla korekcji

Funkcja MEAFRAME daje Frame korekcyjny. Jeżeli ten Frame zostanie powiązany z nastawnym Frame \$P_UIFR[1], który był aktywny przy wywołaniu funkcji, (np. G54), wówczas uzyskuje się nastawny Frame dla dalszych przeliczeń w celu wykonywania ruchów albo obróbki.

Powiązanie przy pomocy ADDFRAME

Jeżeli ten Frame korekcyjny ma w łańcuchu Frame działać w innym miejscu albo gdy przed ustawianym Frame są aktywne jeszcze inne Frame, wówczas funkcja ADDFRAME() może zostać użyta do powiązania w jednym z Frame bazowych kanału albo we Frame systemowym.

We frame nie może być przy tym aktywne:

- lustrzane odbicie przy pomocy MIRROR
- skalowanie przy pomocy SCALE

Parametrami wejściowymi dla wartości zadanych i rzeczywistych są współrzędne obrabianego przedmiotu. W systemie podstawowym sterowania współrzędne te należy stale podawać metrycznie albo calowo (G71/G70) i jako wymiar odniesiony do promienia (DIAMOF).

Literatura:

Dalsze informacje dot. ADDFRAME patrz:

Podręcznik działania Funkcje podstawowe; K2: Osie, Układy współrzędnych, Frame

5.7 Frame globalne dla NCU

Funkcja

Frame globalne dla NCU występują w NCU tylko jeden raz dla wszystkich kanałów. Frame globalne dla NCU mogą być czytane i zapisywane ze wszystkich kanałów. Uaktywnienie frame globalnych dla NCU następuje w każdorazowym kanale.

Przez frame globalne **osie kanału i osie maszyny** z przesunięciami mogą być skalowane i poddawane lustrzanemu odbiciu.

Zależności geometryczne i łańcuchy frame

W przypadku frame globalnych nie istnieje zależność geometryczna między osiami. Dlatego nie mogą być wykonywane żadne obroty, ani programowanie identyfikatorów osi geometrycznych.

- Do frame globalnych nie dają się użyć żadne rotacje. Zaprogramowanie obrotu jest odrzucane z alarmem: "18310 Kanał %1 blok %2 Frame: obrót niedopuszczalny".
- Powiązanie frame globalnych i frame kanałowych jest możliwe. Frame wynikowy zawiera wszystkie składowe frame łącznie z rotacją dla wszystkich osi. Przyporządkowanie Frame ze składowymi obrotowymi do Frame globalnego jest odrzucane z alarmem "Frame: obrót niedopuszczalny".

Frame globalne dla NCU

Globalne dla NCU frame bazowe \$P_NCBFR[n]

Można zaprojektować do 8 frame bazowych globalnych dla NCU:

Równocześnie mogą występować kanałowe frame bazowe.

Frame globalne mogą być czytane i zapisywane ze wszystkich kanałów NCU. Przy zapisie frame globalnych użytkownik powinien zatroszczyć się o koordynację kanałów. Można to realizować np. przez znaczniki Wait (WAITMC).

Producent maszyny

Liczba globalnych Frame bazowych jest projektowana poprzez dane maszynowe.

Literatura:

Podręcznik działania Funkcje podstawowe; Osie, Układy współrzędnych, Frame (K2)

Globalne dla NCU Frame ustawiane \$P_UIFR[n]

Wszystkie Frame ustawiane G500, G54...G599 mogą być projektowane jako globalne dla NCU, lub jako specyficzne dla kanału.

Producent maszyny

Wszystkie Frame ustawiane mogą przy pomocy danej maszynowej MD18601 \$MN_MM_NUM_GLOBAL_USER_FRAMES zostać przeprojektowane na Frame globalne.

Jako identyfikatory osi w przypadku poleceń programowania frame mogą być stosowane identyfikatory kanału i identyfikatory osi maszyny. Programowanie identyfikatorów osi geometrycznych jest odrzucane z alarmem.

5.7.1 Frame kanałowe (\$P_CHBFR, \$P_UBFR)

Funkcja

Frame ustawiane albo Frame bazowe mogą poprzez program obróbki i poprzez BTSS być zapisywane i czytane przez obsługę i przez PLC.

Przesunięcie dokładne jest możliwe również dla frame globalnych. Wyłączanie Frame globalnych następuje tak, jak w przypadku Frame specyficznych dla kanału poprzez G53, G153, SUPA i G500.

Producent maszyny

Przez daną maszynową MD28081 \$MC_MM_NUM_BASE_FRAMES można zaprojektować liczbę Frame bazowych w kanale. Konfiguracja standardowa jest tak zaprojektowana, że jest co najmniej jeden frame bazowy na kanał. Maksymalnie jest możliwych 8 frame bazowych na kanał. Dodatkowo do 8 frame bazowych w kanale może być jeszcze 8 frame globalnych NCU.

Frame specyficzne dla kanału

\$P_CHBFR[n]

Poprzez zmienną systemową \$P_CHBFR[n] mogą być czytane i zapisywane Frame bazowe. Przy zapisaniu Frame bazowego powiązany łączny Frame bazowy nie jest uaktywniany, lecz uaktywnienie następuje dopiero przy wykonaniu instrukcji G500, G54...G599. Zmienna służy przeważnie jako pamięć dla procesów zapisu na frame bazowym z HMI lub PLC. Te zmienne frame są zabezpieczane przez wykonanie kopii zapasowej danych.

Pierwszy frame bazowy w kanale

Zapisanie na predefiniowaną zmienną \$P_UBFR uaktywnia Frame bazowy z indeksem tablicy 0 nie równocześnie, lecz uaktywnienie następuje dopiero z wykonaniem instrukcji G500, G54 ... G599. Zmienna może być zapisywana i czytana również w programie.

\$P_UBFR

\$P_UBFR jest identyczna z \$P_CHBFR[0]. Standardowo jest zawsze jeden frame bazowy w kanale, tak że zmienna systemowa jest kompatybilna ze starszymi wersjami. Jeżeli nie ma specyficznego dla kanału Frame bazowego, jest przy zapisie albo odczycie wyprowadzany alarm "Frame: Instrukcja niedopuszczalna".

5.7.2 Frame działające w kanale

Funkcja

Frame działające w kanale są wprowadzane z programu obróbki przez odnośne zmienne systemowe tych frame. Należą do tego również frame systemowe. Przez te zmienne systemowe można w programie obróbki czytać i zapisywać aktualny frame systemowy.

Aktualne frame działające w kanale

Przegląd

Aktualne frame systemowe

\$P_PARTFRAME

dla:

TCARR i PAROT

\$P_SETFRAME

Ustawienie wartości rzeczywistej i zadrapanie

\$P_EXTFRAME

Zewnętrzne przesunięcie punktu zerowego

\$P_NCBFRAME[n]

Aktualne frame bazowe globalne dla NCU

\$P_CHBFRAME[n]

Aktualne kanałowe frame bazowe

\$P_BFRAME

Aktualny 1. frame bazowy w kanale

\$P_ACTBFRAME

Całkowity frame bazowy

\$P_CHBFRMASK i \$P_NCBFRMASK

Całkowity frame bazowy

\$P_IFRAME

Aktualny frame ustawiany

Aktualne frame systemowe

dla:

\$P_TOOLFRAME

TOROT i TOFRAME

\$P_WPFRAME

Punkty odniesienia narzędzia

\$P_TRAFRAME

Transformacje

\$P_PFRAME

Aktualny programowany frame.

Aktualny frame systemowy

dla:

\$P_CYCFRAME

Cykle

P_ACTFRAME

Aktualny frame całkowity

Powiązanie FRAME

Frame aktualny składa się z całkowitego frame bazowego

\$P_NCBFRAME[n] Aktualne Frame bazowe globalne dla NCU

Poprzez zmienną systemową \$P_NCBFRAME[n] mogą być czytane i zapisywane aktualne globalne elementy tablicy Frame bazowego. Wynikający całkowity frame bazowy jest wliczany przez proces zapisu w kanale.

Zmieniony frame będzie aktywny tylko w kanale, w którym został zaprogramowany. Jeżeli frame ma zostać zmieniony dla wszystkich kanałów NCU, konieczne jest równoczesne zapisanie \$P_NCBFR[n] i \$P_NCBFRAME[n]. Inne kanały muszą wówczas jeszcze uaktywnić Frame przy pomocy np. G54. Przy zapisie frame bazowego całkowity frame bazowy jest obliczany na nowo.

\$P_CHBFRAME[n] Aktualne frame bazowe kanału

Poprzez zmienną systemową \$P_CHBFRAME[n] mogą być czytane i zapisywane aktualne elementy tablicy Frame bazowego kanału. Wynikający całkowity frame bazowy jest wliczany przez proces zapisu w kanale. Przy zapisie frame bazowego całkowity frame bazowy jest obliczany na nowo.

\$P_BFRAME Aktualny 1. frame bazowy w kanale

Poprzez predefiniowaną zmienną Frame \$P_BFRAME aktualny Frame bazowy z indeksem tablicy 0, który obowiązuje w kanale, może być czytany i zapisywany w programie obróbki. Zapisany frame bazowy jest natychmiast wliczany.

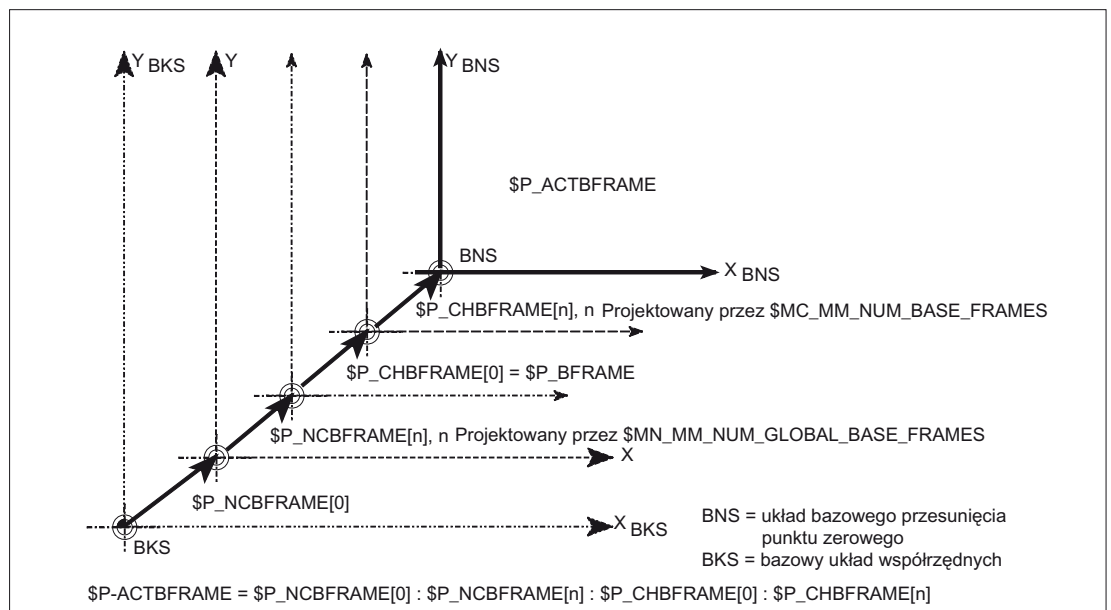
\$P_BFRAME jest identyczna z \$P_CHBFRAME[0]. Zmienna systemowa ma standardowo zawsze poprawną wartość. Jeżeli nie ma specyficznego dla kanału Frame bazowego, jest przy zapisie albo odczycie wyprowadzany alarm "Frame: Instrukcja niedopuszczalna".

\$P_ACTBFRAME Całkowity frame bazowy

Zmienna \$P_ACTBFRAME oblicza powiązany całkowity Frame bazowy. Zmienna może być tylko czytana.

\$P_ACTBFRAME odpowiada:

$\$P_NCBFRAME[0] : \dots : \$P_NCBFRAME[n] : \$P_CHBFRAME[0] : \dots : \$P_CHBFRAME[n]$.



\$P_CHBFRMASK i \$P_NCBFRMASK Całkowity frame bazowy

Poprzez zmienne systemowe \$P_CHBFRMASK i \$P_NCBFRMASK użytkownik może wybrać, jakie Frame bazowe chce włączyć do obliczenia "całkowitego" Frame bazowego. Zmienne mogą być programowane tylko w programie i czytane przez BTSS. Wartość zmiennych jest interpretowana jako maska bitowa i podaje, który element tablicy Frame bazowego \$P_ACTBFAME jest brany do obliczenia.

Przy pomocy \$P_CHBFRMASK można zadać, które specyficzne dla kanału Frame bazowe, a przy pomocy \$P_NCBFRMASK, które globalne dla NCU Frame bazowe są wliczane.

Przez zaprogramowanie zmiennych całkowity Frame bazowy i Frame całkowity jest obliczany na nowo. Po Reset i w ustawieniu podstawowym wartość \$P_CHBFRMASK i \$P_NCBFRMASK jest następująca:

```
$P_CHBFRMASK = $MC_CHBFRAME_RESET_MASK
```

```
$P_NCBFRMASK = $MC_CHBFRAME_RESET_MASK
```

Przykład:

```
$P_NCBFRMASK = 'H81' ;$P_NCBFRAME[0] : $P_NCBFRAME[7]
```

```
$P_CHBFRMASK = 'H11' ;$P_CHBFRAME[0] : $P_CHBFRAME[4]
```

\$P_IFRAME Aktualny ustawiany frame

Poprzez predefiniowaną zmienną Frame \$P_IFRAME aktualny ustawiany Frame, który obowiązuje w kanale, może być czytany i zapisywany w programie obróbki. Zapisany ustawiany frame jest wliczany natychmiast.

W przypadku globalnych dla NCU frame ustawianych zmieniony frame działa tylko w tym kanale, w którym jest zaprogramowany. Jeżeli Frame ma zostać zmieniony dla wszystkich kanałów NCU, musi równocześnie nastąpić opisanie \$P_UIFR[n] i \$P_IFRAME. Inne kanały muszą wówczas jeszcze uaktywnić odpowiedni Frame przy pomocy np. G54.

\$P_PFRAME Aktualny frame programowany

\$P_PFRAME jest to Frame programowany, który wynika z zaprogramowania TRANS/ATRANS, G58/G59, ROT/AROT, SCALE/ASCALE, MIRROR/AMIRROR wzgl. z przyporządkowania CTRANS, CROT, CMIRROR, CSCALE do programowanego FRAME.

Aktualna, programowana zmienna frame, która tworzy odniesienie między układem ustawianego przesunięcia punktu zerowego (ENS) i układem współrzędnych obrabianego przedmiotu (WKS).

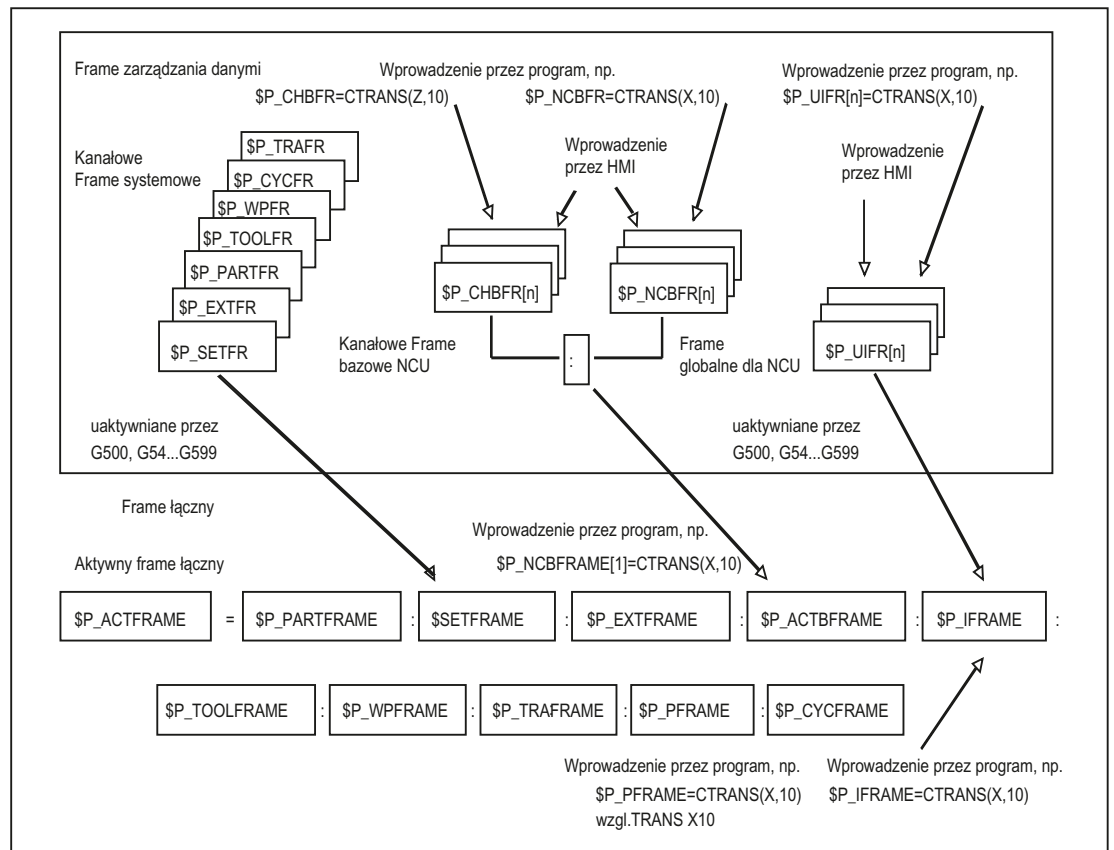
P_ACTFRAME Aktualny frame całkowity

Aktualny wynikający frame łączny \$P_ACTFRAME wynika teraz jako powiązanie wszystkich frame bazowych, aktualnego frame ustawianego i frame programowanego. Aktualny frame jest uaktualniany zawsze wtedy, gdy zmienia się jego składowa.

\$P_ACTFRAME odpowiada:

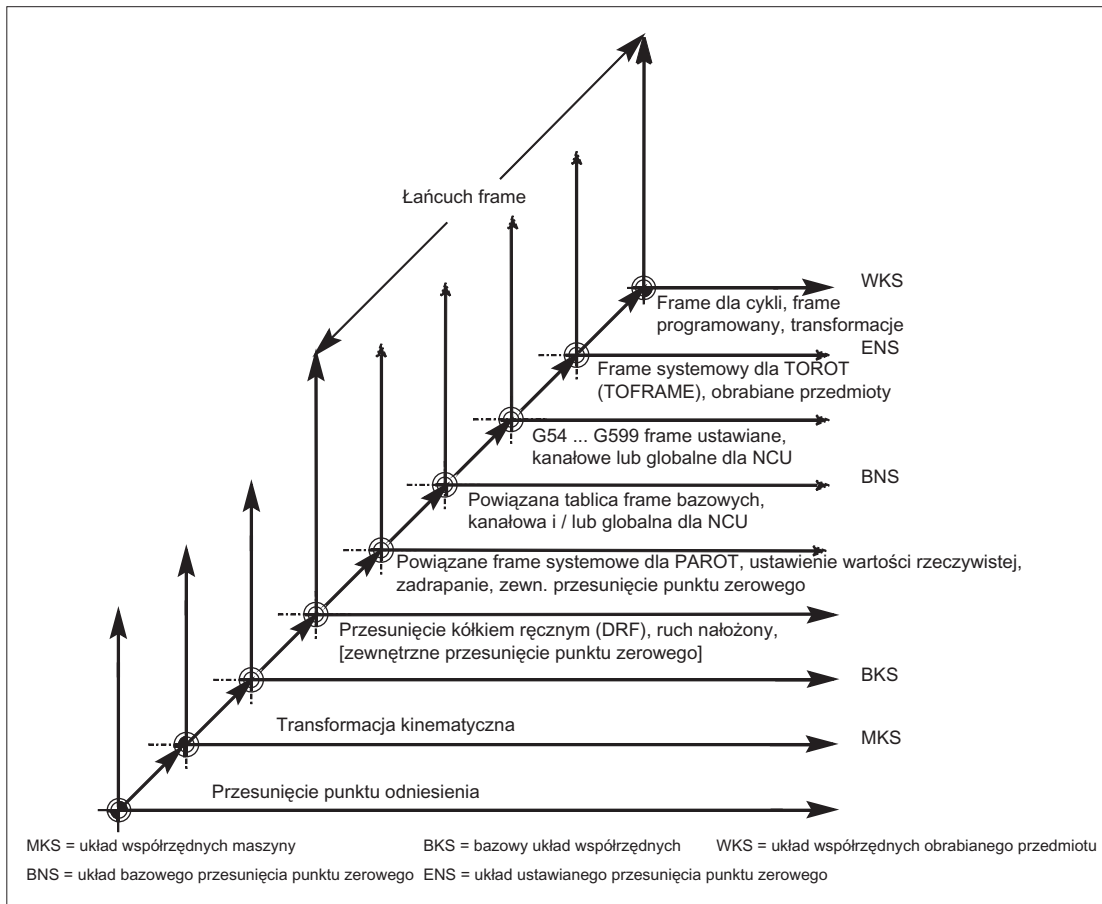
\$P_PARTFRAME : \$P_SETFRAME : \$P_EXTFRAME : \$P_ACTBFRAME : \$P_IFRAME :

\$P_TOOLFRAME : \$P_WPFRAME : \$P_TRAFRAME : \$P_PFRAME : \$P_CYCFRAME



Powiązanie frame

Aktualny frame składa się z całkowitego frame bazowego, frame ustawianego, frame systemowego i frame programowanego według wyżej podanego frame całkowitego.



Transformacje

6.1 Ogólne programowanie rodzajów transformacji

Funkcja ogólna

W celu dopasowywania sterowania do różnych kinematyk maszyny można wybrać programowanie rodzajów transformacji przy pomocy odpowiednich parametrów. Przez te parametry można dla wybranej transformacji odpowiednio uzgodnić zarówno orientację narzędzia w przestrzeni, jak też ruchy orientacji.

Przy transformacjach trzy-, cztero- i pięcioosiowych zaprogramowane dane dot. pozycji odnoszą się zawsze do wierzchołka narzędzia, który jest ortogonalnie aktualizowany do znajdującej się w przestrzeni powierzchni obróbki. Współrzędne kartezjańskie są przeliczane z bazowego układu współrzędnych na układ współrzędnych maszyny i odnoszą się do osi geometrycznych. Te opisują punkt roboczy. Wirtualne osie obrotowe opisują orientację narzędzia w przestrzeni i są programowane przy pomocy TRAORI.

Przy transformacji kinematycznej pozycje mogą być programowane w kartezjańskim układzie współrzędnych. Sterowanie transformuje zaprogramowane przy pomocy TRANSMIT, TRACYL i TRAANG ruchy kartezjańskiego układu współrzędnych na ruchy realnych osi maszyny.

Programowanie

Transformacje trzy-, cztero- i pięcioosiowe TRAORI

Uzgodniona transformacja orientacji jest uaktywniana przy pomocy polecenia TRAORI i trzech możliwych parametrów dla numeru transformacji, wektora orientacji i offsetów osi obrotowych.

TRAORI (numer transformacji, wektor orientacji, offsety osi obrotowych)

Transformacje kinematyczne

Do transformacji kinematycznych należą uzgodnione transformacje TRANSMIT (numer transformacji)

TRACYL (średnica robocza, numer transformacji)

TRAANG (kąt osi ustawionej skośnie, numer transformacji)

Wyłączenie aktywnej transformacji

Przy pomocy TRAF00F można wyłączyć właśnie aktywną transformację.

Transformacja orientacji

Transformacje trzy-, cztero- i pięcioosiowe TRAORI

Do optymalnej obróbki powierzchni ukształtowanych przestrzennie w przestrzeni roboczej maszyny, obrabiarki potrzebują oprócz trzech osi liniowych X, Y i Z, jeszcze osi dodatkowych. Dodatkowe osie opisują orientację w przestrzeni i są dalej nazywane osiami orientacji. Są one do dyspozycji jako osie obrotu w przypadku czterech typów maszyn z różną kinematyką.

1. Dwuosiowa głowica skrętna, np. głowica narzędziowa Kardana z osią obrotową równoległą do osi liniowej przy nieruchomym stole narzędziowym.
2. Dwuosiowy stół obrotowy, np. nieruchoma głowica skrętna ze stołem narzędziowym obracającym wokół dwóch osi.
3. Jednoosiowa głowica skrętna i jednoosiowy stół obrotowy, np. obrotowa głowica skrętna z narzędziem obróconym przy stole narzędziowym obracającym wokół jednej osi.
4. Dwuosiowa głowica skrętna i jednoosiowy stół obrotowy, np. przy stole narzędziowym obracającym wokół jednej osi i obrotowa głowica skrętna z narzędziem obracającym wokół własnej osi.

Transformacje 3- i 4-osiowe są specjalnymi specjalnymi formami transformacji 5-osiowej i są programowane analogicznie do transformacji 5-osiowych.

"Transformacja rodzajowa 3-/4-/5-/6-osiowa pokrywa swoim zakresem funkcji prostopadle umieszczone osie obrotowe jak też transformacje dla głowicy frezarskiej Kardana i może jak każda inna transformacja orientacji być uaktywniana przy pomocy TRAORI również dla tych czterech typów maszyn. W przypadku rodzajowej transformacji 5/6 osiowej orientacja narzędzia ma kolejny trzeci stopień swobody, przy którym oprócz dowolnego kierunku narzędzia w przestrzeni narzędzie może być obracane wokół własnej osi.

Literatura: /FB3/ Podręcznik działania Funkcje specjalne; Transformacja 3- do 5-osiowej (F2)

Niezależne od kinematyki położenie podstawowe orientacji narzędzia

ORIRESET

Jeżeli przy pomocy TRAORI jest aktywna transformacja orientacji, wówczas można przy pomocy ORIRESET podać położenia podstawowe do 3 osi orientacji z opcjonalnymi parametrami A, B, C. Przyporządkowanie kolejności zaprogramowanych parametrów do osi obrotowych następuje według kolejności osi orientacji ustalonej przez transformację. Zaprogramowanie ORIRESET(A, B, C) powoduje, że osie orientacji wykonują ruch liniowy i synchroniczny od swojej aktualnej pozycji do podanej pozycji położenia podstawowego.

Transformacje kinematyczne

TRANSMIT i TRACYL

Przy obróbkach frezarskich na tokarkach można dla uzgodnionej transformacji programować albo

1. obróbkę czołową w zamocowaniu obrotowym przy pomocy TRANSMIT albo
2. obróbkę dowolnie przebiegających rowków na elementach walcowych przy pomocy TRACYL

TRAANG

Jeżeli oś dosuwu, np. dla technologii szlifowania, ma być dosuwana również skośnie, wówczas można przy pomocy TRAANG dla uzgodnionej transformacji zaprogramować parametryzowalny kąt.

Ruch kartezjański PTP

Do transformacji kinematycznej należy również "ruch kartezjański PTP" przy którym można programować do 8 różnych położeń przegubu STAT=. Pozycje są programowane w kartezjańskim układzie współrzędnych, przy czym ruch maszyny następuje we współrzędnych maszyny.

Literatura:

/FB2/ Podręcznik działania Funkcje rozszerzające; Transformacja kinematyczna (M1)

Transformacje powiązane

Każdorazowo można połączyć dwie transformacje jedna za drugą. W przypadku powiązanej w ten sposób drugiej transformacji składowe ruchu osi są przejmowane z pierwszej transformacji.

Jako pierwsza transformacja są możliwe:

- Transformacja orientacji TRAORI
- Transformacja biegunowa TRANSMIT
- Transformacja walcowa TRACYL
- Transformacja osi skośnej TRAANG

Druga transformacja musi być skośną osią TRAANG

6.1.1 Ruchy orientacji przy transformacjach

Ruchy postępowe i ruchy orientacji

Ruchy programowanych orientacji zależą pierwotnie od typu maszyny. Przy transformacji trzy-, cztero- i pięcioosiowej z TRAORI osie obrotowe albo osie skrętne osie, osie liniowe opisują ruchy orientacji narzędzia.

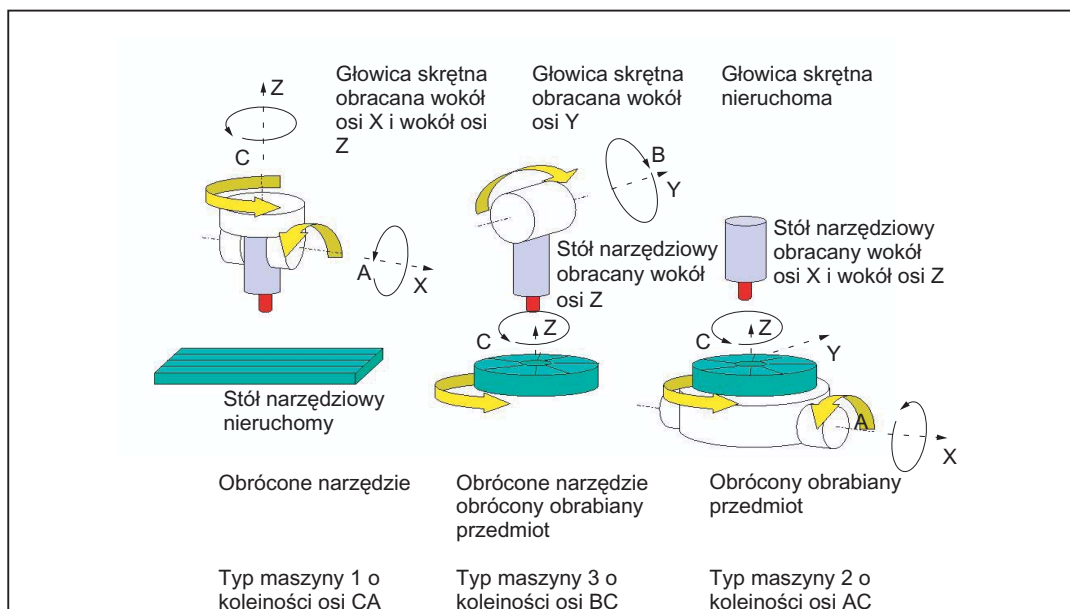
Zmiany pozycji osi obrotowych uczestniczących w transformacji orientacji prowadzą do ruchów wyrównawczych pozostałych osi maszyny. Położenie wierzchołka narzędzia pozostaje przy tym niezmienione.

Ruchy orientacji narzędzia mogą być programowane przez identyfikatory osi obrotowych A..., B..., C... osi wirtualnych zależnie od zastosowania albo przez podanie kątów Eulera wzgl. kątów RPY albo wektorów kierunkowych wzgl. wektorów normalnej powierzchni, wektorów normalizowanych dla osi obrotu stożka albo dla orientacji pośredniej na pobocznicę stożka.

Przy transformacji kinematycznej z TRANSMIT, TRACYL i TRAANG sterowanie transformuje zaprogramowane ruchy kartezjańskiego układu współrzędnych na ruchy realnych osi maszyny.

Kinematyka maszyny przy transformacji trzy-, cztero- i pięcioosiowej

Obrotowe może być albo narzędzie albo stół narzędziowy z maksymalnie dwoma osiami obrotowymi. Kombinacja każdorazowo jednoosiowej głowicy skrętniej i stołu obrotowego jest również możliwa.



Typ maszyny

Programowanie orientacji

Transformacja trzyosiowa
typy maszyn 1 i 2

Programowanie orientacji narzędzia tylko w płaszczyźnie, która jest prostopadła do osi obrotowej. Istnieją **dwie** osie ruchu postępowego (liniowe) i **jedna** oś rotacyjna (obrotowa).

Transformacja czteroosiowa
typy maszyn 1 i 2

Programowanie orientacji narzędzia tylko w płaszczyźnie, która jest prostopadła do osi obrotowej. Istnieją **trzy** osie ruchu postępowego (liniowe) i **jedna** oś rotacyjna (obrotowa).

Transformacja pięcioosiowa
typ maszyny 3
jednoosiowa głowica skrętnaobrotowymi.

Programowanie transformacji orientacji. Kinematyka z **trzema** osiami liniowymi i **dwoma** ortogonalnymi osiami

i jednoosiowy stół obrotowy

Osie obrotowe są równoległe do dwóch z trzech osi liniowych. Pierwsza oś obrotowa jest poruszana od dwóch kartezjańskich osi liniowych. Obraca ona trzecią osią liniową z narzędziem. Druga oś obrotowa obraca obrabianym przedmiotem.

Transformacje rodzajowe 5/6-osiowe

Typ maszyny	Programowanie transformacji orientacji
Transformacja rodzajowa pięć-/sześćosiowa typy maszyn 4	Programowanie transformacji orientacji. Kinematyka z trzema osiami liniowymi i trzema ortogonalnymi osiami obrotowymi.
Dwuosiowa głowica skrętna z obrotowym narzędziem i jednoosiowy stół obrotowy	Osie obrotowe są równoległe do dwóch z trzech osi liniowych. Pierwsza oś obrotowa jest poruszana od dwóch kartezjańskich osi liniowych. Obraca ona trzecią oś liniową z narzędziem. Druga oś obrotowa obraca obrabianym przedmiotem. Podstawowa orientacja narzędzia może zostać zaprogramowana przez dodatkowy obrót wokół niego o kąt THETA.

Przy wywołaniu "transformacji rodzajowej trzy-, cztero- i pięć-/sześćosiowej" można dodatkowo przekazać orientację podstawową narzędzia. Ograniczenia odnośnie kierunków osi obrotowych już nie obowiązują. Jeżeli osie obrotowe nie są dokładnie prostopadłe do siebie lub występujące osie obrotowe nie są ustawione dokładnie równoległe do osi liniowych, "rodzajowa transformacja pięć-/sześćosiowa" może dać lepsze wyniki orientacji narzędzia.

Transformacje kinematyczne TRANSMIT, TRACYL i TRAANG

Dla obróbek frezarskich na tokarkach albo skośnie dosuwanej osi przy szlifowaniu obowiązują zależnie od transformacji w przypadku standardowym następujące układy osi:

TRANSMIT	Uaktywnienie transformacji biegunowej
Obróbka czołowa w zamocowaniu tokarskim	jedna oś obrotowa jedna oś dosuwu prostopadła do osi toczenia jedna oś podłużna równoległa do osi toczenia
TRACYL	Uaktywnienie transformacji poboczniczy walca
Obróbka dowolnie przebiegających rowków na elementach cylindrycznych	jedna oś obrotowa jedna oś dosuwu prostopadła do osi toczenia jedna oś podłużna równoległa do osi toczenia
TRAANG	Uaktywnienie transformacji z osią skośną
Obróbka ze skośną osią dosuwu	jedna oś obrotowa jedna oś dosuwu z parametryzowanym kątem jedna oś podłużna równoległa do osi toczenia

Ruch kartezjański PTP

Ruch maszyny następuje we współrzędnych maszyny i jest programowany przy pomocy:

TRAORI	Uaktywnienie transformacji
PTP ruch punkt do punktu	Ruch do pozycji w kartezjańskim układzie współrzędnych (MKS)
CP	Ruch po torze osi kartezjańskich w (BKS)
STAT	Położenie przegubów jest zależne od transformacji
TU	Kąt, o który osie wykonują ruch po najkrótszej drodze

Ruch PTP przy rodzajowej transformacji 5/6-osiowej

Ruch maszyny następuje we współrzędnych maszyny, a orientacja narzędzia może zostać zaprogramowana zarówno przy pomocy pozycji osi obrotowych jak również przy pomocy niezależnych od kinematyki wektorów kąt Eulera wzgl. kąt RPY albo wektorów biegunowych.

Jest przy tym możliwa interpolacja osi obrotowej, interpolacja wektorowa z interpolacją wielkiego okręgu lub interpolacja wektora orientacji na poboczniczy stożka.

Przykład, transformacja trzy- do pięcioosiowej w przypadku głowicy frezarskiej Kardana

Obrabiarka ma co najmniej 5 osi, z tego

- Trzy osie ruchu prostoliniowego, które przesuwają punkt roboczy do dowolnej pozycji w przestrzeni roboczej.
- Dwie obrotowe osie skrętne, które są umieszczone pod projektowalnym kątem (najczęściej 45 stopni), umożliwiają narzędziu przyjmowanie zorientowań w przestrzeni, które przy umieszczeniu pod kątem 45 stopni ograniczają się do półkuli.

6.1.2 Przegląd transformacji orientacji TRAORI

Możliwe rodzaje programowania w związku z TRAORI

Typ maszyny	Programowanie przy aktywnej transformacji TRAORI
<p>Typy maszyn 1, 2 albo 3, dwuosiowa głowica skrętna albo dwuosiowy stół obrotowy albo kombinacja jednoosiowej głowicy skrętnej i jednoosiowego stołu obrotowego.</p>	<p>Kolejność osi orientacji i kierunek orientacji narzędzia można projektować albo w odniesieniu do maszyny przez dane maszynowe zależnie od kinematyki maszyny albo w odniesieniu do obrabianego przedmiotu z programowalną orientacją niezależnie od kinematyki maszyny</p> <p>Kierunki obrotu osi orientacji w systemie odniesienia są programowane przy pomocy:</p> <ul style="list-style-type: none"> - ORIMKS układ odniesienia = układ współrzędnych maszyny - ORIWKS układ odniesienia = układ współrzędnych obrabianego przedmiotu <p>Położeniem podstawowym jest ORIWKS.</p> <p>Programowanie osi orientacji przy pomocy:</p> <ul style="list-style-type: none"> A, B, C pozycji osi maszyny bezpośrednio A2, B2, C2 programowanie kąta osi wirtualnych przy pomocy - ORIEULER przez kąt Eulera (standard) - ORIRPY przez kąt RPY - ORIVIRT1 przez wirtualne osie orientacji 1. definicja - ORIVIRT2 przez wirtualne osie orientacji 2. definicja <p>z rozróżnieniem rodzaju interpolacji:</p> <p>interpolacja liniowa</p> <ul style="list-style-type: none"> - ORIAXES osi orientacji lub osi maszyny <p>interpolacja dużego okręgu (interpolacja wektora orientacji)</p> <ul style="list-style-type: none"> - ORIVECT osi orientacji <p>Programowanie osi orientacji przez podanie A3, B3, C3 komponentów wektorowych (normalna kierunkowa/powierzchni)</p> <p>Programowanie wynikowej orientacji narzędzia</p> <ul style="list-style-type: none"> A4, B4, C4 wektora normalnej powierzchni na początku bloku A5, B5, C5 wektora normalnej powierzchni na końcu bloku LEAD kąt wyprzedzenia dla orientacji narzędzia TILT kąt boczny dla orientacji narzędzia

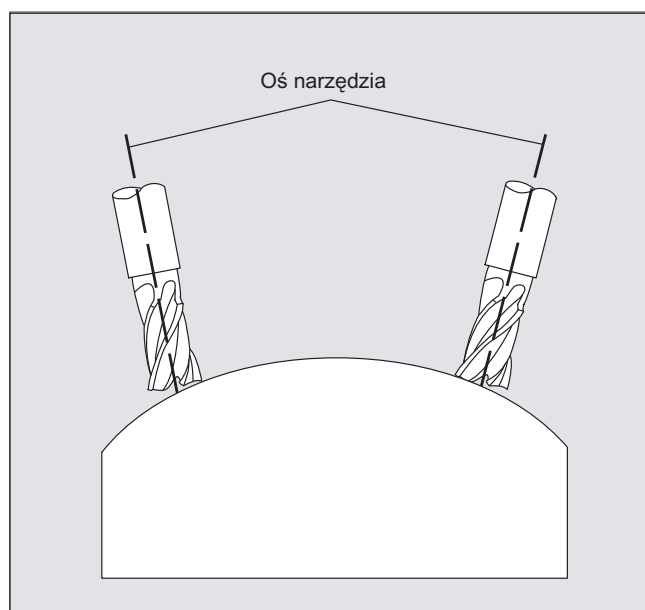
Typ maszyny	Programowanie przy aktywnej transformacji TRAORI
	<p>Interpolacja wektora orientacji na pobocznicy stożka Zmiany orientacji na znajdującej się dowolnie w przestrzeni pobocznicy stożka przez interpolację:</p> <ul style="list-style-type: none"> - ORIPLANE w płaszczyźnie (interpolacja dużego okręgu) - ORICONCW na pobocznicy stożka w kierunku ruchu wskazówek zegara - ORICONCCW na pobocznicy stożka przeciwnie do ruchu wskazówek zegara <p>A6, B6, C6 wektora kierunkowego (oś obrotu stożka) -OICONIO interpolacja na pobocznicy stożka z: A7, B7, C7 wektory pośrednie (orientacja początkowa i końcowa) albo</p> <ul style="list-style-type: none"> - ORICONTO na pobocznicy stożka przejście styczne <p>Zmiany orientacji odniesione do toru przy pomocy</p> <ul style="list-style-type: none"> - ORICURVE zadanie ruchu dwóch punktów kontaktowych przez PO[XH]=(xe, x2, x3, x4, x5) wielomiany orientacji max 5. stopnia PO[YH]=(ye, y2, y3, y4, y5) wielomiany orientacji max 5. stopnia PO[ZH]=(ze, z2, z3, z4, z5) wielomiany orientacji max 5. stopnia <ul style="list-style-type: none"> - ORIPATHS wygładzanie przebiegu orientacji przy pomocy A8, B8, C8. Fazy przeorientowania narzędzia odpowiada: Kierunek i długość drogi narzędzia przy cofnięciu
<p>Typy maszyn 1 i 3</p> <p>Dalsze typy maszyn z dodatkowym obrotem narzędzia wokół siebie samego wymagają 3. osi obrotowej</p> <p>Transformacja orientacji, jak np. rodzajowa transformacja 6-osiowa. Obroty wektora orientacji.</p>	<p>Programowanie obrotów orientacji narzędzia przy pomocy LEAD kąt wyprzedzenia w stosunku do wektora normalnej powierzchni</p> <p>PO[PHI] programowanie wielomianu max 5. stopnia</p> <p>TILT kąt boczny, obrót wokół stycznej do toru (kierunek Z)</p> <p>PO[PSI] programowanie wielomianu max 5. stopnia</p> <p>THETA kąt obrotu (obróć wokół kierunku narzędzia w Z)</p> <p>THETA= wartość, która jest uzyskiwana na końcu bloku</p> <p>THETA=AC(...) przełączenie pojedynczymi blokami na absolutne podawanie wymiarów</p> <p>THETA=IC(...) przełączenie pojedynczymi blokami na przyrostowe podawanie wymiarów</p> <p>THETA=Θ_e Interpolowanie kąta programowanego G90/G91</p> <p>PO[THT]=(..) Programowanie wielomianu max 5. stopnia</p> <p>Programowanie wektora obrotu</p> <ul style="list-style-type: none"> - ORIROTA obrót absolutny - ORIROTR względny wektor obrotu - ORIROTT styczny wektor obrotu
<p>Odniesiona do toru orientacja dla zmian orientacji w stosunku do toru albo obrót wektora obrotu stycznie do toru</p>	<p>Zmiany orientacji w stosunku do toru przy pomocy</p> <ul style="list-style-type: none"> - ORIPATH orientacja narzędzia w odniesieniu do toru - ORIPATHS dodatkowo w przypadku załamania w przebiegu orientacji <p>Programowanie wektora obrotu</p> <ul style="list-style-type: none"> - ORIROTC styczny wektor obrotu, obrót do stycznej do toru

6.2 Transformacje trzy-, cztero- i pięcioosiowe (TRAORI)

6.2.1 Zależności ogólne głowicy narzędziowej Kardana

Funkcja

Aby uzyskać optymalne warunki skrawania przy obrabianiu powierzchni przestrzennie zakrzywionych, kąt ustawienia narzędzia musi się dawać zmieniać.

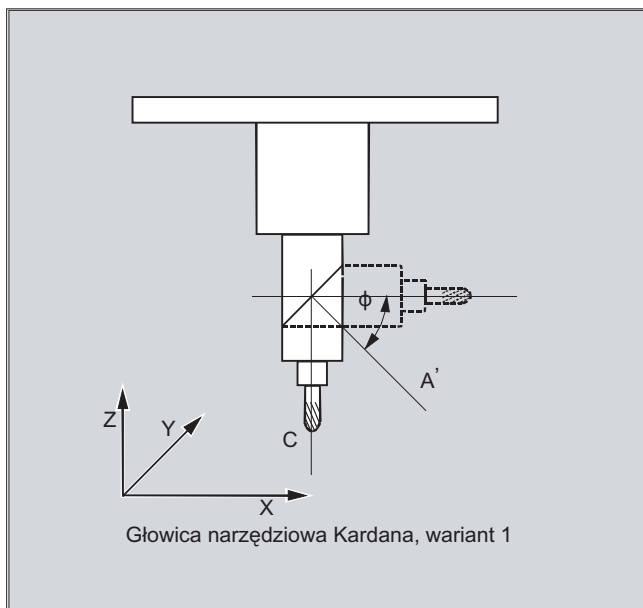


Przy pomocy jakiej konstrukcji maszyny jest to uzyskiwane, jest zapisane w danych osi.

Transformacja 5-osiowa

Głowica narzędziowa kardana

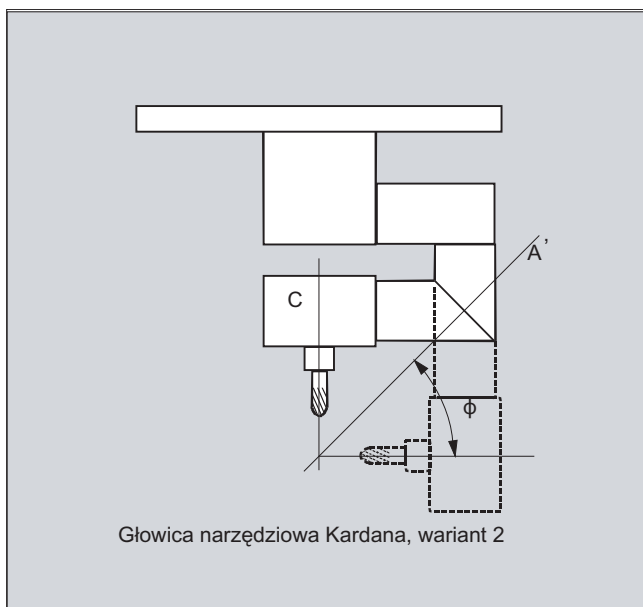
Tutaj trzy osie liniowe (X, Y, Z) i dwie osie orientacji ustalają kąt ustawienia i punkt roboczy narzędzia. Jedna z obydwu osi orientacji jest przyłożona jako oś skośna, tutaj w przykładzie A' - w wielu przypadkach jako układ 45°.



W pokazanych tutaj przykładach widać układy na przykładzie z głowicą narzędziową Kardana kinematyki maszyny CA!

Producent maszyny

Kolejność osi orientacji i kierunek orientacji narzędzia są zależnie od kinematyki maszyny ustawiane przez dane maszynowe.



W niniejszym przykładzie A' leży pod kątem ϕ względem osi X

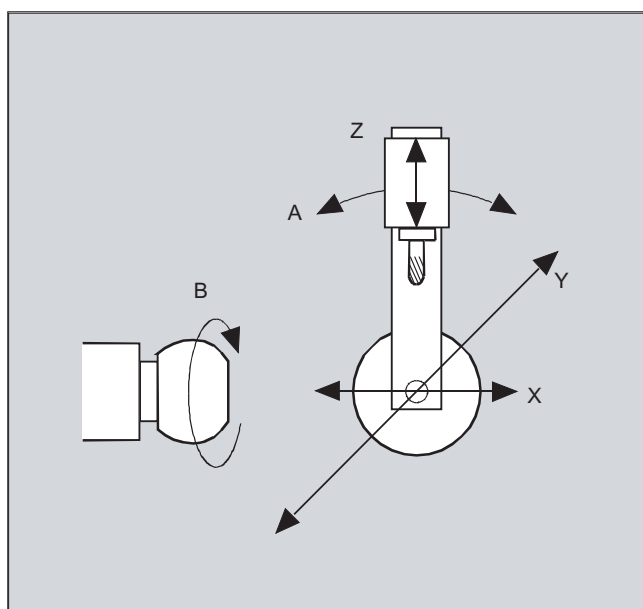
Ogólnie obowiązują następujące możliwe zależności:

A' leży pod kątem φ względem	osi X
B' leży pod kątem φ względem	osi Y
C' leży pod kątem φ względem	osi Z

Kąt φ można projektować w zakresie 0° do $+89^\circ$ przez dane maszynowe.

Ze skrzyżną osią liniową

Chodzi przy tym o układ z poruszającym obrabianym przedmiotem i poruszającym narzędziem. Kinematyka składa się z trzech osi liniowych (X, Y, Z) i dwóch prostopadle usytuowanych osi obrotowych. Pierwsza oś obrotowa jest poruszana np. przez sanie krzyżowe o dwóch osiach liniowych, narzędzie jest ustawione równoległe do trzeciej osi liniowej. Druga oś obrotu obraca obrabiany przedmiot. Trzecia oś liniowa (oś skrzyżna) leży w płaszczyźnie sani krzyżowych.



Kolejność osi rotacyjnych i kierunek orientacji narzędzia są zależnie od kinematyki maszyny ustawiane przez dane maszynowe.

Obowiązują następujące możliwe zależności:

Osie:	Kolejności osi:
1. Oś obrotowa	A A B B C C
2. Oś obrotowa	B C A C A B
Skrzyżona oś liniowa	Z Y Z X Y X

Dalsze objaśnienia dot. konfigurowalnych kolejności osi dla kierunku orientacji narzędzia patrz

Literatura: /FB3/ Podręcznik działania Funkcje specjalne; transformacja 3- do 5-osiowej (F2), punkt Głowica frezarska Kardana, "Parametryzacja".

6.2.2 Transformacje trzy-, cztero- i pięcioosiowe (TRAORI)

Funkcja

Użytkownik może projektować dwie wzgl. trzy osie ruchu postępowego i jedną oś obrotową. Transformacje zakładają, że oś obrotowa jest ustawiona ortogonalnie na płaszczyźnie orientacji.

Orientacja narzędzia jest możliwa tylko w płaszczyźnie, która jest prostopadła do osi obrotowej. Transformacja obsługuje typy maszyn o ruchomym narzędziu i ruchomym obrabianym przedmiocie.

Projektowanie i programowanie transformacji trzy- i czteroosiowych jest analogiczne do transformacji pięcioosiowych.

Literatura:

Podręcznik działania Funkcje specjalne; Transformacje wieloosiowe (F2)

Składnia

TRAORI (<n>)

TRAORI (<n>, <X>, <Y>, <Z>, <A>,)

TRAFOOF

Znaczenie

TRAORI:	Uaktywnia pierwszą uzgodnioną transformację orientacji
TRAORI (<n>):	Uaktywnia transformację orientacji uzgodnioną przy pomocy n
<n>:	Numer transformacji
	Wartość: 1 albo 2
	Przykład:
	TRAORI(1) uaktywnia transformację orientacji 1
<X>, <Y>, <Z>:	Składowa wektora orientacji, na którą wskazuje narzędzie.
<A>, :	Programowalny offset dla osi obrotowych
TRAFOOF:	Wyłączenie transformacji

Orientacja narzędzia

Zależnie od wybranego kierunku orientacji narzędzia aktywna płaszczyzna robocza (G17, G18, G19) musi być tak ustawiona w programie NC, by korekcja długości narzędzia działała w kierunku orientacji narzędzia.

Wskazówka

Po włączeniu transformacji dane dot. pozycji odnoszą się zawsze do wierzchołka narzędzia (X, Y, Z). Zmiany pozycji osi obrotowych uczestniczących w transformacji prowadzą do ruchów wyrównawczych pozostałych osi maszyny, dzięki czemu pozycja wierzchołka narzędzia pozostaje nie zmieniona.

Transformacja orientacji jest zawsze skierowana od wierzchołka narzędzia do jego uchwytu.

Offset dla osi orientacji

Przy uaktywnieniu transformacji orientacji można bezpośrednio zaprogramować dodatkowy offset dla osi orientacji.

Wolno pominąć parametry, gdy przy programowaniu zostanie zachowana prawidłowa kolejność.

Przykład:

TRAORI (, , , , A,B) ; gdy ma zostać wprowadzony tylko jeden jedyny offset

Alternatywnie do bezpośredniego programowania można ten dodatkowy offset dla osi orientacji również przejąć automatycznie z aktualnie aktywnego przesunięcia punktu zerowego. Przejęcie jest projektowane przez dane maszynowe.

Przykłady

TRAORI (1,0,0,1)	; Podstawowa orientacja narzędzia wskazuje w kierunku Z
TRAORI (1,0,1,0)	; Podstawowa orientacja narzędzia wskazuje w kierunku Z
TRAORI (1,0,1,1)	; Podstawowa orientacja narzędzia wskazuje w kierunku Y/Z (odpowiada położeniu -45°)

6.2.3 Warianty programowania orientacji i położenie podstawowe (OTIRESET)**Programowanie orientacji narzędzia przy TRAORI**

W połączeniu z programowaną transformacją orientacji TRAORI mogą dodatkowo do osi liniowych X, Y, Z być przez identyfikatory osi obrotowych A..., B..., C... programowane pozycje osi albo osie wirtualne z kątami albo komponentami wektorowymi. Dla osi orientacji i osi maszyny są możliwe różne rodzaje interpolacji. Niezależnie od tego, jakie wielomiany orientacji PO[kąt] i wielomiany osi PO[os] są właśnie aktywne, może być zaprogramowanych wiele różnych rodzajów wielomianów jak np. G1, G2, G3, CIP lub POLY.

Zmiana orientacji narzędzia może zostać zaprogramowana przez wektory orientacji. Przy tym orientacja końcowa każdego bloku może nastąpić albo przez programowanie bezpośrednie wektora albo przez programowanie pozycji osi obrotowych.

Wskazówka

Warianty programowania orientacji przy transformacji trzy- do pięcioosiowej

Przy transformacji trzy- do pięcioosiowej warianty

1. A, B, C bezpośrednie podanie pozycji osi maszyny
2. A2, B2, C2 programowanie kąta osi wirtualnych poprzez kąt Eulera albo kąt RPY
3. A3, B3, C3 podanie składowych wektora
4. LEAD, TILT podanie kąta wyprzedzenia i kąta bocznego w odniesieniu do toru i powierzchni
5. A4, B4, C4 i A5, B5, C5 wektor normalny do powierzchni na początku i na końcu bloku
6. A6, B6, C6 i A7, B7, C7 interpolacja wektora orientacji na poboczniczy stożka
7. A8, B8, C8 przeorientowanie narzędzia, kierunku i długości drogi ruchu wycofania

wzajemnie wykluczają się.

Mieszane programowanie wartości jest uniemożliwiane przez komunikaty alarmowe.

Położenie podstawowe orientacji narzędzia ORIRESET

Przez zaprogramowanie ORIRESET(A, B, C) ruchy w osiach orientacji są wykonywane liniowo i synchronicznie od ich aktualnej pozycji do podanej pozycji położenia podstawowego.

Jeżeli dla osi nie zostanie zaprogramowana pozycja położenia podstawowego, wówczas jest stosowana zdefiniowana pozycja z przynależnej danej maszynowej

\$MC_TRAFO5_ROT_AX_OFFSET_1/2. Ewentualnie aktywne frame osi obrotowych nie są przy tym uwzględniane.

Wskazówka

Tylko gdy jest aktywna transformacja orientacji z TRAORI(...), może zostać zaprogramowane położenie podstawowe orientacji narzędzia niezależnie od kinematyki przy pomocy ORIRESET(...) bez alarmu 14101.

Przykłady

```

1. Przykład dla kinematyki maszyny CA (nazwy osi maszyny C, A)
ORIRESET(90, 45)           ;C na 90 stopni, A na 45 stopni
ORIRESET(, 30)            ;C na $MC_TRAFO5_ROT_AX_OFFSET_1/2[0], A na 30 stopni
ORIRESET( )               ;C na $MC_TRAFO5_ROT_AX_OFFSET_1/2[0],
                           ;A na $MC_TRAFO5_ROT_AX_OFFSET_1/2[1]

2. Przykład dla kinematyki maszyny CAC (nazwy osi kanału C, A, B)
ORIRESET(90, 45, 90)      ;C na 90 stopni, A na 45 stopni, B na 90 stopni
ORIRESET( )               ;C na $MC_TRAFO5_ROT_AX_OFFSET_1/2[0],
                           ;A na $MC_TRAFO5_ROT_AX_OFFSET_1/2[1],
                           ;B na $MC_TRAFO5_ROT_AX_OFFSET_1/2[2]
    
```

Programowanie obrotów LEAD, TILT i THETA

Obroty orientacji narzędzia w przypadku transformacji trzy- do pięcioosiowej są programowane przy pomocy kątów wyprzedzenia LEAD i kątów bocznych TILT.

Przy transformacji z trzecią osią obrotową są zarówno dla orientacji z komponentami wektorowymi, jak również z podaniem kątów LEAD, TILT dozwolone dodatkowe zaprogramowania C2 (obroty wektora orientacji).

Z dodatkową trzecią osią obrotową może zostać zaprogramowany obrót narzędzia wokół siebie samego z kątem obrotu THETA.

6.2.4 Programowanie orientacji narzędzia (A..., B..., C..., LEAD, TILT)

Funkcja

Są następujące możliwości programowania orientacji narzędzia:

1. Bezpośrednie programowanie ruchu osi obrotowych. Zmiana orientacji następuje zawsze w bazowym układzie współrzędnych wzgl. układzie współrzędnych maszyny. Osie orientacji wykonują zawsze ruch, jako osie synchroniczne.
2. Programowanie w kątach Eulera albo RPY według definicji kąta przez A_2 , B_2 , C_2 .
3. Programowanie wektora biegunowego przez A_3 , B_3 , C_3 . Wektor biegunowy jest skierowany od wierzchołka narzędzia w kierunku uchwytu.
4. Programowanie wektora normalnej powierzchni na początku bloku przy pomocy A_4 , B_4 , C_4 i na końcu bloku przy pomocy A_5 , B_5 , C_5 (frezowanie czołowe).
5. Programowanie przez kąt wyprzedzenia LEAD i kąt boczny TILT
6. Programowanie osi obrotu stożka jako wektor znormalizowany przez A_6 , B_6 , C_6 , albo orientacji pośredniej na pobocznicę stożka przez A_7 , B_7 , C_7 , patrz punkt "Programowanie orientacji wzdłuż pobocznicę stożka (ORIPLANE, ORICONxx)".
7. Programowanie przeorientowania, kierunku i długości drogi narzędzia podczas ruchu cofnięcia przez A_8 , B_8 , C_8 , patrz punkt "Wyglądanie przebiegu orientacji" (ORIPATHS A8=, B8=, C8=)

Wskazówka

We wszystkich przypadkach programowanie orientacji jest dopuszczalne tylko wtedy, gdy transformacja orientacji jest włączona.

Zaletą: Te programy dają się przenosić na każdą kinematykę maszyny.

Definicja orientacji narzędzia przez G-Code

Wskazówka

Producent maszyny

Przez daną maszynową można przełączać między kątami Eulera i kątami RPY. Przy odpowiednich ustawieniach danych maszynowych jest możliwe przełączanie zarówno zależnie jak i niezależnie od aktywnego G-Code grupy 50. Są do dyspozycji następujące możliwości wyboru:

1. Gdy obydwie dane maszynowe do definicji osi orientacji i definicji kąta orientacji przez G-Code są ustawione na zero.
Kąty zaprogramowane przy pomocy A_2 , B_2 , C_2 są **w zależności od danej maszynowej** definicja kąta zaprogramowania orientacji interpretowane, albo jako kąty Eulera albo kąty RPY.
2. Gdy dana maszynowa do definicji osi orientacji przez G-Code jest ustawiona na jeden, przełączenie następuje **zależnie** od aktywnego G-Code grupy 50:
Kąty zaprogramowane przy pomocy A_2 , B_2 , C_2 są interpretowane według jednego z aktywnych G-Code $ORIEULER$, $ORIRPY$, $ORIVIRT1$, $ORIVIRT2$, $ORIXPOS$ i $ORIPY2$. Wartości zaprogramowane z osiami orientacji są interpretowane odpowiednio do aktywnego G-Code grupy 50 również jako kąt orientacji.
3. Gdy dana maszynowa dla definicji kątów orientacji przez G-Code jest ustawiona na jeden, a dana maszynowa dla definicji osi orientacji przez G-Code jest ustawiona na zero, przełączenie następuje **niezależnie** od aktywnego G-Code grupy 50:
Kąty zaprogramowane przy pomocy A_2 , B_2 , C_2 są interpretowane według jednego z aktywnych G-Code $ORIEULER$, $ORIRPY$, $ORIVIRT1$, $ORIVIRT2$, $ORIXPOS$ i $ORIPY2$. Wartości zaprogramowane z osiami orientacji są niezależnie od aktywnego G-Code grupy 50 zawsze interpretowane jako pozycje osi obrotowych.

Programowanie

G1 X Y Z A B C

G1 X Y Z A2= B2= C2=

G1 X Y Z A3== B3== C3==

G1 X Y Z A4== B4== C4==

G1 X Y Z A5== B5== C5==

LEAD=

TILT=

Programowanie ruchu osi obrotowych

Programowanie w kątach Eulera

Programowanie wektora biegunowego

Programowanie wektora normalnej powierzchni na początku bloku

Programowanie wektora normalnej powierzchni na końcu bloku

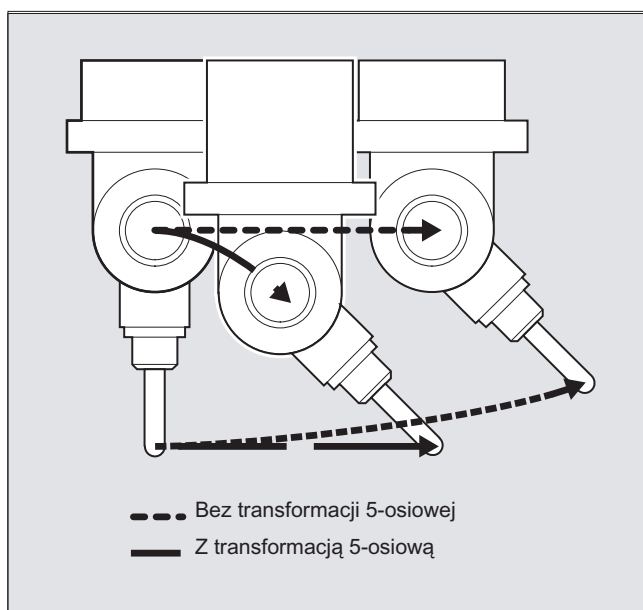
Kąt wyprzedzenia do programowania orientacji narzędzia

Kąt w kierunku bocznym do programowania orientacji narzędzia

Parametry

G...	Podanie rodzaju ruchu osi obrotowych
X Y Z	Podanie osi liniowych
A B C	Podanie pozycji osi obrotowych maszyny
A2 B2 C2	Programowanie kąta (kąt Eulera albo RPY) osi wirtualnych wzgl. osi orientacji
A3 B3 C3	Podanie składowych wektora biegunowego
A4 B4 C4	Podanie, np. przy frezowaniu czołowym, składowej wektora normalnej powierzchni na początku bloku
A5 B5 C5	Podanie, np. przy frezowaniu czołowym, składowej wektora normalnej powierzchni na końcu bloku
LEAD	Kąt w stosunku do wektora normalnej powierzchni, w płaszczyźnie wyznaczonej przez styczną do toru i wektor normalnej powierzchni
TILT	Kąt w płaszczyźnie, prostopadle do stycznej do toru w stosunku do wektora normalnej powierzchni

Przykład, zestawienie bez i z transformacją 5-osiową



Dalsze informacje

Z reguły programy 5-osiowe są wytwarzane przez systemy CAD/CAM, a nie wprowadzane na sterowaniu. Dlatego poniższe objaśnienia są skierowane głównie do programistów postprocesorów.

Rodzaj programowania orientacji jest ustalany w G-Code grupa 50:

Funkcja G	Programowanie orientacji
ORIEULER	poprzez kąt Eulera
ORIRPY	poprzez kąt RPY (kolejność obrotów ZYX)
ORIVIRT1	poprzez wirtualne osie orientacji (definicja 1)
ORIVIRT2	poprzez wirtualne osie orientacji (definicja 2)
ORIXPOS	poprzez wirtualne osie orientacji z pozycjami osi obrotowych
ORIPY2	poprzez kąt RPY (kolejność obrotów XYZ)

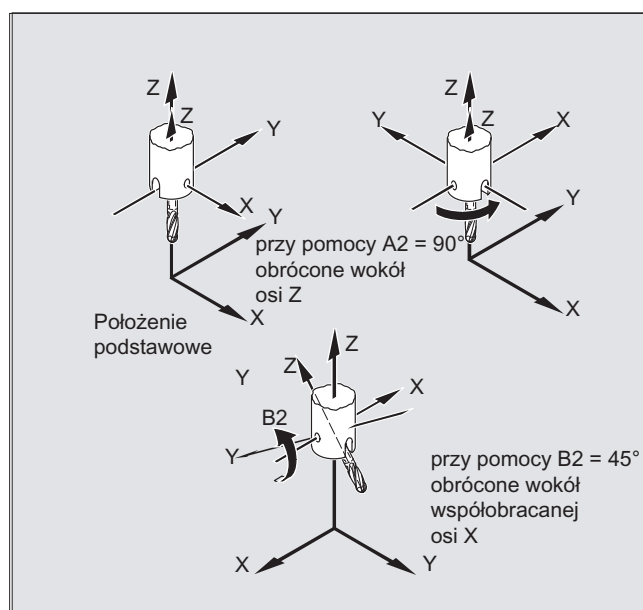
Wskazówka

Przez dane maszynowe producent maszyny może definiować różne warianty. Proszę przestrzegać danych producenta maszyny.

Programowanie w kątach Eulera ORIEULER

Wartości programowane przy programowaniu orientacji przy pomocy A_2 , B_2 , C_2 są interpretowane jako kąty Eulera (w stopniach).

Wektor orientacji uzyskuje się, gdy wektor w kierunku Z zostanie obrócony najpierw przy pomocy A_2 wokół osi Z, następnie przy pomocy B_2 wokół nowej osi X, a na końcu przy pomocy C_2 wokół nowej osi Z.



W tym przypadku wartość C_2 (obrót wokół nowej osi Z) jest bez znaczenia i nie musi być programowana.

Programowanie w kątach RPY ORIRPY

Wartości programowane przy programowaniu orientacji przy pomocy A_2 , B_2 , C_2 są interpretowane jako kąty RPY (w stopniach).

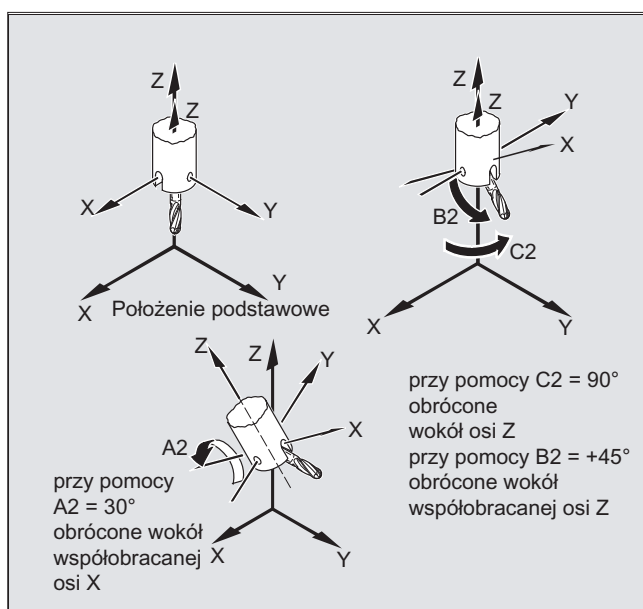
Wskazówka

W przeciwieństwie do programowania w kątach Eulera tutaj wszystkie trzy wartości mają wpływ na wektor orientacji.

Przy definicji kąta z kątem orientacji przez kąt RPY obowiązuje dla osi orientacji:

$\$MC_ORI_DEF_WITH_G_CODE = 0$

Wektor orientacji uzyskuje się, gdy wektor w kierunku Z zostanie obrócony najpierw przy pomocy C_2 wokół osi Z, następnie przy pomocy B_2 wokół nowej osi Y, a na końcu przy pomocy A_2 wokół nowej osi X.



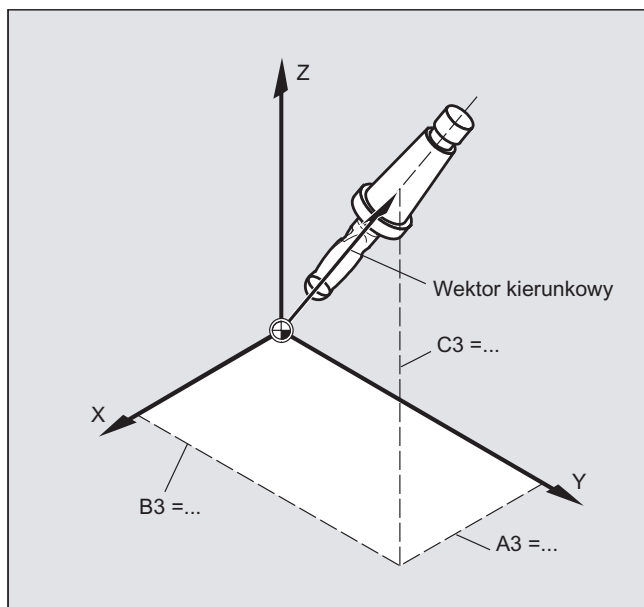
Jeżeli dana maszyna dla definicji osi orientacji przez G-Code jest $\$MC_ORI_DEF_WITH_G_CODE = 1$, wówczas obowiązuje:

Wektor orientacji wynika przez obrót wektora w kierunku Z najpierw o A_2 wokół osi Z, następnie o B_2 wokół nowej osi Y, a na końcu o C_2 wokół nowej osi X.

Programowanie wektora biegunowego

Składowe wektora biegunowego są programowane przy pomocy A_3 , B_3 , C_3 . Wektor jest skierowany w kierunku uchwytu narzędzia; długość wektora jest przy tym bez znaczenia.

Nie zaprogramowane składowe wektora są ustawiane na zero.



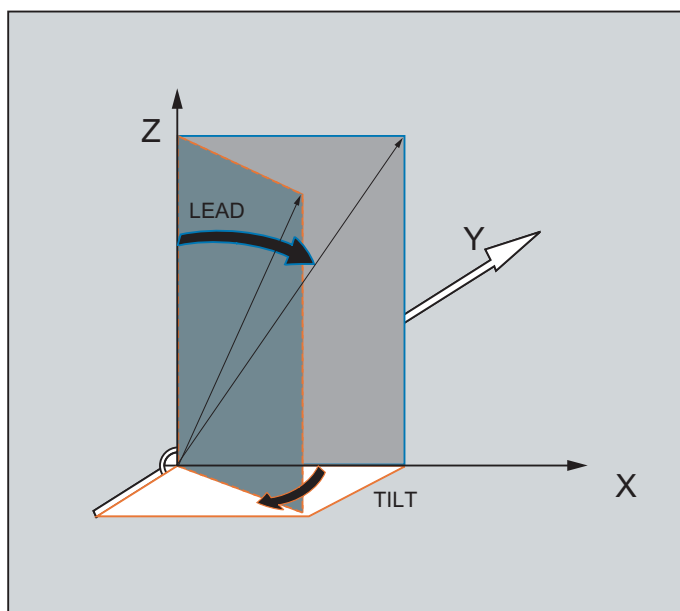
Programowanie orientacji narzędzia przy pomocy LEAD= i TILT=

Wynikająca orientacja narzędzia jest obliczana z:

- Styczna do toru
- Wektor normalnej powierzchni na początku bloku A_4 , B_4 , C_4 i na końcu bloku A_5 , B_6 , C_5
- Kąt wyprzedzenia $LEAD$ w płaszczyźnie wyznaczonej przez styczną do toru i wektor normalnej powierzchni
- Kąt boczny $TILT$ na końcu bloku prostopadłe do stycznej do toru i względem wektora normalnej powierzchni

Zachowanie się na narożach wewnętrznych (przy korekcji narzędzia 3D):

Gdy blok jest skracany na narożu wewnętrznym, wynikająca orientacja narzędzia jest uzyskiwana również na końcu bloku.

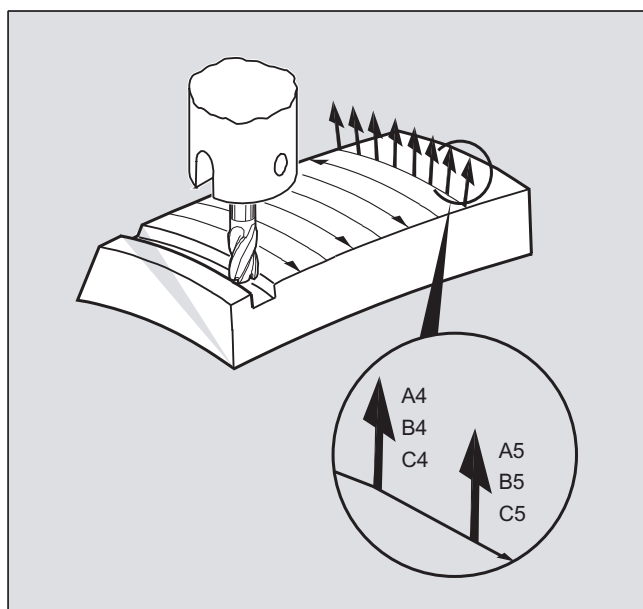


Rysunek 6-1 Definicja orientacji narzędzia przy pomocy LEAD= i TILT=

6.2.5 Frezowanie czołowe (A4, B4, C4, A5, B5, C5)

Funkcja

Frezowanie czołowe służy do obróbki powierzchni dowolnie zakrzywionych.



Dla tego rodzaju frezowania trójwymiarowego potrzeba wierszowego opisu torów 3D na powierzchni obrabianego przedmiotu.

Obliczenia są przeprowadzane przy uwzględnieniu kształtu narzędzia i jego wymiarów - zazwyczaj w CAM. Obliczone na gotowo bloki NC są następnie przez postprocesory wczytywane do sterowania.

Programowanie zakrzywienia toru

Opis powierzchni

Opis zakrzywienia toru następuje przez wektory normalnych powierzchni przy pomocy następujących składowych:

A4, B4, C4 wektor startowy na początku bloku

A5, B5, C5 wektor końcowy na końcu bloku

Jeżeli w bloku jest podany tylko wektor startowy, wektor normalnej powierzchni pozostaje stały przez cały blok. Gdy w bloku jest podany tylko wektor końcowy, wówczas interpolacja odbywa się od wartości końcowej poprzedniego bloku przez interpolację dużego okręgu do zaprogramowanej wartości końcowej.

Gdy jest zaprogramowany wektor startowy i końcowy, wówczas między obydwoma kierunkami interpolacja następuje również przez interpolację dużego okręgu. Przez to dają się wytwarzać w sposób ciągły gładkie drogi po torze.

W położeniu podstawowym wektory normalnej powierzchni są niezależnie od aktywnej płaszczyzny G17 do G19 skierowane w kierunku Z.

Długość wektora jest bez znaczenia.

Nie zaprogramowane składowe wektora są ustawiane na zero.

Przy aktywnym ORIWKS (patrz "Odniesienie osi orientacji (ORIWKS, ORIMKS) (Strona 327)") wektory normalne do powierzchni odnoszą się do aktywnego Frame i przy obrocie Frame są równocześnie obracane.

Producent maszyny

Wektor normalnej powierzchni musi być prostopadły do stycznej do toru w ramach wartości granicznej ustawionej przez daną maszynową, w przeciwnym przypadku jest wyprowadzany alarm.

6.2.6 Odniesienie osi orientacji (ORIWKS, ORIMKS)

Funkcja

Przy programowaniu orientacji w układzie współrzędnym obrabianego przedmiotu przez

- kąt Eulera wzgl. kąt RPY albo
- wektor orientacji

można ustawić przebieg ruchu obrotowego przez `ORIMKS/ORIWKS`.

Wskazówka

Producent maszyny

Rodzaj interpolacji orientacji jest ustalany daną maszynową:

MD21104 \$MC_ORI_IPO_WITH_G_CODE

= FALSE: Odniesieniem są funkcje G ORIWKS i ORIMKS

= TRUE: Odniesieniem są funkcje G 51. grupy (ORIAxes, ORIVect, ORIPLANE, ...)

Składnia

`ORIMKS=...`

`ORIWKS=...`

Znaczenie

`ORIMKS` Obrót w układzie współrzędnych maszyny

`ORIWKS` Obrót w układzie współrzędnych obrabianego przedmiotu

Wskazówka

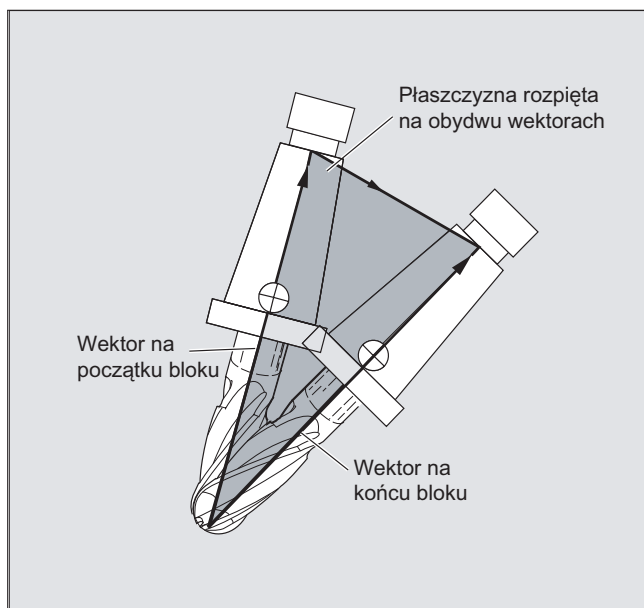
`ORIWKS` jest ustawieniem podstawowym. Jeżeli w przypadku programu 5-osiowego nie jest z góry jasne, na której maszynie ma on być wykonywany, należy z zasady wybrać `ORIWKS`. Jakie ruchy maszyna rzeczywiście wykonuje, zależy od jej kinematyki.

Przy pomocy `ORIMKS` można programować rzeczywiste ruchy maszyny, np. aby uniknąć kolizji z przyrządami itp.

Opis

W przypadku **ORIMKS** wykonywany ruch narzędzia jest **zależny** od kinematyki maszyny. Przy zmianie orientacji z wierzchołkiem narzędzia stałym w przestrzeni następuje interpolacja liniowa między pozycjami osi obrotowej.

W przypadku **ORIWKS** wykonywany ruch narzędzia jest **niezależny** od kinematyki maszyny. Przy zmianie orientacji z wierzchołkiem narzędzia stałym w przestrzeni narzędzie porusza się w płaszczyźnie wyznaczonej przez wektor początkowy i wektor końcowy.



Położenia osobliwe

Wskazówka

ORIWKS

Ruchy orientacji w zakresie położenia osobliwego maszyny 5-osiowej wymagają dużych ruchów w osiach maszyny. (Na przykład w przypadku skrętnej głowicy tokarskiej z C, jako osią obrotu i A jako osią skrętu wszystkie położenia z A=0 są położeniami osobliwymi.)

Producent maszyny

Aby nie przeciążyć osi maszyny, układ prowadzenia prędkości bardzo zmniejsza prędkość po torze w pobliżu miejsc osobliwych.

Przy pomocy

```
$MC_TRAFO5_NON_POLE_LIMIT
```

```
$MC_TRAFO5_POLE_LIMIT
```

transformacja może zostać tak sparametryzowana, by ruchy orientacji w pobliżu bieguna były prowadzone przez biegun i by była możliwa nieprzerwana obróbka.

Miejsca szczególne są traktowane tylko przy pomocy MD \$MC_TRAFO5_POLE_LIMIT.

Literatura:

/FB3/ Podręcznik działania Funkcje specjalne; transformacja 3- do 5-osiowej (F2), punkt "Miejsca osobliwe i ich traktowanie".

6.2.7 Programowanie osi orientacji (ORIXES, ORIVECT, ORIEULER, ORIRPY, ORIRPY2, ORIVIRT1, ORIVIRT2)

Funkcja

Funkcja "Osie orientacji" opisuje orientację narzędzia w przestrzeni i jest osiągalna przez zaprogramowanie offsetu dla osi obrotowych. Dalszy trzeci stopień swobody można osiągnąć przez dodatkowy obrót narzędzia wokół siebie samego. Ta orientacja narzędzia następuje dowolnie w przestrzeni przez trzecią oś obrotową i wymaga transformacji sześciooosiowej. Obrót narzędzia wokół siebie jest ustalany zależnie od rodzaju interpolacji wektorów obrotu z kątem obrotu THETA (patrz "Obroty orientacji narzędzia (ORIROTA, ORIROTR, ORIROTT, ORIROTC, THETA) (Strona 339)").

Osie orientacji są programowane przez identyfikatory osi A2, B2, C2.

Składnia

N... ORIXES/ORIVECT	; interpolacja liniowa lub interpolacja dużego okręgu
N... G1 X Y Z A B C	
N... ORIPLANE	; interpolacja orientacji płaszczyzny
N... ORIEULER/ORIRPY/ORIRPY2	; kąt orientacji, kąt Eulera/RPY
N... G1 X Y Z A2= B2= C2=	; programowanie kąta osi wirtualnych
N... ORIVIRT1/ORIVIRT2	; definicja wirtualnych osi orientacji 1/2
N... G1 X Y Z A3= B3= C3=	; programowanie wektora kierunku

Wskazówka

Dla zmian orientacji wzdłuż znajdującej się w przestrzeni poboczniczy stożka mogą być programowane dalsze offsety obrotowej osi orientacji, patrz "Programowanie orientacji wzdłuż poboczniczy stożka (ORIPLANE, ORICONCW, ORICONCCW, ORICONTO, ORICONIO) (Strona 332)".

Znaczenie

ORIXES:	Interpolacja liniowa osi maszyny albo osi orientacji
ORIVECT:	Interpolacja dużego okręgu (identyczna z ORIPLANE)
ORIMKS:	Obrót w układzie współrzędnych maszyny
ORIWKS:	Obrót w układzie współrzędnych obrabianego przedmiotu Opis patrz "Odniesienie osi orientacji (ORIWKS, ORIMKS) (Strona 327)".
A= B= C=:	Programowanie pozycji osi maszyny
ORIEULER:	Programowanie orientacji przez kąt Eulera
ORIRPY:	Programowanie orientacji przez kąt RPY Kolejnością obrotów jest XYZ, przy czym obowiązuje: <ul style="list-style-type: none"> • A2 jest kątem obrotu wokół X • B2 jest kątem obrotu wokół Y • C2 jest kątem obrotu wokół Z
ORIRPY2:	Programowanie orientacji przez kąt RPY Kolejnością obrotów jest ZYX, przy czym obowiązuje: <ul style="list-style-type: none"> • A2 jest kątem obrotu wokół Z • B2 jest kątem obrotu wokół Y • C2 jest kątem obrotu wokół X
A2= B2= C2=:	Programowanie kąta osi wirtualnych
ORIVIRT1/ORIVIRT2:	Programowanie orientacji przez wirtualne osie orientacji Definicja 1: Ustalenie według MD21120 \$MC_ORIAX_TURN_TAB_1 Definicja 2: Ustalenie według MD21130 \$MC_ORIAX_TURN_TAB_2
A3= B3= C3=:	Programowanie wektora biegunowego osi kierunkowej

Opis

Producent maszyny

Przy pomocy MD21102 \$MC_ORI_DEF_WITH_G_CODE ustala się, jak są definiowane programowane kąty A2, B2, C2:

Definicja następuje według MD21100 \$MC_ORIENTATION_IS_EULER (standard) albo według grupy G 50 (ORIEULER, ORIRPY, ORIVIRT1, ORIVIRT2).

Przy pomocy MD21104 \$MC_ORI_IPO_WITH_G_CODE następuje ustalenie, który rodzaj interpolacji działa: ORIWKS/ORIMKS czy ORIXES/ORIVECT.

Tryb pracy JOG

Kąty orientacji są w tym trybie pracy zawsze interpolowane liniowo. Przy ruchu ciągłym i przyrostowym przez przyciski ruchu może wykonywać ruch tylko jedna oś orientacji. Przez kółka ręczne osie orientacji mogą wykonywać ruch równocześnie.

Dla ruchu ręcznego osi orientacji działa specyficzny dla kanału przełącznik korekcyjny posuwu wzgl. przełącznik korekcyjny posuwu szybkiego przy nałożeniu tego posuwu.

Przy pomocy następujących danych maszynowych jest możliwe oddzielne zadanie prędkości

MD21160 \$MC_JOG_VELO_RAPID_GEO

MD21165 \$MC_JOG_VELO_GEO

MD21150 \$MC_JOG_VELO_RAPID_ORI

MD21155 \$MC_JOG_VELO_ORI

Wskazówka**SINUMERIK 840D sl z "Pakiem transformacji Handling"**

Przy pomocy funkcji "Kartezjański ruch ręczny" można w trybie JOG, niezależnie od siebie ustawić przesunięcie osi geometrycznych w systemach odniesienia MKS, WKS i TKS.

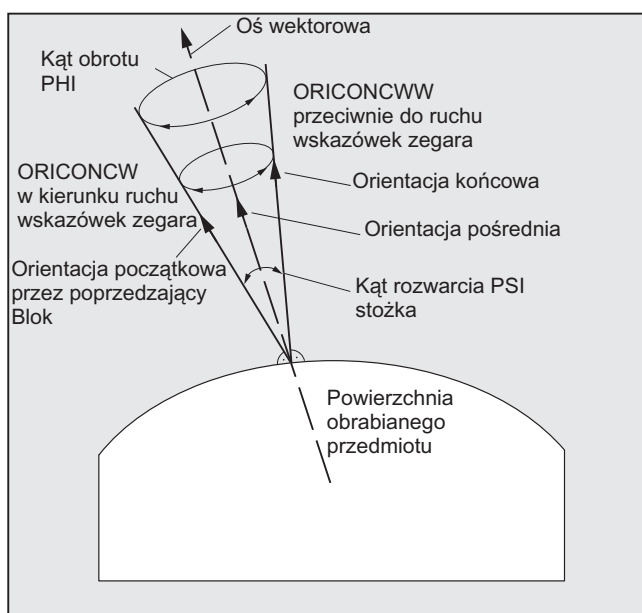
Literatura:

Podręcznik działania Funkcje rozszerzające; Transformacja kinematyczna (M1)

6.2.8 Programowanie orientacji wzdłuż pobocznic stożka (ORIPLANE, ORICONCW, ORICONCCW, ORICONTO, ORICONIO)

Funkcja

Przy pomocy orientacji rozszerzonej jest możliwe wykonywanie zmian orientacji wzdłuż powierzchni pobocznicowej stożka znajdującej się w przestrzeni. Interpolacja wektora orientacji na pobocznicę stożka następuje przy pomocy poleceń modalnych ORICONxx. Dla interpolacji w płaszczyźnie można programować orientację końcową przy pomocy ORIPLANE. Generalnie orientacja startowa jest ustalana przez poprzedzające bloki.



Programowanie

Orientacja końcowa jest ustalana albo przez podanie zaprogramowania kąta jako kąt Eulera lub kąt RPY przy pomocy A2, B2, C2, albo przez zaprogramowanie pozycji osi obrotowych przy pomocy A, B, C. Dla osi orientacji wzdłuż pobocznic stożka są wymagane dalsze dane programowe:

- Oś obrotu stożka jako wektor z A_6, B_6, C_6
- Kąt rozwarcia Ψ z identyfikatorem NUT
- Orientacja pośrednia w pobocznicę stożka z A_7, B_7, C_7

Wskazówka**Programowanie wektora biegunowego A6, B6, C6 dla osi obrotu stożka**

Zaprogramowanie orientacji końcowej nie jest bezwarunkowo wymagane. Gdy orientacja końcowa nie jest podana, wówczas jest interpolowana pełna pobocznica stożka 360 stopni.

Programowanie kąta rozwarcia stożka z NUT=kąt

Podanie orientacji końcowej jest bezwzględnie wymagane.

Kompletna pobocznica stożka 360 stopni nie może być w ten sposób interpolowana.

Programowanie orientacji pośredniej A7, B7, C7 w pobocznicy stożka

Podanie orientacji końcowej jest bezwzględnie wymagane. Zmiana orientacji i kierunek obrotu są jednoznacznie ustalane przez trzy wektory orientacji startowej, końcowej i pośredniej. Wszystkie trzy wektory muszą przy tym być różne. Jeżeli zaprogramowana orientacja pośrednia jest równoległa do orientacji startowej albo końcowej, jest przeprowadzana liniowa interpolacja dużego okręgu orientacji w płaszczyźnie, która jest wyznaczana przez wektor startowy i końcowy

Rozszerzona interpolacja orientacji na pobocznicy stożka

```
N... ORICONCW lub ORICONCCW
N... A6= B6= C6= A3= B3= C3=
lub
N... ORICONT0
N... G1 X Y Z A6= B6= C6=
lub
N... ORICONIO
N... G1 X Y Z A7= B7= C7=
N... PO[PHI]=(a2, a3, a4, a5)
N... PO[PSI]=(b2, b3, b4, b5)
```

Interpolacja na pobocznicy stożka z wektorem biegunowym stożka w kierunku / przeciwie do ruchu wskazówek zegara i orientacją końcową albo przejściem stycznym i podaniem orientacji końcowej albo podaniem orientacji końcowej i orientacji pośredniej na pobocznicy stożka z wielomianami kąta obrotu i wielomianami kąta rozwarcia

Parametry

ORIPLANE	Interpolacja na płaszczyźnie(Interpolacja dużego okręgu)
ORICONCW	Interpolacja na pobocznicy stożka w kierunku ruchu wskazówek zegara
ORICONCCW	Interpolacja na pobocznicy stożka przeciwie do ruchu wskazówek zegara
ORICONT0	Interpolacja na pobocznicy stożka przejście styczne
A6= B6= C6=	Programowanie osi obrotu stożka (wektor znormalizowany)
NUT=kąt	Kąt rozwarcia stożka w stopniach

NUT=+179	Kąt ruchu mniejszy albo równy 180 stopni
NUT=-181	Kąt ruchu większy albo równy 180 stopni
ORICONIO	Interpolacja na poboczniczy stożka
A7= B7= C7=	Orientacja pośrednia (programowanie jako wektor znormalizowany)
PHI	Kąt obrotu orientacji wokół osi kierunkowej stożka
PSI	Kąt rozwarcia stożka
Możliwe wielomiany PO[PHI]=(a2, a3, a4, a5) PO[PSI]=(b2, b3, b4, b5)	Oprócz poszczególnych kątów można programować również wielomiany maksymalnie 5. stopnia

Przykład: Różne zmiany orientacji

Kod programu	Komentarz
...	
N10 G1 X0 Y0 F5000	
N20 TRAORI(1)	; Transformacja orientacji wł.
N30 ORIVECT	; Interpolowanie orientacji narzędzia jako wektor.
...	; Orientacja narzędzia na płaszczyźnie.
N40 ORIPLANE	; Wybór interpolacji dużego okręgu.
N50 A3=0 B3=0 C3=1	
N60 A3=0 B3=1 C3=1	; Orientacja na płaszczyźnie Y/Z obrócona o 45 stopni, na końcu bloku jest osiągnięta orientacja (0,1/√2,1/√2).
...	
N70 ORICONCW	; Programowanie orientacji na poboczniczy stożka:
N80 A6=0 B6=0 C6=1 A3=0 B3=0 C3=1	; Kąt orientacji jest interpolowany na poboczniczy stożka z kierunkiem (0,0,1), aż do orientacji (1/√2,0,1/√2) w kierunku ruchu wskazówek zegara, kąt obrotu wynosi przy tym 270 stopni
N90 A6=0 B6=0 C6=1	; Orientacja narzędzia przechodzi pełny obrót na tej samej poboczniczy stożka.

Opis

Jeżeli zmiany orientacji mają zostać opisane na poboczniczy stożka leżącej dowolnie w przestrzeni, wówczas musi być znany wektor, wokół którego orientacja narzędzia ma wykonać obrót. Poza tym musi zostać zadana orientacja startowa i orientacja końcowa. Orientacja startowa wynika z bloku poprzedzającego, a orientacja końcowa musi albo zostać zaprogramowana albo ustalona przez inne warunki.

Programowanie w płaszczyźnie ORIPLANE odpowiada ORIVECT

Programowanie interpolacji wielkiego okręgu razem z wielomianami kąta odpowiada interpolacji liniowej i wielomianowej konturów. Orientacja narzędzia jest interpolowana w płaszczyźnie, która jest wyznaczona przez orientację startową i końcową. Gdy zostaną dodatkowo zaprogramowane wielomiany, wówczas wektor orientacji może ulec odchyleniu od płaszczyzny.

Programowanie okręgów w płaszczyźnie G2/G3, CIP i CT

Orientacja rozszerzona odpowiada interpolacji okręgów na płaszczyźnie. Odnosnie odpowiednich możliwości programowania okręgów z podaniem punktu środkowego albo promienia jak G2/G3, okręgu przez punkt pośredni CIP i okręgów stycznych CT patrz

Literatura: Podręcznik programowania Podstawy, "Programowanie poleceń dot. drogi".

Programowanie orientacji**Interpolacja wektora orientacji na pobocznicy stożka ORICONxx**

Dla interpolacji orientacji na pobocznicy stożka mogą być wybierane cztery różne rodzaje interpolacji z grupy G-Code 51:

1. Interpolacja na pobocznicy stożka w kierunku ruchu wskazówek zegara `ORICONCW` z podaniem orientacji końcowej i kierunku stożka albo kąta rozwarcia. Wektor biegunowy jest programowany przy pomocy identyfikatorów `A6`, `B6`, `C6`, a kąt rozwarcia stożka przy pomocy identyfikatora `NUT=` zakres wartości w przedziale 0 do 180 stopni.
2. Interpolacja na pobocznicy stożka przeciwnie do ruchu wskazówek zegara `ORICONCWW` z podaniem orientacji końcowej i kierunku stożka lub kąta rozwarcia. Wektor biegunowy jest programowany przy pomocy identyfikatorów `A6`, `B6`, `C6`, a kąt rozwarcia stożka przy pomocy identyfikatora `NUT=` zakres wartości w przedziale 0 do 180 stopni.
3. Interpolacja na pobocznicy stożka `ORICONIO` z podaniem orientacji końcowej i orientacji pośredniej, która jest programowana przy pomocy identyfikatorów `A7`, `B7`, `C7`.
4. Interpolacja na pobocznicy stożka `ORICONTO` z przejściem stycznym i podaniem orientacji końcowej. Wektor biegunowy jest programowany przy pomocy identyfikatorów `A6`, `B6`, `C6`.

6.2.9 Zadanie orientacji dwóch punktów kontaktowych (ORICURVE, PO[XH]=, PO[YH]=, PO[ZH]=)**Funkcja****Programowanie zmiany orientacji przez drugą krzywą w przestrzeni ORICURVE**

Kolejna możliwość programowania zmian orientacji polega na tym, by oprócz wierzchołka narzędzia wzdłuż krzywej w przestrzeni, programować również ruch drugiego punktu kontaktowego narzędzia przy pomocy `ORICURVE`. Przez to mogą zostać jednoznacznie ustalone zmiany orientacji narzędzia, jak przy programowaniu samego wektora narzędzia.

Producent maszyny

Proszę przestrzegać wskazówek producenta maszyny dot. identyfikatorów osi ustawianych przez daną maszynową odnośnie programowania 2. toru orientacji narzędzia.

Programowanie

Przy tym rodzaju interpolacji mogą dla obydwu krzywych w przestrzeni być programowane punkty przy pomocy `G1` wzgl. wielomiany przy pomocy `POLY`. Okręgi i ewolwenty są niedopuszczalne. Dodatkowo może zostać uaktywniona interpolacja spline przy pomocy `BSPLINE` i funkcja "Połączenie krótkich bloków Spline".

Literatura:

Podręcznik działania Funkcje podstawowe; Przechodzenie płynne, Zatrzymanie dokładne, LookAhead (B1), Rozdział: Połączenie krótkich bloków Spline

Inne rodzaje spline `ASPLINE` i `CSPLINE`, jak też uaktywnienie kompresora przy pomocy `COMPON`, `COMPCURV` lub `COMPCAD` są niedopuszczalne.

Ruch dwóch punktów kontaktowych narzędzia można zadać przy programowaniu wielomianów orientacji dla współrzędnych do maksymalnie 5. stopnia.

Rozszerzona interpolacja orientacji z dodatkową krzywą w przestrzeni i wielomiany dla współrzędnych

N... ORICURVE
 N... PO[XH]=(xe, x2, x3, x4, x5)
 N... PO[YH]=(ye, y2, y3, y4, y5)
 N... PO[ZH]=(ze, z2, z3, z4, z5)

Podanie ruchu drugiego punktu kontaktowego narzędzia i dodatkowe wielomiany każdorazowych współrzędnych

Parametry

ORICURVE	Interpolacja orientacji z zadaniem ruchu dwóch punktów kontaktowych narzędzia.
XH YH ZH	Identyfikator współrzędnych drugiego punktu kontaktowego narzędzia dodatkowego konturu jako krzywa w przestrzeni
Możliwe wielomiany PO[XH]=(xe, x2, x3, x4, x5) PO[YH]=(ye, y2, y3, y4, y5) PO[ZH]=(ze, z2, z3, z4, z5)	Oprócz każdorazowych punktów końcowych można dodatkowo programować krzywe w przestrzeni przy pomocy wielomianów.
xe, ye, ze	Punkty końcowe krzywej w przestrzeni
xi, yi, zi	Współczynniki wielomianów maksymalnie 5. stopnia

Wskazówka**Identyfikatory XH YH ZH do programowania 2. toru orientacji**

Identyfikatory muszą zostać tak wybrane, by nie powstał konflikt z innymi identyfikatorami osi liniowych

osie X Y Z

i osiami obrotowymi jak

A2 B2 C2 kąt Eulera wzgl. kąt RPY

A3 B3 C3 wektory kierunkowe

A4 B4 C4 wzgl. A5 B5 C5 wektory normalnej powierzchni

A6 B6 C6 wektory obrotu wzgl. A7 B7 C7 współrzędne punktu pośredniego lub innymi parametrami interpolacji.

6.3 Wielomiany orientacji (PO[kąt], PO[współrzędna])

Funkcja

Niezależnie od tego, jaka interpolacja wielomianowa G-Code grupy 1 jest właśnie aktywna, mogą być programowane dwa różne typy wielomianów orientacji do maksymalnie 5. stopnia przy transformacji trzy- do pięcioosiowej.

1. Wielomiany dla **kąta**: kąt wyprzedzenia LEAD, kąt boczny TILT w odniesieniu do płaszczyzny, która jest wyznaczana przez orientację początkową i końcową.
2. Wielomiany dla **współrzędnych**: XH, YH, ZH drugiej krzywej w przestrzeni dla orientacji narzędzia punktu odniesienia na narzędziu.

Przy transformacji sześćoosiowej można w celu zorientowania narzędzia dodatkowo zaprogramować obrót wektora obrotu THT z wielomianami do maksymalnie 5. stopnia dla obrotów samego narzędzia.

Składnia

Wielomiany orientacji typu 1 dla **kąta**

N... PO[PHI] = (a2, a3, a4, a5)
N... PO[PSI] = (b2, b3, b4, b5)

Transformacja trzy- do pięcioosiowej

Transformacja trzy- do pięcioosiowej

Wielomiany orientacji typu 2 dla **współrzędnych**

N... PO[XH] = (xe, x2, x3, x4, x5)
N... PO[YH] = (ye, y2, y3, y4, y5)
N... PO[ZH] = (ze, z2, z3, z4, z5)

Identyfikatory dla współrzędnych drugiego toru orientacji dla orientacji narzędzia

Dodatkowo można w obydwu przypadkach programować wielomiany dla **obrotu** przy transformacji sześciosiowej z

N... PO [THT] = (c2, c3, c4, c5)

lub

N... PO [THT] = (d2, d3, d4, d5)

odniesiona do toru interpolacja obrotu narzędzia

absolutna, względna i styczna interpolacja do zmiany orientacji

wektora orientacji. Jest to możliwe wtedy, gdy transformacja obsługuje wektor obrotu z offsetem programowalnym i interpolowalnym przez kąt obrotu THETA.

Znaczenie

PO [PHI]	Kąt w płaszczyźnie między orientacją startową i końcową
PO [PSI]	Kąt, który opisuje odchylenie orientacji od płaszczyzny między orientacją startową i końcową
PO [THT]	Kąt obrotu zaprogramowanej przez obrót wektora obrotu przy pomocy THETA grupy 54 G-Code
PHI	Kąt wyprzedzenia LEAD
PSI	Kąt boczny TILT
THETA	Obrót wokół kierunku narzędzia w Z
PO [XH]	Współrzędna X punktu odniesienia na narzędziu
PO [YH]	Współrzędna Y punktu odniesienia na narzędziu
PO [ZH]	Współrzędna Z punktu odniesienia na narzędziu

Opis

Wielomiany orientacji nie mogą być programowane

- gdy interpolacje spline ASPLINE, BSPLINE, CSPLINE są aktywne. Wielomiany typu 1 dla kąta orientacji są możliwe dla każdego rodzaju interpolacji oprócz Spline tzn. przy interpolacji liniowej z posuwem szybkim G0 wzgl. z posuwem G1 przy interpolacji wielomianowej z POLY i przy interpolacji okręgu wzgl. ewolwenty z G2, G3, CIP, CT, INVCW i INCCCW. Wielomiany typu 2 dla współrzędnych orientacji są natomiast możliwe tylko wtedy, gdy jest aktywna interpolacja liniowa z posuwem szybkim G0 wzgl. z posuwem G1 albo interpolacja wielomianowa z POLY.
- gdy orientacja jest interpolowana przy pomocy interpolacji osi ORIAXES. W tym przypadku mogą być programowane bezpośrednio wielomiany z PO[A] i PO[B] dla osi orientacji A i B.

Wielomiany orientacji typu 1 z ORIVECT, ORIPLANE i ORICONxx

Przy interpolacji dużego okręgu i interpolacji na poboczniczy stożka z ORIVECT, ORIPLANE i ORICONxx są możliwe tylko wielomiany orientacji typu 1.

Wielomiany orientacji typu 2 z ORICURVE

Gdy jest aktywna interpolacja z dodatkową krzywą w przestrzeni ORICURVE, kartezyjskie składowe wektora orientacji są interpolowane i są możliwe tylko wielomiany orientacji typu 2.

6.4 Obroty orientacji narzędzia (ORIROTA, ORIOTR, ORIOTT, ORIOTC, THETA)

Funkcja

Jeżeli w przypadku typów maszyn z ruchomym narzędziem również musi być możliwa zmiana orientacji narzędzia, wówczas każdy blok jest programowany z orientacją końcową. Zależnie od kinematyki maszyny można programować albo kierunek orientacji osi orientacji albo kierunek obrotu wektora orientacji THETA. Dla tych wektorów obrotu można programować różne rodzaje interpolacji:

- ORIROTA: Kąt obrotu do kierunku obrotu zadanego absolutnie.
- ORIOTR: Kąt obrotu w stosunku do płaszczyzny między orientacją startową i końcową.
- ORIOTT: Kąt obrotu w stosunku do zmiany wektora orientacji.
- ORIOTC: Styczny kąt obrotu do stycznej do toru.

Składnia

Tylko gdy jest aktywny rodzaj interpolacji ORIROTA, można programować kąt obrotu albo wektor obrotu na cztery możliwe sposoby:

1. Bezpośrednio pozycje osi obrotowych A , B , C
2. Kąt Eulera (w stopniach) przez A_2 , B_2 , C_2
3. Kąt RPY (w stopniach) przez A_2 , B_2 , C_2
4. Wektor kierunku przez A_3 , B_3 , C_3 (kąt obrotu przy pomocy $THETA=<wartość>$)

W przypadku gdy ORIOTR albo ORIOTT są aktywne, kąt obrotu można programować już tylko bezpośrednio przy pomocy THETA.

Obrót można również zaprogramować w oddzielnym bloku, przy czym nie następuje zmiana orientacji. Przy tym ORIOTR i ORIOTT nie mają znaczenia. W tym przypadku kąt obrotu jest zawsze interpretowany w odniesieniu do kierunku absolutnego (ORIROTA).

N... ORIROTA	Ustalenie interpolacji wektora obrotu
N... ORIOTR	
N... ORIOTT	
N... ORIOTC	
N... A3= B3= C3= THETA=<wartość>	Ustalenie obrotu wektora orientacji
N... PO[THT]=(d2, d3, d4, d5)	Interpolacja kąta obrotu z wielomianem 5. stopnia

Znaczenie

ORIROTA:	Kąt obrotu do kierunku obrotu zadanego absolutnie
ORIOTR:	Kąt obrotu w stosunku do płaszczyzny między orientacją startową i końcową
ORIOTT:	Kąt obrotu jako styczny wektor obrotu do zmiany orientacji

ORIROTC:	Kąt obrotu jako styczny wektor obrotu do stycznej do toru
THETA:	Obrót wektora orientacji
THETA=<wartość>:	Kąt obrotu w stopniach, który jest osiągnięty na końcu bloku
THETA=Θe:	Kąt obrotu z kątem końcowym Θ _e wektora obrotu
THETA=AC (...):	Przełączanie pojedynczymi blokami na absolutne podawanie wymiaru
THETA=AC (...):	Przełączanie pojedynczymi blokami na przyrostowe podawanie wymiaru
Θe:	Kąt końcowy wektora obrotu zarówno absolutny z G90, jak też względny z G91 (przyrostowe podawanie wymiarów) jest aktywny
PO [THT] = (...):	Wielomian dla kąta obrotu

Przykład: Obroty orientacji

Kod programu	Komentarz
N10 TRAORI	; Uaktywnienie transformacji orientacji
N20 G1 X0 Y0 Z0 F5000	; Orientacja narzędzia
N30 A3=0 B3=0 C3=1 THETA=0	; W kierunku Z z kątem obrotu 0
N40 A3=1 B3=0 C3=0 THETA=90	; W kierunku X i obrót o 90 stopni
N50 A3=0 B3=1 C3=0 PO [THT] = (180, 90)	; Orientacja
N60 A3=0 B3=1 C3=0 THETA=IC(-90)	; W kierunku Y i obrót na 180 stopni
N70 ORIROTT	; Pozostaje stała i obrót na 90 stopni
N80 A3=1 B3=0 C3=0 THETA=30	; Kąt obrotu w stosunku do zmiany orientacji ; Kąt obrotu w kącie 30 stopni do płaszczyzny X-Y

Przy interpolacji od bloku N₄₀ kąt obrotu jest interpolowany liniowo od wartości startowej 0 do wartości końcowej 90 stopni. W bloku N₅₀ kąt obrotu zmienia się z 90 stopni do 180 stopni według parabol $\theta(u) = +90u^2$. W N₆₀ może również zostać wykonany obrót, bez zmiany orientacji.

W przypadku N₈₀ orientacja narzędzia jest obracana z kierunku Y do kierunku X. Przy tym zmiana orientacji leży w płaszczyźnie X-Y, a wektor obrotu tworzy z tą płaszczyzną kąt 30 stopni.

Opis

ORIROTA

Kąt obrotu THETA jest interpolowany względem absolutnie ustalonego kierunku w przestrzeni. Podstawowy kierunek obrotu jest definiowany przez dane maszynowe

ORIROTR

Kąt obrotu THETA jest interpretowany względem płaszczyzny, która jest wyznaczana przez orientację początkową i orientację końcową.

ORIROTT

Kąt obrotu θ jest interpretowany względem zmiany orientacji. Dla $\theta=0$ wektor obrotu jest interpolowany stycznie do zmiany orientacji i różni się tylko wtedy od ORIROTR , gdy dla orientacji został zaprogramowany co najmniej jeden wielomian dla "kąta wychylenia PSI". Z tego wynika zmiana orientacji, która nie przebiega w płaszczyźnie. Przez dodatkowo zaprogramowany kąt obrotu θ można wówczas np. tak interpolować wektor obrotu, by tworzył on zawsze określoną wartość do zmiany orientacji.

ORIROTC

Wektor obrotu jest interpolowany w stosunku do stycznej do toru z offsetem programowalnym przez kąt θ . Dla kąta offsetu może przy tym być również programowany wielomian $\text{PO}[\theta] = (c2, c3, c4, c5)$ maksymalnie 5. stopnia.

6.5 Orientacje w stosunku do toru**6.5.1 Rodzaje orientacji w stosunku do toru****Funkcja**

Przy pomocy tej rozszerzonej funkcji orientacja względna jest osiągnięta nie tylko na końcu bloku, lecz na całym przebiegu toru. Orientacja osiągnięta w poprzedzającym bloku jest przy pomocy interpolacji dużego okręgu prowadzona do zaprogramowanej orientacji końcowej. Zasadniczo są dwie możliwości programowania żądanej orientacji w stosunku do toru:

1. Orientacja narzędzia jak również obrót narzędzia jest przy pomocy ORIPATH , ORPATHTS interpolowana w stosunku do toru.
2. Wektor orientacji jest programowany i interpolowany jak zazwyczaj dotychczas. Przy pomocy ORIROTC jest inicjalizowany obrót wektora orientacji w stosunku do stycznej do toru.

Składnia

Rodzaj interpolacji orientacji i obrotu narzędzia jest programowany przy pomocy:

N... ORIPATH	Orientacja w stosunku do toru
N... ORIPATHS	Orientacja w stosunku do toru z wygładzaniem przebiegu orientacji
N... ORIROTC	Odniesiona do toru interpolacja wektora obrotu

Załamanie orientacji wywołane przez naroże w przebiegu toru można wygładzić przy pomocy ORIPATHS . Kierunek i długość drogi ruchu cofnięcia są programowane przez wektor o składowych $A8=X$, $B8=Y$, $C8=Z$.

Przy pomocy `ORIPATH/ORIPATHS` można programować różne odniesienia do stycznej do toru przez trzy kąty

- `LEAD`= podanie kąta do przodu w odniesieniu do toru i powierzchni
- `TILT`= podanie kąta bocznego w odniesieniu do toru i powierzchni
- `THETA`= kąt obrotu

dla całego przebiegu toru. Do kąta obrotu `THETA` mogą przy pomocy `PO[THT]=(...)` być dodatkowo programowane wielomiany maksymalnie 5. stopnia.

Wskazówka

Producent maszyny

Proszę przestrzegać danych producenta maszyny. Przez projektowalne dane maszynowe i ustawcze mogą do rodzaju orientacji w stosunku do toru zostać dokonane dalsze ustawienia. Dalsze objaśnienia patrz

Literatura:

/FB3/ Podręcznik działania Funkcje specjalne; Transformacje 3- do 5-osiowych (F2), Rozdział: "Orientacja"

Znaczenie

Interpolację kątów `LEAD` i `TILT` można przez daną maszynową różnie ustawiać:

- Odniesienie orientacji narzędzia zaprogramowane przy pomocy `LEAD` i `TILT` jest dotrzymane przez cały blok.
 - Kąt do przodu `LEAD`: obrót wokół kierunku prostopadle do stycznej i wektora normalnej
`TILT`: obrót orientacji wokół wektora normalnej.
 - Kąt do przodu `LEAD`: obrót wokół kierunku prostopadle do stycznej i wektora normalnej Kąt boczny `TILT`: obrót orientacji wokół kierunku stycznej do toru.
 - Kąt obrotu `THETA`: obrót narzędzia wokół siebie samego z dodatkową trzecią osią obrotową, jako osią orientacji przy transformacji sześciooosiowej.
-

Wskazówka

Odniesiona do toru orientacja razem z `OSC`, `OSS`, `OSSE`, `OSD`, `OST` jest niedopuszczalna

Odniesiona do toru interpolacja orientacji `ORIPATH` wzgl. `ORIPATHS` i `ORIOTC` nie może być programowana razem z wygładzaniem przebiegu orientacji przy pomocy jednego z G-Code z grupy 34. W tym celu musi być aktywne `OSOF`.

6.5.2 Odniesiony do toru obrót orientacji narzędzia (ORIPATH, ORIPATHS, kąt obrotu)

Funkcja

W przypadku transformacji sześcioośiowej można w celu orientacji narzędzia dowolnie w przestrzeni również obrócić narzędzie wokół siebie samego przy pomocy trzeciej osi obrotowej. Przy odniesionym do toru obrocie orientacji narzędzia przy pomocy ORIPATH lub ORIPATHS można zaprogramować dodatkowy obrót przez kąt obrotu THETA. Alternatywnie do tego kąty LEAD i TILT mogą być programowane przez wektor, który leży na płaszczyźnie prostopadle do kierunku narzędzia.

Producent maszyny

Proszę przestrzegać danych producenta maszyny. Przez daną maszynową można różnie ustawić interpolację kątów LEAD i TILT.

Składnia

Obrót orientacji narzędzia i narzędzia

Rodzaj orientacji narzędzia w stosunku do toru jest uaktywniany przy pomocy ORIPATH albo ORIPATHS.

N... ORIPATH	Uaktywnienie rodzaju orientacji w odniesieniu do toru
N... ORIPATHS	Uaktywnienie rodzaju orientacji w odniesieniu do toru z wygładzaniem przebiegu orientacji
Uaktywnienie trzech możliwych kątów z działaniem obrotowym:	
N... LEAD=	Kąt dla zaprogramowanej orientacji w stosunku do wektora normalnej powierzchni
N... TILT=	Kąt dla zaprogramowanej orientacji w płaszczyźnie prostopadle do stycznej do toru w stosunku do wektora normalnej powierzchni
N... THETA=	Kąt obrotu w stosunku do zmiany orientacji wokół kierunku narzędzia trzeciej osi obrotowej

Wartości kątów na końcu bloku są programowane przy pomocy LEAD=wartość, TILT=wartość wzgl. THETA=wartość. Dodatkowo do stałych kątów mogą dla wszystkich trzech kątów być programowane wielomiany maksymalnie 5. stopnia.

N... PO[PHI]=(a2, a3, a4, a5)	Wielomian dla kąta wyprzedzenia LEAD
N... PO[PSI]=(b2, b3, b4, b5)	Wielomian dla kąta bocznego TILT
N... PO[THT]=(d2, d3, d4, d5)	Wielomian dla kąta obrotu THETA

Przy programowaniu wyższe współczynniki wielomianu, które wynoszą zero, można pominąć. Przykład PO[PHI]=a2 daje parabolę dla kąta wyprzedzenia LEAD.

Znaczenie

Orientacja narzędzia odniesiona do toru

ORIPATH	Orientacja narzędzia odniesiona do toru
ORIPATHS	Orientacja narzędzia w odniesieniu do toru, załamanie w przebiegu orientacji jest wygładzane
LEAD	Kąt w stosunku do wektora normalnej powierzchni, w płaszczyźnie wyznaczonej przez styczną do toru i wektor normalnej powierzchni
TILT	Obrót orientacji wokół kierunku Z lub obrót wokół stycznej do toru
THETA	Obrót wokół kierunku narzędzia do Z
PO [PHI]	Wielomian orientacji dla kąta wyprzedzenia LEAD
PO [PSI]	Wielomian orientacji dla kąta bocznego TILT
PO [THT]	Wielomian orientacji dla kąta obrotu THETA

Wskazówka

Kąt obrotu THETA

Do obrotu narzędzia z trzecią osią obrotową jako osią orientacji wokół siebie samego jest wymagana transformacja sześćoosiowa.

6.5.3 Interpolacja obrotu narzędzia względem toru (ORIOTC, THETA)

Funkcja

Interpolacja z wektorami obrotu

Do zaprogramowanego przy pomocy ORIOTC obrotu narzędzia w stosunku do stycznej do toru można interpolować wektor obrotu również z offsetem programowanym przez kąt obrotu THETA. Przy tym można dla kąta offsetu z PO[THT] programować wielomian do maksymalnie 5. stopnia.

Składnia

N... ORIOTC	Inicjalizacja obrotu narzędzia względem stycznej do toru
N... A3= B3= C3= THETA=wartość	Ustalenie obrotu wektora orientacji
N... A3= B3= C3= PO[THT]=(c2, c3, c4, c5)	Interpolacja kąta offsetu z wielomianem maksymalnie 5. stopnia

Obrót można również zaprogramować w oddzielnym bloku, przy czym nie następuje zmiana orientacji.

Znaczenie

Odniesiona do toru interpolacja obrotu narzędzia przy transformacji sześciooosiowej

ORIROTC	Ustawienie stycznego wektora obrotu do stycznej do toru
THETA=wartość	Kąt obrotu w stopniach, który jest osiąganym na końcu bloku
THETA= θ_e	Kąt obrotu z kątem końcowym Θ_e wektora obrotu
THETA=AC (...)	Przełączanie pojedynczymi blokami na absolutne podawanie wymiaru
THETA=IC (...)	Przełączanie pojedynczymi blokami na przyrostowe podawanie wymiaru
PO [THT] = (c2, c3, c4, c5)	Interpolacja kąta offsetu z wielomianem 5. stopnia

Wskazówka**Interpolacja wektora obrotu ORIROTC**

Ustawienie również obrotu narzędzia w stosunku do stycznej do toru przeciwnie do kierunku jego orientacji jest możliwe tylko w przypadku transformacji sześciooosiowej.

Przy aktywnym ORIROTC

Wektora obrotu ORIROTA nie można zaprogramować. W przypadku zaprogramowania jest wyprowadzany ALARM 14128 "Absolutne zaprogramowanie obrotu narzędzia przy aktywnym ORIROTC".

Kierunek orientacji narzędzia przy transformacji trzy- do pięcioosiowej

Kierunek orientacji narzędzia może być programowany jak zazwyczaj przy transformacji trzy- do pięcioosiowej przez kąt Eulera wzgl. kąt RPY albo wektory biegunowe. Również są możliwe zmiany orientacji narzędzia w przestrzeni przez programowanie interpolacji dużego okręgu ORIVECT, interpolacji liniowej osi orientacji ORIAXES, wszystkie interpolacje na poboczniczy stożka ORICONxx, jak też interpolacji dodatkowo do krzywej w przestrzeni z dwoma punktami kontaktowymi narzędzia ORICURVE.

G	Podanie rodzaju ruchu osi obrotowych
X Y Z	Podanie osi liniowych
ORIAXES	Interpolacja liniowa osi maszyny albo osi orientacji
ORIVECT	Interpolacja dużego okręgu (identyczna z ORIPLANE)
ORIMKS	Obrót w układzie współrzędnych maszyny
ORIWKS	Obrót w układzie współrzędnych obrabianego przedmiotu
	Opis patrz punkt "Obroty orientacji narzędzia"
A= B= C=	Programowanie pozycji osi maszyny
ORIEULER	Programowanie orientacji przez kąt Eulera
ORIRPY	Programowanie orientacji przez kąt RPY
A2= B2= C2=	Programowanie kąta osi wirtualnych

ORIVIRT1	Programowanie orientacji przez wirtualne osie orientacji
ORIVIRT2	(Definicja 1), ustalenie według MD \$MC_ORIAX_TURN_TAB_1
	(Definicja 2), ustalenie według MD \$MC_ORIAX_TURN_TAB_2
A3= B3= C3=	Programowanie wektora biegunowego osi kierunkowej
ORIPLANE	Interpolacja na płaszczyźnie (Interpolacja dużego okręgu)
ORICONCW	Interpolacja na poboczniczy stożka w kierunku ruchu wskazówek zegara
ORICONCCW	Interpolacja na poboczniczy stożka przeciwnie do ruchu wskazówek zegara
ORICONTO	Interpolacja na poboczniczy stożka przejście styczne
A6= B6= C6=	Programowanie osi obrotu stożka (wektor znormalizowany)
NUT=kał	Kąt rozwarcia stożka w stopniach
NUT=+179	Kąt ruchu mniejszy albo równy 180 stopni
NUT=-181	Kąt ruchu większy albo równy 180 stopni
ORICONIO	Interpolacja na poboczniczy stożka
A7= B7= C7=	Orientacja pośrednia (programowanie jako wektor znormalizowany)
ORICURVE	Interpolacja orientacji z zadaniem ruchu dwóch punktów
XH YH ZH np. z	kontaktowych narzędzia. Oprócz każdorazowych punktów
wielomianami	końcowych można programować dodatkowe wielomiany
PO [XH] = (xe, x2, x3,	krzywych w przestrzeni.
x4, x5)	

Wskazówka

Gdy orientacja narzędzia jest interpolowana z aktywnym ORIAXES przez osie orientacji, wówczas odniesiona do toru inicjalizacja kąta obrotu jest spełniana tylko na końcu bloku.

6.5.4 Wygładzanie przebiegu orientacji (ORIPATHS A8=, B8=, C8=)

Funkcja

Przy wykazujących stałe przyśpieszenie zmianach orientacji na konturze, przerwania ruchów po torze, które szczególnie mogą występować na narożnikach konturu, są niepożądane. Wynikające z tego załamanie w przebiegu orientacji można wygładzić przez wstawienie własnego bloku pośredniego. Zmiana orientacji następuje wówczas ze stałym przyśpieszeniem, gdy podczas przeorientowania jest aktywne również ORIPATHS. W tej fazie może zostać przeprowadzony ruch cofnięcia narzędzia.

Producent maszyny

Proszę przestrzegać wskazówek producenta maszyny dot. ewentualnie predefiniowanych danych maszynowych i ustawczych, przy pomocy których ta funkcja jest uaktywniana.

Przez daną maszynową można ustawić, jak wektor cofnięcia jest interpretowany:

1. W układzie współrzędnych narzędzia współrzędna Z jest definiowana przez kierunek narzędzia.
2. W układzie współrzędnych obrabianego przedmiotu współrzędna Z jest definiowana przez aktywną płaszczyznę.

Dalsze objaśnienia do funkcji "Orientacja odniesiona do toru" patrz

Literatura:

Podręcznik działania Funkcje specjalne; Transformacje wieloosiowe (F2)

Składnia

Dla stałych orientacji narzędzia w odniesieniu do całego toru są na narożniku konturu wymagane dalsze dane programowe. Kierunek i długość drogi tego ruchu jest programowany przez wektor ze składowymi A8=X, B8=Y, C8=Z:

```
N... ORIPATHS A8=X B8=Y C8=Z
```

Znaczenie

ORIPATHS	Orientacja narzędzia w odniesieniu do toru, załamanie w przebiegu orientacji jest wygładzane.
A8= B8= C8=	Składowe wektora dla kierunku i długości drogi
X, Y, Z	Ruch cofnięcia w kierunku narzędzia

Wskazówka**Programowanie wektora biegunowego A8, B8, C8**

Gdy długość wektora jest równa zero, ruch cofnięcia nie następuje.

ORIPATHS

Odniesiona do toru orientacja narzędzia staje się aktywna z ORIPATHS. W przeciwnym przypadku orientacja jest przy pomocy liniowej interpolacji dużego okręgu transferowana z orientacji startowej do końcowej.

6.6 Kompresja orientacji (COMPON, COMPCURV, COMPCAD)

Funkcja

Programy NC, w których jest aktywna transformacja orientacji ($_{TRAORI}$) i są programowane orientacje narzędzi (obojętnie jakiego rodzaju), mogą być kompresowane przy zachowaniu zadanych tolerancji.

Programowanie

Orientacja narzędzia

W przypadku gdy jest aktywna transformacja orientacji ($_{TRAORI}$), w przypadku maszyn 5-osiowych orientacja narzędzia może być programowana następująco (niezależnie od kinematyki):

- Programowanie **wektora** kierunkowego przez:

A3=<...> B3=<...> C3=<...>

- Programowanie **kątów** Eulera wzgl. **kątów** RPY przez:

A2=<...> B2=<...> C2=<...>

Obrót narzędzia

W przypadku maszyn **6-osiowych** można dodatkowo do orientacji narzędzia zaprogramować jeszcze jego obrót.

Programowanie kąta obrotu następuje przy pomocy:

THETA=<...>

Patrz "Obroty orientacji narzędzia (Strona 339)".

Wskazówka

Bloki NC, w których dodatkowo jest zaprogramowany obrót, dają się kompresować tylko wtedy, gdy kąt obrotu zmienia się **liniowo**. Tzn. dla kąta obrotu nie może być zaprogramowany wielomian przy pomocy $PO_{[THT]} = (\dots)$.

Ogólna forma kompresowalnego bloku NC

Ogólna forma kompresowalnego bloku NC może dlatego wyglądać następująco:

N... X=<...> Y=<...> Z=<...> A3=<...> B3=<...> C3=<...> THETA=<...> F=<...>

wzgl.

N... X=<...> Y=<...> Z=<...> A2=<...> B2=<...> C2=<...> THETA=<...> F=<...>

Wskazówka

Wartości pozycji mogą być podawane bezpośrednio (np. X90) lub pośrednio przez przyporządkowania parametrów (np. $X=R1*(R2+R3)$).

Programowanie orientacji narzędzia przez pozycje osi obrotowych

Orientacja narzędzia może być również podana przez pozycje osi obrotowych, np. w formie:

N... X=<...> Y=<...> Z=<...> A=<...> B=<...> C=<...> THETA=<...> F=<...>

W tym przypadku kompresja jest przeprowadzana na dwa różne sposoby, zależnie od tego, czy interpolacja dużego okręgu jest przeprowadzana czy nie. Gdy interpolacja wielkiego okręgu nie następuje, wówczas skompresowana zmiana orientacji jest przedstawiana w zwykły sposób przez wielomiany osiowe dla osi obrotowych.

Dokładność konturu

Zależnie od ustawionego trybu kompresji (MD20482 \$MC_COMPRESSOR_MODE) przy kompresji działają dla osi geometrycznych i osi orientacji albo zaprojektowane tolerancje specyficzne dla osi (MD33100 \$MA_COMPRESS_POS_TOL) albo następujące tolerancje specyficzne dla kanału, ustawiane przez dane ustawcze:

SD42475 \$SC_COMPRESS_CONTUR_TOL (maksymalne odchylenie konturu)

SD42476 \$SC_COMPRESS_ORI_TOL (maksymalne odchylenie kątowne orientacji narzędzia)

SD42477 \$SC_COMPRESS_ORI_ROT_TOL (maksymalne odchylenie kątowne dla kąta obrotu narzędzia) (dostępne tylko w przypadku maszyn 6-osiowych)

Literatura:

Podręcznik działania Funkcje podstawowe; Transformacja 3- do 5-osiowej (F2),
Rozdział: "Kompresja orientacji"

Uaktywnienie / wyłączenie aktywności

Funkcje kompresora są włączane przez modalne G-Code COMPON, COMPCURV wzgl. COMPCAD.

Funkcja kompresora ulega zakończeniu przy pomocy COMPOF.

Patrz "Kompresja bloków NC (COMPON, COMPCURV, COMPCAD) (Strona 245)".

Wskazówka

Ruch orientacji jest kompresowany tylko przy aktywnej interpolacji dużego okręgu (tzn. zmiana orientacji narzędzia następuje w płaszczyźnie, która jest wyznaczana przez orientację startową i końcową).

Interpolacja dużego okręgu jest przeprowadzana pod następującymi warunkami:

- MD21104 \$MC_ORI_IPO_WITH_G_CODE = 0,
ORIWKS jest aktywne a
orientacja jest programowana przy pomocy wektorów (z A3, B3, C3 wzgl. A2, B2, C2).
- MD21104 \$MC_ORI_IPO_WITH_G_CODE = 1 i
ORIVECT lub ORIPLANE jest aktywne.

Orientacja narzędzia może być zaprogramowana albo jako wektor biegunowy, albo z pozycjami osi obrotowych. Jeżeli jest aktywny jeden z G-Code ORICON_{xx} albo ORICURVE albo są zaprogramowane wielomiany dla kąta orientacji (PO[PHI] i PO[PSI]), nie jest przeprowadzana interpolacja dużego okręgu.

Przykład

W poniższym przykładzie programowania jest kompresowany okrąg, który jest przybliżony przez ciąg prostych. Orientacja narzędzia porusza się przy tym synchronicznie do niego na poboczniczy stożka. Chociaż kolejne zaprogramowane zmiany orientacji przebiegają niestale, funkcja kompresora generuje gładki przebieg orientacji.

Programowanie	Komentarz
DEF INT LICZBA=60	
DEF REAL PROMIEŃ=20	
DEF INT COUNTER	
DEF REAL KĄT	
N10 G1 X0 Y0 F5000 G64	
\$SC_COMPRESS_CONTUR_TOL=0.05	; Maksymalne odchylenie konturu = 0.05 mm
\$SC_COMPRESS_ORI_TOL=5	; Maksymalne odchylenie orientacji = 5 stopni
TRAORI	
COMPCURV	
	; Następuje ruch po okręgu, który jest tworzony z prostych. Orientacja porusza się przy tym po stożku wokół osi Z z kątem rozwarcia 45 stopni.
N100 X0 Y0 A3=0 B3=-1 C3=1	
N110 FOR COUNTER=0 TO LICZBA	
N120 WINKEL=360*COUNTER/LICZBA	
N130 X=PROMIEŃ*cos(KĄT) Y=PROMIEŃ*sin(KĄT)	
A3=sin(KĄT) B3=-cos(KĄT) C3=1	
N140 ENDFOR	

6.7 Wygładzanie przebiegu orientacji (ORISON, ORISOF)

Funkcja

Przy pomocy funkcji "Wygładzanie przebiegu orientacji (ORISON)" można wygładzać wahania orientacji przez wiele zmian. Przez to osiąga się gładki przebieg zarówno orientacji, jak też konturu.

Warunek

Funkcja "Wygładzanie przebiegu orientacji (ORISON)" jest dostępna tylko w systemach z transformacją 5/6-osiową.

Składnia

```
ORISON
...
ORISOF
```

Znaczenie

ORISON:	Wygładzanie przebiegu orientacji WŁ. Działanie: Modalnie
ORISOF:	Wygładzanie przebiegu orientacji WYŁ. Działanie: Modalnie

Dane ustawcze

Wygładzanie przebiegu orientacji następuje przy zachowaniu:

- zadanej tolerancji maksymalnej (maksymalne odchylenie kątowe orientacji narzędzia w stopniach)

i

- zadanej maksymalnej drogi po torze.

Te zadane wielkości są definiowane przez dane ustawcze:

- SD42678 \$SC_ORISON_TOL (tolerancja wygładzania przebiegu orientacji)
- SD42680 O\$SC_ORISON_DIST (droga po torze dla wygładzania przebiegu orientacji)

Przykład

Kod programu	Komentarz
...	
TRAORI()	; Włączenie transformacji orientacji.
ORISON	; Włączenie wygładzania orientacji.
\$SC_ORISON_TOL=1.0	; Tolerancja wygładzania orientacji = 1,0 stopień.
G91	
X10 A3=1 B3=0 C3=1	
X10 A3=-1 B3=0 C3=1	
X10 A3=1 B3=0 C3=1	
X10 A3=-1 B3=0 C3=1	
X10 A3=1 B3=0 C3=1	
X10 A3=-1 B3=0 C3=1	
X10 A3=1 B3=0 C3=1	
X10 A3=-1 B3=0 C3=1	
X10 A3=1 B3=0 C3=1	
X10 A3=-1 B3=0 C3=1	
...	
ORISOF	; Wyłączenie wygładzania orientacji.
...	

Orientacja ulega skrętowni o 90 stopni w płaszczyźnie XZ, od -45 do +45 stopni. W wyniku wygładzenia przebiegu orientacji nie osiąga już ona maksymalnych wartości kątowych -45 wzgl. +45 stopni.

Dalsze informacje

Liczba bloków

Wygładzanie przebiegu orientacji następuje przez zaprojektowaną liczbę bloków, która jest zapisana w danej maszynowej MD28590 \$MC_MM_ORISON_BLOCKS.

Wskazówka

Jeżeli wygładzanie przebiegu orientacji zostanie uaktywnione przy pomocy ORISON, bez zaprojektowania w tym celu wystarczającej pamięci bloków (MD28590 < 4), wówczas następuje alarm i funkcja nie może zostać wykonana.

Maksymalna długość drogi w bloku

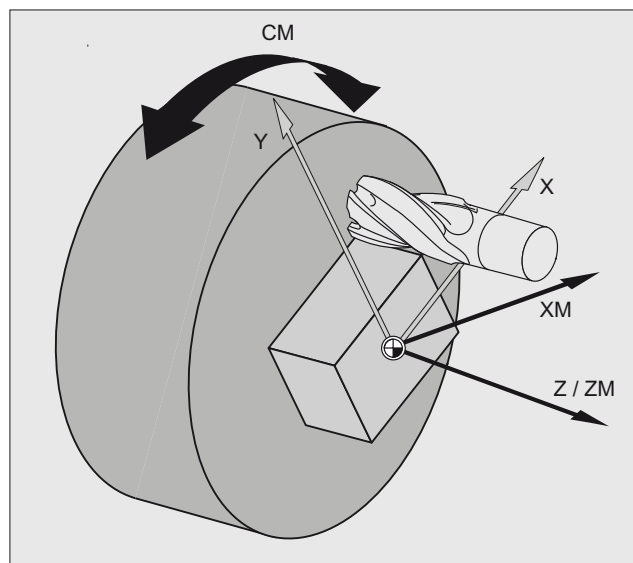
Przebieg orientacji jest wygładzany tylko w takich blokach, w których droga ruchu jest mniejsza, niż zaprojektowana droga maksymalna w bloku (MD20178 \$MC_ORISON_BLOCK_PATH_LIMIT). Bloki o dłuższych drogach ruchu przerywają wygładzanie i są wykonywane jak zaprogramowano.

6.8 Transformacja kinematyczna

6.8.1 Obróbka frezarska na częściach toczonych (TRANSMIT)

Funkcja

Rozkaz TRANSMIT aktywuje transformacje do obróbki strony czołowej na tokarkach.



X	Oś geometryczna
Y	Oś geometryczna
Z	Oś geometryczna
XM	Oś maszyny
ZM	Oś maszyny
CM	Oś maszyny

Literatura

Szczegółowe informacje dot. funkcji TRANSMIT znajdują się w:

Podręcznik działania Funkcje rozszerzone; rozdział "Transformacja kinematyczna" > "Transformacja części czołowej TRANSMIT"

Składnia

TRANSMIT
TRANSMIT (n)
TRAFOOF

Znaczenie

TRANSMIT:	Włączyć TRANSMIT z pierwszym zestawem danych TRANSMIT
TRANSMIT (n):	Włączyć TRANSMIT z następnym zestawem danych TRANSMIT
TRAFOOF:	Wyłączenie transformacji

Wskazówka

Aktywna transformacja TRANSMIT zostanie wyłączona, jeśli w kanale zostanie włączona inna transformacja np. TRACYL, TRAANG, TRAORI.

Przykład

Kod programu	Komentarz
N10 T1 D1 G54 G17 G90 F5000 G94	; Wybór narzędzia
N20 G0 X20 Z10 SPOS=45	; Ruch do pozycji wyjściowej
N30 TRANSMIT	Włączyć TRANSMIT z pierwszym zestawem danych TRANSMIT
N40 ROT RPL=-45	; Ustawienie frame
N50 ATRANS X-2 Y10	
N60 G1 X10 Y-10 G41 OFFN=1	; Obróbka zgrubna czopa kwadratowego; naddatek 1 mm (OFFN)
N70 X-10	
N80 Y10	
N90 X10	
N100 Y-10	
N110 G0 Z20 G40 OFFN=0	; wymiana narzędzia
N120 T2 D1 X15 Y-15	
N130 Z10 G41	
N140 G1 X10 Y-10	; Obróbka wykańczająca czopa kwadratowego
N150 X-10	
N160 Y10	
N170 X10	
N180 Y-10	
N190 Z20 G40	; Cofnięcie wyboru frame
N200 TRANS	
N210 TRAFOOF	; wyłączenie TRANSMIT
N220 G0 X20 Z10 SPOS=45	; Ruch do pozycji wyjściowej
N230 M30	

Opis

Literatura

Szczegółowy opis funkcji znajduje się w:

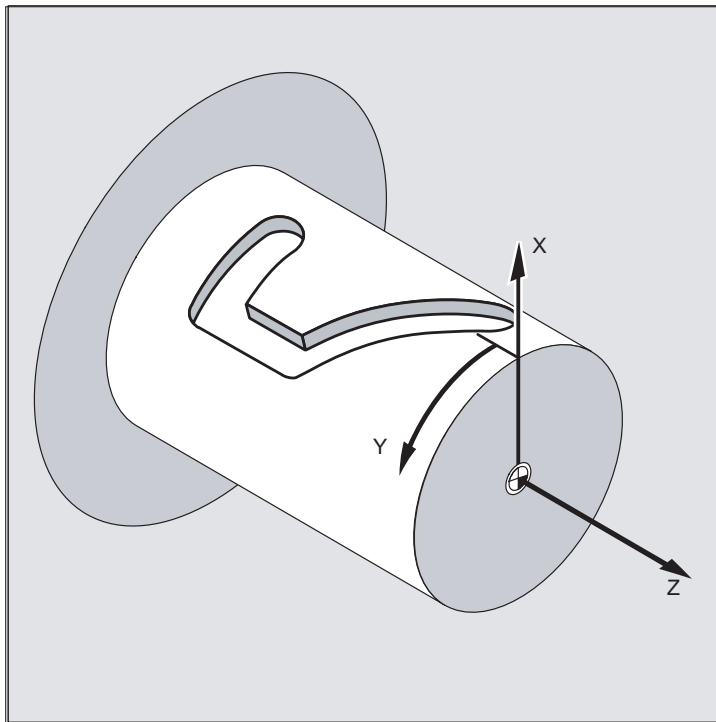
Podręcznik działania Funkcje rozszerzone; rozdział "M1 Transformacja kinematyczna" > "TRANSMIT"

6.8.2 Transformacja poboczniczy walca (TRACYL)

Funkcja

Rozkaz TRACYL aktywuje transformację do obróbki rowków na ciałach cylindrycznych.

Przebieg rowków jest programowany w odniesieniu do rozwiniętej, równej powierzchni poboczniczy walca.



TRACYL typy transformacji

Są trzy formy transformacji poboczniczy walca:

- TRACYL Bez korekcji ścianki rowka (typ transformacji 512):
- TRACYL z korekcją ścianki rowka (typ transformacji 513):
- TRACYL z zaprogramowaną korekcją ścianki rowka (typ transformacji 514):

Przy transformacji poboczniczy walca z korekcją ścianki rowka oś stosowana do korekcji powinna być ustawiona na zero ($y=0$), aby rowek był wykonywany środkowo względem zaprogramowanej jego linii środkowej.

Korzystanie z osi

Następujące osie nie mogą być stosowane jako osie pozycjonowania, wzgl. osie wahlwe:

- oś geometryczna w kierunku obwodowym poboczniczy walca (oś Y)
- dodatkowa oś liniowa w przypadku korekcji ścianki rowka (oś Z)

Składnia

TRACYL (d)

TRACYL (d, n)

TRACYL (d, n, k)

TRAFOOF

Znaczenie

TRACYL (d)

Włączanie TRACYL z pierwszym zestawem danych TRACYL oraz średnicy roboczej d

TRACYL (d, n)

Włączanie TRACYL z następnym zestawem danych TRACYL oraz średnicy roboczej d

d

Średnica odniesienia lub robocza

Zaprogramowana wartość F powinna być większa od 1.

n

Numer zestawu danych TRACYL (**opcjonalnie**)

Zakres wartości: 1, 2

k

Parametr jest istotny tylko dla typu transformacji 514:

- k = 0: bez korekcji ścianki rowka
- k = 0: Z korekcją ścianki rowka

Jeśli nie podano parametru, w danej maszynowej TRACYL_DEFAULT_MODE_<t> działa sparametryzowane położenie podstawowe. (**opcjonalnie**)

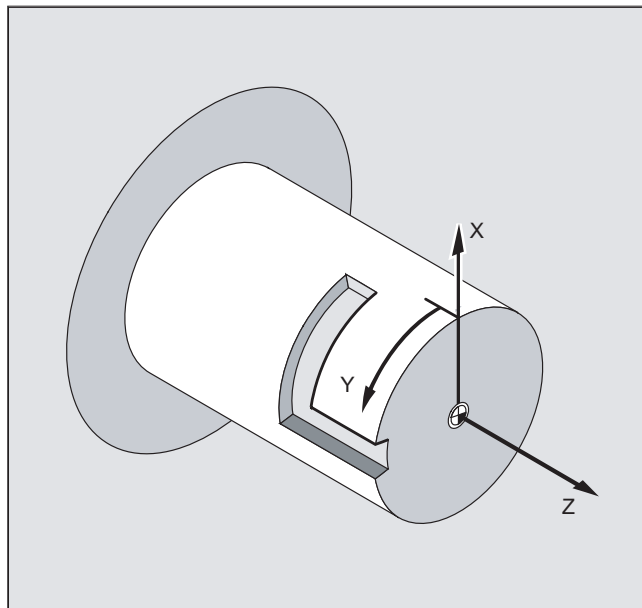
TRAFOOF

Wyłączenie transformacji

Wskazówka

Aktywna transformacja TRACYL zostanie wyłączona, jeśli w kanale zostanie włączona inna transformacja np. TRANSMI, TRAANG, TRAORI.

Przykład



Definicja narzędzia

Kod programu	Komentarz
; Parametry narzędzia	
\$TC_DP1 [1,1]=120	; Typ narzędzia: Frez
\$TC_DP2 [1,1]=0	; Położenie ostrza: Tylko dla narzędzi tokarskich

Kod programu	Komentarz
; Geometria: Korekcja długości	
: Wektor korekcji długości: Obliczenie według	
\$TC_DP3 [1,1]=8.	; Typ
\$TC_DP4 [1,1]=9.	; Płaszczyzna
\$TC_DP5 [1,1]=7.	

Kod programu	Komentarz
; Geometria: Promień	
\$TC_DP6 [1,1]=6.	; Promień
\$TC_DP7 [1,1]=0	; Szerokość rowka b dla piły do rowków, promień zaokrąglenia dla narzędzi frezarskich
\$TC_DP8 [1,1]=0	; Występ k tylko dla piły do rowków
\$TC_DP9 [1,1]=0	
\$TC_DP10 [1,1]=0	
\$TC_DP11 [1,1]=0	; Kąt dla stożkowych narzędzi frezarskich

Kod programu	Komentarz
	; Zużycie Korekcja długości i promienia
\$TC_DP12 [1,1]=0	; Pozostałe parametry do \$TC_DP24 = 0 (miara bazy/adapter)

Włączenie transformacji pobocznic walca

Kod programu	Komentarz
N10 T1 D1 G54 G90 F5000 G94	; Wybór narzędzia, kompensacja zamocowania
N20 SPOS=0	; Ruch do pozycji wyjściowej
N30 G0 X25 Y0 Z105 CC=200	
N40 TRACYL (40)	; TRACYL z pierwszym zestawem danych TRACYL i ; Włączenie średnicy roboczej 40mm
N50 G19	; płaszczyzna YZ

Wykonanie rowka hakowego:

Kod programu	Komentarz
N60 G1 X20	; Dosuw narzędzia do dna rowka
N70 OFFN=12	; Odstęp ścianki rowka 12 mm w stosunku do ; Ustalenie linii środkowej rowka
N80 G1 Z100 G42	; Dosunięcie do prawej ścianki rowka
N90 G1 Z50	; fragment rowka równoległy do osi walca
N100 G1 Y10	; fragment rowka równoległy do obwodu
N110 OFFN=4 G42	; Dosunięcie do lewej ścianki rowka ; Odstęp ścianki rowka 4 mm w stosunku do ; Ustalenie linii środkowej rowka
N120 G1 Y70	; fragment rowka równoległy do obwodu
N130 G1 Z100	; fragment rowka równoległy do osi walca
N140 G1 Z105 G40	; Odsunięcie od ścianki rowka
N150 G1 X25	; Odsunięcie
N160 TRAF00F	; Wyłączenie transformacji
N170 G0 X25 Y0 Z105 CC=200	; Ruch do pozycji wyjściowej
N180 M30	

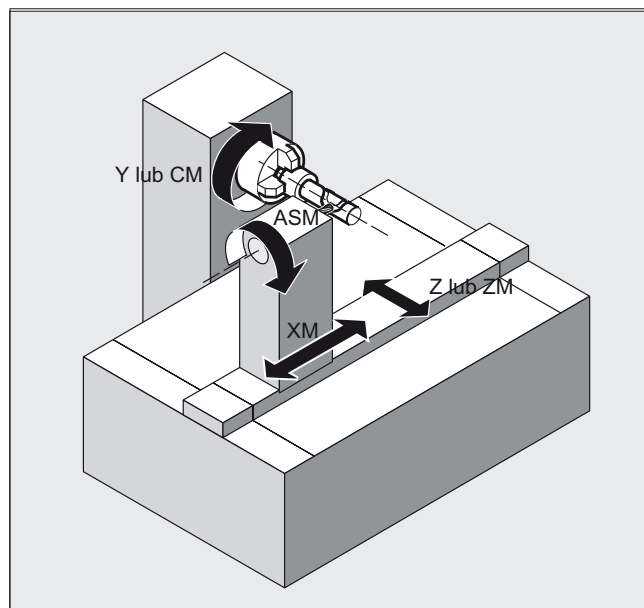
Opis

Transformacja poboczniczy walca bez korekcji ścianki rowka

Sterowanie transformuje zaprogramowane ruchy postępowe w walcowym układzie współrzędnych na ruchy postępowe w osiach maszyny:

- Oś obrotowa (Y / CM)
- Oś dosuwu prostopadła do osi obrotu (XM)
- Oś podłużna równoległa do osi obrotu (ZM)

Osie liniowe są prostopadłe do siebie.



XM	Oś dosuwu prostopadła do osi toczenia
Z / ZM	Oś liniowa równoległa do osi toczenia
Y / CM	Oś Y transformacji/ oś obrotowa
ASM	Wrzeciono robocze

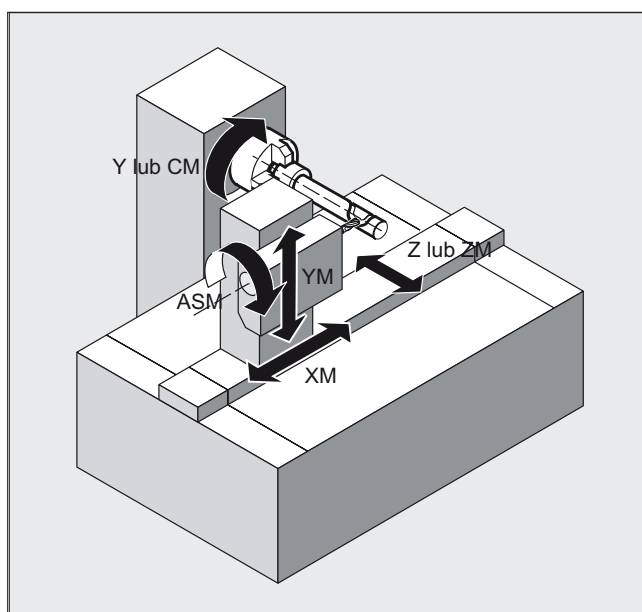
Rysunek 6-2 Transformacja poboczniczy walca bez korekcji ścianki rowka

Transformacja poboczniczy walca z korekcją ścianki rowka

Sterowanie transformuje zaprogramowane ruchy postępowe w walcowym układzie współrzędnych na ruchy postępowe w osiach maszyny:

- Oś obrotowa (Y / CM)
- Oś dosuwu prostopadła do osi obrotu (XM)
- Oś dodatkowa do korekcji ścianki rowka w stosunku prostopadłym do płaszczyzny X-Z (YM)
- Oś podłużna równoległa do osi obrotu (ZM)

Osie liniowe są prostopadłe do siebie.



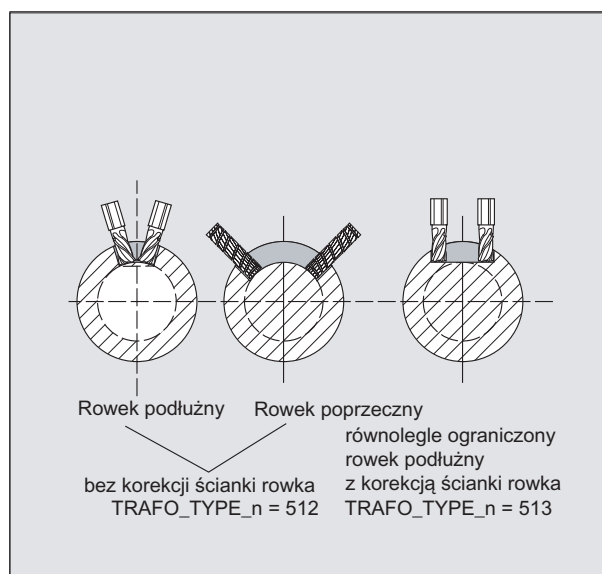
XM	Oś dosuwu prostopadła do osi toczenia
YM	Oś dodatkowa do korekcji ścianki rowka w stosunku prostopadłym do płaszczyzny X-Z
Z / ZM	Oś liniowa równoległa do osi toczenia
Y / CM	Oś Y transformacji/ oś obrotowa
ASM	Wrzeciono robocze

Rysunek 6-3 Transformacja poboczniczy walca z korekcją ścianki rowka

Zbocza rowka

Przy transformacji pobocznic walca bez korekcji ścianki rowka zbocza rowka wzdłużne do obrotowej są tylko wtedy równoległe, gdy szerokość rowka odpowiada średnicy narzędzia.

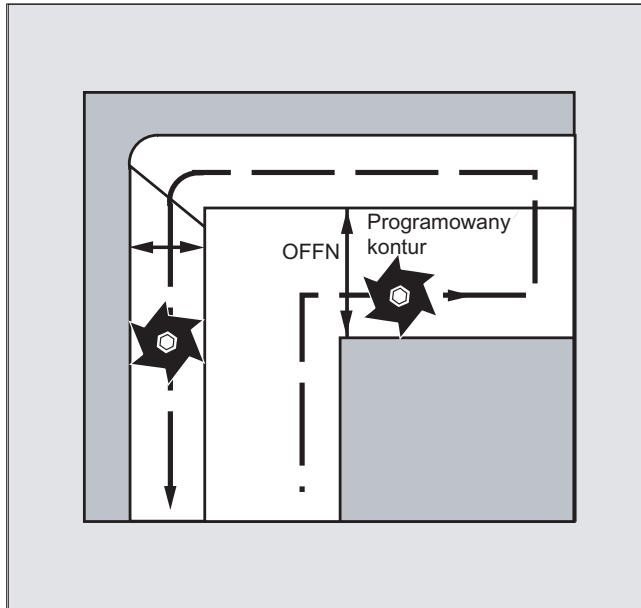
Zbocza rowków, które przebiegają równoległe do obwodu (poprzeczne) nie są równoległe na początku i końcu.



Offset konturu normalny OFFN (typ transformacji 513)

Żeby frezować rowki za pomocą `TRACYL`, w podprogramie programowana jest linia środkowa rowka, a poprzez parametr `OFFN` połowa jego szerokości.

Aby uniknąć uszkodzenia ścianki rowka, `OFFN` działa dopiero z wybraną korekcją promienia narzędzia. Ponadto musi być $OFFN \geq$ promień narzędzia, aby wykluczyć uszkodzenia przeciwległej ścianki rowka.



Program obróbki do frezowania rowka składa się z reguły z następujących kroków:

1. Wybór narzędzia
2. Wybór TRACYL
3. Wybór odpowiedniego przesunięcia współrzędnych (FRAME)
4. Pozycjonowanie
5. Programowanie OFFN
6. Wybór korekcji promienia narzędzia (WRK)
7. Blok dosuwu (zrealizowanie WRK i dosunięcie do ścianki rowka)
8. Kontur linii środkowej rowka
9. Cofnięcie wyboru WRK
10. Blok odsunięcia (cofnięcie realizacji korekcji promienia narzędzia i odsunięcie od ścianki rowka)
11. Pozycjonowanie
12. TRAF00F
13. Ponowny wybór pierwotnego przesunięcia współrzędnych (FRAME)

Cechy szczególne

- Korekcja promienia narzędzia
WRK nie jest wliczana do ścianki rowka, tylko do programowanego środka rowka. Żeby narzędzie mogło przejechać na lewo od ścianki rowka, zamiast G41 należy zaprogramować G42. Bądź podać wartość OFFN ze znakiem ujemnym.
- OFFN z TRACYL działa inaczej, niż bez TRACYL. Ponieważ OFFN jest wliczane również bez TRACYL przy aktywnej WRK, OFFN powinno po TRAF00F ponownie zostać ustawione na zero.
- Zmiana OFFN w ramach programu obróbki jest możliwa. W ten sposób linia środkowa rowka może zostać przesunięta ze środka (patrz rysunek).
- Za pomocą TRACYL oraz narzędzia, którego średnica jest mniejsza niż szerokość rowka nie jest możliwe wykonanie takiej samej geometrii ścianki rowka jak w przypadku narzędzia, którego średnica jest równa szerokości rowka. TRACYL minimalizuje błąd. W celu poprawy dokładności zaleca się wybrać średnicę narzędzia tylko nieco mniejszą od szerokości rowka.

Wskazówka

OFFN i WRK

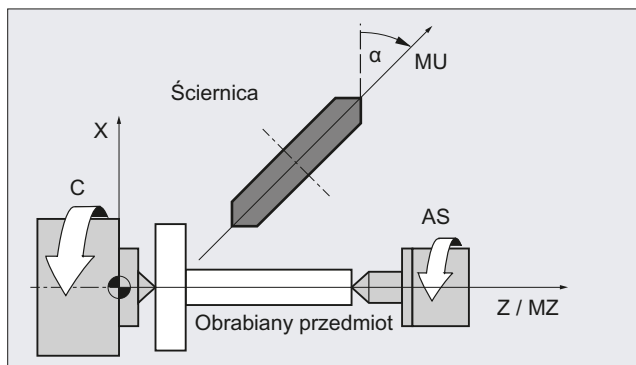
W przypadku TRAF0_TYPE_n = 512 wartość OFFN działa jako naddatek do WRK.

W przypadku TRAF0_TYPE_n = 513 dla OFFN programowana jest połowa szerokości rowka. Przejście po konturze następuje z OFFN-WRK.

6.8.3 Oś skośna (TRAANG)

Funkcja

Transformacja kąta skośnego bądź "Oś skośna" (TRAANG) służy do ułatwienia programowania w szlifiarkach za pomocą osi liniowej, która nie jest ustawiona w sposób prostopadły względem osi toczenia.



- X Oś geometryczna
- Z Oś geometryczna
- MZ Oś maszyny
- MU Oś maszyny
- α Kąt osi skośnej

Składnia

TRAANG
 TRAANG (
 TRAANG (, <n>
 TRAANG (α)
 TRAANG (α , <n>
 TRAFOOF

Znaczenie

Element	Opis
TRAANG TRAANG () :	Włączenie TRAANG z pierwszym zestawem danych TRAANG i ostatnio obowiązującym kątem α
TRAANG (, n) :	Włączenie TRAANG z następnym zestawem danych TRAANG i ostatnio obowiązującym kątem α
TRAANG (α) :	Włączenie TRAANG z pierwszym zestawem danych TRAANG i kątem α
TRAANG (α , <n>) :	Włączenie TRAANG z następnym zestawem danych TRAANG i kątem α

Element	Opis
α :	Kąt osi skośnej (opcjonalnie) Uwaga Bez podania kąta w danej maszynowej obowiązuje wartość sparametryzowana: \$MC_TRAANG_ANGLE_<n>, z n = numer zestawu danych
	Zakres wartości: $-90^\circ < \alpha < +90^\circ$
<n>:	Numer zestawu danych TRAANG (opcjonalnie!)
	Zakres wartości: 1, 2
TRAFOOF:	Wyłączenie transformacji

Przykład

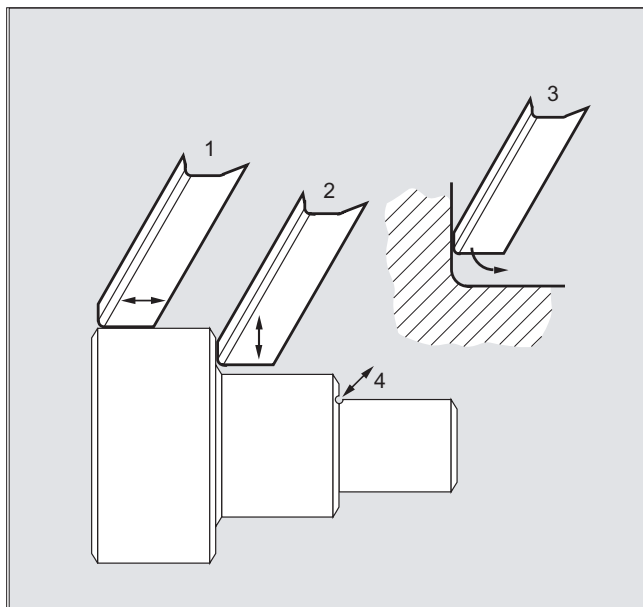
Osie geometryczne: Z, X

Oś maszyny: MU

Kod programu	Komentarz
N10 G0 G90 Z0 MU=10 G54 F5000 G18 G64 T1 D1	; płaszczyzny XZ, wybór narzędzia, ; Kompensacja zamocowania
N20 TRAANG(45)	; TRAANG z pierwszym zestawem danych TRAANG i ; Włączenie kąta 45°
N30 G0 Z10 X5	; Ruch do pozycji wyjściowej
N40 WAITP(Z)	; Czekanie na oś Z, w której ma zostać wykonany ruch
N50 OSP1[Z]=10 OSP2[Z]=5 OST1[Z]=-2 OST2[Z]=-2 FA[Z]=5000	; Ruch wahliwy ; OSP1/OSP2: lewy/prawy punkt nawrotu ; OST1/OST2: punkt zatrzymania w lewym/prawym punkcie nawrotnym
N60 OS[Z]=1	; Włączenie ruchu wahliwego
N70 POS[X]=4.5 FA[X]=50	; Pozycjonowanie równoległe osi X
N80 OS[Z]=0	; Wyłączenie ruchu wahliwego
N90 WAITP(Z)	; Czekanie na oś Z, w której ma zostać wykonany ruch
N100 TRAFOOF	; Wyłączenie transformacji
N110 G0 Z10 MU=10	; Odsunięcie
N120 M30	

Dalsze informacje

Zastosowania



- 1 Szlifowanie wzdłużne
- 2 Szlifowanie poprzeczne
- 3 Szlifowanie konturowe
- 4 Obróbka wcinająca skośna

Literatura

Szczegółowy opis funkcji znajduje się w:

Podręcznik działania Funkcje rozszerzone; rozdział "M1 Transformacja kinematyczna" > "Transformacja kąta skośnego TRAANG"

6.8.4 Programowanie osi skośnej (G5, G7)

Funkcja

Funkcje G7 oraz G5 służą do ułatwienia programowania obróbki wcinającej skośnej na szlifiarkach z transformacją "Oś skośna" (TRAANG), tak że przy wcinaniu ruch wykonuje oś skośna.

