SIEMENS

SINUMERIK

SINUMERIK 840D sl
NC variable and interface signals

Parameter Manual

Valid for
Control system
SINUMERIK 840D sl / 840DE sl
Software
CNC software, version 4.7 SP2

10/2015
6FC5397-3CP40-5BA3

Legal information

Warning notice system

This manual contains notices you have to observe in order to ensure your personal safety, as well as to prevent damage to property. The notices referring to your personal safety are highlighted in the manual by a safety alert symbol, notices referring only to property damage have no safety alert symbol. These notices shown below are graded according to the degree of danger.

⚠ DANGER
indicates that death or severe personal injury **will** result if proper precautions are not taken.

⚠ WARNING
indicates that death or severe personal injury **may** result if proper precautions are not taken.

⚠ CAUTION
indicates that minor personal injury can result if proper precautions are not taken.

NOTICE
indicates that property damage can result if proper precautions are not taken.

If more than one degree of danger is present, the warning notice representing the highest degree of danger will be used. A notice warning of injury to persons with a safety alert symbol may also include a warning relating to property damage.

Qualified Personnel

The product/system described in this documentation may be operated only by **personnel qualified** for the specific task in accordance with the relevant documentation, in particular its warning notices and safety instructions. Qualified personnel are those who, based on their training and experience, are capable of identifying risks and avoiding potential hazards when working with these products/systems.

Proper use of Siemens products

Note the following:

⚠ WARNING
Siemens products may only be used for the applications described in the catalog and in the relevant technical documentation. If products and components from other manufacturers are used, these must be recommended or approved by Siemens. Proper transport, storage, installation, assembly, commissioning, operation and maintenance are required to ensure that the products operate safely and without any problems. The permissible ambient conditions must be complied with. The information in the relevant documentation must be observed.

Trademarks

All names identified by ® are registered trademarks of Siemens AG. The remaining trademarks in this publication may be trademarks whose use by third parties for their own purposes could violate the rights of the owner.

Disclaimer of Liability

We have reviewed the contents of this publication to ensure consistency with the hardware and software described. Since variance cannot be precluded entirely, we cannot guarantee full consistency. However, the information in this publication is reviewed regularly and any necessary corrections are included in subsequent editions.

Siemens AG
Division Digital Factory
Postfach 48 48
90026 NÜRNBERG
GERMANY

Order number: 6FC5397-3CP40-5BA3
Ⓟ 11/2015 Subject to change

Preface

SINUMERIK documentation

The SINUMERIK documentation is organized in the following categories:

- General documentation
- User documentation
- Manufacturer/service documentation

Additional information

You can find information on the following topics at www.siemens.com/motioncontrol/docu:

- Ordering documentation/overview of documentation
- Additional links to download documents
- Using documentation online (find and search in manuals/information)

Please send any questions about the technical documentation (e.g. suggestions for improvement, corrections) to the following address:

docu.motioncontrol@siemens.com

My Documentation Manager (MDM)

Under the following link you will find information to individually compile OEM-specific machine documentation based on the Siemens content:

www.siemens.com/mdm

Training

For information about the range of training courses, refer under:

- www.siemens.com/sitrain
  SITRAIN - Siemens training for products, systems and solutions in automation technology
- www.siemens.com/sinutrain
  SinuTrain - training software for SINUMERIK

FAQs

You can find Frequently Asked Questions in the Service&Support pages under Product Support. http://support.automation.siemens.com

SINUMERIK

You can find information on SINUMERIK under the following link:

www.siemens.com/sinumerik

Target group

This publication is intended for project engineers, commissioning engineers, machine operators and service and maintenance personnel.

Benefits

The intended target group can use the Parameter Manual to test and commission the system or the plant correctly and safely.

Utilization phase: Setup and commissioning phase

Standard scope

This documentation describes the functionality of the standard scope. Extensions or changes made by the machine manufacturer are documented by the machine manufacturer.

Other functions not described in this documentation might be executable in the control. This does not, however, represent an obligation to supply such functions with a new control or when servicing.

Furthermore, for the sake of clarity, this documentation does not contain all detailed information about all types of the product and cannot cover every conceivable case of installation, operation or maintenance.

Technical Support

You will find telephone numbers for other countries for technical support in the Internet under http://www.siemens.com/automation/service&support

Table of contents

1 Fundamental safety instructions

1.1 General safety instructions

> ⚠ **WARNING**
>
> **Risk of death if the safety instructions and remaining risks are not carefully observed**
>
> If the safety instructions and residual risks are not observed in the associated hardware documentation, accidents involving severe injuries or death can occur.
>
> - Observe the safety instructions given in the hardware documentation.
> - Consider the residual risks for the risk evaluation.

> ⚠ **WARNING**
>
> **Danger to life or malfunctions of the machine as a result of incorrect or changed parameterization**
>
> As a result of incorrect or changed parameterization, machines can malfunction, which in turn can lead to injuries or death.
>
> - Protect the parameterization (parameter assignments) against unauthorized access.
> - Respond to possible malfunctions by applying suitable measures (e.g. EMERGENCY STOP or EMERGENCY OFF).

1.2 Industrial security

Note

Industrial security

Siemens provides products and solutions with industrial security functions that support the secure operation of plants, solutions, machines, equipment and/or networks. They are important components in a holistic industrial security concept. With this in mind, Siemens' products and solutions undergo continuous development. Siemens recommends strongly that you regularly check for product updates.

For the secure operation of Siemens products and solutions, it is necessary to take suitable preventive action (e.g. cell protection concept) and integrate each component into a holistic, state-of-the-art industrial security concept. Third-party products that may be in use should also be considered. For more information about industrial security, visit this address (http://www.siemens.com/industrialsecurity).

To stay informed about product updates as they occur, sign up for a product-specific newsletter. For more information, visit this address (http://support.automation.siemens.com).

WARNING

Danger as a result of unsafe operating states resulting from software manipulation

Software manipulation (e.g. by viruses, Trojan horses, malware, worms) can cause unsafe operating states to develop in your installation which can result in death, severe injuries and/or material damage.

- Keep the software up to date.
  You will find relevant information and newsletters at this address (http://support.automation.siemens.com).
- Incorporate the automation and drive components into a holistic, state-of-the-art industrial security concept for the installation or machine.
  You will find further information at this address (http://www.siemens.com/industrialsecurity).
- Make sure that you include all installed products into the holistic industrial security concept.

Introduction

2

2.1 NC variables

General

The NC variables can be accessed from the user interface or from the PLC via the operator panel interface (OPI).

The structure of the NC variables and their processing is described in Section Explanations on the NC variables (Page 17).

The tables of the NC variables contain references to further manuals with detailed information.

The list of manuals contains the manuals referred to, see Section References (Page 15)

2.2 Interface signals

This manual contains an overview of the interface assignment. Please refer to other manuals for a detailed description of the functions.

References

References to the detailed description of an interface signal are via the unique section numbers in the Basic, Extended and Special Function Manuals: /.../

Example:

DB10.DBX108.5:	Drives in cyclic operation /Z1-A2/
Z1:	Function Manual, 840D sl Basic Functions, NC/PLC interface signals (Z1)
A2:	Function Manual, 840D sl Basic Functions; Various interface signals and functions (A2)

An overview of the manuals is provided in Section References (Page 15).

Inverse signals

Inverse signals are designated with "*".

Example:

Signals from the machine control panel, EB n + 2, DBX4: *Spindle stop:

1: Spindle stop is not requested

0: Spindle stop is requested

Abbreviations

Information about abbreviations and their meaning is provided in Section List of abbreviations (Page 833).

2.3 References

General

A detailed description of the NC variables and signals can be found in the following references. The reference to the appropriate manual is made up as follows:

- Abbreviation of the manual
- Abbreviation of the sub-book

Example: References for the NC variables

W1: Function Manual, Basic Functions; W1: Tool offset
FBW: Function Manual, Tool Management

Example: References for the NC/PLC interface signals

DB10.DBB0: Disable of the digital NC inputs /Z1-A2/

FB1 Function Manual, Basic Functions
A2 A2: Various NC/PLC interface signals and functions

List of manuals

Reference is made to the following manuals:

Manual abbreviation	Manual name	Abbreviation of the sub-books
/FB1/	Function Manual, Basic Functions	A2, A3, B1, B2, F1, G2, H2, K1, K2, N2, P1, P3, P4, R1, S1, V1, W1, Z1
/FB2/	Function Manual, Extended Functions	A4, B3, H1, K3, K5, M1, M5, N3, N4, P2, P5, R2, S3, S7, T1, W3, W4, Z2
/FB3/	Function Manual, Special Functions	F2, G1, K6, K7, K8, K9, M3, R3, S9, T3, TE01, TE02, TE1, TE3, TE4, TE6, TE7, TE8, TE9, V2, W5, W6, Z3
/IHsl/	Commissioning Manual, Basic Software and Operating Software	IM9, BE2, IM7, IM8, IM10
/FBWsl/	Function Manual, Tool Management	
/SCE/	System Manual, Ctrl-Energy	
/FBSIsl/	SINUMERIK Safety Integrated Function Manual	
/PGAsl/	Programming Manual Advanced	
/LIS3sl/	List Manual, System Variables	

Additional references

- For SINAMICS drives, also note the following documents:
  - SINAMICS S120, Commissioning Manual
  - SINAMICS S120/S150, List Manual
- For SINUMERIK Integrate (previously ePS Network Services) (remote diagnostics, control and condition monitor services, data, workflow and administration services), also note the following document:
  - Function Manual, SINUMERIK Integrate for production
- The input and output images of the machine control panel and the handheld units can be found in the following document:
  - Manual, Operator Components and Networking

NC variable 3

3.1 Explanations on the NC variables

3.1.1 NC areas

NC areas

The NC variables are organized in data blocks that are assigned to the following NC areas.

Table 3-1 Assignment of the TO areas

Area	NC variable
NC (N)	Contains all variables that apply to the entire NC, e.g. • System data (Y) • Protection areas (PA) • G groups (YNCFL), etc.
Mode group (B)	Contains all variables that apply to the mode group, e.g. • Status data (S)
Channel (C)	Contains all variables that apply to the relevant channel, e.g. • System data (Y) • Protection areas (PA) • Global status data (S)
Tool (T)	Contains all variables that apply to the tools on the machine, e.g. • Tool offset data (TO) • General tool data (TD) • Tool monitoring data (TS), etc. Each tool area (T) is assigned to a channel.
Axis (A)	Contains machine and setting data that applies to the relevant axis or spindle. See List Manual 1, Section: Axis-specific machine data
Feed/main drive (V/H)	Contains machine data or machine data as service values that apply to the relevant drive.

3.1.2 Data blocks

Overview of the available data blocks

The following table contains an overview of available data blocks for variables of the NC and their assignment to the individual areas.

Only those data blocks whose variables can be read or written with direct access are represented.

Data blocks whose variables can be freely defined by the programmer (e.g. global user data) are read from the operating software or PLC using other mechanisms.

Data block	Area						
	A	B	C	H	N	T	V
ETP			x				
ETPD					x		
DIAGN			x				
FA			x		x		
FB			x		x		
FE			x				
FU			x		x		
M	x				x		
NIB			x				
PA			x		x		
RP			x				
S		x	x	x	x		x
SALA					x		
SALAL					x		
SALAP					x		
SE	x		x		x		
SEGA			x				
SEMA			x		x		
SGA			x				
SINF			x				
SMA			x		x		
SNCF			x				
SPARP			x				
SPARPF			x				
SPARPI			x				
SPARPP			x				
SSP			x		x		
SSP2			x		x		
SSYNAC			x				
SYNACT			x				
TD						x	
EP						x	
TG						x	
TM						x	
TMC						x	
TMV						x	
TO						x	

Data block	Area						
	A	B	C	H	N	T	V
TP						x	
TPM						x	
TS						x	
TT						x	
TU						x	
TUE						x	
TUM						x	
TUP						x	
TUS						x	
TV						x	
AD						x	
AEV						x	
TC						x	
TOE						x	
TOET						x	
TOS						x	
TOST						x	
TOT						x	
VSYN		x					
Y		x			x		
YNCFL					x		

References

Further information on which data blocks this mechanism is used, can be found in the following manual: Function Manual, Basic functions; P3: Basic PLC program

3.1.3 Variable types

Access to an NC variable

In general, the NC variables are stored as structures or arrays of structures (tables) in the areas. This means the following details must be specified in the address for access to an NC variable:

- Area and area number
- Block
- NC variable name (or column number)
- Line number

NC variable type

Generally, a distinction can be made between three variants for the NC variables:

- NC variables that consist of one line
- NC variables that consist of several lines
- NC variables that consist of several columns and lines

Single-line NC variable

A single-line NC variable consists of just a single value. The following information is required to access an NC variable of this type:

- Area (and possibly the area number)
- Block
- NC variable name

Table 3-2 Single-line NC variant type

numMachAxes					
Number of available machine axes					
-				Word	r
Multi-line: No					

Example for reading the number of machine axes in channel 1:

HMI:
/Channel/Configuration/numMachAxes[u1]

HMI:
P_C_Y_numMachAxes

PLC with NC variable selector:

Area	C[.]
Block	Y
NC variable	numMachAxes
Area number	1

Multi-line NC variable

This NC variable is generally defined as a one-dimensional array. The following information is required to access an NC variable of this type:

- Area and possibly the area number
- Block
- NC variable name
- Line number

Table 3-3 Multi-line variant type

actFeedRate	**$AA_VACTB[x]**				**S5**
Actual axial feed value (only when the axis is a positioning axis, "spec" = 1)					
%				Double	r
Multi-line: Yes	Axis index		numMachAxes		

Example for reading the current speed of axis 3 in channel 1:

HMI:
/Channel/MachineAxis/actFeedRate[u1, 3]
HMI:
P_C_SEMA_actFeedRate

PLC with NC variable selector:

Area	C[.]
Block	SEMA
NC variable	actFeedRate[.]
Area number	1
Line	3

Multi-line and multi-column NC variable

This NC variable is generally defined as a two-dimensional array. The following information is required to access an NC variable of this type:

- Area and possibly the area number
- Block
- NC variable name
- Column number
- Line number

In this example, the complete data block consists only of these two-dimensional NC variables.

Table 3-4 Multi-line and multi-column variant type

cuttEdgeParam	**$TC_DPx[y,z]**				
Compensation value parameters for a tool edge					
mm, inch or user-defined	0			Double	wr
Multi-line: Yes	(EdgeNo - 1) * numCuttEdgeParams + ParameterNo		numCuttEdgeParams * numCuttEdges		

Example of reading and writing the current cutting edge data for cutting edge 3 / parameter 1 of tool 3 in T area 1.

The example assumes that each tool cutting edge is defined with (numCuttEdgeParams =) 25 parameters:

HMI:
/Tool/Compensation/cuttEdgeParam[u1,c3, 51]

HMI:
P_T_TO_cuttEdgeParam
[y,z] array which is queried in SINUMERIK Operate.

PLC with NC variable selector:

Area	T[.]
Block	TO
NC variable	cuttEdgeParam[.]
Area number	1
Column	3
Line	51

3.1.4 Data types

The control provides the following data types that can be used for programming:

Table 3-5 Data types

Data type	Size
BOOL	1 bit
CHAR	8 bits without sign
Byte	8 bits with sign
Word	16 bits without sign
Short integer	16 bits with sign
Double-word	32 bits without sign
Long integer	32 bits with sign
Float	32-bit floating-point
REAL	32 bit
Double	64-bit floating-point
STRING	Null-terminated string

3.1.5 Structure of the data tables

Table fields

Table 3-6 Meaning of table fields

NC variable name	Reference to assigned machine data				Lit.
NC variable brief description / NC variable description <DescriptionValue range>					
Physical unit	Default value	Lower limit	Upper limit	Format / field length	w / r
Multi-line: Yes/no	Line index description		Maximum line index		

Lit. Reference according to the list of references

See: Chapter References (Page 15)

w / r

- w Variable may be overwritten
- r Variable can be read

3.2 System data

3.2.1 Area N, Block Y : Global system data

OEM-MMC: Linkitem /NckConfiguration/...

The machine tool builder or user configures the control with the help of the machine data. Configuration can only be performed with certain access rights. The configuration of the NC can be read in the system data regardless of current access rights.

accessLevel					
Level of the access rights currently set. Can be changed by entering the password or turning the keyswitch. 0 = access level SIEMENS 1 = access level machine tool builder 2 = access level system start-up engineer (machine tool builder) 3 = access level end user with password 4 = access level key switch 3 5 = access level key switch 2 6 = access level key switch 1 7 = access level key switch 0					
-				UWord	r
Multi-line: no					

anLanguageOnHmi	$AN_LANGUAGE_ON_HMI	

Current language set on HMI -- [*] available languages
1 German [*]
2 French [*]
3 English (United Kingdom) [*]
4 Spanish [*]
5 Portuguese (Portugal)
6 Italian [*]
7 Dutch [*]
8 Chinese (Simplified) [*]
9 Swedish [*]
10 German (Austria)
11 German (Liechtenstein)
12 German (Luxembourg)
13 German (Switzerland)
15 Norwegian (Bokmål)
16 Norwegian (Nynorsk)
18 Hungarian [*]
19 Finnish [*]
20 French (Belgium)
21 French (Canada)
22 French (Luxembourg)
23 French (Principality of Monaco)
24 French (Switzerland)
26 Greek [*]
28 Czech [*]
30 English (United States)
31 English (Australia)
32 English (Belize)
33 English (Canada)
34 English (Caribbean)
35 English (India)
36 English (Ireland)
37 English (Jamaica)
38 English (Malaysia)
39 English (New Zealand)
40 Spanish (Argentina)
41 Spanish (Bolivia)
42 Spanish (Chile)
43 Spanish (Colombia)
44 Spanish (Costa Rica)
45 Spanish (Dominican Republic)
46 Spanish (Ecuador)
47 Spanish (El Salvador)
48 Spanish (Guatemala)
49 Spanish (Honduras)
50 Portuguese (Brazil) [*]
53 Polish [*]
55 Danish [*]
57 Russian [*]
59 Albanian
60 Italian (Switzerland)

anLanguageOnHmi	$AN_LANGUAGE_ON_HMI	
62 Bosnian (Latin, Bosnia and Herzegovina) 63 Bosnian (Cyrillic, Bosnia and Herzegovina) 65 Croatian (Croatia) [*] 66 Croatian (Latin, Bosnia and Herzegovina) 68 Slovak [*] 69 Slovenian [*] 70 Dutch (Belgium) 72 Romanian [*] 73 Romansh (Switzerland) 75 Bulgarian [*] 76 Estonian 77 Georgian 78 Latvian 79 Lithuanian 80 Chinese (Traditional) [*] 81 Chinese (Hong Kong S.A.R.) 82 Chinese (Macao S.A.R.) 83 Chinese (Singapore) 85 Korean [*] 87 Japanese [*] 88 Macedonian 89 Turkish [*] 90 Swedish (Finland) 92 Ukrainian 93 Afrikaans 94 Alsatian (France) 95 Amharic (Ethiopia) 96 Armenian 97 Azeri 98 Bashkir (Russia) 99 Belarusian 100 Arabic (Saudi Arabia) 101 Arabic (Algeria) 102 Arabic (Bahrain) 103 Arabic (Egypt) 104 Arabic (Iraq) 105 Arabic (Jordan) 106 Arabic (Kuwait) 107 Arabic (Lebanon) 108 Arabic (Libya) 109 Arabic (Morocco) 110 Arabic (Oman) 111 Arabic (Qatar) 112 Arabic (Syria) 113 Arabic (Tunisia) 114 Arabic (U.A.E.) 115 Arabic (Yemen) 118 Assamese 119 Bengali 120 Gujarati 121 Hindi		

anLanguageOnHmi	$AN_LANGUAGE_ON_HMI	
122 Indonesian [*] 123 Kannada 124 Konkani 125 Malayalam 126 Marathi 127 Oriya 128 Punjabi 129 Sanskrit 130 English (Republic of the Philippines) 131 English (Singapore) 132 English (South Africa) 133 English (Trinidad and Tobago) 134 English (Zimbabwe) 137 Pashto (Afghanistan) 138 Dari (Afghanistan) 139 Urdu 140 Spanish (Mexico) 141 Spanish (Nicaragua) 142 Spanish (Panama) 143 Spanish (Paraguay) 144 Spanish (Peru) 145 Spanish (Puerto Rico) 146 Spanish (Spain) 147 Spanish (United States) 148 Spanish (Uruguay) 149 Spanish (Venezuela) 151 Serbian (Latin, Bosnia and Herzegovina) 152 Serbian (Cyrillic, Bosnia and Herzegovina) 155 Malay (Brunei Darussalam) 156 Quechua (Bolivia) 158 Inuktitut (Latin, Canada) 159 Inuktitut (Syllabics, Canada) 160 Mohawk (Mohawk) 162 Mapudungun (Chile) 164 Tibetan (PRC) 165 Yi (PRC) 166 Mongolian (Traditional Mongolian, PRC) 167 Uighur (PRC) 169 Tamazight (Latin, Algeria) 171 Catalan 172 Basque 173 Galician 175 Sami, Northern (Finland) 176 Sami, Inari (Finland) 177 Sami, Skolt (Finland) 180 Breton (France) 181 Corsican (France) 182 Occitan (France) 184 Faroese 186 Tamil 187 Telugu		

anLanguageOnHmi	$AN_LANGUAGE_ON_HMI	
190 Welsh (United Kingdom) 192 Lower Sorbian (Germany) 193 Upper Sorbian (Germany) 195 Greenlandic (Greenland) 196 Icelandic 198 Irish 200 Persian 201 Syriac 203 Hebrew 204 Kazakh 205 Kyrgyz 206 Yakut (Russia) 207 Tatar 208 Uzbek 210 Khmer (Cambodia) 211 Lao 212 Thai [*] 213 Vietnamese [*] 214 Sinhala (Sri Lanka) 215 Filipino (Philippines) 216 Tajik (Cyrillic, Tajikistan) 217 Turkmen 220 Serbian (Cyrillic) 221 Serbian (Latin) 224 K'iche (Guatemala) 225 Kiswahili 226 Luxembourgish 227 Divehi 228 Maltese 229 Mongolian 230 Malay [*] 231 Nepali (Nepal) 232 Hausa (Latin, Nigeria) 233 Igbo (Nigeria) 234 Yoruba (Nigeria) 235 Frisian (Netherlands) 236 Sami, Southern (Norway) 237 Sami, Northern (Norway) 238 Sami, Lule (Norway) 239 Maori (New Zealand) 240 Quechua (Peru) 241 Kinyarwanda (Rwanda) 242 Wolof (Senegal) 243 Sami, Southern (Sweden) 244 Sami, Northern (Sweden) 245 Sami, Lule (Sweden) 246 Sesotho sa Leboa (South Africa) 247 Setswana (South Africa) 248 isiXhosa (South Africa) 249 isiZulu (South Africa) 250 Quechua (Ecuador)		

anLanguageOnHmi	$AN_LANGUAGE_ON_HMI				
-	2	0	255	UWord	rw
Multi-line: no			1		

axisType					
Axis types for all machine axes (necessary for start-up): If a machine axis is addressed via the M module, the units and values are returned with reference to the axis type accessible via this variable. (The absolute machine axis index 1-N_Y_maxnumGlobMachAxes is specified via the line index) 0 = Linear axis 1 = Rotary axis					
-				UWord	r
Multi-line: yes	Absolute machine axis number		maxnumGlobMachAxes		

basicLengthUnit					
Global basic unit 0 = mm 1 = inch 4 = userdef					
-				UWord	r
Multi-line: no					

chanAssignment	MD 10010: ASSIGN_CHAN_TO_MODE_GROUP[x] x=ChannelNo				K1
Assignment of each channel to mode group 0 = channel does not exist n = channel assigned to mode group n (n is maximum numBAGs (BAG = mode group))					
-				UWord	r
Multi-line: yes	Channel number		maxnumChannels		

driveTypeSupport					
Type of supported drive 0 = stepper 1 = digital					
-				UWord	r
Multi-line: no					

exportRestricted					
Export restriction Identification of the software subject to export restriction according to the Federal Office of Economics and Export Control, and Export Control and Customs Compare OPI N/Y exportRestricted					
-	1			Bool	r
Multi-line: no					

externCncSystem					
CNC system whose part programs must be processed on the SINUMERIK control. 0: No external language defined 1: System ISO Dialect0 Milling (obsolete) 2: System ISO Dialect0 Turning (obsolete) 3: External language via OEM application (from P6.2) 4: System ISO Dialect0 Milling (from P7.) 5: System ISO Dialect0 Turning (from P7.) etc.					
-				UWord	r
Multi-line: yes	1		1		

extraCuttEdgeParams					
Bit string that specifies which TO edge parameters are available in addition to the 25 standard parameters. Bit 0: Cutting edge parameter no. 26 valid (ISO Dialect Milling H No.) Bit 1: Cutting edge parameter no. 27 valid (Orientation of the cutting edge) Bit 2: Cutting edge parameter no. 28 valid (L1 of the orientation of the cutting edge) Bit 3: Cutting edge parameter no. 29 valid (L2 of the orientation of the cutting edge) Bit 4: Cutting edge parameter no. 30 valid (L3 of the orientation of the cutting edge) Bit 5: Cutting edge parameter no. 31 valid (L1 of the orientation of the cutting edge normal) Bit 6: Cutting edge parameter no. 32 valid (L2 of the orientation of the cutting edge normal) Bit 7: Cutting edge parameter no. 33 valid (L3 of the orientation of the cutting edge normal) Bit 8: Cutting edge parameter no. 34 valid (number of cutting edge teeth, always set) Bit 9: Cutting edge parameter no. 35 valid (basic angle of rotation of the cutting edge, always set) etc.					
-				UWord	r
Multi-line: yes	1		1		

handWheelNr					
Handwheel number, as required for selection via the PLC user interface.					
-		0		UWord	r
Multi-line: yes	Handwheel number		numHandWheels		

kindOfSumcorr	$MN_MM_KIND_OF_SUMCORR				
Characteristics of total offsets in NCK: Bit No. Value Meaning 0 0 Total offsets are saved at the same time as the tool data. 1 Total offsets are not saved at the same time as the tool data. 1 0 Setup offsets are saved at the same time as the tool data. 1 Setup offsets are not saved at the same time as the tool data. 2 0 If the "Tool management" function is in use: The existing total/setup offsets are not affected when tool status "active" is set. 1 When tool status "active" is set, the existing total offsets are set to zero. The setup offsets are not affected. 3 0 If the "Tool management" function plus "Adapter" is in use: Transformation of total offsets 1 No transformation of total offsets 4 0 No setup offset data sets 1 Setup offset data sets are created additionally, in which case the total offset equals the product of total offset + "fine total offset".					
-				UWord	r
Multi-line: yes	1				

maskToolManagement	$MN_MM_TOOL_MANAGEMENT_MASK				
Settings for NCK tool management function Activation of tool management memory with "0" means: The set tool management data do not occupy any memory space. Bit 0=1: Memory for TM-specific data is made available Bit 1=1: Memory for monitoring data is made available Bit 2=1: Memory for user data (CC data) is made available Bit 3=1: Memory for "Consider adjacent location" is made available Bit 5=0: Parameters and function for tool wear monitoring are not available. Bit 5=1: Parameters and function for tool wear monitoring are available and, if bit 1 = 1, the wear monitoring function is also available. Bit 6=0: The wear group function is not available; i.e. parameters $TC_MAMP3, $TC_MAP9 cannot be programmed, $TC_MPP5 is not defined for magazine locations of type 1. Bit 6=1: The wear group function is available; i.e. parameters $TC_MAMP3, $TC_MAP9 can be programmed and wear groups defined. $TC_MPP5 contains the wear group number for location type 1. Bit 7=1: Tool adapter data sets are available. Bit 8=1: Total offsets are available. Bit 9=1: Tools in a turret are handled in OPI variable modules such that they are not "displayed" in tool half-locations, but always displayed in a turret location. Please note, therefore, that tools in a turret remain (in display terms) in their turret location in the event of a tool change. Bit 9=0: Default response; Tools in a turret are "displayed" in the OPI in their actual (according to data) location.					
-	0			Long Integer	r
Multi-line: yes	1				

maxCuttingEdgeNo	$MN_MAX_CUTTING_EDGE_NO				
Maximum value of D number 1 to 32000					
-	9	1	32000	UWord	r
Multi-line: yes	1				

maxNoOfChannels					
Maximum number of channels that can be activated. This defines the upper limit of the option data $ON_NUM_CHANNELS.					
-	1	1		UWord	r
Multi-line: yes	1		1		

maxNoOfProgLevel					
Maximum number of program levels present in the system. This defines the upper limit of the option data $ON_NUM_CHANNELS.					
-				UWord	r
Multi-line: yes	1		1		

maxNumAdapter	$MN_MM_NUM_TOOL_ADAPTER				
Maximum number of tool adapter data sets available in NCK >0: Maximum number of adapter data sets. 0: Adapter data cannot be defined. Edge-specific parameters $TC_DP21, $TC_DP22, $TC_DP23 are available, i.e. active tool management function with adapters is not in use. -1: An adapter is automatically assigned to each magazine location, i.e. the number of adapters provided internally corresponds to the number of magazine locations set in machine data $MN_MM_NUM_MAGAZINE_LOCATION.					
-	0	-1	600	Long Integer	r
Multi-line: yes	1				

maxNumNcusInNcuCluster					
Maximum number of NCUs in the NCU link					
-				UWord	r
Multi-line: yes	1		1		

maxNumPlacesPerMultitool	-				
Maximum number of places per multi-tool; specified by $MN_MAX_TOOLS_PER_MULTITOOL					
-				UWord	r
Multi-line: yes	1		1		

maxNumSumCorr	$MN_MM_NUM_SUMCORR				
Total number of total offsets in NCK A setting of -1 means that the number of total offsets equals the number of edges * number of total offsets per edge. A setting of > 0 and < number of edges * number of total offsets per edge means that a maximum number of total offsets equalling "number of total offsets per edge" can be defined per edge, but need not be, i.e. it is thus possible to use the buffer memory more economically. In other words, only the edges have a total offset data set for which data can be defined explicitly.					
-				Long Integer	r
Multi-line: yes	1				

maxnumAlarms					
Size of NCK alarm buffer (maximum number of pending alarms)					
-				UWord	r
Multi-line: no					

maxnumChannels					
Maximum number of available channels					
-				UWord	r
Multi-line: no					

maxnumContainer					
Maximum number of available axis containers					
-		0		UWord	r
Multi-line: yes	1		1		

maxnumContainerSlots					
Maximum number of available slots per axis container					
-				UWord	r
Multi-line: yes	1		1		

maxnumCuttEdges_Tool	$MN_MAX_CUTTING_EDGE_PER_TOOL				
Max. number of edges per tool 1 to 12					
-	9			UWord	r
Multi-line: yes	1				

maxnumDrives					
Maximum number of available drives					
-				UWord	r
Multi-line: no					

maxnumEdgeSC	$MN_MAX_SUMCORR_PERCUTTING_EDGE				
Max. number of total offsets per edge 0 to 6					
-	0			UWord	r
Multi-line: yes	1				

maxnumEventTypes					
Maximum number of event types for the trace protocolling					
-				UWord	r
Multi-line: no					

maxnumGlobMachAxes					
Maximum number of available machine axes					
-				UWord	r
Multi-line: no					

maxnumTraceProtData					
Maximum number of data per data list for trace protocolling					
-				UWord	r
Multi-line: no					

maxnumTraceProtDataList					
Maximum number of data per data list for trace protocolling					
-				UWord	r
Multi-line: no					

modeSpindleToolRevolver	MD $MN_MM_TOOL_MANAGEMENT_MASK Bit 9				
Representation of tool currently in use in modules magazine location data (T / TP, magazine data, location data) and tool data (T / TD, tool data, general data and T / TV, tool data, directory) 0: Previous method: During operation, the tool is removed (in data terms) from its circular magazine location and loaded to the spindle location in the buffer magazine. 1: During operation, the tool remains in its circular magazine locations in the OPI modules. This applies to OPI modules magazine location data (T / TP, magazine data and location data) and tool data (T / TD, tool data, general data and T / TV, tool data, directory and T / AEV, working offsets, directory).					
-				UWord	r
Multi-line: yes	1				

nckLogbookSeekPos					
NCK logbook					
-				Long Integer	rw
Multi-line: no			1		

nckType					
NCK type 0: 840D pl 1000: FM-NC 2000: 810D pl 3000: 802S 4000: 802D pl 5000: 840Di pl (up to and including SW 6) 6000: SOLUTIONLINE 10700: 840D sl 14000: 802D sl T/M 14000: 802D sl N/G or C/U 15000: 840Di sl					
-				UWord	r
Multi-line: no					

nckVersion	$AN_NCK_VERSION				
NCK version Only the digits before the comma of the floating point number are evaluated, the digits after the comma may contain identifiers for development-internal intermediate releases. The digits before the comma includes the official NCK identifier for the software release: For software release 3.4 the value of the variable is 34,....					
-				Double	r
Multi-line: no					

ncuPerformanceClass					
NCU power class 0: No special power class 1: Powerline 2-n: Reserved					
-	0	0		UWord	r
Multi-line: yes	1		1		

numAnalogInp	MD 10300: FASTIO_ANA_NUM_INPUTS				A2
Number of HW analog inputs					
-				UWord	r
Multi-line: no					

numAnalogOutp	MD 10310: FASTIO_ANA_NUM_OUTPUTS				A2
Number of HW analog outputs					
-				UWord	r
Multi-line: no					

numBAGs					
Number of available mode groups					
-				UWord	r
Multi-line: no					

numBasisFrames	$MN_MM_NUM_GLOBAL_BASE_FRAMES				
Number of channel-independent basic frames					
-	0			UWord	r
Multi-line: yes	1		1		

numChannels					
Number of active channels					
-				UWord	r
Multi-line: no					

numContainer					
Number of currently available axis containers					
-		0	maxnumContainer	UWord	r
Multi-line: yes	1		1		

numContainerSlots					
Number of currently available slots per axis container					
-			maxnumContainerSlots	UWord	r
Multi-line: yes	Index of axis container		numContainer		

numCuttEdgeParams					
Number of P elements of a cutting edge					
-				UWord	r
Multi-line: no					

numCuttEdgeParams_tao	$MN_MM_NUM_CCS_TOA_PARAM				
Number of Siemens application cutting edge data in module TAO !! Reserved for SIEMENS applications !!					
-	0	0	10	UWord	r
Multi-line: yes	1		1		

numCuttEdgeParams_tas	$MN_MM_NUM_CCS_MON_PARAM				
Number of Siemens application monitoring data in module TAS !! Reserved for SIEMENS applications !!					
-	0	0	10	UWord	r
Multi-line: yes	1		1		

numCuttEdgeParams_ts					
Number of P elements of a cutting edge in module TS (tool monitoring data)					
-				UWord	r
Multi-line: no					

numCuttEdgeParams_tu	MD 18096: MM_CC_TOA_PARAM				
Number of P elements of a cutting edge in module TUE (cutting edge data for OEM)					
-				UWord	r
Multi-line: no					

numCuttEdgeParams_tus	$MN_MM_NUM_CC_MON_PARAM				
Number of parameters in the user monitoring data of a cutting edge in the module TUS					
-	0	0	10	UWord	r
Multi-line: yes	1		1		

numDigitInp	MD 10350: FASTIO_DIG_NUM_INPUTS				A2
Number of HW digital inputs					
-				UWord	r
Multi-line: no					

numDigitOutp	MD 10360: FASTIO_DIG_NUM_OUTPUTS				A2
Number of HW digital outputs					
-				UWord	r
Multi-line: no					

numDrives					
Reserved					
-				UWord	r
Multi-line: no					

numGCodeGroups					
Number of NC instruction groups					
-				UWord	r
Multi-line: no					

numGCodeGroupsFanuc					
Number of NC instruction groups in ISO Dialect mode (the number for the turning and milling versions is not the same)					
-				UWord	r
Multi-line: yes	1		1		

numGlobMachAxes					
Number of active machine axes					
-				UWord	r
Multi-line: no					

numGlobalGFrames	$MN_MM_NUM_GLOBAL_G_FRAMES				
Number of channel-independent G frames					
-	0			UWord	r
Multi-line: yes	1		1		

numHandWheels					
Maximum number of handwheels					
-				UWord	r
Multi-line: no					

numMagLocParams_tap	$MN_MM_NUM_CCS_MAGLOC_PARAM				
Number of Siemens application magazine location data in module TAP !! Reserved for SIEMENS applications !!					
-	0	0	10	UWord	r
Multi-line: yes	1		1		

numMagLocParams_u	$MN_MM_NUM_CC_MAGLOC_PARAM				
Number of parameters of the magazine user data for a tool magazine place in the module TUP					
-	0	0	10	UWord	r
Multi-line: yes	1		1		

numMagParams_tam	$MN_MM_NUM_CCS_MAGAZINE_PARAM				
Number of Siemens application magazine data in module TAM !! Reserved for SIEMENS applications !!					
-	0	0	10	UWord	r
Multi-line: yes	1		1		

numMagParams_u	$MN_MM_NUM_CC_MAGAZINE_PARAM				
Number of parameters of the magazine user data for a tool magazine in the TUM module					
-	0	0	10	UWord	r
Multi-line: yes	1		1		

numMagPlaceParams					
Number of parameters of a magazine location					
-				UWord	r
Multi-line: yes	1				

numMagPlacesMax	MD 18086: MM_NUM_MAGAZINE_LOCATION				FBW
Maximum number of magazine locations					
-				UWord	r
Multi-line: no					

numMagsMax	MD 18084: MM_NUM_MAGAZINE				FBW
Maximum number of magazines					
-				UWord	r
Multi-line: no					

numMultiToolParams	-				
Amount of multi-tool data in T/MTD module					
-				UWord	r
Multi-line: yes	1		1		

numMultiToolParams_mtad	$MN_MM_NUM_CCS_MULTITOOL_PARAM				
Amount of Siemens-specific multi-tool data in the MTAD module. Reserved for SIEMENS applications					
-				UWord	r
Multi-line: yes	1		1		

numMultiToolParams_mtud	$MN_MM_NUM_CC_MULTITOOL_PARAM				
Amount of OEM-specific multi-tool data in the MTUD module. Reserved for OEM applications					
-				UWord	r
Multi-line: yes	1		1		

numMultiToolPlaceParams	-				
Amount of multi-tool place data in the T/MTP module					
-				UWord	r
Multi-line: yes	1		1		

numMultiToolPlaceParams_mtap	$MN_MM_NUM_CCS_MTLOC_PARAM				
Amount of Siemens-specific multi-tool place data in the MTAP module. Reserved for SIEMENS applications					
-				UWord	r
Multi-line: yes	1		1		

numMultiToolPlaceParams_mtup	$MN_MM_NUM_CC_MTLOC_PARAM				
Amount of OEM-specific multi-tool place data in the MTUP module. Reserved for OEM applications					
-				UWord	r
Multi-line: yes	1		1		

numOfISOCorr					
Number of offset values in the ISO offset memory in ISO2 and ISO3 modes. This defines the upper limit of the option data $ON_NUM_CHANNELS.					
-	98			UWord	r
Multi-line: yes	1		1		

numParams_Adapt					
Number of parameters per adapter					
-	4			UWord	r
Multi-line: yes	1				

numParams_SC					
Number of total offset parameters per total offset set					
-	9			UWord	r
Multi-line: yes	1				

numPlaceMulti					FBW
Number of possible multiple assignments of a location to magazines					
-				UWord	r
Multi-line: no					

numPlaceMultiParams					FBW
Number of parameters of a multiple assignment					
-				UWord	r
Multi-line: no					

numSearchRunToolParams					
Number of parameters in the data actToolDataBeforeSearch in the area C, module S					
-	3	0		UWord	r
Multi-line: no			1		

numToBaust	MD 18110: MM_NUM_TOA_MODULES				
Number of T areas					
-				UWord	r
Multi-line: no					

numToolHolderParams					
Number of parameters in the data toolHolderData in the area C, module S Number of parameters in toolHolderData. If the "flat D number" is active, the value = 0 will be returned.					
-	5	0		UWord	r
Multi-line: no			1		

numToolParams_tad	$MN_MM_NUM_CCS_TDA_PARAM				
Number of Siemens application tool data in module TAD !! Reserved for SIEMENS applications !!					
-	0	0	10	UWord	r
Multi-line: yes	1		1		

numToolParams_tu	MD 18094: MM_CC_TDA_PARAM				
Number of P elements of a tool in module TU (tool data for OEM)					
-				UWord	r
Multi-line: no					

numUserFrames	$MN_MM_NUM_GLOBAL_USER_FRAMES				
Number of channel-independent user frames					
-	0			UWord	r
Multi-line: yes	1		1		

toolChangeMFunc	MD 22560: TOOL_CHANGE_M_CODE				W1
Number of M function for tool change 0 = change on T selection (standard for turning) 1 = change on selection M1.. 99999 = change on selection M99999 (standard for milling M06)					
-				Long Integer	r
Multi-line: no					

typeOfCuttingEdge					
Type of D-number programming see MD: MM_TYPE_OF_CUTTING_EDGE Value Meaning 0 no 'flat D number management' active 1 flat D numbers active					
-				UWord	r
Multi-line: yes	1		1		

userScale					
User unit table with 13 elements (see Start-up Guide 2.4 and machine data) 0 = table not active 1 = table active					
-				UWord	r
Multi-line: no			1		

3.2.2 Area C, Block Y : Channel-specific system data

OEM-MMC: Linkitem /ChannelConfiguration/...

The machine tool builder or user configures the control with the help of the machine data. Configuration can only be performed with certain access rights. The configuration of the NC can be read in the system data regardless of current access rights.

channelName	MD 20000: CHAN_NAME				K1
Channel name					
-				String [32]	r
Multi-line: no					

maskToolManagement	$MC_TOOL_MANAGEMENT_MASK	
Channel-specific settings for NCK tool management function Activation of TM memory by "0" means: The set tool management data do not use any memory space. Value=0: TM deactivated Bit 0=1: TM active: The tool management functions are enabled for the current channel. Bit 1=1: TM monitoring function active: Functions required to monitor tools (tool life and number of workpieces) are enabled. Bit 2=1: OEM functions active: The memory for user data can be utilized. Bit 3=1: Consideration of adjacent location active Bits 0 to 3 must be set identically to machine data MM_TOOL_MANAGEMENT_MASK (18080). Bit 4=1: The PLC has the possibility of issuing another request for tool change preparation with modified parameters. Part program is halted in response to T selection or M06 until it has been acknowledged by the PLC program. Bit 5=1: The main run/PLC synchronization in response to a tool change for the main spindle is executed simultaneously with the transport acknowledgement. Bit 6=1: The main run/PLC synchronization in response to a tool change for the auxiliary spindle is executed simultaneously with the transport acknowledgement. Bit 7=1: The main run/PLC synchronization in response to a tool change for the main spindle is not executed until the PLC acknowledgement confirms that the tool change is complete. Bit 8=1: The main run/PLC synchronization in response to a tool change for the auxiliary spindle is not executed until the PLC acknowledgement confirms that the tool change is complete.. Bit 9: Reserved Bit 10=1: M06 is delayed until the preparation acknowledgement has been output by the PLC. The change signal (e.g. M06) is not output until the tool selection (DBX [n+0].2) has been acknowledged. The part program is halted in response to M06 until the T selection has been acknowledged. Bit 11=1: The preparation command is output even if a preparation command has already been output for the same tool. This setting is useful, for example, if the chain is to be positioned when "Tx" is first called and if the second call is to initiate a check as to whether the tool is in the correct location for a tool change (e.g. in front of tool-change station). Bit 12=1: The preparation command is executed even if the tool is already loaded in the spindle, i.e. the T selection signal (DB72.DBXn. 2) is set even if it has already been set for the same tool. (Tx...Tx) Bit 13=1: Only on systems with sufficient memory space (NCU572, NCU573): Recording of tool sequences in a diagnostics buffer. The commands are fetched from the diagnostics buffer in response to Reset and stored in a file in the passive file system, NCATR xx.MPF under part program. The trace file is useful for the Hotline in the event of errors and is not described in detail here. Bit 14=1: Automatic tool change in response to Reset and Start according to machine data MD20120 TOOL_RESET_NAME MD20110 RESET_MODE_MASK MD20124 TOOL_MANAGEMENT_TOOLHOLDER. If machine data RESET_MODE_MASK is in use, then this bit		

maskToolManagement	$MC_TOOL_MANAGEMENT_MASK				
must be set as well. If RESET_MODE_MASK is set such that the tool stored in TOOL_RESET_NAME must be loaded in response to RESET, then the select and change command is output to the user interface (DB 72) in response to RESET or Start. If machine data RESET_MODE_MASK is set such that the active tool must remain active after M30 or RESET and if the active tool is disabled in the spindle (by user), then a change command for a replacement tool is output to the user interface in response to RESET. If no replacement tool is available, then an error message is output. Bit 15=1: No return transport of tool when several preparation commands are output. (Tx->Tx) Bit 16=1: T location number is active Bit 17=1: Tool life decrementation can be started/stopped via the PLC.					
-	0			Long Integer	r
Multi-line: yes	1				

mmcCmd					
Command from NCK to HMI The string is made up of the following characters: 1st Character acknowledgement mode: "N" no acknowledgement "S" synchronous acknowledgement "A" asynchronous acknowledgement 2. - 6th character: five-digit sequence number in ASCII that is generated by the NCK 7. - 207th character: Command string which ends with "\0"					
-				String [206]	r
Multi-line: no					

mmcCmdPrep					
Command from the NCK-preparation task to the HMI (e.g. for calling external subprograms)					
-				String [206]	r
Multi-line: yes	1		1		

mmcCmdQuit					
Acknowledgement from HMI for command from NCK to HMI The string is made up of the following characters: 1st Character acknowledgement code: "P" programmed "B" busy "F" failed "E" executed 2. - 6th character: five-digit sequence number in ASCII for acknowledgement code "B", "F" or "E", generated by NCK 7. - 201th character: additional communication-specific information for acknowledgement code "B", "F" or "E", ends with "\0"					
-				String [200]	w
Multi-line: no					

mmcCmdQuitPrep					
Acknowledgemnt by HMI for an NCK-preparation command to the HMI (e.g. for calling external subprograms)					
-				String [200]	rw
Multi-line: yes	1		1		

numActAxes					
Number of active tools in channel. Channel axis gaps are not included in count which means that value might be lower than numMachAxes. The following applies: numMachAxes >= numGeoAxes + numAuxAxes numActAxes = numGeoAxes + numAuxAxes					
-	0	0	numMachAxes	UWord	r
Multi-line: yes	1		1		

numAuxAxes					
Number of auxiliary axes					
-				UWord	r
Multi-line: no					

numBasisFrames	$MC_MM_NUM_BASE_FRAMES				
Number of basic frames in channel					
-	0			UWord	r
Multi-line: yes	1		1		

numContourInProtArea					
Maximum number of polygon elements per protection zone					
-				UWord	r
Multi-line: no					

numGFrames	MD 28080: MM_NUM_G_FRAMES				
Number of G frames in this channel					
-				UWord	r
Multi-line: no					

numGeoAxes					
Number of geometry axes and orientation axes					
-				UWord	r
Multi-line: no					

numMachAxes					
No. of highest channel axis. This also corresponds to the number of axes in the channel provided there are no gaps in the axis sequence.					
-	0	1		UWord	r
Multi-line: yes	1		1		

numOriAxes					
Number of orientation axes in channel					
-	0			UWord	r
Multi-line: yes	1		1		

numProtArea	MD 28200: MM_NUM_PROTECT_AREA_CHAN				S7
Maximum number of protection zones					
-				UWord	r
Multi-line: no					

numRParams	MD 28050: MM_NUM_R_PARAM				S7
Number of channel-specific R variables					
-				UWord	r
Multi-line: no					

numSpindles					
Number of spindles					
-				UWord	r
Multi-line: no					

numSpindlesLog					
Number of logical spindles. Specifies the number of lines in module SSP2.					
-				UWord	r
Multi-line: no			1		

numToolEdges	MD 18100: MM_NUM_CUTTING_EDGES_IN_TOA				S7
Number of tool edges in this channel					
-				UWord	r
Multi-line: no					

numUserFrames	MD 28080: MM_NUM_USER_FRAMES				S7
Number of user frames in this channel					
-				UWord	r
Multi-line: no					

oemProtText					
OEM text to be entered next in the logging buffer.					
-				String [128]	r
Multi-line: yes	1		1		

progProtText					
Programmable text to be entered next in the logging buffer					
-				String [128]	r
Multi-line: yes	1		1		

punchNibActivation	MD 26012: PUNCHNIB_ACTIVATION				N4
Activation of punching and nibbling functions 0 = option not available 1 = option available					
-				UWord	r
Multi-line: no			1		

stringIsFileId		

As soon as PI _N_STRGIS is executed, the result from interpretation of the transferred string is stored as file ID in this variable. moduleId may be used instead of fileId.
Also refer to NC command STRINGIS.

Meaning Name of definition file in NCK (explanation) OPI area OPI modules (domain name)

0 String is unknown
1 GCODE (G code - Siemens and/or ISO)
2 NCADDRES (NCK NC address character)
3 NCADDRES_CHAN (channel NC address character)
4 NCNAM (named NCK NC address)
5 NCNAM_CHAN (named channel NC address)
6 FRAME (frame variable)
7 TOOLCORR (tool parameter) TO (=4)
8 MACHDAT_NCK (machine data) NCK (=0) M (= 0x1A)
9 MACHDAT_CHAN (machine data) CHAN (=2) M (=-0x1A)
10 MACHDAT_AXIS (machine data) AXIS (=3) M (=-0x1A)
11 R_PARAM (R variable) CHAN (=2) RP (= 0x15)
12 AC_MARKER (synchronized action marker)
13 AC_PARAM (synchronized action parameter)
14 PRED_FUNC (NC language function)
15 SYSDAT_NCK (status variable)
16 SYSDAT_CHAN (status variable)
17 SYSDAT_AXIS (status variable)
18 USER_NCK _N_SGUD_DEF NCK (=0) GD1 (= 0x36)
19 USER_CHAN _N_SGUD_DEF CHAN (=2) GD1 (= 0x36)
20 USER_AXIS _N_SGUD_DEF AXIS (=3) GD1 (= 0x36)
21 USERMACRO _N_SMAC_DEF
_N_MMAC_DEF
_N_UMAC_DEF
22 EEC (leadscrew parameter)
23 QEC (quadrant error parameter)
24 CEC (cross error compensation parameter)
25 TOOLMAGAZINE (magazine parameter) TO (=4)
26 PROTAREA (protection area parameter)
27 PROTAREA_CHAN (protection area parameter)
28 USER_NCK2 _N_MGUD_DEF NCK (=0) GD2 (= 0x2D)
29 USER_NCK3 _N_UGUD_DEF NCK (=0) GD3 (= 0x2E)
30 USER_NCK4 _N_GUD4_DEF NCK (=0) GD4 (= 0x2F)
31 USER_NCK5 _N_GUD5_DEF NCK (=0) GD5 (= 0x30)
32 USER_NCK5 _N_GUD6_DEF NCK (=0) GD6 (= 0x31)
33 USER_NCK5 _N_GUD7_DEF NCK (=0) GD7 (= 0x32)
34 USER_NCK5 _N_GUD8_DEF NCK (=0) GD8 (= 0x33)
35 USER_NCK5 _N_GUD9_DEF NCK (=0) GD9 (= 0x34)
36 USER_CHAN2 _N_MGUD_DEF CHAN (=2) GD2 (= 0x2D)
37 USER_CHAN3 _N_UGUD_DEF CHAN (=2) GD3 (= 0x2E)
38 USER_CHAN4 _N_GUD4_DEF CHAN (=2) GD4 (= 0x2F)
39 USER_CHAN5 _N_GUD5_DEF CHAN (=2) GD5 (= 0x30)
40 USER_CHAN6 _N_GUD6_DEF CHAN (=2) GD6 (= 0x31)
41 USER_CHAN7 _N_GUD7_DEF CHAN (=2) GD7 (= 0x32)
42 USER_CHAN8 _N_GUD8_DEF CHAN (=2) GD8 (= 0x33)

stringIsFileId					
43 USER_CHAN9 _N_GUD9_DEF CHAN (=2) GD9 (= 0x34) 44 reserved 45 reserved 46 reserved 47 reserved 48 reserved 49 reserved 50 reserved 51 reserved 52 TOOLCARRIER (orientable tool carrier parameter TO (=4) 53 GCODESEXT (G code) NCK (=0) (G codes FANUC) 54 FRAME_NCK (frame) NCK (=0) 55 CYC_PARAM_CHAN (global cycle transfer parameter)r 56 TOOLENVMOD (tool environment parameter) 57 SYNAGUD_CHAN (synchronized action capable GUD CHAN (=2) GD1 (= 0x36) 58 SYNAGUD_CHAN2 (synchronized action capable GUD CHAN (=2) GD2 (= 0x2D) 59 SYNAGUD_CHAN3 (synchronized action capable GUD CHAN (=2) GD3 (= 0x2E) 60 SYNAGUD_CHAN4 (synchronized action capable GUD CHAN (=2) GD4 (= 0x2F) 61 SYNAGUD_CHAN5 (synchronized action capable GUD CHAN (=2) GD5 (= 0x30) 62 SYNAGUD_CHAN6 (synchronized action capable GUD CHAN (=2) GD6 (= 0x31) 63 SYNAGUD_CHAN7 (synchronized action capable GUD CHAN (=2) GD7 (= 0x32) 64 SYNAGUD_CHAN8 (synchronized action capable GUD CHAN (=2) GD8 (= 0x33) 65 SYNAGUD_CHAN9 (synchronized action capable GUD CHAN (=2) GD9 (= 0x34) 66 NKIN (kinematic chain parameter) 67 NPA (3D protection area parameter) 68 WAL_CS (work area in specific coordinate system) 69 TOOLISO22CORR (tool correction parameter with ISO2.2) 70 TOOLISO32CORR (tool correction parameter with ISO3.2) 71 EPS_PARAM (ePS services parameters (reserved for ePS use only !!) >= 200 LUD (LUD / PUD - program local variable) Note: via OPI normally only a subset of the NCK data blocks is made known. Note: missing column entries means that no definition file has been defined for this line, or that in OPI no domain name has been defined. Various NCK data may be accessed through OPI variable blocks instead of through domain names. Example: tool data, frame data, etc. It is possible that several OPI variable blocks exist for a value of stringIsFileId.					
-				UWord	r
Multi-line: no					

stringIsMeaning					
As soon as PI _N_STRGIS is executed, the result from interpretation of the transferred string is stored as a code in this variable. For example, the code for $P_TOOL is value 207. Also refer to NC command STRINGIS. 000 = itemName string is unknown in the NCK 100 = itemName string is a language construct, but is not programmable (option/function is not active) 2xx = itemName string is a permissible language construct (option/function is active) 2xx = is defined by: 200 = no interpretation possible 201 = DIN address / NC address (e.g. MEAS) 202 = G code (e.g. G04, INVCW) 203 = NC language function (= command with return value, parameter Passing) (e.g. GETMDACT) 204 = NC language procedure (= command without return value, with parameter Passing) (e.g. SBLOF) 205 = NC key word (e.g. DEFINE) 206 = machine/setting/option data (= parameter starting with $M / $S / $O) 207 = NC system parameter (= parameter starting with R and $) 208 = cycle name (name created by cycle) 209 = GUD variable (name created by GUD definitions 210 = macro name (name created by macro definition file) 211 = LUD variable (name created by active program) 212 = no Siemens G code, but an ISO G code 400 = NC address, which is not xx=01 or xx=10, and which is also not: G or R (e.g. T, D, F, H, L, M)					
-	0	0	4000	UWord	r
Multi-line: no					

stringIsSymbolId					
As soon as PI _N_STRGIS is executed, the result from interpretation of the transferred string is stored as symbol ID in this variable. The symbol ID is found in the NCK module, which is specified in variable stringIsFileId. This value can be found also in the corresponding ACC and ACX file.					
-				UWord	r
Multi-line: no					

systemFrameMask	$MC_MM_SYSTEM_FRAME_MASK				
Configuring screenform for channel-specific system frames Indicates in bit-coded form which system frames are available					
-	0	0		UWord	r
Multi-line: yes	1		1		

toNo	MD 28085: MM_LINK_TOA_UNIT				W1
Number of T area that is assigned to the channel					
-				UWord	r
Multi-line: no					

toolDataChangeBufferSize	$MC_MM_TOOL_DATA_CHANGE_BUFFER_SIZE				
Size of the effective ring buffer for the tool data modifications in the OPI block TDC (0x56). This value is the maximum column number in the OPI block TDC. If a number of channels works with a TO unit, the setting with the lowest channel number applies. The value = 0 is returned if the the ring buffer is not active ($MN_TOOL_DATA_CHANGE_COUNTER, Bit2=0 and Bit3=0).					
-	0	0		UWord	r
Multi-line: yes	1		1		

3.2.3 Area N, Block PA : Global protection zones

OEM-MMC: Linkitem /NckProtectedArea/...

Up to 10 protection zones can be defined. Each protection zone is described by a polygon function consisting of up to 10 elements. The module PA contains the individual coordinates of the polygon elements. The protection zones are addressed via the variable indices. The physical unit of the parameters can be read from the variable "basicLengthUnit" in the module Y in area N.

The classification as NCK or channel-specific protection zones does not affect the protection zone monitoring function, but indicates the area in which the protection zone is registered.

MDD_PA_CENT_ABS_0	$SN_PA_CENT_ABS[x,0] x = Number protection zone				A3
Absolute abscissa value of arc centre of 1st contour element					
mm, inch, user defined				Double	r
Multi-line: yes	Number of protection zone		numProtArea		

MDD_PA_CENT_ABS_1	$SN_PA_CENT_ABS[x,1] x = Number protection zone				A3
Absolute abscissa value of arc centre of 2nd contour element					
mm, inch, user defined				Double	r
Multi-line: yes	Number of protection zone		numProtArea		

MDD_PA_CENT_ABS_2	$SN_PA_CENT_ABS[x,2] x = Number protection zone				A3
Absolute abscissa value of arc centre of 3rd contour element					
mm, inch, user defined				Double	r
Multi-line: yes	Number of protection zone		numProtArea		

MDD_PA_CENT_ABS_3	$SN_PA_CENT_ABS[x,3] x = Number protection zone				A3
Absolute abscissa value of arc centre of 4th contour element					
mm, inch, user defined				Double	r
Multi-line: yes	Number of protection zone		numProtArea		

MDD_PA_CENT_ABS_4	$SN_PA_CENT_ABS[x,4] x = Number protection zone				A3
Absolute abscissa value of arc centre of 5th contour element					
mm, inch, user defined				Double	r
Multi-line: yes	Number of protection zone		numProtArea		

MDD_PA_CENT_ABS_5	$SN_PA_CENT_ABS[x,5] x = Number protection zone				A3
Absolute abscissa value of arc centre of 6th contour element					
mm, inch, user defined				Double	r
Multi-line: yes	Number of protection zone		numProtArea		

MDD_PA_CENT_ABS_6	$SN_PA_CENT_ABS[x,6] x = Number protection zone				A3
Absolute abscissa value of arc centre of 7th contour element					
mm, inch, user defined				Double	r
Multi-line: yes	Number of protection zone		numProtArea		

MDD_PA_CENT_ABS_7	$SN_PA_CENT_ABS[x,7] x = Number protection zone				A3
Absolute abscissa value of arc centre of 8th contour element					
mm, inch, user defined				Double	r
Multi-line: yes	Number of protection zone		numProtArea		

MDD_PA_CENT_ABS_8	$SN_PA_CENT_ABS[x,8] x = Number protection zone				A3
Absolute abscissa value of arc centre of 9th contour element					
mm, inch, user defined				Double	r
Multi-line: yes	Number of protection zone		numProtArea		

MDD_PA_CENT_ABS_9	$SN_PA_CENT_ABS[x,9] x = Number protection zone				A3
Absolute abscissa value of arc centre of 10th contour element					
mm, inch, user defined				Double	r
Multi-line: yes	Number of protection zone		numProtArea		

MDD_PA_CENT_ORD_0	$SN_PA_CENT_ORD[x,0] x = Number protection zone				A3
Absolute ordinate value of arc centre of 1st contour element					
mm, inch, user defined				Double	r
Multi-line: yes	Number of protection zone		numProtArea		

MDD_PA_CENT_ORD_1	$SN_PA_CENT_ORD[x,1] x = Number protection zone				A3
Absolute ordinate value of arc centre of 2nd contour element					
mm, inch, user defined				Double	r
Multi-line: yes	Number of protection zone		numProtArea		

MDD_PA_CENT_ORD_2	$SN_PA_CENT_ORD[x,2] x = Number protection zone				A3
Absolute ordinate value of arc centre of 3rd contour element					
mm, inch, user defined				Double	r
Multi-line: yes	Number of protection zone		numProtArea		

MDD_PA_CENT_ORD_3	$SN_PA_CENT_ORD[x,3] x = Number protection zone				A3
Absolute ordinate value of arc centre of 4th contour element					
mm, inch, user defined				Double	r
Multi-line: yes	Number of protection zone		numProtArea		

MDD_PA_CENT_ORD_4	$SN_PA_CENT_ORD[x,4] x = Number protection zone				A3
Absolute ordinate value of arc centre of 5th contour element					
mm, inch, user defined				Double	r
Multi-line: yes	Number of protection zone		numProtArea		

MDD_PA_CENT_ORD_5	$SN_PA_CENT_ORD[x,5] x = Number protection zone				A3
Absolute ordinate value of arc centre of 6th contour element					
mm, inch, user defined				Double	r
Multi-line: yes	Number of protection zone		numProtArea		

MDD_PA_CENT_ORD_6	$SN_PA_CENT_ORD[x,6] x = Number protection zone				A3
Absolute ordinate value of arc centre of 7th contour element					
mm, inch, user defined				Double	r
Multi-line: yes	Number of protection zone		numProtArea		

MDD_PA_CENT_ORD_7	$SN_PA_CENT_ORD[x,7] x = Number protection zone				A3
Absolute ordinate value of arc centre of 8th contour element					
mm, inch, user defined				Double	r
Multi-line: yes	Number of protection zone		numProtArea		

MDD_PA_CENT_ORD_8	$SN_PA_CENT_ORD[x,8] x = Number protection zone				A3
Absolute ordinate value of arc centre of 9th contour element					
mm, inch, user defined				Double	r
Multi-line: yes	Number of protection zone		numProtArea		

MDD_PA_CENT_ORD_9	$SN_PA_CENT_ORD[x,9] x = Number protection zone				A3
Absolute ordinate value of arc centre of 10th contour element					
mm, inch, user defined				Double	r
Multi-line: yes	Number of protection zone		numProtArea		

MDD_PA_CONT_ABS_0	$SN_PA_CONT_ABS[x,0] x = Number protection zone				A3
Absolute abscissa value of end point of 1st contour element					
mm, inch, user defined				Double	r
Multi-line: yes	Number of protection zone		numProtArea		

MDD_PA_CONT_ABS_1	$SN_PA_CONT_ABS[x,1] x = Number protection zone				A3
Absolute abscissa value of end point of 2nd contour element					
mm, inch, user defined				Double	r
Multi-line: yes	Number of protection zone		numProtArea		

MDD_PA_CONT_ABS_2	$SN_PA_CONT_ABS[x,2] x = Number protection zone				A3
Absolute abscissa value of end point of 3rd contour element					
mm, inch, user defined				Double	r
Multi-line: yes	Number of protection zone		numProtArea		

MDD_PA_CONT_ABS_3	$SN_PA_CONT_ABS[x,3] x = Number protection zone				A3
Absolute abscissa value of end point of 4th contour element					
mm, inch, user defined				Double	r
Multi-line: yes	Number of protection zone		numProtArea		

MDD_PA_CONT_ABS_4	$SN_PA_CONT_ABS[x,4] x = Number protection zone				A3
Absolute abscissa value of end point of 5th contour element					
mm, inch, user defined				Double	r
Multi-line: yes	Number of protection zone		numProtArea		

MDD_PA_CONT_ABS_5	$SN_PA_CONT_ABS[x,5] x = Number protection zone				A3
Absolute abscissa value of end point of 6th contour element					
mm, inch, user defined				Double	r
Multi-line: yes	Number of protection zone		numProtArea		

MDD_PA_CONT_ABS_6	$SN_PA_CONT_ABS[x,6] x = Number protection zone				A3
Absolute abscissa value of end point of 7th contour element					
mm, inch, user defined				Double	r
Multi-line: yes	Number of protection zone		numProtArea		

MDD_PA_CONT_ABS_7	$SN_PA_CONT_ABS[x,7] x = Number protection zone				A3
Absolute abscissa value of end point of 8th contour element					
mm, inch, user defined				Double	r
Multi-line: yes	Number of protection zone		numProtArea		

MDD_PA_CONT_ABS_8	$SN_PA_CONT_ABS[x,8] x = Number protection zone				A3
Absolute abscissa value of end point of 9th contour element					
mm, inch, user defined				Double	r
Multi-line: yes	Number of protection zone		numProtArea		

MDD_PA_CONT_ABS_9	$SN_PA_CONT_ABS[x,9] x = Number protection zone				A3
Absolute abscissa value of end point of 10th contour element					
mm, inch, user defined				Double	r
Multi-line: yes	Number of protection zone		numProtArea		

MDD_PA_CONT_ORD_0	$SN_PA_CONT_ORD[x,0] x = Number protection zone				A3
Absolute ordinate value of end point of 1st contour element					
mm, inch, user defined				Double	r
Multi-line: yes	Number of protection zone		numProtArea		

MDD_PA_CONT_ORD_1	$SN_PA_CONT_ORD[x,1] x = Number protection zone				A3
Absolute ordinate value of end point of 2nd contour element					
mm, inch, user defined				Double	r
Multi-line: yes	Number of protection zone		numProtArea		

MDD_PA_CONT_ORD_2	$SN_PA_CONT_ORD[x,2] x = Number protection zone				A3
Absolute ordinate value of end point of 3rd contour element					
mm, inch, user defined				Double	r
Multi-line: yes	Number of protection zone		numProtArea		

MDD_PA_CONT_ORD_3	$SN_PA_CONT_ORD[x,3] x = Number protection zone				A3
Absolute ordinate value of end point of 4th contour element					
mm, inch, user defined				Double	r
Multi-line: yes	Number of protection zone		numProtArea		

MDD_PA_CONT_ORD_4	$SN_PA_CONT_ORD[x,4] x = Number protection zone				A3
Absolute ordinate value of end point of 5th contour element					
mm, inch, user defined				Double	r
Multi-line: yes	Number of protection zone		numProtArea		

MDD_PA_CONT_ORD_5	$SN_PA_CONT_ORD[x,5] x = Number protection zone				A3
Absolute ordinate value of end point of 6th contour element					
mm, inch, user defined				Double	r
Multi-line: yes	Number of protection zone		numProtArea		

MDD_PA_CONT_ORD_6	$SN_PA_CONT_ORD[x,6] x = Number protection zone				A3
Absolute ordinate value of end point of 7th contour element					
mm, inch, user defined				Double	r
Multi-line: yes	Number of protection zone		numProtArea		

MDD_PA_CONT_ORD_7	$SN_PA_CONT_ORD[x,7] x = Number protection zone				A3
Absolute ordinate value of end point of 8th contour element					
mm, inch, user defined				Double	r
Multi-line: yes	Number of protection zone		numProtArea		

MDD_PA_CONT_ORD_8	$SN_PA_CONT_ORD[x,8] x = Number protection zone				A3
Absolute ordinate value of end point of 9th contour element					
mm, inch, user defined				Double	r
Multi-line: yes	Number of protection zone		numProtArea		

MDD_PA_CONT_ORD_9	$SN_PA_CONT_ORD[x,9] x = Number protection zone				A3
Absolute ordinate value of end point of 10th contour element					
mm, inch, user defined				Double	r
Multi-line: yes	Number of protection zone		numProtArea		

MDD_PA_MINUS_LIM	$SN_PA_MINUS_LIM[x] x = Number protection zone				A3
Limitation in the minus direction of the protection zone in the axis that is perpendicular to the polygon definition (applicate)					
mm, inch, user defined				Double	r
Multi-line: yes	Number of protection zone		numProtArea		

MDD_PA_PLUS_LIM	$SN_PA_PLUS_LIM[x] x = Number protection zone				A3
Limitation in the plus direction of the protection zone in the axis that is perpendicular to the polygon definition (applicate)					
mm, inch, user defined				Double	r
Multi-line: yes	Number of protection zone		numProtArea		

MDU_PA_ACTIV_IMMED	$SN_PA_ACTIV_IMMED[x] x = Number protection zone				A3
Code for "active immediately after referencing", i.e. the protection zone is active as soon as the control has been started up and the axes have been referenced 0 = protection zone is not active immediately 1 = protection zone is active immediately					
-				UWord	r
Multi-line: yes	Number of protection zone		numProtArea		

MDU_PA_CONT_NUM	$SN_PA_CONT_NUM[x] x = Number protection zone				A3
Number of valid contour elements					
-		0	numContourInPr otArea	UWord	r
Multi-line: yes	Number of protection zone		numProtArea		

MDU_PA_CONT_TYP_0	$SN_PA_CONT_TYP[x,0] x = Number protection zone				A3
Contour type of 1st contour element 0 = G1 1 = G2 2 = G3					
-				UWord	r
Multi-line: yes	Number of protection zone		numProtArea		

MDU_PA_CONT_TYP_1	$SN_PA_CONT_TYP[x,1] x = Number protection zone				A3
Contour type of 2nd contour element 0 = G1 1 = G2 2 = G3					
-				UWord	r
Multi-line: yes	Number of protection zone		numProtArea		

MDU_PA_CONT_TYP_2	$SN_PA_CONT_TYP[x,2] x = Number protection zone				A3
Contour type of 3rd contour element 0 = G1 1 = G2 2 = G3					
-				UWord	r
Multi-line: yes	Number of protection zone		numProtArea		

MDU_PA_CONT_TYP_3	$SN_PA_CONT_TYP[x,3] x = Number protection zone				A3
Contour type of 4th contour element 0 = G1 1 = G2 2 = G3					
-				UWord	r
Multi-line: yes	Number of protection zone		numProtArea		

MDU_PA_CONT_TYP_4	$SN_PA_CONT_TYP[x,4] x = Number protection zone				A3
Contour type of 5th contour element 0 = G1 1 = G2 2 = G3					
-				UWord	r
Multi-line: yes	Number of protection zone		numProtArea		

MDU_PA_CONT_TYP_5	$SN_PA_CONT_TYP[x,5] x = Number protection zone				A3
Contour type of 6th contour element 0 = G1 1 = G2 2 = G3					
-				UWord	r
Multi-line: yes	Number of protection zone		numProtArea		

MDU_PA_CONT_TYP_6	$SN_PA_CONT_TYP[x,6] x = Number protection zone				A3
Contour type of 7th contour element 0 = G1 1 = G2 2 = G3					
-				UWord	r
Multi-line: yes	Number of protection zone		numProtArea		

MDU_PA_CONT_TYP_7	$SN_PA_CONT_TYP[x,7] x = Number protection zone				A3
Contour type of 8th contour element 0 = G1 1 = G2 2 = G3					
-				UWord	r
Multi-line: yes	Number of protection zone		numProtArea		

MDU_PA_CONT_TYP_8	$SN_PA_CONT_TYP[x,8] x = Number protection zone				A3
Contour type of 9th contour element 0 = G1 1 = G2 2 = G3					
-				UWord	r
Multi-line: yes	Number of protection zone		numProtArea		

MDU_PA_CONT_TYP_9	$SN_PA_CONT_TYP[x,9] x = Number protection zone				A3
Contour type of 10th contour element 0 = G1 1 = G2 2 = G3					
-				UWord	r
Multi-line: yes	Number of protection zone		numProtArea		

MDU_PA_LIM_3DIM	$SN_PA_LIM_3DIM[x] x = Number protection zone				A3
Code for limitation of protection zone in the axis that is perpendicular to polygon definition (applicate) 0 = no limitation 1 = limitation in positive direction 2 = limitation in negative direction 3 = limitation in both directions					
-				UWord	r
Multi-line: yes	Number of protection zone		numProtArea		

MDU_PA_ORI	$SN_PA_ORI[x] x = Number protection zone				A3
Code for plane assignment of protection zone 0 = G17 1 = G18 2 = G19					
-				UWord	r
Multi-line: yes	Number of protection zone		numProtArea		

MDU_PA_TW	$SN_PA_T_W[x] x = Number protection zone				A3
Code for workpiece or tool-oriented protection zone 0 = workpiece-related 1 = reserved 2 = reserved 3 = tool-related					
-				UWord	r
Multi-line: yes	Number of protection zone		numProtArea		

kinElemAxis	$NK_AXIS				
Machine axis or OEM object name					
-				String [32]	rw
Multi-line: yes	Number of the chain element		$MN_MM_MAXNUM_KIN_CHAIN_ELEM		

kinElemAxisOffset	$NK_A_OFF				
Axis offset					
mm, inch, degree, user defined				Double	rw
Multi-line: yes	Number of the chain element		$MN_MM_MAXNUM_KIN_CHAIN_ELEM		

kinElemName	$NK_NAME				
Name of a kinematic element					
-				String [32]	rw
Multi-line: yes	Number of the chain element		$MN_MM_MAXNUM_KIN_CHAIN_ELEM		

kinElemNameOld					
Name of a kinematic element (old). Using this variable, you can access the same data as with kinElemName (column index 1030). On the one hand, the variable is necessary due to an address shift compared with software version 83, and, on the other hand, to lock the (old) column index 1040 for other applications.					
-				String [32]	rw
Multi-line: yes	Number of the chain element		$MN_MM_MAXNUM_KIN_CHAIN_ELEM		

kinElemNext	$NK_NEXT				
Reference to the next kinematic element					
-				String [32]	rw
Multi-line: yes	Number of the chain element		$MN_MM_MAXNUM_KIN_CHAIN_ELEM		

kinElemNextOld					
Reference to next kinematic element (old). Using this variable, you can access the same data as with kinElemNameNext (column index 1032). On the one hand, the variable is necessary due to an address shift compared with software version 83, and, on the other hand, to lock the (old) column index 1041 for other applications.					
-				String [32]	rw
Multi-line: yes	Number of the chain element		$MN_MM_MAXNUM_KIN_CHAIN_ELEM		

kinElemOffDir0	$NK_OFF_DIR[0]				
Offset or direction component in the X direction					
mm, inch, user defined				Double	rw
Multi-line: yes	Number of the chain element		$MN_MM_MAXNUM_KIN_CHAIN_ELEM		

kinElemOffDir1	$NK_OFF_DIR[1]				
Offset or direction component in the Y direction					
mm, inch, user defined				Double	rw
Multi-line: yes	Number of the chain element		$MN_MM_MAXNUM_KIN_CHAIN_ELEM		

kinElemOffDir2	$NK_OFF_DIR[2]				
Offset or direction component in the Z direction					
mm, inch, user defined				Double	rw
Multi-line: yes	Number of the chain element		$MN_MM_MAXNUM_KIN_CHAIN_ELEM		

kinElemParallel	$NK_PARALLEL				
Reference to 1st element of a diverging chain					
-				String [32]	rw
Multi-line: yes	Number of the chain element		$MN_MM_MAXNUM_KIN_CHAIN_ELEM		

kinElemSwitchIndex	$NK_SWITCH_INDEX				
Index of a switch in the kinematic chain.					
-				Long Integer	rw
Multi-line: yes	Number of the chain element		$MN_MM_MAXNUM_KIN_CHAIN_ELEM		

kinElemSwitchPos	$NK_SWITCH_POS				
Position of a switch in a kinematic chain.					
-				UDoubleword	rw
Multi-line: yes	Number of the chain element		$MN_MM_MAXNUM_KIN_CHAIN_ELEM		

kinElemType	$NK_TYPE				
Type of kinematic element					
-				String [32]	rw
Multi-line: yes	Number of the chain element		$MN_MM_MAXNUM_KIN_CHAIN_ELEM		

kinSwitch	$NK_SWITCH				
Position of a switch in a kinematic chain.					
-				Long Integer	rw
Multi-line: yes	Number of the switch		$MN_MM_MAXNUM_KIN_SWITCHES		

modelChangeCounter					
Modification counter of the machine model					
-				UWord	r
Multi-line: yes	1: Kinematic modification counter 2: Activation status modification counter 3: Protection area geometry modification counter 4: Create/delete protection areas modification counter 5: Busy: model is being modified 6: In the case of an alarm during model preparation: alarm number 7: In the case of an alarm during model preparation: type of the erroneous element (0 = unknown, 1 = kinematic element, 2 = protection area, 3 = protection area element, 4 = collision pair) 8: In the case of an alarm during model preparation: index of the erroneous element (beginning with 1)		8		

pa3D1stProt	$NP_1ST_PROT				
Name of the first element of the protection zone					
-				String [32]	rw
Multi-line: yes	Number of the protection zone		MN_MM_MAXNUM_3D_PROT_AREAS		

pa3DAuxIndex0	$NP_INDEX[0]				
1st index for definition of variable protection zones					
-				UDoubleword	rw
Multi-line: yes	Number of the protection zone		MN_MM_MAXNUM_3D_PROT_AREAS		

pa3DAuxIndex1	$NP_INDEX[1]				
2nd index for definition of variable protection zones					
-				UDoubleword	rw
Multi-line: yes	Number of the protection zone		MN_MM_MAXNUM_3D_PROT_AREAS		

pa3DAuxIndex2	$NP_INDEX[2]				
3rd index for definition of variable protection zones					
-				UDoubleword	rw
Multi-line: yes	Number of the protection zone		MN_MM_MAXNUM_3D_PROT_AREAS		

pa3DBitIndex	$NP_BIT_NO				
Index of the bits assigned on the VDI interface					
-				UDoubleword	rw
Multi-line: yes	Number of the protection zone		MN_MM_MAXNUM_3D_PROT_AREAS		

pa3DChainElem	$NP_CHAIN_ELEM				
Name of the kinematic element carrying the protection zone					
-				String [32]	rw
Multi-line: yes	Number of the protection zone		MN_MM_MAXNUM_3D_PROT_AREAS		

pa3DCollPair0	$NP_COLL_PAIR[n, 0]				
1. Name of a protection area of a collision pair					
-				String [32]	rw
Multi-line: yes	Number of a collision pair		MM_MAXNUM_3D_PROT_AREAS * (MM_MAXNUM_3D_PROT_AREAS - 1) / 2		

pa3DCollPair1	$NP_COLL_PAIR[n, 1]				
2. Name of a protection area of a collision pair					
-				String [32]	rw
Multi-line: yes	Number of a collision pair		MM_MAXNUM_3D_PROT_AREAS * (MM_MAXNUM_3D_PROT_AREAS - 1) / 2		

pa3DCollPairSafetyDist	$NP_SAFETY_DIST				
Safety distance of a collision pair					
mm, inch, user defined				Double	rw
Multi-line: yes	Number of a collision pair		MM_MAXNUM_3D_PROT_AREAS * (MM_MAXNUM_3D_PROT_AREAS - 1) / 2		

pa3DElemAdd	$NP_ADD				
Name of a protection zone to be inserted					
-				String [32]	rw
Multi-line: yes	Number of the protection zone element		$MN_MM_MAXNUM_3D_PROT_AREA_ELEM		

pa3DElemAngle	$NP_ANG				
Turning angle					
-				Double	rw
Multi-line: yes	Number of the protection zone element		$MN_MM_MAXNUM_3D_PROT_AREA_ELEM		

pa3DElemColor	$NP_COLOR				
Color and transparency of a protection area element in QT format (HighByte: Transparency. Bytes 0 - 2: RGB					
-				UDoubleword	rw
Multi-line: yes	Number of the protection zone		MN_MM_MAXNUM_3D_PROT_AREA_ELEM		

pa3DElemDLevel	$NP_D_LEVEL				
Detailing level of a protection area element					
-				UDoubleword	rw
Multi-line: yes	Number of the protection zone		MN_MM_MAXNUM_3D_PROT_AREAS		

pa3DElemDir0	$NP_DIR[0]				
X components of rotary axis					
-				Double	rw
Multi-line: yes	Number of the protection zone element		$MN_MM_MAXNUM_3D_PROT_AREA_ELEM		

pa3DElemDir1	$NP_DIR[1]				
Y components of rotary axis					
-				Double	rw
Multi-line: yes	Number of the protection zone element		$MN_MM_MAXNUM_3D_PROT_AREA_ELEM		

pa3DElemDir2	$NP_DIR[2]				
Z components of rotary axis					
-				Double	rw
Multi-line: yes	Number of the protection zone element		$MN_MM_MAXNUM_3D_PROT_AREA_ELEM		

pa3DElemFileName	$NP_FILENAME				
Name of file containing the description of a protection area element of the "FILE" type.					
-				String [32]	rw
Multi-line: yes	Number of the protection zone element		$MN_MM_MAXNUM_3D_PROT_AREA_ELEM		

pa3DElemName	$NP_NAME				
Name of the protection zone element					
-				String [32]	rw
Multi-line: yes	Number of the protection zone element		$MN_MM_MAXNUM_3D_PROT_AREA_ELEM		

pa3DElemNext	$NP_NEXT				
Name of the next protection zone element					
-				String [32]	rw
Multi-line: yes	Number of the protection zone element		$MN_MM_MAXNUM_3D_PROT_AREA_ELEM		

pa3DElemNextP	$NP_NEXTP				
Name of next parallel protection area element					
-				String [32]	rw
Multi-line: yes	Number of the protection zone element		$MN_MM_MAXNUM_3D_PROT_AREA_ELEM		

pa3DElemOffset0	$NP_OFF[0]				
X components of the offset					
mm, inch, user defined				Double	rw
Multi-line: yes	Number of the protection zone element		$MN_MM_MAXNUM_3D_PROT_AREA_ELEM		

pa3DElemOffset1	$NP_OFF[1]				
Y components of the offset					
mm, inch, user defined				Double	rw
Multi-line: yes	Number of the protection zone element		$MN_MM_MAXNUM_3D_PROT_AREA_ELEM		

pa3DElemOffset2	$NP_OFF[2]				
Z components of the offset					
mm, inch, user defined				Double	rw
Multi-line: yes	Number of the protection zone element		$MN_MM_MAXNUM_3D_PROT_AREA_ELEM		

pa3DElemPara0	$NP_PARA[0]				
1st geometry parameter of protection zone element					
mm, inch, user defined				Double	rw
Multi-line: yes	Number of the protection zone element		$MN_MM_MAXNUM_3D_PROT_AREA_ELEM		

pa3DElemPara1	$NP_PARA[1]				
2nd geometry parameter of protection zone element					
mm, inch, user defined				Double	rw
Multi-line: yes	Number of the protection zone element		$MN_MM_MAXNUM_3D_PROT_AREA_ELEM		

pa3DElemPara2	$NP_PARA[2]				
3rd geometry parameter of protection zone element					
mm, inch, user defined				Double	rw
Multi-line: yes	Number of the protection zone element		$MN_MM_MAXNUM_3D_PROT_AREA_ELEM		

pa3DElemType	$NP_TYPE				
Type of the protection zone element					
-				String [32]	rw
Multi-line: yes	Number of the protection zone element		$MN_MM_MAXNUM_3D_PROT_AREA_ELEM		

pa3DElemUsage	$NP_USAGE				
Use of the protection area element to visualize ('V' or 'v"), to avoid collisions ('C' or 'c') or both ('A' or 'a'). ASCII code of one of the following letters: 'A' 'a', 'C', 'c', 'V', 'v'					
-				Character	rw
Multi-line: yes	Number of the protection zone		MN_MM_MAXNUM_3D_PROT_AREA_ELEM		

pa3DInitStat	$NP_INIT_STAT				
Initialization status of the protection zone ASCII code of one of the following letters: 'A', 'a', 'I', 'i', 'P', 'p'					
-				Character	rw
Multi-line: yes	Number of the protection zone		MN_MM_MAXNUM_3D_PROT_AREAS		

pa3DProtColor	$NP_PROT_COLOR				
Color and transparency of the protection area in QT format (HighByte: Transparency. Bytes 0 - 2: RGB					
-				UDoubleword	rw
Multi-line: yes	Number of the protection zone		MN_MM_MAXNUM_3D_PROT_AREAS		

pa3DProtDLevel	$NP_PROT_D_LEVEL				
Detailing level of a protection area					
-				UDoubleword	rw
Multi-line: yes	Number of the protection zone		MN_MM_MAXNUM_3D_PROT_AREAS		

pa3DProtDState					
Status of the PI service _N_PROT_D. The line index has the following meaning: 1: Counter for PI calls 2: Current status (0=not calculated, 1=running calculation, 2=ready) 3: Number of a potentially occuring alarm during calculation (0=no alarm)					
-				UWord	r
Multi-line: yes	Information on the PI service _N_PROT_D		3		

pa3DProtDistance					
Distance vector of the PI service _N_PROT_D. The line index has the following meaning: 1: X component 2: Y component 3: Z component 4. Absolute value of the vector					
mm, inch, user defined				Double	r
Multi-line: yes	Component		4		

pa3DProtName	$NP_PROT_NAME				
Name of a protection zone					
-				String [32]	rw
Multi-line: yes	Number of the protection zone		MN_MM_MAXNUM_3D_PROT_AREAS		

pa3DProtType	$NP_PROT_TYPE				
Type of protection area. Valid are the values "MACHINE" or "TOOL". No difference made between lower and upper cases.					
-				String [32]	rw
Multi-line: yes	Number of the protection zone		MN_MM_MAXNUM_3D_PROT_AREAS		

pa3DState					
Activation state of a protection zone					
-	0	0	3	Character	r
Multi-line: yes	Number of the protection zone		$MN_MM_MAXNUM_3D_PROT_AREAS		

pa3DTElemAngle	$NP_T_ANG				
Turning angle					
-				Double	r
Multi-line: yes	Number of the tool protection area element		$MN_MM_MAXNUM_3D_T_PROT_ELEM		

pa3DTElemDir0	$NP_T_DIR[0]				
X components of rotary axis					
-				Double	r
Multi-line: yes	Number of the tool protection area element		$MN_MM_MAXNUM_3D_T_PROT_ELEM		

pa3DTElemDir1	$NP_T_DIR[1]				
Y components of rotary axis					
-				Double	r
Multi-line: yes	Number of the tool protection area element		$MN_MM_MAXNUM_3D_T_PROT_ELEM		

pa3DTElemDir2	$NP_T_DIR[2]				
Z components of rotary axis					
-				Double	r
Multi-line: yes	Number of the tool protection area element		$MN_MM_MAXNUM_3D_T_PROT_ELEM		

pa3DTElemFileName	$NP_T_FILENAME				
Name of file containing the description of the tool protection area element of the "FILE" type.					
-				String [32]	r
Multi-line: yes	Number of the tool protection area element		$MN_MM_MAXNUM_3D_T_PROT_ELEM		

pa3DTElemName	$NP_T_NAME				
Name of the tool protection area element					
-				String [32]	r
Multi-line: yes	Number of the tool protection area element		$MN_MM_MAXNUM_3D_T_PROT_ELEM		

pa3DTElemOffset0	$NP_T_OFF[0]				
X components of the offset					
mm, inch, user defined				Double	r
Multi-line: yes	Number of the tool protection area element		$MN_MM_MAXNUM_3D_T_PROT_ELEM		

pa3DTElemOffset1	$NP_T_OFF[1]				
Y components of the offset					
mm, inch, user defined				Double	r
Multi-line: yes	Number of the tool protection area element		$MN_MM_MAXNUM_3D_T_PROT_ELEM		

pa3DTElemOffset2	$NP_T_OFF[2]				
Z components of the offset					
mm, inch, user defined				Double	r
Multi-line: yes	Number of the tool protection area element		$MN_MM_MAXNUM_3D_T_PROT_ELEM		

pa3DTElemPara0	$NP_T_PARA[0]				
1st geometry parameter of the tool protection area element					
mm, inch, user defined				Double	r
Multi-line: yes	Number of the tool protection area element		$MN_MM_MAXNUM_3D_T_PROT_ELEM		

pa3DTElemPara1	$NP_T_PARA[1]				
2nd geometry parameter of the tool protection area element					
mm, inch, user defined				Double	r
Multi-line: yes	Number of the tool protection area element		$MN_MM_MAXNUM_3D_T_PROT_ELEM		

pa3DTElemPara2	$NP_T_PARA[2]				
3rd geometry parameter of the tool protection area element					
mm, inch, user defined				Double	r
Multi-line: yes	Number of the tool protection area element		$MN_MM_MAXNUM_3D_T_PROT_ELEM		

pa3DTElemType	$NP_T_TYPE				
Type of the tool protection area element					
-				String [32]	r
Multi-line: yes	Number of the tool protection area element		$MN_MM_MAXNUM_3D_T_PROT_ELEM		

trafoDatAuxPos0	$NT_AUX_POS[n,0]				
X component of the auxiliary position for measuring cycles					
mm, inch, user defined				Double	r
Multi-line: yes	Number of transformer data record		$MN_MM_NUM_TRAFO_DATA_SETS		

trafoDatAuxPos1	$NT_AUX_POS[n,1]				
Y component of the auxiliary position for measuring cycles					
mm, inch, user defined				Double	r
Multi-line: yes	Number of transformer data record		$MN_MM_NUM_TRAFO_DATA_SETS		

trafoDatAuxPos2	$NT_AUX_POS[n,2]				
Z component of the auxiliary position for measuring cycles					
mm, inch, user defined				Double	r
Multi-line: yes	Number of transformer data record		$MN_MM_NUM_TRAFO_DATA_SETS		

trafoDatBaseOrient0	$NT_BASE_ORIENT[n, 0]				
X component of basic tool orientation					
-				Double	r
Multi-line: yes	Number of transformer data record		$MN_MM_NUM_TRAFO_DATA_SETS		

trafoDatBaseOrient1	$NT_BASE_ORIENT[n, 1]				
Y component of basic tool orientation					
-				Double	r
Multi-line: yes	Number of transformer data record		$MN_MM_NUM_TRAFO_DATA_SETS		

trafoDatBaseOrient2	$NT_BASE_ORIENT[n, 2]				
Z component of basic tool orientation					
-				Double	r
Multi-line: yes	Number of transformer data record		$MN_MM_NUM_TRAFO_DATA_SETS		

trafoDatBaseOrientNormal0	$NT_BASE_ORIENT_NORMAL[n, 0]				
X component of the normal vector of orientation					
-				Double	r
Multi-line: yes	Number of transformer data record		$MN_MM_NUM_TRAFO_DATA_SETS		

trafoDatBaseOrientNormal1	$NT_BASE_ORIENT_NORMAL[n, 1]				
Y component of the normal vector of orientation					
-				Double	r
Multi-line: yes	Number of transformer data record		$MN_MM_NUM_TRAFO_DATA_SETS		

trafoDatBaseOrientNormal2	$NT_BASE_ORIENT_NORMAL[n, 2]				
Z component of the normal vector of orientation					
-				Double	r
Multi-line: yes	Number of transformer data record	$MN_MM_NUM_TRAFO_DATA_SETS			

trafoDatCloseChainP	$NT_CLOSE_CHAIN_P[n]				
Element whose end point is used as a reference point for closing the part chain.					
-				String [32]	r
Multi-line: yes	Number of transformer data record	$MN_MM_NUM_TRAFO_DATA_SETS			

trafoDatCloseChainT	$NT_CLOSE_CHAIN_T[n]				
Element whose end point is used as a reference point for closing the tool chain.					
-				String [32]	r
Multi-line: yes	Number of transformer data record	$MN_MM_NUM_TRAFO_DATA_SETS			

trafoDatCntrl	$NT_CNTRL[n]				
This data is a bit-coded control word, with which the behavior can be influenced in specific situations. The individual bits have the following meaning: Bit 0 : Not assigned Bit 1 - 3: The orientation axis assigned to the bit (Bit 1: First orientation axis, Bit 2: Second orientation axis, Bit 3: Third orientation axis) is interpreted as speed-controlled spindle. We currently only support the cases in which either the first or the third orientation axis is parameterized as a spindle (turning on milling machines or 5-axis milling on machines on which the third orientation axis is not operated in position control mode). Bit 4 - 6: The orientation axis assigned to the bit (Bit 4: First orientation axis, Bit 5: Second orientation axis, Bit 6: Third orientation axis) is Hirth-toothed. For the Hirth toothing, only the machine data $MA_INDEX_AX_NUMERATOR, $MA_INDEX_AX_DENOMINATOR and $MA_INDEX_AX_OFFSET are evaluated. Der Inhalt des Maschinendatums $MA_HIRTH_IS_ACTIVE wird nicht ausgewertet, d.h. die Achse muss nicht als echte Hirthachse prametriert sein. Ist die Achse als Moduloachse parameriert, wird das Maschinendatum $MA_INDEX_AX_NUMERATOR durch das Maschinendatum $MA_MODULO_RANGE ersetzt. Die Abstände der zulässigen Achspositionen ergeben sich dann durch $MA_MODULO_RANGE / $MA_INDEX_AX_DENOMINATOR. Das Maschinendatum $MA_INDEX_AX_OFFSET wird auch bei Moduloachsen ausgewertet. Bit 7 - 8: Sind diese Bits gesetzt, werden an den Startpunkten der Teilketten (Bit7: Part-Kette; Bit 8: Tool-Kette) bei Bedarf intern automatisch zusätzliche konstante Kettenelmnente eingefügt, die eine Veebindung vom Endpunkt der Kette zum Maschinennullpunkt herstellen ("Kette schließen"). Bit 9 - 31: Nicht belegt					
-				Long Integer	r
Multi-line: yes	Number of transformer data record	$MN_MM_NUM_TRAFO_DATA_SETS			

trafoDatCorrElemP0	$NT_CORR_ELEM_P[n, 0]				
Name of the 1st correction element in the part chain					
-				String [32]	r
Multi-line: yes	Number of transformer data record	$MN_MM_NUM_TRAFO_DATA_SETS			

trafoDatCorrElemP1	$NT_CORR_ELEM_P[n, 1]				
Name of the 2nd correction element in the part chain					
-				String [32]	r
Multi-line: yes	Number of transformer data record	$MN_MM_NUM_TRAFO_DATA_SETS			

trafoDatCorrElemP2	$NT_CORR_ELEM_P[n, 2]				
Name of the 3rd correction element in the part chain					
-				String [32]	r
Multi-line: yes	Number of transformer data record	$MN_MM_NUM_TRAFO_DATA_SETS			

trafoDatCorrElemP3	$NT_CORR_ELEM_P[n, 3]				
Name of the 4st correction element in the part chain					
-				String [32]	r
Multi-line: yes	Number of transformer data record	$MN_MM_NUM_TRAFO_DATA_SETS			

trafoDatCorrElemT0	$NT_CORR_ELEM_T[n, 0]				
Name of the 1st correction element in the tool chain					
-				String [32]	r
Multi-line: yes	Number of transformer data record	$MN_MM_NUM_TRAFO_DATA_SETS			

trafoDatCorrElemT1	$NT_CORR_ELEM_T[n, 1]				
Name of the 2nd correction element in the tool chain					
-				String [32]	r
Multi-line: yes	Number of transformer data record	$MN_MM_NUM_TRAFO_DATA_SETS			

trafoDatCorrElemT2	$NT_CORR_ELEM_T[n, 2]				
Name of the 3rd correction element in the tool chain					
-				String [32]	r
Multi-line: yes	Number of transformer data record		$MN_MM_NUM_TRAFO_DATA_SETS		

trafoDatCorrElemT3	$NT_CORR_ELEM_T[n, 3]				
Name of the 4st correction element in the tool chain					
-				String [32]	r
Multi-line: yes	Number of transformer data record		$MN_MM_NUM_TRAFO_DATA_SETS		

trafoDatGeoAxName0	$NT_GEO_AX_NAME[n, 0]				
Name of first geometry axis					
-				String [32]	r
Multi-line: yes	Number of transformer data record		$MN_MM_NUM_TRAFO_DATA_SETS		

trafoDatGeoAxName1	$NT_GEO_AX_NAME[n, 1]				
Name of second geometry axis					
-				String [32]	r
Multi-line: yes	Number of transformer data record		$MN_MM_NUM_TRAFO_DATA_SETS		

trafoDatGeoAxName2	$NT_GEO_AX_NAME[n, 2]				
Name of third geometry axis					
-				String [32]	r
Multi-line: yes	Number of transformer data record		$MN_MM_NUM_TRAFO_DATA_SETS		

trafoDatHirthInc0	$NT_HIRTH_INC[n, 0]				
Angle increment of 1st rotary axis with Hirth tooth system					
Degree, user defined				Double	r
Multi-line: yes	Number of transformer data record		$MN_MM_NUM_TRAFO_DATA_SETS		

trafoDatHirthInc1	$NT_HIRTH_INC[n, 1]				
Angle increment of 2nd rotary axis with Hirth tooth system					
Degree, user defined				Double	r
Multi-line: yes	Number of transformer data record	$MN_MM_NUM_TRAFO_DATA_SETS			

trafoDatHirthInc2	$NT_HIRTH_INC[n, 2]				
Angle increment of 3rd rotary axis with Hirth tooth system					
Degree, user defined				Double	r
Multi-line: yes	Number of transformer data record	$MN_MM_NUM_TRAFO_DATA_SETS			

trafoDatHirthOff0	$NT_HIRTH_OFF[n, 0]				
Angle offset of 1st rotary axis with Hirth tooth system					
Degree, user defined				Double	r
Multi-line: yes	Number of transformer data record	$MN_MM_NUM_TRAFO_DATA_SETS			

trafoDatHirthOff1	$NT_HIRTH_OFF[n, 1]				
Angle offset of 2nd rotary axis with Hirth tooth system					
Degree, user defined				Double	r
Multi-line: yes	Number of transformer data record	$MN_MM_NUM_TRAFO_DATA_SETS			

trafoDatHirthOff2	$NT_HIRTH_OFF[n, 2]				
Angle offset of 3rd rotary axis with Hirth tooth system					
Degree, user defined				Double	r
Multi-line: yes	Number of transformer data record	$MN_MM_NUM_TRAFO_DATA_SETS			

trafoDatIdent0	$NT_IDENT[n, 0]				
Identifier 0, ID number 0, no significance in NCK					
-				Long Integer	r
Multi-line: yes	Number of transformer data record	$MN_MM_NUM_TRAFO_DATA_SETS			

trafoDatIdent1	$NT_IDENT[n, 1]				
Identifier 1, ID number 1, no significance in NCK					
-				Long Integer	r
Multi-line: yes	Number of transformer data record	$MN_MM_NUM_TRAFO_DATA_SETS			

trafoDatIdent2	$NT_IDENT[n, 2]				
Identifier 2, ID number 2, no significance in NCK					
-				Long Integer	r
Multi-line: yes	Number of transformer data record		$MN_MM_NUM_TRAFO_DATA_SETS		

trafoDatIgnoreToolOrient	$NT_IGNORE_TOOL_ORIENT[n]				
If this parameter is set, the orientation stored in the transformation data ($NT_BASE_ORIENT, $NT_BASE_ORIENT_NORMAL) is always used, regardless of the basic orientation contained in the tool data of an active tool, i.e. the orientation defined in the transformation data set takes precedence over the tool orientation.					
-				Bool	r
Multi-line: yes	Number of transformer data record		$MN_MM_NUM_TRAFO_DATA_SETS		

trafoDatName	$NT_NAME[n]				
Name of transformer data record					
-				String [32]	r
Multi-line: yes	Number of transformer data record		$MN_MM_NUM_TRAFO_DATA_SETS		

trafoDatPChainLastElem	$NT_P_CHAIN_LAST_ELEM[n]				
Name of last element of kin. chain for workpiece					
-				String [32]	r
Multi-line: yes	Number of transformer data record		$MN_MM_NUM_TRAFO_DATA_SETS		

trafoDatPoleLimit	$NT_POLE_LIMIT[n]				
End angle tolerance with interpolation through pole					
Degree, user defined				Double	r
Multi-line: yes	Number of transformer data record		$MN_MM_NUM_TRAFO_DATA_SETS		

trafoDatPoleSideFix	$NT_POLE_SIDE_FIX[n]				
Restriction of working area in front of/behind pole or no restriction, i.e. traversal through the pole. The assigned values have the following meanings: 0: No restriction of the working area. Traversal through the pole allowed. 1: Working area of linear axis for positions >=0, (if tool length compensation parallel to linear axis = 0) 2: Working area of linear axis for positions <=0, (if tool length compensation parallel to linear axis = 0)					
-				Long Integer	r
Multi-line: yes	Number of transformer data record		$MN_MM_NUM_TRAFO_DATA_SETS		

trafoDatPoleTol	$NT_POLE_TOL[n]				
End angle tolerance for pole interpolation					
Degree, user defined				Double	r
Multi-line: yes	Number of transformer data record		$MN_MM_NUM_TRAFO_DATA_SETS		

trafoDatRotAxCnt0	$NT_ROT_AX_CNT[n, 0]				
Number of relevant rotary axes in the part chain					
-				Long Integer	r
Multi-line: yes	Number of transformer data record		$MN_MM_NUM_TRAFO_DATA_SETS		

trafoDatRotAxCnt1	$NT_ROT_AX_CNT_[n, 1]				
Number of relevant rotary axes in the tool chain					
-				Long Integer	r
Multi-line: yes	Number of transformer data record		$MN_MM_NUM_TRAFO_DATA_SETS		

trafoDatRotAxMax0	$NT_ROT_AX_MAX[n, 0]				
Maximum position of 1st manual rotary axis					
Degree, user defined				Double	r
Multi-line: yes	Number of transformer data record		$MN_MM_NUM_TRAFO_DATA_SETS		

trafoDatRotAxMax1	$NT_ROT_AX_MAX[n, 1]				
Maximum position of 2nd manual rotary axis					
Degree, user defined				Double	r
Multi-line: yes	Number of transformer data record		$MN_MM_NUM_TRAFO_DATA_SETS		

trafoDatRotAxMax2	$NT_ROT_AX_MAX[n, 2]				
Maximum position of 3rd manual rotary axis					
Degree, user defined				Double	r
Multi-line: yes	Number of transformer data record		$MN_MM_NUM_TRAFO_DATA_SETS		

trafoDatRotAxMin0	$NT_ROT_AX_MIN[n, 0]				
Minimum position of 1st manual rotary axis					
Degree, user defined				Double	r
Multi-line: yes	Number of transformer data record		$MN_MM_NUM_TRAFO_DATA_SETS		

trafoDatRotAxMin1	$NT_ROT_AX_MIN[n, 1]				
Minimum position of 2nd manual rotary axis					
Degree, user defined				Double	r
Multi-line: yes	Number of transformer data record		$MN_MM_NUM_TRAFO_DATA_SETS		

trafoDatRotAxMin2	$NT_ROT_AX_MIN[n, 2]				
Minimum position of 3rd manual rotary axis					
Degree, user defined				Double	r
Multi-line: yes	Number of transformer data record		$MN_MM_NUM_TRAFO_DATA_SETS		

trafoDatRotAxName0	$NT_ROT_AX_NAME[n, 0]				
Name of first rotary axis					
-				String [32]	r
Multi-line: yes	Number of transformer data record		$MN_MM_NUM_TRAFO_DATA_SETS		

trafoDatRotAxName1	$NT_ROT_AX_NAME[n, 1]				
Name of second rotary axis					
-				String [32]	r
Multi-line: yes	Number of transformer data record		$MN_MM_NUM_TRAFO_DATA_SETS		

trafoDatRotAxName2	$NT_ROT_AX_NAME[n, 2]				
Name of third rotary axis					
-				String [32]	r
Multi-line: yes	Number of transformer data record		$MN_MM_NUM_TRAFO_DATA_SETS		

trafoDatRotAxPos0	$NT_ROT_AX_POS[n, 0]				
Position of the 1st manual rotary axis					
Degree, user defined				Double	r
Multi-line: yes	Number of transformer data record		$MN_MM_NUM_TRAFO_DATA_SETS		

trafoDatRotAxPos1	$NT_ROT_AX_POS[n, 1]				
Position of the 2nd manual rotary axis					
Degree, user defined				Double	r
Multi-line: yes	Number of transformer data record		$MN_MM_NUM_TRAFO_DATA_SETS		

trafoDatRotAxPos2	$NT_ROT_AX_POS[n, 2]				
Position of the 3rd manual rotary axis					
Degree, user defined				Double	r
Multi-line: yes	Number of transformer data record		$MN_MM_NUM_TRAFO_DATA_SETS		

trafoDatRotOffsetFromFrame	$NT_ROT_OFFSET_FROM_FRAME[n]				
Accept rotary offset for transformer selection from WO					
-				Bool	r
Multi-line: yes	Number of transformer data record		$MN_MM_NUM_TRAFO_DATA_SETS		

trafoDatTChainLastElem	$NT_T_CHAIN_LAST_ELEM[n]				
Name of last element of kin. chain for tool					
-				String [32]	r
Multi-line: yes	Number of transformer data record		$MN_MM_NUM_TRAFO_DATA_SETS		

trafoDatTRefElem	$NT_T_REF_ELEM[n]				
Reference point for tool length calculation					
-				String [32]	r
Multi-line: yes	Number of transformer data record		$MN_MM_NUM_TRAFO_DATA_SETS		

trafoDatTrafoIncludesTool	$NT_TRAFO_INCLUDES_TOOL[n]				
This system variable indicates whether the tool is handled internally or externally in the case of an active transformation.					
-				Bool	r
Multi-line: yes	Number of transformer data record		$MN_MM_NUM_TRAFO_DATA_SETS		

trafoDatTrafoIndex	$NT_TRAFO_INDEDX[n]				
A transformation defined by kinematic chains can also be activated by conventional language commands, such as TRAORI(<n>) or TRANSMIT(<n>), instead of the call TRAFOON(<Name>), if a value not equal to zero is entered in this system data, and the transformation type is compatible with the conventionally named transformation type. The hundreds and thousands digits indicate the channel in which the transformation can be called with a conventional language command. If both these digits are empty (zero), the definition applies to the first channel; this means that, for example, the entries "1" and "101" are equivalent. In order for a transformation defined by kinematic chains to be called with a conventional language command, the three lowest-value decimal positions of this system data must not be zero. The orientation transformation indicated by the index 1 is also activated for compatibility reasons with the conventional call syntax with TRAORI(0), TRAORI() or TRAORI but not with TRAORI(1). The same applies to the other transformation types (TRANSMIT, TRACYL and TRAANG).					
-				Long Integer	r
Multi-line: yes	Number of transformer data record		$MN_MM_NUM_TRAFO_DATA_SETS		

trafoDatTrafoType	$NT_TRAFO_TYPE				
Transformer type					
-				String [32]	r
Multi-line: yes	Number of transformer data record		$MN_MM_NUM_TRAFO_DATA_SETS		

3.2.4 Area C, Block PA : Channel-specific protection zones

OEM-MMC: Linkitem /ChannelProtectedArea/...

Up to 10 protection zones can be defined. Each protection zone is described by a polygon function consisting of up to 10 elements. The maximum permissible number of protection zones is specified in "numProtArea" in the module Y in area C. The maximum permissible number of polygon definition elements is specified in "numContourInProtArea" in module Y in area C. Module PA contains the individual coordinates of the polygon elements. The protection zones are addressed via the variable indices.

The classification as NCK or channel-specific protection zone does not affect the protection zone monitoring function but simply indicates the area in which the protection zone is registered.

The physical unit actually used for length quantities is defined in "/C/SGA/extUnit" in module SGA in area C.