Należy zaprogramować przy tym tylko żadaną pozycję końcową ruchu wcinającego w X i Z. Wychodząc od aktualnej pozycji osi X, zaprogramowanej pozycji końcowej oraz kąta α osi skośnej NC obliczy przynależną pozycję startową dla G7 i nastąpi dosunięcie.

Pozycja startowa wynika z punktu przecięcia obu prostych:

- Prosta równoległa do osi Z w odstępnie od aktualnej pozycji osi X
- Prosta równoległa do osi skośnej zaprogramowanej przez pozycję docelową

Za pomocą następującej G5 nastąpi ruch osi skośnej na zaprogramowaną pozycję docelową.

Literatura

Podręcznik działania Funkcje rozszerzone; rozdział "M1 Transformacja kinematyczna" > "Transformacja kąta skośnego TRAANG"

Składnia

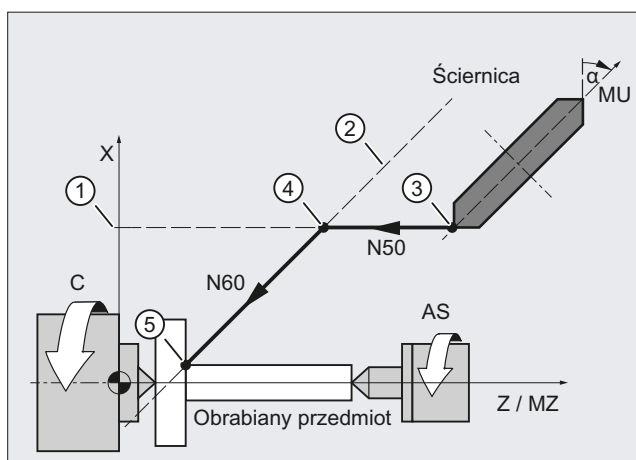
G7 <Endpos_X> <Endpos_Z>

G5 <Endos_X>

Znaczenie

G7:	Obliczenie oraz dosunięcie do pozycji startowej wcinania skośnego
G5:	Ruch w osiach skośnych w kierunku zaprogramowanej pozycji końcowej
<Endpos_X>:	Pozycja końcowa osi X
<Endpos_Z>:	Pozycja końcowa osi Z

Przykład



- ① Równoległe do osi Z w odstępnie od aktualnej pozycji osi X
- ② Równoległe do osi skośnej poprzez zaprogramowaną pozycję końcową
- ③ Pozycja wyjściowa
- ④ Wcinanie: Pozycja startowa,
- ⑤ Wcinanie: Pozycja końcowa
- X Oś geometryczna
- Z Oś geometryczna
- MZ Oś maszyny
- MU Oś maszyny

Kod programu	Komentarz
N.. G18	Wybór płaszczyzny XZ
N40 TRAANG(45,0)	1. Włączenie transformacji TRAANG , kąte = 45°
N50 G7 X40 Z70 F4000	Obliczenie oraz dosunięcie do pozycji startowej
N60 G5 X40 F100	Ruch w osi skośnej do pozycji końcowej
N70 ...	

6.9 Ruch kartezjański PTP

Funkcja

Przy pomocy tej funkcji pozycja może zostać zaprogramowana w kartezjańskim układzie współrzędnych, ruch maszyny następuje jednak we współrzędnych maszyny. Funkcja ta może być stosowana na przykład przy zmianie położenia przegubu, gdy przy tym ruch prowadzi przez położenie osobiwe.

Wskazówka

Funkcja ma sens tylko w połączeniu z aktywną transformacją. Ponadto "ruch PTP" jest dopuszczalny tylko w połączeniu z G0 i G1.

Składnia

```
N... TRAORI
N... STAT='B10' TU='B100' PTP
N... CP
```

Ruch PTP przy rodzajowej transformacji 5/6-osiowej

Jeżeli przy aktywnej generycznej transformacji 5/6 osiowej z PTP zostanie uaktywniony ruch punkt-do-punktu w układzie współrzędnym maszyny (`ORIMKS`), można programować orientację narzędzia z pozycjami osi obrotowych, z niezależnymi od kinematyki wektorami, kątami Eulera wzgl. RPY albo wektorami biegunowymi.

- Pozycje osi obrotowych: N... G1 X Y Z A B C
- Kąt Eulera w konwencji ZY'X" (kąt RPY) albo konwencji ZX'Z":
N... ORIRPY wzgl. N... ORIEULER
N... G1 X Y Z A2 B2 C2
- Wektory kierunku: N... G1 X Y Z A3 B3 C3

Przy tym może być aktywna zarówno interpolacja osi obrotowej, jak też interpolacja wektorowa z interpolacją dużego okręgu `ORIVECT` albo interpolacja z wektorem orientacji na pobocznicy stożka `ORICONxxx`.

Wieloznaczności orientacji z wektorami

Przy programowaniu orientacji z wektorami występują wieloznaczności w możliwych pozycjach osi obrotowych. Przyjmowane pozycje osi obrotowych mogą przy tym być wybierane przez programowanie `STAT = <...>`. Gdy zostanie zaprogramowane

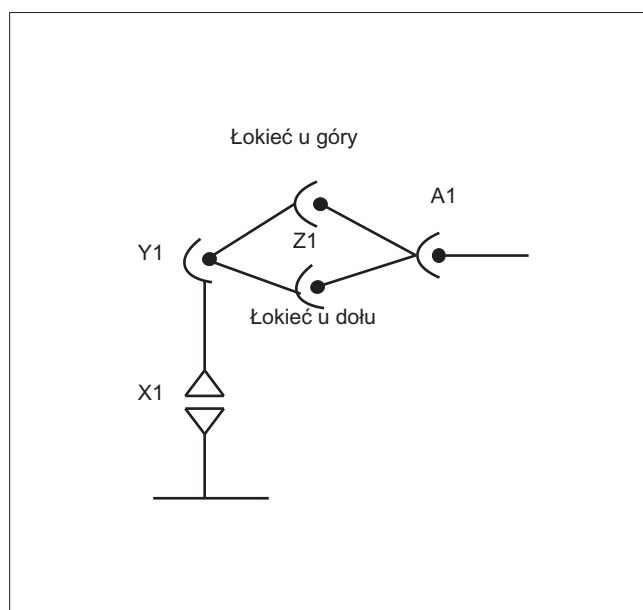
`STAT = 0` (odpowiada to ustawieniu standardowemu),
następuje ruch do pozycji znajdujących się w najmniejszej odległości od pozycji startowych.
Gdy zostanie zaprogramowane

`STAT = 1`,
następuje ruch do pozycji znajdujących się w dalszej odległości od pozycji startowych.

Znaczenie

Polecenie	Znaczenie
PTP:	Point to Point (ruch punkt do punktu) Działanie: modalnie
CP:	Continuous Path (kartezjański ruch po torze) Działanie: modalnie
STAT=:	Położenie przegubów. Wartość jest zależna od transformacji. Działanie: modalnie Wartość STAT działa tylko przy interpolacji wektorowej.
TU=:	Kąt osi Zakres wartości: ±360 stopni Działanie: pojedynczymi blokami Wartość TU działa przy interpolacji wektorowej i interpolacji osi obrotowej.

Przykład



Kod programu	Komentarz
N10 G0 X0 Y-30 Z60 A-30 F10000	; Położenie wyjściowe → łokieć u góry
N20 TRAORI(1)	; Transformacja wł.
N30 X1000 Y0 Z400 A0	
N40 X1000 Z500 A0 STAT='B10'	; Zmiana orientacji bez transformacji
TU='B100' PTP	; → Łokieć u dołu
N50 X1200 Z400 CP	; Transformacja ponownie aktywna
N60 X1000 Z500 A20	
N70 M30	

Przykład Ruch PTP przy rodzajowej transformacji 5-osiowej

Założenia: Podstawą jest prostokątna kinematyka CA.

Kod programu	Komentarz
TRAORI	; Transformacja kinematyka CA wł.
PTP	; Włączenie ruchu PTP
N10 A3=0 B3=0 C3=1	; Pozycje osi obrotowych C = 0 A = 0
N20 A3=1 B3=0 C3=1	; Pozycje osi obrotowych C = 90 A = 45
N30 A3=1 B3=0 C3=0	; Pozycje osi obrotowych C = 90 A = 90
N40 A3=1 B3=0 C3=1 STAT=1	; Pozycje osi obrotowych C = 270 A = -45

Wybór jednoznacznego ruchu do pozycji osi obrotowej:

W bloku N40 osie obrotowe w wyniku zaprogramowania $STAT = 1$ wykonują dłuższą drogę od swojego punktu startowego (C=90, A=90) do punktu końcowego (C=270, A=-45), zamiast krótszej drogi jak byłoby to w przypadku $STAT = 0$ (C=90, A=45).

Opis

Przełączanie między ruchami w układzie kartezyjskim i ruchami w osiach maszyny następuje przy pomocy poleceń PTP i CP.

Ruch PTP przy rodzajowej transformacji 5/6-osiowej

Przy ruchu PTP w przeciwieństwie do transformacji 5/6 osiowej TCP zazwyczaj nie pozostaje na stałym miejscu, gdy zmienia się tylko orientacja. Następuje ruch liniowy do transformowanych pozycji końcowych wszystkich osi transformacji (3 osie liniowe i max 3 osie obrotowe), bez trwającej nadal aktywności transformacji we właściwym sensie.

Ruch PTP jest wyłączany przez zaprogramowanie modalnego G-Code CP.

Różne transformacje są zawarte w druku:

/FB3/ Podręcznik działania Funkcje specjalne; Pakiet transformacji Handling (TE4).

Programowanie położenia (STAT=)

Położenie maszyny nie jest jednoznacznie określone tylko przez podanie pozycji w układzie współrzędnych kartezyjskich i orientacji narzędzia. W zależności od tego o jaką kinematykę chodzi, istnieje do 8 różnych wzgl. rozróżnianych położeń przegubu. Są one przez to specyficzne dla transformacji. Aby pozycję kartezyjską móc jednoznacznie przeliczyć na kąt osi, położenie przegubów musi zostać podane przy pomocy polecenia $STAT=$. Polecenie "STAT" zawiera jeden bit jako wartość binarną dla każdego z możliwych położeń.

Bity położenia, które należy programować przy "STAT", patrz:

/FB2/ Podręcznik działania Funkcje rozszerzające; Transformacja kinematyczna (M1), punkt "Ruch kartezyjski PTP".

Programowanie kątów osi (TU=)

Aby móc jednoznacznie wykonać ruch do kąta osi $< \pm 360$ stopni, informacja ta musi zostać zaprogramowana poleceniem "TU=".

Osie wykonują ruch po najkrótszej drodze:

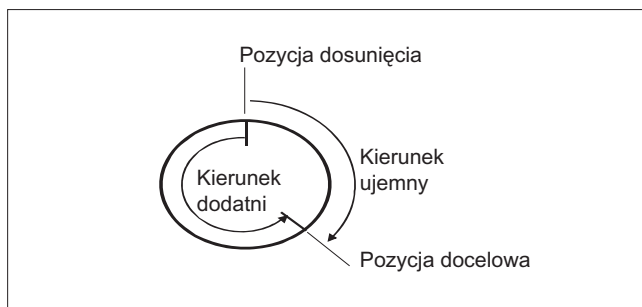
- gdy w przypadku pozycji nie zostanie zaprogramowane TU,
- w przypadku osi, które mają zakres ruchu $> \pm 360$ stopni.

Przykład:

Ruch do pozycji podanej na rysunku może nastąpić w kierunku ujemnym albo dodatnim. Pod adresem A1 jest programowany kierunek.

A1=225°, TU=Bit 0, → kierunek dodatni

A1=-135°, TU=Bit 1, → kierunek ujemny



Przykład Ewaluacja TU dla rodzajowej transformacji 5/6-osiowej i pozycje docelowe

Zmienna TU zawiera dla każdej osi, która wchodzi do transformacji, jeden bit, który sygnalizuje kierunek ruchu. Przyporządkowanie bitów TU odpowiada widokowi osi obrotowych w kanale. Informacja TU jest poddawana ewaluacji tylko dla max 3 możliwych osi obrotowych, które wchodzi do transformacji:

Bit0: oś 1, bit $TU = 0$: 0 stopni \leq kąt osi obrotowej < 360 stopni

Bit1: oś 2, bit $TU = 1$: -360 stopni $<$ kąt osi obrotowej < 0 stopni

Pozycją startową osi obrotowej jest $C = 0$, przez zaprogramowanie $C = 270$ oś obrotowa wykonuje ruch do następujących pozycji docelowych:

$C = 270$: bit $TU = 0$, dodatni kierunek obrotu

$C = -90$: bit $TU = 1$, ujemny kierunek obrotu

Dalsze zachowanie się

Zmiana trybu pracy

Funkcja "ruch kartezjański PTP" ma sens tylko w rodzajach pracy AUTO i MDA. Przy zmianie trybu pracy na JOG aktualne ustawienie pozostaje zachowane.

Gdy jest ustawiony G-Code PTP , osie wykonują ruch w MKS. Gdy jest ustawiony G-Code CP , osie wykonują ruch w WKS.

Power On/RESET

Po Power On albo po Reset ustawienie jest zależne od danej maszynowej $\$MC_GCODE_RESET_VALUES[48]$. Standardowo jest ustawiony rodzaj ruchu "CP".

REPOS

Jeżeli podczas bloku przerwania była ustawiona funkcja "ruch kartezjański PTP", również z PTP podlega pozycjonowaniu przywracającemu.

Ruchy nałożone

Przesunięcie DRF albo zewnętrzne przesunięcie punktu zerowego są w przypadku ruchu kartezjańskiego PTP możliwe tylko z ograniczeniem. Przy zmianie z ruchu PTP na ruch CP, w BKS nie może być żadnych nałożeń.

Ścięcie między ruchami CP i PTP

Między blokami jest przy pomocy `G641` możliwe programowane ścięcia przejścia.

Wielkość obszaru ścięcia jest to droga po torze w mm albo calach, od której wzgl. do której przejście między blokami jest ścinane. Wielkość należy podawać następująco:

- dla bloków G0 przy pomocy `ADISPOS`
- dla wszystkich innych poleceń drogowych przy pomocy `ADIS`

Obliczenie drogi po torze odpowiada uwzględnieniu adresów F w przypadku bloków nie G0. Posuw jest dotrzymywany w osiach podanych w `FGROUP(...)`.

Obliczenie posuwu

Dla bloków CP są do obliczenia stosowane osie kartezjańskie bazowego układu współrzędnych.

Dla bloków PTP są stosowane do obliczenia odpowiednie osie układu współrzędnych maszyny.

6.9.1 PTP przy TRANSMIT**Funkcja**

Przy pomocy PTP w przypadku TRANSMIT następuje optymalny pod względem czasu ruch do bloków G0 i G1. Zamiast liniowego ruchu osi w bazowym układzie współrzędnych (CP), ruch liniowy wykonują osie maszyny (PTP). Przez to przebieg osi maszyny w pobliżu bieguna ma taki wpływ, że punkt końcowy bloku może zostać znacznie szybciej uzyskany.

Program obróbki jest nadal pisany w kartezjańskim układzie współrzędnych obrabianego przedmiotu i wszystkie przesunięcia współrzędnych, obroty i programowania frame pozostają obowiązujące. Symulacja na HMI jest również wyświetlana w kartezjańskim układzie współrzędnych obrabianego przedmiotu.

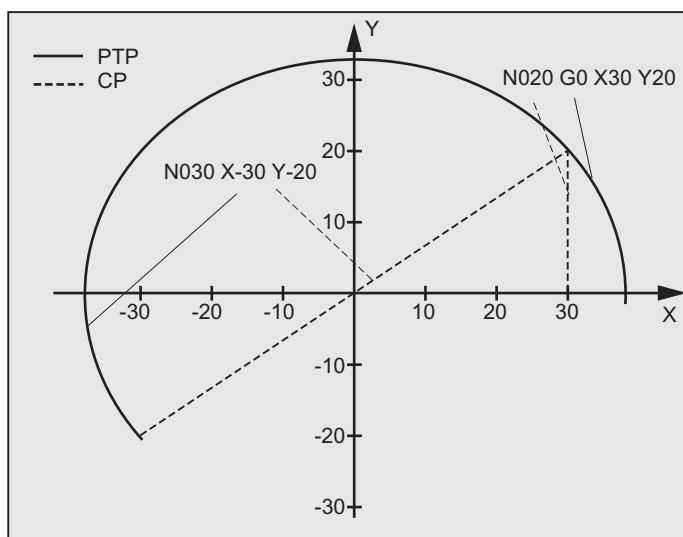
Składnia

```
N... TRANSMIT
N... PTPG0
N... G0 ...
...
N... G1 ...
```

Znaczenie

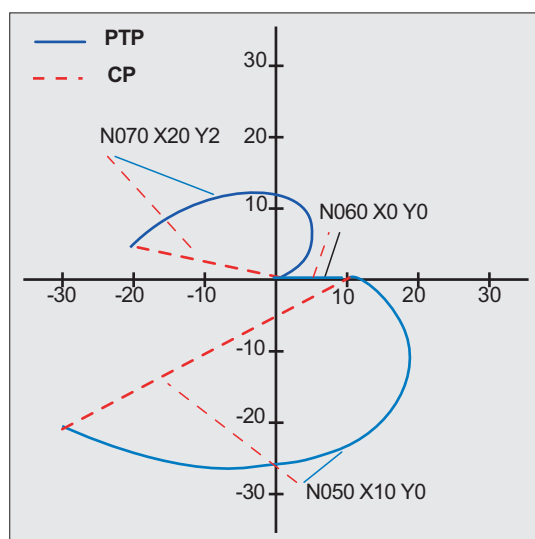
TRANSMIT	Uaktywia pierwszą uzgodnioną funkcję TRANSMIT (patrz punkt "Obróbki frezarskie na częściach toczonych: TRANSMIT") PTPG0
Point to Point G0	(ruch punkt do punktu automatycznie do każdego bloku G0, a następnie ponownie ustawienie CP) Ponieważ STAT i TU są modalne, obowiązuje zawsze ostatnia zaprogramowana wartość.
PTP	Point to Point (ruch punkt do punktu) Dla TRANSMIT, PTP oznacza, że przejście ze spirali kartezjańskiej na spiralę Archimedeasa następuje albo wokół bieguna albo z bieguna. Wynikające stąd ruchy narzędzia przebiegają wyraźnie inaczej, niż w przypadku CP i są przedstawione w poszczególnych przykładach programowania.
STAT=	Rozwiązanie wieloznaczności odnośnie bieguna.
TU=	TU jest bez znaczenia w przypadku PTP przy TRANSMIT

Przykład, obejście bieguna przy pomocy PTP i TRANSMIT



Kod programu	Komentarz
N001 G0 X30 Z0 F10000 T1 D1 G90	; Położenie wyjściowe, wymiar absolutny
N002 SPOS=0	
N003 TRANSMIT	; Transformacja TRANSMIT
N010 PTPG0	; Do każdego bloku G0 automatycznie PTP, a następnie ponownie CP
N020 G0 X30 Y20	
N030 X-30 Y-20	
N120 G1 X30 Y20	
N110 X30 Y0	
M30	

Przykład, wyjście z bieguna przy pomocy PTP i TRANSMIT



Programowanie	Komentarz
N001 G0 X90 Z0 F10000 T1 D1 G90	; Położenie wyjściowe
N002 SPOS=0	
N003 TRANSMIT	; Transformacja TRANSMIT
N010 PTPG0	; Do każdego bloku G0 automatycznie PTP, a następnie ponownie CP
N020 G0 X90 Y60	
N030 X-90 Y-60	
N040 X-30 Y-20	
N050 X10 Y0	
N060 X0 Y0	
N070 X-20 Y2	
N170 G1 X0 Y0	
N160 X10 Y0	
N150 X-30 Y-20	
M30	

Opis

PTP i PTPG0

PTPG0 jest uwzględniane przy wszystkich transformacjach, które mogą wykonywać PTP. We wszystkich innych przypadkach PTPG0 jest bez znaczenia.

Bloki G0 są wykonywane w trybie CP.

Wybór PTP wzgl. PTPG0 następuje w programie obróbki albo przez cofnięcie wyboru CP w danej maszynowej \$MC_GCODE_RESET_VALUES[48].

**OSTROŻNIE****Warunki brzegowe**

Oдноśnie ruchów narzędzia i kolizji obowiązuje wiele warunków brzegowych i określone wykluczenia funkcji jak:

Z PTP nie może być aktywna żadna korekcja promienia narzędzia (WRK).

Z PTPG0 następuje ruch przez CP przy aktywnej korekcji promienia narzędzia (WRK).

Z PTP miękkie dosunięcie i odsunięcie (WAB) jest niemożliwe.

Z PTPG0 następuje ruch przez CP z miękkim dosunięciem i odsunięciem (WAB).

Z PTP cykle skrawania warstwowego (CONTPRON, CONTDCON) są niemożliwe.

Przy pomocy PTPG0 jest w cyklach skrawania warstwowego (CONTPRON, CONTDCON) wykonywany ruch przez CP.

Faza (CHF, CHR) i zaokrąglenie (RND, RNDM) są ignorowane.

Kompresor nie toleruje się z PTP i jego wybór jest automatycznie cofany w blokach PTP.

Nalożenie osi w interpolacji nie może się zmienić podczas sekcji PTP.

Przy G643 następuje automatyczne przełączenie na ścinanie z dokładnością osiową G642.

Przy aktywnym PTP osie transformacji nie mogą być równocześnie osiami pozycjonowania.

Literatura:

/FB2/ Podręcznik działania Funkcje rozszerzające; Transformacja kinematyczna (M1), punkt "Ruch kartezjański PTP"

PTP przy TRACON:

PTP można używać również z TRACON, gdy pierwsza powiązana transformacja obsługuje PTP.

Znaczenie STAT= i TU= przy TRANSMIT

Gdy oś obrotowa ma się obrócić o 180 stopni wzgl. kontur przy CP prowadzić przez biegun, osie obrotowe mogą zależnie od danej maszynowej \$MC_TRANSMIT_POLE_SIDE_FIX_1/2 [48] wykonać obrót o +/- 180 stopni i wykonać ruch zgodnie lub przeciwnie do ruchu wskazówek zegara. Również można ustawić, czy ruch jest wykonywany przez biegun, czy następuje obrót wokół bieguna.

6.10 Warunki brzegowe przy wyborze transformacji

Funkcja

Wybór transformacji jest możliwy przez program obróbki lub MDA. Należy przy tym uwzględnić:

- Nie jest wstawiany blok pośredni ruchu (fazy/zaokrąglenia).
- Ciąg bloków spline musi być zakończony; jeżeli nie, ukazuje się komunikat.
- Musi być cofnięty wybór korekcji dokładnej narzędzia (FTOCOF); jeżeli nie, ukazuje się komunikat.
- Musi być cofnięty wybór korekcji promienia narzędzia (G40); jeżeli nie, ukazuje się komunikat.
- Uaktywniona korekcja długości narzędzia jest przejmowana przez sterowanie do transformacji.
- Sterowanie cofa wybór aktualnego frame, działającego przed transformacją.
- Aktywne ograniczenie obszaru pracy jest dla osi objętych transformacją cofane przez sterowanie (odpowiada WALIMOF).
- Nadzór obszaru ochrony jest cofany.
- Praca z przechodzeniem płynnym i ścinanie narożników są przerywane.
- Wszystkie osie podane w danej maszynowej muszą być zsynchronizowane w odniesieniu do bloku.
- Zamienione osie są zamieniane z powrotem, jeżeli nie, ukazuje się komunikat.
- W przypadku osi zależnych jest wyprowadzany komunikat.

Wymiana narzędzia

Wymiana narzędzia jest dopuszczalna tylko przy cofniętym wyborze korekcji promienia narzędzia.

Zmiana korekcji długości narzędzia i wybór/cofnięcie wyboru korekcji promienia narzędzia nie mogą być zaprogramowane w tym samym bloku.

Zmiana frame

Wszystkie instrukcje, które odnoszą się tylko do bazowego układu współrzędnych, są dozwolone (FRAME, korekcja promienia narzędzia). Zmiana frame w przypadku G91 (wymiar przyrostowy) nie jest jednak - inaczej niż przy nie aktywnej transformacji - traktowana oddzielnie. Będący do przebycia przyrost jest poddawany ewaluacji w układzie współrzędnych obrabianego przedmiotu - niezależnie od tego, jaki frame działał w poprzedzającym bloku.

Wykluczenia

Osie objęte transformacją nie mogą być stosowane:

- jako oś preset (alarm),
- do dosunięcia do punktu stałego (alarm),
- do bazowania (alarm).

6.11 Cofnięcie wyboru transformacji (TRAFOOF)

Funkcja

Przy pomocy polecenia `TRAFOOF` są wyłączane wszystkie aktywne transformacje i frame.

Wskazówka

Aktywność później potrzebnych frame musi zostać włączona przez ponowne zaprogramowanie.

Należy przy tym uwzględnić:

Dla cofnięcia wyboru transformacji obowiązują te same warunki brzegowe co dla wyboru (patrz punkt "Warunki brzegowe przy wyborze transformacji").

Składnia

`TRAFOOF`

Znaczenie

`TRAFOOF` Polecenie wyłączenia wszystkich aktywnych transformacji/frame

6.12 Transformacje powiązane (TRACON, TRAFOOF)

Funkcja

Każdorazowo **dwie** transformacje mogą być połączone jedna za drugą (powiązane), tak że składowe ruchu dla osi z pierwszej transformacji są danymi początkowymi dla powiązanej drugiej transformacji. Składowe ruchu z drugiej transformacji działają na osie maszyny.

Łańcuch może obejmować **dwie** transformacje

Wskazówka

Narzędzie jest zawsze przyporządkowywane do pierwszej transformacji łańcucha. Następna transformacja zachowuje się wówczas tak, jakby aktywna długość narzędzia wynosiła zero. Działają tylko ustawione przez dane maszynowe długości bazowe narzędzia (`_BASE_TOOL_`) dla pierwszej transformacji łańcucha.

Producent maszyny

Proszę przestrzegać wskazówek producenta maszyny i ew. transformacji predefiniowanych przez dane maszynowe.

Transformacje i transformacje powiązane są opcjami. Na temat dostępności określonych transformacji w łańcuchu w określonych sterowaniach informuje każdorazowo aktualny katalog.

Zastosowania

- Szlifowanie konturów, które zostały zaprogramowane jako tworząca rozwinięcia walca (TRACYL) przy pomocy ściernicy ustawionej skośnie, np. szlifowanie narzędzi.
- Obróbka dokładna konturu nieokrągłego utworzonego przy pomocy TRANSMIT przy pomocy ściernicy ustawionej skośnie.

Składnia

TRACON(trf,par) Jest włączana transformacja powiązana.
TRAF00F

Znaczenie

TRACON	Transformacja powiązana jest włączana. Uaktywniona przedtem inna transformacja jest przez TRACON() implicite wyłączana.
TRAF00F	Ostatnio włączona (powiązana) transformacja jest wyłączana.
trf	Numer powiązanej transformacji: 0 albo 1 dla pierwszej/jedynej powiązanej transformacji. Jeżeli w tym miejscu nic nie zaprogramowano, jest to równoznaczne z podaniem wartości 0 albo 1, tzn. jest uaktywniana pierwsza/jedyna transformacja. 2 dla drugiej powiązanej transformacji. (wartości nierówne 0 - 2 wytwarzają komunikat błędu).
par	Jeden albo wiele rozdzielonych przecinkiem parametrów transformacji w powiązaniu, które oczekują parametrów, np. kąt osi skośnej. Przy nie ustawionych parametrach działają ustawienia domyślne albo parametry ostatnio używane. Przez postawienie przecinków należy zadbać o to, by podane parametry były poddawane ewaluacji w takiej kolejności, w jakiej są oczekiwane, jeżeli dla przedtem umieszczonych parametrów mają działać ustawienia domyślne. W szczególności przy podaniu co najmniej jednego parametru musi przed nim znajdować się przecinek, również gdy podanie trf nie jest konieczne, a więc na przykład TRACON(, 3.7).

Warunek

Drugą transformacją musi być "oś skośna" (TRAANG). Jako pierwsza transformacja są możliwe:

- Transformacje orientacji (TRAORI), łącznie z głowicą frezarską Kardana
- TRANSMIT
- TRACYL
- TRAANG

Dla użycia polecenia włączenia transformacji powiązanej warunkiem jest, by poszczególne transformacje do powiązania i będąca do uaktywnienia transformacja powiązana były zdefiniowane przez dane maszynowe.

Warunków brzegowych i przypadków specjalnych podanych w poszczególnych opisach transformacji należy przestrzegać również w ramach powiązania.

Informacje dot. projektowania danych maszynowych transformacji można znaleźć w:

/FB2/ Podręcznik działania Funkcje rozszerzające; Transformacje kinematyczne (M1) i

/FB3/ Podręcznik działania Funkcje specjalne; Transformacje 3- do 5-osiowych (F2).

Łańcuchy kinematyczne

7.1 Usunięcie komponentów (DELOBJ)

Funkcja

Za pomocą funkcji DELOBJ następujące komponenty mogą zostać "usunięte":

- Elementy łańcuchów kinematycznych
- Obszary ochrony oraz elementy obszaru ochrony dla geometrycznego modelowania maszyn
- pary kolizyjne
- dane transformacji

W tym celu wszystkie zmienne systemowe komponentów zostaną ustawione na wartości domyślnej.

Usunięcie jednego komponentu nie działa od razu, ale dopiero po nowym przeliczeniu modelu, w którym te komponenty są stosowane.

Przykład:

Model: Model funkcji unikania kolizji

Komponenty usunięte: Element

Usunięcie będzie skuteczne dopiero po ponownym żądaniu przeliczenia modelu kolizji przez PROTA.

Składnia

```
[<RetVal>=] DELOBJ (<CompType> [ , , , <NoAlarm> ] )
[<RetVal>=] DELOBJ (<CompType> , <Index1> [ , , <NoAlarm> ] )
[<RetVal>=] DELOBJ (<CompType> [ , <Index1> ] [ , <Index2> ] [ , <NoAlarm> ] )
```

Znaczenie

DELOBJ:	Usuwanie elementów łańcuchów kinematycznych, obszarów ochrony oraz elementów obszarów ochrony geometrycznego modelowania maszyn, par kolizyjnych i danych transformacji		
<RetVal>:	Zwracana wartość funkcji		
	Typ danych:	INT	
	Zakres wartości:	0, -1, -2, ... -7	
	Wartość:	0	Żaden błąd nie wystąpił.
		-1	Wywołanie funkcji bez parametrów. Min. musi zostać podany parametr <CompType>.
		-2	<CompType> oznacza nieznanne komponenty
		-3	<Index1> jest mniejszy niż -1
		-4	<Index1> jest większy niż zaprojektowana liczba komponentów
		-5	<Index1> przy usuwaniu grupy komponentów ma wartość różną od -1
-6		<Index2> jest mniejszy niż <Index1>	
-7	<Index2> jest większy niż zaprojektowana liczba komponentów		
<CompType>:	Rodzaj komponentów do usunięcia		
	Typ danych:	STRING	
	Wartość:	"KIN_CHAIN_ELEM"	Usuwa wszystkie zmienne systemowe: <ul style="list-style-type: none"> • \$NK_...
		"PROT_AREA"	Usuwa zmienne systemowe: <ul style="list-style-type: none"> • \$NP_PROT_NAME • \$NP_CHAIN_NAME • \$NP_CHAIN_ELEM • \$NP_1ST_PROT
		"PROT_AREA_ELEM"	Usuwa zmienne systemowe: <ul style="list-style-type: none"> • \$NP_NAME • \$NP_NEXT • \$NP_ADD • \$NP_TYPE • \$NP_PARA • \$NP_OFF • \$NP_DIR • \$NP_ANG
"PROT_AREA_COLL_PAIRS"		Usuwa zmienne systemowe: <ul style="list-style-type: none"> • \$NP_COLL_PAIR • \$NP_SAFETY_DIST 	

	"PROT_AREA_ALL"	Usuwa wszystkie zmienne systemowe: • \$NP_...	
	"TRAFO_DATA"	Usuwa wszystkie zmienne systemowe: • \$NT_...	
	"TRAFO_OBJS"	Usuwa wszystkie aktualnie niekatywne transformacje kinematyczne. Parametry <Index1> i <Index2> nie są uwzględniane.	
<Index1>:	Indeks Start dla usuwanych komponentów (opcjonalnie)		
	Typ danych:	INT	
	Zakres wartości:	$-1 \leq x \leq$ (maksymalna liczba zaprojektowanych komponentów - 1)	
	Wartość:	-1	Wszystkie komponenty podanego rodzaju zostaną usunięte. <Index2> nie jest uwzględniany.
		> -1	Indeks (pierwszych) usuwanych komponentów
Wartość domyślna:	-1		
<Index2>:	Indeks Koniec dla usuwanych komponentów (opcjonalnie) Jeśli nie zaprogramowano <Index2> , usunięte zostaną tylko zmienne systemowe bazowanych w <Index1> komponentów		
	Typ danych:	INT	
	Zakres wartości:	$\langle \text{Index1} \rangle \leq x \leq$ (maksymalna liczba zaprojektowanych komponentów - 1)	
	Wartość domyślna:	Tylko zmienne systemowe komponentów bazowanych w <Index1> zostaną usunięte.	
	<NoAlarm>:	Wyłączenie alarmu (opcjonalnie)	
	Typ danych:	BOOL	
	Wartość:	FALSE	W przypadku błędu (<RetVal> < 0) obróbka programowa zostanie zatrzymana i wyświetli się alarm.
		TRUE	W przypadku błędu obróbka programowa nie zostanie zatrzymana i nie wyświetli się żaden alarm. Przypadek zastosowania: Specyficzna dla użytkownika reakcja odpowiednio do wartości zwrotnej
	Wartość domyślna:	FALSE	

Literatura

Podręcznik działania Funkcje specjalne; Rozdział K7: Łańcuchy kinematyczne

7.2 Określanie indeksu poprzez nazwy (NAMETOINT)

Funkcja

W polach zmiennych systemowych typu STRING wprowadzono specyficzne dla użytkownika łańcuchy znaków lub nazw. Jeżeli identyfikator zmiennych systemowych oraz nazwa są znane, funkcja NAMETOINT za pomocą nazwy określa przynależny do zmiennej systemowej indeks.

Składnia

<RetVal>=NAMETOINT (<SysVar>, <Name> [, <NoAlarm>])

Znaczenie

NAMETOINT:	Określanie indeksu zmiennych systemowych			
<RetVal>:	Zwracana wartość funkcji			
	Typ danych:	INT		
	Zakres wartości:	-1 ≤ x ≤ (maks. liczba zaprojektowanych komponentów -1)		
	Wartość:	-1	Nie znaleziono szukanej nazwy lub wystąpił błąd.	
≥ 0		Indeks zmiennych systemowych		
<SysVar>:	Nazwa pola zmiennych systemowych typu STRING			
	Typ danych:	STRING		
	Zakres wartości:	Nazwy wszystkich pól zmiennych systemowych NC typu STRING		
<Name>:	Łańcuch znaków lub nazwa, dla których powinien zostać określony indeks zmiennych systemowych.			
	Typ danych:	STRING		
<NoAlarm>:	Wyłączenie alarmu (opcjonalnie)			
	Typ danych:	BOOL		
	Wartość:	FALSE	W przypadku błędu (<RetVal> < 0) obróbka programowa zostanie zatrzymana i wyświetli się alarm.	
		TRUE	W przypadku błędu obróbka programowa nie zostanie zatrzymana i nie wyświetli się żaden alarm. Przypadek zastosowania: Specyficzna dla użytkownika reakcja odpowiednio do wartości zwrotnej	

Przykład

Kod programu	Komentarz
DEF INT INDEX	
\$NP_PROT_NAME [27] = "Osłona"	
...	
INDEX=NAMETOINT (" \$NP_PROT_NAME ", "Osłona")	; INDEX==27

Literatura

Podręcznik działania Funkcje specjalne; Rozdział K7: Łańcuchy kinematyczne

Uniknięcie kolizji za pomocą łańcuchów kinematycznych

8

Wskazówka

Obszary ochrony

Podane w następnych rozdziałach obszary ochrony odnoszą się do funkcji "Geometryczne modelowanie maszyny"

Literatur:

Podręcznik działania Funkcje specjalne, Rozdział "Geometryczne modelowanie maszyny"

8.1 Sprawdzenie pod kątem pary kolizyjnej (COLLPAIR)

Funkcja

Przy unikaniu kolizji za pomocą łańcuchów kinematycznych za pomocą funkcji COLLPAIR można określić, czy dwa obszary ochrony tworzą parę kolizyjną tzn. czy są kontrolowane pod kątem kolizji, gdy oba są aktywne.

Składnia

```
[<RetVal>=] COLLPAIR (<Name_1>, <Name_2> [, <NoAlarm>])
```

Znaczenie

COLLPAIR:	Kontrola pod kątem przynależności do pary kolizyjnej		
<RetVal>:	Zwracana wartość funkcji		
	Typ danych:	INT	
	Wartość:	≥ 0	Oba obszary ochronne tworzą jedną parę kolizyjną. Wartość zwrotna == Indeks pary kolizyjnej m
		-1	Albo podano mniej niż dwa łańcuchy, albo minimum jeden z nich stanowi łańcuch zerowy.
		-2	Podany w pierwszym parametrze obszar ochrony nie został znaleziony.
		-3	Podany w drugim parametrze obszar ochrony nie został znaleziony.
		-4	Żaden z obu podanych obszarów ochrony nie został znaleziony.
-5		Oba podane obszary ochrony zostały znalezione, ale nie tworzą one jednej pary kolizyjnej.	

<Name_1>:	Nazwa pierwszego obszaru ochrony		
	Typ danych:	STRING	
	Zakres wartości:	Sparametryzowane nazwy obszarów ochrony	
<Name_2>:	Nazwa drugiego obszaru ochrony		
	Typ danych:	STRING	
	Zakres wartości:	Sparametryzowane nazwy obszarów ochrony	
<NoAlarm>:	Wyłączenie alarmu (opcjonalnie)		
	Typ danych:	BOOL	
	Wartość:	FALSE	W przypadku błędu (<RetVal> < 0) obróbka programowa zostanie zatrzymana i wyświetli się alarm.
		TRUE	W przypadku błędu obróbka programowa nie zostanie zatrzymana i nie wyświetli się żaden alarm. Przypadek zastosowania: Specyficzna dla użytkownika reakcja odpowiednio do wartości zwrotnej
	Wartość domyślna:	FALSE	

Literatura

Podręcznik działania Funkcje specjalne; Rozdział K9: Unikanie kolizji

8.2 Żądanie nowego obliczenia modelu kolizji (PROTA)

Funkcja

Poprzez zapis parametrów systemowych dla łańcuchów kinematycznych, geometryczne modelowanie maszyny i/lub unikanie kolizji w programie obróbki bądź przez interfejs BTSS model kolizji może zostać zmieniony. Żeby zmiany stały się skuteczne, musi nastąpić żądanie ponownego obliczenia modelu za pomocą procedury PROTA.

Status obszaru ochrony

Poprzez wywołanie procedury PROTA zostanie przeorowane nowe obliczenie modelu kolizji **bez** zmiany obszaru ochrony. **Nowo**zdefiniowany obszar ochrony posiada potem status inicjalizacji. Obszar ochrony, którego nazwa została zmieniona **nie**obowiązuje jako nowo zdefiniowany. Jego status pozostaje niezmieniony.

Poprzez wywołanie procedury PROTA za pomocą parametrów "R" zostanie przeorowane nowe obliczenie modelu kolizji **ze** zmianą statusu obszaru ochrony. Przy tym wszystkie obszary ochrony zostaną ponownie przełączone do statusu inicjalizacji odpowiednio do \$NP_INIT_STAT.

Wskazówka

Wymiana narzędzia oraz automatycznie stworzone obszary ochrony narzędzia

Po wymianie narzędzia w przypadku stworzonych automatycznie obszarów ochrony narzędzia model kolizji jest zawsze domyślnie od nowa obliczany.

Składnia

PROTA
 PROTA (<Par>)

Znaczenie

PROTA:	Żądanie nowego obliczenia modelu kolizji	
	Zatrzymanie przebiegu wyprzedzającego:	tak
<Par>:	Parametry	
	Typ danych:	STRING
	Zakres wartości:	"R" Po nowym obliczeniu modelu kolizji wszystkie obszary ochrony zostaną ponownie przełączone do statusu inicjalizacji.

Warunki brzegowe

Symulacja

Procedura PROTA nie może być stosowana w programach obróbki w związku z symulacją (simNCK).

Przykład:

Kod programu	Komentarz
...	
IF \$P_SIM == FALSE	;IF nieaktywna symulacja
PROTA	;THEN nowe obliczenie modelu kolizji
ENDIF	;ENDIF
...	

Literatura

Podręcznik działania Funkcje specjalne; Rozdział K9: Unikanie kolizji

8.3 Ustawienie stanu obszaru ochrony (PROTS)

Funkcja

Przy unikaniu kolizji za pomocą łańcuchów kinematycznych za pomocą procedury PROTS można ustawić stan obszarów ochrony.

Składnia

```
PROTS (<Status>)
PROTS (<Status>, <Name_1>)
PROTS (<Status>, <Name_1>, ..., <Name_n>)
```

Znaczenie

PROTS:	Ustawienie stanu obszaru ochrony			
	Zatrzymanie przebiegu wyprzedzającego:	nie		
	W oddzielnym bloku:	tak		
<State>:	Status, w jakim należy ustawić podane obszary ochrony			
	Typ danych:	CHAR		
	Wartość:	"A" lub "a"	Status: Aktywny	
		"I" lub "i"	Status: Nieaktywny	
		"P" lub "p"	Status: Wstępnie aktywowany (Aktywacja następuje poprzez interfejs NC/PLC)	
"R" lub "r"		Status = \$NP_INIT_STAT (status inicjalizacji)		
<Name_1> ... <Name_n>:	Nazwa jednego lub kilku obszarów ochrony, jakie należy ustawić dla podanego statusu			
	Uwaga: Maksymalna liczba obszarów ochrony, które mogą zostać podane jako parametry, jest zależny tylko od maksymalnie możliwej liczby znaków przypadającej na wiersz programu.			
	Typ danych:	STRING		
	Zakres wartości:	Sparametryzowane nazwy obszarów ochrony		
		Uwaga: Jeżeli nie podano nazwy, podany status zostanie ustawiony dla wszystkich zdefiniowanych obszarów ochrony.		

Literatura

Podręcznik działania Funkcje specjalne; Rozdział K9: Unikanie kolizji

8.4 Określenie odległości dwóch obszarów ochrony (PROTD)

Funkcja

Przy unikaniu kolizji za pomocą łańcuchów kinematycznych za pomocą procedury PROTD można obliczyć odległość od dwóch obszarów ochrony.

Właściwości funkcji:

- Obliczenie odległości następuje niezależnie od statusu obszarów ochrony (aktywny, nieaktywny, wstępnie aktywowany).
- Obliczenie odległości następuje za pomocą pozycji obowiązujących na końcu bloku poprzedzającego.
- Nałożenia, które zostaną uwzględnione w przebiegu głównym (np. przesunięcie DRF albo ekstremalne przesunięcia punktu zerowego) zapisują się wraz z wartościami obowiązującymi do **momentu interpretacji** funkcji w obliczeniu odległości.

Wskazówka

Synchronizacja

Przy stosowaniu funkcji PROTD wyłącznie w gestii użytkownika leży synchronizacja przebiegu głównego oraz poprzedzającego wykonana ewentualnie za pomocą zatrzymania przebiegu poprzedzającego STOPRE.

Kolizja

Jeśli zachodzi kolizja pomiędzy podanymi obszarami ochrony, funkcja podaje odległość 0,0. Kolizja zachodzi gdy odległość wynosi:

$0,0 < \text{odległość} < \text{MD10619 } \$\text{MN_COLLISION_TOLERANCE}$ (tolerancja dla kontroli kolizji)

Odstęp bezpieczeństwa dla kontroli kolizji (MD10622 \$MN_COLLISION_SAFETY_DIST) nie jest uwzględniany podczas obliczania odległości:

Składnia

```
[<RetVal>=] PROTD (<Name_1>, <Name_2>, VAR <Vector> [, <System>])  
[<RetVal>=] PROTD (, , VAR <Vector> [, <System>])
```

Znaczenie

PROTD:	Oblicza odległość dwóch podanych obszarów ochrony:		
	Zatrzymanie przebiegu wyprzedzającego:	nie	
	W oddzielnym bloku:	tak	
<RetVal>:	Zwracana wartość funkcji: Wartość absolutna odległości obu obszarów ochrony lub 0,0 w przypadku kolizji (patrz wyżej: akapit Kolizja)		
	Typ danych:	REAL	
	Zakres wartości:	$0.0 \leq x \leq +\text{maks. wartość REAL}$	
<Name_1> , <Name_2>:	Nazwy obu obszarów ochrony, których odległość powinna zostać obliczona. (opcjonalnie)		
	Typ danych:	STRING	
	Zakres wartości:	Sparametryzowane nazwy obszarów ochrony	
	Wartość domyślna:	Jeżeli nie podano żadnych obszarów ochrony, funkcja oblicza aktualną odległość najmniejszą ze wszystkich zawartych w modelu kolizji uaktywnionych bądź wstępnie aktywowanych obszarów ochrony.	
<Vector>:	Wartość zwrotna: 3-wymiarowy wektor odstępu od obszaru ochrony <Name_2> do obszaru ochrony <Name_1> z:		
	<ul style="list-style-type: none"> • <Vector>[0]: współrzędna X w układzie współrzędnych kartezjańskich • <Vector>[1]: współrzędna Y w układzie współrzędnych kartezjańskich • <Vector>[2]: współrzędna Z w układzie współrzędnych kartezjańskich W przypadku kolizji: <Vector> == Wektor zerowy		
	Typ danych:	VAR REAL[3]	
	Zakres wartości:	<Vector>[n]: $0,0 \leq x \leq +\text{maks. wartość REAL}$	
<System>:	System miar (calowy/metryczny) dla odstępu i wektora odstępu (opcjonalnie)		
	Typ danych:	BOOL	
	Wartość:	FALSE	System miar odpowiednio do aktualnie aktywnych funkcji G z grupy G 13 (G70, G71, G700, G710).
		TRUE	System miar odpowiednio do ustawionego systemu podstawowego: MD52806 \$MN_ISO_SCALING_SYSTEM
	Wartość domyślna:	FALSE	

Literatura

Podręcznik działania Funkcje specjalne; Rozdział K9: Unikanie kolizji

Korekcje narzędzi

9.1 Pamięć korekcji

Funkcja

Budowa korekcji pamięci

Każde pole danych można wywołać przy pomocy numeru T i numeru D (oprócz "płaskiego nr D") i zawiera ono oprócz danych geometrycznych dla narzędzia jeszcze dalsze wpisy, np. typ narzędzia.

Płaska struktura numerów D

"Płaska struktura numerów D" jest stosowana, gdy zarządzanie narzędziami następuje poza NCK. W tym przypadku numery D z przynależnymi zestawami korekt narzędzia są tworzone bez przyporządkowania do narzędzi

W programie obróbki można dalej programować T. To T nie ma jednak odniesienia do zaprogramowanego numeru D.

Dane ostrzy nadawane przez użytkownika

Przez daną maszynową mogą być konfigurowane dane użytkownika dot. ostrza. Proszę przestrzegać danych producenta maszyny.

Parametry narzędzia

Wskazówka

Poszczególne wartości w pamięci korekcji

Poszczególne wartości pamięci korekcji P1 do P25 dają się czytać i zapisywać z programu przez zmienne systemowe. Wszystkie pozostałe parametry są zarezerwowane.

Parametry narzędzi \$TC_DP6 do \$TC_DP8, \$TC_DP10 i \$TC_DP11, jak też \$TC_DP15 do \$TC_DP17, \$TC_DP19 i \$TC_DP20 mają w zależności od typu narzędzia inne znaczenie.

Parametr narzędzia numer (DP)	Znaczenie zmiennej systemowej	Wskazówka
\$TC_DP1	Typ narzędzia	Przegląd patrz lista
\$TC_DP2	Położenie ostrza	Tylko dla narzędzi tokarskich

Parametr narzędzia numer (DP)	Znaczenie zmiennej systemowej	Wskazówka
Geometria	Korekcja długości	
\$TC_DP3	Długość 1	Obliczenie według
\$TC_DP4	Długość 2	Typ i płaszczyzna
\$TC_DP5	Długość 3	
Geometria	Promień	
\$TC_DP6 ¹⁾ \$TC_DP6 ²⁾	Promień 1 / długość 1 Średnica d	Narz. frezarskie/tokarskie/szlifierskie Piła do rowków
\$TC_DP7 ¹⁾ \$TC_DP7 ²⁾	Długość 2 / promień naroża frezy stożkowe Szerokość rowka b promień naroża	Narzędzia frezarskie Piła do rowków
\$TC_DP8 ¹⁾ \$TC_DP8 ²⁾	Promień zaokrąglenia 1 dla narzędzi frezarskich Występ k	Narzędzia frezarskie Piła do rowków
\$TC_DP9 ^{1) 3)}	Promień zaokrąglenia 2	Zarezerwowano
\$TC_DP10 ¹⁾	Kąt 1 strona czołowa narzędzia	Stożkowe narzędzia frezarskie
\$TC_DP11 ¹⁾	Kąt 2 oś wzdłużna narzędzia	Stożkowe narzędzia frezarskie
Zużycie	Korekcja długości i promienia	
\$TC_DP12	Długość 1	
\$TC_DP13	Długość 2	
\$TC_DP14	Długość 3	
\$TC_DP15 ¹⁾ \$TC_DP15 ²⁾	Promień 1 / długość 1 Średnica d	Narz. frezarskie/tokarskie/szlifierskie Piła do rowków
\$TC_DP16 ¹⁾ \$TC_DP16 ³⁾	Długość 2 / promień naroża frezy stożkowe Szerokość rowka b promień naroża	Narzędzia frezarskie Piła do rowków
\$TC_DP17 ¹⁾ \$TC_DP17 ²⁾	Promień zaokrąglenia 1 dla narzędzi frezarskich Występ k	Frezowanie / frezowanie czołowe 3D Piła do rowków
\$TC_DP18 ^{1) 3)}	Promień zaokrąglenia 2	zarezerwowano
\$TC_DP19 ¹⁾	Kąt 1 strona czołowa narzędzia	Stożkowe narzędzia frezarskie
\$TC_DP20 ¹⁾	Kąt 2 oś wzdłużna narzędzia	Stożkowe narzędzia frezarskie
Wymiar bazowy/przystawka	Korekcje długości	
\$TC_DP21	Długość 1	
\$TC_DP22	Długość 2	
\$TC_DP23	Długość 3	
Technologia		
\$TC_DP24	Kąt przystawienia	Tylko dla narzędzi tokarskich
\$TC_DP25		zarezerwowano

¹⁾ Obowiązuje również dla narzędzi frezarskich do frezowania czołowego 3D

²⁾ W przypadku piły do rowków typ narzędzia

³⁾ zarezerwowano (nie jest obsługiwane przez SINUMERIK 840D sl)

Uwagi

Dla wielkości geometrycznych (np. długość 1 albo promień) jest wiele wpisywanych składowych. Są one sumowane tworząc wielkość wynikową (np. długość całkowita 1, promień całkowity), która następnie działa.

Niepotrzebne korekcje należy wyposażyć w wartość zero.

Parametry narzędzia \$TC-DP1 do \$TC-DP23 z narzędziami konturowymi

Wskazówka

Parametry narzędzi, które nie są wymienione w tablicy, jak np. \$TC_DP7, nie są poddawane ewaluacji, tzn. ich treść jest bez znaczenia.

Parametr narzędzia numer (DP)	Znaczenie	Ostrza Dn	Wskazówka
\$TC_DP1	Typ narzędzia		400 do 599
\$TC_DP2	Położenie ostrza		
Geometria	Korekcja długości		
\$TC_DP3	Długość 1		
\$TC_DP4	Długość 2		
\$TC_DP5	Długość 3		
Geometria	Promień		
\$TC_DP6	Promień		
Geometria	Kąt graniczny		
\$TC_DP10	Minimalny kąt graniczny		
\$TC_DP11	Maksymalny kąt graniczny		
Zużycie	Korekcja długości i promienia		
\$TC_DP12	Zużycie długość 1		
\$TC_DP13	Zużycie długość 2		
\$TC_DP14	Zużycie długość 3		
\$TC_DP15	Zużycie promień		
Zużycie	Kąt graniczny		
\$TC_DP19	Zużycie min kąt graniczny		
\$TC_DP20	Zużycie max kąt graniczny		
Wymiar bazy/przystawka	Korekcje długości		
\$TC_DP21	Długość 1		
\$TC_DP22	Długość 2		
\$TC_DP23	Długość 3		

Wartość podstawowa i wartość zużycia

Wielkości wynikowe wynikają każdorazowo z sumy z wartości podstawowej i wartości zużycia (np. \$TC_DP6 + \$TC_DP15 dla promienia). Do długości narzędzia pierwszego ostrza jest poza tym jeszcze dodawany wymiar bazowy (\$TC_DP21 – \$TC_DP23). Dodatkowo na tę długość narzędzia działają wszystkie inne wielkości, które również w przypadku zwykłego narzędzia mogą mieć wpływ na jego efektywną długość (przystawka, orientowalny nośnik narzędzi, dane ustawcze).

Kąty graniczne 1 i 2

Kąty graniczne 1 wzgl. 2 odnoszą się każdorazowo do wektora od punktu środkowego ostrza do punktu odniesienia ostrza i są liczone przeciwnie do kierunku ruchu wskazówek zegara.

9.2 Korekcje addytywne

9.2.1 Wybranie korekcji addytywnych (DL)

Funkcja

Korekcje addytywne mogą być traktowane jako korekcje procesu programowalne w czasie obróbki. Odnoszą się one do danych geometrycznych ostrza i są przez to częścią składową danych ostrza narzędzia.

Dostęp do danych korekcji addytywnej jest uzyskiwany przez numer DL (DL: Location dependent; korekcje odnośnie każdorazowego miejsca użycia) i wprowadzane przez interfejs graficzny.

Zastosowanie

Przez korekcje addytywne mogą być wyrównywane błędy uwarunkowane miejscem użycia.

Składnia

DL=<numer>

Znaczenie

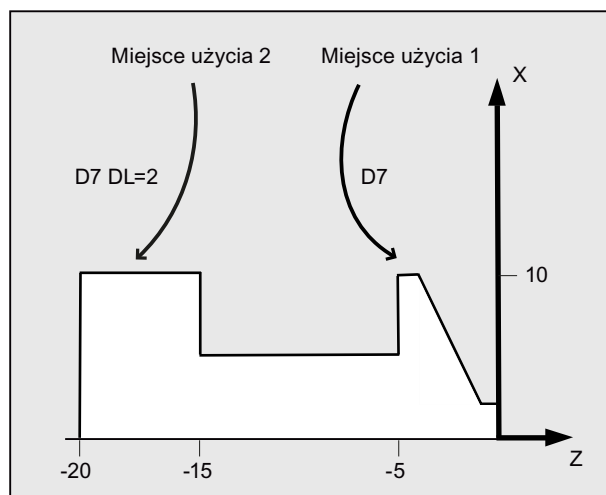
Element	Opis
DL:	Polecenie do uaktywnienia korekcji addytywnej
<numer>:	Poprzez parametr <numer> jest podawany będący do uaktywnienia addytywne zestaw danych korekcyjnych narzędzia.

Wskazówka

Ustalenie liczby i uaktywnienie korekcji addytywnych następuje przez dane maszynowe (→ przestrzegać danych producenta maszyny!).

Przykład

To samo ostrze jest stosowane do dwóch gniazd łożyskowych:



Kod programu	Komentarz
N110 T7 D7	; Głowica rewolwerowa jest pozycjonowana na miejscu 7 D7 i DL=1 są uaktywniane i realizowane w następnym bloku.
N120 G0 X10 Z1	
N130 G1 Z-6	
N140 G0 DL=2 Z-14	; Dodatkowo do D7 jest uaktywniane DL=2 i realizowane w następnym bloku.
N150 G1 Z-21	
N160 G0 X200 Z200	; Ruch do punktu wymiany narzędzia
...	

9.2.2 Ustalenie zużycia i wartości ustawczych (\$TC_SCPxy[t,d], \$TC_ECPxy[t,d])

Funkcja

Wartości zużycia i ustawcze mogą być czytane i zapisywane przez zmienne systemowe. Logika orientuje się przy tym według logiki odpowiednich zmiennych systemowych dla narzędzi i ostrzy.

Zmienne systemowe

Zmienna systemowa	Znaczenie
\$TC_SCPxy[<t>,<d>]	Wartości zużycia, które przez xy są przyporządkowane do każdorazowego parametru geometrycznego, przy czym x odpowiada numerowi wartości zużycia, a y tworzy odniesienie do parametru geometrycznego.
\$TC_ECPxy[<t>,<d>]	Wartości ustawcze, które przez xy są przyporządkowane do każdorazowego parametru geometrycznego, przy czym x odpowiada numerowi wartości ustawczej, a y tworzy odniesienie do parametru geometrycznego.
<t>: numer T narzędzia <d>: numer D ostrza narzędzia	

Wskazówka

Ustalone wartości zużycia i ustawcze są dodawane do parametrów geometrycznych i pozostałych parametrów korekcyjnych (numery D).

Przykład

Wartość zużycia długości 1 dla ostrza <d> narzędzia <t> jest ustalona na wartość 1.0.

Parametr: \$TC_DP3 (długość 1, w przypadku narzędzi tokarskich)

Wartości zużycia: \$TC_SCP13 do \$TC_SCP63

Wartości ustawcze: \$TC_ECP13 bis \$TC_ECP63

\$TC_SCP43 [<t>,<d>] = 1.0

9.2.3 Skasowanie korekcji addytywnych (DELDL)

Funkcja

Przy pomocy polecenia `DELDL` są kasowane addytywne korekcje ostrza narzędzia (zwolnienie pamięci). Są przy tym kasowane zarówno ustalone wartości zużycia, jak też wartości ustawcze.

Składnia

```
DELDL [<t> , <d>]
DELDL [<t>]
DELDL
<Status>=DELDL [<t> , <d>]
```

Znaczenie

DELDL	Polecenie do kasowania korekcji addytywnych
<t>	Numer T narzędzia
<d>	Numer D ostrza narzędzia
DELDL [<t>, <d>]	Są kasowane wszystkie addytywne korekcje ostrza <d> narzędzia <t>.
DELDL [<t>]	Są kasowane wszystkie addytywne korekcje wszystkich ostrzy narzędzia <t>.
DELDL	Są kasowane wszystkie korekcje addytywne wszystkich ostrzy wszystkich narzędzi jednostki TO (dla kanału, w którym zostanie zaprogramowane polecenie).
<status>	Status kasowania
	Wartość Znaczenie:
	ć:
0	Skasowanie zostało przeprowadzone z powodzeniem.
-	Skasowanie nie zostało przeprowadzone (gdy parametryzowanie określa dokładnie jedno ostrze), albo jest niekompletne (gdy parametryzowanie określa wiele ostrzy).

Wskazówka

Wartości zużycia i wartości ustawczych aktywnych narzędzi nie można skasować (zachowuje się analogicznie do zachowania się D wzgl. danych narzędzia przy kasowaniu).

9.3 Traktowanie specjalne korekcji narzędzia**Funkcja**

Przy pomocy danych ustawczych SD42900 do SD42960 można sterować ewaluacją znaków długości narzędzia i zużycia.

Dotyczy to również zachowania się składowych zużycia przy lustrzanym odbiciu osi geometrycznych albo przy zmianie płaszczyzny obróbki, a również do kompensacji temperatury w kierunku narzędzia.

Wartości zużycia

Gdy dalej powołujemy się na wartości zużycia, należy pod tym pojęciem każdorazowo rozumieć sumę właściwych wartości zużycia (\$TC_DP12 do \$TC_DP20) i korekcji sumarycznych z wartościami zużycia (\$SCPX3 do \$SCPX11) i wartościami ustawczymi (\$ECPX3 do \$ECPX11).

Bliższe dane dot. korekcji sumarycznych patrz:

Literatura:

Podręcznik działania Zarządzanie narzędziami

Dane ustawcze

Dana ustawcza	Znaczenie
SD42900 \$SC_MIRROR_TOOL_LENGTH	Lustrzane odbicie komponentów długości narzędzia i komponentów wymiaru bazowego.
SD42910 \$SC_MIRROR_TOOL_WEAR	Lustrzane odbicie wartości zużycia komponentów długości narzędzia.
SD42920 \$SC_WEAR_SIGN_CUTPOS	Reakcja na znak składowej zużycia w zależności od położenia ostrza.
SD42930 \$SC_WEAR_SIGN	Odwraca znaki wymiarów zużycia.
SD42935 \$SC_WEAR_TRANSFORM	Transformacja wartości zużycia.
SD42940 \$SC_TOOL_LENGTH_CONST	Przyporządkowanie komponentów długości narzędzia do osi geometrycznych.
SD42950 \$SC_TOOL_LENGTH_TYPE	Przyporządkowanie komponentów długości narzędzia niezależnie od typu narzędzia.
SD42960 \$SC_TOOL_TEMP_COMP	Wartość kompensacji temperatury w kierunku narzędzia. Działa również przy istniejącej orientacji narzędzia.

Literatura

Podręcznik działania Funkcje podstawowe; Korekcja narzędzia (W1)

Dalsze informacje**Uaktywnienie zmienionych danych ustawczych**

Nowa reakcja na składowe narzędzia przy zmianie opisanych danych ustawczych działa dopiero wtedy, gdy następny raz zostanie wybrane ostrze narzędzia. Jeżeli narzędzie jest już aktywne i ma zacząć działać zmieniona reakcja na dane tego narzędzia, narzędzie to musi zostać wybrane ponownie.

Odpowiednio jest w przypadku, gdy zmienia się wynikowa długość narzędzia, ponieważ zmieniony został stan lustrzanego odbicia osi. Narzędzie musi po poleceniu lustrzanego odbicia zostać ponownie wybrane, aby zaczęły działać zmienione komponenty długości narzędzia.

Orientowalne nośniki narzędzi i nowe dane ustawcze

Dane ustawcze SD42900 do SD42940 nie działają na komponenty ewentualnie aktywnego orientowalnego nośnika narzędzi. Narzędzie wchodzi jednak zawsze ze swoją całą wynikową długością (długość narzędzia + zużycie + wymiar bazowy) do obliczenia z orientowanym nośnikiem narzędzi. Przy obliczaniu wynikającej długości całkowitej są uwzględniane wszystkie zmiany, które zostały spowodowane przez dane ustawcze; tzn. wektory orientowanego nośnika narzędzi są niezależne od płaszczyzny obróbki.

Wskazówka

Często przy zastosowaniu orientowalnych nośników narzędzi ma sens zdefiniowanie wszystkich narzędzi dla systemu podstawowego nie poddanego lustrzanemu odbiciu, również tych, które są stosowane tylko przy obróbce w lustrzanym odbiciu. Przy obróbce z osiami poddanymi lustrzanemu odbiciu nośnik narzędzi jest wówczas tak obracany, że rzeczywiste położenie narzędzia jest prawidłowo opisywane. Wszystkie składowe długości narzędzia działają wówczas automatycznie we właściwym kierunku, tak że zbędne staje się sterowanie reakcją poszczególnych składowych przez dane ustawcze zależnie od stanu lustrzanego odbicia poszczególnych osi.

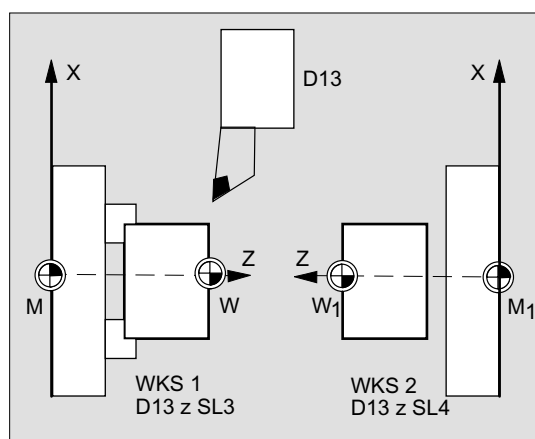
Dalsze możliwości zastosowania

Zastosowanie funkcji orientowanych nośników narzędzi może mieć sens również wtedy, gdy w maszynie nie jest fizycznie przewidziana żadna możliwość obracania narzędzi, ale są one na stałe zainstalowane z różnym zorientowaniem. Zwymiarowania narzędzi można wówczas jednolicie dokonać w zorientowaniu podstawowym, a wymiary mające znaczenie dla obróbki wynikają z obrotu wirtualnego nośnika narzędzi.

9.3.1 Lustrzane odbicie długości narzędzia

Funkcja

Przy pomocy danych ustawczych SD42900 `$$SC_MIRROR_TOOL_LENGTH` i SD42910 `$$SC_MIRROR_TOOL_WEAR` nierównych zeru można poddać lustrzanemu odbiciu komponenty długości narzędzia i komponenty wymiarów bazowych z wartościami zużycia w ich przynależnych osiach.



SD42900 `$$SC_MIRROR_TOOL_LENGTH`

Dana ustawcza **nie** równa zeru:

Lustrzanemu odbiciu podlegają również komponenty długości narzędzia (`$TC_DP3`, `$TC_DP4` i `$TC_DP5`) i komponenty wymiarów bazowych (`$TC_DP21`, `$TC_DP22` i `$TC_DP23`), których przynależne osie są poddane lustrzanemu odbiciu, - przez zmianę znaku liczby.

Wartości zużycia **nie** podlegają równoczesnemu lustrzanemu odbiciu. Gdy mają one również zostać poddane lustrzanemu odbiciu, musi być ustawiona dana ustawcza SD42910 \$SC_MIRROR_TOOL_WEAR.

SD42910 \$SC_MIRROR_TOOL_WEAR

Dana ustawcza **nierówna** zeru:

Wartości zużycia komponentów długości narzędzia, których przynależne osie są poddane lustrzanemu odbiciu, są również poddawane lustrzanemu odbiciu - przez odwrócenie znaku liczby.

9.3.2 Reakcja na znak zużycia

Funkcja

Przy pomocy ustawionych danych ustawczych SD42920 \$SC_WEAR_SIGN_CUTPOS i SD42930 \$SC_WEAR_SIGN nierówne zeru, można poddać odwróceniu ewaluację znaku komponentów zużycia.

SD42920 \$SC_WEAR_SIGN_CUTPOS

Dana ustawcza **nierówna** zeru:

W przypadku narzędzi o mającym znaczeniu położeniu ostrza (narzędzia tokarskie i szlifierskie, typy narzędzi 400) reakcja na znak liczby składowych zużycia w płaszczyźnie obróbki zależy od położenia ostrza. W przypadku typów narzędzi bez mającego znaczenie położenia ostrza ta dana ustawcza nie ma znaczenia

W poniższej tabelicy są przez X zaznaczone wymiary, których znak jest odwracany przez SD42920 (nierówna 0):

Położenie ostrza	Długość 1	Długość 2
1		
2		X
3	X	X
4	X	
5		
6		
7		X
8	X	
9		

Wskazówka

Ewaluacja znaku przez SD42920 i SD42910 jest niezależna od siebie. Gdy np. znak podania wymiaru zostanie zmieniony przez obydwie dane ustawcze, znak wynikowy pozostanie bez zmian.

SD42930 \$SC_WEAR_SIGN

Dana ustawcza **nierówna** zeru:

Znak wszystkich wymiarów zużycia jest odwracany. Działa on zarówno na długość narzędzia jak również na pozostałe wielkości promienia narzędzia, promień zaokrąglenia, itd.

Gdy zostanie wprowadzona dodatnia miara zużycia, narzędzie stanie się przez to "krótsze" i "cieńsze", patrz punkt "Korekcja narzędzia, traktowanie specjalne", początek działania zmienionych danych ustawczych".

9.3.3 Układ współrzędnych aktywnej obróbki (TOWSTD, TOWMCS, TOWWCS, TOWBCS, TOWTCS, TOWKCS)

Funkcja

Zależnie od kinematyki maszyny albo od występowania orientowalnego nośnika narzędzi wartości zużycia zmierzone w jednym z tych układów współrzędnych są przenoszone lub transformowane na odpowiedni układ współrzędnych.

Układy współrzędnych aktywnej obróbki

Z poniższych układów współrzędnych mogą wynikać offsety długości narzędzia, które zużycie jako składową długości narzędzia przez odpowiedni G-Code z grupy 56 wliczają do aktywnego narzędzia.

- Układ współrzędnych maszyny (MKS)
- Bazowy układ współrzędnych (BKS)
- Układ współrzędnych obrabianego przedmiotu (WKS)
- Układ współrzędnych narzędzia (TCS)
- Układ współrzędnych narzędzia transformacji kinematycznej (KCS)

Składnia

TOWSTD
TOWMCS
TOWWCS
TOWBCS
TOWTCS
TOWKCS

Znaczenie

TOWSTD	Wartość ustawienia podstawowego korekcji w długości narzędzia, wartość zużycia
TOWMCS	Korekcje w długości narzędzia w MKS
TOWWCS	Korekcje w długości narzędzia w WKS
TOWBCS	Korekcje w długości narzędzia w BKS
TOWTCS	Korekcje długości narzędzia w punkcie odniesienia nośnika narzędzi (orientowalny nośnik narzędzi)
TOWKCS	Korekcje długości narzędzia głowicy narzędziowej (transformacji kinematycznej)

Dalsze informacje

Cechy odróżniające

W poniższej tabelicy są przedstawione najważniejsze cechy odróżniające:

G-Code	Wartość zużycia	Aktywny orientowany nośnik narzędzi
TOWSTD	Wartość ustawienia podstawowego, długość narzędzia	Wartości zużycia podlegają obrotowi.
TOWMCS	Wartość zużycia w MKS. TOWMCS jest identyczny z TOWSTD, gdy nie jest aktywny orientowalny nośnik narzędzi.	Obraca się tylko wektor wynikowej długości narzędzia bez uwzględnienia zużycia.
TOWWCS	Wartość zużycia jest w układzie współrzędnych obrabianego przedmiotu przeliczana na układ współrzędnych maszyny.	Wektor narzędzia jest obliczany bez uwzględnienia zużycia, jak w przypadku TOWMCS.
TOWBCS	Wartość zużycia jest w bazowym układzie współrzędnych przeliczana na układ współrzędnych maszyny.	Wektor narzędzia jest obliczany bez uwzględnienia zużycia, jak w przypadku TOWMCS.
TOWTCS	Wartość zużycia jest w układzie współrzędnych narzędzia przeliczana na układ współrzędnych maszyny.	Wektor narzędzia jest obliczany bez uwzględnienia zużycia, jak w przypadku TOWMCS.

TOWWCS , TOWBCS, TOWTCS: Wektor zużycia jest dodawany do wektora narzędzia.

Transformacja liniowa

Długość narzędzia jest możliwa tylko wtedy do sensownego zdefiniowania w układzie współrzędnych maszyny, gdy układ ten wynika z bazowego układu współrzędnych przez transformację liniową.

Transformacja nieliniowa

Jeżeli np. przy pomocy TRANSMIT jest aktywna transformacja nieliniowa, wówczas przy podaniu MKS jako żądanego układu współrzędnych jest automatycznie stosowany BKS.

Bez transformacji kinematycznej i bez orientowanego nośnika narzędzi

Jeżeli z powodu transformacji kinematycznej jest aktywny jeszcze jeden orientowany nośnik narzędzi, wówczas oprócz WKS są kombinowane wszystkie dalsze cztery układy współrzędnych. Przez to tylko układ współrzędnych obrabianego przedmiotu różni się od pozostałych. Ponieważ należy poddać ewaluacji wyłącznie długości narzędzi, translacje między układami współrzędnych nie mają żadnego znaczenia.

Literatura:

Dalsze informacje dot. korekcji narzędzi patrz:

Podręcznik działania Funkcje podstawowe; Korekcja narzędzia (W1)

Wliczenie wartości zużycia

Dana ustawcza **SD42935 \$SC_WEAR_TRANSFORM** ustala, która z trzech składowych zużycia:

- Zużycie
- Korekcje sumaryczne dokładne
- Korekcje sumaryczne zgrubne

ma zostać poddana obrotowi przez transformację dostosowawczą albo orientowany nośnik narzędzi, gdy jest aktywny jeden z następujących G-Code:

- **TOWSTD** położenie podstawowe dla korekcji w długości narzędzia
- **TOWMCS** wartości zużycia w układzie współrzędnych maszyny (MKS)
- **TOWWCS** wartości zużycia w układzie współrzędnych obrabianego przedmiotu (WKS)
- **TOWBCS** wartości zużycia (BKS) w bazowym układzie współrzędnych
- **TOWTCS** Wartości zużycia w układzie współrzędnych narzędzia na zamocowaniu uchwytu narzędzia (T odniesienie nośnika narzędzi)
- **TOWKCS** Wartości zużycia w układzie współrzędnych głowicy narzędziowej przy transformacji kinetycznej

Wskazówka

Ewaluacja poszczególnych składowych zużycia (przyporządkowanie do osi geometrycznych, ewaluacja znaku) ma wpływ:

- aktywna płaszczyzna
 - transformacja dostosowawcza
 - następujące dane ustawcze:
 - SD42910 \$SC_MIRROR_TOOL_WEAR
 - SD42920 \$SC_WEAR_SIGN_CUTPOS
 - SD42930 \$SC_WEAR_SIGN
 - SD42940 \$SC_TOOL_LENGTH_CONST
 - SD42950 \$SC_TOOL_LENGTH_TYPE
-

9.3.4 Długość narzędzia i zmiana płaszczyzny

Funkcja

Przy pomocy ustawionych danych ustawczych SD42940 \$SC_TOOL_LENGTH_CONST nierównych zero można dla narzędzi tokarskich i szlifierskich przy zmianie płaszczyzny przyporządkować komponenty długości narzędzia, jak długość, zużycie i wymiar bazowy do osi geometrycznych.

SD42940 \$SC_TOOL_LENGTH_CONST

Dana ustawcza **nierówna** zeru:

Przyporządkowanie składowych długości narzędzia (długość, zużycie i wymiar bazowy) do osi geometrycznych przy zmianie płaszczyzny obróbki (G17 - G18) nie ulega zmianie.

Poniższa tablica pokazuje przyporządkowanie składowych długości narzędzia do osi geometrii dla narzędzi tokarskich i szlifierskich (typ narzędzia 400 do 599):

Treść	Długość 1	Długość 2	Długość 3
17	Y	X	Z
*)	X	Z	Y
19	Z	Y	X
-17	X	Y	Z
-18	Z	X	Y
-19	Y	Z	X

*) Każda wartość nierówna 0, która jest nierówna jednej z sześciu wymienionych wartości, jest traktowana jako wartość 18.

Poniższa tablica pokazuje przyporządkowanie składowych długości narzędzia do osi geometrii dla wszystkich innych narzędzi (typ narzędzia < 400 lub > 599):

Płaszczyzna obróbki	Długość 1	Długość 2	Długość 3
*)	Z	Y	X
18	Y	X	Z
19	X	Z	Y
-17	Z	X	Y
-18	Y	Z	X
-19	X	Y	Z

*) Każda wartość nierówna 0, która jest nierówna jednej z sześciu wymienionych wartości, jest traktowana jako wartość 17.

Wskazówka

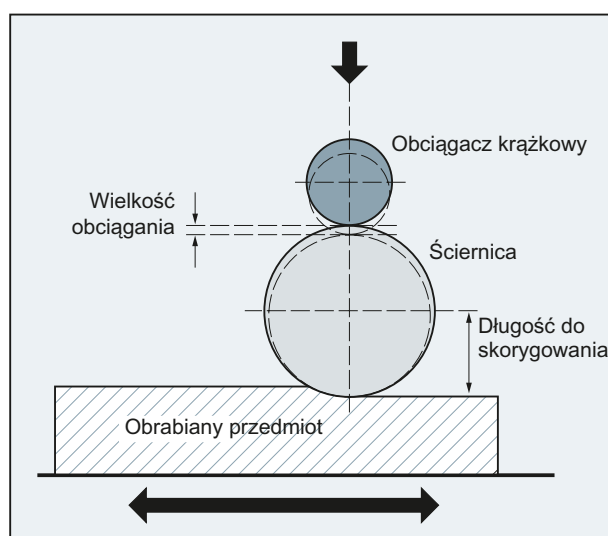
Przy przedstawianiu w tablicach zakładamy, że osie geometryczne do 3 mają identyfikatory X, Y i Z. Dla przyporządkowania korekcji do osi jest miarodajny nie identyfikator osi lecz ich kolejność.

9.4 Korekcja narzędzia online (PUTFTOCF, FCTDEF, PUTFTOC, FTOCON, FTOCOF)

Funkcja

Przy pomocy funkcji "Korekcja narzędzia online" jest w przypadku narzędzi szlifierskich natychmiast wliczana korekcja długości narzędzia, która wynika z obróbki.

Przykładem zastosowania jest obciąganie CD, w przypadku którego ściernica jest obciągana równoległe do obróbki:



Korekcję długości narzędzia można zmienić z kanału obróbki albo kanału równoległego (kanał obciągacza).

Do zapisania korekcji narzędzia online są zależnie od pożądanej chwili procesu obciągania stosowane różne funkcje:

- Zapis ciągły pojedynczymi blokami (PUTFTOCF)

Przy pomocy PUTFTOCF proces obciągania następuje równocześnie z obróbką.

Korekcja narzędzia jest zmieniana w kanale obróbkowym w sposób ciągły według funkcji wielomianowej 1., 2. albo 3. stopnia, która przedtem musi zostać zdefiniowana przy pomocy FCTDEF.

PUTFTOCF działa zawsze pojedynczymi blokami, tzn. w następnym bloku ruchu postępowego.

- Zapis ciągły modalnie: ID=1 DO FTOC (patrz "Akcje synchroniczne (Strona 561)")
- Zapis nieciągły (PUTFTOC)

Przy pomocy PUTFTOC proces obciągania następuje nie w tym samym czasie co obróbka z kanału równoległego. Podana przy pomocy PUTFTOC wartość korekcji natychmiast działa w kanale docelowym.

Wskazówka

Korekcja narzędzia online może być stosowana tylko w przypadku narzędzi szlifierskich.

Składnia

Włączenie/wyłączenie korekcji narzędzia online w kanale docelowym:

```
FTOCON
...
FTOCOF
```

Zapisanie korekcji narzędzia online:

- Ciągłe pojedynczymi blokami:

```
FCTDEF (<funkcja>, <LLimit>, <ULimit>, <a0>, <a1>, <a2>, <a3>)
PUTFTOCF (<funkcja>, <wartość odniesienia>, <parametr narzędzia>, <kanał>, <wrzeciono>)
...
```

- Nieciągła:

```
PUTFTOC (<wartość korekcji>, <parametr narzędzia>, <kanał>, <wrzeciono>)
...
```

Znaczenie

FTOCON:	<p>Włączenie korekcji narzędzia online</p> <p>FTOCON musi być programowane w tym kanale, w którym korekcja narzędzia online ma działać.</p>																
FTOCOF:	<p>Anulowanie korekcji narzędzia online</p> <p>Przy pomocy FTOCOF korekcja nie jest dalej realizowana, w specyficznych dla ostrza danych korekcyjnych jest jednak skorygowana kompletna wartość napisana przy pomocy PUTFTOC/PUTFTOCF.</p> <p>Wskazówka:</p> <p>W celu ostatecznego wyłączenia aktywności korekcji narzędzia online musi po FTOCOF nastąpić jeszcze wybór/cofnięcie wyboru narzędzia (T...).</p>																
FCTDEF:	<p>Przy pomocy FCTDEF jest definiowana funkcja wielomianowa dla PUTFTOCF.</p> <p>Parametr:</p> <table> <tr> <td><funkcja>:</td> <td>Numer funkcji wielomianowej</td> </tr> <tr> <td></td> <td>Typ: INT</td> </tr> <tr> <td><LLimit>:</td> <td>Dolna wartość graniczna</td> </tr> <tr> <td></td> <td>Typ: REAL</td> </tr> <tr> <td><ULimit>:</td> <td>Górna wartość graniczna</td> </tr> <tr> <td></td> <td>Typ: REAL</td> </tr> <tr> <td><a0> ... <a3>:</td> <td>Współczynniki funkcji wielomianowej</td> </tr> <tr> <td></td> <td>Typ: REAL</td> </tr> </table>	<funkcja>:	Numer funkcji wielomianowej		Typ: INT	<LLimit>:	Dolna wartość graniczna		Typ: REAL	<ULimit>:	Górna wartość graniczna		Typ: REAL	<a0> ... <a3>:	Współczynniki funkcji wielomianowej		Typ: REAL
<funkcja>:	Numer funkcji wielomianowej																
	Typ: INT																
<LLimit>:	Dolna wartość graniczna																
	Typ: REAL																
<ULimit>:	Górna wartość graniczna																
	Typ: REAL																
<a0> ... <a3>:	Współczynniki funkcji wielomianowej																
	Typ: REAL																

9.4 Korekcja narzędzia online (PUTFTOCF, FCTDEF, PUTFTOC, FTOCON, FTOCOF)

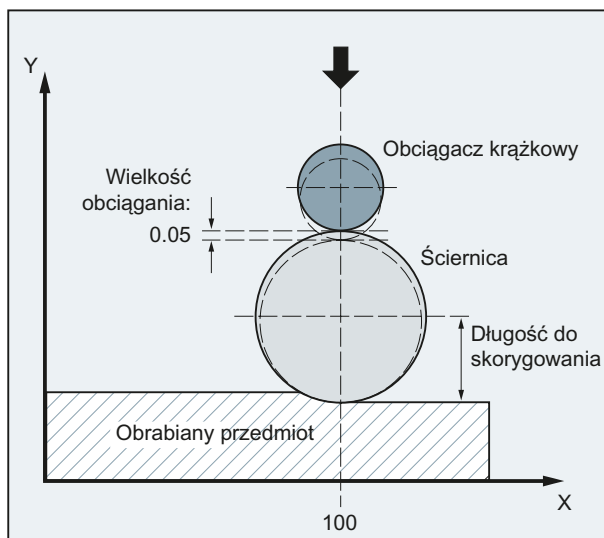
PUTFTOCF:	Wywołanie funkcji "Zapis korekcji narzędzia online ciągly pojedynczymi blokami"
	Parametr:
<funkcja>:	Numer funkcji wielomianowej Typ: INT
	Wskazówka: Musi być zgodny przy podaniu przy FCTDEF.
<wartość odniesienia>:	Zmienna wartość odniesienia, od której ma zostać wyprowadzona korekcja (np. zmieniająca się wartość rzeczywista). Typ: VAR REAL
<parametr narzędzia>:	Numer parametru zużycia (długość 1, 2 lub 3), w którym wartość korekcji ma zostać dodana. Typ: INT
<kanał>:	Numer kanału, w którym korekcja narzędzia online ma działać. Typ: INT
	Wskazówka: Podanie jest wymagane tylko wtedy, gdy korekcja ma nie działać w aktywnym kanale.
<wrzeciono>:	Numer wrzeciona, dla którego korekcja narzędzia online ma działać. Typ: INT
	Wskazówka: Podanie jest wymagane tylko wtedy, gdy zamiast aktywnego, pracującego narzędzia ma zostać skorygowana nie aktywna ściernica.
PUTFTOC:	Wywołanie funkcji "Zapis korekcji narzędzia online nieciągly"
	Parametr:
<wartość korekcji>:	Wartość korekcji, która ma zostać dodana w parametrze zużycia. Typ: REAL
<parametr narzędzia>:	patrz PUTFTOCF
<kanał>:	Numer kanału, w którym korekcja narzędzia online ma działać. Typ: INT
<wrzeciono>:	patrz PUTFTOCF

Przykład

Szlifierka do płaszczyzn z:

- Y: oś dosuwu ściernicy
- V: oś dosuwu obciążacza krążkowego
- Kanał obróbki: kanał 1 z osiami X, Z, Y
- Kanał obciążania: kanał 2 z osią V

Po rozpoczęciu ruchu szlifowania ściernica powinna przy X100 zostać obciążona o wielkość 0,05. Wielkość obciążenia powinna przy pomocy "zapis korekcji narzędzia online ciągle" działać w przypadku narzędzia szlifierskiego.



Program obróbki w kanale 1:

Kod programu	Komentarz
...	
N110 G1 G18 F10 G90	; Położenie podstawowe
N120 T1 D1	; Wybór aktualnego narzędzia.
N130 S100 M3 X100	; Wrzeczono wł., ruch do pozycji wyjściowej.
N140 INIT(2, "ABRICHT", "S")	; Wybór programu obciążania w kanale 2.
N150 START(2)	; Uruchomienie programu obciążania w kanale 2.
N160 X200	; Ruch do pozycji docelowej.
N170 FTOCON	; Włączenie korekcji online.
N... G1 X100	; Dalsza obróbka.
N... M30	

Program obciążania w kanale 2:

Kod programu	Komentarz
...	
N40 FCTDEF(1,-1000,1000,-\$AA_IW[V],1)	; Zdefiniowanie funkcji: prosta o nachyleniu=1.
N50 PUTFTOCF(1,\$AA_IW[V],3,1)	; Zapis ciągły korekcji narzędzia online: wychodząc od ruchu osi V jest korygowana długość 3 aktualnej ściernicy w kanale 1.
N60 V-0.05 G1 F0.01 G91	; Ruch dosuwu do obciążania, tylko w tym bloku działa PUTFTOCF.
...	
N... M30	

Dalsze informacje

Ogólnie na temat korekcji narzędzia online

Przy zapisie ciągłym (na takt interpolatora) po włączeniu funkcji ewaluacji każda zmiana jest addytywnie przeliczana w pamięci zużycia (aby uniknąć skoków wartości zadanej).

W każdym przypadku obowiązuje: Korekcja narzędzia online może działać w każdym kanale dla każdego wrzeciona i długości 1, 2 **albo** 3 parametrów zużycia.

Przyporządkowanie długości do osi geometrycznych następuje na podstawie aktualnej płaszczyzny roboczej.

Przyporządkowanie wrzeciona do narzędzia następuje przez dane narzędzia w przypadku GWPSON wzgl. TMON, o ile nie chodzi o aktywną ściernicę.

Korygowany jest zawsze parametr zużycia dla aktualnej strony ściernicy wzgl. lewej strony ściernicy w przypadku narzędzi nie aktywnych.

Wskazówka

Przy identycznej korekcji dla wielu stron ściernicy należy poprzez instrukcję powiązania zadbać o to, by wartości zostały automatycznie przejęte dla drugiej strony ściernicy.

Jeżeli dla kanału obróbki zostaną zadane korekcje online, wówczas wartości zużycia dla aktualnego narzędzia w tym kanale nie wolno zmienić z programu obróbki, ani w drodze czynności obsługowej.

Korekcja narzędzia online jest również uwzględniana dla stałej prędkości obwodowej ściernicy (SUG), jak też nadzoru narzędzia TMON.

9.5 Uaktywnienie korekcji narzędzia 3D (CUT3DC..., CUT3DF...)

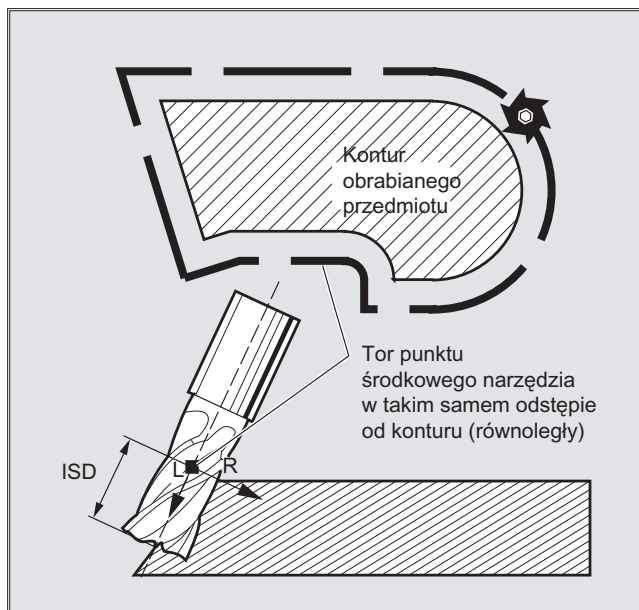
9.5.1 Uaktywnienie korekcji narzędzia 3D (CUT3DC, CUT3DF, CUT3DFS, CUT3DFF, ISD)

Funkcja

W przypadku korekcji promienia narzędzia dla narzędzi cylindrycznych jest uwzględniana zmieniona orientacja narzędzia.

Dla wyboru korekcji promienia narzędzia 3D obowiązują takie same polecenia programowe, jak w przypadku korekcji promienia narzędzia 2D. Przez $G41/G42$ jest podawana korekcja na lewo/prawo w kierunku ruchu. Zachowanie się przy dosunięciu jest zawsze $NORM$. Korekcja promienia narzędzia 3D działa tylko przy wybranej transformacji 5-osiowej.

Korekcja promienia narzędzia 3D jest określana również jako korekcja 5D, ponieważ w tym przypadku jest do dyspozycji 5 stopni swobody dla położenia narzędzia w przestrzeni.



Różnica między korekcją promienia narzędzia 2 1/2D i 3D

W przypadku korekcji promienia narzędzia 3D orientacja narzędzia jest zmienna. W przypadku korekcji promienia narzędzia 2 1/2D obliczenie jest wykonywane tylko z jednym narzędziem o stałej orientacji.

Składnia

CUT3DC
CUT3DFS
CUT3DFF
CUT3DF
ISD=<wartość>

Znaczenie

CUT3DC	Uaktywnienie korekcji narzędzia 3D dla frezowania obwodowego
CUT3DFS	Korekcja narzędzia D dla frezowania czołowego ze stałą orientacją. Orientacja narzędzia jest ustalona przez G17 - G19 i frame nie mają na nią wpływu.
CUT3DFE	Korekcja narzędzia D dla frezowania czołowego ze stałą orientacją. Orientacją narzędzia jest kierunek ustalony przez G17 - G19 i ew. obrócony przez frame.
CUT3DF	Korekcja narzędzia D dla frezowania czołowego ze zmianą orientacji (tylko przy aktywnej transformacji 5-osiowej).
G40 X... Y... Z...	Do wyłączenia: blok liniowy G0/G1 z osiami geometrycznymi
ISD	Głębokość wgłębienia

Wskazówka

Polecenia działają modalnie i są w tej samej grupie co CUT2D i CUT2DF. Cofnięcie wyboru następuje dopiero z następnym ruchem w aktualnej płaszczyźnie. Obowiązuje to zawsze dla G40 i jest niezależne od polecenia CUT.

Bloki pośrednie przy aktywnej korekcji promienia narzędzia 3D są dozwolone. Obowiązują ustalenia dla korekcji promienia narzędzia 2 1/2D.

Warunki brzegowe

- **G450/G451 i DISC**

Na narożach zewnętrznych jest zawsze wstawiany blok ruchu po okręgu. G450/G451 nie mają znaczenia.

Na polecenie DISC nie ma reakcji.

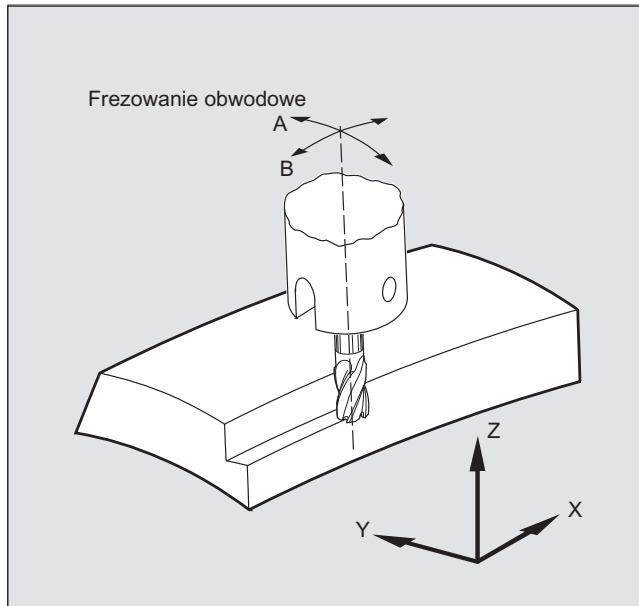
Przykład

Kod programu	Komentarz
N10 A0 B0 X0 Y0 Z0 F5000	
N20 T1 D1	; Wywołanie narzędzia, wywołanie wartości korekcyjnych narzędzia.
N30 TRAORI(1)	; Wybór transformacji
N40 CUT3DC	; Wybór korekcji promienia narzędzia 3D
N50 G42 X10 Y10	; Wybór korekcji promienia narzędzia
N60 X60	
N70 ...	

9.5.2 Korekcja narzędzia 3D: frezowanie obwodowe, frezowanie czołowe

Frezowanie obwodowe

Używany tutaj wariant frezowania obwodowego jest realizowany przez zadanie toru (linii prowadzącej) i przynależnej orientacji. Przy tym rodzaju obróbki kształt narzędzia na torze jest bez znaczenia. Decyduje tylko promień na punkcie przyporu narzędzia.

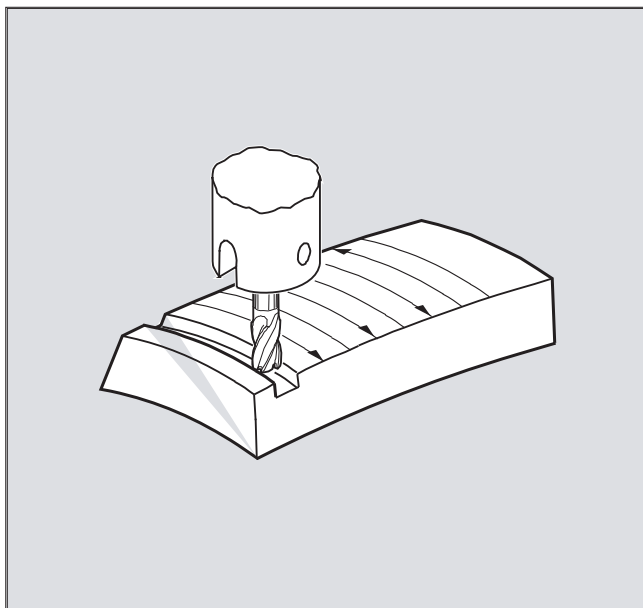


Wskazówka

Funkcja korekcji promienia narzędzia 3D ogranicza się do narzędzi cylindrycznych.

Frezowanie czołowe

Dla tego rodzaju frezowania trójwymiarowego potrzeba wierszowego opisu torów 3D na powierzchni obrabianego przedmiotu. Obliczenia są przeprowadzane przy uwzględnieniu kształtu i wymiarów narzędzia - zazwyczaj w CAM. Postprocesor zapisuje do programu obróbki – oprócz bloków NC– orientację narzędzi (przy aktywnej transformacji 5-osiowej) i G-Code dla pożądanej korekcji narzędzia 3D. Przez to operator maszyny ma możliwość - odmiennie od narzędzia użytego do obliczenia torów NC - użycia nieznacznie mniejszych narzędzi.



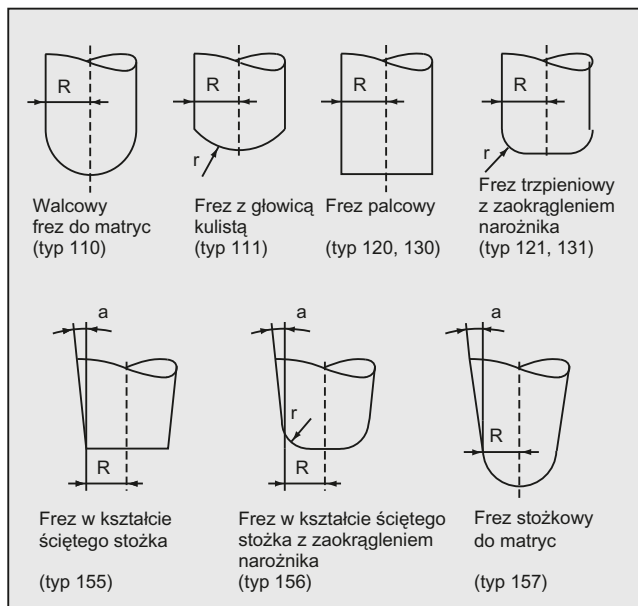
Przykład:

Bloki NC były obliczane z użyciem frezu 10 mm. Tutaj można by również pracować ze średnicą frezu 9,9 mm, przy czym należy się liczyć ze zmienionym profilem chropowatości.

9.5.3 Korekcja narzędzia 3D: kształty narzędzi i dane narzędzi dla frezowania czołowego

Kształty frezu, dane narzędzia

Poniżej zestawiono kształty narzędzi, możliwe dla frezowania czołowego, i wartości graniczne danych narzędzi. Kształt części chwytowej narzędzia nie jest uwzględniany. Typy narzędzi 120 i 156 są identyczne pod względem działania.



Jeżeli w programie NC zostanie podany numer typu inny, niż pokazany na rysunku, system automatycznie stosuje typ narzędzia 110 (frez walcowy do matryc). Przy przekroczeniu wartości granicznych danych narzędzia jest wyprawdzany alarm.

Typ frezu	Nr typu	R	r	a
Frez walcowy do matryc	110	> 0	-	-
Frez z głowicą kulistą	111	> 0	> R	-
Frez trzpieniowy, frez z głowicą kątową	120, 130	> 0	-	-
Frez trzpieniowy, frez z głowicą kątową z zaokrągleniem naroża	121, 131	> r	> 0	-
Frez w kształcie ściętego stożka	155	> 0	-	> 0
Frez w kształcie ściętego stożka z zaokrągleniem naroża	156	> 0	> 0	> 0
Frez stożkowy do matryc	157	> 0	-	> 0

- R = promień części trzpieniowej (promień narzędzia)
 r = promień naroża
 a = kąt między osią wzdłużną narzędzia i górnym końcem powierzchni torusowej
 - = parametr nie jest brany do obliczeń

Dane narzędzi	Parametry narzędzia	
Wymiary narzędzia	Geometria	Zużycie
R	\$TC_DP6	\$TC_DP15
r	\$TC_DP7	\$TC_DP16
a	\$TC_DP11	\$TC_DP20

Korekcja długości narzędzia

Jako punkt odniesienia dla korekcji długości służy wierzchołek narzędzia (punkt przecięcia oś wzdłużna/powierzchnia).

Korekcja narzędzia 3D, zmiana narzędzia

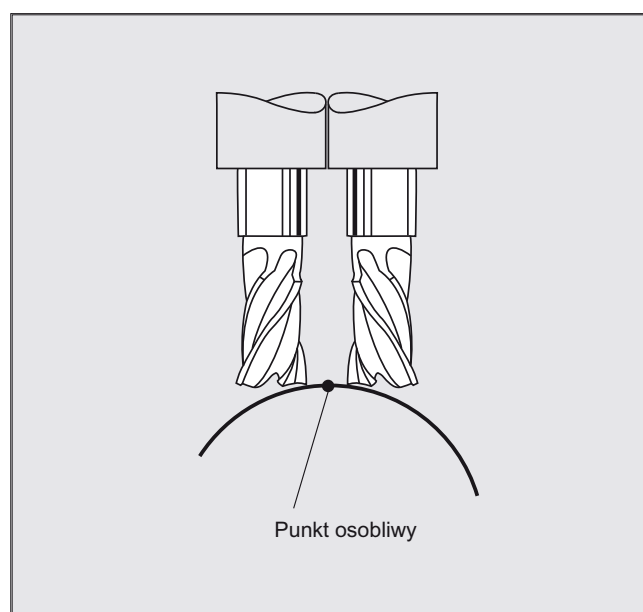
Nowe narzędzie ze zmienionymi wymiarami (R, r, a) lub innym kształtem wolno podać tylko z zaprogramowaniem G41 lub G42 (przejście G40 na G41 wzgl. G42, ponowne zaprogramowanie G41 wzgl. G42). Wszystkie inne dane narzędzia, np. długości narzędzi, pozostają nie uwzględnione przez tę regułę, tak że takie narzędzia można wprowadzać do pozycji roboczej również bez ponownego G41 wzgl. G42.

9.5.4 Korekcja narzędzia 3D: Korekcja po torze, zakrzywienie toru, głębokość wgłębienia (CUT3DC, ISD)

Funkcja

Korekcja na torze

Przy frezowaniu czołowym musi zostać rozważony przypadek, gdy punkt styku na powierzchni narzędzia przeskakuje. Jak w tym przykładzie przy obróbce powierzchni wypukłej narzędziem ustawionym pionowo. Zastosowanie pokazane na rysunku może być traktowane jako przypadek graniczny.



Ten przypadek graniczny jest nadzorowany przez sterowanie w ten sposób, że na bazie przyłożeń kątowych między narzędziem i wektorami normalnej powierzchni są rozpoznawane skokowe zmiany punktu obróbki. W tych miejscach sterowanie wstawia bloki liniowe, tak że ruch może zostać wykonany.

W celu obliczenia bloków liniowych są w danych maszynowych dla kąta w kierunku bocznym zapisane dopuszczalne zakresy kąta. W przypadku gdy ustalone w danych maszynowych wartości graniczne dopuszczalnych zakresów kąta zostaną przekroczone, system generuje alarm.

Zakrzywienie toru

Zakrzywienie toru nie jest nadzorowane. Również tutaj zaleca się stosowanie tylko takich narzędzi, przy pomocy których można pracować bez naruszenia konturu.

Głębokość wgłębienia (ISD)

Głębokość wgłębienia ISD podlega ewaluacji tylko przy aktywnej korekcji promienia narzędzia 3D.

Przy pomocy polecenia programowego `ISD` (Insertion Depth) jest programowana głębokość wgłębienia narzędzia przy frezowaniu obwodowym. Przez to jest możliwa zmiana położenia punktu obróbki na powierzchni pobocznicowej narzędzia.

Składnia

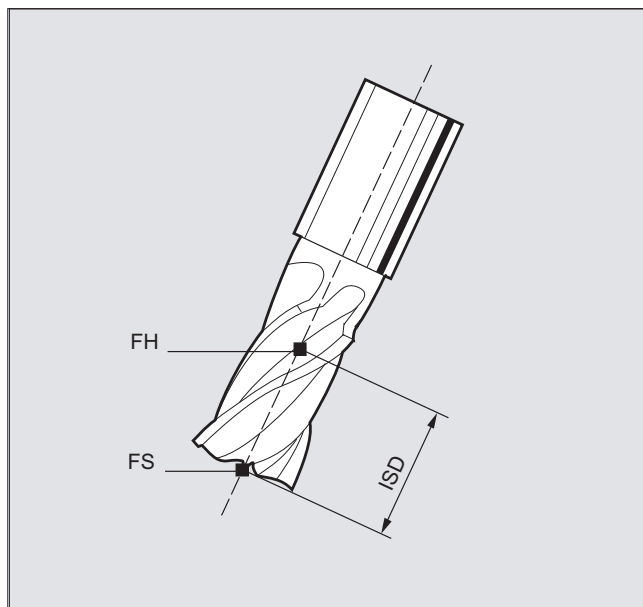
Korekcja narzędzia 3D, frezowanie obwodowe
CUT3DC
ISD=<wartość>

Znaczenie

CUT3DC	Uaktywnienie korekcji narzędzia 3D dla frezowania obwodowego, np. do frezowania wnęki o skośnych ściankach bocznych.
ISD	Przy pomocy polecenia <code>ISD</code> jest podawany odstęp (<wartość>) między wierzchołkiem frezu (FS) i punktem pomocniczym frezu (FH).

Punkt pomocniczy frezu

Punkt pomocniczy frezu (FH) powstaje przez rzut zaprogramowanego punktu obróbki na oś narzędzia.



Dalsze informacje

Frezowanie wnęki o skośnych ściankach dla frezowania obwodowego przy pomocy CUT3DC

Przy tej korekcji promienia narzędzia 3D jest kompensowane odchylenie promienia frezu przez dokonanie dosunięcia w kierunku normalnej powierzchni obrabianej. Niezmieniona pozostaje przy tym płaszczyzna, w której leży powierzchnia czołowa frezu, gdy głębokość wgłębienia I_{SD} pozostaje taka sama. Frez o np. mniejszym promieniu w stosunku do narzędzia normalnego nie osiągnąłby wówczas dna wnęki, które również stanowi płaszczyznę ograniczającą. Dla automatycznego dosuwu narzędzia w sterowaniu musi być znana ta płaszczyzna ograniczająca, patrz punkt "Frezowanie obwodowe 3D z powierzchniami ograniczającymi".

Dalsze informacje dot. nadzoru na kolizję patrz:

Literatura:

Podręcznik programowania Podstawy; punkt "Korekcje narzędzia".

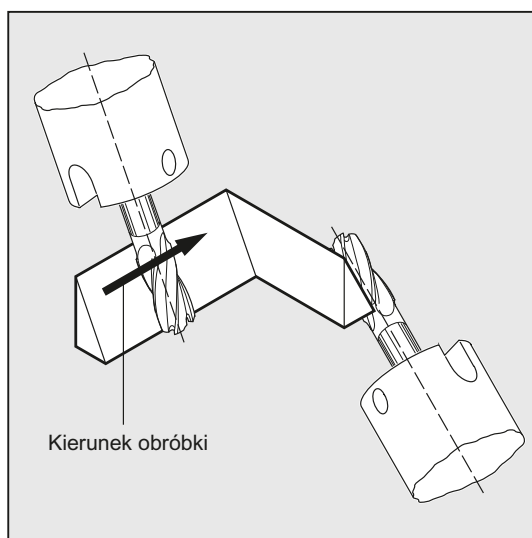
9.5.5 Korekcja narzędzia 3D: naroża wewnętrzne/zewnętrzne i metoda punktu przecięcia (G450/G451)

Funkcja

Naroża wewnętrzne/zewnętrzne

Naroża zewnętrzne i wewnętrzne są traktowane oddzielnie. Określenie naroże wewnętrzne lub zewnętrzne jest zależne od orientacji narzędzia.

Przy zmianie orientacji na narożu może wystąpić przypadek, że typ naroża zmieni się podczas obróbki. Jeżeli ten przypadek wystąpi, obróbka zostanie przerwana z komunikatem błędu.



Składnia

G450
G451

Znaczenie

G450 Okrąg przejściowy (narzędzie obchodzi naroża obrabianego przedmiotu po torze kołowym).
G451 Punkt przecięcia równoległych (narzędzie wychodzi z materiału w narożu)

Dalsze informacje

Metoda punktu przecięcia dla korekcji 3D

Przy frezowaniu obwodowym 3D następuje teraz na narożach zewnętrznych reakcja na G-Code G450/G451, tzn. może nastąpić dosunięcie do punktu przecięcia krzywych offsetu. Do wersji opr. 4 na narożach zewnętrznych zawsze był wstawiany okrąg. Dostępna metoda punktu przecięcia jest w przypadku typowych programów 3D wytworzonych przez CAD szczególnie korzystna. Składają się one często z krótkich bloków definiujących proste (dla aproksymacji gładkich krzywych), przy których przejścia między sąsiadującymi blokami są prawie styczne.

W przypadku korekcji promienia narzędzia na stronie zewnętrznej konturu wstawiano dotychczas z zasady okręgi w celu obchodzenia naroży zewnętrznych. Ponieważ te bloki przy prawie stycznych przejściach stają się bardzo krótkie, wynikają niepożądane załamania prędkości.

W tych przypadkach są analogicznie do korekcji promienia $2 \frac{1}{2} D$ przedłużane obydwie uczestniczące krzywe, następuje ruch do punktu przecięcia obydwu krzywych.

Punkt przecięcia jest określany przez przedłużenie krzywych offsetu obydwu bloków i określenie punktu ich przecięcia na płaszczyźnie prostopadłej do orientacji narzędzia na narożu. Jeżeli taki punkt przecięcia nie istnieje, naroże jest traktowane jak dotychczas, tzn. jest wstawiany okrąg.

Dalsze informacje dot. metody punktu przecięcia patrz:

Literatura:

Podręcznik działania Funkcje specjalne; Korekcja promienia narzędzia 3D (W5)

9.5.6 Korekcja narzędzia 3D: frezowanie obwodowe 3D z płaszczyznami ograniczającymi

Dopasowania frezowania obwodowego 3D do warunków programów CAM

Programy NC generowane przez systemy CAM z reguły aproksymują tor punktu środkowego znormalizowanego narzędzia przy pomocy dużej liczby krótkich bloków liniowych. Aby te tak wytworzone bloki wielu konturów częściowych możliwie dokładnie odwzorowywały kontur oryginalny, jest konieczne dokonanie pewnych dopasowań w programie obróbki.

Ważne informacje, które byłyby konieczne dla optymalnej korekcji, ale w programie obróbki nie są już dostępne, muszą zostać zastąpione przez odpowiednie środki. Poniżej przedstawiono typowe metody wyrównania krytycznych przejść albo bezpośrednio w programie obróbki albo przy określaniu realnego konturu (np. przez dosunięcie narzędzia).

Zastosowania

Dodatkowo oprócz typowych przypadków zastosowania, gdzie w miejsce narzędzia normalnego realne narzędzie przechodzi po torze punktu środkowego, zostaną również omówione narzędzia cylindryczne z korekcją 3D. Przy tym programowany tor odnosi się do konturu na powierzchni obróbki. Obowiązująca tutaj powierzchnia ograniczająca jest niezależna od narzędzia. Jak przy zwykłej korekcji promienia narzędzia do obliczenia offsetu prostopadłego w stosunku do płaszczyzny ograniczającej jest brany promień całkowity.

9.5.7 Korekcja narzędzia 3D: Uwzględnienie powierzchni ograniczającej (CUT3DCC, CUT3DCCD)

Funkcja

Frezowanie obwodowe 3D przy użyciu narzędzi realnych

Przy frezowaniu obwodowym 3D z ciągłą albo stałą zmianą orientacji narzędzia jest często programowany tor punktu środkowego narzędzia dla zdefiniowanego narzędzia znormalizowanego. Ponieważ w praktyce często nie ma do dyspozycji pasujących narzędzi znormalizowanych, można użyć narzędzi nie odbiegającego zanadto od narzędzia znormalizowanego.

Przy pomocy `CUT3DCCD` jest dla realnego narzędzia różniące się uwzględniana powierzchnia ograniczająca, którą opisałoby zaprogramowane narzędzie znormalizowane. Program NC opisuje tor punktu środkowego narzędzia znormalizowanego.

Przy pomocy `CUT3DCC` jest przy zastosowaniu narzędzi cylindrycznych uwzględniana powierzchnia ograniczająca, którą osiągnęło by zaprogramowane narzędzie znormalizowane. Program NC opisuje kontur na powierzchni obróbki.

Składnia

`CUT3DCCD`
`CUT3DCC`

Znaczenie

<code>CUT3DCCD</code>	Uaktywnienie korekcji narzędzia 3D dla frezowania obwodowego z płaszczyznami ograniczającymi z narzędziem różnicowym na torze punktu środkowego narzędzia: Dosunięcie do powierzchni ograniczającej.
<code>CUT3DCC</code>	Uaktywnienie korekcji narzędzia 3D dla frezowania obwodowego z płaszczyznami ograniczającymi z korekcją promienia 3D: Kontur na powierzchni obróbki

Wskazówka

Korekcja promienia narzędzia z G41, G42

Dla korekcji promienia narzędzia z `G41`, `G42` przy aktywnym `CUT3DCCD` albo `CUT3DCC` musi być opcja "transformacja orientacji".

Narzędzia znormalizowane z zaokrągleniem naroża

Zaokrąglenie naroża narzędzia znormalizowanego jest opisywane przez parametr narzędzia $\$TC_DP7$. Z parametru narzędzia $\$TC_DP16$ wynika odchylenie zaokrąglenia naroża realnego narzędzia w stosunku do narzędzia znormalizowanego.

Przykład

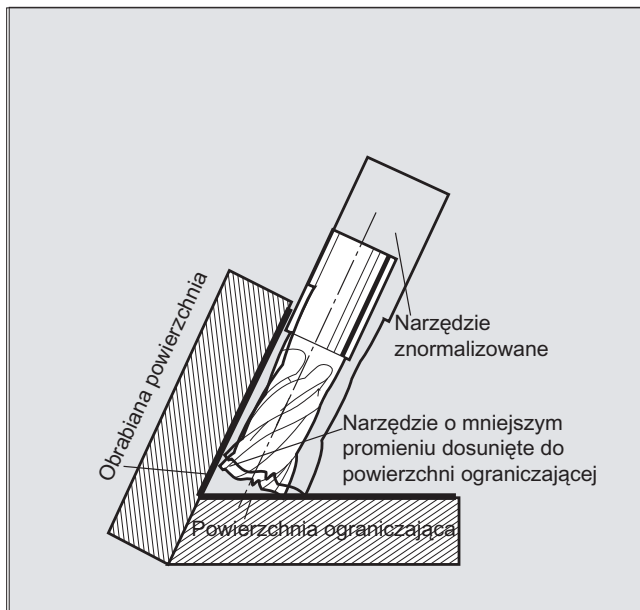
Wymiary narzędzia dla frezu torusowego o promieniu zmniejszonym w stosunku do promienia normalnego.

Typ narzędzia	R = promień części trzpieniowej	r = promień naroża
Narzędzie znormalizowane z zaokrągleniem naroża	$R = \$TC_DP6$	$r = \$TC_DP7$
Narzędzie realne z zaokrągleniem naroża: Typy narzędzi 121 i 131 frez torusowy (frez trzpieniowy)	$R' = \$TC_DP6 + \$TC_DP15 + OFFN$	$r' = \$TC_DP7 + \TC_DP16
W tym przykładzie zarówno $\$TC_DP15 + OFFN$, jak też $\$TC_DP16$ są ujemne. Ewaluacji jest poddawany typ narzędzia ($\$TC_DP1$).		
Dopuszczone są tylko typy frezów o cylindrycznej części trzpieniowej (frez walcowy albo palcowy) jak też frezy torusowe (typ 121 i 131), a w przypadku granicznym cylindryczny frez do matryc (typ 110).	W przypadku dopuszczonych typów frezów promień naroża r jest równy promieniowi części trzpieniowej R. Wszystkie inne dopuszczalne typy narzędzi są interpretowane jako frez walcowy, a ewentualnie podany wymiar zaokrąglenia naroża nie jest uwzględniany.	
Dopuszczalne są wszystkie typy narzędzi o numerach 1 - 399 za wyjątkiem numerów 111 i 155 do 157.		

Dalsze informacje

Tor punktu środkowego narzędzia z dosuwem do płaszczyzny ograniczającej CUT3DCCD

Gdy jest stosowane narzędzie, które w porównaniu z pasującym narzędziem znormalizowanym ma mniejszy promień, wówczas frez dosuwany w kierunku wzdłużnym jest prowadzony na tyle dalej, aż ponownie dotknie on dna wnęki. Przez to z naroża, które jest tworzone przez płaszczyznę obróbki i płaszczyznę ograniczającą, materiał jest wybierany na tyle, na ile pozwala na to narzędzie. Chodzi przy tym o mieszany sposób obróbki, frezowania obwodowego i czołowego. Analogicznie do narzędzia o zmniejszonym promieniu, w przypadku narzędzia o powiększonym promieniu następuje odpowiednie dosunięcie w kierunku przeciwnym.



W stosunku do wszystkich innych korekcji narzędzi grupy G-Code 22 podany dla CUT3DCCD parametr narzędzia $\$TC_DP6$ nie ma żadnego znaczenia dla promienia narzędzia i nie wpływa na uzyskiwaną korekcję.

Offset korekcji wynika z sumy:

- wartości zużycia promienia narzędzia (parametr narzędzia $\$TC_DP15$)
- i offsetu narzędzia $OFFN$ zaprogramowanego w celu obliczenia offsetu prostopadłego do powierzchni ograniczającej.

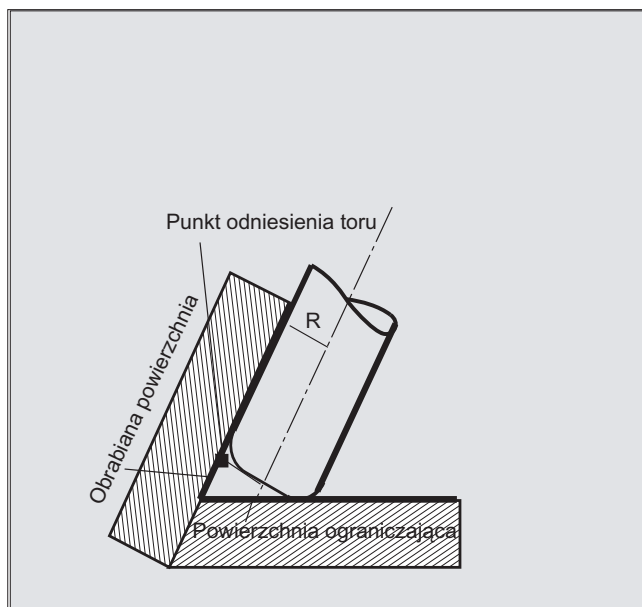
To, czy obrabiana powierzchnia leży na lewo czy na prawo od toru, nie można odczytać ze sporządzonego programu obróbki. Dlatego zakłada się dodatni promień i ujemną wartość zużycia oryginalnego narzędzia. Ujemna wartość zużycia opisuje zawsze narzędzie o zmniejszonej średnicy.

Zastosowanie narzędzi walcowych

Przy zastosowaniu narzędzi walcowych dosuw jest wymagany tylko wtedy, gdy powierzchnia obróbki i płaszczyzna ograniczająca tworzą kąt ostry (mniejszy, niż 90 stopni). Gdy są stosowane frezy torusowe (walec z zaokrąglonym narożem), wówczas zarówno w przypadku kątów ostrych jak i rozwartych wymaga to dosuwu w kierunku wzdłużnym narzędzia.

Korekcja promienia 3D z CUT3DCC, kontur na obrabianej powierzchni

Jeżeli jest aktywne `CUT3DCC` z frezem torusowym, wówczas zaprogramowany tor odnosi się do fikcyjnego frezu palcowego o takiej samej średnicy. Wynikający stąd punkt odniesienia toru jest przy zastosowaniu frezu torusowego przedstawiony na poniższym rysunku.



Jest dopuszczalne, by kąt między powierzchnią obrabianą i powierzchnią ograniczającą również w ramach bloku przechodził z kąta ostrego na rozwarty albo na odwrót.

W stosunku do narzędzia znormalizowanego stosowane realne narzędzie może być albo większe albo mniejsze. Przy tym wynikający promień naroża nie może stać się ujemny, a znak wynikającego promienia narzędzia musi pozostać zachowany.

W przypadku `CUT3DCC` program obróbki NC odnosi się do konturu na obrabianej powierzchni. Jest przy tym, jak przy zwykłej korekcji promienia narzędzia brany cały promień, który składa się z sumy:

- promienia narzędzia (parametr narzędzia `$TC_DP6`)
- wartości zużycia (parametr narzędzia `$TC_DP15`)
- i offsetu narzędzia `OFFN` zaprogramowanego w celu obliczenia offsetu prostopadłego do powierzchni ograniczającej.

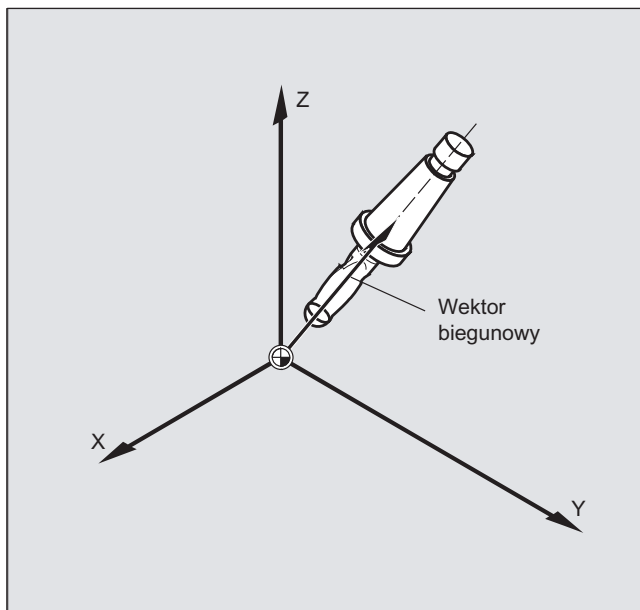
Położenie płaszczyzny ograniczającej jest określone z różnicy obydwu wartości:

- Wymiary narzędzia znormalizowanego
- Promień narzędzia (parametr narzędzia `$TC_DP6`)

9.6 Orientacja narzędzia (ORIC, ORID, OSOF, OSC, OSS, OSSE, ORIS, OSD, OST)

Funkcja

Pod orientacją narzędzia rozumie się geometryczne ustawienie narzędzia w przestrzeni. W przypadku 5-osiowej maszyny obróbkowej orientacja osi jest ustawialna przez polecenia programowe.



Ruchy ścinania naroży orientacji uaktywnione przy pomocy `OSD` i `OST` są w zależności od rodzaju interpolacji dla orientacji narzędzia tworzone różnie:

Przy aktywnej interpolacji wektorowej wygładzony przebieg orientacji jest interpolowany również przy pomocy interpolacji wektorowej. Natomiast przy aktywnej interpolacji osi obrotowej orientacja jest wygładzana bezpośrednio przy pomocy ruchów osi obrotowej.

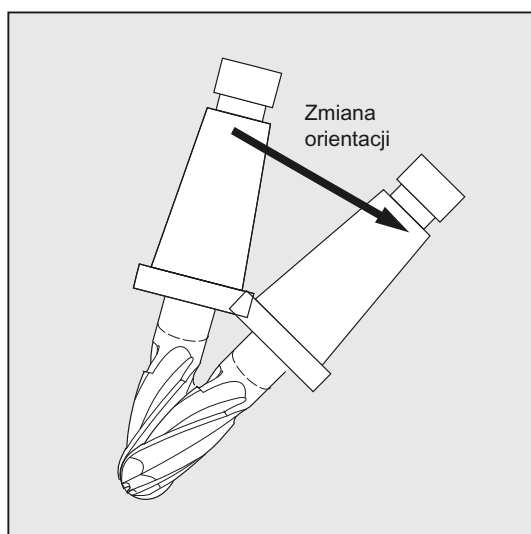
Programowanie

Programowanie zmiany orientacji:

Zmiana orientacji narzędzia może być programowana przez:

- bezpośrednio programowanie osi obrotowych A, B, C (interpolacja osi obrotowej)
- kąt Eulera lub kąt RPY
- wektor biegunowy (interpolacja wektorowa przez podanie `A3` lub `B3` lub `C3`)
- `LEAD/TILT` (frezowanie czołowe)

Układem współrzędnych odniesienia jest albo układ współrzędnych maszyny (`ORIMKS`) albo aktualny układ współrzędnych obrabianego przedmiotu (`ORIWKS`).



Programowanie orientacji narzędzia:

Polecenie	Znaczenie
ORIC:	Orientacja i ruch po torze równoległe
ORID:	Orientacja i ruch po torze kolejno
OSOF:	Brak wygładzania orientacji
OSC:	Orientacja stała
OSS:	Wygładzanie orientacji tylko na początku bloku
OSSE:	Wygładzanie orientacji na początku i końcu bloku
ORIS:	Prędkość zmiany orientacji przy włączonym wygładzaniu orientacji w stopniach na mm (obowiązuje dla OSS i OSSE)
OSD:	Wygładzanie orientacji przez zadanie drogi wygładzania przy pomocy danej ustawczej: SD42674 \$SC_ORI_SMOOTH_DIST
OST:	Wygładzanie orientacji przez zadanie tolerancji kątowej w stopniach przy interpolacji wektorowej przy pomocy danej ustawczej SD42676 \$SC_ORI_SMOOTH_TOL Przy interpolacji osi obrotowej zadana tolerancja jest przyjmowana, jako maksymalne odchylenie osi orientacji.

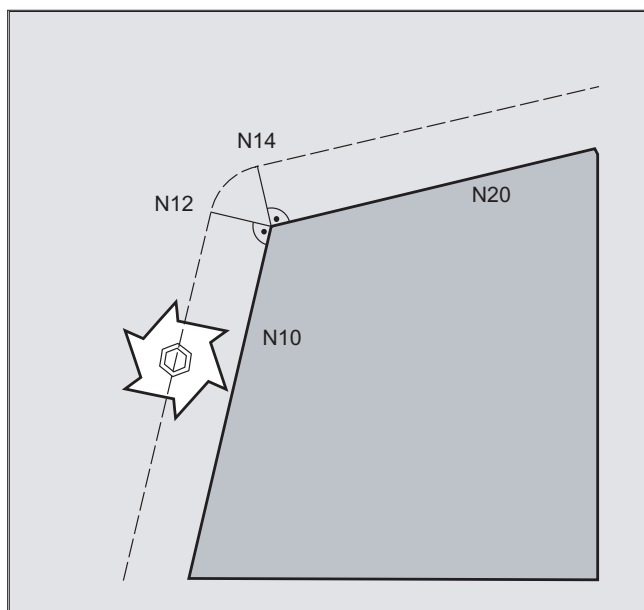
Wskazówka

Wszystkie polecenia do wygładzania orientacji narzędzia (OSOF, OSC, OSS, OSSE, OSD i OST) są ujęte w grupie funkcji G 34. Działają one modalnie, tzn. zawsze może działać tylko jedno z tych poleceń.

Przykłady

Przykład 1: ORIC

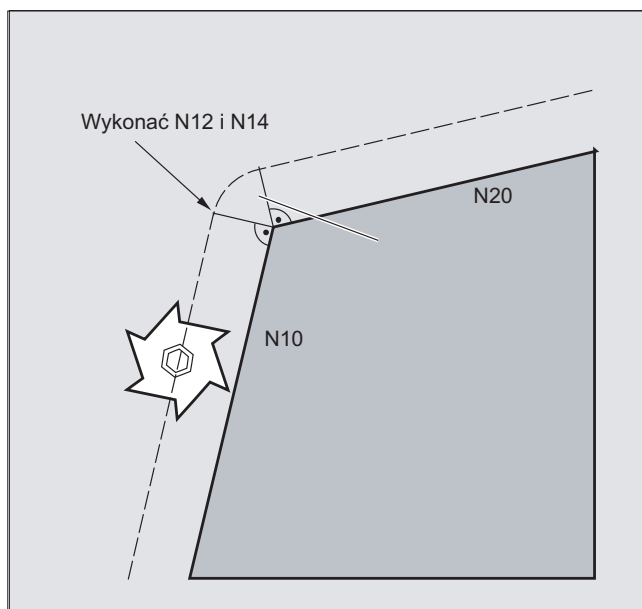
Jeżeli między blokami ruchu postępowego N10 i N20 zaprogramowano dwa albo więcej bloków ze zmianami orientacji (np. A2=... B2=... C2=...) i jest aktywne ORIC, wówczas wstawiony blok okręgu jest dzielony na te bloki pośrednie odpowiednio do wartości bezwzględnej zmian kąta.



Kod programu	Komentarz
ORIC	
N8 A2=... B2=... C2=...	
N10 X... Y... Z...	
N12 C2=... B2=...	; Blok okręgu, który jest wstawiany na narożu zewnętrznym, dzieli się na N12 i N14 odpowiednio do zmiany orientacji. Ruch kołowy i zmiana orientacji są przy tym wykonywane równolegle.
N14 C2=... B2=...	
N20 X =...Y=... Z=... G1 F200	

Przykład 2: ORID

Gdy jest aktywne `ORID`, wówczas wszystkie bloki są wykonywane między obydwooma blokami ruchu postępowego na końcu pierwszego bloku ruchu postępowego. Blok okręgu ze stałą orientacją jest wykonywany bezpośrednio przed drugim blokiem ruchu postępowego.



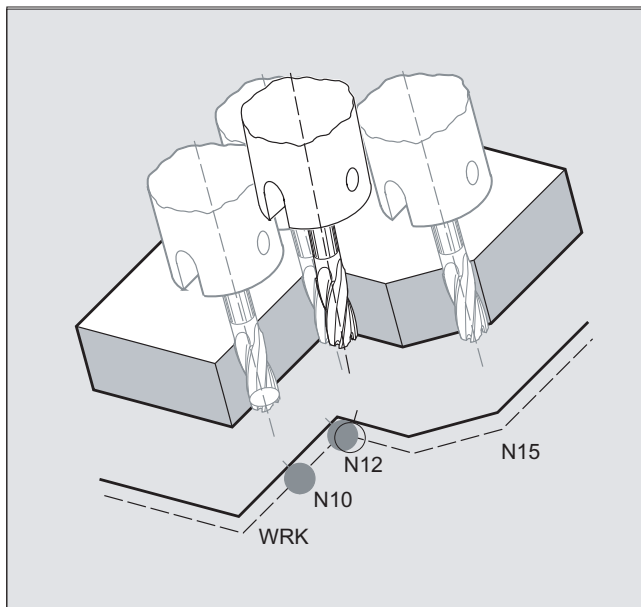
Kod programu	Komentarz
ORID	
N8 A2=... B2=... C2=...	
N10 X... Y... Z...	
N12 A2=... B2=... C2=...	; Bloki N12 i N14 są wykonywana na końcu N10. Następnie jest realizowany blok okręgu z aktualną orientacją.
N14 M20	; Funkcje pomocnicze itd.
N20 X... Y... Z...	

Wskazówka

Dla rodzaju zmiany orientacji na narożu zewnętrznym jest miarodajne polecenie programowe, które jest aktywne w pierwszym bloku ruchu postępowego naroża zewnętrznego.

Bez zmiany orientacji: Jeżeli orientacja nie jest na końcu bloku zmieniana, wówczas przekrój narzędzia jest okręgiem, który dotyka obydwu konturów.

Przykład 3: Zmiana orientacji na narożu wewnętrznym



Kod programu

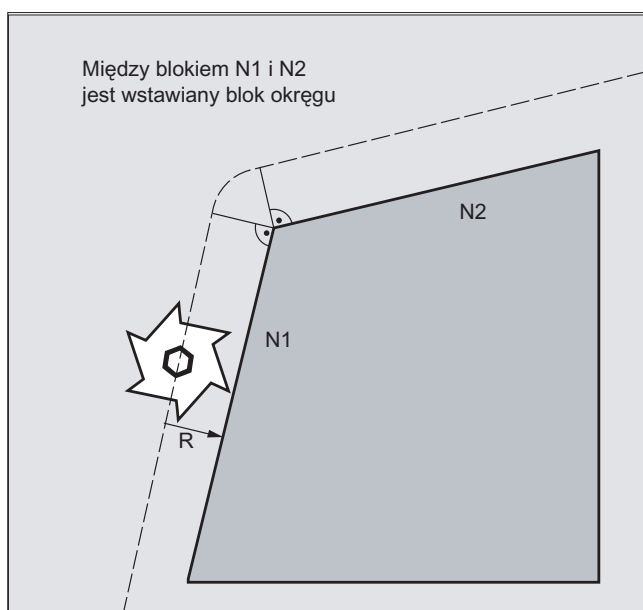
```
ORIC  
N10 X ...Y... Z... G1 F500  
N12 X ...Y... Z... A2=... B2=... C2=...  
N15 X ...Y... Z... A2=... B2=... C2=...
```

Dalsze informacje

Zachowanie się na narożach zewnętrznych

Na narożu zewnętrznym jest zawsze wstawiany blok okręgu o promieniu frezu.

Przy pomocy poleceń programowych `ORIC` lub `ORID` można ustalić, czy zmiany orientacji, które zostały zaprogramowane między blokiem `N1` i `N2`, mają być wykonywane przed rozpoczęciem wstawionego bloku okręgu czy równocześnie z nim.



Jeżeli na narożach zewnętrznych jest konieczna zmiana orientacji, wówczas może ona nastąpić do wyboru równoległe do interpolacji albo oddzielnie od ruchu po torze.

W przypadku `ORID` są najpierw wykonywane wstawione bloki bez ruchu po torze. Blok okręgu jest wstawiany bezpośrednio przed drugim z obydwu bloków ruchu, przy pomocy których jest tworzone naroże.

Jeżeli na narożu zewnętrznym jest wstawionych wiele bloków orientacji i jest wybrane `ORIC`, wówczas ruch kołowy jest dzielony na poszczególne wstawione bloki odpowiednio do zawartych w nich wielkości zmian orientacji.

Wygładzanie orientacji z OSD i OST

Przy wygładzaniu przy pomocy `G642` maksymalne odchylenie dla osi konturowych i osi orientacji nie może być bardzo różne. Mniejsza tolerancja z obydwu określa kształt ruchu wygładzania wzgl. tolerancję kąta, stosunkowo duże wygładzanie przebiegu orientacji, bez konieczności większych odchyień od konturu.

Przez uaktywnienie `OSD` lub `OST` jest możliwe, z zadaną drogą wygładzania lub tolerancją kątową dokonywanie "znakomitego" wygładzania bardzo małych odchyień przebiegu orientacji bez dużych odchyień od konturu.

Wskazówka

W odróżnieniu od wygładzania konturu (i od przebiegu orientacji) przy pomocy `G642` nie jest przy wygładzaniu orientacji przy pomocy `OSD` lub `OST` tworzony własny blok, lecz ruch wygładzania jest wstawiany bezpośrednio dla zaprogramowanych oryginalnych bloków.

Przy pomocy `OSD` lub `OST` nie mogą być wygładzane przejścia między blokami, przy których następuje zmiana rodzaju interpolacji dla orientacji narzędzia (wektor → oś obrotowa, oś obrotowa → wektor). Te przejścia między blokami mogą ewentualnie być wygładzane przy pomocy zwykłych funkcji ścinania naroży `G641`, `G642` wzgl. `G643`.

9.7 Dowlolne nadawanie numerów D, numer ostrza

9.7.1 Dowlolne nadawanie numerów D , numer ostrza (adres CE)

Numer D

Numery D mogą być stosowane jako numery korekcji. Dodatkowo można przez adres CE adresować numer ostrza. Przez parametr systemowy \$TC_DPCE można zapisać numer ostrza.

Ustawienie domyślne: nr korekcji == nr ostrza

Przez dane maszynowe ustala się maksymalną liczbę numerów D (numerów ostrza) i maksymalną liczbę ostrzy na narzędzie (→ producent maszyny). Kolejne polecenia mają sens tylko wtedy, gdy maksymalny numer ostrza (MD18105) został ustalony większy, niż liczba ostrzy na narzędzie (MD18106). Proszę przestrzegać danych producenta maszyny.

Wskazówka

Oprócz względnego nadawania numerów D, numery D mogą być również nadawane jako "płaskie" wzgl. "absolutne" numery D (1-32000) bez odniesienia do numeru T (w ramach funkcji "Płaska struktura numerów D").

Literatura

Podręcznik działania Funkcje podstawowe; Korekcja narzędzia (W1)

9.7.2 Dowlolne nadawanie numerów D: sprawdzenie numerów D (CHKDNO)

Funkcja

Przy pomocy polecenia `CHKDNO` następuje sprawdzenie, czy istniejące numery D są jednoznacznie nadane. Numery D wszystkich narzędzi zdefiniowanych w ramach jednej jednostki TO mogą wystąpić tylko jeden raz. Narzędzia zamienne nie są przy tym uwzględniane.

Składnia

```
state=CHKDNO (Tno1, Tno2, Dno)
```

Znaczenie

state	= TRUE:	Numer D zostały jednoznacznie nadane dla sprawdzonego zakresu.
	=FALSE:	Nastąpiła kolizja numerów D albo parametryzacja jest niepoprawna. Przez Tno1, Tno2 i Dno są przekazywane parametry, które prowadziły do kolizji. Ewaluacja tych danych jest możliwa w programie obróbki.
CHKDNO (Tno1, Tno2)		Są sprawdzane wszystkie numery D wymienionych narzędzi.
CHKDNO (Tno1)		Są sprawdzane wszystkie numery D dla Tno1 w stosunku do wszystkich innych narzędzi.
CHKDNO		Są sprawdzane wszystkie numery D wszystkich narzędzi w stosunku do wszystkich innych narzędzi.

9.7.3 Dowlolne nadawanie numerów D: Zmiana nazw numerów D (GETDNO, SETDNO)**Funkcja**

Numer D muszą być nadawane jednoznacznie. Dwa różne ostrza jednego narzędzia nie mogą mieć tego samego numeru D.

GETDNO

To polecenie daje numer D określonego ostrza (ce) narzędzia o T-numerze t. Jeżeli nie istnieje numer D dla podanych parametrów, wówczas następuje ustawienie d=0. Jeżeli numer D jest nie poprawny, jest zwracana wartość większa, niż 32000.

SETDNO

Przy pomocy tego polecenia przyporządkowuje się wartość d do numeru D ostrza ce narzędzia t. Przez state jest zwracany wynik tej instrukcji (TRUE albo FALSE). Jeżeli nie istnieje zestaw danych do wprowadzonych parametrów, jest zwracane FALSE. Błędy składni powodują alarm. Numeru D nie można explicite ustawić na 0.

Składnia

```
d = GETDNO (t,ce)
state = SETDNO (t,ce,d)
```

Znaczenie

d	Numer D ostrza narzędzia
t	Numer T narzędzia
ce	Numer ostrza (numer CE) narzędzia
state	Podaje, czy można było bezbłędnie wykonać polecenie (TRUE albo FALSE).

Przykład, zmiana nazwy numeru D

Programowanie	Komentarz
\$TC_DP2 [1,2]=120	;
\$TC_DP3 [1,2] = 5.5	;
\$TC_DPCE [1,2] = 3	; Numer ostrza CE
...	;
N10 def int DNrStary, DNrNowy = 17	;
N20 DNrStary = GETDNO(1,3)	;
N30 SETDNO(1,3,DNrNowy)	;

Przez to do ostrza CE=3 jest przyporządkowywana nowa wartość D 17. Teraz sięganie do danych tego ostrza następuje przez numer D 17; zarówno przez parametry systemowe, jak też w programowaniu przy pomocy adresu NC.

9.7.4 Dowlone nadawanie numerów D: Określenie numeru T do zadanego numeru D (GETACTTD)

Funkcja

Przy pomocy polecenia `GETACTTD` określa się do absolutnego numeru D przynależny numer T. Nie ma badania na jednoznaczność. Gdy jest wiele takich samych numerów D w ramach jednej jednostki TO, jest zwracany numer T pierwszego znalezionej narzędzia. Przy stosowaniu "płaskich" numerów D stosowanie tego polecenia nie ma sensu, ponieważ zawsze zostanie zwrócona wartość "1" (w zarządzaniu danymi nie ma numeru T).

Składnia

```
status=GETACTTD (Tnr, Dnr)
```

Znaczenie

Dnr	Numer D, dla którego ma być szukany numer T.
Tnr	Znaleziony numer T
status	Wartość: Znaczenie:
0	Numer T został znaleziony. Tnr otrzymuje wartość numeru T.
-1	Do podanego numeru D nie istnieje numer T; Tnr=0.
-2	Numer D nie jest absolutny. Tnr zawiera wartość pierwszego znalezionej narzędzia, który zawiera numer o wartości Dnr.
-5	Funkcja nie mogła zostać wykonana z innego powodu.

9.7.5 Dowolne nadawanie numerów D: ustawienie nie obowiązywania numerów D (DZERO)

Funkcja

Polecenie `DZERO` służy do wsparcia podczas przezbrajania. Tak zaznaczone zestawy danych korekcyjnych nie są już sprawdzane przez polecenie `CHKDNO`. Aby je ponownie udostępnić, numer D musi zostać ponownie ustawiony przy pomocy `SETDNO`.

Składnia

`DZERO`

Znaczenie

`DZERO` Zaznacza wszystkie numery D jednostki TO jako nie obowiązujące.

9.8 Kinematyka nośnika narzędzi

Warunki

Nośnik narzędzi może orientować narzędzie tylko wówczas we wszystkich możliwych kierunkach w przestrzeni, gdy

- są dwie osie obrotowe v_1 i v_2 .
- osie obrotowe są prostopadłe w stosunku do siebie.
- oś wzdłużna narzędzia jest prostopadła do drugiej osi obrotu v_2 .

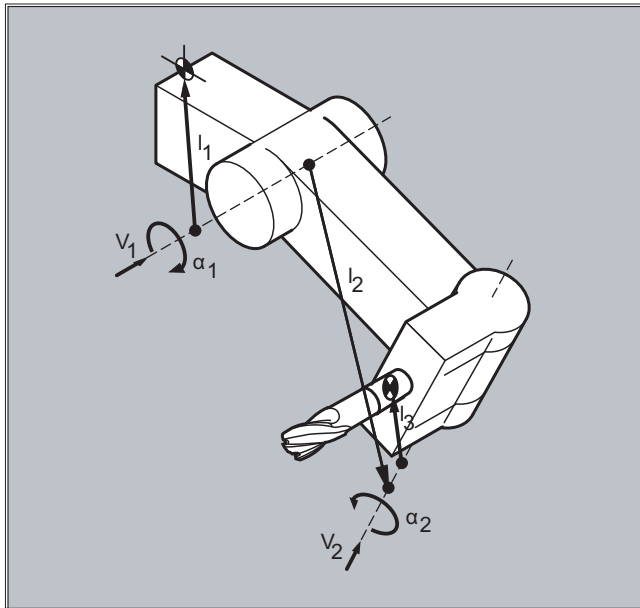
Dodatkowo w przypadku maszyn, przy których muszą dawać się ustawić wszystkie możliwe orientacje, obowiązuje następujący wymóg:

- orientacja narzędzia musi być prostopadła do pierwszej osi obrotowej v_1 .

Funkcja

Kinematyka nośnika narzędzi o maksymalnie dwóch osiach obrotu v_1 lub v_2 jest opisywana przez 17 zmiennych systemowych $\$TC_CARR1[m]$ do $\$TC_CARR17[m]$. Opis nośnika narzędzi składa się z:

- wektorowej odległości od pierwszej osi obrotu do punktu odniesienia nośnika narzędzi I_1 , wektorowej odległości od pierwszej do drugiej osi obrotu I_2 , wektorowej odległości od drugiej osi obrotu do punktu odniesienia narzędzia I_3 .
- wektorów biegunowych obydwu osi obrotowych v_1 , v_2 .
- kątów obrotu α_1 , α_2 wokół obydwu osi. Kąty obrotu są liczone w kierunku ruchu wskazówek zegara patrząc w kierunku wektorów osi obrotu.



Dla maszyn z **rozdzielną kinematyką** (obrotowe jest zarówno narzędzie, jak i obrabiany przedmiot) zmienne systemowe są rozszerzone o wpisy $\$TC_CARR18[m]$ do $\$TC_CARR23[m]$.

Parametry

Funkcja zmiennych systemowych dla orientowanych nośników narzędzi			
Określenie	Składowa x	Składowa y	Składowa z
I_1 wektor offsetu	$\$TC_CARR1[m]$	$\$TC_CARR2[m]$	$\$TC_CARR3[m]$
I_2 wektor offsetu	$\$TC_CARR4[m]$	$\$TC_CARR5[m]$	$\$TC_CARR6[m]$
v_1 oś obrotowa	$\$TC_CARR7[m]$	$\$TC_CARR8[m]$	$\$TC_CARR9[m]$
v_2 oś obrotowa	$\$TC_CARR10[m]$	$\$TC_CARR11[m]$	$\$TC_CARR12[m]$
α_1 kąt obrotu	$\$TC_CARR13[m]$		
α_2 kąt obrotu	$\$TC_CARR14[m]$		
I_3 wektor offsetu	$\$TC_CARR15[m]$	$\$TC_CARR16[m]$	$\$TC_CARR17[m]$

Rozszerzenia zmiennych systemowych dla orientowanych nośników narzędzi			
Określenie	Składowa x	Składowa y	Składowa z
l ₄ wektor offsetu	\$TC_CARR18[m]	\$TC_CARR19[m]	\$TC_CARR20[m]
Identyfikator osi oś obrotowa v ₁ oś obrotowa v ₂	Identyfikatory osi obrotowych v ₁ v ₂ (ustawieniem domyślnym jest zero) \$TC_CARR21[m] \$TC_CARR22[m]		
Typ kinematyki	\$TC_CARR23[m]		
Tool	Typ kinematyki T ->	Typ kinematyki P ->	Typ kinematyki M
Part	Tylko narzędzie jest obrotowe (ustawienie domyślne)	Tylko obrabiany przedmiot jest obrotowy	Obrabiany przedmiot & narzędzie są obrotowe
Mixed mode			
Offset osi obrotowej v ₁ osi obrotowej v ₂	Kąty w stopniach osi obrotowych v ₁ v ₂ przy przyjęciu położenia podstawowego \$TC_CARR24[m] \$TC_CARR25[m]		
Offset kątowy osi obrotowej v ₁ osi obrotowej v ₂	Offset połączenia Hirtha w stopniach osi obrotowych v ₁ v ₂ \$TC_CARR26[m] \$TC_CARR27[m]		
Przyrost kątowy v ₁ oś obrotowa v ₂ oś obrotowa	Przyrost połączenia Hirtha w stopniach osi obrotowych v ₁ v ₂ \$TC_CARR28[m] \$TC_CARR29[m]		
Pozycja min Oś obrotowa v ₁ Oś obrotowa v ₂	Limit softwareowy pozycji minimalnej osi obrotowych v ₁ v ₂ \$TC_CARR30[m] \$TC_CARR31[m]		
Pozycja max Oś obrotowa v ₁ Oś obrotowa v ₂	Limity softwareowe pozycji maksymalnej osi obrotowych v ₁ v ₂ \$TC_CARR32[m] \$TC_CARR33[m]		
Nazwa nośnika narzędzi	Zamiast liczby nośnik narzędzi może otrzymać nazwę. \$TC_CARR34[m]		
Użytkownik: Nazwa osi 1 Nazwa osi 2 Oznaczenie	Zamierzone zastosowanie przez użytkownika w ramach cykli pomiarowych. \$TC_CARR35[m] \$TC_CARR36[m] \$TC_CARR37[m]		
Pozycja	\$TC_CARR38[m]	\$TC_CARR39[m]	\$TC_CARR40[m]
Przesunięcie dokładne	Parametry, które mogą być dodawane do wartości w parametrach bazowych.		
l ₁ wektor offsetu	\$TC_CARR41[m]	\$TC_CARR42[m]	\$TC_CARR43[m]
l ₂ wektor offsetu	\$TC_CARR44[m]	\$TC_CARR45[m]	\$TC_CARR46[m]
l ₃ wektor offsetu	\$TC_CARR55[m]	\$TC_CARR56[m]	\$TC_CARR57[m]
l ₄ wektor offsetu	\$TC_CARR58[m]	\$TC_CARR59[m]	\$TC_CARR60[m]
v ₁ oś obrotowa	\$TC_CARR64[m]		
v ₂ oś obrotowa	\$TC_CARR65[m]		

Wskazówka**Objaśnienia do parametrów**

Przy pomocy "m" jest każdorazowo podawany numer opisywanego nośnika narzędzi \$TC_CARR47 do \$TC_CARR54, jak też \$TC_CARR61 do \$TC_CARR63 nie są zdefiniowane i przy próbie dostępu w celu zapisu albo odczytu prowadzą do alarmu.

Punkty początkowe wzgl. końcowe wektorów odległości na osiach mogą być dowolnie wybierane. Kąty obrotu α_1 , α_2 wokół obydwu osi są w podstawowym stanie nośnika narzędzi zdefiniowane jako 0° . Kinematyka nośnika narzędzi może w ten sposób być opisywana na dowolnie wiele sposobów.

Nośniki narzędzi o tylko jednej albo nie posiadające osi obrotowych mogą być opisywane przez ustawienie na zero wektorów kierunkowych jednej albo obydwu osi obrotowych. W przypadku nośnika narzędzi bez osi obrotowej wektory odległości działają, jak dodatkowe korekcje narzędzia, na których składowe nie wpływa przełączanie płaszczyzn obróbki (G17 do G19).

Rozszerzenia parametrów**Parametry osi obrotowych**

Zmienne systemowe zostały rozszerzone o wpisy \$TC_CARR24[m] do \$TC_CARR33[m] i opisane następująco:

Offset osi obrotowych v_1 , v_2	Zmiana pozycji osi obrotowej v_1 lub v_2 przy podstawowym położeniu orientowanego nośnika narzędzi.
Offset kątowy/ przyrost kątowy Osie obrotowe v_1 , v_2	Offset lub przyrost kątowy połączenia Hirtha osi obrotowych v_1 i v_2 . Zaprogramowany albo obliczony kąt jest zaokrąglany do najbliższej wartości, która wynika przy całkowitym z $\phi = s + n * d$.
Pozycja minimalna i maksymalna Osie obrotowe v_1 , v_2	Pozycja minimalna/maksymalna osi obrotowej, kąt graniczny (Software-Limit) osi obrotowej v_1 i v_2 .

Parametry dla użytkownika

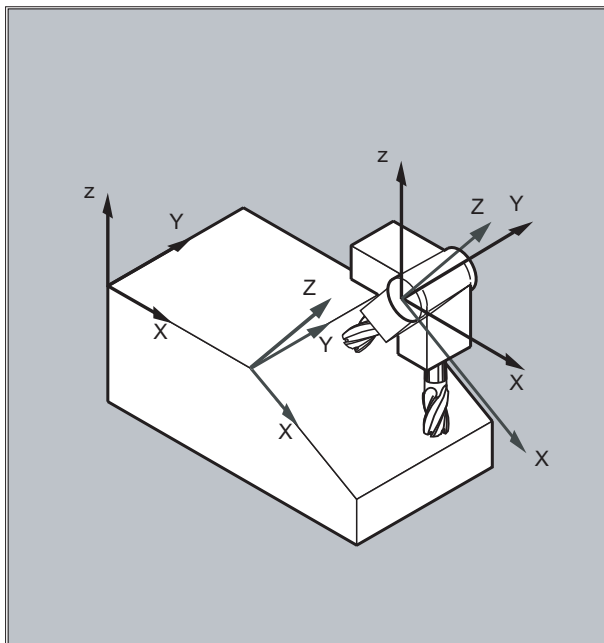
\$TC_CARR34 do \$TC_CARR40 zawierają parametry, które są do dowolnej dyspozycji użytkowników i do w. opr. 6.4 standardowo nie są dalej wykorzystywane w ramach NCK albo nie mają znaczenia.

Parametry przesunięcia dokładnego

\$TC_CARR41 bis \$TC_CARR65 zawierają parametry przesunięcia dokładnego, które mogą być dodawane do wartości w parametrach bazowych. Wartość przesunięcia dokładnego przyporządkowana do parametru bazowego wynika, gdy do numeru parametru zostanie dodana wartość 40.

Przykład

Nośnik narzędzi zastosowany w poniższym przykładzie daje się kompletnie opisać przez obrót wokół osi Y.



Kod programu	Komentarz
N10 \$TC_CARR8 [1,1]=1	; Definicja składowej Y pierwszej osi obrotu nośnika narzędzi 1.
N20 \$TC_DP1 [1,1]= 120	; Definicja frezu trzpieniowego.
N30 \$TC_DP3 [1,1]=20	; Definicja frezu trzpieniowego o długości 20 mm.
N40 \$TC_DP6 [1,1]=5	; Definicja frezu trzpieniowego o promieniu 5 mm.
N50 ROT Y37	; Definicja frame z obrotem 37° wokół osi Y.
N60 X0 Y0 Z0 F10000	; Ruch do pozycji wyjściowej.
N70 G42 CUT2DF TCOFR TCARR=1 T1 D1 X10	; Ustawienie korekcji promienia, korekcji długości narzędzia w obróconym frame, wybór nośnik narzędzi 1, narzędzie 1.
N80 X40	; Przeprowadzenie obróbki przy obrocie 37°.
N90 Y40	
N100 X0	
N110 Y0	
N120 M30	

Dalsze informacje

Kinematyka rozłączona

Dla maszyn z rozdzieloną kinematyką (obrotowe jest zarówno narzędzie, jak i obrabiany przedmiot) zmienne systemowe są rozszerzone o wpisy $\$TC_CARR18 [m]$ do $\$TC_CARR23 [m]$.

Obrotowy stół narzędziowy składający się z:

- odległości wektorowej drugiej osi obrotu v_2 od punktu odniesienia obrotowego stołu narzędziowego I_4 trzeciej osi obrotowej.

Osie obrotowe składające się z:

- obydwu identyfikatorów kanału dla odniesienia osi obrotowych v_{1i} v_2 , do których pozycji ewentualnie następuje dostęp przy określaniu orientacji orientowalnego nośnika narzędzi.

Typ kinematyki z jedną z wartości T, P albo M:

- Typ kinematyki T: Obrotowe jest tylko narzędzie.
- Typ kinematyki P: Obrotowy jest tylko obrabiany przedmiot.
- Typ kinematyki M: Obrotowe jest narzędzie i obrabiany przedmiot.

Skasowanie danych nośnika narzędzi

Przy pomocy $\$TC_CARR1 [0] = 0$ mogą być kasowane dane wszystkich zestawów danych nośników narzędzi.

Typ kinematyki $\$TC_CARR23 [T] = T$ musi zostać podany jako jedna z trzech dopuszczalnych liter dużych albo małych (T,P,M) i z tego powodu nie powinien być kasowany.

Zmiana danych nośnika narzędzi

Każdą z opisanych wartości można zmienić przez przyporządkowanie nowej wartości w programie obróbki. Każdy znak inny niż T, P albo M prowadzi do alarmu przy próbie uaktywnienia orientowalnego nośnika narzędzi.

Odczyt danych nośnika narzędzi

Każda z zapisanych wartości może zostać przeczytana przez przyporządkowanie do zmiennej w programie obróbki.

Przesunięcia dokładne

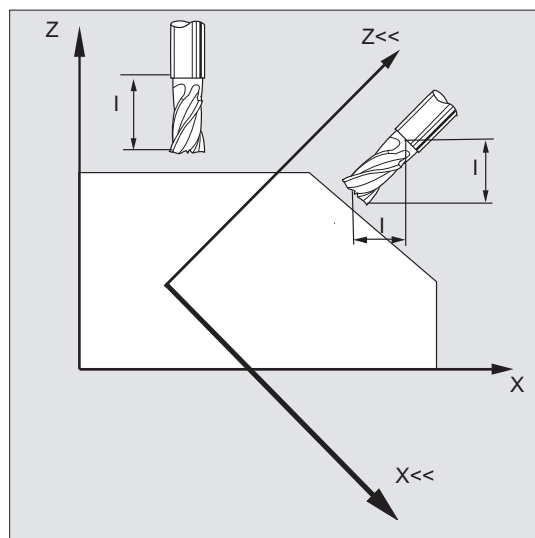
Niedopuszczalna wartość przesunięcia dokładnego zostaje rozpoznana dopiero wtedy, gdy zostanie uaktywniony orientowalny nośnik narzędzi, który zawiera taką wartość i jednocześnie dana nastawcza $SD42974 \$SC_TOCARR_FINE_CORRECTION = TRUE$.

Wielkość dopuszczalnego przesunięcia dokładnego jest przez dane maszynowe ograniczana do maksymalnie dopuszczalnej wartości.

9.9 Korekcja długości narzędzia dla orientowalnych nośników narzędzi (TCARR, TCOABS, TCOFR, TCOFRX, TCOFRY, TCOFRZ)

Funkcja

Ze zmienioną orientacją narzędzia w przestrzeni zmieniają się również jego składowe długości.



Po przebrojeniu, np. przez ustawienie ręczne albo wymianę nośnika narzędzi o stałym ustawieniu przestrzennym, komponenty długości narzędzia muszą dlatego zostać określone na nowo. Następuje to przy pomocy poleceń drogowych TCOABS i TCOFR.

W przypadku orientowalnego nośnika narzędzi aktywnego frame można przy wyborze narzędzia przy pomocy TCOFRZ, TCOFRY i TCOFRX określić kierunek, w którym narzędzie ma być zwrócone.