MDD_PA_CENT_ABS_0	$SC_PA_CENT_ABS[x,0] x = Number protection zone				A3
Absolute abscissa value of arc centre of 1st contour element					
mm, inch, user defined				Double	r
Multi-line: yes	Number of protection zone		numProtArea		

MDD_PA_CENT_ABS_1	$SC_PA_CENT_ABS[x,1] x = Number protection zone				A3
Absolute abscissa value of arc centre of 2nd contour element					
mm, inch, user defined				Double	r
Multi-line: yes	Number of protection zone		numProtArea		

MDD_PA_CENT_ABS_2	$SC_PA_CENT_ABS[x,2] x = Number protection zone				A3
Absolute abscissa value of arc centre of 3rd contour element					
mm, inch, user defined				Double	r
Multi-line: yes	Number of protection zone		numProtArea		

MDD_PA_CENT_ABS_3	$SC_PA_CENT_ABS[x,3] x = Number protection zone				A3
Absolute abscissa value of arc centre of 4th contour element					
mm, inch, user defined				Double	r
Multi-line: yes	Number of protection zone		numProtArea		

MDD_PA_CENT_ABS_4	$SC_PA_CENT_ABS[x,4] x = Number protection zone				A3
Absolute abscissa value of arc centre of 5th contour element					
mm, inch, user defined				Double	r
Multi-line: yes	Number of protection zone		numProtArea		

MDD_PA_CENT_ABS_5	$SC_PA_CENT_ABS[x,5] x = Number protection zone				A3
Absolute abscissa value of arc centre of 6th contour element					
mm, inch, user defined				Double	r
Multi-line: yes	Number of protection zone		numProtArea		

MDD_PA_CENT_ABS_6	$SC_PA_CENT_ABS[x,6] x = Number protection zone				A3
Absolute abscissa value of arc centre of 7th contour element					
mm, inch, user defined				Double	r
Multi-line: yes	Number of protection zone		numProtArea		

MDD_PA_CENT_ABS_7	$SC_PA_CENT_ABS[x,7] x = Number protection zone				A3
Absolute abscissa value of arc centre of 8th contour element					
mm, inch, user defined				Double	r
Multi-line: yes	Number of protection zone		numProtArea		

MDD_PA_CENT_ABS_8	$SC_PA_CENT_ABS[x,8] x = Number protection zone				A3
Absolute abscissa value of arc centre of 9th contour element					
mm, inch, user defined				Double	r
Multi-line: yes	Number of protection zone		numProtArea		

MDD_PA_CENT_ABS_9	$SC_PA_CENT_ABS[x,9] x = Number protection zone				A3
Absolute abscissa value of arc centre of 10th contour element					
mm, inch, user defined				Double	r
Multi-line: yes	Number of protection zone		numProtArea		

MDD_PA_CENT_ORD_0	$SC_PA_CENT_ORD[x,0] x = Number protection zone				A3
Absolute ordinate value of arc centre of 1st contour element					
mm, inch, user defined				Double	r
Multi-line: yes	Number of protection zone		numProtArea		

MDD_PA_CENT_ORD_1	$SC_PA_CENT_ORD[x,1] x = Number protection zone				A3
Absolute ordinate value of arc centre of 2nd contour element					
mm, inch, user defined				Double	r
Multi-line: yes	Number of protection zone		numProtArea		

MDD_PA_CENT_ORD_2	$SC_PA_CENT_ORD[x,2] x = Number protection zone				A3
Absolute ordinate value of arc centre of 3rd contour element					
mm, inch, user defined				Double	r
Multi-line: yes	Number of protection zone		numProtArea		

MDD_PA_CENT_ORD_3	$SC_PA_CENT_ORD[x,3] x = Number protection zone				A3
Absolute ordinate value of arc centre of 4th contour element					
mm, inch, user defined				Double	r
Multi-line: yes	Number of protection zone		numProtArea		

MDD_PA_CENT_ORD_4	$SC_PA_CENT_ORD[x,4] x = Number protection zone				A3
Absolute ordinate value of arc centre of 5th contour element					
mm, inch, user defined				Double	r
Multi-line: yes	Number of protection zone		numProtArea		

MDD_PA_CENT_ORD_5	$SC_PA_CENT_ORD[x,5] x = Number protection zone				A3
Absolute ordinate value of arc centre of 6th contour element					
mm, inch, user defined				Double	r
Multi-line: yes	Number of protection zone		numProtArea		

MDD_PA_CENT_ORD_6	$SC_PA_CENT_ORD[x,6] x = Number protection zone				A3
Absolute ordinate value of arc centre of 7th contour element					
mm, inch, user defined				Double	r
Multi-line: yes	Number of protection zone		numProtArea		

MDD_PA_CENT_ORD_7	$SC_PA_CENT_ORD[x,7] x = Number protection zone				A3
Absolute ordinate value of arc centre of 8th contour element					
mm, inch, user defined				Double	r
Multi-line: yes	Number of protection zone		numProtArea		

MDD_PA_CENT_ORD_8	$SC_PA_CENT_ORD[x,8] x = Number protection zone				A3
Absolute ordinate value of arc centre of 9th contour element					
mm, inch, user defined				Double	r
Multi-line: yes	Number of protection zone		numProtArea		

MDD_PA_CENT_ORD_9	$SC_PA_CENT_ORD[x,9] x = Number protection zone				A3
Absolute ordinate value of arc centre of 10th contour element					
mm, inch, user defined				Double	r
Multi-line: yes	Number of protection zone		numProtArea		

MDD_PA_CONT_ABS_0	$SC_PA_CONT_ABS[x,0] x = Number protection zone				A3
Absolute abscissa value of end point of 1st contour element					
mm, inch, user defined				Double	r
Multi-line: yes	Number of protection zone		numProtArea		

MDD_PA_CONT_ABS_1	$SC_PA_CONT_ABS[x,1] x = Number protection zone				A3
Absolute abscissa value of end point of 2nd contour element					
mm, inch, user defined				Double	r
Multi-line: yes	Number of protection zone		numProtArea		

MDD_PA_CONT_ABS_2	$SC_PA_CONT_ABS[x,2] x = Number protection zone				A3
Absolute abscissa value of end point of 3rd contour element					
mm, inch, user defined				Double	r
Multi-line: yes	Number of protection zone		numProtArea		

MDD_PA_CONT_ABS_3	$SC_PA_CONT_ABS[x,3] x = Number protection zone				A3
Absolute abscissa value of end point of 4th contour element					
mm, inch, user defined				Double	r
Multi-line: yes	Number of protection zone		numProtArea		

MDD_PA_CONT_ABS_4	$SC_PA_CONT_ABS[x,4] x = Number protection zone				A3
Absolute abscissa value of end point of 5th contour element					
mm, inch, user defined				Double	r
Multi-line: yes	Number of protection zone		numProtArea		

MDD_PA_CONT_ABS_5	$SC_PA_CONT_ABS[x,5] x = Number protection zone				A3
Absolute abscissa value of end point of 6th contour element					
mm, inch, user defined				Double	r
Multi-line: yes	Number of protection zone		numProtArea		

MDD_PA_CONT_ABS_6	$SC_PA_CONT_ABS[x,6] x = Number protection zone				A3
Absolute abscissa value of end point of 7th contour element					
mm, inch, user defined				Double	r
Multi-line: yes	Number of protection zone		numProtArea		

MDD_PA_CONT_ABS_7	$SC_PA_CONT_ABS[x,7] x = Number protection zone				A3
Absolute abscissa value of end point of 8th contour element					
mm, inch, user defined				Double	r
Multi-line: yes	Number of protection zone		numProtArea		

MDD_PA_CONT_ABS_8	$SC_PA_CONT_ABS[x,8] x = Number protection zone				A3
Absolute abscissa value of end point of 9th contour element					
mm, inch, user defined				Double	r
Multi-line: yes	Number of protection zone		numProtArea		

MDD_PA_CONT_ABS_9	$SC_PA_CONT_ABS[x,9] x = Number protection zone				A3
Absolute abscissa value of end point of 10th contour element					
mm, inch, user defined				Double	r
Multi-line: yes	Number of protection zone		numProtArea		

MDD_PA_CONT_ORD_0	$SC_PA_CONT_ORD[x,0] x = Number protection zone				A3
Absolute ordinate value of end point of 1st contour element					
mm, inch, user defined				Double	r
Multi-line: yes	Number of protection zone		numProtArea		

MDD_PA_CONT_ORD_1	$SC_PA_CONT_ORD[x,1] x = Number protection zone				A3
Absolute ordinate value of end point of 2nd contour element					
mm, inch, user defined				Double	r
Multi-line: yes	Number of protection zone		numProtArea		

MDD_PA_CONT_ORD_2	$SC_PA_CONT_ORD[x,2] x = Number protection zone				A3
Absolute ordinate value of end point of 3rd contour element					
mm, inch, user defined				Double	r
Multi-line: yes	Number of protection zone		numProtArea		

MDD_PA_CONT_ORD_3	$SC_PA_CONT_ORD[x,3] x = Number protection zone				A3
Absolute ordinate value of end point of 4th contour element					
mm, inch, user defined				Double	r
Multi-line: yes	Number of protection zone		numProtArea		

MDD_PA_CONT_ORD_4	$SC_PA_CONT_ORD[x,4] x = Number protection zone				A3
Absolute ordinate value of end point of 5th contour element					
mm, inch, user defined				Double	r
Multi-line: yes	Number of protection zone		numProtArea		

MDD_PA_CONT_ORD_5	$SC_PA_CONT_ORD[x,5] x = Number protection zone				A3
Absolute ordinate value of end point of 6th contour element					
mm, inch, user defined				Double	r
Multi-line: yes	Number of protection zone		numProtArea		

MDD_PA_CONT_ORD_6	$SC_PA_CONT_ORD[x,6] x = Number protection zone				A3
Absolute ordinate value of end point of 7th contour element					
mm, inch, user defined				Double	r
Multi-line: yes	Number of protection zone		numProtArea		

MDD_PA_CONT_ORD_7	$SC_PA_CONT_ORD[x,7] x = Number protection zone				A3
Absolute ordinate value of end point of 8th contour element					
mm, inch, user defined				Double	r
Multi-line: yes	Number of protection zone		numProtArea		

MDD_PA_CONT_ORD_8	$SC_PA_CONT_ORD[x,8] x = Number protection zone				A3
Absolute ordinate value of end point of 9th contour element					
mm, inch, user defined				Double	r
Multi-line: yes	Number of protection zone		numProtArea		

MDD_PA_CONT_ORD_9	$SC_PA_CONT_ORD[x,9] x = Number protection zone				A3
Absolute ordinate value of end point of 10th contour element					
mm, inch, user defined				Double	r
Multi-line: yes	Number of protection zone		numProtArea		

MDD_PA_MINUS_LIM	$SC_PA_MINUS_LIM[x] x = Number protection zone				A3
Limitation in the minus direction of the protection zone in the axis that is perpendicular to the polygon definition (applicate)					
mm, inch, user defined				Double	r
Multi-line: yes	Number of protection zone		numProtArea		

MDD_PA_PLUS_LIM	$SC_PA_PLUS_LIM[x] x = Number protection zone				A3
Limitation of the protection zone in the plus direction of the axis that is perpendicular to the polygon definition (applicate)					
mm, inch, user defined				Double	r
Multi-line: yes	Number of protection zone		numProtArea		

MDU_PA_ACTIV_IMMED	$SC_PA_ACTIV_IMMED[x] x = Number protection zone				A3
Code for "active immediately after referencing", i.e. the protection zone is active as soon as the control has been started up and the axes have been referenced 0 = protection zone is not active immediately 1 = protection zone is active immediately					
-				UWord	r
Multi-line: yes	Number of protection zone		numProtArea		

MDU_PA_CONT_NUM	$SC_PA_CONT_NUM[x] x = Number protection zone				A3
Number of valid contour elements					
-		0	numContourInProtArea	UWord	r
Multi-line: yes	Number of protection zone		numProtArea		

MDU_PA_CONT_TYP_0	$SC_PA_CONT_TYP[x,0] x = Number protection zone				A3
Contour type of 1st contour element 0 = G1 1 = G2 2 = G3					
-				UWord	r
Multi-line: yes	Number of protection zone		numProtArea		

MDU_PA_CONT_TYP_1	$SC_PA_CONT_TYP[x,1] x = Number protection zone				A3
Contour type of 2nd contour element 0 = G1 1 = G2 2 = G3					
-				UWord	r
Multi-line: yes	Number of protection zone		numProtArea		

MDU_PA_CONT_TYP_2	$SC_PA_CONT_TYP[x,2] x = Number protection zone				A3
Contour type of 3rd contour element 0 = G1 1 = G2 2 = G3					
-				UWord	r
Multi-line: yes	Number of protection zone		numProtArea		

MDU_PA_CONT_TYP_3	$SC_PA_CONT_TYP[x,3] x = Number protection zone				A3
Contour type of 4th contour element 0 = G1 1 = G2 2 = G3					
-				UWord	r
Multi-line: yes	Number of protection zone		numProtArea		

MDU_PA_CONT_TYP_4	$SC_PA_CONT_TYP[x,4] x = Number protection zone				A3
Contour type of 5th contour element 0 = G1 1 = G2 2 = G3					
-				UWord	r
Multi-line: yes	Number of protection zone		numProtArea		

MDU_PA_CONT_TYP_5	$SC_PA_CONT_TYP[x,5] x = Number protection zone				A3
Contour type of 6th contour element 0 = G1 1 = G2 2 = G3					
-				UWord	r
Multi-line: yes	Number of protection zone		numProtArea		

MDU_PA_CONT_TYP_6	$SC_PA_CONT_TYP[x,6] x = Number protection zone				A3
Contour type of 7th contour element 0 = G1 1 = G2 2 = G3					
-				UWord	r
Multi-line: yes	Number of protection zone		numProtArea		

MDU_PA_CONT_TYP_7	$SC_PA_CONT_TYP[x,7] x = Number protection zone				A3
Contour type of 8th contour element 0 = G1 1 = G2 2 = G3					
-				UWord	r
Multi-line: yes	Number of protection zone		numProtArea		

MDU_PA_CONT_TYP_8	$SC_PA_CONT_TYP[x,8] x = Number protection zone				A3
Contour type of 9th contour element 0 = G1 1 = G2 2 = G3					
-				UWord	r
Multi-line: yes	Number of protection zone		numProtArea		

MDU_PA_CONT_TYP_9	$SC_PA_CONT_TYP[x,9] x = Number protection zone				A3
Contour type of 10th contour element 0 = G1 1 = G2 2 = G3					
-				UWord	r
Multi-line: yes	Number of protection zone		numProtArea		

MDU_PA_LIM_3DIM	$SC_PA_LIM_3DIM[x] x = Number protection zone				A3
Code for limitation of protection zone in the axis that is perpendicular to polygon definition (applicate) 0 = no limitation 1 = limitation in positive direction 2 = limitation in negative direction 3 = limitation in both directions					
-				UWord	r
Multi-line: yes	Number of protection zone		numProtArea		

MDU_PA_ORI	$SC_PA_ORI[x] x = Number protection zone				A3
Code for plane assignment of protection zone 0 = G17 1 = G18 2 = G19					
-				UWord	r
Multi-line: yes	Number of protection zone		numProtArea		

MDU_PA_TW	$SC_PA_T_W[x] x = Number protection zone				A3
Code for workpiece or tool-oriented protection zone 0 = workpiece-related 1 = reserved 2 = reserved 3 = tool-related					
-				UWord	r
Multi-line: yes	Number of protection zone		numProtArea		

acCollPos					
Contact point between two collision bodies during a collision alarm. Corresponds with system variable $AC_COLLPOS					
mm, inch, user defined				Double	r
Multi-line: yes	1,2,3 = X,Y,Z coordinate of position		3		

collisionAlarm	OD19830 $ON_COLLISION_MASK				
Collision of two protection zones					
-				UWord	r
Multi-line: yes	1: 0=No collision in the channel, otherwise alarm number 2: Number of the first protection zone 3: Number of the second protection zone		3		

declarProtObject					
Declaration of a variable protection zone 0=No object 1=WORKPIECE 2=FIXTURE					
-				UWord	r
Multi-line: yes	1		1		

declarProtObjectReal					
Real parameters of the declaration of a variable protection zone					
-				Double	r
Multi-line: yes	Number of the real parameter. The number and significance of the real parameters depend on the fourth string parameter (declarProtObjectString, row index 4).		10		

declarProtObjectString					
String parameter(s) of the declaration of a variable protection zone					
-				String [32]	r
Multi-line: yes	Number of the string parameter		4		

fixtureStatus					
Error status after calling the procedure FIXTURE Corresponds to the system variable $P_FIX_STAT					
-				short Integer	r
Multi-line: no					

workpieceStatus					
Error status after calling the procedure WORKPIECE Corresponds to the system variable $P_WP_STAT					
-				short Integer	r
Multi-line: no					

3.2.5 Area N, Block YNCFL : NCK instruction groups

OEM-MMC: Linkitem /NckFunctionGrouping/...

All G functions currently configured for the channels are made available for reading by the NCK. They are configured via machine data. Since the G functions are organized in groups, only one of which can be active at a time, this module is organized as a table.

There are two columns for each G group. The 1st column lists the number of G functions in a group (/N/YNCFL/Gruppe_NUM), this corresponds to the number of rows in each subsequent column. This second column contains all the G functions belonging to a group (/N/YNCFL/Gruppe).

As a result, the data for a certain G group are calculated via a column offset.

The column offset of each variable is:

2 * (G group number - 1)

The number of G groups is given in the variable "numGCodeGroups" in area N / module Y. The resultant the maximum column offset of the variables is thus 2 * numGCodeGroups.

The G functions currently active are listed in area C / module SNCF.

Gruppe					
Instruction group					
-				String [16]	r
Multi-line: yes	Serial number		Gruppe_NUM		

Gruppe_NUM					
Number of G functions in the relevant group					
-				UWord	r
Multi-line: yes	1		1		

3.3 Status data of the system

3.3.1 Area N, Block S : Global state data

OEM-MMC: Linkitem /NckState/...

During NC operation different internal states occur and system-specific data may change during operation. To distinguish those from system variables, they are classified as state data.

A distinction is made between:

- NCK-specific state data

- Mode-group-specific state data

- Channel-specific state data

- Drive-specific state data (FDD)

- Drive-specific state data (MSD)

aDbb	$A_DBB[x] x = ByteNo				
Data byte from/to the PLC					
-				UWord	rw
Multi-line: yes	Position offset within an I/O area				

aDbd	$A_DBD[x] x = Offset				
Data double word (32 bits) from/to the PLC					
-				Long Integer	rw
Multi-line: yes	Position offset within an I/O area. The offset refers to the byte, with the count beginning at 0. Permissible values for x are thus 0, 4, 8 etc.				

aDbr	$A_DBR[x] x = Offset				
Real data (32 bits) from/to the PLC					
-				Double	rw
Multi-line: yes	Position offset within an I/O area				

aDbsb	$A_DBSB				
PLC data byte					
-	0	-128	127	Long Integer	r
Multi-line: yes	Position offset within the I/O range 0-	1023			

aDbsw	$A_DBSW				
PLC data word					
-	0	-32768	32767	Long Integer	r
Multi-line: yes	Position offset within the I/O range 0-	1022			

aDbw	$A_DBW[x] x = Offset				
Data word (16 bits) from/to the PLC					
-				UWord	rw
Multi-line: yes	Position offset within an I/O area				

aDlb	$A_DLB[index]				
Data byte (8 bits) in link area					
-				UWord	rw
Multi-line: yes	Position offset within link data area				

aDld	$A_DLD[index]				
Data double word (32 bits) in link data area					
-				Long Integer	rw
Multi-line: yes	Position offset within link data area				

aDlr	$A_DLR[index]				
Read data (32 bits) in link data area					
-				Double	rw
Multi-line: yes	Position offset within link data area				

aDlw	$A_DLW[index]				
Data word (16 bits) in link data area					
-				UWord	rw
Multi-line: yes	Position offset within link data area				

aDpInConf	$A_DP_IN_CONF				
PROFIBUS configured input data areas					
-	0	0	0xffffffff	Long Integer	r
Multi-line: yes	1		1		

aDpInLength	$A_DP_IN_LENGTH				
PROFIBUS length of input data area					
-	0	0	128	Long Integer	r
Multi-line: yes	RangeIndex		32		

aDpInState	$A_DP_IN_STATE				
PROFIBUS status of input data area					
-	0	0	3	Long Integer	r
Multi-line: yes	RangeIndex		32		

aDpInValid	$A_DP_IN_VALID				
PROFIBUS valid input data areas					
-	0	0	0xffffffff	Long Integer	r
Multi-line: yes	1		1		

aDpOutConf	$A_DP_OUT_CONF				
PROFIBUS configured output data areas					
-	0	0	0xffffffff	Long Integer	r
Multi-line: yes	1		1		

aDpOutLength	$A_DP_OUT_LENGTH				
PROFIBUS length of output data area					
-	0	0	128	Long Integer	r
Multi-line: yes	RangeIndex		32		

aDpOutState	$A_DP_OUT_STATE				
PROFIBUS status of output data area					
-	0	0	3	Long Integer	r
Multi-line: yes	RangeIndex		32		

aDpOutValid	$A_DP_OUT_VALID				
PROFIBUS valid output data areas					
-	0	0	0xffffffff	Long Integer	r
Multi-line: yes	1		1		

aDpbIn	$A_DPB_IN				
PROFIBUS input byte (unsigned)					
-	0	0	255	Long Integer	r
Multi-line: yes	LowByte: RangeOffset HighByte: RangeIndex (0-31)				

aDpbOut	$A_DPB_OUT				
PROFIBUS output byte (unsigned)					
-	0	0	255	Long Integer	r
Multi-line: yes	LowByte: RangeOffset HighByte: RangeIndex (0-31)				

aDprIn	$A_DPR_IN				
PROFIBUS input data (32 bit REAL)					
-	0			Double	r
Multi-line: yes	LowByte: RangeOffset HighByte: RangeIndex (0-31)				

aDprOut	$A_DPR_OUT				
PROFIBUS output data (32 bit REAL)					
-	0			Double	r
Multi-line: yes	LowByte: RangeOffset HighByte: RangeIndex (0-31)				

aDpsbIn	$A_DPSB_IN				
PROFIBUS input byte (signed)					
-	0	-128	127	Long Integer	r
Multi-line: yes	LowByte: RangeOffset HighByte: RangeIndex (0-31)				

aDpsbOut	$A_DPSB_OUT				
PROFIBUS output byte (signed)					
-	0	-128	127	Long Integer	r
Multi-line: yes	LowByte: RangeOffset HighByte: RangeIndex (0-31)				

aDpsdIn	$A_DPSD_IN				
PROFIBUS input data double word (signed)					
-	0			Long Integer	r
Multi-line: yes	LowByte: RangeOffset HighByte: RangeIndex (0-31)				

aDpsdOut	$A_DPSD_OUT				
PROFIBUS output data double word (signed)					
-	0			Long Integer	r
Multi-line: yes	LowByte: RangeOffset HighByte: RangeIndex (0-31)				

aDpswIn	$A_DPSW_IN				
PROFIBUS input word (signed)					
-	0	-32768	32767	Long Integer	r
Multi-line: yes	LowByte: RangeOffset HighByte: RangeIndex (0-31)				

aDpswOut	$A_DPSW_OUT				
PROFIBUS output word (signed)					
-	0	-32768	32767	Long Integer	r
Multi-line: yes	LowByte: RangeOffset HighByte: RangeIndex (0-31)				

aDpwIn	$A_DPW_IN				
PROFIBUS input word (unsigned)					
-	0	0	65535	Long Integer	r
Multi-line: yes	LowByte: RangeOffset HighByte: RangeIndex (0-31)				

aDpwOut	$A_DPW_OUT				
PROFIBUS output word (unsigned)					
-	0	0	65535	Long Integer	r
Multi-line: yes	LowByte: RangeOffset HighByte: RangeIndex (0-31)				

aInco	$A_INCO[x] x = InputNo				
Comperator input NC					
-				UWord	r
Multi-line: yes	Input number		2		

aInsip					
Corresponds to safeIntInpValPlcBit 0: Input not set 1: Input set					
-	0	0	1	UWord	r
Multi-line: yes	Bit number		64		

aPbbIn	$A_PBB_IN[index]				
Data byte (8bits) in PLC input/output area IN (also available on 810D CCU2) Neg. values are also permitted in spite of TYPE_UWORD					
-				UWord	r
Multi-line: yes	Position offset within PLC input/ output area				

aPbbOut	$A_PBB_OUT[index]				
Data byte (8 bits) in PLC input/output area OUT (also available on 810D CCU2) Neg. values are also permitted in spite of TYPE_UWORD					
-				UWord	rw
Multi-line: yes	Position offset within PLC input/ output area				

aPbdIn	$A_PBD_IN[index]				
Data double word (32bits) in PLC input/output area IN					
-				Long Integer	r
Multi-line: yes	Position offset within PLC input/ output area				

aPbdOut	$A_PBD_OUT[index]				
Data double word (32 bits) in the PLC input/output area OUT (also available on 810D CCU2)					
-				Long Integer	rw
Multi-line: yes	Position offset within PLC input/ output area				

aPbrIn	$A_PBR_IN[index]				
Real data (32bits) in PLC input/output area IN (also available on 810D CCU2)					
-				Double	r
Multi-line: yes	Position offset within PLC input/ output area				

aPbrOut	$A_PBR_OUT[index]				
Real data (32 bits) in the PLC input/output area OUT (also available on 810D CCU2)					
-				Double	rw
Multi-line: yes	Position offset within PLC input/ output area				

aPbwIn	$A_PBW_IN[index]				
Data word (16bits) in PLC input/output area IN (also available on 810D CCU2) Neg. values are also permitted in spite of TYPE_UWORD					
-				UWord	r
Multi-line: yes	Position offset within PLC input/ output area				

aPbwOut	$A_PBW_OUT[index]				
Data word (16 bits) in the PLC input/output area OUT (also available on 810D CCU2) Neg. values are also permitted in spite of TYPE_UWORD					
-				UWord	rw
Multi-line: yes	Position offset within PLC input/ output area				

aProbe	$A_PROBE				
Probe status 0: Not deflected 1: Deflected					
-	0	0	1	UWord	r
Multi-line: yes	Probe number		2		

aProbeLimited	$A_PROBE_LIMITED				
Contains the accumulated number of DP communication cycles where at least one limitation was effective. An increasing value signals that the frequency of the measuring probe signals must be reduced (i.e. by reducing the speed of the toothed wheel to be measured).					
-	0	0		UWord	rw
Multi-line: yes	Probe number		2		

aStopesi	$A_STOPESI				
Current Safety Integrated Stop E for some axis Value 0: No Stop E Value not equal to 0: A Stop E is currently applied for some axis					
-	0	0		UWord	r
Multi-line: yes	1		1		

aXfaultsi	$A_XFAULTSI				
Information about Stop F for a safety axis: Bit 0 = 1: An actual value error has been discovered in the cross-check between NCK and the drive of any safety axis. Bit 1 = 1: An error has been discovered in the cross-check between NCK and the drive of any axis, and the waiting time until Stop B is triggered is running or has expired in this axis ($MA_SAFE_STOP_SWITCH_TIME_F)					
-	0	0		Long Integer	r
Multi-line: yes	1		1		

accIndex					
Global upload starting point for ACC entries. If a value is set here, upload access to _N_xx_yyy_ACC modules starts from this entry.					
-	1			UWord	rw
Multi-line: no					

anActivateCollCheck	$AN_ACTIVATE_COLL_CHECK				
Status of the field ActivateCollcheck on the interface PLC->NCK (DB10.DBX234.0 - DB10.DBX241.7). Data are made available in groups of 4 bytes, i.e. with index 1 you receive the first 4 bytes (DB10.DBX234.0 - DB10.DBX237.7), with index 2 the second 4 bytes (DB10.DBX238.0 - DB10.DBX241.7)					
-	0			Long Integer	r
Multi-line: yes	Index may be 1 or 2		2		

anAuxfuListChanno	$AN_AUXFU_LIST_CHANNO[n]				
Channel number of the auxiliary function collected in the channel. The variable is only valid in combination with block search type 5 (SERUPRO).					
-	0	0	MD_MAXNUM_AUXFU_CHANNELS	Long Integer	rw
Multi-line: yes	List index		1280		

anAuxfuListEndindex	$AN_AUXFU_LIST_ENDINDEX				
The variable determines the last valid index for the global auxiliary function list.					
-	0	-1	MD_MAXNUM_AUXFU_LIST_INDEX	Long Integer	r
Multi-line: yes	1		1		

anAuxfuListGroupindex	$AN_AUXFU_LIST_GROUPINDEX[n]				
Group index of the auxiliary function collected in the channel. The variable is only valid in combination with block search type 5 (SERUPRO).					
-	0	0	MD_MAXNUM_AUXFU_GROUPS - 1	Long Integer	rw
Multi-line: yes	List index		1280		

anAxCtAS	$AN_AXCTAS[n]				
Current container rotation, i.e. by how many slots the axis container has been currently advanced. The original container assignment is valid after Power On and outputs value 0. maxCount = Maximum number of assigned locations in the axis container - 1					
-	0	0	maxnumContainerSlots - 1	UWord	r
Multi-line: yes	Container no.		numContainer		

anAxCtSwA	$AN_AXCTSWA[CTn]				
A rotation is currently being executed on the axis container.					
-	0	0	1	UWord	r
Multi-line: yes	Container no.		numContainer		

anAxEsrTrigger	$AN_ESR_TRIGGER				
(Global) control signal "Start Stop/Retract". With a signal edge change from 0 to 1, the reactions parameterized beforehand in axial MD $MA_ESR_REACTION and enabled via system variable $AA_ESR_ENABLE are started. Independent drive reactions subsequently require a Power-Off / Power-On, independent NC reactions require at least an opposite edge change in the relevant system variable as well as a Reset. 0: FALSE 1: TRUE					
-	0	0	1	UWord	r
Multi-line: yes	1		1		

anAxctSwE	$AN_AXCTSWE				
Has a release for rotation been granted for a slot on an axis container? Bit mask, each bit corresponds to a slot, e.g. 0x5 corresponds to slots 1 and 3. Bit == 1: The slot of an axis container has been released for rotation. Bit == 0: The slot of an axis container has not been released for rotation. Example: Axis container with 4 slots: 'Hfff5' slot 1 and 3 have been released for rotation. As soon as a slot has been released for axis container rotation, bit == 1 is also output for unused slots. See example 'Hfff0'. If the slots of an axis container are distributed across several NCUs, the current state of the slots on other NCUs is only displayed if all of the slots on the other NCU have been released for axis container rotation.					
-	0	0	0xffffffff	UDoubleword	r
Multi-line: yes	Container no.		numContainer		

anCecDirection	$AN_CEC_DIRECTION				
The variable activates the direction-dependent activation of the compensation table: 0: Both traversing directions of the basic axis 1: Positive traversing direction of the basic axis -1: Negative traversing direction of the basic axis					
-	0			Long Integer	r
Multi-line: yes	Number of the compensation table		62		

anCecInputAxis	$AN_CEC_INPUT_AXIS				
The variable describes the number of the axis whose setpoint is used as entry for the compensation table. The value -1 indicates that no axis has been programmed.					
-	-1			Long Integer	r
Multi-line: yes	Number of the compensation table	62			

anCecInputNcu	$AN_CEC_INPUT_NCU				
The variable indicates the number of the NCU, for which the basic axis is calculated. It returns the value 0 if no NCU has been programmed.					
-	0			UWord	r
Multi-line: yes	Number of the compensation table	62			

anCecIsModulo	$AN_CEC_IS_MODULO				
The variable indicates whether the values of the corresponding compensation table shall be repeated cyclically: TRUE: Cyclic repetition of the compensation table FALSE: No cyclic repetition of the compensation table					
-	FALSE			Bool	r
Multi-line: yes	Number of the compensation table	62			

anCecMax	$AN_CEC_MAX				
The variable indicates the end position of the compensation table.					
-	0.0			Double	r
Multi-line: yes	Number of the compensation table	62			

anCecMin	$AN_CEC_MIN				
The variable indicates the start position of the compensation table.					
-	0.0			Double	r
Multi-line: yes	Number of the compensation table	62			

anCecMultByTable	$AN_CEC_MULT_BY_TABLE				
The variable indicates the number of the table whose output value shall be multiplied with the output value of the compensation table.					
-	0			UWord	r
Multi-line: yes	Number of the compensation table	62			

anCecOutputAxis	$AN_CEC_OUTPUT_AXIS				
The variable describes the number of the axis onto which the output of the compensation table has an effect. The value -1 indicates that no axis has been programmed,					
-	-1			Long Integer	r
Multi-line: yes	Number of the compensation table		62		

anCecOutputNcu	$AN_CEC_OUTPUT_NCU				
The variable indicates the number of the NCU which the compensation axis is calculated to. It returns the value 0 if no NCU has been programmed.					
-	0			UWord	r
Multi-line: yes	Number of the compensation table		62		

anCecStep	$AN_CEC_STEP				
The variable indicates the distance of the offset values.					
-	0.0			Double	r
Multi-line: yes	Number of the compensation table		62		

anCecType	$AN_CEC_TYPE				
The variable specifies the compensation table type 0: no special table type 1: cylinder error compensation table type					
-	FALSE			UWord	r
Multi-line: yes	Number of the compensation table		62		

anCollCheckOff	$AN_COLL_CHECK_OFF				
Status of the byte DeactivateCollCheckGroups on the interface PLC->NCK (DB10.DBB58) for the operating-mode-dependant suppression of the collision avoidance for groups of protection areas.					
-	0			Long Integer	r
Multi-line: yes	1		1		

anCollIpoActive	$AN_COLL_IPO_ACTIVE				
The system variable indicates if the main run monitoring of the collision avoidance is active.					
-				UWord	r
Multi-line: no					

anCollIpoLimit	$AN_COLL_IPO_LIMIT				
The system variable indicates if the main run monitoring of the collision avoidance leads to a velocity reduction.					
-				UWord	r
Multi-line: no					

anCollLoad	$AN_COLL_LOAD				
Gives the required calculation time in ms - required for certain operations in connection with collision avoidance. The operation is defined by index i. i = 0: Time requirement for last call of PROTA i = 1: Time requirement for last call of collision avoidance during preprocessing i = 2: Time requirement for last call of the calculation of free space (real-time monitoring) The variables can be reset by describing with value 0. Every write attempt with a value other than 0 is refused with an error message.					
s, user defined				Double	rw
Multi-line: yes	Choice of functions		3		

anCollMemAvailable	$AN_COLL_MEM_AVAILABLE				
Collision calculation requires internal memory, the size of which is either calculated automatically from the number of available protection zones, protection zone elements, facets and the number of machine axes, or it can be explicitly defined by machine data $MN_MM_MAXNUM_3D_COLLISION. The size of the reserved memory area in kbytes can be read with system variable $AN_COLL_MEM_AVAILABLE.					
-	0	0		Double	r
Multi-line: yes	1		1		

anCollMemUseAct	$AN_COLL_MEM_USE_ACT				
Collision calculation requires internal memory, the size of which is either calculated automatically from the number of available protection zones, protection zone elements, facets and the number of machine axes, or it can be explicitly defined by machine data $MN_MM_MAXNUM_3D_COLLISION. The size of the reserved memory area in kbytes can be read with system variable $AN_COLL_MEM_AVAILABLE. The system variable $AN_COLL_MEM_USE_ACT returns the current (that is the last calculated) memory space required for collision calculation as a percentage of the reserved memory area. It can be reset by writing with the value 0. Any attempt to write any other value than 0 is rejected with an error message.					
-	0	0		Double	rw
Multi-line: yes	1		1		

anCollMemUseMax	$AN_COLL_MEM_USE_MAX				
Collision calculation requires internal memory, the size of which is either calculated automatically from the number of available protection zones, protection zone elements, facets and the number of machine axes, or it can be explicitly defined by machine data $MN_MM_MAXNUM_3D_COLLISION. The size of the reserved memory area in kbytes can be read with system variable $AN_COLL_MEM_AVAILABLE. The system variable $AN_COLL_MEM_USE_MAX returns the maximum memory space required for collision calculation as a percentage of the reserved memory area. It can be reset by writing with the value 0. Any attempt to write any other value than 0 is rejected with an error message.					
-	0	0		Double	rw
Multi-line: yes	1		1		

anCollMemUseMin	$AN_COLL_MEM_USE_MIN				
Collision calculation requires internal memory, the size of which is either calculated automatically from the number of available protection zones, protection zone elements, facets and the number of machine axes, or it can be explicitly defined by machine data $MN_MM_MAXNUM_3D_COLLISION. The size of the reserved memory area in kbytes can be read with system variable $AN_COLL_MEM_AVAILABLE. The system variable $AN_COLL_MEM_USE_MIN returns the minimum memory space required for collision calculation as a percentage of the reserved memory area. It can be reset by writing with the value 0. Any attempt to write any other value than 0 is rejected with an error message.					
-	0	0		Double	rw
Multi-line: yes	1		1		

anCollPairsAct	$AN_COLL_PAIRS_ACT				
The collision avoidance function can monitor a maximum number of protection area pairs. This number is determined by machine data 18898 $MN_MM_MAXNUM_3D_COLL_PAIRS. The system variable $AN_COLL_PAIRS_ACT states how many of them are currently being used.					
-	0			Long Integer	r
Multi-line: yes	1		1		

anCollState	$AN_COLL_STATE[i]				
The system variable indicates if a protection area can currently be part of collision monitoring. However, the following requirements must be met first: 1. The activation status of the protection area is active ("A") or the activation status is PLC-controlled ("P") and the interface bit assigned to the protection area is set. 2. The protection area group ("Machine", "TOOL" etc.) has been activated in the current operating mode via the associated interface bit. A protection area for which this system variable gives the value TRUE only then enters real collision monitoring when it is part of at least one collision pair ($NP_COLL_PAIR). The other partner must also be an active protection area.					
-				UWord	r
Multi-line: yes	Number of a protection area		$MN_MM_MAXNUM_3D_PROT_AREAS		

anCollStateCond	$AN_COLL_STATE_COND[i]				
The system variable indicates if a protection area can currently be part of the collision monitoring. The individual conditions which have to be fulfilled so that a protection area can actively prevent collisions are also shown. The variable is coded as follows: Bit 0: The protection area is monitored (this bit has the same significance as the system variable $AN_COLL_STATE). Bit 1: The protection area is included in the internally mapped model. Bit 2: The protection area has the status 'P' (PLC-controlled). Bit 3: The protection area has the status 'A' (active). Bit 4: All axes which can move the protection area are referenced. Bit 5: Indicates whether a PLC bit is assigned to the protection area. Bit 6: Status of the interface bit assigned to the SB. An active protection area (bit 0 = TRUE) only then enters real collision monitoring when it is part of at least one collision pair ($NP_COLL_PAIR). The other partner must also be an active protection area.					
-				Long Integer	r
Multi-line: yes	Number of a protection area		$MN_MM_MAXNUM_3D_PROT_AREAS		

anFacetsAct	$AN_FACETS_ACT				
Machine parts can be modeled from 3D facets for the collision avoidance function. The maximum number of 3D facets is limited by the machine data 18895 $MN_MM_MAXNUM_3D_FACETS. The anFacetsAct variable indicates how many facets are currently being used.					
-	0			Long Integer	r
Multi-line: yes	1		1		

anFacetsAvailable	$AN_FACETS_AVAILABLE				
Machine parts can be modeled from 3D facets for the collision avoidance function. The maximum number of 3D facets is limited by the machine data 18895 $MN_MM_MAXNUM_3D_FACETS. The anFacetsAvailable variable indicates how many facets are still available.					
-	0			Long Integer	r
Multi-line: yes	1		1		

anFacetsInternAct	$AN_FACETS_INTERN_ACT				
Variable machine parts such as tools are automatically modeled from 3D facets by means of the collision avoidance function. The maximum number of 3D facets is limited by the machine data 18894 $MN_MM_MAXNUM_3D_FACETS_INTERN. The anFacetsInternAct variable indicates how many facets are currently being used.					
-	0			Long Integer	r
Multi-line: yes	1		1		

anFacetsInternAvailable	$AN_FACETS_INTERN_AVAILABLE				
Variable machine parts such as tools are automatically modeled from 3D facets by means of the collision avoidance function. The maximum number of 3D facets is limited by the machine data 18894 $MN_MM_MAXNUM_3D_FACETS_INTERN. The anFacetsInternAvailable variable indicates how many facets are still available.					
-	0			Long Integer	r
Multi-line: yes	1		1		

anFacetsInternMax	$AN_FACETS_INTERN_MAX				
Variable machine parts such as tools are automatically modeled from triangle areas by means of the Collision prevention function. The maximum number of triangles is limited by the machine data 18894 $MN_MM_MAXNUM_3D_FACETS_INTERN. The variable anFacetsInternMax indicates how many have been used so far.					
-	0			Long Integer	rw
Multi-line: yes	1		1		

anFacetsInternMin	$AN_FACETS_INTER_MIN				
Variable machine parts such as tools are automatically modeled from 3D facets by means of the collision avoidance function. The maximum number of 3D facets is limited by the machine data 18894 $MN_MM_MAXNUM_3D_FACETS_INTERN. The anFacetsInternMin variable indicates the minimum number of facets used so far.					
-	0			Long Integer	rw
Multi-line: yes	1		1		

anFacetsMax	$AN_FACETS_MAX				
Machine parts can be modeled from 3D facets for the collision avoidance function. The maximum number of 3D facets is limited by the machine data 18895 $MN_MM_MAXNUM_3D_FACETS. The anFacetsMax variable indicates the maximum number of facets used so far.					
-	0			Long Integer	rw
Multi-line: yes	1		1		

anFacetsMin	$AN_FACETS_MIN				
Machine parts can be modeled from 3D facets for the collision avoidance function. The maximum number of 3D facets is limited by the machine data 18895 $MN_MM_MAXNUM_3D_FACETS. The anFacetsMin variable indicates the minimum number of facets used so far.					
-	0			Long Integer	rw
Multi-line: yes	1		1		

anIpoActLoad	$AN_IPO_ACT_LOAD				
Current IPO runtime including the runtime of the synchronized actions of all channels					
-	0	0		Double	r
Multi-line: yes	1		1		

anIpoChanax	$AN_IPO_CHANAX				
For a global axis number returned by valpoNcChanax, the channel and channel axis number that define the writing interpolator of the axis are output. Here the channel is output as from position 100, and the channel axis number as from position 1, e.g. 1005 - channel 10 channel axis 5. If the axis with the specified global axis number is not used on this NCU, 0 will be returned.					
-	0	0		UDoubleword	r
Multi-line: yes	Global axis number as output by valpoNcChanax		170		

anIpoLoadLimit	$AN_IPO_LOAD_LIMIT				
IPO utilization limit reached 0: Utilization limit not reached 1: Utilization limit reached					
-	0	0	1	UWord	r
Multi-line: yes	1		1		

anIpoLoadPercent	$AN_IPO_LOAD_PERCENT				
Ratio of curr. IPO runtime / IPO cycle					
-	0	0		Double	r
Multi-line: yes	1		1		

anIpoMaxLoad	$AN_IPO_MAX_LOAD				
Maximum IPO runtime including the runtime of the synchronized actions of all channels					
-	0	0		Double	r
Multi-line: yes	1		1		

anIpoMinLoad	$AN_IPO_MIN_LOAD				
Minimum IPO runtime including the runtime of the synchronized actions of all channels					
-	0	0		Double	r
Multi-line: yes	1		1		

anKinChainElemAct	$AN_KIN_CHAIN_ELEM_ACT				
The kinematic chains can only use a maximum number of elements. This number is determined by machine data 18880 $MN_MM_MAXNUM_KIN_CHAIN_ELEM. The system variable $AN_KIN_CHAIN_ELEM_ACT states how many of them are currently being used.					
-	0			Long Integer	r
Multi-line: yes	1		1		

anLaiAxIsAxctax	$AN_LAI_AX_IS_AXCTAX				
Bit mask that displays whether an axis in the logical NCK machine axis image (machine data 10002 $MN_AXCONF_LOGIC_MACHAX_TAB) is an axis in an axis container (machine data 1270x/1271x $MN_AXCT_AXCONF_ASSIGN_TABi).					
-	0	0	0xffffffff	UDoubleword	r
Multi-line: yes	1		1		

anLaiAxIsLeadLinkax	$AN_LAI_AX_IS_LEADLINKAX				
Bit mask that displays whether an axis in the logical NCK machine axis image (machine data 10002 $MN_AXCONF_LOGIC_MACHAX_TAB) is a lead link axis, i.e. the same machine axis is referred to on several NCUs via MD10002 $MN_AXCONF_LOGIC_MACHAX_TAB and the axial MD30554 $MA_AXCONF_ASSIGN_MASTER_NCU specifies which NCU is the master NCU, which generates the setpoint value for the position controller following power-up.					
-	0	0	0xffffffff	UDoubleword	r
Multi-line: yes	1		1		

anLaiAxIsLinkax	$AN_LAI_AX_IS_LINKAX				
Bit mask that displays whether an axis in the logical NCK machine axis image (machine data 10002 $MN_AXCONF_LOGIC_MACHAX_TAB) is a link axis (axis physically connected to another NCU).					
-	0	0	0xffffffff	UDoubleword	r
Multi-line: yes	1		1		

anLaiAxToIpoNcChanax	$AN_LAI_AX_TO_IPO_NC_CHANAX				
If the LAI axis is currently interpolated to this NCU, the channel and channel axis number which define the interpolator of the axis are output. If the LAI axis is currently interpolated to a different NCU, the NCU identifier of the interpolated NCU and the global axis number of the machine axis are output. This global axis number can then be used to transfer the interpolated channel and the channel axis number to the other NCU, with NCU ID 2, with $AN_IPO_CHANAX[103]. If no LAI axis is used, 0 is returned. The channel is output as from position 100, and the channel axis number as from position 1, e.g. 1005 - channel 10 channel axis 5. These values are always lower than 10000. Here the NCU is output as from position 10000, e.g. 20103: NCU 2 and the global axis number is 103.					
-	0	0		UDoubleword	r
Multi-line: yes	Number (index + 1) in the logical NCK machine axis image (machine data 10002 $MN_AXCONF_LOGIC_MACHAX_TAB)		>maxnumGlobMachAxes		

anLaiAxToMachax	$AN_LAI_AX_TO_MACHAX				
The NCU and machine axis are output for an LAI axis, representing the physical image of the axis. Here the NCU ID is output as from position 10000, e.g. 20005: NCU 2 axis 5. Without an NCU link, i.e. if there is only one NCU, only the number of the machine axis will be output. In this case, the NCU ID is equal to zero. If the LAI axis is not used, 0 is returned.					
-	0	0		UDoubleword	r
Multi-line: yes	Number (index + 1) in the logical NCK machine axis image (machine data 10002 $MN_AXCONF_LOGIC_MACHAX_TAB)		>maxnumGlobMachAxes		

anLinkCommState	$AN_LINK_COMM_STATE				
Status of the NCU link communication between all NCUs in the NCU link cluster. Decimal values of the variable: 0: NCU link communication is not active (MD18780 $MN_MM_NCU_LINK_MASK) 1: NCU link communication is active (MD18780 $MN_MM_NCU_LINK_MASK) and functions correctly, that is signs of life are received from all NCUs in the cluster 2: NCU link communication is active (MD18780 $MN_MM_NCU_LINK_MASK), but does not function correctly (e.g. start-up with inactive link, communication error etc.)					
-	0			UWord	r
Multi-line: yes	1		1		

anLinkConnRcv	$AN_LINK_CONN_RCV				
Number of link variable changes per cycle from the specified to the current NCU number. The variable $AN_LINK_CONN_RCV[NCU-No] shows the transmission capacity reserved for non-cyclic messages from NCU-No to NCU-Curr in bytes. Systems without a NCU link return the value 0.					
-	0			Long Integer	r
Multi-line: yes	Currently, the index may have a value between 1 and 16		maxNumNcusInNcuCluster		

anLinkConnSizeLinkvar	$AN_LINK_CONN_SIZE_LINKVAR				
Gross number of bytes required for a link variable to be transmitted in each PTP relationship The assignment of a link variable (e.g. $a_dlb[9] = 1) loads the non-cyclic link connections with a message of length $AN_LINK_CONN_SIZE_LINKVAR. It is irrelevant here whether a double-link or a byte-link variable is written. This enables the customer to estimate the maximum number of transmittable link variables per IPO cycle ($AN_LINK_CONN_SND[NCU-No] / $AN_LINK_CONN_SIZE_LINKVAR= number of link-variable changes per IPO cycle from NCU-Curr to NCU-No).					
-	0			Long Integer	r
Multi-line: yes	1		1		

anLinkConnSnd	$AN_LINK_CONN_SND				
Number of link variable changes per cycle from the current to the specified NCU number. The index NCU-No of the variable $AN_LINK_CONN_SDN[NCU-No] currently ranges from 1 to 16. The variable returns the number of bytes reserved from the current NCU-Curr to the NCU-No in order to exchange any non-cyclic messages. Depending on the utilization of this transmission capacity, SIEMENS can supply new SDB blocks for the CBE-30 which reduce the total transmission capacity from NCU-Curr to NCU-No. This makes the link faster and thus the servo cycle shorter. Note: If NCU-Curr == NCU-No the variable returns "0".					
-	0			Long Integer	r
Multi-line: yes	Currently, the index may have a value between 1 and 16		maxNumNcusInNcuCluster		

anLinkTransRateLast	$AN_LINK_TRANS_RATE_LAST				
Number of link variables that should have been sent off in a previous IPO cycle					
-	0			UWord	r
Multi-line: yes	1		1		

anLinkTransRateLastSum	$AN_LINK_TRANS_RATE_LAST_SUM				
Number of link variables for the send direction to the specified NCU number that could have been sent off in a previous IPO cycle.					
-	0			UWord	r
Multi-line: yes	Currently, the index may have a value between 1 and 16		maxNumNcusInNcuCluster		

anPoweronState	$AN_POWERON_STATE				
The bit-coded variable indicates the state of the NCK power on. All bits = 0: NCK power on has not started. Bit0=1: The NCK power on has started, i.e. all NCK objects (channels etc.) have already been created and are being initialized. Bit1=1: The main run states can now be read. This means that all stations have been initialized, and that power on Reset has been executed together with the Reset INIT blocks. Bit2=1: User interventions (Reset, Stop etc.) are now possible and purposeful. This means that any configured Safety ProgEvent has been correctly completed or possibly could not be executed because of alarms. Any configured PowerOn ProgEvent is executed next provided that its execution is not prevented by alarms. Bit24=1: The NCK power on has finished together will all the ProgEvents that could be executed automatically (Safety ProgEvent, PowerOn ProgEvent). The bit does not indicate whether or not an error occurred during the power on (see Bit25). Bit25=1: The NCK power on finished with errors. This means, for example, that an error occurred while the stations were being initialized, during the Reset INIT blocks or the execution of the Safety ProgEvent. Other alarms indicate the exact problem, and the alarm responses indicate which actions can be executed.					
-	0			UDoubleword	r
Multi-line: yes	1		1		

anPrepActLoad	$AN_PREP_ACT_LOAD				
Current preprocessing run time throughout all channels					
-	0	0		Double	r
Multi-line: yes	1		1		

anPrepActLoadGross	$AN_PREP_ACT_LOAD_GROSS				
Current gross preprocessing run time throughout all channels					
-	0	0		Double	r
Multi-line: yes	1		1		

anPrepMaxLoad	$AN_PREP_MAX_LOAD				
Longest preprocessing run time throughout all channels					
-	0	0		Double	r
Multi-line: yes	1		1		

anPrepMaxLoadGross	$AN_PREP_MAX_LOAD_GROSS				
Longest gross preprocessing run time throughout all channels					
-	0	0		Double	r
Multi-line: yes	1		1		

anPrepMinLoad	$AN_PREP_MIN_LOAD				
Shortest preprocessing run time throughout all channels					
-	0	0		Double	r
Multi-line: yes	1		1		

anPrepMinLoadGross	$AN_PREP_MIN_LOAD_GROSS				
Shortest gross preprocessing run time throughout all channels					
-	0	0		Double	r
Multi-line: yes	1		1		

anProtAreaElemAct	$AN_PROT_AREA_ELEM_ACT				
The collision avoidance function can monitor a maximum number of protection area elements. This number is determined by machine data 18892 $MN_MM_MAXNUM_3D_PROT_AREA_ELEM. The system variable $AN_PROT_AREA_ELEM_ACT states how many of them are currently being used.					
-	0			Long Integer	r
Multi-line: yes	1		1		

anProtAreasAct	$AN_PROT_AREAS_ACT				
The collision avoidance function can monitor a maximum number of protection areas. This number is determined by machine data 18890 $MN_MM_MAXNUM_3D_PROT_AREAS. The system variable $AN_PROT_AREAS_ACT states how many of them are currently being used.					
-	0			Long Integer	r
Multi-line: yes	1		1		

anRebootDelayTime	$AN_REBOOT_DELAY_TIME				
Time until reboot					
s, user defined	0	0		Double	r
Multi-line: yes	1		1		

anRobin	$AN_ROBIN[index]				
The system variable $AN_ROBIN[index] reads the relevant byte in the NCK-PLC interface robotic status.					
-	0			UWord	r
Multi-line: yes	Byte no.		8		

anRobout	$AN_ROBOUT[index]				
The system variable $AN_ROBOUT[index] reads the relevant byte in the NCK-PLC robotic control interface.					
-	0			UWord	r
Multi-line: yes	Byte no.		8		

anSLTrace	$AN_SLTRACE				
This variable is reserved for the application SinUTrace and Operate-Trace. It serves as a trigger variable for the logging function. The following coding is recommended: 0: Inactive 1: Start logging requested 2: Stop logging requested The value is generally set by the part program, the reset by the application via OPI.					
-	0			Long Integer	rw
Multi-line: yes	1		1		

anServoActLoad	$AN_SERVO_ACT_LOAD				
Current runtime of the position controller					
-	0	0		Double	r
Multi-line: yes	1		1		

anServoMaxLoad	$AN_SERVO_MAX_LOAD				
Maximum runtime of the position controller					
-	0	0		Double	r
Multi-line: yes	1		1		

anServoMinLoad	$AN_SERVO_MIN_LOAD				
Minimum runtime of the position controller					
-	0	0		Double	r
Multi-line: yes	1		1		

anSimChanMask	$AN_SIM_CHAN_MASK				
Bit-coded mask of channels to be considered in the synchronized multi-channel simulation. The variable is valid only in connection with the synchronized simulation (see Bit4 $MN_PROG_TEST_MASK).					
-	0	0	0x3FF	Long Integer	rw
Multi-line: no					

anSimMaxIpoStep	$AN_SIM_MAX_IPOSTEP				
Using this variable you can specify the maximum step width in real-time IPO cycles. After each step width an event is output to the HMI interface. This allows setting the number of breakpoints. If value 0 is specified, then the system determines the maximum possible step width. The variable is valid only in connection with the synchronized simulation (see Bit4 $MN_PROG_TEST_MASK).					
-	0	0		Long Integer	rw
Multi-line: no					