Składnia

TCARR= [<m>]
 TCOABS
 TCOFR
 TCOFRZ
 TCOFRY
 TCOFRX

Znaczenie

TCARR= [<m>]:	Zażądanie nośnika narzędzi o numerze "m"
TCOABS:	Obliczenie składowych długości narzędzia z aktualnego zorientowania nośnika narzędzi
TCOFR:	Określenie składowych długości narzędzia ze zorientowania aktywnego frame

TCOFRZ:	Orientowalny nośnik narzędzi z aktywnego frame, którego narzędzie wskazuje w kierunku Z
TCOFRY:	Orientowalny nośnik narzędzi z aktywnego frame, którego narzędzie wskazuje w kierunku Y
TCOFRX:	Orientowalny nośnik narzędzi z aktywnego frame, którego narzędzie wskazuje w kierunku X

Dalsze informacje

Korekcja długości narzędzia z orientacji nośnika (TCOABS)

TCOABS oblicza korekcję promienia narzędzia z aktualnych kątów orientacji nośnika narzędzi; zapisanych w zmiennych systemowych \$TC_CARR13 i \$TC_CARR14.

Odnosnie definicji kinematyki nośnika narzędzi przy pomocy zmiennych systemowych patrz "Kinematyka nośnika narzędzi (Strona 433)".

W celu ponownego obliczenia korekcji długości narzędzia przy zmianie frame, wrzeczono musi zostać jeszcze raz wybrane.

Kierunek narzędzia z aktywnego frame

Orientowalny nośnik narzędzi można tak ustawić, że narzędzie będzie zwrócone w następujących kierunkach:

- przy pomocy TCOFR lub TCOFRZ w kierunku Z
- przy pomocy TCOFRY w kierunku Y
- przy pomocy TCOFRX w kierunku X

Przełączenie między TCOFR i TCOABS powoduje nowe obliczenie korekcji długości narzędzia.

Zażądanie nośnika narzędzi (TCARR)

Przy pomocy TCARR następuje z numerem nośnika narzędzi m zażądanie jego danych geometrycznych (pamięć korekcji).

Przy pomocy m=0 następuje cofnięcie wyboru aktywnego nośnika narzędzi.

Dane geometryczne nośnika narzędzi stają się aktywne dopiero po wywołaniu narzędzia. Wybrane narzędzie pozostaje aktywne po zmianie nośnika narzędzi.

Aktualne dane geometryczne nośnika narzędzi mogą być również definiowane w programie obróbki przez odpowiednie zmienne systemowe.

Nowe obliczenie korekcji długości narzędzia (TCOABS) przy zmianie frame

W celu ponownego obliczenia korekcji długości narzędzia przy zmianie frame, wrzeczono musi zostać jeszcze raz wybrane.

Wskazówka

Orientacja narzędzia musi zostać ręcznie dopasowana do aktywnego frame.

Przy obliczaniu korekcji długości narzędzia są w ramach kroku pośredniego obliczane również kąty obrotu nośnika narzędzi. Ponieważ w przypadku nośników narzędzi o dwóch osiach obrotu zazwyczaj istnieją dwie pary kątów obrotu, przy pomocy których orientacja narzędzia może zostać dopasowana do aktywnego frame, wartości zapisane w zmiennych systemowych muszą przynajmniej w przybliżeniu odpowiadać mechanicznie ustawionym kątom obrotu.

Wskazówka

Orientacja narzędzia

Kątów skrętu obliczonych przez zorientowanie frame sterowanie nie jest w stanie sprawdzić pod względem możliwości ustawienia na maszynie

Jeżeli osie obrotu nośnika narzędzi są konstrukcyjnie tak usytuowane, że orientacja narzędzia, obliczona przez orientację frame, nie może zostać osiągnięta, zostanie wywołany alarm.

Kombinacja korekcji dokładnej narzędzia i funkcji dotyczących korekcji długości narzędzia przy ruchomych nośnikach narzędzi jest niedopuszczalna. Przy próbie równoczesnego wywołania obydwu funkcji następuje komunikat błędu.

Przy pomocy `TOFRAME` jest możliwe zdefiniowanie frame na podstawie kierunku zorientowania wybranego nośnika narzędzi. Dokładniejsze informacje patrz punkt "Frame".

Przy aktywnej transformacji orientacji (transformacja 3-, 4-, 5-osiowa) można wybrać nośnik narzędzi o orientacji odbiegającej od położenia zerowego, bez wyprowadzenia przy tym alarmu.

Przekazywane parametry cykli standardowych i pomiarowych

Dla przekazywanych parametrów cykli standardowych i pomiarowych obowiązują zdefiniowane zakresy wartości.

W przypadku wartości kątowych zakres wartości jest ustalony następująco:

- Obrót wokół 1. osi geometrycznej: -180 stopni do +180 stopni
- Obrót wokół 2. osi geometrycznej: -90 stopni do +90 stopni
- Obrót wokół 3. osi geometrycznej: -180 stopni do +180 stopni

Patrz punkt Frame "Obrót programowany (ROT, AROT, RPL)".

Wskazówka

Przy przekazywaniu wartości kątowych do cyklu standardowego lub pomiarowego należy przestrzegać:

Wartości mniejsze, niż dokładność obliczania przez NC należy zaokrąglić do zera!

Dokładność obliczania przez NC dla pozycji końcowej jest ustalona w danej maszynowej:

MD10210 \$MN_INT_INCR_PER_DEG

9.10 Korekcja długości narzędzia online (TOFFON, TOFFOF)

Funkcja

Poprzez zmienną systemową \$AA_TOFF[<n>] można trójwymiarowo w czasie rzeczywistym nałożyć efektywne długości narzędzia odpowiednio do trzech jego kierunków.

Jako indeks <n> są stosowane trzy identyfikatory osi geometrycznych. Przez to liczba aktywnych kierunków korekcji jest ustalona przez osie geometryczne aktywne w tym samym czasie.

Wszystkie korekcje mogą być równocześnie aktywne.

Funkcja korekcji długości narzędzia online daje się stosować przy:

- Transformacja orientacji TRAORI
- Orientowalne nośniki narzędzi TCARR

Wskazówka

Korekcja długości narzędzia online jest **opcją**, która przedtem musi zostać udostępniona. Funkcja ta ma sens tylko w połączeniu z aktywną transformacją orientacji albo aktywnym orientowalnym nośnikiem narzędzi.

Składnia

```
TRAORI
TOFFON(<kierunek korekcji>[,<wartość offsetu>])
WHEN TRUE DO $AA_TOFF[<kierunek korekcji>]           ; W akcjach synchronicznych.
...
TOFFOF(<kierunek korekcji>)
```

Dalsze objaśnienia dot. programowania korekcji długości narzędzia online w akcjach synchronicznych ruchu patrz "Akcje synchroniczne (Strona 561)".

Znaczenie

TOFFON:	Uaktywnienie korekcji długości narzędzia online
<kierunek korekcji>:	Kierunek narzędzia (x, y, z), w którym ma działać korekcja długości narzędzia online.
<wartość offsetu>:	Przy uaktywnieniu można dla odpowiedniego kierunku korekcji podać offset, który jest natychmiast realizowany.
TOFFOF:	Cofnięcie korekcji długości narzędzia online
	Wartości korekcji w podanym kierunku korekcji są cofane i jest wyzwalane zatrzymanie przebiegu wyprzedzającego.

Przykłady

Przykład 1: Wybór korekcji długości narzędzia

Kod programu	Komentarz
MD21190 \$MC_TOFF_MODE =1	; Następuje ruch do wartości
MD21194 \$MC_TOFF_VELO[0] =1000	absolutnych.
MD21196 \$MC_TOFF_VELO[1] =1000	
MD21194 \$MC_TOFF_VELO[2] =1000	
MD21196 \$MC_TOFF_ACCEL[0] =1	
MD21196 \$MC_TOFF_ACCEL[1] =1	
MD21196 \$MC_TOFF_ACCEL[2] =1	
N5 DEF REAL XOFFSET	
N10 TRAORI(1)	; Transformacja wł.
N20 TOFFON(Z)	; Uaktywnienie korekcji długości narzędzia online dla kierunku Z narzędzia.
N30 WHEN TRUE DO \$AA_TOFF[Z]=10 G4 F5	; Dla kierunku Z narzędzia jest interpolowana korekcja długości narzędzia wynosząca 10.
...	
N100 XOFFSET=\$AA_TOFF_VAL[X]	; Przyporządkowanie aktualnej korekcji w kierunku X.
N120 TOFFON(X,-XOFFSET) G4 F5	; Dla kierunku X narzędzia korekcja długości narzędzia jest ponownie cofana na 0.

Przykład 2: Cofnięcie wyboru korekcji długości narzędzia

Kod programu	Komentarz
N10 TRAORI(1)	; Transformacja wł.
N20 TOFFON(X)	; Uaktywnienie korekcji długości narzędzia online dla kierunku X.
N30 WHEN TRUE DO \$AA_TOFF[X]=10 G4 F5	; Dla kierunku X narzędzia jest interpolowana korekcja długości narzędzia wynosząca 10.
...	
N80 TOFFOF(X)	; Offset pozycji kierunku X narzędzia jest kasowany: ...\$AA_TOFF[X]=0 Nie jest wykonywany ruch w żadnej osi. Do aktualnej pozycji w WKS jest doliczany offset pozycji odpowiednio do aktualnej orientacji.

Dalsze informacje

Przygotowanie bloków

Przy przetwarzaniu bloków w przebiegu wyprzedzającym jest uwzględniany działający w przebiegu głównym aktualny offset długości narzędzia. Aby móc w dużym stopniu korzystać z maksymalnie dopuszczalnych prędkości w osiach, jest wymagane zatrzymanie przygotowywania bloków przez zatrzymanie przebiegu wyprzedzającego `STOPRE`, podczas gdy jest budowany offset narzędzia.

Offset narzędzia jest w momencie przebiegu wyprzedzającego znany też zawsze wtedy, gdy korekcje długości narzędzia po starcie programu nie są już zmieniane, albo gdy po zmianie korekcji długości narzędzia zostało przetworzonych więcej bloków, niż może pomieścić bufor IPO między przebiegiem wyprzedzającym i przebiegiem głównym.

Zmienna `$AA_TOFF_PREP_DIFF`

Wielkość różnicy między aktualną korekcją działającą w interpolatorze i korekcją, która działała w chwili przetwarzania bloku, można odczytać w zmiennej `$AA_TOFF_PREP_DIFF[<n>]`.

Ustawienie danych maszynowych i danych ustawczych

Dla korekcji długości narzędzia online są do dyspozycji następujące dane systemowe:

- MD20610 `$MC_ADD_MOVE_ACCEL_RESERVE` (rezerwa przyspieszenia dla ruchu nałożonego)
- MD21190 `$MC_TOFF_MODE`

Zawartość zmiennej systemowej `$AA_TOFF[<n>]` jest realizowana albo całkowana, jako wartość absolutna.

- MD21194 `$MC_TOFF_VELO` (prędkość korekcji długości narzędzia online)
- MD21196 `$MC_TOFF_ACCEL` (przyspieszenie korekcji długości narzędzia online)
- Dana ustawcza do zadania wartości granicznych:
SD42970 `$SC_TOFF_LIMIT` (górną granicą wartości korekcji długości narzędzia)

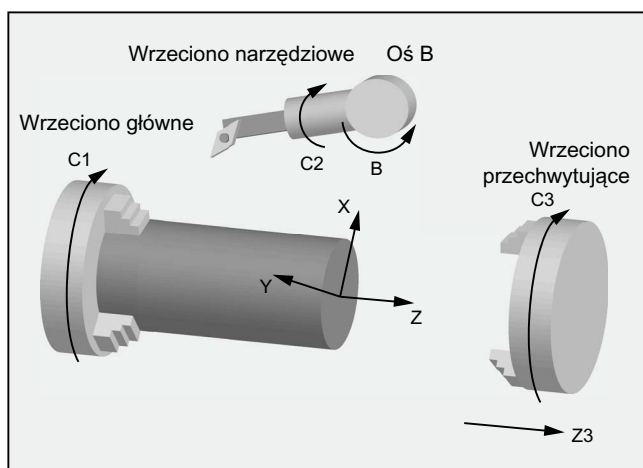
Literatura:

Podręcznik działania Funkcje specjalne; F2 Transformacje wieloosiowe

9.11 Modyfikacja danych skrawania w przypadku narzędzi obrotowych (CUTMOD)

Funkcja

Przy pomocy funkcji "Modyfikacja danych ostrzy w przypadku narzędzi obrotowych" mogą przy korekcji narzędzia być uwzględniane zmienione warunki geometryczne, które wynikają przy obrocie narzędzi (przeważnie narzędzi tokarskich, ale też wiertarskich i frezarskich) w stosunku do obrabianego przedmiotu.



Rysunek 9-1 Narzędzie obrotowe w przypadku tokarki

Aktualny obrót narzędzia jest przy tym zawsze określany z aktualnie aktywnego orientowalnego nośnika narzędzi (patrz "Korekcja długości narzędzia dla orientowalnych nośników narzędzi (Strona 439)").

Funkcja jest uaktywniana przy pomocy polecenia CUTMOD.

Składnia

CUTMOD=<wartość>

Znaczenie

CUTMOD	Polecenie do włączenia funkcji "Modyfikacja danych ostrzy w przypadku narzędzi obrotowych"
<wartość>	Do polecenia CUTMOD mogą zostać przyporządkowane następujące wartości:
0	Aktywność funkcji jest wyłączona. Wartości dawane przez zmienne systemowe \$P_AD... są równe korespondującym parametrom narzędzia.
> 0	Funkcja jest uaktywniana, gdy jest aktywny orientowalny nośnik narzędzi o podanym numerze, tzn. uaktywnienie jest powiązane z określonym orientowalnym nośnikiem narzędzi. Wartości dawane przez zmienne systemowe \$P_AD... są ewentualnie modyfikowane w stosunku do korespondujących parametrów narzędzia zależnie od aktywnego obrotu. Wyłączenie aktywności określonego orientowalnego nośnika narzędzi wyłącza aktywność funkcji tymczasowo, uaktywnienie innego orientowalnego nośnika narzędzi wyłącza jej aktywność w sposób ciągły. W pierwszym przypadku funkcja jest dlatego ponownie uaktywniana przy ponownym wybraniu tego samego orientowalnego nośnika narzędzi, w drugim przypadku jest konieczny ponowny wybór, również wtedy, gdy w późniejszym czasie orientowalny nośnik narzędzi o podanym numerze zostanie ponownie uaktywniony. Reset nie wpływa na funkcję.
-1	Funkcja jest uaktywniana zawsze wtedy, gdy orientowalny nośnik narzędzi jest aktywny. Przy zmianie nośnika narzędzi albo przy cofnięciu jego wyboru i późniejszym ponownym wyborze CUTMOD nie musi być ponownie ustawiane.
-2	Funkcja jest uaktywniana zawsze wtedy, gdy jest aktywny orientowalny nośnik narzędzi, którego numer jest równy numerowi aktualnie aktywnego orientowalnego nośnika narzędzi. Gdy żaden orientowalny nośnik narzędzi nie jest aktywny, jest to równoznaczne z CUTMOD=0. Gdy orientowalny nośnik narzędzi jest aktywny, jest to jednoznaczne z bezpośrednim podaniem aktualnego numeru nośnika narzędzi.
< -2	Wartości mniejsze od -2 są ignorowane, tzn. ten przypadek jest tak traktowany, jakby CUTMOD nie zaprogramowano. Wskazówka: Ten zakres wartości nie powinien być stosowany, ponieważ jest on zarezerwowany dla ewentualnych późniejszych rozszerzeń.

Wskazówka

SD42984 \$SC_CUTDIRMOD

Funkcja uaktywniana przez polecenie CUTMOD zastępuje funkcję uaktywnianą przez daną ustawczą SD42984 \$SC_CUTDIRMOD. Ta funkcja jest nadal bez zmian do dyspozycji. Ponieważ jednak nie ma sensu używanie obydwu funkcji równolegle, może zostać ona uaktywniona tylko wtedy, gdy CUTMOD jest równe zero.

Przykład

Poniższy przykład odnosi się do narzędzia o położeniu 3 ostrza i orientowanym nośniku narzędzi, który może obracać narzędzie wokół osi B.

Wartości liczbowe w komentarzach podają każdorazowo pozycje końcowe bloków we współrzędnych maszyny (MKS) w kolejności X, Y, Z.

Kod programu	Komentarz		
N10 \$TC_DP1 [1,1]=500			
N20 \$TC_DP2 [1,1]=3	; Położenie ostrza		
N30 \$TC_DP3 [1,1]=12			
N40 \$TC_DP4 [1,1]=1			
N50 \$TC_DP6 [1,1]=6			
N60 \$TC_DP10 [1,1]=110	; Kąt uchwytu		
N70 \$TC_DP11 [1,1]=3	; Kierunek skrawania		
N80 \$TC_DP24 [1,1]=25	; Kąt przystawienia		
N90 \$TC_CARR7 [2]=0 \$TC_CARR8 [2]=1 \$TC_CARR9 [2]=0	; Oś B		
N100 \$TC_CARR10 [2]=0 \$TC_CARR11 [2]=0 \$TC_CARR12 [2]=1	; Oś C		
N110 \$TC_CARR13 [2]=0			
N120 \$TC_CARR14 [2]=0			
N130 \$TC_CARR21 [2]=X			
N140 \$TC_CARR22 [2]=X			
N150 \$TC_CARR23 [2]="M"			
N160 TCOABS CUTMOD=0			
N170 G18 T1 D1 TCARR=2	X	Y	Z
N180 X0 Y0 Z0 F10000	; 12.000	0.000	1.000
N190 \$TC_CARR13 [2]=30			
N200 TCARR=2			
N210 X0 Y0 Z0	; 10.892	0.000	-5.134
N220 G42 Z-10	; 8.696	0.000	-17.330
N230 Z-20	; 8.696	0.000	-21.330
N240 X10	; 12.696	0.000	-21.330
N250 G40 X20 Z0	; 30.892	0.000	-5.134
N260 CUTMOD=2 X0 Y0 Z0	; 8.696	0.000	-7.330
N270 G42 Z-10	; 8.696	0.000	-17.330
N280 Z-20	; 8.696	0.000	-21.330
N290 X10	; 12.696	0.000	-21.330
N300 G40 X20 Z0	; 28.696	0.000	-7.330
N310 M30			

Objaśnienia:

W bloku N180 jest najpierw wybierane narzędzie przy CUTMOD=0 i nie obróconym orientowalnym nośniku narzędzi. Ponieważ wszystkie wektory offsetowe orientowalnego nośnika narzędzi wynoszą 0, następuje ruch do pozycji, która odpowiada długościom narzędzi podanych w \$TC_DP3 [1, 1] i \$TC_DP4 [1, 1].

W bloku N200 jest uaktywniany orientowalny nośnik narzędzi z obrotem 30° wokół osi B. Ponieważ położenie ostrza z powodu CUTMOD=0 nie jest zmodyfikowane, miarodajny jest jak dotychczas stary punkt odniesienia ostrza. Dlatego w bloku N210 następuje ruch do pozycji, która zachowuje stary punkt odniesienia ostrza w punkcie zerowym (tzn. wektor (1, 12) jest w płaszczyźnie Z/X obracany o 30°).

W bloku N260 w odróżnieniu od bloku N200 działa CUTMOD=2. Ze względu na obrót orientowalnego nośnika narzędzi zmodyfikowanym położeniem ostrza staje się 8. Z tego wynikają również odmienne pozycje osi.

W blokach N220 wzgl. N270 jest każdorazowo uaktywniana korekcja promienia narzędzia (WRK). Różne położenie ostrza w obydwu fragmentach programu nie ma wpływu na pozycje końcowe bloków, w których jest aktywna WRK, odpowiednie pozycje są dlatego identyczne. Dopiero w blokach cofnięcia wyboru N260 wzgl. N300 znów oddziałują różne położenia ostrza.

Dalsze informacje

Działanie zmodyfikowanych danych ostrza

Zmodyfikowane położenie ostrza i zmodyfikowany punkt odniesienia ostrza działają natychmiast przy zaprogramowaniu również dla już aktywnego narzędzia. Wybór narzędzia nie jest do tego konieczny.

Wpływ aktywnej płaszczyzny roboczej

Dla określenia zmodyfikowanego położenia ostrza, kierunku skrawania i kąta uchwytu wzgl. kąta przyłożenia jest miarodajne widzenie ostrza w każdorazowo aktywnej płaszczyźnie (G17 - G19).

Gdy jednak dana ustawcza SD42940 \$SC_TOOL_LENGTH_CONST (zmiana składowych długości narzędzia przy zmianie płaszczyzny) zawiera poprawną wartość nierówną zeru (plus lub minus 17, 18 albo 19), wówczas jej treść określa płaszczyznę, w której są traktowane odnośne wielkości.

Zmienne systemowe

Są do dyspozycji następujące zmienne systemowe:

Zmienne systemowe	Znaczenie
\$P_CUTMOD_ANG / \$AC_CUTMOD_ANG	Daje (nie zaokrąglony) kąt w aktywnej płaszczyźnie obróbki, który został przyjęty za podstawę dla modyfikacji danych ostrza (położenie ostrza, kierunek skrawania, kąt przstawienia i kąt uchwytu) w przypadku funkcji uaktywnionych przy pomocy CUTMOD wzgl. \$SC_CUTDIRMOD. \$P_CUTMOD_ANG odnosi się do aktualnego stanu w przebiegu wyprzedzającym, \$AC_CUTMOD_ANG do aktualnego bloku przebiegu głównego.

9.11 Modyfikacja danych skrawania w przypadku narzędzi obrotowych (CUTMOD)

Zmienne systemowe	Znaczenie
\$P_CUTMOD / \$AC_CUTMOD	<p>Czyta aktualnie obowiązującą wartość, która została ostatnio zaprogramowana poleceniem CUTMOD (numer nośnika narzędzi, dla którego ma zostać uaktywniona modyfikacja danych ostrza).</p> <p>Jeżeli ostatnio zaprogramowana wartość była CUTMOD = -2 (uaktywnienie z aktualnie aktywnym orientowalnym nośnikiem narzędzi), wówczas jest w \$P_CUTMOD zwracana nie wartość -2, lecz numer orientowalnego nośnika narzędzi, aktywnego w chwili programowania.</p> <p>\$P_CUTMOD odnosi się do aktualnego stanu w przebiegu wyprzedzającym, \$AC_CUTMOD do aktualnego bloku przebiegu głównego.</p>
\$P_CUT_INV / \$AC_CUT_INV	<p>Daje wartość TRUE, gdy narzędzie jest tak obrócone, że kierunek obrotów wrzeczona musi zostać odwrócony. W tym celu w bloku, do którego odnosi się każdorazowa operacja odczytu, muszą być spełnione następujące cztery warunki:</p> <ol style="list-style-type: none"> 1. Jest aktywne narzędzie tokarskie albo szlifierskie (typy narzędzi 400 do 599 i / albo SD42950 \$SC_TOOL_LENGTH_TYPE = 2). 2. Sterowanie ostrzami zostało uaktywnione przy pomocy polecenia językowego CUTMOD. 3. Jest aktywny orientowalny nośnik narzędzi, który został określony przez numeryczną wartość CUTMOD. 4. Orientowalny nośnik narzędzi obraca narzędzie wokół osi w płaszczyźnie obróbki (typowo oś C) tak, by uzyskana normalna ostrza narzędzia w stosunku do położenia wyjściowego była obrócona o więcej, niż 90° (typowo 180°). <p>Jeżeli przynajmniej jeden z wymienionych warunków nie jest spełniony, treść zmiennej jest FALSE. Dla narzędzi, których położenie ostrza nie jest zdefiniowane, wartość zmiennej jest zawsze FALSE.</p> <p>\$P_CUT_INV odnosi się do aktualnego stanu w przebiegu wyprzedzającym, a \$AC_CUT_INV do aktualnego bloku przebiegu głównego.</p>

Wszystkie zmienne przebiegu głównego (\$AC_CUTMOD_ANG, \$AC_CUTMOD i \$AC_CUT_INV) mogą być czytane w akcjach synchronicznych. Dostęp w celu odczytu z przebiegu wyprzedzającego generuje zatrzymanie przebiegu wyprzedzającego.

Zmodyfikowane dane ostrza:

W przypadku gdy jest aktywny obrót narzędzia, zmodyfikowane dane są udostępniane w następujących zmiennych systemowych:

Zmienna systemowa	Znaczenie
\$P_AD[2]	Położenie ostrza
\$P_AD[10]	Kąt uchwytu
\$P_AD[11]	Kierunek skrawania
\$P_AD[24]	Kąt przystawienia

Wskazówka

Dane są w stosunku do korespondujących parametrów narzędzia (\$TC_DP2[... , ...] itd.) zawsze wtedy modyfikowane, gdy funkcja "modyfikacja danych ostrza w przypadku narzędzi obrotowych" została uaktywniona poleceniem `CUTMOD` i jest aktywny orientowalny nośnik narzędzi, który powoduje obrót narzędzia.

Literatura

Dalsze informacje dot funkcji "modyfikacja danych ostrza w przypadku narzędzi obrotowych" patrz:

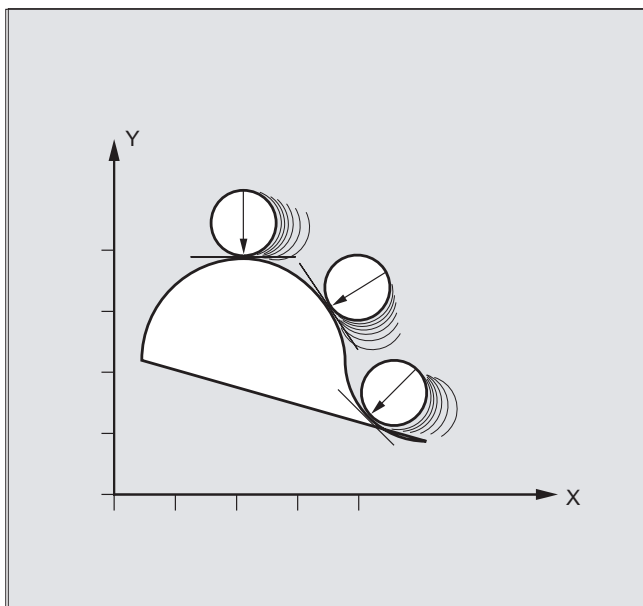
Podręcznik działania Funkcje podstawowe; Korekcja narzędzia (W1)

Zachowanie się w ruchu po torze

10.1 Sterowanie styczne (TANG, TANGON, TANGOF, TLIFT, TANGDEL)

Funkcja

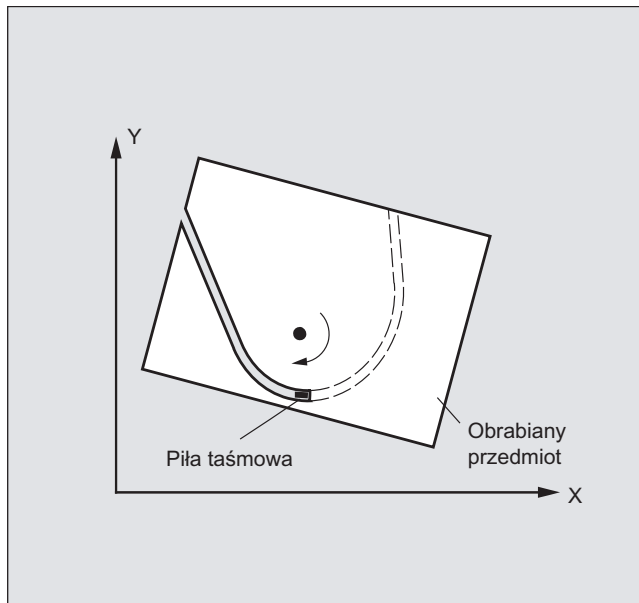
Oś nadążna jest aktualizowana według stycznej po torze ustalonym przez osie wiodące. Przez to narzędzie może zostać ustawione równoległe do konturu. Przez kąt zaprogramowany w instrukcji `TANGON` można przyłożyć narzędzie w stosunku do stycznej.



Zastosowanie

Sterowanie styczne może np. być stosowane przy:

- stycznym przyłożeniu narzędzia obrotowego przy cięciu
- aktualizacji ustawienia obrabianego przedmioty w przypadku piły taśmowej (patrz poniższy rysunek)
- przystawieniu narzędzia obciążającego do ściernicy
- przyłożeniu kółka tnącego przy cięciu szkła albo papieru
- stycznym doprowadzaniu drutu przy spawaniu 5-osiowym



Składnia

Definicja aktualizacji stycznej:

TANG(<oś nadożna>, <oś wiodaca1>, <oś wiodaca2>, <współczynnik sprzężenia>, <KS>, <Opt>)

Włączenie sterowania stycznego:

TANGON(<oś nadożna>, <kąt>, <odległość>, <tolerancja kątowa>)

Wyłączenie sterowania stycznego:

TANGOF(<oś nadożna>)

Włączenie funkcji "Wstawianie bloku pośredniego na narożach konturu":

TLIFT(<oś nadożna>)

Instrukcja TLIFT jest podawana bezpośrednio po przyporządkowaniu osi przy pomocy TANG(...).

Wyłączenie funkcji "Wstawianie bloku pośredniego na narożach konturu":

Powtórzyć instrukcję TANG(...) bez TLIFT(<oś nadożna>).

Skasowanie definicji aktualizacji stycznej:

TANGDEL(<oś nadożna>)

Istniejąca, zdefiniowana przez użytkownika aktualizacja styczna musi zostać skasowana, gdy ma zostać zdefiniowana nowa aktualizacja styczna z taką samą osią nadożną w wywołaniu przygotowawczym TANG. Skasowanie jest możliwe tylko wtedy, gdy sprzężenie jest wyłączone przy pomocy TANGOF (<oś nadożna>).

Znaczenie

TANG:	Instrukcja przygotowawcza dla definicji aktualizacji stycznnej
TANGON:	Włączenie sterowania stycznego dla podanej osi nadążnej
TANGOF:	Wyłączenie sterowania stycznego dla podanej osi nadążnej
TLIFT:	Włączenie funkcji "Wstawianie bloku pośredniego na narożach konturu"
TANGDEL:	Skasowanie definicji aktualizacji stycznnej
<oś nadążna>:	Oś nadążna: stycznie aktualizowana dodatkowa oś obrotowa
<oś wiodąca1>, <oś wiodąca2>:	Osie wiodące: osie uczestniczące w tworzeniu konturu, z których jest określana stycznna dla aktualizacji
<współczynnik sprzężenia>:	Współczynnik sprzężenia: zależność między zmianą kąta stycznnej i osi aktualizowanej Ustawienie 1 domyślne: Wskazówka: Współczynnik sprzężenia wynoszący 1 nie musi być explicite programowany.
<KS>:	Symbol literowy układu współrzędnych "B": Bazowy układ współrzędnych (ustawienie domyślne) Wskazówka: <KS> = "B" nie musi być explicite programowane. "W": Układ współrzędnych obrabianego przedmiotu (nieдоступny)
<Opt>:	Optymalizacja "S": Standard (ustawienie domyślne) Wskazówka: <Opt> = "S" nie musi być explicite programowane. "P": Automatyczne dopasowanie przebiegu czasowego osi stycznnej i konturu Wskazówka: Przy pomocy <Opt> = "P" jest przy ograniczeniu prędkości osi wiodących również uwzględniana dynamika osi nadążnej. To ustawienie jest zalecane przede wszystkim przy zastosowaniu transformacji kinematycznych.
<kąt>:	Kąt offsetu osi nadążnej
<odległość>:	Droga ścinania dla osi nadążnej (wymagana przy <Opt> = "P")
<tolerancja kątowa>:	Tolerancja kątowa osi nadążnej (opcjonalnie; ewaluacja tylko przy <Opt> = "P") Wskazówka: Parametry <odległość> i <tolerancja kątowa> w zamierzony sposób ograniczają błąd pomiędzy osią aktualizowaną i styczną osi wiodących.

Przykłady

Przykład 1: Definicja i włączenie aktualizacji styczniej

Kod programu	Komentarz
N10 TANG (C, X, Y, 1, "B", "P")	; Definicja aktualizacji styczniej: oś obrotowa C powinna postępować za osiami geometrycznymi X i Y.
N20 TANGON (C, 90)	; Oś C jest osią nadażną. Jest ona przy każdym ruchu osi uczestniczących w tworzeniu konturu obracana do pozycji 90° w stosunku do styczniej do toru.
...	

Wskazówka

Programowania uproszczone

TANG (C, X, Y, 1, "B", "P") można programować w sposób uproszczony jako TANG (C, X, Y, , , "P").

Przykład 2: Zmiana płaszczyzny

Kod programu	Komentarz
N10 TANG (A, X, Y, 1)	; 1. definicja aktualizacji styczniej.
N20 TANGON (A)	; Uaktywnienie sprzężenia.
N30 X10 Y20	; Promień
...	
N80 TANGOF (A)	; Wyłączenie 1. sprzężenia.
N90 TANGDEL (A)	; Skasowanie 1. definicji.
...	
TANG (A, X, Z)	; 2. definicja aktualizacji styczniej.
TANGON (A)	; Uaktywnienie nowego sprzężenia.
...	
N200 M30	

Przykład 3: Przełączenie osi geometrycznych i TANGDEL

Alarm nie jest generowany.

Kod programu	Komentarz
N10 GEOAX (2, Y1)	; Y1 jest osią geometryczną 2.
N20 TANG (A, X, Y)	; 1. definicja aktualizacji styczniej.
N30 TANGON (A, 90)	; Uaktywnienie aktualizacji z Y1
N40 G2 F8000 X0 Y0 I0 J50	
N50 TANGOF (A)	; Wyłączenie aktywności aktualizacji z Y1.
N60 TANGDEL (A)	; Skasowanie 1. definicji.
N70 GEOAX (2, Y2)	; Y2 jest nową osią geometryczną 2.

Kod programu	Komentarz
N80 TANG(A, X, Y)	; 2. definicja aktualizacji stycznnej.
N90 TANGON(A, 90)	; Uaktywnienie aktualizacji z Y2.
...	

Przykład 4: Aktualizacja stycznna z optymalizacją automatyczną

Y1 jest osią geometryczną 2.

Kod programu	Komentarz
...	
N80 G0 C0	
N100 F=50000	
N110 G1 X1000 Y500	
N120 TRAORI	
N130 G642	; Wygładzanie przy dotrzymaniu maksymalnego dozwolonego odchylenia od toru.
N171 TRANS X50 Y50	
N180 TANG(C, X, Y, 1, , "P")	; Definicja aktualizacji stycznnej z automatyczną optymalizacją prędkości ruchu po torze.
N190 TANGON(C, 0, 5.0, 2.0)	; Włączenie aktualizacji stycznnej z optymalizacją automatyczną: droga wygładzania 5 mm, tolerancja katowa 2 stopnie.
N210 G1 X1310 Y500	
N215 G1 X1420 Y500	
N220 G3 X1500 Y580 I=AC(1420) J=AC(580)	
N230 G1 X1500 Y760	
N240 G3 X1360 Y900 I=AC(1360) J=AC(760)	
N250 G1 X1000 Y900	
N280 TANGOF(C)	
N290 TRAFOOF	
N300 M02	

Dalsze informacje

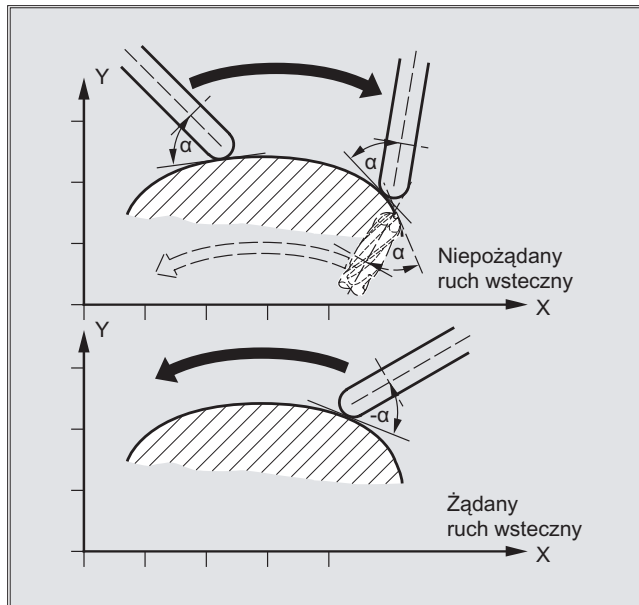
Definiowanie osi nadążnej i osi wiodącej

Definicja osi nadążnych i wiodących następuje przy pomocy TANG.

Współczynnik sprzężenia podaje zależność między zmianą kąta stycznnej i osi nadążnej. Jego wartość wynosi z reguły 1 (ustawienie domyślne).

Kąt graniczny przez ograniczenie obszaru pracy

Przy ruchach po torze prowadzonych w jedną i drugą stronę stycznna skacze w punkcie nawrotnym toru o 180° , odpowiednio zmienia się ustawienie osi nadążnej. Z reguły to zachowanie się nie ma sensu: Ruch powrotny powinien być wykonywany w tym samym ujemnym kącie offsetu co ruch tamtą stronę.



W tym celu musi zostać ograniczony obszar pracy osi nadążnej (G_{25} , G_{26}). Ograniczenie obszaru pracy musi być aktywne w chwili nawrotu toru ($WALIMON$). Gdy kąt offsetu leży poza ograniczeniem obszaru pracy, następuje próba dojścia z offsetem ujemnym z powrotem do dopuszczalnego zakresu pracy.

Wstawianie bloku pośredniego na narożach konturu (TLIFT)

Na narożu konturu skokowo zmienia się stycznna, a przez to pozycja zadana osi aktualizowanej. Oś próbuje normalnie wyrównać ten skok przy pomocy swojej maksymalnej możliwej prędkości. Przy tym jednak na pewnym odcinku na konturze za narożem wynika odchylenie od pożądanego stycznego przyłożenia. Jeżeli z powodów technologicznych nie można tego tolerować, można przy pomocy instrukcji `TLIFT` spowodować, że sterowanie zatrzyma się na narożu i w automatycznie wytworzonym bloku pośrednim obróci oś aktualizowaną do nowego kierunku stycznnej.

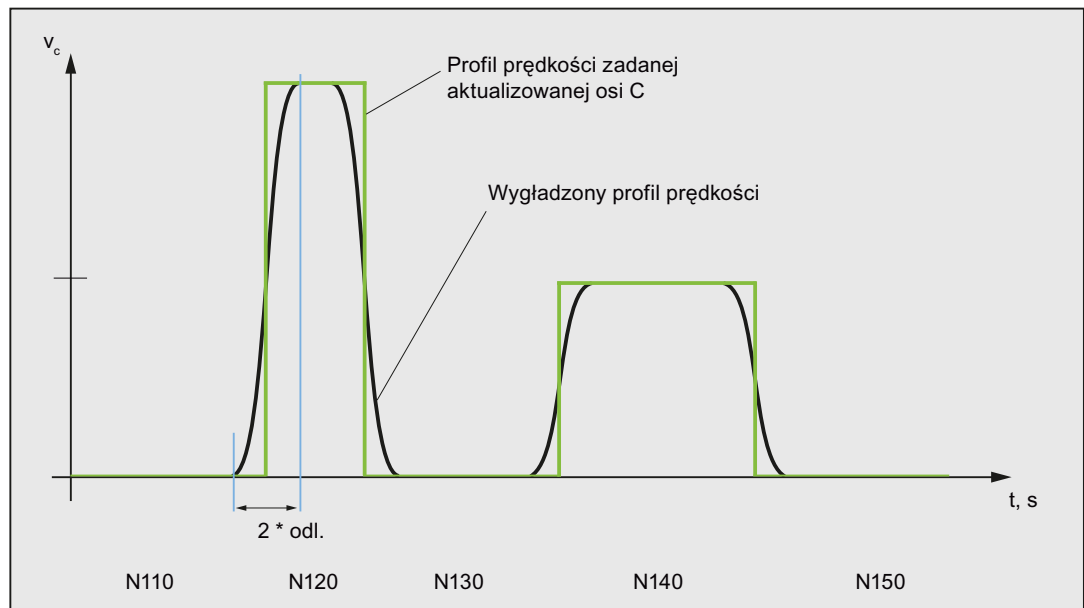
Obrót następuje z zaprogramowaną osią uczestniczącą w tworzeniu konturu, gdy oś aktualizowana wykonywała już ruch jako oś uczestnicząca w tworzeniu konturu. Przez funkcję `TFGREF [<oś>] = 0.001` można tutaj uzyskać prędkość maksymalną osi aktualizowanej.

Jeżeli oś aktualizowana nie wykonywała jeszcze ruchu jako oś uczestnicząca w tworzeniu konturu, wówczas wykonuje ruch jako oś pozycjonowania. Prędkość jest wówczas zależna od prędkości pozycjonowania zapisanej w danej maszynie.

Obrót następuje z maksymalną prędkością osi aktualizowanej.

Możliwość optymalizacji

Jeżeli jest wybrana optymalizacja automatyczna (<Opt> = "P") i dla osi nadążnej są podane parametry droga ścinania (<odległość>) i tolerancja kątowna (<tolerancja kątowna>), wówczas przy aktualizacji stycznej skoki prędkości osi nadążnej w wyniku skoków konturu osi wiodącej są ścinane wzgl. wygładzane. Przy tym oś nadążna jest prowadzona wyprzedzająco (patrz wykres), aby utrzymywać jak najmniejsze odchylenie.



Definiowanie zmiany kąta

Zmiana kąta, od którego jest wstawiany automatyczny blok pośredni, jest definiowana przez następującą daną maszynową:

MD37400 \$MA_EPS_TLIFT_TANG_STEP (kąt stycznej dla rozpoznania naroża)

Wpływ na transformacje

Pozycja aktualizowanej osi obrotowej może być wartością wejściową dla transformacji.

Pozycjonowanie explicite osi nadążnej

Jeżeli jedna z osi aktualizowanych odpowiednio do osi prowadzących jest explicite pozycjonowana, wówczas podanie pozycji działa addytywnie do zaprogramowanego kąta offsetu.

Dopuszczalne są wszystkie zadania drogi (ruchy po torze i ruchy pozycjonowania).

Status sprzężenia

W programie obróbki NC można odpytać status sprzężenia przy pomocy zmiennej systemowej \$AA_COUP_ACT[<os>]:

Wartość	Znaczenie
0	Nie jest aktywne żadne sprzężenie
1,2,3	Aktualizacja styczna aktywna

10.2 Przebieg posuwu (FNORM, FLIN, FCUB, FPO)

Funkcja

W celu bardziej elastycznego zadawania przebiegu posuwu jego programowanie według DIN66025 jest rozszerzane o przebiegi liniowe i sześciennie.

Przebiegi sześciennie mogą być programowane bezpośrednio albo jako spline interpolujące. Przez to dają się - zależnie od zakrzywienia obrabianego przedmiotu - programować ciągle gładkie przebiegi prędkości.

Te przebiegi prędkości umożliwiają wolne od szarpnięć zmiany przyspieszenia, a przez to wykonywanie równomiernych powierzchni obrabianych przedmiotów.

Składnia

F... FNORM
F... FLIN
F... FCUB
F=FPO (... , ... , ...)

Znaczenie

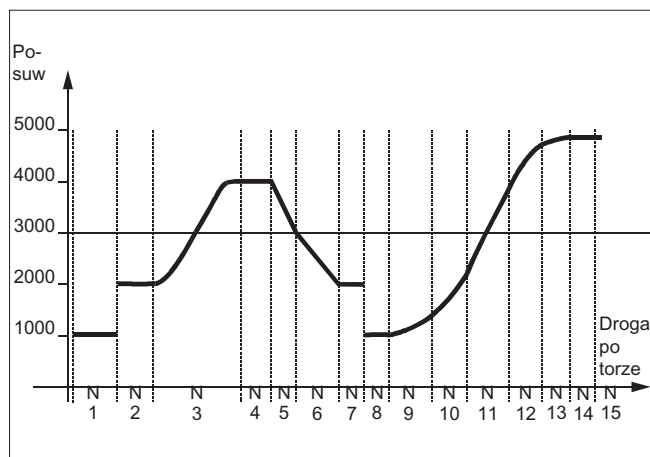
FNORM	Ustawienie podstawowe. Wartość posuwu jest zadawana przez drogę ruchu po torze w bloku i obowiązuje następnie jako wartość modalna.
FLIN	Profil prędkości ruchu po torze liniowy : Wartość posuwu jest uzyskiwana liniowo z aktualnej wartości na początku bloku do końca bloku na drodze ruchu po torze i następnie obowiązuje jako wartość modalna. To zachowanie się można kombinować z G93 i G94.
FCUB	Profil prędkości ruchu po torze sześcienny : Programowane pojedynczymi blokami wartości F - w odniesieniu do punktu końcowego bloku - są łączone przez spline. Spline rozpoczyna się i kończy stycznie do poprzedzającego wzgl. następnego podania posuwu i działa z G93 i G94. Jeżeli w bloku brak jest adresu F, wówczas jest przyjmowana ostatnio zaprogramowana wartość F.
F=FPO...	Profil prędkości ruchu po torze przez wielomian : Adres F określa przebieg posuwu przez wielomian od aktualnej wartości do końca bloku. Wartość końcowa obowiązuje następnie jako wartość modalna.

Optymalizacja posuwu na zakrzywionych torach ruchu

Wielomian posuwu $F=FPO$ i spline posuwu $FCUB$ powinny być zawsze wykonywane ze stałą prędkością skrawania CFC . Przez to daje się wytworzyć wykazujący stałe przyspieszenie profil posuwu zadanego.

Przykład: różne profile posuwu

W tym przykładzie można znaleźć programowanie i przedstawienie graficzne różnych profili posuwu.

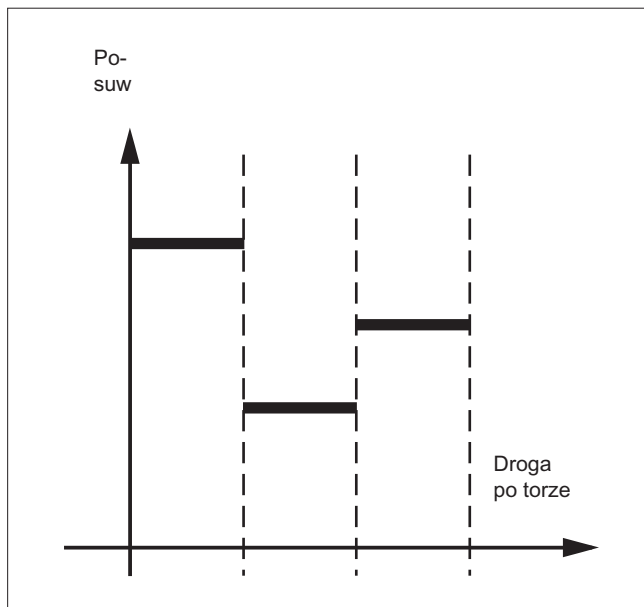


Kod programu	Komentarz
N1 F1000 FNORM G1 X8 G91 G64	; Stały profil posuwu, podanie w wymiarze przyrostowym
N2 F2000 X7	; Skokowa zmiana prędkości zadanej
N3 F=FPO(4000, 6000, -4000)	; Profil posuwu przez wielomian z posuwem 4000 na końcu bloku
N4 X6	; Posuw wielomianowy 4000 obowiązuje jako wartość modalna
N5 F3000 FLIN X5	; Liniowy profil posuwu
N6 F2000 X8	; Liniowy profil posuwu
N7 X5	Posuw liniowy obowiązuje jako wartość modalna
N8 F1000 FNORM X5	; Stały profil posuwu ze skokową zmianą przyspieszenia
N9 F1400 FCUB X8	; Wszystkie dalsze programowane pojedynczymi blokami wartości F są łączone ze spline
N10 F2200 X6	
N11 F3900 X7	
N12 F4600 X7	
N13 F4900 X5	; Wyłączenie profilu spline
N14 FNORM X5	
N15 X20	

FNORM

Adres posuwu F określa posuw ruchu po torze jako wartość stałą według DIN 66025.

Więcej informacji na ten temat można znaleźć w podręczniku programowania "Podstawy".

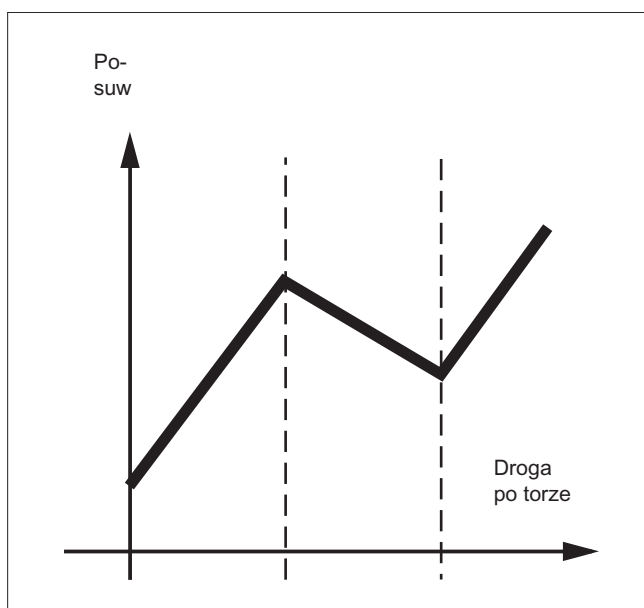


FLIN

Przebieg posuwu jest uzyskiwany od aktualnej wartości posuwu do zaprogramowanej wartości F liniowo do końca bloku.

Przykład:

```
N30 F1400 FLIN X50
```



FCUB

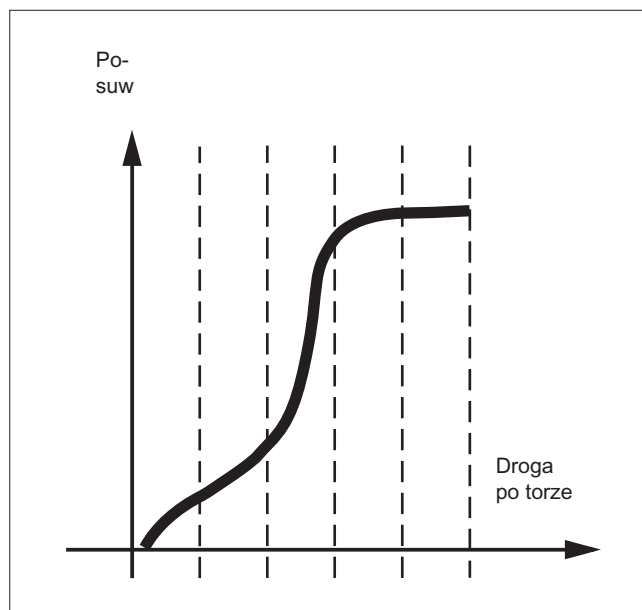
Posuw jest uzyskiwany od aktualnej wartości posuwu do zaprogramowanej wartości F do końca bloku w przebiegu sześciennym. Sterowanie łączy przez spline wszystkie zaprogramowane pojedynczymi blokami wartości posuwu z aktywnym FCUB. Wartości posuwu służą tutaj jako punkty oparcia do obliczenia interpolacji spline.

Przykład:

```
N50 F1400 FCUB X50
```

```
N60 F2000 X47
```

```
N70 F3800 X52
```



F=FPO(.....)

Przebieg posuwu jest programowany bezpośrednio przez wielomian. Podanie współczynników wielomianu następuje analogicznie do interpolacji wielomianowej.

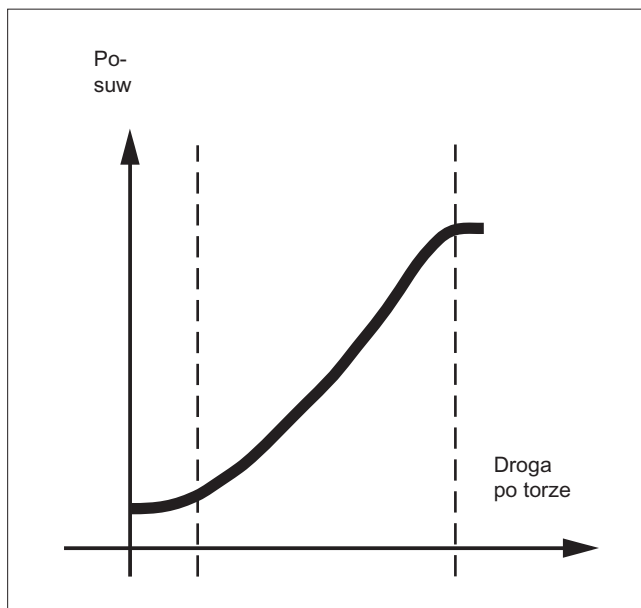
Przykład:

```
F=FPO(endfeed, quadf, cubf)
```

endfeed, quadf i cubf są przedtem zdefiniowanymi zmiennymi.

endfeed:	posuw na końcu bloku
quadf:	kwadratowy współczynnik wielomianowy
cubf:	sześcienny współczynnik wielomianowy

Przy aktywnym `FCUB` spline dołącza na początku i końcu bloku do przebiegu ustalonego przez `FPO`.



Warunki brzegowe

Niezależnie od zaprogramowanego przebiegu posuwu obowiązują funkcje do programowania zachowania się w ruchu po torze.

Programowany przebieg posuwu obowiązuje zasadniczo bezwzględnie - niezależnie od G90 albo G91.

Przebieg posuwu FLIN i FCUB działa z

G93 i G94.

FLIN i FCUB **nie działa** przy

G95, G96/G961 i G97/G971.

Aktywny kompresor COMPON

Przy aktywnym kompresorze COMPON obowiązuje przy połączeniu wielu bloków w jeden segment Spline:

FNORM:

Dla segmentu spline obowiązuje słowo F ostatniego przynależnego bloku.

FLIN:

Dla segmentu spline obowiązuje słowo F ostatniego przynależnego bloku.

Zaprogramowana wartość F obowiązuje do końca segmentu i następnie jest uzyskiwana liniowo.

FCUB:

Utworzony Spline posuwu odbiega maksymalnie o wartość zdefiniowaną w danej maszynowej MD20172 \$MC_COMPRESS_VELO_TOL od zaprogramowanych punktów końcowych.

F=FPO(.....)

Te bloki nie są kompresowane.

10.3 Przyśpieszenie

10.3.1 Tryb przyśpieszenia (BRISK, BRISKA, SOFT, SOFTA, DRIVE, DRIVEA)

Funkcja

Do programowania trybu przyśpieszenia są do dyspozycji następujące polecenia programu obróbki:

- BRISK, BRISKA

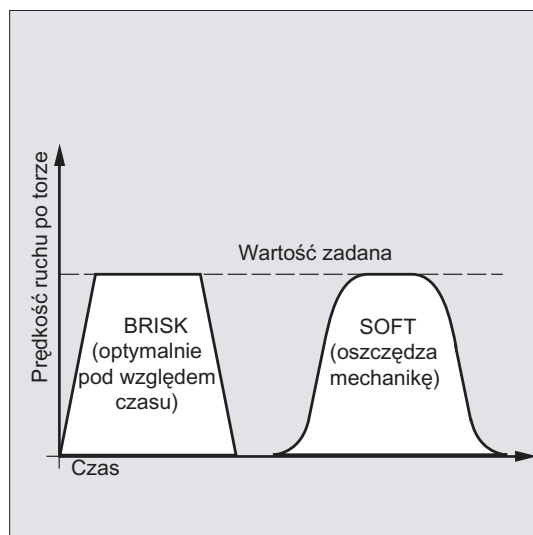
Poszczególne osie wzgl. osie uczestniczące w tworzeniu konturu wykonują ruch z max przyśpieszeniem, aż do osiągnięcia zaprogramowanej prędkości posuwu (**bez ograniczenia przyśpieszenia drugiego stopnia**).

- SOFT, SOFTA

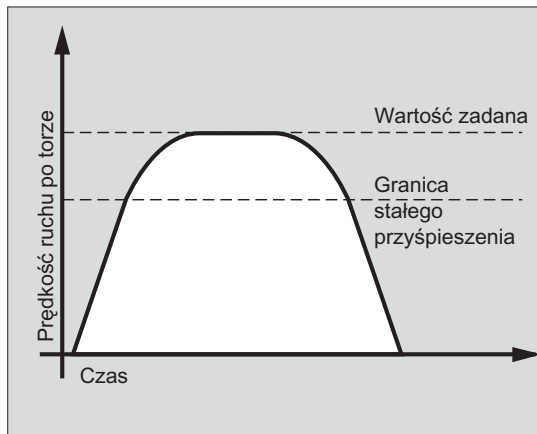
Poszczególne osie wzgl. osie uczestniczące w tworzeniu konturu wykonują ruch ze stałym przyśpieszeniem, aż do osiągnięcia zaprogramowanej prędkości posuwu (**z ograniczeniem przyśpieszenia drugiego stopnia**).

- DRIVE, DRIVEA

Poszczególne osie wzgl. osie uczestniczące w tworzeniu konturu wykonują ruch z maksymalnym przyśpieszeniem, aż do zaprojektowanej granicy prędkości (ustawienie MD!). Potem następuje zmniejszenie przyśpieszenia (ustawienie MD!), aż do osiągnięcia zaprogramowanej prędkości posuwu.



Rysunek 10-1 Przebieg prędkości ruchu po torze przy BRISK i SOFT



Rysunek 10-2 Przebieg prędkości ruchu po torze przy DRIVE.

Składnia

BRISK
BRISKA (<oś1>, <oś2>, ...)
SOFT
SOFTA (<oś1>, <oś2>, ...)
DRIVE
DRIVEA (<oś1>, <oś2>, ...)

Znaczenie

BRISK:	Polecenie do włączenia "przyspieszenia bez ograniczenia przyspieszenia drugiego stopnia" dla osi uczestniczących w tworzeniu konturu.
BRISKA:	Polecenie do włączenia "przyspieszenia bez ograniczenia przyspieszenia drugiego stopnia" dla ruchów w pojedynczych osiach (JOG, JOG/INC, oś pozycjonowania, oś ruchu wahliwego, itd.).
SOFT:	Polecenie do włączenia "przyspieszenia z ograniczeniem przyspieszenia drugiego stopnia" dla osi uczestniczących w tworzeniu konturu.
SOFTA:	Polecenie do włączenia "przyspieszenia z ograniczeniem przyspieszenia drugiego stopnia" dla ruchów w pojedynczych osiach (JOG, JOG/INC, oś pozycjonowania, oś ruchu wahliwego, itd.).
DRIVE:	Polecenie do włączenia zmniejszonego przyspieszenia powyżej zaprojektowanej granicy prędkości (MD35220 \$MA_ACCEL_REDUCTION_SPEED_POINT) dla osi uczestniczących w tworzeniu konturu.
DRIVEA:	Polecenie do włączenia zmniejszonego przyspieszenia powyżej zaprojektowanej granicy prędkości MD35220 \$MA_ACCEL_REDUCTION_SPEED_POINT) dla ruchów w pojedynczych osiach (JOG, JOG/INC, oś pozycjonowania, oś ruchu wahliwego, itd.).
(<oś1>, <oś2>, ...):	Pojedyncze osie, dla których ma obowiązywać wywołany tryb przyspieszenia.

Warunki brzegowe

Zmiana trybu przyśpieszenia podczas obróbki

Gdy w programie obróbki tryb przyśpieszenia zostanie zmieniony podczas obróbki (BRISKA ↔ SOFT), wówczas również w trybie przechodzenia płynnego następuje na przejściu zmiana bloku z zatrzymaniem dokładnym na końcu bloku.

Przykłady

Przykład 1: SOFT i BRISKA

```
Kod programu  
-----  
N10 G1 X... Y... F900 SOFT  
N20 BRISKA (AX5, AX6)  
...
```

Przykład 2: DRIVE i DRIVEA

```
Kod programu  
-----  
N05 DRIVE  
N10 G1 X... Y... F1000  
N20 DRIVEA (AX4, AX6)  
...
```

Literatura

Podręcznik działania Funkcje podstawowe; Przyśpieszenie (B2)

10.3.2 Sterowanie przyśpieszeniem w przypadku osi nadążnych (VELOLIMA, ACCLIMA, JERKLIMA)

Funkcja

W przypadkach sprzężeń osi (aktualizacja styczna, holowanie, sprzężenie wartości wiodącej, przekładnia elektroniczna; patrz Sprzężenia osi (Strona 503)) osie/wrzeciona nadążne wykonują ruch zależnie od jednej lub wielu osi/wrzecion wiodących.

Na ograniczenia dynamiki osi/wrzecion nadążnych można wpływać przy pomocy funkcji VELOLIMA, ACCLIMA i JERKLIMA z programu obróbki lub z akcji synchronicznych, również przy już aktywnym sprzężeniu osi.

Wskazówka

Funkcja JERKLIMA jest dostępna nie dla wszystkich rodzajów sprzężeń.

Literatura:

- Podręcznik działania Funkcje specjalne; Sprzężenia osi (M3)
 - Podręcznik działania Funkcje rozszerzające; wrzeciono synchroniczne (S3)
-

Wskazówka

Dostępność w przypadku SINUMERIK 828D

Funkcje VELOLIMA, ACCLIMA i JERKLIMA mogą w przypadku SINUMERIK 828D być stosowane tylko w połączeniu z funkcją "holowanie!"

Składnia

VELOLIMA (<oś>) =<wartość>
ACCLIMA (<oś>) =<wartość>
JERKLIMA (<oś>) =<wartość>

Znaczenie

VELOLIMA: Polecenie do korekcji sparametryzowanej **prędkości** maksymalnej
ACCLIMA: Polecenie do korekcji sparametryzowanego **przyspieszenia** maksymalnego
JERKLIMA: Polecenie do korekcji sparametryzowanego maksymalnego **przyspieszenia drugiego stopnia**
<oś>: Oś nadążna, której ograniczenia dynamiki mają zostać skorygowane
<wartość>: Procentowa wartość korekcji

Przykłady

Przykład 1: Korekcja ograniczeń dynamiki dla osi nadążnej (AX4)

Kod programu	Komentarz
...	
VELOLIMA [AX4] =75	; Korekcja ograniczenia do 75% zapisanej w danej maszynowej prędkości maksymalnej w osi.
ACCLIMA [AX4] =50	; Korekcja ograniczenia do 50% zapisanego w danej maszynowej osiowego przyspieszenia maksymalnego.
JERKLIMA [AX4] =50	; Korekcja ograniczenia do 50% zapisanego w danej maszynowej osiowego maksymalnego przyspieszenia drugiego stopnia.
...	

Przykład 2: Przekładnia elektroniczna

Oś 4 jest przez sprzężenie "przekładnia elektroniczna" sprzężona z osią X. Zdolność przyśpieszenia osi nadszajnej jest ograniczona do 70% przyśpieszenia maksymalnego. Maksymalna dopuszczalna prędkość jest ograniczona do 50% prędkości maksymalnej. Po dokonanych włączeniu sprzężenia maksymalna dopuszczalna prędkość jest ponownie ustawiana na 100%.

Kod programu	Komentarz
...	
N120 ACCLIMA [AX4]=70	; Zmniejszone przyśpieszenie maksymalne.
N130 VELOLIMA [AX4]=50	; Zmniejszona prędkość maksymalna.
...	
N150 EGON (AX4, "FINE", X, 1, 2)	; Włączenie sprzężenia przekładni elektronicznej.
...	
N200 VELOLIMA [AX4]=100	; Pełna maksymalna prędkość.
...	

Przykład 3: Wpływ na sprzężenie wartości wiodącej przez statyczną akcję synchroniczną

Oś 4 jest sprzężana z osią X przez sprzężenie z wartością wiodącą. Przyśpieszenie jest poprzez statyczną akcję synchroniczną 2 od pozycji 100 ograniczone do 80 %.

Kod programu	Komentarz
...	
N120 IDS=2 WHENEVER \$AA_IM[AX4] > 100 DO ACCLIMA[AX4]=80	; Akcja synchroniczna
N130 LEADON (AX4, X, 2)	; Sprzężenie wartości wiodącej wł.
...	

10.3.3 Uaktywnienie specyficznych dla technologii wartości dynamiki (DYNNORM, DYNPOS, DYNROUGH, DYNSEMIFIN, DYNFINISH)

Funkcja

Przy pomocy grupy G "technologia" można dla 5 różnych technologicznych kroków obróbki uaktywnić pasującą dynamikę.

Wartości dynamiki i polecenia G dają się projektować, a przez to są zależne od ustawień danych maszynowych (→ producenta maszyny!).

Literatura:

Podręcznik działania Funkcje podstawowe; Przechodzenie płynne, Zatrzymanie dokładne, LookAhead (B1)

Składnia

Uaktywnienie wartości dynamiki:

DYNNORM
DYNPOS
DYNROUGH
DYNSEMIFIN
DYNFINISH

Wskazówka

Wartości dynamiki działają już w tym bloku, w którym zostało zaprogramowane przynależne polecenie G. Nie następuje zatrzymanie obróbki.

Zapis albo odczyt określonego elementu tablicy:

R<m>=\$MA... [n, X]
\$MA... [n, X] =<wartość>

Znaczenie

DYNNORM:	Polecenie G do uaktywnienia normalnej dynamiki
DYNPOS:	Polecenie G do uaktywnienia dynamiki dla pozycjonowania, gwintowania otworu
DYNROUGH:	Polecenie G do uaktywnienia dynamiki dla obróbki zgrubnej
DYNSEMIFIN:	Polecenie G do uaktywnienia dynamiki dla obróbki wykańczającej
DYNFINISH:	Polecenie G do uaktywnienia dynamiki dla obróbki wykańczającej
R<m>:	Parametr obliczeniowy o numerze <m>
\$MA... [n, X]:	Dana maszynowa z elementem tablicy określającym dynamikę
<n>:	Indeks tablicy Zakres wartości: 0 ... 4 0 Normalna dynamika (DYNNORM) 1 Dynamika dla pozycjonowania (DYNPOS) 2 Dynamika dla obróbki zgrubnej (DYNROUGH) 3 Dynamika dla obróbki wykańczającej (DYNSEMIFIN) 4 Dynamika dla obróbki wykańczającej (DYNFINISH)
<X> :	Adres osi
<wartość>:	Wartość dynamiki

Przykłady

Przykład 1: Uaktywnienie wartości dynamiki

Kod programu	Komentarz
DYNNORM G1 X10	; Położenie podstawowe
DYNPOS G1 X10 Y20 Z30 F...	; Tryb pozycjonowania, gwintowanie otworu
DYNROUGH G1 X10 Y20 Z30 F10000	; Obróbka zgrubna
DYNSEMIFIN G1 X10 Y20 Z30 F2000	; Obróbka wykańczająca
DYNFINISH G1 X10 Y20 Z30 F1000	; Obróbka wykańczająca

Przykład 2: Odczyt lub zapis określonego elementu tablicy

Przyspieszenie maksymalne dla obróbki zgrubnej, oś X.

Kod programu	Komentarz
R1=\$MA_MAX_AX_ACCEL[2,X]	; Odczyt
\$MA_MAX_AX_ACCEL[2,X]=5	; Zapis

10.4 Ruch ze sterowaniem wyprzedzającym (FFWON, FFWOF)

Funkcja

Dzięki sterowaniu wyprzedzającemu zależna od prędkości droga wybiegu ruchu po torze jest zredukowana do zera. Ruch ze sterowaniem wyprzedzającym umożliwia większą dokładność ruchu po torze, a przez to lepsze wyniki produkcji.

Składnia

FFWON

FFWOF

Znaczenie

FFWON: Polecenie do **włączenia** sterowania wyprzedzającego

FFWOF: Polecenie do **wyłączenia** sterowania wyprzedzającego

Wskazówka

Poprzez dane maszynowe ustala się rodzaj sterowania wyprzedzającego i które osie uczestniczące w tworzeniu konturu mają wykonywać ruch z takim sterowaniem.

Standard: Sterowanie wyprzedzające zależne od prędkości

Opcja: Sterowanie wyprzedzające zależne od przyspieszenia

Przykład

Kod programu
N10 FFWON
N20 G1 X... Y... F900 SOFT

10.5 Programowalna dokładność konturu (CPRECON, CPRECOF)

Funkcja

Funkcja "Programowalna dokładność konturu" zmniejsza błąd toru na konturach zakrzywionych, dzięki automatycznemu dopasowaniu prędkości.

Będąc do utrzymania dokładność konturu zadaje się w zależności od zaprojektowania maszyny (MD20470 \$MC_MC_CPREC_WITH_FFW; patrz dane producenta maszyny) albo poprzez daną ustawczą \$SC_CONTPREC lub poprzez zaprogramowaną tolerancję konturu CTOL. Im mniejsza jest wartość i im mniejszy współczynnik K_V osi geometrycznych, tym bardziej jest zmniejszany posuw po torze na konturach zakrzywionych.

Funkcja "Programowalna dokładność konturu" jest włączana i wyłączana poprzez instrukcje CPRECON i CPRECOF w programie NC.

Składnia

CPRECON
...
CPRECOF

Znaczenie

CPRECON:	Wywołanie funkcji G: włączyć "Programowalną dokładność konturu" Działanie: modalnie
CPRECOF:	Wywołanie funkcji G: wyłączyć "Programowalną dokładność konturu" Działanie: modalnie

CPRECON i CPRECOF tworzą razem grupę funkcji G 39 (programowalna dokładność konturu).

Wskazówka

Poprzez daną ustawczą \$SC_MINFEED (minimalny posuw po torze przy CPRECON) użytkownik może zadać prędkość minimalną posuwu po torze.

Posuw nie jest ograniczany poniżej tej wartości, chyba że została zaprogramowana niższa wartość F lub dynamiczne ograniczenia osi wymuszają mniejszą prędkość.

Wskazówka

Funkcja "Programowalna dokładność konturu" dotyczy tylko osi geometrycznych toru. Na prędkość osi pozycjonowania nie ma ona wpływu.

Przykład

Kod programu	Komentarz
N10 G0 X0 Y0	
N20 CPRECON	; Włączenie "Programowalnej dokładności konturu".
N30 G1 G64 X100 F10000	; Obróbka z 10 m/min w trybie przechodzenia płynnego.
N40 G3 Y20 J10	; Automatyczne ograniczenie posuwu w bloku ruchu kołowego.
N50 G1 X0	; Posuw ponownie bez ograniczenia (10 m/min).
...	
N100 CPRECOF	; Wyłączenie "Programowalnej dokładności konturu".
N110 G0 ...	

Literatura

Odnosnie programowania CTOL patrz "Programowana tolerancja konturu/orientacji (CTOL, OTOL, ATOL) (Strona 496)"

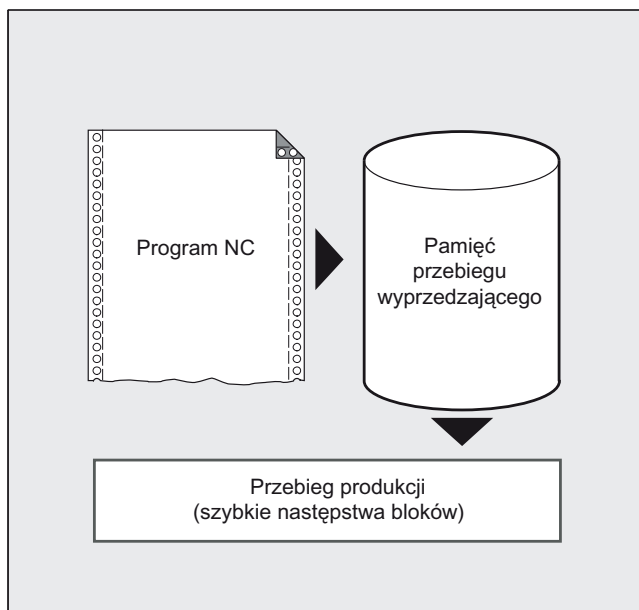
Dla uzyskania bardziej szczegółowych informacji dot. funkcji "Programowalna dokładność konturu" patrz:

Podręcznik działania, Funkcje specjalne; Nadzór tunelu konturowego (K6), Rozdział: "Programowalna dokładność konturu"

10.6 Przebieg programu z pamięcią przebiegu wyprzedzającego (STOPFIFO, STARTFIFO, FIFOCTRL, STOPRE)

Funkcja

W zależności od stopnia rozbudowy sterowanie dysponuje określoną wielkością tzw. pamięci przebiegu wyprzedzającego, która zapisuje przygotowane bloki przed ich wykonaniem i w trakcie obróbki wyprowadza jako szybkie sekwencje bloków. Dają się przez to wykonywać krótkie drogi z dużymi prędkościami. O ile pozostały czas sterowania to dopuszcza, pamięć przebiegu wyprzedzającego jest z zasady napełniana.



Oznakowanie segmentu obróbki

Segment obróbki, który ma zostać poddany pośredniemu zapisaniu w pamięci przebiegu wyprzedzającego, jest w programie obróbki oznakowywany na początku przez `STOPFIFO` i na końcu przez `STARTFIFO`. Wykonywanie przygotowanych i poddanych pośredniemu zapisaniu bloków rozpoczyna się dopiero po poleceniu `STARTFIFO` albo gdy pamięć przebiegu wyprzedzającego jest zapełniona.

Automatyczne sterowanie pamięcią przebiegu wyprzedzającego

Automatyczne sterowanie pamięcią przebiegu wyprzedzającego jest wywoływane poleceniem `FIFOCTRL`. `FIFOCTRL` działa najpierw dokładnie tak, jak `STOPFIFO`. Przy każdym programowaniu następuje czekanie, aż pamięć przebiegu wyprzedzającego zostanie wypełniona, następnie rozpoczyna się wykonywanie. Różne jest natomiast zachowanie się przy opróżnianiu pamięci przebiegu wyprzedzającego: za pomocą `FIFOCTRL` prędkość po torze jest coraz bardziej zmniejszana od poziomu napełnienia 2/3, żeby zapobiec kompletnemu opróżnieniu pamięci i hamowaniu aż do stanu zatrzymania.

Zatrzymanie przebiegu wyprzedzającego

Przygotowywanie i pośrednie zapisywanie bloków jest zatrzymywane, gdy w bloku jest zaprogramowane polecenie `STOPRE`. Kolejny blok jest wykonywany dopiero wtedy, gdy wszystkie przedtem przygotowane i zapisane bloki są całkowicie wykonane. Poprzedni blok jest zatrzymywany w zatrzymaniu dokładnym (jak G9).

UWAGA**Anulowanie programu**

Przy włączonej korekcji narzędzia i w przypadku interpolacji Spline nie należy programować `STOPRE`, ponieważ spowoduje to przerwanie przynależnych do siebie ciągów bloków.

Składnia

Tabela 10- 1 Oznakowanie segmentu obróbki:

```

| STOPFIFO
| ...
| STARTFIFO

```

Tabela 10- 2 Automatyczne sterowanie pamięcią przebiegu wyprzedzającego:

```

| ...
| FIFOCTRL
| ...

```

Tabela 10- 3 Zatrzymanie przebiegu wyprzedzającego:

```

| ...
| STOPRE
| ...

```

Wskazówka

Polecenia `STOPFIFO`, `STARTFIFO`, `FIFOCTRL` i `STOPRE` muszą być programowane w oddzielnym bloku.

Znaczenie

STOPFIFO:	STOPFIFO oznakowuje początek segmentu obróbki, który ma zostać poddany pośredniemu zapisaniu w pamięci przebiegu wyprzedzającego. Przy pomocy STOPFIFO wykonywanie jest zatrzymywane i pamięć pośrednia jest napelniana aż: <ul style="list-style-type: none">• nastąpi rozpoznanie STARTFIFO albo STOPRElub• pamięć przebiegu wyprzedzającego zostanie wypełnionalub• nastąpi dojście do końca programu.
STARTFIFO:	Przy pomocy STARTFIFO następuje szybkie wykonywanie segmentu obróbki, równoległe z tym następuje napelnianie pamięci przebiegu wyprzedzającego.
FIFOCTRL:	Włączenie automatycznego sterowania pamięcią przebiegu wyprzedzającego
STOPRE:	Zatrzymanie przebiegu wyprzedzającego

Wskazówka

Napelnianie pamięci przebiegu wyprzedzającego nie jest wykonywane wzgl. jest przerywane, gdy segment obróbki zawiera polecenia, które wymuszają pracę nie buforowaną (bazowanie do punktu odniesienia, funkcje pomiarowe, ...).

Wskazówka

Przy dostępie do danych stanu maszyny (\$SA...) sterowanie wytwarza wewnętrzne zatrzymanie przebiegu wyprzedzającego.

Przykład: Zatrzymanie przebiegu wyprzedzającego

Kod programu	Komentarz
...	
N30 MEAW=1 G1 F1000 X100 Y100 Z50	; Blok pomiarowy z sonda pomiarową pierwszego wejścia pomiarowego i interpolacją prostoliniową.
N40 STOPRE	; Zatrzymanie przebiegu wyprzedzającego.
...	

10.7 Warunkowo przerywalne segmenty programu (DELAYFSTON, DELAYFSTOF)

Funkcja

Warunkowo przerywalne segmenty programu obróbki są nazywane obszarami Stop-Delay. W ramach określonych segmentów programu nie ma następować **zatrzymywanie**, a również **posuw** nie ma być zmieniany. W istocie krótkie segmenty programu, które np. służą do wykonywania gwintu, powinny być chronione przed wszelkimi zdarzeniami zatrzymania. Ewentualne zatrzymanie działa dopiero wtedy, gdy fragment programu został wykonany do końca.

Składnia

```
DELAYFSTON
...
DELAYFSTOF
```

Wskazówka

Polecenia DELAYFSTON i DELAYFSTOF znajdują się oddzielnie w jednym wierszu programu obróbki.

Znaczenie

DELAYFSTON:	Zdefiniowanie początku obszaru, w którym "łagodne" zatrzymania ulegają zwłóce, aż zostanie osiągnięty koniec obszaru Stop-Delay.
DELAYFSTOF:	Zdefiniowanie końca obszaru Stop-Delay

Wskazówka

DELAYFSTON i DELAYFSTOF są dopuszczalne tylko w programach obróbki, ale nie w akcjach synchronicznych.

Wskazówka

W przypadku MD11550 \$MN_STOP_MODE_MASK Bit 0 = 0 (domyślna) obszar Stop-Delay jest implicite definiowany, gdy G331/G332 jest aktywne i jest zaprogramowany ruch po torze lub G4.

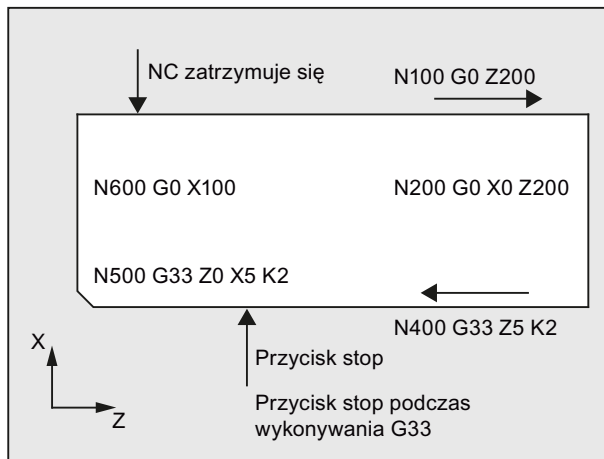
Przykłady

Przykład 1: Zagnieżdżanie obszarów Stop-Delay na dwóch poziomach programu

Kod programu	Komentarz
N10010 DELAYFSTON()	; bloki z N10xxx płaszczyzna programowa 1.
N10020 R1 = R1 + 1	
N10030 G4 F1	; Obszar Stop-Delay rozpoczyna się.
...	
N10040 Podprogram2	
...	
...	; Interpretacja podprogramu 2.
N20010 DELAYFSTON()	; Nie działa, powtórne rozpoczęcie, 2. poziom.
...	
N20020 DELAYFSTOF()	; Nie działa, koniec na innym poziomie.
N20030 RET	
N10050 DELAYFSTOF()	; Koniec obszaru Stop-Delay na tym samym poziomie.
...	
N10060 R2 = R2 + 2	
N10070 G4 F1	; Obszar Stop-Delay kończy się. Zatrzymania działają od teraz bezpośrednio.

Przykład 2

W pętli jest powtarzany następujący blok programu:



Na rysunku widać, że użytkownik w zakresie Stop-Delay naciska "Stop", a NC rozpoczyna proces hamowania poza tym zakresem, tzn. w bloku N100. W wyniku tego NC zatrzymuje się w przednim obszarze N100.

Kod programu

```

...
N99 MY_LOOP:
N100 G0 Z200
N200 G0 X0 Z200
N300 DELAYFSTON()
N400 G33 Z5 K2 M3 S1000
N500 G33 Z0 X5 K3
N600 G0 X100
N700 DELAYFSTOF()
N800 GOTOB MY_LOOP

```

Dalsze informacje**Zdarzenia stopu**

W obszarze Stop-Delay zmiana posuwu i blokada posuwu są ignorowane. Działają one dopiero za obszarem Stop-Delay.

Są rozróżniane zdarzenia stopu:

- "Łagodne" z darzenia stopu Reakcja: delayed
- "Twarde" zdarzenia stopu Reakcja: immediate

Wybór kilku zdarzeń stopu, które zatrzymują przynajmniej na krótki czas:

Nazwa zdarzenia	Reakcja	Parametry przerwania
RESET	immediate	NST: DB21, ... DBX7.7 i DB11 DBX20.7
PROG_END	Alarm 16954	NC-Prog: M30
INTERRUPT	delayed	NST: FC-9 i ASUP DB10 DBB1
SINGLEBLOCKSTOP	delayed	Wykonywanie pojedynczymi blokami w obszarze Stop-Delay włączone: NC zatrzymuje się na końcu 1. bloku poza obszarem Stop-Delay. Wykonywanie pojedynczymi blokami wybrane jeszcze przed obszarem Stop-Delay: NST: "zatrzymanie NC na granicy bloków" DB21, ... DBX7.2
STOPPROG	delayed	NST: DB21,... DBX7.3 i DB11 DBX20.5
PROG_STOP	Alarm 16954	NC-Prog: M0 i M1
WAITM	Alarm 16954	NC-Prog: WAITM
WAITE	Alarm 16954	NC-Prog: WAITE
STOP_ALARM	immediat	Alarm: projektowanie alarmu STOPBYALARM
RETREAT_MOVE_THREAD	Alarm 16954	NC-Prog: Alarm 16954 przy LFON (Stopp & Fastlift w G33 niemożliwe)

Nazwa zdarzenia	Reakcja	Parametry przerwania
WAITMC	Alarm 16954	NC-Prog: WAITMC
NEWCONF_PREP_STOP	Alarm 16954	NC-Prog: NEWCONF
SYSTEM_SHUTDOWN	immediate	System-Shutdown przy 840Di sl
ESR	delayed	Rozszerzone zatrzymanie i wycofanie
EXT_ZERO_POINT	delayed	Zewnętrzne przesunięcie punktu zerowego
STOPRUN	Alarm 16955	BTSS: PI "_N_FINDST" STOPRUN

Objaśnienie reakcji:

immediate ("twarde" zdarzenie stopu)	Zatrzymuje natychmiast również w obszarze Stop-Delay
delayed ("łagodne" zdarzenie stopu)	Zatrzymanie (również krótkotrwałe) następuje dopiero za obszarem Stop-Delay.
Alarm 16954	Program jest przerywany, ponieważ w obszarze Stop-Delay są zastosowane niedozwolone polecenia programowe.
Alarm 16955	Program jest kontynuowany, w obszarze Stop-Delay nastąpiła niedozwolona akcja.
Alarm 16957	Obszar programu (obszar Stop-Delay), który jest określony przez DELAYFSTON i DELAYFSTOF, nie mógł zostać uaktywniony. Przez to każde zatrzymanie w obszarze działa natychmiast i nie podlega zwłoce.

Podsumowanie dalszych reakcji na zdarzenia zatrzymania patrz:

Literatura:

Podręcznik działania Funkcje podstawowe; BAG, Kanał, Praca programowa, Zachowanie się przy zresetowaniu (K1)

Zalety obszaru Stop-Delay

Segment programu jest wykonywany bez załamania prędkości.

Jeżeli użytkownik, po nastąpieniu zatrzymania, przerwie program przez zresetowanie, wówczas przerwany blok programu znajduje się za chronionym obszarem. Ten blok programu nadaje się wówczas jako cel szukania dla późniejszego szukania.

Jak długo obszar Stop-Delay jest wykonywany, następujące osie przebiegu głównego nie są zatrzymywane:

- osie rozkazowe i
- osie pozycjonowania, które wykonują ruch z POSA

Polecenie programu obróbki^{G4} jest dopuszczalne w obszarze Stop-Delay, natomiast inne polecenia programu obróbki, które prowadzą do przejściowego zatrzymania (np. WAITM) są niedopuszczalne.

G₄ uaktywnia wzgl. podtrzymuje aktywność obszaru Stop-Delay, jak ruch po torze.

Przykład: ingerencje w posuw

Gdy korektor zostanie przed obszarem Stop-Delay obniżony do 6%, wówczas działa w tym obszarze.

Jeżeli korektor zostanie w obszarze Stop-Delay obniżony ze 100% do 6%, wówczas obszar ten jest realizowany do końca ze 100%, a następnie ruch odbywa się z 6%.

Blokada posuwu nie działa w obszarze Stop-Delay, zatrzymanie następuje dopiero po opuszczeniu tego obszaru.

Nakładanie się / kaskadowanie

Jeżeli dwa obszary Stop-Delay nakładają się na siebie, jeden z poleceń językowych, a drugi z danej maszyny MD11550 \$MN_STOP_MODE_MASK, wówczas jest tworzony łączny możliwy obszar Stop-Delay.

Następujące punkty regulują współpracę poleceń językowych DELAYFSTON i DELAYFSTOF z kaskadowaniem i końcem podprogramu:

- Z końcem podprogramu, w którym wywołano DELAYFSTON następuje implicite uaktywnienie DELAYFSTOF.
- DELAYFSTON w obszarze Stop-Delay pozostaje bez działania.
- Jeżeli podprogram 1 w obszarze Stop-Delay wywoła podprogram 2, wówczas podprogram 2 mieści się kompletnie w obszarze Stop-Delay. W szczególności nie działa DELAYFSTOF w podprogramie 2.

Wskazówka

REPOSA jest końcem podprogramu i DELAYFSTON jest cofane w każdym przypadku.

Jeżeli "twarde" zdarzenie zatrzymania natrafi na "obszar Stop-Delay", wówczas wybór całego obszaru jest cofany! Oznacza to, że gdy w tym segmencie programu nastąpi kolejne dowolne zatrzymanie, jest ono natychmiast realizowane. Dopiero nowe zaprogramowanie (ponowny DELAYFSTON) pozwala na rozpoczęcie nowego obszaru Stop-Delay.

Jeżeli przycisk Stop zostanie naciśnięty przed obszarem Stop-Delay, a NCK musi w celu wyhamowania wejść w ten obszar, wówczas NCK zatrzymuje się w obszarze Stop-Delay i wybór tego obszaru pozostaje cofnięty!

Gdy wejście w obszar Stop-Delay nastąpi z override 0%, wówczas obszar Stop-Delay **nie** jest akceptowany!

Dotyczy to wszystkich "łagodnych" zdarzeń zatrzymania.

Przy pomocy STOPALL można hamować w obszarze Stop-Delay. Jednak w wyniku STOPALL stają się aktywne wszystkie inne zdarzenia zatrzymania, które dotychczas podlegały zwłoce.

Zmienne systemowe

Obszar Stop-Delay może zostać rozpoznany przy pomocy \$P_DELAYFST w programie obróbki. Gdy bit 0 zmiennej systemowej jest ustawiony na 1, wówczas wykonywanie programu obróbki znajduje się w tym momencie w obszarze Stop-Delay.

Obszar Stop-Delay może zostać rozpoznany przy pomocy \$AC_DELAYFST w akcjach synchronicznych. Gdy bit 0 zmiennej systemowej jest ustawiony na 1, wówczas wykonywanie programu obróbki znajduje się w tym momencie w obszarze Stop-Delay.

Kompatybilność

Ustawienie domyślne danej maszynowej MD11550 \$MN_STOP_MODE_MASK Bit 0 = 0 powoduje implicite obszar Stop-Delay podczas grupy G-Code G331/G332 i gdy jest zaprogramowany ruch po torze G4.

Bit 0 = 1 umożliwia zatrzymanie podczas grupy G-Code G331/G332 i gdy jest zaprogramowany ruch po torze wzgl. G4. Do definicji obszaru Stop-Delay muszą zostać użyte polecenia DELAYFSTON/DELAYFSTOF.

10.8 Zablokowanie miejsca w programie dla SERUPRO (IPTRLOCK, IPTRUNLOCK)

Funkcja

Dla określonych skomplikowanych sytuacji mechanicznych na maszynie jest konieczne uniemożliwienie szukania bloku SERUPRO.

Przy pomocy programowanego wskaźnika przerwania jest możliwość ingerencji, by przy "szukaniu miejsca przerwania" rozpocząć obróbkę przed miejscem nadającym się do przeszukiwania.

Jest możliwość również definiowania obszarów niezdatnych do przeszukiwania w obszarach programu obróbki, w których NCK jeszcze nie może ponownie rozpocząć obróbki. Z przerwaniem programu NCK zapamiętuje ostatnio wykonany blok, którego można szukać poprzez interfejs graficzny HMI.

Składnia

IPTRLOCK
IPTRUNLOCK

Polecenia znajdują się w oddzielnych wierszach programu obróbki i umożliwiają programowany wskaźnik przerwania

Znaczenie

IPTRLOCK Początek segmentu programu niezdatnego do przeszukiwania
IPTRUNLOCK Koniec segmentu programu niezdatnego do przeszukiwania

Obydwa polecenia są dopuszczalne tylko w programach obróbki ale **nie** w akcjach synchronicznych.

Przykład

Kaskadowanie niezdatnych do przeszukiwania fragmentów programu na dwóch poziomach programowych z implicite IPTRUNLOCK. Implicite IPTRUNLOCK w podprogramie 1 kończy obszar niezdatny do przeszukiwania.

Kod programu	Komentarz
N10010 IPTRLOCK()	
N10020 R1 = R1 + 1	
N10030 G4 F1	; Blok zatrzymania, który rozpoczyna niezdatny do przeszukiwania segment programu.
...	
N10040 Podprogram2	
...	; Interpretacja podprogramu 2.
N20010 IPTRLOCK ()	; Nie działa, powtórne rozpoczęcie.
...	
N20020 IPTRUNLOCK ()	; Nie działa, koniec na innym poziomie.
N20030 RET	
...	
N10060 R2 = R2 + 2	
N10070 RET	; Koniec segmentu programu niezdatnego do przeszukiwania.
N100 G4 F2	; Program główny jest kontynuowany.

Przerwanie na 100 daje wówczas znów wskaźnik przerwania.

Ujęcie i szukanie obszarów niezdatnych do przeszukiwania

Fragmenty programu niezdatne do przeszukiwania programu są zaznaczane przy pomocy poleceń językowych IPTRLOCK i IPTRUNLOCK.

Polecenie IPTRLOCK zamraża wskaźnik przerwania na pojedynczym bloku wykonywalnym w przebiegu głównym (SBL1). Ten blok jest dalej nazywany blokiem zatrzymania. Gdy po IPTRLOCK nastąpi przerwanie programu, wówczas na interfejsie graficznym HMI można szukać tego tzw. bloku zatrzymania.

Ponowne ustawienie na aktualnym bloku

Wskaźnik przerwania jest przy pomocy IPTRUNLOCK dla następnego segmentu programu ustawiany na aktualnym bloku w punkcie przerwania.

Po znalezieniu celu szukania można z tym samym blokiem zatrzymania powtórzyć nowy cel szukania.

Edytowany przez użytkownika wskaźnik przerwania, musi zostać ponownie usunięty poprzez HMI.

Zasady przy kaskadowaniu

Następujące punkty regulują współpracę poleceń językowych IPTRLOCK i IPTRUNLOCK z kaskadowaniem i końcem podprogramu:

1. Z końcem podprogramu, w którym wywołano IPTRLOCK następuje implicite uaktywnienie IPTRUNLOCK .
2. IPTRLOCK w obszarze niezdatnym do przeszukiwania pozostaje bez działania.
3. Jeżeli podprogram1 w obszarze niezdatnym do przeszukiwania wywoła podprogram2, wówczas cały podprogram2 pozostaje niezdatny do przeszukiwania. W szczególności nie działa IPTRUNLOCK w podprogramie2.

Dalsze informacje na ten temat patrz /FB1/ Podręcznik działania Funkcje podstawowe; BAG, Kanał, praca programowa (K1).

Zmienna systemowa

Obszar niezdatny do przeszukiwania można rozpoznać przy pomocy \$S_IPTRLOCK w programie obróbki.

Automatyczny wskaźnik przerwania

Funkcja automatycznego wskaźnika przerwania automatycznie ustala przedtem ustalony rodzaj sprzężenia jako niezdatny do przeszukiwania. Przy pomocy danej maszynowej jest dla

- przekładni elektronicznej przy EGON
- osiowego sprzężenia wartości wiodącej przy LEADON

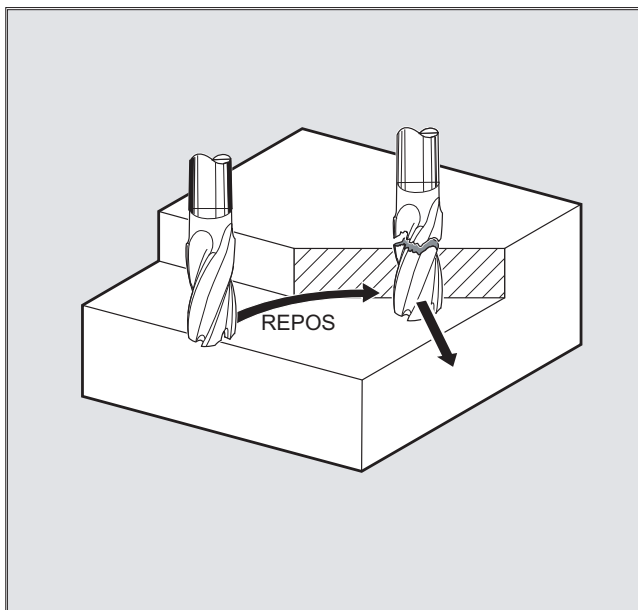
uaktywniany automatyczny wskaźnik przerwania. Jeżeli zaprogramowane i uaktywniane poprzez daną maszynową automatyczne wskaźniki przerwania zachodzą na siebie, wówczas jest tworzony największy możliwy obszar niezdatny do przeszukiwania.

10.9 Dosunięcie przywracające do konturu (REPOSA, REPOSL, REPOSQ, REPOSQA, REPOSH, REPOSHA, DISR, DISPR, RMIBL, RMBBL, RMEBL, RMNBL)

Funkcja

Gdy podczas obróbki nastąpi przerwanie bieżącego programu i odsunięcie narzędzia - na przykład z powodu jego pęknięcia albo w celu przeprowadzenia pomiaru - można dokonać ponownego sterowanego programem dosunięcia narzędzia do konturu w wybranym punkcie.

Polecenie REPOS działa jak powrót z podprogramu (np. jako M17). Kolejne bloki w procedurze przerwania nie są już wykonywane.



Na temat przerwania przebiegu programu patrz też "Procedura przerwania (ASUP) (Strona 123)".

Składnia

```

REPOSA RMIBL DISPR=...
REPOSA RMBBL
REPOSA RMEBL
REPOSA RMNBL
REPOSL RMIBL DISPR=...
REPOSL RMBBL
REPOSL RMEBL
REPOSL RMNBL
REPOSQ RMIBL DISPR=... DISR=...
REPOSQ RMBBL DISR=...
REPOSQ RMEBL DISR=...
REPOSQA DISR=...
REPOSH RMIBL DISPR=... DISR=...
REPOSH RMBBL DISR=...
REPOSH RMEBL DISR=...
REPOSHA DISR=...

```

Znaczenie

Droga dosunięcia

REPOSA:	Dosuw po prostej we wszystkich osiach
REPOSL:	Dosuw po prostej
REPOSQ DISR=... :	Dosuw po ćwierćokręgu o promieniu DISR
REPOSQA DISR=... :	Dosuw we wszystkich osiach po ćwierćokręgu o promieniu DISR
REPOSH DISR=... :	Dosuw po półokręgu o średnicy DISR
REPOSHA DISR=... :	Dosuw we wszystkich osiach po półokręgu o promieniu DISR

Punkt dosunięcia przywracającego

RMIBL:	Dosunięcie do punktu przerwania
RMIBL DISPR=...:	Punkt wejścia w odstępie DISPR w mm/calach przed punktem przerwania
RMBBL:	Dosunięcie do punktu początkowego bloku
RMEBL:	Dosunięcie do punktu końcowego bloku
RMEBL DISPR=... :	Dosunięcie do punktu końcowego bloku w odstępie DISPR przed punktem końcowym
RMNBL:	Dosunięcie do najbliższego położonego punktu toru
A0 B0 C0 :	Osie, w których ma nastąpić dosunięcie

Wskazówka

Kompatybilność

Aby utrzymać kompatybilność ze starszymi wersjami oprogramowania, można zainstalować tryb dosunięcia REPOS, a także modalne rozkazy G RMI, RMB, RME i RMN. Należy przy tym pamiętać, aby w poleceniu PROC zaprogramować ASUP z atrybutem SAVE. W innym przypadku modalny tryb dosunięcia REPOS miałby wpływ również na następne procesy repozycjonowania w sposób niezamierzony, jeśli odbiegałby on od ustawienia domyślnego RMI.

Przykład: Dosunięcie po prostej, REPOSA, REPOSL

Narzędzie wykonuje ruch do punktu dosunięcia przywracającego bezpośrednio po prostej.

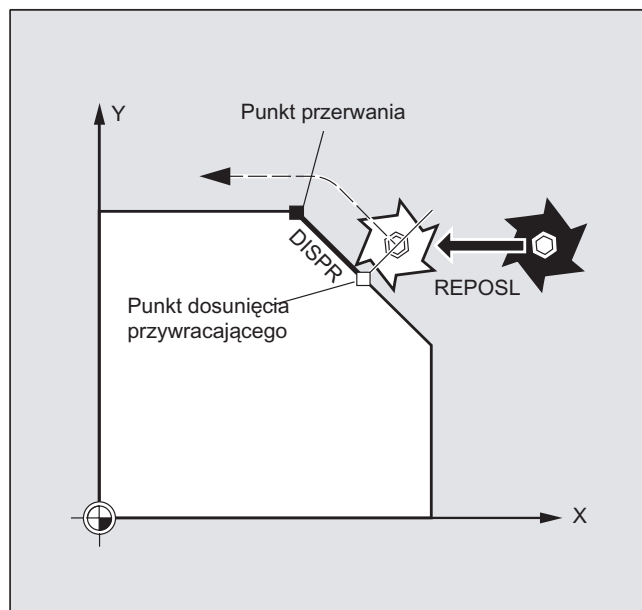
Przy pomocy REPOSA następuje automatycznie ruch we wszystkich osiach. W przypadku REPOSL można podać osie, w których ma zostać wykonany ruch.

Przykład:

```
REPOSL RMIBL DISPR=6 F400
```

lub

```
REPOSA RMIBL DISPR=6 F400
```

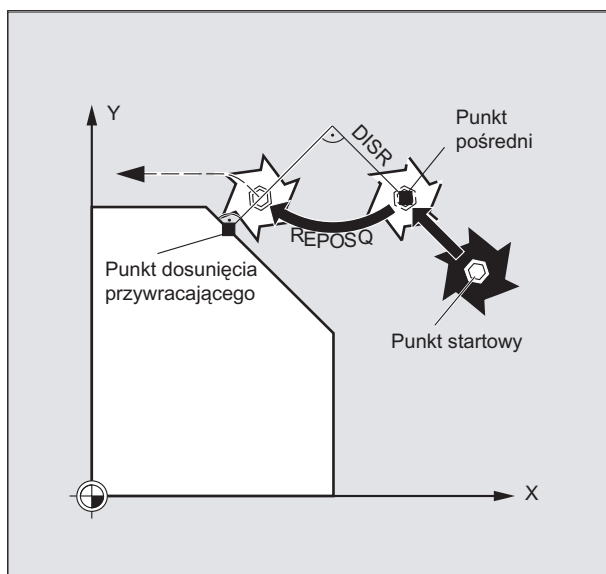


Przykład: Dosunięcie po ćwierćokręgu, REPOSQ, REPOSQA

Narzędzie wykonuje ruch do punktu dosunięcia przywracającego po ćwierćokręgu o promieniu $DISR=...$. Niezbędny punkt pośredni między punktem startowym i punktem dosunięcia przywracającego sterowanie oblicza automatycznie.

Przykład:

REPOSQ RMIBL DISR=10 F400

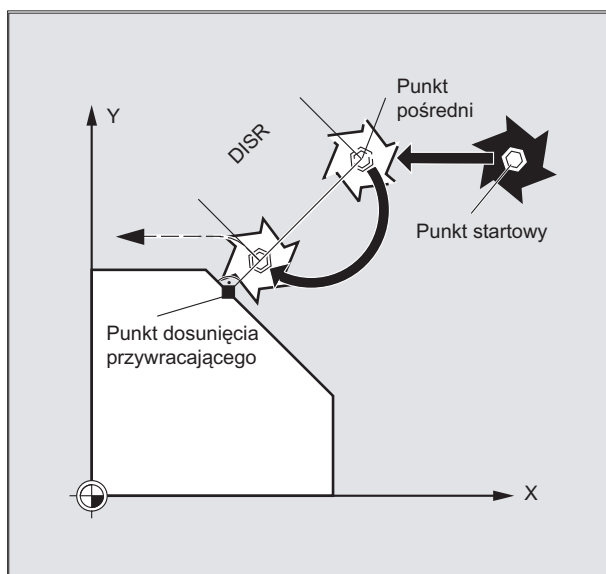


Przykład: Dosunięcie narzędzia po półokręgu, REPOSH, REPOSHA

Narzędzie wykonuje ruch do punktu dosunięcia przywracającego po półokręgu o średnicy $DISR=...$. Niezbędny punkt pośredni między punktem startowym i punktem dosunięcia przywracającego sterowanie oblicza automatycznie.

Przykład:

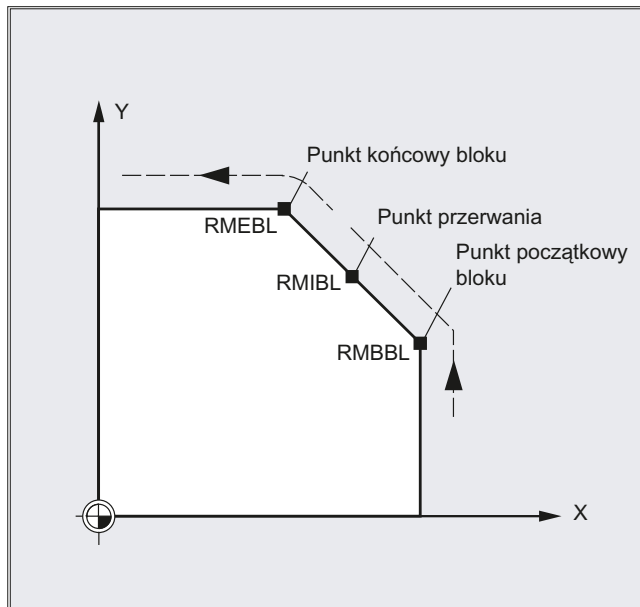
REPOSH RMIBL DISR=20 F400



Ustalenie punktu dosunięcia przywracającego (nie dla dosunięcia SERUPRO przy pomocy RMNBL)

W odniesieniu do bloku NC, w którym przebieg programu został przerwany, możecie wybierać między trzema punktami dosunięcia przywracającego:

- RMIBL, punkt przerwania
- RMBBL, punkt początkowy bloku wzgl. ostatni punkt końcowy
- RMEBL, punkt końcowy bloku



Przy pomocy RMIBL DISPR=... wzgl. RMEBL DISPR=... można ustalić punkt dosunięcia przywracającego, który leży przed punktem przerwania wzgl. przed punktem końcowym bloku.

Przy pomocy DISPR=... opisujemy drogę po konturze w mm/calach, o którą punkt dosunięcia przywracającego leży przed punktem przerwania wzgl. punktem końcowym. Punkt ten może leżeć maksymalnie w punkcie początkowym bloku - również dla większych wartości.

Jeżeli DISPR=... nie zostanie zaprogramowane, obowiązuje DISPR=0, a przez to punkt przerwania (w przypadku RMIBL) wzgl. punkt końcowy bloku (w przypadku RMEBL).

Znak DISPR

Znak DISPR podlega ewaluacji. W przypadku znaku dodatniego zachowanie jest jak dotychczas.

W przypadku znaku ujemnego dosunięcie następuje za punktem przerwania wzgl. w przypadku RMBBL za punktem startowym.

Odstęp punkt przerwania - punkt rozpoczęcia wynika z wartości bezwzględnej DISPR. Ten punkt może również dla większych wartości absolutnych leżeć maksymalnie w punkcie końcowym bloku.

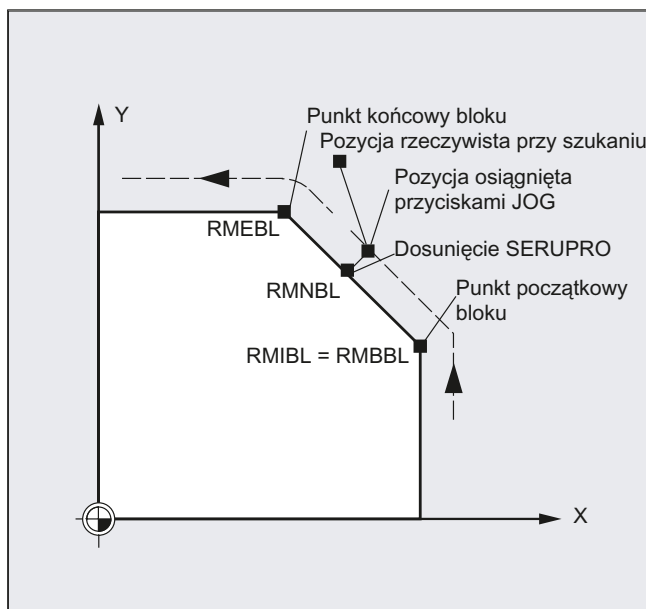
Przykład zastosowania:

Poprzez czujnik zostaje rozpoznane zbliżenie się do łapy dociskowej. Zostaje wyzwolone ASUP, przy pomocy którego łapa dociskowa jest omijana.

Następnie z ujemnym DISPR następuje repozycjonowanie na punkt za łapą i program jest kontynuowany.

Dosuw SERUPRO przy pomocy RMNBL

Jeżeli przy obróbce zostanie w dowolnym miejscu wymuszone przerwanie, wówczas następuje przy pomocy dosunięcia SERUPRO pod RMNBL dosunięcie po najkrótszej drodze od miejsca przerwania, aby następnie przeprowadzić obróbkę tylko po pozostałej drodze. W tym celu użytkownik uruchamia proces SERUPRO do bloku przerwania i przyciskami JOG pozycjonuje przed wadliwym miejscem bloku docelowego.



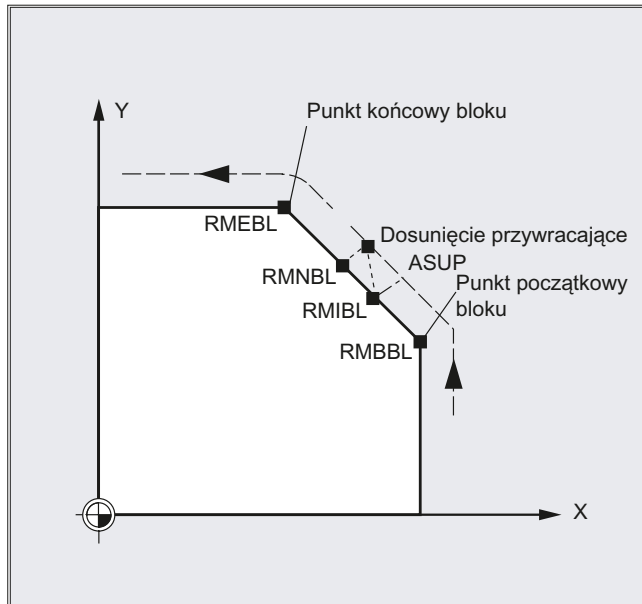
Wskazówka

SERUPRO

Dla SERUPRO są identyczne RMIBL i RMBBL. RMNBL nie jest ograniczone tylko do SERUPRO lecz obowiązuje ogólnie.

Dosunięcie do najbliższej położonego punktu toru RMNBL

W momencie interpretacji przez REPOSA po przerwaniu blok dosunięcia z RMNBL nie jest jeszcze raz rozpoczynany od początku, lecz jest wykonywana tylko pozostała droga. Następuje dosunięcie do najbliższej położonego punktu na torze w przerwany blok.



Status dla obowiązującego trybu REPOS

Obowiązujący tryb REPOS przerwany blok można odczytać poprzez akcje synchroniczne przy pomocy zmiennej \$AC_REPOS_PATH_MODE:

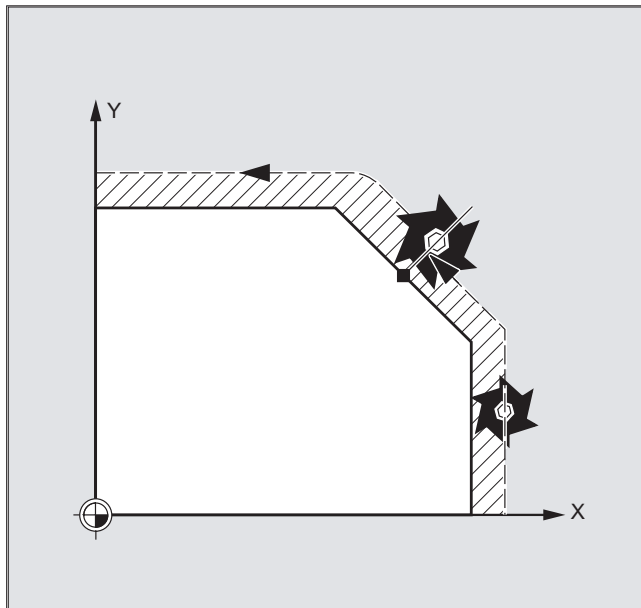
- 0 Dosunięcie nie zdefiniowane
- 1 RMBBL: Dosunięcie do początku
- 2 RMIBL: Dosunięcie do punktu przerwania
- 3 RMEBL: Dosunięcie do punktu końcowego bloku
- 4 RMNBL: Dosunięcie do najbliższej położonego punktu toru przerwany blok.

Dosunięcie z nowym narzędziem

W przypadku gdy nastąpiło zatrzymanie przebiegu programu z powodu pęknięcia narzędzia:

Z zaprogramowaniem nowego numeru D program jest kontynuowany od punktu dosunięcia przywracającego ze zmienionymi wartościami korekcji narzędzia.

Przy zmienionych wartościach korekcji narzędzia możliwe jest, że nie będzie już można dokonać ponownego dosunięcia do punktu przerwania. W tym przypadku dosunięcie następuje do punktu położonego najbliższej punktu przerwania na nowym konturze (ewentualnie zmodyfikowanego o DISPR).



Dosunięcie do konturu

Ruch, którym narzędzie ponownie zbliża się do konturu, daje się programować. Adresy osi ruchu podajemy z wartością zero.

Przy pomocy poleceń REPOSA, REPOSQA i REPOSHA są automatycznie rezyzyjonowane wszystkie osie. Podanie osi nie jest konieczne.

Przy zaprogramowaniu REPOSL, REPOSQ i REPOSH wszystkie osie geometryczne wykonują ruch automatycznie, również bez podania w poleceniu. Wszystkie inne osie muszą zostać podane w poleceniu.

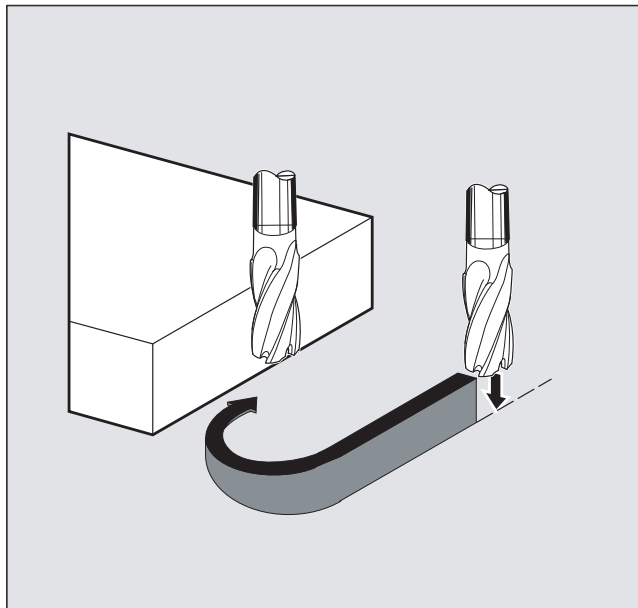
Dla ruchów kołowych REPOSH i REPOSQ obowiązuje:

Ruch po okręgu jest wykonywany w podanej płaszczyźnie obróbki G17 do G19.

W przypadku gdy w bloku dosuwu zostanie podana trzecią oś geometryczna (kierunek dosuwu), ruch do punktu dosunięcia przywracającego na wypadek, że pozycja narzędzia i zaprogramowana pozycja w kierunku dosuwu nie są zgodne, jest wykonywany po linii śrubowej.

W następujących przypadkach następuje automatyczne przełączenie na dosunięcie liniowe REPOSL:

- Nie podano wartości DISR.
- Nie ma zdefiniowanego kierunku dosunięcia (przerwanie programu w bloku bez informacji o ruchu).
- W przypadku kierunku dosunięcia prostopadłe do aktualnej płaszczyzny roboczej.



10.10 Sterowanie prowadzeniem ruchu

10.10.1 Procentowa korekcja przyśpieszenia drugiego stopnia (JERKLIM)

Funkcja

Przy pomocy polecenia NC `JERKLIM` można zmniejszyć albo zwiększyć ustawione poprzez daną maszynową maksymalnie możliwe przyśpieszenie drugiego stopnia osi przy ruchu po torze w krytycznych fragmentach programu.

Warunek

Tryb przyśpieszenia SOFT musi być aktywny.

Działanie

Funkcja działa:

- w trybach pracy AUTOMATYKA.
- tylko na osie uczestniczące w tworzeniu konturu.

Składnia

`JERKLIM[<oś>]=<wartość>`

Znaczenie

<code>JERKLIM:</code>	Polecenie do korekcji przyśpieszenia drugiego stopnia
<code><oś>:</code>	Oś maszyny, której wartość graniczna przyśpieszenia drugiego stopnia ma zostać dopasowana.
<code><wartość>:</code>	Procentowa wartość korekcji, odniesiona do zaprojektowanego maksymalnego przyspieszenia drugiego stopnia w osi przy ruchu po torze (MD32431 <code>\$MA_MAX_AX_JERK</code>). Zakres wartości: 1 ... 200 Wartość 100 powoduje brak sterowania przyśpieszeniem drugiego stopnia.

Wskazówka

Zachowanie się JERKLIM na końcu programu obróbki i przy zresetowaniu kanału jest projektowane przy pomocy bitu 0 w danej maszynowej MD32320 `$MA_DYN_LIMIT_RESET_MASK`:

- Bit 0 = 0:
Zaprogramowana wartość JERKLIM jest przez zresetowanie kanału / M30 cofana do 100%.
 - Bit 0 = 1:
Zaprogramowana wartość JERKLIM pozostaje zachowana po zresetowaniu kanału / M30.
-

Przykład

Kod programu	Komentarz
...	
N60 JERKLIM[X]=75	; Sanie osi w kierunku X mogą przyspieszać/zwalniać tylko z max 75% dopuszczalnego dla osi przyśpieszenia drugiego stopnia.
...	

10.10.2 Procentowa korekcja prędkości (VELOLIM)

Funkcja

Przy pomocy polecenia `VELOLIM` można w programie obróbki albo akcji synchronicznej zmniejszyć ustawioną poprzez daną maszynową maksymalnie możliwą szybkość osi wzgl. maksymalnie możliwą, zależną od stopnia przekładni, prędkość obrotową wrzeciona.

Działanie

Funkcja działa:

- w trybach pracy AUTOMATYKA.
- na osie uczestniczącej w tworzeniu konturu i osie pozycjonowania.
- na wrzeciono w trybie wrzeciona/osi

Składnia

`VELOLIM [<os/wrzeciono>] = <wartość>`

Znaczenie

`VELOLIM:` Polecenie do korekcji prędkości

`<os/wrzeciono>:` Oś albo wrzeciono, którego wartość graniczna prędkości albo prędkości obrotowej ma zostać dopasowana.

VELOLIM dla wrzecion

Poprzez daną maszynową (MD30455 \$MA_MISC_FUNCTION_MASK, Bit 6) można dla programowania ustawić w programie obróbki, czy `VELOLIM` działa niezależnie od aktualnego zastosowania jako wrzeciono lub oś (bit 6 = 1), czy ma być programowalne oddzielnie dla każdego rodzaju pracy (bit 6 = 0). Jeżeli jest zaprojektowane oddzielne działanie, wówczas wybór jest dokonywany przez identyfikator przy programowaniu:

- Identyfikator `S<n>` trybów pracy wrzeciona
- Identyfikator osi, np. "c", dla trybu pracy jako oś

<wartość>:	Procentowa wartość korekcji
	Wartość korekcji odnosi się:
	<ul style="list-style-type: none">w przypadku osi / wrzecion w trybie pracy jako oś (gdy MD30455 bit 6 == 0): do zaprojektowanej maksymalnej prędkości osi (MD32000 \$MA_MAX_AX_VELO).w przypadku wrzecion w trybie pracy jako wrzeciono albo oś (gdy MD30455 bit 6 == 1): do maksymalnej prędkości obrotowej aktywnego stopnia przekładni (MD35130 \$MA_GEAR_STEP_MAX_VELO_LIMIT[<n>])
	Zakres wartości: 1 ... 100
	Wartość 100 nie powoduje wpływu na prędkość wzgl. prędkość obrotową.

Wskazówka

Zachowanie się na końcu programu obróbki i zresetowaniu kanału

Zachowanie się `VELOLIM` na końcu programu obróbki i przy zresetowaniu kanału jest ustawiane poprzez daną maszynową MD32320 `$MA_DYN_LIMIT_RESET_MASK`, bit 0

Rozpoznanie aktywnego ograniczenia prędkości obrotowej w trybie pracy jako wrzeciono

Ograniczenie prędkości obrotowej przez `VELOLIM` (mniejsze niż 100%) można w trybie pracy jako wrzeciono rozpoznać poprzez następującą zmienną systemową:

- `$AC_SMAXVELO` (maksymalna możliwa prędkość obrotowa wrzeciona)
- `$AC_SMAXVELO_INFO` (oznaczenie przyczyny ograniczenia prędkości obrotowej)

Przykłady

Przykład 1: Ograniczenie prędkości osi maszyny

Kod programu	Komentarz
...	
N70 VELOLIM[X]=80	; Ruch sań osi w kierunku X powinien być wykonywany tylko z max 80% prędkości dopuszczalnej dla osi.
...	

Przykład 2: Ograniczenie prędkości obrotowej wrzeciona

Kod programu	Komentarz
N05 VELOLIM[S1]=90	; Ograniczenie maksymalnej prędkości obrotowej wrzeciona 1 do 90% z 1000 obr/min.
...	
N50 VELOLIM[C]=45	; Ograniczenie prędkości obrotowej do 45% z 1000 obr/min, niech C będzie identyfikatorem osi S1.
...	

Ustawienia danych maszynowych dla wrzeciona 1 (AX5)

- Maksymalna prędkość obrotowa stopnia przekładni 1 = 1000 obr/min:
 MD35130 \$MA_GEAR_STEP_MAX_VELO_LIMIT[1, AX5] = 1000
- Zaprogramowanie VELOLIM działa wspólnie dla pracy wrzeciona i osi niezależnie od zaprogramowanego identyfikatora:
 MD30455 \$MA_MISC_FUNCTION_MASK[AX5], bit 6 = 1

10.10.3 Przykład programu dla JERKLIM i VELOLIM

Poniższy program przedstawia przykład zastosowania procentowego ograniczenia przyspieszenia drugiego stopnia i prędkości:

Kod programu	Komentarz
N1000 G0 X0 Y0 F10000 SOFT G64	
N1100 G1 X20 RNDM=5 ACC[X]=20 ACC[Y]=30	
N1200 G1 Y20 VELOLIM[X]=5	; Ruch sań osi w kierunku X powinien być wykonywany tylko z max 5% prędkości dopuszczalnej dla osi.
JERKLIM[Y]=200	; Sanie osi w kierunku Y mogą przyspieszać/zwalniać z max 200% dopuszczalnego dla osi przyspieszenia drugiego stopnia.
N1300 G1 X0 JERKLIM[X]=2	; Sanie osi w kierunku X mogą przyspieszać/zwalniać tylko z max 2% dopuszczalnego dla osi przyspieszenia drugiego stopnia.
N1400 G1 Y0	
M30	

10.11 Programowana tolerancja konturu/orientacji (CTOL, OTOL, ATOL)

Funkcja

Przy pomocy poleceń CTOL, OTOL i ATOL można w programie NC dopasować ustalone przez dane maszynowe i ustawcze tolerancje obróbki dla funkcji kompresora (COMPON, COMPCURV, COMPCAD), rodzajów ścinania naroży G642, G643, G645, OST i wygładzania orientacji ORISON.

Zaprogramowane wartości obowiązują, aż zostaną na nowo zaprogramowane albo skasowane przez przyporządkowanie ujemnej wartości. Są one ponadto kasowane na końcu programu, przy zresetowaniu kanału, zresetowaniu BAG, zresetowaniu NCK (start ciepły) i Power On (start zimny). Po skasowaniu obowiązują ponownie wartości z danych maszynowych i ustawczych.

Składnia

```
CTOL=<wartość>  
OTOL=<wartość>  
ATOL [<oś>] =<wartość>
```

Znaczenie

CTOL	<p>Polecenie do programowania tolerancji konturu</p> <p>CTOL obowiązuje dla:</p> <ul style="list-style-type: none">• wszystkich funkcji kompresora• wszystkich rodzajów ścinania naroży oprócz G641 i G644 <p><wartość>: Wartością tolerancji konturu jest podana długość.</p> <p>Typ: REAL</p> <p>Jednostka: cal/mm (zależnie od aktualnego ustawienia podawania wymiarów)</p>
OTOL	<p>Polecenie do programowania tolerancji orientacji</p> <p>OTOL obowiązuje dla:</p> <ul style="list-style-type: none">• wszystkich funkcji kompresora• wygładzania orientacji ORISON• wszystkich rodzajów ścinania naroży oprócz G641, G644, OSD <p><wartość>: Wartością tolerancji orientacji jest podany kąt.</p> <p>Typ: REAL</p> <p>Jednostka: stopień</p>

ATOL	Polecenie do programowania tolerancji specyficznej dla osi
	ATOL obowiązuje dla:
	<ul style="list-style-type: none"> • wszystkich funkcji kompresora • wygładzania orientacji ORISON • wszystkich rodzajów ścinania naroży oprócz G641, G644, OSD
<oś>:	Nazwa osi, dla której ma zostać zaprogramowana tolerancja osi
<wartość>:	Wartością tolerancji osi jest zależnie od typu osi (liniowa lub obrotowa) podana długość lub podany kąt.
	Typ: REAL
	Jednostka: Dla osi liniowych: cal/mm (zależnie od aktualnego ustawienia podawania wymiarów)
	Dla osi obrotowych: stopień

Wskazówka

CTOL i OTOL mają pierwszeństwo przed ATOL.

Warunki brzegowe

Frame skalujące

Frame skalujące działają na zaprogramowane tolerancje w taki sam sposób jak na pozycje w osiach, tzn. tolerancja względna pozostaje taka sama.

Przykład

Kod programu	Komentarz
COMPCAD G645 G1 F10000	; Uaktywnienie funkcji kompresora COMPCAD.
X... Y... Z...	; Tutaj działają dane maszynowe i ustawcze.
X... Y... Z...	
X... Y... Z...	
CTOL=0.02	; Od tego miejsca działa tolerancja konturu 0,02 mm.
X... Y... Z...	
X... Y... Z...	
X... Y... Z...	
ASCALE X0.25 Y0.25 Z0.25	; Od tego miejsca działa tolerancja konturu 0,005 mm.
X... Y... Z...	
X... Y... Z...	
X... Y... Z...	

Kod programu	Komentarz
CTOL=-1	; Od tego miejsca działają dane maszynowe i ustawcze.
X... Y... Z...	
X... Y... Z...	
X... Y... Z...	

Dalsze informacje

Odczyt wartości tolerancji

Dla dalej idących przypadków zastosowania albo do diagnozy, aktualnie obowiązujące tolerancje dla funkcji kompresora (COMPON, COMPCURV, COMPCAD), rodzajów ścinania narożników G642, G643, G645, OST i wygładzania orientacji ORISON są niezależnie od rodzaju powstania możliwe do odczytania poprzez zmienne systemowe.

- W akcjach synchronicznych lub z zatrzymaniem przebiegu wyprzedzającego w programie obróbki przez zmienne systemowe:

\$AC_CTOL	Tolerancja konturu, która działała przy przygotowywaniu aktualnego bloku przebiegu głównego Jeżeli nie działa żadna tolerancja konturu, \$AC_CTOL daje pierwiastek sumy kwadratów tolerancji osi geometrycznych.
\$AC_OTOL	Tolerancja orientacji, która działała przy przygotowywaniu aktualnego bloku przebiegu głównego Jeżeli żadna tolerancja orientacji nie działa, \$AC_OTOL podczas aktywnej transformacji orientacji daje pierwiastek sumy kwadratów tolerancji osi orientacji, w przeciwnym przypadku wartość "-1".
\$AA_ATOL[<os>]	Tolerancja osi, która działała przy przygotowywaniu aktualnego bloku przebiegu głównego W przypadku gdy jest aktywna tolerancja konturu, \$AA_ATOL[<os geometryczna>] daje tolerancję konturu podzieloną przez pierwiastek liczby osi geometrycznych. W przypadku gdy jest aktywna tolerancja orientacji i transformacja orientacji, \$AA_ATOL[<os orientacji>] daje tolerancję orientacji podzieloną przez pierwiastek liczby osi orientacji.

Wskazówka

Gdy nie zostały zaprogramowane wartości tolerancji, wówczas zmienne \$A nie są dostatecznie zróżnicowane, aby rozróżnić możliwie różne tolerancje poszczególnych funkcji, ponieważ mogą podać tylko jedną wartość.

Takie przypadki mogą wystąpić, gdy dane maszynowe i ustawcze ustawiają różne tolerancje funkcji kompresora, ścinania naroży i wygładzania orientacji. Zmienne dają wówczas największą wartość, która występuje przy właśnie aktywnych funkcjach.

Gdy jest np. aktywna funkcja kompresora z tolerancją orientacji 0,1° i wygładzanie orientacji ORISON z 1°, zmienna \$AC_OTOL daje wartość "1". Gdy wygładzanie orientacji zostanie wyłączone, czyta się już tylko wartość "0.1".

- Bez zatrzymania przebiegu wyprzedzającego w programie obróbki przez zmienne systemowe:

\$P_CTOL	Programowana tolerancja konturu
\$P_OTOL	Programowana tolerancja orientacji
\$PA_ATOL	Programowana tolerancja osi

Wskazówka

Gdy nie są zaprogramowane żadne wartości tolerancji, wówczas zmienne \$P dają wartość "-1".

10.12 Tolerancja przy ruchach G0 (STOLF)

Współczynnik tolerancji G0

Ruchy G0 (posuw szybki, ruchy dosuwu) mogą w odróżnieniu od obróbki być wykonywane z większą tolerancją. Ma to tę zaletę, że skraca się czas ruchów G0.

Ustawienie tolerancji w przypadku ruchów G0 następuje przez zaprojektowanie współczynnika tolerancji G0 (MD20560 \$MC_G0_TOLERANCE_FACTOR).

Okno tolerancji G0 działa tylko wtedy, gdy:

- Jest aktywna jedna z następujących funkcji:
 - Funkcje kompresora: COMPON, COMPCURV i COMPCAD
 - Funkcje ścinania naroży: G642 i G645
 - Ścinanie orientacji: OST
 - Wygładzanie orientacji: ORISON
 - Wygładzanie orientacji względem toru: ORIPATH

- Wiele (≥ 2) bloków G0 następuje jeden po drugim.

W przypadku pojedynczego G0 współczynnik tolerancji G0 nie działa, ponieważ **przy przejściu** z ruchu nie G0 na ruch G0 (i na odwrót) zasadniczo działa "mniejsza tolerancja" (tolerancja obróbki)!

Funkcja

Przez zaprogramowanie `STOLF` w programie obróbki zaprojektowany współczynnik tolerancji G0 (MD20560) można tymczasowo zastąpić. Wartość w MD20560 nie jest przy tym zmieniana. Po zresetowaniu lub zakończeniu programu zaprojektowany współczynnik tolerancji ponownie działa.

Składnia

`STOLF=<współczynnik tolerancji>`

Znaczenie

`STOLF:` Polecenie do programowania współczynnika tolerancji G0
`<współczynnik tolerancji>:` Współczynnik tolerancji G0
Współczynnik może być zarówno większy od 1, jak i mniejszy od 1. Normalnie będą jednak dla ruchów G0 możliwe do ustawiania większe tolerancje.
Przy `STOLF=1.0` (odpowiada zaprojektowanej wartości standardowej) działają dla ruchów G0 te same tolerancje, co dla ruchów nie G0.

Zmienne systemowe

Współczynnik tolerancji G0 działający w programie obróbki lub w aktualnym bloku IPO można odczytać poprzez zmienne systemowe.

- W akcjach synchronicznych lub z zatrzymaniem przebiegu wyprzedzającego w programie obróbki przez zmienną systemową:

\$AC_STOLF Aktywny współczynnik tolerancji G0
Współczynnik tolerancji G0, który działał przy przygotowywaniu aktualnego bloku przebiegu głównego.

- Bez zatrzymania przebiegu wyprzedzającego w programie obróbki przez zmienną systemową:

\$P_STOLF Zaprogramowany współczynnik tolerancji G0

Jeżeli w aktywnym programie obróbki nie jest zaprogramowana wartość ze `STOLF`, wówczas te dwie zmienne systemowe dają wartość ustawioną przez MD20560 `$MC_G0_TOLERANCE_FACTOR`.

Jeżeli w bloku nie jest aktywny posuw szybki (G0), wówczas te zmienne systemowe dają zawsze wartość 1.

Przykład

Kod programu	Komentarz
COMPCAD G645 G1 F10000	; Funkcja kompresora COMPCAD
X... Y... Z...	; Tutaj działają dane maszynowe i ustawcze.
X... Y... Z...	
X... Y... Z...	
G0 X... Y... Z...	
G0 X... Y... Z...	; Tutaj działa dana maszynowa <code>\$MC_G0_TOLERANCE_FACTOR</code> (np. =3), a więc tolerancja ścinania naroży z <code>\$MC_G0_TOLERANCE_FACTOR*\$MA_COMPRESS_POS_TOL</code> .
CTOL=0.02	
STOLF=4	
G1 X... Y... Z...	; Od tego miejsca działa tolerancja konturu 0,02 mm.
X... Y... Z...	
X... Y... Z...	
G0 X... Y... Z...	
X... Y... Z...	; Od tego miejsca działa G0 współczynnik tolerancji 4, a więc tolerancja konturu 0,08 mm.

10.13 Zmiana bloku przy aktywnym sprzężeniu (CPBC)

Funkcja

Przy pomocy polecenia CPBC jest zadawane kryterium zmiany bloku, które musi być spełnione, aby w programie obróbki przy aktywnym sprzężeniu została przeprowadzona zmiana bloku.

Składnia

CPBC [<oś nadążna>] = <kryterium>

Znaczenie

CPBC:	Kryterium zmiany bloku przy aktywnym sprzężeniu
<oś nadążna>:	Identyfikator osi nadążnej
<kryterium>:	Kryterium zmiany bloku
	Typ: STRING
Wartość	Znaczenie: Zmiana bloku nastąpi
"NOC"	niezależnie od stanu sprzężenia
"IPOSTOP"	przy ruchu synchronicznym po stronie wartości zadanej
"COARSE"	przy ruchu synchronicznym "zgrubnie" po stronie wartości rzeczywistej
"FINE"	przy ruchu synchronicznym "dokładnie" po stronie wartości rzeczywistej

Przykład

Kod programu

```
; Zmiana bloku następuje przy:  
; - sprzężenie z osią nadążną X2 == aktywne  
; - ruch synchroniczny po stronie wartości zadanej == aktywny  
CPBC[X2]="IPOSTOP"
```

Sprężenia osi

11.1 Nadążanie (TRAILON, TRAILOF)

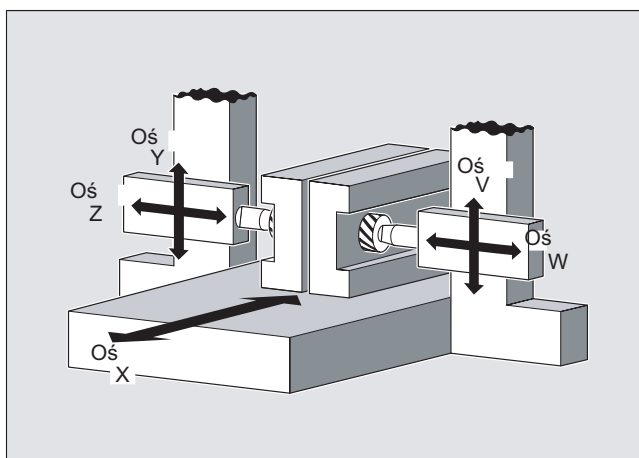
Funkcja

Przy ruchu zdefiniowanej osi wiodącej przyporządkowane jej osie nadążne wykonują, przy uwzględnieniu współczynnika sprzężenia, drogi ruchu wyprowadzone od osi wiodącej.

Oś wiodąca i osie nadążne tworzą razem zespół nadążania.

Zakresy zastosowania

- Ruch w osi za pośrednictwem osi symulowanej. Oś wiodąca jest osią symulowaną, a oś nadążna osią rzeczywistą. W ten sposób można wykonywać ruch w osi rzeczywistej z uwzględnieniem współczynnika sprzężenia.
- Obróbka dwustronna z dwoma zespołami nadążania:
 1. Oś wiodąca Y, oś nadążna V
 2. Oś wiodąca Z, oś nadążna W



Składnia

TRAILON(<oś nadążna>, <oś wiodąca>, <współczynnik sprzężenia>)

TRAILOF(<oś nadążna>, <oś wiodąca>, <oś wiodąca 2>)

TRAILOF(<oś nadążna>)

Znaczenie

TRAILON	Polecenie do włączenia i definicji zespołu nadążania
	Działanie: Modalnie
<oś nadążna>	Parametr 1: Określenie osi nadążnej
	Wskazówka: Oś nadążna może także być osią wiodącą dla dalszych osi nadążnych. W ten sposób mogą być budowane różne zespoły nadążania.
<oś wiodąca>	Parametr 2: Określenie osi wiodącej
<współczynnik sprzężenia>	Parametr 3: Współczynnik sprzężenia
	Współczynnik sprzężenia podaje pożądany stosunek dróg osi nadążnej i osi wiodącej: <Współczynnik sprzężenia> = droga osi nadążnej/droga osi wiodącej
	Typ: REAL
	Ustawienie 1
	domyślne:
	Wprowadzenie wartości ujemnej powoduje przeciwny kierunek ruchu osi wiodącej i osi nadążnej.
	Jeżeli przy programowaniu współczynnik sprzężenia nie zostanie podany, wówczas automatycznie obowiązuje współczynnik 1.
TRAILOF	Polecenie do wyłączenia zespołu nadążania
	Działanie: Modalnie
	TRAILOF z 2 parametrami wyłącza tylko sprzężenie z podaną osią wiodącą:
	TRAILOF (<oś nadążna>, <oś wiodąca>)
	Jeżeli oś nadążna ma 2 osie wiodące, może nastąpić wezwanie do wyłączenia obydwu sprzężeń TRAILOF z 3 parametrami:
	TRAILOF (<oś nadążna>, <oś wiodąca>, <oś wiodąca 2>)
	Ten sam wynik daje zaprogramowanie TRAILOF bez podania osi wiodącej:
	TRAILOF (<oś nadążna>)

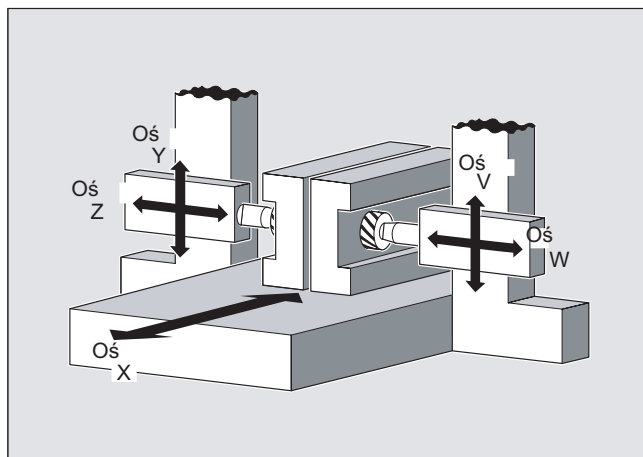
Wskazówka

Nadążanie następuje zawsze w bazowym układzie współrzędnych (BKS).

Liczba zespołów nadążania, możliwych do równoczesnego uaktywnienia, jest ograniczona tylko przez możliwości kombinacji osi istniejących w maszynie.

Przykład

Przedmiot ma być obrabiany dwustronnie z przedstawioną konstelacją osi. W tym celu należy stworzyć 2 zespoły nadażania



Kod programu	Komentarz
...	
N100 TRAILON(V,Y)	; Włączenie 1. zespołu nadażania
N110 TRAILON(W,Z,-1)	; Włączenie 2. zespołu nadażania. Współczynnik sprzężenia ujemny: Oś nadażna wykonuje ruch w kierunku przeciwnym względem osi wiodącej.
N120 G0 Z10	; Dosuw w osi Z i W w przeciwnym kierunku osi.
N130 G0 Y20	; Dosuw w osi Y i V w tym samym kierunku osi.
...	
N200 G1 Y22 V25 F200	; Nałożenie zależnego i niezależnego ruchu osi nadażnej V.
...	
TRAILOF(V,Y)	; Wyłączenie 1. zespołu nadażania.
TRAILOF(W,Z)	; Wyłączenie 2. zespołu nadażania.

Dalsze informacje

Typy osi

Zespół osi nadążnych może składać się z dowolnych kombinacji osi liniowych i obrotowych. Jako oś wiodąca może przy tym również zostać zdefiniowana również oś symulowana.

Osie nadążne

Osi nadążnej mogą równocześnie zostać przyporządkowane maksymalnie 2 osie wiodące. Przyporządkowanie następuje w różnych zespołach nadążania.

Oś nadążna może być programowana przy pomocy wszystkich będących do dyspozycji poleceń ruchu (G0, G1, G2, G3, ...). Dodatkowo oprócz niezależnie definiowanych dróg oś nadążna wykonuje ruchy po drogach wyprowadzonych z jej osi wiodących ze współczynnikami sprzężenia.

Ograniczenie dynamiki

Ograniczenie dynamiki jest zależne od rodzaju uaktywnienia zespołu nadążania:

- Uaktywnienie w programie obróbki

Jeżeli uaktywnienie nastąpi w programie obróbki i wszystkie osie wiodące znajdują się jako osie programowe w kanale uaktywniającym, wówczas przy wykonywaniu ruchów osi wiodących dynamika wszystkich osi nadążnych jest tak uwzględniana, by żadna z nich nie została przeciążona.

Jeżeli uaktywnienie następuje w programie obróbki z osiami wiodącymi, które nie są aktywne jako osie programowe w kanale uaktywniającym (\$AA_TYP ≠ 1), przy wykonywaniu ruchów osi wiodących dynamika osi nadążnej nie jest uwzględniana. W wyniku tego w przypadku osi nadążnych może dojść do przeciążenia z dynamika mniejszą, niż potrzebna dla sprzężenia.

- Uaktywnienie w akcji synchronicznej

Jeżeli uaktywnienie nastąpi w akcji synchronicznej, przy wykonywaniu ruchów w osiach wiodących dynamika osi nadążnych nie jest uwzględniana. W wyniku tego w przypadku osi nadążnych może dojść do przeciążenia z dynamika mniejszą, niż potrzebna dla sprzężenia.

OSTROŻNIE

Przeciążenie osi

Jeżeli zespół nadążania zostanie uaktywniony

- w akcjach synchronicznych
- w programie obróbki z osiami wiodącymi, które nie są osiami programowymi w kanale osi nadążnej,

, wówczas w zakresie szczególnej odpowiedzialności użytkownika/producenta maszyny leży podjęcie odpowiednich środków, by w wyniku ruchów postępowych osi wiodącej nie doszło do przeciążenia osi nadążnych.

Status sprzężenia

Status sprzężenia osi można odpytać w programie obróbki przy pomocy zmiennych systemowych:

\$AA_COUP_ACT[<oś>]

Wartość	Znaczenie
0	Nie jest aktywne żadne sprzężenie
8	Nadążanie aktywne

Wyświetlenie osi nadążnej w przypadku osi obrotowych moduło

Jeżeli oś wiodąca i oś nadążna są osiami obrotowymi moduło, sumują się ruchy postępowe osi wiodącej $n * 360^\circ$, gdzie $n = 1, 2, 3...$, na wyświetleniu pozostałej drogi osi holowanej, aż do wyłączenia sprzężenia.

Przykład: Segment programu z TRAILON, oś wiodąca B i oś nadążna C

Kod programu	Komentarz
TRAILON(C,B,1)	; Włączenie sprzężenia
G0 B0	; Pozycja wyjściowa
	; Wyświetlenie pozostałej drogi na początku bloku:
G91 B360	; B=360, C=360
G91 B720	; B=720, C=1080
G91 B360	; B=360, C=1440

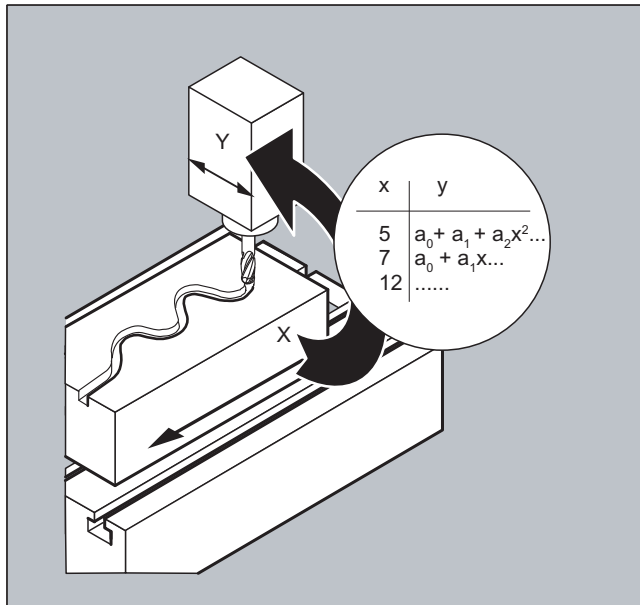
11.2 Tablice krzywych (CTAB)

Funkcja

Przy pomocy tablic krzywych można programować zależności pozycji i prędkości między dwoma osiami (wiodącą i nadążną). Definicja tablic krzywych następuje w programie obróbki.

Zastosowanie

Tablice krzywych zastępują mechaniczne tarcze krzywkowe. Tablica krzywych tworzy przy tym podstawę osiowego sprzężenia wartości wiodącej w ten sposób, że stwarza funkcjonalną zależność między wartością wiodącą i nadążną. Przy odpowiednim zaprogramowaniu sterowanie z przyporządkowanych do siebie pozycji osi wiodącej i nadążnej oblicza wielomian, który odpowiada tarczy krzywkowej.

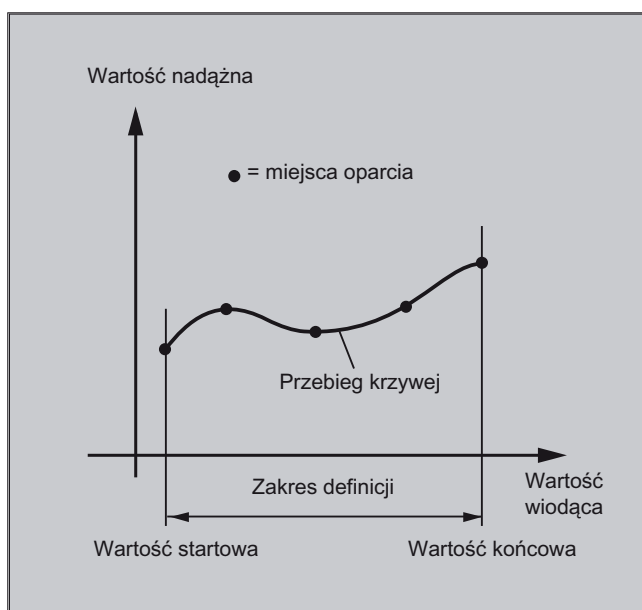


11.2.1 Definiowanie tablic krzywych (CTABDEF, CTABEND)

Funkcja

Tablica krzywych stanowi program obróbki albo jego segment, który charakteryzuje się poleceniem CTABDEF na początku i CTABEND na końcu.

W ramach tego segmentu programu obróbki są przez polecenia ruchu do poszczególnych pozycji osi wiodącej przyporządkowywane jednoznaczne pozycje osi nadążnej, które służą jako węzły interpolacji do obliczania przebiegu krzywej w formie wielomianu do maksymalnie 5. stopnia.



Warunek

Dla definicji tablic krzywych musi przez odpowiednie zaprojektowanie MD być zarezerwowane miejsce w pamięci (→ producent maszyny!).

Składnia

```
CTABDEF (<oś nadążna>, <oś wiodąca>, <n>, <okresowość> [, <miejsce
zapisania>])
...
CTABEND
```

Znaczenie

CTABDEF ()	Początek definicji tablicy krzywych
CTABEND	Koniec definicji tablicy krzywych
<oś nadążna>	Oś, której ruch ma zostać obliczony poprzez tablicę krzywych
<oś wiodąca>	Oś, która daje wartości wiodące do obliczania ruchu osi nadążnej

<n>	Numer (ID) tablicy krzywych Numer tablicy krzywych jest jednoznaczny i niezależny od miejsca w pamięci. Nie może być tablic o takim samym numerze w pamięci statycznej i pamięci dynamicznej NC.
<okresowość>	Okresowość tablicy 0 Tablica nie jest okresowa (jest wykonywana tylko jeden raz, również w przypadku osi obrotowych) 1 Tablica jest okresowa względem osi wiodącej 2 Tablica jest okresowa względem osi wiodącej i osi nadążnej
<miejsce zapisania>	Podanie miejsca zapisania (opcjonalne) "SRAM" Tablica krzywych jest tworzona w stacyjnej pamięci NC. "DRAM" Tablica krzywych jest tworzona w dynamicznej pamięci NC.
	Wskazówka: Gdy dla tego parametru nie zostanie zaprogramowana żadna wartość, wówczas zostanie użyte standardowe miejsce w pamięci, ustawione przy pomocy MD20905 \$MC_CTAB_DEFAULT_MEMORY_TYPE.

Wskazówka**Zastąpienie**

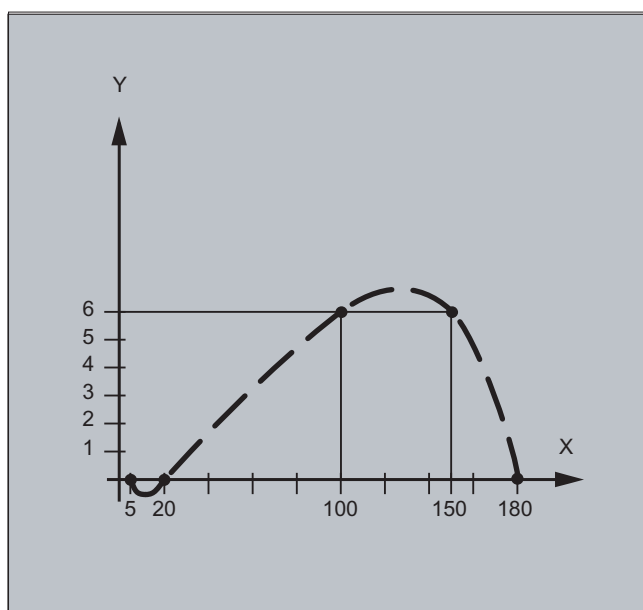
Tablica krzywych jest zastępowana, gdy tylko przy ponownej definicji tablicy zostanie użyty jej numer (<n>) (wyjątek: tablica krzywych jest aktywna w sprzężeniu osi albo zablokowana przy pomocy CTABLOCK). **Przy zastępowaniu tablic krzywych nie jest wyprowadzane odpowiednie ostrzeżenie!**

Przykłady**Przykład 1: Fragment programu jako definicja tablicy krzywych**

Segment programu ma bez zmian być używany do definicji tablicy krzywych. Występujące tam polecenie do zatrzymania przebiegu wyprzedzającego STOPRE może pozostać i natychmiast staje się ponownie aktywne, gdy tylko segment programu nie jest już używany do definicji tablicy, a CTABDEF i CTABEND zostały usunięte.

Kod programu	Komentarz
...	
CTABDEF (Y,X,1,1)	; Definicja tablicy krzywych.
...	
IF NOT (\$P_CTABDEF)	
STOPRE	
ENDIF	
...	
CTABEND	

Przykład 2: Definicja nieperiodycznej tablicy krzywych



Kod programu	Komentarz
N100 CTABDEF(Y,X,3,0)	; Początek definicji nieperiodycznej tablicy krzywych o numerze 3.
N110 X0 Y0	; 1. instrukcja ruchu, ustala wartości startowe i 1. węzeł interpolacji: Wartość wiodąca: 0, wartość nadażna: 0
N120 X20 Y0	; 2. węzeł interpolacji: Wartość wiodąca: 0...20, wartość nadażna: wartość startowa...0
N130 X100 Y6	; 3. węzeł interpolacji: Wartość wiodąca: 20...100, wartość nadażna: 0...6
N140 X150 Y6	; 4. węzeł interpolacji: Wartość wiodąca: 100...150, wartość nadażna: 6...6
N150 X180 Y0	; 5. węzeł interpolacji: Wartość wiodąca: 150...180, wartość nadażna: 6...0
N200 CTABEND	; Koniec definicji. Tablica krzywych jest w swoim wewnętrznym przedstawieniu wytwarzana jako wielomian maksymalnie 5. stopnia. Obliczenie przebiegu krzywej z podanymi węzłami interpolacji jest zależne od modalnie wybranego rodzaju interpolacji (interpolacja kołowa, liniowa, spline). Jest odtwarzany stan programu obróbki przed początkiem definicji.

Przykład 3: Definicja okresowej tablicy krzywych

Definicja periodycznej tablicy krzywych o numerze 2, zakres wartości prowadzącej od 0 do 360, ruch osi nadążnej od 0 do 45 i z powrotem do 0:

Kod programu	Komentarz
N10 DEF REAL DEPPPOS	
N20 DEF REAL GRADIENT	
N30 CTABDEF(Y,X,2,1)	; Początek definicji.
N40 G1 X=0 Y=0	
N50 POLY	
N60 PO[X]=(45.0)	
N70 PO[X]=(90.0) PO[Y]=(45.0,135.0,-90)	
N80 PO[X]=(270.0)	
N90 PO[X]=(315.0) PO[Y]=(0.0,-135.0,90)	
N100 PO[X]=(360.0)	
N110 CTABEND	; Koniec definicji.
;Test krzywej przez sprzężenie Y z X:	
N120 G1 F1000 X0	
N130 LEADON(Y,X,2)	
N140 X360	
N150 X0	
N160 LEADOF(Y,X)	
N170 DEPPPOS=CTAB(75.0,2,GRADIENT)	; Odczyt funkcji tablicy przy wartości wiodącej 75.0.
N180 G0 X75 Y=DEPPPOS	; Pozycjonowanie osi wiodącej i nadążnej.
;Po włączeniu sprzężenia nie jest konieczna synchronizacja osi nadążnej.	
N190 LEADON(Y,X,2)	
N200 G1 X110 F1000	
N210 LEADOF(Y,X)	
N220 M30	

Dalsze informacje**Wartość startowa i wartość końcowa tablicy krzywych**

Wartością startową początku obszaru definicji tablicy krzywych jest pierwsze podanie przynależnych do siebie pozycji osi (pierwsza instrukcja ruchu) w ramach definicji tablicy krzywych. Wartość końcowa obszaru definicji tablicy krzywych jest odpowiednio określana przez ostatnie polecenie ruchu.

Dostępny zakres językowy

W ramach definicji tablicy krzywych jest do dyspozycji cały zakres języka NC.

Wskazówka

Następujące dane są niedopuszczalne w definicjach tablic krzywych:

- Zatrzymanie przebiegu wyprzedzającego
 - Skoki w ruchu osi wiodącej (np. przy zmianie transformacji)
 - instrukcja ruchu tylko dla osi nadążnej
 - Odwrócenie kierunku ruchu osi wiodącej, tzn. pozycja osi wiodącej musi być zawsze jednoznaczna
 - Instrukcja CTABDEF i CTABEND na różnych poziomach programu.
-

Działanie instrukcji modalnych

Wszystkie modalnie działające instrukcje, które nastąpią w ramach definicji tablicy krzywych, nie obowiązują z zakończeniem definicji tablicy krzywych. Program obróbki, w którym następuje definicja tablicy, znajduje się przez to w takim samym stanie przed i po definicji.

Przyporządkowania do parametrów R

Przyporządkowania do parametrów R w ramach definicji tablicy są cofane po .CTABEND.

Przykład:

Kod programu	Komentarz
...	
R10=5 R11=20	; R10=5
...	
CTABDEF	
G1 X=10 Y=20 F1000	
R10=R11+5	; R10=25
X=R10	
CTABEND	
...	; R10=5

Uaktywnienie ASPLINE, BSPLINE, CSPLINE

Jeżeli w ramach definicji tablicy krzywych CTABDEF ... CTABEND nastąpi uaktywnienie ASPLINE, BSPLINE lub CSPLINE, wówczas przed tym uaktywnieniem spline powinien zostać zaprogramowany co najmniej jeden punkt startowy. Natychmiastowego uaktywnienia po CTABDEF należałoby unikać, ponieważ wówczas spline zależy od aktualnej pozycji osi przed definicją tablicy krzywych.

Przykład:

Kod programu
...
CTABDEF (Y, X, 1, 0)
X0 Y0

Kod programu

```
ASPLINE
X=5 Y=10
X10 Y40
...
CTABEND
```

Powtórne zastosowanie tablic krzywych

Zależność funkcjonalna osi wiodącej i nadążnej, obliczona poprzez tablicę krzywych, pozostaje zachowana pod wybranym numerem tablicy po zakończeniu programu i po POWER OFF, w przypadku gdy tablica jest zapisana w statycznej pamięci NC (SRAM).

Tablica, która została zapisana w pamięci dynamicznej (DRAM), jest przy POWER ON kasowana i musi ewentualnie zostać utworzona jeszcze raz.

Raz sporządzona tablica krzywych daje się zastosować do dowolnych kombinacji osi wiodącej i nadążnej i jest niezależna od tego, których osi użyto do sporządzenia tablicy.

Zastąpienie tablicy krzywych

Tablica krzywych jest zastępowana, gdy tylko przy ponownej definicji tablicy zostanie użyty jej numer.

Wyjątek: Tablica krzywych jest aktywna w sprzężeniu osi albo zablokowana przy pomocy CTABLOCK.

Wskazówka

Przy zastępowaniu tablic krzywych nie jest wyprowadzane odpowiednie ostrzeżenie!

Czy definicja tablicy krzywych jest aktywna?

Przy pomocy zmiennej systemowej \$P_CTABDEF można z programu obróbki w każdym czasie odczytać, czy definicja tablicy krzywych jest aktywna.

Zniesienie definicji tablicy krzywych

Segment programu obróbki po wyłączeniu instrukcji do definicji tablic krzywych przed nawias może być ponownie stosowany jako realny program obróbki.

Ładowanie tablic krzywych przez "wykonywanie ze źródła zewnętrznego"

Przy wykonywaniu tablic krzywych ze źródła zewnętrznego wielkość bufora doładowywania (DRAM) musi zostać tak dobrana przez MD18360 \$MN_MM_EXT_PROG_BUFFER_SIZE , by cała definicja tablicy mogła zostać zapisana w buforze. Wykonywanie programu obróbki zostanie w przeciwnym przypadku przerwane z alarmem 15150.

Skoki osi nadążnej

Zależnie od ustawienia danej maszynowej:

MD20900 \$MC_CTAB_ENABLE_NO_LEADMOTION

mogą być tolerowane skoki osi nadążnej przy braku ruchów w osi wiodącej.

11.2.2 Sprawdzenie istnienia pliku tablicy krzywych (CTABEXISTS)

Funkcja

Przy pomocy polecenia `CTABEXISTS` można sprawdzić, czy określony numer tablicy krzywych istnieje w pamięci NC.