anSyncActLoad	$AN_SYNC_ACT_LOAD				
Current runtime for synchronized actions					
-	0	0		Double	r
Multi-line: yes	1		1		

anSyncMaxLoad	$AN_SYNC_MAX_LOAD				
Maximum runtime for synchronized actions					
-	0	0		Double	r
Multi-line: yes	1		1		

anSyncToIpo	$AN_SYNC_TO_IPO				
Percentage of Synact / IPO computing time					
-	0	0		Double	r
Multi-line: yes	1		1		

anTProtElemAct	$AN_T_PROT_ELEM_ACT				
The collision avoidance function can monitor a maximum number of tool protection area elements. This number is determined by machine data 18893 $MN_MM_MAXNUM_3D_T_PROT_ELEM. The system variable $AN_T_PROT_ELEM_ACT states how many of them are currently being used.					
-	0			Long Integer	r
Multi-line: yes	1		1		

anTimer	$AN_TIMER[n]				
Global NCK timer in seconds.					
s, user defined	0	0		Double	r
Multi-line: yes	Index in $AN_TIMER[n]		$MN_MM_NUM_AN_TIMER		

anVModelStatus	$AN_VMODEL_STATUS				
System variable for the status of the VRML model 1: MODIFIED_STATE: The model has been modified internally This is the original status. It may change, if machine parameters such as protection zones change. 2: COPIED_STATE: This status is generated outside the NCK, if the model file is ready for being displayed. 3: DISPLAYED_STATE: If the NCK sends the instruction for display of the model in the display program.					
-	1	1	3	Long Integer	rw
Multi-line: yes	1		1		

analogInpVal	$A_INA[x] x = AnaloginputNo				
Value of HW analog input					
A or V				Double	r
Multi-line: yes	Number of analog input		numAnalogInp		

analogOutpVal	$A_OUTA[x] x = AnalogoutputNo				
Number of HW analog output					
A or V				Double	rw
Multi-line: yes	Number of analog output		numAnalogOutp		

axisActivInNcu					
Display indicating whether the axis is active, i.e. whether it can be traversed via a channel of its own NCU or via another NCU (link axis). This data can be utilized by HMIs in order to suppress the display of any non-active axes. Bits 0-31 stand for the axes of the NCU. Bit n = 1: Axis can be traversed. Bit n = 0: Axis cannot be traversed.					
-				Long Integer	r
Multi-line: yes	1		1		

badMemFfs					
Only with 840D-powerline: Number of bytes which are defective in the Flash File System (FFS)					
-	0			Long Integer	r
Multi-line: yes	1		1		

basisFrameMask	$P_NCBFRMASK				
Display indicating which channel-independent basic frames are active. Every bit in the mask indicates whether the appropriate basic frame is active. Bit0 = 1st basic frame, Bit1 = 2nd basic frame, etc.					
-				UWord	r
Multi-line: yes	1		1		

checkSumForAcxData					
Current fingerprint of selected ACX data, in order to check relatively quickly whether ACX data has changed. The data required for the OPI access is located at the start of the downloaded ACX file.					
-				String [32]	r
Multi-line: yes	1: _N_NC_TEA_ACX 2: _N_CH_TEA_ACX 3: _N_AX_TEA_ACX 4: _N_NC_SEA_ACX 5: _N_CH_SEA_ACX 6: _N_AX_SEA_ACX 7: _N_NC_GD1_ACX 8: _N_NC_GD2_ACX 9: _N_NC_GD3_ACX 10: _N_NC_GD4_ACX 11: _N_NC_GD5_ACX 12: _N_NC_GD6_ACX 13: _N_NC_GD7_ACX 14: _N_NC_GD8_ACX 15: _N_NC_GD9_ACX 16: _N_CH_GD1_ACX 17: _N_CH_GD2_ACX 18: _N_CH_GD3_ACX 19: _N_CH_GD4_ACX 20: _N_CH_GD5_ACX 21: _N_CH_GD6_ACX 22: _N_CH_GD7_ACX 23: _N_CH_GD8_ACX 24: _N_CH_GD9_ACX 25: _N_NC_FUN_ACX 26: _N_NC_GCD_ACX 27: _N_NC_NCN_ACX 28: _N_NC_SYD_ACX 29: _N_CH_SYD_ACX 30: _N_AX_SYD_ACX 31: _N_NC_KYW_ACX 32: _N_NC_MAC_ACX 33: _N_NC_GCI_ACX		32		

completeDocAcxChangeCnt					
Modification counter of ACX for the configuration of DO of all SINAMICS on all PROFIBUS segments (_N_COMPLETE_DOC_ACX) that is incremented when the ACX is changed. If the contents of ACX is or becomes invalid, the modification counter will be set to 0. If the contents of ACX is valid again, the modification counter will be reset to the value it had before the contents of ACX became invalid, and will simultaneously be incremented (only the value), if the contents of ACX has really changed. == 0: Contents of _N_COMPLETE_DOC_ACX is invalid != 0: Contents of _N_COMPLETE_DOC_ACX is valid					
-	0	0		UWord	r
Multi-line: no			1		

completeDotAcxChangeCnt					
Modification counter of ACX that describes all SINAMICS DO types known to the OPI (_N_COMPLETE_DOT_ACX) and that is incremented when ACX changes. If the contents of ACX is or becomes invalid, the modification counter will be set to 0. If the contents of ACX is valid again, the modification counter will be reset to the value it had before the contents of ACX became invalid and will be incremented (only the value) simultaneously, if the contents of ACX has really changed. == 0: Contents of _N_COMPLETE_DOT_ACX is invalid != 0: Contents of _N_COMPLETE_DOT_ACX is valid					
-	0	0		UWord	r
Multi-line: no			1		

completeDpcAcxChangeCnt					
Modification counter of ACX for the PROFIBUS configuration of all PROFIBUS segments (_N_COMPLETE_DPC_ACX) that is incremented when ACX is changed. If the contents of ACX is or becomes invalid, the modification counter will be set to 0. If the contents of ACX is valid again, the modification counter will be reset to the value it had before the contents of ACX became invalid, and will simultaneously be incremented (only the value), if the contents of ACX has really changed. == 0: Contents of _N_COMPLETE_DPC_ACX is invalid != 0: Contents of _N_COMPLETE_DPC_ACX is valid					
-	0	0		UWord	r
Multi-line: no			1		

diagnoseDataFfs					
Only with 840D-powerline: Diagnostic data for Flash File System (FFS)					
-	0			Double	r
Multi-line: yes	1: realspace (bytes) 2: formspace (bytes) 3: freespace (%) 4: delspace (%) 5: badspace (%) 6: actlowwater (%) 7: lowwater (%) 8: reorgmode (%)		8		

digitInpVal	$A_IN[x] x = DigitalinputNo				
Value of HW digital input 0 = low 1 = high					
-				UWord	r
Multi-line: yes	Number of digital input		numDigitInp		

digitOutpVal	$A_OUT[x] x = DigitaloutputNo				
Value of HW digital output 0 = low 1 = high					
-				UWord	rw
Multi-line: yes	Number of digital output		numDigitOutp		

driveType					
Digital drive type. Coded according to machine data 13040, but additional code. Note: As long as the OPI variable contains the identifier 0x100 "Drive type unknown" after an NCK ramp-up, the information is not yet consistent and must not be evaluated. As soon as the identifier 0x100 is deleted, in NCU systems with SIMODRIVE 611D drives it can be assumed that the content can only change after renewed link to the NCK. (e.g. after modification of the drive modules), i.e. it need not be cyclically checked for change. 0x100: Drive type unknown. 0x200: This identifier is entered in addition to the code according to the machine data 13040 if a 611D-Performance2 module is detected. For other codes, see MD 13040.					
-	0	0		UWord	r
Multi-line: no			maxnumDrives		

driveTypeChangeCnt					
This counter is incremented by 1 every time driveType is modified. The next value after 65535 is 0.					
-	0	0		UWord	r
Multi-line: no			1		

freeDirectorys					
Number of directories that can be created					
-				UWord	r
Multi-line: yes	1		1		

freeFiles					
Number of files that can be created					
-				UWord	r
Multi-line: yes	Type of memory: 1: MMF (Solutionline) / SRAM (Powerline) 2: DRAM 3: MMF 4: SRAM		4		

freeMem					
Free SRAM in bytes					
-				Long Integer	r
Multi-line: yes	1		1		

freeMemDram					
Free memory in bytes					
-				Long Integer	r
Multi-line: yes	1		1		

freeMemDramEPassF					
Memory in bytes available in the passive file system for executing from external drives					
-	0	0		Long Integer	r
Multi-line: yes	1		1		

freeMemDramMPassF					
Free memory of the passive file system of the "Machine manufacturer" area in bytes					
-	0	0		Long Integer	r
Multi-line: yes	1		1		

freeMemDramPassF					
Memory available in passive file system (DRAM no. 1) in bytes					
-	0	0		Long Integer	r
Multi-line: yes	1		1		

freeMemDramSPassF					
Free memory of the passive file system of the "Control manufacturer" area in bytes					
-	0	0		Long Integer	r
Multi-line: yes	1		1		

freeMemDramTPassF					
Free memory of the passive file system of the "Temp" area in bytes					
-	0	0		Long Integer	r
Multi-line: yes	1		1		

freeMemDramUPassF					
Free memory of the passive file system of the "User" area in bytes					
-	0	0		Long Integer	r
Multi-line: yes	1		1		

freeMemFfs					
Only with 840D-powerline: Number of bytes that are still available in the Flash File System (FFS)					
-	0			Long Integer	r
Multi-line: yes	1		1		

freeMemISram					
Free internal SRAM					
-	0	0		Long Integer	r
Multi-line: yes	1		1		

freeMemSettings					
Free SRAM for the part programs and persistent data for the present (possibly not yet active) memory layout. It is updated when memory-configuring machine data is changed. The size of the current file system is taken into account.					
-	0	0		Long Integer	r
Multi-line: yes	1		1		

freeMemSettingsDram					
Free DRAM memory for data for the present (may not yet be active) memory layout. Is updated when memory- configuring machine data is changed.					
-	0	0		Long Integer	r
Multi-line: yes	1		1		

freeMemSettingsISram					
Free internal SRAM for the current (possibly not yet active) memory layout. It is updated when the memory-configuring machine data is updated.					
-	0	0		Long Integer	r
Multi-line: yes	1		1		

freeMemSramPassF					
Memory available in passive file system (SRAM) in bytes					
-	0	0		Long Integer	r
Multi-line: yes	1		1		

freeMemWarrant					
Guaranteed free memory for part programs and persistent data (from catalog NC60)					
-	0	0		Long Integer	r
Multi-line: yes	1		1		

freeMemWarrantDram					
Guaranteed memory (DRAM)					
-	0	0		Long Integer	r
Multi-line: yes	1		1		

freeProtokolFiles					
Logging: Number of protocol files that can still be created					
-	0	0	1	UWord	r
Multi-line: yes	User No. (1-10)		10		

fsInfoAllChangeCounter					
Total change counter fsInfoPathName					
-				UDoubleword	r
Multi-line: yes	No. of info object		fsInfoCount		

fsInfoChangeCounter					
Content change counter fsInfoPathName					
-				UDoubleword	r
Multi-line: yes	No. of info object		fsInfoCount		

fsInfoChangeDateTime					
Change time of the file system object fsInfoPathName.					
-				String [13]	r
Multi-line: yes	No. of info object		fsInfoCount		

fsInfoChangeDateTimeSub					
Change time of the files contained in the fsInfoPathName directory.					
-				String [13]	r
Multi-line: yes	No. of info object		fsInfoCount		

fsInfoCount					
Number of file system info objects					
-				UWord	r
Multi-line: yes	1		1		

fsInfoFileLength					
Length of the file system object fsInfoPathName.					
-				UDoubleword	r
Multi-line: yes	No. of info object		fsInfoCount		

fsInfoObjStatus					
Status of file system object fsInfoPathName. Bit-coded (may be added to later): Bit0 = 0: Object not loaded in the NCK. Bit0 = 1: Object loaded in the NCK. Bit1 = 0: Object is a file. Bit1 = 1: Object is a directory					
-		0	3	UWord	r
Multi-line: yes	No. of info object		fsInfoCount		

fsInfoPartition					
Partition and lifetime of the file fsInfoPathName SRP: SRAM persistent USV: DRAM User volatile USP: DRAM User persistent SIP: DRAM Siemens persistent MAV: DRAM Manufacturer volatile MAP: DRAM Manufacturer persistent TMV: DRAM Temporary volatile D1V: DRAM System 1 volatile EXV: DRAM External volatile EXP: DRAM External persistent					
-				String [160]	r
Multi-line: yes	No. of info object		fsInfoCount		

fsInfoPathName					
Name of a file or directory to be observed					
-				String [160]	rw
Multi-line: yes	No. of info object		fsInfoCount		

fsInfoRights					
Access protection and lifetime of file fsInfoPathName 0-7 ASCII-coded for read / write / execute / show / delete rights					
-	"77777"			String [6]	r
Multi-line: yes	No. of info object		fsInfoCount		

fsInfoSeekw					
First line enabled for modification in part program fsInfoPathName					
-				UDoubleword	r
Multi-line: yes	No. of info object		fsInfoCount		

fsInfoUsed					
Assignment of file system info object. The client first searches for a free info object by reading the list fsInfoUsed. This is assigned by writing fsInfoUsed with 1. If there is no error in the assignment, the desired file or directory can be selected by writing the fsInfoPathName. The information about this object can then be read from the other variables. 0: Info object free. It must be explicitly released by writing 0. 1: Info object assigned. If an assigned object is assigned again then this is acknowledged negatively.					
-		0	1	UWord	rw
Multi-line: yes	No. of info object		fsInfoCount		

handWheelTestDiffPulses					
Define differential handwheel pulses for handwheel simulation via OPI					
-				Long Integer	rw
Multi-line: yes	Handwheel number		numHandWheels		

handwheelStatus					
Handwheel status 0 = PASSIVE 1 = ACTIVE					
-				UWord	rw
Multi-line: yes	Number of handwheel		numHandWheels		

hwMLFB					
MLFB of the NCU module					
-				String [24]	r
Multi-line: yes	1		1		

hwProductSerialNr					
Unique hardware number of the NCU module With solution line, this is the serial number of the CF card.					
-				String [16]	r
Multi-line: yes	1		1		

hwProductSerialNrL					
Unique hardware number of the NCU module					
-				String [32]	r
Multi-line: yes	1		1		

licenseKeyInputCount					
Statement of how often the license key can still be entered > 0: LicenseKey can still be entered x times = 0: LicenseKey can no longer be entered. An NCK Power On is required before a new entry can be made.					
-	3	0	3	UWord	r
Multi-line: yes	1		1		

licenseStatus					
Licensing status 0: Licensed, 1: Insufficiently licensed 2: Not licensed 3: PIN is expanded 4: PIN is OK 5: PIN entered incorrectly 6: PIN missing 7: Trial license active 8: Trial license expired					
-	0			UWord	r
Multi-line: yes	1		1		

measFctCmd					
Start of motion measuring function 0 = Function not active or abort 1 = Activate start of motion for all system axes					
-		0	1	UWord	rw
Multi-line: yes	1		1		

mmcCmdPrepCounter					
Counter that is incremented with each call of EXTCALL					
-				UWord	r
Multi-line: yes	1		1		

nckAliveAndWell	DB10, DBX104.7				A4
NCK sign-of-life This value is incremented every time the variable is read, which means that an HMI can determine whether the NCK is still operating correctly by reading the variable cyclically. The value itself has no meaning. Cyclic result acknowledgements in relation to this variable are generated even if the NCK is otherwise no longer operating cyclic services owing to problems with block cycle times. However, this response can be guaranteed only if the variable is not mixed with others in one request, i.e. nckAliveAndWell must be the only variable linked to the cluster. As long as a cyclic read service is set for this variable, one of the HMI-CPU-Ready signals is set in the PLC interface. Which of the signals is set is determined on the one hand by the line number and, on the other hand, by the client's "gloports": The following applies to powerline: An HMI communicates with an MPI via the gloports 0x20-0x2f --> DB10.DBX108 bit2 is set An HMI communicates with an OPI via the gloports 0x10-0x1f --> DB10.DBX108 bit3 is set In the n:m grouping, the 2nd HMI identifies itself by line=2 -> DB10.DBX108 bit1 is set The following applies to solution line: HMIs (int./ext.) communicate via the gloports 0x10-0x17 -> DB10.DBX108 bit3 is set Reserved for later expansions: DB10.DBX108 bit1 Reserved for later expansions: DB10.DBX108 bit2 Note: The related NCK-CPU-Ready signal is stored in DB10,DBX104.7.					
-				UWord	r
Multi-line: yes	HMI No.		2		

nckMode					
Mode in which the NCK works. The mode can be set with the PI _N_NCKMOD. Bit0: NCK works accelerated in simulation mode/DRY_RUN. This mode is currently provided only for the VNCK. Bit1: NCK is slowed down to give the simulation more computing time on the same processor. NC start not possible. Bit2: PowerOn-Init_Finish; power-up initialization finished. Meaning: State = 1 All channels of the NCK have been powered up and have interpreted the DEF files. Thus the HMI may receive consistent data from the NCK. Example of a call from HMI to NCK: What macros are available? What GUDs are available? Note: With Init-Finish==1, the POWER-ON Progevent has NOT yet been run. State = 0 Power up has not yet been finished, or initialization could not been executed due to a fatal alarm Bit3: PowerOn-Ready ; power-up finished Meaning: State == 1 NCK has finished initialization, AND the POWER-ON Progevent has been executed. OR the POWER-ON Progevent could not be executed due to an alarm. Note: The next RESET will "catch up" on the POWER-ON Progevent. This will no longer influence PowerOn-Ready. Without PowerOn Progevent bit 3 and bit 2 are the same. Bit4: NCK works accelerated in simulation mode. This mode is not currently provided for the VNCK.					
-	0	0	f	UWord	r
Multi-line: yes	1		1		

nckModeAccFact					
NCK acceleration factor. The acceleration factor can be set with the PI _N_NCKMOD. The NCK executes the programs in SERUPRO mode. This mode is currently provided only for the VNCK. 0 VNCK executes a program at normal speed. >0 VNCK executes a program at accelerated speed. nckModeAccFact defines the acceleration factor.					
-	0	0		Double	r
Multi-line: yes	1		1		

ncuLinkActive					
Display indicating whether NCU link is activated (via machine data setting) Based on display, HMI decides whether link-specific calculations and displays are required. 0: NCU link not activated 1: NCU link activated					
-	0	0	1	UWord	r
Multi-line: yes	1		1		

nettoMemFfs					
Only with 840D-powerline: Net number of bytes which are available for the Flash File System (FFS). This memory stores the files contents and management data (e.g. file names).					
-	0			Long Integer	r
Multi-line: yes	1		1		

numAlarms					
Number of pending general alarms					
-				UWord	r
Multi-line: no					

numFilesPerDir					
Maximum number of files per directory (see: $MN_MM_NUM_FILES_PER_DIR)					
-				UWord	r
Multi-line: yes	1		1		

numSubDirsPerDir					
Maximum number of subdirectories per directory see: $MN_MM_NUM_SUBDIR_PER_DIR					
-				UWord	r
Multi-line: yes	1		1		

numTraceProtocDataList	$MM_PROTOC_NUM_ETPD_STD_LIST				
Logging: Number of standard data lists per user					
-		0		UWord	r
Multi-line: yes	User No. (1-10)		10		

numTraceProtocOemDataList	$MM_PROTOC_NUM_ETPD_OEM_LIST				
Logging: Number of OEM data lists per user					
-	0	0		UWord	r
Multi-line: yes	User No. (1-10)		10		

passFChangeCounter					
Counter is incremented by 1 when there is a change in the passive file system (but not for changes to the FFS)					
-	0	0		UWord	r
Multi-line: yes	1		1		

pnRobin	$PN_ROBIN[index]				
The system variable $PN_ROBIN[index] reads the relevant byte in the NCK-PLC interface robotic status.					
-	0			UWord	r
Multi-line: yes	Byte no.		8		

pnRobout	$PN_ROBOUT[index]				
The system variable $PN_ROBOUT[index] reads the relevant byte in the NCK-PLC robotic control interface.					
-	0			UWord	r
Multi-line: yes	Byte no.		8		

protCnfgAutoLoad					
Protokoll.: Controls automatic load from trace session from description file during the NCK Start up process 0: Function inactive 1: Logging session should be loaded automatically from a description file during NCK ramp-up 2: Same as (1), but automatic loading is terminated by stop trigger 3: Same as (1), but the description file is updated at each status change					
-	0	0	3	UWord	rw
Multi-line: yes	User no. (1-10)		10		

protCnfgAutoLoadFile					
Log: Directory path and file name from which the trace session is to be loaded during NCK start-up					
-				String [64]	rw
Multi-line: yes	User no. (1-10)		10		

protCnfgAutoSave					
Log: Configuration of the backup of the trace session 0: Function inactive 1: Automatically backs up the logging session in a description file when logging is terminated. 2: Automatically backs up the logging session and diagnostic information in a description file when logging is terminated.					
-	0	0	2	UWord	rw
Multi-line: yes	User no. (1-10)		10		

protCnfgCtl					
Log: Control word to manipulate the trace session description file 0: Function inactive 1: Backs up the logging session in a description file 2: Backs up the logging session and diagnostic information in a description file 3: Loads the logging session from a description file, and all active triggers are deactivated 4: Loads the logging session from a description file 5: Deletes the logging file					
-	0	0	5	UWord	rw
Multi-line: yes	User no. (1-10)		10		

protCnfgFilename					
Log: Name of session backup file					
-				String [64]	rw
Multi-line: yes	User no. (1-10)		10		

protCnfgStat					
Log: Result of the most recent backup or loading of a session backup file 0: No Error					
-	0			UWord	rw
Multi-line: yes	User no. (1-10)		10		

protSessAccR					
Logging: Access rights of the session					
-				String [32]	rw
Multi-line: yes	User No. (1-10)		10		

protSessComm					
Logging: Comments on session					
-				String [128]	rw
Multi-line: yes	User No. (1-10)		10		

protSessConn					
Logging: Connection of the session					
-				String [32]	rw
Multi-line: yes	User No. (1-10)		10		

protSessName					
Logging: Name of the session					
-				String [32]	rw
Multi-line: yes	User No. (1-10)		10		

protSessPrior					
Logging: Priority of the session					
-				String [32]	rw
Multi-line: yes	User No. (1-10)		10		

protocLastValNetIpoCycle					
Logging: Runtime of all events of all channels of a user in the last IPO cycle					
-	0	0		Double	r
Multi-line: yes	User No. (1-10)		10		

protocMaxValNetIpoCycle					
Logging: Maximum run time of all events of all channels of a user					
-	0	0		Double	rw
Multi-line: yes	User No. (1-10)		10		

protocStrtMaskInt16					
Logging: Integer 16 bit screen form with which the start trigger variable is logically AND-ed before the comparison is made with the trigger value. There is no logic operation with the value 0.					
-	0	0		UWord	rw
Multi-line: yes	low byte: user no. (1-10); high byte: variable index (0-1)		10		

protocStrtMaskInt32					
Logging: Integer 32 bit screen form with which the start trigger variable is logically AND-ed before the comparison is made with the trigger value. There is no logic operation with the value 0.					
-	0	0		Long Integer	rw
Multi-line: yes	low byte: user no. (1-10); high byte: variable index (0-1)		10		

protocStrtMatchCount					
Logging: Specifies how often the comparison must match before the start trigger fires. The trigger will only fire, if all trigger conditions are fulfilled.					
-	0	0		UWord	rw
Multi-line: yes	User No. (1-10)		10		

protocStrtNumEvDelay					
Logging: Number of events which are still to be omitted after the occurrence of the trigger event before logging is started.					
-	0	0		UWord	rw
Multi-line: yes	User No. (1-10)		10		

protocStrtOperation					
Logging: There are two start trigger variables. Each of these two variables is monitored using protocTrigType. The result from each of these monitoring actions is: Trigger: yes/no Since there are two variables, there are also two results, which can now be combined with a logic operator. This operator is specified with this variable 0: No logic operation, only the first variable is to be taken into consideration 1: NOT (unary, the result of the first variable is negated, no second variable 2: AND (the result of the first variable is combined with the result of the second variable with logic AND 3: OR (the result of the first variable is combined with the result of the second variable with logic OR 4: XOR (the result of the first variable is combined with the result of the second variable with logic XOR					
-	0	0		UWord	rw
Multi-line: yes	User No. (1-10)		10		

protocStrtRemMatchCount					
Logging: Specifies how often the comparison must match before the start trigger fires. The trigger will only fire, if all trigger conditions are fulfilled.					
-	0	0		UWord	r
Multi-line: yes	User No. (1-10)		10		

protocStrtState					
Logging: Status of the start triggering 0: Passive (trigger inactive) 1: Active (trigger is active, but has not yet responded) 2: Delay (trigger has responded and is still waiting the delay time) 3: Firing (trigger has responded, but must still respond more frequently until the triggering takes place) 4: Done (trigger has responded and is inactive)					
-	0	0		UWord	rw
Multi-line: yes	User No. (1-10)		10		

protocStrtType					
Logging: Type of start triggering 0: Monitoring for equality 1: Monitoring for more than or equal to 2: Monitoring for greater than 3: Monitoring for less than or equal to 4: Monitoring for less than 5: Monitoring for inequality 6: Monitoring for value change 7: Monitoring for increasing values 8: Monitoring for falling values					
-	0	0		UWord	rw
Multi-line: yes	low byte: user no. (1-10); high byte: variable index (0-1)		10		

protocStrtValueInt16					
Logging: Integer 16 bit value with which the start trigger variable is to be compared					
-	0	0		UWord	rw
Multi-line: yes	low byte: user no. (1-10); high byte: variable index (0-1)		10		

protocStrtValueInt32					
Logging: Integer 32 bit value with which the start trigger variable is to be compared					
-	0	0		Long Integer	rw
Multi-line: yes	low byte: user no. (1-10); high byte: variable index (0-1)		10		

protocStrtValueReal32					
Logging: Real 32 bit value with which the start trigger variable is to be compared					
-	0	0		Float	rw
Multi-line: yes	low byte: user no. (1-10); high byte: variable index (0-1)		10		

protocStrtValueReal64					
Logging: Real 64 bit value with which the start trigger variable is to be compared					
-	0	0		Double	rw
Multi-line: yes	low byte: user no. (1-10); high byte: variable index (0-1)		10		

protocStrtVarArea					
Logging: Variable which is to be monitored for start triggering.					
-	0	0		UWord	rw
Multi-line: yes	low byte: user no. (1-10); high byte: variable index (0-1)		10		

protocStrtVarCol					
Logging: Variable which is to be monitored for the start triggering. Statement of the "Col"					
-	0	0		UWord	rw
Multi-line: yes	low byte: user no. (1-10); high byte: variable index (0-1)		10		

protocStrtVarRow					
Logging: Variable which is to be monitored for the start triggering. Statement of the "Row"					
-	0	0		UWord	rw
Multi-line: yes	low byte: user no. (1-10); high byte: variable index (0-1)		10		

protocStrtVarType					
Logging: Variable which is to be monitored for the start triggering. Statement of the "Type"					
-	0	0		UWord	rw
Multi-line: yes	low byte: user no. (1-10); high byte: variable index (0-1)		10		

protocStrtVarUnit					
Logging: Variable which is to be monitored for the start triggering. Statement of the "Unit".					
-	0	0		UWord	rw
Multi-line: yes	low byte: user no. (1-10); high byte: variable index (0-1)		10		

protocTrigMaskInt16					
Logging: Integer 16-bit mask with which the trigger variable is logically ANDed before the comparison with the trigger value is made. Variable is not ANDed if value is 0.					
-	0	0		UWord	rw
Multi-line: yes	low byte: user no. (1-10); high byte: variable index (0-1)		10		

protocTrigMaskInt32					
Logging: Integer 32-bit mask with which the trigger variable is logically ANDed before the comparison with the trigger value is made. Variable is not ANDed if value is 0.					
-	0	0		Long Integer	rw
Multi-line: yes	low byte: user no. (1-10); high byte: variable index (0-1)		10		

protocTrigMatchCount					
Logging: Specifies how often the comparison in total must match before the trigger fires. The trigger will only fire, if all trigger conditions are fulfilled.					
-	0	0		UWord	rw
Multi-line: yes	User No. (1-10)		10		

protocTrigNumEvDelay					
Logging: Number of events to be recorded after the trigger event has occurred before the logging operation is stopped.					
-	0	0		UWord	rw
Multi-line: yes	User No. (1-10)		10		

protocTrigOperation					
Logging: There are two start trigger variables. Each of these two variables is monitored using protocTrigType. The result from each of these monitoring actions is: Trigger: yes/no Since there are two variables, there are also two results, which can now be combined with a logic operator. This operator is specified with this variable 0: No logic operation, only the first variable is to be taken into consideration 1: NOT (unary, the result of the first variable is negated, no second variable 2: AND (the result of the first variable is combined with the result of the second variable with logic AND 3: OR (the result of the first variable is combined with the result of the second variable with logic OR 4: XOR (the result of the first variable is combined with the result of the second variable with logic XOR					
-	0	0		UWord	rw
Multi-line: yes	User No. (1-10)		10		

protocTrigRemMatchCount					
Logging: Specifies how often the comparison must match before the trigger fires. The trigger will only fire, if all trigger conditions are fulfilled.					
-	0	0		UWord	r
Multi-line: yes	User No. (1-10)		10		

protocTrigState					
Logging: Triggering status 0: Passive (trigger inactive) 1: Active (trigger is active, but has not yet responded) 2: Delay (trigger has responded and is still waiting the delay time) 3: Firing (trigger has responded, but must still respond more frequently until the triggering takes place) 4: Done (trigger has responded and is inactive)					
-	0	0		UWord	rw
Multi-line: yes	User No. (1-10)		10		

protocTrigType					
Logging: Triggering method 0: Monitoring for equality 1: Monitoring for more than or equal to 2: Monitoring for greater than 3: Monitoring for less than or equal to 4: Monitoring for less than 5: Monitoring for inequality 6: Monitoring for value change 7: Monitoring for increasing values 8: Monitoring for falling values					
-	0	0		UWord	rw
Multi-line: yes	low byte: user no. (1-10); high byte: variable index (0-1)		10		

protocTrigValueInt16					
Logging: Integer 16-bit value with which trigger variable must be compared					
-	0	0		UWord	rw
Multi-line: yes	low byte: user no. (1-10); high byte: variable index (0-1)		10		

protocTrigValueInt32					
Logging: Integer 32-bit value with which trigger variable must be compared					
-	0	0		Long Integer	rw
Multi-line: yes	low byte: user no. (1-10); high byte: variable index (0-1)		10		

protocTrigValueReal32					
Logging: Real 32-bit value with which trigger variable must be compared					
-	0	0		Float	rw
Multi-line: yes	low byte: user no. (1-10); high byte: variable index (0-1)		10		

protocTrigValueReal64					
Logging: Real 64-bit value with which trigger variable must be compared					
-	0	0		Double	rw
Multi-line: yes	low byte: user no. (1-10); high byte: variable index (0-1)		10		

protocTrigVarArea					
Logging: Variable which is to be monitored for the start triggering. Specification of "Area".					
-	0	0		UWord	rw
Multi-line: yes	low byte: user no. (1-10); high byte: variable index (0-1)		10		

protocTrigVarCol					
Logging: Variable to be monitored for triggering. Specification of "Col".					
-	0	0		UWord	rw
Multi-line: yes	low byte: user no. (1-10); high byte: variable index (0-1)		10		

protocTrigVarRow					
Logging: Variable to be monitored for triggering. Specification of "Row".					
-	0	0		UWord	rw
Multi-line: yes	low byte: user no. (1-10); high byte: variable index (0-1)		10		

protocTrigVarType					
Logging: Variable to be monitored for triggering. Specification of "Type".					
-	0	0		UWord	rw
Multi-line: yes	low byte: user no. (1-10); high byte: variable index (0-1)		10		

protocTrigVarUnit					
Logging: Variable to be monitored for triggering. Specification of "Unit".					
-	0	0		UWord	rw
Multi-line: yes	low byte: user no. (1-10); high byte: variable index (0-1)		10		

safeExtInpValNckBit	$A_INSE[n]				
External NCK input of the SI programmable logic from the NCK periphery					
-	0	0	1	UWord	r
Multi-line: yes	Input number		safeMaxNumExtInput		

safeExtInpValNckWord	$A_INSED[n]				
Image of the external NCK inputs of the SI programmable logic					
-	0			Long Integer	r
Multi-line: yes	1: image of the system variables $A_INSED[1] 2: image of the system variables $A_INSED[2] 3: image of the system variables $A_INSED[3] 4: image of the system variables $A_INSED[4] 5: image of the system variables $A_INSED[5] 6: image of the system variables $A_INSED[6]		safeMaxNumExtInput / 32		

safeExtInpValPlcBit	$A_INSEP[n]				
External PLC input of the SI programmable logic from the PLC periphery					
-	0	0	1	UWord	r
Multi-line: yes	Input number		safeMaxNumExtInput		

safeExtInpValPlcWord	$A_INSEPD[n]				
Image of the external PLC inputs of the SI programmable logic					
-	0			Long Integer	r
Multi-line: yes	1: image of the system variables $A_INSEPD[1] 2: image of the system variables $A_INSEPD[2] 3: image of the system variables $A_INSEPD[3] 4: image of the system variables $A_INSEPD[4] 5: image of the system variables $A_INSEPD[5] 6: image of the system variables $A_INSEPD[6]		safeMaxNumExtInput / 32		

safeExtInputQuality					
Property of an external NCK-SPL input signal					
-	0	0	3	UWord	r
Multi-line: yes	Input number		safeMaxNumExtInput		

safeExtOutpValNckBit	$A_OUTSE[n]				
External NCK output of the SI programmable logic to the NCK periphery					
-	0	0	1	UWord	r
Multi-line: yes	Output number		safeMaxNumExtOutput		

safeExtOutpValNckWord	$A_OUTSED[n]				
Image of the external NCK outputs of the SI programmable logic					
-	0			Long Integer	r
Multi-line: yes	1: image of the system variables $A_OUTSED[1] 2: image of the system variables $A_OUTSED[2] 3: image of the system variables $A_OUTSED[3] 4: image of the system variables $A_OUTSED[4] 5: image of the system variables $A_OUTSED[5] 6: image of the system variables $A_OUTSED[6]		safeMaxNumExtOutput / 32		

safeExtOutpValPlcBit	$A_OUTSEP[n]				
External PLC output of the SI programmable logic to the PLC periphery					
-	0	0	1	UWord	r
Multi-line: yes	Output number		safeMaxNumExtOutput		

safeExtOutpValPlcWord	$A_OUTSEPD[n]				
Image of the external PLC outputs of the SI programmable logic					
-	0			Long Integer	r
Multi-line: yes	1: image of the system variables $A_OUTSEPD[1] 2: image of the system variables $A_OUTSEPD[2] 3: image of the system variables $A_OUTSEPD[3] 4: image of the system variables $A_OUTSEPD[4] 5: image of the system variables $A_OUTSEPD[5] 6: image of the system variables $A_OUTSEPD[6]		safeMaxNumExtOutput / 32		

safeExtOutputQuality					
Property of an external NCK-SPL output signal					
-	0	0	3	UWord	r
Multi-line: yes	Output number		safeMaxNumExtOutput		

safeFdpActCycle					
Current value of the F_DP communication cycle in seconds					
s, user defined	0.0			Double	r
Multi-line: yes	1		1		

safeFdpMaxCycle					
Maximum value of F_DP communication cycle in s					
s, user defined	0.0			Double	r
Multi-line: yes	1		1		

safeFrdpAckReqNck	$A_FRDP_ACK_REQ[n]				
After a communication error, the F_DP communication is back in cyclic operation. A user acknowledgement is required to release normal operation with output of the process values. 0 = No user acknowledgement required 1 = User acknowledgement required					
-	0	0	1	UWord	r
Multi-line: yes	3		16		

safeFrdpActComTime					
Current F_RECVDP communication time in seconds					
s, user defined	0.0			Double	r
Multi-line: yes	3		16		

safeFrdpDiagNck	$A_FRDP_DIAG[n]				
Diagnostics data for F_RECVDP communication/system error 10H = Timeout detected (TO) 20H = Sequence number error detected (SN) 40H = CRC error detected (CRC) 2000H = Deviations in the F telegram data detected (TD) 4000H = Sign-of-life monitoring error detected (LS) 8000H = Asynchronous error state detected (SF)					
-	0	0	0xFFFFFFFF	UDoubleword	r
Multi-line: yes	3		16		

safeFrdpDriverStateNck					
Current state of the F_RECVDP driver 0 = Not parameterized 1 = Initialization 2 = F_RECVDP ready: waiting for F_SENDDP 3 = F_SENDDP ready: waiting for sequence number = 1 4 = F_SENDDP and F_RECVDP ready: waiting for user acknowledgement after error 5 = Normal operation					
-	0	0	5	UWord	r
Multi-line: yes	3		16		

safeFrdpErrReacNck	$A_FRDP_ERR_REAC[n]				
The user can define the error reaction irrespective of the machining situation or the coupling of the communication partners. 0 = Alarm 27350 + stop D/E 1 = Alarm 27350 2 = Alarm 27351 (display only, self-clearing) 3 = No reaction					
-	0	0	3	UDoubleword	r
Multi-line: yes	3		16		

safeFrdpErrReacPlc					
The user can define the error reaction irrespective of the machining situation or the coupling of the communication partners. 0 = Alarm 27350 + stop D/E 1 = Alarm 27350 2 = Alarm 27351 (display only, self-clearing) 3 = No reaction					
-	0	0	3	UDoubleword	r
Multi-line: yes	3		16		

safeFrdpErrorNck	$A_FRDP_ERROR[n]				
An F_RECVDP communication error has been detected. The cause is stated in the diagnostics data. 0 = No communication error 1 = Communication error detected					
-	0	0	1	UWord	r
Multi-line: yes	3		16		

safeFrdpFDataNck					
F user data received					
-	0	0	0xFFFF	UDoubleword	r
Multi-line: yes	3		16		

safeFrdpMaxComTime					
Maximum value of F_RECVDP communication time in s					
s, user defined	0.0			Double	r
Multi-line: yes	3		16		

safeFrdpSendModeNck	$A_FRDP_SENDMODE[n]				
Current operating mode of the F-CPU of the F_SENDDP communication partner 0: FALSE: The F-CPU is in safety mode 1: TRUE: The F-CPU is in deactivated safety mode					
-	0	0	1	UWord	r
Multi-line: yes	3		16		

safeFrdpSubsNck	$A_FRDP_SUBS[n]				
The user can define substitute values. These are then output to the application during power up, and communication errors are output instead of the process values.					
-	0	0	0xFFFF	UDoubleword	r
Multi-line: yes	3		16		

safeFrdpSubsOnNck	$A_FRDP_SUBS_ON[n]				
During power up and in the event of a communication error, substitute values are output. ERROR = 0 AND SUBS_ON = 1 => Power up ERROR = 1 AND SUBS_ON = 1 => Communication error 0 = Process values are output 1 = Substitute values are output					
-	0	0	1	UWord	r
Multi-line: yes	3		16		

safeFrdpSubsPlc					
The user can define substitute values. These are then output to the application during power up, and communication errors are output instead of the process values.					
-	0	0	0xFFFF	UDoubleword	r
Multi-line: yes	3		16		

safeFsdpActComTime					
Current F_SENDDP communication time in seconds The communication time is the time from F_SENDDP sending the message telegram until the arrival of the correct acknowledge telegram from F_RECVDP.					
s, user defined	0.0			Double	r
Multi-line: yes	3		16		

safeFsdpDiagNck	$A_FSDP_DIAG[n]				
Diagnostics data for F_SENDDP communication/system error 10H = Timeout detected (TO) 20H = Sequence number error detected (SN) 40H = CRC error detected (CRC) 2000H = Deviations in the F telegram data detected (TD) 4000H = Sign-of-life monitoring error detected (LS) 8000H = Asynchronous error state detected (SF)					
-	0	0	0xFFFFFFFF	UDoubleword	r
Multi-line: yes	3		16		

safeFsdpDriverStateNck					
Current state of the F_SENDDP driver 0 = Not parameterized 1 = Initialization 2 = F_SENDDP ready: waiting for F_RECVDP 3 = F_RECVDP ready: waiting for sequence no. = 1 4 = F_SENDDP and F_RECVDP ready: waiting for user acknowledgement from F_RECVDP 5 = Normal operation					
-	0	0	5	UWord	r
Multi-line: yes	3		3		

safeFsdpErrReacNck	$A_FSDP_ERR_REAC[n]				
The user can define the error reaction irrespective of the machining situation or the coupling of the communication partners. 0 = Alarm 27350 + stop D/E 1 = Alarm 27350 2 = Alarm 27351 (display only, self-clearing) 3 = No reaction					
-	0	0	3	UDoubleword	r
Multi-line: yes	3		16		

safeFsdpErrReacPlc					
The user can define the error reaction irrespective of the machining situation or the coupling of the communication partners. 0 = Alarm 27350 + stop D/E 1 = Alarm 27350 2 = Alarm 27351 (display only, self-clearing) 3 = No reaction					
-	0	0	3	UDoubleword	r
Multi-line: yes	3		16		

safeFsdpErrorNck	$A_FSDP_ERROR[n]				
A communication error has been detected. The cause is stated in the diagnostics data 0 = No communication error 1 = Communication error detected					
-	0	0	1	UWord	r
Multi-line: yes	3		16		

safeFsdpFDataNck					
The F user data sent from F_SENDDP to F_RECVDP					
-	0	0	0xFFFF	UDoubleword	r
Multi-line: yes	3		16		

safeFsdpMaxComTime					
Maximum value of the F_SENDDP communication time in s After a communication error, the maximum value is reset to 0 by the user acknowledgement.					
s, user defined	0.0			Double	r
Multi-line: yes	3		16		

safeFsdpStatusSubsNck					
Status signal in the acknowledgement telegram from F_RECVDP to F_SENDDP. F_RECVDP informs F_SENDDP with this signal that there is a communication error, and it is currently outputting substitute values. F_RECVDP resets the signal when it receives a user acknowledgement 0 = F_RECVDP outputs process values 1 = F_RECVDP outputs substitute values					
-	0	0	1	UWord	r
Multi-line: yes	3		16		

safeFsdpSubsOnNck	$A_FSDP_SUBS_ON[n]				
The communication relationship is not in normal operation. If F_RECVDP is active, it outputs substitute values. The signal is set during the start-up of the F communication and in the event of a communication error. ERROR = 0 AND SUBS_ON = 1 => Power up ERROR = 1 AND SUBS_ON = 1 => Communication error 0 = Process values are output by F_RECVDP 1 = Substitute values are output by F_RECVDP					
-	0	0	1	UWord	r
Multi-line: yes	3		16		

safeIntInpValNckBit	$A_INSI[n]				
Internal NCK input of the SI programmable logic from the NCK's SI monitoring channel					
-	0	0	1	UWord	r
Multi-line: yes	Input number		safeMaxNumIntInput		

safeIntInpValNckWord	$A_INSID[n]				
Image of the internal NCK inputs of the SI programmable logic from the NCK's SI monitoring channel					
-	0			Long Integer	r
Multi-line: yes	1: image of the system variables $A_INSID[1] 2: image of the system variables $A_INSID[2] 3: image of the system variables $A_INSID[3] 4: image of the system variables $A_INSID[4] 5: image of the system variables $A_INSID[5] 6: image of the system variables $A_INSID[6]		safeMaxNumIntInput / 32		

safeIntInpValPlcBit	$A_INSIP[n]				
Internal PLC input of the SI programmable logic					
-	0	0	1	UWord	r
Multi-line: yes	Input number		safeMaxNumIntInput		

safeIntInpValPlcWord	$A_INSIPD[n]				
Image of the internal PLC inputs of the SI programmable logic					
-	0			Long Integer	r
Multi-line: yes	1: image of the system variables $A_INSIPD[1] 2: image of the system variables $A_INSIPD[2] 3: image of the system variables $A_INSIPD[3] 4: image of the system variables $A_INSIPD[4] 5: image of the system variables $A_INSIPD[5] 6: image of the system variables $A_INSIPD[6]		safeMaxNumIntInput / 32		

safeIntInputQuality					
Property of an internal NCK-SPL input signal					
-	0	0	3	UWord	r
Multi-line: yes	Input number		safeMaxNumIntInput		

safeIntOutpValNckBit	$A_OUTSI[n]				
Internal NCK output of the SI programmable logic to the NCK's SI monitoring channel					
-	0	0	1	UWord	r
Multi-line: yes	Output number		safeMaxNumIntOutput		

safeIntOutpValNckWord	$A_OUTSID[n]				
Image of the internal NCK outputs of the SI programmable logic to the NCK's SI monitoring channel					
-	0			Long Integer	r
Multi-line: yes	1: image of the system variables $A_OUTSID[1] 2: image of the system variables $A_OUTSID[2] 3: image of the system variables $A_OUTSID[3] 4: image of the system variables $A_OUTSID[4] 5: image of the system variables $A_OUTSID[5] 6: image of the system variables $A_OUTSID[6]		safeMaxNumIntOutput / 32		

safeIntOutpValPlcBit	$A_OUTSIP[n]				
Internal PLC output of the SI programmable logic					
-	0	0	1	UWord	r
Multi-line: yes	Output number		safeMaxNumIntOutput		

safeIntOutpValPlcWord	$A_OUTSIPD[n]				
Image of the internal PLC outputs of the SI programmable logic					
-	0			Long Integer	r
Multi-line: yes	1: image of the system variables $A_OUTSIPD[1] 2: image of the system variables $A_OUTSIPD[2] 3: image of the system variables $A_OUTSIPD[3] 4: image of the system variables $A_OUTSIPD[4] 5: image of the system variables $A_OUTSIPD[5] 6: image of the system variables $A_OUTSIPD[6]		safeMaxNumIntOutput / 32		

safeIntOutputQuality					
Property of an internal NCK-SPL output signal					
-	0	0	3	UWord	r
Multi-line: yes	Output number		safeMaxNumIntOutput		

safeMarkerNck	$A_MARKERSI[n]				
NCK flag for the SI programmable logic					
-	0	0	1	UWord	r
Multi-line: no			safeMaxNumMarker		

safeMarkerNckWord	$A_MARKERSID[n]				
NCK flag words for the safe programmable logic					
-	0	0		Long Integer	r
Multi-line: yes	1: image of the system variables $A_MARKERSID[1] 2: image of the system variables $A_MARKERSID[2] 3: image of the system variables $A_MARKERSID[3] 4: image of the system variables $A_MARKERSID[4] 5: image of the system variables $A_MARKERSID[5] 6: image of the system variables $A_MARKERSID[6]		safeMaxNumMarker / 32		

safeMarkerPlc	$A_MARKERSIP[n]				
Image of the PLC flag-variable for SI programmable logic					
-	0	0	1	UWord	r
Multi-line: no			safeMaxNumMarker		

safeMarkerPlcWord	$A_MARKERSIPD[n]				
Image of the PLC flag words for the safe programmable logic					
-	0	0		Long Integer	r
Multi-line: yes	1: image of the system variables $A_MARKERSIPD[1] 2: image of the system variables $A_MARKERSIPD[2] 3: image of the system variables $A_MARKERSIPD[3] 4: image of the system variables $A_MARKERSIPD[4] 5: image of the system variables $A_MARKERSIPD[5] 6: image of the system variables $A_MARKERSIPD[6]		safeMaxNumMarker / 32		

safeMaxNumExtInput					
Maximum number of external inputs of the Safe Programmable Logic 64 = SPL with a max. of 64 INSE 192 = SPL with a max. of 192 INSE					
-	0	0		Long Integer	r
Multi-line: yes	1		1		

safeMaxNumExtOutput					
Maximum number of external outputs of the SPL 64 = SPL with a max. of 64 OUTSE 192 = SPL with a max. of 192 OUTSE					
-	0	0		Long Integer	r
Multi-line: yes	1		1		

safeMaxNumIntInput					
Maximum number of internal inputs of the SPL 64 = SPL with a max. of 64 INSI 192 = SPL with a max. of 192 INSI					
-	0	0		Long Integer	r
Multi-line: yes	1		1		

safeMaxNumIntOutput					
Maximum number of internal outputs of the SPL 64 = SPL with a max. of 64 OUTSI 192 = SPL with a max. of 192 OUTSI					
-	0	0		Long Integer	r
Multi-line: yes	1		1		

safeMaxNumMarker					
Maximum number of markers for the SPL 64 = SPL with a max. of 64 MARKERSI 192 = SPL with a max. of 192 MARKERSI					
-	0	0		Long Integer	r
Multi-line: yes	1		1		

safeMaxNumPlcInOut					
Maximum number of Safety signals from the PLC to the NCK and vice versa 32 = SPL with a max. of 32 PLCSIIN and 32 PLCSIOUT 96 = SPL with a max. of 96 PLCSIIN and 96 PLCSIOUT					
-	0	0		Long Integer	r
Multi-line: yes	1		1		

safeMode					
Configured safety mode 0 = Not updated/invalid 1 = SINUMERIK Safety Integrated (Drive Based) 2 = SINUMERIK Safety Integrated plus (F-PLC) (without Drive Based) 3 = SINUMERIK Safety Integrated plus (F-PLC) 4 = SINUMERIK Safety Integrated (SPL)					
-	0	0		UWord	r
Multi-line: yes	1		1		

safeNumActiveFrdp					
Number of active F_RECVDP connections					
-	0	0	16	UWord	r
Multi-line: yes	1		1		

safeNumActiveFsdp					
Number of active F_SENDDP connections					
-	0	0	16	UWord	r
Multi-line: yes	1		1		

safePlcIn	$A_PLCSIIN[index]				
Bit image of the single channel safety signals from PLC to NCK					
-	0	0	1	UWord	r
Multi-line: yes	Index for $A_PLCSIIN[]		safeMaxNumPlcInOut		

safePlcOut	$A_PLCSIOUT[index]				
Bit image of the single channel safety signals from NCK to PLC					
-	0	0	1	UWord	r
Multi-line: yes	Index for $A_PLCSIOUT[]		safeMaxNumPlcInOut		

safePsActComTime					
Current communication time in s. The communication time is the time from the sending of a telegram by the PROFIsafe host until the arrival of the correct response telegram from the PROFIsafe slave/device					
s, user defined	0	0		Double	r
Multi-line: yes	PROFIsafe driver no.		safePsMaxnumDrivers		

safePsActCycle					
Current value of the PROFIsafe communication cycle in s					
s, user defined	0.0			Double	r
Multi-line: yes	1		1		

safePsAddress					
PROFIsafe address 0 = Not parameterized >0 = PROFIsafe address					
-	0			UWord	r
Multi-line: yes	PROFIsafe driver no.		safePsMaxnumDrivers		

safePsDiagHost					
Diagnostic data PROFIsafe-host communication and system error 0x0004 = Error checksum (CRC) 0x0008 = Error timeout (TO) 0x0010 = Activate substitute values in slave 0x0100 = Host initialization active 0x0200 = Error consecutive number (CN) 0x0400 = Error host state NCK/PLC (SF) 0x0800 = Empty telegram detected (EA) 0x1000 = Timeout host internal (TF)					
-	0	0	0xffff	UWord	r
Multi-line: yes	PROFIsafe driver no.		safePsMaxnumDrivers		

safePsDiagSlave					
Status data from PROFIsafe slave 0x0002 = Error in slave application 0x0004 = Error checksum (CRC) 0x0008 = Error timeout (TO) 0x0010 = Substitute values activated					
-	0	0	0xffff	UWord	r
Multi-line: yes	PROFIsafe driver no.		safePsMaxnumDrivers		

safePsDriverError					
A communication error has been detected. The cause is stated in the diagnostics data.					
-	0			Bool	r
Multi-line: yes	PROFIsafe driver no.		safePsMaxnumDrivers		

safePsDriverMode					
PROFIsafe link mode 0 = Not parameterized 1 = Inactive 2 = Active					
-	0	0	2	UWord	r
Multi-line: yes	PROFIsafe driver no.		safePsMaxnumDrivers		

safePsDriverState					
Current state of PROFIsafe driver 0 = Not parameterized 1 = Communication setup 2 = Communication setup: waiting for error-free telegrams 3 = Communication: waiting for error-free telegrams with the expected consecutive number 4 = Communication: normal operation 5 = Communication: waiting for acknowledgment after error					
-	0	0	5	UWord	r
Multi-line: yes	PROFIsafe driver no.		safePsMaxnumDrivers		

safePsDriverVersion					
PROFIsafe version F driver 0 = Not parameterized 1 = PROFIsafe V1 2 = PROFIsafe V2					
-	0	0	2	UWord	r
Multi-line: yes	PROFIsafe driver no.		safePsMaxnumDrivers		

safePsFDataIn					
F user data received from the PROFIsafe driver					
-	0	0	0xffffffff	UDoubleword	r
Multi-line: yes	PROFIsafe driver no. + (subslot number - 1) * safePsMaxnumDrivers		safePsMaxnumDrivers * safePsMaxnumSubSlots		

safePsFDataOut					
F user data sent by the PROFIsafe driver					
-	0	0	0xffffffff	UDoubleword	r
Multi-line: yes	PROFIsafe driver no. + (subslot number - 1) * safePsMaxnumDrivers		safePsMaxnumDrivers * safePsMaxnumSubSlots		

safePsHostAddress					
PROFIsafe host address F module 0 = Not parameterized >0 = PROFIsafe host address					
-	0			UWord	r
Multi-line: yes	PROFIsafe driver no.		safePsMaxnumDrivers		

safePsMaxComTime					
Maximum value of the communication time in s. The maximum value is reset to 0 by a Reset after a communication error					
s, user defined	0	0		Double	r
Multi-line: yes	PROFIsafe driver no.		safePsMaxnumDrivers		

safePsMaxCycle					
Maximum value of the PROFIsafe communication cycle in s					
s, user defined	0.0			Double	r
Multi-line: yes	1		1		

safePsMaxnumDrivers					
Maximum number of PROFIsafe drivers					
-	0	0		UWord	r
Multi-line: yes	1		1		

safePsMaxnumSubSlots					
Maximum number of subslots F user data					
-	0	0		UWord	r
Multi-line: yes	1		1		

safePsModuleSlotNo					
Slot number F module 0 = Not parameterized >0 = Slot number					
-	0	0		UWord	r
Multi-line: yes	PROFIsafe driver no.		safePsMaxnumDrivers		

safePsModuleType					
F module type 0 = Not parameterized 1 = F input module 2 = F output module 3 = F input/output module					
-	0	0	3	UWord	r
Multi-line: yes	PROFIsafe driver no.		safePsMaxnumDrivers		

safePsNumActiveDrivers					
Number of active PROFIsafe drivers					
-	0	0	safePsMaxnum Drivers	UWord	r
Multi-line: yes	1		1		

safePsNumDisabledDrivers					
Number of inactive PROFIsafe drivers					
-	0	0	safePsMaxnum Drivers	UWord	r
Multi-line: yes	1		1		

safePsNumSubSlotsIn					
Number of subslots F user data telegram incoming direction 0 = Not parameterized >0 = Number of subslots					
-	0	0	safePsMaxnum SubSlots	UWord	r
Multi-line: yes	PROFIsafe driver no.		safePsMaxnumDrivers		

safePsNumSubSlotsOut					
Number of subslots F user data telegram outgoing direction 0 = Not parameterized >0 = Number of subslots					
-	0	0	safePsMaxnum SubSlots	UWord	r
Multi-line: yes	PROFIsafe driver no.		safePsMaxnumDrivers		

safePsParamMaxComTime					
Maximum configured communication time in s. The communication time is the time from the sending of a telegram by the PROFIsafe host until the arrival of the correct response telegram from the PROFIsafe slave/device					
s, user defined	0	0		Double	r
Multi-line: yes	PROFIsafe driver no.		safePsMaxnumDrivers		

safePsSlaveAddress					
PROFIBUS slave address F module >0 = Not parameterized >0 = PROFIBUS slave address					
-	0	0		UWord	r
Multi-line: yes	PROFIsafe driver no.		safePsMaxnumDrivers		

safeSplStatus					
Status of components and parameter settings required for operation of Safe Programmable Logic Bit 0: SPL interfaces $A_INSE, $A_OUTSE, $A_INSI or $A_OUTSI have been parameterized Bit 1: SPL program file SAFE.SPF loaded Bit 2: NCK is waiting for PLC to power up Bit 3: PLC is in cyclic operating mode. PLC can communicate with drive. Bit 4: Interrupt for ASUB start of SPL must be assigned (FB4 call started) Bit 5: Interrupt for ASUB start of SPL has been assigned (FB4 call ended) Bit 6: Interrupt processing for SPL start called (FC9 call started) Bit 7: Interrupt processing for SPL start ended (FC9 call ended) Bit 8: SPL start by call from PROG_EVENT file Bit 9: NCK data cross-checking has been started Bit10: PLC data cross-checking has been started Bit11: Cyclic SPL checksum check active Bit12: All SPL protective mechanisms active Bit13: SPL program execution ended Bit14: SPL start by PowerOn safety event					
-	0	0		UWord	r
Multi-line: no			1		

safeTimerNck	$A_TIMERSI				
NCK timer-variable for the SI programmable logic					
s, user defined	0.0			Double	r
Multi-line: no			8		

safeXcmpCmd	$A_CMDSI[index]				
Command word for cross-checking (KDV) between NCK and PLC 0:No command 1:Extension of time window for different signal levels in cross-checking operation between NCK and PLC					
-	0	0	1	UWord	r
Multi-line: no			32		

safeXcmpLevel	$A_LEVELSID				
Fill-level display for cross-checking operation (KDV) between NCK and PLC. Specifies the current number of signals of different levels between the NCK and PLC)					
-	0	0		Long Integer	r
Multi-line: no			1		

safeXcmpState	$A_STATSID				
Cross-checking (KDV) error has occurred between NCK and PLC. 0: No error has occurred					
-	0	0		Long Integer	r
Multi-line: no			1		

scalingSystemCounter					
Modification counter for dimension system					
-				UWord	r
Multi-line: yes	1		1		

semaDataAvailable					
Display indicating whether complete SEMA data are available for individual axes. This is the case if a channel can be assigned to the relevant NCU axis, thus allowing the data in the channel context to be accessed. This does not apply to link axes as these are traversed by a channel of another NCU. This data can be utilized by HMIs in order to conceal specific, inaccessible data in link axis data displays. Bits 0-31 stand for the axes of the NCU. Bit n = 1: Data can be accessed easily Bit n = 0: Not all SEMA data are accessible					
-				Long Integer	r
Multi-line: yes	1		1		

simo611dSupport					
This data specifies the extent to which the system supports 611 drives. Bit 0 set: NCK software supports 611D drives Bit 1 set: Hardware supports 611D drives (only if bit 0 is also set).					
-	0	0		UWord	r
Multi-line: yes	1		1		

stopCond					
Number of the NC stop state in the NCK More than one stop state can be active simultaneously. The highest priority stop state appears below the first line, this is followed by those with lower priorities. The documentation explains the meanings of the individual stop states.					
-	0	0		UWord	r
Multi-line: yes	Number of the active stop state		stopCondNumNck		

stopCondChan					
Channel in which the NC stop state was reported More than one stop state can be active simultaneously. The highest priority stop state appears below the first line, this is followed by those with lower priorities. The documentation explains the meanings of the individual stop states.					
-	0	1	maxnumChannels	UWord	r
Multi-line: yes	Number of the active stop state		stopCondNumNck		

stopCondChangeCounter					
Modification counter for stop states in the NCK. This is incremented as soon as one of the stop states has changed.					
-				UWord	r
Multi-line: yes	1		1		

stopCondNumNck					
Number of active stop states in the NCK Specifies the number of occupied lines in stopCond					
-				UWord	r
Multi-line: yes	1		1		

stopCondPar					
Stop state parameters in the NCK. More than one stop state can be active simultaneously. The highest priority stop state appears below the first line, this is followed by those with lower priorities.					
-				UWord	r
Multi-line: yes	High byte: No. of the active stop state Low byte: No. of the parameter				

stopCondParA					
Stop state parameters in the NCK. More than one stop state can be active simultaneously. The highest priority stop state appears below the first line, this is followed by those with lower priorities.					
-				String [32]	r
Multi-line: yes	High byte: No. of the active stop state Low byte: No. of the parameter				

stopCondTime					
BCD time stamp for stop state in the NCK. More than one stop state can be active simultaneously. The highest priority stop state appears below the first line, this is followed by those with lower priorities.					
-				Date+Time	r
Multi-line: yes	Number of the active stop state		stopCondNumNck		

swLicensePIN					
PIN for licensing					
-				String [128]	rw
Multi-line: yes	1		1		

sysTimeBCD					
Time represented in PLC format: <month>.<day>.<year> <hours>:<minutes>:<seconds>.<milliseconds> <weekday> <status> <weekday> can take following values: "SUN", "MON", "TUE", "WED", "THU", "FRI", "SAT"					
-				Date+Time	r
Multi-line: no					

sysTimeNCSC					
NCSC system time in microseconds					
μs	0	0		Long Integer	r
Multi-line: yes	1		1		

sysTimeNCSCatTraceStart					
Logging: NCSC time stamp for the trace start time in μs					
μs	0	0		Long Integer	r
Multi-line: yes	User no. (1-10)		10		

sysTimeNCSCatTraceTrig					
Logging: NCSC time stamp for the trace start trigger time in µs					
µs	0	0		Long Integer	r
Multi-line: yes	User no. (1-10)		10		

sysTimeNCSCdiffTraceStart					
Logging: Time difference for the trace start time in µs					
µs	0	0		Long Integer	r
Multi-line: yes	User no. (1-10)		10		

sysTimeNCSCdiffTraceTrig					
Logging: time difference for the trace start trigger time in µs					
µs	0	0		Long Integer	r
Multi-line: yes	User no. (1-10)		10		

sysTimeSinceStartup					
System run time in seconds since NCK ramp-up					
s, user defined	0	0		Double	r
Multi-line: yes	1		1		

sysTimeUdword					
Time sysTimeBCD in special data format: 6 bits for second (bits of the lowest priority) 6 bits for minute 5 bits for hour 5 bits for day 4 bits for month 6 bits for the last two digits of the year This coding is suitable for using the absolute time as trigger for logging. Also see: protocStrtValueInt32 and protocTrigValueInt32					
-				UDoubleword	r
Multi-line: no					

tlkNr					
Unique copy number for the temporary license key					
-				String [32]	r
Multi-line: yes	1		1		

tlkPIN					
Temporary license key					
-				String [128]	r
Multi-line: yes	1		1		

tlkStatus					
Status of the temporary license key 0: active 1: inactive 10: incorrect input 11: max. number of incorrect inputs exceeded 200: internal error (TLK_BUFFER_TOO_SMALL)					
-	1			UWord	r
Multi-line: yes	1		1		

totalDirectorys					
Maximum number of directories which may be created see: $MN_MM_NUM_DIR_IN_FILESYSTEM					
-				UWord	r
Multi-line: yes	1		1		

totalFiles					
Maximum number of files which may be created (see: $MM_NUM_FILES_IN_FILESYSTEM)					
-				UWord	r
Multi-line: yes	Type of memory: 1: MMF (Solutionline) / SRAM (Powerline) 2: DRAM 3: MMF 4: SRAM		4		

totalMem					S7
Total SRAM in bytes (user memory)					
-				Long Integer	r
Multi-line: yes	1				

totalMemDram					
total DRAM in bytes					
-				Long Integer	r
Multi-line: yes	1		1		

totalMemDramEPassF					
Size in bytes of the passive file system for executing from external drives					
-	0	0		Long Integer	r
Multi-line: yes	1		1		

totalMemDramMPassF					
Size of the passive file system of the "Machine manufacturer" area in bytes					
-	0	0		Long Integer	r
Multi-line: yes	1		1		

totalMemDramPassF					
Size of passive file system (DRAM No. 1) in bytes					
-	0	0		Long Integer	r
Multi-line: yes	1		1		

totalMemDramSPassF					
Size of the passive file system of the "Control manufacturer" area in bytes					
-	0	0		Long Integer	r
Multi-line: yes	1		1		

totalMemDramTPassF					
Size of the passive file system of the "Temp" area in bytes					
-	0	0		Long Integer	r
Multi-line: yes	1		1		

totalMemDramUPassF					
Size of the passive file system of the "User" area in bytes					
-	0	0		Long Integer	r
Multi-line: yes	1		1		

totalMemFfs					
Only with 840D-powerline: Number of bytes reserved on the PCMCIA card for the Flash File System (FFS)					
-	0			Long Integer	r
Multi-line: yes	1		1		

totalMemISram					
Total internal SRAM in bytes					
-	0	0		Long Integer	r
Multi-line: yes	1		1		

totalMemSramPassF					
Size of passive file system (SRAM) in bytes					
-	0	0		Long Integer	r
Multi-line: yes	1		1		

totalProtokolFiles	$MM_PROTOC_NUM_FILES				
Logging: Maximum number of log files which may be created					
-	0	0	1	UWord	r
Multi-line: yes	User No. (1-10)		10		

traceProtocolActive	$A_PROTOC				
Logging: User status 0 = Not active 1 = Active					
-	0	0	1	UWord	r
Multi-line: yes	User No. (1-10)		10		

traceProtocolLock	$A_PROT_LOCK				
Logging: Recording disable of a user 0: No disable 1: Disable 2: Disable, events from protocHmiEvent, however, are enabled					
-	0	0	1	UWord	rw
Multi-line: yes	User No. (1-10)		10		

traceStopAction					
Logging: Actions on ending the recording Bit0: Automatic restart after the stop trigger has ended the trace Bit1: Reserved Bit2: The parameterization of the session is stored in an ACX file The file name is formed from the log file with the extension "_U00_ACX". Bit3: The parameterization of the session, including diagnostic data, is stored in an ACX file The file name is formed from the log file with the extension "_U00_ACX".					
-	0	0		UWord	rw
Multi-line: yes	User No. (1-10)		10		

usedDirectorys					
Number of directories that have already been created					
-				UWord	r
Multi-line: yes	1		1		

usedFiles					
Number of files that have already been created					
-				UWord	r
Multi-line: yes	Type of memory: 1: MMF (Solutionline) / SRAM (Powerline) 2: DRAM 3: MMF 4: SRAM		4		

usedMem					S7
Used memory in bytes					
-				Long Integer	r
Multi-line: yes	1				

usedMemDram					
Used DRAM in bytes					
-				Long Integer	r
Multi-line: yes	1		1		

usedMemDramEPassF					
Memory in bytes occupied by the passive file system for executing from external drives					
-	0	0		Long Integer	r
Multi-line: yes	1		1		

usedMemDramMPassF					
Occupied memory of the passive file system of the "Machine manufacturer" area in bytes					
-	0	0		Long Integer	r
Multi-line: yes	1		1		

usedMemDramPassF					
Memory used in passive file system (DRAM No. 1) in bytes					
-	0	0		Long Integer	r
Multi-line: yes	1		1		

usedMemDramSPassF					
Occupied memory of the passive file system of the "Control manufacturer" area in bytes					
-	0	0		Long Integer	r
Multi-line: yes	1		1		

usedMemDramTPassF					
Occupied memory of the passive file system of the "Temp" area in bytes					
-	0	0		Long Integer	r
Multi-line: yes	1		1		

usedMemDramUPassF					
Occupied memory of the passive file system of the "User" area in bytes					
-	0	0		Long Integer	r
Multi-line: yes	1		1		

usedMemFfs					
Only with 840D-powerline: Number of bytes assigned in the Flash File System (FFS)					
-	0			Long Integer	r
Multi-line: yes	1		1		

usedMemISram					
Occupied internal SRAM					
-	0	0		Long Integer	r
Multi-line: yes	1		1		

usedMemSramPassF					
Memory used in passive file system (SRAM) in bytes					
-	0	0		Long Integer	r
Multi-line: yes	1		1		

usedOptionsNotLicensed					
List of options which are not licensed					
-				String [200]	r
Multi-line: yes	1		1		

usedProtokolFiles					
Logging: Number of protocol files that have already been created					
-	0	0	1	UWord	r
Multi-line: yes	User No. (1-10)		10		

vaDpActTel	$VA_DP_ACT_TEL[n, Achse]				
Word for word image of the PROFIBUS actual value message frames from drives on the PROFIBUS/PROFIdrive					
-	0	0		Long Integer	r
Multi-line: yes	100 * axis index + word offset in the message frame		100 * numMachAxes + 19		

3.3.2 Area N, Block SALA : Alarms: List organized according to time, oldest alarm appears first

OEM-MMC: Linkitem /NckSequencedAlarms/...

The NCK alarms are sorted in a list in the order they occurred, the oldest alarm appears at the top of the list. The alarm parameters are transferred as ASCII strings, the first character contains the type information for that parameter. The following types are used:

S: General string, e.g. part program name

A: Axis name / spindle name

K: Channel name

N: Block number

Y: System error

D: Drive number

If a parameters is not assigned, an "S" is transferred.

All variables in this module are privileged variables! This means that cyclic acknowledgements are sent for these variables even if the cyclic services are no longer serviced by the NCK because of block cycle problems.

Attention: Privileged variables lose this characteristic if they are combined with non-privileged variables in a request. -> Do not combine alarm variables with other variables in a cluster!

In addition it is presumed that the cyclic services are set "on change" for the alarm variables and are not combined with other variables (not even with privileged variables) in the same request.

The module SALA only contains the alarms that are generated in the NCK. It contains neither PLC nor HMI alarms. In order to read all alarms, the OEM-HMI user should use the alarm server functions and not read the SALA module directly.

alarmNo					DA
Ordinal number of an alarm (how many alarms since control ON) 0 = unknown alarm					
-				Long Integer	r
Multi-line: yes	Alarm list index The maximum alarm list index can be read via variable numAlarms in module S.		16		

clearInfo					DA
Acknowledgement criterion for an alarm 1 = Power On 2 = Reset 3 = Cancel 4 = Alarm is cancelled by NCK-software 5 = Alarm is cancelled by starting a program 6 = Alarm is cancelled by RESET in all channels of the bags (from SW 4.1) 7 = Alarm is cancelled by RESET in all channels of the NC (from SW 4.1)					
-				Long Integer	r
Multi-line: no			1		

fillText1					DA
Parameter 1 of the alarm					
-				String [32]	r
Multi-line: yes	Alarm list index The maximum alarm list index can be read via variable numAlarms in module S.		16		

fillText2					DA
Parameter 2 of the alarm					
-				String [32]	r
Multi-line: yes	Alarm list index The maximum alarm list index can be read via variable numAlarms in module S.		16		

fillText3					DA
Parameter 3 of the alarm					
-				String [32]	r
Multi-line: yes	Alarm list index The maximum alarm list index can be read via variable numAlarms in module S.		16		

fillText4					DA
Parameter 4 of the alarm					
-				String [32]	r
Multi-line: yes	Alarm list index The maximum alarm list index can be read via variable numAlarms in module S.		16		

textIndex					
Alarm number (actual alarm)					
-				Long Integer	r
Multi-line: yes	Alarm list index The maximum alarm list index can be read via variable numAlarms in module S.		16		

timeBCD					
Time stamp of an alarm Time stamp, displayed in PLC format DATE_AND_TIME					
-				Date+Time	r
Multi-line: yes	Alarm list index The maximum alarm list index can be read via variable numAlarms in module S.		16		

3.3.3 Area N, Block SALAP : Alarms: List organized according to priority

OEM-MMC: Linkitem /NckTopPrioAlarm/...

The alarms of the NCK are arranged in a list in the order of their priorities. The highest-priority alarm is set to position 1. The alarm list is sorted according to the following criteria:

1. Sorting criterion: deletion criterion (highest priority at 1st position)

    - NC power OFF/ON

    - Press reset key

    - Press "Delete alarm"

    - Press "NC start"

    - Press "Recall" key

2. Sorting criterion: alarm occurrence time

The alarm parameters are transferred as ASCII strings, the first character contains the type information for the parameter. The following types are used:

S: General string, e.g. part program name

A: Axis name / spindle name

K: Channel name

N: Block name

Y: System error

D: Drive number

If a parameter is not assigned, an "S" is transferred.

All variables in this module are privileged variables! This means that cyclic acknowledgements are sent for these variables even if the cyclic services are no longer serviced by the NCK because of block cycle problems.

Notice: Privileged variables lose this characteristic if they are combined with non-privileged variables in a request. -> Do not combine alarm variables with other variables in a cluster!

In addition it is presumed that the cyclic services are set "on change" for the alarm variables and are not combined with other variables (not even with privileged variables) in the same request.

The module SALAP only contains the alarms that are generated in the NCK. It contains neither PLC nor HMI alarms. In order to read all alarms, the OEM-HMI user should use the alarm server functions and not read the SALAP module directly.

alarmNo					DA
Ordinal number of an alarm (how many alarms since control ON) 0 = unknown alarm					
-				Long Integer	r
Multi-line: yes	Alarm list index The maximum alarm list index can be read via variable numAlarms in module S.		16		

clearInfo					DA
Acknowledgement criterion for an alarm 1 = Power On 2 = Reset 3 = Cancel 4 = Alarm is cancelled by NCK-software 5 = Alarm is cancelled by starting a program 6 = Alarm is cancelled by RESET in all channels of the bags (from SW 4.1) 7 = Alarm is cancelled by RESET in all channels of the NC (from SW 4.1)					
-				Long Integer	r
Multi-line: no					

fillText1					DA
Parameter 1 of the alarm					
-				String [32]	r
Multi-line: yes	Alarm list index The maximum alarm list index can be read via variable numAlarms in module S.		16		

fillText2					DA
Parameter 2 of the alarm					
-				String [32]	r
Multi-line: yes	Alarm list index The maximum alarm list index can be read via variable numAlarms in module S.		16		

fillText3					DA
Parameter 3 of the alarm					
-				String [32]	r
Multi-line: yes	Alarm list index The maximum alarm list index can be read via variable numAlarms in module S.		16		

fillText4					DA
Parameter 4 of the alarm					
-				String [32]	r
Multi-line: yes	Alarm list index The maximum alarm list index can be read via variable numAlarms in module S.		16		

textIndex					
Alarm number (actual alarm)					
-				Long Integer	r
Multi-line: yes	Alarm list index The maximum alarm list index can be read via variable numAlarms in module S.		16		

timeBCD					
Time stamp of an alarm Time stamp, displayed in PLC format DATE_AND_TIME					
-				Date+Time	r
Multi-line: yes	Alarm list index The maximum alarm list index can be read via variable numAlarms in module S.		16		

3.3.4 Area N, Block SALAL : Alarms: Liste organized according to time, most recent alarm appears first

OEM-MMC: Linkitem /NckLastAlarm/...

The NCK alarms are sorted in a list in the order they occurred, the most recent alarm appears at the bottom of the list. The alarm parameters are transferred as ASCII strings, the first character contains the type information for that parameter. The following types are used:

S: General string, e.g. part program name

A: Axis name / spindle name

K: Channel name

N: Block number

Y: System error

D: Drive number

If a parameters is not assigned, an "S" is transferred.

All variables in this module are privileged variables! This means that cyclic acknowledgements are sent for these variables even if the cyclic services are no longer serviced by the NCK because of block cycle problems.

Attention: Privileged variables lose this characteristic if they are combined with non-privileged variables in a request. -> Do not combine alarm variables with other variables in a cluster!

In addition it is presumed that the cyclic services are set "on change" for the alarm variables and are not combined with other variables (not even with privileged variables) in the same request.

The module SALA only contains the alarms that are generated in the NCK. It contains neither PLC nor HMI alarms. In order to read all alarms, the OEM-HMI user should use the alarm server functions and not read the SALA module directly.

alarmNo					DA
Ordinal number of an alarm (how many alarms since control ON) 0 = unknown alarm					
-				Long Integer	r
Multi-line: yes	Alarm list index The maximum alarm list index can be read via variable numAlarms in module S.		16		

clearInfo					DA
Acknowledgement criterion for an alarm 1 = Power On 2 = Reset 3 = Cancel 4 = Alarm is cancelled by NCK-software 5 = Alarm is cancelled by starting a program 6 = Alarm is cancelled by RESET in all channels of the bags (from SW 4.1) 7 = Alarm is cancelled by RESET in all channels of the NC (from SW 4.1)					
-				Long Integer	r
Multi-line: no					

fillText1					DA
Parameter 1 of the alarm					
-				String [32]	r
Multi-line: yes	Alarm list index The maximum alarm list index can be read via variable numAlarms in module S.		16		

fillText2					DA
Parameter 2 of the alarm					
-				String [32]	r
Multi-line: yes	Alarm list index The maximum alarm list index can be read via variable numAlarms in module S.		16		

fillText3					DA
Parameter 3 of the alarm					
-				String [32]	r
Multi-line: yes	Alarm list index The maximum alarm list index can be read via variable numAlarms in module S.		16		

fillText4					DA
Parameter 4 of the alarm					
-				String [32]	r
Multi-line: yes	Alarm list index The maximum alarm list index can be read via variable numAlarms in module S.		16		

textIndex					
Alarm number (actual alarm)					
-				Long Integer	r
Multi-line: yes	Alarm list index The maximum alarm list index can be read via variable numAlarms in module S.		16		

timeBCD					
Time stamp of an alarm Time stamp, displayed in PLC format DATE_AND_TIME					
-				Date+Time	r
Multi-line: yes	Alarm list index The maximum alarm list index can be read via variable numAlarms in module S.		16		

3.3.5 Area N, Block SMA : State data: Machine axes

OEM-MMC: Linkitem /NckMachineAxis/...

All state data that are dependent on machine movement and are defined specifically for machine axes (geometry and special axes) are combined in module SMA. Supplementary information is to be found in module SEMA. The individual variables are defined as fields where the line index is the number of the machine axis (assigned to the current channel). The variable "name" in module SMA with the line index in question identifies the axis.

The assignment of the line indices in modules SMA and SEMA is identical.

actIncrVal					H1
Active INC weighting of the axis 0 = INC_10000 1 = INC_1000 2 = INC_100 3 = INC_10 4 = INC_1 5 = INC_VAR 6 = INC_JOG_CONT 7 = no incremental mode set					
-				UWord	r
Multi-line: yes	Axis index		numMachAxes		

actToolBasePos					
Tool base position. Physical unit is defined in the variable extUnit (from this module)					
mm, inch, degree, user defined				Double	r
Multi-line: yes	Axis index		numMachAxes		

cmdToolBasePos					
Tool base position, desired value . Physical unit is defined in variable extUnit (in this module).					
mm, inch, degree, user defined				Double	r
Multi-line: yes	Axis index		numMachAxes		

extUnit					
Current physical unit of the axis position 0 = mm 1 = inch 2 = degree 3 = indexing position 4 = userdef					
-				UWord	r
Multi-line: yes	Axis index		numMachAxes		

name					
Axis name					
-				String [32]	r
Multi-line: yes	Axis index		numMachAxes		

status					
Axis status 0 = travel command in plus direction 1 = travel command in minus direction 2 = exact position coarse reached 3 = exact position fine reached					
-				UWord	r
Multi-line: yes	Axis index		numMachAxes		

toolBaseDistToGo					
Tool base distance-to-go. Physical unit is defined in the variable extUnit (in this module).					
mm, inch, degree, user defined				Double	r
Multi-line: yes	Axis index		numMachAxes		

toolBaseREPOS					
Tool base REPOS. Physical unit is defined in the variable extUnit (in this module).					
mm, inch, degree, user defined				Double	r
Multi-line: yes	Axis index		numMachAxes		

varIncrVal					
Settable value for INC_VAR. The physical value depends on whether the axis is linear or rotary. Linear axis: unit is 1 mm Rotary axis: unit is 1/1000 degrees					
mm, inch, degree, user defined				Double	r
Multi-line: yes	Axis index		numMachAxes		

3.3.6 Area N, Block SEMA : State data: Machine axes (extension of SMA)

OEM-MMC: Linkitem /NckMachineAxis/...

All state data that are dependent on machine movement and are defined specifically for machine axes (geometry and special axes) are combined in module SMA. Supplementary information is to be found in module SEMA. The individual variables are defined as fields where the line index is the number of the machine axis (assigned to the current channel). The variable "name" in module SMA with the line index in question identifies the axis.

The assignment of the line indices in modules SMA and SEMA is identical.

PRESETActive					
Preset state 0 = no preset active 1 = preset active					
-				UWord	r
Multi-line: yes	Axis Number		maxnumGlobMachAxes		

PRESETVal					
The function PRESETON (...) programs a work offset for an axis. The value of the offset is stored in the variable 'PRESETVal'. The variable can be overwritten by the part program and by the HMI.					
mm, inch, user defined				Double	rw
Multi-line: yes	Axis Number		maxnumGlobMachAxes		

aaAcc					
Current axial acceleration value					
m/s2, 1000 inch/ s2, rev/s2, user defined	0	0		Double	r
Multi-line: yes	Axis Number		maxnumGlobMachAxes		

aaAccPercent					
Current acceleration value for single-axis interpolation in percent					
-	0	0		UWord	r
Multi-line: yes	Axis Number		maxnumGlobMachAxes		

aaActIndexAxPosNo					
Current indexing position; the display depends on $MN_INDEX_AX_NO_MODE and the division (via table or equidistant)					
-	0			Long Integer	r
Multi-line: yes	Axis Number		maxnumGlobMachAxes		

aaAlarmStat					
Display indicating whether alarms are active for a PLC-controlled axis. The relevant coded alarm reactions can be used as a source for the "Extended Stop and Retract" function. The data is bit-coded, allowing, where necessary, individual states to be masked or evaluated separately (bits not listed supply a value of 0) Bit2 = 1: NOREADY (active rapid deceleration + cancelation of servo enable) Bit6 = 1: STOPBYALARM (rampm stop in all channel axes) Bit9 = 1: SETVDI (VDI interface signal "Setting alarm") Bit13 = 1: FOLLOWUPBYALARM (Follow-up)					
-	0			UWord	r
Multi-line: yes	Axis Number		maxnumGlobMachAxes		

aaAxChangeStat					
Axis status with respect to axis replacement 0: Axis can be replaced 1: Axis is linked to the channel, but can become the PLC, command or reciprocating axis 2: Axis cannot be replaced					
-	0	0	2	UWord	r
Multi-line: yes	Axis Number		maxnumGlobMachAxes		

aaAxChangeTyp					
Axis type with respect to axis replacement 0: Axis assigned to the NC program 1: Axis assigned to the PLC or active as command axis or reciprocating axis 2: Other channel has interpolation right 3: Neutral axis 4: Neutral axis controlled from the PLC 5: Other channel has interpolation right; axis is requested for the NC program 6: Other channel has interpolation right; axis is requested as neutral axis 7: Axis is PLC axis or is active as command axis or reciprocating axis; axis is requested for the NC program 8: Axis is PLC axis or is active as command axis or reciprocating axis; axis is requested as neutral axis					
-	0	0	8	UWord	r
Multi-line: yes	Axis Number		maxnumGlobMachAxes		

aaBcsOffset					
Sum of all axial offsets of an axis, such as DRF, online tool offset, $AA_OFF and ext. WO.					
-	0			Double	r
Multi-line: yes	Axis Number		maxnumGlobMachAxes		

<table>
<tr><th colspan="3">aaBrakeCondB</th><th colspan="2"></th><th></th></tr>
<tr><td colspan="6">Shows the pending braking requests (conditions) for the interpolator stop of the axis / spindle.
A braking request consists of a collision direction relating to a coordinate axis in the BCS and a braking priority relating to the machining step.
If the axis / spindle receives a current braking request on account of these requirement(s), bit 0 is set in $AA_BRAKE_STATE[X] (in the next IPO cycle).
The highest deceleration priority in positive direction is shown in bits 0 to 3:
0x0: No pending deceleration request
0x1: Priority 1 covers all positioning actions (G0, POS, SPOS)
0x2: Priority 2 covers DYNNORM and all priority 1 motions
0x3: Priority 3 covers DYNPOS and all priority 1 to 2 motions
0x4: Priority 4 covers DYNROUGH and all priority 1 to 3 motions
0x5: Priority 5 covers DYNSEMIFIN and all priority 1 to 4 motions
0x6: Priority 6 covers all motions (including DYNFINISH). The request could also have been triggered by a CP SW limit stop.
0x7: Priority 7 covers all motions. The request was triggered by the VDI interface signal DB31,..DBX4.3 "Feed stop / Spindle stop".
Deceleration always takes place, irrespective of the direction of motion.
0xD: Priority 13 covers all motions. Axial deceleration takes place with an emergency stop deceleration ramp.
The highest deceleration priority in negative direction is shown in bits 16 to 19:
0x0 to 0xD: Same meaning as bits 0 to 3
All other bits are reserved and not set.
If the value of the variable is shown in hexadecimal format, the fifth character from the right shows the deceleration priority
in negative direction while the first number from the right shows the deceleration priority in positive direction.</td></tr>
<tr><td>-</td><td>0</td><td>0</td><td>0x70007</td><td>UDoubleword</td><td>r</td></tr>
<tr><td>Multi-line: yes</td><td colspan="2">Axis Number</td><td colspan="3">maxnumGlobMachAxes</td></tr>
</table>

aaBrakeCondM					
Shows the pending braking requests (conditions) for the interpolator stop of the axis / spindle. A braking request consists of a collision direction relating to a coordinate axis in the MCS and a braking priority relating to the machining step. The highest braking priorities in the positive direction are indicated in bits 0 to 3: 0x0: No pending braking request 0x1: Priority 1 covers all positioning actions (G0, POS, SPOS) 0x2: Priority 2 covers DYNNORM and all priority 1 motions 0x3: Priority 3 covers DYNPOS and all priority 1 to 2 motions 0x4: Priority 4 covers DYNROUGH and all priority 1 to 3 motions 0x5: Priority 5 covers DYNSEMIFIN and all priority 1 to 4 motions 0x6: Priority 6 covers all motions (including DYNFINISH). The request could also have been triggered by a CP SW limit stop. 0x7: Priority 7 covers all motions. The request was triggered by the VDI interface signal DB31,..DBX4.3 "Feed stop / Spindle stop". Braking always takes place, irrespective of the direction of motion. 0xD: Priority 13 covers all motions. Axial braking with an emergency stop braking ramp. The highest braking priority in the negative direction is indicated in bits 16 to 19: 0x0 to 0xD: Same meaning as bits 0 to 3 All other bits are reserved and not set. If the value of the variable is shown in hexadecimal format, the fifth character from the right shows the braking priority in the negative direction and the first character from the right shows it in the positive direction.					
-	0	0	0x70007	UDoubleword	r
Multi-line: yes	Axis Number		maxnumGlobMachAxes		

aaBrakeState					
Returns for the axis / spindle whether braking has been initiated on account of the request of aaBrakeCondB or a CP SW limit stop or a VDI interface signal DB31,..DBX4.3 "Feed stop / Spindle stop".					
-	0	0	1	UDoubleword	r
Multi-line: yes	Axis Number		maxnumGlobMachAxes		

aaChanNo					
The variable supplies the number of the channel in which the axis is currently being interpolated. With value 0, the axis could not be assigned to any channel.					
-	0	0		UWord	r
Multi-line: yes	Axis Number		maxnumGlobMachAxes		

aaCollPos					
Position of a machine axis with risk of collision					
mm, inch, degree, user defined	0			Double	r
Multi-line: yes	Axis Number		numMachAxes		

aaCoupAct					
Current coupling state of the slave spindle					
-				UWord	r
Multi-line: yes	Axis Number		maxnumGlobMachAxes		

aaCoupCorr					
This variable is used to execute the function "Correct synchronism error". It returns the compensation value for the position offset for the generic coupling with CPFRS = "MCS". The actual values of this spindle are compared with the setpoints for the duration (MD 30455 MISC_FUNCTION_MASK, bit 7) of the activation of the VDI interface signal DB31...,DBX31.6 'Correct synchronism' for the following spindle with coupling active. The difference is the compensation value, which can be read with this variable.					
-	0			Double	r
Multi-line: yes	Axis Number		maxnumGlobMachAxes		

aaCoupCorrDist					
Generic coupling: path still to be retracted for aaCoupCorr					
-	0			Double	r
Multi-line: yes	Axis Number		maxnumGlobMachAxes		

aaCoupOffs					
Position offset of the synchronous spindle desired value					
-				Double	r
Multi-line: yes	Axis Number		maxnumGlobMachAxes		

aaCurr					
Actual value of the axis/spindle current in A (Only available for PROFIdrive drives)					
A				Double	r
Multi-line: yes	Axis Number		maxnumGlobMachAxes		

aaDepAxO					
Dependency on other axes. Returns an axis code for the defined axis AX containing all the machine axes that have a mechanical dependency on the defined axis.					
-	0	0		Long Integer	r
Multi-line: yes	Axis Number		maxnumGlobMachAxes		

aaDtbb					
Axis-specific distance from the beginning of the block in the BCS for positioning and synchronous axes used in synchronous actions (note: SYNACT only)					
-				Double	r
Multi-line: yes	Axis Number		maxnumGlobMachAxes		

aaDtbreb					
The estimated total distance until the end of deceleration is reached, BCS					
-	0	0		Double	r
Multi-line: yes	Axis Number		maxnumGlobMachAxes		

aaDtbrebCmd					
Command share of the overall deceleration distance of axis ax in the BCS. The value is the estimated deceleration distance of the axis up to standstill.					
-	0	0		Double	r
Multi-line: yes	Axis Number		maxnumGlobMachAxes		

aaDtbrebCorr					
Offset section of the deceleration distance, BCS					
-	0	0		Double	r
Multi-line: yes	Axis Number		maxnumGlobMachAxes		

aaDtbrebDep					
Dependent section of the decelaration distance, BCS					
-	0	0		Double	r
Multi-line: yes	Axis Number		maxnumGlobMachAxes		

aaDtbrem					
The estimated total distance until the end of deceleration is reached, MCS					
-	0	0		Double	r
Multi-line: yes	Axis Number		maxnumGlobMachAxes		

aaDtbremCmd					
Specified section of the decelaration distance, MCS					
-	0	0		Double	r
Multi-line: yes	Axis Number		maxnumGlobMachAxes		

aaDtbremCorr					
Offset section of the deceleration distance, MCS					
-	0	0		Double	r
Multi-line: yes	Axis Number		maxnumGlobMachAxes		

aaDtbremDep					
Dependent section of the decelaration distance, MCS					
-	0	0		Double	r
Multi-line: yes	Axis Number		maxnumGlobMachAxes		

aaDteb					
Axis-specific distance to the end of the block in the BCS for positioning and synchronous axes used in synchronous actions (note: SYNACT only)					
-				Double	r
Multi-line: yes	Axis Number		maxnumGlobMachAxes		

aaDtepb					
Axis-specific distance-to-go of infeed during oscillation in the BCS (note: SYNACT only)					
-				Double	r
Multi-line: yes	Axis Number		maxnumGlobMachAxes		

aaEnc1Active					
First measuring system is active 0: Measuring system is not active 1: Measuring system is active					
-	0	0	1	UWord	r
Multi-line: yes	Axis Number		maxnumGlobMachAxes		

aaEnc2Active					
Second measuring system is active 0: Measuring system is not active 1: Measuring system is active					
-	0	0	1	UWord	r
Multi-line: yes	Axis Number		maxnumGlobMachAxes		

aaEncActive					
Measuring system is active 0: Measuring system is not active 1: Measuring system is active					
-	0	0	1	UWord	r
Multi-line: yes	Axis Number		maxnumGlobMachAxes		

aaEsrEnable					
(Axial) enabling of reactions of "Extended Stop and Retract" function. The selected axial ESR reaction must be parameterized in MD $MA_ESR_REACTION. beforehand. The corresponding Stop or Retract reactions can be activated via $AN_ESR_TRIGGER (or for individual drives in the event of communications failure/ DC-link undervoltage), generator-mode operation is automatically activated in response to undervoltage conditions. 0: FALSE 1: TRUE					
-	0	0	1	UWord	r
Multi-line: yes	Axis Number		maxnumGlobMachAxes		

aaEsrStat					
(Axial) status checkback signals of "Extended Stop and Retract" function, which can be applied as input signals for the gating logic of the ESR (synchronous actions). The data is bit-coded. Individual states can therefore be masked or evaluated separately if necessary: Bit0 = 1: Generator mode is activated Bit1 = 1: Retract operation is activated Bit2 = 1: Stop operation is activated Bit3 = 1: Risk of undervoltage (DC-link voltage monitoring, voltage has dropped below warning threshold) Bit4 = 1: Speed has dropped below minimum generator mode threshold (i.e. no more regenerative rotation energy is available).					
-	0			UWord	r
Multi-line: yes	Axis Number		maxnumGlobMachAxes		

aaEsrTrigger					
Activation of "NC-controlled ESR" for PLC-controlled axis					
-	0	0	1	UWord	r
Multi-line: yes	Axis Number		maxnumGlobMachAxes		

aaFixPointSelected					
Selected fixed point: Number of the fixed point that is to be approached					
-	0			UDoubleword	r
Multi-line: yes	Axis Number		maxnumGlobMachAxes		

aaIbnCorr					
Current BZS setpoint value of an axis including override components					
-	0			Double	r
Multi-line: yes	Axis Number		maxnumGlobMachAxes		

aaIenCorr					
Current SZS setpoint value of an axis including override components					
-	0			Double	r
Multi-line: yes	Axis Number		maxnumGlobMachAxes		

aaInSync					
Synchronization status of the following axis with master value coupling and ELG 0: Synchronization is not running 1: Synchronization is running, i.e. following axis is being synchronized					
-	0	0	1	UWord	r
Multi-line: yes	Axis Number		maxnumGlobMachAxes		

aaInposStat					
Status for the programmed position 0: No status available (axis/spindle is outside of the programmed position) 1: Travel motion pending 2: Position setpoint reached 3: Position reached with 'exact stop coarse' 4: Position reached with 'exact stop fine'					
-	0	0	4	UWord	r
Multi-line: yes	Axis Number		maxnumGlobMachAxes		

aaIpoNcChanax					
If the axis is currently interpolated to this NCU, the channel and channel number which define the interpolator of the axis are output. If the axis is currently interpolated to a different NCU, the NCU identifier of the interpolated NCU and the global axis number of the machine axis are output. This global axis number can then be used to transfer the interpolated channel and the channel axis number to the other NCU, with NCU ID 2, with $AN_IPO_CHANAX[203]. The axis must be assigned to at least one channel on this NCU, otherwise 0 will be returned. The channel is output as from position 100, and the channel axis number is output as from position 1, e.g. 1005 - channel 10 channel axis 5. These values are always lower than 10000. The NCU is output as from position 10000, e.g. 20203: NCU 2 and the global axis number is 203.					
-	0	0		UDoubleword	r
Multi-line: yes	Axis Number		maxnumGlobMachAxes		

aaJerkCount					
Total traverse processes of an axis with jerk					
-		0		Double	r
Multi-line: yes	Axis Number		maxnumGlobMachAxes		

aaJerkTime					
Total traverse time of an axis with jerk					
s, user defined		0		Double	r
Multi-line: yes	Axis Number		maxnumGlobMachAxes		

aaJerkTotal					
Overall total jerk of an axis					
-		0		Double	r
Multi-line: yes	Axis Number		maxnumGlobMachAxes		

aaJogPosAct					
Position reached for JOG to position					
-	0	0	1	UWord	r
Multi-line: yes	Axis Number		maxnumGlobMachAxes		

aaJogPosSelected					
JOG to position is active					
-	0	0	1	UWord	r
Multi-line: yes	Axis Number		maxnumGlobMachAxes		

aaLeadP					
Actual lead value position					
mm, inch, degree, user defined				Double	r
Multi-line: yes	Axis Number		maxnumGlobMachAxes		

aaLeadPTurn					
Current master value - position component lost as a result of modulo reduction					
mm, inch, degree, user defined	0	0		Double	r
Multi-line: yes	Axis Number		maxnumGlobMachAxes		

aaLeadSp					
Simulated lead value - position					
mm, inch, degree, user defined				Double	r
Multi-line: yes	Axis Number		maxnumGlobMachAxes		

aaLeadSv					
Simulated leading value velocity					
mm/min, inch/min, user defined				Double	r
Multi-line: yes	Axis Number		maxnumGlobMachAxes		

aaLeadV					
Actual lead value - velocity					
mm/min, inch/min, user defined				Double	r
Multi-line: yes	Axis Number		maxnumGlobMachAxes		

aaLoad					
Drive load in % (only available for PROFIdrive drives)					
%				Double	r
Multi-line: yes	Axis Number		maxnumGlobMachAxes		

aaLoadSmooth					
Smoothed drive load in %					
%				Double	r
Multi-line: yes	Axis Number		maxnumGlobMachAxes		

aaMachax					
The NCU and machine axis are output for an axis, representing the physical image of the axis. The machine axis must be assigned to at least one channel on this NCU, otherwise 0 will be returned. Without an NCU link, i.e. if there is only one NCU, only the number of the machine axis will be output. In this case, the NCU ID is equal to zero. The NCU ID is output as from position 10000, e.g. 20005: NCU 2 axis 5.					
-	0	0		UDoubleword	r
Multi-line: yes	Axis Number		maxnumGlobMachAxes		

aaMaslDef					
Each slave axis currently coupled via master-slave delivers the machine axis number of the corresponding master axis. Zero is displayed as default if the coupling is not configured. A master axis also shows default value zero. 0: No coupling for this axis configured, or axis is master axis, or no coupling active >0: Machine axis number of the master axis with which the slave axis is currently coupled					
-	0	0	numGlobMachAxes	UWord	r
Multi-line: yes	Axis Number		maxnumGlobMachAxes		

aaMaslState					
Each slave axis currently coupled via master-slave delivers the machine axis number of the corresponding master axis. Zero is displayed as default for inactive coupling. A master axis also shows default value zero. 0: No coupling for this axis configured, or axis is master axis, or no coupling active >0: Machine axis number of the master axis with which the slave axis is currently coupled					
-	0	0	numGlobMachAxes	UWord	r
Multi-line: yes	Axis Number		maxnumGlobMachAxes		

aaMeaAct					
Axial measuring active 0: Measuring system is not active 1: Measuring system is active					
-	0	0	1	UWord	r
Multi-line: yes	Axis Number		maxnumGlobMachAxes		

aaMm					
Latched probe position in the machine coordinate system					
mm, inch, degree, user defined				Double	rw
Multi-line: yes	Axis Number		maxnumGlobMachAxes		

aaMm1					
Access to measurement result of trigger event in the MCS					
mm, inch, degree, user defined				Double	rw
Multi-line: yes	Axis Number		maxnumGlobMachAxes		

aaMm2					
Access to measurement result of trigger event in the MCS					
mm, inch, degree, user defined				Double	rw
Multi-line: yes	Axis Number		maxnumGlobMachAxes		

aaMm3					
Access to measurement result of trigger event in the MCS					
mm, inch, degree, user defined				Double	rw
Multi-line: yes	Axis Number		maxnumGlobMachAxes		

aaMm4					
Access to measurement result of trigger event in the MCS					
mm, inch, degree, user defined				Double	rw
Multi-line: yes	Axis Number		maxnumGlobMachAxes		

aaOff					
Superimposed position offset from synchronous actions					
mm, inch, degree, user defined				Double	r
Multi-line: yes	Axis Number		maxnumGlobMachAxes		

aaOffLimit					
Limit for axial correction $AA_OFF reached (Note: for SYNACT only) 0: Limit value not reached 1: Limit value reached in positive axis direction 11: Limit value reached in negative axis direction					
-				UWord	r
Multi-line: yes	Axis Number		maxnumGlobMachAxes		

aaOffVal					
Integrated value of overlaid motion for an axis. The negative value of this variable can be used to cancel an overlaid motion. e.g. $AA_OFF[axis] = -$AA_OFF_VAL[axis]					
-	0			Double	r
Multi-line: yes	Axis Number		maxnumGlobMachAxes		

aaOnFixPoint					
Current fixed point, number of the fixed point at which the axis stands					
-	0			UDoubleword	r
Multi-line: yes	Axis Number		maxnumGlobMachAxes		

aaOscillBreakPos1					
Oscillation interrupt position 1					
mm, inch, degree, user defined				Double	r
Multi-line: yes	Axis Number		maxnumGlobMachAxes		

aaOscillBreakPos2					
Oscillation interrupt position 2					
mm, inch, degree, user defined				Double	r
Multi-line: yes	Axis Number		maxnumGlobMachAxes		

aaOscillReversePos1					
Current reverse position 1 for oscillation in the BCS. For synchronous actions the value of the setting data $SA_OSCILL_REVERSE_POS1 is evaluated online; (note: SYNACT only)					
mm, inch, degree, user defined				Double	r
Multi-line: yes	Axis Number		maxnumGlobMachAxes		

aaOscillReversePos2					
Current reverse position 2 for oscillation in the BCS; For synchronous actions the value of the setting data $SA_OSCILL_REVERSE_POS1 is evaluated online; (note: SYNACT only)					
mm, inch, degree, user defined				Double	r
Multi-line: yes	Axis Number		maxnumGlobMachAxes		

aaOvr					
Axial override for synchronous actions					
-				Double	r
Multi-line: yes	Axis Number		maxnumGlobMachAxes		

aaPlcOvr					
Axial override specified by PLC for motion-synchronous actions					
-	100	0		Double	r
Multi-line: yes	Axis Number		maxnumGlobMachAxes		

aaPolfa					
The programmed retraction position of the single axis					
mm, inch, degree, user defined				Double	r
Multi-line: yes	Axis Number		maxnumGlobMachAxes		

aaPolfaValid					
States whether the retraction of the single axis is programmed 0: No retraction programmed for the single axis 1: Retraction programmed as position 2: Retraction programmed as distance					
-	0	0	2	UWord	r
Multi-line: yes	Axis Number		maxnumGlobMachAxes		

aaPosRes					
Axis status "Position restored". The value TRUE shows that the position of the axis has been restored after the voltage breakdown. ($MA_ENC_REFP_STATE[] = 3). After referencing of the axis, the value goes to FALSE. 1 = TRUE: Axis position not restored 0 = FALSE: Axis position restored					
-				UWord	r
Multi-line: yes	Axis Number		maxnumGlobMachAxes		

aaPower					
Drive power in W (only available for PROFIdrive drives)					
W				Double	r
Multi-line: yes	Axis Number		maxnumGlobMachAxes		

aaPowerSmooth					
Smoothed drive power in W (only available for PROFIdrive drives)					
W				Double	r
Multi-line: yes	Axis Number		maxnumGlobMachAxes		

aaProgIndexAxPosNo					
Programmed indexing position 0: No indexing axis, therefore no indexing position available >0: Number of the programmed indexing position					
-	0	0		UWord	r
Multi-line: yes	Axis Number		maxnumGlobMachAxes		

aaRef					
Axis is referenced 0: Axis is not referenced 1: Axis is referenced					
-	0	0	1	UWord	r
Multi-line: yes	Axis Number		maxnumGlobMachAxes		

aaReposDelay					
REPOS suppression active 0: REPOS suppression is currently not active for this axis 1: REPOS suppression is currently active for this axis					
-	0	0	1	UWord	r
Multi-line: yes	Axis Number		maxnumGlobMachAxes		

aaScPar					
Current setpoint parameter set					
-	0	0		Long Integer	r
Multi-line: yes	Axis Number		maxnumGlobMachAxes		

aaSnglAxStat					
Display status of a PLC-controlled axis 0: Not a single axis 1: Reset 2: Ended 3: Interrupted 4: Active 5: Alarm					
-	0			UWord	r
Multi-line: yes	Axis Number		maxnumGlobMachAxes		

aaSoftendn					
Software end position, negative direction					
-				Double	r
Multi-line: yes	Axis Number		maxnumGlobMachAxes		

aaSoftendp					
Software end position, positive direction					
-				Double	r
Multi-line: yes	Axis Number		maxnumGlobMachAxes		

aaStat					
Axis state 0: no axis state available 1: travel command is active 2: axis has reached the IPO end. only for channel axes 3: axis in position (exact stop coarse) for all axes 4: axis in position (exact stop fine) for all axes					
-				UWord	r
Multi-line: yes	Axis Number		maxnumGlobMachAxes		

aaSync					
Coupling status of the following axis with master value coupling 0: No synchronism 1: Synchronism coarse 2: Synchronism fine 3: Synchronism coarse and fine					
-				UWord	r
Multi-line: yes	Axis Number		maxnumGlobMachAxes		

aaSyncDiff					
Setpoint synchronism difference					
mm, inch, degree, user defined	0	0		Double	r
Multi-line: yes	Axis Number		maxnumGlobMachAxes		

aaSyncDiffStat					
Status of the setpoint synchronism difference -4: No valid value in aaSyncDiff, coupled motion from part program -3: Reserved -2: Reserved -1: No valid value in aaSyncDiff 0: No valid value in aaSyncDiff, coupling not active 1: Valid value in aaSyncDiff					
-	0	-4	1	Long Integer	r
Multi-line: yes	Axis Number		maxnumGlobMachAxes		

aaTorque					
Drive torque setpoint in Nm (only available for PROFIdrive drives)					
Nm				Double	r
Multi-line: yes	Axis Number		maxnumGlobMachAxes		

aaTotalOvr					
The total axial override for motion-synchronous actions					
-	100	0		Double	r
Multi-line: yes	Axis Number		maxnumGlobMachAxes		

aaTravelCount					
Total traverse processes of an axis					
-		0		Double	r
Multi-line: yes	Axis Number		maxnumGlobMachAxes		

aaTravelCountHS					
Total traverse processes of an axis at high speed					
-		0		Double	r
Multi-line: yes	Axis Number		maxnumGlobMachAxes		

aaTravelDist					
Total travel path of an axis in mm or degrees					
mm, inch, degree, user defined		0		Double	r
Multi-line: yes	Axis Number		maxnumGlobMachAxes		

aaTravelDistHS					
Total travel path of an axis at high speed in mm or degrees					
mm, inch, degree, user defined		0		Double	r
Multi-line: yes	Axis Number		maxnumGlobMachAxes		

aaTravelTime					
Total traverse time of an axis in seconds					
s, user defined		0		Double	r
Multi-line: yes	Axis Number		maxnumGlobMachAxes		

aaTravelTimeHS					
Total traverse time of an axis at high speed in seconds					
s, user defined		0		Double	r
Multi-line: yes	Axis Number		maxnumGlobMachAxes		

aaTyp					
Axis type 0: axis in other channel 1: channel axis of same channel 2: neutral axis 3: PLC axis 4: reciprocating axis 5: neutral axis, currently traversing in JOG 6: slave axis coupled via master value 7: coupled motion slave axis 8: command axis 9: compile cycle axis					
-				UWord	r
Multi-line: yes	Axis Number		maxnumGlobMachAxes		

aaType					
Cross-channel axis type 0: Axis type cannot be determined 1: NC program axis 2: Neutral axis 3: PLC axis 4: Reciprocating axis 5: Neutral axis that is currently executing a JOG or homing motion 6: Following axis coupled to the master value 7: Coupled motion of the following axis, activated in a synchronized action 8: Command axis 9: Compile Cycle axis 10: Coupled slave axis (master-slave function.) 11: Program axis that is currently executing a JOG or homing motion					
-	0	0	11	UWord	r
Multi-line: yes	Axis Number		maxnumGlobMachAxes		

aaVactB					
Axis velocity in basic coordinate system					
mm/min, inch/min, user defined	0.0			Double	r
Multi-line: yes	Axis Number		maxnumGlobMachAxes		

aaVactM					
Axis velocity in machine coordinate system					
mm/min, inch/min, user defined	0.0			Double	r
Multi-line: yes	Axis Number		maxnumGlobMachAxes		

aaVc					
Additive correction value for path feed or axial feed					
mm/min, inch/min, user defined				Double	r
Multi-line: yes	Axis Number		maxnumGlobMachAxes		

acRpValid					
Reapproach position valid 0: Reapproach position not valid 1: Reapproach position valid					
-	0	0	1	UWord	r
Multi-line: yes	Axis Number		maxnumGlobMachAxes		

ackSafeMeasPos					
Confirmation of SI actual position 0 = not confirmed 0x00AC = confirmed					
-				UWord	rw
Multi-line: yes	Axis Number		maxnumGlobMachAxes		