Składnia

```
CTABEXISTS (<n>)
```

Znaczenie

<code>CTABEXISTS</code>	Sprawdza, czy tablica krzywych o numerze <code><n></code> występuje w statycznej albo dynamicznej pamięci NC
0	Tablica nie istnieje
1	Tablica istnieje
<code><n></code>	Numer (ID) tablicy krzywych

11.2.3 Kasowanie tablic krzywych (CTABDEL)

Funkcja

Przy pomocy `CTABDEL` można kasować tablice krzywych.

Wskazówka

Tablice krzywych, które są aktywne w sprzężeniu osi, nie mogą zostać skasowane.

Składnia

```
CTABDEL (<n>)  
CTABDEL (<n>, <m>)  
CTABDEL (<n>, <m>, <miejsce zapisania>)  
CTABDEL ()  
CTABDEL (, , <miejsce zapisania>)
```

Znaczenie

CTABDEL	Polecenie do kasowania tablic krzywych
<n>	Numer (ID) kasowanej tablicy krzywych
	Przy kasowaniu zakresu tablic krzywych CTABDEL(<n>, <m>) następuje przez <n> podanie numeru pierwszej tablicy zakresu.
<m>	Przy kasowaniu zakresu tablic krzywych CTABDEL(<n>, <m>) następuje przez <m> podanie numeru ostatniej tablicy zakresu. <m> musi być większe od <n>!
<miejsce zapisania>	Podanie miejsca zapisania (opcjonalne) Przy kasowaniu bez podania miejsca zapisania są kasowane podane tablice krzywych w pamięci statycznej i dynamicznej. Przy kasowaniu z podaniem miejsca zapisania są z podanych tablic krzywych kasowane te, które znajdują się w podanej pamięci. Pozostałe pozostają. "SRAM" Skasowanie w stacyjnej pamięci NC "DRAM" Skasowanie w dynamicznej pamięci NC

Gdy CTABDEL zostanie zaprogramowane bez podania kasowanej tablicy krzywych, wówczas zostaną skasowane **wszystkie** tablice krzywych wzgl. tablice krzywych w podanej pamięci:

CTABDEL()	Kasuje wszystkie tablice krzywych w statycznej i dynamicznej pamięci NC
CTABDEL(, , "SRAM")	Kasuje wszystkie tablice krzywych w statycznej pamięci NC
CTABDEL(, , "DRAM")	Kasuje wszystkie tablice krzywych w dynamicznej pamięci NC

Wskazówka

Gdy przy kasowaniu wielokrotnym CTABDEL(<n>, <m>) lub CTABDEL() co najmniej jedna z kasowanych tablic krzywych jest aktywna w sprzężeniu, wówczas polecenie skasowania nie jest wykonywane, tzn. **żadna** z zaadresowanych tablic nie zostanie skasowana.

11.2.4 Zablokowanie tablic krzywych przed skasowaniem i zastąpieniem (CTABLOCK, CTABUNLOCK)

Funkcja

Tablice krzywych mogą przez ustawienie blokad być chronione przed nie zamierzonym skasowaniem i zastąpieniem. Ustawioną blokadę można w każdym czasie wyłączyć.

Składnia

Ustawienie blokady:

```
CTABLOCK (<n>)
CTABLOCK (<n>, <m>)
CTABLOCK (<n>, <m>, <miejsce zapisania>)
CTABLOCK ()
CTABLOCK (, , <miejsce zapisania>)
```

Wyłączenie blokady:

```
CTABUNLOCK (<n>)
CTABUNLOCK (<n>, <m>)
CTABUNLOCK (<n>, <m>, <miejsce zapisania>)
CTABUNLOCK ()
CTABUNLOCK (, , <miejsce zapisania>)
```

Znaczenie

CTABLOCK	Polecenie do ustawienia blokady przed skasowaniem/zastąpieniem
CTABUNLOCK	Polecenie do wyłączenia blokady przed skasowaniem/zastąpieniem
	CTABUNLOCK zwalnia tablice krzywych zablokowane przy pomocy CTABLOCK. Tablice krzywych, które działają w aktywnym sprzężeniu, pozostają nadal zablokowane i nie mogą zostać skasowane. Blokada przy pomocy CTABLOCK zostanie wyłączona, gdy zostanie zniesiona blokada przez aktywne sprzężenie z wyłączeniem aktywności sprzężenia. Przez to ta tablica może zostać skasowana. Ponowne wywołanie CTABUNLOCK nie jest konieczne.
<n>	Numer (ID) blokowanej/zwalnianej tablicy krzywych Przy blokowaniu/zwalnianiu zakresu tablic krzywych CTABLOCK (<n>, <m>)/CTABUNLOCK (<n>, <m>) następuje przez <n> podanie numeru pierwszej tablicy zakresu.
<m>	Przy blokowaniu/zwalnianiu zakresu tablic krzywych CTABLOCK (<n>, <m>)/CTABUNLOCK (<n>, <m>) następuje przez <m> podanie numeru ostatniej tablicy zakresu. <m> musi być większe od <n>!

<miejsce zapisania>	Podanie miejsca zapisania (opcjonalne) Przy ustawieniu/wyłączeniu blokady bez podania miejsca zapisania są blokowane/zwalniane podane tablice krzywych w pamięci statycznej i dynamicznej. Przy ustawieniu/wyłączeniu blokady z podaniem miejsca zapisania są z podanych tablic krzywych blokowane/zwalniane te, które znajdują się w podanej pamięci. Pozostałe nie są blokowane/zwalniane. "SRAM" Ustawienie/wyłączenie blokady w stacyjnej pamięci NC "DRAM" Ustawienie/wyłączenie blokady w dynamicznej pamięci NC
---------------------	--

Gdy `CTABLOCK/CTABUNLOCK` zostanie zaprogramowane bez podania blokowanej/zwalnianej tablicy krzywych, wówczas zostaną zablokowane/zwolnione **wszystkie** tablice krzywych wzgl. tablice krzywych w podanej pamięci:

<code>CTABLOCK()</code>	Blokuje wszystkie tablice krzywych w statycznej i dynamicznej pamięci NC
<code>CTABLOCK(, , "SRAM")</code>	Blokuje wszystkie tablice krzywych w statycznej pamięci NC
<code>CTABLOCK(, , "DRAM")</code>	Blokuje wszystkie tablice krzywych w dynamicznej pamięci NC
<code>CTABUNLOCK()</code>	Odblokowuje wszystkie tablice krzywych w statycznej i dynamicznej pamięci NC
<code>CTABUNLOCK(, , "SRAM")</code>	Odblokowuje wszystkie tablice krzywych w statycznej pamięci NC
<code>CTABUNLOCK(, , "DRAM")</code>	Odblokowuje wszystkie tablice krzywych w dynamicznej pamięci NC

11.2.5 Tablice krzywych: określenie właściwości tablicy (CTABID, CTABISLOCK, CTABMENTYP, CTABPERIOD)

Funkcja

Przy pomocy tych poleceń mogą być odpytywane ważne właściwości tablicy krzywych (numer tablicy, stan zablokowania, miejsce zapisania, okresowość).

Składnia

```
CTABID(<p>)
CTABID(<p>,<miejsce zapisania>)
CTABISLOCK(<n>)
CTABMENTYP(<n>)
TABPERIOD(<n>)
```

Znaczenie

CTABID	Daje numer tablicy , który w podanej pamięci jest wpisany jako <p>-ta tablica krzywych. Przykład: CTABID(1, "SRAM") daje numer pierwszej tablicy krzywych w statycznej pamięci NC. Pierwsza tablica krzywych odpowiada przy tym tablicy krzywych posiadającej najwyższy numer. Wskazówka: Jeżeli między kolejnymi wywołaniami CTABID zostanie zmieniona kolejność tablic krzywych w pamięci, np. przez skasowanie tablic krzywych przez CTABDEL, CTABID(<p>, ...) o tym samym numerze <p> może dać inną tablicę krzywych, niż poprzednio.
CTABISLOCK	Zwraca stan zablokowania tablicy krzywych o numerze <n>: 0 Tablica nie jest zablokowana 1 Tablica jest zablokowana przez CTABLOCK 2 Tablica jest zablokowana przez aktywne sprzężenie 3 Tablica jest zablokowana przez CTABLOCK i aktywne sprzężenie -1 Tablica nie istnieje
CTABMEMTYP	Daje miejsce zapisania tablicy krzywych o numerze <n>: 0 Skasowanie w statycznej pamięci NC 1 Tablica w dynamicznej pamięci NC -1 Tablica nie istnieje
CTABPERIOD	Daje okresowość tablicy krzywych o numerze <n>: 0 Tablica jest nieperiodyczna 1 Tablica jest periodyczna w osi wiodącej 2 Tablica jest periodyczna w osi wiodącej i nadążnej -1 Tablica nie istnieje
<p>	Numer wpisu w pamięci
<n>	Numer (ID) tablicy krzywych
<miejsce zapisania>	Podanie miejsca zapisania (opcjonalne) "SRAM" statyczna pamięć NC "DRAM" dynamiczna pamięć NC Wskazówka: Gdy dla tego parametru nie zostanie zaprogramowana żadna wartość, wówczas zostanie użyte standardowe miejsce w pamięci, ustawione przy pomocy MD20905 \$MC_CTAB_DEFAULT_MEMORY_TYPE.

11.2.6 Odczyt wartości tablicy krzywych (CTABTSV, CTABTEV, CTABTSP, CTABTEP, CTABSSV, CTABSEV, CTAB, CTABINV, CTABTMIN, CTABTMAX)

Funkcja

Następujące wartości tablicy krzywych można czytać w programie obróbki:

- Wartości osi nadążnej i osi wiodącej na początku i na końcu tablicy krzywych
- Wartości osi nadążnej na początku i na końcu segmentu krzywej
- Wartość osi nadążnej do wartości osi wiodącej
- Wartość osi wiodącej do wartości osi nadążnej
- Minimalna i maksymalna wartość osi nadążnej
 - w całym zakresie definicji tablicy krzywych
 - lub
 - w zdefiniowanym przedziale tablicy krzywych

Składnia

```
CTABTSV(<n>, <gradient> [, <os nadążna>])
CTABTEV(<n>, <gradient> [, <os nadążna>])
CTABTSP(<n>, <gradient> [, <os wiodąca>])
CTABTEP(<n>, <gradient> [, <os wiodąca>])
CTABSSV(<wartość wiodąca>, <n>, <gradient> [, <os nadążna>])
CTABSEV(<wartość wiodąca>, <n>, <gradient> [, <os nadążna>])
CTAB(<wartość wiodąca>, <n>, <gradient> [, <os nadążna>, <os wiodąca>])
CTABINV(<wartość nadążna>, <wartość przybliżona>, <n>, <gradient> [, <os nadążna>, <os wiodąca>])
CTABTMIN(<n> [, <os nadążna>])
CTABTMAX(<n> [, <os nadążna>])
CTABTMIN(<n>, <a>, <b> [, <os nadążna>, <os wiodąca>])
CTABTMAX(<n>, <a>, <b> [, <os nadążna>, <os wiodąca>])
```

Znaczenie

CTABTSV:	Odczyt wartości osi nadążnej na początku tablicy krzywych nr <n>
CTABTEV:	Odczyt wartości osi nadążnej na końcu tablicy krzywych nr <n>
CTABTSP:	Odczyt wartości osi wiodącej na początku tablicy krzywych nr <n>
CTABTEP:	Odczyt wartości osi wiodącej na końcu tablicy krzywych nr <n>
CTABSSV:	Odczyt wartości osi nadążnej na początku segmentu krzywej należącego do podanej wartości osi wiodącej (<wartość wiodąca>)
CTABSEV:	Odczyt wartości osi nadążnej na końcu segmentu krzywej należącego do podanej wartości osi wiodącej (<wartość wiodąca>)
CTAB:	Odczyt wartości osi nadążnej do podanej wartości osi wiodącej (<wartość wiodąca>)
CTABINV:	Odczyt wartości osi wiodącej do podanej wartości osi nadążnej (<wartość nadążna>)

CTABTMIN:	Określenie minimalnej wartości osi nadążnej: <ul style="list-style-type: none"> w całym zakresie definicji tablicy krzywych lub w zdefiniowanym przedziale <code><a> ... </code>
CTABTMAX:	Określenie maksymalnej wartości osi nadążnej: <ul style="list-style-type: none"> w całym zakresie definicji tablicy krzywych lub w zdefiniowanym przedziale <code><a> ... </code>
<n>:	Numer (ID) tablicy krzywych
<gradient>:	W parametrze <code><gradient></code> jest zwracane nachylenie funkcji tablicy krzywych w określonej pozycji
<oś nadążna>:	Oś, której ruch ma zostać obliczony poprzez tablicę krzywych (opcjonalnie)
<oś wiodąca>:	Oś, która daje wartości wiodące do obliczania ruchu osi nadążnej (opcjonalnie)
<wartość nadążna>:	Wartość osi nadążnej do odczytu przynależnej wartości osi wiodącej CTABINV
<wartość wiodąca>:	Wartość osi wiodącej: <ul style="list-style-type: none"> do odczytu przynależnej wartości osi nadążnej przy CTAB lub do wyboru segmentu krzywej przy CTABSSV/CTABSEV
<wartość przybliżona>:	Przyporządkowanie wartości osi wiodącej do wartości osi nadążnej przy CTABINV nie zawsze musi być jednoznaczne. CTABINV potrzebuje dlatego jako parametru wartości przybliżonej oczekiwanej wartości osi wiodącej.
<a>:	Dolna granica przedziału wartości wiodącej przy CTABTMIN/CTABTMAX
:	Górna granica przedziału wartości wiodącej przy CTABTMIN/CTABTMAX
	Hinweis: Interwał wartości wiodącej <code><a> ... </code> musi znajdować się w obszarze definicji tablicy krzywych.

Przykłady

Przykład 1:

Określenie wartości osi nadążnej i osi wiodącej na początku i na końcu tablicy krzywych, jak też minimalnej i maksymalnej wartości osi nadążnej w całym zakresie definicji tablicy krzywych.

Kod programu	Komentarz
N10 DEF REAL STARTPOS	
N20 DEF REAL ENDPOS	
N30 DEF REAL STARTPARA	

Kod programu	Komentarz
N40 DEF REAL ENDPARA	
N50 DEF REAL MINVAL	
N60 DEF REAL MAXVAL	
N70 DEF REAL GRADIENT	
...	
N100 CTABDEF(Y,X,1,0)	; Początek definicji tablicy
N110 X0 Y10	; Pozycja startowa 1. segment tablicy
N120 X30 Y40	; Pozycja końcowa 1. segment tablicy = pozycja startowa 2. segment tablicy
N130 X60 Y5	; Pozycja końcowa 2. segment tablicy = ...
N140 X70 Y30	
N150 X80 Y20	
N160 CTABEND	; Koniec definicji tablicy.
...	
N200 STARTPOS=CTABTSV(1,GRADIENT)	; Wartość osi nadążnej na początku tablicy krzywych = 10
N210 ENDPOS=CTABTEV(1,GRADIENT)	; Wartość osi nadążnej na końcu tablicy krzywych = 20
N220 STARTPARA=CTABTSP(1,GRADIENT)	; Wartość osi wiodącej na początku tablicy krzywych = 0
N230 ENDPARA=CTABTEP(1,GRADIENT)	; Wartość osi wiodącej na końcu tablicy krzywych = 80
N240 MINVAL=CTABTMIN(1)	; Wartość minimalna osi nadążnej przy Y=5
N250 MAXVAL=CTABTMAX(1)	; Wartość maksymalna osi nadążnej przy Y=40

Przykład 2:

Określenie wartości osi nadążnej na początku i na końcu segmentu krzywej należącego do wartości osi wiodącej X=30.

Kod programu	Komentarz
N10 DEF REAL STARTPOS	
N20 DEF REAL ENDPOS	
N30 DEF REAL GRADIENT	
...	
N100 CTABDEF(Y,X,1,0)	; Początek definicji tablicy.
N110 X0 Y0	; Pozycja startowa 1. segment tablicy
N120 X20 Y10	; Pozycja końcowa 1. segment tablicy = pozycja startowa 2. segment tablicy
N130 X40 Y40	; Pozycja końcowa 2. segment tablicy = ...
N140 X60 Y10	
N150 X80 Y0	
N160 CTABEND	; Koniec definicji tablicy.
...	
N200 STARTPOS=CTABSSV(30.0,1,GRADIENT)	; Pozycja startowa Y w 2. segmencie = 10
N210 ENDPOS=CTABSEV(30.0,1,GRADIENT)	; Pozycja końcowa Y w 2. segmencie = 40

Dalsze informacje

Zastosowanie w akcjach synchronicznych

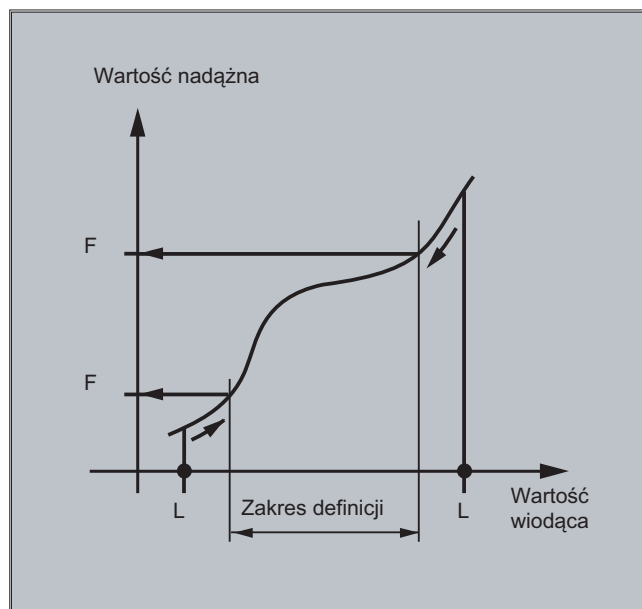
Wszystkie polecenia do odczytu wartości tablic krzywych mogą być stosowane również w akcjach synchronicznych (patrz też punkt "akcje synchroniczne ruchu").

Przy stosowaniu poleceń CTABINV, CTABTMIN i CTABTMAX należy zwracać uwagę, by:

- w chwili wykonania była dostępna wystarczająca moc NC
lub
- przed wywołaniem została odpytana liczba segmentów tablicy krzywych, aby ewentualnie moc podzielić odnośną tablicę

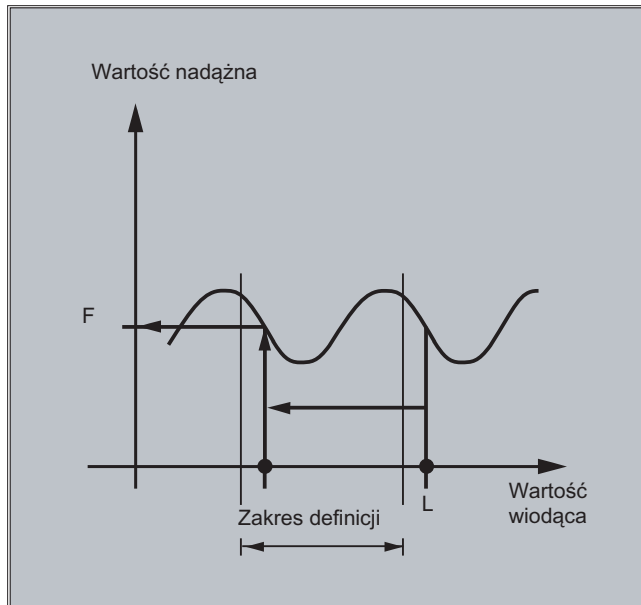
CTAB przy nieperiodycznych tablicach krzywych

Gdy podana <wartość wiodąca> leży poza zakresem definicji, jest jako wartość nadążna wyprowadzana granica górna wzgl. dolna.



CTAB przy okresowych tablicach krzywych

Jeżeli podana <wartość wiodąca> leży poza zakresem definicji, jest interpretowana wartość wiodąca modulo zakresu definicji i wprowadzana odpowiednia wartość nadążna:



Wartość przybliżona CTABINV

Polecenie `CTABINV` potrzebuje przybliżonej wartości oczekiwanej wartości wiodącej. `CTABINV` zwraca wartość wiodącą, która jest najbliższa wartości przybliżonej. Wartością przybliżoną może być np. wartość wiodąca z poprzedniego taktu interpolacji.

Nachylenie funkcji tablicy krzywych

Wyprowadzenie nachylenia (<gradient>) umożliwia obliczenie prędkości osi wiodącej lub nadążnej w odpowiedniej pozycji.

Podanie osi wiodącej lub nadążnej

Opcjonalne podanie osi wiodącej i/albo nadążnej jest ważne, w przypadku gdy oś wiodąca i oś nadążna są zaprojektowane w różnych jednostkach długości.

CTABSSV, CTABSEV

Polecenia `CTABSSV` i `CTABSEV` w następujących przypadkach **nie** nadają się, do odpytania zaprogramowanych segmentów:

- Są zaprogramowane okręgi albo ewolwenty.
- Jest aktywne fazowanie wzgl. zaokrąglanie przy pomocy `CHF/RND`.
- Jest aktywne ścinanie narożników przy pomocy `G643`.
- Jest aktywna kompresja bloków NC przy pomocy `COMPON/COMP CURV/COMP CAD`.

11.2.7 Tablice krzywych: sprawdzić wykorzystanie zasobów (CTABNO, CTABNOMEM, CTABFNO, CTABSEGID, CTABSEG, CTABFSEG, CTABMSEG, CTABPOLID, CTABPOL, CTABFPOL, CTABMPOL)

Funkcja

Przy pomocy tych poleceń programista ma możliwość poinformowania się o aktualnym wykorzystaniu zasobów dla tablic krzywych, segmentów tablicy i wielomianów.

Składnia

```
CTABNO
CTABNOMEM(<miejsce zapisania>)
CTABFNO(<miejsce zapisania>)
CTABSEGID(<n>,<miejsce zapisania>)
CTABSEG(<miejsce zapisania>,<rodzaj segmentu>)
CTABFSEG(<miejsce zapisania>,<rodzaj segmentu>)
CTABMSEG(<miejsce zapisania>,<rodzaj segmentu>)
CTABPOLID(<n>)
CTABPOL(<miejsce zapisania>)
CTABFPOL(<miejsce zapisania>)
CTABMPOL(<miejsce zapisania>)
```

Znaczenie

CTABNO	Określenie liczby całkowitej zdefiniowanych tablic krzywych (w statycznej i dynamicznej pamięci NC)
CTABNOMEM	Określenie liczby zdefiniowanych tablic krzywych w podanym <miejscu zapisania>
CTABFNO	Określenie liczby jeszcze możliwych tablic krzywych w podanym <miejscu zapisania>
CTABSEGID	Określenie liczby segmentów krzywej o podanym <rodzaju segmentu>, które są stosowane przez tablicę krzywych o numerze <n>
CTABSEG	Określenie liczby zastosowanych segmentów krzywej o podanym <rodzaju segmentu> w podanym <miejscu zapisania>
CTABFSEG	Określenie liczby jeszcze możliwych segmentów krzywej o podanym <rodzaju segmentu> w podanym <miejscu zapisania>
CTABMSEG	Określenie liczby maksymalnie możliwych segmentów krzywej o podanym <rodzaju segmentu> w podanym <miejscu zapisania>
CTABPOLID	Określenie liczby wielomianów krzywej, które są stosowane przez tablicę krzywych o numerze <n>
CTABPOL	Określenie liczby zastosowanych wielomianów krzywych w podanym <miejscu zapisania>
CTABFPOL	Określenie liczby jeszcze możliwych wielomianów krzywych w podanym <miejscu zapisania>
CTABMPOL	Określenie liczby maksymalnie możliwych wielomianów krzywych w podanym <miejscu zapisania>
<n>	Numer (ID) tablicy krzywych

<miejsce zapisania> Podanie miejsca zapisania (opcjonalne)
"SRAM" **stacyczna** pamięć NC
"DRAM" **dynamiczna** pamięć NC

Wskazówka:
Gdy dla tego parametru nie zostanie zaprogramowana żadna wartość, wówczas zostanie użyte standardowe miejsce w pamięci, ustawione przy pomocy MD20905 \$MC_CTAB_DEFAULT_MEMORY_TYPE.

<rodzaj segmentu> Podanie rodzaju segmentu (opcjonalne)
"L" Segmenty liniowe
"P" Segmenty wielomianowe

Wskazówka:
Jeżeli dla tego parametru nie zostanie zaprogramowana żadna wartość, wówczas zostanie wyprowadzona suma fragmentów liniowych i wielomianowych.

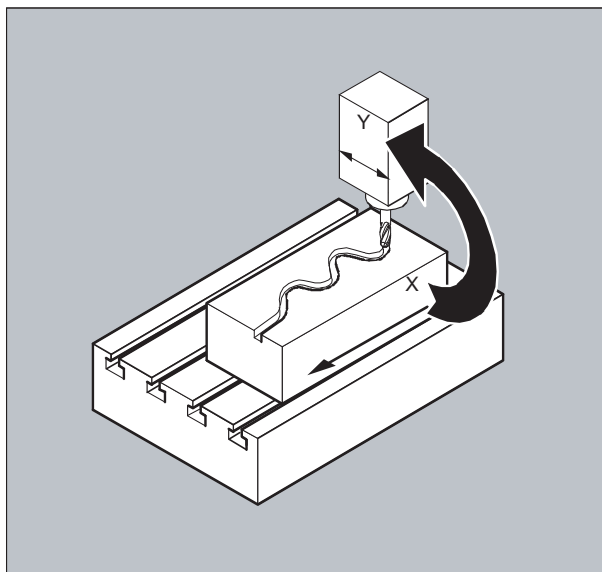
11.3 Osiowe sprzężenie wartości wiodącej (LEADON, LEADOF)

Wskazówka

Ta funkcja nie jest dostępna dla SINUMERIK 828D!

Funkcja

W przypadku osiowego sprzężenia wartości wiodącej oś wiodąca i oś nadążna wykonują ruch synchroniczny. Przy tym każdorazowa pozycja osi nadążnej jest przez tablicę krzywej wzgl. obliczony z niej wielomian jednoznacznie przyporządkowana do - ew. symulowanej - pozycji osi wiodącej.



Oś wiodącą nazywa się ta oś, która daje wartości wejściowe dla tablicy krzywych. **Oś nadążną** nazywa się ta oś, która przyjmuje pozycje obliczone poprzez tablicę krzywej.

Sprzężenie wartości rzeczywistej i zadanej

Jako wartości wiodące, a więc wartości wyjściowe do określenia pozycji osi nadążnej mogą być stosowane:

- Wartości rzeczywiste pozycji osi wiodącej: Sprzężenie wartości rzeczywistej
- Wartości zadane pozycji osi wiodącej: Sprzężenie wartości zadanej

Sprzężenie wartości wiodącej obowiązuje zawsze w bazowym układzie współrzędnych.

Odnośnie sporządzania tablic krzywych patrz rozdział "Tablice krzywych".

Składnia

LEADON(<oś nadążna>, <oś wiodąca>, <n>)

LEADOF(<oś nadążna>, <oś wiodąca>)

albo wyłączenie bez podania osi wiodącej:

LEADOF(oś nadążna)

Sprzężenie wartości wiodącej może zostać włączone i wyłączone zarówno z programu obróbki, jak i podczas ruchu z akcji synchronicznych.

Znaczenie

LEADON:	Włączenie sprzężenia wartości wiodącej
LEADOF:	Wyłączenie sprzężenia wartości wiodącej
<oś nadążna>:	Oś nadążna
<oś wiodąca>:	Oś wiodąca
<n>:	Numer tablicy krzywych
\$SA_LEAD_TYPE:	Przełączenie między sprzężeniem wartości zadanej i rzeczywistej

Wyłączenie sprzężenia wartości wiodącej, LEADOF

Z wyłączeniem sprzężenia wartości wiodącej oś nadążna staje się ponownie normalną osią rozkazową!

Osiowe sprzężenie wartości wiodącej i różne stany robocze, RESET

Zależnie od ustawienia w danej maszynowej sprzężenia wartości wiodącej są wyłączane przez RESET.

Przykład, sprzężenie wartości wiodącej z akcji synchronicznej

W przypadku prasy zwykle mechaniczne sprzężenie między osią wiodącą (wał stempla) i osiami systemu transportowego składającego się z osi transportowych i osi pomocniczych ma zostać zastąpione przez elektroniczny system sprzęgający.

Przykład pokazuje, jak w przypadku prasy mechaniczny system transportowy jest zastępowany przez system elektroniczny. Procesy sprzęgania i rozprzegania są realizowane jako **statyczne akcje synchroniczne**.

Od osi wiodącej LW (wał stempla) są sterowane osie transportowe i osie pomocnicze definiowane, jako osie nadążne poprzez tablice krzywych.

Osie nadążne

X oś posuwu wzgl. wzdłużna
 YL oś zamykania wzgl. poprzeczna
 ZL oś skoku
 U posuw wałków, oś pomocnicza
 V głowica prostująca, oś pomocnicza
 W smarowanie, oś pomocnicza

Akcje

Jako akcje występują w akcjach synchronicznych np.:

- Sprzężenie, LEADON(<oś nadążna>, <oś wiodąca>, <numer tablicy krzywych>)
- Rozłączenie sprzężenia, LEADOF(<oś nadążna>, <oś wiodąca>)
- Ustawienie wartości rzeczywistej, PRESETON(<oś>, <wartość>)
- Ustawienie znacznika, \$AC_MARKER[i]=<wartość>
- Rodzaj sprzężenia: rzeczywista/wirtualna wartość wiodąca
- Ruch do pozycji osi, POS [<oś>]=<wartość>

Warunki

Jako warunki podlegają ewaluacji szybkie wejścia cyfrowe, zmienne czasu rzeczywistego \$AC_MARKER i porównania pozycji, powiązane operatorem logicznym AND.

Wskazówka

W poniższym przykładzie zmiany wiersza, wcięcia i **tlusty** druk zastosowano wyłącznie w tym celu, by zwiększyć czytelność programowania. Dla sterowania wszystko, co jest pod jednym numerem wiersza, jest jednym wierszem.

Komentarz

Kod programu	Komentarz
	; Definiuje wszystkie statyczne akcje synchroniczne.
	; ****Cofnięcie znacznika
N2 \$AC_MARKER[0]=0 \$AC_MARKER[1]=0 \$AC_MARKER[2]=0 \$AC_MARKER[3]=0 \$AC_MARKER[4]=0 \$AC_MARKER[5]=0 \$AC_MARKER[6]=0 \$AC_MARKER[7]=0	
	; **** E1 0=>1 sprzężenie transport WŁ.
N10 IDS=1 EVERY (\$A_IN[1]==1) AND (\$A_IN[16]==1) AND (\$AC_MARKER[0]==0) DO LEADON(X,LW,1) LEADON(YL,LW,2) LEADON(ZL,LW,3) \$AC_MARKER[0]=1	
	; **** E1 0=>1 sprzężenie posuw wałków WŁ.
N20 IDS=11 EVERY (\$A_IN[1]==1) AND (\$A_IN[5]==0) AND (\$AC_MARKER[5]==0) DO LEADON(U,LW,4) PRESETON(U,0) \$AC_MARKER[5]=1	
	; **** E1 0->1 sprzężenie głowica prostująca WŁ.
N21 IDS=12 EVERY (\$A_IN[1]==1) AND (\$A_IN[5]==0) AND (\$AC_MARKER[6]==0) DO LEADON(V,LW,4) PRESETON(V,0) \$AC_MARKER[6]=1	

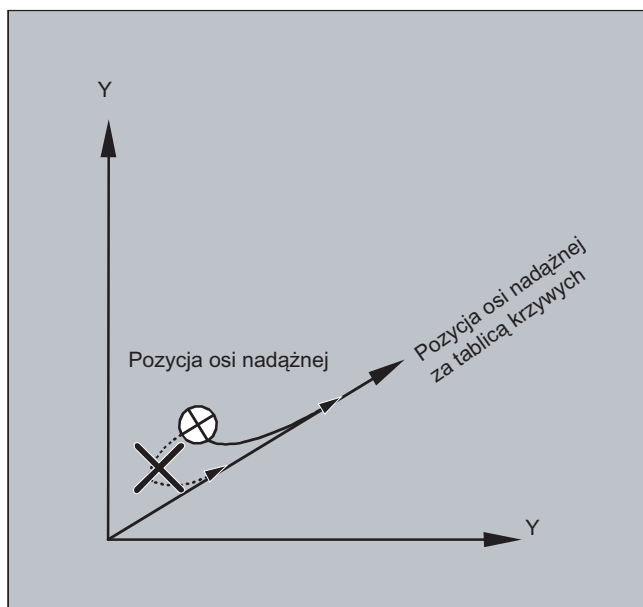
Kod programu	Komentarz
	; **** E1 0->1 sprzężenie smarowanie WŁ.
N22 IDS=13 EVERY (\$A_IN[1]==1) AND (\$A_IN[5]==0) AND (\$AC_MARKER[7]==0)	
DO LEADON(W,LW,4) PRESETON(W,0) \$AC_MARKER[7]=1	
	; **** E2 0->1 sprzężenie WYŁ.
N30 IDS=3 EVERY (\$A_IN[2]==1)	
DO LEADOF(X,LW) LEADOF(YL,LW) LEADOF(ZL,LW) LEADOF(U,LW) LEADOF(V,LW) LEADOF(W,LW) \$AC_MARKER[0]=0	
\$AC_MARKER[1]=0 \$AC_MARKER[3]=0 \$AC_MARKER[4]=0 \$AC_MARKER[5]=0 \$AC_MARKER[6]=0 \$AC_MARKER[7]=0	
....	
N110 G04 F01	
N120 M30	

Opis

Sprzężenie wartości wiodącej wymaga synchronizacji osi wiodącej i osi nadążnej. Tę synchronizację można uzyskać tylko wtedy, gdy oś nadążna przy włączeniu sprzężenia wartości wiodącej znajduje się w zakresie tolerancji przebiegu krzywej obliczonego z tablicy krzywych.

Zakres tolerancji pozycji osi nadążnej jest zdefiniowany przez daną maszynową MD 37200: COUPLE_POS_POL_COARSE A_LEAD_TYPE.

Jeżeli oś nadążna przy włączeniu sprzężenia wartości wiodącej nie znajduje się jeszcze w odpowiedniej pozycji, wówczas praca synchroniczna jest podejmowana automatycznie, gdy tylko obliczona wartość zadana pozycji osi nadążnej zbliży się do rzeczywistej pozycji tej osi. Oś nadążna wykonuje przy tym podczas procesu synchronizacji ruch w kierunku, który jest zdefiniowany przez prędkość zadaną osi nadążnej (obliczona z prędkości osi wiodącej i według tablicy krzywych CTAB).

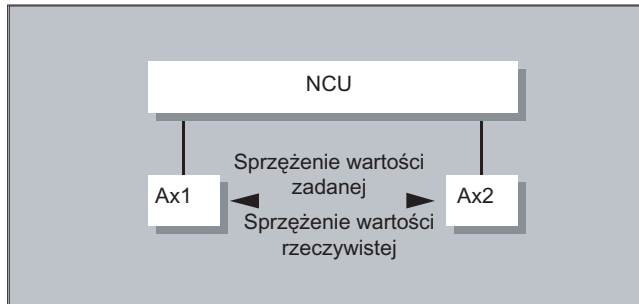


Bez pracy synchronicznej

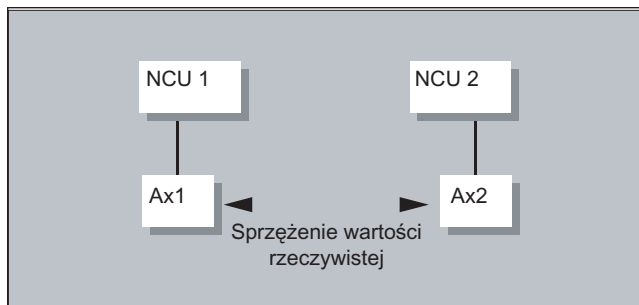
Jeżeli obliczona pozycja osi nadążnej oddala się z włączeniem sprzężenia wartości wiodącej od aktualnej pozycji osi nadążnej, praca synchroniczna nie jest podejmowana.

Sprzężenie wartości rzeczywistej i zadanej

Sprzężenie wartości zadanej daje w porównaniu ze sprzężeniem wartości rzeczywistej polepszony przebieg synchronizacji między osią wiodącą i osią nadążną i dlatego jest standardowo ustawione jako domyślne.



Sprzężenie wartości zadanej jest możliwe tylko wtedy, gdy oś wiodąca i nadążna są interpolowane przez tą samą NCU. W przypadku zewnętrznej osi wiodącej oś nadążna może zostać sprzężona z osią wiodącą tylko przez wartości rzeczywiste.



Przełączenie jest możliwe przez daną ustawczą \$SA_LEAD_TYPE.

Przełączanie między sprzężeniem wartości rzeczywistej i zadanej powinno następować zawsze przy zatrzymanej osi nadążnej. Ponieważ tylko na postoju następuje po przełączeniu ponowna synchronizacja.

Przykład zastosowania

Czytanie wartości rzeczywistych może przy dużych wstrząsach maszyny następować z błędami. Przy zastosowaniu sprzężenia wartości wiodącej na linii pras może dlatego w operacjach roboczych o większych wstrząsach być konieczne przełączenie ze sprzężenia wartości rzeczywistej na sprzężenie wartości zadanej.

Symulacja wartości wiodącej przy sprzężeniu wartości zadanej

Przez daną maszynową interpolator osi wiodącej daje się oddzielić od serwonapędu. Przez to przy sprzężeniu wartości zadanej wartości zadane mogą być wytwarzane bez rzeczywistego ruchu osi wiodącej.

Wartości wiodące wytwarzane przez sprzężenie wartości zadanej są w celu użycia np. w akcjach synchronicznych możliwe do odczytania z następujących zmiennych:

- | | |
|---------------|---------------------------|
| - \$AA_LEAD_P | Wartość wiodąca, pozycja |
| - \$AA_LEAD_V | Wartość wiodąca, prędkość |

Utworzenie wartości wiodących

Wartości wiodące mogą być wytwarzane do wyboru przy pomocy innych samemu programowanych metod. Tak wytworzone wartości wiodące są zapisywane do zmiennych

- \$AA_LEAD_SP Wartość wiodąca, pozycja
- \$AA_LEAD_SV Wartość wiodąca, prędkość

i z nich czytane. W celu użycia tych zmiennych musi być ustawiona dana ustawcza \$SA_LEAD_TYPE = 2.

Status sprzężenia

W programie obróbki NC można odpytać na status sprzężenia przy pomocy następującej zmiennej systemowej:

```
$AA_COUP_ACT[os]
```

0: Nie jest aktywne żadne sprzężenie
16: Sprzężenie wartości wiodącej aktywne

Zarządzanie statusem w przypadku akcji synchronicznych

Procesy łączenia i sprzęgania są zarządzane poprzez zmienne czasu rzeczywistego:

```
$AC_MARKER[i] = n
```

gdzie:
i numer znacznika
n wartość statusu

11.4 Przekładnia elektroniczna (EG)

Funkcja

Przy pomocy funkcji "przekładnia elektroniczna" jest możliwe sterowanie ruchem **osi nadążnej** po liniowym bloku ruchu w zależności od maksymalnie pięciu **osi wiodących**. Zależności między osiami wiodącymi i osią nadążną są dla każdej osi wiodącej definiowane poprzez współczynnik sprzężenia.

Obliczona składowa ruchu osi nadążnej jest tworzona przez dodawanie z poszczególnych składowych ruchu osi wiodących pomnożonych przez każdorazowy współczynnik sprzężenia. Przy uaktywnieniu zespołu osi EG można spowodować synchronizację osi nadążnej na zdefiniowaną pozycję. Zespół przekładni można z programu obróbki:

- zdefiniować,
- włączyć,
- wyłączyć,
- skasować

.

Ruch osi nadążnej można do wyboru wyprowadzić z

- wartości zadanych osi wiodących, jak też
- wartości rzeczywistych osi wiodących.

Jako rozszerzenie mogą być również realizowane nieliniowe zależności między osiami wiodącymi i osią nadążną poprzez **tablice krzywych** (patrz punkt "Zachowanie się w ruchu po torze"). Przekładnie elektroniczne mogą być kaskadowane, tzn. oś nadążna przekładni elektronicznej może być osią wiodącą dla kolejnej przekładni elektronicznej.

11.4.1 Zdefiniowanie przekładni elektronicznej (EGDEF)

Funkcja

Zespół osi przekładni elektronicznej jest ustalany przez podanie osi nadążnej i co najmniej jednej, ale co najwyżej pięciu osi wiodących z każdorazowym typem sprzężenia.

Warunek

Warunek dla definicji zespołu osi przekładni elektronicznej:

Dla osi nadążnej nie może być jeszcze zdefiniowane żadne sprzężenie osi (wzgl. istniejące musi zostać przedtem skasowane przy pomocy EGDEL).

Składnia

EGDEF(oś nadążna, oś wiodąca1, typ sprzężenia1, oś wiodąca , typ sprzężenia2, ...)

Znaczenie

EGDEF

Oś nadążna

Oś wiodąca1

Oś wiodąca5

Typ sprzężenia1

Typ sprzężenia5

Definicja przekładni elektronicznej

Oś, na którą wpływają osie wiodące

Osie, które wpływają na oś nadążną

Typ sprzężenia

Typ sprzężenia nie musi być taki sam dla wszystkich osi wiodących i dlatego dla każdej osi wiodącej należy go podać oddzielnie.

Wartość: Znaczenie:

0 Na oś nadążną wpływa **wartość rzeczywista** odpowiedniej osi wiodącej.

1 Na oś nadążną wpływa **wartość zadana** odpowiedniej osi wiodącej.

Wskazówka

Współczynniki sprzężenia są przy definicji zespołu sprzężenia przekładni elektronicznej domyślnie ustawione na zero.

Wskazówka

`EGDEF` wyzwała zatrzymanie przebiegu wyprzedzającego. Definicję przekładni przy pomocy `EGDEF` należy również wtedy stosować bez zmian, gdy w przypadku systemów jedna lub wiele osi wiodących wpływa poprzez **tablicę krzywych** na oś nadążną.

Przykład

Kod programu	Komentarz
<code>EGDEF(C,B,1,Z,1,Y,1)</code>	; Definicja zespołu osi przekładni elektronicznej. Osie wiodące B, Z, Y wpływają na oś nadążną C poprzez wartość zadaną.

11.4.2 Włączenie przekładni elektronicznej (EGON, EGONSYN, EGONSYNE)**Funkcja**

Są trzy warianty włączenia zespołu osi przekładni elektronicznej.

Składnia**Wariant 1:**

Zespół osi przekładni elektronicznej jest włączany bez synchronizacji selektywnie przy pomocy:

```
EGON(FA, "tryb zmiany bloku", LA1, Z1, N1, LA2, Z2, N2, . . . , LA5, Z5, N5)
```

Wariant 2:

Zespół osi przekładni elektronicznej jest włączany z synchronizacją selektywnie przy pomocy:

```
EGONSYN(FA, "tryb zmiany bloku", SynPosFA, [, LAi, SynPosLAI, Zi, Ni])
```

Wariant 3:

Zespół osi przekładni elektronicznej jest z synchronizacją selektywnie włączany, a tryb dosunięcia zadawany przy pomocy:

```
EGONSYNE(FA, "tryb zmiany bloku", SynPosFA, tryb  
dosunięcia [, LAi, SynPosLAI, Zi, Ni])
```

Znaczenie

Wariant 1:

FA	Oś nadążna
Tryb zmiany bloku	Mogą być używane następujące tryby:
	"NOC" Zmiana bloku następuje natychmiast
	"FINE" Zmiana bloku następuje przy "ruchu synchronicznym dokładnie"
	"COARSE" Zmiana bloku następuje przy "ruchu synchronicznym zgrubnie"
	"IPOSTOP" Zmiana bloku następuje przy pracy synchronicznej po stronie wartości zadanej
LA1, ... LA5	Osie wiodące
Z1, ... Z5	Licznik dla współczynnika sprzężenia i
N1, ... N5	Mianownik dla współczynnika sprzężenia i
	Współczynnik sprzężenia i = licznik i / mianownik i

Wolno programować tylko osie wiodące, które przedtem zostały wyspecyfikowane przy pomocy EGDEF. Musi zostać zaprogramowana co najmniej jedna oś wiodąca.

Wariant 2:

FA	Oś nadążna
Tryb zmiany bloku	Mogą być używane następujące tryby:
	"NOC" Zmiana bloku następuje natychmiast
	"FINE" Zmiana bloku następuje przy "ruchu synchronicznym dokładnie"
	"COARSE" Zmiana bloku następuje przy "ruchu synchronicznym zgrubnie"
	"IPOSTOP" Zmiana bloku następuje przy pracy synchronicznej po stronie wartości zadanej
[, LAi , SynPosLAi , Zi , Ni]	(nawiasów kwadratowych nie należy pisać)
	Min. 1, max 5 sekwencji:
LA1, ... LA5	Osie wiodące
SynPosLAi	Pozycja synchronizacji dla i-tej osi wiodącej
Z1, ... Z5	Licznik dla współczynnika sprzężenia i
N1, ... N5	Mianownik dla współczynnika sprzężenia i
	Współczynnik sprzężenia i = licznik i / mianownik i

Wolno programować tylko osie wiodące, które przedtem zostały wyspecyfikowane przy pomocy EGDEF. Poprzez zaprogramowane "pozycje synchroniczne" dla osi nadążnej (SynPosFA) dla osi wiodących (SynPosLA) są definiowane pozycje, w których zespół sprzężenia jest uważany za *synchroniczny*. O ile przekładnia elektroniczna przy włączeniu nie znajduje się w stanie synchronicznym, oś nadążna wykonuje ruch do swojej zdefiniowanej pozycji synchronicznej.

Wariant 3:

Parametry odpowiadają tym z wariantu 2 plus:

Tryb dosunięcia	Mogą być używane następujące tryby:
"NTGT"	Ruch w optymalnym czasie do najbliższej luki międzyzębnej
"NTGP"	Ruch po optymalnej drodze do najbliższej luki międzyzębnej
"ACN"	Ruch osi obrotowej w kierunku ujemnym absolutnie
"ACP"	Ruch osi obrotowej w kierunku dodatnim absolutnie
"DCT"	W optymalnym czasie do zaprogramowanej pozycji synchronicznej
"DCP"	Po optymalnej drodze do zaprogramowanej pozycji synchronicznej

Wariant 3 ma wpływ tylko na osie nadażne modułu, które są sprzężone z osiami wiodącymi modułu. Optymalizacja czasu uwzględnia granice prędkości osi nadażnej.

Dalsze informacje**Opis wariantów włączenia**

Wariant 1:

Pozycje osi wiodących, jak też osi nadażnej w chwili włączenia są zapisywane jako "pozycje synchroniczne". "Pozycje synchroniczne" mogą być czytane przy pomocy zmiennych systemowych \$AA_EG_SYN.

Wariant 2:

Gdy w zespole sprzężenia są osie modułu, ich pozycje podlegają redukcji modułu. Jest przez to zagwarantowany ruch do najbliższej możliwej pozycji synchronicznej (tzw. *synchronizacja względna*: np. najbliższa luka międzyzębna). Jeżeli dla osi nadażnej nie zostało udzielone "zezwoenie nałożenie osi nadażnej" sygnał interfejsowy DB(30 + numer osi), DBX 26 bit 4, nie następuje ruch do pozycji synchronicznej. Zamiast tego program jest zatrzymywany na bloku EGONSYN i jest wyświetlany samokasujący alarm 16771, aż zostanie ustawiony w/w sygnał.

Wariant 3:

Odstęp zębów (w stopniach) wynika z: $360 \cdot Zi/Ni$. W przypadku gdy oś nadażna stoi w chwili wywołania, optymalizacja drogi daje takie samo zachowanie się, jak optymalizacja czasu.

W przypadku osi nadażnej będącej już w ruchu następuje przy pomocy NTGP niezależnie od aktualnej prędkości osi nadażnej synchronizacja na najbliższą lukę międzyzębną. W przypadku osi nadażnej będącej już w ruchu następuje przy pomocy NTGT zależnie od aktualnej prędkości osi nadażnej synchronizacja na najbliższą lukę międzyzębną. Oś jest w tym celu ew. również hamowana.

Tablice krzywych

Jeżeli dla jednej z osi wiodących jest stosowana **tablica krzywych**, wówczas:

- Ni mianownik współczynnika sprzężenia liniowego musi zostać ustawiony na 0. (mianownik 0 byłby dla sprzężeń liniowych niedopuszczalny). Mianownik jest dla sterowania znakiem, że
- Zi należy interpretować jako numer będącej do zastosowania tablicy krzywych. Tablica krzywych o podanym numerze musi w chwili włączenia być już zdefiniowana.
- LAi Podanie osi wiodącej odpowiada podaniu osi wiodącej w przypadku sprzężenia poprzez współczynnik sprzężenia (sprzężenie liniowe).

Dalsze wskazówki dot. użycia tablic krzywych i kaskadowania przekładni elektronicznych i ich synchronizacji można znaleźć w:

Literatura:

Podręcznik działania Funkcje specjalne; Sprężenia osi i ESR (M3), punkt "Nadążanie i sprzężenie wartości wiodącej".

Zachowanie się przekładni elektronicznej przy power on, RESET, zmianie trybu pracy, szukaniu

- Po power on **żadne** sprzężenie nie jest aktywne.
- Aktywne sprzężenia pozostają zachowane po RESET i zmianie trybu pracy.
- Przy szukaniu bloku polecenia przełączenia, skasowania, definicji przekładni elektronicznej nie są wykonywane i nie są gromadzone, lecz pomijane.

Zmienne systemowe przekładni elektronicznej

Przy pomocy zmiennych systemowych przekładni elektronicznej program obróbki może odczytywać aktualne stany zespołu osi przekładni elektronicznej i ew. reagować na nie.

Zmienne systemowe przekładni elektronicznej są oznaczone następująco:

\$AA_EG_ ...

lub

\$VA_EG_ ...

Literatura:

Podręcznik zmiennych systemowych

11.4.3 Wyłączenie przekładni elektronicznej (EGOFS, EGOFC)

Funkcja

Dla wyłączenia aktywnego zespołu osi przekładni elektronicznej istnieją 3 warianty.

Programowanie

Wariant 1:

Składnia

EGOFS (oś nadążna)

Znaczenie

Przekładnia elektroniczna jest wyłączana Oś nadążna jest hamowana do stanu zatrzymanego. Wywołanie wyzwała zatrzymanie przebiegu wyprzedzającego.

Wariant 2:

Składnia

EGOFS (oś nadążna, oś wiodąca1, ..., oś wiodąca5)

Znaczenie

Ta parametryzacja polecenia pozwana na **selektywny** wpływ poszczególnych osi wiodących na ruch osi nadążnej.

Musi zostać podana co najmniej jedna oś wiodąca. Wpływ podanych osi wiodących na oś nadążną zostaje w sposób celowy wyłączony. Wywołanie wyzwała zatrzymanie przebiegu wyprzedzającego. Jeżeli pozostają jeszcze aktywne osie wiodące, wówczas oś nadążna pod ich wpływem nadal wykonuje ruch. Gdy wszystkie oddziaływania osi wiodących są w ten sposób wyłączone, oś nadążna jest hamowana do stanu zatrzymanego.

Wariant 3:

Składnia

EGOFC (wrzeciono nadążne1)

Znaczenie

Przekładnia elektroniczna jest wyłączana Wrzeciono nadążne wykonuje nadal ruch z prędkością obrotową / prędkością aktualną w chwili wyłączenia. Wywołanie wyzwała zatrzymanie przebiegu wyprzedzającego.

Wskazówka

Ten wariant jest dozwolony tylko dla wrzecion.

11.4.4 Skasowanie definicji przekładni elektronicznej (EGDEL)

Funkcja

Zespół osi przekładni elektronicznej musi być wyłączony, zanim definicja będzie mogła zostać skasowana.

Programowanie

Składnia

EGDEL (oś nadążna)

Znaczenie

Definicja sprzężenia zespołu osi jest kasowana. Aż do osiągnięcia maksymalnej liczby równocześnie uaktywnionych zespołów osi, jest ponownie możliwe zdefiniowanie dalszych zespołów osi przy pomocy EGDEF. Wywołanie wyzwała zatrzymanie przebiegu wyprzedzającego.

11.4.5 Posuw na obrót (G95) / przekładnia elektroniczna (FPR)

Funkcja

Poleceniem FPR można również oś nadążną przekładni elektronicznej podać jako oś określającą posuw na obrót. Dla tego przypadku obowiązuje następujące zachowanie się:

- Posuw jest zależny od prędkości zadanej osi nadążnej przekładni elektronicznej.
- Prędkość zadana jest obliczana z prędkości wrzecion wiodących i osi wiodących modulo (które nie są osiami uczestniczącymi w tworzeniu konturu) i im przyporządkowanych współczynników sprzężenia.
- Składowe prędkości osi liniowych wzgl. osi wiodących nie modulo i ruchy nałożone osi nadążnej nie są uwzględniane.

11.5 Wrzeciono synchroniczne

Funkcja

W pracy synchronicznej jest wrzeciono wiodące (LS) i wrzeciono nadążne (FS), tzw. **para wrzecion synchronicznych**. Wrzeciono nadążne podąża przy aktywnym sprzężeniu (praca synchroniczna) za ruchami wrzeciona wiodącego odpowiednio do ustalonej zależności funkcyjnej.

Pary wrzecion synchronicznych dają się dla każdej maszyny zarówno projektować na stałe przy pomocy kanałowych danych maszynowych albo definiować specyficznie dla zastosowania przez program obróbki CNC. W kanale NC może równocześnie pracować do 2 par wrzecion synchronicznych.

Sprzężenie może zostać z programu obróbki

- zdefiniowane lub zmienione
- włączone
- wyłączone
- skasowane

.

Ponadto można zależnie od wersji oprogramowania

- czekać na warunek pracy synchronicznej
- zmienić zachowanie się pod względem zmiany bloku
- wybrać rodzaj sprzężenia, albo sprzężenie wartości zadanej albo sprzężenie wartości rzeczywistej, lub zadać przesunięcie kątowe między wrzecionem wiodącym i wrzecionem nadążnym
- przy włączeniu sprzężenia przejąć poprzednie zaprogramowanie wrzeciona nadążnego
- skorygować albo zmierzone albo już znane odchylenie ruchu synchronicznego

.

11.5.1 Wrzeciono synchroniczne: Programowanie (COUPDEF, COUPDEL, COUPON, COUPONC, COUPOF, COUPOFS, COUPRES, WAITC)

Funkcja

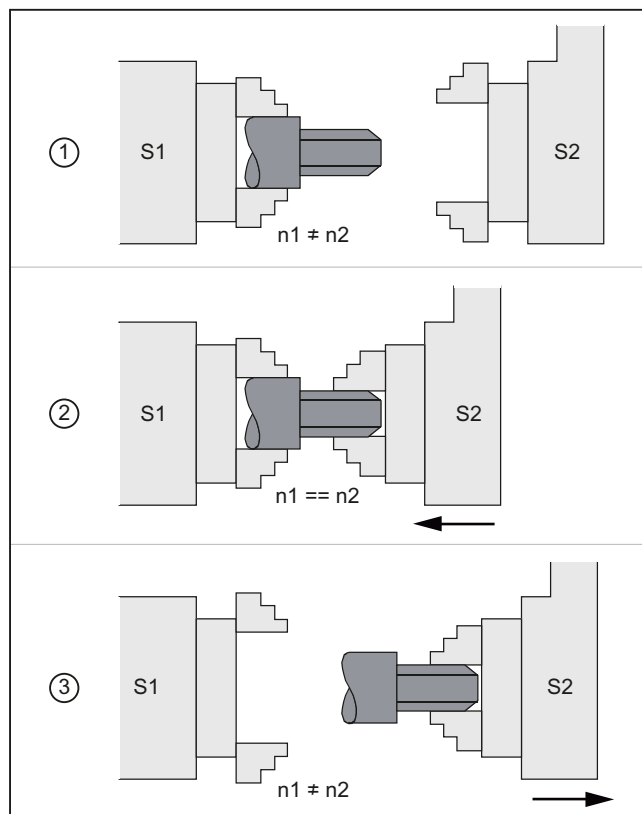
Funkcja "Wrzeciono synchroniczne" umożliwia synchroniczny pod względem liczby obrotów proces wrzeciona nadążnego (FS) oraz wiodącego (LS) za pomocą zaprogramowanego stopunku przełożenia.

Funkcja udostępnia następujące tryby:

- Synchronizacja prędkości obrotowych ($n_{FS} = n_{LS}$)
- Synchronizacja położenia ($\phi_{FS} = \phi_{LS}$)
- Synchronizacja położenia z przesunięciem kątowym ($\phi_{FS} = \phi_{LS} + \Delta\phi$)

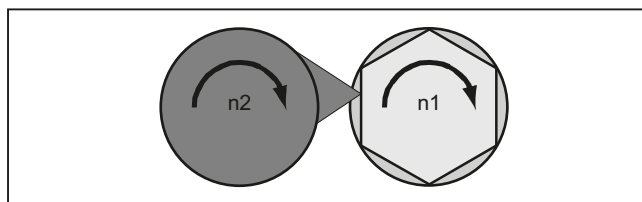
Przykłady zastosowania:

- Przekazanie przedmiotu obrabianego podczas ruchu np. do obróbki tylnej strony, stosunku przełożenia: 1:1



- ① Synchronizacja liczby obrotów
- ② Przekazanie przedmiotu obrabianego
- ③ Obróbka tylnej strony

- Obróbka konturów wielobocznych (toczenie wieloboku), synchroniczność prędkości obrotowej, stosunek przełożenia: $n_1:n_2$



Składnia

COUPDEF (<FS>, <LS>, <ÜFS>, <ÜLS>, <zmiana bloku>, <rodzaj sprzężenia>)

COUPON (<FS>, <LS>, <POSFS>)

COUPONC (<FS>, <LS>)

COUPOF (<FS>, <LS>, <POSFS>, <POSLS>)

COUPOFS (<FS>, <LS>)

COUPOFS (<FS>, <LS>, <POSFS>)

COUPRES (<FS>, <LS>)

COUPDEL (<FS>, <LS>)
 WAITC (<FS>, <zmiana bloku>, <LS>, <zmiana bloku>)

Wskazówka

Skrócony sposób zapisu

W przypadku instrukcji COUPOF, COUPOFS, COUPRES i COUPDEL jest możliwy skrócony sposób zapisu bez podania wrzeciona wiodącego.

Znaczenie

COUPDEF:	Definicja/zmiana sprzężenia specyficzna dla użytkownika
COUPON:	Włączenie sprzężenia. Wychodząc od aktualnej prędkości obrotowej wrzeciono nadążne synchronizuje się na wrzeciono wiodące
COUPONC:	Przejęcie sprzężenia z poprzednim zaprogramowaniem od M3 S... albo M4 S...
	Różnica prędkości obrotowej wrzeciona nadążnego jest przyjmowana natychmiast.
COUPOF:	Wyłączenie sprzężenia. <ul style="list-style-type: none"> Z natychmiastową zmianą bloku: COUPOF (<S2>, <S1>) Zmiana bloku dopiero po przejściu pozycji wyłączenia <POSFS> wzgl. <POSLS>: COUPOF (<S2>, <S1>, <POSFS>) COUPOF (<S2>, <S1>, <POSFS>, <POSLS>)
COUPOFS:	Wyłączenie sprzężenia z zatrzymaniem wrzeciona nadążnego. Zmiana bloku jak najszybciej z natychmiastową zmianą bloku: COUPOFS (<S2>, <S1>) Zmiana bloku dopiero po przejściu pozycji wyłączenia: COUPOFS (<S2>, <S1>, <POSFS>)
COUPRES:	Cofnięcie parametrów sprzężenia do zaprojektowanych MD i SD
COUPDEL:	Skasowanie sprzężenia definiowanego przez użytkownika
WAITC:	Czekanie na warunek pracy synchronicznej (NOC są anulowane na IPO przy zmianie bloku)
<FS>:	Określenie wrzeciona nadążnego

Parametry opcjonalne:

<LS>:	Określenie wrzeciona wiodącego
	Podanie z numerem wrzeciona: z. B. S2, S1
<ZFS>, <NLS>:	Stosunek przełożenia między FS i LS. $\langle ZFS \rangle / \langle NLS \rangle = \text{Licznik} / \text{Mianownik}$ Ustawienie domyślne: $\langle ZFS \rangle / \langle NLS \rangle = 1.0$; Podanie mianownika opcjonalne

<zmiana bloku>:	Zachowanie się pod względem zmiany bloku Zmiana bloku następuje: "NOC" natychmiast "FINE" z osiągnięciem "ruchu synchronicznego dokładnie" "COARSE" z osiągnięciem "ruchu synchronicznego zgrubnie" "IPOSTOP" z osiągnięciem IPOSTOP, tzn. po uzyskaniu ruchu synchronicznego po stronie wartości zadanej (ustawienie domyślne)
<rodzaj sprzężenia>:	Zachowanie się pod względem zmiany bloku działa modalnie. Rodzaj sprzężenia: Sprzężenie między FS i LS. "DV" Sprzężenie wartości zadanej (ustawienie domyślne) "AV" Sprzężenie wartości rzeczywistej "VV" Sprzężenie prędkości Rodzaj sprzężenia działa modalnie.
<POSFS>:	Przesunięcie kątowe między wrzeczionem wiodącym i nadążnym Zakres wartości: 0°... 359,999°
<POSFS>, <POSLS>:	Pozycje wyłączenia wrzecziona nadążnego i wrzecziona wiodącego "Zmiana bloku jest zwalniana po przejściu POS _{FS} , POS _{LS} " Zakres wartości: 0°... 359,999°

Przykłady

Praca z wrzeczionem wiodącym i wrzeczionem nadążnym

Kod programu	Komentarz
	Wrzecziono wiodące = wrzecziono master = wrzecziono 1 Wrzecziono nadążne = wrzecziono 2
N05 M3 S3000 M2=4 S2=500	Wrzecziono wiodące obraca się z 3000 obr/min, wrzecziono nadążne obraca się z 500 obr/min.
N10 COUPDEF(S2,S1,1,1,"NOC","Dv")	Definicja sprzężenia (może być też projektowane).
...	
N70 SPCON	Wrzecziono wiodące na regulację położenia (sprzężenie wartości zadanej).
N75 SPCON(2)	Wrzecziono nadążne na regulację położenia.
N80 COUPON(S2,S1,45)	Sprzężenie w ruchu na pozycję offsetową = 45 stopni.
...	
N200 FA[S2]=100	Prędkość pozycjonowania = 100 stopni/min
N205 SPOS[2]=IC(-90)	Nałożenie ruchu 90 stopni w kierunku ujemnym.
N210 WAITC(S2,"Fine")	Czekanie na ruch synchroniczny "dokładnie".
N212 G1 X... Y... F...	Obróbka
...	

Kod programu	Komentarz
N215 SPOS[2]=IC(180)	Nałożenie ruchu 180 stopni w kierunku dodatnim.
N220 G4 S50	Czas oczekiwania = 50 obrotów wrzeciona wiodącego
N225 FA[S2]=0	Uaktywnienie zaprojektowanej prędkości (MD).
N230 SPOS[2]=IC(-7200)	20 obrotów. Ruch z zaprojektowaną prędkością w kierunku ujemnym.
...	
N350 COUPOF(S2,S1)	Odsprężenie w ruchu, S=S2=3000
N355 SPOSA[2]=0	Zatrzymanie wrzeciona nadażnego na zero stopni.
N360 G0 X0 Y0	
N365 WAITS(2)	Czekanie na wrzeciono 2.
N370 M5	Zatrzymanie wrzeciona nadażnego.
N375 M30	

Programowanie różnicy prędkości obrotowej

Kod programu	Komentarz
	Wrzeciono wiodące = wrzeciono master = wrzeciono 1 Wrzeciono nadażne = wrzeciono 2
N01 M3 S500	Wrzeciono wiodące obraca się z 500 obr/min.
N02 M2=3 S2=300	Wrzeciono nadażne obraca się z 300 obr/min.
...	
N10 G4 F1	Czas oczekiwania wrzeciona wiodącego.
N15 COUPDEF (S2,S1,-1)	Współczynnik sprzężenia ze stosunkiem przełożenia -1:1
N20 COUPON (S2,S1)	Uaktywnienie sprzężenia. Prędkość obrotowa wrzeciona nadażnego wynika z prędkości obrotowej wrzeciona wiodącego i współczynnika sprzężenia
...	
N26 M2=3 S2=100	Programowanie różnicy prędkości obrotowej.

Przykłady przejęcia ruchu do różnicy prędkości obrotowej

1. Włączenie sprzężenia przy uprzednim zaprogramowaniu wrzeciona nadażnego przy pomocy COUPON

Kod programu	Komentarz
	Wrzeciono wiodące = wrzeciono master = wrzeciono 1 Wrzeciono nadażne = wrzeciono 2
N05 M3 S100 M2=3 S2=200	Wrzeciono wiodące obraca się z prędkością 100 obr/min, wrzeciono nadażne obraca się z 200 obr/min.
N10 G4 F5	Czas oczekiwania wrzeciona master = 5 sekund
N15 COUPDEF(S2,S1,1)	Stosunek przełożenia wrzeciono nadażne do wrzeciona wiodącego wynosi 1,0 (ustawienie domyślne).
N20 COUPON(S2,S1)	Sprężenie w ruchu na wrzeciono wiodące.
N10 G4 F5	Wrzeciono nadażne obraca się ze 100 obr/min.

2. Włączenie sprzężenia przy uprzednim zaprogramowaniu wrzeciona nadążnego przy pomocy COUPONC

Kod programu	Komentarz
	Wrzeciono wiodące = wrzeciono master = wrzeciono 1 Wrzeciono nadążne = wrzeciono 2
N05 M3 S100 M2=3 S2=200	Wrzeciono wiodące obraca się z prędkością 100 obr/min, wrzeciono nadążne obraca się z prędkością 200 obr/min.
N10 G4 F5	Czas oczekiwania wrzeciona master = 5 sekund
N15 COUPDEF (S2, S1, 1)	Stosunek przełożenia wrzeciono nadążne do wrzeciona wiodącego wynosi 1,0 (ustawienie domyślne).
N20 COUPONC (S2, S1)	Sprzężenie w ruchu na wrzeciono wiodące i przejście poprzedniej prędkości obrotowej do S2.
N10 G4 F5	S2 obraca się ze 100obr/min + 200obr/min = 300obr/min

3. Włączenie sprzężenia przy zatrzymanym wrzecionie nadążnym z COUPON

Kod programu	Komentarz
	Wrzeciono wiodące = wrzeciono master = wrzeciono 1 Wrzeciono nadążne = wrzeciono 2
N05 SPOS=10 SPOS[2]=20	Wrzeciono nadążne S2 w trybie pozycjonowania.
N15 COUPDEF (S2, S1, 1)	Stosunek przełożenia wrzeciono nadążne do wrzeciona wiodącego wynosi 1,0 (ustawienie domyślne).
N20 COUPON (S2, S1)	Sprzężenie w ruchu na wrzeciono wiodące.
N10 G4 F1	Sprzężenie jest zamykane, S2 zatrzymuje się na 20 stopniach.

4. Włączenie sprzężenia przy zatrzymanym wrzecionie nadążnym z COUPONC

Wskazówka

Tryb pozycjonowania lub tryb osi

Jeżeli wrzeciono nadążne przed sprzężeniem znajduje się w trybie pozycjonowania albo pracy jako oś, wówczas wrzeciono nadążne przy COUPON (<FS>, <LS>) i COUPONC (<FS>, <LS>) zachowuje się tak samo.

Wskazówka

Wrzeciono wiodące i praca w trybie osi

Jeżeli wrzeciono wiodące przed definicją sprzężenia znajduje się w trybie pracy jako oś, również po włączeniu sprzężenia działa wartość graniczna prędkości z danej maszynowej:

MD32000 \$MA_MAX_AX_VELO (maksymalna prędkość osi)

W celu uniknięcia tego zachowania się oś musi przed definicją sprzężenia zostać przełączona na tryb wrzeciona (M3 S... lub M4 S...).

Dalsze informacje

Projektowane sprzężenie

Przy projektowaniu sprzężenia wrzeciono wiodące i nadążne są ustalane przez daną maszynową. Zaprojektowanych wrzecion nie można zmienić w programie obróbki. Parametryzacja sprzężenia może nastąpić w programie obróbki przy pomocy `COUPDEF` (warunek: nie jest ustalona ochrona przed zapisem).

Sprzężenie definiowane przez użytkownika

Przy pomocy `COUPDEF` można na nowo zdefiniować lub zmienić sprzężenie w programie obróbki. Jeżeli jest już aktywne sprzężenie, musi ono przed definicją nowego sprzężenia zostać najpierw skasowane przy pomocy `COUPDEL`.

Sprzężenie jest kompletnie definiowane przez:

`COUPDEF (<FS>, <LS>, <ÜFS>, <ÜLS>`, zachowanie się przy zmianie bloku, rodzaj sprzężenia)

Wrzeciono nadążne (FS) i wrzeciono wiodące (LS)

Przy pomocy nazw osi wrzeciona nadążnego (FS) i wrzeciona wiodącego (LS) sprzężenie jest określone jednoznacznie. Nazwy osi muszą być programowane z każdą instrukcją `COUPDEF`. Inne parametry sprzężenia działają modalnie i muszą być programowane tylko wtedy, gdy są zmieniane.

Przykład:

`COUPDEF (S2, S1)`

Stosunek przełożenia

Stosunek przełożenia podawany jest jako stosunek prędkości obrotowej wrzecina nadążnego i wrzecina wiodącego:

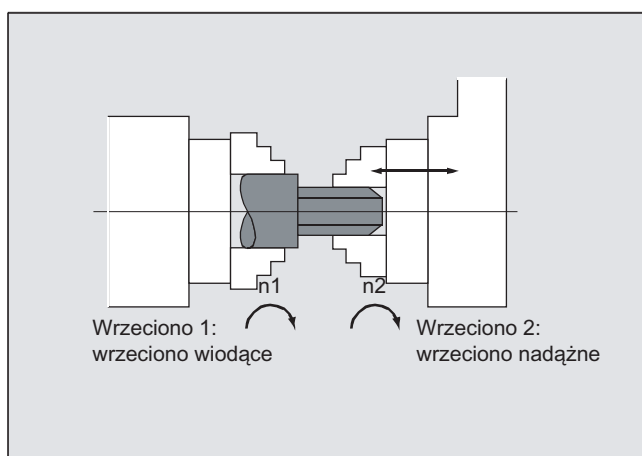
Wrzeciono nadążne/Wrzeciono wiodące = Licznik/Mianownik

Licznik musi zostać zaprogramowany. Mianownik musi zostać zaprogramowany. Dla mianownika ustalona wartość domyślna wynosi 1.0.

Przykład:

Wrzeciono nadążne S2 i wrzeciono wiodące S1, stosunek przełożenia = 1 / 1

`COUPDEF (S2, S1, 1.0)`



Wskazówka

Stosunek przełożenia można również zmienić w ruchu podczas włączonego sprzężenia.

Zachowanie się przy zmianie bloku NOC, FINE, COARSE, IPOSTOP

Przy programowaniu zachowania się przy zmianie bloku jest możliwy następujący skrócony sposób zapisu:

- "NO": natychmiast (ustawienie domyślne)
- "FI": z osiągnięciem "ruchu synchronicznego dokładnie"
- "CO": z osiągnięciem "ruchu synchronicznego zgrubnie"
- "IP": z osiągnięciem IPOSTOP, tzn. po uzyskaniu ruchu synchronicznego po stronie wartości zadanej

Rodzaj sprzężenia

Wskazówka

Rodzaj sprzężenia wolno jest zmienić tylko przy wyłączonym sprzężeniu.

Włączenie pracy synchronicznej COUPON, <POSFS>

- Włączenie sprzężenia z dowolnym przesunięciem kątowymk między LS i FS:
 - COUPON (S2, S1)
 - COUPON (S2)
- Włączenie sprzężenia z zależnością kątową <POSFS>
<POSFS> odnosi się do pozycji 0° wrzeciona wiodącego w dodatnim kierunku obrotów
obszar wartości <POSFS>: 0°... 359,999°
 - COUPON (S2, S1, 30)

Wskazówka

Przesunięcie kątowe może zostać zmienione również przy aktywnym sprzężeniu.

Pozycjonowanie wrzeciona nadążnego

Również przy włączonym sprzężeniu wrzeciona synchronicznego wrzeciono nadążne można pozycjonować niezależnie od wrzeciona wiodącego w obszarze $\pm 180^\circ$.

- Pozycjonowanie wrzeciona nadążnego za pomocą SPOS

Przykład: SPOS [2] = IC (-90)

Dalsze informacje dot. SPOS znajdują się w:

Literatura:

Podręcznik programowania Podstawy

Różnica prędkości obrotowej

Różnica prędkości obrotowej powstaje podczas pracy ze sterowaniem prędkością obrotową oraz aktywnego sprzężenia wrzecion synchronicznych poprzez podlegające ewaluacji nałożenie prędkości obrotowej wrzeciona nadążnego na podstawie ruchu wrzeciona wiodącego oraz prędkości obrotowej wrzeciona wiodącego na podstawie zaprogramowania wrzeciona:

- Sprzężenie wrzeciona synchronicznego za pomocą COUPONC
- $S_{FS} = \text{prędkość obrotowa} [M_{FS} = \text{kierunek obrotu}]$

Wskazówka

Warunki brzegowe

- Dla kierunku obrotu M_3 / M_4 należy również na nowo zaprogramować prędkość obrotową $S \dots$
- Nałożenie prędkości obrotowej wrzeciona ($M_{kierunek obrotu} S_{FS}$) przez ruch wrzeciona wiodącego podczas sprzężenia wrzecion synchronicznych COUPONC działa tylko wtedy, gdy jest zezwolenie dla nałożenia.
- Dynamika wrzeciona wiodącego musi zostać na tyle ograniczona, by przy nałożeniu wrzeciona nadążnego jego wartości graniczne dynamiki nie zostały przekroczone.

Odnosnie dalszych informacji dotyczących różnicy prędkości obrotowej patrz: :

Literatura:

Podręcznik działania Funkcje rozszerzające; Wrzeciono synchroniczne (S3)

Prędkość, przyspieszenie FA, ACC, OVRA, VELOLIMA

Osiową prędkość i przyspieszenie wrzeciona nadążnego można programować przy pomocy:

- $FA[SPI(S_{<n>})]$ wzgl. $FA[S_{<n>}]$ (prędkość osiowa)
- $ACC[SPI(S_{<n>})]$ wzgl. $ACC[S_{<n>}]$ (przyspieszenie osiowe)
- $OVRA[SPI(S_{<n>})]$ wzgl. $OVRA[S_{<n>}]$ (override osiowy)
- $VELOLIMA[SPI(S_{<n>})]$ wzgl. $VELOLIMA[S_{<n>}]$ (osiowe zwiększenie lub zmniejszenie prędkości)

Przy pomocy $<n> = 1, 2, 3, \dots$ (numery wrzecion nadążnych)

Literatura

Podręcznik programowania Podstawy

Wskazówka

Zmniejszenie albo zwiększenie maksymalnego osiowego przyspieszenia nie działa w przypadku wrzecion.

Odnosnie dalszych informacji dotyczących dynamiki osiowej znajdują się w :

Literatura:

Podręcznik funkcjonowania Funkcje rozszerzające; Osie obrotowe (R2)

Programowane zachowanie się przy zmianie bloku WAITC

Przy pomocy `WAITC` może zostać zadane zachowanie się przy zmianie bloku, np. po zmianie parametrów sprzężenia albo procesów pozycjonowania, z różnymi warunkami ruchu synchronicznego (zgrubnie, dokładnie, `IPOSTOP`). Jeżeli warunki ruchu synchronicznego nie są podane, obowiązuje zachowanie się podane przy definicji `COUPDEF`.

Przykłady

- Czekanie na osiągnięcie warunku ruchu synchronicznego `FINE` w przypadku wrzeciona nadążnego `S2` i `COARSE` w przypadku wrzeciona nadążnego `S4`:
`WAITC(S2, "FINE", S4, "COARSE")`
- Czekanie na osiągnięcie warunku ruchu synchronicznego zgodnie z `COUPDEF`: `WAITC()`

Wyłączenie sprzężenia COUPOF

Przy pomocy `COUPOF` można zadać zachowanie się sprzężenia przy wyłączeniu:

- Wyłączenie sprzężenia z natychmiastową zmianą bloku:
 - `COUPOF(S2, S1)` (z podaniem wrzeciona wiodącego)
 - `COUPOF(S2)` (bez podania wrzeciona wiodącego)
- Wyłączenie sprzężenia po przejściu pozycji wyłączenia. Zmiana bloku następuje po przejściu pozycji wyłączenia.
 - `COUPOF(S2, S1, 150)` (pozycja wyłączenia FS: 150°)
 - `COUPOF(S2, S1, 150, 30)` (pozycja wyłączenia FS: 150°, LS: 30°)

Wyłączenie sprzężenia z zatrzymaniem wrzeciona nadążnego COUPOFS

Przy pomocy `COUPOFS` można zadać zachowanie się sprzężenia przy wyłączeniu z zatrzymaniem wrzeciona nadążnego:

- Wyłączenie sprzężenia z zatrzymaniem wrzeciona nadążnego z natychmiastową zmianą bloku:
 - `COUPOFS(S2, S1)` (z podaniem wrzeciona wiodącego)
 - `COUPOFS(S2)` (bez podania wrzeciona wiodącego)
- Wyłączenie sprzężenia po przejściu pozycji wyłączenia z zatrzymaniem wrzeciona nadążnego. Zmiana bloku następuje po przejściu pozycji wyłączenia.
 - `COUPOFS(S2, S1, 150)` (pozycja wyłączenia FS: 150°)

Skasowanie sprzężeń COUPDEL

Przy pomocy `COUPDEL` sprzężenie jest kasowane:

- `COUPDEL(S2, S1)` (z podaniem wrzeciona wiodącego)
- `COUPDEL(S2)` (bez podania wrzeciona wiodącego)

Cofnięcie parametrów sprzężenia COUPRES

Przy pomocy `COUPRES` są uaktywniane wartości sprzężenia, sparametryzowane w danych maszynowych i ustawczych:

- `COUPRES(S2, S1)` (z podaniem wrzeciona wiodącego)
- `COUPRES(S2)` (bez podania wrzeciona wiodącego)

Zmienne systemowe

- Aktualny stan sprzężenia wrzeciona nadążnego

Aktualny stan sprzężenia wrzeciona nadążnego może zostać przeczytany w sposób kodowany bitowo poprzez:

<wartość> = \$AA_COUP_ACT [<FS>]

Bit	<wartość >	Znaczenie
-	0	Nie jest aktywne żadne sprzężenie
2	4	Sprzężenie wrzeciona synchronicznego aktywne
Wskazówka		
<ul style="list-style-type: none"> • Wszystkie inne wartości odnoszą się do pracy osi • Jeśli wrzeciono stanowi wrzeciono nadążne o kilku sprzężeniach, jako stan sumaryczny zostanie zwrócona wartość stanu sprzężenia wszystkich sprzężeń. 		

- Aktualne przesunięcie kątowe

Aktualne przesunięcie kątowe wrzeciona nadążnego do wrzeciona wiodącego może zostać odczytane poprzez:

- \$AA_COUP_OFFS [<FS>] (przesunięcie kątowe po stronie wartości zadanej)
- \$VA_COUP_OFFS [<FS>] (przesunięcie kątowe po stronie wartości rzeczywistej)

Przykład zastosowania

Korekcja różnicy przesunięcia kąтового w programie NC po wyłączeniu trybu śledzenia:

Różnica przesunięcia kąowego = Zaprogramowane przesunięcie kątowe - zmienna systemowa

Literatura

Szczegółowe informacje dot. zmiennych systemowych znajdują się w:

Podręcznik Lista zmiennych systemowych

11.6 Sprzężenie rodzajowe (CP...)

Funkcja

"Sprzężenie ogólne" jest ogólną funkcją sprzężenia, w której zebrane są wszystkie właściwości istniejących rodzajów sprzężenia (nadażanie, sprzężenie wartości wiodącej, przekładnia elektroniczna oraz wrzeciono synchroniczne).

Funkcja umożliwia elastyczny rodzaj programowania:

- Użytkownik może dla swoich aplikacji w sposób losowy wybrać właściwości sprzężenia (zasada modułów).
- Każda właściwość sprzężenia jest pojedynczo programowana.
- Możliwe są zmiany właściwości sprzężenia zdefiniowanego sprzężenia (np. współczynnik sprzężenia).
- Późniejsze używanie dalszych właściwości sprzężenia jest możliwe.
- Układ współrzędnych odniesienia osi nadażnej (bazowy układ współrzędnych lub układ współrzędnych maszyny) jest programowany.
- Określone właściwości sprzężenia mogą być również programowane w akcjach synchronicznych.

Literatura: Podręcznik działania Akcje synchroniczne

Wskazówka

Dotychczasowe wywołania sprzężenia dla nadażania (TRAIL*), sprzężenia wartości wiodącej (LEAD*), napędu elektronicznego (EG*) oraz wrzeciona synchronicznego (COUP*) są dalej wspierane poprzez cykle dopasowania.

Przegląd wszystkich słów kluczowych oraz właściwości sprzężenia

Następująca tabela przedstawia przegląd wszystkich słów kluczowych dla sprzężenia ogólnego, a tym samym programowanych właściwości sprzężenia:

Słowo kluczowe	Właściwość sprzężenia/Znaczenie	Składnia
CPDEF	Utworzenie trybu sprzężenia	CPDEF= (<FAX>)
CPDEL	Skasowanie trybu sprzężenia	CPDEL= (<FAX>)
CPLA	Definicja osi wiodącej	CPLA [<FAX>] = (<LAX>)
CPLDEF	Definicja osi wiodącej i utworzenie trybu sprzężenia (również możliwe za pomocą CPDEF + CPLA)	CPLDEF [<FAX>] = (<LAX>) lub CPDEF= (<FAX>) CPLA [<FAX>] = (<LAX>)
CPLDEL	Skasowanie osi wiodącej modułu sprzężenia (również możliwe za pomocą CPDEF + CPLA)	CPLDEL [<FAX>] = (<LAX>) lub CPDEL= (<FAX>) CPLA [<FAX>] = (<LAX>)

Słowo kluczowe	Właściwość sprzężenia/Znaczenie	Składnia		
CPON	Włączenie modułu sprzężenia	CPON= (<FAX>)		
CPOF	Wyłączenie modułu sprzężenia	CPOF= (<FAX>)		
CPLON	Włączenie osi wiodącej modułu sprzężenia	CPLON [<FAX>] =<LAX>		
CPLOF	Wyłączenie osi wiodącej modułu sprzężenia	CPLOF [<FAX>] =<LAX>		
CPLNUM	Licznik współczynnika sprzężenia	CPLNUM [FAX, LAX] =<wartość>		
CPLDEN	Mianownik współczynnika sprzężenia	CPLDEN [FAX, LAX] =<wartość>		
CPLCTID	Numer tablicy krzywych	CPLCTID [FAX, LAX] =<wartość>		
CPLSETVAL	Odniesienie sprzężenia	CPLSETVAL [FAX, LAX] = "<Odniesienie sprzężenia>"		
		"<Odniesienie sprzężenia>":	"CMDPOS"	Sprzężenie wartości zadanej
			"CMDVEL"	Sprzężenie prędkości
			"ACTPOS"	Sprzężenie wartości rzeczywistej
CPFRS	Układ odniesienia współrzędnych	CPFRS [FAX] = "<Odniesienie sprzężenia>"		
		"<Odniesienie sprzężenia>":	"BCS"	Bazowy układ współrzędnych
			"MCS"	Układ współrzędnych maszyny
CPBC	Kryterium zmiany bloku	CPBC [FAX] = "<Kryterium zmiany bloku>"		
		"<Kryterium zmiany bloku>":	"NOC"	Zmiana bloku następuje niezależnie od stanu sprzężenia.
			"IPOSTOP"	Zmiana bloku następuje przy pracy synchronicznej po stronie wartości zadanej.
			"COARSE"	Zmiana bloku następuje przy ruchu synchronicznym "zgrubnie" po stronie wartości rzeczywistej.
			"FINE"	Zmiana bloku następuje przy ruchu synchronicznym "dokładnie" po stronie wartości rzeczywistej.
CPFPOS + CPON	Pozycja synchroniczna osi nadążnej przy włączeniu	CPON=FAX CPFPOS [FAX] =<wartość>		
CPLPOS + CPON	Pozycja synchroniczna osi wiodącej przy załączeniu	CPLPOS [FAX, LAX] =<wartość>		

Słowo kluczowe	Właściwość sprzężenia/Znaczenie	Składnia		
CPFMSON	Tryb synchronizacji	CPFMSON[Fax] = "<Tryb synchronizacji>"		
		"<Tryb synchronizacji>":	"CFAST"	Sprzężenie zostanie zamknięte w optymalnym czasie.
			"CCOARSE"	Sprzężenie zostanie dopiero wtedy włączone, gdy zgodnie z regułą sprzężenia żądana pozycja osi nadążnej będzie się znajdowała w obszarze aktualnej pozycji osi nadążnej.
			"NTGT"	Nastąpi ruch w optymalnym czasie do najbliższej luki międzyzębnej.
			"NTGP"	Nastąpi ruch po optymalnej drodze do najbliższej luki międzyzębnej.
			"NRGT"	Ruch w optymalnym czasie do segmentu zgodnie ze stosunkiem ilości przekładni do liczby zębów.
			"NRGP"	Ruch w optymalnym czasie do segmentu zgodnie ze stosunkiem ilości przekładni do liczby zębów.
			"ACN"	Tylko dla osi obrotowych! Oś obrotowa wykonuje ruch w ujemnym kierunku osi do pozycji synchronicznej. Synchronizacja następuje od razu.
			"ACP"	Tylko dla osi obrotowych! Oś obrotowa wykonuje ruch w dodatnim kierunku osi do pozycji synchronicznej. Synchronizacja następuje od razu.
			"DCT"	Tylko dla osi obrotowych! Oś obrotowa wykonuje ruch w optymalnym czasie do zaprogramowanej pozycji synchronicznej. Synchronizacja następuje od razu.
		"DCP"	Tylko dla osi obrotowych! Oś obrotowa wykonuje ruch po optymalnej drodze do zaprogramowanej pozycji synchronicznej. Synchronizacja następuje od razu.	

Słowo kluczowe	Właściwość sprzężenia/Znaczenie	Składnia		
CPFMON	Zachowanie się osi nadążnej przy włączeniu	CPFMON [FAx] = "<Podtrzymanie włączenia>"		
		"<Podtrzymanie włączenia>":	"STOP"	Tylko przy wrzecionach! Aktywny ruch wrzeciona nadążnego zostanie zatrzymany przed włączeniem.
			"CONT"	Tylko dla wrzecion oraz osi przebiegu głównego! Aktualny ruch osi nadążnej/wrzeciona nadążnego zostanie przejęty do sprzężenia jako ruch początkowy.
			"ADD"	Tylko przy wrzecionach! Składowe ruchu sprzężenia działają dodatkowo dla aktualnego ruchu nałożonego, tzn. aktualny ruch osi nadążnej/wrzeciona nadążnego zostanie zachowany jako ruch nałożony.
CPFMOF	Zachowanie się osi nadążnej przy kompletnym wyłączeniu	CPFMOF [FAx] = "<Podtrzymanie wyłączenia>"		
		"<Podtrzymanie wyłączenia>":	"STOP"	Zatrzymanie wrzeciona nadażnego/osi nadażnej. Aktywny ruch nałożony zostanie zahamowany do stanu zatrzymania. Następnie zostanie otwarte sprzężenie
			"CONT"	Tylko dla wrzecion oraz osi przebiegu głównego! Wrzeciono nadażne wykonuje nadal ruch z prędkością obrotową / prędkością aktualną w chwili wyłączenia.
CPFPOS + CPOF	Pozycja wyłączenia osi nadażnej przy wyłączeniu	CPOF = (FAx) CPFPOS [FAx] = <wartość>		
CPMRESET	Zachowanie się sprzężenia przy RESET	CPMRESET [FAx] = "<Podtrzymanie Reset>"		
		"<Podtrzymanie Reset>":	"NONE"	Aktualny stan sprzężenia pozostaje zachowany.
			"ON"	Jeżeli zostanie utworzony odpowiedni moduł sprzężenia, zostanie ono włączone. Wszystkie zdefiniowane powiązania osi wiodących zostaną aktywowane. Następuje to również wtedy, gdy wszystkie albo część tych powiązań jest już aktywnych, tzn. również przy całkowicie aktywnym sprzężeniu następuje nowa synchronizacja.

Słowo kluczowe	Właściwość sprzężenia/Znaczenie	Składnia	
		"OF"	Aktywny ruch nałożony zostanie zahamowany do stanu zatrzymania. Następnie sprzężenie zostanie wyłączone. Jeśli dany moduł sprzężenia zostanie utworzony bez wyraźnej definicji (CPDEF), zostanie on usunięty. W innym przypadku pozostaje nadal tworzony, tzn. może być dalej używany.
		"OFC"	Możliwe tylko przy wrzecionach! Wrzeciono nadążne wykonuje nadal ruch z prędkością obrotową / prędkością aktualną w chwili wyłączenia. Sprzężenie zostanie wyłączone. Jeśli dany moduł sprzężenia zostanie utworzony bez wyraźnej definicji (CPDEF), zostanie on usunięty. W innym przypadku pozostaje nadal tworzony, tzn. może być dalej używany.
		"DEL"	Aktywny ruch nałożony zostanie zahamowany do stanu zatrzymania. Potem sprzężenie zostanie dezaktywowane, a następnie usunięte.
		"DELC"	Możliwe tylko przy wrzecionach! Wrzeciono nadążne wykonuje nadal ruch z prędkością obrotową / prędkością aktualną w chwili wyłączenia. Sprzężenie zostanie dezaktywowane, a następnie usunięte.
CPMSTART	Zachowanie się sprzężenia przy starcie programu obróbki	CPMSTART [FAX] = "<Podtrzymanie Start>":	<div data-bbox="911 1417 1066 1509">"NONE"</div> <div data-bbox="911 1509 1066 1780">"ON"</div> <div data-bbox="1066 1417 1437 1509">Aktualny stan sprzężenia pozostaje zachowany.</div> <div data-bbox="1066 1509 1437 1780">Sprzężenie włączone. Wszystkie zdefiniowane powiązania osi wiodących zostaną aktywowane. Następuje to również wtedy, gdy wszystkie albo część tych powiązań jest już aktywnych, tzn. również przy całkowicie aktywnym sprzężeniu następuje nowa synchronizacja.</div>

Słowo kluczowe	Właściwość sprzężenia/Znaczenie	Składnia	
			"OF" Sprzężenie zostanie wyłączone. Jeśli dany moduł sprzężenia zostanie utworzony bez wyraźnej definicji (CPDEF), zostanie on usunięty. W innym przypadku pozostaje nadal tworzony, tzn. może być dalej używany.
			"DEL" Sprzężenie zostanie dezaktywowane, a następnie usunięte.
CPMPRT	Zachowanie się sprzężenia przy starcie programu obróbki pod przebiegiem szukania za pomocą testu programu	CPMPRT [FAx] = "<Podtrzymanie Start>" "<Podtrzymanie Start>":	patrz CPMSTART
CPLINTR	Wartość przesunięcia wartości wejściowej dla osi wiodącej	CPLINTR [FAx, LAx] = <wartość>	
CPLINSC	Współczynnik skalowania wartości wejściowej dla osi wiodącej	CPLINSC [FAx, LAx] = <wartość>	
CPLOUTTR	Wartość przesunięcia wartości wyjściowej sprzężenia	CPLOUTTR [FAx, LAx] = <wartość>	
CPLOUTSC	Współczynnik skalowania wartości wyjściowej sprzężenia	CPLOUTSC [FAx, LAx] = <wartość>	
CPSYNCOV	Wartość progowa dla ruchu synchronicznego pozycjonowania "zgrubnie"	CPSYNCOV [FAx] = <wartość>	
CPSYNFIP	Wartość progowa dla ruchu synchronicznego pozycjonowania "dokładnie"	CPSYNFIP [FAx] = <wartość>	
CPSYNCOV2	Druga wartość progowa dla ruchu synchronicznego pozycjonowania "zgrubnie"	CPSYNCOV2 [FAx] = <wartość>	
CPSYNFIP2	Druga wartość progowa dla ruchu synchronicznego pozycjonowania "dokładnie"	CPSYNFIP2 [FAx] = <wartość>	
CPSYNCOV	Wartość progowa dla ruchu synchronicznego szybkiego "zgrubnie"	CPSYNCOV [FAx] = <wartość>	
CPSYNFIV	Wartość progowa dla ruchu synchronicznego szybkiego "dokładnie"	CPSYNFIV [FAx] = <wartość>	

Słowo kluczowe	Właściwość sprzężenia/Znaczenie	Składnia										
CPMBRAKE	Zachowanie się osi nadążnej przy określonych sygnałach i rozkazach Stop	CPMBRAKE [FAx] =<wartość kodowana bitowo>										
CPMVDI	Zachowanie się osi nadążnej na określone sygnały interfejsu NC/PLC	CPMVDI [FAx] =<wartość kodowana bitowo>										
CPMALARM	Ukrywanie specjalnych, dotyczących sprzężenia wyprawdzeń alarmów	CPMALARM [FAx] =<wartość kodowana bitowo>										
CPSETTYPE	Typ sprzężenia	CPSETTYPE [FAx] = "<Typ sprzężenia>" "<Typ sprzężenia>": <table border="1" data-bbox="970 728 1442 1090"> <tr> <td>"CP"</td> <td>Dowolne programowanie</td> </tr> <tr> <td>"TRAIL"</td> <td>Typ sprzężenia "Nadążanie"</td> </tr> <tr> <td>"LEAD"</td> <td>Typ sprzężenia "Sprzężenie wartości wiodącej"</td> </tr> <tr> <td>"EG"</td> <td>Typ sprzężenia "Przekładnia elektroniczna"</td> </tr> <tr> <td>"COUP"</td> <td>Typ sprzężenia "Sprzężenie synchroniczne"</td> </tr> </table>	"CP"	Dowolne programowanie	"TRAIL"	Typ sprzężenia "Nadążanie"	"LEAD"	Typ sprzężenia "Sprzężenie wartości wiodącej"	"EG"	Typ sprzężenia "Przekładnia elektroniczna"	"COUP"	Typ sprzężenia "Sprzężenie synchroniczne"
"CP"	Dowolne programowanie											
"TRAIL"	Typ sprzężenia "Nadążanie"											
"LEAD"	Typ sprzężenia "Sprzężenie wartości wiodącej"											
"EG"	Typ sprzężenia "Przekładnia elektroniczna"											
"COUP"	Typ sprzężenia "Sprzężenie synchroniczne"											

FAx: Oś nadążna/ wrzecziono nadążne

LAx: Oś wiodąca/ wrzecziono wiodące

Wskazówka

Właściwości sprzężenia, które nie zostały wyraźnie zaprogramowane (w programie obróbki lub w akcjach synchronicznych), będą działały w oparciu o ich ustawienia standardowe.

W zależności od ustawienia słowa kluczowego CPSETTYPE zamiast ustawień standardowych (CPSETTYPE="CP") mogą działać również ustawione domyślnie właściwości sprzężenia.

Literatura

Odnosnie szczegółowych informacji odnośnie sprzężenia ogólnego patrz:

- Podręcznik działania Funkcje specjalne; M3: Ssprzężenia osi, rozdział: "Sprzężenie ogólne"

11.7 Sprzężenie Master/Slave (MASLDEF, MASLDEL, MASLON, MASLOF, MASLOFS)

Funkcja

"Sprzężenie Master/Slave" umożliwia:

- sprzężenie osi Slave z osią Master w czasie postoju uczestniczących osi.
- sprzężenie i rozłączenie wrzecion **obracających się**, o sterowanej prędkości obrotowej.
- projektowanie dynamiczne.

Wskazówka

W przypadku osi i wrzecion w trybie pozycjonowania sprzężenie jest załączane i rozłączane tylko w czasie postoju.

Składnia

```
MASLON(<Slave_1>, <Slave_2>, ...)  
MASLOF(<Slave_1>, <Slave_2>, ...)  
MASLOFS(<Slave_1>, <Slave_2>, ...)
```

Projektowanie dynamiczne:

```
MASLDEF(<Slave_1>, <Slave_2>, ..., <Master>)  
MASLDEL(<Slave_1>, <Slave_2>, ...)
```

Znaczenie

MASLON:	Włączenie sprzężenia tymczasowego Master/Slave
	<Slave_1>, ...: Osie Slave
	Osie, które w zespole Master/Slave mają być prowadzone przez oś Master
MASLOF:	Rozłączenie aktywnego sprzężenia Master/Slave
	<Slave_1>, ...: Osie Slave
MASLOFS:	Rozłączenie sprzężenia Master/Slave i automatyczne wyhamowanie wrzecion Slave (patrz wskazówka "Zachowanie się sprzężonych wrzecion w pracy ze sterowaną prędkością obrotową!")
	<Slave_1>, ...: Osie Slave
MASLDEF:	Utworzenie/zmiana zespołu Master/Slave z programu obróbki
	<Slave_1>, ...: Osie Slave
	<Master>: Oś Master
	Oś, która ma prowadzić oś Slave zdefiniowaną w zespole Master/Slave

MASLDEL: Rozłączenie sprzężenia Master/Slave i skasowanie definicji zespołu
<Slave_1>, ...: Osie Slave
Wskazówka:
Definicje Master/Slave, zaprojektowane w danych maszynowych, pozostają zachowane.

Wskazówka

Zachowanie się sprzężenia w przypadku wrzecion w trybie sterowania prędkością obrotową

W przypadku wrzecion w trybie sterowania prędkością obrotową zachowanie się `MASLON`, `MASLOF`, `MASLOFS` i `MASLDEL` pod względem sprzężenia jest ustalane explicite przez następującą daną maszynową:

MD37263 \$MA_MS_SPIND_COUPLING_MODE

W ustawieniu standardowym przy pomocy MD37263 = 0 sprzężenie i rozłączenie osi slave następuje wyłącznie w stanie zatrzymania uczestniczących osi. `MASLOFS` odpowiada `MASLOF`.

Przy MD37263 = 1 instrukcja sprzężenia jest wykonywana bezpośrednio, a przez to również w ruchu. Sprzężenie jest w przypadku `MASLON` natychmiast tworzone, a przy `MASLOFS` albo `MASLOF` natychmiast rozłączane. Obracające się w tym momencie wrzeciona Slave zachowują przy `MASLOF` swoje prędkości obrotowe, aż do ponownego zaprogramowania prędkości obrotowej. W przypadku `MASLOFS` są natomiast automatycznie hamowane.

Wskazówka

W przypadku `MASLOF/MASLOFS` odpada implicite zatrzymanie przebiegu wyprzedzającego. Ze względu na brak zatrzymania przebiegu wyprzedzającego zmienne systemowe \$P nie dają dla osi Slave zaktualizowanych wartości, aż do chwili ponownego zaprogramowania.

Wskazówka

Dla osi Slave wartość rzeczywista może przez `PRESETON` być synchronizowana na taką samą wartość osi Master. W tym celu trwale sprzężenie Master/Slave musi zostać na krótki czas rozłączone, aby wartość rzeczywistą nie bazowanej osi Slave przy pomocy POWER ON ustawić na wartość osi Master. Następnie trwale sprzężenie jest ponownie tworzone.

Sprzężenie trwale Master/Slave jest uaktywniane przez następującą MD:

MD37262 \$MA_MS_COUPLING_ALWAYS_ACTIVE = 1

Nie ma ono wpływu na polecenia językowe sprzężenia tymczasowego.

Przykłady

Przykład 1: Dynamiczne projektowanie sprzężenia Master/Slave

Dynamiczna konfiguracja sprzężenia Master/Slave z programu obróbki.
Oś właściwa po obróceniu pojemnika osi powinna stać się osią Master.

Kod programu	Komentarz
MASLDEF (AUX,S3)	; S3 master dla AUX
MASLON (AUX)	; Sprzężenie wł. dla AUX
M3=3 S3=4000	; Kierunek obróbki w prawo
MASLDEL (AUX)	; Skasowanie zaprojektowania i rozłączenie sprzężenia
AXCTSWE (CT1)	; Obrót pojemnika

Przykład 2: Sprzężenie wartości rzeczywistej osi Slave

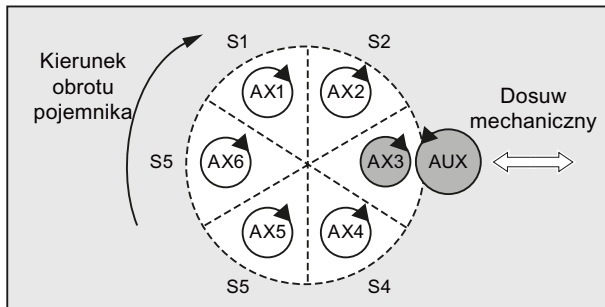
Sprzężenie wartości rzeczywistej osi Slave na taką samą wartość osi Master przez PRESETON. Przy permanentnym sprzężeniu Master/Slave ma na osi Slave zostać zmieniona wartość rzeczywista przez PRESETON.

Kod programu	Komentarz
N37262 \$MA_MS_COUPLING_ALWAYS_ACTIVE[AX2]=0	; Wyłączenie na krótko sprzężenia permanentnego.
N37263 NEWCONF	
N37264 STOPRE	
MASLOF (Y1)	; Sprzężenie tymczasowe wł.
N5 PRESETON (Y1,0,Z1,0,B1,0,C1,0,U1,0)	; Ustawienie wartości rzeczywistej nie wybazowanych osi Slave, ponieważ są one uaktywnione przy pomocy POWER ON.
N37262 \$MA_MS_COUPLING_ALWAYS_ACTIVE[AX2]=1	; Uaktywnienie sprzężenia permanentnego.
N37263 NEWCONF	

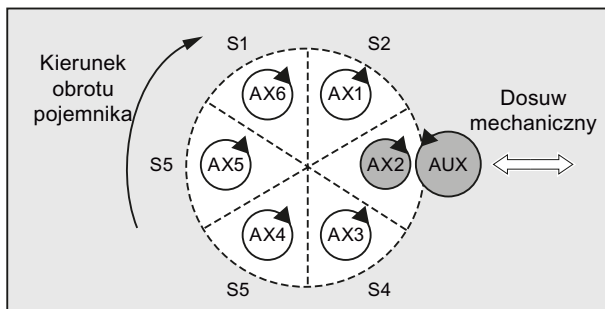
Przykład 3: Sekwencja sprzężenia położenie 3 / pojemnik CT1

Aby po obróceniu pojemnika mogło zostać dokonane sprzężenie z innym wrzecionem, przedtem stare sprzężenie musi zostać rozłączone, zaprojektowane skasowane i zaprojektowane nowe sprzężenie.

Sytuacja wyjściowa:



Po obróceniu o jeden slot:



Akcje synchroniczne

12.1 Definicja akcji synchronicznej

Akcja synchroniczna jest definiowana w bloku programu obróbki. W ramach tego bloku nie wolno programować żadnych innych poleceń, które nie są częścią składową akcji synchronicznej.

Akcja synchroniczna składa się z następujących elementów:

Część warunkowa				Część akcyjna		
Opcjonalnie Obowiązywanie, nr ident.	Opcjonalnie			Słowo kluczowe	Opcjonalnie: Funkcja G	Akcje
	Częstość	Opcjonalnie Funkcja G	Warunek			
n. p.	n. p.	G...	Wyrażenie logiczne	DO	G...	Akcja 1 ... Akcja n
ID=<Nr>	WHENEVER					
IDS=<Nr>	FROM					
	WHEN					
	EVERY					

n. p. = "nie podano"

Składnia

```
DO <akcja 1> ... <Akcja n>
<częstość> [<funkcja G>] <warunek> DO <akcja 1> ... <Akcja n>
ID=<Nr> <częstość> [<funkcja G>] <warunek> DO <akcja 1> ... <Akcja n>
IDS=<Nr> <częstość> [<funkcja G>] <warunek> DO <akcja 1> ... <Akcja n>
```

Literatura

Szczegółowy opis funkcjonalności akcji synchronicznych znajduje się w:

Podręcznik funkcjonowania Akcje synchroniczne

Ruch wahliwy

13.1 Ruch wahliwy asynchroniczny (OS, OSP1, OSP2, OST1, OST2, OSCTRL, OSNSC, OSE, OSB)

Funkcja

Oś wahliwa wykonuje ruch między dwoma punktami nawrotu 1 i 2 z podanym posuwem w jedną i drugą stronę, aż ruch ten zostanie wyłączony.

Inne osie mogą podczas ruchu wahliwego dowolnie interpolować. Poprzez ruch po torze albo przy pomocy osi pozycjonowania można uzyskać ciągły dosuw. Nie ma jednak przy tym **żadnego związku** między ruchem wahliwym i ruchem dosuwu.

Właściwości asynchronicznego ruchu wahliwego

- Asynchroniczny ruch wahliwy działa specyficznie dla osi poza granice bloku.
- Poprzez program obróbki jest zagwarantowane synchroniczne z blokiem włączenie ruchu wahliwego.
- Wspólna interpolacja wielu osi i nałożenie odcinków ruchu wahliwego są niemożliwe.

Programowanie

Poprzez następujące polecenia jest możliwe odpowiadające wykonywaniu programu NC włączanie i sterowanie asynchronicznym ruchem wahliwym z programu obróbki.

Zaprogramowane wartości są synchronicznie do bloków zapisywane w przebiegu głównym w odpowiednich danych ustawczych i działają, aż do następnej zmiany.

Składnia

```
OSP1 [<oś>] = <wartość>  OSP2 [<oś>] = <wartość>
OST1 [<oś>] = <wartość>  OST2 [<oś>] = <wartość>
FA [<oś>] = <wartość>
OSCTRL [<oś>] = (<opcja ustawiona>, <opcja cofnięta>)
OSNSC [<oś>] = <wartość>
OSE [<oś>] = <wartość>
OSB [<oś>] = <wartość>
OS [<oś>] = 1
OS [<oś>] = 0
```

Znaczenie

<os>	Nazwa osi wahliwej
OS	Włączenie, wyłączenie ruchu wahliwego: OS Wartość: 1 Włączenie ruchu wahliwego 0 Wyłączenie ruchu wahliwego
OSP1	Ustalenie pozycji ruchu nawrotnego 1
OSP2	Ustalenie pozycji ruchu nawrotnego 2 Wskazówka: Jeżeli jest aktywny ruch przyrostowy, wówczas pozycja jest obliczana przyrostowo względem ostatniej zaprogramowanej w programie NC, odpowiedniej pozycji nawrotu.
OST1	Ustalenie czasu zatrzymania w punkcie nawrotu 1 w [s]
OST2	Ustalenie czasu zatrzymania w punkcie nawrotu 2 w [s] <wartość>: -2 Interpolacja jest kontynuowana bez czekania na zatrzymanie dokładne -1 Czekanie na zatrzymanie dokładne zgrubnie 0 Czekanie na zatrzymanie dokładne dokładnie >0 Czekanie na zatrzymanie dokładne dokładnie, a następnie odczekanie podanego czasu zatrzymania Wskazówka: Jednostka czasu zatrzymania jest identyczna z czasem zatrzymania zaprogramowanym poprzez G4.
FA	Ustalenie prędkości posuwu Jako prędkość posuwu obowiązuje zdefiniowana prędkość posuwu osi pozycjonowania. Gdy prędkość posuwu nie jest zdefiniowana, obowiązuje wartość zapisana w danej maszynowej.
OSCTRL	Podanie opcji ustawienia i cofnięcia Wartości opcji 0 - 3 kodują zachowanie się w punktach nawrotu przy wyłączeniu. Można wybrać jeden z wariantów 0 - 3. Pozostałe ustawienia dają się według potrzeby kombinować z wybranym wariantem. Wiele opcji jest ze sobą łączonych przy pomocy znaku plus (+). <wartość>: 0 Przy wyłączeniu ruchu wahliwego zatrzymać w najbliższym punkcie nawrotu (ustawienie domyślne) Wskazówka: Możliwe tylko przez cofnięcie wartości 1 i 2. 1 Przy wyłączeniu ruchu wahliwego zatrzymanie w punkcie nawrotu 1 2 Przy wyłączeniu ruchu wahliwego zatrzymanie w punkcie nawrotu 2 3 Przy wyłączeniu ruchu wahliwego nie ma ruchu do punktu nawrotnego, w przypadku gdy skoki wykaszania nie są zaprogramowane 4 Po wykaszaniu ruch do pozycji końcowej

	8	Jeżeli ruch wahliwy zostanie przerwany przez skasowanie pozostałej drogi, powinny następnie zostać wykonane skoki wyskrzania i ew. ruch do pozycji końcowej.
	16	Jeżeli ruch wahliwy zostanie przerwany przez skasowanie pozostałej drogi, powinien jak przy wyłączeniu nastąpić ruch do odpowiedniej pozycji nawrotu.
	32	Zmieniony posuw jest aktywny dopiero od następnego punktu nawrotu
	64	FA równe 0, FA = 0: nałożenie drogi jest aktywne FA nierówne 0, FA <>0: nałożenie prędkości jest aktywne
	128	W przypadku osi obrotowej DC (najkrótsza droga)
	256	Skok wyskrzania jest wykonywany jako skok podwójny.(standard) 1=skok wyskrzania jest wykonywany jako skok pojedynczy.
	512	Najpierw wykonać ruch do pozycji startowej
OSNSC		Ustalenie liczby skoków wyskrzania
OSE		Ustalenie pozycji końcowej (w WKS), do której ma nastąpić ruch po wyłączeniu ruchu wahliwego Wskazówka: Przy zaprogramowaniu OSE działa dla OSCTRL implicite opcja 4.
OSB		Ustalenie pozycji startowej (w WKS), do której ma nastąpić ruch przed włączeniem ruchu wahliwego Ruch do pozycji startowej następuje przed punktem nawrotu 1. Jeżeli pozycja startowa jest zgodna z pozycją nawrotu 1, wówczas następnie następuje ruch do pozycji nawrotu 2. Przy osiągnięciu pozycji startowej nie działa czas zatrzymania, również gdy pozycja startowa jest zgodna z pozycją nawrotu 1, zamiast tego następuje czekanie na zatrzymanie dokładne dokładnie. Ustawiony warunek zatrzymania dokładnego jest dotrzymywany. Wskazówka: Aby nastąpił ruch do pozycji startowej, musi być ustawiona dana ustawcza SD43770 \$SA_OSCILL_CTRL_MASK bit 9.

Przykłady

Przykład 1: Oś ruchu wahliwego ma wykonywać ruch wahliwy między dwoma punktami nawrotu

Oś ruchu wahliwego Z ma wykonywać ruch wahliwy między 10 i 100. Ruch do punktu nawrotu 1 z zatrzymaniem dokładnym dokładnie, ruch do punktu nawrotu 2 z zatrzymaniem dokładnym zgrubnie. Posuw dla osi ruchu wahliwego ma wynosić 250. Na końcu obróbki mają nastąpić 3 skoki wyskrzania i oś ma dojść do pozycji końcowej 200. Posuw dla osi dosuwu ma wynosić 1, koniec dosuwu w kierunku X ma być osiągnięty przy pozycji 15.

Kod programu	Komentarz
WAITP(X, Y, Z)	; Położenie wyjściowe.
G0 X100 Y100 Z100	; Przełączenie na tryb osi pozycjonowania.
WAITP(X, Z)	

Kod programu	Komentarz
OSP1 [Z]=10 OSP2 [Z]=100	; Punkt nawrotny 1 < punkt nawrotny 2
OSE [Z]=200	; Pozycja końcowa.
OST1 [Z]=0 OST2 [Z]=-1	; Czas zatrzymania na U1: zatrzymanie dokładne dokładnie ; Czas zatrzymania na U2: zatrzymanie dokładne zgrubnie
FA [Z]=250 FA [X]=1	; Posuw w osi ruchu wahliwego, posuw w osi dosuwu.
OSCTRL [Z] = (4, 0)	; Opcje ustawiania.
OSNSC [Z] = 3	; 3 skoki wyiskrzania.
OS [Z] = 1	; ;Wystartowanie ruchu wahliwego
WHEN \$A_IN [3] == TRUE DO DELDTG (X)	; Skasowanie pozostałej drogi.
POS [X] = 15	; Położenia wyjściowe oś X
POS [X] = 50	Położenie końcowe oś X.
OS [Z] = 0	; Zatrzymanie ruchu wahliwego.
M30	

Wskazówka

Sekwencję poleceń OSP1 [Z] = . . . do OSNSC [Z] = . . . można również zaprogramować w jednym bloku.

Przykład 2: Ruch wahliwy ze zmianą online pozycji nawrotu

Dane ustawcze wymagane dla asynchronicznego ruchu wahliwego mogą zostać ustawione w danych ustawczych.

Gdy dane ustawcze zostaną zapisane bezpośrednio w programie obróbki, wówczas zmiana działa już w chwili przebiegu wyprzedzającego. Synchroniczne zachowanie się może zostać osiągnięte przez zatrzymanie przebiegu wyprzedzającego (STOPRE).

Kod programu	Komentarz
\$SA_OSCILL_REVERSE_POS1 [Z] = -10	
\$SA_OSCILL_REVERSE_POS2 [Z] = 10	
GO X0 Z0	
WAITP (Z)	
ID=1 WHENEVER \$AA_IM [Z] < \$\$AA_OSCILL_REVERSE_POS1 [Z] DO \$AA_OVR [X] = 0	; Gdy wartość rzeczywista
ID=2 WHENEVER \$AA_IM [Z] < \$\$AA_OSCILL_REVERSE_POS2 [Z] DO \$AA_OVR [X] = 0	osi ruchu wahliwego przekroczyła punkt nawrotny, oś dosuwu jest zatrzymywana.
OS [Z] = 1 FA [X] = 1000 POS [X] = 40	; Włączenie ruchu wahliwego.
OS [Z] = 0	; Wyłączenie ruchu wahliwego.
M30	

Dalsze informacje

Oś ruchu wahliwego

Dla ruchu wahliwego obowiązują:

- Każda oś może zostać użyta jako oś ruchu wahliwego.
- Równocześnie może być aktywnych wiele osi ruchu wahliwego (maksymalnie: liczba osi pozycjonowania).
- Dla osi ruchu wahliwego jest zawsze - niezależnie od polecenia G aktualnie obowiązującego w programie - aktywna interpolacja liniowa G1.

Oś ruchu wahliwego może:

- być osią wejściową dla transformacji dynamicznej
- być osią wiodącą w przypadku osi gantry i osi nadążnych
- wykonywać ruch:
 - bez ograniczenia przyspieszenia drugiego stopnia (**BRISK**)
lub
 - z ograniczeniem przyspieszenia drugiego stopnia (**SOFT**)
lub
 - z załamaną charakterystyką przyspieszenia (jak osie pozycjonowania)

Punkty nawrotu ruchu wahliwego

Przy ustalaniu pozycji ruchu wahliwego należy uwzględnić aktualne przesunięcia:

- Podanie absolutne

`OSP1[Z]=<wartość>`

Pozycja punkt nawrotny = suma przesunięć + wartość zaprogramowana

- Podanie względne

`OSP1[Z]=IC(<wartość>)`

Pozycja punktu nawrotnego = punkt nawrotny 1 + zaprogramowana wartość

Przykład:

Kod programu
N10 OSP1[Z]=100 OSP2[Z]=110
...
...
N40 OSP1[Z]=IC(3)

WAITP

Gdy ruch wahliwy ma być wykonywany przy pomocy osi geometrycznej, musi ona przy pomocy `WAITP` zostać zwolniona dla tego ruchu.

Po zakończonym ruchu wahliwym oś ruchu wahliwego jest przy pomocy `WAITP` ponownie wpisywana jako oś pozycjonowania i może być ponownie normalnie stosowana.

Ruch wahliwy z akcjami synchronicznymi ruchu i czasami zatrzymania

Po upływie ustawionych czasów zatrzymania następuje przy ruchu wahliwym wewnętrzna zmiana bloku (widać po nowych pozostałych drogach w osi). Przy zmianie bloku jest sprawdzana funkcja wyłączenia. Funkcja wyłączenia jest przy tym ustalana odpowiednio do ustawień sterujących przebiegiem ruchu (OSCTRL). *Na to zachowanie się pod względem czasu można wpływać przez korektor posuwu.*

Ewentualnie jest następnie wykonywany jeszcze jeden skok ruchu wahliwego, zanim zostaną wystartowane skoki wyiskrzania albo nastąpi ruch do pozycji końcowej. *Powstaje przy tym wrażenie, że zmienia się zachowanie się pod względem wyłączenia. Tak jednak nie jest.*

13.2 Ruch wahliwy sterowany przez akcje synchroniczne (OSCILL)**Funkcja**

Przy tego rodzaju ruchu wahliwym ruch dosuwu jest dozwolony tylko w punktach nawrotu lub w ramach zdefiniowanych obszarów nawrotu.

Zależnie od wymogu ruch wahliwy może podczas dosuwu

- być kontynuowany lub
- zostać zatrzymany, aż dosuw zostanie kompletnie wykonany.

Składnia

1. Ustalenie parametrów ruchu wahliwego
2. Zdefiniowanie akcji synchronicznych ruchu
3. Przyporządkowanie osi, ustalenie dosuwu

Znaczenie

OSP1[<os ruchu wahliwego>]=	Pozycja punktu nawrotu 1
OSP2[<os ruchu wahliwego>]=	Pozycja punktu nawrotu 2
OST1[<os ruchu wahliwego>]=	Czas zatrzymania w punkcie nawrotu 1 w sekundach
OST2[<os ruchu wahliwego>]=	Czas zatrzymania w punkcie nawrotu 2 w sekundach
FA[<os ruchu wahliwego>]=	Posuw w osi ruchu wahliwego
OSCTRL[<os ruchu wahliwego>]=	Opcje ustawienia wzgl. cofnięcia
OSNSC[<os ruchu wahliwego>]=...	Liczba skoków wyiskrzania
OSE[<os ruchu wahliwego>]=	Pozycja końcowa
WAITP(<os ruchu wahliwego>)	Zezwolenie na ruch wahliwy osi

Przyporządkowanie osi, dosuw

OSCILL[<oś ruchu wahliwego>]=(<oś dosuwu 1>,<oś dosuwu 2>,<oś dosuwu 3>)

POSP[<oś dosuwu>]=(<pozycja końcowa>,<długość częściowa>,<tryb>)

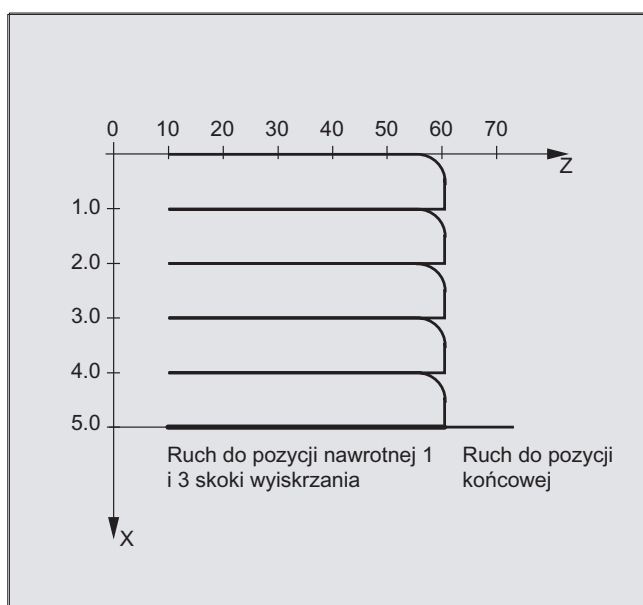
OSCILL:	Przyporządkowanie osi dosuwu do osi ruchu wahliwego
POSP:	Ustalenie dosuwów całkowitych i częściowych (patrz punkt "Zarządzanie plikami i programami")
Pozycja końcowa:	Pozycja końcowa dla osi dosuwu, po tym jak wszystkie dosuwy są wykonane.
Długość częściowa:	Wielkość dosuwu częściowego w punkcie nawrotnym / obszarze nawrotu
Tryb:	Podział dosuwu całkowitego na dosuwy częściowe = dwa skrawy resztowe o takiej samej wielkości (ustawienie domyślne); = wszystkie dosuwy częściowe o takiej samej wielkości

Akcje synchroniczne ruchu

WHEN... .. DO	jeżeli..., wówczas...
WHENEVER ... DO	zawsze gdy..., wówczas...

Przykład

W punkcie nawrotnym 1 dosuw nie powinien następować. W punkcie nawrotnym 2 dosuw powinien następować już w odstępie ii2 przed tym punktem 2 a oś ruchu wahliwego nie powinna w punkcie nawrotnym czekać na zakończenie dosuwu częściowego. Oś Z jest osią ruchu wahliwego, a oś X osią dosuwu.



1. parametr dla ruchu wahliwego

Kod programu	Komentarz
DEF INT ii2	; Zdefiniowanie zmiennej dla obrębu nawrotu 2
OSP1[Z]=10 OSP2[Z]=60	; Zdefiniowanie punktu nawrotu 1 i 2
OST1[Z]=0 OST2[Z]=0	; Punkt nawrotu 1: zatrzymanie dokładne dokładnie Punkt nawrotu 2: zatrzymanie dokładne dokładnie
FA[Z]=150 FA[X]=0,5	; Posuw oś ruchu wahliwego Z, posuw oś dosuwu X
OSCTRL[Z]=(2+8+16,1)	; Wyłączenie ruchu wahliwego w punkcie nawrotu 2; po RWL wyiskrzanie i ruch do pozycji końcowej; po RWL ruch do odpowiedniej pozycji nawrotu
OSNC[Z]=3	; Skoki wyiskrzania
OSE[Z]=70	; Pozycja końcowa = 70
ii2=2	; Ustawienie zakresu nawrotu
WAITP(Z)	; Zezwolenie na ruch wahliwy osi Z

2. Akcja synchroniczna ruchu

Kod programu	Komentarz
WHENEVER \$AA_IM[Z]<\$SA_OSCILL_REVERSE_POS2[Z] DO -> \$AA_OVR[X]=0 \$AC_MARKER[0]=0	; Zawsze, gdy aktualna pozycja osi ruchu wahliwego Z w MKS jest mniejsza, niż początek obszaru nawrotu 2, ustaw osiowy korektor osi dosuwu X na 0% i znacznik o indeksie 0 na wartość 0.
WHENEVER \$AA_IM[Z]>=\$SA_OSCILL_REVERSE_POS2[Z] DO \$AA_OVR[Z]=0	; Zawsze, gdy aktualna pozycja osi ruchu wahliwego Z w MKS jest większa albo równa pozycji nawrotu 2, ustaw osiowy korektor osi ruchu wahliwego Z na 0%.
WHENEVER \$AA_DTEPW[X]==0 DO \$AC_MARKER[0]=1	; Zawsze gdy pozostała droga dosuwu częściowego jest równa 0, ustaw znacznik o indeksie 0 na wartość 1.
WHENEVER \$AC_MARKER[0]==1 DO \$AA_OVR[X]=0 \$AA_OVR[Z]=100	; Zawsze gdy znacznik 0 jest równy 1, ustaw osiowy korektor osi dosuwu X na 0%. Przez to zapobiega się zbyt wczesnemu dosuwowi (oś ruchu wahliwego Z jeszcze nie opuściła obszaru nawrotu 2, oś dosuwu X jest jednak gotowa do ponownego dosuwu). Ustaw osiowy korektor osi ruchu wahliwego Z z 0% (akcja 2. akcji synchronicznej) w celu wykonania ruchu ponownie na 100%.

-> musi zostać zaprogramowane w jednym bloku

3. Uruchomienie ruchu wahliwego

Kod programu	Komentarz
OSCILL [Z] = (X) POSP [X] = (5, 1, 1)	; Wystartowanie osi Do osi ruchu wahliwego Z jest przyporządkowywana oś X jako oś dosuwu.
M30	Oś X powinna do pozycji 5 wykonywać ruch krokami po 1. ; Koniec programu

Opis

- 1. Ustalenie parametrów ruchu wahliwego**
Przed blokiem ruchu, który zawiera przyporządkowanie osi dosuwu i osi ruchu wahliwego, jak też ustalenie dosuwu, należy ustalić parametry ruchu wahliwego (patrz "Asynchroniczny ruch wahliwy").
- 2. Ustalenie akcji synchronicznych ruchu**
Poprzez warunki synchronizacji następuje:
Zatrzymanie dosuwu, aż oś ruchu wahliwego znajdzie się w ramach obszaru nawrotu (ii1, ii2) albo w punkcie nawrotu (U1, U2).
Zatrzymanie ruchu wahliwego podczas dosuwu w punkcie nawrotu.
Ponowne **uruchomienie ruchu wahliwego** po zakończonym dosuwie częściowym.
Ustalenie **startu następnego dosuwu częściowego**.
- 3. Przyporządkowanie osi wahliwej i osi dosuwu**, jak też ustalenie **dosuwu całkowitego i częściowego**.

Ustalenie parametrów ruchu wahliwego

Przyporządkowanie osi ruchu wahliwego i osi dosuwu: OSCILL

OSCILL[oś ruchu wahliwego] = (oś dosuwu1, oś dosuwu2, oś dosuwu3)

Przy pomocy polecenia `OSCILL` następują przyporządkowania osi i uruchomienie ruchu wahliwego.

Maksymalnie do jednej osi ruchu wahliwego mogą zostać przyporządkowane 3 osie dosuwu.

Wskazówka

Przed uruchomieniem ruchu wahliwego muszą być ustalone warunki synchronizacji dla zachowania się osi.

Ustalenie dosuwów: POSP

POSP[os dosuwu] = (poz. końcowa, długość częściowa, tryb)

Przy pomocy polecenia POSP sterowanie jest informowane o:

- Dosuw całkowity (przez pozycję końcową)
- Wielkość każdorazowego dosuwu częściowego w punkcie nawrotu lub w obszarze nawrotu
- Zachowanie się pod względem dosuwu częściowego przy osiągnięciu pozycji końcowej (przez tryb)

Tryb = 0	Dla obydwu ostatnich dosuwów częściowych następuje podział pozostałej drogi do punktu docelowego na 2 pozostałe kroki o takiej samej wielkości (ustawienie domyślne).
Tryb = 1	Wszystkie dosuwy częściowe mają taką samą wielkość. Są one obliczane z dosuwu całkowitego.

Ustalenie akcji synchronicznych ruchu

Wymienione poniżej akcje synchroniczne ruchu są najogólniej używane do wykonywania ruchu wahliwego.

Są tutaj przykładowe rozwiązania dla poszczególnych wymogów, które posłużą jako moduły do sporządzania specyficznych dla użytkownika ruchów wahliwych.

Wskazówka

W indywidualnym przypadku warunki synchronizacji mogą być również inaczej zaprogramowane.

Słowa kluczowe

WHEN ... DO ...	jeżeli..., wówczas...
WHENEVER ... DO	zawsze gdy..., wówczas...

Funkcje

Przy pomocy poniżej opisanych szczegółowo środków językowych można zrealizować następujące funkcje:

1. Dosuw w punkcie nawrotu.
2. Dosuw w obszarze nawrotu.
3. Dosuw w obydwu punktach nawrotu
4. Zatrzymanie ruchu wahliwego w punkcie nawrotu.
5. Ponownie uruchomienie ruchu wahliwego.
6. Niezbyt wczesny start dosuwu częściowego.

Dla wszystkich akcji synchronicznych przedstawionych tutaj jako przykład obowiązują założenia:

- Punkt nawrotu 1 < punkt nawrotu 2
- Z = oś ruchu wahliwego
- X = oś dosuwu

Wskazówka

Dla bliższych objaśnień patrz punkt "Akcje synchroniczne ruchu".

Przyporządkowanie osi ruchu wahliwego i osi dosuwu, jak też ustalenie dosuwu całkowitego i częściowego

Dosuw w obszarze nawrotu

Ruch dosuwu powinien rozpoczynać się w ramach obszaru nawrotu, zanim punkt nawrotu zostanie osiągnięty.

Te akcje synchroniczne zapobiegają ruchowi dosuwu, aż oś ruchu wahliwego znajdzie się w obszarze nawrotu.

Przy danych założeniach (patrz wyżej) wynikają następujące instrukcje:

Obszar nawrotu 1: Zawsze gdy aktualna pozycja osi ruchu wahliwego w MKS jest większa, niż początek obszaru nawrotu 1, ustaw korektor osi dosuwu na 0%.

```
WHENEVER
$AA_IM[Z]>$SA_OSCILL_RESERV
E_POS1[Z]+1i1 DO $AA_OVR[X]
=-0
```

Obszar nawrotu 2: Zawsze gdy aktualna pozycja osi ruchu wahliwego w MKS jest mniejsza, niż początek obszaru nawrotu 2, ustaw korektor osi dosuwu na 0%.

```
WHENEVER
$AA_IM[Z]<$SA_OSCILL_RESERV
E_POS2[Z]+1i2 DO $AA_OVR[X]
=-0
```

Dosuw w punkcie nawrotu

Dopóki oś ruchu wahliwego nie doszła do punktu nawrotu, nie następuje żaden ruch osi dosuwu.

Przy danych założeniach (patrz wyżej) wynikają następujące instrukcje:

Obszar nawrotu 1: Zawsze gdy aktualna pozycja osi ruchu wahliwego Z w MKS jest większa albo mniejsza, niż pozycja punktu nawrotu 1, ustaw korektor osi dosuwu X na 0% i korektor osi ruchu wahliwego Z na 100%.

```
WHENEVER
$AA_IM[Z]<>$SA_OSCILL_RESERV
VE_POS1[Z] DO $AA_OVR[X] =
0
→ $AA_OVR[Z] = 100
```

Obszar nawrotu 2: Zawsze gdy aktualna pozycja osi ruchu wahliwego Z w MKS jest większa albo mniejsza, niż pozycja punktu nawrotu 2, ustaw korektor osi dosuwu X na 0% i korektor osi ruchu wahliwego Z na 100%.

Dla punktu nawrotu2:

```
WHENEVER
$AA_IM[Z]<>$SA_OSCILL_RESERV
VE_POS2[Z] DO $AA_OVR[X] =
0
→ $AA_OVR[Z] = 100
```

Zatrzymanie ruchu wahliwego w punkcie nawrotu

Oś ruchu wahliwego jest zatrzymywana w punkcie nawrotu, równocześnie rozpoczyna się ruch dosuwu. Ruch wahliwy jest kontynuowany, gdy ruch dosuwu jest całkowicie wykonany.

Równocześnie ta akcja synchroniczna może zostać użyta do tego, by uruchomić ruch dosuwu, w przypadku gdy został on zatrzymany przez poprzednią akcję synchroniczną, która jeszcze działa.

Przy danych założeniach (patrz wyżej) wynikają następujące instrukcje:

Obszar nawrotu 1: Zawsze gdy aktualna pozycja osi ruchu wahliwego w MKS jest równa pozycji nawrotu 1, ustaw korektor osi ruchu wahliwego na 0% i korektor osi dosuwu na 100%

```
WHENEVER  
$SA_IM[Z]==$SA_OSCILL_RESER  
VE_POS1[Z] DO $AA_OVR[X] =  
0  
→ $AA_OVR[Z] = 100
```

Obszar nawrotu 2: Zawsze, gdy aktualna pozycja osi ruchu wahliwego Z w MKS jest równa pozycji nawrotu 2, ustaw korektor osi ruchu wahliwego X na 0% i korektor osi dosuwu na 100%.

```
WHENEVER  
$SA_IM[Z]==$SA_OSCILL_RESER  
VE_POS2[Z] DO $AA_OVR[X] =  
0  
→ $AA_OVR[Z] = 100
```

Ewaluacja online punktu nawrotu

Jeżeli po prawej stronie porównania znajduje się zmienna przebiegu głównego oznaczona przez \$\$, wówczas obydwie zmienne są na bieżąco ewaluowane w takcie IPO i porównywane ze sobą.

Wskazówka

Więcej informacji na ten temat patrz punkt "Akcje synchroniczne ruchu".

Ponownie uruchomienie ruchu wahliwego

Ta akcja synchroniczna ruchu jest używana do tego, by kontynuować ruch osi ruchu wahliwego, gdy ruch dosuwu częściowego jest zakończony.

Przy danych założeniach (patrz wyżej) wynikają następujące instrukcje:

```
WHENEVER $AA_DTEPW[X]==0 DO  
$AA_OVR[Z] =-100
```

Zawsze, gdy pozostała droga dla dosuwu częściowego w osi dosuwu X w WKS jest równa zero, ustaw korektor osi ruchu wahliwego na 100%.

Następny dosuw częściowy

Po dokonany dosuwie musi zostać uniemożliwiony zbyt wczesny start następnego dosuwu częściowego.

W tym celu jest stosowany specyficzny dla kanału znacznik (`$AC_MARKER[indeks]`), który jest ustawiany na końcu dosuwu częściowego (częściowa pozostała droga $\equiv 0$) i kasowany przy opuszczaniu obszaru nawrotu. Następnie przy pomocy akcji synchronicznej następuje uniemożliwienie następnego ruchu dosuwu.

Przy danych założeniach (patrz wyżej) wynikają np. dla punktu nawrotu 1 następujące instrukcje:

- | | |
|--|--|
| <p>1. Ustawienie znacznika:</p> <pre>WHENEVER \$AA_DTEPW[X]==0 DO \$AC_MARKER[1] = 1</pre> | <p>Zawsze gdy pozostała droga dla dosuwu częściowego w osi dosuwu X w WKS jest równa zero, ustaw znacznik o indeksie 1 na 1.</p> |
| <p>2. Skasowanie znacznika</p> <pre>WHENEVER \$AA_IM[Z]<> \$SA_OSCILL_RESERVE_POS1[Z] DO \$AC_MARKER[1] =-0</pre> | <p>Zawsze, gdy aktualna pozycja osi ruchu wahliwego Z w MKS jest większa, albo mniejsza niż pozycja punktu nawrotu 1, ustaw znacznik 1 na 0.</p> |
| <p>3. Uniemożliwienie dosuwu</p> <pre>WHENEVER \$AC_MARKER[1]==1 DO \$AA_OVR[X] = 0</pre> | <p>Zawsze gdy znacznik 1 jest taki sam, ustaw korektor osi dosuwu X na 0%.</p> |

Tłoczenie i cięcie

14.1 Uaktywnienie, wyłączenie aktywności

14.1.1 Tłoczenie i cięcie wł. lub wył. (SPOF, SON, PON, SONS, PONS, PDELAYON, PDELAYOF, PUNCHACC)

Funkcja

Uaktywnienie/wyłączenie aktywności tłoczenia i cięcia

Przy pomocy `PON` i `SON` jest uaktywniana funkcja tłoczenia wzgl. cięcia. `SPOF` powoduje zakończenie wszystkich funkcji specyficznych dla tłoczenia i cięcia. Modalnie działające polecenia `PON` i `SON` wykluczają się wzajemnie, tzn., `PON` powoduje wyłączenie aktywności `SON` i na odwrót.

Tłoczenie/cięcie z rekordem początkowym

Funkcje `SONS` i `PONS` również włączają funkcje tłoczenia wzgl. cięcia.

W przeciwieństwie do działającego w przypadku `SON/PON` sterowanie skokiem na płaszczyźnie interpolacji, w przypadku tych funkcji następuje sygnałowo-techniczne sterowanie wyzwaniem skoku na płaszczyźnie serwo. Dzięki temu można pracować z większymi częstotliwościami skoków, a przez to z wyższą wydajnością tłoczenia.

Podczas ewaluacji sygnału w rekordzie początkowym są zablokowane wszystkie funkcje, które prowadzą do zmiany zmiany pozycji osi cięcia albo tłoczenia (np. ruch kółkiem ręcznym, zmiana frame przez PLC, funkcje pomiarowe).

Tłoczenie ze zwłoką

`PDELAYON` powoduje opóźnione wyprowadzenie skoku tłoczenia. Modalnie działające polecenie spełnia funkcję przygotowawczą i dlatego z reguły znajduje się przed `PON`. Po `PDELAYOF` następuje normalne dalsze tłoczenie.

Wskazówka

Czas zwłoki jest ustawiany w danej ustawczej `SD42400 $SC_PUNCH_DWELLTIME`.

Przyśpieszenie zależne od drogi

Przy pomocy `PUNCHACC` można ustalić charakterystykę przyśpieszenia, która zależnie od odstępów otworów definiuje różne przyśpieszenia.

Drugi interfejs tłoczenia

Maszyny, które mają na przemian używać drugiego interfejsu tłoczenia (druga jednostka tłocząca lub podobne urządzenie), można przełączyć na drugą parę szybkich cyfrowych wejść i wyjść sterowania (para I/O). Dla obydwu interfejsów tłoczenia są dostępne wszystkie funkcje tłoczenia/cięcia. Przełączanie między pierwszym i drugim interfejsem tłoczenia następuje przez polecenia SPIF1 i SPIF2.

Wskazówka

Warunek: Poprzez dane maszynowe musi być zdefiniowana druga para I/O dla funkcji tłoczenia (→ patrz dane producenta maszyny!).

Składnia

```
PON G... X... Y... Z...
SON G... X... Y... Z...
SONS G... X... Y... Z...
PONS G... X... Y... Z...
PDELAYON
PDELAYOF
PUNCHACC (<Smin>, <Amin>, <Smax>, <Amax>)
SPIF1/SPIF2
SPOF
```

Znaczenie

PON	Uaktywnienie tłoczenia
SON	Uaktywnienie cięcia
PONS	Uaktywnienie tłoczenia z rekordem początkowym
SONS	Uaktywnienie cięcia z rekordem początkowym
SPOF	Wyłączenie aktywności tłoczenia/cięcia
PDELAYON	Uaktywnienie tłoczenia ze zwłoką
PDELAYOF	Wyłączenie aktywności tłoczenia ze zwłoką
PUNCHACC	Uaktywnienie przyśpieszenia zależnego od drogi
	Parametr:
<Smin>	Najmniejszy odstęp otworów
<Amin>	Przyśpieszenie początkowe
	<Amin> może być większe, niż <Amax>.
<Smax>	Największy odstęp otworów
<Amax>	Przyśpieszenie końcowe
	<Amax> może być większe, niż <Amin>.

SPIF1	Uaktywnienie pierwszego interfejsu tłoczenia Sterowanie skokami następuje przez pierwszą parę szybkich I/O.
SPIF2	Uaktywnienie drugiego interfejsu tłoczenia Sterowanie skokami następuje przez drugą parę szybkich I/O.

Wskazówka:
Po RESET albo rozruchu sterowania jest aktywny pierwszy interfejs tłoczenia. Gdy jest używany tylko jeden interfejs tłoczenia, nie musi być on programowany.

Przykłady

Przykład 1: Uaktywnienie cięcia

Kod programu	Komentarz
...	
N70 X50 SPOF	; Pozycjonowanie bez wyzwolenia tłoczenia.
N80 X100 SON	; Uaktywnienie cięcia, wyzwolenie skoku przed ruchem (X=50) i na końcu zaprogramowanego ruchu (X=100).
...	

Przykład 2: Tłoczenie ze zwłoką

Kod programu	Komentarz
...	
N170 PDELAYON X100 SPOF	; Pozycjonowanie bez wyzwolenia tłoczenia, uaktywnienie zwłocznego wyzwolenia tłoczenia.
N180 X800 PON	; Uaktywnienie tłoczenia. Po dojściu do pozycji końcowej skok tłoczenia jest wyprowadzany ze zwłoką.
N190 PDELAYOF X700	; Wyłączenie aktywności tłoczenia ze zwłoką, normalne wyzwolenie tłoczenia na końcu zaprogramowanego ruchu.
...	

Przykład 3: Tłoczenie z dwoma interfejsami stanu

Kod programu	Komentarz
...	
N170 SPIF1 X100 PON	; Na końcu bloku następuje wyzwolenie skoku na pierwszym szybkim wyjściu. Sygnał "skok aktywny" jest nadzorowany na pierwszym wejściu.
N180 X800 SPIF2	; Drugie wyzwolenie skoku następuje na drugim szybkim wyjściu. Sygnał "skok aktywny" jest nadzorowany na drugim wejściu.
N190 SPIF1 X700	; Sterowanie wszystkimi dalszymi skokami następuje przy pomocy pierwszego interfejsu.
...	

Dalsze informacje

Tłoczenie i cięcie z rekordem początkowym (PONS/SONS)

Tłoczenie i cięcie z rekordem początkowym nie jest możliwe równocześnie w wielu kanałach. PONS wzgl. SONS można uaktywnić tylko w jednym kanale.

Przyspieszenie zależne od drogi (PUNCHACC)

Przykład:

PUNCHACC (2, 50, 10, 100)

Odstępy otworów poniżej 2 mm:

Ruch odbywa się z przyspieszeniem 50% przyspieszenia maksymalnego.

Odstępy otworów od 2 mm do 10 mm:

Przyspieszenie rośnie do 100% proporcjonalnie do odstępów.

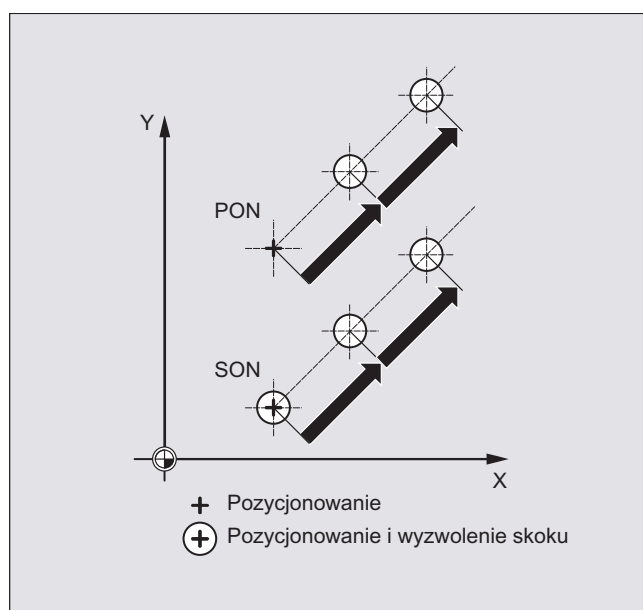
Odstępy otworów większe, niż 10 mm:

Ruch z przyspieszeniem 100%.

Wyzwolenie pierwszego skoku

Wyzwolenie pierwszego skoku po uaktywnieniu funkcji następuje w przypadku cięcia i tłoczenia różnie pod względem czasowym:

- PON/PONS:
 - Wszystkie skoki – również skok w pierwszym bloku po uaktywnieniu – następują na końcu bloku.
- SON/SONS:
 - Pierwszy skok po uaktywnieniu cięcia następuje, już na początku bloku.
 - Wszystkie dalsze skoki są wyzwolane każdorazowo na końcu bloku.



Tłoczenie i cięcie na miejscu

Wyzwolenie skoku następuje tylko wtedy, gdy blok zawiera informację o ruchu dla osi tłoczenia albo cięcia (osie aktywnej płaszczyzny).

Aby mimo to wyzwolić skok w tym samym miejscu, programuje się jedną z osi tłoczenia / cięcia z drogą ruchu 0.

Praca z użyciem narzędzi obrotowych**Wskazówka**

Aby narzędzia obrotowe wprowadzić stycznie na zaprogramowany tor, należy stosować sterowanie styczne.

Zastosowanie poleceń M

Przy pomocy techniki makr jest jak dotychczas możliwe używanie specjalnych funkcji M zamiast poleceń językowych (kompatybilność). Przy tym obowiązują następujące zgodności ze starymi systemami:

M20, M23	△	SPOF
M22	△	SON
M25	△	PON
M26	△	PDELAYON

Przykład pliku makr:

Kod programu	Komentarz
DEFINE M25 AS PON	; Tłoczenie wł.
DEFINE M125 AS PONS	; Tłoczenie z rekordem początkowym wł.
DEFINE M22 AS SON	; Cięcie wł.
DEFINE M122 AS SONS	; Cięcie z rekordem początkowym wł.
DEFINE M26 AS PDELAYON	; Tłoczenie ze zwłoką wł.
DEFINE M20 AS SPOF	; Tłoczenie, cięcie wł.
DEFINE M23 AS SPOF	; Tłoczenie, cięcie wł.

Przykład programowania:

Kod programu	Komentarz
...	
N100 X100 M20	; Pozycjonowanie bez wyzwolenia tłoczenia.
N110 X120 M22	; Uaktywnienie cięcia, przed i po ruchu wyzwolenie skoku.
N120 X150 Y150 M25	; Uaktywnienie tłoczenia, wyzwolenie skoku na końcu ruchu.
...	

14.2 Automatyczny podział drogi

Funkcja

Podział w odcinkach częściowych

Przy uaktywnionym tłoczeniu wzgl. cięciu zarówno SPP, jak też SPN powodują podział całkowitego odcinka ruchu, zaprogramowanego dla osi uczestniczących w tworzeniu konturu, na pewną liczbę odcinków częściowych o takiej samej długości (równoległy podział drogi). Wewnętrznie każdy odcinek częściowy odpowiada jednemu blokowi.

Liczba skoków

Przy tłoczeniu pierwszy skok następuje w punkcie końcowym pierwszego odcinka częściowego, przy cięciu natomiast - w punkcie startowym pierwszego odcinka częściowego. Na całym odcinku ruchu wynikają przez to następujące liczby:

Tłoczenie: liczba skoków = liczba odcinków częściowych

Cięcie: liczba skoków = liczba odcinków częściowych + 1

Funkcje pomocnicze

Funkcje pomocnicze są wykonywane w pierwszym z utworzonych bloków.

Składnia

SPP=

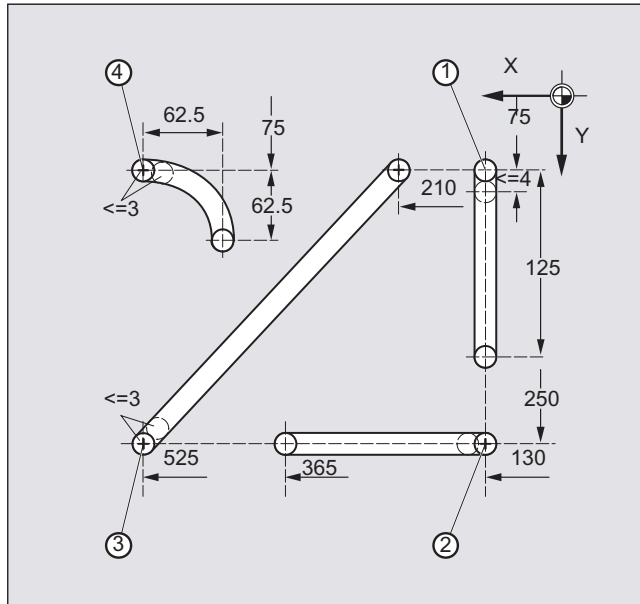
SPN=

Znaczenie

SPP	Wielkość odcinka częściowego (maksymalny odstęp skoków); działa modalnie
SPN	Liczba odcinków częściowych na blok; działa pojedynczymi blokami

Przykład 1

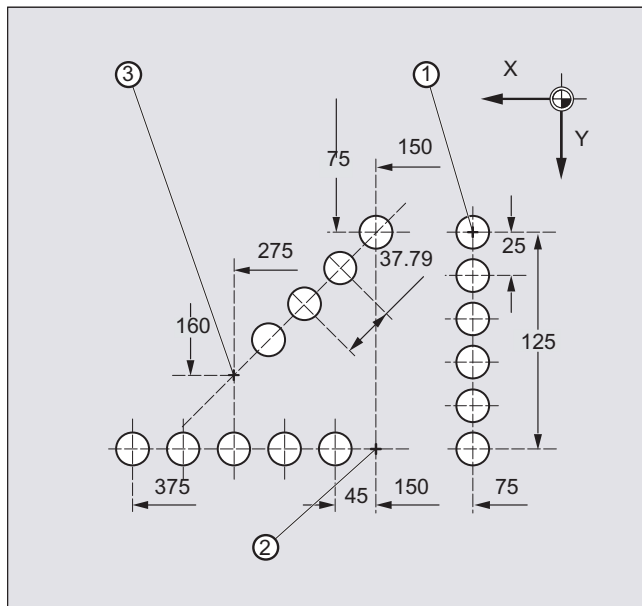
Zaprogramowane odcinki cięcia powinny automatycznie zostać podzielone na odcinki częściowe o takiej samej wielkości.



Kod programu	Komentarz
N100 G90 X130 Y75 F60 SPOF	; Pozycjonowanie na punkt startowy 1
N110 G91 Y125 SPP=4 SON	; Cięcie wł.; maksymalna długość odcinka podziałowego dla automatycznego podziału drogi: 4 mm
N120 G90 Y250 SPOF	; Cięcie wył.; pozycjonowanie na punkt startowy 2
N130 X365 SON	; Cięcie wł.; maksymalna długość odcinka podziałowego dla automatycznego podziału drogi: 4 mm
N140 X525 SPOF	; Cięcie wył.; pozycjonowanie na punkt startowy 3
N150 X210 Y75 SPP=3 SON	; Cięcie wł.; maksymalna długość odcinka podziałowego dla automatycznego podziału drogi: 3 mm
N160 X525 SPOF	; Cięcie wył.; pozycjonowanie na punkt startowy 4
N170 G02 X-62.5 Y62.5 I J62.5 SPP=3 SON	; Cięcie wł.; maksymalna długość odcinka podziałowego dla automatycznego podziału drogi: 3 mm
N180 G00 G90 Y300 SPOF	; Cięcie wył.

Przykład 2

Dla poszczególnych szeregów otworów powinien nastąpić automatyczny podział drogi. Dla podziału jest każdorazowo podawana maksymalna długość odcinka częściowego (wartość SPP).



Kod programu	Komentarz
N100 G90 X75 Y75 F60 PON	; Pozycjonowanie na punkt startowy 1; wytłoczenie pojedynczego otworu
N110 G91 Y125 SPP=25	; Maksymalna długość odcinka podziałowego dla automatycznego podziału drogi: 25 mm
N120 G90 X150 SPOF	; Tłoczenie wył.; pozycjonowanie na punkt startowy 2
N130 X375 SPP=45 PON	; Tłoczenie wł.; maksymalna długość odcinka podziałowego dla automatycznego podziału drogi: 45 mm
N140 X275 Y160 SPOF	; Tłoczenie wył.; pozycjonowanie na punkt startowy 3
N150 X150 Y75 SPP=40 PON	; Tłoczenie wł.; zamiast zaprogramowanej długości odcinka częściowego 40 mm jest stosowana obliczona długość odcinka 37,79 mm.
N160 G00 Y300 SPOF	; Tłoczenie wył.; pozycjonowanie

14.2.1 Podział drogi w przypadku osi uczestniczących w tworzeniu konturu

Długość odcinka częściowego SPP

Przy pomocy SPP podaje się maksymalny odstęp skoków, a przez to maksymalną długość odcinków częściowych, na które całkowity odcinek ruchu ma zostać podzielony. Wyłączenie polecenia następuje przy pomocy $SPOF$ albo $SPP=0$.

Przykład:

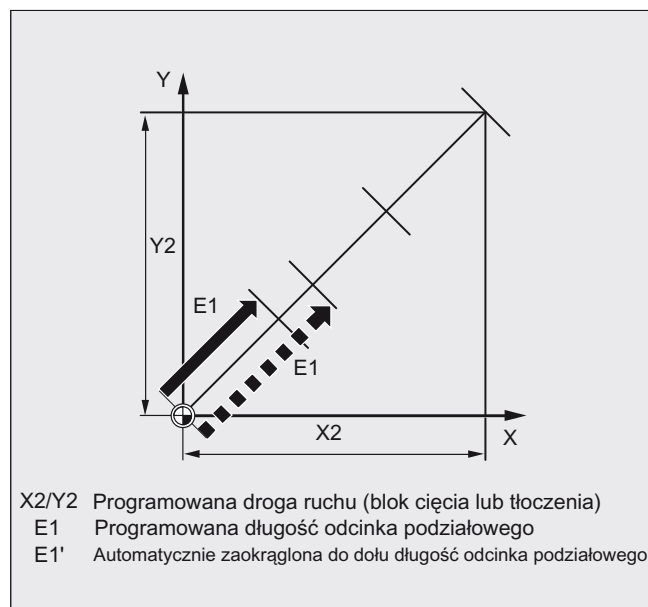
```
N10 SON X0 Y0
```

```
N20 SPP=2 X10
```

Całkowita droga ruchu 10 mm jest dzielona na 5 odcinków po 2 mm ($SPP=2$).

Wskazówka

Podział drogi przy pomocy SPP następuje zawsze równolegle: wszystkie odcinki częściowe mają taką samą długość. Oznacza to, że zaprogramowana wielkość odcinka częściowego (wartość SPP) obowiązuje tylko wtedy, gdy iloraz drogi całkowitej i wartości SPP jest liczbą całkowitą. Jeżeli tak nie jest, wówczas wielkość odcinka częściowego jest tak redukowana, że jest uzyskiwany iloraz całkowitoliczbowy.



Przykład:

```
N10 G1 G91 SON X10 Y10
```

```
N20 SPP=3.5 X15 Y15
```

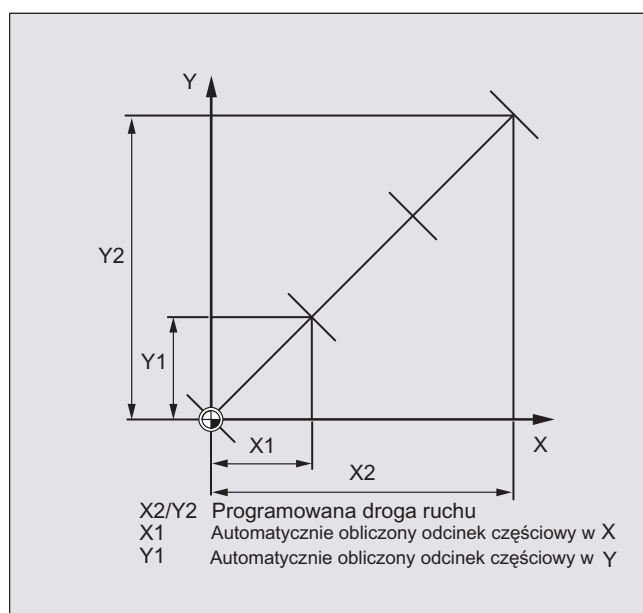
W przypadku drogi całkowitej 15 mm i odcinka częściowego 3,5 mm uzyskuje się iloraz nie całkowitoliczbowy (4.28). Przez to następuje zmniejszenie wartości SPP do najbliższego możliwego ilorazu całkowitoliczbowego. W tym przypadku uzyskuje się długość odcinka częściowego wynoszącą 3 mm.

Liczba odcinków częściowych SPN

Przy pomocy SPN definiuje się liczbę odcinków częściowych, która ma zostać utworzona z całkowitej drogi ruchu. Długość odcinków częściowych jest obliczana automatycznie. Ponieważ SPN działa pojedynczymi blokami, musi przedtem przy pomocy PON albo SON zostać uaktywnione tłoczenie albo cięcie.

SPP i SPN w tym samym bloku

Jeżeli w tym samym bloku zaprogramuje się zarówno długość odcinka częściowego (SPP), jak też liczbę takich odcinków (SPN), wówczas dla tego bloku obowiązuje SPN , dla wszystkich dalszych SPP . Jeżeli SPP uaktywniono już przed SPN , wówczas po bloku z SPN jest ono ponownie aktywne.



Wskazówka

O ile tłoczenie/cięcie jest zasadniczo dostępne w sterowaniu, programowanie automatycznego podziału drogi przy pomocy SPN lub SPP daje się uaktywnić również niezależnie od tej technologii.

14.2.2 Podział drogi w przypadku pojedynczych osi

Jeżeli oprócz osi uczestniczących w tworzeniu konturu również pojedyncze osie są zdefiniowane jako oś tłoczenia-cięcia, wówczas również one mogą podlegać automatycznemu podziałowi drogi.

Zachowanie się pojedynczej osi w przypadku SPP

Zaprogramowana długość odcinka częściowego (S_{PP}) odnosi się w zasadzie do osi uczestniczących w tworzeniu konturu. Dlatego w bloku, w którym oprócz ruchu pojedynczej osi i wartości SPP nie jest zaprogramowana żadna oś uczestnicząca w tworzeniu konturu, wartość SPP jest ignorowana.

Jeżeli w bloku jest zaprogramowana zarówno pojedyncza oś, jak też oś uczestnicząca w tworzeniu konturu, wówczas zachowanie się pojedynczej osi zależy od ustawienia odpowiedniej danej maszynowej.

1. Ustawienie standardowe

Droga pojedynczej osi jest dzielona równomiernie na bloki pośrednie wytwarzane przez SPP.