actCouppPosOffset					S3
Position offset of an axis to a leading axis / leading spindle (actual value)					
mm, inch, degree, user defined		0	360	Double	r
Multi-line: yes	Axis Number		maxnumGlobMachAxes		

actFeedRate					S5
Actual value of axis-specific feedrate, if the axis is a positioning axis.					
mm/min, inch/min, user defined				Double	r
Multi-line: yes	Axis Number		maxnumGlobMachAxes		

actIndexAxPosNo					
Current indexing position number 0 = no indexing position >0 = indexing position number					
-				UWord	r
Multi-line: yes	Axis Number		maxnumGlobMachAxes		

actSpeedRel					
Actual value of rotary speed (referring to the maximum speed in %), for linear drives actual value of the velocity.					
%				Double	r
Multi-line: yes	Axis Number		maxnumGlobMachAxes		

actValResol					
Actual value resolution. The physical unit is defined in measUnit (in this module)					
mm, inch, degree, user defined				Double	r
Multi-line: yes	Axis Number		maxnumGlobMachAxes		

activeSvOverride					
Currently active SG override factor in the NCK					
-	-1	-1	100	Long Integer	r
Multi-line: yes	Axis Number		maxnumGlobMachAxes		

amSetupState					
State variable of the PI Service Automatic set-up of an asynchronous motor 0 = inactive 1 = wait for PLC enable 2 = wait for key NC-start 3 = active 4 = stopped by Servo + fine code in the upper byte 5 = stopped by 611D + fine code in the upper byte 6 = stopped by NCK + fine code in the upper byte					
-	0	0	0xff06	UWord	r
Multi-line: yes	Axis Number		maxnumGlobMachAxes		

axComp					
Sum of axis-specific compensation values (CEC Cross Error compensation and temperature compensation). The physical unit is defined in measUnit (in this module).					
mm, inch, degree, user defined				Double	r
Multi-line: yes	Axis Number		maxnumGlobMachAxes		

axisActiveInChan					
Flag indicating whether axis is active in this channel 0 = inactive 1 = active					
-				UWord	r
Multi-line: yes	Axis Number		maxnumGlobMachAxes		

axisFeedRateUnit					
Unit of axial feedrate 0 = mm/min 1 = inch/min 2 = degree/min					
-				UWord	r
Multi-line: yes	Axis Number		maxnumGlobMachAxes		

chanAxisNoGap					
Display of existing axis, i.e. no axis gap in channel. 0: Axis does not exist 1: Axis does exist					
-	0	0	1	UWord	r
Multi-line: yes	Axis Number		maxnumGlobMachAxes		

chanNoAxisIsActive					
Channel number in which the channel axis is currently active 0 = axis is not assigned to any channel 1 to maxnumChannels (Area.:N / Module:Y) = channel number					
-				UWord	r
Multi-line: yes	Axis Number		maxnumGlobMachAxes		

clampStatus					
Axis is connected (VDI input signal) Bit 0 = 1: Axis is connected					
-	0	0	1	UWord	r
Multi-line: no			maxnumGlobMachAxes		

cmdContrPos					
Desired value of position after fine interpolation					
mm, inch, degree, user defined				Double	r
Multi-line: yes	Axis Number		maxnumGlobMachAxes		

cmdCouppPosOffset					S3
Position offset of an axis referring to the leading axis / leading spindle (desired value)					
mm, inch, degree, user defined		0	360	Double	r
Multi-line: yes	Axis Number		maxnumGlobMachAxes		

cmdFeedRate					
Desired value of axis-specific feedrate for a positioning axis.					
mm/min, inch/min, user defined				Double	r
Multi-line: yes	Axis Number		maxnumGlobMachAxes		

cmdSpeedRel					
Speed setpoint (as % of the maximum speed), velocity setpoint in the case of linear motors.					
%				Double	r
Multi-line: yes	Axis Number		maxnumGlobMachAxes		

contrConfirmActive					
Controller enable 0 = no controller enable 1 = controller enable					
-				UWord	r
Multi-line: yes	Axis Number		maxnumGlobMachAxes		

contrMode					
Identifier for controller mode servo 0 = position control 1 = speed control 2 = stop 3 = park 4 = follow-up (set the mode through VDI interface and partly through part program)					
-				UWord	r
Multi-line: yes	Axis Number		maxnumGlobMachAxes		

displayAxis					
Identifier indicating whether axis is displayed by HMI as a machine axis. 0 = Do not display at all 0xFFFF = Always display everything bit 0 = Display in actual-value window bit 1 = Display in reference point window bit 2 = Display in Preset / Basic offset / Scratching bit 3 = Display in handwheel selection					
-	0xFFFF	0	0xFFFF	UWord	r
Multi-line: yes	Axis Number		maxnumGlobMachAxes		

distPerDriveRevol					
Rotary drive: Load-side path corresponding to one revolution of the drive. Is returned in the unit of the internal computational resolution INT_INCR_PER_MM (for linear axes) or INT_INCR_PER_DEG (for rotary axes / spindles) taking into account gear ratios etc. In the case of linear axes, the pitch of the ball screw is also included in the calculation. In the case of linear motors, a fixed value of "1mm" is used for the ball screw pitch instead of the non-existent ball screw.					
mm, inch, degree, user defined				Double	r
Multi-line: yes	Axis Number		maxnumGlobMachAxes		

drfVal					
DRF value					
-	0			Double	r
Multi-line: yes	Axis Number		maxnumGlobMachAxes		

drive2ndTorqueLimit					
2nd torque limit. With linear motors: 2nd force limit 0 = inactive 1 = active					
-				UWord	r
Multi-line: yes	Axis Number		maxnumGlobMachAxes		

driveActMotorSwitch					
Actual motor wiring (star/delta) 0 = star 1 = delta					
-				UWord	r
Multi-line: yes	Axis Number		maxnumGlobMachAxes		

driveActParamSet					
Number of the actual drive parameter set					
-		1	8	UWord	r
Multi-line: yes	Axis Number		maxnumGlobMachAxes		

driveClass1Alarm					
Message ZK1 drive alarm 0 = no alarm set 1= alarm set (fatal error occured)					
-				UWord	r
Multi-line: yes	Axis Number		maxnumGlobMachAxes		

driveContrMode					
Control mode of drive 0 = current control 1 = speed control					
-				UWord	r
Multi-line: yes	Axis Number		maxnumGlobMachAxes		

driveCoolerTempWarn					
Heatsink temperature monitoring 0 = temperature OK 1 = overtemperature					
-				UWord	r
Multi-line: yes	Axis Number		maxnumGlobMachAxes		

driveDdsPerMds					
Number of drive data sets assigned to a motor data set. Refer to the SINAMICS S120 Function Manual for more information about drive and motor data sets.					
-				UWord	r
Multi-line: yes	Axis Number		maxnumGlobMachAxes		

driveDesMotorSwitch					
Motor wiring selection (star/delta) 0 = star 1 = delta					
-				UWord	r
Multi-line: yes	Axis Number		maxnumGlobMachAxes		

driveDesParamSet					
Desired parameter set of the drive					
-		1	8	UWord	r
Multi-line: yes	Axis Number		maxnumGlobMachAxes		

driveFastStop					
Ramp-function generator rapid stop 0 = not stopped 1 = stopped					
-				UWord	r
Multi-line: yes	Axis Number		maxnumGlobMachAxes		

driveFreqMode					
I/F mode					
-				UWord	r
Multi-line: yes	Axis Number		maxnumGlobMachAxes		

driveImpulseEnabled					
Enable inverter impulse (checkback signal to impulseEnable) 0 = not enabled 1 = enabled					
-				UWord	r
Multi-line: yes	Axis Number		maxnumGlobMachAxes		

driveIndex					
Drive assignment (logical drive number) 0 = drive does not exist 1 to 15 = logical drive number					
-		0	15	UWord	r
Multi-line: yes	Axis Number		maxnumGlobMachAxes		

driveIntegDisable					
Integrator disable 0 = not disabled 1 = disabled					
-				UWord	r
Multi-line: yes	Axis Number		maxnumGlobMachAxes		

driveLinkVoltageOk					
State of the DC link voltage 0 = OK 1 = not OK					
-				UWord	r
Multi-line: yes	Axis Number		maxnumGlobMachAxes		

driveMotorTempWarn					
Motor temperature warning 0 = temperature OK 1 = overtemperature					
-				UWord	r
Multi-line: yes	Axis Number		maxnumGlobMachAxes		

driveNumCrcErrors					
CRC errors on the drive bus (Transmission errors when writing data to the drive; values may range up to FFFFH) 0 = no error					
-				UWord	r
Multi-line: yes	Axis Number		maxnumGlobMachAxes		

driveParked					
Parking axis 0 = no parking axis 1 = parking axis					
-				UWord	r
Multi-line: yes	Axis Number		maxnumGlobMachAxes		

drivePowerOn					
Drive switched on 0 = drive not switched on 1 = drive switched on					
-				UWord	r
Multi-line: yes	Axis Number		maxnumGlobMachAxes		

driveProgMessages					
Configurable messages (via machine data)					
-				UWord	r
Multi-line: yes	Axis Number		maxnumGlobMachAxes		

driveReady					
Drive ready 0 = drive not ready 1 = drive ready					
-				UWord	r
Multi-line: yes	Axis Number		maxnumGlobMachAxes		

driveRunLevel					
Current state reached during the boot process (range: coarse state (0 to 5) * 100 + fine state (up to 22) Booting the firmware ---> 0 XX entering the configuration ---> 1XX hardware-init, communication-init loading, converting data ---> 2XX changing bus addressing ---> 3XX preparing synchronization ---> 4XX activating interrupt ---> 519 XX ==> fine state					
-				UWord	r
Multi-line: yes	Axis Number		maxnumGlobMachAxes		

driveSetupMode					
Set-up mode 0 = inactive 1 = active					
-				UWord	r
Multi-line: yes	Axis Number		maxnumGlobMachAxes		

driveSpeedSmoothing					
Smoothing the desired value of the rotary speed, for linear drives: smoothing the desired value of the velocity 0 = no smoothing 1 = smoothing					
-				UWord	r
Multi-line: yes	Axis Number		maxnumGlobMachAxes		

effComp1					
Sum of the compensation values for encoder 1. The value results from: Temperature compensation, backlash compensation, quadrant error compensation, beam sag compensation, leadscrew error compensation. The physical unit is defined in measUnit (in this module).					
mm, inch, degree, user defined				Double	r
Multi-line: yes	Axis Number		maxnumGlobMachAxes		

effComp2					
Sum of the compensation values for encoder 2. The value results from: Temperature compensation, backlash compensation, quadrant error compensation, beam sag compensation, leadscrew error compensation. The physical unit is defined in measUnit (in this module).					
mm, inch, degree, user defined				Double	r
Multi-line: yes	Axis Number		maxnumGlobMachAxes		

enc1IsOn					
Operating status of position measuring system 1 0 = Position measuring system 1 parked (or is not configured), may be removed 1 = Position measuring system 1 is passive 2 = Position measuring system 1 is active (e.g. position control)					
-		0	2	UWord	r
Multi-line: yes	Axis Number		maxnumGlobMachAxes		

enc2IsOn					
Operating status of position measuring system 2 0 = Position measuring system 2 parked (or is not configured), may be removed 1 = Position measuring system 2 is passive 2 = Position measuring system 2 is active (e.g. position control)					
-		0	2	UWord	r
Multi-line: yes	Axis Number		maxnumGlobMachAxes		

encChoice					
Active encoder 0 = does not exist 1 = encoder 1 2 = encoder 2					
-				UWord	r
Multi-line: yes	Axis Number		maxnumGlobMachAxes		

fctGenState					
State of the function generator					
-				UWord	r
Multi-line: yes	Axis Number		maxnumGlobMachAxes		

feedRateOvr					
Feedrate override (only if axis is a positioning axis)					
%				Double	r
Multi-line: yes	Axis Number		maxnumGlobMachAxes		

focStat					
Current status of "Travel with limited torque" function 0-2 0: FOC not active 1: FOC modal active (programming of FOCON[]) 2: FOC non-modal active (programming of FOC[])					
-	0	0	2	UWord	r
Multi-line: yes	Axis Number		maxnumGlobMachAxes		

fxsInfo					
Additional information on travel to fixed stop if $VA_FXS[]=2, or OPI variable fxsStat=2. 0 No additional information available 1 No approach motion programmed 2 Programmed end position reached, movement ended 3 Abort by NC RESET (Reset key) 4 Fixed stop window exited 5 Torque reduction was rejected by drive 6 PLC has canceled enable signals					
-	0	0	6	UWord	r
Multi-line: yes	Axis Number		maxnumGlobMachAxes		

fxsStat					
State after travelling to fixed stop 0 = normal control, no clamping 1 = fixed stop reached, clamping active 2 = selection failed 3 = selection active 4 = stop detected 5 = deselection active					
-				UWord	r
Multi-line: yes	Axis Number		maxnumGlobMachAxes		

handwheelAss					
Number of handwheel assigned to axis 0 = no handwheel assigned 1 to 3 = handwheel number					
-		0	3	UWord	r
Multi-line: yes	Axis Number		maxnumGlobMachAxes		

impulseEnable					
Impulse enable for drive 0 = not enabled 1 = enabled					
-				UWord	r
Multi-line: yes	Axis Number		maxnumGlobMachAxes		

index					
Absolute axis index referred to MD					
-				UWord	r
Multi-line: yes	Axis Number		maxnumGlobMachAxes		

isDriveUsed					
One or more machine axes are assigned to each drive. The drive can only be controlled at any one time by one of these machine axes. The machine manufacturer makes the selection. The status of the drive control changes dynamically.					
-	0	0	1	UWord	r
Multi-line: yes	Axis Number		maxnumGlobMachAxes		

kVFactor					
position control gain factor					
16.667 1/s				Double	r
Multi-line: yes	Axis Number		maxnumGlobMachAxes		

lag					
Following error = desired value of position after fine interpolation - actual value of position. The physical unit is defined in measUnit (in this module).					
mm, inch, degree, user defined				Double	r
Multi-line: yes	Axis Number		maxnumGlobMachAxes		

logDriveNo					
Drive assignment (logical drive number) 0 = not available 1 to 15 = drive number					
-		0	15	UWord	r
Multi-line: yes	Axis Number		maxnumGlobMachAxes		

measFctState					
State of the probing function					
-				UWord	r
Multi-line: yes	Axis Number		maxnumGlobMachAxes		

measPos1					
Actual value of position for encoder 1. The physical unit is defined in measUnit (in this module).					
mm, inch, degree, user defined				Double	r
Multi-line: yes	Axis Number		maxnumGlobMachAxes		

measPos2					
Actual value of position for encoder 2. The physical unit is defined in measUnit (in this module).					
mm, inch, degree, user defined				Double	r
Multi-line: yes	Axis Number		maxnumGlobMachAxes		

measPosDev					
Actual position difference between the two encoders. The physical unit is defined in measUnit (in this module).					
mm, inch, degree, user defined				Double	r
Multi-line: yes	Axis Number		maxnumGlobMachAxes		

measUnit					
Unit for service values of the drives 0 = mm 1 = inch 2 = grd					
-				UWord	r
Multi-line: yes	Axis Number		maxnumGlobMachAxes		

paramSetNo					
Number of parameter set					
-		1	8	UWord	r
Multi-line: yes	Axis Number		maxnumGlobMachAxes		

preContrFactTorque					
Feed forward control factor torque					
Nm				Double	r
Multi-line: yes	Axis Number		maxnumGlobMachAxes		

preContrFactVel					
Feed forward control factor velocity					
-				Double	r
Multi-line: yes	Axis Number		maxnumGlobMachAxes		

preContrMode					
Feed forward control mode 0 = inactive 1 = velocity feed forward 2 = torque feed forward					
-				UWord	r
Multi-line: yes	Axis Number		maxnumGlobMachAxes		

progIndexAxPosNo					
Programmed indexing position number 0 = no indexing position >0 = indexing position number					
-				UWord	r
Multi-line: yes	Axis Number		maxnumGlobMachAxes		

qecLrnIsOn					
Quadrant error compensation learning active 0 = inactive 1 = Neuronal-QEC learning active 2 = Standard-QEC active 3 = Standard-QEC with adaptation of correction value active 4 = Neuronal-QEC active 5 = Neuronal-QEC with adaptation of measuring time active 6 = Neuronal-QEC with adaptation of decay time of correction value active 7 = Neuronal-QEC with adaptation of measuring time and decay time of correction value active					
-		0	7	UWord	r
Multi-line: yes	Axis Number		maxnumGlobMachAxes		

refPtBusy					
Axis is being referenced 0 = axis is not being referenced 1 = axis is being referenced					
-				UWord	r
Multi-line: yes	Axis Number		maxnumGlobMachAxes		

refPtCamNo					
Reference point cam 0 = no cam approached 1 = cam 1 2 = cam 2 3 = cam 3 4 = cam 4					
-				UWord	r
Multi-line: yes	Axis Number		maxnumGlobMachAxes		

refPtPhase					
Referencing phases 0 = False 1 = Phase 1 2 = Phase 2 3 = Phase 3 4 = Phase 4					
-				UWord	r
Multi-line: yes	Axis Number		maxnumGlobMachAxes		

refPtStatus					
Identifier indicating whether an axis requires referencing and is referenced. Note regarding exchange axes: An exchange axis need only ever be referenced in the channel to which it is currently assigned. A referenced exchange axis is thus logged onto the channel in which it is traversing with value "3" (requires referencing and referenced) and in other channels with value "1" (does not require referencing, but referenced). Set bits have the following meanings: Bit0: current measuring system has been referenced Bit1: current measuring system requires referencing (A busy signal affects the state)					
-	Achsindex			UWord	r
Multi-line: no			maxnumGlobMachAxes		

resolvStatus1					
Encoder status for measuring system 1 0 = Undefined 1 = Referenced 2 = Activated 3 = Limit frequency exceeded					
-				UWord	r
Multi-line: yes	Axis Number		maxnumGlobMachAxes		

resolvStatus2					
Encoder status for measuring system 2 0 = Undefined 1 = Referenced 2 = Activated 3 = Limit frequency exceeded					
-				UWord	r
Multi-line: yes	Axis Number		maxnumGlobMachAxes		

safeAcceptCheckPhase					
Flag for NCK-side acceptance test phase, the human-machine interface can determine which acceptance test phase is present on the NCK. 0: NCK has acceptance test phase inactive = 0 0ACH: NCK has acceptance test phase active					
-	0	0	0ACH	UWord	r
Multi-line: yes	Axis Number		maxnumGlobMachAxes		

safeAcceptTestMode					
SI PowerOn alarms can be acknowledged by Reset in acceptance test mode 0: Acceptance test mode: SI PowerOn alarms cannot be acknowledged by Reset 0ACH: Acceptance test mode: SI PowerOn alarms can be acknowledged by Reset					
-	0	0	0FFH	UWord	rw
Multi-line: yes	Axis Number		maxnumGlobMachAxes		

safeAcceptTestPhase					
Flag for acceptance test phase 0: Acceptance test Wizard not selected, activate NCK-side alarm suppression 0ACH: Dialogs for acceptance test support selected, deactivate NCK-side alarm suppression					
-	0	0	0FFH	UWord	rw
Multi-line: yes	Axis Number		maxnumGlobMachAxes		

safeAcceptTestSE					
Flag for NCK-side SE acceptance test. The human-machine interface starts checking the safe limit positions during the acceptance test 0: NCK has SE acceptance test inactive = 0. The single channel SW limit positions are activated. 0ACH: NCK is to activate SE acceptance test. The single channel SW limit positions are deactivated in this way.					
-	0	0	0ACH	UWord	r
Multi-line: yes	Axis Number		maxnumGlobMachAxes		

safeAcceptTestState					
Flag for acceptance test status, the human-machine interface can determine which acceptance test mode is present on the NCK. 0: NCK has inactive acceptance test mode 0CH: Acceptance test mode not activated because SI PowerOn alarms already present. The causes of the SI PowerOn alarms must be eliminated first. 0DH: Acceptance test mode not activated, the HMI writes invalid values in safeAcceptTestMode to the NCK. 0ACH: NCK has active acceptance test mode					
-	0	0	0FFH	UWord	r
Multi-line: yes	Axis Number		maxnumGlobMachAxes		

safeActPosDiff					
Current actual value difference betw. NCK and drive monitoring channels					
mm, inch, degree, user defined	0.0			Double	r
Multi-line: yes	Axis Number		maxnumGlobMachAxes		

safeActVeloDiff					
Current speed difference between NCK and drive monitoring channels					
mm/min, inch/min, user defined	0.0			Double	r
Multi-line: yes	Axis Number		maxnumGlobMachAxes		

safeActVeloLimit					
Safe limit of actual speed -1 => no actual speed limit active >= 0 => limit of actual speed is active					
mm, inch, degree, user defined		-1		Double	r
Multi-line: no			maxnumGlobMachAxes		

safeActiveCamTrack					
Status Safe cam track (active/inactive) Bit 0 = 1/0: Safe cam track 1 active/inactive Bit 1 = 1/0: Safe cam track 2 active/inactive Bit 2 = 1/0: Safe cam track 3 active/inactive Bit 3 = 1/0: Safe cam track 4 active/inactive					
-	0	0	0xF	UWord	r
Multi-line: no			maxnumGlobMachAxes		

safeAxisType					
Type of axial safety monitoring 0 = No SINUMERIK Safety Integrated active 1 = SINUMERIK Safety Integrated (SPL) active 2 = SINUMERIK Safety Integrated plus (F-PLC) active					
-				UWord	r
Multi-line: yes	Axis Number		maxnumGlobMachAxes		

safeDesVeloLimit					
Safe limit of desired speed -1 => no desired speed limit active >= 0 => desired speed limit is active					
mm, inch, degree, user defined		-1		Double	r
Multi-line: no			maxnumGlobMachAxes		

safeFctEnable					
Safe operation active (Safety Integrated / SPL) 0 = inactive >0 = active					
-				UWord	r
Multi-line: yes	Axis Number		maxnumGlobMachAxes		

safeInputSig					
Safe input signals of the axis					
-				UWord	r
Multi-line: yes	Axis Number		maxnumGlobMachAxes		

safeInputSig2					
Safe input signals part 2					
-		0	0xffff	UWord	r
Multi-line: no			maxnumGlobMachAxes		

safeInputSigDrive					
Safe input signals of the drive					
-				UWord	r
Multi-line: yes	Axis Number		maxnumGlobMachAxes		

safeInputSigDrive2					
Safe input signals of the drive part 2					
-		0	0xffff	UWord	r
Multi-line: no			maxnumGlobMachAxes		

safeMaxVeloDiff					
Maximum speed difference between NCK and drive monitoring channels since last NCK Reset					
mm/min, inch/min, user defined	0.0			Double	r
Multi-line: yes	Axis Number		maxnumGlobMachAxes		

safeMeasPos					
Safe actual position of the axis. The physical unit is defined in the variable measUnit (in this module).					
mm, inch, degree, user defined				Double	r
Multi-line: yes	Axis Number		maxnumGlobMachAxes		

safeMeasPosDrive					
Safe actual position of drive. The physical unit is defined in measUnit (in this module).					
mm, inch, degree, user defined				Double	r
Multi-line: yes	Axis Number		maxnumGlobMachAxes		

safeOutputSig					
Safe output signals of the axis					
-				UWord	r
Multi-line: yes	Axis Number		maxnumGlobMachAxes		

safeOutputSig2					
Safe output signals part 2					
-		0	0xffff	UWord	r
Multi-line: no			maxnumGlobMachAxes		

safeOutputSigCam					
Results of the NCK safe cam evaluation					
-	0	0	3FFFFFFF	Long Integer	r
Multi-line: no			maxnumGlobMachAxes		

safeOutputSigCamDrive					
Results of the drive safe cam evaluation					
-	0	0	3FFFFFFF	Long Integer	r
Multi-line: no			maxnumGlobMachAxes		

safeOutputSigDrive					
Safe output signals of the drive					
-				UWord	r
Multi-line: yes	Axis Number		maxnumGlobMachAxes		

safeOutputSigDrive2					
Safe output signals of the drive part 2					
-		0	0xffff	UWord	r
Multi-line: no			maxnumGlobMachAxes		

safePosCtrlActive					
Axis monitors absolute position 0 = Axis does not monitor absolute position (no SE/SN) 1 = Axis monitors absolute position					
-	0	0	1	UWord	r
Multi-line: no			maxnumGlobMachAxes		

safeStopOtherAxis					
Stop on another axis 0: No stop on another axis 1: Stop on another axis					
-	0	0	1	UWord	r
Multi-line: yes	Axis Number		maxnumGlobMachAxes		

spec					
Axis specification 0 = path axis 1 = positioning axis					
-				UWord	r
Multi-line: yes	Axis Number		maxnumGlobMachAxes		

spindleModePiState					
Status of the spindle mode switchover for this machine axis by PI service _N_SPIMOD 0 = PI service not selected 10 = PI service active 50 = PI service ended successfully 101 = PI service rejected because axis/spindle is not known in the channel 102 = PI service rejected because axis/spindle is not available in the channel 104 = PI service rejected because axis/spindle is not defined as a spindle. 105 = PI service rejected because axis/spindle is a permanently assigned PLC axis/spindle 106 = PI service rejected because axis/spindle is an active following axis/spindle 107 = PI service rejected because axis/spindle is a transformed spindle/axis 108 = PI service rejected because axis/spindle is not available as a command axis 200 = PI service rejected because of an internal error					
-	0	0	999	UWord	r
Multi-line: yes	Axis Number		maxnumGlobMachAxes		

stateContrActive					
State controller (not available) 1 = TRUE 0 = FALSE					
-				UWord	r
Multi-line: yes	Axis Number		maxnumGlobMachAxes		

subSpec					T1
Subspecification 0 = normal axis 1 = indexing axis					
-				UWord	r
Multi-line: yes	Axis Number		maxnumGlobMachAxes		

torqLimit					
Torque limitation value (referring to the nominal value of the drive). For linear motors: force limitation value.					
%				Double	r
Multi-line: yes	Axis Number		maxnumGlobMachAxes		

traceState1					
State of trace channel 1 0 = idle state 1 = recording started 2 = trigger reached 3 = recording ended 4 = recording aborted					
-				UWord	r
Multi-line: yes	Axis Number		maxnumGlobMachAxes		

traceState2					
State of trace channel 2 0 = idle state 1 = recording started 2 = trigger reached 3 = recording ended 4 = recording aborted					
-				UWord	r
Multi-line: yes	Axis Number		maxnumGlobMachAxes		

traceState3					
State of trace channel 3 0 = idle state 1 = recording started 2 = trigger reached 3 = recording ended 4 = recording aborted					
-				UWord	r
Multi-line: yes	Axis Number		maxnumGlobMachAxes		

traceState4					
State of trace channel 4 0 = idle state 1 = recording started 2 = trigger reached 3 = recording ended 4 = recording aborted					
-				UWord	r
Multi-line: yes	Axis Number		maxnumGlobMachAxes		

trackErrContr					
Position controller difference (actual value / desired value of position)					
mm, inch, degree, user defined				Double	r
Multi-line: yes	Axis Number		maxnumGlobMachAxes		

trackErrDiff					
Contour deviation (difference actual value of position and calculated dynamical model)					
mm, inch, degree, user defined				Double	r
Multi-line: yes	Axis Number		maxnumGlobMachAxes		

type					
Axis type In the case of a link axis, the initial setting of the machine data is returned according to axisType. The difference between spindle and rotary axis cannot be determined because there is no possibility of accessing the other NCUs. Thus, in this case, there is no value 2 for spindle. 0 = linear axis 1 = rotary axis 2 = spindle					
-				UWord	r
Multi-line: yes	Axis Number		maxnumGlobMachAxes		

vaAbsoluteEnc1DeltaInit					
Enc1: Initial difference					
-	0	0		UDoubleword	r
Multi-line: yes	Axis Number		maxnumGlobMachAxes		

vaAbsoluteEnc1ErrCnt					
Enc 1: Error counter for absolute encoder					
-	0	0		Long Integer	r
Multi-line: yes	Axis Number		maxnumGlobMachAxes		

vaAbsoluteEnc1State					
Enc1: Status of absolute encoder interface Bit0: Interface is active Bit1: Error during parity check Bit2: Error bit alarm Bit3: Error bit CRC error Bit4: Start bit missing with EnDat transfer					
-	0	0		Long Integer	r
Multi-line: yes	Axis Number		maxnumGlobMachAxes		

vaAbsoluteEnc1ZeroMonMax					
Enc1:Maximum of vaEnc1ZeroMonAct with absolute encoder					
-	0	0		UDoubleword	r
Multi-line: yes	Axis Number		maxnumGlobMachAxes		

vaAbsoluteEnc2DeltaInit					
Enc2: Initial difference					
-	0	0		UDoubleword	r
Multi-line: yes	Axis Number		maxnumGlobMachAxes		

vaAbsoluteEnc2ErrCnt					
Enc 2: Error counter for absolute encoder					
-	0	0		Long Integer	r
Multi-line: yes	Axis Number		maxnumGlobMachAxes		

vaAbsoluteEnc2State					
Enc2: Status of absolute encoder interface Bit0: Interface is active Bit1: Error during parity check Bit2: Error bit alarm Bit3: Error bit CRC error Bit4: Start bit missing with EnDat transfer					
-	0	0		Long Integer	r
Multi-line: yes	Axis Number		maxnumGlobMachAxes		

vaAbsoluteEnc2ZeroMonMax					
Enc2:Maximum of vaEnc2ZeroMonAct with absolute encoder					
-	0	0		UDoubleword	r
Multi-line: yes	Axis Number		maxnumGlobMachAxes		

vaCcCompValTotal					
Axial OA total compensation value of the compile cycles					
mm, inch, degree, user defined	0			Double	r
Multi-line: yes	Axis Number		maxnumGlobMachAxes		

vaCecCompVal					
Axial sag compensation value					
mm, inch, degree, user defined	0			Double	r
Multi-line: yes	Axis Number		maxnumGlobMachAxes		

vaCpSync2					
Second synchronism monitoring of the following axis / spindle 0: Monitoring not active Bit 0 = 1: Monitoring 'Synchronism(2) coarse' active Bit 1 = 1: Synchronism(2) coarse available Bit 2 = 1: Monitoring 'Synchronism(2) fine' active Bit 3 = 1: Synchronism(2) fine available					
-				UWord	r
Multi-line: yes	Axis index of the following axis		maxnumGlobMachAxes		

vaCurr					
Drive actual current value					
-	0			Double	r
Multi-line: yes	Axis Number		maxnumGlobMachAxes		

vaDistTorque					
Disturbing torque/max. torque (motor end, York)					
%	0	-100	100	Double	r
Multi-line: yes	Axis Number		maxnumGlobMachAxes		

vaDpe					
Status of power enable of a machine axis 0 - 1					
-	0	0	1	UWord	r
Multi-line: yes	Axis Number		maxnumGlobMachAxes		

vaEnc1CompVal					
Leadscrew error compensation (LEC) value encoder 1					
mm, inch, degree, user defined	0			Double	r
Multi-line: yes	Axis Number		maxnumGlobMachAxes		

vaEnc1ZeroMonAccessCnt					
Enc1: Update counter					
-	0	0		UDoubleword	r
Multi-line: yes	Axis Number		maxnumGlobMachAxes		

vaEnc1ZeroMonAct					
Enc1: Zero monitoring values					
-	0	0		UDoubleword	r
Multi-line: yes	Axis Number		maxnumGlobMachAxes		

vaEnc1ZeroMonErrCnt					
Enc 1: Error counter for zero mark monitoring					
-	0	0		Long Integer	r
Multi-line: yes	Axis Number		maxnumGlobMachAxes		

vaEnc1ZeroMonInit					
Enc1:Hardware counter value of the basic zero mark					
-	0	0		UDoubleword	r
Multi-line: yes	Axis Number		maxnumGlobMachAxes		

vaEnc2CompVal					
Leadscrew error compensation (LEC) value encoder 2					
mm, inch, degree, user defined	0			Double	r
Multi-line: yes	Axis Number		maxnumGlobMachAxes		

vaEnc2ZeroMonAccessCnt					
Enc2: Update counter					
-	0	0		UDoubleword	r
Multi-line: yes	Axis Number		maxnumGlobMachAxes		

vaEnc2ZeroMonAct					
Enc2: Zero monitoring values					
-	0	0		UDoubleword	r
Multi-line: yes	Axis Number		maxnumGlobMachAxes		

vaEnc2ZeroMonErrCnt					
Enc 2: Error counter for zero mark monitoring					
-	0	0		Long Integer	r
Multi-line: yes	Axis Number		maxnumGlobMachAxes		

vaEnc2ZeroMonInit					
Enc2:Hardware counter value of the basic zero mark					
-	0	0		UDoubleword	r
Multi-line: yes	Axis Number		maxnumGlobMachAxes		

vaFoc					
Actual status of "ForceControl" 0: ForceControl not active 1: Modal ForceControl active 2: Non-modal ForceControl active					
-	0	0	2	UWord	r
Multi-line: yes	Axis Number		maxnumGlobMachAxes		

vaFxs					
Actual status of "Travel to fixed stop" 0: Axis not at fixed stop 1: Successful travel to fixed stop 2: Unsuccessful travel to fixed stop 3: Travel to fixed stop selection active 4: Fixed stop has been detected 5: Travel to fixed stop deselection active					
-	0	0	5	UWord	r
Multi-line: yes	Axis Number		maxnumGlobMachAxes		

vaIm					
Encoder actual value in the machine coordinate system (measured active measuring system)					
mm, inch, degree, user defined	0	0		Double	r
Multi-line: yes	Axis Number		maxnumGlobMachAxes		

vaIm1					
Actual value in the machine coordinate system (measured encoder 1)					
mm, inch, degree, user defined	0	0		Double	r
Multi-line: yes	Axis Number		maxnumGlobMachAxes		

vaIm2					
Actual value in the machine coordinate system (measured encoder 2)					
mm, inch, degree, user defined	0	0		Double	r
Multi-line: yes	Axis Number		maxnumGlobMachAxes		

vaIpoNcChanax					
If the machine axis is currently interpolated to this NCU, the channel and channel axis number which define the interpolator of the axis are output. If the machine axis is currently interpolated to a different NCU, the NCU identifier of the interpolated NCU and the global axis number of the machine axis are output. This global axis number can then be used to transfer the interpolated channel and the channel axis number to the other NCU, with NCU ID 2, with $AN_IPO_CHANAX[103]. If no machine axis is used, 0 will be returned. The channel is output as from position 100, and the channel axis number as from position 1, e.g. 1005 - channel 10 channel axis 5. These values are always lower than 10000. The NCU is output as from position 10000, e.g. 20103: NCU 2 and the global axis number is 103.					
-	0	0		UDoubleword	r
Multi-line: yes	Axis Number		maxnumGlobMachAxes		

vaLagError					
Axis following error					
-	0			Double	r
Multi-line: yes	Axis Number		maxnumGlobMachAxes		

vaLoad					
Drive utilization in %					
-	0	-100	100	Double	r
Multi-line: yes	Axis Number		maxnumGlobMachAxes		

vaMotClampingState					
Starting from the position of the draw-bar (value of the S1), this variable determines the clamping state. A maximum speed is assigned to each state. These are stored in the drive parameters p5043[0..6]. The following values are possible: 0: Sensor not present 1: Initial state, speed limit 0 rpm 2: Alarm, speed limit 0 rpm 3: Tool released / ejected, speed limit see drive parameter p5043[0] 4: Clamping (by spring force), speed limit see drive parameter p5043[1] 5: Releasing (by compressed air), speed limit see drive parameter p5043[2] 6: Releasing (by compressed air), speed limit see drive parameter p5043[3] 7: Clamped with tool, speed limit see drive parameter p5043[4] 8: Clamped with tool, speed limit see drive parameter p5043[4] 9: Further clamping (by spring force), speed limit see drive parameter p5043[5] 10: Clamped without tool, speed limit see drive parameter p5043[6] 11: Alarm, speed limit 0 rpm					
-	0	0		UDoubleword	r
Multi-line: yes	Axis Number		maxnumGlobMachAxes		

vaMotSensorAna					
This variable determines the analog measured value of sensor S1. At a resolution of 1 mV, the analog value 0 - 10 V is mapped by a maximum of +10000 increments.					
-	0	0		UDoubleword	r
Multi-line: yes	Axis Number		maxnumGlobMachAxes		

vaMotSensorConf					
The configuration of the motor sensors can be queried with this variable. The variable is bit-coded, and has the following meanings: Bit0 = 1: Sensor system present. Bit1 = 1: Sensor S1 present. Analog measured value for position of the draw-bar. Bit2 = 0: Bit3 = 0: Bit4 = 1: Sensor S4 present. Digital value for the piston end position Bit5 = 1: Sensor S5 present. Digital value for the angular position of the shaft.					
-	0	0		UDoubleword	r
Multi-line: yes	Axis Number		maxnumGlobMachAxes		

vaMotSensorDigi					
This variable determines the states of the digital sensors S4 and S5. The variable is bit-coded, and has the following meanings: Bit0 = 0: Bit1 = 0: Bit2 = 0: Bit3 = 0: Bit4 = 1: Sensor S4 piston end position Bit5 = 1: Sensor S5 angular position of the shaft					
-	0	0		UDoubleword	r
Multi-line: yes	Axis Number		maxnumGlobMachAxes		

vaPosctrlMode					
Position controller mode" 0: Position control 1: Speed control 2: Holding 3: Parking 4: Tracking					
-	0	0	4	UWord	r
Multi-line: yes	Axis Number		maxnumGlobMachAxes		

vaPower					
Active drive power					
-	0			Double	r
Multi-line: yes	Axis Number		maxnumGlobMachAxes		

vaPressureA					
Pressure on A end of the cylinder in bar (only for Hydraulic)					
-	0			Double	r
Multi-line: yes	Axis Number		maxnumGlobMachAxes		

vaPressureB					
Pressure on B end of the cylinder in bar (only for Hydraulic)					
-	0			Double	r
Multi-line: yes	Axis Number		maxnumGlobMachAxes		

vaSce					
Status of speed controller enable					
-	0	0	1	UWord	r
Multi-line: yes	Axis Number		maxnumGlobMachAxes		

vaStopSi					
Stop from Safety Integrated -1: No stop 0: Stop A 1: Stop B 2: Stop C 3: Stop D 4: Stop E 5: Stop F 10: Test stop of NC 11: Test of ext. pulse suppression					
-	0			Long Integer	r
Multi-line: yes	Axis Number		maxnumGlobMachAxes		

vaSyncDiff					
Actual value synchronism difference for all types of coupling					
mm, inch, degree, user defined	0			Double	r
Multi-line: yes	Axis Number		maxnumGlobMachAxes		

vaSyncDiffStat					
Status of the actual value synchronism difference -4: Reserved -3: No valid value in $VA_SYNCDIFF, tangential control -2: No valid value in $VA_SYNCDIFF, master value coupling and simulated master value -1: No valid value in $VA_SYNCDIFF 0: No valid value in $VA_SYNCDIFF, coupling not active 1: Valid value in $VA_SYNCDIFF					
mm, inch, degree, user defined	0	-4	1	Long Integer	r
Multi-line: yes	Axis Number		maxnumGlobMachAxes		

vaTempCompVal					
Axial temperature compensation value					
mm, inch, degree, user defined	0			Double	r
Multi-line: yes	Axis Number		maxnumGlobMachAxes		

vaTorque					
Drive torque setpoint					
-	0			Double	r
Multi-line: yes	Axis Number		maxnumGlobMachAxes		

vaTorqueAtLimit					
Status "effective torque equals specified torque limit" 0: Effective torque lower than torque limit 1: Effective torque has reached torque limit					
-	0	0	1	UWord	r
Multi-line: yes	Axis Number		maxnumGlobMachAxes		

vaVactm					
Axis velocity actual value on the load side in the MCS					
mm/min, inch/min, user defined				Double	r
Multi-line: yes	Axis Number		maxnumGlobMachAxes		

vaValveLift					
Actual valve lift in mm (only for Hydraulic)					
-	0			Double	r
Multi-line: yes	Axis Number		maxnumGlobMachAxes		

vaXfaultSi					
Stop F through cross-checking error active Bit 0 set: An actual value error has been discovered in the cross-check between NCK and drive Bit 1 set: Some error has been discovered in the cross-check between NCK and drive and the waiting time until Stop B ($MA_SAFE_STOP_SWITCH_TIME_F) is running or has expired					
-	0			Long Integer	r
Multi-line: yes	Axis Number		maxnumGlobMachAxes		

3.3.7 Area N, Block SSP : State data: Spindle

OEM-MMC: Linkitem /NckSpindle/...

All status data that refer to the spindle are combined in the module SSP. The individual variables are defined as arrays where the row index is the number of the spindle (assigned to the current channel). The spindle can be identified by reading the variables "name" or "index" in the same module with the respective row index.

The number of spindles can be read from "numSpindles" in the module Y in the area C.

Values of 0 or ' ' are supplied for axes which are not spindles. The value SSP:index = 0 indicates that the axis is not a spindle.

acConstCutS					
Current constant cutting rate					
m/min, ft/min, user defined	0			Double	r
Multi-line: yes	Axis index		maxnumGlobMachAxes		

acSMode					
Spindle mode 0: No spindle present in channel or spindle is active in another channel or is being used by PLC (FC18) or by synchronized actions. 1: Open-loop speed control mode 2: Positioning mode 3: Synchronous mode 4: Axis mode					
-	1	0	4	UWord	r
Multi-line: yes	Axis index		maxnumGlobMachAxes		

acSType					
Type of spindle programming Spindle programmed as: 0 Spindle not programmed 1 Spindle speed, S in rev/min 2 Cutting rate, SVC in m/min or ft/min 3 Constant cutting rate, S in m/min or ft/min 4 Constant grinding wheel peripheral speed, S in m/s or ft/s					
mm/min, inch/min, user defined	0	0		Double	r
Multi-line: no					

acSVC					
Programmed, active cutting rate					
mm/min, inch/min, user defined	0	0		Double	r
Multi-line: no					

acSmaxAcc					
Active acceleration of the spindle This variable returns the active acceleration of the spindle for spindle mode. Bit 14 of $AC_SPIND_STATE (spindle accelerating) is set for the duration of the acceleration to the defined setpoint speed. Bit 15 of $AC_SPIND_STATE (spindle braking) is set for the duration of the braking to the defined setpoint speed. Apart from that, the acceleration-determining machine and setting data can be determined with the system variable $AC_SMAXACC_INFO. If the spindle is in axis mode, then $AC_SMAXACC does not return the current acceleration, instead the machine data (MAX_AX_VELO, MAX_AX_ACCEL, ...) typical for axis mode are active.					
Rev/s2, user defined				Double	r
Multi-line: yes	Axis index		maxnumGlobMachAxes		

acSmaxAccInfo					
Identifier for the active spindle acceleration data The system variable provides additional information about $AC_SMAXACC and returns the definitive machine data as identifier/index. The index can be used to determine the active acceleration data on the basis of the following table of existing spindle accelerations. The number range is oriented to the system variable $AC_SMAXVELO_INFO: 0 No acceleration limitation (SERUPRO) 1 Not used 2 Acceleration in speed control mode without position control in the current gear stage MD 35200 GEAR_STEP_SPEEDCTRL_ACCEL 3 Not used 4 Acceleration in the current gear stage based on position control MD 35210 GEAR_STEP_POSCTRL_ACCEL (SPCON, SPOS, poss. with COUPON,..) 5 Not used 6 Not used 7 Not used 8 Not used 9 Acceleration limited by preparation calculations 10 Not used 11 Not used 12 Acceleration limited by axis mode. In the case of a synchronous spindle, the axis mode is enforced by the leading spindle. 13 Acceleration of the superimposed motion of the following spindle limited to the residual dynamics remaining following the coupling 14 Acceleration of the leading spindle due to missing following spindle dynamics or a high transformation ratio 15 Acceleration of the master spindle MD 35212 GEAR_STEP_POSCTRL_ACCEL2 in the case of tapping with G331, G332 (only when the second data set is configured accordingly) 16 Acceleration limited by the configuration of ACC or ACCFXS (synchronized action) 17 Acceleration limited by tool parameter $TC_TP_MAX_ACCEL 18 Not used 19 Acceleration limited in JOG mode by MD 32301 MA_JOG_MAX_ACCEL 20 Acceleration limited due to NCU link 21 Not used 22 Acceleration limited by programming ACCLIMA 23 Not used In oscillation mode (gear stage change), the variable returns the value for spindle mode (speed control mode).					
-				Long Integer	r
Multi-line: yes	Axis index		maxnumGlobMachAxes		

acSmaxVelo					
Maximum spindle speed This variable returns the maximum spindle speed for spindle mode. This is formed from the smallest active speed limitation, and cannot be exceeded by speed programming or override > 100%. A speed limitation is indicated by the VDI interface signal DB31..,DBX83.1 'Setpoint speed limited' and by $AC_SPIND_STATE, bit 10 (setpoint speed limited). The cause of the speed limitation (machine, setting data, G code, VDI interface signal etc.) can also be determined with the system variable $AC_SMAXVELO_INFO. If the spindle is in axis mode, then the speed is not limited by $AC_SMAXVELO but instead the machine data (MAX_AX_VELO, ...) typical for axis mode are active.					
rev/min, user defined				Double	r
Multi-line: yes	Axis index		maxnumGlobMachAxes		

acSmaxVeloInfo					
Identifier (index) for the speed-limiting data (machine/setting data, etc.) The system variable provides additional information about $AC_SMAXVELO and returns the definitive data (machine, setting data, G code, VDI interface etc.) as identifier/index. The index can be used to determine the speed-limiting data on the basis of the following table of existing spindle speed limitations. 0 No limitation (SERUPRO) 1 Maximum speed (chuck speed) of spindle MD 35100 SPIND_VELO_LIMIT 2 Speed limited to maximum speed in the current gear stage MD 35130 GEAR_STEP_MAX_VELO_LIMIT 3 Speed limited due to position control to 90% of the minimum from MD 35100 and MD 35130 (SPCON, SPOS, poss. with COUPON,..) 4 Speed limited due to position control to MD 35132 GEAR_STEP_PC_MAX_VELO_LIMIT 5 Speed limited to SD 43220 SPIND_MAX_VELO_G26 (G26 S.. or specification from HMI) 6 Speed limited to MD 35160 SPIND_EXTERN_VELO_LIMIT based on the set VDI interface signal DB31,...DBX3.6 7 Speed limited to SD 43230 SPIND_MAX_VELO_LIMS at constant cutting speed (G96, G961, G962, G97, LIMS) 8 Speed limited to safe speed (SG) by Safety Integrated 9 Speed limited by preparation calculations 10 Limitation by drive parameter SINAMICS p1082 to maximum speed of the drive 11 Speed limitation to MD 36300 ENC_FREQ_LIMIT with functions that require a functioning measuring system, e.g. position control and G95, G96, G97, G973, G33, G34, G35 for the master spindle. The limitation takes into account the encoder speed, the MS arrangement (direct/indirect), MS limiting frequency and the current parameter set 12 Speed limited by axis mode. In the case of a synchronous spindle, axis mode is enforced by the leading spindle. 13 Speed of the superimposed motion of the following spindle limited to the residual dynamics remaining following the coupling. A larger proportion of the superimposed motion can be achieved by reducing the speed of the leading spindle, e.g. by programming G26 S, VELOLIM for the leading spindle or VELOLIMA for the following spindle. The coupling factor must be taken into account. 14 Speed of the leading spindle limited due to missing following spindle dynamics or a high transformation ratio 15 Speed of the master spindle limited to MD 35550 DRILL_VELO_LIMIT in the case of tapping with G331, G332 16 Speed limitation due to the programming of VELOLIM 17 Speed limitation by tool parameter $TC_TP_MAX_VELO 18 Not used 19 Not used 20 Speed limited due to NCU link 21 Speed limited by SD43235 SD_SPIND_USER_VELO_LIMIT, user-controlled speed limitation, e.g. tensioning device, chuck speed 22 Speed limited by the programming of VELOLIMA 23 Speed limited by the clamping state of the tool. In the case of a Weiss spindle, the clamping state can be read from $VA_MOT_CLAMPING_STATE[axn]. In oscillation mode (gear stage change), the variable returns the value for spindle mode (speed control mode).					
-				Long Integer	r
Multi-line: yes	Axis index		maxnumGlobMachAxes		

acSminVelo					
Minimum spindle speed This variable returns the minimum spindle speed for speed control mode. This is formed from the highest active speed increase, and cannot be undershot by speed programming or override < 100%. A speed increase is indicated by the VDI interface signal DB31..,DBX83.2 'Setpoint speed increased' and by $AC_SPIND_STATE, bit 11 (setpoint speed increased). The cause of the speed increase (machine, setting data, G code, VDI interface signal etc.) can also be determined with the system variable $AC_SMINVELO_INFO. If the spindle is in axis or positioning mode, then the speed is not increased by $AC_SMINVELO.					
rev/min, user defined				Double	r
Multi-line: yes	Axis index		maxnumGlobMachAxes		

acSminVeloInfo					
Identifier (index) for the speed-limiting data (machine/setting data, etc.) The system variable provides additional information about $AC_SMAXVELO, and returns the definitive data (machine, setting data, G code, VDI interface, etc.) as identifier/index. The speed-limiting data can be determined with the index from the following table of existing spindle speed limitations. The system variable provides additional information about $AC_SMINVELO, and returns the speed increasing data (machine, setting data) as identifier/index. The speed-increasing data can be determined with the index from the following table of existing spindle speed increases. 0 Not used 1 Not used 2 Lower speed limit (minimum speed) of the current gear stage MD 35140 GEAR_STEP_MIN_VELO_LIMIT 3 Not used 4 Not used 5 Lower speed limit (minimum speed) from SD 43210 SPIND_MIN_VELO_G25 (G25 S.. or specification from HMI) In oscillation mode (gear stage change) and axis mode, the variable returns the values from spindle mode.					
-				Long Integer	r
Multi-line: yes	Axis index		maxnumGlobMachAxes		

acSpindState					
This variable returns the selected states of the spindle. For positioning and axis modes, the variable $AA_INPOS_STATE[Sn] can also be read. Bit 0: "Constant cutting speed active" (VDI interface signal DB31..,DBX84.0) Bit 1: "SUG active" (VDI interface signal DB31..,DBX84.1) Bit 2: "CLGON active" (VDI interface signal DB31..,DBX84.2) Bit 3: "Tapping without compensating chuck" (VDI interface signal DB31..,DBX84.3) Bit 4: "Synchronous mode" (following spindle with synchronous spindle coupling) (VDI interface signal DB31..,DBX84.4) Bit 5: "Positioning mode" (VDI interface signal DB31..,DBX84.5) Bit 6: "Oscillating mode" (gear stage change) (VDI interface signal DB31..,DBX84.6) Bit 7: "Speed control mode" (VDI interface signal DB31..,DBX84.7) Bit 8: "Spindle programmed" (e.g. M3, M4 S.., FC18, ..) (VDI interface signal DB31..,DBX64.4/5 or 6/7) Bit 9: "Speed limit exceeded" (VDI interface signal DB31..,DBX83.0) Bit 10: "Setpoint speed limited" (VDI interface signal DB31..,DBX83.1), active if the speed would be greater than the maximum speed as a result of programming or override ($AC_SMAXVELO) Bit 11: "Setpoint speed increased" (VDI interface signal DB31..,DBX83.2) active if the speed would be less than the minimum speed as a result of programming or override (system variable $AC_SMINVELO) Bit 12: "Spindle in setpoint range" (VDI interface signal DB31..,DBX83.5) Bit 13: "Actual direction of rotation right" (VDI interface signal DB31..,DBX83.7) Bit 14: "Spindle accelerating" remains active as long as the spindle is accelerating to the defined setpoint speed on the setpoint side. Bit 15: "Spindle braking" remains active as long as the spindle is braking to the defined setpoint speed or comes to a standstill on the setpoint side. Bit 16: "Spindle stopped" (VDI interface signal DB31..,DBX61.4) Bit 17: "Tool with dynamic limitation active" (VDI interface signal DB31..,DBX85.0) Bit 18: Reserved Bit 19: "Spindle in position" (VDI interface signal DB31..,DBX85.5) Bit 20: "Position control active" (VDI interface signal DB31..,DBX61.5) Bit 21: "Referenced/synchronized 1" (VDI interface signal DB31..,DBX60.4) Bit 22: "Referenced/synchronized 2" (VDI interface signal DB31..,DBX60.5) Bit 23: Direction of spindle rotation inverted by interface signal "Invert M3/M4" (DB31..,DBX17.6)					
-				Long Integer	r
Multi-line: yes	Axis index		maxnumGlobMachAxes		

actGearStage					
Actual gear stage of spindle					
-				UWord	r
Multi-line: yes	Axis index		maxnumGlobMachAxes		

actSpeed					
Spindle speed actual value					
rev/min, user defined				Double	r
Multi-line: yes	Axis index		maxnumGlobMachAxes		

channelNo					
Number of channel in which spindle is configured					
-				UWord	r
Multi-line: yes	Axis index		maxnumGlobMachAxes		

cmdAngPos					
Spindle position (SPOS)					
Degree, user defined				Double	r
Multi-line: yes	Axis index		maxnumGlobMachAxes		

cmdConstCutSpeed					
Constant cutting rate of the master spindle. The requested value for the master spindle differs from SSP:cmdSpeed only if G96 is active. (For a certain OEM customer this variable is now available retroactively in software version 3.2)					
mm/min, inch/min, user defined	0.0			Double	r
Multi-line: yes	Axis index		maxnumGlobMachAxes		

cmdGearStage					
Requested gear stage					
-				UWord	r
Multi-line: yes	Axis index		maxnumGlobMachAxes		

cmdGwps					
Programmed SUG desired value (SUG is the function "constant perimeter speed of grinding wheel")					
m/s, ft/s				Double	r
Multi-line: yes	Axis index		maxnumGlobMachAxes		

cmdSpeed					
Spindle speed desired value					
rev/min , m/min				Double	r
Multi-line: yes	Axis index		maxnumGlobMachAxes		

driveLoad					
Load					
%				Double	r
Multi-line: yes	Axis index		maxnumGlobMachAxes		

gwpsActive					
SUG programming active (SUG is the function "constant perimeter speed of grinding wheel) 0 = inactive 1 = active					
-				UWord	r
Multi-line: yes	Axis index		maxnumGlobMachAxes		

index					
Absolute axis index referred to MD					
-				UWord	r
Multi-line: yes	Axis index		maxnumGlobMachAxes		

name					
Spindle name Note: If several logical spindles are referred to one physical spindle with active spindle conversion and access is made via area N of module SSP2, then the name of the first suitable logical spindle is output.					
-				String [32]	r
Multi-line: yes	Axis index		maxnumGlobMachAxes		

namePhys					
Name of assigned physical spindle, identical to "name" variable.					
-				String [32]	r
Multi-line: yes	Axis index		maxnumGlobMachAxes		

opMode					
Spindle mode 0 = spindle mode 1 = oscillation mode (gear step changeover) 2 = positioning mode 3 = synchronous mode 4 = axis mode					
-				UWord	r
Multi-line: yes	Axis index		maxnumGlobMachAxes		

<table>
<tr><th>pSMode</th><th colspan="4"></th><th></th></tr>
<tr><td colspan="6">Last programmed spindle mode
0: No spindle configured in channel or spindle is active in another channel
or in use by the PLC (FC18) or by synchronized actions.
1: Speed control mode
2: Positioning mode
3: Synchronous mode
4: Axis mode</td></tr>
<tr><td>-</td><td></td><td></td><td></td><td>UWord</td><td>r</td></tr>
<tr><td>Multi-line: yes</td><td colspan="2">Axis index</td><td colspan="3">maxnumGlobMachAxes</td></tr>
</table>

<table>
<tr><th>pSModeS</th><th colspan="4"></th><th></th></tr>
<tr><td colspan="6">Last programmed spindle mode with block search
0: No spindle configured in channel or spindle is active in another channel
or in use by the PLC (FC18) or by synchronized actions.
1: Speed control mode
2: Positioning mode
3: Synchronous mode
4: Axis mode</td></tr>
<tr><td>-</td><td></td><td></td><td></td><td>UWord</td><td>r</td></tr>
<tr><td>Multi-line: yes</td><td colspan="2">Axis index</td><td colspan="3">maxnumGlobMachAxes</td></tr>
</table>

<table>
<tr><th>psModePos</th><th colspan="4"></th><th></th></tr>
<tr><td colspan="6">If the spindle is in positioning mode (pSMode = 2) or axis mode
(pSMode = 4), the value actToolEdgeCenterPosEns is returned, otherwise 0.</td></tr>
<tr><td>-</td><td>0</td><td></td><td></td><td>Double</td><td>r</td></tr>
<tr><td>Multi-line: yes</td><td colspan="2">Axis index</td><td colspan="3">maxnumGlobMachAxes</td></tr>
</table>

<table>
<tr><th>psModePosBKS</th><th colspan="4"></th><th></th></tr>
<tr><td colspan="6">If the spindle is in positioning mode (pSMode = 2) or axis mode
(pSMode = 4), the value actProgPosBKS is returned, otherwise 0.</td></tr>
<tr><td>-</td><td>0</td><td></td><td></td><td>Double</td><td>r</td></tr>
<tr><td>Multi-line: yes</td><td colspan="2">Axis index</td><td colspan="3">maxnumGlobMachAxes</td></tr>
</table>

<table>
<tr><th>psModePosS</th><th colspan="4"></th><th></th></tr>
<tr><td colspan="6">If the spindle is in positioning mode (pSMode = 2) or axis mode
(pSMode = 4), the value cmdToolEdgeCenterPosEnsS is returned, otherwise 0.</td></tr>
<tr><td>-</td><td>0</td><td></td><td></td><td>Double</td><td>r</td></tr>
<tr><td>Multi-line: yes</td><td colspan="2">Axis index</td><td colspan="3">maxnumGlobMachAxes</td></tr>
</table>

speedLimit					
Current speed limitation for spindle					
rev/min , m/min				Double	r
Multi-line: yes	Axis index		maxnumGlobMachAxes		

speedOvr					
Spindle override					
%				Double	r
Multi-line: yes	Axis index		maxnumGlobMachAxes		

spindleType					
Spindle type 0 = master spindle 1 = no master spindle					
-				UWord	r
Multi-line: yes	Axis index		maxnumGlobMachAxes		

status					
Spindle status Bit0 = following spindle Bit1 = leading spindle Bit2 = master spindle Bit3 = constant cutting rate (G96) active Bit0 = following spindle Bit1 = leading spindle					
-				UWord	r
Multi-line: yes	Axis index		maxnumGlobMachAxes		

turnState					
State of spindle rotation value range to be read via BTSS variable 0 = clockwise 1 = counter-clockwise 2 = stop value range to be read via $ variable 3 = clockwise 4 = counter-clockwise 5 = stop					
-				UWord	r
Multi-line: yes	Axis index		maxnumGlobMachAxes		

vcSGear					
Variable $VC_SGEAR[spino] determines the currently active spindle gear stage. $AC_SGEAR[spino] determines the defined gear stage in the main run. During search the actual gear stage may differ from the defined gear stage, because during search the gear stages are not changed. Using $VC_SGEAR[spino] and $AC_SGEAR[spino] it can be checked whether a gear stage change is to be performed after a search. The following values are possible: 1: 1st gear stage active .... 5: 5th gear stage active 1: 1st gear stage active .... 5: 5th gear stage active					
-	0	0	5	short Integer	r
Multi-line: no					

3.3.8 Area N, Block SSP2 : State data: Spindle

OEM-MMC: Linkitem /NckLogicalSpindle/...