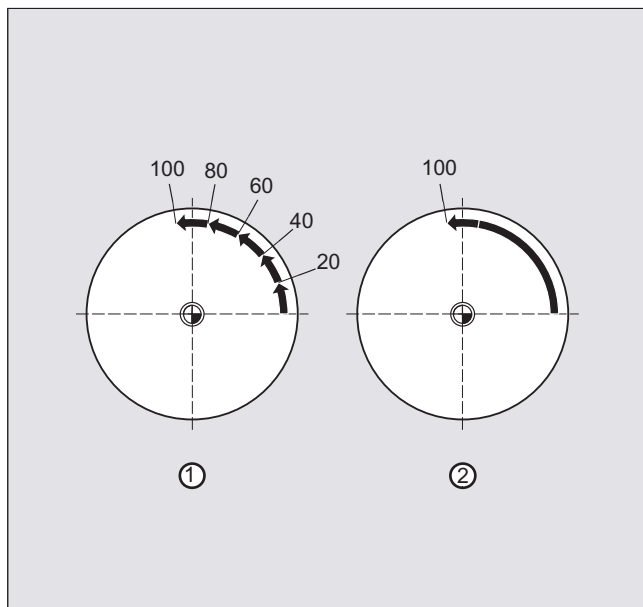
Przykład:

```
N10 G1 SON X10 A0
```

```
N20 SPP=3 X25 A100
```

Poprzez odcinek skoku 3 mm jest przy całkowitej drodze ruchu osi X (oś uczestnicząca w tworzeniu konturu) wynoszącej 15 mm wytwarzanych 5 bloków.

Oś A obraca się przez to w każdym bloku o 20° .



1. Pojedyncza oś bez podziału drogi

Pojedyncza oś przebywa swoją całkowitą drogę w pierwszym z wytworzonych bloków.

2. Różny podział drogi

Zachowanie się pojedynczej osi jest zależne od interpolacji osi uczestniczących w tworzeniu konturu:

- Interpolacja kołowa: Podział drogi
- Interpolacja liniowa: Brak podziału drogi

Zachowanie się w przypadku SPN

Zaprogramowana liczba odcinków częściowych obowiązuje również, gdy nie jest równocześnie zaprogramowana oś uczestnicząca w tworzeniu konturu.

Warunek: pojedyncza oś jest zdefiniowana jako oś tłoczenia-cięcia.

Szlifowanie

15.1 Specyficzny dla szlifowania nadzór narzędzia w programie obróbki (TMON, TMOF)

Funkcja

Przy pomocy polecenia TMON można dla narzędzi szlifierskich (typ 400 - 499) uaktywnić w programie NC nadzór geometrii i prędkości obrotowej. Nadzór pozostaje aktywny, aż zostanie w programie obróbki wyłączony przez polecenie TMOF.

Wskazówka

Proszę przestrzegać danych producenta maszyny!

Warunek

Specyficzne dla szlifowania parametry narzędzia \$TC_TPG1 do \$TC_TPG9 muszą być ustawione.

Składnia

TMON (<T-Nr. >)
TMOF (<T-Nr. >)

Znaczenie

TMON	Polecenie do włączenia specyficznego dla szlifowania nadzoru narzędzia
TMOF	Polecenie do wyłączenia specyficznego dla szlifowania nadzoru narzędzia
<nr T>	Podanie numeru T
	Wskazówka:
	Konieczne tylko wtedy, gdy narzędzie o tym numerze T nie jest aktywne.
TMOF (0)	Wyłączenie nadzoru dla wszystkich narzędzi

Dalsze informacje

Parametry narzędzia specyficzne dla szlifowania

Parametry	Znaczenie	Typ danych
\$TC_TPG1	Numer wrzeciona	INT
\$TC_TPG2	Instrukcja powiązania Są automatycznie utrzymywane identyczne parametry dla lewej i prawej strony ściernicy.	INT
\$TC_TPG3	Minimalny promień ściernicy	REAL
\$TC_TPG4	Minimalna szerokość ściernicy	REAL
\$TC_TPG5	Aktualna szerokość ściernicy	REAL
\$TC_TPG6	Maksymalna prędkość obrotowa	REAL
\$TC_TPG7	Maksymalna prędkość obwodowa	REAL
\$TC_TPG8	Kąt ściernicy skośnej	REAL
\$TC_TPG9	Numer parametru dla obliczenia promienia	INT

Literatura:

Podręcznik działania Funkcje podstawowe; Korekcja narzędzia (W1)

Włączenie nadzoru narzędzia przez wybór narzędzia

W zależności od danej maszynowej można dla narzędzi szlifierskich (typ 400 - 499) włączać nadzór narzędzia razem z jego wyborem

W każdej chwili może dla każdego wrzeciona być aktywny tylko **jeden** nadzór.

Nadzór geometrii

Nadzorowany jest aktualny promień ściernicy i jej aktualna szerokość.

Nadzorowanie wartości zadanej prędkości obrotowej na wartość graniczną następuje cyklicznie przy uwzględnieniu korektora wrzeciona.

Jako wartość graniczna prędkości obrotowej obowiązuje mniejsza wartość, która przy porównaniu prędkości maksymalnej z prędkością obliczoną wynika z maksymalnej prędkości obwodowej ściernicy i aktualnego promienia ściernicy.

Prace bez numeru T i numeru D

Dla danej maszynowej można ustawić numer **T** i standardowy numer **D**,

który nie musi już być programowany i działa po power on / reset.

Przykład: Praca z tą samą ściernicą

Poprzez daną maszynową można ustawić, że aktywne narzędzie przy zresetowaniu pozostaje zachowane (patrz Dowolne nadawanie numerów D, numer ostrza (Strona 430)).

Dalsze funkcje

16.1 Funkcje osi (AXNAME, AX, SPI, AXTOSPI, ISAXIS, AXSTRING, MODAXVAL)

Funkcja

`AXNAME` jest stosowane np. przy sporządzaniu ogólnie obowiązujących cykli, gdy nazwy osi nie są znane.

`AX` jest stosowane do pośredniego programowania osi geometrycznych i synchronicznych. Identyfikator osi jest przy tym zapisywany w zmiennej typu `AXIS` lub dawany przez polecenie jak `AXNAME` albo `SPI`.

`SPI` jest stosowane, gdy są programowane funkcje osi dla wrzeciona, np. wrzeciona synchronicznego.

`AXTOSPI` jest stosowane, aby zamienić identyfikator osi na indeks wrzeciona (funkcja odwrotna do `SPI`).

`AXSTRING` jest stosowane, aby zamienić identyfikator osi (typ danych `AXIS`) na łańcuch znaków (funkcja odwrotna do `AXNAME`).

`ISAXIS` jest stosowane w powszechnie obowiązujących cyklach, aby zapewnić, że określona oś geometryczna występuje, a przez to następne wywołanie `$P_AXNX` nie zostanie przerwane z błędem.

`MODAXVAL` jest stosowane, aby w przypadku osi obrotowych modulo określić pozycję modulo.

Składnia

```
AXNAME("String")
AX[AXNAME("String")]
SPI(n)

AXTOSPI(A) albo AXTOSPI(B) albo AXTOSPI(C)
AXSTRING(SPI(n))
ISAXIS(<numer osi geometrycznej>)
<pozycja modulo>=MODAXVAL(<oś>,<pozycja osi>)
```

Znaczenie

AXNAME	Konwertuje wejściowy ciąg znaków na identyfikator osi; wejściowy ciąg znaków musi zawierać obowiązującą nazwę osi.
AX	Zmienny identyfikator osi
SPI	Konwertuje numer wrzeciona na identyfikator osi; przekazany parametr musi zawierać poprawny numer wrzeciona.
n	Numer wrzeciona
AXTOSPI	Konwertuje identyfikator osi na indeks wrzeciona typu Integer. AXTOSPI odpowiada funkcji odwrotnej do SPI.
X, Y, Z	Identyfikator osi typu AXIS jako zmienna albo stała
AXSTRING	Jest wyprowadzany łańcuch znaków z przynależnym numerem wrzeciona.
ISAXIS	Sprawdza, czy istnieje podana oś geometryczna.
MODAXVAL	W przypadku osi obrotowych modulo określa pozycję modulo; odpowiada ona reszcie modulo względem sparametryzowanego zakresu modulo (wynosi w ustawieniu standardowym 0 do 360 stopni; przez MD30340 MODULO_RANGE_START i MD30330 \$MA_MODULO_RANGE można zmienić początek i wielkość zakresu modulo).

Wskazówka

Rozszerzenia SPI

Funkcja osi SPI(n) może być teraz stosowana również do odczytu i zapisu komponentów frame. Dzięki temu mogą być zapisywane frame np. o składni `$P_PFRAME[SPI(1),TR]=2.22`.

Przez dodatkowe zaprogramowanie pozycji osi przez adres `AX[SPI(1)]=<pozycja osi>` może zostać wykonany ruch w osi. Warunkiem jest, by wrzeciono znajdowało się w trybie pozycjonowania albo pracy jako oś.

Przykłady

Przykład 1: AXNAME, AX, ISAXIS

Kod programu	Komentarz
OVRA[AXNAME("oś poprzeczna")]=10	; Korektor dla osi poprzecznej
AX[AXNAME("oś poprzeczna")]=50.2	; Pozycja końcowa dla osi poprzecznej
OVRA[SPI(1)]=70	; Korektor dla wrzeciona 1
AX[SPI(1)]=180	; Pozycja końcowa dla wrzeciona 1
IF ISAXIS(1) == FALSE GOTOF DALEJ	; Czy jest odcięta?
AX[\$P_AXN1]=100	; Ruch w odciętej wykonany
DALEJ:	

Przykład 2: AXSTRING

Przy programowaniu przy pomocy AXSTRING[SPI(n)] nie jest już jako numer wrzeczona wyprowadzany indeks osi, której wrzeczono jest przyporządkowane, lecz łańcuch znaków "Sn".

Kod programu	Komentarz
AXSTRING[SPI(2)]	; Jest wyprowadzany łańcuch znaków "S2".

Przykład 3: MODAXVAL

Ma zostać określona pozycja modulo osi obrotowej modulo A.

Wartością wyjściową dla obliczenia jest pozycja osi 372.55.

Sparametryzowany zakres modulo wynosi 0 do 360 stopni:

MD30340 MODULO_RANGE_START = 0

MD30330 \$MA_MODULO_RANGE = 360

Kod programu	Komentarz
R10=MODAXVAL(A,372.55)	; Obliczona pozycja modulo R10 = 12.55.

Przykład 4: MODAXVAL

Jeżeli programowany identyfikator osi nie odnosi się do osi obrotowej modulo, wówczas będąca do konwersji wartość (<pozycja osi>) jest zwracana bez zmian.

Kod programu	Komentarz
R11=MODAXVAL(X,372.55)	; X jest osią liniową; R11 = 372.55.

16.2 Przełączalne osie geometryczne (GEOAX)

Funkcja

Przy pomocy funkcji "Przełączane osie geometryczne" można z programu obróbki zmienić zespół osi geometrycznych skonfigurowany przez dane maszynowe. Przy tym oś kanału zdefiniowana jako synchroniczna oś dodatkowa może zastąpić dowolną oś geometryczną.

Składnia

GEOAX (<n>, <oś kanałowa>, <n>, <oś kanałowa>, <n>, <oś kanałowa>)
 GEOAX ()

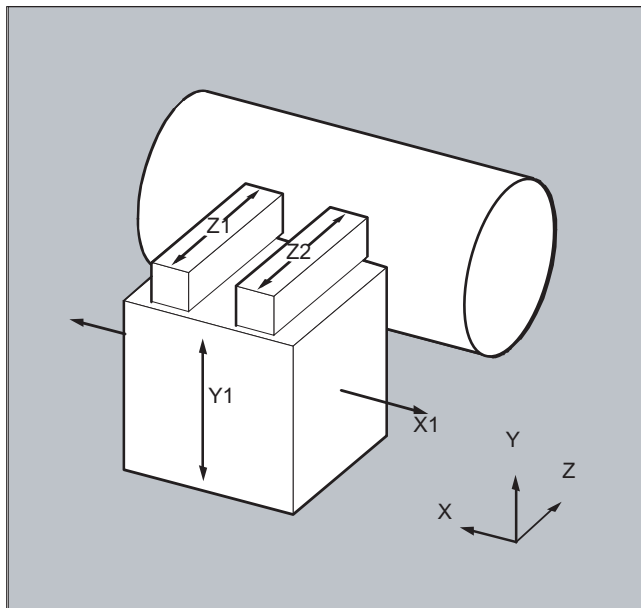
Znaczenie

GEOAX (...)	Polecenie do przełączania osi geometrycznych Wskazówka: GEOAX () bez podania parametrów wywołuje konfigurację podstawową osi geometrycznych.
<n>	Przy pomocy tego parametru jest podawany numer osi geometrycznej, której ma zostać przyporządkowana podana następnie oś kanału. Zakres wartości: 1, 2 lub 3 Wskazówka: Przy pomocy <n>=0 można bez zastępowania usunąć podaną następnie oś kanałową z zespołu osi geometrycznych.
<oś kanałowa>	Przy pomocy tego parametru jest podawana nazwa osi kanałowej, która ma zostać przyjęta do zespołu osi geometrycznych.

Przykłady

Przykład 1: Włączanie dwóch osi na przemian jako oś geometryczna

Ruchy saniami narzędziowymi mogą być wykonywane przez osie kanałowe X1, Y1, Z1, Z2:



Osie geometryczne są tak zaprojektowane, że po włączeniu najpierw działa Z1 jako 3. oś geometryczna pod nazwą "Z" i razem z X1 i Y1 tworzy zespół osi geometrycznych.

W programie obróbki teraz osie Z1 i Z2 mają na przemian być używane jako oś geometryczna Z:

Kod programu	Komentarz
...	
N100 GEOAX(3,Z2)	; Jako 3. oś geometryczna (Z) działa oś kanałowa Z2.
N110 G1 ...	
N120 GEOAX(3,Z1)	; Jako 3. oś geometryczna (Z) działa oś kanałowa Z1.
...	

Przykład 2: Przełączanie osi geometrycznych przy 6 osiach kanału

Maszyna posiada 6 osi kanałowych o nazwach XX, YY, ZZ, U, V, W.

Podstawowe ustawienie konfiguracji osi geometrycznych przez dane maszynowe jest następujące:

oś kanałowa XX = 1. oś geometryczna (oś X)

oś kanałowa YY = 2. oś geometryczna (oś Y)

oś kanałowa ZZ = 3. oś geometryczna (oś Z)

Kod programu	Komentarz
N10 GEOAX()	; Konfiguracja podstawowa osi geometrycznych działa.
N20 G0 X0 Y0 Z0 U0 V0 W0	; Wszystkie osie przesuwem szybkim do pozycji 0.
N30 GEOAX(1,U,2,V,3,W)	; Oś kanałowa U staje się pierwszą (X), V drugą (Y), a W trzecią osią geometryczną (Z).
N40 GEOAX(1,XX,3,ZZ)	; Oś kanałowa XX staje się pierwszą (X), ZZ trzecią osią geometryczną (Z). Oś kanału V pozostaje drugą osią geometryczną (Y).
N50 G17 G2 X20 I10 F1000	; Okrąg w płaszczyźnie X/Y. Ruch wykonują osie kanałowe XX i V.
N60 GEOAX(2,W)	; Oś kanału W staje się drugą osią geometryczną (Y).
N80 G17 G2 X20 I10 F1000	; Okrąg w płaszczyźnie X/Y. Ruch wykonują osie kanałowe XX i W.
N90 GEOAX()	; Cofnięcie do stanu podstawowego.
N100 GEOAX(1,U,2,V,3,W)	; Oś kanałowa U staje się pierwszą (X), V drugą (Y), a W trzecią osią geometryczną (Z).
N110 G1 X10 Y10 Z10 XX=25	; Każda z osi kanału U, V, W wykonuje ruch do pozycji 10. XX jako oś dodatkowa wykonuje ruch do pozycji 25.
N120 GEOAX(0,V)	; V jest wyłączana z zespołu osi geometrycznych. U i W są nadal pierwszą (X) i trzecią osią geometryczną (Z). Druga oś geometryczna (Y) pozostaje nie zajęta.
N130 GEOAX(1,U,2,V,3,W)	; Oś kanałowa U pozostaje pierwszą (X), V staje się drugą (Y), W pozostaje trzecią osią geometryczną (Z).
N140 GEOAX(3,V)	; V staje się trzecią osią geometryczną (Z), przy czym W jest zastępowana, a przez to usuwana z zespołu osi geometrycznych. Druga oś geometryczna (Y) jest jak dotychczas nie zajęta.

Wskazówka

Konfiguracja osi

Przyporządkowanie między osiami geometrycznymi, osiami dodatkowymi, osiami kanałowymi i osiami maszyny, jak też ustalenie nazw poszczególnych typów osi jest dokonywane poprzez następujące dane maszynowe:

MD20050 \$MC_AXCONF_GEOAX_ASSIGN_TAB (przyporządkowanie osi geometrycznej do osi kanałowej)

MD20060 \$MC_AXCONF_GEOAX_NAME_TAB (nazwa osi geometrycznej w kanale)

MD20070 \$MC_AXCONF_MACHAX_USED (numer osi maszyny obowiązujący w kanale)

MD20080 \$MC_AXCONF_CHANAX_NAME_TAB (nazwa osi kanałowej w kanale)

MD10000 \$MN_AXCONF_MACHAX_NAME_TAB (nazwa osi maszyny)

MD35000 \$MA_SPIND_ASSIGN_TO_MACHAX (przyporządkowanie wrzeciona do osi maszyny)

Literatura:

Podręcznik działania Funkcje podstawowe; Osie, Układy współrzędnych frame (K2)

Ograniczenia

- Przełączenie osi geometrycznych jest niemożliwe w przypadku:
 - aktywnej transformacji
 - aktywnej interpolacji spline
 - aktywnej korekcji promienia narzędzia
 - aktywnej korekcji dokładnej narzędzia
- Jeżeli oś geometryczna i oś kanałowa mają taką samą nazwę, zmiana osi geometrycznej jest niemożliwa.
- Żadna z osi uczestniczących w przełączeniu nie może uczestniczyć w akcji, która może trwać poza granice bloku.
- Przy pomocy polecenia `GEOAX` mogą zostać zastąpione tylko osie geometryczne już istniejące przy włączeniu (a więc nie mogą być definiowane nowe).
- Zamiana osi przy pomocy `GEOAX` podczas przygotowywania tablicy konturu (`CONTPRON`, `CONTDCON`) prowadzi do alarmu.

Warunki brzegowe

Stan osi po zastąpieniu

Oś zastąpiona przez przełączenie w zespole osi geometrycznych jest po procesie przełączenia poprzez swoją nazwę osi kanałowej możliwa do programowania jako oś dodatkowa.

Frame, obszary ochrony, ograniczenia obszaru pracy

Z przełączeniem osi geometrycznych są kasowane wszystkie frame, obszary ochrony i ograniczenia obszaru pracy.

Współrzędne biegunowe

Zamiana osi geometrycznych przy pomocy `GEOAX` ustawia analogicznie do zmiany płaszczyzny przy pomocy `G17-G19` modalne współrzędne biegunowe na wartość 0.

DRF, NPV

Ewentualne przesunięcie kółkiem ręcznym (DRF) albo zewnętrzne przesunięcie punktu zerowego (NPV) nadal działa po przełączeniu.

Konfiguracja podstawowa osi geometrycznych

Polecenie `GEOAX ()` wywołuje podstawową konfigurację zespołu osi geometrycznych.

Po POWER ON i przy przełączeniu na tryb pracy "bazowanie do punktu odniesienia" następuje automatycznie przełączenie z powrotem na konfigurację podstawową.

Korekcja długości narzędzia

Aktywna korekcja długości narzędzia działa również po procesie przełączenia. Obowiązuje ona jednak dla nowo przyjętych albo zmienionych pod względem pozycji osi geometrycznych jako jeszcze nie zrealizowana. Przy pierwszym poleceniu ruchu dla tej osi geometrycznej wynikowa droga ruchu składa się w związku z tym z sumy korekcji długości narzędzia i zaprogramowanej drogi ruchu.

Osie geometryczne, które przy przełączeniu zachowują swoją pozycję w zespole osi, zachowują również swój status odnośnie korekcji długości narzędzia.

Konfiguracja osi geometrycznych przy aktywnej transformacji

Konfiguracja osi geometrycznych obowiązująca w aktywnej transformacji (ustalona poprzez dane maszynowe) nie daje się zmienić poprzez funkcję "przełączane osie geometryczne".

Jeżeli w związku z transformacjami jest konieczność zmiany konfiguracji osi geometrycznych, wówczas jest to możliwe tylko przez kolejną transformację.

Zmieniona przez `GEOAX` konfiguracja osi geometrycznych jest kasowana przez uaktywnienie transformacji.

Jeżeli ustawienia danych maszynowych dla transformacji i dla przełączania osi geometrycznych są ze sobą sprzeczne, wówczas pierwszeństwo mają ustawienia w transformacji.

Przykład:

Niech będzie aktywna transformacja. Według danych maszynowych w przypadku RESET transformacja powinna zostać zachowana, równocześnie jednak w przypadku RESET powinna zostać stworzona podstawowa konfiguracja osi geometrycznych. W tym przypadku pozostaje zachowana konfiguracja osi geometrycznych, która została ustalona przy pomocy transformacji.

16.3 Pojemnik osi (AXCTSWE, AXCTSWED, AXCTSWEC)

Funkcja

Poprzez polecenia AXCTSWE wzgl. AXCTSWED jest zwalniany obrót podanego pojemnika osi.

Poprzez polecenie AXCTSWEC można cofnąć już udzielone zezwolenie na obrót pojemnika osi.

Składnia

AXCTSWE (<ID>)
 AXCTSWED (<ID>)
 AXCTSWEC (<ID>)

Znaczenie

AXCTSWE:	<p>Zezwolenie na obrót pojemnika osi</p> <p>Wykonywanie programu nie jest zatrzymywane przez AXCTSWE.</p> <p>Obrót następuje, gdy tylko wszystkie kanały uczestniczące w pojemniku osi udzieliły zezwolenia.</p>
AXCTSWED:	<p>Zezwolenie na obrót pojemnika osi bez uwzględnienia innych kanałów uczestniczących w pojemniku.</p> <p>Wskazówka</p> <ul style="list-style-type: none"> • Wariant polecenia do uproszczenia uruchamiania programu obróbki wzgl. akcji synchronicznej. • Zachowanie się odnośnie innych kanałów uczestniczących w pojemniku osi można zadać przez: MD12760 \$MN_ AXCT_FUNCTION_MASK, Bit 0
AXCTSWEC:	<p>Cofnięcie zezwolenia na obrót pojemnika osi</p> <p>Wskazówka</p> <p>Zezwolenie na obrót pojemnika osi może zostać cofnięte tylko wtedy, gdy obrót jeszcze nie rozpoczął się: \$AN_AXCTSWA[<pojemnik osi>] == 0</p> <p>Na temat zmiennej systemowej patrz "Pojemnik osi (AXCTSWE, AXCTSWED, AXCTSWEC) (Strona 598)"</p>
<ID>:	<p>Identyfikator pojemnika osi albo osi pojemnikowej:</p> <p>CT<numer>: Identyfikator domyślny pojemnika osi: MD12750 \$MN_ AXCT_NAME_TAB Przykład: CT1</p> <p><pojemnik>: Specyficzny dla użytkownika identyfikator pojemnika osi: MD12750 \$MN_ AXCT_NAME_TAB Przykład: CONTAINER_1</p> <p><oś>: Identyfikator osi pojemnikowej znanej w kanale</p>

Wskazówka**Długość kroku**

Wielkość kroku obrotu pojemnika osi jest ustawiana poprzez daną ustawczą:

SD41700 \$SN_AXCT_SWWIDTH

Dalsze informacje**Diagnoza**

Aktualny status pojemnika osi można przeczytać poprzez następującą zmienną systemową:

Zmienna systemowa	Typ	Opis
\$AC_AXCTSWA[<nazwa>]	BOOL	Specyficzny dla kanału status pojemnika osi
\$AN_AXCTSWA[<pojemnik osi>]	BOOL	Specyficzny dla NCU status pojemnika osi
\$AN_AXCTSWE[<pojemnik osi>]	INT	Specyficzny dla slotu status obrotu pojemnika osi Zmienna systemowa daje bitami status slotów pojemnika osi. Każdy bit odpowiada jednemu slotowi.
\$AN_AXCTAS[<pojemnik osi>]	INT	Liczba miejsc (slotów) o które pojemnik osi został aktualnie przełączony.

Obrót pojemnika osi z implicite GET / GETD

Poprzez poniższą daną maszynową można ustawić, że poleceniem AXCTSWE wszystkie osie pojemnikowe kanału przy pomocy domniemanego GET / GETD są przywoływane do kanału. Zamiana osi jest możliwa dopiero po dokonanych obrocie pojemnika.

MD10722 \$MN_AXCHANGE_MASK, Bit 1 = 1

Wskazówka

Obrót pojemnika osi przy pomocy domniemanego GET / GETD **nie** jest przeprowadzany dla osi w stanie "oś przebiegu głównego" (np. oś PLC), ponieważ oś do obrotu pojemnika musiałaby opuścić ten stan.

16.4 Czekanie na obowiązującą pozycję osi (WAITENC)

Funkcja

Przy pomocy polecenia językowego `WAITENC` można w programie NC czekać, aż dla osi zaprojektowanych przy pomocy MD34800 `$MA_WAIT_ENC_VALID = 1` będą do dyspozycji zsynchronizowane wzgl. restaurowane pozycje.

W stanie czekania może nastąpić przerwanie, np. przez start ASUP lub przez zmianę trybu pracy na JOG. Z kontynuowaniem programu jest ew. znów przyjmowany stan czekania.

Wskazówka

Stan czekania jest wyświetlany na interfejsie graficznym przez stan zatrzymania "Czekanie na system pomiarowy".

Składnia

`WAITENC` można programować w części programowej dowolnego programu NC.

Programowanie musi nastąpić w oddzielnym bloku:

```

...
WAITENC
...

```

Przykład

`WAITENC` jest stosowane np. w sterowanym zdarzeniami programie użytkownika `.../_N_CMA_DIR/_N_PROG_EVENT_SPF`, jak pokazuje to poniższy przykład zastosowania.

Przykład zastosowania: Wycofanie narzędzia po POWER OFF z transformacją orientacji

Obróbka z orientacją narzędzia została przerwana przez zanik napięcia.

Przy następnym rozruchu jest wywoływany sterowany zdarzeniami program użytkownika `.../_N_CMA_DIR/_N_PROG_EVENT_SPF`.

W sterowanym zdarzeniami programie użytkownika następuje przy pomocy `WAITENC` czekanie na zsynchronizowane wzgl. restaurowane pozycje osi, aby następnie móc obliczyć frame, który ustawia WKS w kierunku narzędzia.

Kod programu	Komentarz
...	
IF \$P_PROG_EVENT == 4	; Rozruch.
IF \$P_TRAFO <> 0	; Transformacja została wybrana.
WAITENC	; Czekanie na poprawne pozycje osi orientacji.
TOROTZ	; Obrót osi Z WKS w kierunku osi narzędzia.

Kod programu	Komentarz
ENDIF	
M17	
ENDIF	
...	

Następnie narzędzie może zostać odsunięte w trybie pracy JOG przez ruch wycofania w kierunku osi narzędzia.

16.5 Programowalne przełączenie zestawu parametrów (SCPARA)

Funkcja

Przy pomocy polecenia `SCPARA` można dla osi zażądać przełączenia na określony zestaw parametrów.

Wskazówka

Bez przełączenia parametrów podczas obróbki gwintu

Przy nacinaniu gwintu `G33` i gwintowaniu otworu `G331 / G332` zestaw parametrów jest wybierany przez sterowanie i nie można go zmienić.

Zablokowane przełączenie zestawu parametrów

Przełączenia zestawu parametrów można zażądać również poprzez interfejs NC/PLC. Aby uniknąć konfliktów przy przełączaniu, można poprzez interfejs NC/PLC zablokować przełączenie parametrów NC (`SCPARA`):

DB31, ... DBX9.3 (zadanie zestawu parametrów zablokowane przez NC))

Wskazówka

Gdy zostanie przez `SCPARA` zażądane przełączenie zestawu parametrów, podczas gdy jest zablokowane przez interfejs NC/PLC, przełączenie nie jest akceptowane bez wyprowadzenia komunikatu błędu.

Składnia

`SCPARA [<os>] = <wartość>`

Znaczenie

SCPARA:	Polecenie: przełącz zestaw parametrów
<oś>:	Identyfikator osi (oś kanału)
	Typ: AXIS
<wartość>:	Numer zestawu parametrów: 1, 2, 3, ... maks. numer zestawu parametrów

Przykład

Kod programu	Komentarz
...	
N110 SCPARA[X]= 3	; Wybór: oś X, 3. zestaw parametrów
...	

Dalsze informacje

Zezwolenie na przełączenie zestawu parametrów

Przełączenie zestawu parametrów osi musi być explicite zwolnione:

```
MD35590 $MA_PARAMSET_CHANGE_ENABLE[<oś>]
```

Odczyt zestawu parametrów

Numer wybranego zestawu parametrów (zadany zestaw parametrów) można przeczytać przez zmienną systemową \$AA_SCPAR.

Literatura

Szczegółowe informacje dot. zestawów parametrów znajdują się w:

Podręcznik działania Funkcje podstawowe; Rozdział "Prędkości, systemy wartości zadanej/rzeczywistej, Regulacja (G2)" > "Regulacja" > "Zestawy parametrów regulatora położenia"

16.6 Sprawdzenie występującego zakresu językowego NC (STRINGIS)

Funkcja

Przy pomocy funkcji `STRINGIS (...)` można sprawdzić, czy podany łańcuch znaków jest dostępny jako element języka programowania NC w aktualnym zakresie językowym.

Definicja

```
INT STRINGIS (STRING <nazwa>)
```

Składnia

STRINGIS (<nazwa>)

Znaczenie

STRINGIS: Funkcja z wartością zwrótną
 <nazwa>: Nazwa sprawdzanego elementu języka programowania NC
 Wartość zwrótna: Formatem wartości zwrótej jest yxx (dziesiętny).

Elementy języka programowania NC

Następujące elementy języka programowania NC mogą zostać sprawdzone:

- G-Code wszystkich istniejących grup funkcji G np. G0, INVCW, POLY, ROT, KONT, SOFT, CUT2D, CDON, RMBBL, SPATH
- Adresy DIN lub NC jak np. ADIS, RNDM, SPN, SR, MEAS
- Funkcje np. TANG (...) lub GETMDACT
- Procedury np. SBLOF.
- Słowa kluczowe np. ACN, DEFINE albo SETMS
- Dane systemowe np. dane maszynowe \$M... , dane ustawcze \$S... albo dane opcji \$O...
- Zmienne systemowe \$A... , \$V... , \$P...
- Parametry obliczeniowe R...
- Nazwy uaktywnionych cykli
- Zmienne GUD i LUD
- Nazwy makr
- Nazwy etykiet

Wartość zwrótna

Wartość zwrótna ma znaczenie tylko w pierwszych 3 miejscach dziesiętnych. Formatem wartości zwrótej jest yxx, gdzie y = informacja bazowa, a xx = informacja szczegółowa.

Wartość zwrótna	Znaczenie
000	Łańcuch znaków 'name' nie jest znany w niniejszym systemie ¹⁾
100	Łańcuch znaków 'name' jest elementem języka programowania NC, ale jego programowanie jest aktualnie niemożliwe (opcja/funkcja jest nieaktywna)

16.6 Sprawdzenie występującego zakresu językowego NC (STRINGIS)

Wartość zwrotna	Znaczenie
2xx	Łańcuch znaków 'name' jest programowalnym elementem języka programowania NC (opcja/funkcja jest aktywna). Informacja szczegółowa xx zawiera dalsze informacje o rodzaju elementu:
	xx Znaczenie
	01 Adres DIN lub adres NC ²⁾
	02 G-Code (np. G04, INVCW)
	03 Funkcja z wartością zwrótną
	04 Funkcja bez wartości zwrótej
	05 Słowo kluczowe (np. DEFINE)
	06 Dana maszynowa (\$M...), ustawcza (\$S...) lub dana opcji (\$O...)
	07 Parametry systemowe, np. zmienne systemowe (\$...) albo parametry obliczeniowe (R...)
	08 Cykl (Cykl musi być załadowany w NCK, a programy cykli muszą być aktywne ³⁾)
	09 Zmienna GUD (Zmienna GUD musi być zdefiniowana w pliku definicji GUD, a zmienne GUD muszą być uaktywnione)
	10 Nazwa makra (Makro musi być zdefiniowane w plikach definicji makr i makra muszą być uaktywnione ⁴⁾)
	11 Zmienna LUD aktualnego programu obróbki
	12 ISO G-Code (Tryb językowy ISO musi być aktywny)
400	Łańcuch znaków 'name' jest adresem NC, który nie został rozpoznany jako xx == 01 albo xx == 10 i który nie jest G lub R ²⁾
y00	Specyficzne przyporządkowanie nie jest możliwe
<p>1) Zależnie od sterowania jest ewentualnie znany tylko podzbiór poleceń językowych NC Siemens, np. SINUMERIK 802D sl. Na tych sterowaniach jest dla łańcuchów znaków, które zasadniczo są poleceniami językowymi NC Siemens, jest zwracana wartość 0. To zachowanie się może zostać zmienione przez MD10711 \$MN_NC_LANGUAGE_CONFIGURATION. Przy MD10711 = 1 jest wówczas dla poleceń językowych NC Siemens zawsze zwracana wartość 100.</p> <p>2) Adresami NC są następujące litery: A, B, C, E, I, J, K, Q, U, V, W, X, Y, Z. Te adresy NC mogą być również programowane z rozszerzeniem. Rozszerzenie adresu może zostać podane przy sprawdzaniu przy pomocy STRINGIS. Przykład: 201 == STRINGIS("A1"). Litery: D, F, H, L, M, N, O, P, S, T są adresami NC albo funkcjami pomocniczymi, które są stosowane jako definiowane przez użytkownika. Dla nich jest zawsze zwracana wartość 400. Przykład: 400 == STRINGIS("D"). Te adresy NC nie mogą przy sprawdzaniu przy pomocy STRINGIS być podawane z rozszerzeniem. Przykład: 000 == STRINGIS("M02"), albo 400 == STRINGIS("M").</p> <p>3) Nazwy parametrów cykli nie mogą być sprawdzane przy pomocy STRINGIS.</p> <p>4) Adresy zdefiniowane jako makro, np. G, H, M, L są identyfikowane jako makro.</p>	

Przykłady

W poniższych przykładach przyjmuje się, że element językowy NC podany jako łańcuch znaków, o ile szczególnie nie zaznaczono, zasadniczo daje się programować w sterowaniu.

1. Łańcuch znaków "T" jest definiowany jako funkcja pomocnicza:

```
400 == STRINGIS("T")
000 == STRINGIS("T3")
```

2. Łącuch znaków "X" jest definiowany jako oś:

```
201 == STRINGIS ("X")
```

```
201 == STRINGIS ("X1")
```

3. Łącuch znaków "A2" jest definiowany jako adres NC z rozszerzeniem:

```
201 == STRINGIS ("A")
```

```
201 == STRINGIS ("A2")
```

4. Łącuch znaków "INVCW" jest definiowany jako nazwany G-Code:

```
202 == STRINGIS ("INVCW")
```

5. Łącuch znaków "\$MC_GCODE_RESET_VALUES" jest definiowany jako dana maszynowa:

```
206 == STRINGIS ("MC_GCODE_RESET_VALUES")
```

6. Łącuch znaków "GETMDACT" jest funkcją językową NC:

```
203 == STRINGIS ("GETMDACT ")
```

7. Łącuch znaków "DEFINE" jest słowem kluczowym:

```
205 == STRINGIS ("DEFINE")
```

8. Łącuch znaków "\$TC_DP3" jest parametrem systemowym (komponent długości narzędzia):

```
207 == STRINGIS ("TC_DP3")
```

9. Łącuch znaków "\$TC_TP4" jest parametrem systemowym (wielkość narzędzia):

```
207 == STRINGIS ("TC_TP4")
```

10. Łącuch znaków "\$TC_MPP4" jest parametrem systemowym (stan miejsca w magazynie):

```
– Zarządzanie magazynem narzędzi jest aktywne: 207 == STRINGIS ("TC_MPP4") ;
```

```
– Zarządzanie magazynem narzędzi nie jest aktywne: 000 == STRINGIS ("TC_MPP4")
```

Patrz też w punkcie: Zarządzanie magazynem narzędzi.

11. Łącuch znaków "MACHINERY_NAME" jest definiowany jako zmienna GUD:

```
209 == STRINGIS ("MACHINERY_NAME")
```

12. Łącuch znaków "LONGMACRO" jest definiowany jako makro:

```
210 == STRINGIS ("LONGMACRO")
```

13. Łącuch znaków "MYVAR" jest definiowany jako zmienna LUD:

```
211 == STRINGIS ("MYVAR")
```

14. Łącuch znaków "XYZ" nie jest poleceniem znanym w NCK, zmienną GUD, nazwą makra albo cyklu:

```
000 == STRINGIS ("XYZ")
```

Zarządzanie magazynem narzędzi

Jeżeli funkcja zarządzania magazynem narzędzi nie jest aktywna, STRINGIS daje dla parametrów systemowych zarządzania magazynem narzędzi, niezależnie od danej maszynowej

- MD10711 \$MN_NC_LANGUAGE_CONFIGURATION

zawsze wartość 000.

Tryb ISO

Gdy jest aktywna funkcja "Tryb ISO":

- MD18800 \$MN_MM_EXTERN_LANGUAGE (uaktywnienie zewnętrznych języków NC)
- MD10880 \$MN_MM_EXTERN_CNC_SYSTEM (system sterowania do zaadaptowania)

STRINGIS najpierw sprawdza podany łańcuch znaków jako G-Code SINUMERIK. Jeżeli łańcuch znaku nie jest G-Code SINUMERIK, następuje następnie sprawdzenie jako G-Code ISO.

Programowane przełączenia (G_{290} (SINUMERIK Mode), G_{291} (ISO Mode)) nie mają żadnego wpływu na STRINGIS.

Przykład

Dane maszynowe mające znaczenie dla funkcji STRINGIS(...) mają następujące wartości:

- MD10711 \$MN_NC_LANGUAGE_CONFIGURATION = 2 (Jako znane są traktowane tylko te polecenia NC, których opcje są ustawione)
- MD19410 \$ON_TRAFO_TYPE_MASK = 'H0' (opcja: Transformacje)
- MD10700 \$MN_PREPROCESSING_LEVEL='H43' (przetwarzanie wstępne cykli aktywne)

Następujący program przykładowy jest wykonywany bez komunikatu błędu:

Kod programu	Komentarz
N1 R1=STRINGIS("TRACYL")	; R1 == 0, ponieważ TRACYL z powodu brakującej
	; opcji transformacji jest rozpoznawane
	; jako "nieznane"
N2 IF STRINGIS("TRACYL") == 204	;
N3 TRACYL(1,2,3)	; N3 jest pomijane
N4 ELSE	
N5 G00	; i zamiast tego wykonywane N5
N6 ENDIF	
N7 M30	

16.7 Interaktywne wywołanie okna z programu obróbki (MMC)

Funkcja

Poprzez polecenie MMC mogą z programu obróbki na HMI być wyświetlane okna dialogowe definiowane przez użytkownika (obrazy dialogowe).

Wygląd okna dialogowego jest ustalany przez czysto tekstowe projektowanie (plik COM w katalogu cykli), oprogramowanie systemowe HMI pozostaje przy tym bez zmian.

Okna dialogowe zdefiniowane przez użytkownika nie mogą być wywoływane równocześnie w różnych kanałach.

Składnia

MMC (<rozkaz>, <tryb kwitowania>)

Znaczenie

MMC:	identyfikator podprogramu
<rozkaz>:	Parametry typu STRING
	Zawiera rozkaz MMC, np. w następującej formie:
	"CYCLES, PICTURE_ON, T_SK.COM, BILD, MGUD.DEF, BILD_3.AWB, TEST_1, A1"
	CYCLES: Okno obsługowe, w którym są wykonywane zaprojektowane dialogi z użytkownikiem.
	PICTURE_ON wzgl.
	PICTURE_OFF: Polecenie: wybór wzgl. cofnięcie wyboru obrazu
	T_SK.COM: Plik Com: Nazwa pliku okna dialogowego (cykle użytkownika). Tutaj jest ustalony wygląd obrazów dialogowych. Na obrazie dialogowym mogą być wyświetlane zmienne użytkownika i/albo teksty komentarzy.
	BILD: Nazwa obrazu dialogowego: Poszczególne obrazy są wybierane przez nazwy obrazów dialogowych.
	MGUD.DEF: Plik definicji danych użytkownika, do którego następuje dostęp przy odczycie/zapisie zmiennych.
	BILD_3.AWB: Plik graficzny
	TEST_1: Czas wyświetlania albo zmienna kwitowania
	A1: Zmienne tekstowe..."

<tryb kwitowania>:	Parametr typu CHAR
Wartość: <"N">:	bez kwitowania Wykonywanie programu jest kontynuowane po wysłaniu rozkazu. Nadawca nie jest powiadamiany, gdy rozkazu nie można pomyślnie wykonać.
"S":	kwitowanie synchroniczne Wykonywanie programu jest tak długo zatrzymywane, aż komponent odbierający pokwituje rozkaz. Przy pozytywnym pokwitowaniu jest wykonywane następne polecenie. Przy ujemnym pokwitowaniu jest wprowadzany komunikat błędu.
"A":	kwitowanie asynchroniczne Wykonywanie programu jest kontynuowane po wysłaniu polecenia. Pokwitowanie jest zapisywane w zmiennej kwitowania (predefiniowana zmienna systemowa) i należy ją explicite odczytać z programu. Parametr następujący po trybie kwitowania jest numerem zmiennej kwitowania.

Gdy <tryb kwitowania> nie zostanie zaprogramowany, jest jako ustawienie standardowe stosowane kwitowanie synchroniczne.

16.8 Czas przebiegu programu / licznik obrabianych przedmiotów

16.8.1 Czas przebiegu programu / licznik obrabianych przedmiotów (przeгляд)

W celu wsparcia operatora obrabiarki są udostępniane informacje dot. czasu przebiegu programu i liczby obrabianych przedmiotów.

Te informacje mogą być przetwarzane jako zmienne systemowe w programie NC i/albo PLC. Równocześnie te informacje są do dyspozycji w celu wyświetlenia na interfejsie graficznym.

16.8.2 Czas przebiegu programu

Funkcja

Funkcja "czas przebiegu programu" udostępnia wewnętrzny w NC zegar do nadzoru procesów technologicznych, które mogą być odczytywane przez zmienne systemowe specyficzne dla NC i specyficzne dla kanału, w programie obróbki i w akcjach synchronicznych.

Przerzutnik do pomiaru czasu przebiegu (\$AC_PROG_NET_TIME_TRIGGER) jest jedyną zapisywalną zmienną systemową funkcji i służy do selektywnego pomiaru segmentów programu. Tzn. przez zapisywanie przerzutnika w programie NC pomiar czasu może być włączany i wyłączany.

Zmienna systemowa	Znaczenie	Aktywność
Specyficzna dla NC		
\$AN_SETUP_TIME	Czas od ostatniego rozruchu sterowania z wartościami standardowymi ("start zimny") w minutach Jest przy każdym rozruchu sterowania z wartościami standardowymi automatycznie cofany na "0".	<ul style="list-style-type: none"> Zawsze aktywny
\$AN_POWERON_TIME	Czas od ostatniego normalnego rozruchu sterowania ("start ciepły") w minutach Jest przy każdym normalnym rozruchu sterowania automatycznie cofany na "0".	
Specyficzna dla kanału		
\$AC_OPERATING_TIME	Całkowity czas przebiegu programów NC w trybie pracy Automatyka w sekundach Wartość jest przy każdym rozruchu sterowania automatycznie cofana na "0".	<ul style="list-style-type: none"> Uaktywnienie przez MD27860 Tylko tryb pracy AUTOMATYKA
\$AC_CYCLE_TIME	Czas przebiegu wybranego programu NC w sekundach Wartość jest ze startem nowego programu NC automatycznie cofana na "0".	
\$AC_CUTTING_TIME	Czas obróbki w sekundach Mierzony jest czas ruchu osi uczestniczących w tworzeniu konturu (jest aktywna co najmniej jedna) bez aktywnego posuwu szybkiego we wszystkich programach NC między NC-Start i końcem programu / reset NC. Pomiar jest dodatkowo przerywany przy aktywnym czasie oczekiwania. Jest przy każdym rozruchu sterowania z wartościami standardowymi wartość jest automatycznie cofana na "0".	

Zmienna systemowa	Znaczenie	Aktywność	
\$AC_ACT_PROG_NET_TIME	Aktualny czas przebiegu netto aktualnego programu NC w sekundach Jest ze startem nowego programu NC automatycznie cofany na "0".	<ul style="list-style-type: none"> Zawsze aktywny Tylko tryb pracy AUTOMATYKA 	
\$AC_OLD_PROG_NET_TIME	Czas przebiegu netto w sekundach programu właśnie poprawnie zakończonego przez M30		
\$AC_OLD_PROG_NET_TIME_COUNT	Zmiany na \$AC_OLD_PROG_NET_TIME Po POWER ON zmienna \$AC_OLD_PROG_NET_TIME_COUNT jest ustawiona na "0". \$AC_OLD_PROG_NET_TIME_COUNT jest zwiększana zawsze wtedy, gdy sterowanie zapisało \$AC_OLD_PROG_NET_TIME na nowo.		
\$AC_PROG_NET_TIME_TRIGGER	Przerzutnik do pomiaru czasu przebiegu:		<ul style="list-style-type: none"> Tylko tryb pracy AUTOMATYKA
	0	Stan neutralny Przerzutnik nie jest aktywny.	
	1	Zakończenie Kończy pomiar i kopiuje wartość z \$AC_ACT_PROG_NET_TIME do \$AC_OLD_PROG_NET_TIME. \$AC_ACT_PROG_NET_TIME jest ustawiana na "0" i następnie mierzy dalej.	
	2	Start Uruchamia pomiar i ustawia przy tym \$AC_ACT_PROG_NET_TIME na "0". \$AC_OLD_PROG_NET_TIME nie jest zmieniana.	
	3	Stop Zatrzymuje pomiar. Nie zmienia \$AC_OLD_PROG_NET_TIME i utrzymuje stałą \$AC_ACT_PROG_NET_TIME, aż do kontynuacji.	
4	Kontynuacja Kontynuacja pomiaru, tzn. przedtem zatrzymany pomiar jest ponownie podejmowany. \$AC_ACT_PROG_NET_TIME biegnie dalej. \$AC_OLD_PROG_NET_TIME nie jest zmieniana.		
Przez POWER ON wszystkie zmienne systemowe są cofane na "0"!			
Literatura: Szczegółowy opis wyszczególnionych zmiennych systemowych znajduje się w: Podręcznik działania Funkcje podstawowe; BAG, kanał, praca programowa, zachowanie się przy zresetowaniu (K1), punkt: Czas przebiegu programu			

Wskazówka**Producent maszyny**

Włączenie uaktywnialnego zegara następuje przez daną maszynową MD27860 \$MC_PROCESSTIMER_MODE.

Zachowanie się aktywnych pomiarów czasu w przypadku określonych funkcji (np. GOTOS, korektor = 0%, aktywny posuw w pracy próbnej, test programu, ASUP, PROG_EVENT, ...) jest konfigurowane przez dane maszynowe MD27850 \$MC_PROG_NET_TIMER_MODE i MD27860 \$MC_PROCESSTIMER_MODE.

Literatura:

Podręcznik działania Funkcje podstawowe; BAG, Kanał, Praca programowa, Zachowanie się przy zresetowaniu (K1), punkt: Czas przebiegu programu

Wskazówka**Pozostały czas dla obrabianego przedmiotu**

Gdy są kolejno obrabiane takie same przedmioty, można z wartości zegara:

- czas obróbki ostatnio obrobionego przedmiotu (patrz \$AC_OLD_PROG_NET_TIME)
- i
- aktualny czas obróbki (patrz \$AC_ACT_PROG_NET_TIME)

określić pozostający czas obróbki przedmiotu.

Pozostały czas jest oprócz aktualnego czasu obróbki wyświetlany na otoczce graficznej.

Wskazówka**Zastosowanie STOPRE**

Zmienne systemowe \$AC_OLD_PROG_NET_TIME i \$AC_OLD_PROG_NET_TIME_COUNT nie wytwarzają implicite zatrzymania przebiegu wyprzedzającego. Przy zastosowaniu w programie obróbki nie stanowi to problemu, gdy wartość zmiennej systemowej pochodzi z poprzedniego przebiegu programu. Gdy jednak przerzutnik jest w celu pomiaru czasu przebiegu (\$AC_PROG_NET_TIME_TRIGGER) zapisywany z wysoką częstotliwością i przez to \$AC_OLD_PROG_NET_TIME bardzo często zmienia się, wówczas powinno w programie obróbki być implicite stosowane STOPRE.

Warunki brzegowe

- **Szukanie bloku**

Przy szukaniu bloku nie są określane czasy przebiegu programu.

- **REPOS**

Czas trwania procesu REPOS jest doliczany do aktualnego czasu obróbki (\$AC_ACT_PROG_NET_TIME).

Przykłady

Przykład 1: Pomiar czasu trwania "mySubProgrammA"

Kod programu

```
...
N50 DO $AC_PROG_NET_TIME_TRIGGER=2
N60 FOR ii= 0 TO 300
N70 mySubProgrammA
N80 DO $AC_PROG_NET_TIME_TRIGGER=1
N95 ENDFOR
N97 mySubProgrammB
N98 M30
```

Gdy program wykonał wiersz N80, w \$AC_OLD_PROG_NET_TIME znajduje się czas przebiegu netto dla "mySubProgrammA".

Wartość \$AC_OLD_PROG_NET_TIME:

- pozostaje zachowana po M30.
- jest aktualizowana po każdym przejściu pętli.

Przykład 2: Pomiar czasu trwania "mySubProgrammA" i "mySubProgrammC"

Kod programu

```
...
N10 DO $AC_PROG_NET_TIME_TRIGGER=2
N20 mySubProgrammA
N30 DO $AC_PROG_NET_TIME_TRIGGER=3
N40 mySubProgrammB
N50 DO $AC_PROG_NET_TIME_TRIGGER=4
N60 mySubProgrammC
N70 DO $AC_PROG_NET_TIME_TRIGGER=1
N80 mySubProgrammD
N90 M30
```

16.8.3 Licznik obrabianych przedmiotów

Funkcja

Funkcja "Licznik obrabianych przedmiotów" udostępnia różne liczniki, które mogą być stosowane w szczególności do wewnętrznego w sterowaniu liczenia obrabianych przedmiotów.

Liczniki istnieją jako specyficzne dla kanału zmienne systemowe z dostępem w celu zapisu i odczytu w zakresie wartości 0 do 999 999 999.

Zmienna systemowa	Znaczenie
\$AC_REQUIRED_PARTS	Liczba przedmiotów do obrobienia (zadana liczba obrabianych przedmiotów) W tym liczniku można zdefiniować liczbę obrabianych przedmiotów, przy osiągnięciu której rzeczywista liczba obrobionych przedmiotów \$AC_ACTUAL_PARTS jest zerowana.
\$AC_TOTAL_PARTS	Liczba łączna obrobionych przedmiotów (łączna rzeczywista) Ten licznik podaje liczbę przedmiotów obrobionych od chwili startu. Wartość jest tylko przy rozruchu sterowania z wartościami standardowymi automatycznie cofana na "0".
\$AC_ACTUAL_PARTS	Liczba obrobionych przedmiotów (rzeczywista) W tym liczniku jest rejestrowana liczba wszystkich przedmiotów obrobionych od chwili startu. Przy osiągnięciu zadanej liczby obrobionych przedmiotów (\$AC_REQUIRED_PARTS) licznik jest automatycznie cofany na "0" (\$AC_REQUIRED_PARTS > 0).
\$AC_SPECIAL_PARTS	Liczba obrobionych przedmiotów liczonych przez użytkownika Ten licznik pozwala użytkownikowi na liczenie obrabianych przedmiotów według własnej definicji. Można zdefiniować wyprowadzenie alarmu przy osiągnięciu zadanej liczby sztuk (\$AC_REQUIRED_PARTS). Zerowanie licznika użytkownik musi przeprowadzić sam.

Wskazówka

Wszystkie liczniki są przy rozruchu sterowania z wartościami standardowymi zerowane i mogą być niezależnie od ich uaktywnienia czytane/zapisywane.

Wskazówka

Przez kanałowe dane maszynowe można wpływać na uaktywnienie licznika, czas zerowania i algorytm liczenia.

Wskazówka**Liczenie obrabianych przedmiotów przy pomocy polecenia M definiowanego przez użytkownika**

Przez dane maszynowe można ustawić, by impulsy liczenia dla różnych liczników obrabianych przedmiotów zamiast przez koniec programu M2/M30 były wyzwalane przez definiowane przez użytkownika polecenia M.

Literatura

Dalsze informacje dot. działania "liczników obrabianych przedmiotów" patrz:

- Podręcznik działania Funkcje podstawowe; BAG, Kanał, Praca programowa, Zachowanie się przy zresetowaniu (K1), punkt: Licznik obrabianych przedmiotów

16.9 Process DataShare - Wyprowadzenie do zewnętrznego urządzenia/pliku (EXTOPEN, WRITE, EXTCLOSE)

Funkcja

Przy pomocy funkcji "Process DataShare" jest możliwe zapisanie danych z programu obróbki na zewnętrznym urządzeniu / w zewnętrznym pliku, np. w celu protokolowania danych produkcyjnych albo w celu sterowania agregatami dodatkowymi na sterowaniu.

Wyprowadzenie do zewnętrznego urządzenia/pliku następuje w trzech krokach:

1. Otwarcie zewnętrznego urządzenia/pliku

Przy pomocy polecenia `EXTOPEN` następuje otwarcie zewnętrznego urządzenia/pliku dla kanału w celu zapisu.

2. Zapis danych

Wyprowadzana dana może zostać przygotowana przy pomocy funkcji string języka NC ("Operacja na łańcuchach znaków (Strona 81)"), np. `SPRINT`. Sam zapis następuje poprzez polecenie `WRITE`.

3. Zamknięcie zewnętrznego urządzenia/pliku

Przy pomocy polecenia `EXTCLOSE` albo poprzez dojście do końca programu (`M30`), jak też przy zresetowaniu kanału jest zwalniane zajęte w kanale zewnętrzne urządzenie/plik.

Wskazówka

W jednym programie obróbki/kanały można zająć więcej, niż jedno zewnętrzne urządzenie/plik.

Dostępność

Funkcja jest dostępna:

- tylko w programach obróbki (**nie** w akcjach synchronicznych).
- równolegle we wszystkich kanałach obróbki NCK dla wszystkich dostępnych (projektowanych) urządzeń wyjścia.

Dla urządzenia wyjścia można zadać przy jego otwieraniu, czy ma być ono używane wyłącznie przez jeden kanał, czy może być przydzielone na wiele kanałów, które chcą do niego wyprowadzać dane ("shared"-modus).

Składnia

```
DEF INT <błąd>
DEF STRING[<n>] <wyprowadzenie>
...
EXTOPEN(<błąd>,"<urządzenie zewnętrzne>",<tryb obróbki>,<tryb użycia>,
<tryb zapisu>)
...
<wyprowadzenie>="wyprowadzane dane"
WRITE(<błąd>,"<urządzenie zewnętrzne>",<wyprowadzenie>)
...
EXTCLOSE(<błąd>,"<urządzenie zewnętrzne>")
```

Znaczenie

EXTOPEN:	Polecenie do otwarcia zewnętrznego urządzenia/pliku
<błąd>:	<p>Parametr 1: Zmienna do zwrotu wartości błędu</p> <p>Na podstawie wartości błędu można ocenić w programie powodzenie operacji i odpowiednio postąpić.</p> <p>Typ: INT</p> <p>Wartości:</p> <ul style="list-style-type: none"> 0 Nie ma błędu 1 Urządzenia zewnętrznego nie można otworzyć 2 Urządzenie zewnętrzne nie jest zaprojektowane 3 Urządzenie zewnętrzne zaprojektowane z niepoprawną ścieżką 4 Brak praw dostępu do urządzenia zewnętrznego 5 Tryb użycia: urządzenie zewnętrzne już jest zajęte "na wyłączność" 6 Tryb użycia: urządzenie zewnętrzne już jest zajęte w trybie "shared" 7 Długość pliku większa, niż LOCAL_DRIVE_MAX_FILESIZE 8 Maksymalna liczba urządzeń zewnętrznych jest przekroczona 9 Opcja dla LOCAL_DRIVE nie ustawiona 11 Interfejs V.24 jest już zajęty przez funkcję Easy-Message (tylko 828D) 12 Tryb zapisu: podanie sprzeczne z extdev.ini 16 Zaprogramowano niepoprawną ścieżkę zewnętrzną 22 Urządzenie zewnętrzne nie zamontowane
<urządzenie zewnętrzne>:	<p>Parametr 2: Identyfikator symboliczny otwieranego zewnętrznego urządzenia/pliku</p> <p>Typ: STRING</p> <p>Identyfikator symboliczny składający się z:</p> <ul style="list-style-type: none"> 1. logicznej nazwy urządzenia 2. po której ew. następuje ścieżka pliku (dołączona przy pomocy "/"). <p>Są zdefiniowane następujące logiczne nazwy urządzeń:</p>

"LOCAL_DRIVE":	Lokalna CompactFlash Card (predefiniowana)
"CYC_DRIVE":	zarezerwowane podanie stacji do zastosowania w cyklach SIEMENSA (predefiniowane)
"/dev/ext/1",...	Dostępne stacje sieciowe
"/dev/ext/9":	Wskazówka: Wymagane zaprojektowanie w pliku extdev.ini!
"/dev/cyc/1", "/dev/cyc/2":	zarezerwowane podania stacji do zastosowania w cyklach SIEMENSA Wskazówka: Wymagane zaprojektowanie w pliku extdev.ini!
"/dev/v24":	Interfejs V.24 Wskazówka: Wymagane zaprojektowanie w pliku extdev.ini!

Ścieżka pliku:

- Do "LOCAL_DRIVE" i "CYC_DRIVE" musi zostać podana ścieżka pliku, np.:
"LOCAL_DRIVE/my_dir/my_file.txt"
- Logiczne nazwy urządzeń "/dev/ext/1...9" i "/dev/cyc/1...2" mogą poprzez zaprojektowanie:
 - już wskazywać na plik, wówczas wolno podać tylko logiczną nazwę urządzenia, np.:
"/dev/ext/4"
 - albo na katalog, wówczas musi zostać podana ścieżka pliku, np.
"/dev/ext/5/my_dir/my_file.txt"
- Do "/dev/v24" nie może być podwieszona ścieżka pliku.

Wskazówka:

Dla logicznych nazw urządzeń "/dev/ext/1...9", "/dev/v24" i "/dev/cyc/1...2" pisanie dużymi/małymi literami jest ignorowane, przy podaniu ścieżki do pliku pisanie dużymi/małymi literami ma znaczenie. Dla "LOCAL_DRIVE" i "CYC_DRIVE" jest dopuszczalne pisanie tylko dużymi literami.

<tryb obróbki>:

Parametr 3: Tryb wykonywania dla poleceń WRITE dla tego urządzenia/pliku

Typ: STRING

Wartości: "SYN": Zapis synchroniczny
 Wykonywanie programu jest zatrzymywane, aż proces zapisu będzie zakończony.
 Pomyślne zakończenie zapisu synchronicznego można sprawdzić poprzez ocenę zmiennej błędu polecenia WRITE.

"ASYN": Zapis asynchroniczny
 Wykonywanie programu nie jest przerywane poprzez polecenie WRITE.

Wskazówka:
 Zmienna błędu polecenia WRITE w tym trybie ma zawsze wartość 0 (nie ma błędu) i nie informuje. W tym trybie nie ma pewności, że polecenie WRITE zostało wykonane pomyślnie.

<tryb użycia>:

Parametr 4: Tryb użycia dla tego urządzenia/pliku

Typ: STRING

Wartości: "SHARED": Następuje zażądanie urządzenia/pliku w trybie "shared". Inne kanały mogą współużywać urządzenie, tzn. również w tym trybie otwierać.

"EXCL": Stosowanie urządzenia/pliku następuje w kanale na wyłączność, żaden inny kanał nie może współużywać urządzenia.

<tryb zapisu>:

Parametr 5: Tryb zapisu dla poleceń WRITE do tego urządzenia/pliku (opcjonalnie)

Typ: STRING

Wartości: "APP": Zawieszenie
 Plik pozostaje zachowany w swojej treści, wywołania zapisu powodują dołączenie na końcu.

"OVR": Zastąpienie
 Treść pliku jest kasowana i sporządzana na nowo poprzez następujące dalej wywołania zapisu.

Wskazówka:

Przy pomocy tego parametru **nie** można zastąpić trybu zapisu zaprojektowanego w extdev.ini. W przypadku konfliktu wywołanie EXTOPEN jest kwitowane z błędem.

WRITE: Polecenie zapisu wyprowadzanych danych
Opis patrz "Zapisanie pliku (WRITE) (Strona 143)!"

EXTCLOSE: Polecenie do zamknięcia otwartego zewnętrznego urządzenia/pliku

<bład>: **Parametr 1:** Zmienna do zwrotu wartości błędu
 Typ: INT
 Wartości: 0 Nie ma błędu
 16 Zaprogramowano niepoprawną ścieżkę zewnętrzną
 21 Błąd przy zamknięciu urządzenia zewnętrznego

<urządzenie zewnętrzne>: **Parametr 2:** Identyfikator symboliczny zamykanego zewnętrznego urządzenia/pliku
 Opis patrz pod EXTOPEN!

Wskazówka:
 Identyfikator musi być identyczny z identyfikatorem podanym w poleceniu EXTOPEN!

Przykład

```

Kod programu
-----
N10  DEF INT RESULT
N20  DEF BOOL EXTDEVICE
N30  DEF STRING[80] WYPROWADZENIE
N40  DEF INT PHASE
N50  EXTOPEN(RESULT,"LOCAL_DRIVE/my_file.txt","SYN","SHARED")
N60  IF RESULT > 0
N70      MSG("Błąd przy EXTOPEN:" << RESULT)
N80  ELSE
N90      EXTDEVICE=TRUE
N100 ENDIF
...
N200 PHASE=4
N210 IF EXTDEVICE
N220     WYPROWADZENIE=SPRINT("Koniec fazy: %D",PHASE)
N230     WRITE(RESULT,"LOCAL_DRIVE/my_file.txt",AUSGABE)
N240 ENDIF
...
    
```

Dalsze informacje

Wpływ na pracę z przechodzeniem płynnym

Polecenia `EXTOPEN`, `WRITE` i `EXTCLOSE` wywołują każdorazowo zatrzymanie przebiegu wyprzedzającego i przez to przerywają pracę z przechodzeniem płynnym.

Zachowanie się przy szukaniu bloku

Podczas "Szukania bloku z obliczaniem" nie następuje wyprowadzenie przy pomocy `WRITE`. Są jednak zbierane polecenia `EXTOPEN` i `EXTCLOSE` i -- po dojściu do celu szukania -- jest ustawiane ich działanie przy pomocy NC-Start. Następujące dalej polecenia `WRITE` znajdują dzięki temu takie samo warunki, jak przy normalnym wykonywaniu programu.

Przy szukaniu z obliczaniem w trybie "Test programu" (`SERUPRO`) `EXTOPEN`, `WRITE` i `EXTCLOSE` są wykonywane, jak przy normalnym wykonywaniu programu.

Zachowanie się przy reset

Z zakończeniem programu obróbki i zresetowaniem kanału są zamykane wszystkie otwarte w tym kanale zewnętrzne urządzenia/pliki.

Dostępne urządzenia zewnętrzne

Jako zewnętrzne urządzenia/pliki mogą być do dyspozycji:

- Pliki na lokalnej CompactFlash Card

Pod pojęciem lokalnej CompactFlash Card rozumie się pamięć, do której odsyła się z HMI przy pomocy identyfikatora symbolicznego `LOCAL_DRIVE`. W przypadku SINUMERIK 840 jest to stacja lokalna, w przypadku SINUMERIK 828D jest to CompactFlash Card użytkownika.

Wskazówka

Do wyprowadzenia na urządzenie `LOCAL_DRIVE` jest w przypadku SINUMERIK 840D wymagana opcja "dodatkowe xxx MB pamięci użytkownika HMI na CF-Card NCU". Dla SINUMERIK 828D musi być CompactFlash Card użytkownika, opcja nie jest tutaj wymagana.

- Pliki w stacji sieciowej
- Interfejs V.24

Wskazówka

W celu wyprowadzenia na interfejs V.24 jest w przypadku SINUMERIK 840D si wymagany moduł opcji NCU RS232. W przypadku SINUMERIK 828D wyprowadzenie następuje na zintegrowany interfejs V.24 (warunek: `MD51233 $MNS_ENABLE_GSM_MODEM = 0`).

Projektowanie

Zaprojektowanie stosowanych urządzeń zewnętrznych następuje w pliku /oem/sinumerik/nck/extdev.ini wzgl. /user/sinumerik/nck/extdev.ini. Jeżeli występują obydwa pliki, wówczas pierwszeństwo mają wpisy w obszarze użytkownika. Plik można zmieniać w obszarze obsługi URUCHOMIENIE pod DANE SYSTEMOWE/Karta CF.

Wskazówka

Dla zastosowania LOCAL_DRIVE i CYC_DRIVE nie jest wymagane zaprojektowanie w pliku extdef.ini. Obydwa urządzenia są zawsze dostępne, gdy tylko jest ustawiona odpowiednia opcja, wzgl. gdy jest CompactFlash Card użytkownika.

W segmencie [ExternalDevices] pliku extdev.ini są definiowane/wymieniane stosowane urządzenia zewnętrzne. Jako urządzenia można podać jedno urządzenie szeregowe (/dev/v24) i do dziewięciu plików albo katalogów (/dev/ext/1...9). Notacją podawania jest pisownia Linux. Wiersze, które rozpoczynają się od ";", są komentarzami i nie są czytane.

Za wyjątkiem /dev/v24 urządzenia mogą zostać określone, jako ścieżka katalogu - zakończona dołączonym "/" - albo jako ścieżka pliku - a więc z dołączoną w pełni określoną ścieżką, kończącą się nazwą pliku (bez końcowego "/"). Do urządzenia ze ścieżką katalogu musi przy zastosowaniu w programie obróbki zostać jednocześnie podana nazwa pliku (ścieżka).

Oprócz /dev/v24 definicja urządzenia następuje poprzez trzy dane rozdzielone przecinkiem "serwer", "ścieżka" i optymalny "tryb zapisu".

Do plików wzgl. katalogów (działa wówczas na wszystkie pliki w katalogu) można podać, czy plik po otwarciu ma zostać zastąpiony ("O" = overwrite), czy też wyprowadzane dane mają być dołączane do pliku ("A" = append). Wartością standardową jest "A". Nie istniejący plik/katalog jest przy otwieraniu tworzony na nowo.

Dla interfejsu V.24 urządzenia są podawane tylko ustawienia szybkości transmisji, bity danych, bity stopu, parzystość, protokół i ewentualnie koniec, w takiej kolejności.

Dla plików, które są tworzone/zapisywane na LOCAL_DRIVE, następuje poprzez daną LOCAL_DRIVE_MAX_FILESIZE ustawienie maksymalnej wielkości pliku w bajtach - obowiązuje jednolicie dla wszystkich plików. Wielkość pliku jest sprawdzana przy wykonaniu polecenia EXTOOPEN w trybie append. Opcjonalnie można przy pomocy danej LOCAL_DRIVE_FILE_MODE ustalić tryb zapisu ("O" = overwrite, "A" = append). Wartością standardową jest "A".

Wskazówka

Szablon do skopiowania pliku projektowego extdev.ini jest dostępny w katalogu /siemens/sinumerik/nck.

Wskazówka

Zmiany w pliku extdev.ini działają dopiero po zrestartowaniu/rozruchu NCK.

Wskazówka**Urządzenia USB**

Dla SINUMERIK 828D można, jako cel jednego z urządzeń zdefiniować również "usb" (bez podania partycji!) dla urządzenia USB na stronie czołowej. Dostęp do urządzenia na USB można uzyskać z programu obróbki tylko pośrednio poprzez symboliczny identyfikator urządzenia " /dev/ext/x".

W przypadku SINUMERIK 840D si mogą jako urządzenia USB być projektowane tylko statycznie połączone interfejsy USB jednej TCU. Projektowanie następuje poprzez SERVER:/PATH, jako specyfikacja dla "Server", przy czym SERVER jest nazwą TCU, a /PATH określa interfejs USB. Dostęp do poszczególnych interfejsów USB jednej TCU jest uzyskiwany przy pomocy "dev0-0", "dev0-1", "dev1-0". Podanie ścieżki rozpoczyna się zawsze od "/Partition", przy czym partycja może być podana poprzez jej dwucyfrowy numer lub jej nazwę i ew. przedłużona przez ścieżkę pliku do pożądanego celu, a więc np.

```
/dev/ext/8 = "TCU4:/dev0-0, /01/, A"
```

```
/dev/ext/8 = "TCU4:/dev0-0, /01/mydir.dir"
```

```
/dev/ext/8 = "TCU4:/dev0-0, /myfirstpartition/Mydir.dir/myfile.txt, O"
```

Przykłady:

```
[ExternalDevices]
```

```
; Wiersz komentarzy
```

```
; przykład dla V24
```

```
; /dev/v24 = "9600, 8, 1, none, rts [, etx]"
```

```
; przykłady dla dysków sieciowych
```

```
; /dev/ext/1 = "[USERNAME[/DOMAIN][%PASSWORD]@]SERVER/SHARE/, /, A"
```

```
; /dev/ext/2 = "[USERNAME[/DOMAIN][%PASSWORD]@]SERVER/SHARE, /myfile.txt, O"
```

```
; /dev/ext/3 = "[USERNAME[/DOMAIN][%PASSWORD]@]SERVER/SHARE, /mydir/, A"
```

```
; /dev/ext/4 = "SERVER:/dev0-0, /01/, A"
```

```
; ...
```

```
; SINUMERIK 828 only (USB)
```

```
; /dev/ext/9 = "usb, / [ , O]"
```

```
; domyślnie: Numer partycji = 1
```

```
; SIEMENS only
```

```
; /dev/cyc/1= "[USERNAME[/DOMAIN][%PASSWORD]@]SERVER/SHARE, /mydir/, A"
```

```
; /dev/cyc/2= "[USERNAME[/DOMAIN][%PASSWORD]@]SERVER/SHARE/mydir, /, A"
```

```
LOCAL_DRIVE_MAX_FILESIZE = 50000
```

```
LOCAL_DRIVE_FILE_MODE = "O"
```

Działanie parametru EXTOPEN <tryb zapisu>

Poprzez podanie trybu zapisu, jak też przy projektowaniu w pliku extdev.ini, jak też przy wywołaniu EXTOPEN może dochodzić do konfliktów praw, które przy wywołaniu EXTOPEN są ew. kwitowane z błędem:

Wartość z extdev.ini	Wartość parametru EXROPEN		
	"OVR"	"APP"	-
"O"	O	Błąd	O
"A"	Błąd	A	A
-	O	A	A
Wyjaśnienie: O: Działa tryb "Zastępowanie". A: Działa tryb "Dołączenie". Błąd: Wywołanie EXTOPEN jest kwitowane z błędem.			

LOCAL_DRIVE: Atrybuty pliku

Pliki utworzone przy pomocy EXTOPEN na LOCAL_DRIVE uzyskują następujące atrybuty:

- Właściciel: "user" Ustawione prawo zapisu/odczytu
- Grupa: "operator" Ustawione prawo zapisu/odczytu

Maksymalna liczba otwartych urządzeń zewnętrznych

Dla wszystkich kanałów NC może być w sumie otwartych maksymalnie 10 urządzeń wyjścia. Dodatkowo dla cykli Siemensa zarezerwowano dodatkowo 2 wpisy.

Do tych urządzeń może być równocześnie aktywnych maksymalnie 5 zleceń.

16.10 Alarmy (SETAL)

Funkcja

W programie NC mogą być ustawiane alarmy. Są one przedstawiane na interfejsie graficznym w specjalnym polu. Z alarmem jest każdorazowo związana reakcja sterowania odpowiednio do kategorii alarmu.

Literatura:

Dalej idące informacje dot. reakcji na alarm patrz podręcznik uruchomienia.

Składnia

SETAL(<numer alarmu> [, <łańcuch znaków>])

Znaczenie

SETAL:	Słowo kluczowe do zaprogramowania alarmu. SETAL musi być programowane w oddzielnym bloku NC.										
<numer alarmu>:	Zmienna typu INT. Zawiera numer alarmu. Obowiązujący zakres numerów alarmów leży między 60000 i 69999, z czego 60000 do 64999 zarezerwowano dla cykli SIEMENS, a numery 65000 do 69999 są do dyspozycji użytkownika.										
<łańcuch znaków>:	Przy programowaniu alarmów cykli użytkownika może dodatkowo zostać podany łańcuch znaków z max 4 parametrami. W tych parametrach mogą być definiowane zmienne teksty użytkownika. Są jednak do dyspozycji również następujące predefiniowane parametry:										
	<table><thead><tr><th>Parametry</th><th>Znaczenie</th></tr></thead><tbody><tr><td>%1</td><td>Numer kanału</td></tr><tr><td>%2</td><td>Numer bloku, etykieta</td></tr><tr><td>%3</td><td>Indeks tekstu dla alarmów cykli</td></tr><tr><td>%4</td><td>Dodatkowy parametr alarmu</td></tr></tbody></table>	Parametry	Znaczenie	%1	Numer kanału	%2	Numer bloku, etykieta	%3	Indeks tekstu dla alarmów cykli	%4	Dodatkowy parametr alarmu
Parametry	Znaczenie										
%1	Numer kanału										
%2	Numer bloku, etykieta										
%3	Indeks tekstu dla alarmów cykli										
%4	Dodatkowy parametr alarmu										

Wskazówka

Teksty alarmów muszą być projektowane na interfejsie graficznym.

Wskazówka

Jeżeli wyprowadzenie alarmu ma nastąpić w języku aktywnym na interfejsie graficznym, użytkownik potrzebuje informacji o języku aktualnie ustawionym na HMI. Tę informację można odczytać w programie obróbki i w akcjach synchronicznych poprzez zmienną systemową \$AN_LANGUAGE_ON_HMI (patrz Aktualny język w HMI (Strona 821)).

Przykład

Kod programu	Komentarz
...	
N100 SETAL(65000)	; Ustawienie alarmu nr 65000
...	

16.11 Rozszerzone zatrzymanie i wycofanie (ESR)

Funkcja

Funkcja "Rozszerzone zatrzymanie i wycofanie", dalej określana jako ESR, stwarza możliwość elastycznego reagowania na błędy w zależności od procesu:

- **Rozszerzone zatrzymanie**

Jeżeli specyficzna sytuacja błędu na to pozwala, wszystkie osie zwolnione do zatrzymania rozszerzonego są zatrzymywane w sposób uporządkowany.

- **Wycofanie**

Skrawające narzędzie jest jak najszybciej wycofywane od obrabianego przedmiotu.

- **Tryb generatora (funkcja napędowa SINAMICS "Regulacja Vdc")**

Przy spadku poniżej sparametryzowanej wartości napięcia obwodu pośredniego, np. w przypadku zaniku napięcia sieciowego, energia elektryczna potrzebna do wycofania jest wytwarzana przez odzysk energii hamowania przewidzianego w tym celu napędu (tryb generatora).

Źródła wyzwalania

Źródła ogólne (zewnętrzne dla NC / globalne albo specyficzne dla BAG/kanału)

- Wejścia cyfrowe (np. na zespole konstrukcyjnym NCU) wzgl. wewnętrzne w sterowaniu, odczytywalne odwzorowanie wyjść cyfrowych (\$A_IN, \$A_OUT)
- Stan kanału (\$AC_STAT)
- Sygnały VDI (\$A_DBB)
- Komunikaty zbiorcze liczby alarmów (\$AC_ALARM_STAT)

Źródła osiowe

- Próg wycofania awaryjnego osi holowanej (praca synchroniczna sprzężenia elektronicznego, \$VC_EG_SYNCDIFF[<oś nadążna>])
- Napęd: próg ostrzegania dla obwodu pośredniego (groźący niedobór napięcia), \$AA_ESR_STAT[<oś>]
- Napęd: próg minimalnej prędkości obrotowej generatora (nie ma już zdolnej do zwrotu energii ruchu obrotowego), \$AA_ESR_STAT[<oś>].

Logika powiązania statycznych akcji synchronicznych: powiązanie źródeł/reakcji

Elastyczne możliwości powiązania statycznych akcji synchronicznych są używane do tego, aby na podstawie źródeł w stosunkowo krótkim czasie wyzwać określone reakcje.

Powiązanie wszystkich odnośnych źródeł przy pomocy statycznych akcji synchronicznych leży w rękach użytkownika. Może on reagować na źródłowe zmienne systemowe jako całość, albo też selektywnie przy pomocy masek bitowych i dokonać odpowiedniego powiązania z pożądanymi reakcjami. Statyczne akcje synchroniczne działają we wszystkich tryby pracy.

Literatura:

Podręcznik działania Akcje synchroniczne

Uaktywnienie**Zwolnienie funkcji**

Funkcje pracy generatora, zatrzymania, wycofania, uzyskują zezwolenie przez ustawienie odnośnego sygnału sterującego \$AA_ESR_ENABLE. Sygnał ten może być zmieniany z akcji synchronicznych.

Wyzwolenie funkcji

ESR jest wyzwolane wspólnie dla wszystkich zwolnionych osi przez ustawienie zmiennej systemowej \$AC_ESR_TRIGGER.

Praca generatora staje się "automatycznie" aktywna w napędzie przy rozpoznaniu grożącego zbyt małego napięcia w obwodzie pośrednim.

Wystarczające dla napędu zatrzymanie i/albo wycofanie staje się aktywne przy rozpoznaniu przerwania komunikacji (między NC i napędem), jak też przy rozpoznaniu niedomiaru napięcia obwodu pośredniego w napędzie (zakładając konfigurację i zezwolenie).

Wystarczające dla napędu zatrzymanie i/albo wycofanie może dodatkowo zostać wyzwolone również ze strony NC przez ustawienie odpowiedniego sygnału sterującego \$AN_ESR_TRIGGER" (polecenie Broadcast dla wszystkich napędów).

Literatura

Informacje szczegółowe dot. ESR patrz:

Podręcznik działania Funkcje specjalne; Rozszerzone zatrzymanie i wycofanie (R3)

16.11.1 ESR prowadzone przez NC**16.11.1.1 Wycofanie prowadzone przez NC (POLF, POLFA, POLFMASK, POLFMLIN)****Funkcja**

Dla wycofania prowadzonego przez NC są wymagane określone warunki wyjściowe (patrz "Wycofanie prowadzone przez NC (POLF, POLFA, POLFMASK, POLFMLIN) (Strona 625)"). Gdy te warunki są spełnione, wówczas jest uaktywniane szybkie wycofanie (LIFTFAST) dla skonfigurowanych w kanale osi wycofania, przez ustawienie zmiennej systemowej \$AC_ESR_TRIGGER (lub \$AA_ESR_TRIGGER dla pojedynczych osi).

Składnia

```
POLF [<oś>]=<pozycja>
POLFA (<oś>,<typ>,<wartość>)
POLFMASK (<oś_1>,<oś_2>,...)
POLFMLIN (<oś_1>,<oś_2>,...)
```

Dla POLFA są dozwolone następujące formy skrócone:

```
POLFA (<oś>,<typ>) ; forma skrócona dla wycofania w pojedynczej osi
POLFA (oś,0/1/2) ; szybkie wyłączenie aktywności albo uaktywnienie
POLFA (oś,0,$AA_POLFA[oś]) ; powoduje zatrzymanie przebiegu
wyprzedzającego
POLFA (oś,0) ; nie powoduje zatrzymania przebiegu wyprzedzającego
```

Znaczenie

POLF:	<p>Adres dla podania pozycji docelowej osi wycofania</p> <p>POLF działa modalnie.</p> <p><oś>: Nazwa osi geometrycznej albo osi kanału/maszyny, w której następuje wycofanie</p> <p><pozycja>: Pozycja wycofania</p> <p>Typ: REAL</p> <p>Dla osi geometrycznej obowiązuje WKS, w innym przypadku MKS.</p> <p>W przypadku takich samych identyfikatorów dla osi geometrycznej i osi kanału/maszyny wycofanie następuje w układzie współrzędnych obrabianego przedmiotu.</p>
POLFA:	<p>Predefiniowane wywołanie podprogramu dla podania pozycji wycofania pojedynczych osi</p> <p><oś>: Identyfikator kanału</p> <p><Typ>: Tryb podania pozycji</p> <p>Typ: INT</p> <p>Wartość0: zaznaczenie wartości pozycji jako nie obowiązującej</p> <p>: 1: wartość pozycji jest absolutna 2: wartość pozycji jest przyrostowa (odległość)</p> <p>Uwaga: Jeżeli oś nie jest osią pojedynczą albo brakuje typu wzgl. typ=0, wówczas jest generowany odpowiedni alarm.</p> <p><pozycja>: Pozycja wycofania (patrz wyżej)</p> <p>Uwaga: Wartość pozycji jest przejmowana również z typem=0. Wartość ta jest wówczas jednak zaznaczona jako nie obowiązująca i musi dla wycofania zostać zaprogramowana na nowo.</p>

POLFMASK:	<p>Predefiniowane wywołanie podprogramu dla wyboru osi, w których po wyzwoleniu szybkiego cofnięcia ma nastąpić wycofanie niezależnie od siebie.</p> <p><oś_1>, ...: Nazwy osi, które w przypadku szybkiego cofnięcia powinny wykonać ruch do swoich pozycji zdefiniowanych przy pomocy POLF.</p> <p>Wszystkie podane osie muszą znajdować się w tym samym układzie współrzędnych.</p> <p>POLFMASK() bez podania osi wyłącza aktywność szybkiego odsunięcia dla wszystkich osi, które są wycofywane niezależnie od siebie.</p>
POLFMLIN:	<p>Predefiniowane wywołanie podprogramu do wyboru osi, w których po wyzwoleniu szybkiego cofnięcia ma nastąpić wycofanie w zależności liniowej.</p> <p><oś_1>, ...: patrz wyżej</p> <p>POLFMLIN() bez podania osi wyłącza aktywność szybkiego odsunięcia dla wszystkich osi, które są wycofywane w zależności liniowej.</p>

Wskazówka

Zanim poprzez POLFMASK albo POLFMLIN będzie można udzielić zezwolenia na szybkie cofnięcie do stałej pozycji, musi dla wybranych osi zostać zaprogramowana pozycja przy pomocy POLF.

Wskazówka

Jeżeli osie uzyskają kolejno zezwolenie przy pomocy POLFMASK, POLFMLIN albo POLFMLIN, POLFMASK, obowiązuje dla danej osi każdorazowo ostatnie ustalenie.

Wskazówka

Pozycje zaprogramowane przy pomocy POLF i uaktywnianie przez POLFMASK albo POLFMLIN są kasowane przy starcie programu obróbki. Oznacza to, że użytkownik musi w każdym programie obróbki na nowo zaprogramować wartości dla POLF i wybrane osie w POLFMASK wzgl. POLFMLIN.

Wskazówka

Jeśli przy stosowaniu form skróconych POLFA zostanie zmieniony tylko typ, wówczas użytkownik musi zagwarantować, by pozycja wycofania lub droga wycofania zawierała sensowną wartość. W szczególności należy na nowo nastawić pozycję wycofania i drogę wycofania po POWER ON.

Przykład

Wycofanie pojedynczej osi:

Kod programu	Komentarz
MD37500 \$MA_ESR_REACTION[AX1]=21	; Wycofanie prowadzone przez NC.
...	
\$AA_ESR_ENABLE[AX1]=1	
POLFA(AX1,1,20.0)	; Do AX1 jest przydzielana osiowa pozycja wycofania 20.0 (absolutna).
\$AA_ESR_TRIGGER[AX1]=1	; Od tego miejsca zaczyna się wycofanie.

Dalsze informacje

Warunki dla wycofania prowadzonego przez NC

- W Kanale jest zaprojektowana oś wycofania dla wycofania prowadzonego przez NC:
MD37500 \$MA_ESR_REACTION = 21
- ESR musi być zwolnione dla tej osi:
\$AA_ESR_ENABLE = 1
- Czasy zwłoki są zdefiniowane:
MD21380 \$MC_ESR_DELAY_TIME1
MD21381 \$MC_ESR_DELAY_TIME2
- W programie obróbki są zaprogramowane specyficzne dla osi pozycje wycofania z POLF.
- Osie są przy pomocy POLFMASK/POLFMLIN wybrane dla wycofania prowadzonego przez NC.
- Dla ruchu wycofania sygnały zezwoleń muszą być, jak i pozostać ustawione.

Zwolnienie i wystartowanie wycofania prowadzonego przez NC

Gdy jest ustawiona zmienna systemowa \$AC_ESR_TRIGGER = 1 i gdy w tym kanale jest skonfigurowana oś wycofania (tzn. MD37500: \$MA_ESR_REACTION = 21) i dla tej osi jest ustawiona \$AA_ESR_ENABLE = 1, wówczas jest w tym kanale uaktywniane szybkie wycofanie (LIFTFAST).

Skonfigurowany przy pomocy POLF (wzgl. LFPOS) ruch wycofania osi wybranej(ych) przy pomocy POLFMASK albo POLFMLIN zastępuje ruch po torze ustalony dla tej(tych) osi w programie obróbki.

Dla wycofania jest do dyspozycji maksymalnie suma czasów MD21380 \$MC_ESR_DELAY_TIME1 i MD21381 \$MC_ESR_DELAY_TIME2. Po upływie tego czasu jest również dla osi wycofania rozpoczynane szybkie hamowanie z następną aktualizacją.

Wskazówka

Rozszerzony ruch wycofania (tzn. LIFTFAST/LFPOS) wyzwolony przez \$AC_ESR_TRIGGER) jest **nieprzerwalny** i może zostać przedwcześnie zakończony tylko poprzez zatrzymanie awaryjne.

Wskazówka

Wycofanie wyzwolone przez \$AC_ESR_TRIGGER jest zablokowane przed wielokrotnym wycofaniem.

Wycofanie w pojedynczej osi

Przy wycofaniu w pojedynczej osi musi przy pomocy POLFA być zaprogramowana pozycja wycofania pojedynczej osi i muszą zostać dotrzymane następujące warunki:

- \$AA_ESR_ENABLE = 1
- <os> musi w chwili wyzwolenia (\$AA_ESR_TRIGGER = 1) być osią pojedynczą.
- <typ> musi być albo 1 albo 2.

Kierunek wycofania przy szybkim wycofaniu

W chwili uaktywnienia szybkiego wycofania jest uaktywniany obowiązujący Frame.

Wskazówka

Frame z obrotem wpływają poprzez POLF również na kierunek, w którym następuje wycofanie.

Zamiana osi

Osie wycofania należy zawsze przyporządkowywać dokładnie jednemu kanałowi NC i nie mogą one być zamieniane pomiędzy kanałami. Przy próbie zamiany osi wycofania z innym kanałem, jest generowany alarm. Dopiero gdy przy pomocy \$AA_ESR_ENABLE[AX] = 0 aktywność tej osi została wyłączona, można ją przełączyć na nowy kanał. Po dokonanej zamianie osi można je ponownie obciążyć przy pomocy \$AA_ESR_ENABLE[AX] = 1.

Osie neutralne

Osie neutralne nie mogą wykonywać ESR prowadzonego przez NC.

16.11.1.2 Zatrzymanie prowadzone przez NC**Funkcja**

Dla zatrzymania skonfigurowanego w kanale jest przez ustawienie zmiennej systemowej \$AC_ESR_TRIGGER (wzgl. \$AA_ESR_TRIGGER dla pojedynczych osi) uaktywniane zatrzymanie prowadzone przez NC.

Warunki

- W Kanale jest zaprojektowana oś zatrzymania dla zatrzymania prowadzonego przez NC:
MD37500 \$MA_ESR_REACTION = 22
- ESR musi być zwolnione dla tej osi:
\$AA_ESR_ENABLE = 1
- Czasy zwłoki są zdefiniowane:
MD21380 \$MC_ESR_DELAY_TIME1 (czas zwłoki osi ESR)
MD21381 \$MC_ESR_DELAY_TIME2 (czas ESR dla hamowania interpolacji)

Przebieg

Poprzez czas w MD21380 oś interpoluje bez przeszkód dalej, jak zaprogramowano. Po upływie czasu w MD21380 rozpoczynane jest interpolacyjnie prowadzone hamowanie (zatrzymanie według charakterystyki). Dla hamowania prowadzonego interpolacyjnie jest do dyspozycji maksymalnie przedział czasu w MD21381. Po upływie tego czasu jest rozpoczynane szybkie hamowanie z następną aktualizacją.

Przykład

Zatrzymanie pojedynczej osi:

Kod programu	Komentarz
MD37500 \$MC_ESR_REACTION[AX1]=22	; Zatrzymanie prowadzone przez NC.
MD21380 \$MC_ESR_DELAY_TIME1[AX1]=0.3	
MD21381 \$MC_ESR_DELAY_TIME2[AX1]=0.06	
...	
\$AA_ESR_ENABLE[AX1]=1	
\$AA_ESR_TRIGGER[AX1]=1	; Od tego miejsca rozpoczyna się zatrzymanie.

16.11.2 ESR wystarczające dla napędu

16.11.2.1 Zaprojektowanie zatrzymania niezależnego dla napędu (ESRS)

Funkcja

Przy pomocy funkcji `ESRS(...)` są projektowane parametry napędu dla "zatrzymania" niezależnego dla napędu funkcji ESR.

Składnia

```
ESRS(<oś_1>,<czas zatrzymania_1>[,...,<oś_n>,<czas zatrzymania_n>])
```

Znaczenie

ESRS (...):	Funkcja dla zapisu parametrów napędu dla funkcji ESR "zatrzymanie"
	Funkcja:
	<ul style="list-style-type: none"> • musi znajdować się w oddzielnym bloku. • wyzwala zatrzymanie przebiegu wyprzedzającego. • nie może być stosowana w akcjach synchronicznych.
<oś_1>, ... , <oś_n>:	<p>Oś, która powinna zostać zaprojektowana dla zatrzymania wystarczającego dla napędu</p> <p>W napędzie jest dla tej osi pisany parametr napędu p0888 (konfiguracja):</p> <p>p0888 = 1</p> <p>Typ: AXIS</p> <p>Zakres wartości: Identyfikator kanału</p>
<czas zatrzymania_1>, ... , <czas zatrzymania_n>:	<p>Okres czasu, w którym napęd po wystąpieniu błędu nadal wykonuje ruch z aktualną wartością zadaną prędkości obrotowej</p> <p>W napędzie jest dla podanej osi pisany parametr napędu p0892 (stopień czasowy):</p> <p>p0892 = <czas zatrzymania></p> <p>Jednostka: s</p> <p>Typ: REAL</p> <p>Zakres wartości: 0.00 - 20.00</p> <p>W jednym wywołaniu funkcji można zaprogramować maksymalnie 5 osi; n = 5</p>

16.11.2.2 Zaprojektowanie wycofania niezależnego dla napędu (ESRR)

Funkcja

Przy pomocy funkcji `ESRR(...)` są projektowane parametry napędu dla "wycofania" niezależnego dla napędu funkcji ESR.

Składnia

```
ESRR(<oś_1>, <droga wycofania_1>, <prędkość wycofania_1> [, ... , <oś_n>, <droga wycofania_n>, <prędkość wycofania_n>])
```

Znaczenie

ESRR(...):	<p>Funkcja do zapisu parametrów napędu dla funkcji ESR "wycofanie"</p> <p>Funkcja:</p> <ul style="list-style-type: none"> • musi znajdować się w oddzielnym bloku. • wyzwala zatrzymanie przebiegu wyprzedzającego. • nie może być stosowana w akcjach synchronicznych.
<oś_1>, ..., <oś_n>:	<p>Oś, dla której ma zostać zaprojektowane wycofanie wystarczające dla napędu</p> <p>W napędzie jest dla tej osi pisany parametr napędu p0888 (konfiguracja):</p> <p>p0888 = 2</p> <p>Typ: AXIS</p> <p>Zakres wartości: Identyfikator kanału</p>
<droga wycofania_1>, ..., <droga wycofania_n>:	<p>Droga wycofania jest dla napędu przeliczana na prędkość obrotową wycofania. Wartość jest dla podanej osi zapisywana w parametrze napędu p0893 (prędkość obrotowa):</p> <p>p0893 = (<droga wycofania _n> przeliczona na prędkość obrotową wycofania)</p> <p>Jednostka: mm/min, cale/min, stopni/min (zależnie od jednostki osi)</p> <p>Typ: REAL</p> <p>Zakres wartości: MIN - MAX</p>
<Prędkość wycofania_1>, ..., <Prędkość wycofania_n>:	<p>Prędkość wycofania jest dla napędu przeliczana na czas trwania. Wartość jest dla podanej osi zapisywana w parametrze napędu p0892 (stopień czasowy) [s]:</p> <p>p0892 = <droga wycofania_n> / <prędkość wycofania _n></p> <p>Jednostka: mm/min, cale/min, stopni/min (zależnie od jednostki osi)</p> <p>Typ: REAL</p> <p>Zakres wartości: 0.00 - MAX</p>

W jednym wywołaniu funkcji można zaprogramować maksymalnie 5 osi; n = 5

Własne programy obróbki

17.1 Funkcje wspierające skrawanie

Funkcje

Do skrawania są udostępniane gotowe cykle obróbkowe. Ponadto jest możliwość sporządzania własnych programów skrawania, przy pomocy niżej wymienionych funkcji.

- Sporządzenie tablicy konturu (CONTPRON)
- Sporządzenie kodowanej tablicy konturu (CONTDCON)
- Wyłączenie przygotowania konturu (EXECUTE)
- Obliczenie punktu przecięcia między dwoma elementami konturu (INTERSEC)
(Tylko dla tablic, które zostały sporządzone przez CONTPRON.)
- Wykonywanie pojedynczymi blokami elementów konturowych tablicy (EXECTAB)
(Tylko dla tablic, które zostały sporządzone przez CONTPRON.)
- Obliczenie danych okręgu (CALCDAT)

Wskazówka

Można stosować te funkcje nie tylko do skrawania, lecz uniwersalnie.

Warunki

Przed wywołaniem funkcji CONTPRON albo CONTDCON musi:

- nastąpić ruch do punktu startowego, który pozwala na bezkolizyjną obróbkę.
- być wyłączona korekcja promienia ostrza przy pomocy G40.

17.2 Sporządzenie tablicy konturu (CONTPRON)

Funkcja

Przy pomocy polecenia `CONTPRON` jest włączane przygotowanie konturu. Następnie wywoływane bloki NC nie są wykonywane, lecz dzielone na poszczególne ruchy i zapisywane w tablicy konturu. Każdemu elementowi konturu odpowiada jeden wiersz w dwuwymiarowej tablicy konturu. Jest zwracana liczba obliczonych podcięć.

Składnia

Włączenie przygotowania konturu:

`CONTPRON (<tablica konturu>, <rodzaj obróbki>, <podcięcia>, <kierunek obróbki>)`

Wyłączenie przygotowania konturu i powrót do normalnego trybu pracy:

`EXECUTE (<BŁĄD>)`

Patrz "Wyłączenie przygotowania konturu (EXECUTE) (Strona 648)"

Znaczenie

`CONTPRON`

Polecenie do włączenia przygotowania konturu do sporządzenia tablicy konturu

`<tablica konturu>`

Nazwa tablicy konturu

`<rodzaj obróbki>`

Parametr rodzaju obróbki

Typ: CHAR

Wartość: "G" Toczenie wzdłużne: Obróbka wewnętrzna

"L" Toczenie wzdłużne: Obróbka zewnętrzna

"N" Toczenie poprzeczne: Obróbka wewnętrzna

"P" Toczenie poprzeczne: Obróbka zewnętrzna

`<podcięcia>`

Zmienna wynikowa liczby występujących elementów podcięć

Typ: INT

`<kierunek obróbki>`

Parametr kierunku obróbki

Typ: INT

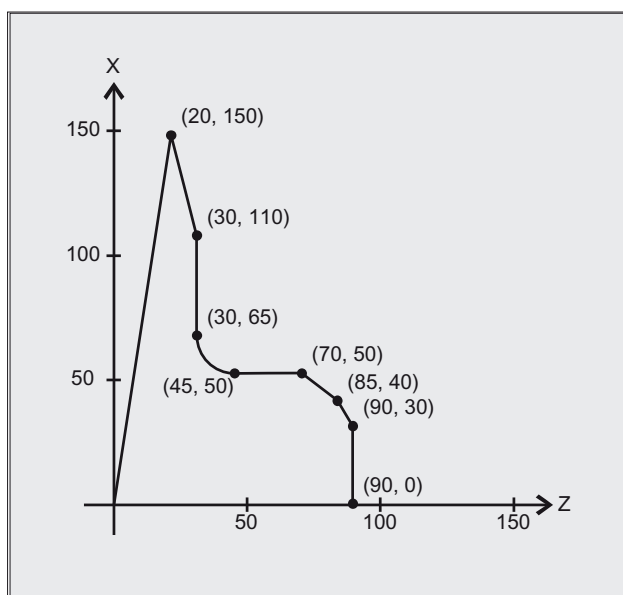
Wartość: 0 Przygotowanie konturu do przodu (wartość standardowa)

1 Przygotowanie konturu w obydwu kierunkach

Przykład 1

Sporządzenie tablicy konturu z:

- nazwą "KTAB"
- max 30 elementami konturu (okręgi, proste)
- zmienną wynikową liczby występujących elementów podcięć
- zmienną komunikatów błędów



Program NC:

Kod programu	Komentarz
N10 DEF REAL KTAB[30,11]	; Tablica konturu o nazwie KTAB i max 30 elementami konturu, wartość parametru 11 (liczba kolumn tablicy) jest wielkością stałą.
N20 DEF INT ANZHINT	; Zmienna liczby elementów podcięć o nazwie ANZHINT.
N30 DEF INT BŁAD	; Zmienna dla zwrotnego komunikatu błędu (0=nie ma błędu, 1=błąd).
N40 G18	
N50 CONTPRON(KTAB,"G",ANZHINT)	; Włączenie przygotowania konturu.
N60 G1 X150 Z20	; N60 do N120: opis konturu
N70 X110 Z30	
N80 X50 RND=15	
N90 Z70	
N100 X40 Z85	
N110 X30 Z90	
N120 X0	
N130 EXECUTE(BŁAD)	; Zakończenie wypełniania tablicy konturu, przełączenie na normalną pracę programową.
N140 ...	; Dalsze opracowywanie tablicy.

Tablica konturu KTAB:

Indeks Wiersz z	Kolumna									
	(0)	(1)	(2)	(3)	(4)	(5)	(6)	(7)	(8)	(9)
7	7	11	0	0	20	150	0	82.40535663	0	0
0	2	11	20	150	30	110	-1111	104.0362435	0	0
1	3	11	30	110	30	65	0	90	0	0
2	4	13	30	65	45	50	0	180	45	65
3	5	11	45	50	70	50	0	0	0	0
4	6	11	70	50	85	40	0	146.3099325	0	0
5	7	11	85	40	90	30	0	116.5650512	0	0
6	0	11	90	30	90	0	0	90	0	0
0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0

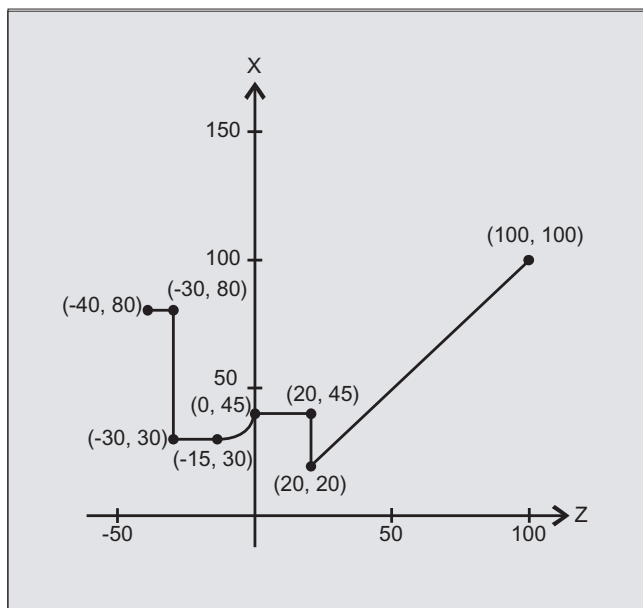
Objaśnienie treści kolumn:

- (0) Wskaźnik na następny element konturu (na jego numer wiersza)
- (1) Wskaźnik na poprzedni element konturu
- (2) Kodowanie trybu konturu dla ruchu
Możliwe wartości dla X = abc
a = 10² G90 = 0 G91 = 1
b = 10¹ G70 = 0 G71 = 1
c = 10⁰ G0 = 0 G1 = 1 G2 = 2 G3 = 3
- (3), (4) Punkt początkowy elementów konturu
(3) = odcięta, (4) = rzędna w aktualnej płaszczyźnie
- (5), (6) Punkt końcowy elementów konturu
(5) = odcięta, (6) = rzędna w aktualnej płaszczyźnie
- (7) Wskaźnik max/min : oznacza lokalne maksima i minima w konturze
- (8) Wartość maksymalna między elementem konturu i odcięta (przy obróbce wzdłużnej) wzgl. odcięta (przy obróbce poprzecznej). Kąt jest zależny od zaprogramowanego rodzaju obróbki.
- (9), (10) Współrzędne punktu środkowego elementu konturu, gdy jest to blok kołowy.
(9) = odcięta, (10) = rzędna

Przykład 2

Sporządzenie tablicy konturu z

- nazwą KTAB
- max 92 elementami konturu (okręgi, proste)
- tryb pracy: toczenie wzdłużne, obróbka zewnętrzna
- przygotowaniem do przodu i do tyłu



Program NC:

Kod programu	Komentarz
N10 DEF REAL KTAB[92,11]	; Tablica konturu o nazwie KTAB i max 92 elementami konturu, wartość parametru 11 jest wielkością stałą.
N20 DEF CHAR BT="L"	; Tryb pracy dla CONTPRON: toczenie wzdłużne, obróbka zewnętrzna
N30 DEF INT HE=0	; Liczba elementów podcięcia =0
N40 DEF INT MODE=1	; Przygotowanie do przodu i do tyłu
N50 DEF INT ERR=0	; Zwrotna sygnalizacja błędu
...	
N100 G18 X100 Z100 F1000	
N105 CONTPRON(KTAB,BT,HE,MODE)	; Włączenie przygotowania konturu.
N110 G1 G90 Z20 X20	
N120 X45	
N130 Z0	
N140 G2 Z-15 X30 K=AC(-15) I=AC(45)	
N150 G1 Z-30	

Kod programu	Komentarz
N160 X80	
N170 Z-40	
N180 EXECUTE (ERR)	; Zakończenie wypełniania tablicy konturu, przełączenie na normalną pracę programową.
...	

Tablica konturu KTAB:

Po zakończeniu przygotowania konturu, kontur jest do dyspozycji w obydwu kierunkach.

Indeks	Kolumna										
	(0)	(1)	(2)	(3)	(4)	(5)	(6)	(7)	(8)	(9)	(10)
0	6 ¹⁾	7 ²⁾	11	100	100	20	20	0	45	0	0
1	0 ³⁾	2	11	20	20	20	45	-3	90	0	0
2	1	3	11	20	45	0	45	0	0	0	0
3	2	4	12	0	45	-15	30	5	90	-15	45
4	3	5	11	-15	30	-30	30	0	0	0	0
5	4	7	11	-30	30	-30	45	-1111	90	0	0
6	7	0 ⁴⁾	11	-30	80	-40	80	0	0	0	0
7	5	6	11	-30	45	-30	80	0	90	0	0
8	1 ⁵⁾	2 ⁶⁾	0	0	0	0	0	0	0	0	0
	...										
83	84	0 ⁷⁾	11	20	45	20	80	0	90	0	0
84	90	83	11	20	20	20	45	-1111	90	0	0
85	0 ⁸⁾	86	11	-40	80	-30	80	0	0	0	0
86	85	87	11	-30	80	-30	30	88	90	0	0
87	86	88	11	-30	30	-15	30	0	0	0	0
88	87	89	13	-15	30	0	45	-90	90	-15	45
89	88	90	11	0	45	20	45	0	0	0	0
90	89	84	11	20	45	20	20	84	90	0	0
91	83 ⁹⁾	85 ¹⁰⁾	11	20	20	100	100	0	45	0	0

Objaśnienie treści kolumn i uwagi do wierszy 0, 1, 6, 8, 83, 85 i 91

Obowiązują wyjaśnienia treści kolumn wymienione w przykładzie 1.

Zawsze w wierszu 0 tablicy:

- 1) Poprzedzający: wiersz n zawiera koniec konturu w kierunku do przodu
- 2) Następny: wiersz n jest końcem tablicy konturu w kierunku do przodu

Po jednym razie w ramach elementów konturu do przodu:

- 3) Poprzedzający: początek konturu (w kierunku do przodu)
- 4) Następny: koniec konturu (w kierunku do przodu)

Zawsze na wiersz końca tablicy konturu (do przodu) +1:

5) Poprzedzający: liczba podcięć w kierunku do przodu

6) Następny: liczba podcięć w kierunku do tyłu

Po jednym razie w ramach elementów konturu do tyłu:

7) Następny: koniec konturu (w kierunku do przodu)

8) Poprzedzający: początek konturu (w kierunku do tyłu)

Zawsze w ostatnim wierszu tablicy:

9) Poprzedzający: wiersz n jest początkiem tablicy konturu (w kierunku do tyłu)

10) Następny: wiersz n zawiera początek konturu (w kierunku do tyłu)

Dalsze informacje**Dozwolone polecenia ruchu, układ współrzędnych**

Dla programowania konturu są dopuszczalne następujące polecenia G:

- Grupa G 1: G0, G1, G2, G3

Dodatkowo możliwe są:

- Zaokrąglenie i faza
- Programowanie okręgu przez CIP i CT

Funkcje spline, wielomian i gwint prowadzą do błędów.

Zmiany układu współrzędnych przez włączenie frame są między CONTPRON i EXECUTE niedopuszczalne. To samo dotyczy zmiany między G70 i G71 wzgl. G700 i G710.

Zamiana osi geometrycznych przy pomocy GEOAX podczas przygotowywania tablicy konturu prowadzi do alarmu.

Elementy podcięć

Opis konturu poszczególnych elementów podcięć może nastąpić do wyboru w podprogramie albo w poszczególnych blokach.

Skrawanie niezależne od zaprogramowanego kierunku konturu

Przygotowanie konturu z CONTPRON zostało tak rozszerzone, że po jego wywołaniu tablica konturu jest do dyspozycji niezależnie od zaprogramowanego kierunku.

17.3 Sporządzenie kodowanej tablicy konturu (CONTDCON)

Funkcja

Przy przygotowywaniu konturu włączonym przy pomocy `CONTDCON` następnie wywoływane bloki są korzystnie dla pamięci zapisywane w formie kodowanej w 6-kolumnowej tablicy konturu. Każdemu elementowi konturu odpowiada jeden wiersz tablicy. Znając niżej podane kodowania można np. dla cykli zestawiać programy DIN-Code z wierszy tablicy. W wierszu tablicy o numerze 0 są zapisywane dane punktu wyjściowego.

Składnia

Włączenie przygotowania konturu:

`CONTDCON (<tablica konturu>, <kierunek obróbki>)`

Wyłączenie przygotowania konturu i powrót do normalnego trybu pracy:

`EXECUTE (<BŁĄD>)`

Patrz "Wyłączenie przygotowania konturu (EXECUTE) (Strona 648)"

Znaczenie

`CONTDCON`

Polecenie do włączenia przygotowania konturu do sporządzenia kodowanej tablicy konturu

`<tablica konturu>`

Nazwa tablicy konturu

`<kierunek obróbki>`

Parametr kierunku obróbki

Typ: INT

Wartość: 0 Przygotowanie konturu według sekwencji bloków konturowych (wartość standardowa)

1 jest niedopuszczalne

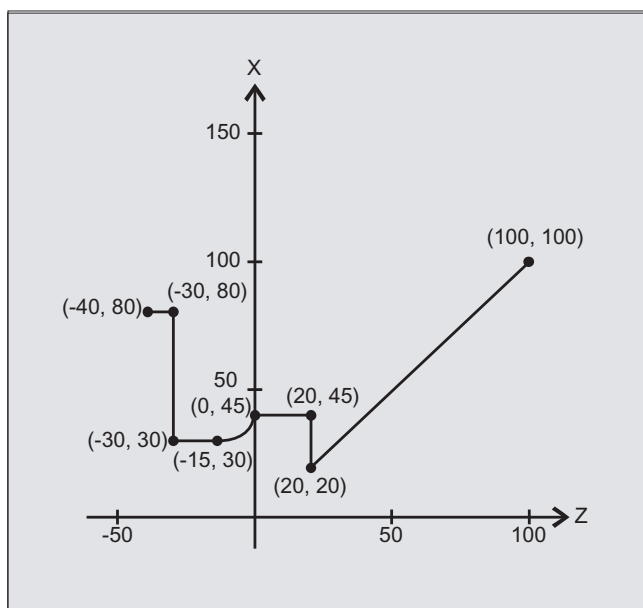
Wskazówka

G-Code dopuszczone dla `CONTDCON` w tablicowanym segmencie programu są bardziej obszerne, niż w przypadku `CONTPRON`. Ponadto są równocześnie zapisywane posuwy i typ posuwu na segment konturu.

Przykład

Sporządzenie tablicy konturu z:

- nazwą "KTAB"
- elementami konturu (okręgi, proste)
- Tryb pracy: Toczenie
- Kierunek obróbki: do przodu



Program NC:

Kod programu	Komentarz
N10 DEF REAL KTAB[9,6]	; Tablica konturu o nazwie KTAB i 9 wierszach. Pozwalają one na 8 bloków konturu. Wartość parametru 6 (liczba kolumn tablicy) jest stałą wielkością.
N20 DEF INT MODE = 0	; Zmienna kierunku obróbki. Wartość standardowa 0: tylko w zaprogramowanym kierunku konturu.
N30 DEF INT ERROR = 0	; Zmienna dla zwrotnego komunikatu błędu.
...	
N100 G18 G64 G90 G94 G710	
N101 G1 Z100 X100 F1000	
N105 CONTDCON (KTAB, MODE)	; Wywołanie przygotowania konturu (MODE wolno pominąć).
N110 G1 Z20 X20 F200	; Opis konturu.
N120 G9 X45 F300	
N130 Z0 F400	
N140 G2 Z-15 X30 K=AC(-15) I=AC(45) F100	
N150 G64 Z-30 F600	
N160 X80 F700	
N170 Z-40 F800	
N180 EXECUTE(ERROR)	; Zakończenie wypełniania tablicy konturu, przełączenie na normalną pracę programową.
...	

Tablica konturu KTAB:

	Indeks kolumny					
	0	1	2	3	4	5
Indeks wiersza	Tryb konturu	Punkt końcowy odcięta	Punkt końcowy rzędna	Punkt środkowy odcięta	Punkt środkowy rzędna	Posuw
0	30	100	100	0	0	7
1	11031	20	20	0	0	200
2	111031	20	45	0	0	300
3	11031	0	45	0	0	400
4	11032	-15	30	-15	45	100
5	11031	-30	30	0	0	600
6	11031	-30	80	0	0	700
7	11031	-40	80	0	0	800
8	0	0	0	0	0	0

Objaśnienie treści kolumn:

Wiersz 0: kodowanie dla punktu startowego:

- Kolumna 0: 10^0 (miejsce jednostek): G0 = 0
 10^1 (miejsce dziesiątek): G70 = 0, G71 = 1, G700 = 2, G710 = 3
- Kolumna 1: Punkt startowy odcięta
- Kolumna 2: Punkt startowy rzędna
- Kolumna 3-4: 0
- Kolumna 5: Indeks wiersza ostatniego elementu konturu w tablicy

Wiersze 1-n: wpisy elementów konturu

- Kolumna 0: 10^0 (miejsce jednostek): G0 = 0, G1 = 1, G2 = 2, G3 = 3
 10^1 (miejsce dziesiątek): G70 = 0, G71 = 1, G700 = 2, G710 = 3
 10^2 (miejsce setek): G90 = 0, G91 = 1
 10^3 (miejsce tysięcy): G93 = 0, G94 = 1, G95 = 2, G96 = 3
 10^4 (miejsce dziesiątek tysięcy): G60 = 0, G44 = 1, G641 = 2, G642 = 3
 10^5 (miejsce setek tysięcy): G9 = 1
- Kolumna 1: Punkt końcowy odcięta
- Kolumna 2: Punkt końcowy rzędna
- Kolumna 3: Punkt środkowy odcięta przy interpolacji kołowej
- Kolumna 4: Punkt środkowy rzędna przy interpolacji kołowej
- Kolumna 5: Posuw

Dalsze informacje

Dozwolone polecenia ruchu, układ współrzędnych

Dla programowania konturu są dopuszczalne następujące grupy G i polecenia G:

Grupa G1:	G0, G1, G2, G3
Grupa G10:	G60, G64, G641, G642
Grupa G11:	G9
Grupa G13:	G70, G71, G700, G710
Grupa G14:	G90, G91
Grupa G15:	G93, G94, G95, G96, G961

Dodatkowo możliwe są:

- Zaokrąglenie i faza
- Programowanie okręgu przez `CIP` i `CT`

Funkcje spline, wielomian i gwint prowadzą do błędów.

Zmiany układu współrzędnych przez włączenie frame są między `CONTDCON` i `EXECUTE` niedopuszczalne. To samo dotyczy zmiany między `G70` i `G71` wzgl. `G700` i `G710`.

Zamiana osi geometrycznych przy pomocy `GEOAX` podczas przygotowywania tablicy konturu prowadzi do alarmu.

Kierunek obróbki

Tablica konturu utworzona przy pomocy `CONTDCON` jest przewidziana do skrawania w zaprogramowanym kierunku konturu.

17.4 Obliczenie punktu przecięcia między dwoma elementami konturu (INTERSEC)

Funkcja

`INTERSEC` określa punkt przecięcia dwóch znormalizowanych elementów konturu z tablicy konturu utworzonej przy pomocy `CONTPRON`.

Składnia

```
<Status>=INTERSEC(<tablica konturu_1>[<element konturu_1>],
<tablica konturu_2>[<element konturu_2>], <punkt przecięcia>, <rodzaj
obróbki>)
```

Znaczenie

INTERSEC	Słowo kluczowe do obliczenia punktu przecięcia dwóch elementów konturu z tablicy konturu utworzonej przy pomocy <code>CONTPRON</code>
<status>	Zmienna statusu punktu przecięcia Typ: <code>BOOL</code> Wartość: <code>TRUE</code> Punkt przecięcia znaleziony <code>FALSE</code> Nie znaleziono punktu przecięcia
<tablica konturu_1>	Nazwa pierwszej tablicy konturu
<element konturu_1>	Numer elementu konturu pierwszej tablicy konturu
<tablica konturu_2>	Nazwa drugiej tablicy konturu
<element konturu_2>	Numer elementu konturu drugiej tablicy konturu
<punkt przecięcia>	Współrzędne punktu przecięcia w aktywnej płaszczyźnie (<code>G17 / G18 / G19</code>) Typ: <code>REAL</code>
<rodzaj obróbki>	Parametr rodzaju obróbki Typ: <code>INT</code> Wartość: <code>0</code> Obliczenie punktu przecięcia w płaszczyźnie aktywnej z parametrem 2 (wartość standardowa) <code>1</code> Obliczenie punktu przecięcia niezależnie od przekazanej płaszczyzny

Wskazówka

Należy pamiętać, że zmienne przed ich zastosowaniem muszą być zdefiniowane.

Przekazanie konturów wymaga dotrzymania wartości zdefiniowanych przy pomocy `CONTPRON`:

Parametry	Znaczenie
2	Kodowanie trybu konturu dla ruchu
3	Punkt początkowy konturu odcięta
4	Punkt początkowy konturu rzędna
5	Punkt końcowy konturu odcięta
6	Punkt końcowy konturu rzędna
9	Współrzędna punktu środkowego dla odciętej (tylko w przypadku konturu kołowego)
10	Współrzędna punktu środkowego dla rzędnej (tylko w przypadku konturu kołowego)

Przykład

Określenie punktu przecięcia elementu 3 konturu tablicy TABNAME1 i elementu 7 konturu tablicy TABNAME2. Współrzędne punktu przecięcia w aktywnej płaszczyźnie są zapisywane w zmiennej ISCOORD (1. element = odcięta, 2. element = rzędna). Jeżeli punkt przecięcia nie istnieje, następuje przeskok do KEINSCH (nie znaleziono punktu przecięcia).

Kod programu	Komentarz
DEF REAL TABNAME1[12,11]	; Tablica 1 konturu
DEF REAL TABNAME2[10,11]	; Tablica 2 konturu
DEF REAL ISCOORD[2]	; Zmienna współrzędnych punktu przecięcia.
DEF BOOL ISPOINT	; Zmienna statusu punktu przecięcia.
DEF INT MODE	; Zmienna rodzaju obróbki.
...	
MODE=1	; Obliczenie niezależnie od aktywnej płaszczyzny.
N10 ISPOINT=INTERSEC(TABNAME1[3],TABNAME2[7],ISCOORD,MODE)	; Wywołanie punktu przecięcia elementów konturu.
N20 IF ISPOINT==FALSE GOTOF KEINSCH	; Skok do KEINSCH.
...	

17.5 Przejście pojedynczymi blokami elementów konturu w tablicy (EXECTAB)

Funkcja

Przy pomocy polecenia EXECTAB można pojedynczymi blokami wykonywać elementy konturowe tablicy, które zostały utworzone np. poleceniem CONTPRON.

Składnia

```
EXECTAB(<tablica konturu>[<element konturu>])
```

Znaczenie

EXECTAB	Polecenie wykonania ruchu po elemencie konturu
<tablica konturu>	Nazwa tablicy konturu
<element konturu>	Numer elementu konturu

Przykład

Ruchy po elementach konturu 0 do 2 tablicy KTAB mają być wykonywane pojedynczymi blokami.

Kod programu	Komentarz
N10 EXECTAB(KTAB[0])	; Ruch po elemencie 0 tablicy KTAB.
N20 EXECTAB(KTAB[1])	; Ruch po elemencie 1 tablicy KTAB.
N30 EXECTAB(KTAB[2])	; Ruch po elemencie 2 tablicy KTAB.

17.6 Obliczenie danych okręgu (CALCDAT)

Funkcja

Przy pomocy polecenia `CALCDAT` można z trzech albo czterech znanych punktów okręgu obliczyć promień i współrzędne punktu środkowego okręgu. Podane punkty muszą być różne. W przypadku 4 punktów, które nie leżą dokładnie na okręgu, jest dla punktu środkowego okręgu i promienia wybierana wartość średnia.

Składnia

```
<Status>=CALCDAT(<punkty
okręgu>[<liczba>,<rodzaj>],<liczba>,<wynik>)
```

Znaczenie

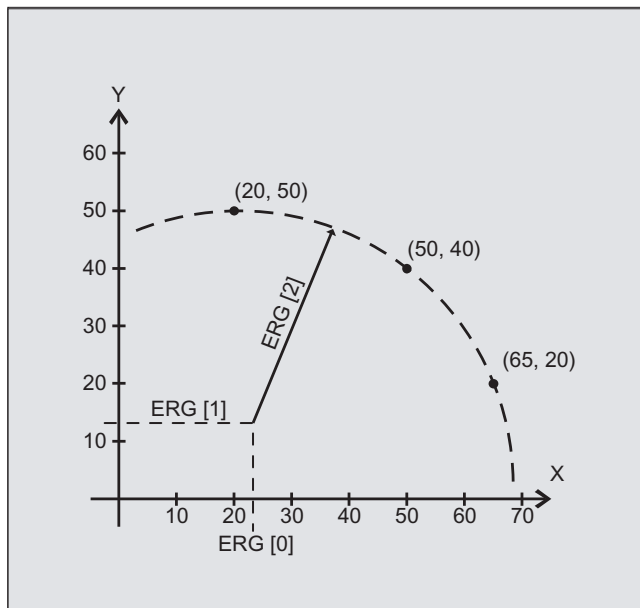
CALCDAT	Polecenie do obliczenia promienia i współrzędnych punktu środkowego okręgu z 3 albo 4 punktów
<status>	Zmienna statusu obliczenia okręgu Typ: BOOL Wartość: TRUE Podane punkty leżą na okręgu. FALSE Podane punkty nie leżą na okręgu.
<punkty na okręgu>[]	Zmienna do podania punktów okręgu z parametrami: <liczba> Liczba punktów na okręgu (3 albo 4) <rodzaj> Rodzaj podania współrzędnych, np. 2 dla współrzędnych 2-punktowych
<liczba>	Parametr liczby punktów stosowanych do obliczenia (3 albo 4)
<wynik>[3]	Zmienna dla wyniku: Podanie współrzędnych punktu środkowego okręgu i promienia 0 Współrzędna punktu środkowego okręgu: wartość ujemnej 1 Współrzędna punktu środkowego okręgu: wartość rzędnej 2 Promień

Wskazówka

Należy pamiętać, że zmienne przed ich zastosowaniem muszą być zdefiniowane.

Przykład

Z trzech punktów ma nastąpić obliczenie, czy leżą one na jednym segmencie okręgu.



Kod programu	Komentarz
N10 DEF REAL PKT [3,2] = (20,50,50,40,65,20)	; Zmienna do podania punktów okręgu
N20 DEF REAL ERG [3]	; Zmienna dla wyniku
N30 DEF BOOL STATUS	; Zmienna statusu
N40 STATUS=CALCDAT (PKT,3,ERG)	; Wywołanie obliczonych danych okręgu.
N50 IF STATUS == FALSE GOTOF ERROR	; Skok do błędu

17.7 Wyłączenie przygotowania konturu (EXECUTE)

Funkcja

Przy pomocy polecenia `EXECUTE` następuje wyłączenie przygotowania konturu i równocześnie przełączenie z powrotem na normalny tryb wykonywania.

Składnia

`EXECUTE (<BŁĄD>)`

Znaczenie

<code>EXECUTE</code>	Polecenie do zakończenia przygotowania konturu
<code><BŁĄD></code>	Zmienna sygnalizacji zwrotnej błędu
	Typ: INT
	Wartość zmiennej pokazuje, czy można było bezbłędnie przygotować kontur:
0	Błąd
1	Nie ma błędu

Przykład

```
Kod programu
...
N30 CONTPRON(...)
N40 G1 X... Z...
...
N100 EXECUTE(...)
...
```


Zewnętrzne programowanie cykli

18.1 Cykle technologiczne

18.1.1 Wprowadzenie

Treść

W tym punkcie są udokumentowane cykle technologiczne od wersji 2.6 do sporządzania zewnętrznych programów NC.

Budowa

Dokumentacja ma następującą strukturę:

- **Programowanie**
Nazwa cyklu i kolejność wywoływania przekazywanych parametrów
- **Parametry**
Tablica do objaśnienia poszczególnych parametrów

Opis parametrów

W tablicy znajdują się nazwy wewnętrznie stosowanych parametrów i objaśnienie dot. znaczenia i możliwego zakresu wartości. Poza tym objaśniono zależności między parametrami. Kolumna odsyłania do parametru w oknie ma służyć w tym celu, by przy przetwarzaniu wstecznym wygenerowanych na zewnątrz wywołań cykli znaleźć ponownie zaprogramowane wartości w sterowaniu.

Parametr "tylko dla powierzchni"

W tablicy parametry są oznakowane przez "tylko dla powierzchni". Nie mają one znaczenia dla funkcji cyklu. Są one potrzebne tylko do tego, aby wywołania cykli móc poddać kompletnemu przetwarzaniu wstecznemu. Jeżeli nie są one zaprogramowane, można mimo to poddać cykl przetwarzaniu wstecznemu, pola są wówczas odpowiednio zaznaczone kolorem i muszą zostać wypełnione w oknie.

Parametr "zarezerwowano"

Parametry, które są opisane przez "zarezerwowano", muszą być programowane z 0 albo spacją między przecinkami, aby przyporządkowanie następnych wywoływanych parametrów pasowało do wewnętrznych parametrów cyklu. Wyjątek: w przypadku parametrów string wartość "" albo spacja między przecinkami.

Kompatybilność

Cykle technologiczne od wersji 2.6 stanowią rozwinięcie pakietów cykli dla 840Dsl do GIV 1.5 (cykle do wersji 7.5). Programy NC z wywołaniami cykli tych wcześniejszych wersji oprogramowania mogą być nadal wykonywane.

Większość cykli została rozszerzona o nowe przekazywane parametry albo został rozszerzony zakres wartości istniejących parametrów, aby móc programować nowe funkcje (jak np. często stosowany parametr `_VARI` dla rodzaju obróbki).

Pojęcie "kompatybilność" w tej dokumentacji wskazuje na wprowadzane wartości, które nie były przedtem programowane. Gdy wartości zostaną zgodnie z tym wprowadzone, cykl przebiega tak, jak w wersji 7.5.

Powtarzanie cykli na szablonie pozycji

Cykle wiercenia i frezowania mogą być powtarzane na szablonie pozycji (wywołania modalne). Przed cyklem należy wówczas w tym samym wierszu napisać `MCALL`, np. `MCALL CYCLE83 (...)`.

Wskazówka

Jeżeli określone przekazywane parametry (np. `_VARI`, `_GMODE`, `_DMODE`, `_AMODE`) są zaprogramowane pośrednio jako parametry, jest przy przetwarzaniu wstecznym otwierane okno wprowadzania, nie może jednak zostać zapisane, ponieważ do określonych pól wyboru nie ma jednoznacznego przyporządkowania.

18.1.2 Wiercenie, nawiercanie - CYCLE81

Programowanie

CYCLE81 (REAL RTP, REAL RFP, REAL SDIS, REAL DP, REAL DPR, REAL _DTB,
INT _GMODE, INT _DMODE, INT _AMODE)

Parametry

Nr	Okno param etrów	Parametry wewnętrzne	Objaśnienie
1	RP	RTP	Płaszczyzna wycofania (abs)
2	Z0	RFP	Punkt odniesienia (abs)
3	SC	_SDIS	Odstęp bezpieczeństwa (addytywnie do punktu odniesienia, wprowadzić bez znaku)
4	Z1/∅	_DP	Głębokość wiercenia (abs)/średnica nakiełkowania (abs), patrz _GMODE
5	Z1	-DPR	Głębokość wiercenia (przr.)
6	DT	_DTB	Czas oczekiwania na końcowej głębokości wiercenia, patrz _AMODE
7		_GMODE	Tryb geometrii (oszacowanie zaprogramowanych wartości geometrycznych) JEDNOSTKI: Zarezerwowano DZIESIĄTKI: Nawiercanie odniesione do głębokości/średnicy 0 = kompatybilność, głębokość 1 = średnica
8		_DMODE	Tryb wyświetlania JEDNOSTKI: Płaszczyzna obróbki G17/G18/G19 0 = kompatybilność, aktywna pozostaje płaszczyzna działająca przed wywołaniem cyklu 1 = G17 (aktywna tylko w cyklu) 2 = G18 (aktywna tylko w cyklu) 3 = G19 (aktywna tylko w cyklu)
9		_AMODE	Tryb alternatywny JEDNOSTKI: Głębokość wiercenia Z1 (abs/ink) 0 = kompatybilność, z programowania DP/DPR 1 = przyrostowo 2 = absolutnie DZIESIĄTKI: Czas oczekiwania DT na końcowej głębokości wiercenia w sekundach/obrotach 0 = kompatybilność, ze znaku DTB (> 0 sekund albo < 0 obrotów) 1 = w sekundach 2 = w obrotach

18.1.3 Wiercenie, pogłębianie czółowe - CYCLE82

Programowanie

CYCLE82 (REAL RTP, REAL RFP, REAL SDIS, REAL DP, REAL DPR, REAL DTB,
INT _GMODE, INT _DMODE, INT _AMODE)

Parametry

Nr	Okno param etrów	Parametry wewnętrzne	Objaśnienie
1	RP	RTP	Płaszczyzna wycofania (abs)
2	Z0	RFP	Punkt odniesienia (abs)
3	SC	SDIS	Odstęp bezpieczeństwa (addytywnie do punktu odniesienia, wprowadzić bez znaku)
4	Z1	DP	Głębokość wiercenia (abs), patrz _AMODE
5	Z1	DPR	Głębokość wiercenia (ink), patrz _AMODE
6	DT	DTB	Czas oczekiwania na końcowej głębokości wiercenia, patrz _AMODE
7		_GMODE	Tryb geometrii (oszacowanie zaprogramowanych wartości geometrycznych) JEDNOSTKI: Zarezerwowano DZIESIĄTKI: Głębokość wiercenia odniesiona do wierzchołka/trzonu 0 = kompatybilność, wierzchołek 1 = trzon
8		_DMODE	Tryb wyświetlania JEDNOSTKI: Płaszczyzna obróbki G17/G18/G19 0 = kompatybilność, aktywna pozostaje płaszczyzna działająca przed wywołaniem cyklu 1 = G17 (aktywna tylko w cyklu) 2 = G18 (aktywna tylko w cyklu) 3 = G19 (aktywna tylko w cyklu)
9		_AMODE	Tryb alternatywny JEDNOSTKI: Głębokość wiercenia Z1 (abs./przyr.) 0 = kompatybilność, z programowania DP/DPR 1 = przyrostowo 2 = absolutnie DZIESIĄTKI: Czas oczekiwania DT na końcowej głębokości wiercenia w sekundach/obrotach 0 = kompatybilność, ze znaku DT (> 0 sekund/ < 0 obrotów) 1 = w sekundach 2 = w obrotach

18.1.4 Rozwiercanie - CYCLE85

Programowanie

CYCLE85 (REAL RTP, REAL RFP, REAL SDIS, REAL DP, REAL DPR, REAL DTB, REAL FFR, REAL RFF, INT _GMODE, INT _DMODE, INT _AMODE)

Parametry

Nr	Okno param etrów	Parametry wewnętrzne	Objaśnienie
1	RP	RTP	Płaszczyzna wycofania (abs)
2	Z0	RFP	Punkt odniesienia (abs)
3	SC	SDIS	Odstęp bezpieczeństwa (addytywnie do punktu odniesienia, wprowadzić bez znaku)
4	Z1	DP	Głębokość wiercenia (abs), patrz _AMODE
5	Z1	DPR	Głębokość wiercenia (ink), patrz _AMODE
6	DT	DTB	Czas oczekiwania na końcowej głębokości wiercenia, patrz _AMODE
7	F	FFR	Posuw
8	FR	RFF	Posuw przy wycofaniu
9		_GMODE	Zarezerwowano
10		_DMODE	Tryb wyświetlania JEDNOSTKI: Płaszczyzna obróbki G17/G18/G19 0 = kompatybilność, aktywna pozostaje płaszczyna działająca przed wywołaniem cyklu 1 = G17 (aktywna tylko w cyklu) 2 = G18 (aktywna tylko w cyklu) 3 = G19 (aktywna tylko w cyklu)
11		_AMODE	Tryb alternatywny (wiercenie) JEDNOSTKI: Głębokość wiercenia Z1 (abs/ink) 0 = kompatybilność, z programowania DP/DPR 1 = przyrostowo 2 = absolutnie DZIESIĄTKI: Czas oczekiwania DT na końcowej głębokości wiercenia w sekundach/obrotach 0 = kompatybilność, ze znaku DT (> 0 sekund albo < 0 obrotów) 1 = w sekundach 2 = w obrotach

18.1.5 Wiercenie głębokich otworów - CYCLE83

Programowanie

```
CYCLE83 (REAL RTP, REAL RFP, REAL SDIS, REAL DP, REAL DPR, REAL FDEP,
REAL FDPR, REAL _DAM, REAL DTB, REAL DTS, REAL FRF, INT VARI, INT
_AXN, REAL _MDEP, REAL _VRT, REAL _DTD, REAL _DIS1, INT _GMODE, INT
_DMODE, INT _AMODE)
```

Parametry

Nr	Okno param etrów	Parametry wewnętrzne	Objaśnienie
1	RP	RTP	Płaszczyzna wycofania (abs)
2	Z0	RFP	Punkt odniesienia (abs)
3	SC	SDIS	Odstęp bezpieczeństwa (addytywnie do punktu odniesienia, wprowadzić bez znaku)
4	Z1	DP	Końcowa głębokość wiercenia (abs.), patrz _AMODE
5	Z1	DPR	Końcowa głębokość wiercenia (przr.), patrz _AMODE
6	D	FDEP	1. Głębokość wiercenia (abs), patrz _AMODE
7	D	FDPR	1. Głębokość wiercenia (ink), patrz _AMODE
8	DF	_DAM	Wartość bezwzględna/procentowa dla każdego kolejnego dosuwu (wartość degresywna/procentowa), patrz _AMODE
9	DTB	DTB	Czas oczekiwania na głębokości wiercenia, patrz _AMODE
10	DTS	DTS	Czas oczekiwania w punkcie początkowym (tylko przy usuwaniu wiórów), patrz _AMODE
11	FD1	FRF	Wielkość procentowa posuwu dla pierwszego dosuwu, patrz _AMODE
12		VARI	Rodzaj obróbki JEDNOSTKI: Łamanie wiórów/usuwanie wiórów 0 = łamanie wiórów 1 = usuwanie wiórów
13		_AXN	Oś narzędzia: 0 = 3. Oś geometryczna 1 = 1. Oś geometryczna 2 = 2. Oś geometryczna > 2 = 3. Oś geometryczna
14	V1	_MDEP	Dosuw minimalny (tylko przy wielkości procentowej dla degresji)
15	V2	_VRT	Wartość wycofania po każdym kroku obróbki (tylko przy łamaniu wiórów) > 0 = zmienna wielkość wycofania 0 = wartość standardowa 1 mm
16	DT	_DTD	Czas oczekiwania na końcowej głębokości wiercenia, patrz _AMODE
17	V3	_DIS1	Odstęp wcześniejszego zatrzymania (tylko przy usuwaniu wiórów), patrz _AMODE

Nr	Okno parametrów	Parametry wewnętrzne	Objaśnienie
18		_GMODE	<p>Tryb geometrii (oszacowanie zaprogramowanych wartości geometrycznych)</p> <p>JEDNOSTKI: Zarezerwowano</p> <p>DZIESIĄTKI: Głębokość wiercenia odniesiona do wierzchołka/trzonu</p> <p>0 = wierzchołek</p> <p>1 = trzon</p>
19		_DMODE	<p>Tryb wyświetlania</p> <p>JEDNOSTKI: Płaszczyzna obróbki G17/G18/G19</p> <p>0 = kompatybilność, aktywna pozostaje płaszczyzna działająca przed wywołaniem cyklu</p> <p>1 = G17 (aktywna tylko w cyklu)</p> <p>2 = G18 (aktywna tylko w cyklu)</p> <p>3 = G19 (aktywna tylko w cyklu)</p>
20		_AMODE	<p>Tryb alternatywny</p> <p>JEDNOSTKI: Głębokość wiercenia = końcowa głębokość wiercenia Z1 (abs/ink)</p> <p>0 = kompatybilność, z programowania DP/DPR</p> <p>1 = przyrostowo</p> <p>2 = absolutnie</p> <p>DZIESIĄTKI: Czas oczekiwania na głębokości wiercenia DTB w sekundach/obrotach</p> <p>0 = kompatybilność ze znaku DTB (> 0 sekund lub < 0 obrotów)</p> <p>1 = w sekundach</p> <p>2 = w obrotach</p> <p>SETKI: Czas oczekiwania w punkcie początkowym DTS w sekundach/obrotach</p> <p>0 = kompatybilność ze znaku DTS (> 0 sekund albo < 0 obrotów)</p> <p>1 = w sekundach</p> <p>2 = w obrotach</p> <p>TYSIĄCE: Czas oczekiwania DT na końcowej głębokości wiercenia w sekundach/obrotach</p> <p>0 = kompatybilność ze znaku _DTD (> 0 sekund albo < 0 obrotów)</p> <p>1 = w sekundach</p> <p>2 = w obrotach</p> <p>DZIESIĄTKI TYSIĘCY: 1. Głębokość wiercenia D (abs/ink)</p> <p>0 = kompatybilność, z programowania FDEP/FDPR</p> <p>1 = przyrostowo</p> <p>2 = absolutnie</p> <p>SETKI TYSIĘCY: Wartość bezwzględna/procentowa DAM dla każdego kolejnego dosuwu (degresja)</p> <p>0 = kompatybilność, ze znaku DAM (> 0 wartość bezwzględna lub < 0 współczynnik 0.001 do 1.0)</p> <p>1 = wartość bezwzględna</p> <p>2 = wartość procentowa (0.001 do 100 %)</p> <p>MILIONY: Odstęp wcześniejszego zatrzymania V3 automatycznie/ręcznie</p>

Nr	Okno param etrów	Parametry wewnętrzne	Objaśnienie
			0 = kompatybilność ze znaku _DIS1 (= 0 automatycznie lub > 0 ręcznie) 1 = automatycznie (jest obliczane w cyklu) 2 = ręcznie (zaprogramowana wartość)
			DZIESIĄTKI MILIONÓW: Współczynnik posuwu dla pierwszego dosuwu FRF, jako współczynnik/wartość procentowa
			0 = kompatybilność, jako współczynnik (0.001 do 1.0, FRF = 0 oznacza 100%) 1 = wartość procentowa (0.001 do 999,999 %)

18.1.6 Wytaczanie - CYCLE86

Programowanie

CYCLE86 (REAL RTP, REAL RFP, REAL SDIS, REAL DP, REAL DPR, REAL DTB, INT SDIR, REAL RPA, REAL RPO, REAL RPAP, REAL POSS, INT _GMODE, INT _DMODE, INT _AMODE)

Parametry

Nr	Okno param etrów	Parametry wewnętrzne	Objaśnienie
1	RP	RTP	Płaszczyzna wycofania (abs)
2	Z0	RFP	Punkt odniesienia (abs)
3	SC	SDIS	Odstęp bezpieczeństwa (addytywnie do punktu odniesienia, wprowadzić bez znaku)
4	Z1	DP	Głębokość wiercenia (abs), patrz _AMODE
5	Z1	DPR	Głębokość wiercenia (ink), patrz _AMODE
6	DT	DTB	Czas oczekiwania na końcowej głębokości wiercenia, patrz _AMODE
7	DIR	SDIR	Kierunek obrotów wrzeciona 3 = M3 4 = M4
8	DX	RPA	Wartość wycofania w kierunku X
9	DY	RPO	Wartość wycofania w kierunku Y
10	DZ	RPAP	Wartość wycofania w kierunku Z
11	SPOS	POSS	Pozycja wrzeciona dla wycofania (zorientowan zatrzymanie wrzeciona, w stopniach)
12		_GMODE	Tryb geometrii JEDNOSTKI: Tryb wycofania 0 = wycofanie, kompatybilność 1 = bez wycofania

Nr	Okno parametrów	Parametry wewnętrzne	Objaśnienie
13		_DMODE	Tryb wyświetlania JEDNOSTKI: Płaszczyzna obróbki G17/G18/G19 0 = kompatybilność, aktywna pozostaje płaszczyzna działająca przed wywołaniem cyklu 1 = G17 (aktywna tylko w cyklu) 2 = G18 (aktywna tylko w cyklu) 3 = G19 (aktywna tylko w cyklu)
14		_AMODE	Tryb alternatywny JEDNOSTKI: Głębokość wiercenia Z1 (abs/ink) 0 = kompatybilność, z programowania DP/DPR 1 = przyrostowo 2 = absolutnie DZIESIĄTKI: Czas oczekiwania DT na końcowej głębokości wiercenia w sekundach/obrotach 0 = kompatybilność, ze znaku DT (> 0 sekund albo < 0 obrotów) 1 = w sekundach 2 = w obrotach

18.1.7 Gwintowanie otworu bez oprawki kompensacyjnej - CYCLE84

Programowanie

```
CYCLE84 (REAL RTP, REAL RFP, REAL SDIS, REAL DP, REAL DPR, REAL DTB,
INT SDAC, REAL MPIT, REAL PIT, REAL POSS, REAL SST, REAL SST1, INT
_AXN, INT _PITA, INT _TECHNO, INT _VARI, REAL _DAM, REAL _VRT,
STRING[15] _PITM, STRING[5] _PTAB, STRING[20] _PTABA, INT _GMODE,
INT _DMODE, INT _AMODE)
```

Parametry

Nr	Okno parametrów	Parametry wewnętrzne	Objaśnienie
1	RP	RTP	Płaszczyzna wycofania (abs)
2	Z0	RFP	Punkt odniesienia (abs)
3	SC	SDIS	Odstęp bezpieczeństwa (addytywnie do punktu odniesienia, wprowadzić bez znaku)
4	Z1	DP	Głębokość wiercenia = końcowa głębokość gwintowania (abs.), patrz _AMODE
5	Z1	DPR	Głębokość gwintowania = końcowa głębokość gwintowania (przr.), patrz _AMODE
6	DT	DTB	Czas oczekiwania na głębokości gwintowania w sekundach
7	SDE	SDAC	Kierunek obrotów po zakończeniu cyklu

Nr	Okno parametrów	Parametry wewnętrzne	Objaśnienie
8		MPIT	Wielkość gwintu tylko dla "ISO metrycznego zwykłego" (skok jest obliczany wewnętrznie podczas wykonywania)
9	P	PIT	Skok gwintu jako wartość, jednostka miary patrz _PITA
10	α_S^1	POSS	Pozycja wrzeciona dla zorientowanego zatrzymania wrzeciona
11	S	SST	Prędkość obrotowa wrzeciona dla gwintowania otworu
12	SR	SST1	Prędkość obrotowa wrzeciona dla wycofania
13		_AXN	Oś gwintowania: 0 = 3. Oś geometryczna 1 = 1. Oś geometryczna 2 = 2. Oś geometryczna ≥ 3 = 3. Oś geometryczna
14		_PITA	Jednostka miary skoku gwintu (oszacowanie PIT i MPIT) 0 = skok w mm - Oszacowanie MPIT/PIT 1 = skok w mm - Oszacowanie PIT 2 = skok w TPI - Oszacowanie PIT (zwojów gwintu na cal) 3 = skok w calach - Oszacowanie PIT 4 = MODUŁ - Oszacowanie PIT
15		_TECHNO	Technologia ¹⁾ JEDNOSTKI: Zatrzymanie dokładne 0 = zatrzymanie dokładne aktywne, jak przed wywołaniem cyklu 1 = zatrzymanie dokładne G601 2 = zatrzymanie dokładne G602 3 = zatrzymanie dokładne G603 DZIESIĄTKI: Sterowanie wyprzedzające 0 = z/bez sterowania wyprzedzającego aktywne, jak przed wywołaniem cyklu 1 = ze sterowaniem wyprzedzającym FFWON 2 = bez sterowania wyprzedzającego FFWOF SETKI: Przyśpieszenie 0 = SOFT/BRISK/DRIVE aktywne, jak przed wywołaniem cyklu 1 = z ograniczeniem przyśpieszenia drugiego stopnia SOFT 2 = bez ograniczenia przyśpieszenia drugiego stopnia BRISK 3 = zmniejszone przyśpieszenie DRIVE TYSIĄCE: MCALL praca jako wrzeciono 0 = przy MCALL ponownie uaktywnić pracę jako wrzeciono 1 = przy MCALL pozostać w regulacji położenia
16		_VARI	Rodzaj obróbki: JEDNOSTKI: 0 = 1 krok 1 = łamanie wiórów (wiercenie otworów głębokich) 2 = usuwanie wiórów (wiercenie otworów głębokich)

Nr	Okno param etrów	Parametry wewnętrzne	Objaśnienie
			TYSIĄCE: ISO/SIEMENS tryb bez znaczenia dla okna wprowadzania 1 = wywołanie z kompatybilności ISO 0 = wywołanie z kontekstu SIEMENS
17	D	_DAM	Maksymalny dosuw na głębokość (tylko przy usuwaniu/łamaniu wiórów)
18	V2	_VRT	Wartość wycofania po każdym dosuwie (tylko przy łamaniu wiórów), patrz _AMODE
19		_PITM	Łańcuch znaków jako znacznik dla wprowadzenia skoku gwintu ²⁾
20		_PTAB	Łańcuch znaków dla tablicy gwintów ("", "ISO", "BSW", "BSP", "UNC") ²⁾
21		_PTABA	Łańcuch znaków dla wyboru w tablicy gwintów (np. "M 10", "M 12", ...) ²⁾
22		_GMODE	Tryb geometrii (oszacowanie zaprogramowanych wartości geometrycznych) JEDNOSTKI: Zarezerwowano DZIESIĄTKI: Zarezerwowano
23		_DMODE	Tryb wyświetlania JEDNOSTKI: Płaszczyzna obróbki G17/G18/G19 0 = kompatybilność, aktywna pozostaje płaszczyna działająca przed wywołaniem cyklu 1 = G17 (aktywna tylko w cyklu) 2 = G18 (aktywna tylko w cyklu) 3 = G19 (aktywna tylko w cyklu) DZIESIĄTKI: Zarezerwowano SETKI: Zarezerwowano TYSIĄCE: Tryb zgodności (tylko dla okna wprowadzania do przetwarzania wstecznego), gdy MD 52216 Bit0 = 1 ¹⁾ 0 = Parametry technologii zostaną wyświetlone (kompatybilność): Parametry TECHNOLOGII działają 1 = Parametry technologii nie zostaną wyświetlone: Technologia "jak zaprogramowano przed wywołaniem cyklu" działa
24		_AMODE	Tryb alternatywny JEDNOSTKI: Głębokość gwintowania = końcowa głębokość gwintowania Z1 (abs/ink) 0 = kompatybilność, z programowania DP/DPR 1 = przyrostowo 2 = absolutnie DZIESIĄTKI: Zarezerwowano SETKI: Zarezerwowano TYSIĄCE: Kierunek gwintu w prawo/w lewo 0 = kompatybilność, ze znaku PIT/MPIT 1 = w prawo 2 = w lewo DZIESIĄTKI TYSIĘCY: Zarezerwowano SETKI TYSIĘCY: Zarezerwowano MILIONY: Wartość wycofania po każdym dosuwie V2 ręcznie/automatycznie

Nr	Okno param etrów	Parametry wewnętrzne	Objaśnienie
			0 = kompatybilność, z programowania _VRT (> 0 wartość zmienna lub ≤ 0 wartość standardowa 1 mm/0.0394 cala) 1 = automatycznie (wartość standardowa 1mm/0.0394 cala) 2 = ręcznie (jak zaprogramowano pod V2)

¹⁾ Tablice technologii mogą być ukryte w zależności od danej ustawczej SD52216 \$MCS_FUNCTION_MASK_DRILL

²⁾ Parametry 19, 20 i 21 są stosowane tylko przy wyborze gwintu w tablicach gwintów okna wprowadzania. Dostęp do tablic gwintów poprzez definicję cyklu w czasie wykonywania cyklu jest niemożliwy.

18.1.8 Gwintowanie otworu z oprawką kompensacyjną - CYCLE840

Programowanie

```
CYCLE840 (REAL RTP, REAL RFP, REAL SDIS, REAL DP, REAL DPR, REAL DTB,
INT SDR, INT SDAC, INT ENC, REAL MPIT, REAL PIT, INT _AXN, INT
_PITA, INT _TECHNO, STRING[15] _PITM, STRING[5] _PTAB, STRING[20]
_PTAB, INT _GMODE, INT _DMODE, INT _AMODE)
```

Parametry

Nr	Okno param etrów	Parametry wewnętrzne	Objaśnienie
1	RP	RTP	Płaszczyzna wycofania (abs)
2	Z0	RFP	Punkt odniesienia (abs)
3	SC	SDIS	Odstęp bezpieczeństwa (addytywnie do punktu odniesienia, wprowadzić bez znaku)
4	Z1	DP	Głębokość wiercenia (abs), patrz _AMODE
5	Z1	DPR	Głębokość wiercenia (ink), patrz _AMODE
6	DT	DTB	Czas oczekiwania na głębokości gwintowania/na odstęp bezpieczeństwa po wycofaniu, w sekundach, patrz ENC
7		SDR	Kierunek obrotów dla wycofania
8	SDE	SDAC	Kierunek obrotów po zakończeniu cyklu

Nr	Okno parametrów	Parametry wewnętrzne	Objaśnienie
9		ENC	Gwintowanie otworu z przetwornikiem wrzeciona (G33)/gwintowanie otworu bez przetwornika wrzeciona (G63) 0 = z przetwornikiem wrzeciona 20 = z przetwornikiem wrzeciona 11 = bez przetwornika wrzeciona 1 = bez przetwornika wrzeciona - Skok z MPIT/PIT - bez DT - Skok z MPIT/PIT - z DT po wycofaniu na odstęp bezpieczeństwa - Skok z MPIT/PIT - z DT na głębokość gwintowania - Skok z zaprogramowanego posuwu - z DT na głębokość gwintowania (posuw = prędkość obrotowa skok)
10		MPIT	Wielkość gwintu tylko dla "ISO metrycznego zwykłego" (skok jest obliczany wewnętrznie podczas wykonywania) Zakres wartości: 3 do 48 (dla M3 do M48), alternatywnie do PIT
11		PIT	Skok gwintu jako wartość, jednostka miary patrz _PITA) Zakres wartości: > 0, alternatywnie do MPIT
12		_AXN	Oś gwintowania: 0 = 3. Oś geometryczna 1 = 1. Oś geometryczna 2 = 2. Oś geometryczna ≥ 3 = 3. Oś geometryczna
13		_PITA	Jednostka miary skoku gwintu (oszacowanie PIT i MPIT) 0 = skok w mm - Oszacowanie MPIT/PIT 1 = skok w mm - Oszacowanie PIT 2 = skok w TPI - Oszacowanie PIT (zwojów gwintu na cal) 3 = skok w calach - Oszacowanie PIT 4 = MODUŁ - Oszacowanie PIT
14		_TECHNO	Technologia ¹⁾ JEDNOSTKI: Zatrzymanie dokładne 0 = zatrzymanie dokładne, jak przed wywołaniem cyklu 1 = zatrzymanie dokładne G601 2 = zatrzymanie dokładne G602 3 = zatrzymanie dokładne G603 DZIESIĄTKI: Sterowanie wyprzedzające 0 = z/bez sterowania wyprzedzającego aktywne, jak przed wywołaniem cyklu 1 = ze sterowaniem wyprzedzającym FFWON 2 = bez sterowania wyprzedzającego FFWOF
15		_PITM	Łańcuch znaków jako znacznik dla wprowadzenia skoku gwintu ²⁾
16		_PTAB	Łańcuch znaków dla tablicy gwintów ("", "ISO", "BSW", "BSP", "UNC") ²⁾
17		_PTABA	Łańcuch znaków dla wyboru w tablicy gwintów (np. "M 10", "M 12", ...) ²⁾
18		_GMODE	Zarezerwowano

Nr	Okno parametrów	Parametry wewnętrzne	Objaśnienie
19		<code>_DMODE</code>	<p>Tryb wyświetlania</p> <hr/> <p>JEDNOSTKI: Płaszczyzna obróbki G17/G18/G19</p> <p>0 = kompatybilność, aktywna pozostaje płaszczyna działająca przed wywołaniem cyklu 1 = G17 (aktywna tylko w cyklu) 2 = G18 (aktywna tylko w cyklu) 3 = G19 (aktywna tylko w cyklu)</p> <hr/> <p>DZIESIĄTKI: Zarezerwowano</p> <hr/> <p>SETKI: Zarezerwowano</p> <hr/> <p>TYSIĄCE: Tryb zgodności (tylko dla okna wprowadzania do przetwarzania wstecznego), gdy MD 52216 Bit0 = 1¹⁾</p> <p>0 = Parametry technologii zostaną wyświetlone (kompatybilność): Parametry TECHNOLOGII działają 1 = Parametry technologii nie zostaną wyświetlone: Technologia "jak zaprogramowano przed wywołaniem cyklu" działa</p>
20		<code>_AMODE</code>	<p>Tryb alternatywny</p> <hr/> <p>JEDNOSTKI: Głębokość wiercenia Z1 (abs/ink)</p> <p>0 = kompatybilność, z programowania DP/DPR 1 = przyrostowo 2 = absolutnie</p>

¹⁾ Tablice technologii mogą być ukryte w zależności od danej ustawczej SD52216 \$MCS_FUNCTION_MASK_DRILL

²⁾ Parametry 15, 16 i 17 są stosowane tylko przy wyborze gwintu w tablicach gwintów okna wprowadzania. Dostęp do tablic gwintów poprzez definicję cyklu w czasie wykonywania cyklu jest niemożliwy!

18.1.9 Wiercenie otworu z frezowaniem gwintu - CYCLE78

Programowanie

```
CYCLE78 (REAL _RTP, REAL _RFP, REAL _SDIS, REAL _DP, REAL _ADPR, REAL
_FDFR, REAL _LDPR, REAL _DIAM, REAL _PIT, INT _PITA, REAL _DAM, REAL
_MDEP, INT _VARI, INT _CDIR, REAL _GE, REAL _FFD, REAL _FRDP, REAL
_FFR, REAL _FFP2, INT _FFA, STRING[15] _PITM, STRING[20] _PTAB,
STRING[20] _PTABA, INT _GMODE, INT _DMODE, INT _AMODE)
```

Parametry

Nr	Okno parametrów	Parametry wewnętrzne	Objaśnienie
1	RP	<code>_RTP</code>	Płaszczyzna wycofania (abs)
2	Z0	<code>_RFP</code>	Punkt odniesienia osi narzędzia (abs)

Nr	Okno parametrów	Parametry wewnętrzne	Objaśnienie
3	SC	_SDIS	Odstęp bezpieczeństwa (addytywnie do punktu odniesienia, wprowadzić bez znaku)
4	Z1	_DP	Końcowa głębokość wiercenia (abs./przyr.), patrz _AMODE
5		_ADPR	Głębokość nawiercania ze zmniejszonym posuwem po torze (przyr.), działa z VARI ZEHNTAUSEND
6	D	_FDPR	maksymalny dosuw na głębokość (przyr.) $D \geq Z1 \Rightarrow$ dosuw na końcową głębokość wiercenia $D < Z1 \Rightarrow$ cykl głębokiego wiercenia z wieloma dosuwami i usuwaniem wiórów
7	ZR	_LDPR	Pozostała głębokość wiercenia przy wierceniu przelotowym (przyr.), z posuwem FR
8	∅	_DIAM	Średnica nominalna gwintu
9	P	_PIT	Skok gwintu jako wartość liczbowa
10		_PITA	Interpretacja skoku gwintu P 1 = skok w mm/obr 2 = skok w zwojach/cal 3 = skok w calach/obr. 4 = skok jako MODUŁ
11	DF	_DAM	Wartość bezwzględna/procentowa dla każdego kolejnego dosuwu (degresja), patrz _AMODE
12	V1	_MDEP	dosuw minimalny (przyr.), działa tylko przy degresji
13		_VARI	Rodzaj obróbki JEDNOSTKI: Zarezerwowano DZIESIĄTKI: 0 = bez usuwania wiórów przed frezowaniem gwintu (działa tylko na końcową głębokość wiercenia) 1 = usuwanie wiórów przed frezowaniem gwintu (działa tylko na końcową głębokość wiercenia) SETKI: 0 = gwint prawy 1 = gwint lewy TYSIĄCE: 0 = bez pozostałej głębokości wiercenia z posuwem wiercenia FR 1 = pozostała głębokość wiercenia z posuwem wiercenia FR DZIESIĄTKI TYSIĘCY: 0 = bez nawiercania ze zmniejszonym posuwem 1 = nawiercanie ze zmniejszonym posuwem Posuw nawiercania = 0.3 mm/obr, gdy $F1 < 0.15$ mm/obr Posuw nawiercania = 0.1 mm/obr, gdy $F1 \geq 0.15$ mm/obr
14		_CDIR	Kierunek frezowania 0 = frezowanie współbieżne 1 = frezowanie przeciwbieżne 4 = ruch przeciwbieżny + współbieżny (kombinacja obróbki zgrubnej + wykańczającej)

Nr	Okno parametrów	Parametry wewnętrzne	Objaśnienie
15	Z2	_GE	Wielkość wycofania przed frezowaniem gwintu (przr.)
16	F1	_FFD	Posuw wiercenia (mm/min wzgl. cali/min albo mm/obr.)
17	FR	_FRDP	Posuw wiercenia dla pozostałej głębokości wiercenia (mm/min albo mm/obr)
18	F2	-FFR	Posuw dla frezowania gwintu (mm/min albo mm/ostrze)
19	FS	_FFP2	Posuw obróbki wykańczającej dla CDIR=4 (mm/min albo mm/ostrze)
20		_FFA	Interpretacja posuwów JEDNOSTKI: Posuw po torze F1 DZIESIĄTKI: Posuw dla pozostałej głębokości wiercenia FR SETKI: Posuw dla frezowania gwintu F2 TYSIĄCE: Posuw obróbki wykańczającej FS
21		_PITM	Łańcuch znaków jako znacznik dla wprowadzenia skoku gwintu (tylko dla powierzchni) ¹⁾
22		_PTAB	Łańcuch znaków dla tablicy gwintów ("", "ISO", "BSW", "BSP", "UNC") (tylko dla powierzchni) ¹⁾
23		_PTABA	Łańcuch znaków dla wyboru w tablicy gwintów (np. "M 10", "M 12", ...) (tylko dla powierzchni) ¹⁾
24		_GMODE	Tryb geometrii, zarezerwowany
25		_DMODE	Tryb wyświetlania JEDNOSTKI: Płaszczyzna obróbki G17/18/19 0 = kompatybilność, aktywna pozostaje płaszczyzna działająca przed wywołaniem cyklu 1 = G17 (aktywna tylko w cyklu) 2 = G18 (aktywna tylko w cyklu) 3 = G19 (aktywna tylko w cyklu)
26		_AMODE	Tryb alternatywny JEDNOSTKI: Głębokość gwintowania = końcowa głębokość gwintowania Z1 abs./przr. 0 = absolutnie 1 = przyrostowo DZIESIĄTKI: Wartość bezwzględna/procentowa DF dla każdego kolejnego dosuwu (degresja) 0 = wartość bezwzględna 1 = wartość procentowa (0.001 do 100 %)

Wskazówka

¹⁾ Parametry 21, 22 i 23 są stosowane tylko przy wyborze gwintu w tablicach gwintów okna wprowadzania. Dostęp do tablic gwintów poprzez definicję cyklu w czasie wykonywania cyklu jest niemożliwy.