All state data that refer to a spindle, if a spindle converter (logical spindles) is active

acConstCutS					
Current constant cutting rate					
m/min, ft/min, user defined	0			Double	r
Multi-line: yes	Axis index		maxnumGlobMachAxes		

acSMode					
Spindle mode 0: No spindle present in channel or spindle is active in another channel or is being used by PLC (FC18) or by synchronized actions. 1: Open-loop speed control mode 2: Positioning mode 3: Synchronous mode 4: Axis mode					
-	1	0	4	UWord	r
Multi-line: yes	Axis index		maxnumGlobMachAxes		

acSType					
Type of spindle programming Spindle programmed as: 0 Spindle not programmed 1 Spindle speed, S in rev/min 2 Cutting rate, SVC in m/min or ft/min 3 Constant cutting rate, S in m/min or ft/min 4 Constant grinding wheel peripheral speed, S in m/s or ft/s					
mm/min, inch/min, user defined	0	0		Double	r
Multi-line: no					

acSVC					
Programmed, active cutting rate					
mm/min, inch/min, user defined	0	0		Double	r
Multi-line: no					

acSmaxAcc					
Active acceleration of the spindle This variable returns the active acceleration of the spindle for spindle mode. Bit 14 of $AC_SPIND_STATE (spindle accelerating) is set for the duration of the acceleration to the defined setpoint speed. Bit 15 of $AC_SPIND_STATE (spindle braking) is set for the duration of the braking to the defined setpoint speed. Apart from that, the acceleration-determining machine and setting data can be determined with the system variable $AC_SMAXACC_INFO. If the spindle is in axis mode, then $AC_SMAXACC does not return the current acceleration, instead the machine data (MAX_AX_VELO, MAX_AX_ACCEL, ...) typical for axis mode are active.					
Rev/s2, user defined				Double	r
Multi-line: yes	Axis index		maxnumGlobMachAxes		

acSmaxAccInfo					
Identifier for the active spindle acceleration data The system variable provides additional information about $AC_SMAXACC and returns the definitive machine data as identifier/index. The index can be used to determine the active acceleration data on the basis of the following table of existing spindle accelerations. The number range is oriented to the system variable $AC_SMAXVELO_INFO: 0 No acceleration limitation (SERUPRO) 1 Not used 2 Acceleration in speed control mode without position control in the current gear stage MD 35200 GEAR_STEP_SPEEDCTRL_ACCEL 3 Not used 4 Acceleration in the current gear stage based on position control MD 35210 GEAR_STEP_POSCTRL_ACCEL (SPCON, SPOS, poss. with COUPON,..) 5 Not used 6 Not used 7 Not used 8 Not used 9 Acceleration limited by preparation calculations 10 Not used 11 Not used 12 Acceleration limited by axis mode. In the case of a synchronous spindle, the axis mode is enforced by the leading spindle. 13 Acceleration of the superimposed motion of the following spindle limited to the residual dynamics remaining following the coupling 14 Acceleration of the leading spindle due to missing following spindle dynamics or a high transformation ratio 15 Acceleration of the master spindle MD 35212 GEAR_STEP_POSCTRL_ACCEL2 in the case of tapping with G331, G332 (only when the second data set is configured accordingly) 16 Acceleration limited by the configuration of ACC or ACCFXS (synchronized action) 17 Acceleration limited by tool parameter $TC_TP_MAX_ACCEL 18 Not used 19 Acceleration limited in JOG mode by MD 32301 MA_JOG_MAX_ACCEL 20 Acceleration limited due to NCU link 21 Not used 22 Acceleration limited by programming ACCLIMA 23 Not used In oscillation mode (gear stage change), the variable returns the value for spindle mode (speed control mode).					
-				Long Integer	r
Multi-line: yes	Axis index		maxnumGlobMachAxes		

acSmaxVelo					
Maximum spindle speed This variable returns the maximum spindle speed for spindle mode. This is formed from the smallest active speed limitation, and cannot be exceeded by speed programming or override > 100%. A speed limitation is indicated by the VDI interface signal DB31..,DBX83.1 'Setpoint speed limited' and by $AC_SPIND_STATE, bit 10 (setpoint speed limited). The cause of the speed limitation (machine, setting data, G code, VDI interface signal etc.) can also be determined with the system variable $AC_SMAXVELO_INFO. If the spindle is in axis mode, then the speed is not limited by $AC_SMAXVELO but instead the machine data (MAX_AX_VELO, ...) typical for axis mode are active.					
rev/min, user defined				Double	r
Multi-line: yes	Axis index		maxnumGlobMachAxes		

acSmaxVeloInfo					
Identifier (index) for the speed-limiting data (machine/setting data, etc.) The system variable provides additional information about $AC_SMAXVELO and returns the definitive data (machine, setting data, G code, VDI interface etc.) as identifier/index. The index can be used to determine the speed-limiting data on the basis of the following table of existing spindle speed limitations. 0 No limitation (SERUPRO) 1 Maximum speed (chuck speed) of spindle MD 35100 SPIND_VELO_LIMIT 2 Speed limited to maximum speed in the current gear stage MD 35130 GEAR_STEP_MAX_VELO_LIMIT 3 Speed limited due to position control to 90% of the minimum from MD 35100 and MD 35130 (SPCON, SPOS, poss. with COUPON,..) 4 Speed limited due to position control to MD 35132 GEAR_STEP_PC_MAX_VELO_LIMIT 5 Speed limited to SD 43220 SPIND_MAX_VELO_G26 (G26 S.. or specification from HMI) 6 Speed limited to MD 35160 SPIND_EXTERN_VELO_LIMIT based on the set VDI interface signal DB31,...DBX3.6 7 Speed limited to SD 43230 SPIND_MAX_VELO_LIMS at constant cutting speed (G96, G961, G962, G97, LIMS) 8 Speed limited to safe speed (SG) by Safety Integrated 9 Speed limited by preparation calculations 10 Limitation by drive parameter SINAMICS p1082 to maximum speed of the drive 11 Speed limitation to MD 36300 ENC_FREQ_LIMIT with functions that require a functioning measuring system, e.g. position control and G95, G96, G97, G973, G33, G34, G35 for the master spindle. The limitation takes into account the encoder speed, the MS arrangement (direct/indirect), MS limiting frequency and the current parameter set 12 Speed limited by axis mode. In the case of a synchronous spindle, axis mode is enforced by the leading spindle. 13 Speed of the superimposed motion of the following spindle limited to the residual dynamics remaining following the coupling. A larger proportion of the superimposed motion can be achieved by reducing the speed of the leading spindle, e.g. by programming G26 S, VELOLIM for the leading spindle or VELOLIMA for the following spindle. The coupling factor must be taken into account. 14 Speed of the leading spindle limited due to missing following spindle dynamics or a high transformation ratio 15 Speed of the master spindle limited to MD 35550 DRILL_VELO_LIMIT in the case of tapping with G331, G332 16 Speed limitation due to the programming of VELOLIM 17 Speed limitation by tool parameter $TC_TP_MAX_VELO 18 Not used 19 Not used 20 Speed limited due to NCU link 21 Speed limited by SD43235 SD_SPIND_USER_VELO_LIMIT, user-controlled speed limitation, e.g. tensioning device, chuck speed 22 Speed limited by the programming of VELOLIMA 23 Speed limited by the clamping state of the tool. In the case of a Weiss spindle, the clamping state can be read from $VA_MOT_CLAMPING_STATE[axn]. In oscillation mode (gear stage change), the variable returns the value for spindle mode (speed control mode).					
-				Long Integer	r
Multi-line: yes	Axis index		maxnumGlobMachAxes		

acSminVelo					
Minimum spindle speed This variable returns the minimum spindle speed for speed control mode. This is formed from the highest active speed increase, and cannot be undershot by speed programming or override < 100%. A speed increase is indicated by the VDI interface signal DB31..,DBX83.2 'Setpoint speed increased' and by $AC_SPIND_STATE, bit 11 (setpoint speed increased). The cause of the speed increase (machine, setting data, G code, VDI interface signal etc.) can also be determined with the system variable $AC_SMINVELO_INFO. If the spindle is in axis or positioning mode, then the speed is not increased by $AC_SMINVELO.					
rev/min, user defined				Double	r
Multi-line: yes	Axis index		maxnumGlobMachAxes		

acSminVeloInfo					
Identifier (index) for the speed-limiting data (machine/setting data, etc.) The system variable provides additional information about $AC_SMAXVELO, and returns the definitive data (machine, setting data, G code, VDI interface, etc.) as identifier/index. The speed-limiting data can be determined with the index from the following table of existing spindle speed limitations. The system variable provides additional information about $AC_SMINVELO, and returns the speed increasing data (machine, setting data) as identifier/index. The speed-increasing data can be determined with the index from the following table of existing spindle speed increases. 0 Not used 1 Not used 2 Lower speed limit (minimum speed) of the current gear stage MD 35140 GEAR_STEP_MIN_VELO_LIMIT 3 Not used 4 Not used 5 Lower speed limit (minimum speed) from SD 43210 SPIND_MIN_VELO_G25 (G25 S.. or specification from HMI) In oscillation mode (gear stage change) and axis mode, the variable returns the values from spindle mode.					
-				Long Integer	r
Multi-line: yes	Axis index		maxnumGlobMachAxes		

acSpindState					
This variable returns the selected states of the spindle. For positioning and axis modes, the variable $AA_INPOS_STATE[Sn] can also be read. Bit 0: "Constant cutting speed active" (VDI interface signal DB31..,DBX84.0) Bit 1: "SUG active" (VDI interface signal DB31..,DBX84.1) Bit 2: "CLGON active" (VDI interface signal DB31..,DBX84.2) Bit 3: "Tapping without compensating chuck" (VDI interface signal DB31..,DBX84.3) Bit 4: "Synchronous mode" (following spindle with synchronous spindle coupling) (VDI interface signal DB31..,DBX84.4) Bit 5: "Positioning mode" (VDI interface signal DB31..,DBX84.5) Bit 6: "Oscillating mode" (gear stage change) (VDI interface signal DB31..,DBX84.6) Bit 7: "Speed control mode" (VDI interface signal DB31..,DBX84.7) Bit 8: "Spindle programmed" (e.g. M3, M4 S.., FC18, ..) (VDI interface signal DB31..,DBX64.4/5 or 6/7) Bit 9: "Speed limit exceeded" (VDI interface signal DB31..,DBX83.0) Bit 10: "Setpoint speed limited" (VDI interface signal DB31..,DBX83.1), active if the speed would be greater than the maximum speed as a result of programming or override ($AC_SMAXVELO) Bit 11: "Setpoint speed increased" (VDI interface signal DB31..,DBX83.2) active if the speed would be less than the minimum speed as a result of programming or override (system variable $AC_SMINVELO) Bit 12: "Spindle in setpoint range" (VDI interface signal DB31..,DBX83.5) Bit 13: "Actual direction of rotation right" (VDI interface signal DB31..,DBX83.7) Bit 14: "Spindle accelerating" remains active as long as the spindle is accelerating to the defined setpoint speed on the setpoint side. Bit 15: "Spindle braking" remains active as long as the spindle is braking to the defined setpoint speed or comes to a standstill on the setpoint side. Bit 16: "Spindle stopped" (VDI interface signal DB31..,DBX61.4) Bit 17: "Tool with dynamic limitation active" (VDI interface signal DB31..,DBX85.0) Bit 18: Reserved Bit 19: "Spindle in position" (VDI interface signal DB31..,DBX85.5) Bit 20: "Position control active" (VDI interface signal DB31..,DBX61.5) Bit 21: "Referenced/synchronized 1" (VDI interface signal DB31..,DBX60.4) Bit 22: "Referenced/synchronized 2" (VDI interface signal DB31..,DBX60.5) Bit 23: Direction of spindle rotation inverted by interface signal "Invert M3/M4" (DB31..,DBX17.6)					
-				Long Integer	r
Multi-line: yes	Axis index		maxnumGlobMachAxes		

actGearStage					
Actual gear stage of spindle					
-				UWord	r
Multi-line: yes	Axis index		maxnumGlobMachAxes		

actSpeed					
Spindle speed actual value					
rev/min, user defined				Double	r
Multi-line: yes	Axis index		maxnumGlobMachAxes		

channelNo					
Number of channel in which spindle is configured					
-				UWord	r
Multi-line: yes	Logical spindle index		maxnumGlobMachAxes		

cmdAngPos					
Spindle position (SPOS)					
Degree, user defined				Double	r
Multi-line: yes	Axis index		maxnumGlobMachAxes		

cmdConstCutSpeed					
Constant cutting rate of the master spindle. The requested value for the master spindle differs from SSP:cmdSpeed only if G96 is active. (For a certain OEM customer this variable is now available retroactively in software version 3.2)					
mm/min, inch/min, user defined	0.0			Double	r
Multi-line: yes	Axis index		maxnumGlobMachAxes		

cmdGearStage					
Requested gear stage					
-				UWord	r
Multi-line: yes	Axis index		maxnumGlobMachAxes		

cmdGwps					
Programmed SUG desired value (SUG is the function "constant perimeter speed of grinding wheel")					
m/s, ft/s				Double	r
Multi-line: yes	Axis index		maxnumGlobMachAxes		

cmdSpeed					
Spindle speed desired value					
rev/min , m/min				Double	r
Multi-line: yes	Axis index		maxnumGlobMachAxes		

driveLoad					
Load					
%				Double	r
Multi-line: yes	Axis index		maxnumGlobMachAxes		

gwpsActive					
SUG programming active (SUG is the function "constant perimeter speed of grinding wheel) 0 = inactive 1 = active					
-				UWord	r
Multi-line: yes	Axis index		maxnumGlobMachAxes		

index					
Absolute axis index referred to MD					
-				UWord	r
Multi-line: yes	Axis index		maxnumGlobMachAxes		

name					
Spindle name Note: If several logical spindles are referred to one physical spindle with active spindle conversion and access is made via area N of module SSP2, then the name of the first suitable logical spindle is output.					
-				String [32]	r
Multi-line: yes	Axis index		maxnumGlobMachAxes		

namePhys					
Name of assigned physical spindle.					
-				String [32]	r
Multi-line: yes	Axis index		maxnumGlobMachAxes		

opMode					
Spindle mode 0 = spindle mode 1 = oscillation mode (gear step changeover) 2 = positioning mode 3 = synchronous mode 4 = axis mode					
-				UWord	r
Multi-line: yes	Axis index		maxnumGlobMachAxes		

pSMode					
Last programmed spindle mode 0: No spindle configured in channel or spindle is active in another channel or in use by the PLC (FC18) or by synchronized actions. 1: Speed control mode 2: Positioning mode 3: Synchronous mode 4: Axis mode					
-				UWord	r
Multi-line: yes	Axis index		maxnumGlobMachAxes		

pSModeS					
Last programmed spindle mode with block search 0: No spindle configured in channel or spindle is active in another channel or in use by the PLC (FC18) or by synchronized actions. 1: Speed control mode 2: Positioning mode 3: Synchronous mode 4: Axis mode					
-				UWord	r
Multi-line: yes	Axis index		maxnumGlobMachAxes		

psModePos					
If the spindle is in positioning mode (pSMode = 2) or axis mode (pSMode = 4), the value actToolEdgeCenterPosEns is returned, otherwise 0.					
-	0			Double	r
Multi-line: yes	Axis index		maxnumGlobMachAxes		

psModePosBKS					
If the spindle is in positioning mode (pSMode = 2) or axis mode (pSMode = 4), the value actProgPosBKS is returned, otherwise 0.					
-	0			Double	r
Multi-line: yes	Axis index		maxnumGlobMachAxes		

psModePosS					
If the spindle is in positioning mode (pSMode = 2) or axis mode (pSMode = 4), the value cmdToolEdgeCenterPosEnsS is returned, otherwise 0.					
-	0			Double	r
Multi-line: yes	Axis index		maxnumGlobMachAxes		

speedLimit					
Current speed limitation for spindle					
rev/min , m/min				Double	r
Multi-line: yes	Axis index		maxnumGlobMachAxes		

speedOvr					
Spindle override					
%				Double	r
Multi-line: yes	Axis index		maxnumGlobMachAxes		

spindleType					
Spindle type 0 = master spindle 1 = no master spindle					
-				UWord	r
Multi-line: yes	Axis index		maxnumGlobMachAxes		

status					
Spindle status Bit0 = following spindle Bit1 = leading spindle Bit2 = master spindle Bit3 = constant cutting rate (G96) active Bit0 = following spindle Bit1 = leading spindle					
-				UWord	r
Multi-line: yes	Axis index		maxnumGlobMachAxes		

turnState					
State of spindle rotation value range to be read via BTSS variable 0 = clockwise 1 = counter-clockwise 2 = stop value range to be read via $ variable 3 = clockwise 4 = counter-clockwise 5 = stop					
-				UWord	r
Multi-line: yes	Axis index		maxnumGlobMachAxes		

vcSGear					
Variable $VC_SGEAR[spino] determines the currently active spindle gear stage. $AC_SGEAR[spino] determines the defined gear stage in the main run. During search the actual gear stage may differ from the defined gear stage, because during search the gear stages are not changed. Using $VC_SGEAR[spino] and $AC_SGEAR[spino] it can be checked whether a gear stage change is to be performed after a search. The following values are possible: 1: 1st gear stage active .... 5: 5th gear stage active 1: 1st gear stage active .... 5: 5th gear stage active					
-	0	0	5	short Integer	r
Multi-line: no					

3.3.9 Area N, Block FA : Active NCU global frames

OEM-MMC: Linkitem /NckActualFrame/...

There are the following frame indices:

2: IFRAME current settable work offset (only if $MN_MM_NUM_GLOBAL_USER_FRAMES > 0)

6: ACTBFRAME current total of base frames (only if $MN_MM_NUM_GLOBAL_BASE_FRAMES = 0)

The maximum frame index is: 6

linShift	diverse, siehe Bausteinbescheibung				PA
Translation of an active work offset (the physical unit is defined in basicLengthUnit in module Y in area N).					
mm, inch, user defined				Double	r
Multi-line: yes	Frame index * maxnumGlobMachAxes + axis number		20 * maxnumGlobMachAxes		

linShiftFine					
Fine offset for frames, extension of the basic frames and the settable frames.					
mm, inch, user defined				Double	rw
Multi-line: yes	Frame index * maxnumGlobMachAxes + axis number		6 * maxnumGlobMachAxes		

mirrorImgActive	diverse, siehe Bausteinbescheibung				PA
Mirroring enabled in an active work offset 0 = mirroring not active 1 = mirroring active					
-				UWord	r
Multi-line: yes	Frame index * maxnumGlobMachAxes + axis number		20 * maxnumGlobMachAxes		

rotation	diverse, siehe Bausteinbescheibung				PA
Rotation of an active work offset					
Degree				Double	r
Multi-line: yes	Frame index * maxnumGlobMachAxes + axis number		20 * maxnumGlobMachAxes		

rotationCoordinate	diverse, siehe Bausteinbescheibung				
Rotation around a coordinate of an active zero offset 1: Rotation around the first non-existing geometry axis.					
Degree				Double	r
Multi-line: yes	Frame index * maxnumGlobMachAxes + 1		20 * maxnumGlobMachAxes		

scaleFact	diverse, siehe Bausteinbescheibung				PA
Scaling factor of an active work offset					
-				Double	r
Multi-line: yes	Frame index * maxnumGlobMachAxes + axis number		6 * maxnumGlobMachAxes		

3.3.10 Area N, Block FB : NCU global base frames

OEM-MMC: Linkitem /NckBaseFrame/...

This only applies if $MN_MM_NUM_GLOBAL_BASE_FRAMES > 0.

The maximum frame index is: $MN_MM_NUM_GLOBAL_BASE_FRAMES - 1

linShift	$P_NCBFR[x,TR] x=FrameNo, y=Axis				PA
Translation of settable work offset (the physical unit is defined in basicLengthUnit in module Y in area N).					
mm, inch, user defined				Double	rw
Multi-line: yes	Frame index * maxnumGlobMachAxes + axis number		$MN_MM_NUM_GLOBAL_BASE_FRAMES * maxnumGlobMachAxes		

linShiftFine	$P_NCBFR[x,SI] x=FrameNo, y=Axis				
Fine offset with frames, expansion of basic frames and settable frames					
mm, inch, user defined				Double	rw
Multi-line: yes	Frame index * maxnumGlobMachAxes + axis number		$MN_MM_NUM_GLOBAL_BASE_FRAMES * maxnumGlobMachAxes		

mirrorImgActive	$P_NCBFR[x ,MI] x=FrameNo, y=Axis				PA
Mirroring enabled in a settable work offset 0: Mirroring not active 1: Mirroring active					
-				UWord	rw
Multi-line: yes	Frame index * maxnumGlobMachAxes + axis number		$MN_MM_NUM_GLOBAL_BASE_FRAMES * maxnumGlobMachAxes		

rotation	$P_NCBFR[x,y,RT] x=FrameNo, y=Axis				PA
Rotation of a settable work offset					
Degree				Double	rw
Multi-line: yes	Frame index * maxnumGlobMachAxes + axis number		$MN_MM_NUM_GLOBAL_BASE_FRAMES * maxnumGlobMachAxes		

rotationCoordinate	$P_NCBFR[x,y,RT] x=FrameNo, y=1				
Rotation around a coordinate of a settable zero offset 1: Rotation around the first non-existing geometry axis.					
Degree				Double	rw
Multi-line: yes	Frame index * maxnumGlobMachAxes + 1		$MN_MM_NUM_GLOBAL_BASE_FRAMES * maxnumGlobMachAxes		

scaleFact	$P_NCBFR[x,SC] x=FrameNo, y=Axis				PA
Scaling factor of a settable work offset					
-				Double	rw
Multi-line: yes	Frame index * maxnumGlobMachAxes + axis number		$MN_MM_NUM_GLOBAL_BASE_FRAMES * maxnumGlobMachAxes		

3.3.11 Area N, Block FU : NCU global settable frames

OEM-MMC: Linkitem /NckUserFrame/...

This only applies if $MN_MM_NUM_GLOBAL_USER_FRAMES > 0.

The following frame indices are possible:

0 = G500

1 = G54

2 = G55

3 = G56

4 = G57

5 = G505

6 = G506

:

n = G5n

:

99 = G599

The maximum frame index is: $MN_MM_NUM_GLOBAL_USER_FRAMES - 1

The PI service SETUFR has to be called in order to activate the settable frames.

linShift					PA
Translation of settable work offset (the physical unit is defined in basicLengthUnit in module Y in area N).					
mm, inch, user defined				Double	rw
Multi-line: yes	Frame index * maxnumGlobMachAxes + axis number		$MN_MM_NUM_GLOBAL_USER_FRAMES * maxnumGlobMachAxes		

linShiftFine					
Fine offset with frames, expansion of basic frames and settable frames					
mm, inch, user defined				Double	rw
Multi-line: yes	Frame index * maxnumGlobMachAxes + axis number		$MN_MM_NUM_GLOBAL_USER_FRAMES * maxnumGlobMachAxes		

mirrorImgActive					PA
Mirroring enabled in a settable work offset 0 = mirroring not active 1 = mirroring active					
-				UWord	rw
Multi-line: yes	Frame index * maxnumGlobMachAxes + axis number		$MN_MM_NUM_GLOBAL_USER_FRAMES * maxnumGlobMachAxes		

rotation					
Dummy variable, do not use					
-				Double	r
Multi-line: no					

rotationCoordinate					
Coordinate rotation of a settable frame					
-				Double	r
Multi-line: yes	Frame index * maxnumGlobMachAxes + 1		$MN_MM_NUM_GLOBAL_USER_FRAMES * maxnumGlobMachAxes		

scaleFact					PA
Scaling factor of a settable work offset					
-				Double	rw
Multi-line: yes	Frame index * maxnumGlobMachAxes + axis number		$MN_MM_NUM_GLOBAL_USER_FRAMES * maxnumGlobMachAxes		

3.3.12 Area N, Block YFAFL : NCK instruction groups (Fanuc)

OEM-MMC: Linkitem /NckFunctionGroupingFanuc/...

All G functions currently configured for the channels are made available for reading by the NCK. They are configured via machine data. Since the G functions are organized in groups, only one of which can be active at a time, this module is organized as a table.

There are two columns for each G group. The 1st column lists the number of G functions in a group (/N/YFAFL/Gruppe_NUM), this corresponds to the number of rows in each subsequent column. This second column contains all the G functions belonging to a group (/N/YFAFL/Gruppe).

As a result, the data for a certain G group are calculated via a column offset.

The column offset of each variable is:

2 * (G group number - 1)

The number of G groups is given in the variable "numGCodeGroupsFanuc" in area N / module Y. The resultant maximum column offset of the variables is thus 2 * numGCodeGroupsFanuc.

The G functions currently active are listed in area C / module SNCF.

Gruppe					
Instruction group					
-				String [16]	r
Multi-line: yes	Serial number		Gruppe_NUM		

Gruppe_NUM					
Number of Fanuc-G functions in the relevant group					
-		0		UWord	r
Multi-line: yes	1		1		

3.3.13 Area B, Block S : Mode-group-specific state data

OEM-MMC: Linkitem /BagState/...

During NC operation different internal states occur and system-specific data may change during operation. To distinguish those from system variables, they are classified as state data.

A distinction is made between:

- NCK-specific state data

- Mode-group-specific state data

- Channel-specific state data

- Drive-specific state data (FDD)

- Drive-specific state data (MSD)

autoJogState	$AC_AUTO_JOG_STATE				
Status of Automatic+JOG mode 1: Automatic is selected, $MN_JOG_MODE_MASK is set, and the mode group (BAG) is in "Reset". A JOG motion can then be initiated by pressing the +/- buttons or turning the hand wheel in Auto. 2: This mode group has been switched internally to JOG on account of a JOG motion. VDI and OPI still show Automatic. 0: In all other cases					
-	0	0	2	UWord	r
Multi-line: yes	Mode group number		numBAGs		

ncAutoCounter					
Counter which is incremented with each 0->edge of the Auto key					
-	0	0		UWord	r
Multi-line: yes	Mode group number		numBAGs		

ncJogCounter					
Counter which is incremented with each 0->edge of the Jog key					
-	0	0		UWord	r
Multi-line: yes	Mode group number		numBAGs		

ncMDACounter					
Counter which is incremented with each 0->edge of the MDI key					
-	0	0		UWord	r
Multi-line: yes	Mode group number		numBAGs		

opMode	DB11, DBX6.0-6.2				
Active mode 0 = JOG 1 = MDI 2 = AUTO					
-				UWord	r
Multi-line: no					

readyActive	DB11, DBX6.3				
Code whether mode group is ready 0 = not ready 1 = ready					
-				UWord	r
Multi-line: no					

resetActive	DB11, DBX6.7				
Code whether all channels in mode group are in Reset 0 = not all channels in reset 1 = all channels in reset					
-				UWord	r
Multi-line: no					

3.3.14 Area N, Block SALAC : Alarm actions: List in rev. chronol. order, oldest alarm act. appears first

OEM-MMC: Linkitem /NckAlarmEvent/...

In a given alarm, all values in the SALAC module are identical to the corresponding variables in the SALA, SALAP and SALAL modules with the exception of actionType and actionCount.

The same alarm can be found in the various modules by comparing the values of alarmNo.

A client is registered with an alarm server when cyclic reading of the SALAC module has been set.

If the operator panel sets cyclic reading when a data in the module changes and column index 0 has been specified, then the variable server sends the entire data block to the operator panel if the alarm server receives a new alarm action.

Another alarm server client is registered at each cyclic reading of the SALAC module which has been set.

This mechanism therefore functions with more than one user interface connected. Registration is withdrawn when the corresponding cyclic reading is terminated. As cyclic reading only applies to registered clients,

each regular and noncyclic reading returns the default value for the requested variable.

actionCount					
A unique number assigned to Alarm Action. On power ON it is reset to zero by the NCK. For each new alarm action it will be incremented by one.					
-	0			Long Integer	r
Multi-line: yes	1		1		

actionType					
Specifies whether the alarm is deleted or activated. 0: No alarm action pending 1: Alarm set 2: Alarm deleted					
-	0	0	2	Long Integer	r
Multi-line: yes	1		1		

alarmNo					
A unique number assigned to the alarm. It will be incremented by one for each reported alarm.					
-	0			Long Integer	r
Multi-line: yes	1		1		

clearInfo					
Describes the acknowledgement criterion for the alarm. 1 = Power ON 2 = Reset 3 = Delete 4 = Alarm is deleted by NCK software 5 = Alarm is deleted by program call 6 = Alarm is deleted by RESET in all channels of the mode groups (from SW 4.1) 7 = Alarm is deleted by RESET in all channels of the NC (from SW 4.1)					
-	1	1	7	Long Integer	r
Multi-line: yes	1		1		

fillText1					
Parameter 1, ASCII string which is inserted in the standard alarm text to supplement the alarm description.					
-	0			String [32]	r
Multi-line: yes	1		1		

fillText2					
Parameter 2, ASCII string which is inserted in the standard alarm text to supplement the alarm description.					
-	0			String [32]	r
Multi-line: yes	1		1		

fillText3					
Parameter 3, ASCII string which is inserted in the standard alarm text to supplement the alarm description.					
-	0			String [32]	r
Multi-line: yes	1		1		

fillText4					
Parameter 4, ASCII string which is inserted in the standard alarm text to supplement the alarm description.					
-	0			String [32]	r
Multi-line: yes	1		1		

textIndex					
Identifies the alarm for alarm description.					
-	0			Long Integer	r
Multi-line: yes	1		1		

timeBCD					
Date and time of the occurred alarm in BCD format.					
-				Date+Time	r
Multi-line: yes	1		1		

3.4 Status data of the channel

3.4.1 Area C, Block M : Channel-specific machine data

OEM-MMC: Linkitem /ChannelDrive/...

Channel-specific machine data

AXCONF_CHANAX_NAME_TAB	MD 20080: $MC_AXCONF_CHANAX_NAME_TAB				
MD 20080: $MC_AXCONF_CHANAX_NAME_TAB					
-				String [16]	r
Multi-line: no			2		

3.4.2 Area C, Block S : Channel-specific status data

OEM-MMC: Linkitem /ChannelState/...

During NC operation different internal states occur and system-specific data may change during operation. To distinguish those from system variables, they are classified as state data.

A distinction is made between:

- NCK-specific state data

- Mode-group-specific state data

- Channel-specific state data

- Drive-specific state data (FDD)

- Drive-specific state data (MSD)

G0Mode	$AC_G0MODE				
G00 is active and $MC_G0_LINEAR_MODE is FALSE (Siemens mode) or $MC_EXTERN_G0_LINEAR_MODE is FALSE (ISO mode) and therefore non-linear interpolation is active with G0, i.e. the path axes are traversed as positioning axes. 0: G00 not active 1: G00 and linear interpolation active 2: G00 and non-linear interpolation active					
-	0	0	2	UWord	r
Multi-line: yes	1		1		

aGG	$A_GG				
active G function in synchronized action					
-	0	0		UWord	r
Multi-line: yes	Number of the G function group		Gruppe_NUM		

aLinkTransRate	$A_LINK_TRANS_RATE				
Link transfer rate Number of link variables that can still be transferred in the current IPO cycle via the NCU link communication. If this variable is read in the context of the lead, it always outputs the maximum available bandwidth.					
-		0		UWord	r
Multi-line: yes	1		1		

aMonifact	$A_MONIFACT				
Factor for tool life monitoring					
-	0	0		Double	r
Multi-line: yes	1		1		

aTcAckC	$AC_TC_ACKC				
Counter variable: aTcAckC (AcknowledgeCounter) is incremented by 1 every time the PLC acknowledges a tool management command.					
-	0	0		UWord	rw
Multi-line: yes	1		1		

aTcCmdC	$AC_TC_CMDC				
Counter variable: aTcCmdC (CoMmandCounter) is incremented by 1 every time the tool management outputs a command to the PLC.					
-	0	0		UWord	rw
Multi-line: yes	1		1		

aTcDistance	$AC_TC_DISTANCE				
Distance of the multi-tool place of the loaded tool to the reference point					
-				Double	r
Multi-line: yes	1		1		

aTcFct	$AC_TC_FCT				
Command number					
-				UWord	r
Multi-line: yes	1		1		

aTcLfn	$AC_TC_LFN				
Source location number of new tool					
-				UWord	r
Multi-line: yes	1		1		

aTcLfo	$AC_TC_LFO				
Source location number of old tool					
-				UWord	r
Multi-line: yes	1		1		

aTcLmyn	$AC_TC_LMYN				
Owner location number of the new tool					
-		-1	32000	UWord	r
Multi-line: yes	1		1		

aTcLtn	$AC_TC_LTN				
Target location number of new tool					
-				UWord	r
Multi-line: yes	1		1		

aTcLto	$AC_TC_LTO				
Target location number of old tool					
-				UWord	r
Multi-line: yes	1		1		

aTcMfn	$AC_TC_MFN				
Source magazine of new tool					
-				UWord	r
Multi-line: yes	1		1		

aTcMfo	$AC_TC_MFO				
Source magazine number of old tool					
-				UWord	r
Multi-line: yes	1		1		

aTcMmyn	$AC_TC_MMYN				
Owner magazine number of the new tool					
-		-1	32000	UWord	r
Multi-line: yes	1		1		

aTcMtn	$AC_TC_MTN				
Target magazine number of new tool					
-				UWord	r
Multi-line: yes	1		1		

aTcMto	$AC_TC_MTO				
Target magazine number of old tool					
-				UWord	r
Multi-line: yes	1		1		

aTcMtptn	$AC_TC_MTLTN				
Number of the multi-tool place of the loaded tool					
-				UWord	r
Multi-line: yes	1		1		

aTcMttn	$AC_TC_MTTN				
Number of the multi-tool of the loaded tool					
-				UWord	r
Multi-line: yes	1		1		

aTcNumPlaces	$AC_TC_MTNLOC				
Number of defined places in the multi-tool					
-				UWord	r
Multi-line: yes	1		1		

aTcStatus	$AC_TC_STATUS				
Command status					
-				UWord	r
Multi-line: yes	1		1		

aTcThno	$AC_TC_THNO				
Number of toolholder for new tool					
-				UWord	r
Multi-line: yes	1		1		

aTcTno	$AC_TC_TNO				
T number of new tool					
-				UWord	r
Multi-line: yes	1		1		

aTcToolIs	$AC_TC_TOOLIS				
0 = Tool, 1,2,3 = Type of distance coding for the multi-tool					
-				UWord	r
Multi-line: yes	1		1		

aaATol	$AA_ATOL				
aaATol provides the axis tolerance for compressor and smoothing, which was effective during preprocessing of the current main run block.					
mm, inch, degree, user defined	0	0		Double	r
Multi-line: yes	Axis index		numMachAxes		

aaAccLim					
Display of the axial acceleration override programmed by ACC. Depending on $MA_DYN_LIMIT_RESET_MASK, the value can remain active after Reset. The variable always shows the programmed acceleration override and not the currently active acceleration limit.					
-	100	1	200	UWord	r
Multi-line: yes	(Axis index)		numMachAxes		

aaAccLimA	$AA_ACCLIMA[a]				
Axial acceleration override in main run 1-200					
-	100	1	200	UWord	r
Multi-line: yes	(Axis index)		numMachAxes		

aaEgActive	$AA_EG_ACTIVE[a,b]				
Electronic gear: Link to the specified master axis is operative, i.e. activated. 0: Deactivated 1: Activated					
-	0	0	1	UWord	r
Multi-line: yes	(Axis index of the slave axis) * numMachAxes + (axis index of the master axis) + 1		numMachAxes * numMachAxes		

aaEgAx	$AA_EG_AX[n,a]				
Electronic gear: Axis number of nth master axis (1-n). (Axis index = axis number - 1) 1-numMachAxes					
-	0	1	numMachAxes	UWord	r
Multi-line: yes	(Axis index of slave axis) * 5 + (index of master axis) + 1		numMachAxes * 5		

aaEgDenom	$AA_EG_DENOM[a,b]				
Electronic gear: Denominator of link factor for the specified master axis. The link factor of the gear is the result of $AA_EG_NUMERA[a,b]/$AA_EG_DENOM[a,b].					
-	1			Double	r
Multi-line: yes	(Axis index of the slave axis) * numMachAxes + (axis index of the master axis) + 1		numMachAxes * numMachAxes		

aaEgNumLa	$AA_EG_NUM_LA[a]				
Electronic gear: Number of master axes specified with EGDEF. If the axis has not been specified with EGDEF as slave axis, the value is 0. 0-5					
-	0	0	5	UWord	r
Multi-line: yes	(Axis index of slave axis + 1)		numMachAxes		

aaEgNumera	$AA_EG_NUMERA[a,b]				
Electronic gear: Numerator of link factor for the specified master axis. The link factor of the gear is the result of $AA_EG_NUMERA[a,b]/$AA_EG_DENOM[a,b].					
-	0			Double	r
Multi-line: yes	(Axis index of the slave axis) * numMachAxes + (axis index of the master axis) + 1		numMachAxes * numMachAxes		

aaEgSyn	$AA_EG_SYN[a,b]				
Electronic gear: Synchronous position for the specified master axis.					
mm, inch, degree, user defined	0			Double	r
Multi-line: yes	(Axis index of the slave axis) * numMachAxes + (axis index of the master axis) + 1		numMachAxes * numMachAxes		

aaEgSynFa	$AA_EG_SYNFA[a]				
Electronic gear: Synchronous position for the slave axis.					
mm, inch, degree, user defined	0			Double	r
Multi-line: yes	(Axis index of slave axis + 1)		numMachAxes		

aaEgType	$AA_EG_TYPE[a,b]				
Electronic gear: Type of link for the specified master axis 0: Actual-value linkage 1: Setpoint linkage					
-	0	0	1	UWord	r
Multi-line: yes	(Axis index of the slave axis) * numMachAxes + (axis index of the master axis) + 1		numMachAxes * numMachAxes		

aaFgref	$AA_FGREF				
The variable provides the radius used by a rotary axis to contribute to the path travel. The default value is 180 mm/PI = 57,296 mm, which corresponds to a contribution of 1mm per degree. For linear axes the variable always provides 1.					
mm, inch, user defined	0	0		Double	r
Multi-line: yes	Axis index		numMachAxes		

aaFgroup	$AA_FGROUP				
If the travel of an axis influences the path velocity in the current main run block (FGROUP), the variable provides value 1, otherwise 0.					
-	0	0	1	UWord	r
Multi-line: yes	Axis index		numMachAxes		

aaJerkLim					
Display of the axial jerk override programmed by JERKLIM. Depending on $MA_DYN_LIMIT_RESET_MASK, the value can remain active after Reset. The variable always shows the programmed jerk override and not the currently active jerk limit.					
-	100	1	200	UWord	r
Multi-line: yes	(Axis index)		numMachAxes		

aaJerkLimA	$AA_JERKLIMA[a]				
Axial jerk override in run in 1-200					
-	100	1	200	UWord	r
Multi-line: yes	(Axis index)		numMachAxes		

aaMeasP1Valid	$AA_MEAS_P1_VALID				
Save axial measuring point P1 for workpiece and tool measurement 0: Clear axial measuring point 1: Write actual axial values to axial measuring point					
-	0	0	1	Long Integer	rw
Multi-line: yes	Axis index		numMachAxes		

aaMeasP2Valid	$AA_MEAS_P2_VALID				
Save axial measuring point P2 for workpiece and tool measurement 0: Clear axial measuring point 1: Write actual axial values to axial measuring point					
-	0	0	1	Long Integer	rw
Multi-line: yes	Axis index		numMachAxes		

aaMeasP3Valid	$AA_MEAS_P3_VALID				
Save axial measuring point P3 for workpiece and tool measurement 0: Clear axial measuring point 1: Write actual axial values to axial measuring point					
-	0	0	1	Long Integer	rw
Multi-line: yes	Axis index		numMachAxes		

aaMeasP4Valid	$AA_MEAS_P4_VALID				
Save axial measuring point P4 for workpiece and tool measurement 0: Clear axial measuring point 1: Write actual axial values to axial measuring point					
-	0	0	1	Long Integer	rw
Multi-line: yes	Axis index		numMachAxes		

aaMeasPoint1	$AA_MEAS_POINT1				
1st measuring point for workpiece and tool measurement					
mm, inch, user defined	0			Double	rw
Multi-line: yes	Axis index		numMachAxes		

aaMeasPoint2	$AA_MEAS_POINT2				
2nd measuring point for workpiece and tool measurement					
mm, inch, user defined				Double	rw
Multi-line: yes	Axis index		numMachAxes		

aaMeasPoint3	$AA_MEAS_POINT3				
3rd measuring point for workpiece and tool measurement					
mm, inch, user defined				Double	rw
Multi-line: yes	Axis index		numMachAxes		

aaMeasPoint4	$AA_MEAS_POINT4				
4th measuring point for workpiece and tool measurement					
mm, inch, user defined				Double	rw
Multi-line: yes	Axis index		numMachAxes		

aaMeasSetangle	$AA_MEAS_SETANGLE				
Setpoint angle of an axis					
Degree, user defined				Double	rw
Multi-line: yes	Axis index				

aaMeasSetpoint	$AA_MEAS_SETPOINT				
Setpoint position of edge, corner or hole					
mm, inch, user defined				Double	rw
Multi-line: yes	Axis index		numMachAxes		

aaMeasSpValid	$AA_MEAS_SP_VALID				
Save axial setpoint for workpiece and tool measurement 0: Clear axial setpoint 1: Validate axial setpoint					
-	0	0	1	Long Integer	rw
Multi-line: yes	Axis index		numMachAxes		

aaSyncDiff	$AA_SYNCDIFF[]				
Setpoint synchronism difference for all types of coupling					
mm, inch, degree, user defined	0			Double	r
Multi-line: yes	Axis index of the following axis		numMachAxes		

aaSyncDiffStat	$AA_SYNCDIFF_STAT[]				
Status of setpoint synchronism difference -4: No valid value in $AA_SYNCDIFF, coupled motion from part program -3: Reserved -2: Reserved -1: No valid value in $AA_SYNCDIFF 0: No valid value in $AA_SYNCDIFF, coupling not active 1: Valid value in $AA_SYNCDIFF					
-	0	-4	1	Long Integer	r
Multi-line: yes	Axis index of the following axis		numMachAxes		

aaVeloLim					
Display of the axial velocity override programmed by VELOLIM. Depending on $MA_DYN_LIMIT_RESET_MASK, the value can remain active after Reset. The variable always shows the programmed velocity override and not the currently active velocity limit.					
-	100	1	200	UWord	r
Multi-line: yes	(Axis index)		numMachAxes		

aaVeloLimA	$AA_VELOLIMA[a]				
Axial velocity override in main run 1-200					
-	100	1	200	UWord	r
Multi-line: yes	(Axis index)		numMachAxes		

acActToolLengthIndex	$AC_ACT_TOOL_LENGTH_INDEX				
The variable returns the number of the tool length components (1, 2 or 3 corresponding to the length components L1, L2, L3) of the active tool, which is assigned to the geometry axis that was transferred as an index. The assignment does not take into account any rotations (e.g. by kinematic transformations) or frames. It depends on the type of active tool, the active plane, any active adapter transformation and the setting data SD42950 $SC_TOOL_LENGTH_TYPE, SD42940 $SC_TOOL_LENGTH_CONST and SD42942 $SC_TOOL_LENGTH_CONST_T. Active mirrorings of a frame can influence the output value if setting data SD42900 $SC_MIRROR_TOOL_LENGTH, is set, see below. If the tool length component is active with a negative sign, the index is output with a negative sign. This can be the case if the hundreds digit of setting data SD42940 $SC_TOOL_LENGTH_CONST or the hundreds digit of setting data SD42942 $SC_TOOL_LENGTH_CONST_T equals 1, or if a mirroring of the relevant axis is active on account of setting data $SC_MIRROR_TOOL_LENGTH. If both causes are active simultaneously, the resulting sign is positive once more. If no tool is active, the value 0 is returned.					
-	0	-3	3	UWord	r
Multi-line: yes	1		3		

acActToolLengthIndexS	$P_ACT_TOOL_LENGTH_INDEX				
The variable returns the number of the tool length components (1, 2 or 3 corresponding to the length components L1, L2, L3) of the active tool, which is assigned to the geometry axis that was transferred as an index. The assignment does not take into account any rotations (e.g. by kinematic transformations) or frames. It depends on the type of active tool, the active plane, any active adapter transformation and the setting data SD42950 $SC_TOOL_LENGTH_TYPE, SD42940 $SC_TOOL_LENGTH_CONST and SD42942 $SC_TOOL_LENGTH_CONST_T. Active mirrorings of a frame can influence the output value if setting data SD42900 $SC_MIRROR_TOOL_LENGTH, is set, see below. If the tool length component is active with a negative sign, the index is output with a negative sign. This can be the case if the hundreds digit of setting data SD42940 $SC_TOOL_LENGTH_CONST or the hundreds digit of setting data SD42942 $SC_TOOL_LENGTH_CONST_T equals 1, or if a mirroring of the relevant axis is active on account of setting data $SC_MIRROR_TOOL_LENGTH. If both causes are active simultaneously, the resulting sign is positive once more. If no tool is active, the value 0 is returned.					
-	0	-3	3	UWord	r
Multi-line: yes	1		3		

acAlarmStat	$AC_ALARM_STAT				
!=0: Alarms are pending, the appropriate coded alarm reactions can be used as source for "Extended stop and retract". The data is bit-coded. Individual states can therefore be masked or evaluated separately if necessary (bits excluded below produce a value of 0) Bit2 = 1: NOREADY (active rapid deceleration + cancellation of servo enable) Bit6 = 1: STOPBYALARM (ramp stop of all channel axes) Bit9 = 1: SETVDI (VDI interface signal alarm setting) Bit13 = 1: FOLLOWUPBYALARM (follow-up)					
-	0			UWord	r
Multi-line: yes	1		1		

acAsup	$AC_ASUP				
Code number for the cause of activation of an ASUB The reasons are bit-coded. BIT0: Activation because of: User interrupt "ASUB with Blsync". BIT1: Activation because of: User interrupt "ASUB". BIT2: Activation because of: User interrupt "ASUB from channel state Ready". BIT3: Activation because of: User interrupt "ASUB in manual mode". BIT4: Activation because of: Activation because of: User interrupt "ASUB". BIT5: Activation because of: Abort of the subroutine repetition. BIT6: Activation because of: Activation of decoding single block. BIT7: Activation because of: Activation of DDTG. BIT8: Activation because of: Activation of axis synchronization. BIT9: Activation because of: Change of operating mode. BIT10: Activation because of: Program continuation under TeachIn or after TeachIn deactivation. BIT11: Activation because of: Selection overstore. BIT12: Activation because of: Alarm with reaction compensation block with REPOS (COMPBLOCKWITHREORG). BIT13: Activation because of: Retraction motion with G33 and Stop. BIT14: Activation because of: Activation of dry run feed. BIT15: Activation because of: Deactivation of dry run feed. BIT16: Activation because of: Activation of skip block. BIT17: Activation because of: Deactivation of skip block. BIT18: Activation because of: Activate machine data. BIT19: Activation because of: Activate tool offset. BIT20: Activation because of: System ASUB after search type SERUPRO has reached the search target.					
-	0	0		Long Integer	r
Multi-line: yes	1		1		

acAxCtSwA	$AC_AXCTSWA[CTn]				
Channel status of axis container rotation TRUE: The channel has enabled rotation for the axis container and rotation is still in progress. FALSE: Axis container rotation is already finished					
-	0	0	1	UWord	r
Multi-line: yes	Container no.		numContainer		

acCTol	$AC_CTOL				
acCTol provides the contour tolerance for compressor and smoothing which was used to preprocess the current main run block.					
mm, inch, user defined	0	0		Double	r
Multi-line: yes	1		1		

acConeAngle	$AC_CONE_ANGLE				
Currently effective taper angle for taper turning. The taper angle is specified by the setting data $SC_CONE_ANGLE and is effective in the operating mode JOG only.					
Degree	0	-90	90	Double	r
Multi-line: yes	1		1		

acDelt	$AC_DELT				
Stored distance-to-go of the path in the WCS after delete-distance-to-go of the path DELDTG for synchronous action (Note: for SYNACT only)					
-				Double	r
Multi-line: yes	1		1		

acDtbb	$AC_DTBB				
Distance from the beginning of the block in the BCS (Note: SYNACT only)					
-				Double	r
Multi-line: yes	1		1		

acDtbw	$AC_DTBW				
Distance from the beginning of the block in the WCS (Note: for SYNACT only)					
-				Double	r
Multi-line: yes	1		1		

acDteb	$AC_DTEB				
Distance to the end of the block in the BCS (Note: for SYNACT only)					
-				Double	r
Multi-line: yes	1		1		

acDtew	$AC_DTEW				
Distance to the end of the block in the WCS (Note: for SYNACT only)					
-				Double	r
Multi-line: yes	1		1		

acEsrTrigger	$AC_ESR_TRIGGER				
Activation of "NC-controlled ESR"					
-	0	0	1	UWord	r
Multi-line: yes	1		1		