18.1.10 Dowlone pozycje - CYCLE802

Programowanie

```
CYCLE802 (INT _XA, INT _YA, REAL _X0, REAL _Y0, REAL _X1, REAL _Y1,
REAL _X2, REAL _Y2, REAL _X3, REAL _Y3, REAL _X4, REAL _Y4, REAL
_X5, REAL _Y5, REAL _X6, REAL _Y6, REAL _X7, REAL _Y7, REAL _X8,
REAL _Y8, INT _VARI, INT _UMODE, INT _DMODE)
```

Parametry

Nr	Okno param etrów	Param wewnętrzne	Objaśnienie
1		_XA	Alternatywa dla wszystkich pozycji X (9-cyfrowa wartość dziesiętna) Liczba miejsc: 876543210 (Miejsce odpowiada pozycji gwintowania Xn) Liczba miejsc: 1 = absolutnie (1. programowana pozycja zawsze absolutnie!) 2 = przyrostowo
2		_YA	Alternatywa dla wszystkich pozycji Y (9-cyfrowa wartość dziesiętna) Liczba miejsc: 876543210 (Miejsce odpowiada pozycji gwintowania Yn) Liczba miejsc: 1 = pozycja wprowadzona (abs.) 2 = pozycja wprowadzona (przyr.)
3	X0	_X0	1. Pozycja X
4	Y0	_Y0	1. Pozycja Y
5	X1	_X1	2. Pozycja X
6	Y1	_Y1	2. Pozycja Y
7	X2	_X2	3. Pozycja X
8	Y2	_Y2	3. Pozycja Y
9	X3	_X3	4. Pozycja X
10	Y3	_Y3	4. Pozycja Y
11	X4	_X4	5. Pozycja X
12	Y4	_Y4	5. Pozycja Y
13	X5	_X5	6. Pozycja X
14	Y5	_Y5	6. Pozycja Y
15	X6	_X6	7. Pozycja X
16	Y6	_Y6	7. Pozycja Y
17	X7	_X7	8. Pozycja X
18	Y7	_Y7	8. Pozycja Y
19	X8	_X8	9. Pozycja X
20	Y8	_Y8	9. Pozycja Y
21		_VARI	Zarezerwowano

Nr	Okno parametrów	Param wewnętrzne	Objaśnienie
22		_UMODE	Zarezerwowano
23		_DMODE	Tryb wyświetlania
JEDNOSTKI: Płaszczyzna obróbki G17/18/19			
0 = kompatybilność, aktywna pozostaje płaszczyna działająca przed wywołaniem cyklu			
1 = G17 (aktywna tylko w cyklu)			
2 = G18 (aktywna tylko w cyklu)			
3 = G19 (aktywna tylko w cyklu)			

Wskazówka

Niepotrzebne pozycje parametrów X1/Y1 do X8/Y8 można pominąć.

Wartości alternatywne dla _XA, i _YA należy jednak podać kompletnie dla wszystkich 9 pozycji.

18.1.11 Szereg otworów - HOLES1**Programowanie**

```
HOLES1 (REAL SPCA, REAL SPCO, REAL STA1, REAL FDIS, REAL DBH, INT
NUM, INT _VARI, INT _UMODE, STRING[200] _HIDE, INT _NSP, INT _DMODE)
```

Parametry

Nr	Okno parametrów	Param wewnętrzne	Objaśnienie
1	X0	SPCA	Punkt odniesienia dla szeregu otworów w 1. osi (abs.)
2	Y0	SPCO	Punkt odniesienia dla szeregu otworów w 2. osi (abs.)
3	$\alpha 0$	STA1	Podstawowy kąt obrotu (kąt w stosunku do 1. osi)
4	L0	FDIS	Odstęp 1. otworu od punktu odniesienia
5	L	DBH	Odstęp między otworami
6	N	NUM	Liczba otworów
7		_VARI	Zarezerwowano
8		_UMODE	Zarezerwowano
9		_HIDE	Ukryte pozycje <ul style="list-style-type: none"> • max 198 znaków • Podanie kolejnych numerów pozycji, np. "1,3" (pozycje 1 i 3 nie będą wykonywane)

Nr	Okno parametrów	Param wewnętrzne	Objaśnienie
10		_NSP	Zarezerwowano
11		_DMODE	Tryb wyświetlania JEDNOSTKI: Płaszczyzna obróbki G17/18/19 0 = kompatybilność, aktywna pozostaje płaszczyzna działająca przed wywołaniem cyklu 1 = G17 (aktywna tylko w cyklu) 2 = G18 (aktywna tylko w cyklu) 3 = G19 (aktywna tylko w cyklu)

18.1.12 Siatka albo ramka - CYCLE801

Programowanie

```
CYCLE801 (REAL _SPCA, REAL _SPCO, REAL _STA, REAL _DIS1, REAL _DIS2,
INT _NUM1, INT _NUM2, INT _VARI, INT _UMODE, REAL _ANG1, REAL _ANG2,
STRING[200] _HIDE, INT _NSP, INT _DMODE)
```

Parametry

Nr	Okno parametrów	Param wewnętrzne	Objaśnienie
1	X0	_SPCA	Punkt odniesienia dla szablonu pozycji (siatka/ramka) w 1. osi (abs.)
2	Y0	_SPCO	Punkt odniesienia dla szablonu pozycji (siatka/ramka) w 2. osi (abs.)
3	α 0	_STA	Podstawowy kąt obrotu (kąt w stosunku do 1. osi) < 0 = obrót w kierunku ruchu wskazówek zegara > 0 = obrót przeciwnie do kierunku ruchu wskazówek zegara
4	L1	_DIS1	Odstęp kolumn (odstęp pozycji 1. osi, wprowadzić bez znaku liczby)
5	L2	_DIS2	Odstęp wierszy (odstęp pozycji 2. osi, wprowadzić bez znaku liczby)
6	N1	_NUM1	Liczba kolumn
7	N2	_NUM2	Liczba wierszy
8		_VARI	Rodzaj obróbki JEDNOSTKI: Szablon pozycji 0 = siatka 1 = ramka DZIESIĄTKI: Zarezerwowano SETKI: Zarezerwowano
9		_UMODE	Zarezerwowano

Nr	Okno parametrów	Param wewnętrzne	Objaśnienie
10	αX	_ANG1	Kąt ścięcia w stosunku do 1. osi (położenie skośne wierszy w odniesieniu do 1. osi) < 0 = pomiar w kierunku ruchu wskazówek zegara (0 do -90 stopni) > 0 = pomiar przeciwnie do kierunku ruchu wskazówek zegara (0 do 90 stopni)
11	αY	_ANG2	Kąt ścięcia w stosunku do 2. osi (położenie skośne kolumn w odniesieniu do 2. osi) < 0 = pomiar w kierunku ruchu wskazówek zegara (0 do -90 stopni) > 0 = pomiar przeciwnie do kierunku ruchu wskazówek zegara (0 do 90 stopni)
12		_HIDE	Ukryte pozycje <ul style="list-style-type: none"> • max 198 znaków • Podanie kolejnych numerów pozycji, np. "1,3" (pozycje 1 i 3 nie będą wykonywane)
13		_NSP	Zarezerwowano
14		_DMODE	Tryb wyświetlania JEDNOSTKI: Płaszczyzna obróbki G17/18/19 0 = kompatybilność, aktywna pozostaje płaszczyzna działająca przed wywołaniem cyklu 1 = G17 (aktywna tylko w cyklu) 2 = G18 (aktywna tylko w cyklu) 3 = G19 (aktywna tylko w cyklu)

18.1.13 Okrąg otworów - HOLES2

Programowanie

```
HOLES2 (REAL CPA, REAL CPO, REAL RAD, REAL STA1, REAL INDA, INT NUM,
INT _VARI, INT _UMODE, STRING[200] _HIDE, INT _NSP, INT _DMODE)
```

Parametry

Nr	Okno parametrów	Param wewnętrzne	Objaśnienie
1	X0	CPA	Punkt środkowy dla okręgu otworów w 1. osi (abs.)
2	Y0	CPO	Punkt środkowy dla okręgu otworów w 2. osi (abs.)
3	R	RAD	Promień okręgu otworów
4	$\alpha 0$	STA1	Kąt początkowy
5	$\alpha 1$	INDA	Kąt przełączania (tylko w przypadku łuku koła) < 0 = kierunek ruchu wskazówek zegara > 0 = przeciwnie do ruchu wskazówek zegara
6	N	NUM	Liczba pozycji

Nr	Okno param etrów	Param wewnętrzne	Objaśnienie
7		_VARI	Rodzaj obróbki <hr/> JEDNOSTKI: Zarezerwowano <hr/> DZIESIĄTKI: Rodzaj pozycjonowania <hr/> 0 = ruch do pozycji - liniowy 1 = ruch do pozycji - po torze kołowym <hr/> SETKI: : Zarezerwowano <hr/> TYSIĄCE: Szablon okręgu <hr/> 0 = tryb kompatybilności, gdy INDA = 0 wówczas pełny okrąg, INDA <> 0 wówczas łuk koła) 1 = okrąg 2 = łuk koła
8		_UMODE	Zarezerwowano
9		_HIDE	Ukryte pozycje <ul style="list-style-type: none"> • max 198 znaków • Podanie kolejnych numerów pozycji, np. "1,3" (pozycje 1 i 3 nie będą wykonywane)
10		_NSP	Zarezerwowano
11		_DMODE	Tryb wyświetlania <hr/> JEDNOSTKI: Płaszczyzna obróbki G17/18/19 <hr/> 0 = kompatybilność, aktywna pozostaje płaszczyzna działająca przed wywołaniem cyklu 1 = G17 (aktywna tylko w cyklu) 2 = G18 (aktywna tylko w cyklu) 3 = G19 (aktywna tylko w cyklu)

18.1.14 Frezowanie płaszczyzny - CYCLE61

Programowanie

```
CYCLE61 (REAL _RTP, REAL _RFP, REAL _SDIS, REAL _DP, REAL _PA, REAL
_PO, REAL _LENG, REAL _WID, REAL _MID, REAL _MIDA, REAL _FALD, REAL
_FFP1, INT _VARI, INT _LIM, INT _DMODE, INT _AMODE)
```

Parametry

Nr	Okno param etrów	Parametry wewnętrzne	Objaśnienie
1	RP	_RTP	Płaszczyzna wycofania (abs)
2	Z0	_RFP	Punkt odniesienia osi narzędzia, wysokość półfabrykatu (abs)

Nr	Okno parametrów	Parametry wewnętrzne	Objaśnienie
3	SC	_SDIS	Odstęp bezpieczeństwa (addytywnie do punktu odniesienia, wprowadzić bez znaku)
4	Z1	_DP	Wysokość części gotowej (abs./przyr.), patrz _AMODE
5	X0	_PA	Punkt narożny 1 w 1. osi (abs.)
6	Y0	_PO	Punkt narożny 1 w 2. osi (abs.)
7	X1	_LENG	Punkt narożny 2 w 1. osi (abs./przyr.), patrz _AMODE
8	Y1	_WID	Punkt narożny 2 w 2. osi (abs./przyr.), patrz _AMODE
9	DZ	_MID	Maksymalny dosuw na głębokość
10	DXY	_MIDA	Maksymalny dosuw w płaszczyźnie (jednostka, patrz _AMODE)
11	UZ	_FALD	Naddatek na głębokości
12	F	_FFP1	Posuw roboczy
13		_VARI	Rodzaj obróbki JEDNOSTKI: Obróbka 1 = obróbka zgrubna 2 = obróbka wykańczająca DZIESIĄTKI: Kierunek obróbki 1 = równoległe do 1. osi, jeden kierunek 2 = równoległe do 2. osi, jeden kierunek 3 = równoległe do 1. osi, kierunek zmienny 4 = równoległe do 2. osi, kierunek zmienny
14		_LIM	Ograniczenia JEDNOSTKI: Ograniczenie 1. oś minus 0 = nie 1 = tak DZIESIĄTKI: Ograniczenie 1. oś plus 0 = nie 1 = tak SETKI: Ograniczenie 2. oś minus 0 = nie 1 = tak TYSIĄCE: Ograniczenie 2. oś plus 0 = nie 1 = tak
15		_DMODE	Tryb wyświetlania JEDNOSTKI: Płaszczyzna obróbki G17/18/19 0 = kompatybilność, aktywna pozostaje płaszczyzna działająca przed wywołaniem cyklu 1 = G17 (aktywna tylko w cyklu) 2 = G18 (aktywna tylko w cyklu) 3 = G19 (aktywna tylko w cyklu)

Nr	Okno parametrów	Parametry wewnętrzne	Objaśnienie
16		_AMODE	Tryb alternatywny
			JEDNOSTKI: Głębokość końcowa (_DP)
			0 = absolutnie
			1 = przyrostowo
			DZIESIĄTKI: Jednostka dla dosuwu w płaszczyźnie (_MIDA)
			0 = mm
			1 = % średnicy narzędzia
			SETKI: Zarezerwowano
			TYSIĄCE: Długość płaszczyzny
			0 = przyrostowo
			1 = absolutnie
			DZIESIĄTKI TYSIĘCY: Szerokość płaszczyzny
			0 = przyrostowo
			1 = absolutnie

18.1.15 Frezowanie kieszeni prostokątnej - POCKET3

Programowanie

```
POCKET3 (REAL _RTP, REAL _RFP, REAL _SDIS, REAL _DP, REAL _LENG, REAL
_WID, REAL _CRAD, REAL _PA, REAL _PO, REAL _STA, REAL _MID, REAL
_FAL, REAL _FALD, REAL _FFP1, REAL _FFD, INT _CDIR, INT _VARI, REAL
_MIDA, REAL _AP1, REAL _AP2, REAL _AD, REAL _RAD1, REAL _DP1, INT
_UMODE, REAL _FS, REAL _ZFS, INT _GMODE, INT _DMODE, INT _AMODE)
```

Parametry

Nr	Okno parametrów	Parametry wewnętrzne	Objaśnienie
1	RP	_RTP	Płaszczyzna wycofania (abs)
2	Z0	_RFP	Punkt odniesienia osi narzędzia (abs)
3	SC	_SDIS	Odstęp bezpieczeństwa (addytywnie do punktu odniesienia, wprowadzić bez znaku)
4	Z1	_DP	Głębokość kieszeni (abs./przyr.), patrz _AMODE
5	L	_LENG	Długość kieszeni (przyr., wprowadzić ze znakiem)
6	W	_WID	Szerokość kieszeni (przyr., wprowadzić ze znakiem)
7	RP	_CRAD	Promień naroża kieszeni

Nr	Okno parametrów	Parametry wewnętrzne	Objaśnienie
8	X0	_PA	Punkt odniesienia, 1. oś (abs)
9	YO	_PO	Punkt odniesienia, 2. oś (abs)
10	$\alpha 0$	_STA	Kąt obrotu, kąt między osią wzdłużną (L) i 1. osią
11	DZ	_MID	Maksymalny dosuw na głębokość
12	UXY	_FAL	Naddatek w płaszczyźnie
13	UZ	_FALD	Naddatek na głębokości
14	F	_FFP1	Posuw w płaszczyźnie
15	FZ	_FFD	Posuw dla dosuwu na głębokość
16		_CDIR	Kierunek frezowania: 0 = frezowanie współbieżne 1 = frezowanie przeciwbieżne
17		_VARI	Rodzaj obróbki JEDNOSTKI: 1 = obróbka zgrubna 2 = obróbka wykańczająca 4 = obróbka wykańczająca obrzeża 5 = fazowanie DZIESIĄTKI: 0 = z nawiercaniem, dosuw z G0 1 = prostopadle, dosuw z G1 2 = spiralnie 3 = ruch wahadłowy w osi wzdłużnej kieszeni SETKI: Zarezerwowano
18	DXY	_MIDA	Maksymalny dosuw w płaszczyźnie, jednostka patrz _AMODE
19	L1	_AP1	Długość kieszeni półfabrykatu (przyr.)
20	W1	_AP2	Szerokość kieszeni półfabrykatu (przyr.)
21	AZ	_AD	Głębokość kieszeni półfabrykatu (przyr.)
22	ER	_RAD1	Promień toru spiralnego przy zagłębieniu
	EW		Maksymalny kąt zagłębienia dla ruchu wahliwego
23	EP	_DP1	Skok przy zagłębieniu po linii spiralnej
24		_UMODE	Zarezerwowano
25	FS	_FS	Szerokość fazy (przyr.)
26	ZFS	_ZFS	Głębokość zagłębienia (wierzchołka narzędzia) przy fazowaniu (abs./przyr.), patrz _AMODE

Nr	Okno parametrów	Parametry wewnętrzne	Objaśnienie
27		_GMODE	<p>Tryb geometrii</p> <p>JEDNOSTKI: Zarezerwowano</p> <p>DZIESIĄTKI: Zarezerwowano</p> <p>SETKI: Wybór obróbka/tylko obliczenie punktu startowego</p> <p>0 = tryb kompatybilności</p> <p>1 = normalna obróbka</p> <p>TYSIĄCE: Zwymiarowanie przez środek/naróże</p> <p>0 = tryb kompatybilności</p> <p>1 = zwymiarowanie przez środek</p> <p>2 = zwymiarowanie punkt narożny, położenie kieszeni +LENG/+WID</p> <p>3 = zwymiarowanie punkt narożny, położenie kieszeni -LENG/+WID</p> <p>4 = zwymiarowanie punkt narożny, położenie kieszeni +LENG/-WID</p> <p>5 = zwymiarowanie punkt narożny, położenie kieszeni -LENG/-WID</p> <p>DZIESIĄTKI TYSIĘCY: Obróbka kompletna/obróbka poprawkowa</p> <p>0 = tryb kompatybilności (_AP1, _AP2 i _AD traktować jak dotychczas)</p> <p>1 = obróbka kompletna</p> <p>2 = obróbka poprawkowa</p>
28		_DMODE	<p>Tryb wyświetlania</p> <p>JEDNOSTKI: Płaszczyzna obróbki G17/G18/G19</p> <p>0 = kompatybilność, aktywna pozostaje płaszczyzna działająca przed wywołaniem cyklu</p> <p>1 = G17 (aktywna tylko w cyklu)</p> <p>2 = G18 (aktywna tylko w cyklu)</p> <p>3 = G19 (aktywna tylko w cyklu)</p> <p>DZIESIĄTKI: Rodzaj posuwu: Grupa G (G94/G95) dla posuwu dla powierzchni i na głębokość</p> <p>0 = tryb kompatybilności</p> <p>1 = G-Code jak przed wywołaniem cyklu. G94/G95 dla posuwu dla powierzchni i na głębokość takie same</p>
29		_AMODE	<p>Tryb alternatywny</p> <p>JEDNOSTKI: Głębokość kieszeni (Z1)</p> <p>0 = absolutnie (tryb kompatybilności)</p> <p>1 = przyrostowo</p> <p>DZIESIĄTKI: Jednostka dla dosuwu w płaszczyźnie (DXY)</p> <p>0 = mm</p> <p>1 = % średnicy narzędzia</p> <p>SETKI: Głębokość zagłębienia przy fazowaniu (ZFS)</p> <p>0 = absolutnie</p> <p>1 = przyrostowo</p>

18.1.16 Frezowanie kieszeni kołowej - POCKET4

Programowanie

```
POCKET4(REAL _RTP, REAL _RFP, REAL _SDIS, REAL _DP, REAL _CDIAM,
REAL _PA, REAL _PO, REAL _MID, REAL _FAL, REAL _FALD, REAL _FFP1,
REAL _FFD, INT _CDIR, INT _VARI, REAL _MIDA, REAL _AP1, REAL _AD,
REAL _RAD1, REAL _DP1, INT _UMODE, REAL _FS, REAL _ZFS, INT _GMODE,
INT _DMODE, INT _AMODE)
```

Parametry

Nr	Okno param etrów	Parametry wewnętrzne	Objaśnienie
1	RP	_RTP	Płaszczyzna wycofania (abs)
2	Z0	_RFP	Punkt odniesienia osi narzędzia (abs)
3	SC	_SDIS	Odstęp bezpieczeństwa (addytywnie do punktu odniesienia, wprowadzić bez znaku)
4	Z1	_DP	Głębokość kieszeni (abs./przyr.), patrz _AMODE
5	Ø	_DIAM	Średnica albo promień kieszeni, patrz _DMODE
6	X0	_PA	Punkt odniesienia 1. oś (abs)
7	Y0	_PO	Punkt odniesienia 2. oś (abs)
8	DZ	_MID	Maksymalny dosuw na głębokość, patrz _VARI = płaszczyznami Maksymalny skok linii spiralnej, patrz _VARI = linia spiralna
9	UXY	_FAL	Naddatek w płaszczyźnie
10	UZ	_FALD	Naddatek na głębokości
11	F	_FFP1	Posuw dla obróbki płaszczyzny
12	FZ	_FFD	Posuw dla dosuwu na głębokość
13		_CDIR	Kierunek frezowania 0 = frezowanie współbieżne 1 = frezowanie przeciwbieżne
14		_VARI	Rodzaj obróbki <hr/> JEDNOSTKI: 1 = obróbka zgrubna 2 = obróbka wykańczająca 4 = obróbka wykańczająca obrzeża 5 = fazowanie <hr/> DZIESIĄTKI: Rodzaj dosuwu (obróbka zgrubna i wykańczająca) 0 = z nawiercaniem, dosuw z G0 (kieszeń jest po obróbce wstępnej) 1 = prostopadle, dosuw z G1 2 = spiralnie <hr/> SETKI: Zarezerwowano <hr/> TYSIĄCE: 0 = płaszczyznami 1 = spiralnie

Nr	Okno param etrów	Parametry wewnętrzne	Objaśnienie
15	DXY	_MIDA	Maksymalny dosuw w płaszczyźnie, patrz _AMODE, $0 = 0,8 \cdot \text{średnica WZG}$
16	Ø	_AP1	Średnica/promień kieszeni półfabrykatu (przyr.)
17	AZ	_AD	Głębokość kieszeni półfabrykatu (przyr.)
18	ER	_RAD1	Promień toru spiralnego przy zagłębieniu
19	EP	_DP1	Skok przy zagłębieniu po linii spiralnej
20		_UMODE	Zarezerwowano
21	FS	_FS	Szerokość fazy (przyr.)
22	ZFS	_ZFS	Głębokość zagłębienia (wierzchołka narzędzia) przy fazowaniu (abs./przyr.), patrz _AMODE
23		_GMODE	Tryb geometrii JEDNOSTKI: Zarezerwowano DZIESIĄTKI: Zarezerwowano SETKI: Obróbka/obliczenie punktu startowego 0 = tryb kompatybilności 1 = normalna obróbka TYSIĄCE: Zarezerwowano DZIESIĄTKI TYSIĘCY: Obróbka kompletna/obróbka poprawkowa 0 = tryb kompatybilności (_AP1 i _AD traktować jak dotychczas) 1 = obróbka kompletna 2 = obróbka poprawkowa
24		_DMODE	Tryb wyświetlania JEDNOSTKI: Płaszczyzna obróbki G17/18/19 0 = kompatybilność, aktywna pozostaje płaszczyzna działająca przed wywołaniem cyklu 1 = G17 (aktywna tylko w cyklu) 2 = G18 (aktywna tylko w cyklu) 3 = G19 (aktywna tylko w cyklu) DZIESIĄTKI: Rodzaj posuwu: Grupa G (G94/G95) dla posuwu dla powierzchni i na głębokość 0 = tryb kompatybilności 1 = G-Code jak przed wywołaniem cyklu. G94/G95 dla posuwu dla powierzchni i na głębokość takie same SETKI: 0 = tryb kompatybilności (_CDIAM/_AP1 wprowadzone jako promień) 1 = _CDIAM/_AP1 wprowadzone jako średnica

Nr	Okno parametrów	Parametry wewnętrzne	Objaśnienie
25		<code>_AMODE</code>	Tryb alternatywny <hr/> JEDNOSTKI: Głębokość kieszeni (Z1) 0 = absolutnie (tryb kompatybilności) 1 = przyrostowo <hr/> DZIESIĄTKI: Jednostka szerokości dosuwu (DXY) 0 = mm 1 = % średnicy narzędzia <hr/> SETKI: Głębokość zagłębienia przy fazowaniu (ZFS) 0 = absolutnie 1 = przyrostowo

18.1.17 Frezowanie czopa prostokątnego - CYCLE76

Programowanie

```
CYCLE76 (REAL _RTP, REAL _RFP, REAL _SDIS, REAL _DP, REAL _DPR, REAL
_LENG, REAL _WID, REAL _CRAD, REAL _PA, REAL _PO, REAL _STA, REAL
_MID, REAL _FAL, REAL _FALD, REAL _FFP1, REAL _FFD, INT _CDIR, INT
_VARI, REAL _AP1, REAL _AP2, REAL _FS, REAL _ZFS, INT _GMODE, INT
_DMODE, INT _AMODE)
```

Parametry

Nr	Okno parametrów	Parametry wewnętrzne	Objaśnienie
1	RP	<code>_RTP</code>	Płaszczyzna wycofania (abs)
2	Z0	<code>_RFP</code>	Punkt odniesienia osi narzędzia (abs)
3	SC	<code>_SDIS</code>	Odstęp bezpieczeństwa (addytywnie do punktu odniesienia, wprowadzić bez znaku)
4	Z1	<code>_DP</code>	Głębokość czopa (abs.)
5		<code>_DPR</code>	Głębokość czopa (przyr.), w odniesieniu do Z0 (wprowadzić bez znaku)
6	L	<code>_LENG</code>	Długość czopa, patrz <code>_GMODE</code> (wprowadzić bez znaku)
7	W	<code>_WID</code>	Szerokość czopa, patrz <code>_GMODE</code> (wprowadzić bez znaku)
8	R	<code>_CRAD</code>	Promień naroża czopa (wprowadzić bez znaku)
9	X0	<code>_PA</code>	Punkt odniesienia czopa, 1. oś płaszczyzny (abs.)
10	Y0	<code>_PO</code>	Punkt odniesienia czopa, 2. oś płaszczyzny (abs.)
11	$\alpha 0$	<code>_STA</code>	Kąt obrotu, kąt między osią podłużną (L) i 1. osią płaszczyzny
12	DZ	<code>_MID</code>	Maksymalny dosuw na głębokość (przyr., wprowadzić bez znaku)
13	UXY	<code>_FAL</code>	Naddatek w płaszczyźnie (przyr.), naddatek na obrzeżu konturu

Nr	Okno parametrów	Parametry wewnętrzne	Objaśnienie
14	UZ	_FALD	Naddatek na głębokości (przyr.), naddatek na dnie (wprowadzić bez znaku)
15	FX	_FFP1	Posuw po konturze
16	FZ	_FFD	Posuw dla dosuwu na głębokość
17		_CDIR	Kierunek frezowania (wprowadzić bez znaku)
			JEDNOSTKI: 0 = frezowanie współbieżne 1 = frezowanie przeciwbieżne
18		_VARI	Obróbka
			JEDNOSTKI: 1 = obróbka zgrubna 2 = obróbka wykańczająca 5 = fazowanie
19	L1	_AP1	Długość czopa półfabrykatu
20	W1	_AP2	Szerokość czopa półfabrykatu
21	FS	_FS	Szerokość fazy (przyr.)
22	ZFS	_ZFS	Głębokość zagłębienia (wierzchołka narzędzia) przy fazowaniu (abs., przyr.), patrz _AMODE
23		_GMODE	Tryb ewaluacji zaprogramowanych wartości geometrycznych
			JEDNOSTKI: Zarezerwowano
			DZIESIĄTKI: Zarezerwowano
			SETKI: Wybór obróbka albo tylko obliczenie punktu startowego 0 = tryb kompatybilności 1 = normalna obróbka
			TYSIĄCE: Zwymiarowanie czopa przez środek albo naroże 0 = tryb kompatybilności 1 = zwymiarowanie przez środek 2 = zwymiarowanie punktu narożnego, czop +L +W 3 = zwymiarowanie punktu narożnego, czop -L +W 4 = zwymiarowanie punktu narożnego, czop +L -W 5 = zwymiarowanie punktu narożnego, czop -L -W
			DZIESIĄTKI TYSIĘCY: Obróbka kompletna albo obróbka poprawkowa 0 = tryb kompatybilności 1 = obróbka kompletna 2 = obróbka poprawkowa
24		_DMODE	Tryb wyświetlania
			JEDNOSTKI: Płaszczyzna obróbki G17/G18/G19 0 = kompatybilność, aktywna pozostaje płaszczyna działająca przed wywołaniem cyklu 1 = G17 (aktywna tylko w cyklu) 2 = G18 (aktywna tylko w cyklu) 3 = G19 (aktywna tylko w cyklu)

Nr	Okno parametrów	Parametry wewnętrzne	Objaśnienie
25		_AMODE	Tryb alternatywny <hr/> JEDNOSTKI: Głębokość końcowa Z1 (abs/ink) <hr/> 0 = kompatybilność 1 = Z1 (przyr) 2 = Z1 (abs) <hr/> DZIESIĄTKI: Zarezerwowano <hr/> SETKI: Głębokość zagłębienia przy fazowaniu ZFS <hr/> 0 = ZFS (abs) 1 = ZFS (ink)

18.1.18 Frezowanie czopa kołowego - CYCLE77

Programowanie

```
CYCLE77(REAL _RTP, REAL _RFP, REAL _SDIS, REAL _DP, REAL _DPR, REAL
_CDIAM, REAL _PA, REAL _PO, REAL _MID, REAL _FAL, REAL _FALD, REAL
_FFP1, REAL _FFD, INT _CDIR, INT _VARI, REAL _AP1, REAL _FS, REAL
_ZFS, INT _GMODE, INT _DMODE, INT _AMODE)
```

Parametry

Nr	Okno parametry	Parametr wewnętrzne	Objaśnienie
1	RP	_RTP	Płaszczyzna wycofania (abs)
2	Z0	_RFP	Punkt odniesienia osi narzędzia (abs)
3	SC	_SDIS	Odstęp bezpieczeństwa (addytywnie do punktu odniesienia, wprowadzić bez znaku)
4	Z1	_DP	Głębokość czopa (abs.)
5		_DPR	Głębokość czopa (przyr.), w odniesieniu do Z0 (wprowadzić bez znaku)
6	Ø	_CDIAM	Średnica czopa (wprowadzić bez znaku)
7	X0	_PA	Punkt odniesienia czopa, 1. oś płaszczyzny (abs.)
8	Y0	_PO	Punkt odniesienia czopa, 2. oś płaszczyzny (abs.)
9	DZ	_MID	Maksymalny dosuw na głębokość (przyr., wprowadzić bez znaku)
10	UXY	_FAL	Naddatek w płaszczyźnie (przyr.), naddatek na obrzeżu konturu
11	UZ	_FALD	Naddatek na głębokości (przyr.), naddatek na dnie (wprowadzić bez znaku)
12	FX	_FFP1	Posuw po konturze
13	FZ	_FFD	Posuw dla dosuwu na głębokość

Nr	Okno paramet rów	Parametr wewnętrzne	Objaśnienie
14		_CDIR	Kierunek frezowania (wprowadzić bez znaku) JEDNOSTKI: 0 = frezowanie współbieżne 1 = frezowanie przeciwbieżne
15		_VARI	Obróbka JEDNOSTKI: 1 = obróbka zgrubna, aż do naddatku na obróbkę wykańczającą 2 = obróbka wykańczająca (naddatek X/Y/Z = 0) 5 = fazowanie
16	Ø1	_AP1	Średnica czopa półfabrykatu
17	FS	_FS	Szerokość fazy (przyr.)
18	ZFS	_ZFS	Głębokość zagłębienia (wierzchołka narzędzia) przy fazowaniu (abs./przyr.), patrz _AMODE
19		_GMODE	Tryb ewaluacji zaprogramowanych wartości geometrycznych JEDNOSTKI: Zarezerwowano DZIESIĄTKI: Zarezerwowano SETKI: Wybór obróbka/tylko obliczenie punktu startowego 0 = tryb kompatybilności 1 = normalna obróbka TYSIĄCE: Zarezerwowano DZIESIĄTKI TYSIĘCY: Obróbka kompletna/obróbka poprawkowa 0 = tryb kompatybilności (_AP1 traktować jak dotychczas) 1 = obróbka kompletna 2 = obróbka poprawkowa
20		_DMODE	Tryb wyświetlania JEDNOSTKI: Płaszczyzna obróbki G17/G18/G19 0 = kompatybilność, aktywna pozostaje płaszczyzna działająca przed wywołaniem cyklu 1 = G17 (aktywna tylko w cyklu) 2 = G18 (aktywna tylko w cyklu) 3 = G19 (aktywna tylko w cyklu)
21		_AMODE	Tryb alternatywny JEDNOSTKI: Głębokość końcowa Z1 (abs/ink) 0 = kompatybilność 1 = Z1 (przyr) 2 = Z1 (abs) DZIESIĄTKI: Zarezerwowano SETKI: Głębokość zagłębienia przy fazowaniu ZFS 0 = ZFS (abs) 1 = ZFS (ink)

18.1.19 Wielobok - CYCLE79

Programowanie

```
CYCLE79 (REAL _RTP, REAL _RFP, REAL _SDIS, REAL _DP, INT _NUM, REAL
_SWL, REAL _PA, REAL _PO, REAL _STA, REAL _RC, REAL _AP1, REAL
_MIDA, REAL _MID, REAL _FAL, REAL _FALD, REAL _FFP1, INT _CDIR, INT
_VARI, REAL _FS, REAL _ZFS, INT _GMODE, INT _DMODE, INT _AMODE)
```

Parametry

Nr	Okno paramet rów	Parametr wewnętrzne	Objaśnienie
1	RP	_RTP	Płaszczyzna wycofania (abs)
2	Z0	_RFP	Punkt odniesienia osi narzędzia (abs)
3	SC	_SDIS	Odstęp bezpieczeństwa (addytywnie do punktu odniesienia, wprowadzić bez znaku)
4	Z1	_DP	Głębokość wieloboku (abs./przyr.) patrz _AMODE
5	N	_NUM	Liczba krawędzi (1...n)
6	SW/L	_SWL	Rozstaw klucza albo długość krawędzi (zależnie od _VARI) ("SW" w przypadku rozstawu klucza, "L" w przypadku długości krawędzi) Rozstaw klucza tylko przy parzystej liczbie krawędzi i jednej krawędzi
7	X0	_PA	Punkt odniesienia czopa, 1. oś (abs.)
8	Y0	_PO	Punkt odniesienia czopa, 2. oś (abs.)
9	$\alpha 0$	_STA	Kąt obrotu środka krawędzi w stosunku do 1. osi (oś X)
10	R1/FS1	_RC	Zaokrąglenie naroża przy _NUM > 2 (promień/faza, patrz _AMODE) (przyr., wprowadzić bez znaku) ("R1" w przypadku promienia, "FS1" w przypadku fazy)
11	\emptyset	_AP1	Średnica półfabrykatu czopa
12	DXY	_MIDA	Maksymalna szerokość dosuwu (jednostka, patrz _AMODE)
13	DZ	_MID	Maksymalny dosuw na głębokość
14	UXY	_FAL	Naddatek w płaszczyźnie
15	UZ	_FALD	Naddatek na głębokości
16	F	_FFP1	Posuw roboczy
17		_CDIR	Kierunek frezowania 0 = frezowanie współbieżne 1 = frezowanie przeciwbieżne

Nr	Okno paramet rów	Parametr wewnętrzne	Objaśnienie
18		_VARI	Rodzaj obróbki JEDNOSTKI: Obróbka 1 = obróbka zgrubna 2 = obróbka wykańczająca 3 = obróbka wykańczająca obrzeża 5 = fazowanie DZIESIĄTKI: Rozstaw klucza lub długość krawędzi 0 = rozstaw klucza 1 = długość krawędzi
19	FS	_FS	Szerokość fazy (przyr.)
20	ZFS	_ZFS	Głębokość zagłębienia (wierzchołka narzędzia) przy fazowaniu (abs./przyr.), patrz _AMODE
21		_GMODE	Tryb geometrii JEDNOSTKI: Zarezerwowano DZIESIĄTKI: Zarezerwowano SETKI: Obróbka/obliczenie punktu startowego 1 = normalna obróbka
22		_DMODE	Tryb wyświetlania JEDNOSTKI: Płaszczyzna obróbki G17/G18/G19 0 = kompatybilność, aktywna pozostaje płaszczyzna działająca przed wywołaniem cyklu 1 = G17 (aktywna tylko w cyklu) 2 = G18 (aktywna tylko w cyklu) 3 = G19 (aktywna tylko w cyklu)
23		_AMODE	Tryb alternatywny JEDNOSTKI: Głębokość końcowa (_DP) 0 = absolutnie 1 = przyrostowo DZIESIĄTKI: Jednostka dla dosuwu w płaszczyźnie (_MIDA) 0 = mm 1 = % średnicy narzędzia SETKI: Głębokość zagłębienia przy fazowaniu (_ZFS) 0 = absolutnie 1 = przyrostowo TYSIĄCE: Zaokrąglenie naroża (_RC) 0 = promień 1 = faza

18.1.20 Rowek podłużny - SLOT1

Programowanie

SLOT1 (REAL RTP, REAL RFP, REAL SDIS, REAL _DP, REAL _DPR, INT NUM, REAL LENG, REAL WID, REAL _CPA, REAL _CPO, REAL RAD, REAL STA1, REAL INDA, REAL FFD, REAL FFP1, REAL _MID, INT CDIR, REAL _FAL, INT VARI, REAL _MIDF, REAL FFP2, REAL SSF, REAL _FALD, REAL _STA2, REAL _DP1, INT _UMODE, REAL _FS, REAL _ZFS, INT _GMODE, INT _DMODE, INT _AMODE)

Parametry

Nr	Okno param etrów	Parametry wewnętrzne	Objaśnienie
1	RP	RTP	Płaszczyzna wycofania (abs)
2	Z0	RFP	Punkt odniesienia osi narzędzia (abs)
3	SC	SDIS	Odstęp bezpieczeństwa (addytywnie do punktu odniesienia, wprowadzić bez znaku)
4	Z1	_DP	Głębokość rowka (abs.)
5		_DPR	Głębokość rowka (przyr.), w odniesieniu do Z0 (wprowadzić bez znaku)
6		NUM	Liczba rowków = 1
7	L	LENG	Długość rowka
8	W	WID	Szerokość rowka
9	X0	_CPA	Punkt odniesienia, 1. oś płaszczyzny
10	Y0	_CPO	Punkt odniesienia, 2. oś płaszczyzny
11		_RAD	Zarezerwowano
12	α	STA1	Kąt obrotu
13		INDA	Zarezerwowano
14	FZ	FFD	Posuw dla dosuwu na głębokość
15	F	FFP1	Posuw
16	DZ	_MID	Maksymalny dosuw na głębokość
17		CDIR	Kierunek frezowania 0 = frezowanie współbieżne 1 = frezowanie przeciwbieżne
18	UXY	_FAL	Naddatek w płaszczyźnie albo na obrzeżu rowka
19		VARI	Rodzaj obróbki JEDNOSTKI: 0 = zarezerwowano 1 = obróbka zgrubna 2 = obróbka wykańczająca 4 = obróbka wykańczająca na obrzeżu (obróbka tylko obrzeża) 5 = fazowanie DZIESIĄTKI: Dosunięcie

Nr	Okno param etrów	Parametry wewnętrzne	Objaśnienie
			0 = z nawiercaniem, dosuw z G0 (rowek jest po obróbce wstępnej) 1 = prostopadle, dosuw z G1 2 = ruchem spiralnym 3 = ruchem wahadłowym
			SETKI: Zarezerwowano
20	DZF	MIDF	Zarezerwowano
21	FF	FFP2	Zarezerwowano
22	SF	SSF	Zarezerwowano
23	UZ	_FALD	Naddatek na głębokości
24	ER	_STA2	Promień toru spiralnego przy zagłębieniu
	EW		Maksymalny kąt zagłębienia dla ruchu wahlowego
25	EP	_DP1	Głębokość zagłębienia na obrót dla toru spiralnego
26		_UMODE	Zarezerwowano
27	FS	_FS	Szerokość fazy (przyr.) tylko przy fazowaniu
28	ZFS	_ZFS	Głębokość zagłębienia (wierzchołka narzędzia) przy fazowaniu (abs./przyr.), patrz _AMODE
29		_GMODE	Tryb geometrii JEDNOSTKI: Zarezerwowano DZIESIĄTKI: Zarezerwowano SETKI: Wybór obróbka albo tylko obliczenie punktu startowego 1 = normalna obróbka TYSIĄCE: Wymiarowanie punkt odniesienia, położenie rowka 0 = środek 1 = po lewej wewnątrz +L 2 = po prawej wewnątrz -L 3 = po lewej obrzeże +L 4 = po prawej obrzeże -L
30		_DMODE	Tryb wyświetlania JEDNOSTKI: Płaszczyzna obróbki G17/18/19 0 = kompatybilność, aktywna pozostaje płaszczyzna działająca przed wywołaniem cyklu 1 = G17 (aktywna tylko w cyklu) 2 = G18 (aktywna tylko w cyklu) 3 = G19 (aktywna tylko w cyklu) DZIESIĄTKI: Zarezerwowano SETKI: Zarezerwowano TYSIĄCE: Oznaczenie wersji oprogramowania 1 = rozszerzenie funkcji SLOT1

Nr	Okno param etrów	Parametry wewnętrzne	Objaśnienie
31		<code>_AMODE</code>	Tryb alternatywny <hr/> JEDNOSTKI: Głębokość końcowa Z1 (abs/ink) 0 = kompatybilność 1 = Z1 (przyr) 2 = Z1 (abs) <hr/> DZIESIĄTKI: Zarezerwowano <hr/> SETKI: Głębokość zagłębienia przy fazowaniu ZFS 0 = ZFS (abs) 1 = ZFS (ink)

Wskazówka

Cykl jest w porównaniu z wcześniejszymi wersjami oprogramowania wyposażony w nowe funkcje. Skutkuje to tym, że określone parametry nie są już wyświetlane w oknie wprowadzania (`NUM`, `RAD`, `INDA`). Wiele rowków na jednym szablonie pozycji można programować przy pomocy "MCALL" i wywołanie pożądanego szablonu, np. HOLES2.

18.1.21 Rowek kołowy - SLOT2

Programowanie

```
SLOT2 (REAL RTP, REAL RFP, REAL SDIS, REAL _DP, REAL _DPR, INT NUM,
REAL AFSL, REAL WID, REAL _CPA, REAL _CPO, REAL RAD, REAL STA1, REAL
INDA, REAL FFD, REAL FFP1, REAL _MID, INT CDIR, REAL _FAL, INT VARI,
REAL _MIDF, REAL FFP2, REAL SSF, REAL _FFCP, INT _UMODE, REAL _FS,
REAL _ZFS, INT _GMODE, INT _DMODE, INT _AMODE)
```

Parametry

Nr	Okno param etrów	Parametry wewnętrzne	Objaśnienie
1	RP	RTP	Płaszczyzna wycofania (abs)
2	Z0	RFP	Punkt odniesienia osi narzędzia (abs)
3	SC	SDIS	Odstęp bezpieczeństwa (addytywnie do punktu odniesienia, wprowadzić bez znaku)
4	Z1	_DP	Głębokość rowka (abs.)
5		_DPR	Głębokość rowka (przyr.), w odniesieniu do Z0 (wprowadzić bez znaku)
6	N	NUM	Liczba rowków
7	$\alpha 1$	AFSL	Kąt rozwarcia rowka

Nr	Okno parametrów	Parametry wewnętrzne	Objaśnienie
8	W	WID	Szerokość rowka
9	X0	_CPA	Punkt odniesienia = punkt środkowy okręgu, 1. oś płaszczyzny
10	Y0	_CPO	Punkt odniesienia = punkt środkowy okręgu, 2. oś płaszczyzny
11	R	RAD	Promień okręgu
12	$\alpha 0$	STA1	Kąt początkowy
13	$\alpha 2$	INDA	Kąt przełączania
14	FZ	FFD	Posuw dla dosuwu na głębokość
15	F	FFP1	Posuw
16	DZ	_MID	Maksymalny dosuw na głębokość
17		CDIR	Kierunek frezowania 0 = frezowanie współbieżne 1 = frezowanie przeciwbieżne
18	UXY	_FAL	Naddatek na obróbkę wykańczającą w płaszczyźnie albo na obrzeżu
19		VARI	Rodzaj obróbki JEDNOSTKI: 0 = obróbka kompletna 1 = obróbka zgrubna 2 = obróbka wykańczająca 3 = obróbka wykańczająca obrzeża 5 = fazowanie DZIESIĄTKI: 0 = pozycjonowanie pośrednie z prostą G0 1 = pozycjonowanie pośrednie na torze kołowym SETKI: Zarezerwowano TYSIĄCE: 0 = tryb kompatybilności, gdy INDA = 0 wówczas pełny okrąg, INDA <> 0 wówczas łuk koła) 1 = okrąg 2 = łuk koła
20	DZF	_MIDF	Zarezerwowano
21		FFP2	Zarezerwowano
22		SSF	Zarezerwowano
23	FF	_FFCP	Zarezerwowano
24		_UMODE	Zarezerwowano
25	FS	_FS	Szerokość fazy (przyr.)
26	ZFS	_ZFS	Głębokość zagłębienia (wierzchołka narzędzia) przy fazowaniu (abs./przyr.), patrz _AMODE

Nr	Okno parametrów	Parametry wewnętrzne	Objaśnienie
27		_GMODE	<p>Tryb geometrii</p> <hr/> <p>JEDNOSTKI: Zarezerwowano</p> <hr/> <p>DZIESIĄTKI: Zarezerwowano</p> <hr/> <p>SETKI: Wybór obróbka albo tylko obliczenie punktu startowego</p> <hr/> <p>0 = tryb kompatybilności</p> <p>1 = normalna obróbka</p>
28		_DMODE	<p>Tryb wyświetlania</p> <hr/> <p>JEDNOSTKI: Płaszczyzna obróbki G17/18/19</p> <hr/> <p>0 = kompatybilność, aktywna pozostaje płaszczyzna działająca przed wywołaniem cyklu</p> <p>1 = G17 (aktywna tylko w cyklu)</p> <p>2 = G18 (aktywna tylko w cyklu)</p> <p>3 = G19 (aktywna tylko w cyklu)</p> <hr/> <p>DZIESIĄTKI: Zarezerwowano</p> <hr/> <p>SETKI: Zarezerwowano</p> <hr/> <p>TYSIĄCE: Oznaczenie wersji oprogramowania</p> <hr/> <p>1 = funkcje SLOT2 od w. opr. 2.5</p>
29		_AMODE	<p>Tryb alternatywny</p> <hr/> <p>JEDNOSTKI: Głębokość końcowa Z1 (abs/ink)</p> <hr/> <p>0 = kompatybilność</p> <p>1 = Z1 (przyr)</p> <p>2 = Z1 (abs)</p> <hr/> <p>DZIESIĄTKI: Zarezerwowano</p> <hr/> <p>SETKI: Głębokość zagłębienia przy fazowaniu ZFS</p> <hr/> <p>0 = ZFS (abs)</p> <p>1 = ZFS (ink)</p>

18.1.22 Frezowanie rowka otwartego - CYCLE899

Programowanie

```
CYCLE899 (REAL _RTP, REAL _RFP, REAL _SDIS, REAL _DP, REAL _LENG,
REAL _WID, REAL _PA, REAL _PO, REAL _STA, REAL _MID, REAL _MIDA,
REAL _FAL, REAL _FALD, REAL _FFP1, INT _CDIR, INT _VARI, INT _GMODE,
INT _DMODE, INT _AMODE, INT _UMODE, REAL _FS, REAL _ZFS)
```

Parametry

Nr	Okno parametrow	Parametr wewnętrzny	Objaśnienie
1	RP	_RTP	Płaszczyzna wycofania (abs)
2	Z0	_RFP	Punkt odniesienia osi narzędzia (abs)
3	SC	_SDIS	Odstęp bezpieczeństwa (addytywnie do punktu odniesienia, wprowadzić bez znaku)
4	Z1	_DP	Głębokość rowka (abs./przyr.), patrz _AMODE
5	L	_LENG	Długość rowka (przyr.)
6	W	_WID	Szerokość rowka (przyr.)
7	X0	_PA	Punkt odniesienia/pozycja startowa 1. oś (abs)
8	Y0	_PO	Punkt odniesienia/pozycja startowa 2. oś (abs)
9	$\alpha 0$	_STA	Kąt obrotu w stosunku do 1. osi
10	DZ	_MID	Maksymalna głębokość dosuwu (przyr.), tylko dla frezowania trochoidalnego
11	DXY	_MIDA	Maksymalny dosuw w płaszczyźnie, patrz _AMODE
12	UXY	_FAL	Naddatek w płaszczyźnie
13	UZ	_FALD	Naddatek na głębokości
14	F	_FFP1	Posuw
15		_CDIR	Kierunek frezowania
			JEDNOSTKI:
			0 = frezowanie współbieżne
			1 = frezowanie przeciwbieżne
			4 = naprzemiennie
16		_VARI	Obróbka
			JEDNOSTKI:
			1 = obróbka zgrubna
			2 = obróbka wykańczająca
			3 = obróbka wykańczająca dna
			4 = obróbka wykańczająca obrzeża
			5 = obróbka półwykańczająca
			6 = fazowanie
			DZIESIĄTKI: Zarezerwowano
			SETKI: Zarezerwowano

Nr	Okno paramet rów	Parametr wewnętrzne	Objaśnienie
			<p>TYSIĄCE:</p> <p>1 = frezowanie trochoidalne</p> <p>2 = frezowanie wgłębne</p>
17	_GMODE		<p>Oszacowanie wartości geometrycznych</p> <p>JEDNOSTKI: Zarezerwowano</p> <p>DZIESIĄTKI: Zarezerwowano</p> <p>SETKI: Wybór obróbka/tylko obliczenie punktu startowego</p> <p>1 = normalna obróbka</p> <p>TYSIĄCE: Zwymiarowanie przez środek/krawędź</p> <p>0 = zwymiarowanie przez środek</p> <p>1 = zwymiarowanie przez krawędź "lewą" ("- " kierunek 1. osi)</p> <p>2 = zwymiarowanie przez krawędź "prawą" ("+" kierunek 1. osi)</p>
18	_DMODE		<p>Tryb wyświetlania</p> <p>JEDNOSTKI: Płaszczyzna obróbki G17/G18/G19</p> <p>0 = kompatybilność, aktywna pozostaje płaszczyzna działająca przed wywołaniem cyklu</p> <p>1 = G17 (aktywna tylko w cyklu)</p> <p>2 = G18 (aktywna tylko w cyklu)</p> <p>3 = G19 (aktywna tylko w cyklu)</p>
19	_AMODE		<p>Tryb alternatywny</p> <p>JEDNOSTKI: Głębokość rowka Z1</p> <p>0 = absolutnie</p> <p>1 = przyrostowo</p> <p>DZIESIĄTKI: Jednostka dla dosuwu w płaszczyźnie (_MIDA) DXY</p> <p>0 = mm</p> <p>1 = % średnicy narzędzia</p> <p>SETKI: Głębokość zagłębienia przy fazowaniu ZFS</p> <p>0 = absolutnie</p> <p>1 = przyrostowo</p>
20	_UMODE		Zarezerwowano
21	FS	_FS	Szerokość fazy (przyr.)
22	ZFS	_ZFS	Głębokość zagłębienia (wierzchołka narzędzia) przy fazowaniu (abs./przyr.), patrz _AMODE

18.1.23 Otwór podłużny - LONGHOLE

Programowanie

```
LONGHOLE (REAL RTP, REAL RFP, REAL SDIS, REAL _DP, REAL _DPR,
INT NUM, REAL LENG, REAL _CPA, REAL _CPO, REAL RAD, REAL STA1,
REAL INDA, REAL FFD, REAL FFP1, REAL MID, INT _VARI, INT _UMODE,
INT _GMODE, INT _DMODE, INT _AMODE)
```

Parametry

Nr	Okno param etrów	Param wewnętrzne	Objaśnienie
1	RP	RTP	Płaszczyzna wycofania (abs)
2	Z0	_RFP	Punkt odniesienia osi narzędzia (abs)
3	SC	SDIS	Odstęp bezpieczeństwa (addytywnie do punktu odniesienia, wprowadzić bez znaku)
4	Z1	_DP	Głębokość otworu podłużnego (abs.)
5		_DPR	Głębokość otworu podłużnego (przyr.), w odniesieniu do Z0 (wprowadzić bez znaku)
6		NUM	Liczba otworów podłużnych = 1
7	L	LENG	Długość otworu podłużnego
8	X0	_CPA	Punkt odniesienia, 1. oś płaszczyzny
9	Y0	_CPO	Punkt odniesienia, 2. oś płaszczyzny
10		RAD	Zarezerwowano
11	$\alpha 0$	STA1	Kąt obrotu
12		INDA	Zarezerwowano
13	FZ	FFD	Posuw dla dosuwu na głębokość
14	F	FFP1	Posuw
15	DZ	MID	Maksymalny dosuw na głębokość
16		_VARI	Rodzaj obróbki JEDNOSTKI: Rodzaj dosuwu 1 = prostopadle z G1 3 = ruchem wahadłowym SETKI: Zarezerwowano
17		_UMODE	Zarezerwowano

Nr	Okno parametrów	Param wewnętrzny	Objaśnienie
18		<code>_GMODE</code>	<p>Tryb geometrii</p> <hr/> <p>JEDNOSTKI: Zarezerwowano</p> <hr/> <p>DZIESIĄTKI: Zarezerwowano</p> <hr/> <p>SETKI: Wybór obróbka albo tylko obliczenie punktu startowego</p> <p>0 = tryb kompatybilności</p> <p>1 = normalna obróbka</p> <hr/> <p>TYSIĄCE: Wymiarowanie punkt odniesienia, położenie rowka</p> <p>0 = środek</p> <p>1 = po lewej wewnątrz +L</p> <p>2 = po prawej wewnątrz -L</p> <p>3 = po lewej obrzeże +L</p> <p>4 = po prawej obrzeże -L</p>
19		<code>_DMODE</code>	<p>Tryb wyświetlania</p> <hr/> <p>JEDNOSTKI: Płaszczyzna obróbki G17/18/19</p> <p>0 = kompatybilność, aktywna pozostaje płaszczyzna działająca przed wywołaniem cyklu</p> <p>1 = G17 (aktywna tylko w cyklu)</p> <p>2 = G18 (aktywna tylko w cyklu)</p> <p>3 = G19 (aktywna tylko w cyklu)</p> <hr/> <p>DZIESIĄTKI: Rodzaj posuwu: Grupa G (G94/G95) dla posuwu dla powierzchni i na głębokość</p> <p>0 = tryb kompatybilności</p> <p>1 = G-Code jak przed wywołaniem cyklu. G94/G95 dla posuwu dla powierzchni i na głębokość takie same</p> <hr/> <p>SETKI: Zarezerwowano</p> <hr/> <p>TYSIĄCE: Oznaczenie wersji oprogramowania</p> <p>1 = rozszerzenie funkcji LONGHOLE (wymiarowanie punktu odniesienia)</p>
20		<code>_AMODE</code>	<p>Tryb alternatywny</p> <hr/> <p>JEDNOSTKI: Głębokość końcowa Z1 (abs/ink)</p> <p>0 = kompatybilność</p> <p>1 = Z1 (przyr)</p> <p>2 = Z1 (abs)</p>

Wskazówka

Cykl jest w porównaniu z wcześniejszymi wersjami oprogramowania wyposażony w nowe funkcje. Skutkuje to tym, że określone parametry nie są już wyświetlane w oknie wprowadzania (NUM, RAD, INDA). Wiele rowków na jednym szablonie pozycji można programować przy pomocy "MCALL" i wywołania pożądanego szablonu, np. HOLES2.

18.1.24 Frezowanie gwintu - CYCLE70

Programowanie

```
CYCLE70 (REAL _RTP, REAL _RFP, REAL _SDIS, REAL _DP, REAL _DIATH,
REAL _H1, REAL _FAL, REAL _PIT, INT _NT, REAL _MID, REAL _FFR, INT
_TYPH, REAL _PA, REAL _PO, REAL _NSP, INT _VARI, INT _PITA,
STRING[15] _PITM, STRING[20] _PTAB, STRING[20] _PTABA, INT _GMODE,
INT _DMODE, INT _AMODE)
```

Parametry

Nr	Okno param etrów	Parametry wewnętrzne	Objaśnienie
1	RP	_RTP	Płaszczyzna wycofania (abs)
2	Z0	_RFP	Punkt odniesienia osi narzędzia (abs)
3	SC	_SDIS	Odstęp bezpieczeństwa (addytywnie do punktu odniesienia, wprowadzić bez znaku)
4	Z1	_DP	Długość gwintu (abs., przyr.), patrz _AMODE Uwzględnienie wybiegu na dnie otworu (minimum pół skoku)
5	Ø	_DIATH	Średnica nominalna gwintu
6	H1	_H1	Głębokość gwintu
7	U	_FAL	Naddatek na obróbkę wykańczającą
8	P	_PIT	Skok gwintu (wybór_PITA: mm, cale, MODUL, zwojów/cal)
9	NT	_NT	Liczba zębów na płycie skrawającej Długość narzędzia zawsze odniesiona do dolnego zęba!
10	DXY	_MID	Dosuw maksymalny na przejście _MID > _H1: wszystko za jednym przejściem
11	F	_FFR	Posuw frezowania
12		_TYPH	Typ gwintu 0 = gwint wewnętrzny 1 = gwint zewnętrzny
13	X0	_PA	Punkt środkowy gwintu, 1. oś (abs.)
14	Y0	_PO	Punkt środkowy gwintu, 2. oś (abs.)
15	αS	_NSP	Kąt startowy (gwint wielokrotny)
16		_VARI	Rodzaj obróbki JEDNOSTKI: 1 = obróbka zgrubna 2 = obróbka wykańczająca DZIESIĄTKI: 1 = od góry do dołu 2 = od dołu do góry SETKI: 0 = gwint prawy 1 = gwint lewy

Nr	Okno parametrów	Parametry wewnętrzne	Objaśnienie
17		_PITA	Oszacowanie skoku gwintu 0 = tryb kompatybilności 1 = skok w mm 2 = skok w zwojach na cal (TPI) 3 = skok w calach 4 = skok jako MODUŁ
18		_PITM	Łańcuch znaków jako znacznik dla wprowadzenia skoku gwintu (tylko dla powierzchni)
19		_RTAB	Łańcuch znaków dla tablicy gwintów ("", "ISO", "BSW", "BSP", "UNC") (tylko dla powierzchni)
20		_PTABA	Łańcuch znaków dla wyboru w tablicy gwintów (np. "M 10", "M 12", ...) (tylko dla powierzchni)
21		_GMODE	Tryb geometrii JEDNOSTKI: Zarezerwowano DZIESIĄTKI: Zarezerwowano SETKI: Obróbka/obliczenie punktu startowego 0 = tryb kompatybilności 1 = normalna obróbka
22		_DMODE	Tryb wyświetlania JEDNOSTKI: Płaszczyzna obróbki G17/18/19 0 = kompatybilność, aktywna pozostaje płaszczyna działająca przed wywołaniem cyklu 1 = G17 (aktywna tylko w cyklu) 2 = G18 (aktywna tylko w cyklu) 3 = G19 (aktywna tylko w cyklu)
23		_AMODE	Tryb alternatywny JEDNOSTKI: Głębokość gwintu (_DP) 0 = absolutnie 1 = przyrostowo

18.1.25 Cykl grawerowania - CYCLE60

Programowanie

```
CYCLE60 (STRING [200] _TEXT, REAL _RTP, REAL _RFP, REAL _SDIS, REAL
_DP, REAL _DPR, REAL _PA, REAL _PO, REAL _STA, REAL _CP1, REAL _CP2,
REAL _WID, REAL _DF, REAL _FFD, REAL _FFP1, INT _VARI, INT _CODEP,
INT _UMODE, INT _GMODE, INT _DMODE, INT _AMODE)
```

Parametry

Nr	Okno param etrów	Parametry wewnętrzne	Objaśnienie
1		_TEXT	Tekst do wygrawerowania (maksymalnie 100 znaków)
2	RP	_RTP	Płaszczyzna wycofania (abs)
3	Z0	_RFP	Punkt odniesienia osi narzędzia (abs)
4	SC	_SDIS	Odstęp bezpieczeństwa (addytywnie do płaszczyzny odniesienia, wprowadzić bez znaku)
5	Z1	_DP	Głębokość (abs.), patrz _AMODE
6	Z1	_DPR	Głębokość (przyr.), patrz _AMODE
7	X0 R	_PA	Punkt odniesienia, 1. oś płaszczyzny (abs.) - w układzie prostokątnym, patrz _VARI
			Punkt odniesienia, długość (promień) - w układzie biegunowym, patrz _VARI
8	Y0 α0	_PO	Punkt odniesienia, 2. oś płaszczyzny (abs.) - w układzie prostokątnym, patrz _VARI
			Punkt odniesienia, kąt w odniesieniu do 1. osi - w układzie biegunowym, patrz _VARI
9	α1	_STA	Kierunek tekstu, kąt linii tekstu w odniesieniu do 1. osi), patrz _VARI
10	XM LM	_CP1	Punkt środkowy okręgu tekstu, 1. oś płaszczyzny (abs.) - w układzie prostokątnym, patrz _VARI
			Punkt środkowy okręgu tekstu, długość (promień) w odniesieniu do WNP - w układzie biegunowym, patrz _VARI
11	YM αM	_CP2	Punkt środkowy okręgu tekstu, 2. oś płaszczyzny (abs.) - w układzie prostokątnym, patrz _VARI
			Punkt środkowy okręgu tekstu, kąt w odniesieniu do 1. osi - w układzie biegunowym, patrz _VARI
12	W	_WID	Wysokość znaków (wprowadzić bez znaku)
13	DX1 DX2 α2	_DF	Odstęp znaków / szerokość całkowita, patrz _VARI
			Kąt rozwarcia, patrz _VARI
14	FZ	_FFD	Posuw dla dosuwu na głębokość, patrz _DMODE
15	F	_FFP1	Posuw dla obróbki płaszczyzny

Nr	Okno parametrów	Parametry wewnętrzne	Objaśnienie
16		_VARI	<p>Obróbka (orientacja i punkt odniesienia grawerowanego tekstu))</p> <p>JEDNOSTKI: Punkt odniesienia</p> <p>0: prostokątny 1: biegunowy</p> <p>DZIESIĄTKI: Zorientowanie tekstu</p> <p>0: Tekst na linii 1: Tekst na łuku koła u góry 2: Tekst na łuku koła u dołu</p> <p>SETKI: Zarezerwowano</p> <p>TYSIĄCE: Punkt odniesienia tekstu poziomo</p> <p>0: z lewej 1: współśrodkowo 2: z prawej</p> <p>DZIESIĄTKI TYSIĘCY: Punkt odniesienia tekstu pionowo</p> <p>0: z dołu 1: współśrodkowo 2: u góry</p> <p>SETKI TYSIĘCY: Długość tekstu</p> <p>0: Odstęp znaków 1: Całkowita szerokość tekstu (tylko w przypadku tekstu liniowego) 2: Kąt rozwarcia (tylko w przypadku tekstu na łuku koła)</p> <p>MILION: Punkt środkowy okręgu</p> <p>0: prostokątny (kartezjański) 1: biegunowy</p> <p>DZIESIĄTKI MILIONÓW: Pismo w lustrzanym odbiciu</p> <p>0 = kompatybilność 1 = pismo w lustrzanym odbiciu WŁ. 2 = pismo w lustrzanym odbiciu WYŁ.</p>
17		_CODEP	Numer strony kodowej pisma (obecnie tylko 1252)
18		_UMODE	Zarezerwowano
19		_GMODE	<p>Tryb ewaluacji zaprogramowanych wartości geometrycznych</p> <p>JEDNOSTKI: Zarezerwowano</p> <p>DZIESIĄTKI: Zarezerwowano</p> <p>SETKI: Wybór obróbka / tylko obliczenie punktu startowego</p> <p>0 = tryb kompatybilności 1 = normalna obróbka</p>

Nr	Okno parametrów	Parametry wewnętrzne	Objaśnienie
20		<code>_DMODE</code>	Tryb wyświetlania <hr/> JEDNOSTKI: Płaszczyzna obróbki G17/18/19 <hr/> 0 = kompatybilność, aktywna pozostaje płaszczyzna działająca przed wywołaniem cyklu 1 = G17 2 = G18 3 = G19 <hr/> DZIESIĄTKI: Rodzaj posuwu: Grupa G (G94/G95) dla posuwu dla powierzchni i na głębokość <hr/> 0 = tryb kompatybilności 1 = G-Code jak przed wywołaniem cyklu. G94/G95 dla posuwu dla powierzchni i na głębokość takie same
21		<code>_AMODE</code>	Tryb alternatywny <hr/> JEDNOSTKI: Głębokość końcowa (<code>_DP</code> , <code>_DPR</code>) <hr/> 0 = kompatybilność 1 = przyrostowo (<code>_DPR</code>) 2 = absolutnie (<code>_DP</code>)

18.1.26 Wywołanie konturu - CYCLE62

Programowanie

```
CYCLE62 (STRING [140] _KNAME, INT _TYPE, STRING [32] _LAB1, STRING [32]
_LAB2)
```

Parametry

Nr	Okno parametrów	Parametry wewnętrzne	Objaśnienie
1	PRG/ CON	<code>_KNAME</code>	Nazwa konturu albo podprogramu, nie musi być programowana w przypadku <code>_TYPE = 2</code>
2		<code>_TYPE</code>	Określenie wprowadzenia konturu 0 = podprogram 1 = nazwa konturu 2 = etykiety 3 = etykiety w podprogramie
3	LAB1	<code>_LAB1</code>	Etykieta 1, początek konturu
4	LAB2	<code>_LAB2</code>	Etykieta 2, koniec konturu

18.1.27 Frezowanie konturu - CYCLE72

Programowanie

```
CYCLE72 (STRING [141] _KNAME, REAL _RTP, REAL _RFP, REAL _SDIS, REAL
_DP, REAL _MID, REAL _FAL, REAL _FALD, REAL _FFP1, REAL _FFD, INT
_VARI, INT _RL, INT _AS1, REAL _LP1, REAL _FF3, INT _AS2, REAL
_LP2, INT _UMODE, REAL _FS, REAL _ZFS, INT _GMODE, INT _DMODE, INT
_AMODE)
```

Parametry

Nr	Okno param etrów	Parametry wewnętrzne	Objaśnienie
1		_KNAME	Nazwa podprogramu konturu
2	RP	_RTP	Płaszczyzna wycofania (abs)
3	Z0	_RFP	Punkt odniesienia osi narzędzia (abs)
4	SC	_SDIS	Odstęp bezpieczeństwa (addytywnie do punktu odniesienia, wprowadzić bez znaku)
5	Z1	_DP	Punkt końcowy, głębokość końcowa (abs./przr.) patrz _AMODE
6	DZ	_MID	Maksymalny dosuw na głębokość (przr., wprowadzić bez znaku)
7	UXY	_FAL	Naddatek w płaszczyźnie (przr.), naddatek na obrzeżu konturu
8	UZ	_FALD	Naddatek na głębokości (przr.), naddatek na dnie (wprowadzić bez znaku)
9	FX	_FFP1	Posuw po konturze
10	FZ	_FFD	Posuw dla dosuwu na głębokość (lub dosuwu przestrzennego)
11		_VARI	Rodzaj obróbki JEDNOSTKI: Obróbka 1 = obróbka zgrubna 2 = obróbka wykańczająca 5 = fazowanie DZIESIĄTKI: 0 = drogi pośrednie z G0 1 = drogi pośrednie z G1 SETKI: 0 = wycofanie na końcu konturu na punkt odniesienia 1 = wycofanie na końcu konturu na punkt odniesienia + _SDIS 2 = wycofanie na końcu konturu o _SDIS 3 = bez wycofania na końcu konturu, ruch do następnego punktu startowego następuje z posuwem po konturze TYSIĄCE: Zarezerwowano DZIESIĄTKI TYSIĘCY:

Nr	Okno parametrów	Parametry wewnętrzne	Objaśnienie
			0 = obróbka konturu w przód 1 = obróbka konturu w tył Ograniczenia przy obróbce w tył: <ul style="list-style-type: none"> • max 170 elementów konturu (łącznie z fazami i zaokrągleniami) • są poddawane ewaluacji tylko wartości w płaszczyźnie (X/Y) i F
12		_RL	Kierunek obróbki 40 = na środku konturu (G40, dosunięcie i odsunięcie: prosta albo prostopadłe) 41 = na lewo od konturu (G41, dosunięcie i odsunięcie: Prosta albo okrąg) 42 = na prawood konturu (G42, dosunięcie i odsunięcie: Prosta albo okrąg)
13		_AS1	Ruch dosunięcia do konturu JEDNOSTKI: 1 = prosta 2 = ćwierćokrąg 3 = półokrąg 4 = dosunięcie i odsunięcie prostopadłe DZIESIĄTKI: 0 = ostatni ruch, w płaszczyźnie 1 = ostatni ruch, przestrzennie
14	L1	_LP1	Droga dosunięcia, albo promień dosunięcia (przyr., wprowadzić bez znaku)
15	FZ	_FF3	Posuw dla dróg pośrednich (G94/G95 jak na konturze)
16		_AS2	Ruch odsunięcia od konturu (nie przy prostopadłym dosunięciu/odsunięciu) JEDNOSTKI: 1 = prosta 2 = ćwierćokrąg 3 = półokrąg DZIESIĄTKI: 0 = ostatni ruch, w płaszczyźnie 1 = ostatni ruch, przestrzennie
17	L2	_LP2	Droga odsunięcia, albo promień odsunięcia (przyr., wprowadzić bez znaku)
18		_UMODE	Zarezerwowano
19	FS	_FS	Szerokość fazy (przyr.)
20	ZFS	_ZFS	Głębokość zagłębienia (wierzchołka narzędzia) przy fazowaniu (abs./przyr.), patrz _AMODE
21		_GMODE	Tryb ewaluacji zaprogramowanych wartości geometrycznych JEDNOSTKI: Zarezerwowano DZIESIĄTKI: Zarezerwowano SETKI: Wybór obróbka/tylko obliczenie punktu startowego 0 = tryb kompatybilności 1 = normalna obróbka

Nr	Okno parametrów	Parametry wewnętrzne	Objaśnienie
22		<code>_DMODE</code>	<p>Tryb wyświetlania</p> <hr/> <p>JEDNOSTKI: Płaszczyzna obróbki G17/G18/G19</p> <p>0 = kompatybilność, aktywna pozostaje płaszczyzna działająca przed wywołaniem cyklu 1 = G17 (aktywna tylko w cyklu) 2 = G18 (aktywna tylko w cyklu) 3 = G19 (aktywna tylko w cyklu)</p> <hr/> <p>DZIESIĄTKI: Rodzaj posuwu: Grupa G (G94/G95) dla posuwu dla powierzchni i na głębokość</p> <p>0 = tryb kompatybilności 1 = G-Code jak przed wywołaniem cyklu. G94/G95 dla posuwu dla powierzchni i na głębokość takie same</p> <hr/> <p>TYSIĄCE:</p> <p>0 = tryb kompatybilności: Nazwa konturu znajduje się w <code>_KNAME</code> 1 = nazwa konturu jest programowana w <code>CYCLE62</code> i przekazywana w <code>_SC_CONT_NAME</code></p>
23		<code>_AMODE</code>	<p>Tryb alternatywny</p> <hr/> <p>JEDNOSTKI: punkt końcowy Z1 (<code>_DP</code>)</p> <p>0 = absolutnie (tryb kompatybilności) 1 = przyrostowo</p> <hr/> <p>DZIESIĄTKI: Jednostka dla dosuwu w płaszczyźnie</p> <p>0 = mm, cale 1 = zarezerwowano</p> <hr/> <p>SETKI: Głębokość zagłębienia przy fazowaniu (<code>_ZFS</code>)</p> <p>0 = absolutnie 1 = przyrostowo</p>

Wskazówka

Gdy przekazywane parametry poniżej są programowane pośrednio (jako parametry), okno wprowadzania nie jest poddawane przetwarzaniu wstecznemu:

`_VARI, _RL, _AS1, _AS2, _UMODE, _GMODE, _DMODE, _AMODE`

18.1.28 Wiercenie wstępne kieszeni konturowej - CYCLE64

Programowanie

```
CYCLE64 (STRING [100] _PRG, INT _VARI, REAL _RP, REAL _Z0, REAL _SC,
REAL _Z1, REAL _F, REAL _DXY, REAL _UXY, REAL _UZ, INT _CDIR,
STRING [20] _TR, INT _DR, INT _UMODE, INT _GMODE, INT _DMODE, INT
_AMODE)
```

Parametry

Nr	Okno param etrów	Parametry wewnętrzne	Objaśnienie
1	PRG	_PRG	Nazwa programu wiercenia/nawiercania
2		_VARI	Rodzaj obróbki JEDNOSTKI: Zarezerwowano DZIESIĄTKI: Zarezerwowano SETKI: Zarezerwowano TYSIĄCE: Tryb wycofania 0 = wycofanie na płaszczyznę wycofania 1 = wycofanie na punkt odniesienia + odstęp bezpieczeństwa
3	RP	_RP	Płaszczyzna wycofania (abs)
4	Z0	_Z0	Punkt odniesienia (abs)
5	SC	_SC	Odstęp bezpieczeństwa (addytywnie do punktu odniesienia, wprowadzić bez znaku)
6	Z1	_Z1	Głębokość wiercenia/nawiercania, (patrz _AMODE JEDNOSTKI)
7	F	_F	Posuw wiercenia/nawiercania
8	DXY	_DXY	Dosuw w płaszczyźnie - jednostka, (patrz AMODE DZIESIĄTKI)
9	UXY	_UXY	Naddatek w płaszczyźnie
10	UZ	_UZ	Naddatek na głębokości
11		_CDIR	Kierunek frezowania 0 = frezowanie współbieżne 1 = frezowanie przeciwbieżne
12	TR	_TR	Nazwa narzędzia odniesienia
13	DR	_DR	Numer D narzędzia odniesienia
14		_UMODE	Zarezerwowano
15		_GMODE	Tryb ewaluacji zaprogramowanych wartości geometrycznych JEDNOSTKI: Zarezerwowano DZIESIĄTKI: Zarezerwowano SETKI: Wybór obróbka/tylko obliczenie punktu startowego 0 = obróbka normalna (tryb kompatybilności nie jest wymagany) 1 = normalna obróbka 2 = zarezerwowano

Nr	Okno parametrów	Parametry wewnętrzne	Objaśnienie
25		_DMODE	<p>Tryb wyświetlania</p> <p>JEDNOSTKI: Płaszczyzna obróbki G17/18/19</p> <p>0 = kompatybilność, aktywna pozostaje płaszczyzna działająca przed wywołaniem cyklu</p> <p>1 = G17 (aktywna tylko w cyklu)</p> <p>2 = G18 (aktywna tylko w cyklu)</p> <p>3 = G19 (aktywna tylko w cyklu)</p> <p>DZIESIĄTKI: tryb technologii)</p> <p>1 = wiercenie wstępne</p> <p>2 = nawiercanie</p>
26		_AMODE	<p>Tryb alternatywny</p> <p>JEDNOSTKI: Głębokość wiercenia/nawiercania Z1</p> <p>0 = absolutnie (tryb kompatybilności)</p> <p>1 = przyrostowo</p> <p>DZIESIĄTKI: Jednostka dla dosuwu w płaszczyźnie (_DXY)</p> <p>0 = mm</p> <p>1 = % średnicy narzędzia</p>

18.1.29 Frezowanie kieszeni konturowej - CYCLE63

Programowanie

```
CYCLE63 (STRING[100] _PRG, INT _VARI, REAL _RP, REAL _Z0, REAL _SC,
REAL _Z1, REAL _F, REAL _FZ, REAL _DXY, REAL _DZ, REAL _UXY, REAL
_UZ, INT _CDIR, REAL _XS, REAL _YS, REAL _ER, REAL _EP, REAL _EW,
REAL _FS, REAL _ZFS, STRING[20] _TR, INT _DR, INT _UMODE, INT
_GMODE, INT _DMODE, INT _AMODE)
```

Parametry

Nr	Okno parametrów	Parametry wewnętrzne	Objaśnienie
1	PRG	_PRG	Nazwa programu wybierania materiału
2		_VARI	<p>Rodzaj obróbki</p> <p>JEDNOSTKI: Technologia obróbki</p> <p>1 = obróbka zgrubna</p> <p>3 = obróbka wykańczająca dna</p> <p>4 = obróbka wykańczająca obrzeża</p> <p>5 = fazowanie</p> <p>DZIESIĄTKI: Rodzaj dosuwu</p>

Nr	Okno param etrów	Parametry wewnętrzne	Objaśnienie
			0 = zagłębienie w środku 1 = zagłębienie po linii spiralnej 2 = zagłębienie ruchem wahadłowym
			SETKI: Zarezerwowano
			TYSIĄCE: Tryb wycofania 0 = wycofanie na płaszczyznę wycofania 1 = wycofanie na punkt odniesienia + odstęp bezpieczeństwa
			DZIESIĄTKI TYSIĘCY: Punkt startowy przy obróbce zgrubnej i wykańczającej dna 0 = automatycznie 1 = ręcznie
3	RP	_RP	Płaszczyzna wycofania (abs)
4	Z0	_Z0	Punkt odniesienia osi narzędzia (abs)
5	SC	_SC	Odstęp bezpieczeństwa (addytywnie do punktu odniesienia, wprowadzić bez znaku)
6	Z1	_Z1	Głębokość końcowa, (patrz _AMODE JEDNOSTKI)
7	F	_F	Posuw w płaszczyźnie obróbka zgrubna/wykańczająca
8	FZ	_FZ	Posuw dla dosuwu na głębokość
9	DXY	_DXY	Dosuw w płaszczyźnie - jednostka, (patrz AMODE DZIESIĄTKI)
10	DZ	_DZ	Dosuw na głębokość
11	UXY	_UXY	Naddatek w płaszczyźnie
12	UZ	_UZ	Naddatek na głębokości
13		_CDIR	Kierunek frezowania 0 = frezowanie współbieżne 1 = frezowanie przeciwbieżne
14	XS	_XS	Punkt startowy X, absolutnie
15	YS	_YS	Punkt startowy Y, absolutnie
16	ER	_ER	Zagłębienie spiralne: Promień
17	EP	_EP	Zagłębienie spiralne: Skok
18	EW	_EW	Zagłębienie ruchem wahliwym: maksymalny kąt zagłębienia
19	FS	_FS	Szerokość fazy (przyr.) tylko przy fazowaniu
20	ZFS	_ZFS	Głębokość zagłębienia wierzchołka narzędzia przy fazowaniu, (patrz AMODE SETKI)
21	TR	_TR	Nazwa narzędzia odniesienia przy obróbce pozostałego materiału
22	DR	_DR	Numer D narzędzia odniesienia przy obróbce pozostałego materiału
23		_UMODE	Zarezerwowano

Nr	Okno parametrów	Parametry wewnętrzne	Objaśnienie
24		_GMODE	<p>Tryb ewaluacji zaprogramowanych wartości geometrycznych</p> <hr/> <p>JEDNOSTKI: Zarezerwowano</p> <hr/> <p>DZIESIĄTKI: Zarezerwowano</p> <hr/> <p>SETKI: Wybór obróbka/tylko obliczenie punktu startowego</p> <hr/> <p>0 = obróbka normalna (tryb kompatybilności nie jest wymagany)</p> <p>1 = normalna obróbka</p> <p>2 = zarezerwowano</p>
25		_DMODE	<p>Tryb wyświetlania</p> <hr/> <p>JEDNOSTKI: Płaszczyzna obróbki G17/18/19</p> <hr/> <p>0 = kompatybilność, aktywna pozostaje płaszczyzna działająca przed wywołaniem cyklu</p> <p>1 = G17 (aktywna tylko w cyklu)</p> <p>2 = G18 (aktywna tylko w cyklu)</p> <p>3 = G19 (aktywna tylko w cyklu)</p> <hr/> <p>DZIESIĄTKI: Zarezerwowano</p> <hr/> <p>SETKI: Rodzaj technologii</p> <hr/> <p>1 = kieszeń</p> <p>2 = czop</p> <hr/> <p>TYSIĄCE: Obróbka pozostałego materiału</p> <hr/> <p>0 = nie</p> <p>1 = tak</p>
26		_AMODE	<p>Tryb alternatywny</p> <hr/> <p>JEDNOSTKI: Głębokość końcowa Z1</p> <hr/> <p>0 = absolutnie (tryb kompatybilności)</p> <p>1 = przyrostowo</p> <hr/> <p>DZIESIĄTKI: Jednostka dla dosuwu w płaszczyźnie (_DXY)</p> <hr/> <p>0 = mm</p> <p>1 = % średnicy narzędzia</p> <hr/> <p>SETKI: Głębokość zagłębienia przy fazowaniu (_ZFS)</p> <hr/> <p>0 = absolutnie</p> <p>1 = przyrostowo</p>

18.1.30 Skrawanie warstwowe - CYCLE951

Programowanie

```
CYCLE951 (REAL _SPD, REAL _SPL, REAL _EPD, REAL _EPL, REAL _ZPD, REAL
_ZPL, INT _LAGE, REAL _MID, REAL _FALX, REAL _FALZ, INT _VARI, REAL
_RF1, REAL _RF2, REAL _RF3, REAL _SDIS, REAL _FF1, INT _NR, INT
_DMODE, INT _AMODE)
```

Parametry

Nr	Okno parametrow	Parametr wewnętrzny	Objaśnienie
1	X0	_SPD	Punkt odniesienia (abs., zawsze średnica)
2	Z0	_SPL	Punkt odniesienia (abs)
3	X1	_EPD	Punkt końcowy
4	Z1	_EPL	Punkt końcowy
5	XM α1 α2	_ZPD	Punkt pośredni, patrz _DMODE (DZIESIĄTKI)
6	ZM α1 α2	_ZPL	Punkt pośredni, patrz _DMODE (DZIESIĄTKI)
7	Położenie	_POŁOŻENIE	Położenie obrabianego naroża 0 = zewnątrz/z tyłu 1 = zewnątrz/z przodu 2 = wewnątrz/z tyłu 3 = wewnątrz/z przodu
8	D	_MID	Maksymalny dosuw na głębokość przy zagłębieniu
9	UX	_FALX	Naddatek w X
10	UZ	_FALZ	Naddatek w Z
11		_VARI	Rodzaj obróbki JEDNOSTKI: Kierunek skrawania (wzdłużnie lub poprzecznie) w układzie współrzędnych 1 = wzdłużnie 2 = poprzecznie DZIESIĄTKI: 1 = obróbka zgrubna do naddatku na obróbkę wykańczającą 2 = obróbka wykańczająca SETKI: 0 = z dociąganiem po konturze, bez pozostających naroży 1 = bez dociągania po konturze TYSIĄCE: 0 = z promieniem/fazą na narożu 2 1 = z podcięciem na narożu 2

Nr	Okno paramet rów	Parametr wewnętrzne	Objaśnienie
			DZIESIĄTKI TYSIĘCY: 0 = zatrzymać się po obróbce 1 = ruch powrotny do pozycji startowej
12	R1/FS1	_RF1	Promień zaokrąglenia albo szerokość fazy 1, patrz _AMODE (DZIESIĄTKI TYSIĘCY)
13	R2/FS2	_RF2	Promień zaokrąglenia albo szerokość fazy 2, patrz _AMODE (SETKI TYSIĘCY)
14	R3/FS3	_RF3	Promień zaokrąglenia albo szerokość fazy 3, patrz _AMODE (JEDEN MILION)
15	SC	_SDIS	Odstęp bezpieczeństwa
16	F	_FF1	Posuw dla obróbki zgrubnej/wykańczającej
17		_NR	Oznaczenie rodzaju skrawania (odpowiada pionowym przyciskom programowym do wyboru kształtu): 0 = skrawanie 1, naroże 90 stopni bez faz/zaokrągleń 1 = skrawanie 2, naroże 90 stopni z fazami/zaokrągleniami 2 = skrawanie 3, dowolne naroże z fazami/zaokrągleniami
18		_DMODE	Tryb wyświetlania JEDNOSTKI: Płaszczyzna obróbki G17/G18/G19 0 = kompatybilność, aktywna pozostaje płaszczyna działająca przed wywołaniem cyklu 1 = G17 (aktywna tylko w cyklu) 2 = G18 (aktywna tylko w cyklu) 3 = G19 (aktywna tylko w cyklu) DZIESIĄTKI: Wprowadzany kształt _ZPD/_ZPL 0 = Xm/Zm 1 = Xm/α1 2 = Xm/α2 3 = α1/Zm 4 = α2/Zm 5 = α1/α2
21		_AMODE	Tryb alternatywny JEDNOSTKI: Punkt pośredni w X 0 = absolutnie, wartość osi poprzecznej na średnicy 1 = przyrostowo, wartość osi poprzecznej na promieniu DZIESIĄTKI: Punkt pośredni w Z 0 = absolutnie 1 = przyrostowo SETKI: Punkt końcowy w X 0 = absolutnie, wartość osi poprzecznej na średnicy 1 = przyrostowo, wartość osi poprzecznej na promieniu TYSIĄCE: Punkt końcowy w Z 0 = absolutnie 1 = przyrostowo

Nr	Okno paramet rów	Parametr wewnętrzne	Objaśnienie
			DZIESIĄTKI TYSIĘCY: Promień/faza 1
			0 = promień
			1 = faza
			SETKI TYSIĘCY: Promień/faza 2
			0 = promień
			1 = faza
			MILIONY: Promień/faza 3
			0 = promień
			1 = faza

18.1.31 Wytoczenie - CYCLE930

Programowanie

```
CYCLE930 (REAL _SPD, REAL _SPL, REAL _WIDG, REAL _WIDG2, REAL _DIAG,
REAL _DIAG2, REAL _STA, REAL _ANG1, REAL _ANG2, REAL _RCO1, REAL
_RCI1, REAL _RCI2, REAL _RCO2, REAL _FAL, REAL _IDEP1, REAL _SDIS,
INT _VARI, INT _DN, INT _NUM, REAL _DBH, REAL _FF1, INT _NR, REAL
_FALX, REAL _FALZ, INT _DMODE, INT _AMODE)
```

Parametry

Nr	Okno paramet rów	Parametr wewnętrzne	Objaśnienie
1	X0	_SPD	Punkt odniesienia w osi poprzecznej (zawsze średnica)
2	Z0	_SPL	Punkt odniesienia w osi podłużnej
3	B1	_WIDG	Szerokość rowka na dnie
4	B2	_WIDG2	Szerokość rowka u góry (tylko dla powierzchni)
5	T1	_DIAG	Głębokość rowka w punkcie odniesienia, przy abs. i obróbka wzdłużna = średnica, i innym przypadku przyr.
6	T2	_DIAG2	Głębokość zagłębienia w punkcie odniesienia (tylko dla powierzchni), przy abs. i obróbka wzdłużna = średnica, i innym przypadku przyr.
7	α_0	_STA	Kąt skosu ($-180 \leq _STA \leq 180$)
8	α_1	_ANG1	Kąt zbocza 1 ($0 \leq _ANG1 < 90$) na stronie rowka określonej przez punkt odniesienia
9	α_2	_ANG2	Kąt zbocza 2 ($0 \leq _ANG2 < 90$) po drugiej stronie w stosunku do punktu odniesienia
10	R1/FS1	_RCO1	Promień zaokrąglenia albo szerokość fazy 1, na zewnątrz na punkcie odniesienia
11	R2/FS2	_RCI1	Promień zaokrąglenia albo szerokość fazy 2, wewnątrz na punkcie odniesienia

18.1 Cykle technologiczne

Nr	Okno parametrow	Parametry wewnętrzne	Objaśnienie
12	R3/FS3	_RCI2	Promień zaokrąglenia albo szerokość fazy 3, wewnątrz po drugiej stronie w stosunku do punktu odniesienia
13	R4/FS4	_RCO2	Promień zaokrąglenia albo szerokość fazy 4, zewnątrz po drugiej stronie w stosunku do punktu odniesienia
14	U	_FAL	Naddatek w X i Z, patrz _VARI (DZIESIĄTKI TYSIĘCY) (wprowadzić bez znaku)
15	D	_IDEP1	Maksymalny dosuw na głębokość przy zagłębieniu (wprowadzić bez znaku) 0 = 1. Przejście narzędzia bezpośrednio na całej głębokości > 0 = 1. przejście _IDEP1, 2. przejście 2 · _IDEP1, itd.
16	SC	_SDIS	Odstęp bezpieczeństwa (wprowadzić bez znaku)
17		_VARI	Rodzaj obróbki <p>JEDNOSTKI: Zarezerwowano</p> <p>DZIESIĄTKI: Technologia obróbki</p> <p>1 = obróbka zgrubna 2 = obróbka wykańczająca 3 = obróbka zgrubna i wykańczająca</p> <p>SETKI: Położenie wzdłużne/poprzeczne zewnętrzne/wewnętrzne +Z/+Z wzgl. +X/-X</p> <p>1 = wzdłużne/zewnętrzne +Z 2 = poprzeczne/wewnętrzne -X 3 = wzdłużne/wewnętrzne +Z 4 = poprzeczne/wewnętrzne +X 5 = wzdłużne/zewnętrzne -Z 6 = poprzeczne/zewnętrzne -X 7 = wzdłużne/wewnętrzne -Z 8 = poprzeczne/zewnętrzne +X</p> <p>TYSIĄCE: Położenie punktu odniesienia</p> <p>0 = punkt odniesienia u góry 1 = punkt odniesienia u dołu</p> <p>DZIESIĄTKI TYSIĘCY: Określenie zachowania się naddatków na obróbkę wykańczającą</p> <p>0 = naddatek U równoległy do konturu 1 = naddatki UX i UZ oddzielnie</p>
18		_DN	Numer D dla 2. ostrza narzędzia > 0 = numer D dla korekcji narzędzia 2. ostrza noża do toczenia poprzecznego 0 = nie zaprogramowano 2. ostrza
19	N	_NUM	Liczba rowków (0 = 1 rowek)
20	DP	_DBH	Liczba rowków (wymagana tylko wtedy, gdy _NUM > 1)
21	F	_FF1	Posuw

Nr	Okno paramet rów	Parametr y wewnętrzne	Objaśnienie
22		_NR	Oznaczenie kształtu rowka odpowiada pionowym przyciskom programowym do wyboru kształtu 0 = zbocza 90 bez faz/zaokrągleń 1 = skośne zbocza z fazami/zaokrągleniami (bez $\alpha 0$) 2 = jak 1, ale na stożku (z $\alpha 0$)
23	UX	_FALX	Naddatek w osi X, patrz _VARI (DZIESIĄTKI TYSIĘCY) (wprowadzić bez znaku)
24	UZ	_FALZ	Naddatek w osi Z, patrz _VARI (DZIESIĄTKI TYSIĘCY) (wprowadzić bez znaku)
25		_DMODE	Tryb wyświetlania JEDNOSTKI: Płaszczyzna obróbki G17/G18/G19 0 = kompatybilność, aktywna pozostaje płaszczyzna działająca przed wywołaniem cyklu 1 = G17 (aktywna tylko w cyklu) 2 = G18 (aktywna tylko w cyklu) 3 = G19 (aktywna tylko w cyklu)
26		_AMODE	Tryb alternatywny JEDNOSTKI: Zwymiarowanie głębokości (tylko dla powierzchni) 0 = na punkcie odniesienia 1 = po stronie przeciwnej do punktu odniesienia DZIESIĄTKI: Głębokość 0 = absolutnie 1 = przyrostowo SETKI: Zwymiarowanie szerokości (tylko dla powierzchni) 0 = na średnicy zewnętrznej (u góry) 1 = na średnicy wewnętrznej (u dołu) TYSIĄCE: Zaokrąglenie/faza 1 (_RCO1) 0 = promień 1 = faza DZIESIĄTKI TYSIĘCY: Zaokrąglenie/faza 2 (_RCI1) 0 = promień 1 = faza SETKI TYSIĘCY: Zaokrąglenie/faza 3 (_RCI2) 0 = promień 1 = faza MILIONY: Zaokrąglenie/faza 4 (_RCO2) 0 = promień 1 = faza

18.1.32 Podcięcie kształtowe - CYCLE940

Przy pomocy cyklu CYCLE940 można programować różne podcięcia. Po części znacznie różnią się one pod względem parametryzacji.

Dodatkowe kolumny tabeli pokazują, które parametry są potrzebne przy danym kształcie podcięcia. Odpowiadają one pionowym przyciskom programowym wyboru w oknie cykli:

- E: Podcięcie kształt E
- F: Podcięcie kształt F
- A-D: Podcięcie gwintu DIN (kształty A-D)
- T: Podcięcie gwintu (dowolna definicja kształtu)

Programowanie

```
CYCLE940 (REAL _SPD, REAL _SPL, CHAR _FORM, INT _LAGE, REAL _SDIS,
REAL _FFP, INT _VARI, REAL _EPD, REAL _EPL, REAL _R1, REAL _R2, REAL
_STA, REAL _VRT, REAL _MID, REAL _FAL, REAL _FALX, REAL _FALZ, INT
_PITI, STRING[5] _PTAB, STRING[20] _PTABA, INT _DMODE, INT _AMODE)
```

Parametry

Nr	Okno parametrów	Parametry wewnętrzne	Programowanie w przypadku kształtu				Objaśnienie
			E	F	A-D	T	
1	X0	_SPD	x	x	x	x	Punkt odniesienia w osi poprzecznej (zawsze średnica)
2	Z0	_SPL	x	x	x	x	Punkt odniesienia osi wzdłużnej (abs.)
3	FORM	_FORM	x	x	x	x	Kształt podcięcia (duże litery, np. "T") Wybór, z której tablicy mają być brane podcięcia A = zewnątrz, odniesienie DIN76, A = normalne B = zewnątrz, odniesienie DIN76, B = skrócone C = wewnątrz, odniesienie DIN76, C = normalne D = wewnątrz, odniesienie DIN76, D = skrócone E = odniesienie DIN509 F = odniesienie DIN509 T = dowolny kształt
4	POŁOŻENIE	POŁOŻENIE	x	x	x	x	Położenie podcięcia (równoległe do Z) 0 = zewnętrzne +Z: ___ ___ 1 = zewnętrzne -Z: ___/ ___/ 2 = wewnętrzne +Z: /---- /---- 3 = wewnętrzne -Z: ----\ ----\
5	SC	_SDIS	x	x	x	x	Odstęp bezpieczeństwa (przyr.)
6	F	_FFP	x	x	x	x	Posuw dla obróbki (mm/obr.)

Programowanie w przypadku kształtu							
7		_VARI	-	-	x	x	Rodzaj obróbki
JEDNOSTKI: Obróbka							
1 = obróbka zgrubna							
2 = obróbka wykańczająca							
3 = obróbka zgrubna + wykańczająca							
DZIESIĄTKI: Strategia obróbki							
0 o równoległe do konturu							
1 = wzdłużnie							
Podcięcia kształt E i F są zawsze obrabiane w jednym przejściu, jak przy obróbce wykańczającej.							
8	X1	_EPD	x	x	-	-	Naddatek X (abs./przr.), patrz _AMODE)
- - - x Głębokość podcięcia (abs./przr.) patrz _AMODE							
9	Z1	_EPL	-	x	-	-	Naddatek Z
- - - x Szerokość podcięcia (abs./przr.) patrz _AMODE							
10	R1	_R1	-	-	-	x	Promień zaokrąglenia na skosach
11	R2	_R2	-	-	-	x	Promień zaokrąglenia w narożu
12	α	_STA	-	-	x	x	Kąt zagłębiania
13	VX	_VRT	x	x	-	-	Podcięcie X (abs./przr.), patrz _AMODE
- - x x Podcięcie X przy obróbce wykańczającej (abs./przr.), patrz _AMODE							
14	D	_MID	-	-	x	x	Dosuw na głębokość
15	U	_FAL	-	-	x	x	Naddatek równoległy do konturu, patrz _AMODE
16	UX	_FALX	-	-	x	x	Naddatek na obróbkę wykańczającą X
17	UZ	_FALZ	-	-	x	x	Naddatek na obróbkę wykańczającą Z
18	P	_PITI	-	-	x	-	Wybór skoku, kształt A-D, odpowiada M1 ... M68
0 = 0.20 6 = 0.50 12 = 1.25 18 = 3.50							
1 = 0.25 7 = 0.60 13 = 1.50 19 = 4.00							
2 = 0.30 8 = 0.70 14 = 1.75 20 = 4.50							
3 = 0.35 9 = 0.75 15 = 2.00 21 = 5.00							
4 = 0.40 10 = 0.80 16 = 2.50 22 = 5.50							
5 = 0.45 11 = 1.00 17 = 3.00 23 = 6.00							
x x - - Wybór promień/głębokość, kształt E, F							
0 = 0.6 · 0.3 4 = 2.5 · 0.4 8 = 0.1 · 0.1							
1 = 1.0 · 0.4 5 = 4.0 · 0.5 9 = 0.2 · 0.1							
2 = 1.0 · 0.2 6 = 0.4 · 0.2							
3 = 1.6 · 0.3 7 = 0.6 · 0.2							
19		_PTAB	Łańcuch znaków dla tablicy gwintów ("", "ISO", "BSW", "BSP", "UNC")				
(tylko dla powierzchni)							
20		_PTABA	Łańcuch znaków dla wyboru w tablicy gwintów (np. "M 10", "M 12", ...)				
(tylko dla powierzchni)							

				Programowanie w przypadku kształtu		
21	_DMODE			Tryb wyświetlania		
		x	x	x	JEDNOSTKI: Płaszczyzna obróbki G17/18/19	
					0 = kompatybilność, aktywna pozostaje płaszczyzna działająca przed wywołaniem cyklu 1 = G17 (aktywna tylko w cyklu) 2 = G18 (aktywna tylko w cyklu) 3 = G19 (aktywna tylko w cyklu)	
22	_AMODE			Tryb alternatywny		
		x	x	-	x	JEDNOSTKI: Parametr _EPD naddatek X albo głębokość podcięcia
						0 = absolutnie (zawsze średnica) 1 = przyrostowo
		x	x	-	x	DZIESIĄTKI: Parametr _EPL naddatek Z albo szerokość podcięcia
						0 = absolutnie 1 = przyrostowo
		x	x	x	x	SETKI: Parametr _VRT podcięcie X
						0 = absolutnie (zawsze średnica) 1 = przyrostowo
		-	-	x	x	TYSIĄCE: Naddatek na obróbkę wykańczającą
						0 = naddatek równoległy do konturu (_FAL) = naddatek oddzielnie (_FALX/_FALZ)

18.1.33 Toczenie gwintu - CYCLE99

Programowanie

```
CYCLE99(REAL _SPL, REAL _SPD, REAL _FPL, REAL _FPD, REAL _APP, REAL
_ROP, REAL _TDEP, REAL _FAL, REAL _IANG, REAL _NSP, INT _NRC, INT
_NID, REAL _PIT, INT _VARI, INT _NUMTH, REAL _SDIS, REAL _MID, REAL
_GDEP, REAL _PIT1, REAL _FDEP, INT _GST, INT _GUD, REAL _IFLANK, INT
_PITA, STRING[15] _PITM, STRING[20] _PTAB, STRING[20] _PTABA, INT
_DMODE, INT _AMODE)
```

Parametry

Nr	Okno paramet rów	Parametry wewnętrzne	Objaśnienie
1	Z0	_SPL	Punkt odniesienia (abs)
2	X0	_SPD	Punkt odniesienia (abs., zawsze średnica)
3	Z1	_FPL	Punkt końcowy w połączeniu z _AMODE (JEDNOSTKI)
4	X1	_FPD	Punkt końcowy w połączeniu z _AMODE (DZIESIĄTKI)

Nr	Okno paramet rów	Parametry wewnętrzne	Objaśnienie
5	LW/LW2	_APP	Dobieg gwintu w połączeniu z _AMODE (SETKI) lub Dobieg gwintu = wybieg gwintu w połączeniu z _AMODE (SETKI)
6	LR	_ROP	Wybieg gwintu
7	H1	_TDEP	Głębokość gwintu
8	U	_FAL	Naddatek w X i Z
9	DP αP	_IANG	Dosuw po skosie jako odstęp albo kąt, w połączeniu z _AMODE (TYSIĄCE) > 0 = dosuw po zboczu dodatnim < 0 = dosuw po zboczu ujemnym 0 = dosuw po środku
10	$\alpha 0$	_NSP	Przemieszczenie kąta startowego (działa tylko przy 1. zwoju)
11	ND	_NRC	Liczba przejść narzędzia dla obróbki zgrubnej, w połączeniu z _VARI (DZIESIĄTKI TYSIĘCY)
12	NN	_NID	Liczba przejść jałowych narzędzia
13	P	_PIT	Skok gwintu jako wartość, w połączeniu z _PITA
14		_VARI	Rodzaj obróbki JEDNOSTKI: Technologia 1 = gwint zewnętrzny z stałym dosuwem 2 = gwint wewnętrzny z stałym dosuwem 3 = gwint zewnętrzny z dosuwem degresywnym, przekrój skrawania pozostaje stały 4 = gwint wewnętrzny z dosuwem degresywnym, przekrój skrawania pozostaje stały DZIESIĄTKI: Zarezerwowano SETKI: Rodzaj dosuwu 1 = dosuw jednostronny 2 = dosuw naprzemienny TYSIĄCE: Zarezerwowano DZIESIĄTKI TYSIĘCY: Alternatywny dosuw na głębokość 0 = zadanie liczby przejść zgrubnych (_NRC) 1 = zadanie wartości dla 1. dosuwu (_MID) SETKI TYSIĘCY: Rodzaj obróbki 1 = obróbka zgrubna 2 = obróbka wykańczająca 3 = obróbka zgrubna i wykańczająca MILIONY: Kolejność obróbki przy gwincie wielokrotnym 0 = kolejność zwojów rosnąco 1 = kolejność zwojów naprzeciwno
15	N	_NUMTH	Liczba zwojów gwintu
16	VR	_SDIS	Odstęp ruchu powrotnego, przyr.
17	D1	_MID	Pierwsza głębokość dosuwu, w połączeniu z _VARI (DZIESIĄTKI TYSIĘCY)

Nr	Okno paramet rów	Parametry wewnętrzne	Objaśnienie
18	DA	_GDEP	Zmiana głębokości zwoju (działa tylko przy "wielokrotny") 0 = nie uwzględniać zmiany głębokości zwoju > 0 = uwzględniać zmianę głębokości zwoju
19	G	_PIT1	Zmiana skoku na obrót 0 = skok gwintu jest stały (G33) > 0 = skok gwintu staje się większy (G34) < 0 = skok gwintu staje się mniejszy (G35)
20		_FDEP	Wartość zagłębienia (wprowadzić bez znaku)
21	N1	_GST	Zwój startowy N1 = 1...N, w połączeniu z _AMODE (SETKI TYSIĘCY)
22		_GUD	Zarezerwowano
23		_IFLANK	Dosuw po skosie jako szerokość (tylko dla powierzchni)
24		_PITA	Jednostka miary skoku gwintu (oszacowanie PIT i/albo MPIT) 0 = skok w mm - oszacowanie MPIT/PIT 1 = skok w mm - oszacowanie PIT 2 = skok w TPI - oszacowanie PIT (zwojów gwintu na cal) 3 = skok w calach - oszacowanie PIT 4 = MODUŁ - oszacowanie PIT
25		_PITM	Łańcuch znaków jako znacznik dla wprowadzenia skoku gwintu (tylko dla powierzchni) ¹⁾
26		_PTAB	Łańcuch znaków dla tablicy gwintów (tylko dla powierzchni) ¹⁾
27		_PTABA	Łańcuch znaków dla wyboru w tablicy gwintów (tylko dla powierzchni) ¹⁾
28		_DMODE	Tryb wyświetlania JEDNOSTKI: Płaszczyzna obróbki G17/G18/G19 0 = kompatybilność, aktywna pozostaje płaszczyzna działająca przed wywołaniem cyklu 1 = G17 (aktywna tylko w cyklu) 2 = G18 (aktywna tylko w cyklu) 3 = G19 (aktywna tylko w cyklu) DZIESIĄTKI: Rodzaj gwintu 0 = gwint podłużny 1 = gwint poprzeczny 2 = gwint stożkowy
29		_AMODE	Tryb alternatywny JEDNOSTKI: Długość gwintu w Z 0 = absolutnie 1 = przyrostowo DZIESIĄTKI: Długość gwintu w X 0 = absolutnie, wartość osi poprzecznej na średnicy 1 = przyrostowo, wartość osi poprzecznej na promieniu 2 = α SETKI: Oszacowanie drogi wejścia/dobiegu _APP

Nr	Okno parametrów	Parametry wewnętrzne	Objaśnienie
			0 = dobieg gwintu _APP 1 = dobieg gwintu = wybieg gwintu _APP = -_ROP 2 = zadanie drogi dobiegu gwintu _APP = -_APP
			TYSIĄCE: Wybór dosuwu po skosie jako kąt albo szerokość
			0 = kąt dosuwu _IANG 1 = dosuw po skosie _IFLANK
			DZIESIĄTKI TYSIĘCY: Jednokrotny/wielokrotny
			0 = jednokrotny (z przesunięciem kąta startowego _NSP) 1 = wielokrotny
			SETKI TYSIĘCY zwój startowy _GST
			0 = obróbka kompletna 1 = rozpocząć obróbkę od tego zwoju 2 = obrobić tylko ten zwój

Wskazówka

¹⁾ Parametry _PITM, _PTAB i _PTABA są stosowane tylko przy wyborze gwintu w tablicach gwintów okna wprowadzania.
Dostęp do tablic gwintów poprzez definicję cyklu w czasie wykonywania cyklu jest niemożliwy.

18.1.34 Łańcuch gwintów - CYCLE98**Programowanie**

```
CYCLE98 (REAL _PO1, REAL _DM1, REAL _PO2, REAL _DM2, REAL _PO3, REAL
_DM3, REAL _PO4, REAL _DM4, REAL APP, REAL ROP, REAL TDEP, REAL FAL,
REAL _IANG, REAL NSP, INT NRC, INT NID, REAL _PP1, REAL _PP2, REAL
_PP3, INT _VARI, INT _NUMTH, REAL _VRT, REAL _MID, REAL _GDEP, REAL
_IFLANK, INT _PITA, STRING[15] _PITM1, STRING[15] _PITM2, STRING[15]
_PITM3, INT _DMODE, INT _AMODE)
```

Parametry

Nr	Okno parametrów	Parametry wewnętrzne	Objaśnienie
1	Z0	_PO1	Punkt odniesienia w Z (abs.)
2	X1	_DM1	Punkt odniesienia w X (abs.), na średnicy

Nr	Okno paramet rów	Parametr wewnętrzne	Objaśnienie
3	Z1	_PO2	Punkt pośredni 1 w Z (abs./przyr.), patrz _AMODE (JEDNOSTKI)
4	X1 X1 α	_DM2	Punkt pośredni 1 w X (abs./przyr.), patrz _AMODE (DZIESIĄTKI) albo Skos gwintu 1 (-90° do 90°) abs. zawsze średnica, przyr. zawsze promień
5	Z2	_PO3	Punkt pośredni 2 w Z (abs./przyr.), patrz _AMODE (SETKI)
6	X2 X2 α	_DM3	Punkt pośredni 2 w X (abs./przyr.), patrz _AMODE (TYSIĄCE) albo Skos gwintu 2 (-90° do 90°) abs. zawsze średnica, przyr. zawsze promień
7	Z3	_PO4	Punkt końcowy w Z (abs./przyr.), patrz _AMODE (DZIESIĄTKI TYSIĘCY)
8	X3 X3 α	_DM4	Punkt końcowy w X (abs./przyr.), patrz _AMODE (SETKI TYSIĘCY) albo Skos gwintu 3 (-90° do 90°) abs. zawsze średnica, przyr. zawsze promień
9	LW	APP	Dobieg gwintu (przyr., wprowadzić bez znaku)
10	LR	ROP	Dobieg gwintu (przyr., wprowadzić bez znaku)
11	H1	TDEP	Głębokość gwintu (przyr., wprowadzić bez znaku)
12	U	FAL	Naddatek w X i Z
13	DP α P	_IANG	Dosuw po skosie jako odstęp albo kąt, patrz _AMODE (MILION) Dosuw po skosie działa odpowiednio do ustawienia parametru _VARI (SETKI). Definicja trybu kompatybilności _VARI_SETKI = 0: > > 0 = dosuw po jednym zboczach 0 = dostęp prostopadle do gwintu < 0 = dosuw po zboczach naprzemiennie Definicja dla _VARI_SETKI <> 0: > > 0 = dosuw po zboczach dodatnim 0 = dosuw po środku < 0 = dosuw po zboczach ujemnym
14	α 0	NSP	Przesunięcie kąta startowego dla 1. zwoju gwintu
15		NRC	Liczba zgrubnych przejść narzędzia, patrz _VARI (DZIESIĄTKI TYSIĘCY)
16	NN	NID	Liczba przejść jałowych narzędzia
17	P0	_PP1	Skok 1. fragmentu gwintu, patrz _PITA
18	P1	_PP2	Skok 2. fragmentu gwintu, patrz _PITA
19	P2	_PP3	Skok 3. fragmentu gwintu, patrz _PITA
20		_VARI	Obróbka JEDNOSTKI: Technologia 1 = gwint zewnętrzny z stałym dosuwem 2 = gwint wewnętrzny z stałym dosuwem 3 = gwint zewnętrzny z dosuwem degresywnym, przekrój skrawania pozostaje stały 4 = gwint wewnętrzny z dosuwem degresywnym, przekrój skrawania pozostaje stały DZIESIĄTKI: Zarezerwowano

Nr	Okno paramet rów	Parametr wewnętrzne	Objaśnienie
			SETKI: Rodzaj dosuwu
			0 = tryb kompatybilności dla _IANG
			1 = dosuw jednostronny
			2 = dosuw naprzemienny
			TYSIĄCE: Zarezerwowano
			DZIESIĄTKI TYSIĘCY: Alternatywny dosuw na głębokość
			0 = kompatybilność, zadanie liczby przejść zgrubnych (_NRC)
			1 = zadanie wartości dla 1. dosuwu (_MID)
			SETKI TYSIĘCY: Rodzaj obróbki
			0 = kompatybilność (obróbka zgrubna i wykańczająca)
			1 = obróbka zgrubna
			2 = obróbka wykańczająca
			3 = obróbka zgrubna i wykańczająca
			MILIONY: Kolejność obróbki przy gwincie wielokrotnym
			0 = kolejność zwojów rosnąco
			1 = kolejność zwojów naprzeciwko
21	N	_NUMTH	Liczba zwojów gwintu
22		_VRT	Odstęp ruchu wstecznego (ink) 0 = wewnątrz jest stosowana droga wycofania 1 mm niezależnie od systemu miar calowego/metrycznego > 0 = droga wycofania
23	D1	_MID	pierwszy dosuw, patrz _VARI (DZIESIĄTKI TYSIĘCY)
24	DA	_GDEP	Zmiana głębokości zwoju (działa tylko przy "wielokrotny") 0 = nie uwzględniać zmiany głębokości zwoju > 0 = uwzględniać zmianę głębokości zwoju
25		_IFLANK	Dosuw po skosie jako szerokość (tylko dla powierzchni)
26		_PITA	Oszacowanie skoku gwintu 0 = tryb kompatybilności dla skoku gwintu: Oszacowanie _PP1 do _PP3 jak dotychczas odpowiednio do aktywnego systemu metrycznego/calowego 1 = skok w mm 2 = skok w TPI (zwojów gwintu na cal) 3 = skok w calach 4 = MODUŁ
27		_PITM1	Łańcuch znaków jako znacznik dla wprowadzenia skoku gwintu (tylko dla powierzchni)
28		_PITM2	Łańcuch znaków jako znacznik dla wprowadzenia skoku gwintu (tylko dla powierzchni)
29		_PITM3	Łańcuch znaków jako znacznik dla wprowadzenia skoku gwintu (tylko dla powierzchni)

Nr	Okno paramet rów	Parametr wewnętrzne	Objaśnienie
30		_DMODE	<p>Tryb wyświetlania</p> <hr/> <p>JEDNOSTKI: Płaszczyzna obróbki G17/18/19</p> <p>0 = kompatybilność, aktywna pozostaje płaszczyzna działająca przed wywołaniem cyklu</p> <p>1 = G17 (aktywna tylko w cyklu)</p> <p>2 = G18 (aktywna tylko w cyklu)</p> <p>3 = G19 (aktywna tylko w cyklu)</p>
31		_AMODE	<p>Tryb alternatywny</p> <hr/> <p>JEDNOSTKI: 1. Punkt pośredni w Z (Z1)</p> <p>0 = absolutnie</p> <p>1 = przyrostowo</p> <hr/> <p>DZIESIĄTKI: 1. Punkt pośredni w X (X1)</p> <p>0 = absolutnie</p> <p>1 = przyrostowo</p> <p>2 = α</p> <hr/> <p>SETKI: 2. Punkt pośredni w Z (Z2)</p> <p>0 = absolutnie</p> <p>1 = przyrostowo</p> <hr/> <p>TYSIĄCE: 2. Punkt pośredni w X (X2)</p> <p>0 = absolutnie</p> <p>1 = przyrostowo</p> <p>2 = α</p> <hr/> <p>DZIESIĄTKI TYSIĘCY: Punkt końcowy w Z (Z3)</p> <p>0 = absolutnie</p> <p>1 = przyrostowo</p> <hr/> <p>SETKI TYSIĘCY: Punkt końcowy w X (X3)</p> <p>0 = absolutnie</p> <p>1 = przyrostowo</p> <p>2 = α</p> <hr/> <p>MILIONY: Wybór dosuwu po skosie jako kąt albo szerokość</p> <p>0 = kąt dosuwu _IANG</p> <p>1 = dosuw po skosie _IFLANK</p> <hr/> <p>DZIESIĄTKI MILIONÓW: Jednokrotny/wielokrotny</p> <p>0 = tryb kompatybilności (kąt startowy _NSP jest poddawany oszacowaniu)</p> <p>1 = jednokrotny (z przesunięciem kąta startowego _NSP)</p> <p>2 = wielokrotny</p>

18.1.35 Przecinanie - CYCLE92

Programowanie

```
CYCLE92 (REAL _SPD, REAL _SPL, REAL _DIAG1, REAL _DIAG2, REAL _RC,
REAL _SDIS, REAL _SV1, REAL _SV2, INT _SDAC, REAL _FF1, REAL _FF2,
REAL _SS2, REAL _DIAGM, INT _VARI, INT _DN, INT _DMODE, INT _AMODE)
```

Parametry

Nr	Okno paramet rów	Parametr wewnętrzne	Objaśnienie
1	X0	_SPD	Punkt odniesienia (abs., zawsze średnica)
2	Y0	_SPL	Punkt odniesienia (abs)
3	X1	_DIAG1	Głębokość dla zmniejszenia prędkości obrotowej, patrz _AMODE (JEDNOSTKI)
4	X2	_DIAG2	Głębokość końcowa, patrz _AMODE (DZIESIĄTKI)
5	R/FS	_RC	Promień zaokrąglenia albo szerokość fazy, patrz _AMODE (TYSIĄCE)
6	SC	_SDIS	Odstęp bezpieczeństwa (addytywnie do punktu odniesienia, wprowadzić bez znaku)
7	S	_SV1	Stała prędkość skrawania, patrz _AMODE (DZIESIĄTKI TYSIĘCY)
	V		Stała prędkość skrawania
8	SV	_SV2	Maksymalna prędkość obrotowa przy stałej prędkości skrawania
9	DIR	_SDAC	Kierunek obrotów wrzeciona 3 = dla M3 4 = dla M4
10	F	_FF1	Posuw do głębokości zmniejszenia prędkości obrotowej
11	FR	_FF2	Zmniejszony posuw do głębokości końcowej
12	SR	_SS2	Zmniejszona prędkość obrotowa do głębokości końcowej
13	XM	_DIAGM	Głębokość wysunięcia chwytaka obrabianych przedmiotów (abs., zawsze średnica)
14		_VARI	Rodzaj obróbki JEDNOSTKI: Wycofanie 0 = wycofanie na _SPD+ _SDIS 1 = bez wycofania na końcu DZIESIĄTKI: Chwytek obrabianych przedmiotów 0 = nie, nie wykonywać rozkazu M 1 = tak, wywołanie wysunięcia szuflady CUST_TECHCYC(101), zamknięcia szuflady CUST_TECHCYC(102)
15		_DN	Numer D dla 2. ostrza, gdy nie zaprogramowano => D+1
20		_DMODE	Tryb wyświetlania JEDNOSTKI: Płaszczyzna obróbki G17/G18/G19 0 = kompatybilność, aktywna pozostaje płaszczyzna działająca przed wywołaniem cyklu 1 = G17 (aktywna tylko w cyklu) 2 = G18 (aktywna tylko w cyklu) 3 = G19 (aktywna tylko w cyklu)

Nr	Okno paramet rów	Parametr wewnętrzne	Objaśnienie
21		_AMODE	Tryb alternatywny
			JEDNOSTKI: Głębokość dla zmniejszenia prędkości obrotowej (_DIAG1)
			0 = absolutnie, wartość osi poprzecznej na średnicy
			1 = przyrostowo, wartość osi poprzecznej na promieniu
			DZIESIĄTKI: Głębokość końcowa (_DIAG2)
			0 = absolutnie, wartość osi poprzecznej na średnicy
			1 = przyrostowo, wartość osi poprzecznej na promieniu
			SETKI: Zarezerwowano
			TYSIĄCE: Zaokrąglenie/faza (_RC)
			0 = promień
			1 = faza
			DZIESIĄTKI TYSIĘCY: Prędkość obrotowa wrzeciona/prędkość skrawania (_SV1)
			0 = stała prędkość obrotowa wrzeciona
			1 = stała prędkość skrawania

18.1.36 Skrawanie konturu - CYCLE95

Programowanie

```
CYCLE95 (STRING[140] NPP, REAL MID, REAL FALZ, REAL FALX, REAL FAL,
REAL FF1, REAL FF2, REAL FF3, INT VARI, REAL DT, REAL DAM, REAL
_VRT, INT _GMODE, INT _DMODE)
```

Parametry

Nr	Okno param etrów	Parametry wewnętrzne	Objaśnienie
1	CON	NPP	Nazwa konturu
2	D	MID	maksymalny dosuw na głębokość przy obróbce zgrubnej, patrz _GMODE
3	UZ	FALZ	Naddatek w Z
4	UX	FALX	Naddatek w X
5	U	FAL	Naddatek równoległe do konturu (działa w obu osiach)
6	F	FF1	Posuw dla obróbki zgrubnej
7	FY	FF2	Posuw zagłębiania dla podcięcia
8	FS	FF3	Posuw obróbki wykańczającej

Nr	Okno param etrów	Parametry wewnętrzne	Objaśnienie
9		VARI	Rodzaj obróbki <hr/> JEDNOSTKI i DZIESIĄTKI: <hr/> 1 = obróbka zgrubna, wzdłużnie, zewnętrznie 2 = obróbka zgrubna, poprzecznie, zewnętrznie 3 = obróbka zgrubna, wzdłużnie, wewnętrznie 4 = obróbka zgrubna, poprzecznie, wewnętrznie 5 = obróbka wykańczająca, wzdłużnie, zewnętrznie 6 = obróbka wykańczająca, poprzecznie, zewnętrznie 7 = obróbka wykańczająca, wzdłużnie, wewnętrznie 8 = obróbka wykańczająca, poprzecznie, wewnętrznie 9 = obróbka kompletna, wzdłużnie, zewnętrznie 10 = obróbka kompletna, poprzecznie, zewnętrznie 11 = obróbka kompletna, wzdłużnie, wewnętrznie 12 = obróbka kompletna, poprzecznie, wewnętrznie <hr/> SETKI: <hr/> 0 = z dociąganiem po konturze, bez pozostających naroży 1 = bez dociągania po konturze 2 = Dociąganie tylko do poprzedniego punktu przecięcia, mogą powstawać nierówności
10	DT	DT	Czas oczekiwania przy przerywaniu posuwu
11	DI	DAM	Ostęp przerw posuwu
12	VRT	_VRT	Droga cofnięcia od konturu 0 = wewnątrznie jest stosowana droga wycofania 1 mm niezależnie od systemu miar całowego/metrycznego > 0 = droga wycofania
13		_GMODE	Tryb geometrii (oszacowanie zaprogramowanych wartości geometrycznych) <hr/> JEDNOSTKI: Oszacowanie głębokości dosuwu <hr/> 0 = Głębokość dosuwu zostanie przeliczona odpowiednio do Grupy G DIAMON/DIAMOF 1 = Głębokość dosuwu działa jako wartość promienia (niezależnie od DIAMON/DIAMOF)
14		_DMODE	Tryb wyświetlania <hr/> JEDNOSTKI: Płaszczyzna obróbki G17/18/19 <hr/> 0 = kompatybilność, aktywna pozostaje płaszczyna działająca przed wywołaniem cyklu 1 = G17 (aktywna tylko w cyklu) 2 = G18 (aktywna tylko w cyklu) 3 = G19 (aktywna tylko w cyklu) <hr/> TYSIĄCE: <hr/> 0 = Tryb kompatybilności: Nazwa konturu znajduje się w NPP 1 = nazwa konturu jest programowana w CYCLE62 i przekazywana w _SC_CONT_NAME

18.1.37 Toczenie wcinające konturu - CYCLE952

Programowanie

```
CYCLE952 (STRING[100] _PRG, STRING[100] _CON, STRING[100] _CONR, INT
_VARI, REAL _F, REAL _FR, REAL _RP, REAL _D, REAL _DX, REAL _DZ,
REAL _UX, REAL _UZ, REAL _U, REAL _U1, INT _BL, REAL _XD, REAL _ZD,
REAL _XA, REAL _ZA, REAL _XB, REAL _ZB, REAL _XDA, REAL _XDB, INT
_N, REAL _DP, REAL _DI, REAL _SC, INT _DN, INT _GMODE, INT _DMODE,
INT _AMODE, INT _PK, REAL _DCH)
```

Parametry

Nr	Okno param etrów	Parametry wewnętrzne	Objaśnienie
1	PRG	_PRG	Nazwa programu skrawania warstwowego
2	CON	_CON	Nazwa programu z którego jest odczytywany aktualny kontur półfabrykatu (przy obróbce pozostałego materiału)
3	CONR	_CONR	Nazwa programu, w którym jest zapisywany aktualny kontur półfabrykatu (patrz _AMODE DZIESIĄTKI TYSIĘCY)
4		_VARI	Rodzaj obróbki JEDNOSTKI: Rodzaj skrawania 1 = wzdłużnie 2 = poprzecznie 3 = równoległe do konturu DZIESIĄTKI: Obróbka technologiczna (patrz _GMODE SETKI) 1 = obróbka zgrubna 2 = obróbka wykańczająca 3 = zarezerwowano 4 = obróbka zgrubna dwukanałowa 5 = obróbka wykańczająca dwukanałowa SETKI: Kierunek obróbki 1 = kierunek obróbki X - 2 = kierunek obróbki X + 3 = kierunek obróbki Z - 4 = kierunek obróbki Z + TYSIĄCE: Kierunek dosuwu 1 = zewnętrzny X- 2 = wewnętrzny X + 3 = strona czołowa Z - 4 = strona tylna Z + DZIESIĄTKI TYSIĘCY: Określenie zachowania się naddatków na obróbkę wykańczającą 0 = naddatki UX i UZ oddzielnie 1 = naddatek U równoległy do konturu

Nr	Okno param etrów	Parametry wewnętrzne	Objaśnienie
			SETKI TYSIĘCY: Dociąganie
			0 = kompatybilność, dociąganie automatycznie
			1 = z dociąganiem po konturze
			2 = bez dociągania
			3 = dociąganie automatycznie
			MILIONY: Podcięcia
			0 = miejsce nie jest poddawane oszacowaniu przy wcinaniu, - pozostały i toczenie wcinające, - pozostały
			1 = obrabiać podcięcia
			2 = nie obrabiać podcięć
			DZIESIĄTKI MILIONÓW: za/przed osią toczenia
			0 = obróbka przed osią toczenia
			1 = zarezerwowano
5	F FZ	_F	Posuw dla obróbki zgrubnej/wykańczającej Posuw w odciętej dla toczenia wcinaniem
6	FR FX	_FR	Posuw zagłębiania w podcięcie dla obróbki zgrubnej Posuw w rzędnej dla toczenia wcinaniem
7	RP	_RP	Płaszczyzna wycofania przy obróbce wewnętrznej (abs, zawsze średnica)
8	D	_D	Dosuw obróbka zgrubna, (patrz _AMODE JEDNOSTKI)
9	DX	_DX	Dosuw X (patrz _AMODE JEDNOSTKI)
10	DZ	_DZ	Dosuw Z (patrz _AMODE JEDNOSTKI)
11	UX	_UX	Naddatek X (patrz _VARI DZIESIĄTKI TYSIĘCY)
12	UZ	_UZ	Naddatek Z (patrz _VARI DZIESIĄTKI TYSIĘCY)
13	U	_U	Naddatek równoległy do konturu (patrz _VARI DZIESIĄTKI TYSIĘCY)
14	U1	_U1	Dodatkowy naddatek przy obróbce wykańczającej (patrz _AMODE TYSIĄCE)
15	BL	_BL	Definicja półfabrykatu 1 = walec z naddatkiem 2 = naddatek na konturze części gotowej 3 = kontur półfabrykatu jest podany
16	XD	_XD	Definicja półfabrykatu X (patrz _AMODE SETKI TYSIĘCY)
17	ZD	_ZD	Definicja półfabrykatu Z (patrz _AMODE MILIONY)
18	XA	_XA	Granica 1 X (abs, zawsze średnica)
19	ZA	_ZA	Granica 1 Z (abs)
20	XB	_XB	Granica 2 X (patrz _AMODE DZIESIĄTKI MILIONÓW)
21	ZB	_ZB	Granica 2 Z (patrz _AMODE SETKI MILIONÓW)
22	XDA	_XDA	Granica 1 wcinania dla 1. pozycji wcinania ma stronie czołowej (abs, zawsze średnica)
23	XDB	_XDB	Granica 2 wcinania dla 1. pozycji wcinania ma stronie czołowej (abs, zawsze średnica)
24	N	_N	Liczba rowków

Nr	Okno param etrów	Parametry wewnętrzne	Objaśnienie
25	DP	_DP	Odstęp pomiędzy rowkami Rowek wzdłużny: równoległe do osi Z Rowek poprzeczny równoległe do osi X
26	DI	_DI	Odstęp dla przzerwania posuwu 0 = nie ma przzerwania > 0 = z przzerwaniem
27	SC	_SC	Odstęp bezpieczeństwa do obchodzenia przeszkód, przyrostowo
28	D2	_DN	Numer D dla 2. ostrza, gdy nie zaprogramowano ⇒ D+1
29		_GMODE	Tryb geometrii (oszacowanie zaprogramowanych wartości geometrycznych) JEDNOSTKI: Zarezerwowano DZIESIĄTKI: Zarezerwowano SETKI: Wybór obróbka / tylko obliczenie punktu startowego 0 = obróbka normalna (tryb kompatybilności nie jest wymagany) 1 = normalna obróbka 2 = obliczenie pozycji startowej - bez obróbki (tylko dla wywołania z ShopMill/ShopTurn) TYSIĄCE: Ograniczenie 0 = nie 1 = tak DZIESIĄTKI TYSIĘCY: Wprowadzenie granicy 1 X 0 = nie 1 = tak SETKI TYSIĘCY: Wprowadzenie granicy 2 X 0 = nie 1 = tak MILIONY: Wprowadzenie granicy 1 Z 0 = nie 1 = tak DZIESIĄTKI MILIONÓW: Wprowadzenie granicy 2 Z 0 = nie 1 = tak
30		_DMODE	Tryb wyświetlania JEDNOSTKI: Płaszczyzna obróbki G17/18/19 0 = kompatybilność, aktywna pozostaje płaszczyzna działająca przed wywołaniem cyklu 1 = G17 (aktywna tylko w cyklu) 2 = G18 (aktywna tylko w cyklu) 3 = G19 (aktywna tylko w cyklu) DZIESIĄTKI: Rodzaj technologii 1 = skrawanie konturu 2 = obróbka wcinająca konturu 3 = toczenie wcinaniem

Nr	Okno param etrów	Parametry wewnętrzne	Objaśnienie
			SETKI: Obróbka pozostałego materiału 0 = nie 1 = tak
31		_AMODE	Tryb alternatywny JEDNOSTKI: Wybór dosuwu 0 = dosuw DX i DZ przy rodzaju skrawania równoległe do konturu 1 = dosuw D DZIESIĄTKI: Strategia dosuwu 0 = zmienna głębokość skrawania (90 ... 100 %) 1 = stała prędkość skrawania SETKI: Podział na przejścia narzędzia 0 = równomiernie 1 = ustawić na krawędziach TYSIĄCE: Wybór naddatku na konturze U1, obróbka wykańczająca podwójna 0 = nie 1 = tak DZIESIĄTKI TYSIĘCY: Wybór aktualnego półfabrykatu 0 = nie 1 = tak SETKI TYSIĘCY: Wybór naddatku na konturze XD 0 = absolutnie, wartość osi poprzecznej na średnicy 1 = przyrostowo, wartość osi poprzecznej na promieniu MILIONY: Wybór naddatku na półfabrykacie ZD 0 = absolutnie 1 = przyrostowo DZIESIĄTKI MILIONÓW: Wybór granicy 2 XB 0 = absolutnie, wartość osi poprzecznej na średnicy 1 = przyrostowo, wartość osi poprzecznej na promieniu SETKI MILIONÓW: Wybór granicy 2 ZB 0 = absolutnie 1 = przyrostowo JEDEN MILIARD 0 = kanał wiodący 1 = kanał nadążny
32		_PK	Numer kanału partnerskiego, gdy w maszynie są więcej, niż 2 kanały
33	DCH	_DCH	Przesunięcie kanału

18.1.38 Skręt - CYCLE800

Programowanie

```
CYCLE800 (INT _FR, STRING[32] _TC, INT _ST, INT _MODE, REAL _X0, REAL
_Y0, REAL _Z0, REAL _A, REAL _B, REAL _C, REAL _X1, REAL _Y1, REAL
_Z1, INT _DIR, REAL _FR_I , INT _DMODE)
```

Parametry

Nr	Okno param etrów	Parametry wewnętrzne	Objaśnienie
1		_FR	<p>Tryb odsunięcia:</p> <p>0 = bez odsunięcia</p> <p>1 = odsunięcie oś maszyny Z</p> <p>2 = odsunięcie oś maszyny Z, a następnie XY</p> <p>3 = zarezerwowano</p> <p>4 = odsunięcie w kierunku narzędzia maksymalne</p> <p>5 = odsunięcie w kierunku narzędzia przyrostowe</p>
2		_TC	<p>Nazwa zestawu danych skrętu:</p> <p>"" "" (bez nazwy) gdy jest tylko 1 zestaw danych skrętu</p> <p>"0" cofnięcie wyboru zestawu danych skrętu (skasowanie frame skrętu)</p>
3		_ST	<p>Status transformacji</p> <p>JEDNOSTKI:</p> <p>0 = nowy, płaszczyzna skrętu jest kasowana i obliczana na nowo z aktualnymi parametrami</p> <p>1 = addytywnie, płaszczyzna skrętu dodaje się addytywnie do aktywnej płaszczyzny skrętu</p> <p>DZIESIĄTKI: Aktualizacja wierzchołka narzędzia tak/nie (aktywna tylko wtedy, gdy w URUCHOMIENIE jest ustawiona funkcja SKRĘT)</p> <p>0 = nie aktualizować wierzchołka narzędzia</p> <p>1 = aktualizować wierzchołek narzędzia (TRAORI)</p> <p>SETKI: Przystawienie / zorientowanie narzędzia (funkcja jest wyświetlana w oknie wprowadzania SKRĘT narzędzia)</p> <p>0 = nie przystawiać narzędzia</p> <p>1 = przystawić narzędzie (preferowany frez do zarysów promieniowych)</p> <p>2 = zorientować narzędzie tokarskie (gdy jest ustawiona kinematyka osi B dla technologii toczenia w IBN skręt)</p> <p>3 = zorientować narzędzie frezarskie (gdy jest ustawiona kinematyka osi B dla technologii toczenia w IBN skręt)</p> <p>9 = zarezerwowano</p> <p>TYSIĄCE: Wewnętrzny parametr skrętu w JOG</p> <p>DZIESIĄTKI TYSIĘCY: Patrz parametr kierunek _DIR</p> <p>0 = skręt "tak"</p> <p>1 = skręt "nie" kierunek "minus" ³⁾</p> <p>2 = skręt "nie" kierunek "plus" ³⁾</p>

Nr	Okno parametrów	Parametry wewnętrzne	Objaśnienie
			SETKI TYSIĘCY: Patrz parametr kierunek <code>_DIR</code>
			0 = kompatybilność
			1 = wybór kierunku "minus" zoptymalizowany ⁴⁾
			2 = wybór kierunku "plus" zoptymalizowany ⁴⁾
4		<code>_MODE</code> ⁵⁾	Tryb skrętu: Oszacowanie kąta skrętu i kolejności skrętu (kodowanie bitowe!)
			Bit: 7 6
			0 0: Kąt skrętu pojedynczymi osiami -> patrz parametry <code>_A</code> , <code>_B</code> , <code>_C</code>
			0 1: Kąt przestrzenny -> patrz parametry <code>_A</code> , <code>_B</code> ¹⁾
			1 0: Kąt projekcji -> patrz parametry <code>_A</code> , <code>_B</code> , <code>_C</code> ¹⁾
			1 1: Tryb skrętu osie obrotowe bezpośrednio -> patrz parametry <code>_A</code> , <code>_B</code> ¹⁾
			Bit: 5 4 3 2 1 0 (w przypadku kątów przestrzennych bez znaczenia!)
			x x x x 0 1 1. obrót <code>_A</code> wokół X
			x x x x 1 0 1. obrót <code>_A</code> wokół Y
			x x x x 1 1 1. obrót <code>_A</code> wokół Z
			x x 0 1 x x 2. obrót <code>_B</code> wokół X
			x x 1 0 x x 2. obrót <code>_B</code> wokół Y
			x x 1 1 x x 2. obrót <code>_B</code> wokół Z
			0 1 x x x x 3. obrót <code>_C</code> wokół X
			1 0 x x x x 3. obrót <code>_C</code> wokół Y
			1 1 x x x x 3. obrót <code>_C</code> wokół Z
5	X0	<code>_X0</code>	Punkt odniesienia X przed obrotem
6	Y0	<code>_Y0</code>	Punkt odniesienia Y przed obrotem
7	Z0	<code>_Z0</code>	Punkt odniesienia Z przed obrotem
8	X(A)	<code>_A</code>	1. obrót według ustawienia w parametrze <code>_MODE</code>
9	Y(B)	<code>_B</code>	2. obrót według ustawienia w parametrze <code>_MODE</code>
10	Z(C)	<code>_C</code>	3. obrót według ustawienia w parametrze <code>_MODE</code>
11	X1	<code>_X1</code>	Punkt odniesienia X po obrocie
12	Y1	<code>_Y1</code>	Punkt odniesienia Y po obrocie
13	Z1	<code>_Z1</code>	Punkt odniesienia Z po obrocie
14	- albo +	<code>_DIR</code>	Wyzwolić ruch postępowy osi obrotowych (domyślnie = -1!): -1 = pozycjonować na mniejszą wartość osi obrotowej 1 albo ²⁾ +1 = pozycjonować na większą wartość osi obrotowej 1 albo ²⁾ 0 = skręt nie (tylko obliczenie frame skrętu) ^{1) 3)}
15	FR	<code>_FR_I</code>	Wartość (przyr.) odsunięcia w kierunku narzędzia przyrostowo

Nr	Okno parametrów	Parametry wewnętrzne	Objaśnienie
16		<code>_DMODE</code>	<p>Tryb wyświetlania</p> <hr/> <p>JEDNOSTKI: Płaszczyzna obróbki G17/G18/G19</p> <hr/> <p>0 = kompatybilność, aktywna pozostaje płaszczyzna działająca przed wywołaniem cyklu 1 = G17 (aktywna tylko w cyklu) 2 = G18 (aktywna tylko w cyklu) 3 = G19 (aktywna tylko w cyklu)</p>

Wskazówka

Gdy przekazywane parametry poniżej są programowane pośrednio (jako parametry), okno wprowadzania nie jest poddawane przetwarzaniu wstęcznemu: `_FR`, `_ST`, `_TC`, `_MODE`, `_DIR`

- 1) Wybór jest możliwy, gdy w URUCHOMIENIE jest ustawiona funkcja SKRĘT.
- 2) Wybór jest możliwy, gdy w URUCHOMIENIE SKRĘT jest ustawione odniesienie kierunku do osi obrotowej 1 albo 2.

bez pola wyboru przy odniesieniu kierunku nie

- 3) Wybór skręt "nie" może być ukryty SD 55221 bit 0

Skręt "nie" w kierunku "minus" odpowiada `_DIR = 0` i `_ST DZIESIĄTKI TYSIĘCY = 1`

Skręt "nie" w kierunku "plus" odpowiada `_DIR = 0` i `_ST DZIESIĄTKI TYSIĘCY = 2`

- 4) Wybór kierunku osi obrotowej 1 albo 2 następuje również, gdy oś obrotowa z odniesieniem kierunku znajduje się w położeniu biegunowym (wartość pozycji równa zero).

- 5) Przykład kodowania: Obrót pojedynczymi osiami, kolejność obrotów

Binarny: 00011011; dziesiętnie: 27

Identyfikatory osi XYZ odpowiadają osiom geometrycznym kanału NC. Obroty wokół osi XYZ wolno wykonywać pojedynczo. Np. kolejność obrotów ZXZ jest w wywołaniu cyklu CYCLE800 niedozwolona

18.1.39 "Obróbka szybkościowa" - CYCLE832

Programowanie

```
CYCLE832 (REAL S_TOL, INT S_TOLM, REAL S_OTOL)
```

Wskazówka

CYCLE832 nie zwalnia producenta maszyny od niezbędnych zadań optymalizacyjnych przy uruchamianiu maszyny. Dotyczy to optymalizacji osi uczestniczących w obróbce i ustawień NCU (sterowanie wyprzedzające, ograniczenie przyśpieszenia drugiego stopnia, itd.).

Parametry

Nr	Okno param etrów	Parametry wewnętrzne	Objaśnienie
1	Tolera ncja	S_TOL	Tolerancja konturu Tolerancja konturu odpowiada tolerancji osi geometrycznych.
2		S_TOLM	Rodzaj obróbki (technologia) JEDNOSTKI: 0 = cofnięcie wyboru 1 = obróbka wykańczająca (finish) 2 = obróbka półwykańczająca (semifinish) 3 = obróbka zgrubna (rough) DZIESIĄTKI: 0 = kompatybilność ¹⁾ wzgl. bez tolerancji orientacji 1 = tolerancja orientacji w 3. parametrze Aby polepszyć czytelność wywołania cyklu, parametr "rodzaj obróbki" można wprowadzić również tekstem jawnym. Teksty jawne nie są zależne od języka. Są dopuszczalne następujące wprowadzenia: _FINISH = obróbka wykańczająca _SEMIFIN = obróbka półwykańczająca _ROUGH = obróbka zgrubna _ORI_FINISH = obróbka wykańczająca z wprowadzeniem tolerancji orientacji _ORI_SEMIFIN = obróbka półwykańczająca z wprowadzeniem tolerancji orientacji _ORI_ROUGH = obróbka zgrubna z wprowadzeniem tolerancji orientacji OFF = cofnięcie wyboru Uwaga: Pojęcia są oparte o grupę funkcji G 59 (tryb dynamiki dla interpolacji ruchu po torze). Przy pomocy tych tekstów jawnych są pod względem zastosowania klarownie rozdzielane maszyny 3-osiowe i maszyny z wielokrotną transformacją orientacji (TRAORI).

Nr	Okno parametrów	Parametry wewnętrzne	Objaśnienie
3	Tolerancja ORI	S_OTOL	Tolerancja orientacji wzgl. oznaczenie wersji CYCLE832 Parametr tolerancji dla orientacji narzędzia. Jest potrzebny przy wykonywaniu programu obróbki charakteryzującego się dużą prędkością na maszynach z dynamiczną transformacją orientacji (np. obróbka 5-osiowa). Parametr S_OTOL musi zostać zaprogramowany. Dotyczy to również zastosowań na maszynach 3-osiowych w przypadku programów bez orientacji narzędzia (S_OTOL = 1).

1) Tolerancja orientacji wyprowadzona z tolerancji konturu pomnożonej przez współczynnik z danych ustawczych cykliSD55441 do SD55443.

Literatura:

Podręcznik uruchomienia Oprogramowanie podstawowe i oprogramowanie obsługowe; SINUMERIK Operate (IM9), rozdział "Konfiguracja funkcji High Speed Setting (CYCLE832)"

Wskazówka

Przy cofnięciu wyboru CYCLE832 parametr S_TOL musi zostać przekazany z wartością zero.

Przykład: CYCLE832 (0, 0, 1)

Składnia CYCLE832 () jest dla cofnięcia wyboru CYCLE832 również dozwolona.

Przykłady

Przykład 1: CYCLE832 na maszynie 3-osiowej bez transformacji orientacji

a) Wywoływanie cykli z wprowadzeniem tekstu jawnego

Kod programu	Komentarz
G710	; Systemem miar jest system metryczny.
CYCLE832 (0.004, _FINISH, 1)	; Wywołanie CYCLE832 za pomocą: Tolerancja koturu = 0,004 mm, rodzaj obróbki: obróbka wykańczająca
...	; Wykonywanie programu obróbki o dużej prędkości

b) Wywoływanie cykli bez wprowadzenia tekstu jawnego

Kod programu	Komentarz
G710	; patrz wyżej
CYCLE832 (0.004, 1, 1)	; patrz wyżej
...	; patrz wyżej

Przykład 2: CYCLE832 na maszynie 5-osiowej z transformacją orientacji**a) Wywołanie cykli i cofnięcie wyboru z wprowadzeniem tekstu jawnego**

Kod programu	Komentarz
G710	; Systemem miar jest system metryczny.
TRAORI	; Uaktywnienie transformacji orientacji.
CYCLE832(0.3,_ORI_ROUGH,0.8)	; Wywołanie CYCLE832 za pomocą: Tolerancja koturu = 0,3 mm, rodzaj obróbki: Obróbka zgrubna z wprowadzeniem tolerancji orientacji, tolerancja orientacji = 0,8 stopnia
...	; Wykonywanie programu obróbki o dużej prędkości
CYCLE832(0,_OFF,1)	; tolerancja konturu = 0, rodzaj obróbki: cofnięcie wyboru CYCLE832, tolerancja orientacji = 0 stopni

b) Wywołanie cykli i cofnięcie wyboru bez wprowadzenia tekstu jawnego

Kod programu	Komentarz
G710	; patrz wyżej
TRAORI	; patrz wyżej
CYCLE832(0.3,13,0.8)	; patrz wyżej
...	; patrz wyżej
CYCLE832(0,0,1)	; patrz wyżej

Tablice

19.1 Instrukcje

Instrukcja	Art 1)	Znaczenie	W 2)	TP 3)	SA 4)	Opis patrz 5)
1) 2) 3) 4) 5) Objaśnienia patrz legenda (Strona 795).						
:	O	Numer bloku głównego NC, zakończenie znacznika skoku, operator powiązania		+		<i>PGAs/</i> Funkcje arytmetyczne (Strona 70)
*	O	Operator mnożenia		+		<i>PGAs/</i> Funkcje arytmetyczne (Strona 70)
+	O	Operator dodawania		+		<i>PGAs/</i> Funkcje arytmetyczne (Strona 70)
-	O	Operator odejmowania		+		<i>PGAs/</i> Funkcje arytmetyczne (Strona 70)
<	O	Operator porównania, mniejsze		+		<i>PGAs/</i> Funkcje arytmetyczne (Strona 70)
<<	O	Operator powiązania dla łańcuchów znaków		+		<i>PGAs/</i> Funkcje arytmetyczne (Strona 70)
<=	O	Operator porównania, mniejsze lub równe		+		<i>PGAs/</i> Funkcje arytmetyczne (Strona 70)
=	O	Operator przyporządkowania		+		<i>PGAs/</i> Funkcje arytmetyczne (Strona 70)
>=	O	Operator porównania, większe lub równe		+		<i>PGAs/</i> Funkcje arytmetyczne (Strona 70)
/	O	Operator dzielenia		+		<i>PGAs/</i> Funkcje arytmetyczne (Strona 70)
/0 /7		Blok jest maskowany (1. poziom maskowania) Blok jest maskowany (8. poziom maskowania)		+		<i>PGs/</i>
A	A	Nazwa osi	m/s	+		<i>PGAs/</i> Programowanie orientacji narzędzia (A..., B..., C..., LEAD, TILT) (Strona 319)
A2	A	Orientacja narzędzia: kąt RPY lub Eulera	s	+		<i>PGAs/</i> Programowanie orientacji narzędzia (A..., B..., C..., LEAD, TILT) (Strona 319)
A3	A	Orientacja narzędzia: składowa wektora normalnego do kierunku/powierzchni	s	+		<i>PGAs/</i> Programowanie orientacji narzędzia (A..., B..., C..., LEAD, TILT) (Strona 319)

Tablice

19.1 Instrukcje

Instrukcja	Art ¹⁾	Znaczenie	W ²⁾	TP ³⁾	SA ⁴⁾	Opis patrz ⁵⁾
1) 2) 3) 4) 5) Objaśnienia patrz legenda (Strona 795).						
A4	A	Orientacja narzędzia: wektor normalny do powierzchni dla początku bloku	s	+		<i>PGAs/</i> Frezowanie czołowe (A4, B4, C4, A5, B5, C5) (Strona 325)
A5	A	Orientacja narzędzia: wektor normalny do powierzchni dla końca bloku	s	+		<i>PGAs/</i> Frezowanie czołowe (A4, B4, C4, A5, B5, C5) (Strona 325)
ABS	F	Wartość absolutna (bezwzględna)		+	+	<i>PGAs/</i> Funkcje arytmetyczne (Strona 70)
AC	K	Absolutne podanie wymiarów współrzędnych/pozycji	s	+		<i>PGs/</i>
ACC	K	Wpływ na aktualne osiowe przyspieszenie	m	+	+	<i>PGs/</i>
ACCLIMA	K	Wpływ na aktualne maksymalne osiowe przyspieszenie	m	+	+	<i>PGAs/</i> Sterowanie przyspieszeniem w przypadku osi nadążnych (VELOLIMA, ACCLIMA, JERKLIMA) (Strona 465)
ACN	K	Absolutne podanie wymiarów dla osi obrotowych, ruch do pozycji w kierunku ujemnym	s	+		<i>PGs/</i>
ACOS	F	Arcus cosinus (trygonometryczna funkcja)		+	+	<i>PGAs/</i> Funkcje arytmetyczne (Strona 70)
ACP	K	Absolutne podanie wymiarów dla osi obrotowych, ruch do pozycji w kierunku dodatnim	s	+		<i>PGs/</i>
ACTBLOCNO	P	Wyprowadzenie aktualnego numeru bloku alarmowego, również gdy jest aktywne "wyświetlanie aktualnego bloku ukrywane (DISPLOF)!"		+		<i>PGAs/</i> Blokowanie wyświetlania aktualnego bloku (DISPLOF, DISPLON, ACTBLOCNO) (Strona 177)
ADDFRAME	F	Wliczenie i ew. uaktywnienie zmierzonego frame		+	-	<i>PGAs/</i> , <i>FB1s/ (K2)</i> Obliczenie frame z 3 punktów pomiarowych w przestrzeni (MEAFRAME) (Strona 294)
ADIS	A	Droga wygładzania dla funkcji torowych G1, G2, G3, ...	m	+		<i>PGs/</i>
ADISPOS	A	Droga wygładzania dla posuwu szybkiego G0	m	+		<i>PGs/</i>
ADISPOSA	P	Wielkość okna tolerancji dla IPOBRKA	m	+	+	<i>PGAs/</i> Programowane kryterium końca ruchu (FINEA, COARSEA, IPOENDA, IPOBRKA, ADISPOSA) (Strona 273)
ALF	A	Kąt szybkiego cofnięcia	m	+		<i>PGAs/</i> Szybkie cofnięcie od konturu (SETINT LIFTFAST, ALF) (Strona 129)
AMIRROR	G	Programowane lustrzane odbicie	s	+		<i>PGs/</i>

Instrukcja	Art 1)	Znaczenie	W 2)	TP 3)	SA 4)	Opis patrz 5)
1) 2) 3) 4) 5) Objaśnienia patrz legenda (Strona 795).						
AND	K	Logiczne I		+		<i>PGAs/</i> Operacje porównania i operacje logiczne (Strona 73)
ANG	A	Kąt przebiegu konturu	s	+		<i>PGs/</i>
AP	A	Współrzędna kąтова	m/s	+		<i>PGs/</i>
APR	K	Ochrona przed dostępem odczyt/wyświetlanie		+		<i>PGAs/</i> Atrybut: prawa dostępu (APR, APW, APRP, APWP, APRB, APWB) (Strona 40)
APRB	K	Prawo dostępu w celu odczytu, BTSS		+		<i>PGAs/</i> Atrybut: prawa dostępu (APR, APW, APRP, APWP, APRB, APWB) (Strona 40)
APRP	K	Prawo dostępu w celu odczytu, program obróbki		+		<i>PGAs/</i> Atrybut: prawa dostępu (APR, APW, APRP, APWP, APRB, APWB) (Strona 40)
APW	K	Ochrona przed dostępem w celu zapisu		+		<i>PGAs/</i> Atrybut: prawa dostępu (APR, APW, APRP, APWP, APRB, APWB) (Strona 40)
APWB	K	Prawo dostępu w celu zapisu, BTSS		+		<i>PGAs/</i> Atrybut: prawa dostępu (APR, APW, APRP, APWP, APRB, APWB) (Strona 40)
APWP	K	Prawo dostępu w celu zapisu, program obróbki		+		<i>PGAs/</i> Atrybut: prawa dostępu (APR, APW, APRP, APWP, APRB, APWB) (Strona 40)
APX	K	Definicja ochrony przed dostępem dla wykonania podanego elementu językowego		+		<i>PGAs/</i> Redefinicja zmiennych systemowych, zmiennych użytkownika i poleceń językowych NC (REDEF) (Strona 30)
AR	A	Kąt rozwarcia	m/s	+		<i>PGs/</i>
AROT	G	Programowany obrót	s	+		<i>PGs/</i>
AROTS	G	Programowane obrócenia frame z kątami przestrzennymi	s	+		<i>PGs/</i>
AS	K	Definicja makra		+		<i>PGAs/</i> Technika makr (DEFINE ... AS) (Strona 210)
ASCALE	G	Skalowanie programowane	s	+		<i>PGs/</i>
ASIN	F	Funkcja obliczeniowa, arcus sinus		+	+	<i>PGAs/</i> Funkcje arytmetyczne (Strona 70)

Instrukcja	Art ¹⁾	Znaczenie	W ²⁾	TP ³⁾	SA ⁴⁾	Opis patrz ⁵⁾
1) 2) 3) 4) 5) Objaśnienia patrz legenda (Strona 795).						
ASPLINE	G	Akima-Spline	m	+		<i>PGAs/</i> Interpolacja Spline (ASPLINE, BSPLINE, CSPLINE, BAUTO, BNAT, BTAN, EAUTO, ENAT, ETAN, PW, SD, PL) (Strona 234)
ATAN2	F	Arcus tangens2		+	+	<i>PGAs/</i> Funkcje arytmetyczne (Strona 70)
ATOL	K	Specyficzna dla osi tolerancja dla funkcji kompresora, wygładzanie orientacji i rodzaje wygładzania		+		<i>PGAs/</i> Programowana tolerancja konturu/orientacji (CTOL, OTOL, ATOL) (Strona 496)
ATRANS	G	addytywne przesunięcie programowane	s	+		<i>PGs/</i>
AUXFUDEL	P	Skasowanie z globalnej listy, specyficznej dla kanału funkcji pomocniczej		+	-	<i>FB1s/ (H2)</i>
AUXFUDELG	P	Skasowanie z globalnej listy, wszystkich funkcji pomocniczych dla specyficznej dla kanału danej grupy		+	-	<i>FB1s/ (H2)</i>
AUXFUMSEQ	P	Określenie kolejności wyprowadzania funkcji pomocniczych M		+	-	<i>FB1s/ (H2)</i>
AUXFUSYNC	P	Wygenerowanie jako łańcucha znaków kompletnego bloku programu obróbki dla kanałowego SERUPRO-Ende-ASUP, z globalnej listy funkcji pomocniczych.		+	-	<i>FB1s/ (H2)</i>
AX	K	Zmienny identyfikator osi	m/s	+		<i>PGAs/</i> Funkcje osi (AXNAME, AX, SPI, AXTOSPI, ISAXIS, AXSTRING, MODAXVAL) (Strona 591)
AXCTSWE	P	Obrót pojemnika osi		+	-	<i>PGAs/</i> Pojemnik osi (AXCTSWE, AXCTSWED, AXCTSWEC) (Strona 598)
AXCTSWEC	P	Cofnięcie zezwolenia dla obrotu pojemnika osi		+	+	<i>PGAs/</i> Pojemnik osi (AXCTSWE, AXCTSWED, AXCTSWEC) (Strona 598)
AXCTSWED	P	Obrót pojemnika (wariant polecenia do uruchomienia!)		+	-	<i>PGAs/</i> Pojemnik osi (AXCTSWE, AXCTSWED, AXCTSWEC) (Strona 598)
AXIS	K	Identyfikator osi, adres osi		+		<i>PGAs/</i> Definicja zmiennych użytkownika (DEF) (Strona 24)

Instrukcja	Art 1)	Znaczenie	W 2)	TP 3)	SA 4)	Opis patrz 5)
1) 2) 3) 4) 5) Objaśnienia patrz legenda (Strona 795).						
AXNAME	F	Konwertuje wejściowy łańcuch znaków na identyfikator osi		+	-	<i>PGAs!</i> Funkcje osi (AXNAME, AX, SPI, AXTOSPI, ISAXIS, AXSTRING, MODAXVAL) (Strona 591)
AXSTRING	F	Konwertuje łańcuch znaków numer wrzeciona		+	-	<i>PGAs!</i> Funkcje osi (AXNAME, AX, SPI, AXTOSPI, ISAXIS, AXSTRING, MODAXVAL) (Strona 591)
AXTOCHAN	P	Zażądanie osi dla określonego kanału. Jest możliwe z programu NC i z akcji synchronicznej.		+	+	<i>PGAs!</i> Przekazanie osi do innego kanału (AXTOCHAN) (Strona 140)
AXTOINT	F	Zmiana typu danych zmiennej osiowej z AXIS na INT		+	-	<i>PGAs!</i> Konwersje explicite typu danych (AXTOINT, INTTOAX) (Strona 54)
AXTOSPI	F	Konwertuje identyfikator osi na indeks wrzeciona		+	-	<i>PGAs!</i> Funkcje osi (AXNAME, AX, SPI, AXTOSPI, ISAXIS, AXSTRING, MODAXVAL) (Strona 591)
B	A	Nazwa osi	m/s	+		<i>PGAs!</i> Programowanie orientacji narzędzia (A..., B..., C..., LEAD, TILT) (Strona 319)
B2	A	Orientacja narzędzia: kąt RPY lub Eulera	s	+		<i>PGAs!</i> Programowanie orientacji narzędzia (A..., B..., C..., LEAD, TILT) (Strona 319)
B3	A	Orientacja narzędzia: składowa wektora normalnego do kierunku/powierzchni	s	+		<i>PGAs!</i> Programowanie orientacji narzędzia (A..., B..., C..., LEAD, TILT) (Strona 319)
B4	A	Orientacja narzędzia: wektor normalny do powierzchni dla początku bloku	s	+		<i>PGAs!</i> Frezowanie czołowe (A4, B4, C4, A5, B5, C5) (Strona 325)
B5	A	Orientacja narzędzia: wektor normalny do powierzchni dla końca bloku	s	+		<i>PGAs!</i> Frezowanie czołowe (A4, B4, C4, A5, B5, C5) (Strona 325)
B_AND	O	Bitowe I		+		<i>PGAs!</i> Operacje porównania i operacje logiczne (Strona 73)
B_OR	O	Bitowe LUB		+		<i>PGAs!</i> Operacje porównania i operacje logiczne (Strona 73)
B_NOT	O	Bitowa negacja		+		<i>PGAs!</i> Operacje porównania i operacje logiczne (Strona 73)

Instrukcja	Art ¹⁾	Znaczenie	W ²⁾	TP ³⁾	SA ⁴⁾	Opis patrz ⁵⁾
1) 2) 3) 4) 5) Objaśnienia patrz legenda (Strona 795).						
B_XOR	O	Bitowe ALBO		+		<i>PGAs/</i> Operacje porównania i operacje logiczne (Strona 73)
BAUTO	G	Definicja pierwszego segmentu spline przez następne 3 punkty	m	+		<i>PGAs/</i> Interpolacja Spline (ASPLINE, BSPLINE, CSPLINE, BAUTO, BNAT, BTAN, EAUTO, ENAT, ETAN, PW, SD, PL) (Strona 234)
BLOCK	K	Definiuje razem ze słowem kluczowym TO będącą do wykonania część programu w pośrednim przebiegu podprogramu		+		<i>PGAs/</i> Pośrednie wywołanie podprogramu z podaniem części programu do wykonania (CALL BLOCK ... TO ...) (Strona 198)
BLSYNC	K	Wykonywanie procedury przerwania powinno się rozpocząć dopiero z następną zmianą bloku		+		<i>PGAs/</i> Przyporządkowanie i uruchomienie procedury przerwania (SETINT, PRIO, BLSYNC) (Strona 126)
BNAT ⁶⁾	G	Przejście naturalne do pierwszego bloku spline	m	+		<i>PGAs/</i> Interpolacja Spline (ASPLINE, BSPLINE, CSPLINE, BAUTO, BNAT, BTAN, EAUTO, ENAT, ETAN, PW, SD, PL) (Strona 234)
BOOL	K	Typ danych: wartości logiczne TRUE/FALSE lub 1/0		+		<i>PGAs/</i> Definicja zmiennych użytkownika (DEF) (Strona 24)
BOUND	F	Sprawdza, czy wartość leży w zdefiniowanym zakresie. Równość zwraca sprawdzaną wartość.		+	+	<i>PGAs/</i> Minimum, maksimum i zakres zmiennych (MINVAL, MAXVAL i BOUND) (Strona 77)
BRISK ⁶⁾	G	Przyspieszenie skokowe ruchu po torze	m	+		<i>PGAs/</i> Tryb przyspieszenia (BRISK, BRISKA, SOFT, SOFTA, DRIVE, DRIVEA) (Strona 463)
BRISKA	P	Włączenie skokowego przyspieszenia po torze dla zaprogramowanych osi		+	-	<i>PGAs/</i> Tryb przyspieszenia (BRISK, BRISKA, SOFT, SOFTA, DRIVE, DRIVEA) (Strona 463)
BSPLINE	G	B-Spline	m	+		<i>PGAs/</i> Interpolacja Spline (ASPLINE, BSPLINE, CSPLINE, BAUTO, BNAT, BTAN, EAUTO, ENAT, ETAN, PW, SD, PL) (Strona 234)
BTAN	G	Przejście styczne do pierwszego bloku spline	m	+		<i>PGAs/</i> Interpolacja Spline (ASPLINE, BSPLINE, CSPLINE, BAUTO, BNAT, BTAN, EAUTO, ENAT, ETAN, PW, SD, PL) (Strona 234)

Instrukcja	Art 1)	Znaczenie	W 2)	TP 3)	SA 4)	Opis patrz 5)
1) 2) 3) 4) 5) Objaśnienia patrz legenda (Strona 795).						
C	A	Nazwa osi	m/s	+		<i>PGAs/</i> Programowanie orientacji narzędzia (A..., B..., C..., LEAD, TILT) (Strona 319)
C2	A	Orientacja narzędzia: kąt RPY lub Eulera	s	+		<i>PGAs/</i> Programowanie orientacji narzędzia (A..., B..., C..., LEAD, TILT) (Strona 319)
C3	A	Orientacja narzędzia: składowa wektora normalnego do kierunku/powierzchni	s	+		<i>PGAs/</i> Programowanie orientacji narzędzia (A..., B..., C..., LEAD, TILT) (Strona 319)
C4	A	Orientacja narzędzia: wektor normalny do powierzchni dla początku bloku	s	+		<i>PGAs/</i> Frezowanie czołowe (A4, B4, C4, A5, B5, C5) (Strona 325)
C5	A	Orientacja narzędzia: wektor normalny do powierzchni dla końca bloku	s	+		<i>PGAs/</i> Frezowanie czołowe (A4, B4, C4, A5, B5, C5) (Strona 325)
CAC	K	Absolutne dosunięcie do pozycji		+		<i>PGAs/</i> Ruch do pozycji kodowanych (CAC, CIC, CDC, CACP, CACN) (Strona 233)
CACN	K	Ruch do wartości zapisanej w tablicy jest wykonywany absolutnie w kierunku ujemnym		+		<i>PGAs/</i> Ruch do pozycji kodowanych (CAC, CIC, CDC, CACP, CACN) (Strona 233)
CACP	K	Ruch do wartości zapisanej w tablicy jest wykonywany absolutnie w kierunku dodatnim		+		<i>PGAs/</i> Ruch do pozycji kodowanych (CAC, CIC, CDC, CACP, CACN) (Strona 233)
CALCDAT	F	Oblicza promieni i punktów środkowy okręgu z 3 lub 4 punktów		+	-	<i>PGAs/</i> Obliczenie danych okręgu (CALCDAT) (Strona 646)
CALCPOSI	F	Sprawdzenie na naruszenie obszaru ochrony, ograniczenie obszaru pracy i ograniczenia softwareowe		+	-	<i>PGAs/</i>
CALL	K	Pośrednie wywołanie podprogramu		+		<i>PGAs/</i> Pośrednie wywołanie podprogramu (CALL) (Strona 197)
CALLPATH	P	Programowana ścieżka szukania przy wywołaniach podprogramu		+	-	<i>PGAs/</i> Rozszerzenie ścieżki szukania przy wywołaniach podprogramu (CALLPATH) (Strona 201)
CANCEL	P	Anulowanie modalnej akcji synchronicznej		+	-	<i>FBSY</i>
CASE	K	Warunkowe rozgałęzienie programu		+		<i>PGAs/</i> Rozgałęzienie programu (CASE ... OF ... DEFAULT ...) (Strona 102)

Instrukcja	Art ¹⁾	Znaczenie	W ²⁾	TP ³⁾	SA ⁴⁾	Opis patrz ⁵⁾
1) 2) 3) 4) 5) Objaśnienia patrz legenda (Strona 795).						
CDC	K	Bezpośrednie dosunięcie do pozycji		+		<i>PGAs!</i> Ruch do pozycji kodowanych (CAC, CIC, CDC, CACP, CACN) (Strona 233)
CDOF ⁶⁾	G	Nadzór na kolizję WYŁ.	m	+		<i>PGs!</i>
CDOF2	G	Nadzór na kolizję WYŁ., przy frezowaniu obwodowym 3D	m	+		<i>PGs!</i>
CDON	G	Nadzór na kolizję WŁ.	m	+		<i>PGs!</i>
CFC ⁶⁾	G	Stały posuw po konturze	m	+		<i>PGs!</i>
CFIN	G	Stały posuw tylko przy zakrzywieniu wewnętrznym, nie przy zakrzywieniu zewnętrznym	m	+		<i>PGs!</i>
CFINE	F	Przyporządkowanie przesunięcia dokładnego do zmiennej FRAME		+	-	<i>PGAs!</i> Przesunięcie zgrubne i dokładne (CFINE, CTRANS) (Strona 290)
CFTCP	G	Posuw stały w punkcie odniesienia ostrza narzędzia, tor punktu środkowego	m	+		<i>PGs!</i>
CHAN	K	Specyfikacja zakresu obowiązywania danych		+		<i>PGAs!</i> Definicja zmiennych użytkownika (DEF) (Strona 24)
CHANDATA	P	Ustawienie numeru kanału dla dostępu do danych kanału		+	-	<i>PGAs!</i> Pamięć robocza (CHANDATA, COMPLETE, INITIAL) (Strona 217)
CHAR	K	Typ danych: Znak ASCII		+		<i>PGAs!</i> Definicja zmiennych użytkownika (DEF) (Strona 24)
CHF	A	Faza; Wartość = długość fazy	s	+		<i>PGs!</i>
CHKDM	F	Kontrola jednoznaczności w ramach magazynu		+	-	<i>FBWs!</i>
CHKDNO	F	Badanie jednoznaczności numerów D		+	-	<i>PGAs!</i> Dowolne nadawanie numerów D: sprawdzenie numerów D (CHKDNO) (Strona 430)
CHR	A	Faza; Wartość = długość fazy w kierunku ruchu		+		<i>PGs!</i>
CIC	K	Przyrostowe dosunięcie do pozycji		+		<i>PGAs!</i> Ruch do pozycji kodowanych (CAC, CIC, CDC, CACP, CACN) (Strona 233)
CIP	G	Interpolacja kołowa przez punkt pośredni	m	+		<i>PGs!</i>
CLEARM	P	Cofnięcie jednego/wielu znaczników do koordynacji kanałów		+	+	<i>PGAs!</i> Koordynacja programu (INIT, START, WAITM, WAITMC, WAITE, SETM, CLEARM) (Strona 118)

Instrukcja	Art 1)	Znaczenie	W 2)	TP 3)	SA 4)	Opis patrz 5)
1) 2) 3) 4) 5) Objaśnienia patrz legenda (Strona 795).						
CLRINT	P	Cofnięcie wyboru przerwania		+	-	<i>PGAs/</i> Skasowanie przyporządkowania procedury przerwania (CLRINT) (Strona 128)
CMIRROR	F	Lustrzane odbicie jednej osi współrzędnych		+	-	<i>PGAs/</i> Funkcje arytmetyczne (Strona 70)
COARSEA	K	Koniec ruchu przy osiągnięciu "zatrzymania dokładnego zgrubnie"	m	+		<i>PGAs/</i> Programowane kryterium końca ruchu (FINEA, COARSEA, IPOENDA, IPOBRKA, ADISPOSA) (Strona 273)
COLLPAIR	F	Sprawdza przynależności do najazdu kolizyjnego		+		<i>PGAs/</i> Sprawdzenie pod kątem pary kolizyjnej (COLLPAIR) (Strona 385)
COMPCAD	G	Kompresor WŁ.: Zoptymalizowana jakość powierzchni w przypadku programów CAD	m	+		<i>PGAs/</i> Kompresja bloków NC (COMPON, COMPCURV, COMPCAD, COMPOF) (Strona 245)
COMPCURV	G	Kompresor WŁ.: krzywe wielomianowe o stałej krzywiznie	m	+		<i>PGAs/</i> Kompresja bloków NC (COMPON, COMPCURV, COMPCAD, COMPOF) (Strona 245)
COMPLETE		Instrukcja sterownicza dla wyprowadzenia i wczytania danych		+		<i>PGAs/</i> Pamięć robocza (CHANDATA, COMPLETE, INITIAL) (Strona 217)
COMPOF 6)	G	Kompresor WYŁ.	m	+		<i>PGAs/</i> Kompresja bloków NC (COMPON, COMPCURV, COMPCAD, COMPOF) (Strona 245)
COMPON	G	Kompresor WŁ.		+		<i>PGAs/</i> Kompresja bloków NC (COMPON, COMPCURV, COMPCAD, COMPOF) (Strona 245)
CONTDCON	P	Włączenie dekodowania konturu w formie tablicy		+	-	<i>PGAs/</i> Sporządzenie kodowanej tablicy konturu (CONTDCON) (Strona 640)
CONTPRON	P	Włączenie przygotowania konturu		+	-	<i>PGAs/</i> Sporządzenie tablicy konturu (CONTPRON) (Strona 634)
CORROF	P	Wybór wszystkich aktywnych ruchów nałożonych jest cofany		+	-	<i>PGs/</i>
COS	F	Cosinus (trygonometryczna funkcja)		+	+	<i>PGAs/</i> Funkcje arytmetyczne (Strona 70)

Instrukcja	Art ¹⁾	Znaczenie	W ²⁾	TP ³⁾	SA ⁴⁾	Opis patrz ⁵⁾
1) 2) 3) 4) 5) Objaśnienia patrz legenda (Strona 795).						
COUPDEF	P	Definicja zespołu ELG / zespołu wrzecziona synchronicznego		+	-	<i>PGAs/</i> Wrzecziono synchroniczne: Programowanie (COUPDEF, COUPDEL, COUPON, COUPONC, COUPOF, COUPOFS, COUPRES, WAITC) (Strona 539)
COUPDEL	P	Skasowanie zespołu ELG		+	-	<i>PGAs/</i> Wrzecziono synchroniczne: Programowanie (COUPDEF, COUPDEL, COUPON, COUPONC, COUPOF, COUPOFS, COUPRES, WAITC) (Strona 539)
COUPOF	P	Wyłączenie zespołu ELG/pary wrzeczion synchronicznych		+	-	<i>PGAs/</i> Wrzecziono synchroniczne: Programowanie (COUPDEF, COUPDEL, COUPON, COUPONC, COUPOF, COUPOFS, COUPRES, WAITC) (Strona 539)
COUPOFS	P	Wyłączenie zespołu ELG / pary wrzeczion synchronicznych z zatrzymaniem wrzecziona nadążnego		+	-	<i>PGAs/</i> Wrzecziono synchroniczne: Programowanie (COUPDEF, COUPDEL, COUPON, COUPONC, COUPOF, COUPOFS, COUPRES, WAITC) (Strona 539)
COUPON	P	Włączenie zespołu ELG/pary wrzeczion synchronicznych		+	-	<i>PGAs/</i> Wrzecziono synchroniczne: Programowanie (COUPDEF, COUPDEL, COUPON, COUPONC, COUPOF, COUPOFS, COUPRES, WAITC) (Strona 539)
COUPONC	P	Włączenie zespołu ELG / przejęcie pary wrzeczion synchronicznym z poprzedzającym zaprogramowaniem		+	-	<i>PGAs/</i> Wrzecziono synchroniczne: Programowanie (COUPDEF, COUPDEL, COUPON, COUPONC, COUPOF, COUPOFS, COUPRES, WAITC) (Strona 539)
COUPRES	P	Cofnięcie zespołu ELG		+	-	<i>PGAs/</i> Wrzecziono synchroniczne: Programowanie (COUPDEF, COUPDEL, COUPON, COUPONC, COUPOF, COUPOFS, COUPRES, WAITC) (Strona 539)
CP ⁶⁾	G	Ruch po torze	m	+		<i>PGAs/</i> Ruch kartezjański PTP (Strona 369)
CPBC	K	Sprzężenie ogólne: Kryterium zmiany bloku		+	+	<i>FB3s/ (M3)</i>
CPDEF	K	Sprzężenie ogólne: Utworzenie trybu sprzężenia		+	+	<i>FB3s/ (M3)</i>

Instrukcja	Art 1)	Znaczenie	W 2)	TP 3)	SA 4)	Opis patrz 5)
1) 2) 3) 4) 5) Objaśnienia patrz legenda (Strona 795).						
CPDEL	K	Sprzężenie ogólne: Skasowanie trybu sprzężenia		+	+	<i>FB3sl (M3)</i>
CPFMOF	K	Sprzężenie ogólne: Zachowanie się osi nadążnej przy kompletnym wyłączeniu		+	+	<i>FB3sl (M3)</i>
CPFMON	K	Sprzężenie ogólne: Zachowanie się osi nadążnej przy włączeniu		+	+	<i>FB3sl (M3)</i>
CPFMSON	K	Sprzężenie ogólne: Tryb synchronizacji		+	+	<i>FB3sl (M3)</i>
CPFPOS	K	Sprzężenie ogólne: Pozycja synchronizacji osi nadążnej		+	+	<i>FB3sl (M3)</i>
CPFRS	K	Sprzężenie ogólne: Układ odniesienia współrzędnych		+	+	<i>FB3sl (M3)</i>
CPLA	K	Sprzężenie ogólne: Definicja osi wiodącej		+	-	<i>FB3sl (M3)</i>
CPLCTID	K	Sprzężenie ogólne: Numer tablicy krzywych		+	+	<i>FB3sl (M3)</i>
CPLDEF	K	Sprzężenie ogólne: Definicja osi wiodącej i utworzenie trybu sprzężenia		+	+	<i>FB3sl (M3)</i>
CPLDEL	K	Sprzężenie ogólne: Skasowanie osi wiodącej modułu sprzężenia		+	+	<i>FB3sl (M3)</i>
CPLDEN	K	Sprzężenie ogólne: Mianownik współczynnika sprzężenia		+	+	<i>FB3sl (M3)</i>
CPLINSC	K	Sprzężenie ogólne: Współczynnik skalowania wartości wejściowej dla osi wiodącej		+	+	<i>FB3sl (M3)</i>
CPLINTR	K	Sprzężenie ogólne: Wartość przesunięcia wartości wejściowej dla osi wiodącej		+	+	<i>FB3sl (M3)</i>
CPLNUM	K	Sprzężenie ogólne: Licznik współczynnika sprzężenia		+	+	<i>FB3sl (M3)</i>
CPLOF	K	Sprzężenie ogólne: Wyłączenie osi wiodącej modułu sprzężenia		+	+	<i>FB3sl (M3)</i>
CPLON	K	Sprzężenie ogólne: Włączenie osi wiodącej modułu sprzężenia		+	+	<i>FB3sl (M3)</i>
CPLOUTSC	K	Sprzężenie ogólne: Współczynnik skalowania wartości wyjściowej sprzężenia		+	+	<i>FB3sl (M3)</i>
CPLOUTTR	K	Sprzężenie ogólne: Wartość przesunięcia wartości wyjściowej sprzężenia		+	+	<i>FB3sl (M3)</i>
CPLPOS	K	Sprzężenie ogólne: Pozycja synchronizacji osi wiodącej		+	+	<i>FB3sl (M3)</i>
CPLSETVAL	K	Sprzężenie ogólne: Odniesienie sprzężenia		+	+	<i>FB3sl (M3)</i>

Instrukcja	Art ¹⁾	Znaczenie	W ²⁾	TP ³⁾	SA ⁴⁾	Opis patrz ⁵⁾
1) 2) 3) 4) 5) Objaśnienia patrz legenda (Strona 795).						
CPMALARM	K	Sprzężenie ogólne: Ukrywanie specjalnych, dotyczących sprzężenia wyprowadzeń alarmów		+	+	<i>FB3sl (M3)</i>
CPMBRAKE	K	Sprzężenie ogólne: Zachowanie się osi nadążnej przy określonych sygnałach i rozkazach Stop		+	-	<i>FB3sl (M3)</i>
CPMPRT	K	Sprzężenie ogólne: Zachowanie się sprzężenia przy starcie programu obróbki pod przebiegiem szukania za pomocą testu programu		+	+	<i>FB3sl (M3)</i>
CPMRESET	K	Sprzężenie ogólne: Zachowanie się sprzężenia przy RESET		+	+	<i>FB3sl (M3)</i>
CPMSTART	K	Sprzężenie ogólne: Zachowanie się sprzężenia przy starcie programu obróbki		+	+	<i>FB3sl (M3)</i>
CPMVDI	K	Sprzężenie ogólne: Zachowanie się osi nadążnej na określone sygnały interfejsu NC/PLC		+	+	<i>FB3sl (M3)</i>
CPOF	K	Sprzężenie ogólne: Wyłączenie modułu sprzężenia		+	+	<i>FB3sl (M3)</i>
CPON	K	Sprzężenie ogólne: Włączenie modułu sprzężenia		+	+	<i>FB3sl (M3)</i>
CPRECOF ⁶⁾	G	Programowana dokładność konturu WYŁ.	m	+		<i>PGAsl</i> Programowalna dokładność konturu (CPRECON, CPRECOF) (Strona 470)
CPRECON	G	Programowana dokładność konturu WŁ.	m	+		<i>PGAsl</i> Programowalna dokładność konturu (CPRECON, CPRECOF) (Strona 470)
CPRES	K	Sprzężenie ogólne: Uaktywnia zaprojektowane dane synchronicznego sprzężenia wrzeczona		+	-	
CPROT	P	Specyficzny dla kanału obszar ochrony WŁ./WYŁ.		+	-	<i>PGAsl</i> Uaktywnienie / wyłączenie aktywności obszarów ochrony (CPROT, NPROT) (Strona 224)
CPROTDEF	P	Definicja obszaru ochrony specyficznego dla kanału		+	-	<i>PGAsl</i> Ustalenie obszarów ochrony (CPROTDEF, NPROTDEF) (Strona 221)
CPSETTYPE	K	Sprzężenie ogólne: Typ sprzężenia		+	+	<i>FB3sl (M3)</i>
CPSYNCOF	K	Sprzężenie ogólne: Wartość progowa dla ruchu synchronicznego pozycjonowania "zgrubnie"		+	+	<i>FB3sl (M3)</i>

Instrukcja	Art 1)	Znaczenie	W 2)	TP 3)	SA 4)	Opis patrz 5)
1) 2) 3) 4) 5) objaśnienia patrz legenda (Strona 795).						
CPSYNCO2	K	Sprężenie ogólne: Wartość progowa dla ruchu synchronicznego pozycjonowania "zgrubnie" 2		+	+	FB3sl (M3)
CPSYNCOV	K	Sprężenie ogólne: Wartość progowa dla ruchu synchronicznego szybkiego "zgrubnie"		+	+	FB3sl (M3)
CPSYNFIP	K	Sprężenie ogólne: Wartość progowa dla ruchu synchronicznego pozycjonowania "dokładnie"		+	+	FB3sl (M3)
CPSYNFIP2	K	Sprężenie ogólne: Wartość progowa dla ruchu synchronicznego pozycjonowania "dokładnie" 2		+	+	FB3sl (M3)
CPSYNFIV	K	Sprężenie ogólne: Wartość progowa dla ruchu synchronicznego szybkiego "dokładnie"		+	+	FB3sl (M3)
CR	A	Promień okręgu	s	+		PGsl
CROT	F	Obrót aktualnego układu współrzędnych		+	-	PGAsl Funkcje arytmetyczne (Strona 70)
CROTS	F	Programowane obroty frame z kątami przestrzennymi (obroty w podanych osiach)	s	+	-	PGsl
CRPL	F	Obrót frame w dowolnej płaszczyźnie		+	-	FB1sl (K2)
CSCALE	F	Współczynnik skali dla wielu osi		+	-	PGAsl Funkcje arytmetyczne (Strona 70)
CSPLINE	F	Spline sześcienny	m	+		PGAsl Interpolacja Spline (ASPLINE, BSPLINE, CSPLINE, BAUTO, BNAT, BTAN, EAUTO, ENAT, ETAN, PW, SD, PL) (Strona 234)
CT	G	Okrąg z przejściem stycznym	m	+		PGsl
CTAB	F	Określona pozycja osi nadążnej na podstawie pozycji osi wiodącej z tablicy krzywych		+	+	PGAsl Odczyt wartości tablicy krzywych (CTABTSV, CTABTEV, CTABTSP, CTABTEP, CTABSSV, CTABSEV, CTAB, CTABINV, CTABTMIN, CTABTMAX) (Strona 520)
CTABDEF	P	Definicja tablicy WŁ.		+	-	PGAsl Definiowanie tablic krzywych (CTABDEF, CTABEND) (Strona 509)
CTABDEL	P	Skasowanie tablicy krzywych		+	-	PGAsl Kasowanie tablic krzywych (CTABDEL) (Strona 515)

Instrukcja	Art ¹⁾	Znaczenie	W ²⁾	TP ³⁾	SA ⁴⁾	Opis patrz ⁵⁾
1) 2) 3) 4) 5) Objaśnienia patrz legenda (Strona 795).						
CTABEND	P	Definicja tablicy WYŁ.		+	-	<i>PGAs/</i> Definiowanie tablic krzywych (CTABDEF, CTABEND) (Strona 509)
CTABEXISTS	F	Sprawdza tablicę krzywej o numerze n		+	+	<i>PGAs/</i> Sprawdzenie istnienia pliku tablicy krzywych (CTABEXISTS) (Strona 515)
CTABFNO	F	Liczba jeszcze możliwych tablic krzywych w pamięci		+	+	<i>PGAs/</i> Tablice krzywych: sprawdzić wykorzystanie zasobów (CTABNO, CTABNOMEM, CTABFNO, CTABSEGID, CTABSEG, CTABFSEG, CTABMSEG, CTABPOLID, CTABPOL, CTABFPOL, CTABMPOL) (Strona 525)
CTABFPOL	F	Liczba jeszcze możliwych wielomianów w pamięci		+	+	<i>PGAs/</i> Tablice krzywych: sprawdzić wykorzystanie zasobów (CTABNO, CTABNOMEM, CTABFNO, CTABSEGID, CTABSEG, CTABFSEG, CTABMSEG, CTABPOLID, CTABPOL, CTABFPOL, CTABMPOL) (Strona 525)
CTABFSEG	F	Liczba jeszcze możliwych segmentów krzywych w pamięci		+	+	<i>PGAs/</i> Tablice krzywych: sprawdzić wykorzystanie zasobów (CTABNO, CTABNOMEM, CTABFNO, CTABSEGID, CTABSEG, CTABFSEG, CTABMSEG, CTABPOLID, CTABPOL, CTABFPOL, CTABMPOL) (Strona 525)
CTABID	F	Daje numer n-tej tablicy krzywych		+	+	<i>PGAs/</i> Tablice krzywych: określenie właściwości tablicy (CTABID, CTABISLOCK, CTABMEMTYP, CTABPERIOD) (Strona 518)
CTABINV	F	Określona pozycja osi wiodącej na podstawie pozycji osi nadążnej z tablicy krzywych		+	+	<i>PGAs/</i> Odczyt wartości tablicy krzywych (CTABTSV, CTABTEV, CTABTSP, CTABTEP, CTABSSV, CTABSEV, CTAB, CTABINV, CTABTMIN, CTABTMAX) (Strona 520)
CTABISLOCK	F	Zwraca stan zablokowania tablicy krzywych o numerze n		+	+	<i>PGAs/</i> Tablice krzywych: określenie właściwości tablicy (CTABID, CTABISLOCK, CTABMEMTYP, CTABPERIOD) (Strona 518)
CTABLOCK	P	Skasowanie i zastąpienie, zablokowanie		+	+	<i>PGAs/</i> Zablokowanie tablic krzywych przed skasowaniem i zastąpieniem (CTABLOCK, CTABUNLOCK) (Strona 517)

Instrukcja	Art 1)	Znaczenie	W 2)	TP 3)	SA 4)	Opis patrz 5)
1) 2) 3) 4) 5) Objaśnienia patrz legenda (Strona 795).						
CTABMEMTYP	F	Zwraca pamięć, w której jest utworzona tablica krzywych o numerze n.		+	+	<i>PGAs!</i> Tablice krzywych: określenie właściwości tablicy (CTABID, CTABISLOCK, CTABMEMTYP, CTABPERIOD) (Strona 518)
CTABMPOL	F	Liczba maksymalnie możliwych wielomianów w pamięci		+	+	<i>PGAs!</i> Tablice krzywych: sprawdzić wykorzystanie zasobów (CTABNO, CTABNOMEM, CTABFNO, CTABSEGID, CTABSEG, CTABFSEG, CTABMSEG, CTABPOLID, CTABPOL, CTABFPOL, CTABMPOL) (Strona 525)
CTABMSEG	F	Liczba maksymalnie możliwych segmentów krzywych w pamięci		+	+	<i>PGAs!</i> Tablice krzywych: sprawdzić wykorzystanie zasobów (CTABNO, CTABNOMEM, CTABFNO, CTABSEGID, CTABSEG, CTABFSEG, CTABMSEG, CTABPOLID, CTABPOL, CTABFPOL, CTABMPOL) (Strona 525)
CTABNO	F	Liczba zdefiniowanych tablic krzywych w SRAM lub DRAM		+	+	<i>FB3sl (M3)</i>
CTABNOMEM	F	Liczba zdefiniowanych tablic krzywych w SRAM lub DRAM		+	+	<i>PGAs!</i> Tablice krzywych: sprawdzić wykorzystanie zasobów (CTABNO, CTABNOMEM, CTABFNO, CTABSEGID, CTABSEG, CTABFSEG, CTABMSEG, CTABPOLID, CTABPOL, CTABFPOL, CTABMPOL) (Strona 525)
CTABPERIOD	F	Zwraca okresowość tablicy tablicy krzywych o numerze n		+	+	<i>PGAs!</i> Tablice krzywych: określenie właściwości tablicy (CTABID, CTABISLOCK, CTABMEMTYP, CTABPERIOD) (Strona 518)
CTABPOL	F	Liczba już zastosowanych wielomianów w pamięci		+	+	<i>PGAs!</i> Tablice krzywych: sprawdzić wykorzystanie zasobów (CTABNO, CTABNOMEM, CTABFNO, CTABSEGID, CTABSEG, CTABFSEG, CTABMSEG, CTABPOLID, CTABPOL, CTABFPOL, CTABMPOL) (Strona 525)
CTABPOLID	F	Liczba wielomianów krzywej użytych przez tablicę krzywej o numerze n		+	+	<i>PGAs!</i> Tablice krzywych: sprawdzić wykorzystanie zasobów (CTABNO, CTABNOMEM, CTABFNO, CTABSEGID, CTABSEG, CTABFSEG, CTABMSEG, CTABPOLID, CTABPOL, CTABFPOL, CTABMPOL) (Strona 525)

Instrukcja	Art ¹⁾	Znaczenie	W ²⁾	TP ³⁾	SA ⁴⁾	Opis patrz ⁵⁾
1) 2) 3) 4) 5) Objaśnienia patrz legenda (Strona 795).						
CTABSEG	F	Liczba już zastosowanych segmentów krzywej w pamięci		+	+	<i>PGAs!</i> Tablice krzywych: sprawdzić wykorzystanie zasobów (CTABNO, CTABNOMEM, CTABFNO, CTABSEGID, CTABSEG, CTABFSEG, CTABMSEG, CTABPOLID, CTABPOL, CTABFPOL, CTABMPOL) (Strona 525)
CTABSEGID	F	Liczba segmentów krzywej użytych przez tablicę krzywych o numerze n		+	+	<i>PGAs!</i> Tablice krzywych: sprawdzić wykorzystanie zasobów (CTABNO, CTABNOMEM, CTABFNO, CTABSEGID, CTABSEG, CTABFSEG, CTABMSEG, CTABPOLID, CTABPOL, CTABFPOL, CTABMPOL) (Strona 525)
CTABSEV	F	Daje wartość końcową osi nadążnej segmentu tablicy krzywych		+	+	<i>PGAs!</i> Odczyt wartości tablicy krzywych (CTABTSV, CTABTEV, CTABTSP, CTABTEP, CTABSSV, CTABSEV, CTAB, CTABINV, CTABTMIN, CTABTMAX) (Strona 520)
CTABSSV	F	Daje wartość początkową osi nadążnej segmentu tablicy krzywych		+	+	<i>PGAs!</i> Odczyt wartości tablicy krzywych (CTABTSV, CTABTEV, CTABTSP, CTABTEP, CTABSSV, CTABSEV, CTAB, CTABINV, CTABTMIN, CTABTMAX) (Strona 520)
CTABTEP	F	Daje wartość osi wiodącej na końcu tablicy krzywych		+	+	<i>PGAs!</i> Odczyt wartości tablicy krzywych (CTABTSV, CTABTEV, CTABTSP, CTABTEP, CTABSSV, CTABSEV, CTAB, CTABINV, CTABTMIN, CTABTMAX) (Strona 520)
CTABTEV	F	Daje wartość osi nadążnej na końcu tablicy krzywych		+	+	<i>PGAs!</i> Odczyt wartości tablicy krzywych (CTABTSV, CTABTEV, CTABTSP, CTABTEP, CTABSSV, CTABSEV, CTAB, CTABINV, CTABTMIN, CTABTMAX) (Strona 520)
CTABTMAX	F	Daje wartość maksymalną osi nadążnej tablicy krzywych		+	+	<i>PGAs!</i> Odczyt wartości tablicy krzywych (CTABTSV, CTABTEV, CTABTSP, CTABTEP, CTABSSV, CTABSEV, CTAB, CTABINV, CTABTMIN, CTABTMAX) (Strona 520)

Instrukcja	Art 1)	Znaczenie	W 2)	TP 3)	SA 4)	Opis patrz 5)
1) 2) 3) 4) 5) Objaśnienia patrz legenda (Strona 795).						
CTABTMIN	F	Daje wartość minimalną osi nadążnej tablicy krzywych		+	+	<i>PGAs/</i> Odczyt wartości tablicy krzywych (CTABTSV, CTABTEV, CTABTSP, CTABTEP, CTABSSV, CTABSEV, CTAB, CTABINV, CTABTMIN, CTABTMAX) (Strona 520)
CTABTSP	F	Daje wartość osi wiodącej na początku tablicy krzywych		+	+	<i>PGAs/</i> Odczyt wartości tablicy krzywych (CTABTSV, CTABTEV, CTABTSP, CTABTEP, CTABSSV, CTABSEV, CTAB, CTABINV, CTABTMIN, CTABTMAX) (Strona 520)
CTABTSV	F	Daje wartość osi nadążnej na początku tablicy krzywych		+	+	<i>PGAs/</i> Odczyt wartości tablicy krzywych (CTABTSV, CTABTEV, CTABTSP, CTABTEP, CTABSSV, CTABSEV, CTAB, CTABINV, CTABTMIN, CTABTMAX) (Strona 520)
CTABUNLOCK	P	Wyłączenie blokady kasowania i zastąpienia		+	+	<i>PGAs/</i> Zablokowanie tablic krzywych przed skasowaniem i zastąpieniem (CTABLOCK, CTABUNLOCK) (Strona 517)
CTOL	K	Tolerancja konturu dla funkcji kompresora, wygładzanie orientacji i rodzaje wygładzania		+		<i>PGAs/</i> Programowana tolerancja konturu/orientacji (CTOL, OTOL, ATOL) (Strona 496)
CTRANS	F	Przesunięcie punktu zerowego dla wielu osi		+	-	<i>PGAs/</i> Przesunięcie zgrubne i dokładne (CFINE, CTRANS) (Strona 290)
CUT2D 6)	G	Korekcja narzędzia 2D	m	+		<i>PGs/</i>
CUT2DF	G	Korekcja narzędzia 2D Korekcja narzędzia działa w stosunku do aktualnego frame (płaszczyzna skośna).	m	+		<i>PGs/</i>
CUT3DC	G	Korekcja narzędzia 3D, frezowanie obwodowe	m	+		<i>PGAs/</i> Uaktywnienie korekcji narzędzia 3D (CUT3DC..., CUT3DF...) (Strona 410)
CUT3DCC	G	Korekcja narzędzia 3D, frezowanie obwodowe z płaszczyznami ograniczającymi	m	+		<i>PGAs/</i> Korekcja narzędzia 3D: Uwzględnienie powierzchni ograniczającej (CUT3DCC, CUT3DCCD) (Strona 420)
CUT3DCCD	G	Korekcja narzędzia 3D, frezowanie obwodowe z płaszczyznami ograniczającymi z narzędziem różnicowym	m	+		<i>PGAs/</i> Korekcja narzędzia 3D: Uwzględnienie powierzchni ograniczającej (CUT3DCC, CUT3DCCD) (Strona 420)

Instrukcja	Art ¹⁾	Znaczenie	W ²⁾	TP ³⁾	SA ⁴⁾	Opis patrz ⁵⁾
1) 2) 3) 4) 5) Objaśnienia patrz legenda (Strona 795).						
CUT3DF	G	Korekcja narzędzia 3D, frezowanie czołowe	m	+		<i>PGAs/</i> Uaktywnienie korekcji narzędzia 3D (CUT3DC..., CUT3DF...) (Strona 410)
CUT3DFF	G	Korekcja narzędzia 3D frezowanie czołowe ze stałym zorientowaniem narzędzia zależnie od aktywnego frame	m	+		<i>PGAs/</i> Uaktywnienie korekcji narzędzia 3D (CUT3DC..., CUT3DF...) (Strona 410)
CUT3DFS	G	Korekcja narzędzia 3D frezowanie czołowe ze stałym zorientowaniem narzędzia niezależnie od aktywnego frame	m	+		<i>PGAs/</i> Uaktywnienie korekcji narzędzia 3D (CUT3DC..., CUT3DF...) (Strona 410)
CUTCONOF ⁶⁾	G	Stała korekcja narzędzia WYL.	m	+		<i>PGs/</i>
CUTCONON	G	Stała korekcja promienia WŁ.	m	+		<i>PGs/</i>
CUTMOD	K	Włączenie funkcji "Modyfikacja danych korekcyjnych w przypadku narzędzi obrotowych"		+		<i>PGAs/</i> Modyfikacja danych skrawania w przypadku narzędzi obrotowych (CUTMOD) (Strona 445)
CYCLE60	C	Cykl grawerowania		+		<i>PGAs/</i> Cykl grawerowania - CYCLE60 (Strona 693)
CYCLE61	C	Frezowanie płaszczyzny		+		<i>PGAs/</i> Frezowanie płaszczyzny - CYCLE61 (Strona 669)
CYCLE62	C	Wywołanie konturu		+		<i>PGAs/</i> Wywołanie konturu - CYCLE62 (Strona 695)
CYCLE63	C	Frezowanie kieszeni kształtowej		+		<i>PGAs/</i> Frezowanie kieszeni konturowej - CYCLE63 (Strona 700)
CYCLE64	C	Wiercenie wstępne kieszeni kształtowej		+		<i>PGAs/</i> Wiercenie wstępne kieszeni konturowej - CYCLE64 (Strona 699)
CYCLE70	C	Frezowanie gwintu		+		<i>PGAs/</i> Frezowanie gwintu - CYCLE70 (Strona 691)
CYCLE72	C	Frezowanie konturu		+		<i>PGAs/</i> Frezowanie konturu - CYCLE72 (Strona 696)
CYCLE76	C	Frezowanie czopa prostokątnego		+		<i>PGAs/</i> Frezowanie czopa prostokątnego - CYCLE76 (Strona 676)
CYCLE77	C	Frezowanie czopa kołowego		+		<i>PGAs/</i> Frezowanie czopa kołowego - CYCLE77 (Strona 678)

Instrukcja	Art 1)	Znaczenie	W 2)	TP 3)	SA 4)	Opis patrz 5)
1) 2) 3) 4) 5) objaśnienia patrz legenda (Strona 795).						
CYCLE78	C	Wiercenie otworu z frezowaniem gwintu		+		<i>PGAs/</i> Wiercenie otworu z frezowaniem gwintu - CYCLE78 (Strona 662)
CYCLE79	C	Wielokąt		+		<i>PGAs/</i> Wielobok - CYCLE79 (Strona 680)
CYCLE81	C	Wiercenie, nawiercanie		+		<i>PGAs/</i> Wiercenie, nawiercanie - CYCLE81 (Strona 651)
CYCLE82	C	Wiercenie, pogłębianie czołowe		+		<i>PGAs/</i>
CYCLE83	C	Wiercenie otworów głębokich		+		<i>PGAs/</i> Wiercenie głębokich otworów - CYCLE83 (Strona 654)
CYCLE84	C	Gwintowanie otworu bez oprawki kompensacyjnej		+		<i>PGAs/</i> Gwintowanie otworu bez oprawki kompensacyjnej - CYCLE84 (Strona 657)
CYCLE85	C	Rozwiercanie dokładne		+		<i>PGAs/</i> Rozwiercanie - CYCLE85 (Strona 653)
CYCLE86	C	Wytaczanie		+		<i>PGAs/</i> Wytaczanie - CYCLE86 (Strona 656)
CYCLE92	C	Przecięcie		+		<i>PGAs/</i> Przecinanie - CYCLE92 (Strona 717)
CYCLE95	C	Skrawanie konturu		+		<i>PGAs/</i> Skrawanie konturu - CYCLE95 (Strona 718)
CYCLE98	C	Łańcuch gwintów		+		<i>PGAs/</i> Łańcuch gwintów - CYCLE98 (Strona 713)
CYCLE99	C	Toczenie gwintu		+		<i>PGAs/</i> Toczenie gwintu - CYCLE99 (Strona 710)
CYCLE800	C	Skręt		+		<i>PGAs/</i> Skręt - CYCLE800 (Strona 724)
CYCLE801	C	Siatka albo ramka		+		<i>PGAs/</i> Siatka albo ramka - CYCLE801 (Strona 667)
CYCLE802	C	Dowolne pozycje		+		<i>PGAs/</i>
CYCLE832	C	Ustawienia dla obróbki szybkościowej		+		<i>PGAs/</i> "Obróbka szybkościowa" - CYCLE832 (Strona 727)
CYCLE840	C	Gwintowanie otworu z oprawką kompensacyjną		+		<i>PGAs/</i> Gwintowanie otworu z oprawką kompensacyjną - CYCLE840 (Strona 660)

Tablice

19.1 Instrukcje

Instrukcja	Art 1)	Znaczenie	W 2)	TP 3)	SA 4)	Opis patrz 5)
1) 2) 3) 4) 5) Objaśnienia patrz legenda (Strona 795).						
CYCLE899	C	Frezowanie rowka otwartego		+		<i>PGAs/</i> Frezowanie rowka otwartego - CYCLE899 (Strona 687)
CYCLE930	C	Rowek		+		<i>PGAs/</i> Wytoczenie - CYCLE930 (Strona 705)
CYCLE940	C	Kształty pocięć		+		<i>PGAs/</i> Podcięcie kształtowe - CYCLE940 (Strona 708)
CYCLE951	C	Skrawanie		+		<i>PGAs/</i> Skrawanie warstwowe - CYCLE951 (Strona 703)
CYCLE952	C	Obróbka wcinaniem konturu		+		<i>PGAs/</i> Toczenie wcinające konturu - CYCLE952 (Strona 720)
D	A	Numer korekcji narzędzia		+		<i>PGs/</i>
D0	A	Przy zaprogramowaniu D0 korekcje narzędzia nie działają.		+		<i>PGs/</i>
DAC	K	Absolutne, pojedynczymi blokami, specyficzne dla osi programowanie w średnicy	s	+		<i>PGs/</i>
DC	K	Absolutne podawanie wymiarów dla osi obrotowych, bezpośredni ruch do pozycji	s	+		<i>PGs/</i>
DEF	K	Definicja zmiennej		+		<i>PGAs/</i> Definicja zmiennych użytkownika (DEF) (Strona 24)
DEFAULT	K	Gałąź w rozgałęzieniu CASE		+		<i>PGAs/</i> Rozgałęzienie programu (CASE ... OF ... DEFAULT ...) (Strona 102)
DEFINE	K	Słowo kluczowe do definicji makr		+		<i>PGAs/</i> Technika makr (DEFINE ... AS) (Strona 210)
DELAYFSTOF	P	Zdefiniowanie końca obszaru Stopp-Delay	m	+	-	<i>PGAs/</i> Warunkowo przerywalne segmenty programu (DELAYFSTON, DELAYFSTOF) (Strona 475)
DELAYFSTON	P	Zdefiniowanie początku obszaru Stopp-Delay	m	+	-	<i>PGAs/</i> Rozgałęzienie programu (CASE ... OF ... DEFAULT ...) (Strona 102)
DELDL	F	Skasowanie korekcji addytywnych		+	-	<i>PGAs/</i> Skasowanie korekcji addytywnych (DELDL) (Strona 396)
DELDTG	P	Skasowanie pozostałej drogi		-	+	<i>FBSY</i>

Instrukcja	Art 1)	Znaczenie	W 2)	TP 3)	SA 4)	Opis patrz 5)
1) 2) 3) 4) 5) Objaśnienia patrz legenda (Strona 795).						
DELETE	P	Skasowanie podanego pliku. Nazwę pliku można podać ze ścieżką i identyfikatorem pliku.		+	-	<i>PGAsl</i> Skasowanie pliku (DELETE) (Strona 148)
DELMLOWNE R	F	Skasowanie miejsca narzędzia w magazynie		+	-	<i>FBWsl</i>
DEMLRES	F	Skasowanie rezerwacji miejsca w magazynie		+	-	<i>FBWsl</i>
DELMT	P	Skasowanie Multitool		+	-	<i>FBWsl</i>
DELOBJ	F	Kasuje elementy łańcuchów kinematycznych, obszarów ochrony, elementów obszarów ochrony, najazdów kolizyjnych i danych transmisji		+		<i>PGAsl</i> Usunięcie komponentów (DELOBJ) (Strona 381)
DELT	P	Skasowanie narzędzia		+	-	<i>FBWsl</i>
DELTC	P	Skasowanie zestawu danych nośnika narzędzi		+	-	<i>FBWsl</i>
DELTOOLENV	F	Skasowanie zestawów danych do opisu środowisk narzędziowych		+	-	<i>FB1sl (W1)</i>
DIACYCOFA	K	Specyficzne dla osi, modalne programowanie na średnicy: WYŁ. w cyklach	m	+		<i>FB1sl (P1)</i>
DIAM90	G	Programowanie w średnicy dla G90, programowanie w promieniu dla G91	m	+		<i>PGAsl</i>
DIAM90A	K	Specyficzne dla osi, modalne programowanie w średnicy dla G90, programowanie w promieniu dla G91 i IC	m	+		<i>PGsl</i>
DIAMCHAN	K	Przejęcie wszystkich osi z MD funkcji osi do stanu kanału programowania w średnicy		+		<i>PGsl</i>
DIAMCHANA	K	Przejęcie stanu kanału programowania w średnicy		+		<i>PGsl</i>
DIAMCYCOF	G	Specyficzne dla kanału programowanie na średnicy: WYŁ. w cyklach	m	+		<i>FB1sl (P1)</i>
DIAMOF 6)	G	Programowanie na średnicy: WYŁ. Położenie podstawowe patrz producent maszyny	m	+		<i>PGsl</i>
DIAMOFA	K	Specyficzne dla osi, modalne programowanie na średnicy: WYŁ. Położenie podstawowe patrz producent maszyny	m	+		<i>PGsl</i>
DIAMON	G	Programowanie na średnicy: WŁ.	m	+		<i>PGsl</i>

Instrukcja	Art ¹⁾	Znaczenie	W ²⁾	TP ³⁾	SA ⁴⁾	Opis patrz ⁵⁾
1) 2) 3) 4) 5) Objaśnienia patrz legenda (Strona 795).						
DIAMONA	K	Specyficzne dla osi, modalne programowanie na średnicy: WŁ. Udostępnienie patrz producent maszyny	m	+		<i>PGsI</i>
DIC	K	Względnie pojedynczymi blokami, specyficzne dla osi programowanie w średnicy	s	+		<i>PGsI</i>
DILF	A	Droga wycofania (długość)	m	+		<i>PGsI</i>
DISABLE	P	Interrupt WYŁ.		+	-	<i>PGAsI</i> Wyłączenie aktywności / reaktywowanie przyporządkowania procedury przzerwania (DISABLE, ENABLE) (Strona 127)
DISC	A	Zwiększenie okręgu przejściowego, korekcja promienia narzędzia	m	+		<i>PGsI</i>
DISCL	A	Odstęp punktu końcowego szybkiego ruchu dosuwu, od płaszczyzny obróbki		+		<i>PGsI</i>
DISPLOF	PA	Ukrywanie wyświetlania aktualnego bloku		+		<i>PGAsI</i> Blokowanie wyświetlania aktualnego bloku (DISPLOF, DISPLON, ACTBLOCNO) (Strona 177)
DISPLON	PA	Wyłączenie blokowania wyświetlania aktualnego bloku		+		<i>PGAsI</i> Blokowanie wyświetlania aktualnego bloku (DISPLOF, DISPLON, ACTBLOCNO) (Strona 177)
DISPR	A	Różnica torowa repos	s	+		<i>PGAsI</i> Dosunięcie przywracające do konturu (REPOSA, REPOSL, REPOSQ, REPOSQA, REPOSH, REPOSHA, DISR, DISPR, RMIBL, RMBBL, RMEBL, RMNBL) (Strona 483)
DISR	A	Odstęp repos	s	+		<i>PGAsI</i> Dosunięcie przywracające do konturu (REPOSA, REPOSL, REPOSQ, REPOSQA, REPOSH, REPOSHA, DISR, DISPR, RMIBL, RMBBL, RMEBL, RMNBL) (Strona 483)
DISRP	A	Odległość płaszczyzny wycofania od płaszczyzny obróbki przy miękkim dosunięciu i odsunięciu		+		<i>PGsI</i>
DITE	A	Droga wybiegu gwintu	m	+		<i>PGsI</i>
DITS	A	Droga dobiegu gwintu	m	+		<i>PGsI</i>
DIV	K	Dzielenie integer		+		<i>PGAsI</i> Funkcje arytmetyczne (Strona 70)

Instrukcja	Art 1)	Znaczenie	W 2)	TP 3)	SA 4)	Opis patrz 5)
1) 2) 3) 4) 5) Objaśnienia patrz legenda (Strona 795).						
DL	A	Wybór zależnej od miejsca addytywnej korekcji narzędzia (DL, korekcja sumaryczna, ustawiania)	m	+		<i>PGAs/</i> Wybranie korekcji addytywnych (DL) (Strona 394)
DO	A	Słowo kluczowe dla akcji synchronicznej, przy spełnionym warunku wyzwala akcję		-	+	<i>FBSY</i>
DRFOF	P	Wyłączenie przesunięć kółkiem ręcznym (DRF)	m	+	-	<i>PGs/</i>
DRIVE	G	Przyśpieszenie ruchu po torze zależne od prędkości	m	+		<i>PGAs/</i> Tryb przyśpieszenia (BRISK, BRISKA, SOFT, SOFTA, DRIVE, DRIVEA) (Strona 463)
DRIVEA	P	Włączenie załamanej charakterystyki przyśpieszenia dla zaprogramowanych osi		+	-	<i>PGAs/</i> Tryb przyśpieszenia (BRISK, BRISKA, SOFT, SOFTA, DRIVE, DRIVEA) (Strona 463)
DYNFINISH	G	Dynamika dla wygładzania wykańczającego	m	+		<i>PGAs/</i> Uaktywnienie specyficznych dla technologii wartości dynamiki (DYNNORM, DYNPOS, DYNROUGH, DYNSEMIFIN, DYNFINISH) (Strona 467)
DYNNORM 6)	G	Normalna dynamika	m	+		<i>PGAs/</i> Uaktywnienie specyficznych dla technologii wartości dynamiki (DYNNORM, DYNPOS, DYNROUGH, DYNSEMIFIN, DYNFINISH) (Strona 467)
DYNPOS	G	Dynamika dla pozycjonowania, gwintowania otworu	m	+		<i>PGAs/</i> Uaktywnienie specyficznych dla technologii wartości dynamiki (DYNNORM, DYNPOS, DYNROUGH, DYNSEMIFIN, DYNFINISH) (Strona 467)
DYNROUGH	G	Dynamika dla obróbki zgrubnej	m	+		<i>PGAs/</i> Uaktywnienie specyficznych dla technologii wartości dynamiki (DYNNORM, DYNPOS, DYNROUGH, DYNSEMIFIN, DYNFINISH) (Strona 467)
DYNSEMIFIN	G	Dynamika dla obróbki wykańczającej	m	+		<i>PGAs/</i> Uaktywnienie specyficznych dla technologii wartości dynamiki (DYNNORM, DYNPOS, DYNROUGH, DYNSEMIFIN, DYNFINISH) (Strona 467)

Instrukcja	Art ¹⁾	Znaczenie	W ²⁾	TP ³⁾	SA ⁴⁾	Opis patrz ⁵⁾
1) 2) 3) 4) 5) Objaśnienia patrz legenda (Strona 795).						
DZERO	P	Zaznacza wszystkie numery D jednostki TO jako nie obowiązujące		+	-	<i>PGAs/</i> Dowolne nadawanie numerów D: ustawienie nie obowiązywania numerów D (DZERO) (Strona 433)
EAUTO	G	Definicja ostatniego segmentu spline przez ostatnie 3 punkty	m	+		<i>PGAs/</i> Interpolacja Spline (ASPLINE, BSPLINE, CSPLINE, BAUTO, BNAT, BTAN, EAUTO, ENAT, ETAN, PW, SD, PL) (Strona 234)
EGDEF	P	Definicja przekładni elektronicznej		+	-	<i>PGAs/</i> Zdefiniowanie przekładni elektronicznej (EGDEF) (Strona 532)
EGDEL	P	Skasowanie definicji sprzężenia dla osi nadążnej		+	-	<i>PGAs/</i> Skasowanie definicji przekładni elektronicznej (EGDEL) (Strona 538)
EGOFC	P	Wyłączenie przekładni elektronicznej w sposób ciągły		+	-	<i>PGAs/</i> Wyłączenie przekładni elektronicznej (EGOFS, EGOFC) (Strona 537)
EGOFS	P	Wyłączenie przekładni elektronicznej w sposób selektywny		+	-	<i>PGAs/</i> Wyłączenie przekładni elektronicznej (EGOFS, EGOFC) (Strona 537)
EGON	P	Włączenie przekładni elektronicznej		+	-	<i>PGAs/</i> Wyłączenie przekładni elektronicznej (EGOFS, EGOFC) (Strona 537)
EGONSYN	P	Włączenie przekładni elektronicznej		+	-	<i>PGAs/</i> Wyłączenie przekładni elektronicznej (EGOFS, EGOFC) (Strona 537)
EGONSYNE	P	Włączenie przekładni elektronicznej, z zadaniem trybu dosunięcia		+	-	<i>PGAs/</i> Wyłączenie przekładni elektronicznej (EGOFS, EGOFC) (Strona 537)
ELSE	K	Rozgałęzienie programu, gdy warunek IF nie jest spełniony		+		<i>PGAs/</i> Warunkowa instrukcja i rozgałęzienie (IF, ELSE, ENDIF) (Strona 112)
ENABLE	P	Interrupt WŁ.		+	-	<i>PGAs/</i> Wyłączenie aktywności / reaktywowanie przyporządkowania procedury przerywania (DISABLE, ENABLE) (Strona 127)
ENAT ⁶⁾	G	Naturalne przejście krzywej do następnego bloku ruchu	m	+		<i>PGAs/</i> Interpolacja Spline (ASPLINE, BSPLINE, CSPLINE, BAUTO, BNAT, BTAN, EAUTO, ENAT, ETAN, PW, SD, PL) (Strona 234)
ENDFOR	K	Wiersz końcowy pętli FOR		+		<i>PGAs/</i> Pętla FOR (FOR ... TO ..., ENDFOR) (Strona 114)

Instrukcja	Art 1)	Znaczenie	W 2)	TP 3)	SA 4)	Opis patrz 5)
1) 2) 3) 4) 5) Objaśnienia patrz legenda (Strona 795).						
ENDIF	K	Wiersz końcowy rozgałęzienia IF		+		<i>PGAs!</i> Warunkowa instrukcja i rozgałęzienie (IF, ELSE, ENDIF) (Strona 112)
ENDLABEL	K	Znacznik końcowy dla powtórzeń programu obróbki przez REPEAT		+		<i>PGAs!, FB1s! (K1)</i> Powtórzenie programu (REPEAT, REPEATB, ENDLABEL, P) (Strona 104)
ENDLOOP	K	Wiersz końcowy pętli programowej bez końca LOOP		+		<i>PGAs!</i> Pętla programowa bez końca (LOOP, ENDLOOP) (Strona 113)
ENDPROC	K	Wiersz końcowy programu z wierszem początkowym PROC		+		
ENDWHILE	K	Wiersz końcowy z pętlą WHILE		+		<i>PGAs!</i> Pętla programowa z warunkiem na początku pętli (WHILE, ENDPWHILE) (Strona 116)
ESRR	P	Parametryzacja niezależnego napędu dla wycofania ESR		+		<i>PGAs!</i> Zaprojektowanie wycofania niezależnego dla napędu (ESRR) (Strona 631)
ESRS	P	Parametryzacja niezależnego napędu dla zatrzymania ESR		+		<i>PGAs!</i> Zaprojektowanie zatrzymania niezależnego dla napędu (ESRS) (Strona 630)
ETAN	G	Styczne przejście krzywej do następnego bloku ruchu na początku spline	m	+		<i>PGAs!</i> Interpolacja Spline (ASPLINE, BSPLINE, CSPLINE, BAUTO, BNAT, BTAN, EAUTO, ENAT, ETAN, PW, SD, PL) (Strona 234)
EVERY	K	Wykonanie akcji synchronicznej przy przejściu warunku od FALSE do TRUE		-	+	<i>FBSY</i>
EX	K	Wartość kluczowa dla przyporządkowania wartości przy pisaniu wykładniczym		+		<i>PGAs!</i> Predefiniowane zmienne użytkownika: parametry obliczeniowe (R) (Strona 20)
EXECSTRING	P	Przekazanie zmiennej String z będącym do wykonania wierszem programu obróbki		+	-	<i>PGAs!</i> Pośrednie programowanie wierszy programu obróbki (EXECSTRING) (Strona 69)
EXECTAB	P	Wykonanie elementu z tablicy ruchów.		+	-	<i>PGAs!</i> Pośrednie programowanie wierszy programu obróbki (EXECSTRING) (Strona 69)
EXECUTE	P	Wykonanie programu Wł.		+	-	<i>PGAs!</i> Wyłączenie przygotowania konturu (EXECUTE) (Strona 648)

Instrukcja	Art ¹⁾	Znaczenie	W ²⁾	TP ³⁾	SA ⁴⁾	Opis patrz ⁵⁾
1) 2) 3) 4) 5) Objaśnienia patrz legenda (Strona 795).						
EXP	F	Funkcja wykładnicza ex		+	+	<i>PGAs/</i> Funkcje arytmetyczne (Strona 70)
EXTCALL	A	Wykonywanie podprogramu zewnętrznego		+	+	<i>PGAs/</i> Wykonywanie podprogramu zewnętrznego (840D sl) (EXTCALL) (Strona 202)
EXTCLOSE	P	Zamknięcie zewnętrznego urządzenia/pliku otwartego w celu zapisu		+	-	<i>PGAs/</i> Process DataShare - Wyprowadzenie do zewnętrznego urządzenia/pliku (EXTOPEN, WRITE, EXTCLOSE) (Strona 614)
EXTERN	K	Poinformowanie o podprogramie z przekazaniem parametrów		+		<i>PGAs/</i> Wywołanie podprogramu bez przekazania parametrów (Strona 189)
EXTOPEN	P	Otwarcie zewnętrznego urządzenia/pliku dla kanału w celu zapisu		+	-	<i>PGAs/</i> Process DataShare - Wyprowadzenie do zewnętrznego urządzenia/pliku (EXTOPEN, WRITE, EXTCLOSE) (Strona 614)
F	A	Wartość posuwu (w połączeniu z G4 jest pod F również programowany czas oczekiwania)		+	+	<i>PGs/</i>
FA	K	Posuw osiowy	m	+	+	<i>PGs/</i>
FAD	A	Posuw dosuwu dla miękkiego dosunięcia i odsunięcia		+		<i>PGs/</i>
FALSE	K	Stała logiczna: fałsz		+		<i>PGAs/</i> Definicja zmiennych użytkownika (DEF) (Strona 24)
FB	A	Posuw pojedynczymi blokami		+		<i>PGs/</i>
FCTDEF	P	Definicja funkcji wielomianowej		+	-	<i>PGAs/</i> Korekcja narzędzia online (PUTFTOCF, FCTDEF, PUTFTOC, FTOCON, FTOCOF) (Strona 405)
FCUB	G	Posuw zmienny według spline sześciennego	m	+		<i>PGAs/</i> Przebieg posuwu (FNORM, FLIN, FCUB, FPO) (Strona 458)
FD	A	Posuw po torze dla nałożenia ruchu kółkiem ręcznym	s	+		<i>PGs/</i>
FDA	K	Posuw osiowy dla nałożenia ruchu kółkiem ręcznym	s	+		<i>PGs/</i>
FENDNORM ⁶⁾	G	Zwłoka w narożnikach WYŁ.	m	+		<i>PGAs/</i> Zmniejszenie posuwu ze zwłoką na narożach (FENDNORM, G62, G621) (Strona 272)

Instrukcja	Art 1)	Znaczenie	W 2)	TP 3)	SA 4)	Opis patrz 5)
1) 2) 3) 4) 5) Objasnienia patrz legenda (Strona 795).						
FFWOF 6)	G	Sterowanie wyprzedzające WYŁ.	m	+		<i>PGAs/</i> Ruch ze sterowaniem wyprzedzającym (FFWON, FFWOF) (Strona 469)
FFWON	G	Sterowanie wyprzedzające Wł.	m	+		<i>PGAs/</i> Ruch ze sterowaniem wyprzedzającym (FFWON, FFWOF) (Strona 469)
FGREF	K	Promień odniesienia przy osiach obrotowych albo współczynniki odniesienia toru przy osiach orientacji (interpolacja wektorowa)	m	+		<i>PGs/</i>
FGROUP	P	Ustalenie osi z posuwem po torze		+	-	<i>PGs/</i>
FI	K	Parametry dla dostępu do danych frame: Przesunięcie dokładne		+		<i>PGAs/</i> Odczyt i zmiana komponentów frame (TR, FI, RT, SC, MI) (Strona 287)
FIFOCTRL	G	Sterowanie buforem przebiegu wyprzedzającego	m	+		<i>PGAs/</i> Przebieg programu z pamięcią przebiegu wyprzedzającego (STOPFIFO, STARTFIFO, FIFOCTRL, STOPRE) (Strona 472)
FILEDATE	P	Daje datę ostatniego dostępu do pliku w celu zapisu		+	-	<i>PGAs/</i> Odczyt informacji o pliku (FILEDATE, FILETIME, FILESIZE, FILESTAT, FILEINFO) (Strona 155)
FILEINFO	P	Daje sumę FILEDATE, FILESIZE, FILESTAT i FILETIME		+	-	<i>PGAs/</i> Odczyt informacji o pliku (FILEDATE, FILETIME, FILESIZE, FILESTAT, FILEINFO) (Strona 155)
FILESIZE	P	Daje aktualną wielkość pliku		+	-	<i>PGAs/</i> Odczyt informacji o pliku (FILEDATE, FILETIME, FILESIZE, FILESTAT, FILEINFO) (Strona 155)
FILESTAT	P	Daje status pliku praw do odczytu, zapisu, wykonania, wyświetlenia, skasowania (rwxsd)		+	-	<i>PGAs/</i> Odczyt informacji o pliku (FILEDATE, FILETIME, FILESIZE, FILESTAT, FILEINFO) (Strona 155)
FILETIME	P	Daje czas zegarowy ostatniego dostępu do pliku w celu zapisu		+	-	<i>PGAs/</i> Odczyt informacji o pliku (FILEDATE, FILETIME, FILESIZE, FILESTAT, FILEINFO) (Strona 155)
FINEA	K	Koniec ruchu przy osiągnięciu "zatrzymania dokładnego dokładnie"	m	+		<i>PGAs/</i> Programowane kryterium końca ruchu (FINEA, COARSEA, IPOENDA, IPOBRKA, ADISPOSA) (Strona 273)
FL	K	Prędkość graniczna dla osi synchronicznych	m	+		<i>PGs/</i>

Instrukcja	Art 1)	Znaczenie	W 2)	TP 3)	SA 4)	Opis patrz 5)
1) 2) 3) 4) 5) Objaśnienia patrz legenda (Strona 795).						
FLIN	G	Posuw zmienny liniowo	m	+		<i>PGAsl</i> Przebieg posuwu (FNORM, FLIN, FCUB, FPO) (Strona 458)
FMA	K	Wiele posuwów osiowo	m	+		<i>PGsl</i>
FNORM 6)	G	Posuw normalny według DIN66025	m	+		<i>PGAsl</i> Przebieg posuwu (FNORM, FLIN, FCUB, FPO) (Strona 458)
FOC	K	Ograniczenia momentu/siły działające pojedynczymi blokami	s	-	+	<i>FBSY</i>
FOCOF	K	Wyłączenie modalnego ograniczenia momentu/siły	m	-	+	<i>FBSY</i>
FOCON	K	Włączenie modalnego ograniczenia momentu/siły	m	-	+	<i>FBSY</i>
FOR	K	Pętla FOR o stałej liczbie przebiegów		+		<i>PGAsl</i> Pętla FOR (FOR ... TO ..., ENDFOR) (Strona 114)
FP	A	Punkt stały: Numer punktu stałego, do którego ma zostać wykonany ruch	s	+		<i>PGsl</i>
FPO	K	Przebieg posuwu zaprogramowany przez wielomian		+		<i>PGAsl</i> Przebieg posuwu (FNORM, FLIN, FCUB, FPO) (Strona 458)
FPR	P	Oznaczenie osi obrotowej		+	-	<i>PGsl</i>
FPRAOF	P	Wyłączenie posuwu na obrót		+	-	<i>PGsl</i>
FPRAON	P	Włączenie posuwu na obrót		+	-	<i>PGsl</i>
FRAME	K	Typ danych do ustalenia układu współrzędnych		+		<i>PGAsl</i> Definicja nowych frame (DEF FRAME) (Strona 289)
FRC	A	Posuw dla zaokrąglenia i fazy	s	+		<i>PGsl</i>
FRCM	A	Posuw dla zaokrąglenia i fazy modalnie	m	+		<i>PGsl</i>
FROM	K	Akcja jest wykonywana, gdy warunek zostanie raz spełniony i jak długo akcja synchroniczna jest aktywna		-	+	<i>FBSY</i>
FTOC	P	Zmiana korekcji dokładnej narzędzia		-	+	<i>FBSY</i>
FTOCOF 6)	G	Działająca online dokładna korekcja narzędzia WYŁ	m	+		<i>PGAsl</i> Korekcja narzędzia online (PUTFTOCF, FCTDEF, PUTFTOC, FTOCON, FTOCOF) (Strona 405)
FTOCON	G	Działająca online dokładna korekcja narzędzia WŁ.	m	+		<i>PGAsl</i> Korekcja narzędzia online (PUTFTOCF, FCTDEF, PUTFTOC, FTOCON, FTOCOF) (Strona 405)
FXS	K	Ruch do twardego zderzaka wł.	m	+	+	<i>PGsl</i>

Instrukcja	Art 1)	Znaczenie	W 2)	TP 3)	SA 4)	Opis patrz 5)
1) 2) 3) 4) 5) Objaśnienia patrz legenda (Strona 795).						
FXST	K	Granica momentu dla ruchu do zderzaka twardego	m	+	+	PGsl
FXSW	K	Okno nadzoru dla ruchu do zderzaka twardego		+	+	PGsl
FZ	K	Posuw na ostrze	m	+		PGsl
G0	G	Interpolacja liniowa z posuwem szybkim (ruch posuwem szybkim)	m	+		PGsl
G1 6)	G	Interpolacja liniowa z posuwem (interpolacja prostoliniowa)	m	+		PGsl
G2	G	Interpolacja kołowa zgodnie z ruchem wskazówek zegara	m	+		PGsl
G3	G	Interpolacja kołowa przeciwnie do ruchu wskazówek zegara	m	+		PGsl
G4	G	Czas oczekiwania, wstępnie określony	s	+		PGsl
G5	G	Szlifowanie wcinające skośne	s	+		PGAsl Oś skośna (TRAANG) (Strona 364)
G7	G	Ruch wyrównawczy przy szlifowaniu wcinającym skośnym	s	+		PGAsl Oś skośna (TRAANG) (Strona 364)
G9	G	Zmniejszenie prędkości przy zatrzymaniu dokładnym	s	+		PGsl
G17 6)	G	Wybór płaszczyzny roboczej X/Y	m	+		PGsl
G18	G	Wybór płaszczyzny roboczej Z/X	m	+		PGsl
G19	G	Wybór płaszczyzny roboczej Y/Z	m	+		PGsl
G25	G	Dolne ograniczenie obszaru pracy	s	+		PGsl
G26	G	Górne ograniczenie obszaru pracy	s	+		PGsl
G33	G	Nacinanie gwintu o stałym skoku	m	+		PGsl
G34	G	Nacinanie gwintu o skoku liniowo rosnącym	m	+		PGsl
G35	G	Nacinanie gwintu o skoku liniowo malejącym	m	+		PGsl
G40 6)	G	Korekcja promienia narzędzia WYŁ.	m	+		PGsl
G41	G	Korekcja promienia narzędzia na lewo od konturu	m	+		PGsl
G42	G	Korekcja promienia narzędzia na prawo od konturu	m	+		PGsl
G53	G	Blokowanie aktualnego przesunięcia punktu zerowego (pojedynczymi blokami)	s	+		PGsl
G54	G	1.ustawiane przesunięcie punktu zerowego	m	+		PGsl
G55	G	2. Ustawiane przesunięcie punktu zerowego	m	+		PGsl

Tablice

19.1 Instrukcje

Instrukcja	Art ¹⁾	Znaczenie	W ²⁾	TP ³⁾	SA ⁴⁾	Opis patrz ⁵⁾
1) 2) 3) 4) 5) Objaśnienia patrz legenda (Strona 795).						
G56	G	3. Ustawiane przesunięcie punktu zerowego	m	+		<i>PGsl</i>
G57	G	4. Ustawiane przesunięcie punktu zerowego	m	+		<i>PGsl</i>
G58 (840D sl)	G	Osiowe programowane przesunięcie punktu zerowego absolutne, przesunięcie zgrubne	s	+		<i>PGsl</i>
G58 (828D)	G	5. Ustawiane przesunięcie punktu zerowego	m	+		<i>PGsl</i>
G59 (840D sl)	G	Osiowe programowane przesunięcie punktu zerowego addytywne, przesunięcie dokładne	s	+		<i>PGsl</i>
G59 (828D)	G	6. Ustawiane przesunięcie punktu zerowego	m	+		<i>PGsl</i>
G60 ⁶⁾	G	Zmniejszenie prędkości przy zatrzymaniu dokładnym	m	+		<i>PGsl</i>
G62	G	Zaokrąglanie narożników wewnętrznych przy aktywnej korekcji promieni narzędzia (G41, G42)	m	+		<i>PGAsl</i> Zmniejszenie posuwu ze zwłoką na narożach (FENDNORM, G62, G621) (Strona 272)
G63	G	Gwintowanie otworu z oprawką kompensacyjną	s	+		<i>PGsl</i>
G64	G	Praca z płynnym przechodzeniem	m	+		<i>PGsl</i>
G70	G	Całowe podawanie wymiarów dla danych geometrycznych (długości)	m	+	+	<i>PGsl</i>
G71 ⁶⁾	G	Metryczne podawanie wymiarów dla danych geometrycznych (długości)	m	+	+	<i>PGsl</i>
G74	G	Najazd na punkt odniesienia	s	+		<i>PGsl</i>
G75	G	Ruch do punktu stałego	s	+		<i>PGsl</i>
G90 ⁶⁾	G	Absolutne podanie wymiaru	m/s	+		<i>PGsl</i>
G91	G	Przyrostowe podanie wymiaru	m/s	+		<i>PGsl</i>
G93	G	Zależny od czasu posuw 1/min	m	+		<i>PGsl</i>
G94 ⁶⁾	G	Posuw liniowy F w mm/min lub calach/min i stopniach/min	m	+		<i>PGsl</i>
G95	G	Posuw na obrót F w mm/obr. lub calach/obr.	m	+		<i>PGsl</i>
G96	G	Stała prędkość skrawania (jak przy G95) WŁ.	m	+		<i>PGsl</i>
G97	G	Stała prędkość skrawania (jak przy G95) WYŁ.	m	+		<i>PGsl</i>

Instrukcja	Art 1)	Znaczenie	W 2)	TP 3)	SA 4)	Opis patrz 5)
1) 2) 3) 4) 5) Objaśnienia patrz legenda (Strona 795).						
G110	G	Zaprogramowanie bieguna w stosunku do ostatniej zaprogramowanej pozycji zadanej	s	+		<i>PGsl</i>
G111	G	Zaprogramowanie bieguna w stosunku do punktu zerowego aktualnego układu współrzędnych obrabianego przedmiotu	s	+		<i>PGsl</i>
G112	G	Zaprogramowanie bieguna w stosunku do ostatniego obowiązującego bieguna	s	+		<i>PGsl</i>
G140 6)	G	Kierunek dosunięcia WAB ustalony przez G41/G42	m	+		<i>PGsl</i>
G141	G	Kierunek dosunięcia WAB na lewo od konturu	m	+		<i>PGsl</i>
G142	G	Kierunek dosunięcia WAB na prawo od konturu	m	+		<i>PGsl</i>
G143	G	Kierunek dosunięcia WAM zależnie od stycznej	m	+		<i>PGsl</i>
G147	G	Miękkie dosunięcie po prostej	s	+		<i>PGsl</i>
G148	G	Miękkie odsunięcie po prostej	s	+		<i>PGsl</i>
G153	G	Blokowanie aktualnych frame łącznie z frame bazowym	s	+		<i>PGsl</i>
G247	G	Miękkie dosunięcie po ćwierćokręgu	s	+		<i>PGsl</i>
G248	G	Miękkie odsunięcie po ćwierćokręgu	s	+		<i>PGsl</i>
G290 6)	G	Przełączenie na tryb SINUMERIK WŁ.	m	+		<i>FBWsl</i>
G291	G	Przełączenie na tryb ISO2/3 WŁ.	m	+		<i>FBWsl</i>
G331	G	Gwintowanie otworu bez oprawki kompensacyjnej, skok dodatni, obroty w prawo	m	+		<i>PGsl</i>
G332	G	Gwintowanie otworu bez oprawki kompensacyjnej, skok ujemny, obroty w lewo	m	+		<i>PGsl</i>
G340 6)	G	Blok dosunięcia przestrzennego (równocześnie na głębokości i w płaszczyźnie (linia spiralna)	m	+		<i>PGsl</i>
G341	G	Najpierw dosuw w osi prostopadłej (z), następnie dosunięcie w płaszczyźnie	m	+		<i>PGsl</i>
G347	G	Miękkie dosunięcie po półokręgu	s	+		<i>PGsl</i>
G348	G	Miękkie odsunięcie po półokręgu	s	+		<i>PGsl</i>
G450 6)	G	Okrag przejściowy	m	+		<i>PGsl</i>
G451	G	Punkt przecięcia równoległych	m	+		<i>PGsl</i>

Instrukcja	Art ¹⁾	Znaczenie	W ²⁾	TP ³⁾	SA ⁴⁾	Opis patrz ⁵⁾
1) 2) 3) 4) 5) Objaśnienia patrz legenda (Strona 795).						
G460 ⁶⁾	G	Włączenie nadzoru na kolizję dla bloku dosunięcia i odsunięcia	m	+		<i>PGsl</i>
G461	G	Wstawienie okręgu w bloku WRK	m	+		<i>PGsl</i>
G462	G	Wstawienie prostej w bloku WRK	m	+		<i>PGsl</i>
G500 ⁶⁾	G	Wyłączenie wszystkich ustawianych frame, frame bazowe są aktywne	m	+		<i>PGsl</i>
G505 ... G599	G	5 ... 99. Ustawiane przesunięcie punktu zerowego	m	+		<i>PGsl</i>
G601 ⁶⁾	G	Zmiana bloku przy zatrzymaniu dokładnym dokładnie	m	+		<i>PGsl</i>
G602	G	Zmiana bloku przy zatrzymaniu dokładnym zgrubnie	m	+		<i>PGsl</i>
G603	G	Zmiana bloku na końcu bloku IPO	m	+		<i>PGsl</i>
G621	G	Zwłoka na wszystkich narożach	m	+		<i>PGAsl</i> Zmniejszenie posuwu ze zwłoką na narożach (FENDNORM, G62, G621) (Strona 272)
G641	G	Tryb przechodzenia płynnego ze ścinaniem naroży według kryterium drogi (= programowana droga wygładzania)	m	+		<i>PGsl</i>
G642	G	Tryb przechodzenia płynnego ze ścinaniem naroży przy zachowaniu zdefiniowanych tolerancji	m	+		<i>PGsl</i>
G643	G	Tryb przechodzenia płynnego ze ścinaniem naroży przy zachowaniu zdefiniowanych tolerancji (wewnętrznie w bloku)	m	+		<i>PGsl</i>
G644	G	Tryb przechodzenia płynnego ze ścinaniem naroży z maksymalnie możliwą dynamiką	m	+		<i>PGsl</i>
G645	G	Tryb przechodzenia płynnego ze ścinaniem naroży i stycznymi przejściami między blokami przy dotrzymaniu zdefiniowanych tolerancji	m	+		<i>PGsl</i>
G700	G	Całowe podawanie wymiarów dla danych geometrycznych i technologicznych (długości, posuw)	m	+	+	<i>PGsl</i>
G710 ⁶⁾	G	Metryczne podawanie wymiarów dla danych geometrycznych i technologicznych (długości, posuw)	m	+	+	<i>PGsl</i>

Instrukcja	Art 1)	Znaczenie	W 2)	TP 3)	SA 4)	Opis patrz 5)
1) 2) 3) 4) 5) Objaśnienia patrz legenda (Strona 795).						
G810 ⁶⁾ , ..., G819	G	Grupa G zarezerwowana dla użytkownika OEM		+		<i>PGAs/</i> Funkcje specjalne dla użytkownika OEM (OMA1 ... OMA5, OEMIPO1, OEMIPO2, G810 ... G829) (Strona 271)
G820 ⁶⁾ , ..., G829	G	Grupa G zarezerwowana dla użytkownika OEM		+		<i>PGAs/</i> Funkcje specjalne dla użytkownika OEM (OMA1 ... OMA5, OEMIPO1, OEMIPO2, G810 ... G829) (Strona 271)
G931	G	Zadanie posuwu przez czas ruchu	m	+		
G942	G	Zamrożenie posuwu liniowego i stałej prędkości skrawania lub prędkości obrotowej wrzeciona	m	+		
G952	G	Zamrożenie posuwu na obrót i stałej prędkości skrawania albo prędkości obrotowej wrzeciona	m	+		
G961	G	Stała prędkość skrawania i posuw liniowy	m	+		<i>PGs/</i>
G962	G	Posuw liniowy albo posuw na obrót i stała prędkość skrawania	m	+		<i>PGs/</i>
G971	G	Zamrożenie prędkości obrotowej wrzeciona i posuw liniowy	m	+		<i>PGs/</i>
G972	G	Zamrożenie posuwu liniowego lub posuwu na obrót i stałej prędkości obrotowej wrzeciona	m	+		<i>PGs/</i>
G973	G	Posuw na obrót bez ograniczenia prędkości obrotowej wrzeciona	m	+		<i>PGs/</i>
GEOAX	P	Przyporządkowanie nowych osi kanałowych do osi geometrycznych 1 - 3		+	-	<i>PGAs/</i> Przełączalne osie geometryczne (GEOAX) (Strona 593)
GET	P	Zamiana udostępnionych osi między kanałami		+	+	<i>PGAs/</i> Zamiana osi, zamiana wrzecion (RELEASE, GET, GETD) (Strona 136)
GETACTT	F	Określa aktywne narzędzie z grupy narzędzi o takiej samej nazwie		+	-	<i>FBWs/</i>
GETACTTD	F	Określa do absolutnego numeru D przynależny numer T		+	-	<i>PGAs/</i> Dowolne nadawanie numerów D: Określenie numeru T do zadanego numeru D (GETACTTD) (Strona 432)
GETD	P	Zamiana osi bezpośrednio między kanałami		+	-	<i>PGAs/</i> Zamiana osi, zamiana wrzecion (RELEASE, GET, GETD) (Strona 136)
GETDNO	F	Dostarcza numer D ostrza (CE) narzędzia (T)		+	-	<i>PGAs/</i> Dowolne nadawanie numerów D: Zmiana nazw numerów D (GETDNO, SETDNO) (Strona 431)

Instrukcja	Art ¹⁾	Znaczenie	W ²⁾	TP ³⁾	SA ⁴⁾	Opis patrz ⁵⁾
1) 2) 3) 4) 5) Objaśnienia patrz legenda (Strona 795).						
GETEXET	P	Odczyt założonego numeru T		+	-	<i>FBWsl</i>
GETFREELOC	P	Szukanie wolnego miejsca w magazynach dla danego narzędzia		+	-	<i>FBWsl</i>
GETSELT	P	Dostarczenie wstępnie wybranego numeru T		+	-	<i>FBWsl</i>
GETT	F	Określenie numeru T do nazwy narzędzia		+	-	<i>FBWsl</i>
GETTCOR	F	Odczyt długości narzędzia lub komponentów długości narzędzia		+	-	<i>FB1sl (W1)</i>
GETTENV	F	Odczyt numerów T, D i DL		+	-	<i>FB1sl (W1)</i>
GETVARAP	F	Odczyt prawa dostępu do zmiennej systemowej/użytkownika		+	-	<i>PGAsl</i> Odczyt wartości absolutnych/typu danych (GETVARPHU, GETVARAP, GETVARLIM, GETVARDFT, GETVARTYP) (Strona 57)
GETVARDFT	F	Odczyt wartości standardowej zmiennej systemowej/użytkownika		+	-	<i>PGAsl</i> Odczyt wartości absolutnych/typu danych (GETVARPHU, GETVARAP, GETVARLIM, GETVARDFT, GETVARTYP) (Strona 57)
GETVARLIM	F	Odczyt wartości granicznych zmiennej systemowej/użytkownika		+	-	<i>PGAsl</i> Odczyt wartości absolutnych/typu danych (GETVARPHU, GETVARAP, GETVARLIM, GETVARDFT, GETVARTYP) (Strona 57)
GETVARPHU	F	Odczyt jednostki fizycznej zmiennej systemowej/użytkownika		+	-	<i>PGAsl</i> Odczyt wartości absolutnych/typu danych (GETVARPHU, GETVARAP, GETVARLIM, GETVARDFT, GETVARTYP) (Strona 57)
GETVARTYP	F	Odczyt typu danych zmiennej systemowej/użytkownika		+	-	<i>PGAsl</i> Odczyt wartości absolutnych/typu danych (GETVARPHU, GETVARAP, GETVARLIM, GETVARDFT, GETVARTYP) (Strona 57)
GOTO	K	Instrukcja skoku najpierw do przodu a następnie do tyłu (kierunek najpierw do końca a następnie do początku programu)		+		<i>PGAsl</i> Skoki w programie do znaczników skoku (GOTOB, GOTOF, GOTO, GOTOC) (Strona 99)
GOTOB	K	Instrukcja skoku do tyłu (kierunek do początku programu)		+		<i>PGAsl</i> Skoki w programie do znaczników skoku (GOTOB, GOTOF, GOTO, GOTOC) (Strona 99)

Instrukcja	Art 1)	Znaczenie	W 2)	TP 3)	SA 4)	Opis patrz 5)
1) 2) 3) 4) 5) Objaśnienia patrz legenda (Strona 795).						
GOTOC	K	Jak GOTO, ale ukrywanie alarmu 14080 "Cel skoku nie został znaleziony"		+		<i>PGAs/</i> Skoki w programie do znaczników skoku (GOTOB, GOTOF, GOTO, GOTOC) (Strona 99)
GOTOF	K	Instrukcja skoku do przodu (kierunek do końca programu)		+		<i>PGAs/</i> Skoki w programie do znaczników skoku (GOTOB, GOTOF, GOTO, GOTOC) (Strona 99)
GOTOS	K	Skok powrotny do początku programu		+		<i>PGAs/</i> Skok do początku programu (GOTOS) (Strona 98)
GP	K	Słowo kluczowe do pośredniego programowania atrybutów pozycji		+		<i>PGAs/</i> Programowanie pośrednie atrybutów pozycji (GP) (Strona 66)
GWPSOF	P	Cofnięcie wyboru stałej prędkości obwodowej ściernicy (SUG)	s	+	-	<i>PGs/</i>
GWPSON	P	Wybór stałej prędkości obwodowej ściernicy (SUG)	s	+	-	<i>PGs/</i>
H...	A	Wyprowadzenie funkcji pomocniczej do PLC		+	+	<i>PGs/FB1s/ (H2)</i>
HOLES1	C	Szereg otworów		+		<i>PGAs/</i> Szereg otworów - HOLES1 (Strona 666)
HOLES2	C	Okrąg otworów		+		<i>PGAs/</i>
I	A	Parametr interpolacji	s	+		<i>PGs/</i>
I1	A	Współrzędna punktu pośredniego	s	+		<i>PGs/</i>
IC	K	Przyrostowe podawanie wymiarów	s	+		<i>PGs/</i>
ICYCOF	P	Wykonanie wszystkich bloków jednego cyklu technologicznego według ICYCOF w jednym takcie IPO		+	+	<i>FBSY</i>
ICYCON	P	Wykonanie każdego bloku cyklu technologicznego według ICYCON w oddzielnym takcie IPO		+	+	<i>FBSY</i>
ID	K	Oznaczenie dla modalnych akcji synchronicznych	m	-	+	<i>FBSY</i>
IDS	K	Oznaczenie dla modalnych statycznych akcji synchronicznych		-	+	<i>FBSY</i>
IF	K	Rozpoczęcie skoku warunkowego w programie obróbki / cyklu technologicznym		+	+	<i>PGAs/</i> Warunkowa instrukcja i rozgałęzienie (IF, ELSE, ENDIF) (Strona 112)

Instrukcja	Art ¹⁾	Znaczenie	W ²⁾	TP ³⁾	SA ⁴⁾	Opis patrz ⁵⁾
1) 2) 3) 4) 5) Objaśnienia patrz legenda (Strona 795).						
INDEX	F	Określenie indeksu znaku w wejściowym łańcuchu znaków		+	-	<i>PGAs/</i> Szukanie znaku/łańcucha znaków w łańcuchu znaków (INDEX, RINDEX, MINDEX, MATCH) (Strona 86)
INICF	K	Inicjalizacja zmiennych przy NewConfig		+		<i>PGAs/</i> Definicja zmiennych użytkownika (DEF) (Strona 24)
INIPO	K	Inicjalizacja zmiennych przy PowerOn		+		<i>PGAs/</i> Definicja zmiennych użytkownika (DEF) (Strona 24)
INIRE	K	Inicjalizacja zmiennych przy Reset		+		<i>PGAs/</i> Definicja zmiennych użytkownika (DEF) (Strona 24)
INIT	P	Wybór określonego programu NC do wykonywania w określonym kanale		+	-	<i>PGAs/</i> Koordynacja programu (INIT, START, WAITM, WAITMC, WAITE, SETM, CLEARM) (Strona 118)
INITIAL		Utworzenie pliku INI po wszystkich zakresach		+		<i>PGAs/</i> Pamięć robocza (CHANDATA, COMPLETE, INITIAL) (Strona 217)
INT	K	Typ danych: Wartość całkowitoliczbowa ze znakiem		+		<i>PGAs/</i> Definicja zmiennych użytkownika (DEF) (Strona 24)
INTERSEC	F	Obliczenie punktu przecięcia między dwoma elementami konturu		+	-	<i>PGAs/</i> Obliczenie punktu przecięcia między dwoma elementami konturu (INTERSEC) (Strona 643)
INTTOAX	F	Zmiana typu danych zmiennej osiowej z INT na AXIS		+	-	<i>PGAs/</i> Konwersje explicite typu danych (AXTOINT, INTTOAX) (Strona 54)
INVCCW	G	Ruch po ewolwencji, przeciwnie do ruchu wskazówek zegara	m	+		<i>PGs/</i>
INVCW	G	Ruch po ewolwencji, zgodnie z ruchem wskazówek zegara	m	+		<i>PGs/</i>
INVFRAME	F	Obliczenie frame odwrotnego z frame		+	-	<i>FB1s/ (K2)</i>
IP	K	Zmienny parametr interpolacji		+		<i>PGAs/</i> Programowanie pośrednie (Strona 62)
IPOBRKA	P	Kryterium ruchu od punktu początkowego charakterystyki hamowania	m	+	+	
IPOENDA	K	Koniec ruchu przy osiągnięciu "Stop IPO"	m	+		<i>PGAs/</i> Programowane kryterium końca ruchu (FINEA, COARSEA, IPOENDA, IPOBRKA, ADISPOSA) (Strona 273)

Instrukcja	Art 1)	Znaczenie	W 2)	TP 3)	SA 4)	Opis patrz 5)
1) 2) 3) 4) 5) Objasnienia patrz legenda (Strona 795).						
IPTRLOCK	P	Zamrożenie początku niezdatnego do szukania segmentu programu na najbliższy blok funkcjonalny maszyny.	m	+	-	<i>PGAs!</i> Zablokowanie miejsca w programie dla SERUPRO (IPTRLOCK, IPTRUNLOCK) (Strona 480)
IPTRUNLOCK	P	Ustawienie końca niezdatnego do szukania segmentu programu na aktualny blok w chwili przerwania.	m	+	-	<i>PGAs!</i> Zablokowanie miejsca w programie dla SERUPRO (IPTRLOCK, IPTRUNLOCK) (Strona 480)
ISAXIS	F	Sprawdzenie, czy podana jako parametr oś geometryczna 1 istnieje		+	-	<i>PGAs!</i> Funkcje osi (AXNAME, AX, SPI, AXTOSPI, ISAXIS, AXSTRING, MODAXVAL) (Strona 591)
ISD	A	Głębokość wgłębienia	m	+		<i>PGAs!</i> Uaktywnienie korekcji narzędzia 3D (CUT3DC, CUT3DF, CUT3DFS, CUT3DFF, ISD) (Strona 410)
ISFILE	F	Sprawdzenie, czy plik istnieje w pamięci użytkownika NCK		+	-	<i>PGAs!</i> Sprawdzenie istnienia pliku (ISFILE) (Strona 153)
ISNUMBER	F	Sprawdzenie, czy wejściowy łańcuch znaków można zamienić w liczbę		+	-	<i>PGAs!</i> Konwersja typu z STRING (NUMBER, ISNUMBER, AXNAME) (Strona 82)
ISOCALL	K	Pośrednie wywołanie programu programowanego w języku ISO		+		<i>PGAs!</i> Pośrednie wywołanie programu programowanego w języku ISO (ISOCALL) (Strona 199)
ISVAR	F	Sprawdzenie, czy przekazany parametr zawiera zmienną znaną w NC		+	-	<i>PGAs!</i> Sprawdzenie istnienia zmiennej (ISVAR) (Strona 55)
J	A	Parametr interpolacji	s	+		<i>PGs!</i>
J1	A	Współrzędna punktu pośredniego	s	+		<i>PGs!</i>
JERKA	P	Uaktywnienie ustawionego przez MD przyśpieszenia dla programowanych osi		+	-	
JERKLIM	K	Zmniejszenie lub zwiększenie maksymalnego osiowego przyśpieszenia drugiego stopnia	m	+		<i>PGAs!</i> Procentowa korekcja przyśpieszenia drugiego stopnia (JERKLIM) (Strona 492)
JERKLIMA	K	Zmniejszenie lub zwiększenie maksymalnego osiowego przyśpieszenia drugiego stopnia	m	+	+	<i>PGAs!</i> Sterowanie przyśpieszeniem w przypadku osi nadążnych (VELOLIMA, ACCLIMA, JERKLIMA) (Strona 465)
K	A	Parametr interpolacji	s	+		<i>PGs!</i>
K1	A	Współrzędna punktu pośredniego	s	+		<i>PGs!</i>
KONT	G	Obejście konturu przy korekcji narzędzia	m	+		<i>PGs!</i>

Instrukcja	Art ¹⁾	Znaczenie	W ²⁾	TP ³⁾	SA ⁴⁾	Opis patrz ⁵⁾
1) 2) 3) 4) 5) Objaśnienia patrz legenda (Strona 795).						
KONTC	G	Dosunięcie/odsunięcie po krzywej wielomianowej o stałej krzywiznie	m	+		<i>PGsI</i>
KONTT	G	Dosunięcie/odsunięcie po krzywej wielomianowej o stałej pochodnej	m	+		<i>PGsI</i>
L	A	Numer podprogramu	s	+	+	<i>PGAsI</i> Wywołanie podprogramu bez przekazania parametrów (Strona 189)
LEAD	A	Kąt wyprzedzenia 1. Orientacja narzędzia 2. Wielomiany orientacji	m	+		<i>PGAsI</i> Programowanie orientacji narzędzia (A..., B..., C..., LEAD, TILT) (Strona 319)
LEADOF	P	Sprężenie osiowe wartości wiodącej WYŁ.		+	+	<i>PGAsI</i> Osiowe sprężenie wartości wiodącej (LEADON, LEADOF) (Strona 526)
LEADON	P	Sprężenie osiowe wartości wiodącej WŁ.		+	+	<i>PGAsI</i> Osiowe sprężenie wartości wiodącej (LEADON, LEADOF) (Strona 526)
LENTOAX	F	Dostarcza informacji o przyporządkowaniu długości aktywnego narzędzia L1, L2 i L3 do odciętej, rzędnej i aplikaty.		+	-	<i>FB1sI (W1)</i>
LFOF ⁶⁾	G	Szybkie wycofanie przy nacinaniu gwintu WYŁ.	m	+		<i>PGsI</i>
LFON	G	Szybkie wycofanie przy nacinaniu gwintu WŁ.	m	+		<i>PGsI</i>
LFPOS	G	Wycofanie osi podanej przy pomocy POLFMASK lub POLFMLIN do absolutnej pozycji osi zaprogramowanej przy pomocy POLF	m	+		<i>PGsI</i>
LFTXT ⁶⁾	G	Płaszczyzna ruchu wycofania przy szybkim cofnięciu jest określana ze stycznej do toru i aktualnego kierunku narzędzia	m	+		<i>PGsI</i>
LFWP	G	Płaszczyzna ruchu wycofania przy szybkim cofnięciu jest określana przez aktualną płaszczyznę roboczą (G17/G18/G19)	m	+		<i>PGsI</i>
LIFTFAST	K	Szybkie cofnięcie		+		<i>PGsI</i> Szybkie cofnięcie od konturu (SETINT LIFTFAST, ALF) (Strona 129)
LIMS	K	Ograniczenie prędkości obrotowej przy G96/G961 i G97	m	+		<i>PGsI</i>

Instrukcja	Art 1)	Znaczenie	W 2)	TP 3)	SA 4)	Opis patrz 5)
1) 2) 3) 4) 5) Objaśnienia patrz legenda (Strona 795).						
LLI	K	Dolna wartość graniczna zmiennych		+		<i>PGAs/</i> Atrybut: wartości graniczne (LLI, ULI) (Strona 36)
LN	F	Logarytm naturalny		+	+	<i>PGAs/</i> Funkcje arytmetyczne (Strona 70)
LOCK	P	Zablokowanie akcji synchronicznej z ID (zatrzymanie cyklu technologicznego)		-	+	<i>FBSY</i>
LONGHOLE	C	Otwór podłużny		+		<i>PGAs/</i> Otwór podłużny - LONGHOLE (Strona 689)
LOOP	K	Wprowadzenie pętli bez końca		+		<i>PGAs/</i> Pętla programowa bez końca (LOOP, ENDLOOP) (Strona 113)
M0		Zatrzymanie programowane		+	+	<i>PGs/</i>
M1		Zatrzymanie warunkowe		+	+	<i>PGs/</i>
M2		Koniec programu, program główny (jak M30)		+	+	<i>PGs/</i>
M3		Kierunek obrotów wrzeciona w prawo		+	+	<i>PGs/</i>
M4		Kierunek obrotów wrzeciona w lewo		+	+	<i>PGs/</i>
M5		Wrzeczono stop		+	+	<i>PGs/</i>
M6		Wymiana narzędzia		+	+	<i>PGs/</i>
M17		Koniec podprogramu		+	+	<i>PGs/</i>
M19		Pozycjonowanie wrzeciona na pozycji wpisanej w SD43240		+	+	<i>PGs/</i>
M30		Koniec programu, program główny (jak M2)		+	+	<i>PGs/</i>
M40		Automatyczne przełączanie przekładni		+	+	<i>PGs/</i>
M41 ... M45		Stopień przekładni 1 ... 5		+	+	<i>PGs/</i>
M70		Przejsie wrzeciona na pracę jako oś		+	+	<i>PGs/</i>
MASLDEF	P	Zdefiniowanie zespołu osi master/slave		+	+	<i>PGAs/</i> Sprzężenie Master/Slave (MASLDEF, MASLDEL, MASLON, MASLOF, MASLOFS) (Strona 557)
MASLDEL	P	Rozłączenie zespołu osi master/slave i skasowanie definicji zespołu		+	+	<i>PGAs/</i> Sprzężenie Master/Slave (MASLDEF, MASLDEL, MASLON, MASLOF, MASLOFS) (Strona 557)

Instrukcja	Art ¹⁾	Znaczenie	W ²⁾	TP ³⁾	SA ⁴⁾	Opis patrz ⁵⁾
1) 2) 3) 4) 5) Objaśnienia patrz legenda (Strona 795).						
MASLOF	P	Wyłączenie sprzężenia tymczasowego		+	+	<i>PGAs/</i> Sprzężenie Master/Slave (MASLDEF, MASLDEL, MASLON, MASLOF, MASLOFS) (Strona 557)
MASLOFS	P	Wyłączenie sprzężenia tymczasowego z automatycznym zatrzymaniem osi slave		+	+	<i>PGAs/</i> Sprzężenie Master/Slave (MASLDEF, MASLDEL, MASLON, MASLOF, MASLOFS) (Strona 557)
MASLON	P	Włączenie sprzężenia tymczasowego		+	+	<i>PGAs/</i> Sprzężenie Master/Slave (MASLDEF, MASLDEL, MASLON, MASLOF, MASLOFS) (Strona 557)
MATCH	F	Szukanie łańcucha znaków w łańcuchu znaków		+	-	<i>PGAs/</i> Szukanie znaku/łańcucha znaków w łańcuchu znaków (INDEX, RINDEX, MINDEX, MATCH) (Strona 86)
MAXVAL	F	Większa wartość z dwóch zmiennych (arytmetyczna funkcja)		+	+	<i>PGAs/</i> Minimum, maksimum i zakres zmiennych (MINVAL, MAXVAL i BOUND) (Strona 77)
MCALL	K	Modalne wywołanie podprogramu		+		<i>PGAs/</i> Modalne wywołanie podprogramu (MCALL) (Strona 195)
MEAC	K	Ciągły pomiar w osi bez skasowania pozostałej drogi	s	+	+	<i>PGAs/</i> Pomiar osiowy (MEASA, MEAWA, MEAC) (Opcja) (Strona 260)
MEAFRAME	F	Obliczenie frame z punktów pomiarowych		+	-	<i>PGAs/</i> Obliczenie frame z 3 punktów pomiarowych w przestrzeni (MEAFRAME) (Strona 294)
MEAS	A	Pomiar z kasowaniem pozostałej drogi	s	+		<i>PGAs/</i> Pomiar przy pomocy sondy przełączającej, MEAS, MEAW (Strona 257)
MEASA	K	Pomiar w osi z kasowaniem pozostałej drogi	s	+	+	<i>PGAs/</i> Pomiar osiowy (MEASA, MEAWA, MEAC) (Opcja) (Strona 260)
MEASURE	F	Metoda obliczania przy pomiarze obrabianego przedmiotu i narzędzia		+	-	<i>FB1s/ (M5)</i> Pomiar przy pomocy sondy przełączającej, MEAS, MEAW (Strona 257)
MEAW	A	Pomiar bez kasowania pozostałej drogi	s	+		<i>PGAs/</i> Pomiar przy pomocy sondy przełączającej, MEAS, MEAW (Strona 257)

Instrukcja	Art 1)	Znaczenie	W 2)	TP 3)	SA 4)	Opis patrz 5)
1) 2) 3) 4) 5) Objaśnienia patrz legenda (Strona 795).						
MEAWA	K	Pomiar w osi bez skasowania pozostałej drogi	s	+	+	<i>PGAs/</i> Pomiar osiowy (MEASA, MEAWA, MEAC) (Opcja) (Strona 260)
MI	K	Dostęp do danych frame: Lustrzane odbicie		+		<i>PGAs/</i> Odczyt i zmiana komponentów frame (TR, FI, RT, SC, MI) (Strona 287)
MINDEX	F	Określenie indeksu znaku w wejściowym łańcuchu znaków		+	-	<i>PGAs/</i> Szukanie znaku/łańcucha znaków w łańcuchu znaków (INDEX, RINDEX, MINDEX, MATCH) (Strona 86)
MINVAL	F	Mniejsza wartość z dwóch zmiennych (arytmetyczna funkcja)		+	+	<i>PGAs/</i> Minimum, maksimum i zakres zmiennych (MINVAL, MAXVAL i BOUND) (Strona 77)
MIRROR	G	Programowane lustrzane odbicie	s	+		<i>PGAs/</i>
MMC	P	Interaktywne wywołanie z programu obróbki okna dialogowego na HMI		+	-	<i>PGAs/</i> Interaktywne wywołanie okna z programu obróbki (MMC) (Strona 607)
MOD	K	Dzielenie modulo		+		<i>PGAs/</i> Funkcje arytmetyczne (Strona 70)
MODAXVAL	F	Określenie pozycji modulo osi obrotowej modulo		+	-	<i>PGAs/</i> Funkcje osi (AXNAME, AX, SPI, AXTOPI, ISAXIS, AXSTRING, MODAXVAL) (Strona 591)
MOV	K	Uruchomienie osi pozycjonowania		-	+	<i>FBSY</i>
MOVT	A	Podanie punktu końcowego dla ruchu postępowego w kierunku narzędzia				<i>FB1(K2)</i>
MSG	P	Komunikaty programowane	m	+	-	<i>PGs/</i>
MVTOOL	P	Polecenie językowe do wykonania ruchu narzędzia		+	-	<i>FBWs/</i>
N	A	Numer bloku pomocniczego NC		+		<i>PGs/</i>
NAMETOINT	F	Określenie indeksu zmiennych systemowych		+		<i>PGAs/</i> Określanie indeksu poprzez nazwy (NAMETOINT) (Strona 384)
NCK	K	Specyfikacja zakresu obowiązywania danych		+		<i>PGAs/</i> Definicja zmiennych użytkownika (DEF) (Strona 24)
NEWCONF	P	Przejęcie zmienionych danych maszynowych (odpowiada "ustawieniu działania danej maszyny")		+	-	<i>PGAs/</i> Ustawienie działania danych maszynowych (NEWCONF) (Strona 142)
NEWMT	F	Utworzenie nowego Multitool		+	-	<i>FBWs/</i>
NEWT	F	Utworzenie nowego narzędzia		+	-	<i>FBWs/</i>

19.1 Instrukcje

Instrukcja	Art ¹⁾	Znaczenie	W ²⁾	TP ³⁾	SA ⁴⁾	Opis patrz ⁵⁾
1) 2) 3) 4) 5) Objaśnienia patrz legenda (Strona 795).						
NORM ⁶⁾	G	Normalne ustawienie w początkowym, końcowym punkcie przy korekcy narzędzia	m	+		<i>PGAs/</i>
NOT	K	Logiczne NIE (negacja)		+		<i>PGAs/</i> Operacje porównania i operacje logiczne (Strona 73)
NPROT	P	Specyficzny dla maszyny obszar ochrony WŁ./WYŁ.		+	-	<i>PGAs/</i> Uaktywnienie / wyłączenie aktywności obszarów ochrony (CPROT, NPROT) (Strona 224)
NPROTDEF	P	Definicja obszaru ochrony specyficznego dla maszyny		+	-	<i>PGAs/</i> Ustalenie obszarów ochrony (CPROTDEF, NPROTDEF) (Strona 221)
NUMBER	F	Zamiana wejściowego łańcucha znaków na liczbę		+	-	<i>PGAs/</i> Konwersja typu z STRING (NUMBER, ISNUMBER, AXNAME) (Strona 82)
OEMIPO1	G	Interpolacja OEM 1	m	+		<i>PGAs/</i> Funkcje specjalne dla użytkownika OEM (OMA1 ... OMA5, OEMIPO1, OEMIPO2, G810 ... G829) (Strona 271)
OEMIPO2	G	Interpolacja OEM 2	m	+		<i>PGAs/</i> Funkcje specjalne dla użytkownika OEM (OMA1 ... OMA5, OEMIPO1, OEMIPO2, G810 ... G829) (Strona 271)
OF	K	Słowo kluczowe w rozgałęzieniu CASE		+		<i>PGAs/</i> Rozgałęzienie programu (CASE ... OF ... DEFAULT ...) (Strona 102)
OFFN	A	Naddatek do zaprogramowanego konturu	m	+		<i>PGAs/</i>
OMA1	A	Adres OEM 1	m	+		<i>PGAs/</i> Funkcje specjalne dla użytkownika OEM (OMA1 ... OMA5, OEMIPO1, OEMIPO2, G810 ... G829) (Strona 271)
OMA2	A	Adres OEM 2	m	+		<i>PGAs/</i> Funkcje specjalne dla użytkownika OEM (OMA1 ... OMA5, OEMIPO1, OEMIPO2, G810 ... G829) (Strona 271)
OMA3	A	Adres OEM 3	m	+		<i>PGAs/</i> Funkcje specjalne dla użytkownika OEM (OMA1 ... OMA5, OEMIPO1, OEMIPO2, G810 ... G829) (Strona 271)
OMA4	A	Adres OEM 4	m	+		<i>PGAs/</i> Funkcje specjalne dla użytkownika OEM (OMA1 ... OMA5, OEMIPO1, OEMIPO2, G810 ... G829) (Strona 271)

Instrukcja	Art 1)	Znaczenie	W 2)	TP 3)	SA 4)	Opis patrz 5)
1) 2) 3) 4) 5) Objaśnienia patrz legenda (Strona 795).						
OMA5	A	Adres OEM 5	m	+		<i>PGAs/</i> Funkcje specjalne dla użytkownika OEM (OMA1 ... OMA5, OEMIPO1, OEMIPO2, G810 ... G829) (Strona 271)
OR	K	Operator logiczny, połączenie logiczne LUB		+		<i>PGAs/</i> Operacje porównania i operacje logiczne (Strona 73)
ORIXES	G	Interpolacja liniowa osi maszyny albo osi orientacji	m	+		<i>PGAs/</i> Programowanie osi orientacji (ORIXES, ORIVECT, ORIEULER, ORIRPY, ORIRPY2, ORIVIRT1, ORIVIRT2) (Strona 329)
ORIXPOS	G	Kąt orientacji przez wirtualne osie orientacji z pozycjami osi obrotowych	m	+		<i>PGAs/</i> Programowanie orientacji narzędzia (A..., B..., C..., LEAD, TILT) (Strona 319)
ORIC 6)	G	Zmiany orientacji na narożach zewnętrznych są nakładane na wstawiony blok okręgu	m	+		<i>PGAs/</i> Orientacja narzędzia (ORIC, ORID, OSOF, OSC, OSS, OSSE, ORIS, OSD, OST) (Strona 424)
ORICONCCW	G	Interpolacja na kołowej powierzchni poboczniczy przeciwie do ruchu wskazówek zegara	m	+		<i>PGAs/FB3s/ (F3)</i> Programowanie orientacji wzdłuż poboczniczy stożka (ORIPLANE, ORICONCW, ORICONCCW, ORICONTO, ORICONIO) (Strona 332)
ORICONCW	G	Interpolacja na kołowej powierzchni poboczniczy w kierunku ruchu wskazówek zegara	m	+		<i>PGAs/FB3s/ (F4)</i> Programowanie orientacji wzdłuż poboczniczy stożka (ORIPLANE, ORICONCW, ORICONCCW, ORICONTO, ORICONIO) (Strona 332)
ORICONIO	G	Interpolacja na kołowej powierzchni poboczniczy z podaniem orientacji pośredniej	m	+		<i>PGAs/FB3s/ (F4)</i> Programowanie orientacji wzdłuż poboczniczy stożka (ORIPLANE, ORICONCW, ORICONCCW, ORICONTO, ORICONIO) (Strona 332)
ORICONTO	G	Interpolacja na kołowej powierzchni poboczniczy w przejściu stycznym (podanie orientacji końcowej)	m	+		<i>PGAs/FB3s/ (F5)</i> Programowanie orientacji wzdłuż poboczniczy stożka (ORIPLANE, ORICONCW, ORICONCCW, ORICONTO, ORICONIO) (Strona 332)
ORICURVE	G	Interpolacja orientacji z zadaniem ruchu dwóch punktów styku narzędzia	m	+		<i>PGAs/FB3s/ (F6)</i> Zadanie orientacji dwóch punktów kontaktowych (ORICURVE, PO[XH]=, PO[YH]=, PO[ZH]=) (Strona 335)

Instrukcja	Art ¹⁾	Znaczenie	W ²⁾	TP ³⁾	SA ⁴⁾	Opis patrz ⁵⁾
1) 2) 3) 4) 5) Objaśnienia patrz legenda (Strona 795).						
ORID	G	Zmiany orientacji są wykonywane przed blokiem okręgu	m	+		<i>PGAs/</i> Orientacja narzędzia (ORIC, ORID, OSOF, OSC, OSS, OSSE, ORIS, OSD, OST) (Strona 424)
ORIEULER ⁶⁾	G	Kąt orientacji przez kąt Eulera	m	+		<i>PGAs/</i> Programowanie osi orientacji (ORIAxes, ORIVect, ORIEULER, ORIRPY, ORIRPY2, ORIVIRT1, ORIVIRT2) (Strona 329)
ORIMKS	G	Orientacja narzędzia w układzie współrzędnych maszyny	m	+		<i>PGAs/</i> Odniesienie osi orientacji (ORIWKS, ORIMKS) (Strona 327)
ORIPATH	G	Orientacja narzędzia odniesiona do toru	m	+		<i>PGAs/</i> Odniesiony do toru obrót orientacji narzędzia (ORIPATH, ORIPATHS, kąt obrotu) (Strona 343)
ORIPATHS	G	Orientacja narzędzia w odniesieniu do toru, załamanie w przebiegu orientacji jest wygładzane	m	+		<i>PGAs/</i> Odniesiony do toru obrót orientacji narzędzia (ORIPATH, ORIPATHS, kąt obrotu) (Strona 343)
ORIPLANE	G	Interpolacja w płaszczyźnie (odpowiada ORIVect) interpolacja wielkiego okręgu	m	+		<i>PGAs/</i> Programowanie orientacji wzdłuż pobocznicy stożka (ORIPLANE, ORICONCW, ORICONCCW, ORICONTO, ORICONIO) (Strona 332)
ORIRESET	P	Położenie podstawowe orientacji narzędzia z max 3 osiami orientacji		+	-	<i>PGAs/</i> Warianty programowania orientacji i położenie podstawowe (OTIRESET) (Strona 317)
ORIROTA ⁶⁾	G	Kąt obrotu do kierunku obrotu zadanego absolutnie	m	+		<i>PGAs/</i> Obroty orientacji narzędzia (ORIROTA, ORIROTR, ORIROTT, ORIROTC, THETA) (Strona 339)
ORIROTC	G	Styczny wektor obrotu do stycznej do toru	m	+		<i>PGAs/</i> Obroty orientacji narzędzia (ORIROTA, ORIROTR, ORIROTT, ORIROTC, THETA) (Strona 339)
ORIROTR	G	Kąt obrotu w stosunku do płaszczyzny między orientacją startową i końcową	m	+		<i>PGAs/</i> Obroty orientacji narzędzia (ORIROTA, ORIROTR, ORIROTT, ORIROTC, THETA) (Strona 339)
ORIROTT	G	Kąt obrotu w stosunku do zmiany wektora orientacji	m	+		<i>PGAs/</i> Obroty orientacji narzędzia (ORIROTA, ORIROTR, ORIROTT, ORIROTC, THETA) (Strona 339)

Instrukcja	Art 1)	Znaczenie	W 2)	TP 3)	SA 4)	Opis patrz 5)
1) 2) 3) 4) 5) Objaśnienia patrz legenda (Strona 795).						
ORIRPY	G	Kąt orientacji poprzez kąt RPY (XYZ)	m	+		<i>PGAs!</i> Programowanie osi orientacji (ORIAxes, ORIVect, ORIEuler, ORIRPY, ORIRPY2, ORIVIRT1, ORIVIRT2) (Strona 329)
ORIRPY2	G	Kąt orientacji poprzez kąt RPY (ZYX)	m	+		<i>PGAs!</i> Programowanie osi orientacji (ORIAxes, ORIVect, ORIEuler, ORIRPY, ORIRPY2, ORIVIRT1, ORIVIRT2) (Strona 329)
ORIS	A	Zmiana orientacji	m	+		<i>PGAs!</i> Orientacja narzędzia (ORIC, ORID, OSOF, OSC, OSS, OSSE, ORIS, OSD, OST) (Strona 424)
ORISOF 6)	G	Wyglądanie przebiegu orientacji WŁ.	m	+		<i>PGAs!</i> Wyglądanie przebiegu orientacji (ORISON, ORISOF) (Strona 351)
ORISON	G	Wyglądanie przebiegu orientacji WŁ.	m	+		<i>PGAs!</i> Wyglądanie przebiegu orientacji (ORISON, ORISOF) (Strona 351)
ORIVECT 6)	G	Interpolacja dużego okręgu (identyczna z ORIPLANE)	m	+		<i>PGAs!</i> Programowanie osi orientacji (ORIAxes, ORIVect, ORIEuler, ORIRPY, ORIRPY2, ORIVIRT1, ORIVIRT2) (Strona 329)
ORIVIRT1	G	Kąt orientacji przez wirtualne osie orientacji (definicja 1)	m	+		<i>PGAs!</i> Programowanie osi orientacji (ORIAxes, ORIVect, ORIEuler, ORIRPY, ORIRPY2, ORIVIRT1, ORIVIRT2) (Strona 329)
ORIVIRT2	G	Kąt orientacji przez wirtualne osie orientacji (definicja 1)	m	+		<i>PGAs!</i> Programowanie osi orientacji (ORIAxes, ORIVect, ORIEuler, ORIRPY, ORIRPY2, ORIVIRT1, ORIVIRT2) (Strona 329)
ORIWKS 6)	G	Orientacja narzędzia w układzie współrzędnych obrabianego przedmiotu	m	+		<i>PGAs!</i> Odniesienie osi orientacji (ORIWKS, ORIMKS) (Strona 327)
OS	K	Ruch wahadłowy wł./wył.		+		<i>PGAs!</i> Ruch wahlwy asynchroniczny (OS, OSP1, OSP2, OST1, OST2, OSCtrl, OSNSC, OSE, OSB) (Strona 563)
OSB	K	Ruch wahlwy: Punkt startowy	m	+		<i>FB1s!</i> (P5)
OSC	G	Stałe wyglądanie orientacji narzędzia	m	+		<i>PGAs!</i> Orientacja narzędzia (ORIC, ORID, OSOF, OSC, OSS, OSSE, ORIS, OSD, OST) (Strona 424)

Instrukcja	Art ¹⁾	Znaczenie	W ²⁾	TP ³⁾	SA ⁴⁾	Opis patrz ⁵⁾
1) 2) 3) 4) 5) Objaśnienia patrz legenda (Strona 795).						
OSCILL	K	Oś: 1 - 3 osi dosuwu	m	+		<i>PGAs/</i> Ruch wahliwy sterowany przez akcje synchroniczne (OSCILL) (Strona 568)
OSCTRL	K	Opcje ruchu wahadłowego	m	+		<i>PGAs/</i> Ruch wahliwy asynchroniczny (OS, OSP1, OSP2, OST1, OST2, OSCTRL, OSNSC, OSE, OSB) (Strona 563)
OSD	G	Wygładzanie orientacji narzędzia przez zadanie drogi wygładzania przy pomocy SD	m	+		<i>PGAs/</i> Orientacja narzędzia (ORIC, ORID, OSOF, OSC, OSS, OSSE, ORIS, OSD, OST) (Strona 424)
OSE	K	Ruch wahadłowy punkt końcowy	m	+		<i>PGAs/</i> Ruch wahliwy asynchroniczny (OS, OSP1, OSP2, OST1, OST2, OSCTRL, OSNSC, OSE, OSB) (Strona 563)
OSNSC	K	Ruch wahliwy: liczba wyiskrzeń	m	+		<i>PGAs/</i> Ruch wahliwy asynchroniczny (OS, OSP1, OSP2, OST1, OST2, OSCTRL, OSNSC, OSE, OSB) (Strona 563)
OSOF ⁶⁾	G	Wygładzanie orientacji narzędzia WYŁ.	m	+		<i>PGAs/</i> Ruch wahliwy asynchroniczny (OS, OSP1, OSP2, OST1, OST2, OSCTRL, OSNSC, OSE, OSB) (Strona 563)
OSP1	K	Ruch wahliwy: lewy punkt nawrotny	m	+		<i>PGAs/</i> Ruch wahliwy asynchroniczny (OS, OSP1, OSP2, OST1, OST2, OSCTRL, OSNSC, OSE, OSB) (Strona 563)
OSP2	K	Ruch wahadłowy: prawy punkt nawrotny	m	+		<i>PGAs/</i> Ruch wahliwy asynchroniczny (OS, OSP1, OSP2, OST1, OST2, OSCTRL, OSNSC, OSE, OSB) (Strona 563)
OSS	G	Wygładzanie orientacji narzędzia na końcu bloku	m	+		<i>PGAs/</i> Orientacja narzędzia (ORIC, ORID, OSOF, OSC, OSS, OSSE, ORIS, OSD, OST) (Strona 424)
OSSE	G	Wygładzanie orientacji narzędzia na początku i końcu bloku	m	+		<i>PGAs/</i> Orientacja narzędzia (ORIC, ORID, OSOF, OSC, OSS, OSSE, ORIS, OSD, OST) (Strona 424)
OST	G	Wygładzanie orientacji narzędzia przez zadanie tolerancji kątowej w stopniach przy pomocy SD (maksymalne odchylenie od zaprogramowanego przebiegu orientacji)	m	+		<i>PGAs/</i> Orientacja narzędzia (ORIC, ORID, OSOF, OSC, OSS, OSSE, ORIS, OSD, OST) (Strona 424)

Instrukcja	Art 1)	Znaczenie	W 2)	TP 3)	SA 4)	Opis patrz 5)
1) 2) 3) 4) 5) Objaśnienia patrz legenda (Strona 795).						
OST1	K	Ruch wahliwy: punkt zatrzymania w lewym punkcie nawrotnym	m	+		<i>PGAs/</i> Ruch wahliwy asynchroniczny (OS, OSP1, OSP2, OST1, OST2, OSCTRL, OSNSC, OSE, OSB) (Strona 563)
OST2	K	Ruch wahliwy: punkt zatrzymania w prawym punkcie nawrotnym	m	+		<i>PGAs/</i> Ruch wahliwy asynchroniczny (OS, OSP1, OSP2, OST1, OST2, OSCTRL, OSNSC, OSE, OSB) (Strona 563)
OTOL	K	Tolerancja orientacji dla funkcji kompresora, wygładzanie orientacji i rodzaje wygładzania		+		<i>PGAs/</i> Programowana tolerancja konturu/orientacji (CTOL, OTOL, ATOL) (Strona 496)
OVR	K	Korekcja prędkości obrotowej	m	+		<i>PGAs/</i>
OVRA	K	Osiowa korekcja prędkości obrotowej	m	+	+	<i>PGAs/</i>
OVRRAP	K	Korekcja posuwu szybkiego	m	+		<i>PGAs/</i>
P	A	Liczba przebiegów podprogramu		+		<i>PGAs/</i> Liczba powtórzeń programu (P) (Strona 193)
PAROT	G	Zorientowanie układu współrzędnych obrabianego przedmiotu na obrabianym przedmiocie	m	+		<i>PGs/</i>
PAROTOF ⁶⁾	G	Wyłączenie obrotu frame odniesionego do obrabianego przedmiotu	m	+		<i>PGs/</i>
PCALL	K	Wywołanie podprogramów z absolutnym podaniem ścieżki i przekazaniem parametrów		+		<i>PGAs/</i> Wywołanie podprogramu z podaniem ścieżki i parametrami (PCALL) (Strona 200)
PDELAYOF	G	Zwłoka przy "Tłoczenie WYŁ."	m	+		<i>PGAs/</i> Tłoczenie i cięcie wł. lub wył. (SPOF, SON, PON, SONS, PONS, PDELAYON, PDELAYOF, PUNCHACC) (Strona 577)
PDELAYON ⁶⁾	G	Zwłoka przy "Tłoczenie WŁ."	m	+		<i>PGAs/</i> Tłoczenie i cięcie wł. lub wył. (SPOF, SON, PON, SONS, PONS, PDELAYON, PDELAYOF, PUNCHACC) (Strona 577)
PHI	K	Kąt obrotu orientacji wokół osi kierunkowej stożka		+		<i>PGAs/</i> Programowanie orientacji wzdłuż poboczniczy stożka (ORIPANE, ORICONCW, ORICONCCW, ORICONTO, ORICONIO) (Strona 332)
PHU	K	Fizyczna jednostka zmiennej		+		<i>PGAs/</i> Definicja zmiennych użytkownika (DEF) (Strona 24)

Tablice

19.1 Instrukcje

Instrukcja	Art ¹⁾	Znaczenie	W ²⁾	TP ³⁾	SA ⁴⁾	Opis patrz ⁵⁾
1) 2) 3) 4) 5) Objaśnienia patrz legenda (Strona 795).						
PL	A	1. B-Spline: odstęp węzłów 2. Interpolacja wielomianowa: Długość przedziału parametrów przy interpolacji wielomianowej	s	+		<i>PGAs/</i> 1. Interpolacja Spline (ASPLINE, BSPLINE, CSPLINE, BAUTO, BNAT, BTAN, EAUTO, ENAT, ETAN, PW, SD, PL) (Strona 234) 2. Interpolacja wielomianowa (POLY, POLYPATH, PO, PL) (Strona 248)
PM	K	na minutę		+		<i>PGs/</i>
PO	K	Współczynnik wielomianu przy interpolacji wielomianowej	s	+		<i>PGAs/</i> Interpolacja wielomianowa (POLY, POLYPATH, PO, PL) (Strona 248)
POCKET3	C	Cykl technologiczny: Frezowanie kieszeni prostokątnej		+		<i>PGAs/</i> Frezowanie kieszeni prostokątnej - POCKET3 (Strona 671)
POCKET4	C	Cykl technologiczny: Frezowanie kieszeni kołowej		+		<i>PGAs/</i> Frezowanie kieszeni kołowej - POCKET4 (Strona 674)
POLF	K	Pozycja wycofania LIFTFAST	m	+		<i>PGs/PGAs/</i>
POLFA	P	Uruchomienie pozycji wycofania poszczególnych osi z \$AA_ESR_TRIGGER	m	+	+	<i>PGs/</i>
POLFMASK	P	Udostępnienie osi dla wycofania bez zależności między osiami	m	+	-	<i>PGs/</i>
POLFMLIN	P	Udostępnienie osi dla wycofania z zależnością liniową między osiami	m	+	-	<i>PGs/</i>
POLY	G	Interpolacja wielomianowa	m	+		<i>PGAs/</i> Interpolacja wielomianowa (POLY, POLYPATH, PO, PL) (Strona 248)
POLYPATH	P	Interpolacja wielomianowa wybierana dla grup osi AXIS lub VECT	m	+	-	<i>PGAs/</i> Interpolacja wielomianowa (POLY, POLYPATH, PO, PL) (Strona 248)
PON	G	Tłoczenie Wł.	m	+		<i>PGAs/</i> Tłoczenie i cięcie wł. lub wył. (SPOF, SON, PON, SONS, PONS, PDELAYON, PDELAYOF, PUNCHACC) (Strona 577)
PONS	G	Tłoczenie Wł. w takcie IPO	m	+		<i>PGAs/</i> Tłoczenie i cięcie wł. lub wył. (SPOF, SON, PON, SONS, PONS, PDELAYON, PDELAYOF, PUNCHACC) (Strona 577)
POS	K	Pozycjonowanie osi		+	+	<i>PGs/</i>
POSA	K	Pozycjonowanie osi poza granice bloku		+	+	<i>PGs/</i>
POSM	P	Pozycjonowanie magazynu		+	-	<i>FBWs/</i>

Instrukcja	Art 1)	Znaczenie	W 2)	TP 3)	SA 4)	Opis patrz 5)
1) 2) 3) 4) 5) Objaśnienia patrz legenda (Strona 795).						
POSMT	P	Pozycjonowanie Multitool na uchwyt narzędziowy i numer miejsca		+	-	<i>FBWsl</i>
POSP	K	Pozycjonowanie w częściach (ruch wahadłowy)		+		<i>PGsl</i>
POSRANGE	F	Określenie, czy aktualnie interpolowana pozycja zadana osi znajduje się w oknie wokół zadanej pozycji odniesienia		+	+	<i>FBSY</i>
POT	F	Kwadrat (funkcja arytmetyczna)		+	+	<i>PGAsl</i> Funkcje arytmetyczne (Strona 70)
PR	K	Na obrót		+		<i>PGsl</i>
PREPRO	PA	Oznakowanie podprogramów z przygotowaniem		+		<i>PGAsl</i> Oznakowanie podprogramów z przygotowaniem (PREPRO) (Strona 180)
PRESETON	P	Ustawienie wartości zadanej dla programowanych osi		+	+	<i>PGAsl</i> Programowanie orientacji wzdłuż pobocznic stożka (ORIPANE, ORICONCW, ORICONCCW, ORICONTO, ORICONIO) (Strona 332)
PRIO	K	Słowo kluczowe do ustawienia priorytetu przy wykonywaniu przerw		+		<i>PGAsl</i> Przyporządkowanie i uruchomienie procedury przerwania (SETINT, PRIO, BLSYNC) (Strona 126)
PRLOC	K	Inicjalizacja zmiennej po Reset tylko po zmianie lokalnej		+		<i>PGAsl</i> Atrybut: wartość inicjalizacyjna (Strona 33)
PROC	K	Pierwsza instrukcja podprogramu		+		<i>PGAsl</i> Wywołanie podprogramu z podaniem ścieżki i parametrami (PCALL) (Strona 200)
PROTA	P	Żądanie ponownego obliczenia modelu kolizji		+		<i>PGAsl</i> Żądanie nowego obliczenia modelu kolizji (PROTA) (Strona 386)
PROTD	F	Obliczenie odstępów drugiego obszaru ochrony		+		<i>PGAsl</i> Ustawienie stanu obszaru ochrony (PROTS) (Strona 388)
PROTS	P	Ustawienie obszaru ochrony		+		<i>PGAsl</i> Określenie odległości dwóch obszarów ochrony (PROTD) (Strona 389)
PSI	K	Kąt rozwarcia stożka		+		<i>PGAsl</i> Programowanie orientacji wzdłuż pobocznic stożka (ORIPANE, ORICONCW, ORICONCCW, ORICONTO, ORICONIO) (Strona 332)

Instrukcja	Art ¹⁾	Znaczenie	W ²⁾	TP ³⁾	SA ⁴⁾	Opis patrz ⁵⁾
1) 2) 3) 4) 5) Objaśnienia patrz legenda (Strona 795).						
PTP	G	Ruch punkt do punktu	m	+		<i>PGAs/</i> Ruch kartezjański PTP (Strona 369)
PTPG0	G	Ruch punkt do punktu tylko przy G0, w przeciwnym przypadku CP	m	+		<i>PGAs/</i> PTP przy TRANSMIT (Strona 373)
PUNCHACC	P	Zależne od drogi przyspieszenie przy cięciu		+	-	<i>PGAs/</i> Tłoczenie i cięcie wł. lub wył. (SPOF, SON, PON, SONS, PONS, PDELAYON, PDELAYOF, PUNCHACC) (Strona 577)
PUTFTOC	P	Korekcja dokładna narzędzia dla obciążania równoległego		+	-	<i>PGAs/</i> Korekcja narzędzia online (PUTFTOCF, FCTDEF, PUTFTOC, FTOCON, FTOCOF) (Strona 405)
PUTFTOCF	P	Korekcja dokładna narzędzia w zależności od ustalonej przy pomocy FCTDEF funkcji obciążania równoległego		+	-	<i>PGAs/</i> Korekcja narzędzia online (PUTFTOCF, FCTDEF, PUTFTOC, FTOCON, FTOCOF) (Strona 405)
PW	A	B-Spline, waga punktu	s	+		<i>PGAs/</i> Interpolacja Spline (ASPLINE, BSPLINE, CSPLINE, BAUTO, BNAT, BTAN, EAUTO, ENAT, ETAN, PW, SD, PL) (Strona 234)
QU	K	Szybkie wyprowadzenie funkcji dodatkowej (pomocniczej)		+		<i>PGs/</i>
R...	A	Parametr obliczeniowy również jako ustawiany identyfikator adresu z rozszerzeniem numerycznym		+		<i>PGAs/</i> Predefiniowane zmienne użytkownika: parametry obliczeniowe (R) (Strona 20)
RAC	K	Absolutne, pojedynczymi blokami, specyficzne dla osi programowanie na promieniu	s	+		<i>PGs/</i>
RDISABLE	P	Blokada wczytywania		-	+	<i>FBSY</i>
READ	P	Wczytuje w podanym pliku jeden lub wiele wierszy i zapisuje w tablicy przeczytane informacje		+	-	<i>PGAs/</i> Odczyt wierszy w pliku (READ) (Strona 150)
REAL	K	Typ danych: zmienna zmiennoprzecinkowa ze znakiem (liczby rzeczywiste)		+		<i>PGAs/</i> Definicja zmiennych użytkownika (DEF) (Strona 24)
REDEF	K	Ustawienie dla danych maszynowych, elementów językowych NC i zmiennych systemowych, w przypadku których grup użytkowników są one wyświetlane.		+		<i>PGAs/</i> Redefinicja zmiennych systemowych, zmiennych użytkownika i poleceń językowych NC (REDEF) (Strona 30)
RELEASE	P	Udostępnienie osi maszyny do zamiany		+	+	<i>PGAs/</i> Zamiana osi, zamiana wrzecion (RELEASE, GET, GETD) (Strona 136)

Instrukcja	Art 1)	Znaczenie	W 2)	TP 3)	SA 4)	Opis patrz 5)
1) 2) 3) 4) 5) Objaśnienia patrz legenda (Strona 795).						
REP	K	Słowo kluczowe do inicjalizacji wszystkich elementów tablicy o tej samej wartości		+		<i>PGAs!</i> Definicja i inicjalizacja zmiennych tablicowych (DEF, SET, REP) (Strona 46)
REPEAT	K	Powtórzenie pętli programowej		+		<i>PGAs!</i> Powtórzenie programu (REPEAT, REPEATB, ENDLABEL, P) (Strona 104)
REPEATB	K	Powtórzenie wiersza programu		+		<i>PGAs!</i> Powtórzenie programu (REPEAT, REPEATB, ENDLABEL, P) (Strona 104)
REPOSA	G	Dosunięcie przywracające do konturu liniowo we wszystkich osiach	s	+		<i>PGAs!</i> Dosunięcie przywracające do konturu (REPOSA, REPOSL, REPOSQ, REPOSQA, REPOSH, REPOSHA, DISR, DISPR, RMIBL, RMBBL, RMEBL, RMNBL) (Strona 483)
REPOSH	G	Dosunięcie przywracające do konturu po półokręgu	s	+		<i>PGAs!</i> Dosunięcie przywracające do konturu (REPOSA, REPOSL, REPOSQ, REPOSQA, REPOSH, REPOSHA, DISR, DISPR, RMIBL, RMBBL, RMEBL, RMNBL) (Strona 483)
REPOSHA	G	Dosunięcie przywracające do konturu we wszystkich osiach; osie geometryczne po półokręgu	s	+		<i>PGAs!</i> Dosunięcie przywracające do konturu (REPOSA, REPOSL, REPOSQ, REPOSQA, REPOSH, REPOSHA, DISR, DISPR, RMIBL, RMBBL, RMEBL, RMNBL) (Strona 483)
REPOSL	G	Dosunięcie przywracające do konturu liniowe	s	+		<i>PGAs!</i> Dosunięcie przywracające do konturu (REPOSA, REPOSL, REPOSQ, REPOSQA, REPOSH, REPOSHA, DISR, DISPR, RMIBL, RMBBL, RMEBL, RMNBL) (Strona 483)
REPOSQ	G	Dosunięcie przywracające do konturu po ćwierćokręgu	s	+		<i>PGAs!</i> Dosunięcie przywracające do konturu (REPOSA, REPOSL, REPOSQ, REPOSQA, REPOSH, REPOSHA, DISR, DISPR, RMIBL, RMBBL, RMEBL, RMNBL) (Strona 483)
REPOSQA	G	Dosunięcie przywracające do konturu liniowo we wszystkich osiach; osie geometryczne po ćwierćokręgu	s	+		<i>PGAs!</i> Dosunięcie przywracające do konturu (REPOSA, REPOSL, REPOSQ, REPOSQA, REPOSH, REPOSHA, DISR, DISPR, RMIBL, RMBBL, RMEBL, RMNBL) (Strona 483)
RESET	P	Cofnięcie cyklu technologicznego		-	+	<i>FBSY</i>

Instrukcja	Art ¹⁾	Znaczenie	W ²⁾	TP ³⁾	SA ⁴⁾	Opis patrz ⁵⁾
1) 2) 3) 4) 5) Objaśnienia patrz legenda (Strona 795).						
RESETMON	P	Polecenie językowe do uaktywnienia wartości zadanej		+	-	<i>FBWs/</i>
RET	P	Koniec podprogramu		+	+	<i>PGAs/</i> Parametryzowalny powrót z podprogramu (RET ...) (Strona 183)
RIC	K	Względnie pojedynczymi blokami, specyficzne dla osi programowanie na promieniu	s	+		<i>PGs/</i> Zamiana osi, zamiana wrzecion (RELEASE, GET, GETD) (Strona 136)
RINDEX	F	Określenie indeksu znaku w wejściowym łańcuchu znaków		+	-	<i>PGAs/</i> Szukanie znaku/łańcucha znaków w łańcuchu znaków (INDEX, RINDEX, MINDEX, MATCH) (Strona 86)
RMB	G	Dosunięcie przywracające do punktu początkowego bloku	m	+		<i>PGAs/</i> Dosunięcie przywracające do konturu (REPOSA, REPOSL, REPOSQ, REPOSQA, REPOSH, REPOSHA, DISR, DISPR, RMIBL, RMBBL, RMEBL, RMNBL) (Strona 483)
RMBBL	G	Dosunięcie przywracające do punktu początkowego bloku	s	+		<i>PGAs/</i> Dosunięcie przywracające do konturu (REPOSA, REPOSL, REPOSQ, REPOSQA, REPOSH, REPOSHA, DISR, DISPR, RMIBL, RMBBL, RMEBL, RMNBL) (Strona 483)
RME	G	Dosunięcie przywracające do punktu końcowego bloku	m	+		<i>PGAs/</i> Dosunięcie przywracające do konturu (REPOSA, REPOSL, REPOSQ, REPOSQA, REPOSH, REPOSHA, DISR, DISPR, RMIBL, RMBBL, RMEBL, RMNBL) (Strona 483)
RMEBL	G	Dosunięcie przywracające do punktu końcowego bloku	s	+		<i>PGAs/</i> Dosunięcie przywracające do konturu (REPOSA, REPOSL, REPOSQ, REPOSQA, REPOSH, REPOSHA, DISR, DISPR, RMIBL, RMBBL, RMEBL, RMNBL) (Strona 483)
RMI ⁶⁾	G	Dosunięcie przywracające do punktu przerwania	m	+		<i>PGAs/</i> Dosunięcie przywracające do konturu (REPOSA, REPOSL, REPOSQ, REPOSQA, REPOSH, REPOSHA, DISR, DISPR, RMIBL, RMBBL, RMEBL, RMNBL) (Strona 483)
RMIBL ⁶⁾	G	Dosunięcie przywracające do punktu przerwania	s	+		<i>PGAs/</i> Dosunięcie przywracające do konturu (REPOSA, REPOSL, REPOSQ, REPOSQA, REPOSH, REPOSHA, DISR, DISPR, RMIBL, RMBBL, RMEBL, RMNBL) (Strona 483)

Instrukcja	Art 1)	Znaczenie	W 2)	TP 3)	SA 4)	Opis patrz 5)
1) 2) 3) 4) 5) Objaśnienia patrz legenda (Strona 795).						
RMN	G	Dosunięcie przywracające do najbliższej położonego punktu toru	m	+		<i>PGAs/</i> Dosunięcie przywracające do konturu (REPOSA, REPOSL, REPOSQ, REPOSQA, REPOSH, REPOSHA, DISR, DISPR, RMIBL, RMBBL, RMEBL, RMNBL) (Strona 483)
RMNBL	G	Dosunięcie przywracające do najbliższej położonego punktu toru	s	+		<i>PGAs/</i> Dosunięcie przywracające do konturu (REPOSA, REPOSL, REPOSQ, REPOSQA, REPOSH, REPOSHA, DISR, DISPR, RMIBL, RMBBL, RMEBL, RMNBL) (Strona 483)
RND	A	Zaokrąglenie naroża konturu	s	+		<i>PGs/</i>
RNDM	A	Zaokrąglenie modalne	m	+		<i>PGs/</i>
ROT	G	Programowany obrót	s	+		<i>PGs/</i>
ROTS	G	Programowane obroty frame z kątami przestrzennymi	s	+		<i>PGs/</i>
ROUND	F	Zaokrąglenie miejsc po przecinku		+	+	<i>PGAs/</i> Funkcje arytmetyczne (Strona 70)
ROUNDUP	F	Zaokrąglenie wprowadzonej wartości do góry		+	+	<i>PGAs/</i> Zaokrąglenie do góry (ROUNDUP) (Strona 158)
RP	A	Współrzędna promieniowa	m/s	+		<i>PGs/</i>
RPL	A	Obrót w płaszczyźnie	s	+		<i>PGs/</i>
RT	K	Parametry dla dostępu do danych frame: Obrót		+		<i>PGAs/</i> Odczyt i zmiana komponentów frame (TR, FI, RT, SC, MI) (Strona 287)
RTLIOF	G	G0 bez interpolacji liniowej (interpolacja pojedynczych osi)	m	+		<i>PGs/</i>
RTLION 6)	G	G0 z interpolacją liniową	m	+		<i>PGs/</i>
S	A	Prędkość obrotowa wrzeciona (przy G4, G96/G961 inne znaczenie)	m/s	+	+	<i>PGs/</i>
SAVE	PA	Atrybut do zapamiętania funkcji G i Frame przy wywoływaniu podprogramu		+		<i>PGAs/</i> Zapisywanie modalnych funkcji G (SAVE) (Strona 170)
SBLOF	P	Blokowanie wykonywania pojedynczymi blokami		+	-	<i>PGAs/</i> Blokowanie wykonywania pojedynczymi blokami (SBLOF, SBLON) (Strona 171)
SBLON	P	Usunięcie blokady wykonywania pojedynczymi blokami		+	-	<i>PGAs/</i> Blokowanie wykonywania pojedynczymi blokami (SBLOF, SBLON) (Strona 171)
SC	K	Parametry dla dostępu do danych frame: Skalowanie		+		<i>PGAs/</i> Odczyt i zmiana komponentów frame (TR, FI, RT, SC, MI) (Strona 287)
SCALE	G	Skalowanie programowane	s	+		<i>PGs/</i>

Instrukcja	Art ¹⁾	Znaczenie	W ²⁾	TP ³⁾	SA ⁴⁾	Opis patrz ⁵⁾
1) 2) 3) 4) 5) Objaśnienia patrz legenda (Strona 795).						
SCC	K	Selektywne przyporządkowanie osi poprzecznej do G96/G961/G962. Identyfikatory osi mogą być osią geometryczną, kanałową lub maszyny.		+		<i>PGsI</i>
SCPARA	K	Programowanie zestawu parametrów serwo		+	+	<i>PGAsI</i> Programowalne przełączenie zestawu parametrów (SCPARA) (Strona 601)
SD	A	Stopień spline	s	+		<i>PGAsI</i> Interpolacja Spline (ASPLINE, BSPLINE, CSPLINE, BAUTO, BNAT, BTAN, EAUTO, ENAT, ETAN, PW, SD, PL) (Strona 234)
SET	K	Słowo kluczowe do inicjalizacji wszystkich elementów tablicy z wyszczególnionymi wartościami		+		<i>PGAsI</i> Definicja i inicjalizacja zmiennych tablicowych (DEF, SET, REP) (Strona 46)
SETAL	P	Ustawienie alarmu		+	+	<i>PGAsI</i> Alarmy (SETAL) (Strona 622)
SETDNO	F	Przyporządkowanie numeru D do ostrza (CE) narzędzia (T)		+	-	<i>PGAsI</i> Dowolne nadawanie numerów D: Zmiana nazw numerów D (GETDNO, SETDNO) (Strona 431)
SETINT	K	Ustalenie, która procedura przerwania ma zostać uaktywniona, gdy jest aktywne wejście NCK		+		<i>PGAsI</i> Przyporządkowanie i uruchomienie procedury przerwania (SETINT, PRIO, BLSYNC) (Strona 126)
SETM	P	Ustawienie znaczników we własnym kanale		+	+	<i>PGAsI</i> Koordynacja programu (INIT, START, WAITM, WAITMC, WAITE, SETM, CLEARM) (Strona 118)
SETMS	P	Przełączenie z powrotem na wrzeczono wiodące ustalone w danej maszynowej		+	-	
SETMS(n)	P	Wrzeczono n ma obowiązywać, jako wrzeczono wiodące		+		<i>PGsI</i>
SETMTH	P	Ustawienie numeru uchwytu narzędzia master		+	-	<i>FBWsI</i>
SETPIECE	P	Uwzględnienie liczby sztuk dla wszystkich narzędzi, które są przyporządkowane do wrzeczona		+	-	<i>FBWsI</i>
SETTA	P	Ustawienie aktywności narzędzia z zespołu zużycia		+	-	<i>FBWsI</i>
SETTCOR	F	Zmiana komponentów narzędzia przy uwzględnieniu wszystkich warunków brzegowych		+	-	<i>FB1sI (W1)</i>

Instrukcja	Art 1)	Znaczenie	W 2)	TP 3)	SA 4)	Opis patrz 5)
1) 2) 3) 4) 5) objaśnienia patrz legenda (Strona 795).						
SETTIA	P	Wyłączenie aktywności narzędzia z zespołu zużycia		+	-	<i>FBWsl</i>
SF	A	Przesunięcie punktu początkowego przy nacinaniu gwintu	m	+		<i>PGsl</i>
SIN	F	Sinus (trygonometryczna funkcja)		+	+	<i>PGAsl</i> Funkcje arytmetyczne (Strona 70)
SIRELAY	F	Uaktywnienie funkcji bezpieczeństwa sparametryzowanych przy pomocy SIRELIN, SIRELOUT i SIRELTIME.		-	+	<i>FBSlsl</i>
SIRELIN	P	Inicjalizacja wielkości wejściowych modułu funkcjonalnego		+	-	<i>FBSlsl</i>
SIRELOUT	P	Inicjalizacja wielkości wyjściowych modułu funkcjonalnego		+	-	<i>FBSlsl</i>
SIRELTIME	P	Inicjalizacja zegara modułu funkcjonalnego		+	-	<i>FBSlsl</i>
SLOT1	C	Cykl technologiczny: Rowek podłużny		+		<i>PGAsl</i> Rowek podłużny - SLOT1 (Strona 682)
SLOT2	C	Cykl technologiczny: Rowek kołowy		+		<i>PGAsl</i> Rowek kołowy - SLOT2 (Strona 684)
SOFT	G	Przyśpieszenie ruchu po torze z ograniczeniem przyśpieszenia drugiego stopnia	m	+		<i>PGsl</i> Tryb przyśpieszenia (BRISK, BRISKA, SOFT, SOFTA, DRIVE, DRIVEA) (Strona 463)
SOFTA	P	Włączenie przyśpieszenia osi programowanych, z ograniczeniem przyspieszenia drugiego stopnia		+	-	<i>PGsl</i> Tryb przyśpieszenia (BRISK, BRISKA, SOFT, SOFTA, DRIVE, DRIVEA) (Strona 463)
SON	G	Cięcie WŁ.	m	+		<i>PGAsl</i> Tłoczenie i cięcie wł. lub wyt. (SPOF, SON, PON, SONS, PONS, PDELAYON, PDELAYOF, PUNCHACC) (Strona 577)
SONS	G	Cięcie WŁ. w takcie IPO	m	+		<i>PGAsl</i> Tłoczenie i cięcie wł. lub wyt. (SPOF, SON, PON, SONS, PONS, PDELAYON, PDELAYOF, PUNCHACC) (Strona 577)
SPATH 6)	G	Odniesieniem toru dla osi FGROUP jest długość łuku	m	+		<i>PGAsl</i> Ustawiane odniesienie do toru (SPATH, UPATH) (Strona 254)
SPCOF	P	Przełączenie wrzeciona wiodącego lub wrzeciona (n) z regulacji położenia na regulację prędkości obrotowej	m	+	-	<i>PGsl</i>

Instrukcja	Art ¹⁾	Znaczenie	W ²⁾	TP ³⁾	SA ⁴⁾	Opis patrz ⁵⁾
1) 2) 3) 4) 5) Objaśnienia patrz legenda (Strona 795).						
SPCON	P	Przełączenie wrzeciona wiodącego lub wrzeciona (n) z regulacji prędkości obrotowej na regulację położenia	m	+	-	<i>PGAs/</i>
SPI	F	Konwertuje numer wrzeciona na identyfikator osi		+	-	<i>PGAs/</i> Funkcje osi (AXNAME, AX, SPI, AXTOSPI, ISAXIS, AXSTRING, MODAXVAL) (Strona 591)
SPIF1 ⁶⁾	G	Szybkie wejścia wyjścia NCK dla tłoczenia/cięcia bajt 1	m	+		<i>FB2sl (N4)</i>
SPIF2	G	Szybkie wejścia wyjścia NCK dla tłoczenia/cięcia bajt 2	m	+		<i>FB2sl (N4)</i>
SPLINEPATH	P	Ustalenie zespołu spline		+	-	<i>PGAs/</i> Zespół spline (SPLINEPATH) (Strona 244)
SPN	A	Liczba odcinków częściowych na blok	s	+		<i>PGAs/</i> Automatyczny podział drogi (Strona 582)
SPOF ⁶⁾	G	Skok WYŁ., tłoczenie, cięcie WYŁ	m	+		<i>PGAs/</i> Tłoczenie i cięcie wł. lub wył. (SPOF, SON, PON, SONS, PONS, PDELAYON, PDELAYOF, PUNCHACC) (Strona 577)
SPOS	K	Pozycja wrzeciona	m	+	+	<i>PGs/</i>
SPOSA	K	Pozycja wrzeciona poza granice bloku	m	+		<i>PGs/</i>
SPP	A	Długość odcinka częściowego	m	+		<i>PGAs/</i> Automatyczny podział drogi (Strona 582)
SPRINT	F	Zwraca sformatowany wejściowy łańcuch znaków		+		<i>PGAs/</i> Formatowanie łańcucha znaków (SPRINT) (Strona 90)
SQRT	F	Pierwiastek kwadratowy (funkcja arytmetyczna) (square root)		+	+	<i>PGAs/</i> Funkcje arytmetyczne (Strona 70)
SR	A	Droga wycofania przy ruchu wahadłowym dla akcji synchronicznej	s	+		<i>PGs/</i>
SRA	K	Droga wycofania przy ruchu wahadłowym i zewnętrznym posuwie szybkim osiowo dla akcji synchronicznej	m	+		<i>PGs/</i>
ST	A	Czas wyiskrzania ruchem wahadłowym dla akcji synchronicznej	s	+		<i>PGs/</i>

Instrukcja	Art 1)	Znaczenie	W 2)	TP 3)	SA 4)	Opis patrz 5)
1) 2) 3) 4) 5) Objaśnienia patrz legenda (Strona 795).						
STA	K	Czas wyiskrzania ruchem wahadłowym osiowo dla akcji synchronicznej	m	+		<i>PGs/</i>
START	P	Uruchomienie wybranych programów w wielu kanałach równocześnie z bieżącego programu		+	-	<i>PGAs/</i> Koordynacja programu (INIT, START, WAITM, WAITMC, WAITE, SETM, CLEARM) (Strona 118)
STARTFIFO 6)	G	Wykonywanie; równoległe do tego wypełnianie bufora przebiegu	m	+		<i>PGAs/</i> Przebieg programu z pamięcią przebiegu wyprzedzającego (STOPFIFO, STARTFIFO, FIFOCTRL, STOPRE) (Strona 472)
STAT		Położenie przegubów	s	+		<i>PGAs/</i> Ruch kartezyjski PTP (Strona 369)
STOLF	K	Współczynnik tolerancji G0	m	+		<i>PGAs/</i> Tolerancja przy ruchach G0 (STOLF) (Strona 500)
STOPFIFO	G	Zatrzymanie wykonywania; wypełnienie bufora przebiegu, aż nastąpi rozpoznanie STARTFIFO, bufor przebiegu wypełniony albo koniec programu	m	+		<i>PGAs/</i> Przebieg programu z pamięcią przebiegu wyprzedzającego (STOPFIFO, STARTFIFO, FIFOCTRL, STOPRE) (Strona 472)
STOPRE	P	Zatrzymanie przebiegu wyprzedzającego, aż wszystkie przeczytane bloki przebiegu głównego będą wykonane		+	-	<i>PGAs/</i> Przebieg programu z pamięcią przebiegu wyprzedzającego (STOPFIFO, STARTFIFO, FIFOCTRL, STOPRE) (Strona 472)
STOPREOF	P	Anulowanie zatrzymania przebiegu wyprzedzającego		-	+	<i>FBSY</i>
STRING	K	Typ danych: Łańcuch znaków		+		<i>PGAs/</i> Definicja zmiennych użytkownika (DEF) (Strona 24)
STRINGIS	F	Sprawdza istniejący zakres językowy NC, a szczególnie należące do tego polecenia nazwy cykli NC, zmienne użytkownika, makra i nazwy etykiet: czy istnieją one, czy są poprawne, zdefiniowane albo aktywne.		+	-	<i>PGAs/</i> Sprawdzenie występującego zakresu językowego NC (STRINGIS) (Strona 602)
STRLEN	F	Określenie długości łańcucha znaków		+	-	<i>PGAs/</i> Określenie długości łańcucha znaków (STRLEN) (Strona 86)
SUBSTR	F	Określenie indeksu znaku w wejściowym łańcuchu znaków		+	-	<i>PGAs/</i> Wybór częściowego łańcucha znaków (SUBSTR) (Strona 88)

Instrukcja	Art ¹⁾	Znaczenie	W ²⁾	TP ³⁾	SA ⁴⁾	Opis patrz ⁵⁾
1) 2) 3) 4) 5) Objaśnienia patrz legenda (Strona 795).						
SUPA	G	Blokowanie aktualnego przesunięcia punktu zerowego, łącznie z przesunięciami programowanymi, frame systemowymi, przesunięciami kółkiem ręcznym (DRF), zewnętrznym przesunięciem punktu zerowego i ruchem nałożonym	s	+		<i>PGs/</i>
SVC	K	Prędkość skrawania przez narzędzie	m	+		<i>PGs/</i>
SYNFCT	P	Ewaluacja wielomianu zależnie od warunku w akcji synchronicznej ruchu		-	+	<i>FBSY</i>
SYNR	K	Odczyt zmiennej następuje synchronicznie, tzn. w chwili wykonywania		+		<i>PGAs/</i> Definicja zmiennych użytkownika (DEF) (Strona 24)
SYNRW	K	Odczyt i zapis zmiennej następuje synchronicznie, tzn. w chwili wykonywania		+		<i>PGAs/</i> Definicja zmiennych użytkownika (DEF) (Strona 24)
SYNW	K	Zapis zmiennej następuje synchronicznie, tzn. w chwili wykonywania		+		<i>PGAs/</i> Definicja zmiennych użytkownika (DEF) (Strona 24)
T	A	Wywołanie narzędzia (zmiana tylko wtedy, gdy ustalono to w danej maszynowej; poza tym konieczne polecenie M6)		+		<i>PGs/</i>
TAN	F	Tangens (trygonometryczna funkcja)		+	+	<i>PGAs/</i> Funkcje arytmetyczne (Strona 70)
TANG	P	Definicja aktualizacji stycznej zespołu osi		+	-	<i>PGAs/</i> Sterowanie styczne (TANG, TANGON, TANGOF, TLIFT, TANGDEL) (Strona 451)
TANGDEL	P	Skasowanie definicji aktualizacji stycznej zespołu osi		+	-	<i>PGAs/</i> Sterowanie styczne (TANG, TANGON, TANGOF, TLIFT, TANGDEL) (Strona 451)
TANGOF	P	Aktualizacja styczna WYŁ.		+	-	<i>PGAs/</i> Sterowanie styczne (TANG, TANGON, TANGOF, TLIFT, TANGDEL) (Strona 451)
TANGON	P	Aktualizacja styczna WŁ.		+	-	<i>PGAs/</i> Sterowanie styczne (TANG, TANGON, TANGOF, TLIFT, TANGDEL) (Strona 451)

Instrukcja	Art 1)	Znaczenie	W 2)	TP 3)	SA 4)	Opis patrz 5)
1) 2) 3) 4) 5) Objaśnienia patrz legenda (Strona 795).						
TCA (828D: _TCA)	P	Wybór narzędzia / wymiana narzędzia niezależnie od jego statusu		+	-	<i>FBWs/</i>
TCARR	A	Zażądanie nośnika narzędzi (numer "m")		+		<i>PGAs/</i> Korekcja długości narzędzia dla orientowalnych nośników narzędzi (TCARR, TCOABS, TCOFR, TCOFRX, TCOFRY, TCOFRZ) (Strona 439)
TCI	P	Przełoż narzędzie z magazynu pośredniego do magazynu		+	-	<i>FBWs/</i>
TCOABS 6)	G	Określenie składowych długości narzędzia z aktualnej orientacji narzędzia	m	+		<i>PGAs/</i> Korekcja długości narzędzia dla orientowalnych nośników narzędzi (TCARR, TCOABS, TCOFR, TCOFRX, TCOFRY, TCOFRZ) (Strona 439)
TCOFR	G	Określenie składowych długości narzędzia ze zorientowania aktywnego frame	m	+		<i>PGAs/</i> Korekcja długości narzędzia dla orientowalnych nośników narzędzi (TCARR, TCOABS, TCOFR, TCOFRX, TCOFRY, TCOFRZ) (Strona 439)
TCOFRX	G	Określenie orientacji narzędzia aktywnego frame przy wyborze narzędzia, narzędzie skierowane w kierunku X	m	+		<i>PGAs/</i> Korekcja długości narzędzia dla orientowalnych nośników narzędzi (TCARR, TCOABS, TCOFR, TCOFRX, TCOFRY, TCOFRZ) (Strona 439)
TCOFRY	G	Określenie orientacji narzędzia aktywnego frame przy wyborze narzędzia, narzędzie skierowane w kierunku Y	m	+		<i>PGAs/</i> Korekcja długości narzędzia dla orientowalnych nośników narzędzi (TCARR, TCOABS, TCOFR, TCOFRX, TCOFRY, TCOFRZ) (Strona 439)
TCOFRZ	G	Określenie orientacji narzędzia aktywnego frame przy wyborze narzędzia, narzędzie skierowane w kierunku Z	m	+		<i>PGAs/</i> Korekcja długości narzędzia dla orientowalnych nośników narzędzi (TCARR, TCOABS, TCOFR, TCOFRX, TCOFRY, TCOFRZ) (Strona 439)
THETA	A	Kąt obrotu	s	+		<i>PGAs/</i> Obroty orientacji narzędzia (ORIROTA, ORIROTR, ORIROTT, ORIROTC, THETA) (Strona 339)
TILT	A	Kąt boczny	m	+		<i>PGAs/</i> Programowanie orientacji narzędzia (A..., B..., C..., LEAD, TILT) (Strona 319)
TLIFT	P	Przy sterowaniu stycznym wstawianie bloku pośredniego na narożach konturu		+	-	<i>PGAs/</i> Sterowanie styczne (TANG, TANGON, TANGOF, TLIFT, TANGDEL) (Strona 451)

19.1 Instrukcje

Instrukcja	Art ¹⁾	Znaczenie	W ²⁾	TP ³⁾	SA ⁴⁾	Opis patrz ⁵⁾
1) 2) 3) 4) 5) Objaśnienia patrz legenda (Strona 795).						
TML	P	Wybór narzędzia z magazynu - numer miejsca		+	-	<i>FBWsl</i>
TMOF	P	Cofnięcie wyboru nadzoru narzędzia		+	-	<i>PGAsl</i> Specyficzny dla szlifowania nadzór narzędzia w programie obróbki (TMON, TMOF) (Strona 589)
TMON	P	Wybór nadzoru narzędzia		+	-	<i>PGAsl</i> Specyficzny dla szlifowania nadzór narzędzia w programie obróbki (TMON, TMOF) (Strona 589)
TO	K	Określa wartość końcową w pętli FOR		+		<i>PGAsl</i> Pętla FOR (FOR ... TO ..., ENDFOR) (Strona 114)
TOFF	K	Offset długości narzędzia w kierunku komponentu długości narzędzia, który działa równoległe do osi geometrycznej podanej w indeksie.	m	+		<i>PGsl</i>
TOFFL	K	Offset długości narzędzia w kierunku komponentu długości narzędzia L1, L2 lub L3	m	+		<i>PGsl</i>
TOFFOF	P	Cofnięcie korekcji długości narzędzia online		+	-	<i>PGAsl</i> Korekcja długości narzędzia online (TOFFON, TOFFOF) (Strona 442)
TOFFON	P	Uaktywnienie korekcji długości narzędzia online		+	-	<i>PGAsl</i> Korekcja długości narzędzia online (TOFFON, TOFFOF) (Strona 442)
TOFFR	A	Offset promienia narzędzia	m	+		<i>PGsl</i>
TOFRAME	G	Ustawienie osi Z WKS przez obrót frame równoległe do orientacji narzędzia	m	+		<i>PGsl</i>
TOFRAMEX	G	Ustawienie osi X WKS przez obrót frame równoległe do orientacji narzędzia	m	+		<i>PGsl</i>
TOFRAMEY	G	Ustawienie osi Y WKS przez obrót frame równoległe do orientacji narzędzia	m	+		<i>PGsl</i>
TOFRAMEZ	G	Jak TOFRAME	m	+		<i>PGsl</i>
TOLOWER	F	Zamiana liter łańcucha znaków na małe litery		+	-	<i>PGAsl</i> Zmiana na litery małe/duże (TOLOWER, TOUPPER) (Strona 85)
TOOLENV	F	Zapisanie aktualnych stanów, które mają znaczenie dla interpretacji danych narzędzia zapisanych w pamięci		+	-	<i>FB1sl (W1)</i>
TOOLGNT	F	Określenie liczby narzędzi w grupie		+	-	<i>FBWsl</i>

Instrukcja	Art 1)	Znaczenie	W 2)	TP 3)	SA 4)	Opis patrz 5)
1) 2) 3) 4) 5) Objaśnienia patrz legenda (Strona 795).						
TOOLGT	F	Określenie numeru T narzędzia z grupy		+	-	<i>FBWsl</i>
TOROT	G	Ustawienie osi Z WKS przez obrót frame równolegle do orientacji narzędzia	m	+		<i>PGsl</i>
TOROTOF 6)	G	Obroty frame w kierunku narzędzia WYŁ.	m	+		<i>PGsl</i>
TOROTX	G	Ustawienie osi X WKS przez obrót frame równolegle do orientacji narzędzia	m	+		<i>PGsl</i>
TOROTY	G	Ustawienie osi Y WKS przez obrót frame równolegle do orientacji narzędzia	m	+		<i>PGsl</i>
TOROTZ	G	Jak TOROT	m	+		<i>PGsl</i>
TOUPPER	F	Zamiana liter łańcucha znaków na duże litery		+	-	<i>PGAsl</i> Zmiana na litery małe/duże (TOLOWER, TOUPPER) (Strona 85)
TOWBCS	G	Wartości zużycia w bazowym układzie współrzędnych (BKS)	m	+		<i>PGAsl</i> Układ współrzędnych aktywnej obróbki (TOWSTD, TOWMCS, TOWWCS, TOWBCS, TOWTCS, TOWKCS) (Strona 401)
TOWKCS	G	Wartości zużycia w układzie współrzędnych głowicy narzędziowej przy transformacji kinetycznej (od MKS różni się obrotem narzędzia)	m	+		<i>PGAsl</i> Układ współrzędnych aktywnej obróbki (TOWSTD, TOWMCS, TOWWCS, TOWBCS, TOWTCS, TOWKCS) (Strona 401)
TOWMCS	G	Wartości zużycia w układzie współrzędnych maszyny (MKS)	m	+		<i>PGAsl</i> Układ współrzędnych aktywnej obróbki (TOWSTD, TOWMCS, TOWWCS, TOWBCS, TOWTCS, TOWKCS) (Strona 401)
TOWSTD 6)	G	Wartość ustawienia podstawowego korekcji w długości narzędzia	m	+		<i>PGAsl</i> Układ współrzędnych aktywnej obróbki (TOWSTD, TOWMCS, TOWWCS, TOWBCS, TOWTCS, TOWKCS) (Strona 401)
TOWTCS	G	Wartości zużycia w układzie współrzędnych narzędzia (punkt odniesienia nośnika narzędzi T na uchwycie oprawki narzędziowej)	m	+		<i>PGAsl</i> Układ współrzędnych aktywnej obróbki (TOWSTD, TOWMCS, TOWWCS, TOWBCS, TOWTCS, TOWKCS) (Strona 401)
TOWWCS	G	Wartości zużycia w układzie współrzędnych obrabianego przedmiotu (WKS)	m	+		<i>PGAsl</i> Układ współrzędnych aktywnej obróbki (TOWSTD, TOWMCS, TOWWCS, TOWBCS, TOWTCS, TOWKCS) (Strona 401)

Tablice

19.1 Instrukcje

Instrukcja	Art ¹⁾	Znaczenie	W ²⁾	TP ³⁾	SA ⁴⁾	Opis patrz ⁵⁾
1) 2) 3) 4) 5) Objaśnienia patrz legenda (Strona 795).						
TR	K	Składowa przesunięcia zmiennej frame		+		<i>PGAs!</i> Odczyt i zmiana komponentów frame (TR, FI, RT, SC, MI) (Strona 287)
TRAANG	P	Transformacja osi skośnej		+	-	<i>PGAs!</i> Oś skośna (TRAANG) (Strona 364)
TRACON	P	Transformacja kaskadowana		+	-	<i>PGAs!</i> Transformacje powiązane (TRACON, TRAFEOF) (Strona 378)
TRACYL	P	Walec: transformacja dla pobocznic walca		+	-	<i>PGAs!</i> Transformacja pobocznic walca (TRACYL) (Strona 355)
TRAFEOF	P	Wyłączenie transformacji aktywnych w kanale		+	-	<i>PGAs!</i> Transformacje powiązane (TRACON, TRAFEOF) (Strona 378)
TRAILOF	P	Holowanie synchroniczne z osią WYŁ.		+	+	<i>PGAs!</i> Nadążanie (TRAILON, TRAILOF) (Strona 503)
TRAILON	P	Nadążanie asynchroniczne WŁ		+	+	<i>PGAs!</i> Nadążanie (TRAILON, TRAILOF) (Strona 503)
TRANS	G	Przesunięcie programowane	s	+		<i>PGs!</i>
TRANSMIT	P	Transformacja biegunowa (obróbka powierzchni czołowej)		+	-	<i>PGAs!</i> Obróbka frezarska na częściach toczonych (TRANSMIT) (Strona 353)
TRAORI	P	Transformacja 4-, 5-osiowa, transformacja rodzajowa		+	-	<i>PGAs!</i> Transformacje trzy-, cztero- i pięcioosiowe (TRAORI) (Strona 316)
TRUE	K	Stała logiczna: prawda		+		<i>PGAs!</i> Definicja zmiennych użytkownika (DEF) (Strona 24)
TRUNC	F	Odcięcie miejsc po przecinku		+	+	<i>PGAs!</i> Korekcja dokładności w przypadku błędów porównania (TRUNC) (Strona 75)
TU		Kąt osi	s	+		<i>PGAs!</i> Ruch kartezjański PTP (Strona 369)
TURN	A	Liczba zwojów dla linii śrubowej	s	+		<i>PGs!</i>
ULI	K	Górna wartość graniczna zmiennych		+		<i>PGAs!</i> Atrybut: wartości graniczne (LLI, ULI) (Strona 36)
UNLOCK	P	Zezwolenie dla akcji synchronicznej z ID (kontynuowanie cyklu technologicznego)		-	+	<i>FBSY</i>

Instrukcja	Art 1)	Znaczenie	W 2)	TP 3)	SA 4)	Opis patrz 5)
1) 2) 3) 4) 5) Objaśnienia patrz legenda (Strona 795).						
UNTIL	K	Warunek zakończenia pętli REPEAT		+		<i>PGAs!</i> Pętla programowa z warunkiem na początku pętli (WHILE, ENDWHILE) (Strona 116)
UPATH	G	Odniesieniem toru dla osi FGROUP jest parametr krzywej	m	+		<i>PGAs!</i> Ustawiane odniesienie do toru (SPATH, UPATH) (Strona 254)
VAR	K	Słowo kluczowe: Sposób przekazania parametrów		+		<i>PGAs!</i> Wywołanie podprogramu z przekazaniem parametrów (EXTERN) (Strona 191)
VELOLIM	K	Zmniejszenie maksymalnej prędkości osiowej	m	+		<i>PGAs!</i> Procentowa korekcja prędkości (VELOLIM) (Strona 493)
VELOLIMA	K	Zmniejszenie albo zwiększenie maksymalnej prędkości osiowej w osi nadążnej	m	+	+	<i>PGAs!</i> Sterowanie przyśpieszeniem w przypadku osi nadążnych (VELOLIMA, ACCLIMA, JERKLIMA) (Strona 465)
WAITC	P	Czekanie, aż kryterium zmiany bloku sprzężenia dla osi/wrzecion będzie spełnione		+	-	<i>PGAs!</i> Koordynacja programu (INIT, START, WAITM, WAITMC, WAITE, SETM, CLEARM) (Strona 118)
WAITE	P	Czekanie na koniec programu w innym kanale.		+	-	<i>PGAs!</i> Koordynacja programu (INIT, START, WAITM, WAITMC, WAITE, SETM, CLEARM) (Strona 118)
WAITENC	P	Czekanie na synchronizowane lub odtworzone pozycje osi		+	-	<i>PGAs!</i> Czekanie na obowiązującą pozycję osi (WAITENC) (Strona 600)
WAITM	P	Czekanie na znacznik w podanym kanale; zakończenie poprzedniego bloku z zatrzymaniem dokładnym.		+	-	<i>PGAs!</i> Koordynacja programu (INIT, START, WAITM, WAITMC, WAITE, SETM, CLEARM) (Strona 118)
WAITMC	P	Czekanie na znacznik w podanym kanale; zatrzymanie dokładne tylko wtedy, gdy inne kanały jeszcze nie doszły do znacznika.		+	-	<i>PGAs!</i> Koordynacja programu (INIT, START, WAITM, WAITMC, WAITE, SETM, CLEARM) (Strona 118)
WAITP	P	Czekanie na koniec ruchu osi pozycjonowania		+	-	<i>PGs!</i>
WAITS	P	Czekanie na osiągnięcie pozycji wrzeciona		+	-	<i>PGs!</i>
WALCS0 6)	G	Wybór ograniczenia obszaru pracy WKS cofnięty	m	+		<i>PGs!</i>
WALCS1	G	Grupa 1 ograniczenia obszaru pracy WKS aktywna	m	+		<i>PGs!</i>

Instrukcja	Art ¹⁾	Znaczenie	W ²⁾	TP ³⁾	SA ⁴⁾	Opis patrz ⁵⁾
1) 2) 3) 4) 5) Objaśnienia patrz legenda (Strona 795).						
WALCS2	G	Grupa 2 ograniczenia obszaru pracy WKS aktywna	m	+		<i>PGsl</i>
WALCS3	G	Grupa 3 ograniczenia obszaru pracy WKS aktywna	m	+		<i>PGsl</i>
WALCS4	G	Grupa 4 ograniczenia obszaru pracy WKS aktywna	m	+		<i>PGsl</i>
WALCS5	G	Grupa 5 ograniczenia obszaru pracy WKS aktywna	m	+		<i>PGsl</i>
WALCS6	G	Grupa 6 ograniczenia obszaru pracy WKS aktywna	m	+		<i>PGsl</i>
WALCS7	G	Grupa 7 ograniczenia obszaru pracy WKS aktywna	m	+		<i>PGsl</i>
WALCS8	G	Grupa 8 ograniczenia obszaru pracy WKS aktywna	m	+		<i>PGsl</i>
WALCS9	G	Grupa 9 ograniczenia obszaru pracy WKS aktywna	m	+		<i>PGsl</i>
WALCS10	G	Grupa 10 ograniczenia obszaru pracy WKS aktywna	m	+		<i>PGsl</i>
WALIMOF	G	Ograniczenie obszaru pracy BKS WYŁ.	m	+		<i>PGsl</i>
WALIMON ⁶⁾	G	Ograniczenie obszaru pracy BKS WŁ.	m	+		<i>PGsl</i>
WHEN	K	Akcja jest wykonywana cyklicznie, gdy warunek jest spełniony.		-	+	<i>FBSY</i>
WHENEVER	K	Akcja jest wykonywana jeden raz, gdy warunek zostanie raz spełniony.		-	+	<i>FBSY</i>
WHILE	K	Początek pętli programowej WHILE		+		<i>PGAsl</i> Pętla programowa z warunkiem na początku pętli (WHILE, ENDWHILE) (Strona 116)
WRITE	P	Zapisanie tekstu do systemu plików. Wstawia blok na końcu podanego pliku.		+	-	<i>PGAsl</i> Zapisanie pliku (WRITE) (Strona 143)
WRTPR	P	Powoduje zwłokę zlecenia obróbki nie przerywając przy tym trybu przechodzenia płynnego.		+	-	<i>PGAsl</i>
X	A	Nazwa osi	m/s	+		<i>PGsl</i>
XOR	O	Logiczne ALBO		+		<i>PGAsl</i> Operacje porównania i operacje logiczne (Strona 73)
Y	A	Nazwa osi	m/s	+		<i>PGsl</i>
Z	A	Nazwa osi	m/s	+		<i>PGsl</i>

Legenda

- 1) Rodzaj instrukcji:
- A Adres
Identyfikator, któremu jest przyporządkowywana wartość (np. OVR=10). Jest również kilka adresów, które włączają lub wyłączają funkcję bez przyporządkowania wartości (np. CPLON i CPLOF).
 - C Cykl technologiczny
Wcześniej sporządzony program obróbki, w którym zaprogramowany jest określony cykl (proces obróbki), np. gwintowanie otworu albo frezowanie kieszeni. Dopasowanie do konkretnego procesu obróbki następuje przez parametry, które są przekazywane do cyklu podczas jego wywołania.
 - F Funkcje predefiniowane (wartość zwrotna)
Wywołanie funkcji predefiniowanej może znajdować się w wyrażeniu jako argument.
 - G Funkcja G
Funkcje G są podzielone na grupy funkcji. W jednym bloku można zaprogramować tylko jedną funkcję G z danej grupy. Funkcja G może być modalna (odwoływana przez inną funkcję z tej samej grupy) albo działa tylko w bloku, w którym się znajduje (działająca pojedynczymi blokami).
 - K Słowo kluczowe
Identyfikator, który określa składnię bloku. Do słowa kluczowego nie jest przyporządkowywana wartość i przy jego pomocy nie można też włączyć/wyłączyć żadnej funkcji NC.
Przykłady: struktury kontrolne (IF, ELSE, ENDF, WHEN, ...), przebieg programu (GOTOB, GOTO, RET ...)
 - O Operator
Operator operacji matematycznej, porównania albo logicznej
 - P Procedura predefiniowana (brak wartości zwrotnej)
 - P Atrybut programu
 - A Atrybuty programu są umieszczane na końcu wiersza definicji podprogramu:
PROC <nazwa programu>(…) <atrybut programu>
Określają zachowanie się podprogramu podczas wykonywania.
- 2) Działanie instrukcji:
- m Modalnie
 - s Pojedynczymi blokami
- 3) Możliwość zaprogramowania w programie obróbki:
- + programowane
 - nie programowane
- 4) Możliwość programowania w akcjach synchronicznych:
- + programowane
 - nie programowane
 - T programowane tylko w cyklach technologicznych
- 5) Odsyłacz do dokumentu, który zawiera szczegółowy opis instrukcji:
- PGs/ Podręcznik programowania Podstawy
 - PGAs/ Podręcznik programowania Przygotowanie do pracy
 - BNMs/ Podręcznik programowania Cykle pomiarowe

<i>BHDsl</i>	Podręcznik użytkownika Toczenie
<i>BHFsl</i>	Podręcznik użytkownika Frezowanie
<i>FB1sl</i> ()	Podręcznik funkcjonowania Funkcje podstawowe (z alfanumerycznym skrótem odnośnego opisu działania w nawiasach)
<i>FB2sl</i> ()	Podręcznik funkcjonowania Funkcje rozszerzające (z alfanumerycznym skrótem odnośnego opisu działania w nawiasach)
<i>FB3sl</i> ()	Podręcznik funkcjonowania Funkcje specjalne (z alfanumerycznym skrótem odnośnego opisu działania w nawiasach)
<i>FBSsl</i>	Podręcznik działania Safety Integrated
<i>FBSY</i>	Podręcznik funkcjonowania Akcje synchroniczne
<i>FBWsl</i>	Podręcznik funkcjonowania Zarządzanie narzędziami

⁶⁾ Ustawienie standardowe na początku programu (w stanie przy dostawie sterowania, o ile nie zaprogramowano inaczej).

Rysunek 19-1 Legenda do listy instrukcji

19.2 Instrukcje: Dostępność w przypadku SINUMERIK 828D

Instrukcja	Wariant sterowania 828D					
	PPU240.2 / 241.2		PPU260.2 / 261.2		PPU280.2 / 281.2	
	BASIC T	BASIC M	Toczenie	Frezowanie	Toczenie	Frezowanie
• Standardowy						
○ Opcja						
- niedostępna						
:	•	•	•	•	•	•
*	•	•	•	•	•	•
+	•	•	•	•	•	•
-	•	•	•	•	•	•
<	•	•	•	•	•	•
<<	•	•	•	•	•	•
<=	•	•	•	•	•	•
=	•	•	•	•	•	•
>=	•	•	•	•	•	•
/	•	•	•	•	•	•
/0	•	•	•	•	•	•
...						
...						
/7	○	○	○	○	○	○
A	•	•	•	•	•	•
A2	-	-	-	-	-	-
A3	-	-	-	-	-	-
A4	-	-	-	-	-	-
A5	-	-	-	-	-	-
ABS	•	•	•	•	•	•
AC	•	•	•	•	•	•
ACC	•	•	•	•	•	•
ACCLIMA	•	•	•	•	•	•
ACN	•	•	•	•	•	•
ACOS	•	•	•	•	•	•
ACP	•	•	•	•	•	•
ACTBLOCNO	•	•	•	•	•	•
ADDFRAME	•	•	•	•	•	•
ADIS	•	•	•	•	•	•
ADISPOS	•	•	•	•	•	•
ADISPOSA	•	•	•	•	•	•
ALF	•	•	•	•	•	•
AMIRROR	•	•	•	•	•	•
AND	•	•	•	•	•	•
ANG	•	•	•	•	•	•
AP	•	•	•	•	•	•

Instrukcja	Wariant sterowania 828D					
	PPU240.2 / 241.2		PPU260.2 / 261.2		PPU280.2 / 281.2	
	BASIC T	BASIC M	Toczenie	Frezowanie	Toczenie	Frezowanie
• Standardowy						
○ Opcja						
- niedostępna						
APR	•	•	•	•	•	•
APRB	•	•	•	•	•	•
APRP	•	•	•	•	•	•
APW	•	•	•	•	•	•
APWB	•	•	•	•	•	•
APWP	•	•	•	•	•	•
APX	•	•	•	•	•	•
AR	•	•	•	•	•	•
AROT	•	•	•	•	•	•
AROTS	•	•	•	•	•	•
AS	•	•	•	•	•	•
ASCALE	•	•	•	•	•	•
ASIN	•	•	•	•	•	•
ASPLINE	-	○	-	○	-	○
ATAN2	•	•	•	•	•	•
ATOL	-	•	-	•	-	•
ATRANS	•	•	•	•	•	•
AUXFUDEL	•	•	•	•	•	•
AUXFUDELG	•	•	•	•	•	•
AUXFUMSEQ	•	•	•	•	•	•
AUXFUSYNC	•	•	•	•	•	•
AX	•	•	•	•	•	•
AXCTSWE	-	-	-	-	-	-
AXCTSWEC	-	-	-	-	-	-
AXCTSWED	-	-	-	-	-	-
AXIS	•	•	•	•	•	•
AXNAME	•	•	•	•	•	•
AXSTRING	•	•	•	•	•	•
AXTOCHAN	•	•	•	•	•	•
AXTOINT	•	•	•	•	•	•
AXTOSPI	•	•	•	•	•	•
B	•	•	•	•	•	•
B2	-	-	-	-	-	-
B3	-	-	-	-	-	-
B4	-	-	-	-	-	-
B5	-	-	-	-	-	-
B_AND	•	•	•	•	•	•
B_OR	•	•	•	•	•	•

Instrukcja	Wariant sterowania 828D					
	PPU240.2 / 241.2		PPU260.2 / 261.2		PPU280.2 / 281.2	
	BASIC T	BASIC M	Toczenie	Frezowanie	Toczenie	Frezowanie
• Standardowy						
○ Opcja						
- niedostępna						
B_NOT	•	•	•	•	•	•
B_XOR	•	•	•	•	•	•
BAUTO	-	○	-	○	-	○
BLOCK	•	•	•	•	•	•
BLSYNC	•	•	•	•	•	•
BNAT	-	○	-	○	-	○
BOOL	•	•	•	•	•	•
BOUND	•	•	•	•	•	•
BRISK	•	•	•	•	•	•
BRISKA	•	•	•	•	•	•
BSPLINE	-	○	-	○	-	○
BTAN	-	○	-	○	-	○
C	•	•	•	•	•	•
C2	-	-	-	-	-	-
C3	-	-	-	-	-	-
C4	-	-	-	-	-	-
C5	-	-	-	-	-	-
CAC	•	•	•	•	•	•
CACN	•	•	•	•	•	•
CACP	•	•	•	•	•	•
CALCDAT	•	•	•	•	•	•
CALCPOSI	•	•	•	•	•	•
CALL	•	•	•	•	•	•
CALLPATH	•	•	•	•	•	•
CANCEL	•	•	•	•	•	•
CASE	•	•	•	•	•	•
CDC	•	•	•	•	•	•
CDOF	•	•	•	•	•	•
CDOF2	•	•	•	•	•	•
CDON	•	•	•	•	•	•
CFC	•	•	•	•	•	•
CFIN	•	•	•	•	•	•
CFINE	•	•	•	•	•	•
CFTCP	•	•	•	•	•	•
CHAN	•	•	•	•	•	•
CHANDATA	•	•	•	•	•	•
CHAR	•	•	•	•	•	•
CHF	•	•	•	•	•	•

Instrukcja	Wariant sterowania 828D					
	PPU240.2 / 241.2		PPU260.2 / 261.2		PPU280.2 / 281.2	
	BASIC T	BASIC M	Toczenie	Frezowanie	Toczenie	Frezowanie
• Standardowy						
○ Opcja						
- niedostępna						
CHKDM	•	•	•	•	•	•
CHKDNO	•	•	•	•	•	•
CHR	•	•	•	•	•	•
CIC	•	•	•	•	•	•
CIP	•	•	•	•	•	•
CLEARM	-	-	-	-	-	-
CLRINT	•	•	•	•	•	•
CMIRROR	•	•	•	•	•	•
COARSEA	•	•	•	•	•	•
COLLPAIR	-	-	-	-	-	-
COMPCAD	-	○	-	○	-	○
COMPCURV	-	○	-	○	-	○
COMPLETE	•	•	•	•	•	•
COMPOF	-	○	-	○	-	○
COMPON	-	○	-	○	-	○
CONTDCON	•	•	•	•	•	•
CONTPRON	•	•	•	•	•	•
CORROF	•	•	•	•	•	•
COS	•	•	•	•	•	•
COUPDEF	○	-	○	-	○	-
COUPDEL	○	-	○	-	○	-
COUPOF	○	-	○	-	○	-
COUPOFS	○	-	○	-	○	-
COUPON	○	-	○	-	○	-
COUPONC	○	-	○	-	○	-
COUPRES	○	-	○	-	○	-
CP	•	•	•	•	•	•
CPBC	•	•	•	•	•	•
CPDEF	•	•	•	•	•	•
CPDEL	•	•	•	•	•	•
CPFMOF	•	•	•	•	•	•
CPFMON	•	•	•	•	•	•
CPFMSON	•	•	•	•	•	•
CPFPOS	•	•	•	•	•	•
CPFRS	•	•	•	•	•	•
CPLA	•	•	•	•	•	•
CPLCTID	•	•	•	•	•	•
CPLDEF	•	•	•	•	•	•

Instrukcja	Wariant sterowania 828D					
	PPU240.2 / 241.2		PPU260.2 / 261.2		PPU280.2 / 281.2	
	BASIC T	BASIC M	Toczenie	Frezowanie	Toczenie	Frezowanie
• Standardowy						
○ Opcja						
- niedostępna						
CPLDEL	•	•	•	•	•	•
CPLDEN	•	•	•	•	•	•
CPLINSC	•	•	•	•	•	•
CPLINTR	•	•	•	•	•	•
CPLNUM	•	•	•	•	•	•
CPLOF	•	•	•	•	•	•
CPLON	•	•	•	•	•	•
CPLOUTSC	•	•	•	•	•	•
CPLOUTTR	•	•	•	•	•	•
CPLPOS	•	•	•	•	•	•
CPLSETVAL	•	•	•	•	•	•
CPMALARM	•	•	•	•	•	•
CPMBRAKE	•	•	•	•	•	•
CPMPRT	•	•	•	•	•	•
CPMRESET	•	•	•	•	•	•
CPMSTART	•	•	•	•	•	•
CPMVDI	•	•	•	•	•	•
CPOF	•	•	•	•	•	•
CPON	•	•	•	•	•	•
CPRECOF	•	•	•	•	•	•
CPRECON	•	•	•	•	•	•
CPRES	•	•	•	•	•	•
CPROT	•	•	•	•	•	•
CPROTDEF	•	•	•	•	•	•
CPSETTYPE	•	•	•	•	•	•
CPSYNCOF	•	•	•	•	•	•
CPSYNCOF2	•	•	•	•	•	•
CPSYNCOV	•	•	•	•	•	•
CPSYNFIP	•	•	•	•	•	•
CPSYNFIP2	•	•	•	•	•	•
CPSYNFIV	•	•	•	•	•	•
CR	•	•	•	•	•	•
CROT	•	•	•	•	•	•
CROTS	•	•	•	•	•	•
CRPL	•	•	•	•	•	•
CSCALE	•	•	•	•	•	•
CSPLINE	-	○	-	○	-	○
CT	•	•	•	•	•	•

Instrukcja	Wariant sterowania 828D					
	PPU240.2 / 241.2		PPU260.2 / 261.2		PPU280.2 / 281.2	
	BASIC T	BASIC M	Toczenie	Frezowanie	Toczenie	Frezowanie
• Standardowy						
○ Opcja						
- niedostępna						
CTAB	-	-	-	-	-	-
CTABDEF	-	-	-	-	-	-
CTABDEL	-	-	-	-	-	-
CTABEND	-	-	-	-	-	-
CTABEXISTS	-	-	-	-	-	-
CTABFNO	-	-	-	-	-	-
CTABFPOL	-	-	-	-	-	-
CTABFSEG	-	-	-	-	-	-
CTABID	-	-	-	-	-	-
CTABINV	-	-	-	-	-	-
CTABISLOCK	-	-	-	-	-	-
CTABLOCK	-	-	-	-	-	-
CTABMEMTYP	-	-	-	-	-	-
CTABMPOL	-	-	-	-	-	-
CTABMSEG	-	-	-	-	-	-
CTABNO	-	-	-	-	-	-
CTABNOMEM	-	-	-	-	-	-
CTABPERIOD	-	-	-	-	-	-
CTABPOL	-	-	-	-	-	-
CTABPOLID	-	-	-	-	-	-
CTABSEG	-	-	-	-	-	-
CTABSEGID	-	-	-	-	-	-
CTABSEV	-	-	-	-	-	-
CTABSSV	-	-	-	-	-	-
CTABTEP	-	-	-	-	-	-
CTABTEV	-	-	-	-	-	-
CTABTMAX	-	-	-	-	-	-
CTABTMIN	-	-	-	-	-	-
CTABTSP	-	-	-	-	-	-
CTABTSV	-	-	-	-	-	-
CTABUNLOCK	-	-	-	-	-	-
CTOL	-	○	-	○	-	○
CTRANS	●	●	●	●	●	●
CUT2D	●	●	●	●	●	●
CUT2DF	●	●	●	●	●	●
CUT3DC	-	-	-	-	-	-
CUT3DCC	-	-	-	-	-	-
CUT3DCCD	-	-	-	-	-	-

Instrukcja	Wariant sterowania 828D					
	PPU240.2 / 241.2		PPU260.2 / 261.2		PPU280.2 / 281.2	
	BASIC T	BASIC M	Toczenie	Frezowanie	Toczenie	Frezowanie
• Standardowy						
◦ Opcja						
- niedostępna						
CUT3DF	-	-	-	-	-	-
CUT3DFF	-	-	-	-	-	-
CUT3DFS	-	-	-	-	-	-
CUTCONOF	•	•	•	•	•	•
CUTCONON	•	•	•	•	•	•
CUTMOD	•	•	•	•	•	•
CYCLE...	•	•	•	•	•	•
D	•	•	•	•	•	•
D0	•	•	•	•	•	•
DAC	•	•	•	•	•	•
DC	•	•	•	•	•	•
DEF	•	•	•	•	•	•
DEFINE	•	•	•	•	•	•
DEFAULT	•	•	•	•	•	•
DELAYFSTON	•	•	•	•	•	•
DELAYFSTOF	•	•	•	•	•	•
DELDL	•	•	•	•	•	•
DELDTG	•	•	•	•	•	•
DELETE	•	•	•	•	•	•
DELMLOWNER	•	•	•	•	•	•
DEMLRES	•	•	•	•	•	•
DELMT	•	•	•	•	•	•
DELOBJ	-	-	-	-	-	-
DELT	•	•	•	•	•	•
DELTC	•	•	•	•	•	•
DELTOOLNV	•	•	•	•	•	•
DIACYCOFA	•	•	•	•	•	•
DIAM90	•	•	•	•	•	•
DIAM90A	•	•	•	•	•	•
DIAMCHAN	•	•	•	•	•	•
DIAMCHANA	•	•	•	•	•	•
DIAMCYCOF	•	•	•	•	•	•
DIAMOF	•	•	•	•	•	•
DIAMOFA	•	•	•	•	•	•
DIAMON	•	•	•	•	•	•
DIAMONA	•	•	•	•	•	•
DIC	•	•	•	•	•	•
DILF	•	•	•	•	•	•

Instrukcja	Wariant sterowania 828D					
	PPU240.2 / 241.2		PPU260.2 / 261.2		PPU280.2 / 281.2	
	BASIC T	BASIC M	Toczenie	Frezowanie	Toczenie	Frezowanie
• Standardowy						
○ Opcja						
- niedostępna						
DISABLE	•	•	•	•	•	•
DISC	•	•	•	•	•	•
DISCL	•	•	•	•	•	•
DISPLOF	•	•	•	•	•	•
DISPLON	•	•	•	•	•	•
DISPR	•	•	•	•	•	•
DISR	•	•	•	•	•	•
DISRP	•	•	•	•	•	•
DITE	•	•	•	•	•	•
DITS	•	•	•	•	•	•
DIV	•	•	•	•	•	•
DL	-	-	-	-	-	-
DO	•	•	•	•	•	•
DRFOF	•	•	•	•	•	•
DRIVE	•	•	•	•	•	•
DRIVEA	•	•	•	•	•	•
DYNFINISH	•	•	•	•	•	•
DYNNORM	•	•	•	•	•	•
DYNPOS	•	•	•	•	•	•
DYNROUGH	•	•	•	•	•	•
DYNSEMIFIN	•	•	•	•	•	•
DZERO	•	•	•	•	•	•
EAUTO	-	○	-	○	-	○
EGDEF	-	-	-	-	-	-
EGDEL	-	-	-	-	-	-
EGOFC	-	-	-	-	-	-
EGOFS	-	-	-	-	-	-
EGON	-	-	-	-	-	-
EGONSYN	-	-	-	-	-	-
EGONSYNE	-	-	-	-	-	-
ELSE	•	•	•	•	•	•
ENABLE	•	•	•	•	•	•
ENAT	-	○	-	○	-	○
ENDFOR	•	•	•	•	•	•
ENDIF	•	•	•	•	•	•
ENDLABEL	•	•	•	•	•	•
ENDLOOP	•	•	•	•	•	•
ENDPROC	•	•	•	•	•	•

Instrukcja	Wariant sterowania 828D					
	PPU240.2 / 241.2		PPU260.2 / 261.2		PPU280.2 / 281.2	
	BASIC T	BASIC M	Toczenie	Frezowanie	Toczenie	Frezowanie
• Standardowy						
○ Opcja						
- niedostępna						
ENDWHILE	•	•	•	•	•	•
ESRR	•	•	•	•	•	•
ESRS	•	•	•	•	•	•
ETAN	-	○	-	○	-	○
EVERY	•	•	•	•	•	•
EX	•	•	•	•	•	•
EXECSTRING	•	•	•	•	•	•
EXECTAB	•	•	•	•	•	•
EXECUTE	•	•	•	•	•	•
EXP	•	•	•	•	•	•
EXTCALL	•	•	•	•	•	•
EXTCLOSE	•	•	•	•	•	•
EXTERN	•	•	•	•	•	•
EXTOPEN	•	•	•	•	•	•
F	•	•	•	•	•	•
FA	•	•	•	•	•	•
FAD	•	•	•	•	•	•
FALSE	•	•	•	•	•	•
FB	•	•	•	•	•	•
FCTDEF	-	-	-	-	-	-
FCUB	•	•	•	•	•	•
FD	•	•	•	•	•	•
FDA	•	•	•	•	•	•
FENDNORM	•	•	•	•	•	•
FFWOF	•	•	•	•	•	•
FFWON	•	•	•	•	•	•
FGREF	•	•	•	•	•	•
FGROUP	•	•	•	•	•	•
FI	•	•	•	•	•	•
FIFOCTRL	•	•	•	•	•	•
FILEDATE	•	•	•	•	•	•
FILEINFO	•	•	•	•	•	•
FILESIZE	•	•	•	•	•	•
FILESTAT	•	•	•	•	•	•
FILETIME	•	•	•	•	•	•
FINEA	•	•	•	•	•	•
FL	•	•	•	•	•	•
FLIN	•	•	•	•	•	•

Instrukcja	Wariant sterowania 828D					
	PPU240.2 / 241.2		PPU260.2 / 261.2		PPU280.2 / 281.2	
	BASIC T	BASIC M	Toczenie	Frezowanie	Toczenie	Frezowanie
• Standardowy						
○ Opcja						
- niedostępna						
FMA	-	-	-	-	-	-
FNORM	•	•	•	•	•	•
FOCOF	○	-	○	-	○	-
FOCON	○	-	○	-	○	-
FOR	•	•	•	•	•	•
FP	•	•	•	•	•	•
FPO	-	-	-	-	-	-
FPR	•	•	•	•	•	•
FPRAOF	•	•	•	•	•	•
FPRAON	•	•	•	•	•	•
FRAME	•	•	•	•	•	•
FRC	•	•	•	•	•	•
FRCM	•	•	•	•	•	•
FROM	•	•	•	•	•	•
FTOC	•	•	•	•	•	•
FTOCOF	•	•	•	•	•	•
FTOCON	•	•	•	•	•	•
FXS	•	•	•	•	•	•
FXST	•	•	•	•	•	•
FXSW	•	•	•	•	•	•
FZ	•	•	•	•	•	•
G0	•	•	•	•	•	•
G1	•	•	•	•	•	•
G2	•	•	•	•	•	•
G3	•	•	•	•	•	•
G4	•	•	•	•	•	•
G5	•	•	•	•	•	•
G7	•	•	•	•	•	•
G9	•	•	•	•	•	•
G17	•	•	•	•	•	•
G18	•	•	•	•	•	•
G19	•	•	•	•	•	•
G25	•	•	•	•	•	•
G26	•	•	•	•	•	•
G33	•	•	•	•	•	•
G34	•	•	•	•	•	•
G35	•	•	•	•	•	•
G40	•	•	•	•	•	•

Instrukcja	Wariant sterowania 828D					
	PPU240.2 / 241.2		PPU260.2 / 261.2		PPU280.2 / 281.2	
	BASIC T	BASIC M	Toczenie	Frezowanie	Toczenie	Frezowanie
• Standardowy						
○ Opcja						
- niedostępna						
G41	•	•	•	•	•	•
G42	•	•	•	•	•	•
G53	•	•	•	•	•	•
G54	•	•	•	•	•	•
G55	•	•	•	•	•	•
G56	•	•	•	•	•	•
G57	•	•	•	•	•	•
G58	•	•	•	•	•	•
G59	•	•	•	•	•	•
G60	•	•	•	•	•	•
G62	•	•	•	•	•	•
G63	•	•	•	•	•	•
G64	•	•	•	•	•	•
G70	•	•	•	•	•	•
G71	•	•	•	•	•	•
G74	•	•	•	•	•	•
G75	•	•	•	•	•	•
G90	•	•	•	•	•	•
G91	•	•	•	•	•	•
G93	•	•	•	•	•	•
G94	•	•	•	•	•	•
G95	•	•	•	•	•	•
G96	•	•	•	•	•	•
G97	•	•	•	•	•	•
G110	•	•	•	•	•	•
G111	•	•	•	•	•	•
G112	•	•	•	•	•	•
G140	•	•	•	•	•	•
G141	•	•	•	•	•	•
G142	•	•	•	•	•	•
G143	•	•	•	•	•	•
G147	•	•	•	•	•	•
G148	•	•	•	•	•	•
G153	•	•	•	•	•	•
G247	•	•	•	•	•	•
G248	•	•	•	•	•	•
G290	•	•	•	•	•	•
G291	•	•	•	•	•	•

Instrukcja	Wariant sterowania 828D					
	PPU240.2 / 241.2		PPU260.2 / 261.2		PPU280.2 / 281.2	
	BASIC T	BASIC M	Toczenie	Frezowanie	Toczenie	Frezowanie
• Standardowy						
○ Opcja						
- niedostępna						
G331	•	•	•	•	•	•
G332	•	•	•	•	•	•
G340	•	•	•	•	•	•
G341	•	•	•	•	•	•
G347	•	•	•	•	•	•
G348	•	•	•	•	•	•
G450	•	•	•	•	•	•
G451	•	•	•	•	•	•
G460	•	•	•	•	•	•
G461	•	•	•	•	•	•
G462	•	•	•	•	•	•
G500	•	•	•	•	•	•
G505 ... G599	•	•	•	•	•	•
G601	•	•	•	•	•	•
G602	•	•	•	•	•	•
G603	•	•	•	•	•	•
G621	•	•	•	•	•	•
G641	•	•	•	•	•	•
G642	•	•	•	•	•	•
G643	•	•	•	•	•	•
G644	•	•	•	•	•	•
G645	•	•	•	•	•	•
G700	•	•	•	•	•	•
G710	•	•	•	•	•	•
G810 ... G819	-	-	-	-	-	-
G820 ... G829	-	-	-	-	-	-
G931	•	•	•	•	•	•
G942	•	•	•	•	•	•
G952	•	•	•	•	•	•
G961	•	•	•	•	•	•
G962	•	•	•	•	•	•
G971	•	•	•	•	•	•
G972	•	•	•	•	•	•
G973	•	•	•	•	•	•
GEOAX	•	•	•	•	•	•
GET	•	•	•	•	•	•
GETACTT	•	•	•	•	•	•
GETACTTD	•	•	•	•	•	•

Instrukcja	Wariant sterowania 828D					
	PPU240.2 / 241.2		PPU260.2 / 261.2		PPU280.2 / 281.2	
	BASIC T	BASIC M	Toczenie	Frezowanie	Toczenie	Frezowanie
• Standardowy						
○ Opcja						
- niedostępna						
GETD	•	•	•	•	•	•
GETDNO	•	•	•	•	•	•
GETEXET	•	•	•	•	•	•
GETFREELOC	•	•	•	•	•	•
GETSELT	•	•	•	•	•	•
GETT	•	•	•	•	•	•
GETTCOR	•	•	•	•	•	•
GETTENV	•	•	•	•	•	•
GETVARAP	•	•	•	•	•	•
GETVARDFT	•	•	•	•	•	•
GETVARLIM	•	•	•	•	•	•
GETVARPHU	•	•	•	•	•	•
GETVARTYP	•	•	•	•	•	•
GOTO	•	•	•	•	•	•
GOTOB	•	•	•	•	•	•
GOTOC	•	•	•	•	•	•
GOTOF	•	•	•	•	•	•
GOTOS	•	•	•	•	•	•
GP	•	•	•	•	•	•
GWPSOF	•	•	•	•	•	•
GWPSON	•	•	•	•	•	•
H...	•	•	•	•	•	•
HOLES1	•	•	•	•	•	•
HOLES2	•	•	•	•	•	•
I	•	•	•	•	•	•
I1	•	•	•	•	•	•
IC	•	•	•	•	•	•
ICYCOF	•	•	•	•	•	•
ICYCON	•	•	•	•	•	•
ID	•	•	•	•	•	•
IDS	•	•	•	•	•	•
IF	•	•	•	•	•	•
INDEX	•	•	•	•	•	•
INIPO	•	•	•	•	•	•
INIRE	•	•	•	•	•	•
INICF	•	•	•	•	•	•
INIT	-	-	-	-	-	-
INITIAL	•	•	•	•	•	•

Instrukcja	Wariant sterowania 828D					
	PPU240.2 / 241.2		PPU260.2 / 261.2		PPU280.2 / 281.2	
	BASIC T	BASIC M	Toczenie	Frezowanie	Toczenie	Frezowanie
• Standardowy						
○ Opcja						
- niedostępna						
INT	•	•	•	•	•	•
INTERSEC	•	•	•	•	•	•
INTTOAX	•	•	•	•	•	•
INVCCW	-	-	-	-	-	-
INVCW	-	-	-	-	-	-
INVFRAME	•	•	•	•	•	•
IP	•	•	•	•	•	•
IPOBRKA	•	•	•	•	•	•
IPOENDA	•	•	•	•	•	•
IPTRLOCK	•	•	•	•	•	•
IPTRUNLOCK	•	•	•	•	•	•
ISAXIS	•	•	•	•	•	•
ISD	-	-	-	-	-	-
ISFILE	•	•	•	•	•	•
ISNUMBER	•	•	•	•	•	•
ISOCALL	•	•	•	•	•	•
ISVAR	•	•	•	•	•	•
J	•	•	•	•	•	•
J1	•	•	•	•	•	•
JERKA	•	•	•	•	•	•
JERKLIM	•	•	•	•	•	•
JERKLIMA	•	•	•	•	•	•
K	•	•	•	•	•	•
K1	•	•	•	•	•	•
KONT	•	•	•	•	•	•
KONTC	•	•	•	•	•	•
KONTT	•	•	•	•	•	•
L	•	•	•	•	•	•
LEAD						
Orientacja narzędzia	-	-	-	-	-	-
Wielomian orientacji	-	-	-	-	-	-
LEADOF	-	-	-	-	-	-
LEADON	-	-	-	-	-	-
LENTOAX	•	•	•	•	•	•
LFOF	•	•	•	•	•	•
LFON	•	•	•	•	•	•
LFPOS	•	•	•	•	•	•
LFTXT	•	•	•	•	•	•

Instrukcja	Wariant sterowania 828D					
	PPU240.2 / 241.2		PPU260.2 / 261.2		PPU280.2 / 281.2	
	BASIC T	BASIC M	Toczenie	Frezowanie	Toczenie	Frezowanie
• Standardowy						
○ Opcja						
- niedostępna						
LFWP	•	•	•	•	•	•
LIFTFAST	•	•	•	•	•	•
LIMS	•	•	•	•	•	•
LLI	•	•	•	•	•	•
LN	•	•	•	•	•	•
LOCK	•	•	•	•	•	•
LONGHOLE	-	-	-	-	-	-
LOOP	•	•	•	•	•	•
M0	•	•	•	•	•	•
M1	•	•	•	•	•	•
M2	•	•	•	•	•	•
M3	•	•	•	•	•	•
M4	•	•	•	•	•	•
M5	•	•	•	•	•	•
M6	•	•	•	•	•	•
M17	•	•	•	•	•	•
M19	•	•	•	•	•	•
M30	•	•	•	•	•	•
M40	•	•	•	•	•	•
M41 ... M45	•	•	•	•	•	•
M70	•	•	•	•	•	•
MASLDEF	•	•	•	•	•	•
MASLDEL	•	•	•	•	•	•
MASLOF	•	•	•	•	•	•
MASLOFS	•	•	•	•	•	•
MASLON	•	•	•	•	•	•
MATCH	•	•	•	•	•	•
MAXVAL	•	•	•	•	•	•
MCALL	•	•	•	•	•	•
MEAC	-	-	-	-	-	-
MEAFRAME	•	•	•	•	•	•
MEAS	•	•	•	•	•	•
MEASA	-	-	-	-	-	-
MEASURE	•	•	•	•	•	•
MEAW	•	•	•	•	•	•
MEAWA	-	-	-	-	-	-
MI	•	•	•	•	•	•
MINDEX	•	•	•	•	•	•

Instrukcja	Wariant sterowania 828D					
	PPU240.2 / 241.2		PPU260.2 / 261.2		PPU280.2 / 281.2	
	BASIC T	BASIC M	Toczenie	Frezowanie	Toczenie	Frezowanie
• Standardowy						
○ Opcja						
- niedostępna						
MINVAL	•	•	•	•	•	•
MIRROR	•	•	•	•	•	•
MMC	•	•	•	•	•	•
MOD	•	•	•	•	•	•
MODAXVAL	•	•	•	•	•	•
MOV	•	•	•	•	•	•
MOVT	•	•	•	•	•	•
MSG	•	•	•	•	•	•
MVTOOL	•	•	•	•	•	•
N	•	•	•	•	•	•
NAMETOINT	-	-	-	-	-	-
NCK	•	•	•	•	•	•
NEWCONF	•	•	•	•	•	•
NEWMT	•	•	•	•	•	•
NEWT	•	•	•	•	•	•
NORM	•	•	•	•	•	•
NOT	•	•	•	•	•	•
NPROT	•	•	•	•	•	•
NPROTDEF	•	•	•	•	•	•
NUMBER	•	•	•	•	•	•
OEMIPO1	-	-	-	-	-	-
OEMIPO2	-	-	-	-	-	-
OF	•	•	•	•	•	•
OFFN	•	•	•	•	•	•
OMA1	-	-	-	-	-	-
OMA2	-	-	-	-	-	-
OMA3	-	-	-	-	-	-
OMA4	-	-	-	-	-	-
OMA5	-	-	-	-	-	-
OR	•	•	•	•	•	•
ORIAXES	-	-	-	-	-	-
ORIAXPOS	-	-	-	-	-	-
ORIC	-	-	-	-	-	-
ORICONCCW	-	-	-	-	-	-
ORICONCW	-	-	-	-	-	-
ORICONIO	-	-	-	-	-	-
ORICONTO	-	-	-	-	-	-
ORICURVE	-	-	-	-	-	-

Instrukcja	Wariant sterowania 828D					
	PPU240.2 / 241.2		PPU260.2 / 261.2		PPU280.2 / 281.2	
	BASIC T	BASIC M	Toczenie	Frezowanie	Toczenie	Frezowanie
• Standardowy						
◦ Opcja						
- niedostępna						
ORID	-	-	-	-	-	-
ORIEULER	-	-	-	-	-	-
ORIMKS	-	-	-	-	-	-
ORIPATH	-	-	-	-	-	-
ORIPATHS	-	-	-	-	-	-
ORIPLANE	-	-	-	-	-	-
ORIRESET	-	-	-	-	-	-
ORIROTA	-	-	-	-	-	-
ORIROTC	-	-	-	-	-	-
ORIROTR	-	-	-	-	-	-
ORIROTT	-	-	-	-	-	-
ORIRPY	-	-	-	-	-	-
ORIRPY2	-	-	-	-	-	-
ORIS	-	-	-	-	-	-
ORISOF	-	-	-	-	-	-
ORISON	-	-	-	-	-	-
ORIVECT	-	-	-	-	-	-
ORIVIRT1	-	-	-	-	-	-
ORIVIRT2	-	-	-	-	-	-
ORIWKS	-	-	-	-	-	-
OS	-	-	-	-	-	-
OSB	-	-	-	-	-	-
OSC	-	-	-	-	-	-
OSCILL	-	-	-	-	-	-
OSCTRL	-	-	-	-	-	-
OSD	-	-	-	-	-	-
OSE	-	-	-	-	-	-
OSNSC	-	-	-	-	-	-
OSOF	-	-	-	-	-	-
OSP1	-	-	-	-	-	-
OSP2	-	-	-	-	-	-
OSS	-	-	-	-	-	-
OSSE	-	-	-	-	-	-
OST	-	-	-	-	-	-
OST1	-	-	-	-	-	-
OST2	-	-	-	-	-	-
OTOL	-	•	-	•	-	•
OVR	•	•	•	•	•	•

Instrukcja	Wariant sterowania 828D					
	PPU240.2 / 241.2		PPU260.2 / 261.2		PPU280.2 / 281.2	
	BASIC T	BASIC M	Toczenie	Frezowanie	Toczenie	Frezowanie
• Standardowy						
○ Opcja						
- niedostępna						
OVRA	•	•	•	•	•	•
OVRRAP	•	•	•	•	•	•
P	•	•	•	•	•	•
PAROT	•	•	•	•	•	•
PAROTOF	•	•	•	•	•	•
PCALL	•	•	•	•	•	•
PDELAYOF	-	-	-	-	-	-
PDELAYON	-	-	-	-	-	-
PHI	-	-	-	-	-	-
PHU	•	•	•	•	•	•
PL	-	○	-	○	-	○
	-	-	-	-	-	-
PM	•	•	•	•	•	•
PO	-	-	-	-	-	-
POCKET3	•	•	•	•	•	•
POCKET4	•	•	•	•	•	•
POLF	•	•	•	•	•	•
POLFA	•	•	•	•	•	•
POLFMASK	•	•	•	•	•	•
POLFMLIN	•	•	•	•	•	•
POLY	-	-	-	-	-	-
POLYPATH	-	-	-	-	-	-
PON	-	-	-	-	-	-
PONS	-	-	-	-	-	-
POS	•	•	•	•	•	•
POSA	•	•	•	•	•	•
POSM	•	•	•	•	•	•
POSMT	•	•	•	•	•	•
POSP	•	•	•	•	•	•
POSRANGE	•	•	•	•	•	•
POT	•	•	•	•	•	•
PR	•	•	•	•	•	•
PREPRO	•	•	•	•	•	•
PRESETON	•	•	•	•	•	•
PRIO	•	•	•	•	•	•
PRLOC	•	•	•	•	•	•
PROC	•	•	•	•	•	•

Instrukcja	Wariant sterowania 828D					
	PPU240.2 / 241.2		PPU260.2 / 261.2		PPU280.2 / 281.2	
	BASIC T	BASIC M	Toczenie	Frezowanie	Toczenie	Frezowanie
• Standardowy						
○ Opcja						
- niedostępna						
PROTA	-	-	-	-	-	-
PROTD	-	-	-	-	-	-
PROTS	-	-	-	-	-	-
PSI	-	-	-	-	-	-
PTP	•	•	•	•	•	•
PTPG0	•	•	•	•	•	•
PUNCHACC	-	-	-	-	-	-
PUTFTOC	•	•	•	•	•	•
PUTFTOCF	•	•	•	•	•	•
PW	-	○	-	○	-	○
QU	•	•	•	•	•	•
R...	•	•	•	•	•	•
RAC	•	•	•	•	•	•
RDISABLE	•	•	•	•	•	•
READ	•	•	•	•	•	•
REAL	•	•	•	•	•	•
REDEF	•	•	•	•	•	•
RELEASE	•	•	•	•	•	•
REP	•	•	•	•	•	•
REPEAT	•	•	•	•	•	•
REPEATB	•	•	•	•	•	•
REPOSA	•	•	•	•	•	•
REPOSH	•	•	•	•	•	•
REPOSHA	•	•	•	•	•	•
REPOSL	•	•	•	•	•	•
REPOSQ	•	•	•	•	•	•
REPOSQA	•	•	•	•	•	•
RESET	•	•	•	•	•	•
RESETMON	•	•	•	•	•	•
RET	•	•	•	•	•	•
RIC	•	•	•	•	•	•
RINDEX	•	•	•	•	•	•
RMB	•	•	•	•	•	•
RME	•	•	•	•	•	•
RMI	•	•	•	•	•	•
RMN	•	•	•	•	•	•
RND	•	•	•	•	•	•
RNDM	•	•	•	•	•	•

Instrukcja	Wariant sterowania 828D					
	PPU240.2 / 241.2		PPU260.2 / 261.2		PPU280.2 / 281.2	
	BASIC T	BASIC M	Toczenie	Frezowanie	Toczenie	Frezowanie
• Standardowy						
◦ Opcja						
- niedostępna						
ROT	•	•	•	•	•	•
ROTS	•	•	•	•	•	•
ROUND	•	•	•	•	•	•
ROUNDUP	•	•	•	•	•	•
RP	•	•	•	•	•	•
RPL	•	•	•	•	•	•
RT	•	•	•	•	•	•
RTLIOF	•	•	•	•	•	•
RTLION	•	•	•	•	•	•
S	•	•	•	•	•	•
SAVE	•	•	•	•	•	•
SBLOF	•	•	•	•	•	•
SBLON	•	•	•	•	•	•
SC	•	•	•	•	•	•
SCALE	•	•	•	•	•	•
SCC	•	•	•	•	•	•
SCPARA	•	•	•	•	•	•
SD	-	◦	-	◦	-	◦
SET	•	•	•	•	•	•
SETAL	•	•	•	•	•	•
SETDNO	•	•	•	•	•	•
SETINT	•	•	•	•	•	•
SETM	-	-	-	-	-	-
SETMS	•	•	•	•	•	•
SETMS(n)	•	•	•	•	•	•
SETMTH	•	•	•	•	•	•
SETPIECE	•	•	•	•	•	•
SETTA	•	•	•	•	•	•
SETTCOR	•	•	•	•	•	•
SETTIA	•	•	•	•	•	•
SF	•	•	•	•	•	•
SIN	•	•	•	•	•	•
SIRELAY	-	-	-	-	-	-
SIRELIN	-	-	-	-	-	-
SIRELOUT	-	-	-	-	-	-
SIRELTIME	-	-	-	-	-	-
SLOT1	•	•	•	•	•	•
SLOT2	•	•	•	•	•	•

Instrukcja	Wariant sterowania 828D					
	PPU240.2 / 241.2		PPU260.2 / 261.2		PPU280.2 / 281.2	
	BASIC T	BASIC M	Toczenie	Frezowanie	Toczenie	Frezowanie
• Standardowy						
○ Opcja						
- niedostępna						
SOFT	•	•	•	•	•	•
SOFTA	•	•	•	•	•	•
SON	-	-	-	-	-	-
SONS	-	-	-	-	-	-
SPATH	•	•	•	•	•	•
SPCOF	•	•	•	•	•	•
SPCON	•	•	•	•	•	•
SPI	•	•	•	•	•	•
SPIF1	-	-	-	-	-	-
SPIF2	-	-	-	-	-	-
SPLINEPATH	-	○	-	○	-	○
SPN	-	-	-	-	-	-
SPOF	-	-	-	-	-	-
SPOS	•	•	•	•	•	•
SPOSA	•	•	•	•	•	•
SPP	-	-	-	-	-	-
SPRINT	•	•	•	•	•	•
SQRT	•	•	•	•	•	•
SR	-	-	-	-	-	-
SRA	-	-	-	-	-	-
ST	-	-	-	-	-	-
STA	-	-	-	-	-	-
START	-	-	-	-	-	-
STARTFIFO	•	•	•	•	•	•
STAT	•	•	•	•	•	•
STOLF	-	-	-	-	-	-
STOPFIFO	•	•	•	•	•	•
STOPRE	•	•	•	•	•	•
STOPREOF	•	•	•	•	•	•
STRING	•	•	•	•	•	•
STRINGFELD	•	•	•	•	•	•
STRINGIS	•	•	•	•	•	•
STRINGVAR	-	-	-	-	-	-
STRLEN	•	•	•	•	•	•
SUBSTR	•	•	•	•	•	•
SUPA	•	•	•	•	•	•
SVC	•	•	•	•	•	•
SYNFCT	•	•	•	•	•	•

Instrukcja	Wariant sterowania 828D					
	PPU240.2 / 241.2		PPU260.2 / 261.2		PPU280.2 / 281.2	
	BASIC T	BASIC M	Toczenie	Frezowanie	Toczenie	Frezowanie
• Standardowy						
○ Opcja						
- niedostępna						
SYNR	•	•	•	•	•	•
SYNRW	•	•	•	•	•	•
SYNW	•	•	•	•	•	•
T	•	•	•	•	•	•
TAN	•	•	•	•	•	•
TANG	-	-	-	-	-	-
TANGDEL	-	-	-	-	-	-
TANGOF	-	-	-	-	-	-
TANGON	-	-	-	-	-	-
TCA (828D: _TCA)	•	•	•	•	•	•
TCARR	-	•	-	•	-	•
TCI	•	•	•	•	•	•
TCOABS	-	•	-	•	-	•
TCOFR	-	•	-	•	-	•
TCOFRX	-	•	-	•	-	•
TCOFRY	-	•	-	•	-	•
TCOFRZ	-	•	-	•	-	•
THETA	-	-	-	-	-	-
TILT	-	-	-	-	-	-
TLIFT	-	-	-	-	-	-
TML	•	•	•	•	•	•
TMOF	•	•	•	•	•	•
TMON	•	•	•	•	•	•
TO	•	•	•	•	•	•
TOFF	•	•	•	•	•	•
TOFFL	•	•	•	•	•	•
TOFFOF	•	•	•	•	•	•
TOFFON	•	•	•	•	•	•
TOFFR	•	•	•	•	•	•
TOFRAME	•	•	•	•	•	•
TOFRAMEX	•	•	•	•	•	•
TOFRAMEY	•	•	•	•	•	•
TOFRAMEZ	•	•	•	•	•	•
TOLOWER	•	•	•	•	•	•
TOOLENV	•	•	•	•	•	•
TOOLGNT	•	•	•	•	•	•
TOOLGT	•	•	•	•	•	•

Instrukcja	Wariant sterowania 828D					
	PPU240.2 / 241.2		PPU260.2 / 261.2		PPU280.2 / 281.2	
	BASIC T	BASIC M	Toczenie	Frezowanie	Toczenie	Frezowanie
• Standardowy						
○ Opcja						
- niedostępna						
TOROT	•	•	•	•	•	•
TOROTOF	•	•	•	•	•	•
TOROTX	•	•	•	•	•	•
TOROTY	•	•	•	•	•	•
TOROTZ	•	•	•	•	•	•
TOUPPER	•	•	•	•	•	•
TOWBCS	-	•	-	•	-	•
TOWKCS	-	•	-	•	-	•
TOWMCS	-	•	-	•	-	•
TOWSTD	-	•	-	•	-	•
TOWTCS	-	•	-	•	-	•
TOWWCS	-	•	-	•	-	•
TR	•	•	•	•	•	•
TRAANG	-	-	-	-	○	-
TRACON	-	-	-	-	○	-
TRACYL	○	○	○	○	○	○
TRAFOOF	•	•	•	•	•	•
TRAILOF	•	•	•	•	•	•
TRAILON	•	•	•	•	•	•
TRANS	•	•	•	•	•	•
TRANSMIT	○	○	○	○	○	○
TRAORI	-	•	-	•	-	•
TRUE	•	•	•	•	•	•
TRUNC	•	•	•	•	•	•
TU	•	•	•	•	•	•
TURN	•	•	•	•	•	•
ULI	•	•	•	•	•	•
UNLOCK	•	•	•	•	•	•
UNTIL	•	•	•	•	•	•
UPATH	•	•	•	•	•	•
VAR	•	•	•	•	•	•
VELOLIM	•	•	•	•	•	•
VELOLIMA	•	•	•	•	•	•
WAITC	-	-	-	-	○	-
WAITE	-	-	-	-	-	-
WAITENC	-	-	-	-	-	-
WAITM	-	-	-	-	-	-
WAITMC	-	-	-	-	-	-

Instrukcja	Wariant sterowania 828D					
	PPU240.2 / 241.2		PPU260.2 / 261.2		PPU280.2 / 281.2	
	BASIC T	BASIC M	Toczenie	Frezowanie	Toczenie	Frezowanie
• Standardowy						
○ Opcja						
- niedostępna						
WAITP	•	•	•	•	•	•
WAITS	•	•	•	•	•	•
WALCS0	•	•	•	•	•	•
WALCS1	•	•	•	•	•	•
WALCS2	•	•	•	•	•	•
WALCS3	•	•	•	•	•	•
WALCS4	•	•	•	•	•	•
WALCS5	•	•	•	•	•	•
WALCS6	•	•	•	•	•	•
WALCS7	•	•	•	•	•	•
WALCS8	•	•	•	•	•	•
WALCS9	•	•	•	•	•	•
WALCS10	•	•	•	•	•	•
WALIMOF	•	•	•	•	•	•
WALIMON	•	•	•	•	•	•
WHEN	•	•	•	•	•	•
WHENEVER	•	•	•	•	•	•
WHILE	•	•	•	•	•	•
WRITE	•	•	•	•	•	•
WRTPR	•	•	•	•	•	•
X	•	•	•	•	•	•
XOR	•	•	•	•	•	•
Y	•	•	•	•	•	•
Z	•	•	•	•	•	•

19.3 Aktualny język w HMI

Poniższa tablica zawiera wszystkie języki dostępne w interfejsie graficznym.

Aktualnie ustawiony język można odczytać w programie obróbki i w akcjach synchronicznych poprzez następującą zmienną systemową:

\$AN_LANGUAGE_ON_HMI = <wartość>

<wartość>	Język	Skrót języka
1	Niemiecki (Niemcy)	DEU
2	Francuski	FRA
3	Angielski (Zjednoczone Królestwo)	ENG
4	Hiszpański	ESP
6	Włoski	ITA
7	Niderlandzki	NLD
8	Chiński uproszczony	CHS
9	Szwedzki	SVE
18	Węgierski	HUN
19	Fiński	FIN
28	Czeski	CSY
50	Portugalski (Brazylia)	PTB
53	Polski	PLK
55	Duński	DAN
57	Rosyjski	RUS
68	Słowacki	SKY
72	Rumuński	ROM
80	Chiński (tradycyjny)	CHT
85	koreański	KOR
87	Japoński	JPN
89	Turecki	TRK

Wskazówka

Aktualizacja \$AN_LANGUAGE_ON_HMI następuje:

- po rozruchu systemu.
- po zresetowaniu NCK i/albo PLC.
- po przełączeniu na inny NCK w ramach M2N.
- po przełączeniu języka na HMI.

Aneks

A

A.1 Lista skrótów

A	
A	Wyjście
ADI4	Analogowy interfejs napędu dla 4 osi
AC	Sterowanie adaptacyjne
ALM	Active Line Module
ARM	Asynchroniczny silnik obrotowy
AS	System automatyzacji
ASCII	American Standard Code for Information Interchange: Amerykańska norma kodów do wymiany informacji
ASIC	Application Specific Integrated Circuit: Obwód sterownia użytkownika
ASUP	Podprogram asynchroniczny
AUXFU	Auxiliary Function: Funkcja pomocnicza
AWL	Lista instrukcji
AWP	Program użytkownika

B	
BA	Tryb pracy
BAG	Grupa trybów pracy
BCD	Binary Coded Decimals: Liczby dziesiętne zakodowane w kodzie binarnym
BERO	Bezdotkowy wyłącznik krańcowy z sprzężonym zwrotnie oscylatorem
BI	Binector Input
BICO	Binector Connector
BIN	Binary Files: Dane binarne
BIOS	Basic Input Output System
BKS	Bazowy układ współrzędnych
BO	Binector Output
BTSS	Interfejs pulpitu obsługi

C	
CAD	Computer-Aided Design
CAM	Computer-Aided Manufacturing
CC	Compile Cycle: Cykle kompilacyjne

C	
CI	Connector Input
CF-Card	Compact Flash-Card
CNC	Computerized Numerical Control: Komputerowe sterowanie numeryczne
CO	Connector Output
CoL	Certificate of License
COM	Communication
CPA	Compiler Projecting Data: Dane projektowe kompilatora
CRT	Cathode Ray Tube: Kineskop
CSB	Central Service Board: Zespół składowy PLC
CU	Control Unit
CP	Communication Processor
CPU	Central Processing Unit: Jednostka centralna komputera
CR	Carriage Return
CTS	Clear To Send: Komunikat gotowości do wysyłania w przypadku szeregowych interfejsów danych
CUTCOM	Cutter Radius Compensation: Korekcja promienia narzędzia

D	
DAU	Przetwornik cyfrowo-analogowy
DB	Moduł danych (PLC)
DBB	Bajt modułu danych (PLC)
DBD	Podwójne słowo modułu danych (PLC)
DBW	Słowo modułu danych (PLC)
DBX	Bit modułu danych (PLC)
DDE	Dynamic Data Exchange
DIN	Niemiecka Norma Przemysłowa
DIO	Data Input/Output: Wyświetlanie przesyłanych danych
DIR	Directory: Wykaz
DLL	Dynamic Link Library
DO	Drive Object
DPM	Dual Port Memory
DPR	Dual Port RAM
DRAM	Pamięć dynamiczna (nie buforowana)
DRF	Differential Resolver Function: Funkcja selsynu obrotowego (kółko ręczne)
DRIVE-CLiQ	Drive Component Link with IQ
DRY	Dry Run: Posuw próbny
DSB	Decoding Single Block: Dekodowanie pojedynczymi blokami
DSC	Dynamic Servo Control / Dynamic Stiffness Control
DW	Słowo danych
DWORD	Podwójne słowo (aktualnie 32 bity)

E	
E	Wejście
E/A	Wprowadzenie/wyprowadzenie
ENC	Encoder: Przetwornik wartości rzeczywistej
EFP	Zwykły moduł peryferyjny (zespół konstrukcyjny PLC–E/A)
EGB	Elektronicznie zagrożone zespoły/elementy konstrukcyjne
EMV	Kompatybilność elektromagnetyczna
EN	Norma Europejska
EnDat	Interfejs przetwornika
EPROM	Erasable Programmable Read Only Memory: Kasowalna, elektrycznie programowana pamięć do odczytu
ePS Network Services	Zdalne usługi serwisowe poprzez łącza internetowe w celu zdalnej konserwacji maszyny
EQN	Określenie typu przetwornika wartości absolutnej o 2048 sygnałach sinusoidalnych/obrót
ES	Engineering System
ESR	Rozszerzone zatrzymanie i wycofanie
ETC	Przycisk ETC ">"; rozszerzenie paska przycisków w tym samym menu

F	
FB	Moduł funkcjonalny (PLC)
FC	Function Call: Moduł funkcjonalny (PLC)
FEPROM	Flash-EPROM: Pamięć do odczytu i zapisu
FIFO	First In First Out: Pamięć, która pracuje bez podania adresu i której dane są czytane w tej samej kolejności, w jakiej zostały zapisane
FIPO	Interpolator dokładny
FPU	Floating Point Unit: Jednostka zmiennoprzecinkowa
FRK	Korekcja promienia frezu
FST	Feed Stop: Posuw stop
FUP	Plan funkcji (metoda programowania PLC)
FW	Oprogramowanie sprzętowe

G	
GC	Global Control (PROFIBUS: Telegram Broadcast)
GEO	Geometria, np. oś geometryczna
GIA	Gear Interpolation Data: Dane interpolacyjne przekładni
GND	Signal Ground
GP	Program podstawowy (PLC)
GS	Stopień przekładni
GSD	Biblioteka danych stałych sprzętu do opisu PROFIBUS Slave

G	
GSDML	Generic Station Description Markup Language: Bazujący na XML język opisowy do sporządzania pliku GSD
GUD	Global User Data: Globalne dane użytkownika

H	
HEX	Skrót liczby szesnastkowej
HiFu	Funkcja pomocnicza
HLA	Hydrauliczny napęd liniowy
HMI	Human Machine Interface: Interfejs graficzny SINUMERIK
HSA	Napęd wrzeciona głównego
HW	Hardware

I	
IBN	Uruchomienie
IKA	Kompensacja interpolacyjna
IM	Interface-Modul: Przyłączeniowy zespół konstrukcyjny
IMR	Interface-Modul Receive: Przyłączeniowy zespół konstrukcyjny dla odbioru
IMS	Interface-Modul Send: Przyłączeniowy zespół konstrukcyjny dla wysyłania
INC	Increment: Przyrost
INI	Initializing Data: Dane inicjalizacyjne
IPO	Interpolator
ISA	International Standard Architecture
ISO	International Standard Organization

J	
JOG	Jogging: Tryb ustawiania

K	
K _v	Współczynnik wzmocnienia obwodu regulacji
K _p	Wzmocnienie proporcjonalne
K _ü	Stosunek przełożenia
KOP	Plan styków (metoda programowania PLC)

L	
LAI	Logic Machine Axis Image: Logiczne odwzorowanie osi maszyny
LAN	Local Area Network
LCD	Liquid-Crystal Display: Wyświetlacz ciekłokrystaliczny
LED	Light Emitting Diode: Dioda świetlna
LF	Line Feed
LMS	System pomiaru położenia
LR	Regulator położenia
LSB	Least Significant Bit: bit najmniej znaczący
LUD	Local User Data: Dane użytkownika (lokalnie)

M	
MAC	Media Access Control
MAIN	Main program: Program główny (OB1, PLC)
MB	Megabajt
MCI	Motion Control Interface
MCIS	Motion-Control-Information-System
MCP	Machine Control Panel: Pulpit sterowniczy maszyny
MD	Dana maszynowa wzgl. dane maszynowe
MDA	Manual Data Automatic: Wprowadzanie ręczne
MELDW	Słowo komunikatu
MKS	Układ współrzędnych maszyny
MLFB	Określenie wyrobu czytane przez maszynę
MM	Motor Module
MPF	Main Program File: Program główny (NC)
MSTT	Pulpit sterowniczy maszyny

N	
NC	Numerical Control: Sterowanie numeryczne
NCK	Numerical Control Kernel: Jądro sterowania numerycznego z przygotowywaniem bloków, zakresem ruchów, itd.
NCU	Numerical Control Unit: Jednostka sprzętowa NCK
NRK	Określenie systemu operacyjnego NCK
NST	Sygnal interfejsowy
NURBS	Non-Uniform Rational B-Spline
NV	Przesunięcie punktu zerowego
NX	Numerical Extension: Moduł rozszerzania osi

O	
OB	Moduł organizacyjny w PLC
OEM	Original Equipment Manufacturer
OP	Operation Panel: Urządzenie obsługi
OPI	Operation Panel Interface: Przyłącze pulpitu obsługi
OPT	Options: Opcje
OLP	Optical Link Plug: Wtyczka magistralna światłowodowa
OSI	Open Systems Interconnection: Normalizacja dla komunikacji komputerowej

P	
PAA	Odwzorowanie procesu wyjść
PAE	Odwzorowanie procesu wejść
PC	Personal Computer
PCIN	Nazwa oprogramowania do wymiany danych ze sterowaniem
PCMCIA	Personal Computer Memory Card International Association: Normalizacja kart wtykowych pamięci
PCU	PC Unit: PC-Box (jednostka komputerowa)
PG	Przyrząd do programowania
PKE	Identyfikator parametru: Część PKW
PKW	Identyfikator parametru: Wartość (część parametru PPO)
PLC	Programmable Logic Control: Programowalny sterownik logiczny
PN	PROFINET
PNO	Organizacja użytkowników PROFIBUS
PO	POWER ON
POE	Jednostka organizacyjna programu
POS	Pozycja/pozycjonowanie
POSMO A	Positioning Motor Actuator: silnik pozycjonujący
POSMO CA	Positioning Motor Compact AC: Kompletna jednostka napędowa ze zintegrowanym zespołem mocy i sterowania, jak też jednostką pozycjonowania i pamięcią programów; zasilanie prądem przemiennym
POSMO CD	Positioning Motor Compact DC: jak CA, jednak zasilanie prądem stałym
POSMO SI	Positioning Motor Servo Integrated: Silnik pozycjonujący; zasilanie prądem stałym
PPO	Parameter Prozessdaten Objekt ; Cykliczny telegram danych przy przesyłaniu przez PROFIBUS-DP oraz profilu "napędy ze zmienną prędkością obrotową"
PPU	Panel Processing Unit (sprzęt centralny sterowania CNC bazującego na pulpicie, np. SINUMERIK 828D)
PROFIBUS	Process Field Bus: Szeregowa magistrala danych
PRT	Test programu
PSW	Słowo sterujące programu
PTP	Point to Point: Punkt do punktu
PUD	Program Global User Data: Zmienne użytkownika globalne dla programu
PZD	Dane procesowe: Część danych procesu PPO

Q	
QFK	Kompensacja błędu ćwiartki koła

R	
RAM	Random Access Memory: Pamięć do zapisu/odczytu
REF	Funkcja bazowania do punktu odniesienia
REPOS	Funkcja repozycjonowania
RISC	Reduced Instruction Set Computer: Typ procesora o małym zasobie poleceń i szybkim ich wykonywaniu
ROV	Rapid Override: Korekcja posuwu szybkiego
RP	Parametry R, parametry obliczeniowe, predefiniowane zmienne użytkownika
RPA	R-Parameter Active: Obszar pamięci w NCK dla numerów parametrów R
RPY	Roll Pitch Yaw: Rodzaj obrotu układu współrzędnych
RTL	Rapid Traverse Linear Interpolation: Interpolacja liniowa przy posuwie szybkim
RTS	Request To Send: Włączenie wysyłania, sygnał sterujący dla szeregowych interfejsów danych
RTCP	Real Time Control Protocol

S	
SA	Akcja synchroniczna
SBC	Safe Brake Control: Bezpieczne sterowanie hamulcem
SBL	Single Block: Wykonywanie pojedynczymi blokami
SBR	Subroutine: Podprogram (PLC)
SD	Dana ustawcza wzgl. dane ustawcze
SDB	System moduł danych
SEA	Setting Data Active: Oznaczenie (typ pliku) dla danych ustawczych
SERUPRO	Search-Run by Program Test: Wyszukiwanie za pomocą funkcji "Test programu"
SFB	System moduł funkcji
SFC	System Function Call
SGE	Wejście nastawione na bezpieczeństwo
SGA	Wyjście nastawione na bezpieczeństwo
SH	Bezpieczne zatrzymanie
SIM	Single in Line Module
SK	Przycisk programowy
SKP	Skip: Funkcja do pomijania bloku programu obróbki
SLM	Synchroniczny silnik liniowy
SM	Silnik krokowy
SMC	Sensor Module Cabinet Mounted
SME	Sensor Module Externally Mounted

S	
SMI	Sensor Module Integrated
SPF	Sub Program File: Podprogram (NC)
SPS	Sterowanie programowane w pamięci = PLC
SRAM	Pamięć statyczna (buforowana)
SRK	Korekcja promienia ostrza
SRM	Synchroniczny silnik obrotowy
SSFK	Kompensacja błędu skoku śruby pociągowej
SSI	Serial Synchron Interface: Szeregowy interfejs synchroniczny
SSL	Szukanie bloku
STW	Słowo sterowania
SUG	Prędkość obwodowa ściernicy
SW	Oprogramowanie
SYF	System Files: Dane systemowe
SYNACT	Synchronized Action: Akcja synchroniczna

T	
TB	Moduł terminala (SINAMICS)
TCP	Tool Center Point: Wierzchołek narzędzia
TCP/IP	Transport Control Protocol / Internet Protocol
TCU	Thin Client Unit
TEA	Testing Data Active: Oznaczenie dla danych maszynowych
TIA	Totally Integrated Automation
TM	Moduł terminala (SINAMICS)
TO	Tool Offset: Korekcja narzędzia
TOA	Tool Offset Active: Oznaczenie (typ pliku) dla korekcji narzędzia
TRANSMIT	Transform Milling Into Turning: Transformacja współrzędnych dla obróbki frezarskiej na tokarce
TTL	Logika tranzystor-tranzystor (typ interfejsu)
TZ	Cykl technologiczny

U	
UFR	User Frame: Przesunięcie punktu zerowego
UP	Podprogram
USB	Universal Serial Bus
USV	Zasilanie bezprzerwowe

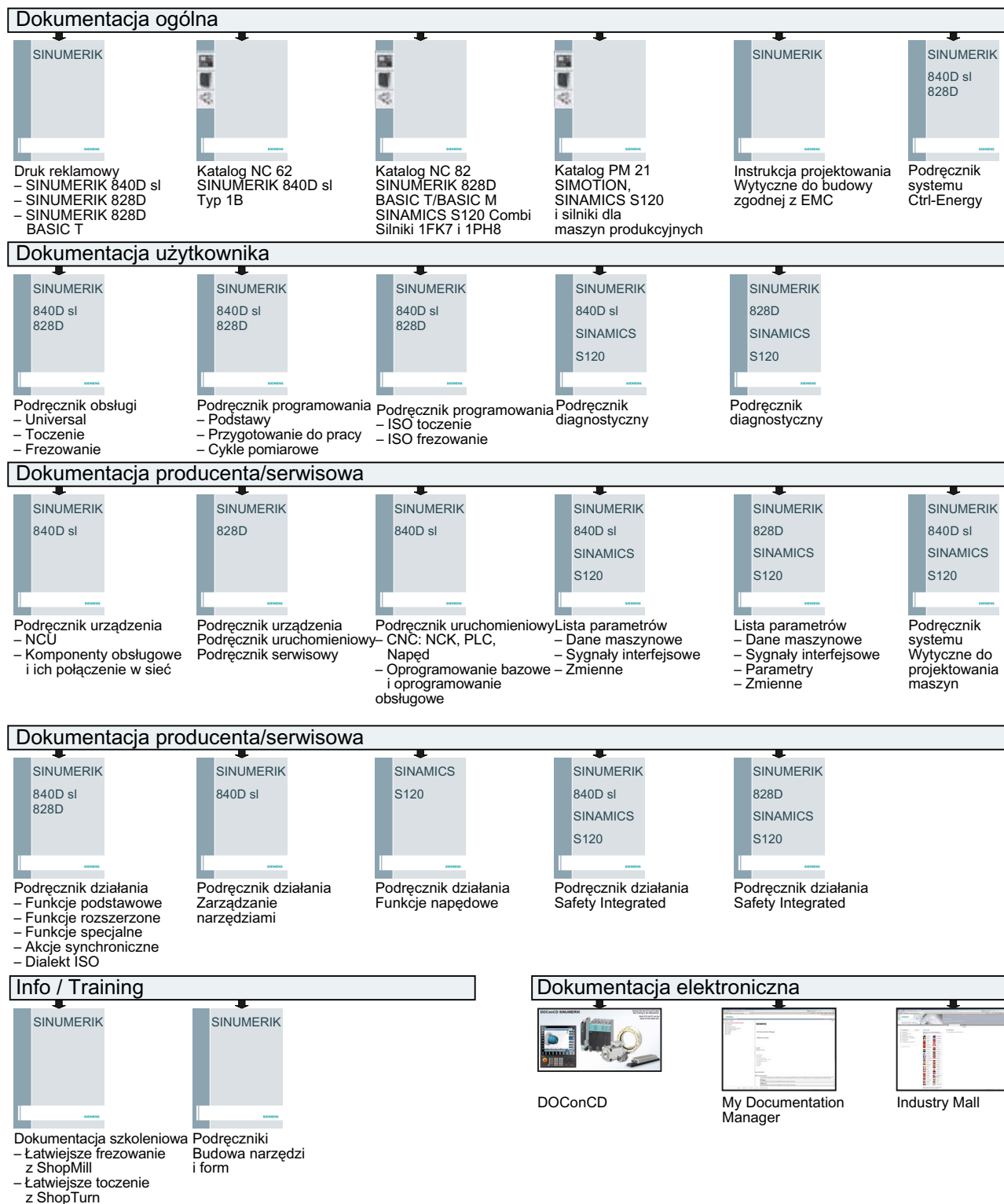
V	
VDI	Wewnętrzny interfejs komunikacyjny między NCK i PLC
VDI	Stowarzyszenie Inżynierów Niemieckich
VDE	Związek Elektrotechników Niemieckich
VI	Voltage Input
VO	Voltage Output
VSA	Napęd posuwu

W	
WAB	Funkcja miękkie dosunięcie i odsunięcie
WKS	Układ współrzędnych obrabianego przedmiotu
WKZ	Narzędzie
WLK	Korekcja długości narzędzia
WOP	Programowanie zorientowane na warsztat
WPD	Work Piece Directory: Katalog obrabianego przedmiotu
WRK	Korekcja promienia narzędzia
WZ	Narzędzie
WZK	Korekcja narzędzia
WZV	Zarządzanie narzędziami
WZW	Wymiana narzędzia

X	
XML	Extensible Markup Language

Z	
ZOA	Zero Offset Active: Oznaczenie przesunięć punktu zerowego
ZSW	Słowo stanu (napędu)

A.2 Przegląd dokumentacji



Glosariusz

Adres

Adres jest oznaczeniem dla określonego argumentu albo zakresu argumentów, np. wejście, wyjście itd.

Adres osi

Patrz → Nazwa osi

Akcje synchroniczne

1. Wyprowadzenie funkcji pomocniczej

Podczas obróbki mogą z programu CNC być wyprowadzane do PLC funkcje technologiczne (→ funkcje pomocnicze). Poprzez te funkcje są np. sterowane urządzenia dodatkowe obrabiarki, jak tuleja wrzecionowa, chwytak, uchwyt tokarski, itd.

2. Szybkie wyprowadzenie funkcji pomocniczej

Dla krytycznych pod względem czasu funkcji łączeniowych można zminimalizować czasy kwitowania dla → funkcji pomocniczych i uniknąć niepotrzebnych punktów zatrzymania w procesie obróbki.

Alarmy

Wszystkie → komunikaty i alarmy są na pulpicie obsługi wyświetlane tekstem jawnym z podaniem daty i czasu zegarowego oraz odpowiedniego symbolu kryterium kasowania. Wyświetlane są osobno alarmy i komunikaty.

1. Alarmy i komunikaty w programie obróbki

Alarmy i komunikaty mogą być bezpośrednio z programu obróbki wyświetlane tekstem jawnym.

2. Alarmy i komunikaty od PLC

Alarmy i komunikaty maszyny mogą być wyświetlane tekstem jawnym z programu PLC. Do tego nie są potrzebne żadne dodatkowe pakiety modułów funkcjonalnych.

Archiwizowanie

Wyprowadzanie plików i/lub katalogów na **zewnętrzne** urządzenie pamięciowe.

Automatyka

Tryb pracy sterowania (praca z ciągiem bloków według DIN): Rodzaj pracy w systemach NC, w których → program obróbki jest wybierany i wykonywany w sposób ciągły.

Bateria buforująca

Bateria buforująca gwarantuje, że → program użytkownika w → CPU jest zapisany w sposób odporny na przerwy zasilania a ustalone obszary danych i znaczniki, czasy i liczniki są utrzymywane.

Bazowy układ współrzędnych

Kartezjański układ współrzędnych, jest przez transformację odwzorowywany na układ współrzędnych maszyny.

W → programie obróbki programista używa nazw osi bazowego układu współrzędnych. Istnieje on, gdy → transformacja nie jest aktywna, równoległe do → układu współrzędnych maszyny. Różnica polega na → nazwie osi.

Blok główny

Blok rozpoczynający się od ":", który zawiera wszystkie dane, aby móc uruchomić przebieg pracy w → programie obróbki.

Blok pomocniczy

Rozpoczynający się od "N" blok zawierający informacje dot. kroku obróbkowego np. podanie pozycji.

Blok programu obróbki

Część → programu obróbki, ograniczona przez Line Feed. Rozróżnia się → bloki główne i → bloki pomocnicze.

Bloki pośrednie

Ruchy postępowe z wybraną → korekcją narzędzia (G_{41}/G_{42}) mogą być przerywane przez ograniczoną liczbę bloków pośrednich (bloki bez ruchów w osiach w płaszczyźnie korekcji), przy czym korekcja narzędzia może być jeszcze prawidłowo obliczana. Dopuszczalną liczbę bloków pośrednich, które sterowanie czyta wyprzedzająco, można ustawić przez parametry systemowe.

Całowy system miar

System miar, który definiuje odległości w calach i ich ułamkach.

CNC

Patrz → NC

COM

Komponent sterowania NC do realizacji i koordynacji komunikacji.

CPU

Central Processor Unit, patrz → sterowanie programowane w pamięci

C-Spline

C-Spline jest najbardziej znanym i najczęściej stosowanym spline. Przejścia w punktach oparcia są krzywą wielomianową o ciągłej pochodnej lub wykazują stałe zakrzywienie. Są stosowane wielomiany 3. stopnia.

Cykle

Chronione podprogramy do wykonywania powtarzających się operacji obróbkowej na → obrabianym przedmiocie.

Cykle standardowe

Dla często powtarzających się zadań obróbkowych są do dyspozycji cykle standardowe:

- dla technologii "wiercenie/frezowanie"
- dla technologii "toczenie"

W oknie obsługowym "Program" są w menu "Obsługa cykli" wyszczególnione dostępne cykle. Po wyborze żądanego cyklu obróbkowego wyświetlane są tekstem jawnym niezbędne parametry dla przyporządkowania wartości.

Dane ustawcze

Dane, które informują sterowanie NC o właściwościach obrabiarki w sposób zdefiniowany przez oprogramowanie systemowe.

Definicja zmiennej

Definicja zmiennej obejmuje ustalenie typu danych i nazwy zmiennej. Przy pomocy nazwy zmiennej można sięgać do jej wartości.

Diagnoza

1. Zakres czynności obsługowych sterowania
2. Sterowanie posiada zarówno program samodiagnozy, jak również pomocnicze możliwości testowe dla serwisu: wyświetlenia statusu, alarmu, serwisowe

Dosunięcie do punktu stałego

Obrabiarki mogą w sposób zdefiniowany dokonywać dosunięcia do punktów stałych jak punkt zmiany narzędzia, punkt załadunku, punkt wymiany palety itd. Współrzędne tych punktów są zapisane w sterowaniu. Sterowanie wykonuje ruch w odnośnych osiach, jeżeli to możliwe, → posuwem szybkim.

DRF

Differential Resolver Function: Funkcja NC, która w połączeniu z elektronicznym kółkiem ręcznym wytwarza przyrostowe przesunięcie punktu zerowego w trybie pracy automatycznej.

Edytor

Edytor umożliwia sporządzenie, zmianę, uzupełnienie, łączenie i wstawianie programów/tekstów/bloków programu.

Edytor tekstów

Patrz → Edytor

Frame

Frame jest to instrukcja obliczeniowa, która zmienia kartezjański układ współrzędnych w inny kartezjański układ współrzędnych. Frame zawiera komponenty → przesunięcie punktu zerowego, → obrót, → skalowanie, → lustrzane odbicie.

Frame programowane

Przy pomocy programowanych → frame mogą dynamicznie, w trakcie wykonywania programu obróbki, być definiowane nowe punkty wyjściowe układu współrzędnych. Rozróżnia się ustalenie bezwzględne na podstawie nowego frame i ustalenie addytywne w odniesieniu do istniejącego punktu wyjściowego.

Funkcje bezpieczeństwa

Sterowanie zawiera stale aktywne nadzory, które rozpoznają zakłócenia w → CNC, sterowaniu adaptacyjnym (→ PLC) i maszynie na tyle wcześnie, że są w dużym stopniu wykluczone uszkodzenia obrabianego przedmiotu, narzędzia albo maszyny. W przypadku zakłócenia przebieg obróbki jest przerywany a napędy są zatrzymywane, przyczyna zakłócenia jest zapisywana w pamięci i jest wyświetlany alarm. Równocześnie PLC otrzymuje informację, że jest aktywny alarm CNC.

Funkcje pomocnicze

Przy pomocy funkcji pomocniczych można w → programach obróbki przekazywać → parametry do → PLC, które wyzwalają reakcje zdefiniowane przez producenta maszyny.

Geometria

Opis → obrabianego przedmiotu w → układzie współrzędnych obrabianego przedmiotu.

Granica zatrzymania dokładnego

Gdy wszystkie osie uczestniczące w tworzeniu konturu osiągną swoją granicę zatrzymania dokładnego, wówczas sterowanie zachowuje się tak, jakby dokładnie osiągnęło punkt docelowy. Następuje przełączenie na następny blok → programu obróbki.

Graniczna prędkość obrotowa

Maksymalna/minimalna prędkość obrotowa (wrzeciona): Przez zadanie w danych maszynowych, → PLC lub → danych ustawczych można ograniczyć maksymalną prędkość obrotową wrzeciona.

Grupa trybów pracy

Technologicznie przynależne do siebie osie i wrzeciona mogą być połączone w grupę trybów pracy (BAG). Osie/wrzeciona jednej BAG mogą być sterowane przez jeden lub wiele → kanałów. Kanałom grupy trybów pracy jest zawsze przyporządkowany taki sam → tryb pracy.

Gwintowanie otworu bez oprawki kompensacyjnej

Przy pomocy tej funkcji można gwintować otwory bez użycia oprawki kompensacyjnej. Dzięki interpolującemu ruchowi wrzeciona, jako osi obrotowej i osi gwintowania gwint jest wykonywany dokładnie na głębokość końcową, np. gwint w otworze nieprzelotowym (warunek: praca wrzeciona jako oś obrotowa).

HIGHSTEP

Podsumowanie możliwości programowania dla → PLC systemu AS300/AS400.

HW-Config

Narzędzie SIMATIC S7 do konfiguracji i parametryzacji komponentów sprzętowych w ramach projektu S7.

Identyfikator

Słowa według DIN 66025 są uzupełniane przez identyfikatory (nazwy) dla zmiennych (zmiennie obliczeniowe, zmiennie systemowe, zmiennie użytkownika), dla podprogramów, słów kluczowych i słów z wieloma literami adresowymi. Znaczenie tych uzupełnień jest pod względem znaczenia równoważne słowom przy budowie bloków. Identyfikatory muszą być jednoznaczne. Tych samych identyfikatorów nie wolno stosować do różnych obiektów.

Inicjalizacja

Ładowanie programu systemowego po Power On.

Interfejs graficzny

Interfejs graficzny (BOF) jest medium, mającym postać ekranu, do wyświetlania dla sterowania CNC. Jest on wyposażony w poziome i pionowe przyciski programowe.

Interpolacja kołowa

Narzędzie → ma wykonać ruch po okręgu między ustalonymi punktami konturu z podanym posuwem a przy tym wykonywać obróbkę.

Interpolacja linii śrubowej

Interpolacja linii śrubowej nadaje się szczególnie do prostego wykonywania gwintów wewnętrznych i zewnętrznych przy pomocy frezów kształtowych i do frezowania rowków smarowych.

Linia śrubowa składa się przy tym z dwóch ruchów:

- Ruch kołowy w płaszczyźnie
- Ruch liniowy prostopadle do tej płaszczyzny

Interpolacja prostoliniowa

Ruch narzędzia następuje po prostej do punktu docelowego i jest przy tym prowadzona obróbka.

Interpolacja spline

Przy pomocy interpolacji spline sterowanie może utworzyć z niewielu zadanych punktów oparcia zadanego konturu gładki przebieg krzywej.

Interpolacja wielomianowa

Przy pomocy interpolacji wielomianowej mogą być wytwarzane najróżniejsze przebiegi krzywych, jak **funkcje prostoliniowe, paraboliczne, potęgowe** (SINUMERIK 840D sl).

Interpolator

Jednostka logiczna → NCK, która po podaniu pozycji docelowych w programie obróbki określa wartości pośrednie dla ruchów będących do wykonania w poszczególnych osiach.

Jednostka TOA

Każdy → zakres TOA może zawierać wiele jednostek TOA. Liczba możliwych jednostek TOA jest ograniczona przez maksymalną liczbę aktywnych → kanałów. Jedna jednostka TOA obejmuje dokładnie jeden moduł danych narzędzi i jeden moduł danych magazynu. Dodatkowo może jeszcze być zawarty moduł danych nośnika narzędzi (opcjonalnie).

Język wysokiego poziomu CNC

Język wysokiego poziomu udostępnia: → zmienne definiowane przez użytkownika, → zmienne systemowe, → technikę makr.

JOG

Tryb pracy sterowania (ustawianie): W trybie pracy JOG można ustawiać maszynę. Poszczególnymi osiami i wrzecionami można przez przyciski kierunkowe wykonywać ruch impulsowy. Dalsze funkcje w trybie pracy JOG to → bazowanie do punktu odniesienia, → repos jak też → Preset (ustawienie wartości rzeczywistej).

Kanał

Kanał charakteryzuje się tym, że niezależnie od innych kanałów może wykonywać → program obróbki. Kanał steruje wyłącznie przyporządkowanymi mu osiami i wrzecionami. Przebiegi programów obróbki w różnych kanałach mogą być koordynowane przez → synchronizację.

Kanał obróbki

Dzięki strukturze kanałowej można przez równoległe wykonywanie ruchów skrócić czasy uboczne, np. ruch manipulatora załadowniczego równocześnie z obróbką. Kanał CNC należy przy tym traktować jako oddzielne sterowanie CNC z dekodowaniem, przetwarzaniem bloków i interpolacją.

Klucz programowania

Znaki i ciągi znaków, które w języku programowania mają dla → programu obróbki ustalone znaczenie.

Koincydencja wyprzedzająca

Zmiana bloku już wtedy, gdy droga po torze zbliżyła się do pozycji końcowej o zadaną wartość delta.

Kompensacja błędu ćwiartki koła

Błędy konturu na przejściach między ćwiartkami, które powstają w wyniku zmieniających się warunków tarcia na prowadnicach, dają się w dużym stopniu wyeliminować przez kompensację błędu ćwiartki. Parametryzowanie kompensacji błędu ćwiartki następuje w drodze testu kształtu kołowego.

Kompensacja błędu skoku śruby pociągowej

Wyrównywanie niedokładności mechanicznych uczestniczącej w posuwie śruby pociągowej tocznej przez sterowanie na podstawie zapisanych wartości pomiarowych odchyień.

Kompensacja interpolacyjna

Przy pomocy kompensacji interpolacyjnej mogą być kompensowane uwarunkowane wykonawczo błędy śruby pociągowej (SSFK) i błędy systemu pomiarowego (MSFK).

Kompensacja luzów

Kompensacja mechanicznych luzów maszyny, np. luzy nawrotu w śrubach pociągowych tocznych. Dla każdej osi można kompensację luzu wprowadzić oddzielnie.

Komunikaty

Wszystkie komunikaty zaprogramowane w programie obróbki i → alarmy rozpoznane przez system są na pulpicie obsługi wyświetlane tekstem jawnym z podaniem daty i czasu zegarowego oraz odpowiedniego symbolu kryterium kasowania. Wyświetlane są osobno alarmy i komunikaty.

Kontur

Obrys → obrabianego przedmiotu

Kontur części gotowej

Kontur części obrobionej na gotowo. Patrz → Półfabrykat.

Kontur obrabianego przedmiotu

Zadany kontur wykonywanego/obrabianego → obrabianego przedmiotu.

Korekcja narzędzia

Uwzględnienie wymiarów narzędzia przy obliczaniu toru.

Korekcja promienia narzędzia

Aby móc bezpośrednio programować pożądaną → kontur obrabianego przedmiotu, sterowanie musi przy uwzględnieniu promienia użytego narzędzia wykonywać ruch po torze równoległym do zaprogramowanego konturu (G41/G42).

Korekcja promienia ostrza

Przy programowaniu konturu zakłada się, że narzędzie jest szpiczaste. Ponieważ jest to w praktyce nie do zrealizowania, promień zakrzywienia użytego narzędzia jest podawany sterowaniu i przez nie uwzględniany. Przy tym punkt środkowy zakrzywienia jest prowadzony wokół konturu z przesunięciem o promień zakrzywienia.

Korektor

Ręczna lub programowana możliwość ingerencji, która pozwala operatorowi na nakładanie programowanych posuwów lub prędkości obrotowych, aby dopasować je do określonego obrabianego przedmiotu lub materiału.

Korektor posuwu

Na zaprogramowaną prędkość jest nakładane aktualne ustawienie prędkości przez → pulpit sterowniczy maszyny albo z → PLC (0-200%). Prędkość posuwu może dodatkowo zostać skorygowana w programie obróbki albo przez programowany współczynnik procentowy (1-200%).

KÜ

Stosunek przełożenia

KV

Współczynnik wzmocnienia obwodu, techniczno-regulacyjna wielkość obwodu regulacji

Look-Ahead

Przy pomocy funkcji **look ahead** uzyskiwane jest, przez "wyprzedzające czytanie" sparametryzowanej liczby bloków zawierających ruch postępowy, optimum prędkości obróbki.

Lustrzane odbicie

Przy lustrzanym odbiciu są zamieniane znaki wartości współrzędnych konturu odnośnie osi. Lustrzanego odbicia można dokonać równocześnie wokół wielu osi.

Masa

Za masę uważa się całość połączonych ze sobą nieaktywnych części środka pracy, które również w przypadku błędu nie mogą uzyskać niebezpiecznego napięcia dotykowego.

MDA

Tryb pracy sterowania: Manual Data Automatic. W rodzaju pracy MDA poszczególne bloki programu albo ich ciągi mogą bez odniesienia do programu głównego albo podprogramu być wprowadzane a następnie natychmiast wykonywane przez naciśnięcie przycisku NC-Start.

Metryczne i calowe podawanie wymiarów:

W programie obróbki wartości pozycji i skoku można programować w calach. Niezależnie od programowanego podawania wymiarów ($G70 / G71$) sterowanie jest ustawiane na system podstawowy.

Metryczny system miar

Znormalizowany system jednostek: dla długości np. mm (milimetr), m (metr).

Moduł

Jako moduły są określane wszystkie pliki, które są potrzebne do sporządzenia i wykonania programu.

Moduł danych

1. Jednostka danych → PLC, do której mogą sięgać programy → HIGHSTEP.
2. Jednostka danych → NC: moduły danych zawierają definicje globalnych danych użytkownika. Dane mogą przy definicji być bezpośrednio inicjalizowane.

Moduł programowy

Moduły programowe zawierają programy główne i podprogramy → programów obróbki.

Nadzór konturu

Jako miara zgodności z konturem jest nadzorowany uchyb nadążania w ramach definiowanego pasma tolerancji. Niedopuszczalnie wysoki uchyb nadążania może wynikać np. z przeciążenia napędu. W takim przypadku następuje alarm i osie są zatrzymywane.

Napęd

Napędem jest ta jednostka CNC, która wykonuje regulację prędkości obrotowej i momentu na podstawie danych NC.

Narzędzie

Działająca w obrabiarce część, która powoduje obróbkę, (np. nóż tokarski, frez, wiertło, promień lasera ...).

Nazwa osi

W celu identyfikacji wszystkie osie kanałowe i → osie maszyny w kanale i w sterowaniu muszą zostać oznaczone jednoznacznie nazwą. Osie → geometryczne oznaczane są jako X, Y, Z. Osie geometryczne które się obracają → osie obrotowe oznaczone są jako A, B, C.

NC

Numerical Control: Sterowanie obejmuje wszystkie komponenty sterowania obrabiarki: → NCK, → PLC, HMI, → COM.

Wskazówka

Dla sterowań SINUMERIK bardziej prawidłowe byłoby określenie sterowanie CNC: Computerized Numerical Control.

NCK

Numerical Control Kernel: Komponent sterowania NC, który wykonuje → programy obróbki i w istotnej części koordynuje przebiegi ruchów na obrabiarce.

NRK

Numeric Robotic Kernel (system operacyjny → NCK)

NURBS

Wewnętrzne w sterowaniu sterowanie prędkością i interpolacja ruchu po torze jest przeprowadzana na bazie NURBS (**N**on **U**niform **R**ational **B**-**S**plines). W ten sposób w sterowaniu dla wszystkich interpolacji do dyspozycji jest jednolita metoda.

Obrabiany przedmiot

Cześć wykonywana / obrabiana przez obrabiarkę.

Obróbka skosów

Obróbka wiertarska i frezarska powierzchni obrabianych przedmiotów, które nie leżą w płaszczyznach współrzędnych maszyny, może być komfortowo prowadzona przy wsparciu przez funkcję "obróbka skosów".

Obrót

Komponent → frame, który definiuje obrót układu współrzędnych o określony kąt.

OEM

Dla producentów maszyn, którzy chcą sporządzać swoje własne interfejsy graficzne albo umieszczać w sterowaniu funkcje specyficzne dla technologii, są przewidziane przestrzenie dla indywidualnych rozwiązań (aplikacje OEM).

Ograniczenie obszaru pracy

Przy pomocy ograniczenia obszaru pracy można dodatkowo do wyłączników krańcowych ograniczyć zakres ruchu osi. Dla każdej osi jedna para wartości służy do opisu chronionej przestrzeni roboczej.

Osie

Osie CNC są odpowiednio do zakresu swojego funkcjonowania podzielone na:

- Osie: osie interpolujące biorące udział w tworzeniu konturu
- Osie pomocnicze: nie interpolujące, osie dosuwu i pozycjonowania poruszające się z specyficznym dla nich posuwem. Osie pomocnicze nie biorą udziału we właściwej obróbce, np. urządzenie podające narzędzia, magazyn narzędzi.

Osie maszyny

Osie fizycznie istniejące w obrabiarce.

Osie synchroniczne

Osie synchroniczne potrzebują dla przebycia swojej drogi takiego samego czasu co osie geometryczne dla swojego ruchu po torze.

Oś bazowa

Oś, której wartość zadana lub rzeczywista jest brana do obliczenia wartości kompensacji.

Oś C

Oś, wokół której następuje sterowany ruch obrotowy i pozycjonowanie przy pomocy wrzeciona obrabianego przedmiotu.

Oś geometryczna

Osie geometryczne tworzą dwu- lub trójwymiarowy układ współrzędnych obrabianego przedmiotu w którym to → w programach obróbki programowana jest geometria obrabianego przedmiotu.

Oś kompensacji

Oś, której wartość zadana albo rzeczywista jest modyfikowana przez wartość kompensacji.

Oś liniowa

Oś liniowa jest to oś, która w przeciwieństwie do osi obrotowej wykonuje ruch po prostej.

Oś obrotowa

Osie obrotowe powodują obrót obrabianego przedmiotu albo narzędzia do danego położenia kąowego.

Oś pozycjonowania

Oś, która wykonuje ruch pomocniczy na obrabiarce. (np. magazyn narzędzi, transport palet). Osie pozycjonowania są to osie, które nie interpolują z → osiami uczestniczącymi w tworzeniu konturu.

Oś uczestnicząca w tworzeniu konturu

Takimi osiami są wszystkie osie → kanału uczestniczące w obróbce, które są przez → interpolator tak prowadzone, że równocześnie rozpoczynają ruch, przyspieszają, zatrzymują się i osiągają punkt końcowy.

Oś wiodąca

Oś wiodąca jest to → oś gantry, która z punktu widzenia operatora i programisty istnieje a przez to można odpowiednio na nią wpływać jak na normalną oś NC.

Oś zaokrągleniowa

Osie zaokrągleniowe powodują obrót obrabianego przedmiotu albo narzędzia do położenia kąowego odpowiadającego rastrowi podziałowemu. Po osiągnięciu rastra oś zaokrągleniowa jest "w pozycji".

Oś zsynchronizowana

Oś zsynchronizowana jest → osią gantry, której pozycja zadana jest stale wyprowadzana od ruchu postępowego → osi wiodącej a przez to wykonuje ruch synchroniczny. Z punktu widzenia operatora i programisty oś zsynchronizowana "nie istnieje".

Pamięć korekcji

Obszar danych w sterowaniu, w którym są zapisane dane korekcyjne narzędzi.

Pamięć ładowania

Pamięć ładowania jest w przypadku CPU 314 sterowania → SPS równa → pamięci roboczej.

Pamięć programów PLC

SINUMERIK 840D sl: W pamięci użytkownika PLC program użytkownika PLC i dane użytkownika są zapisywane razem z programem podstawowym PLC.

Pamięć robocza

Pamięć robocza jest pamięcią RAM w -> CPU, w której procesor podczas wykonywania programu sięga do programu użytkownika.

Pamięć systemowa

Pamięć systemowa jest pamięcią w CPU, w której są zapisywane następujące dane:

- dane, których potrzebuje system operacyjny
- argumenty, czasy, liczniki, znaczniki

Pamięć użytkownika

Wszystkie programy i dane jak programy obróbki, podprogramy, komentarze, korekcje narzędzi, przesunięcia punktu zerowego / frame jak też dane użytkownika dla kanału i programu mogą być zapisywane we wspólnej pamięci użytkownika w CNC.

Parametry R

Parametr obliczeniowy, może być przez programistę → programu obróbki ustawiany i odpytywany w programie dla dowolnych celów.

Peryferyjny zespół konstrukcyjny

Peryferyjne zespoły konstrukcyjne stanowią połączenie między CPU i procesem.

Peryferyjnymi zespołami konstrukcyjnymi są:

- → Cyfrowe zespoły wprowadzania/wyprowadzania
- → Analogowe zespoły wprowadzania/wyprowadzania
- → Symulatorowe zespoły konstrukcyjne

PLC

Programmable Logic Control: → Programowalny sterownik logiczny. Komponenty → NC: Sterowanie adaptacyjne do realizacji logiki kontrolnej obrabiarki.

Podprogram

Określenie "Podprogram" pochodzi z okresu, gdy programy obróbki były na stałe podzielone na → programy główne i podprogramy. Tego podziału już nie ma w dzisiejszym języku SINUMERIK NC. Zasadniczo każdy program obróbki lub każdy → cykl można wywołać w ramach innego programu obróbki, jako podprogram. Jest on wówczas wykonywany na następnym → poziomie programu (x+1) (poziom podprogramu (x+1)).

Podprogram asynchroniczny

Program obróbki, który może zostać uruchomiony asynchronicznie (niezależnie) do aktualnego stanu programu przez sygnał przerwania (np. sygnał "szybkie wejście NC").

Posuw po torze

Posuw po torze działa na → osie uczestniczące w tworzeniu konturu. Stanowi on sumę geometryczną posuwów uczestniczących → osi geometrycznych.

Posuw szybki

Najszybszy ruch w osi. Jest on np. stosowany, gdy narzędzie jest dosuwane z położenia spoczynkowego do → konturu obrabianego przedmiotu albo odsuwane od konturu. Prędkość posuwu szybkiego jest ustawiana specyficznie dla maszyny przez daną maszynową.

Posuw zależny od czasu

Zamiast prędkości posuwu można dla osi zaprogramować czas, którego wymaga posuw po torze w ramach bloku (G93).

Poziom programu

Program obróbki uruchomiony w kanale jest wykonywany jako → program główny na poziomie 0 (poziom programu głównego). Każdy program obróbki wywołany w programie głównym jest wykonywany jako → podprogram na własnym poziomie 1 ... n.

Półfabrykat

Część, od której jest rozpoczynana obróbka.

Praca z płynnym przechodzeniem

Celem płynnego przechodzenia jest uniknięcie większego hamowania → osi uczestniczących w tworzeniu konturu na granicach bloków programu obróbki i przełączanie z możliwie taką samą prędkością ruchu po torze na następny blok.

Prędkość ruchu po torze

Maksymalnie możliwa do zaprogramowania prędkość ruchu po torze jest zależna od dokładności wprowadzania. Przy rozdzielczości np. 0,1 mm maksymalna możliwa do zaprogramowania prędkość ruchu po torze wynosi 1000 m/min.

Procedura przerwania

Procedury przerwania są specjalnymi → podprogramami, które mogą być uruchamiane przez zdarzenia (sygnały zewnętrzne) z procesu obróbki. Wykonywany blok programu obróbki jest przerywany, pozycja przerwania w osiach jest automatycznie zapisywana w pamięci.

Program główny

Określenie "Program główny" pochodzi z okresu, gdy programy obróbki były na stałe podzielone na programy główne i podprogramy. Tego podziału już nie ma w dzisiejszym języku SINUMERIK NC. Zasadniczo każdy program obróbki można wybrać w kanale i uruchomić. Jest on wówczas wykonywany na → poziomie programu 0 (poziom programu głównego). W programie głównym mogą być jako podprogramy wywoływane dalsze programy obróbki lub cykle

Program obróbki

Ciąg poleceń pod adresem sterowania NC, które w sumie powodują wykonanie określonego → obrabianego przedmiotu. Również podjęcie określonej obróbki na danym → półfabrykacie.

Program przesyłania danych PCIN

PCIN jest programem pomocniczym do wysyłania i odbierania danych użytkownika CNC poprzez interfejs szeregowy, jak np. programy obróbki, korekcje narzędzi, itd. Program PCIN pracuje pod MS-DOS na standardowym przemysłowym PC.

Program użytkownika

Programy użytkownika dla systemów automatyzacyjnych S7-300 są sporządzane przy pomocy języka programowania STEP-7. Program użytkownika ma budowę modułową i składa się z poszczególnych modułów.

Podstawowymi typami modułów są:

- Moduły kodowe
Moduły te zawierają polecenia STEP 7.
- Moduły danych
Moduły te zawierają stałe i zmienne dla programu STEP 7.

Programowane ograniczenie pola roboczego

Ograniczenie przestrzeni ruchów narzędzia do przestrzeni zdefiniowanej przez programowane ograniczenia.

Programowanie PLC

PLC jest programowane przy pomocy oprogramowania **STEP 7**. Oprogramowanie do programowania STEP 7 bazuje na standardowym systemie operacyjnym **WINDOWS** i zawiera innowacyjnie rozwinięte funkcje programowania STEP 5.

Przełącznik z kluczykiem

Przełącznik z kluczykiem na pulpicie sterowniczym maszyny posiada 4 położenia, które mają funkcje ustalone w systemie operacyjnym sterowania. Do przełącznika z kluczykiem należą trzy kluczyki o różnych kolorach, które można wyjąć w podanych położeniach

Przeźrenie ochronna

Przeźrenie trójwymiarowa w ramach → przestrzeni roboczej, w którą nie może wejść wierzchołek narzędzia.

Przeźrenie robocza

Przeźrenie trójwymiarowa, w której może poruszać się wierzchołek narzędzia, ze względu na konstrukcję obrabiarki. Patrz → Przeźrenie robocza

Przesunięcie punktu zerowego

Zadanie nowego punktu odniesienia dla układu współrzędnych poprzez odniesienie do istniejącego punktu zerowego i → frame.

1. Ustawiane

Dla każdej osi CNC jest do dyspozycji projektowana liczba ustawianych przesunięć punktu zerowego. Przesunięcia wybieralne poprzez funkcje G działają alternatywnie.

2. Zewnętrzne

Na wszystkie przesunięcia, które ustalają położenie punktu zerowego obrabianego przedmiotu, można dodatkowo nałożyć przesunięcie zewnętrzne przy pomocy kółka ręcznego (przesunięcie DRF) albo z PLC.

3. Programowane

Przy pomocy instrukcji `TRANS` można programować przesunięcia punktu zerowego dla wszystkich osi uczestniczących w tworzeniu konturu i osi pozycjonowania.

Przycisk programowy

Przycisk, którego napis jest reprezentowany przez pole na ekranie, które dynamicznie dopasowuje się do aktualnej sytuacji obsługowej. Dowolnie wykorzystywane przyciski funkcyjne (przyciski programowe) są przyporządkowywane do funkcji definiowanych w oprogramowaniu.

Przyrost

Podanie długości ruchu poprzez liczbę przyrostową (przyrost). Liczba przyrostowa może być zapisana jako → dana ustawcza lub wybrana przez odpowiednio opisane przyciski 10, 100, 1000, 10000.

Przyśpieszenie z ograniczeniem szarpnięcia drugiego stopnia

W celu uzyskania optymalnego zachowania się przy przyśpieszeniu w maszynie przy jednoczesnym oszczędzaniu mechaniki można w programie obróbki przełączać między przyśpieszeniem skokowym i stałym (bez szarpnięcia).

Pulpit sterowniczy maszyny

Pulpit obsługi obrabiarki z elementami obsługi jak przyciski, przełączniki obrotowe itd. i prostymi elementami sygnalizacyjnymi jak diody. Służy on do bezpośredniego wpływania na maszynę poprzez PLC.

Punkt odniesienia

Punkt obrabiarki, do którego odnosi się system pomiarowy → osi maszyny.

Punkt stały maszyny

Punkt jednoznacznie definiowany przez obrabiarkę, np. punkt odniesienia maszyny.

Punkt zerowy maszyny

Stały punkt obrabiarki, do którego można sprowadzić wszystkie (wyprowadzone) systemy miar.

Punkt zerowy obrabianego przedmiotu

Punkt zerowy obrabianego przedmiotu tworzy punkt wyjściowy dla → układu współrzędnych obrabianego przedmiotu. Jest on definiowany przez odstęp od → punktu zerowego maszyny.

Sieć

Sieć jest połączeniem wielu S7-300 i dalszych urządzeń końcowych, np. PG, poprzez → kabel łączący. Poprzez sieć następuje wymiana danych między przyłączonymi urządzeniami.

Skalowanie

Komponent → frame, który powoduje specyficzne dla osi zmiany skali.

Słowa kluczowe

Słowa o ustalonej pisowni, które w języku programowania mają dla → programów obróbki zdefiniowane znaczenie.

Słowo danych

Jednostka danych o wielkości dwóch bajtów w ramach → modułu danych.

Softwareowe wyłączniki krańcowe

Softwareowe wyłączniki krańcowe ograniczają zakres ruchu w osi i zapobiegają najechaniu sań na sprzętowy wyłącznik krańcowy. Na każdą oś można zadać 2 pary wartości, które można oddzielnie uaktywniać przez → PLC.

Sterowanie prędkością

Aby przy ruchach postępowych móc uzyskać akceptowalną prędkość ruchu o bardzo małe wielkości, można ustawić ewaluację wyprzedzającą o wiele bloków (→ Look Ahead).

Sterowanie programowane w pamięci

Sterowania programowane w pamięci (SPS) są to sterowania elektroniczne, których działanie jest zapisane jako program w urządzeniu sterującym. Budowa i okablowanie urządzenia nie zależą więc od jego funkcji. Sterowanie programowane w pamięci ma strukturę komputera; składa się ono z CPU (centralny zespół konstrukcyjny) z pamięcią, zespołów wejścia/wyjścia i wewnętrznego systemu przewodów magistralnych. Peryferia i język programowania są dostosowane do wymogów techniki sterowania.

Sterowanie wyprzedzające, dynamiczne

Niedokładności → konturu, uwarunkowane uchybem nadążania, dają się prawie całkowicie wyeliminować przez dynamiczne, zależne od przyspieszenia sterowanie wyprzedzające. Przez to również przy wysokich → prędkościach po torze uzyskuje się znakomitą dokładność obróbki. Sterowanie wyprzedzające można wybrać i cofnąć specyficznym dla osi przez → program obróbki.

Synchronizacja

Instrukcje w → programach obróbki służące do koordynacji przebiegów w różnych → kanałach w określonych miejscach obróbki.

Szukanie bloku

W celu testowania programów obróbki albo po przerwaniu obróbki można poprzez funkcję "szukanie bloku" wybrać dowolne miejsce w programie obróbki, od którego obróbkę można uruchomić albo kontynuować.

Szybkie cofnięcie od konturu

Przy przybyciu przerwania można przez program obróbkowy CNC spowodować ruch, który umożliwi szybkie cofnięcie narzędzia od właśnie obrabianego konturu. Dodatkowo można zaprogramować kąt wycofania i wielkość drogi. Po szybkim cofnięciu można dodatkowo wykonać procedurę przerwania.

Szybkie cyfrowe wejścia/wyjścia

Poprzez wejścia cyfrowe mogą np. być uruchamiane szybkie procedury programowe CNC (procedury przerwania). Przez wyjścia cyfrowe CNC mogą być uruchamiane szybkie, sterowane przez program funkcje łączeniowe.

Szybkość transmisji

Szybkość przesyłania danych (bitów/s).

Tablica kompensacji

Tablica punktów oparcia. Dla wybranych pozycji osi bazowej wyznacza ona wartości kompensacji w osi kompensacji.

Technika makr

Połączenie pewnej liczby instrukcji pod jednym identyfikatorem. Identyfikator reprezentuje w programie zbiór połączonych instrukcji.

Transformacja

Addytywne lub absolutne przesunięcie punktu zerowego osi.

Tryb pracy

Koncepcja przebiegu pracy sterowania SINUMERIK. Są zdefiniowane tryby pracy → Jog, → MDA, → automatyka.

Układ współrzędnych

Patrz → układ współrzędnych maszyny, → układ współrzędnych obrabianego przedmiotu

Układ współrzędnych maszyny

Układ współrzędnych, który jest odniesiony do osi obrabiarki.

Układ współrzędnych obrabianego przedmiotu

Układ współrzędnych obrabianego przedmiotu ma swój punkt początkowy w → punkcie zerowym obrabianego przedmiotu. Przy programowaniu w układzie współrzędnych obrabianego przedmiotu wymiary i kierunki odnoszą się do tego układu.

V.24

Interfejs szeregowy do wprowadzania/wyprowadzania danych. Przez ten interfejs mogą być ładowane i zapisywane programy obróbki, jak też dane producenta i użytkownika.

Wartość kompensacji

Różnica między zmierzoną przez przetwornik pomiarowy pozycją w osi a pozycją żadaną, zaprogramowaną.

WinSCP

WinSCP jest powszechnie dostępnym programem Open Source dla Windows do przesyłania plików.

Współrzędne biegunowe

Układ współrzędnych, który ustala położenie punktu na płaszczyźnie przez jego odległość od punktu zerowego i kąt, który tworzą promień wodzący i ustaloną oś.

Wymiar absolutny

Podanie celu ruchu w osi przez wymiar, który odnosi się do punktu zerowego aktualnie obowiązującego układu współrzędnych. Patrz → Wymiar przyrostowy.

Wymiar przyrostowy

Również wymiar przyrostowy: Podanie celu ruchu w osi poprzez będący do przebycia odcinek drogi i kierunek w odniesieniu do osiągniętego już punktu. Patrz → Wymiar absolutny

Zakres ruchu

Maksymalny dopuszczalny zakres ruchu w przypadku osi liniowych wynosi ± 9 dekad. Wartość bezwzględna jest zależna od wybranej dokładności wprowadzania i regulacji położenia oraz systemu jednostek (calowy albo metryczny).

Zakres TOA

Zakres TOA obejmuje wszystkie dane narzędzi i magazynów. Standardowo zakres wzgl. zasięg danych pokrywa się z zakresem → "Kanał". Przez dane maszynowe można jednak ustalić, że wiele kanałów będzie dzielić się jedną → jednostką TOA, tak że dla tych kanałów będą wówczas dostępne wspólne dane zarządzania narzędziami.

Zakrzywienie

Zakrzywienie k konturu jest odwrotnością promienia r przylegającego okręgu w punkcie konturu ($k = 1/r$).

Zarządzanie programami obróbki

Zarządzanie programami obróbki może być zorganizowane według → obrabianych przedmiotów. Wielkość pamięci użytkownika określa liczbę zarządzanych programów i danych. Każdy plik (programy i dane) można wyposażyć w nazwę o maksymalnie 24 znakach alfanumerycznych.

Zatrzymanie dokładne

Przy zaprogramowanej instrukcji zatrzymania dokładnego dosunięcie do pozycji podanej w bloku jest dokonywane dokładnie i ew. bardzo powoli. W celu redukcji czasu zbliżania są zdefiniowane dla posuwu szybkiego i posuwu definiowane → granice zatrzymania dokładnego.

Zewnętrzne przesunięcie punktu zerowego

Przesunięcie punktu zerowego zadane z → PLC.

Zmienna systemowa

Zmienna istniejąca bez udziału programisty programującego → program obróbki. Jest ona zdefiniowana przez typ danych i nazwę, która rozpoczyna się od znaku \$. Patrz → Zmienne definiowane przez użytkownika.

Zmienne definiowane przez użytkownika

Użytkownicy mogą dla dowolnego użycia w → programie obróbki albo module danych (globalne dane użytkownika) uzgodnić zmienne definiowane przez użytkownika. Definicja zawiera podanie typu danych i nazwę zmiennej. Patrz → Zmienne systemowe.

Zorientowane wycofanie wrzeciona

RETOOL: Przy przerwaniu pracy (np. przy pęknięciu narzędzia) narzędzie może poprzez polecenie programowe zostać cofnięte o zdefiniowaną drogę przy zadanej orientacji.

Zorientowane zatrzymanie wrzeciona

Zatrzymanie wrzeciona obrabianego przedmiotu w zadnym położeniu kątowym, np. aby w określonym miejscu przeprowadzić dodatkową obróbkę.

Zresetowanie całkowite

Przy zresetowaniu całkowitym są kasowane następujące pamięci → CPU

- → Pamięć robocza
- Obszar zapisu / odczytu → pamięci ładowania
- → Pamięć systemowa
- → Pamięć backup

Indeks

"

"Obróbka szybkościowa" – CYCLE832
programowanie zewnętrzne, 727

\$

\$AA_ATOL, 498
\$AA_COUP_ACT
Przy nadążaniu, 507
Przy osiowym sprzężeniu wartości wiodącej, 531
przy sterowaniu stycznym, 457
\$AA_ESR_ENABLE, 625
\$AA_LEAD_SP, 531
\$AA_LEAD_SV, 531
\$SAC_ACT_PROG_NET_TIME, 610
\$SAC_ACTUAL_PARTS, 613
\$SAC_AXCTSWA, 599
\$SAC_AXCTSWE, 599
\$SAC_CTOL,
\$SAC_CUT_INV, 449
\$SAC_CUTMOD, 449
\$SAC_CUTMOD_ANG, 448
\$SAC_CUTTING_TIME, 609
\$SAC_CYCLE_TIME, 609
\$SAC_DELAYFST, 480
\$SAC_ESR_TRIGGER,
\$SAC_OLD_PROG_NET_TIME, 610
\$SAC_OLD_PROG_NET_TIME_COUNT, 610
\$SAC_OPERATING_TIME, 609
\$SAC_PROG_NET_TIME_TRIGGER, 610
\$SAC_REPOS_PATH_MODE, 489
\$SAC_REQUIRED_PARTS, 613
\$SAC_SMAXVELO, 494
\$SAC_SMAXVELO_INFO, 494
\$SAC_SPECIAL_PARTS, 613
\$SAC_STOLF, 501
\$SAC_TOTAL_PARTS, 613
\$AN_AXCTAS, 599
\$AN_AXCTSWA, 599
\$AN_LANGUAGE_ON_HMI, 821
\$AN_POWERON_TIME, 609
\$AN_SETUP_TIME, 609
\$P_ACTBFRAME, 301
\$P_AD, 449

\$P_BFRAME, 301
\$P_CHBFRAME, 301
\$P_CHBFRMASK, 302
\$P_CTOL, 499
\$P_CUT_INV, 449
\$P_CUTMOD, 449
\$P_CUTMOD_ANG, 448
\$P_DELAYFST, 480
\$P_IFRAME, 302
\$P_NCBFRAME, 300
\$P_NCBFRMASK, 302
\$P_OTOL, 499
\$P_PFRAME, 302
\$P_SIM, 271
\$P_STOLF, 501
\$P_SUBPAR, 164
\$PA_ATOL, 499
\$SA_LEAD_TYPE, 530
\$SC_CONTPREC, 470
\$SC_MINFEED, 471
\$SC_PA_ACTIV_IMMED, 227
\$SN_PA_ACTIV_IMMED, 227
\$TC_CARR1...14, 434
\$TC_CARR18...23, 434
\$TC_CARR18[m], 438
\$TC_DP1 ... 25, 391
\$TC_ECPxy, 395
\$TC_SCPxy, 395
\$TC_TPG1 ... 9, 590

*

* (funkcja obliczeniowa), 70

/

/ (funkcja obliczeniowa), 70

+

+ (funkcja obliczeniowa), 70

<

< (operator porównania), 73

<< (operator powiązania), 84

<= (operator porównania), 73
<> (operator porównania), 73

=

== (operator porównania), 73

>

> (operator porównania), 73
>= (operator porównania), 73

A

ABS, 70
ACCLIMA, 465
ACOS, 70
ACTBLOCNO, 177
ACTFRAME, 279
ADISPOSA, 273
Adresy OEM, 271
Alarmy
 Ustawienie w programie NC, 622
Alarmy cykli, 623
ALF
 dla szybkiego cofnięcia od konturu, 132
AND, 73
APR, 40
APRB, 40
APRP, 40
APW, 40
APWB, 40
APWP, 40
AS, 210
ASIN, 70
A-Spline, 240
ASPLINE, 234
ASUP, 124
Asynchroniczny ruch wahliwy, 563
ATAN2, 70
ATOL, 496
Atrybuty pozycji
 Programowanie pośrednie, 66
Automatyczny podział drogi, 582
Automatyczny wskaźnik przerwania, 482
AV, 539
AX, 591
AXCTSWE, 598
AXCTSWEC, 598
AXCTSWED, 598
AXIS, 24

AXNAME, 82
AXSTRING, 591
AXTOCHAN, 140
AXTOINT, 54
AXTOSPI, 591

B

B_AND, 73
B_NOT, 73
B_OR, 73
B_XOR, 73
BAUTO, 234
BFRAME, 279
BLOCK, 198
Blok zatrzymania, 481
BLSYNC, 126
BNAT, 234
BOOL, 24
BOUND, 77
BRISK, 463
BRISKA, 463
B-Spline, 241
BSPLINE, 234
BTAN, 234

C

CAC, 233
CACN, 233
CACP, 233
CALCDAT, 646
CALL, 197
CALLPATH, 201
CASE, 102
CDC, 233
CFINE, 290
CHAN, 24
CHANDATA, 217
CHAR, 24
CHKDNO, 430
CIC, 233
Cięcie
 aktywowanie/dezaktywowanie, 577
 Automatyczny podział drogi, 582
CLEARARM, 118
CLRINT, 128
COARSE, 539
COARSEA, 273
COLLPAIR, 385
COMCAD, 245

COMPCURV, 245
COMPLETE, 217
COMPOF, 245
COMPON, 245
CONTDCON, 640
CONTPRON, 634
COS, 70
COUPDEF, 539
COUPDEL, 539
COUPOF, 539
COUPOFS, 539
COUPON, 539
COUPONC, 539
COUPRES, 539
CP, 369
CPBC, 551
CPDEF, 550
CPDEL, 550
CPFMOF, 553
CPFMON, 553
CPFMSON, 552
CPFPOS, 553
CPFRS, 551
CPLA, 550
CPLCTID, 551
CPLDEF, 550
CPLDEL, 550
CPLDEN, 551
CPLINSC, 555
CPLINTR, 555
CPLNUM, 551
CPLOF, 551
CPLON, 551
CPLOUTSC, 555
CPLOUTTR, 555
CPLPOS, 551
CPLSETVAL, 551
CPMALARM, 556
CPMBRAKE, 556
CPMPRT, 555
CPMRESET, 553
CPMSTART, 554
CPMVDI, 556
CPOF, 551
CPON, 551
CPRECOF, 470
CPRECON, 470
CPROT, 224
CPROTDEF, 221
CPSETTYPE, 556
CPSYNCOV, 555
CPSYNFIP, 555
CPSYNFIP2, 555
CPSYNFIV, 555
C-Spline, 242
CSPLINE, 234
CTAB, 520
CTABDEF, 509
CTABDEL, 515
CTABEND, 509
CTABEXISTS, 515
CTABFNO, 525
CTABFPOL, 525
CTABFSEG, 525
CTABID, 518
CTABINV, 520
CTABISLOCK, 518
CTABLOCK, 517
CTABMENTYP, 518
CTABMPOL, 525
CTABMSEG, 525
CTABNO, 525
CTABNOMEM, 525
CTABPERIOD, 518
CTABPOL, 525
CTABPOLID, 525
CTABSEG, 525
CTABSEGID, 525
CTABSEV, 520
CTABSSV, 520
CTABTEP, 520
CTABTEV, 520
CTABTMAX, 520
CTABTMIN, 520
CTABTSP, 520
CTABTSV, 520
CTABUNLOCK, 517
CTOL, 496
CTRANS, 290
CUT3DC, 410
CUT3DCC, 420
CUT3DCCD, 420
CUT3DF, 410
CUT3DFF, 410
CUT3DFS, 410
CUTMOD, 445
CYCLE60 - Grawerowanie
 programowanie zewnętrzne, 693
CYCLE61 - Frezowanie płaszczyzny
 programowanie zewnętrzne, 669
CYCLE62 - Wywołanie konturu
 programowanie zewnętrzne, 695

CYCLE63 - Frezowanie konturu kieszeni
 programowanie zewnętrzne, 700
 CYCLE70 - Frezowanie gwintu
 programowanie zewnętrzne, 691
 CYCLE72 - Frezowanie konturu
 programowanie zewnętrzne, 696
 CYCLE76 - Czop prostokątny
 programowanie zewnętrzne, 676
 CYCLE77 - Czop kołowy
 programowanie zewnętrzne, 678
 CYCLE79 - Wielobok
 programowanie zewnętrzne, 680
 CYCLE801 - szablon pozycji Siatka/Ramka
 programowanie zewnętrzne, 667
 CYCLE802
 programowanie zewnętrzne, 665
 CYCLE81 - nawiercanie
 programowanie zewnętrzne, 651
 CYCLE82
 programowanie zewnętrzne, 652
 CYCLE83 - Wiercenie głębokich otworów
 programowanie zewnętrzne, 654
 CYCLE832 - Ustawienia dla obróbki szybkościowej
 programowanie zewnętrzne, 727
 CYCLE84 - Gwintowanie otworu bez oprawki
 kompensacyjnej
 programowanie zewnętrzne, 657
 CYCLE840 - Gwintowanie otworu z oprawką
 kompensacyjną
 programowanie zewnętrzne, 660
 CYCLE85 - Rozwiercanie
 programowanie zewnętrzne, 653
 CYCLE86 - Wytaczanie
 programowanie zewnętrzne, 656
 CYCLE92 - Przecięcie
 programowanie zewnętrzne, 717
 CYCLE930 - Rowek
 programowanie zewnętrzne, 705
 CYCLE940 - Podcięcie
 programowanie zewnętrzne, 708
 CYCLE98 - Łańcuch gwintów
 programowanie zewnętrzne, 713
 CYCLE99 - Toczenie gwintu
 programowanie zewnętrzne, 710
 Czas obróbki, 609
 Czas przebiegu
 Zachowanie się struktur kontrolnych, 111
 Część programu
 Powtórzenie, 104
 Czop kołowy - CYCLE77
 programowanie zewnętrzne, 678
 Czop prostokątny - CYCLE76

programowanie zewnętrzne, 676

D

Dane okręgu
 Obliczenie, 646
 DEF, 24
 DEFAULT, 102
 DEFINE ... AS, 210
 DELAYFSTOF, 475
 DELAYFSTON, 475
 DELDL, 396
 DELETE, 148
 DELOBJ, 381
 DISABLE, 127
 DISPLOF, 177
 DISPLON, 177
 DISPR, 483
 DIV, 70
 DL, 394
 DO, 561
 Dokładność konturu
 programowalna, 470
 Dostępność
 Zależne od systemu, 5
 Dosuw
 Oś, 569
 Ruch, 574
 Dowolne pozycje - CYCLE802
 programowanie zewnętrzne, 665
 DRIVE, 463
 DRIVEA, 463
 DV, 539
 DYNFINISH, 467
 DYNNORM, 467
 DYNPOS, 467
 DYNROUGH, 467
 DYNSEMIFIN, 467

E

EAUTO, 234
 EG
 Przekładnia elektroniczna, 531
 EGDEF, 532
 EGDEL, 538
 EGOFC, 537
 EGOF, 537
 EGON, 533
 EGONSYN, 533
 EGONSYNE, 533

Element konturu
 Odsunięcie, 645
 ELSE, 112
 ENABLE, 127
 ENAT, 234
 ENDFOR, 114
 ENDIF, 112
 ENDLABEL, 104
 ENDLOOP, 113
 ENDWHILE, 116
 ESR, 624
 ESRR, 631
 ESRS, 630
 ETAN, 234
 Etykieta, 104
 EVERY, 561
 EXECSTRING, 69
 EXECTAB, 645
 EXECUTE, 648
 EXP, 70
 EXTCALL, 206
 EXTCLOSE, 614
 EXTERN, 191
 EXTOPEN, 614

F

F10, 221
 FALSE, 24
 FCUB, 458
 FENDNORM, 272
 FFWOF, 469
 FFWON, 469
 FIFOCTRL, 472
 FILEDATE, 155
 FILEINFO, 155
 FILESIZE, 155
 FILESTAT, 155
 FILETIME, 155
 FINE, 539
 FINEA, 273
 FLIN, 458
 FNORM, 458
 FOR, 114
 FPO, 458
 Frame
 Globalne dla NCU, 298
 Łańcuchy frame, 289
 -powiązanie, 304
 Przyporządkowanie, 289
 Specyficzne dla kanału, 299
 Systemowe-, 300

Wywołanie, 288
 FRAME, 24
 Frame systemowe, 300
 Frez
 Punkt pomocniczy (FH), 417
 Wierzchołek (FS), 417
 Frezowanie czołowe, 325
 Frezowanie gwintu - CYCLE70
 programowanie zewnętrzne, 691
 Frezowanie kieszeni konturowej - CYCLE63
 programowanie zewnętrzne, 700
 Frezowanie konturu - CYCLE72
 programowanie zewnętrzne, 696
 Frezowanie obwodowe
 Z powierzchniami ograniczającymi, 419
 Frezowanie płaszczyzny - CYCLE61
 programowanie zewnętrzne, 669
 Frezowanie rowka otwartego - CYCLE899
 programowanie zewnętrzne,
 FROM, 561
 FTOCOF, 405
 FTOCON, 405
 Funkcje OEM, 271
 Funkcje pomocnicze, 582

G

G5, 367
 G62, 272
 G621, 272
 G7, 367
 G810 ... G819, 271
 G820 ... G829, 271
 G-Code
 Programowanie pośrednie, 65
 GEOAX, 593
 GET, 136
 GETACTTD, 432
 GETD, 136
 GETDNO, 431
 GETVARAP, 58
 GETVARDFT, 60
 GETVARLIM, 59
 GETVARPHU, 57
 GETVARTYP, 61
 Głębokość kaskadowania
 Struktury kontrolne, 111
 Głębokość wgłębienia, 416
 GOTO, 99
 GOTOB, 99
 GOTOC, 99
 GOTOF, 99

GOTOS, 98
GP, 66
Grawerowanie (CYCLE60)
 programowanie zewnętrzne, 693
Grupa G
 Technologia, 467
GUD, 24
Gwintowanie otworu bez oprawki kompensacyjnej -
CYCLE84
 programowanie zewnętrzne, 657
Gwintowanie otworu z oprawką kompensacyjną -
CYCLE840
 programowanie zewnętrzne, 660

H

HOLES1 - Szablon pozycji Linia
 programowanie zewnętrzne, 666
HOLES2
 programowanie zewnętrzne, 668

I

ID, 561
IDS, 561
IF, 112
IFRAME, 279
Indeks tablicy, 49
INDEX, 86
INICF, 24
Inicjalizacja
 Tablic, 46
INIPO, 24
INIRE, 24
INIT, 118
INITIAL, 217
INITIAL_INI, 217
Instrukcje
 Lista, 796
INT, 24
Interpolacja wektora obrotu, 339
Interpolacja wielomianowa, 248
INTERSEC, 643
INTTOAX, 54
IPOBRKA, 273
IPOENDA, 273
IPOSTOP, 539
IPTRLOCK, 480
IPTRUNLOCK, 480
ISAXIS, 591
ISD, 410

ISFILE, 153
ISNUMBER, 82
ISOCALL, 199
ISVAR, 55

J

JERKLIM, 492
JERKLIMA, 465

K

Kąt boczny, 320
kąt Eulera, 322
Kąt RPY, 323
Kąt wyprzedzenia, 320
Kieszkań kołowa - POCKET4
 programowanie zewnętrzne, 674
Kieszkań prostokątna - POCKET3
 programowanie zewnętrzne, 671
Kinematyka
 Rozłączona, 438
Komponenty frame
 FI, 287
 SC, 287
 TR, 287
Komponenty Frame
 MI, 287
 RT, 287
Kompresor, 245
Kompresor bloków NC, 245
Kontrolne
 Struktury, 110
Kontur
 dosunięcie przywracające, 483
 Kodowanie, 640
 Przygotowanie, 634
 -tablica, 640
Korekcja długości narzędzia online, 442
Korekcja narzędzia
 Online, 405
 Pamięć korekcji, 391
 Układ współrzędnych dla wartości zużycia, 401
Korekcja narzędzia 3D, 414
 Głębokość wgłębienia, 416
 Korekcja na torze, 415
 Metoda punktu przecięcia, 418
 Zakrzywienie toru, 416
Korekcja promienia narzędzia
 Zwłoka na narożach, 272
Korekcja promienia narzędzia 3D, 410

3D Punkt przecięcia równoległych, 418
 Frezowanie czołowe, 413
 Frezowanie obwodowe, 412
 Naroża wewnętrzne/zewnętrzne, 418
 Okrąg przejściowy, 418
 Korekcje narzędzi
 addytywne, 394
 Kryterium końca ruchu
 Programowane, 273
 KS, 451
 Kształty frezu, 414

L

L..., 189
 LEAD, 319
 LEADOF, 526
 LEADON, 526
 LIFTFAST, 129
 Link
 -Zmienne, 22
 LLI, 36
 LN, 70
 LONGHOLE - Otwór podłużny
 programowanie zewnętrzne, 689
 LOOP, 113
 LUD, 24

Ł

Łańcuch gwintów - CYCLE98
 programowanie zewnętrzne, 713
 Łańcuch znaków
 Długość, 86
 Formatowanie, 90
 Operacje, 81

M

M17, 181
 M30, 181
 Makro, 210
 MASLDEF, 557
 MASLDEL, 557
 MASLOF, 557
 MASLOFS, 557
 MASLON, 557
 MATCH, 86
 MAXVAL, 77
 MCALL, 195
 MEAC, 260

MEAFRAME, 294
 MEAS, 257
 MEASA, 260
 MEAW, 257
 MEAWA, 260
 Miejsca osobliwe, 328
 MINDEX, 86
 MINVAL, 77
 MMC, 607
 MOD, 70
 MODAXVAL, 591
 MPF, 214

N

Nadażanie, 503
 Nadzór narzędzia
 specyficzny dla szlifowania, 589
 NAMETOINT, 384
 Narzędzie
 Korekcja długości, 439
 Korekcja promienia, 397
 -orientacja, 424
 Orientacja przy zmianie frame, 441
 -orientacja, wygładzanie, 351
 -parametry, 391
 Nawiercanie - CYCLE81
 programowanie zewnętrzne, 651
 Nawrót
 Punkt, 568
 NCK, 24
 NEWCONF, 142
 NOC, 539
 Nośnik narzędzi
 Kinematyka, 434
 -Orientowalne, 439
 NOT, 73
 NPROT, 224
 NPROTDEF, 221
 NUMBER, 82
 Numer D
 Dowolne nadanie, 430
 Numer ostrza, 430
 Numery D
 Sprawdzenie, 430
 Zmiana nazwy, 431
 NUT, 332

O

Obrabiany przedmiot

- Katalog główny, 214
 - Katalogi, 214
 - Licznik, 612
 - Obrót
 - wektora orientacji, 339
 - Ochrona
 - Zakresy, 221
 - Odcinki częściowe - automatyczny podział drogi, 582
 - Odniesienie do toru
 - Ustawiane, 254
 - OEMIPO1/2, 271
 - Offset konturu normalny OFFN, 361
 - OMA1 ... OMA5, 271
 - Operatory logiczne, 73
 - Operatory porównania, 73
 - OR, 73
 - ORIXES, 329
 - ORIC, 424
 - ORICONCCW, 332
 - ORICONCW, 332
 - ORICONIO, 332
 - ORICONTO, 332
 - ORICURVE, 335
 - ORID, 424
 - Orientacja narzędzia
 - W stosunku do toru ruchu, 341
 - Orientacja narzędzia odniesiona do toru, 341
 - Orientowane nośniki narzędzi, 434
 - ORIEULER, 329
 - ORIMKS, 327
 - ORIPATH, 343
 - ORIPATHS, 343
 - ORIPLANE, 332
 - ORIRESET(A, B, C), 317
 - ORIROTA, 339
 - ORIROTC
 - przy interpolacji obrotu narzędzia, 344
 - przy obrocie orientacji narzędzia, 339
 - ORIROTR, 339
 - ORIROTT, 339
 - ORIRPY, 329
 - ORIRPY2, 329
 - ORIS, 424
 - ORISOF, 351
 - ORISON, 351
 - ORIVECT, 329
 - ORIVIRT1, 329
 - ORIVIRT2, 329
 - ORIWKS, 327
 - OS, 563
 - OSB, 563
 - OSC, 424
 - OSCILL, 568
 - OSCTRL, 563
 - OSD, 424
 - OSE, 563
 - Osie
 - Nadażanie, 506
 - Osie FGROUP, 254
 - Osie nadażne, 506
 - Osie obrotu
 - Kąt skrętu, 434
 - Wektory kierunku, 434
 - Wektory odległości, 434
 - Osie orientacji, 329
 - Osiowe sprzężenie wartości wiodącej, 526
 - OSNSC, 563
 - OSOF, 424
 - OSP1, 563
 - OSP2, 563
 - OSS, 424
 - OSSE, 424
 - OST, 424
 - OST1, 563
 - OST2, 563
 - Oś
 - Zamiana, 136
 - Oś geometryczna
 - Przełączenie, 593
 - Oś nadażna
 - Przy osiowym sprzężeniu wartości wiodącej, 526
 - przy sterowaniu stycznym, 451
 - Oś skośna (TRAANG), 364
 - Oś wiodąca
 - Przy osiowym sprzężeniu wartości wiodącej, 526
 - przy sterowaniu stycznym, 451
 - OTOL, 496
 - Otwór podłużny - LONGHOLE
 - programowanie zewnętrzne, 689
- P**
- P..., 193
 - P_ACTFRAME, 303
 - Pamięć
 - Pamięć, 213
 - Przebieg wyprzedzający, 472
 - Robocza, 217
 - Pamięć korekcji, 391
 - Pamięć programów
 - Katalogi standardowe, 214
 - Typy plików, 214
 - Pamięć robocza, 217
 - Parametry

- Aktualne, 162
- Formalne, 162
- Narzędzie, 391
- Przekazanie przy wywołaniu podprogramu, 191
- Parametry obliczeniowe
 - Numer n, 20
- Parametry obliczeniowe (R), 20
- PCALL, 200
- PDELAYOF, 577
- PDELAYON, 577
- Pętla bez końca, 113
- Pętla FOR, 114
- Pętla programowa
 - Pętla bez końca, 113
 - Pętla FOR, 114
 - Pętla IF, 112
 - Pętla REPEAT, 117
 - Pętla WHILE, 116
- PFRAME, 279
- PHI
 - przy orientacji wzdłuż poboczniczy stożka, 332
 - Wielomiany orientacji, 338
- PHU, 38
- PL
 - przy interpolacji wielomianowej, 248
 - W przypadku interpolacji Spline, 234
- Plik
 - Informacje, 155
- PO, 248
- PO[PHI]
 - przy obrocie orientacji narzędzia, 343
 - przy orientacji wzdłuż poboczniczy stożka, 332
 - Wielomiany orientacji, 338
- PO[PSI]
 - przy obrocie orientacji narzędzia, 343
 - przy orientacji wzdłuż poboczniczy stożka, 332
 - Wielomiany orientacji, 338
- PO[THT]
 - przy obrocie orientacji narzędzia, 343
 - Wielomiany orientacji, 338
- PO[XH]
 - przy zadaniu orientacji dwóch punktów styku, 335
 - Wielomiany orientacji, 338
- PO[YH]
 - przy zadaniu orientacji dwóch punktów styku, 335
 - Wielomiany orientacji, 338
- PO[ZH]
 - przy zadaniu orientacji dwóch punktów styku, 335
 - Wielomiany orientacji, 338
- POCKET3 - Kieszon prostokątna
 - programowanie zewnętrzne, 671
- POCKET4 - Kieszon kołowa
 - programowanie zewnętrzne, 674
- Podcięcie - CYCLE940
 - programowanie zewnętrzne, 708
- Podprogram
 - Nazwa, 160
 - powrót, parametryzowalny, 183
 - Powtórzenie, 193
 - Programowana ścieżka szukania, 201
 - Wywołanie bez przekazania parametrów, 189
 - Wywołanie z przekazaniem parametrów, 191
 - wywołanie, modalne, 195
 - wywołanie, pośrednie, 197
 - Zastosowanie, 159
- Podział drogi, 586
- Podział drogi w przypadku osi uczestniczących w tworzeniu konturu, 585
- Polecenia programowe
 - Lista, 796
- POLF
 - dla wycofania prowadzonego przez NC, 625
- POLFA, 625
- POLFMASK
 - dla wycofania prowadzonego przez NC, 625
- POLFMLIN
 - dla wycofania prowadzonego przez NC, 625
- POLY, 248
- POLYPATH, 248
- Położenie podstawowe orientacji narzędzia
- ORIRESET, 318
- PON, 586
- PONS, 577
- POSFS, 539
- POT, 70
- Powiązanie
 - Łańcuchy znaków, 84
- Powtórzenie części programu
 - Z programowaniem pośrednim CALL, 198
- Pozostały czas
 - Dla obrabianego przedmiotu, 611
- PREPRO, 180
- PRESETON, 293
- PRIO, 126
- PRLOC, 24
- PROC, 166
- Procedura przerywania, 124
 - Kasowanie, 128
 - Nowe przyporządkowanie, 127
 - Programowany kierunek ruchu, 132
 - Przyporządkowanie i uruchomienie, 126
 - Ruch wycofania, 131
 - Szybkie cofnięcie od konturu, 129
 - Wyłączenie/włączenie, 127

- Process DataShare, 614
- Program
 - Czasy przebiegu, 609
 - Inicjalizacyjny, 218
 - koordynacja, 118
 - Pamięć, 214
 - Powtórzenie, 193
 - Rozgałęzienie, 102
 - Skoki, 99
- Program inicjalizacyjny, 218
- Programowanie orientacji, 329
- Programowanie pośrednie
 - Adresów, 62
 - atrybutów pozycji, 66
 - G-Code, 65
 - wierszy programów obróbki, 69
- PROTA, 386
- PROTD, 389
- PROTS, 388
- Przebieg wyprzedzający
 - Pamięć, 472
- Przecinanie - CYCLE92
 - programowanie zewnętrzne, 717
- Przekładnia elektroniczna, 531
- Przełączane osie geometryczne, 593
- Przesunięcie preset, 293
- Przesunięcie punktu zerowego
 - PRESETON, 293
 - Zewnętrzne przesunięcie punktu zerowego, 292
- Przygotowanie konturu
 - Zwrotna sygnalizacja błędu, 648
- Przyśpieszenie drugiego stopnia
 - Korekcja, 492
 - Ograniczenie, 463
- PSI
 - przy orientacji wzdłuż pobocznic stożka, 332
 - Wielomiany orientacji, 338
- PTP, 369
- PTP przy TRANSMIT, 373
- PTPG0, 373
- PUD, 24
- PUNCHACC, 577
- PUTFTOC, 405
- PUTFTOCF, 405
- PW, 234
- RELEASE, 136
- REP, 46
- REPEAT, 104
- REPEATB, 104
- REPOS, 124
- REPOSA, 483
- REPOSH, 483
- REPOSHA, 483
- REPOSL, 483
- REPOSQ, 483
- REPOSQA, 483
- RET, 182
- RET (parametryzowalny), 183
- RINDEX, 86
- RMBBL, 483
- RMEBL, 483
- RMIBL, 483
- RMNBL, 483
- Rodzaj sprzężenia, 542
- Rodzaje transformacji
 - Funkcja ogólna, 305
- ROUND, 70
- ROUNDUP, 158
- Rowek kołowy - SLOT2
 - programowanie zewnętrzne, 684
- Rowek podłużny - SLOT1
 - programowanie zewnętrzne, 682
- Rozszerzona funkcja pomiarowa, 369
- Rozwiercanie - CYCLE85
 - programowanie zewnętrzne, 653
- Ruch kartezjański PTP, 310
- Ruch pojedynczej osi, 587
- Ruch synchroniczny
 - dokładnie, 542
 - Zgrubnie, 542
- Ruch wahliwy
 - Asynchroniczny, 563
 - Asynchroniczny ruch wahliwy, 563
 - Dosuw częściowy, 571
 - Dosuw w punkcie nawrotu, 573
 - Obszar nawrotu, 571
 - Punkt nawrotu, 571
 - Sterowanie przez akcję synchroniczną, 568
 - Synchroniczny ruch wahliwy, 568
 - Zablokowanie dosuwu, 571

R

- R..., 20
- READ, 150
- REAL, 24
- REDEF, 30

S

- SAVE, 170
- SBLOF, 171
- SBLON, 171
- SCPARA, 601

- SD, 234
SD42475, 349
SD42476, 349
SD42477, 349
SD42678, 351
SD42680, 351
SD42900, 399
SD42910, 399
SD42920, 400
SD42930, 401
SD42935, 403
SD42940, 404
SD42984, 446
SET, 46
SETAL, 622
SETDNO, 431
SETINT, 126
SETM, 118
SIN, 70
Skasowanie pozostałej drogi, 267
Skok
 Do początku programu, 98
 do znaczników skoku, 99
Skrawanie
 Funkcje wspomagające, 633
Skrawanie konturu - CYCLE95
 programowanie zewnętrzne,
Skrawanie warstwowe - CYCLE951
 programowanie zewnętrzne,
Skręt - CYCLE800
 programowanie zewnętrzne,
SLOT1 - Rowek podłużny
 programowanie zewnętrzne, 682
SLOT2 - Rowek kołowy
 programowanie zewnętrzne, 684
SOFT, 463
SOFTA, 463
SON, 577
SONS, 577
SPATH, 254
SPF, 214
SPI, 591
SPIF1, 577
SPIF2, 577
Spline
 Interpolacja, 234
 Typy, 240
SPLINEPATH, 244
SPN, 582
SPOF, 577
SPP, 582
SPRINT, 90
Sprzężenie, 451
 Rodzajowe, 550
Sprzężenie prędkości, 542
Sprzężenie wartości rzeczywistej, 542
Sprzężenie wartości wiodącej
 Sprzężenie wartości rzeczywistej i zadanej, 530
 Synchronizacja osi wiodącej i osi nadążnej, 529
Sprzężenie wartości zadanej, 542
SQRT, 70
START, 118
STARTFIFO, 472
STAT, 369
Status sprzężenia
 Przy nadążaniu, 507
 Przy osiowym sprzężeniu wartości wiodącej, 531
Status zlecenia pomiarowego, 269
Sterowanie styczne, 451
STOLF, 500
STOPFIFO, 472
STOPRE, 472
String
 Powiązanie, 84
STRING, 24
STRINGIS, 602
STRLEN, 86
SUBSTR, 88
Symulacja wartości wiodącej, 530
Synchroniczność liczby obrotów, 539
Synchroniczny ruch wahliwy
 Akcje synchroniczne, 572
 Dosuw w obszarze nawrotu, 573
 Ewaluacja, takt IPO, 574
 Następny dosuw częściowy, 575
 Przyporządkowanie osi ruchu wahliwego i osi dosuwu, 571
 Ruch dosuwu, 573
 Ustalenie dosuwów, 572
 Zatrzymanie w punkcie nawrotu, 574
Synchronizacja położenia, 539
Synchronizacja położenia z zależnością kątową, 539
SYNR, 24
SYNRW, 24
SYNW, 24
System
 Zależna dostępność, 5
Szablon pozycji Linia (HOLES1)
 programowanie zewnętrzne, 666
Szablon pozycji Okrąg - HOLES2
 programowanie zewnętrzne, 668
Szablon pozycji siatka/ramka - CYCLE801
 programowanie zewnętrzne, 667
Szybkie cofnięcie od konturu, 129

- Ś**
- Ścieżka szukania
 - Programowana ścieżka szukania, 201
 - Przy wywołaniu podprogramu, 216
- T**
- Tablica, 46
 - definicja, 46
 - Element, 46
 - TAN, 70
 - TANG, 451
 - TANGDEL, 451
 - TANGOF, 451
 - TANGON, 451
 - TCARR, 439
 - TCOABS, 439
 - TCOFR, 439
 - TCOFRX, 439
 - TCOFRY, 439
 - TCOFRZ, 439
 - THETA
 - przy interpolacji obrotu narzędzia, 344
 - przy obrocie orientacji narzędzia, 339
 - TILT, 319
 - TLIFT, 451
 - Tłoczenie
 - aktywowanie/dezaktywowanie, 577
 - Automatyczny podział drogi, 582
 - TMOF, 589
 - TMON, 589
 - Toczenie gwintu - CYCLE99
 - programowanie zewnętrzne, 710
 - TOFFOF, 442
 - TOFFON, 442
 - Tolerancja
 - W przypadku G0, 500
 - TOLOWER, 85
 - TOUPPER, 85
 - TOWBCS, 401
 - TOWKCS, 401
 - TOWMCS, 401
 - TOWSTD, 401
 - TOWTCS, 401
 - TOWWCS, 401
 - TRAANG, 364
 - TRACON, 378
 - TRACYL, 355
 - TRAFOOF, 378
 - TRAILOF, 503
 - TRAILON, 503
 - Transformacja biegunowa, 309
 - Transformacja orientacji TRAORI
 - Kinematyka maszyny, 308
 - Programowanie orientacji, 317
 - Ruchy postępowe i ruchy orientacji, 307
 - Transformacja rodzajowa 5/6-osiowa, 309
 - Warianty programowania orientacji, 318
 - Transformacja poboczniczy walca, 309
 - Transformacja ze skrętną osią liniową, 315
 - Transformacje
 - Niezależne od kinematyki położenie podstawowe orientacji narzędzia, 306
 - Powiązane, 378
 - Transformacja orientacji, 306
 - Transformacja trzy-, cztery- i pięcioosiowa, 316
 - Transformacje kinematyczne, 306
 - Transformacje powiązane, 307
 - TRANSMIT, 353
 - TRAORI, 316
 - TRUE, 24
 - TRUNC, 70
 - Tryb pracy
 - Przy pomiarze, 266
 - Tryb przyspieszenia, 463
 - TU, 369
 - Typ kinematyki, 438
- U**
- Ujęcie i szukanie obszarów niezdatnych do przeszukiwania, 481
 - ULI, 36
 - UNTIL, 117
 - UPATH, 254
- V**
- VAR, 167
 - VELOLIM, 493
 - VELOLIMA, 465
- W**
- WAITC, 539
 - WAITE, 118
 - WAITENC, 600
 - WAITM, 118
 - Wartość ustawcza, 395
 - Wartość zużycia, 395
 - Warunki brzegowe przy transformacjach, 377
 - Wcinanie - CYCLE952

programowanie zewnętrzne,
 Wektor kierunkowy, 324
 Wektor orientacji THETA, 339
 WHEN, 561
 WHEN-DO, 572
 WHENEVER, 561
 WHENEVER-DO, 572
 WHILE, 116
 Wielobok - CYCLE79
 programowanie zewnętrzne, 680
 Wielomian mianownikowy, 252
 Wiercenie - CYCLE82
 programowanie zewnętrzne, 652
 Wiercenie głębokich otworów - CYCLE83
 programowanie zewnętrzne, 654
 Wiercenie otworu z frezowaniem gwintu - CYCLE78
 programowanie zewnętrzne,
 Wiercenie wstępne kieszeni konturowej - CYCLE64
 programowanie zewnętrzne,
 WRITE, 143
 Wrzecziono
 Zamiana, 136
 Wrzecziono synchroniczne
 Sprzężenie, 539
 Ustalenie pary, 545
 Współczynnik sprzężenia, 503
 Współczynnik tolerancji G0, 500
 Współczynnik wielomianu, 249
 Wycofanie
 Prowadzone przez NC, 625
 wystarczające dla napędu, 631
 Wygładzanie
 Przebieg orientacji, 351
 Wykonywanie pojedynczymi blokami
 Blokowanie, 171
 Wyprowadzenie
 na zewnętrznym urządzeniu/pliku, 614
 Wyświetlanie bloku, 199
 Blokowanie, 177
 Wytaczanie - CYCLE86
 programowanie zewnętrzne, 656
 Wytoczenie - CYCLE930
 programowanie zewnętrzne, 705
 Wywołanie konturu - CYCLE62
 programowanie zewnętrzne, 695
 Wywołanie podprogramu z podaniem ścieżki i
 parametrami, 200
 Wyzwolenie skoku, 580

X

xe ye ze, 335

XH YH ZH, 335
 xi yi zi, 335
 XOR, 73

Z

Zaokrąglenie do góry, 158
 Zatrzymanie
 Prowadzone przez NC, 629
 wystarczające dla napędu, 630
 Zdarzenie przerzutnikowe
 Przy pomiarze, 265
 Zespół nadażania, 503
 Zespół spline, 244
 Zewnętrzne przesunięcie punktu zerowego, 292
 Zmienna
 Konwersja typów, 80
 Zmienna frame
 Definicja nowych frame, 289
 Predefiniowana zmienna frame, 288
 Przyporządkowanie wartości, 284
 Wywoływanie transformacji współrzędnych, 277
 Zmienna Frame
 Predefiniowana zmienna frame, 279
 Przesunięcia punktu zerowego G54 do G599, 283
 Przyporządkowanie do poleceń G54 do G599, 283
 Zmienne
 Definiowane przez użytkownika, 24
 Konwersja typu, 81
 Zmienne systemowe
 Ograniczenie sondy pomiarowej, 269
 Status sondy pomiarowej, 269
 Znacznik skoku
 przy powtórzeniach części programu, 104
 W przypadku skoków w programie, 100
 Znak 0, 81
 Zwłoka na narożach wewnętrznych, 272
 Zwłoka na wszystkich narożach, 272