acFGo	$AC_F_G0				
Max. rapid traverse rate in the block					
mm/min, inch/min, user defined	0	0		Double	r
Multi-line: yes	1		1		

acFZ	$AC_FZ				
Tooth feedrate, setpoint. The physical unit is defined in variable 'feedRateIpoUnit'.					
mm/min, inch/min, user defined	0	0		Double	r
Multi-line: no					

acFct0	$AC_FCT0[x] x = PolynomNo				
a0-coefficient of the nth polynominal for the synchronous action SYNFCT / function FCTDEF n (Note: for SYNACT only)					
-				Double	r
Multi-line: yes	Number of the polynominal		$MC_MM_NUM_FCTDEF_ELEMENTS		

acFct1	$AC_FCT1[x] x = PolynomNo				
a1-coefficient of the nth polynominal for the synchronous action SYNFCT / function FCTDEF n (Note: for SYNACT only)					
-				Double	r
Multi-line: yes	Number of the polynominal		$MC_MM_NUM_FCTDEF_ELEMENTS		

acFct2	$AC_FCT2[x] x = PolynomNo				
a2-coefficient of the nth polynominal for the synchronous action SYNFCT / function FCTDEF n (Note: for SYNACT only)					
-				Double	r
Multi-line: yes	Number of the polynominal		$MC_MM_NUM_FCTDEF_ELEMENTS		

acFct3	$AC_FCT3[x] x = PolynomNo				
a3-coefficient of the nth polynominal for the synchronous action SYNFCT / function FCTDEF n (Note: for SYNACT only)					
-				Double	r
Multi-line: yes	Number of the polynominal		$MC_MM_NUM_FCTDEF_ELEMENTS		

acFctll	$AC_FCTLL[x] x = PolynomNo				
Lower limit of the nth polynominal for the synchronous action SYNFCT / function FCTDEF n (Note: for SYNACT only)					
-				Double	r
Multi-line: yes	Number of the polynominal		$MC_MM_NUM_FCTDEF_ELEMENTS		

acFctul	$AC_FCTUL[x] x = PolynomNo				
Upper limit of the nth polynominal for the synchronous action SYNFCT / function FCTDEF n (Note: for SYNACT only)					
-				Double	r
Multi-line: yes	Number of the polynominal		$MC_MM_NUM_FCTDEF_ELEMENTS		

acFgroupMask	$AC_FGROUP_MASK				
acFgroupMask bit-coded provides the channel axes which are to contribute to the path velocity					
-	0	0	0xFFFF	Long Integer	r
Multi-line: yes	1		1		

acInKeyG					
Grinding: returns the current value of the relevant grinding input.					
-	0	0	1	UWord	r
Multi-line: no			8		

acInKeyGEnable					
Grinding: indicates whether the relevant grinding input has been activated.					
-	0	0	1	UWord	r
Multi-line: no			8		

acInKeyGIsEnable					
Grinding: indicates whether the relevant grinding input is active.					
-	0	0	1	UWord	r
Multi-line: no			8		

acInKeyGRunIn					
Grinding: returns the current value of the relevant grinding input (PLC)					
-	0	0	1	UWord	r
Multi-line: no			8		

acInKeyGRunOut					
Grinding: returns the current value of the relevant grinding input (NCK)					
-	0	0	1	UWord	r
Multi-line: no			8		

acIpoState	$AC_IPO_STATE				
The variable provides selected information about whether certain functions are active: Bit 0: Free-form surface mode is active Bit 1: Compressor active					
-	0	0	0x0003	UWord	r
Multi-line: yes	1		1		

acIwStat	$AC_IW_STAT				
Current position of machine Bit-coded: Bit 0: Tool inv. position Bit 1: Axis 2/3 position Bit 2: Axis 5 position Bit 3-31: Not yet assigned					
-	0			Long Integer	r
Multi-line: yes	1		1		

acIwTu	$AC_IW_TU				
Current position of channel axes Bit-coded: Bit 0: Channel axis 1 position Bit 1: Channel axis 2 position Bit 2: Channel axis 3 position Bit 3: Channel axis 4 position ...					
-	0			Long Integer	r
Multi-line: yes	1		1		

acJogCircleSelected	$AC_JOG_CIRCLE_SELECTED				
JOG in circles has been selected					
-	0	0	1	UWord	r
Multi-line: yes	1		1		

acJogCoord	$AC_JOG_COORD				
Setting the coordinate system for the manual travel 0: Work 1: SZS					
-	0	0	1	Long Integer	rw
Multi-line: no					

acLiftFast	$AC_LIFTFAST				
Information about the execution of LIFTFAST. At the start of the LIFTFAST operation, the variable is set internally by the NC to the value "1". The variable must be reset to the initial state ($AC_LIFTFAST=0) by the evaluating program (if available), in order to be able to detect a subsequent LIFTFAST. 0: Initial state 1: LIFTFAST has been executed					
-	0	0	1	UWord	r
Multi-line: yes	1		1		

acMToolLengthIndex	$AC_M_TOOL_LENGTH_INDEX				
The variable returns the number of the tool length components (1, 2 or 3 corresponding to the length components L1, L2, L3) for milling tools, which is assigned to the geometry axis that was transferred as an index. In this context, milling tools are all tools with a tool type that does not lie between 400 and 599. The assignment does not take into account any rotations (e.g. by kinematic transformations) or frames. It depends on the active plane and the setting data SD42950 $SC_TOOL_LENGTH_TYPE and SD42940 $SC_TOOL_LENGTH_CONST. If the tool length component is active with a negative sign, the index is output with a negative sign. This can be the case if the hundreds digit of setting data SD42940 $SC_TOOL_LENGTH_CONST equals 1.					
-	0	-3	3	UWord	r
Multi-line: yes	1		3		

acMToolLengthIndexS	$P_M_TOOL_LENGTH_INDEX				
The variable returns the number of the tool length components (1, 2 or 3 corresponding to the length components L1, L2, L3) for milling tools, which is assigned to the geometry axis that was transferred as an index. In this context, milling tools are all tools with a tool type that does not lie between 400 and 599. The assignment does not take into account any rotations (e.g. by kinematic transformations) or frames. It depends on the active plane and the setting data SD42950 $SC_TOOL_LENGTH_TYPE and SD42940 $SC_TOOL_LENGTH_CONST. If the tool length component is active with a negative sign, the index is output with a negative sign. This can be the case if the hundreds digit of setting data SD42940 $SC_TOOL_LENGTH_CONST equals 1.					
-	0	-3	3	UWord	r
Multi-line: yes	1		3		

acMea	$AC_MEA				
Touch probe has switched No. of touch probe					
-	0	0	1	UWord	r
Multi-line: yes	No. of touch probe		2		

acMeasActPlane	$AC_MEAS_ACT_PLANE				
Plane setting for measurement calculation 0: G17, 1: G18, 2: G19					
-		0	2	Long Integer	rw
Multi-line: yes	1		1		

acMeasChbfr	$AC_MEAS_CHBFR				
Channel basic frame screen form for setting up the new frame					
-	0	0		Long Integer	rw
Multi-line: no					

acMeasChsfr	$AC_MEAS_CHSFR				
System frame bit screen form for setting up the new frame					
-	0	0		Long Integer	rw
Multi-line: no					

acMeasCornerAngle	$AC_MEAS_CORNER_ANGLE				
Calculated cutting angle of corner					
Degree, user defined				Double	r
Multi-line: yes	1		1		

acMeasCornerSetangle	$AC_MEAS_CORNER_SETANGLE				
User-selectable setpoint cutting angle of corner Permissible input range between 0 and 180 degrees					
Degree, user defined		0	180.0	Double	rw
Multi-line: yes	1		1		

acMeasDNumber	$AC_MEAS_D_NUMBER				
Selected tool edge number					
-		0		Long Integer	rw
Multi-line: yes	1		1		

acMeasDiameter	$AC_MEAS_DIAMETER				
Calculated diameter					
mm, inch, user defined				Double	r
Multi-line: yes	1		1		

acMeasDirApproach	$AC_MEAS_DIR_APPROACH				
Approach direction towards workpiece 0: +x 1: -x 2: +y 3: -y 4: +z 5: -z					
-		0	5	Long Integer	rw
Multi-line: yes	1		1		

acMeasFineTrans	$AC_MEAS_FINE_TRANS				
Correction in fine offset 0: Correction in coarse translation 1: Correction in fine translation					
-		0	1	Long Integer	rw
Multi-line: yes	1		1		

acMeasFrameSelect	$AC_MEAS_FRAME_SELECT				
The frame calculated during workpiece measurement is entered in the selected frame. 0: $P_SETFR 10.. 25: $P_CHBFR[0..15] 50.. 65: $P_NCBFR[0..15] 100.. 199: $P_UIFR[0..99] 1010..1025: $P_CHBFR[0..15] 1050..1065: $P_NCBFR[0..15]					
-		0	1065	Long Integer	rw
Multi-line: yes	1		1		

acMeasInput	$AC_MEAS_INPUT[n]				
Data for the workpiece and tool measurement					
-	0			Double	rw
Multi-line: yes	Index		10		

acMeasLatch	$AC_MEAS_LATCH				
Save measuring points for workpiece and tool measurement 0: Clear measuring point 1: Write current axial values to measuring point					
-	0	0	1	Long Integer	rw
Multi-line: yes	Measuring point no.		4		

acMeasNcbfr	$AC_MEAS_NCBFR				
Global basic frame screen form for setting up the new frame					
-	0	0		Long Integer	rw
Multi-line: no					

acMeasP1Coord	$AC_MEAS_P1_COORD				
Coordinate system of the 1st measuring point 0: Work 1: BCS 2: MCS					
-	0	0		Long Integer	rw
Multi-line: no					

acMeasP2Coord	$AC_MEAS_P2_COORD				
Coordinate system of the 2nd measuring point 0: Work 1: BCS 2: MCS					
-	0	0		Long Integer	rw
Multi-line: no					

acMeasP3Coord	$AC_MEAS_P3_COORD				
Coordinate system of the 3rd measuring point 0: Work 1: BCS 2: MCS					
-	0	0		Long Integer	rw
Multi-line: no					

acMeasP4Coord	$AC_MEAS_P4_COORD				
Coordinate system of the 4th measuring point 0: Work 1: BCS 2: MCS					
-	0	0		Long Integer	rw
Multi-line: no					

acMeasPframe	$AC_MEAS_PFRAME				
Programmable frame is not included					
-	0	0	1	Long Integer	rw
Multi-line: no					

acMeasResults	$AC_MEAS_RESULTS[n]				
Measurement results					
-				Double	r
Multi-line: yes	Index		10		

acMeasScaleunit	$AC_MEAS_SCALEUNIT				
Unit of measurement for input and output values 0: Unit of measurement as configured 1: Unit of measurement in relation to active G code G70/G700/G71/G710					
-		0		Long Integer	rw
Multi-line: yes	1		1		

acMeasSema	$AC_MEAS_SEMA				
Variable for disabling and enabling the measurement interface 0: Not assigned 1: Assigned					
-	0	0	1	Long Integer	rw
Multi-line: yes	1		1		

acMeasSetCoord	$AC_MEAS_SET_COORD				
Coordinate system of the set point 0: Work 1: BCS 2: MCS					
-	0	0		Long Integer	rw
Multi-line: no					

acMeasTNumber	$AC_MEAS_T_NUMBER				
Selected tool number					
-		0		Long Integer	rw
Multi-line: yes	1		1		

acMeasToolLength	$AC_MEAS_TOOL_LENGTH				
Calculated tool length					
mm, inch, user defined				Double	r
Multi-line: yes	1		1		

acMeasToolMask	$AC_MEAS_TOOL_MASK				
Tool setting for the measurement calculation Bit 0: Tool radius is not included in the calculation					
-	0	0		Long Integer	rw
Multi-line: no					

acMeasType	$AC_MEAS_TYPE				
Measurement type specification 0: Default 1: x edge 2: y edge 3: z edge 4: Corner 1 5: Corner 2, 6: Corner 3 7: Corner 4 8: Hole 9: Shaft 10: Tool length 11: Tool diameter 12: Groove 13: Web 14: Actual value setting for geo and special axes 15: Actual value setting for special axes only 16: Edge_2P 17: Plane_Angles 18: Plane_Normal 19: Dimension_1 20: Dimension_2 21: Dimension_3					
-	0	0	21	Long Integer	rw
Multi-line: yes	1		1		

acMeasUifr	$AC_MEAS_UIFR				
Settable data management frame for setting up the new frame					
-	0	0	99	Long Integer	rw
Multi-line: no					

acMeasValid	$AC_MEAS_VALID				
Validity bits for measurement input values Bit 0: $AA_MEAS_POINT1[axis] Bit 1: $AA_MEAS_POINT2[axis] Bit 2: $AA_MEAS_POINT3[axis] Bit 3: $AA_MEAS_POINT4[axis] Bit 4: $AA_MEAS_SETPOINT[axis] Bit 5: $AC_MEAS_WP_SETANGLE Bit 6: $AC_MEAS_CORNER_SETANGLE Bit 7: $AC_MEAS_T_NUMBER Bit 8: $AC_MEAS_D_NUMBER Bit 9: $AC_MEAS_DIR_APPROACH Bit 10: $AC_MEAS_ACT_PLANE Bit 11: $AC_MEAS_FRAME_SELECT Bit 12: $AC_MEAS_TYPE Bit 13: $AC_MEAS_FINE_TRANS					
-		0		Long Integer	rw
Multi-line: yes	1		1		

acMeasWpAngle	$AC_MEAS_WP_ANGLE				
Calculated workpiece position angle					
Degree, user defined				Double	r
Multi-line: yes	1		1		

acMeasWpSetangle	$AC_MEAS_WP_SETANGLE				
User-selectable setpoint workpiece position angle Permissible input range less than +/- 90 degrees					
Degree, user defined		-90.0	90.0	Double	rw
Multi-line: yes	1		1		

acMonMin	$AC_MONMIN				
Ratio of the actual tool monitoring value to the setpoint					
-	0	0		Double	r
Multi-line: yes	1		1		

acMsNum	$AC_MSNUM				
Number of the master spindle 0: No spindle available 1..n: Number of the master spindle					
-	0	0		UWord	r
Multi-line: yes	1		1		

acMthNum	$AC_MTHNUM				
Number of the current master tool holder. Is only meaningful with active magazine management. 0: No master tool holder available 1..n: Number of the master tool holder					
-	0	0		UWord	r
Multi-line: yes	1		1		

acOTol	$AC_OTOL				
acOTol provides the orientation tolerance for compressor and smoothing which was used to preprocess the current main run block.					
Degree, user defined	0	0		Double	r
Multi-line: yes	1		1		

acOvr	$AC_OVR				
Path override for synchronous actions (Note: for SYNACT only)					
-				Double	r
Multi-line: yes	1		1		

acPRTimeA					
For simulation: Estimation of program runtime in seconds - downtime					
s, user defined				Double	rw
Multi-line: yes	1		1		

acPRTimeB					
For simulation: Estimation of program runtime in seconds - blockwise					
s, user defined				Double	r
Multi-line: yes	1		1		

acPRTimeM					
For simulation: Estimation of program runtime in seconds - machining time					
s, user defined				Double	rw
Multi-line: yes	1		1		

acPathAcc	$AC_PATHACC				
Path acceleration for real-time events					
m/s2, 1000 inch/ s2, user defined	0	0		Double	r
Multi-line: yes	1		1		

acPathJerk	$AC_PATHJERK				
Path jerk for real-time events					
mm/s3, 1000 inch / s3, user defined	0	0		Double	r
Multi-line: yes	1		1		

acPathn	$AC_PATHN				
Normalized path parameter (Note: for SYNACT only)					
-				Double	r
Multi-line: yes	1		1		

acPlcOvr	$AC_PLC_OVR				
Path override for synchronized actions specified by the PLC					
-	100	0		Double	r
Multi-line: yes	1		1		

acPltbb	$AC_PLTBB				
Path length from the beginning of the block in the BCS (Note: for SYNACT only)					
-				Double	r
Multi-line: yes	1		1		

acPlteb	$AC_PLTEB				
Path length to the end of the block in the BCS (Note: for SYNACT only)					
-				Double	r
Multi-line: yes	1		1		

acPrepActLoad	$AC_PREP_ACT_LOAD				
Current preprocessing run time					
-	0	0		Double	r
Multi-line: yes	1		1		

acPrepActLoadGross	$AC_PREP_ACT_LOAD_GROSS				
Current preprocessing gross run time					
-	0	0		Double	r
Multi-line: yes	1		1		

acPrepMaxLoad	$AC_PREP_MAX_LOAD				
Longest preprocessing run time					
-	0	0		Double	r
Multi-line: yes	1		1		

acPrepMaxLoadGross	$AC_PREP_MAX_LOAD_GROSS				
Longest preprocessing gross run time					
-	0	0		Double	r
Multi-line: yes	1		1		

acPrepMinLoad	$AC_PREP_MIN_LOAD				
Shortest preprocessing run time					
-	0	0		Double	r
Multi-line: yes	1		1		

acPrepMinLoadGross	$AC_PREP_MIN_LOAD_GROSS				
Shortest preprocessing gross run time					
-	0	0		Double	r
Multi-line: yes	1		1		

acProg	$AC_PROG				
Program status (identical to progStatus but with coding that corresponds to $AC_PROG) 0: aborted (reset) 1: halted (stop) 2: running (active) 3: waiting 4: interrupted					
-	0			UWord	r
Multi-line: yes	1		1		

acPtpSup					
Cartesian point-to-point travel (PTP) is supported by transformation 0: Cart. PTP travel is not supported 1: Cart. PTP travel is supported					
-	0	0	1	UWord	r
Multi-line: yes	1		1		

acSTolF	$AC_STOLF				
acSTolF indicates the G00 tolerance factor for compressor and smoothing with which the current main run block was prepared.					
-	0	0		Double	r
Multi-line: yes	1		1		

acSafeSynaMem	$AC_SAFE_SYNA_MEM				
Free Safety synchronized action elements The maximum number of elements is configured in $MC_MM_NUM_SAFE_SYNC_ELEMENTS					
-	0	0		UWord	r
Multi-line: yes	1		1		

acSimMode					
Variable $AC_SIM_MODE determines the simulation mode. The following values are possible: 0: Simulation not active. 1: Simulation mode active.					
-		0	1	Long Integer	r
Multi-line: yes	1		1		

acSimTimeBlock					
For the simulation: block processing time in seconds					
s, user defined				Double	r
Multi-line: yes	1		1		

acSimTimeStep					
For the simulation: time step in seconds					
s, user defined				Double	r
Multi-line: yes	1		1		

acStat	$AC_STAT				
Channel status (identical to chanStatus but with coding that corresponds to $AC_STAT) 0: reset 1: interrupted 2: active					
-	0			UWord	r
Multi-line: yes	1		1		

acSynaMem	$AC_SYNA_MEM				
Free memory for synchronous actions: Shows how many elements of the memory set with $MC_MM_NUM_SYNC_ELEMENTS are still free.					
-				UWord	r
Multi-line: yes	1		1		

acSynaState	$AC_SYNA_STATE				
The status of a synchronized action can be read using the variable. The line index is the ID of the modal or static synchronized action, for which the status should be read. The data is bit-coded, so that when required, also individual states can be masked or separately evaluated (bits that are not listed supply a value of 0) Bit 0 = 0: No lock Bit 0 = 1: PLC or synchronize actions are locked Bit 1 = 0: PLC is not locked Bit 1 = 1: PLC is locked Bit 2 = 0: Synchronized action is not locked Bit 2 = 1: Synchronized action is locked					
-				UDoubleword	r
Multi-line: no					

acSyncActLoad	$AC_SYNC_ACT_LOAD				
Current runtime for synchronized actions of the last IPO cycle in the channel					
-	0	0		Double	r
Multi-line: yes	1		1		

acSyncAverageLoad	$AC_SYNC_AVERAGE_LOAD				
Average runtime for synchronized actions of an IPO cycle in the channel					
-	0	0		Double	r
Multi-line: yes	1		1		

acSyncMaxLoad	$AC_SYNC_MAX_LOAD				
Longest runtime for synchronized actions of an IPO cycle in the channel					
-	0	0		Double	r
Multi-line: yes	1		1		

acTToolLengthIndex	$AC_T_TOOL_LENGTH_INDEX				
The variable returns the number of the tool length components (1, 2 or 3 corresponding to the length components L1, L2, L3) for turning and grinding tools, which is assigned to the geometry axis that was transferred as an index. In this context, turning and grinding tools are all tools with a tool type that lies between 400 and 599. The assignment does not take into account any rotations (e.g. by kinematic transformations) or frames. It depends on the active plane and the setting data SD42950 $SC_TOOL_LENGTH_TYPE, SD42940 $SC_TOOL_LENGTH_CONST and SD42942 $SC_TOOL_LENGTH_CONST_T. If the tool length component is active with a negative sign, the index is output with a negative sign. This can be the case if the hundreds digit of setting data SD42940 $SC_TOOL_LENGTH_CONST or the hundreds digit of setting data SD42942 $SC_TOOL_LENGTH_CONST_T equals 1.					
-	0	-3	3	UWord	r
Multi-line: yes	1		3		

acTToolLengthIndexS	$P_T_TOOL_LENGTH_INDEX				
The variable returns the number of the tool length components (1, 2 or 3 corresponding to the length components L1, L2, L3) for turning and grinding tools, which is assigned to the geometry axis that was transferred as an index. In this context, turning and grinding tools are all tools with a tool type that lies between 400 and 599. The assignment does not take into account any rotations (e.g. by kinematic transformations) or frames. It depends on the active plane and the setting data SD42950 $SC_TOOL_LENGTH_TYPE, SD42940 $SC_TOOL_LENGTH_CONST and SD42942 $SC_TOOL_LENGTH_CONST_T. If the tool length component is active with a negative sign, the index is output with a negative sign. This can be the case if the hundreds digit of setting data SD42940 $SC_TOOL_LENGTH_CONST or the hundreds digit of setting data SD42942 $SC_TOOL_LENGTH_CONST_T equals 1.					
-	0	-3	3	UWord	r
Multi-line: yes	1		3		

acTaneb	$AC_TANEB				
Tangent angle at the block end point					
-	0	0		Double	r
Multi-line: yes	1		1		

acTc	$AC_TC				
Active tool carrier					
-	0	0		Long Integer	r
Multi-line: yes	1		1		

acTcAckt	$AC_TC_ACKT				
Trigger variable ACKnowledgeTrigger always assumes a value of 1 for an IPO cycle when the PLC acknowledges a command of the tool management.					
-	0			Long Integer	r
Multi-line: yes	1		1		

acTcCmdt	$AC_TC_CMDT				
Trigger variable: CoMmadTrigger always assumes a value of 1 for an IPO cycle when a new command of the magazine management is output to the PLC.					
-	0			Long Integer	r
Multi-line: yes	1		1		

acThreadPitch	$AC_THREAD_PITCH				
Programmed lead					
-	0			Double	r
Multi-line: yes	1		1		

acThreadPitchAct	$AC_THREAD_PITCH_ACT				
Current lead					
-	0			Double	r
Multi-line: yes	1		1		

acThreadPitchInc	$AC_THREAD_PITCH_INC				
Current lead change					
-	0			Double	r
Multi-line: yes	1		1		

acTime	$AC_TIME				
Time from the beginning of the block in seconds (Note: for SYNACT only)					
s				Double	r
Multi-line: yes	1		1		

acTimec	$AC_TIMEC				
Time from the beginning of the block in interpolation cycles (Note: for SYNACT only)					
IPO cycle				Double	r
Multi-line: yes	1		1		

acTimer	$AC_TIMER[x] x = TimerNo				
Time variable in seconds (Note: for SYNACT only)					
s				Double	r
Multi-line: yes	Number of the time variable		$MN_MM_NUM_AC_TIMER		

acToolOAct	$AC_TOOL_O_ACT				
Supplies the setpoint of the current tool orientation in various coordinate systems. Possible values of the line index: 1, 2, 3: Components of the vector in BCS 4, 5, 6: Components of the vector in PCS/WCS 7, 8, 9: Components of the vector in ENS The orientation vector is normalized, i.e. it has the value 1.					
-	0	-1	1	Double	r
Multi-line: no			9		

acToolOCorr	$AC_TOOL_O_CORR				
Supplies the setpoint of the current tool orientation incl. overlays in various coordinate systems. Possible values of the line index: 1, 2, 3: Components of the vector in BCS 4, 5, 6: Components of the vector in PCS/WCS 7, 8, 9: Components of the vector in ENS The orientation vector is normalized, i.e. it has the value 1.					
-	0	-1	1	Double	r
Multi-line: no			9		

acToolOCorrD	$AC_TOOL_O_CORRD				
Supplies the setpoint of the current overlay of the tool orientation in various coordinate systems. Possible values of the line index: 1, 2, 3: Components of the vector in BCS 4, 5, 6: Components of the vector in PCS/WCS 7, 8, 9: Components of the vector in ENS This vector represents the difference between the two vectors acToolOCorr and acToolOAct					
-	0	-1	1	Double	r
Multi-line: no			9		

acToolODiff	$AC_TOOL_O_DIFF				
Supplies the remaining angle between the current vector and end vectors of the tool orientation block in various coordinate systems: Possible values of the line index: 1: Angle in BCS 2: Angle in PCS/WCS 3: Angle in ENS					
-	0	0	180	Double	r
Multi-line: yes	1		3		

acToolOEnd	$AC_TOOL_O_END				
Supplies the end orientation of the current block in various coordinate systems: Possible values of the line index: 1, 2, 3: Components of the vector in BCS 4, 5, 6: Components of the vector in PCS/WCS 7, 8, 9: Components of the vector in ENS The orientation vector is normalized, i.e. it has the value 1.					
-	0	-1	1	Double	r
Multi-line: yes	1: X component		9		

acToolRAct	$AC_TOOL_R_ACT				
Setpoint of the tool rotation in various coordinate systems: Possible values of the line index: 1, 2, 3: Components of the vector in BCS 4, 5, 6: Components of the vector in PCS/WCS 7, 8, 9: Components of the vector in ENS The orientation vector is normalized, i.e. it has the value 1.					
-	0	-1	1	Double	r
Multi-line: yes	1: X component		9		

acToolRCorr	$AC_TOOL_R_CORR				
Supplies the setpoint of the current rotation vector of the tool orientation incl. overlays in various coordinate systems. Possible values of the line index: 1, 2, 3: Components of the vector in BCS 4, 5, 6: Components of the vector in PCS/WCS 7, 8, 9: Components of the vector in ENS The orientation vector is normalized, i.e. it has the value 1.					
-	0	-1	1	Double	r
Multi-line: no			9		

acToolRCorrD	$AC_TOOL_R_CORRD				
Supplies the setpoint of the current overlay of the rotation of the tool in various coordinate systems. Possible values of the line index: 1, 2, 3: Components of the vector in BCS 4, 5, 6: Components of the vector in PCS/WCS 7, 8, 9: Components of the vector in ENS This vector represents the difference between the two vectors acToolRCorr and acToolRAct					
-	0	-1	1	Double	r
Multi-line: no			9		

acToolRDiff	$AC_TOOL_R_DIFF				
Angle remaining between the current and end rotation vectors of the tool orientation block in various coordinate systems: Possible values of the line index: 1: Angle in BCS 2: Angle in PCS/WCS 3: Angle in ENS					
-	0	0	180	Double	r
Multi-line: yes	1		3		

acToolREnd	$AC_TOOL_R_END				
End rotation vector of the current block in various coordinate systems: Possible values of the line index: 1, 2, 3: Components of the vector in BCS 4, 5, 6: Components of the vector in PCS/WCS 7, 8, 9: Components of the vector in ENS The orientation vector is normalized, i.e. it has the absolute value 1.					
-	0	-1	1	Double	r
Multi-line: yes	1: X component		9		

acTotalOvr	$AC_TOTAL_OVR				
Total path override for synchronized actions					
-	100	0		Double	r
Multi-line: yes	1		1		

acTrafo	$AC_TRAFO				
Code number of the active transformation (encoded as for $AC_TRAFO)					
-				UWord	r
Multi-line: yes	1		1		

acTrafoChain	$AC_TRAFO_CHAIN				
Active chained transformation Code numbers of the chained transformations of the active TRACON corresponding to machine data $MC_TRAFO_TYPE_m. 0: No master tool holder available 1..n: Number of the master tool holder					
-	0	0		UWord	r
Multi-line: yes	Index of the chained transformation		4		

acTrafoCorrElemP0	$AC_TRAFO_CORR_ELEM_P[0,n]				
Correction element in the section including the index 0 in the part chain of an active orientation transformation.					
mm, inch, user defined	0			Double	r
Multi-line: yes	Component index (X/Y/Z)		3		

acTrafoCorrElemP1	$AC_TRAFO_CORR_ELEM_P[1,n]				
Correction element in the section including the index 1 in the part chain of an active orientation transformation.					
mm, inch, user defined	0			Double	r
Multi-line: yes	Component index (X/Y/Z)		3		

acTrafoCorrElemP2	$AC_TRAFO_CORR_ELEM_P[2,n]				
Correction element in the section including the index 2 in the part chain of an active orientation transformation.					
mm, inch, user defined	0			Double	r
Multi-line: yes	Component index (X/Y/Z)		3		

acTrafoCorrElemP3	$AC_TRAFO_CORR_ELEM_P[3,n]				
Correction element in the section including the index 3 in the part chain of an active orientation transformation.					
mm, inch, user defined	0			Double	r
Multi-line: yes	Component index (X/Y/Z)		3		

acTrafoCorrElemT0	$AC_TRAFO_CORR_ELEM_T[0,n]				
Correction element in the section with the index 0 in the tool chain of an active orientation transformation.					
mm, inch, user defined	0			Double	r
Multi-line: yes	Component index (X/Y/Z)		3		

acTrafoCorrElemT1	$AC_TRAFO_CORR_ELEM_T[1,n]				
Correction element in the section with the index 1 in the tool chain of an active orientation transformation.					
mm, inch, user defined	0			Double	r
Multi-line: yes	Component index (X/Y/Z)		3		

acTrafoCorrElemT2	$AC_TRAFO_CORR_ELEM_T[2,n]				
Correction element in the section with the index 2 in the tool chain of an active orientation transformation.					
mm, inch, user defined	0			Double	r
Multi-line: yes	Component index (X/Y/Z)		3		

acTrafoCorrElemT3	$AC_TRAFO_CORR_ELEM_T[3,n]				
Correction element in the section with the index 3 in the tool chain of an active orientation transformation.					
mm, inch, user defined	0			Double	r
Multi-line: yes	Component index (X/Y/Z)		3		

acTrafoName	$AC_TRAFO_NAME				
Reads the name of a currently active kinematic transformation. If no transformation is active or if one transformation is active that has not been defined with kinematic chains, then this variable receives a zero string.					
-	"\0"			String [32]	r
Multi-line: yes	1		1		

acTrafoOriaxDirP0	$AC_TRAFO_ORIAX_DIR_P[0,n]				
Direction vector of the orientation axis with the index 0 in the part chain of an active orientation transformation.					
mm, inch, user defined	0			Double	r
Multi-line: yes	Component index (X/Y/Z)		3		

acTrafoOriaxDirP1	$AC_TRAFO_ORIAX_DIR_P[1,n]				
Direction vector of the orientation axis with the index 1 in the part chain of an active orientation transformation.					
mm, inch, user defined	0			Double	r
Multi-line: yes	Component index (X/Y/Z)		3		

acTrafoOriaxDirP2	$AC_TRAFO_ORIAX_DIR_P[2,n]				
Direction vector of the orientation axis with the index 2 in the part chain of an active orientation transformation.					
mm, inch, user defined	0			Double	r
Multi-line: yes	Component index (X/Y/Z)		3		

acTrafoOriaxDirT0	$AC_TRAFO_ORIAX_DIR_T[0,n]				
Direction vector of the orientation axis with the index 0 in the tool chain of an active orientation transformation.					
mm, inch, user defined	0			Double	r
Multi-line: yes	Component index (X/Y/Z)		3		

acTrafoOriaxDirT1	$AC_TRAFO_ORIAX_DIR_T[1,n]				
Direction vector of the orientation axis with the index 1 in the tool chain of an active orientation transformation.					
mm, inch, user defined	0			Double	r
Multi-line: yes	Component index (X/Y/Z)		3		

acTrafoOriaxDirT2	$AC_TRAFO_ORIAX_DIR_T[2,n]				
Direction vector of the orientation axis with the index 2 in the tool chain of an active orientation transformation.					
mm, inch, user defined	0			Double	r
Multi-line: yes	Component index (X/Y/Z)		3		

acTrafoOriaxLoc	$AC_TRAFO_ORIAX_LOC				
The variable supplies the decimal-coded index of an orientation axis in the kinematic chain of an orientation transformation. The ten's place designates the part chain in which the orientation axis is included (0: Part chain; 1: Tool chain) and the unit's place the axis index when counting from the start to the end of the chain.					
-	-1	-3	12	Long Integer	r
Multi-line: yes	(Axis index)		numMachAxes		

acTrafoPar	$AC_TRAFO_PAR[n]				
Supplies the value of parameter 'n' of the current transformation, e.g. the cylinder diameter in the case of TRACYL					
-				Double	r
Multi-line: yes	Number of the parameter (dependent on the transformation type)		8		

acTrafoParSet	$AC_TRAFO_PARSET				
The variable is '0' if no transformation is active. If a conventionally defined (i.e. not by kinematic chains) transformation is active, the variable contains the number of the current transformation data record. If a transformation defined by kinematic chains is active, the variable contains the number of the $NT data record with an offset of 1000, i.e. the first transformation returns the value 1001.					
-	0			UWord	r
Multi-line: yes	1		1		

acTrafoSectionP0	$AC_TRAFO_SECTION_P[0,n]				
Section with the index 0 in the part chain of an active orientation transformation.					
mm, inch, user defined	0			Double	r
Multi-line: yes	Component index (X/Y/Z)		3		

acTrafoSectionP1	$AC_TRAFO_SECTION_P[1,n]				
Section with the index 1 in the part chain of an active orientation transformation.					
mm, inch, user defined	0			Double	r
Multi-line: yes	Component index (X/Y/Z)		3		

acTrafoSectionP2	$AC_TRAFO_SECTION_P[2,n]				
Section with the index 2 in the part chain of an active orientation transformation.					
mm, inch, user defined	0			Double	r
Multi-line: yes	Component index (X/Y/Z)		3		

acTrafoSectionP3	$AC_TRAFO_SECTION_P[3,n]				
Section with the index 3 in the part chain of an active orientation transformation.					
mm, inch, user defined	0			Double	r
Multi-line: yes	Component index (X/Y/Z)		3		

acTrafoSectionT0	$AC_TRAFO_SECTION_T[0,n]				
Section with the index 0 in the tool chain of an active orientation transformation.					
mm, inch, user defined	0			Double	r
Multi-line: yes	Component index (X/Y/Z)		3		

acTrafoSectionT1	$AC_TRAFO_SECTION_T[1,n]				
Section with the index 1 in the tool chain of an active orientation transformation.					
mm, inch, user defined	0			Double	r
Multi-line: yes	Component index (X/Y/Z)		3		

acTrafoSectionT2	$AC_TRAFO_SECTION_T[2,n]				
Section with the index 2 in the tool chain of an active orientation transformation.					
mm, inch, user defined	0			Double	r
Multi-line: yes	Component index (X/Y/Z)		3		

acTrafoSectionT3	$AC_TRAFO_SECTION_T[3,n]				
Section with the index 3 in the tool chain of an active orientation transformation.					
mm, inch, user defined	0			Double	r
Multi-line: yes	Component index (X/Y/Z)		3		

acVactB	$AC_VACTB				
Path velocity in basic coordinate system					
mm/min, inch/min, user defined	0			Double	r
Multi-line: yes	1		1		

acVactBf	$AC_VACTBF				
Path velocity in the BCS. FGroup and FGREF are taken into account.					
mm/min, inch/min, user defined	0	0		Double	r
Multi-line: yes	1		1		

acVactWf	$AC_VACTWF				
Path velocity in the workpiece coordinate system. FGroup and FGREF are taken into account.					
mm/min, inch/min, user defined	0	0		Double	r
Multi-line: yes	1		1		

acVactw	$AC_VACTW				
Path velocity in the work piece coordinate system (Note: for SYNACT only)					
-				Double	r
Multi-line: yes	1		1		

acVc	$AC_VC				
Additive path feedrate correction value for synchronous actions (Note: for SYNACT only)					
-				Double	r
Multi-line: yes	1		1		

actCollPosMcsPacked					
Position of a channel axis with a collision in the MCS. The positions can be read for all configured channel axes. Row index corresponds to 1 1st configured channel axis 2 2nd configured channel axis .. 20 20th configured channel axis					
mm, inch, degree, user defined	0			Double	r
Multi-line: yes	Maximum number of configured channel axes		MAXNUM_AXES_PER_CHAN		

actDLNumber	$P_DLNO				
Number of active total offset DL					
-				UWord	r
Multi-line: yes	1				

actDLNumberS					
Corresponds to actDLNumber for block search with calculation Caution: This variable is not available for the Variable Service, but only for logging in the case of block search events!.					
-				UWord	r
Multi-line: yes	1				

actDNumber	$P_TOOL				
Number of active tool edge					
-		0	9	UWord	r
Multi-line: no					

actDNumberFanuc					
Replaced by actDNumberFanuc32					
-				UWord	r
Multi-line: yes	1		1		

actDNumberFanuc32					
With programming in ISO Dialect mode: Offset memory number radius. Assigned only in conjunction with ISO Dialect M external language.					
-				Long Integer	r
Multi-line: yes	1		1		

actDNumberS					
Corresponds to actNumber for block search with calculation Attention: This variable is available for protocolling block search events only, not for the Variable Service!					
-				UWord	rw
Multi-line: yes	1		1		

actDuploNumber					
Duplo number of active tool					
-	0			UWord	r
Multi-line: no			1		

actFeedRateIpo					
Actual value of the interpolation feedrate. The actual value is the feed actually moved with. (depends on the acceleration profiles, LookAhead, velocity limits etc.) The variable 'feedRateIpoUnit' defines the physical unit.					
mm/min, inch/min, user defined				Double	r
Multi-line: no					

actFeedRateTechIpo					
Interpolation feed extended, actual value. The actual value is the actually traversed feed rate (depends on the acceleration profiles, LookAhead, velocity limits etc.). The variable 'feedRateIpoUnit' defines the physical unit (mm/min, mm/rev or mm/tooth).					
mm/min, inch/min, user defined				Double	r
Multi-line: no					

actFrameIndex	$P_UIFRNUM				
Index of the active set frame (index in G group 8 "Settable work offset"). Frames 0 - 4 (corresponds to G500 ... G57) can be set in the standard version. The number of frames can be changed via machine data MM_NUM_USER_FRAMES. 0 = no frame selected 1 = G54 2 = G55 3 = G56 4 = G57 5 = G505 to 99 = G599					
-				UWord	r
Multi-line: no					

actGrindingFrameIndex	$P_GFRNUM				
Index of the actively set grinding frame. A grinding data management frame becomes an active grinding frame when executing GFRAME0 to GFRAME100. 0 = GFRAME0 = No frame selected 1 = GFRAME1 to 100 = GFRAME100					
-				UWord	r
Multi-line: no					

actHNumberFanuc					
Replaced by actHNumberFanuc32					
-				UWord	r
Multi-line: yes	1		1		

actHNumberFanuc32					
With programming in ISO Dialect mode: Offset memory number length. Assigned only in conjunction with ISO Dialect M external language.					
-				Long Integer	r
Multi-line: yes	1		1		

actIpoType					
Active interpolation mode used for the path motion. This date corresponds to a large degree to the SNCF:ncFktBin for the first G-group. The value differs for automatically generated intermediate blocks only. This is e.g. the case if two lines are connected with an arc by the command RND. The value is the index of the active G-code (analog with SNCF:ncFktBin)					
-				UWord	r
Multi-line: yes	1		1		

actIpoTypeS					
Active mode of interpolation applied during block searches. This data is very similar to SNCF:ncFktBinS for the 1st G group. Its value is different only in the case of automatically generated intermediate blocks, such as when, for example, two straight lines are connected to an arc by means of command RND. The value is the index of the active G function (analogous to SNCF:ncFktBinS).					
-				UWord	r
Multi-line: yes	1		1		

actLanguage					
Active language mode 0: Siemens 1: ISO mode 2: Reserviert for later language expansions					
-				UWord	r
Multi-line: yes	1				

actMTNumber	-				
Number of the multi-tool in which the active tool is contained. The value is zero if the active tool is not contained in a multi-tool.					
-				UWord	r
Multi-line: yes	1		1		

actMTPlaceNumber	$AC_TC_				
Number of the multi-tool place in which the active tool is contained. The value is zero if the active tool is not contained in a multi-tool.					
-				UWord	r
Multi-line: yes	1		1		

actMasterToolHolderNo					
Active number of the master tool holder. Especially for $MC_RESET_MODE_MASK, Bit0=0, this is the value of SETMS or SETMTH last programmed in the RESET status of the NCK. Especially for $MC_RESET_MODE_MASK, Bit0=1, this is the value in the RESET status of the NCK for $MC_SPIND_DEF_MASTER_SPIND (if $MC_TOOL_MANAGEMENT_TOOLHOLDER=0); or $MC_TOOL_MANAGEMENT_TOOLHOLDER (if $MC_TOOL_MANAGEMENT_TOOLHOLDER > 0)					
-		1	max. Anzahl der Kanalachsen	UWord	r
Multi-line: yes	1		1		

actOriToolLength1					
X component in workpiece coordinate system (WCS) of active tool length, taking into account the tool orientation, incl. adapter data, mirroring and TCARR (orientable toolholder).					
-	0			Double	r
Multi-line: yes	1		1		

actOriToolLength2					
Y component in workpiece coordinate system (WCS) of active tool length, taking into account the tool orientation, incl. adapter data, mirroring and TCARR (orientation-capable toolholder).					
-	0			Double	r
Multi-line: yes	1		1		

actOriToolLength3					
Z component in workpiece coordinate system (WCS) of active tool length, taking into account the tool orientation, incl. adapter data, mirroring and TCARR (orientation-capable toolholder).					
-	0			Double	r
Multi-line: yes	1		1		

actParts	$AC_ACTUAL_PARTS				
Total number of workpieces machined in current run: This counter registers the number of workpieces machined since it started. When the required number is reached, the counter is set to zero automatically.					
-	0			Double	rw
Multi-line: no					

actProgNetTime	$AC_ACT_PROG_NET_TIME				
The current net runtime of the current program, that is the time in which the program was stopped, is subtracted. actProgNetTime is automatically reset to zero with the part program start in automatic mode, chanel status RESET. The net runtime does not include the time during which the program pauses on account of override=0. actProgNetTime can be further manipulated with progNetTimeTrigger. Seconds					
s, user defined	0	0		Double	r
Multi-line: yes	1		1		

actTNumber	$P_TOOLNO				W1
Number of active tool					
-		0	32000	UWord	r
Multi-line: no					

actTNumberLong					
Number of the active tool using flat D-numbers with up to 8 digits					
-				Long Integer	r
Multi-line: yes	1		1		

actTNumberS					
Corresponds to actTNumber for block search with calculation. Attention: This variable is available for protocolling the block search events only, not for the Variable Service!					
-				UWord	rw
Multi-line: yes	1		1		

actToolAdapterBaseLength					
Returns the components of the adapter or base dimension of the active tool, that is the contribution of the components $TC_ADPT1[..] - $TC_ADPT3[..] or $TC_DP21[..] - $TC_DP233[..] in different coordinate systems. The adapter and base dimension are mutually exclusive. This means that only one of these two components can contribute a value unequal to zero to the tool length. Both the component and the coordinate system are selected from the line index. Three indices are required for each coordinate system (lengths L1, L2, L3). The following assignment applies: Line indices 1 - 3: Components in the workpiece coordinate system (PCS). Line indices 4 - 6: Components in the basic coordinate system (BCS). Line indices 7 - 9: Components in the machine coordinate system (MCS). Line indices 10 - 12: Components in the tool coordinate system (TCS). Line indices 13 - 15: Components in the settable zero system (SZS).					
mm, inch, user defined				Double	r
Multi-line: yes	1		15		

actToolDataBeforeSearch	$P_....._BEFORE_SEARCH_RUN				
Data for determining the active tool offset before the search, that is in reset status before the start of the search. After reaching the search target, the value is set to the current value for each programming of master tool holder, spindle, D no or DL no. -P1: Master tool holder or spindle before the search ($AC_MTHNUM_BEFORE_SEARCH) Programming "MTH(no)" or "MS(no)" after reaching the search target, this variable then returns the same value as acMthNum. -P2: Active D no. before the search (see $P_D_BEFORE_SEARCH) Programming "D" after reaching the search target, this variable then returns the same value as actDNumber. -P3: Active DL no. before the search (see $P_DL_BEFORE_SEARCH) Programming "DL" after reaching the search target, this variable then returns the same value as actDLNumber.					
-	0	0		UWord	r
Multi-line: yes	Parameter number		numSearchRunToolParams		

actToolEdgeCenterPosEns					
Corresponds to actToolEdgeCenterPosEns in the SEGA block for the three geometry axes The variable consists of three values of the DOUBLE type, i.e. is 24 bytes long.					
-	0	0		Double	r
Multi-line: yes	1		1		

actToolEntryCorrLength					
Returns the components of the total offset of the active tool, that is the contribution of the components $TC_ECPx3[..] - $TC_ECPx5[..] in different coordinate systems. The letter "x" in the components $TC_SCPx3[..] etc. stands for the DL number. Both the component and the coordinate system are selected from the line index. Three indices are required for each coordinate system (lengths L1, L2, L3). The following assignment applies: Line indices 1 - 3: Components in the workpiece coordinate system (PCS). Line indices 4 - 6: Components in the basic coordinate system (BCS). Line indices 7 - 9: Components in the machine coordinate system (MCS). Line indices 10 - 12: Components in the tool coordinate system (TCS). Line indices 13 - 15: Components in the settable zero system (SZS).					
mm, inch, user defined				Double	r
Multi-line: yes	1		15		

actToolGeoLength					
Returns the length components of the geometry component of the active tool, that is the contribution of the components $TC_DP3[..] - $TC_DP5[..] in different coordinate systems. Both the component and the coordinate system are selected from the line index. Three indices are required for each coordinate system (lengths L1, L2, L3). The following assignment applies: Line indices 1 - 3: Components in the workpiece coordinate system (PCS). Line indices 4 - 6: Components in the basic coordinate system (BCS). Line indices 7 - 9: Components in the machine coordinate system (MCS). Line indices 10 - 12: Components in the tool coordinate system (TCS). Line indices 13 - 15: Components in the settable zero system (SZS).					
mm, inch, user defined				Double	r
Multi-line: yes	1		15		

actToolGeoLengthWear					
Returns the components of the wear component of the tool length of the active tool, that is the contribution of the components $TC_DP12[..] - $TC_DP14[..] in different coordinate systems. Both the component and the coordinate system are selected from the line index. Three indices are required for each coordinate system (lengths L1, L2, L3). The following assignment applies: Line indices 1 - 3: Components in the workpiece coordinate system (PCS). Line indices 4 - 6: Components in the basic coordinate system (BCS). Line indices 7 - 9: Components in the machine coordinate system (MCS). Line indices 10 - 12: Components in the tool coordinate system (TCS). Line indices 13 - 15: Components in the settable zero system (SZS).					
mm, inch, user defined				Double	r
Multi-line: yes	1		15		

actToolIdent					W1
Identifier of active tool					
-	"\0"			String [32]	r
Multi-line: no			1		

actToolLength1	$P_TOOLL[1]				W1
Active tool length 1					
mm, inch, user defined				Double	r
Multi-line: no					

actToolLength2	$P_TOOLL[2]				W!
Active tool length 2					
mm, inch, user defined				Double	r
Multi-line: no					

actToolLength3	$P_TOOLL[3]				W1
Active tool length 3					
mm, inch, user defined				Double	r
Multi-line: no					

actToolRadius	$P_TOOLR				W1
Active tool radius					
mm, inch, user defined				Double	r
Multi-line: no					

actToolSumCorrLength					
Returns the components of the total offset of the active tool, that is the contribution of the components $TC_SCPx3[..] - $TC_SCPx5[..] in different coordinate systems. The letter "x" in the components $TC_SCPx3[..] etc. stands for the DL number. Both the component and the coordinate system are selected from the line index. Three indices are required for each coordinate system (lengths L1, L2, L3). The following assignment applies: Line indices 1 - 3: Components in the workpiece coordinate system (PCS). Line indices 4 - 6: Components in the basic coordinate system (BCS). Line indices 7 - 9: Components in the machine coordinate system (MCS). Line indices 10 - 12: Components in the tool coordinate system (TCS). Line indices 13 - 15: Components in the settable zero system (SZS).					
mm, inch, user defined				Double	r
Multi-line: yes	1		15		

actToolToolCarrierLength					
Returns the components of the contribution of the orientable tool carrier (ToolCarrier) to the length of the active tool in different coordinate systems. Both the component and the coordinate system are selected from the line index. Three indices are required for each coordinate system (lengths L1, L2, L3). The following assignment applies: Line indices 1 - 3: Components in the workpiece coordinate system (PCS). Line indices 4 - 6: Components in the basic coordinate system (BCS). Line indices 7 - 9: Components in the machine coordinate system (MCS). Line indices 10 - 12: Components in the tool coordinate system (TCS). Line indices 13 - 15: Components in the settable zero system (SZS).					
mm, inch, user defined				Double	r
Multi-line: yes	1		15		

actToolTotalLength					
Returns the components of the total effective length of the active tool in different coordinate systems. Both the component and the coordinate system are selected from the line index. Three indices are required for each coordinate system (lengths L1, L2, L3). The following assignment applies: Line indices 1 - 3: Components in the workpiece coordinate system (PCS). Line indices 4 - 6: Components in the basic coordinate system (BCS). Line indices 7 - 9: Components in the machine coordinate system (MCS). Line indices 10 - 12: Components in the tool coordinate system (TCS). Line indices 13 - 15: Components in the settable zero system (SZS).					
mm, inch, user defined				Double	r
Multi-line: yes	1		15		

actTransform					
Active transformation					
-	\0			String [32]	r
Multi-line: yes	1		1		

actWaCSCoordSys	$AC_WORKAREA_CS_COORD_SYSTEM				
Coordinate system of the active coordinate system-specific working area limitation Identifier for the coordinate system in which the working area limitation is to apply. The following are valid: 0: Working area limitation in the WCS 3: Working area limitation in the SZS					
-	0			UWord	r
Multi-line: yes	1		1		

actWaCSLimitMinus	$AC_WORKAREA_CS_LIMIT_MINUS				
Position of the coordinate system-specific working area limitation in the negative direction for the addressed axis and working area group. Position of the working area limitation in the negative direction					
mm, inch, degree, user defined	0			Double	r
Multi-line: yes	Channel axis index		numMachAxes		

actWaCSLimitPlus	$AC_WORKAREA_CS_LIMIT_PLUS				
Position of the coordinate system-specific working area limitation in the positive direction for the addressed axis and working area group Position of the working area limitation in the positive direction					
mm, inch, degree, user defined	0			Double	r
Multi-line: yes	Channel axis index		numMachAxes		

actWaCSMinusEnable	$AC_WORKAREA_CS_MINUS_ENABLE				
The coordinate-specific working area limitation in the negative direction of actWaCSLimitMinus is valid. TRUE: The value in the variable actWaCSLimitMinus is valid for the axis. FALSE: There is no limitation in the coordinate-specific working area in the negative direction for this axis.					
-	0			UWord	r
Multi-line: yes	Channel axis index		numMachAxes		

actWaCSPlusEnable	$AC_WORKAREA_CS_PLUS_ENABLE				
The coordinate system-specific working area limitation in the positive direction of actWaCSLimitPlus is valid. TRUE: The value in the variable actWaCSLimitPlus is valid for the axis. FALSE: There is no limitation for this axis in the positive direction in the coordinate-specific working area.					
-	0			UWord	r
Multi-line: yes	Channel axis index		numMachAxes		

actWalimGroupNo	$AC_WORKAREA_CS_GROUP				
Active working area group in the IPO Writing is only possible if the channel has been aborted or stopped 0: Not activated n: $MC_MM_NUM_WORKAREA_CS_GROUPS					
-	0	0	10	UWord	rw
Multi-line: yes	1		1		

allAxesRefActive	DB21-28, DBX36.2				
Code specifying whether all axes are referenced 1 = all axes referenced 0 = at least 1 axis not referenced					
-				UWord	r
Multi-line: no					

allAxesStopped					
Code specifying whether axes are in exact stop 0 = at least one axis is not in exact stop 1 = All axes in exact stop					
-				UWord	r
Multi-line: no					

basisFrameMask	$P_CHBFRMASK				
Display indicating which channel-specific basic frames are active Every bit in the mask indicates whether the appropriate basic frame is active. Bit0 = 1st basic frame, Bit1 = 2nd basic frame, etc.					
-				UWord	r
Multi-line: yes	1		1		

blockProgInfo	$AC_BLOCK_PROGSTATE				
Returns the information of a main run block. Bit-coded: Bit 0: Block is end of main program (M02, M17, M30 or RET(ASUB)) Bit 1: Block is end of subprogram Bit 2: Block is last initializing block					
-	0	0		Long Integer	r
Multi-line: yes	1		1		

blockType	$AC_BLOCKTYPE				
Identifies the type of a block (programmed or generated internally) 0: No internally generated block 1: Internally generated block, but cannot be specified in detail 2: Block was generated on chamfering/rounding 3: Smooth approach and retraction (SAR) 4: Block was generated during tool offset 5: Block was generated on smoothing 6: Block was generated by TLIFT (tangential correction) 7: Block was generated during path segmentation 8: Block was generated by compile cycles					
-	0	0	8	Long Integer	r
Multi-line: yes	1		1		

blockTypeInfo	$AC_BLOCKTYPEINFO				
Detailed information on block type The value range and the meaning of this variable depend on the current value of system variable blockType System variable blockTypeInfo can be used to request additional information on variable blockType. Depending on the value of system variable blockType, various values are then possible: 1. General internally generated block: blockType = 1 blockTypeInfo = 1000 and does not include any additional information. 2. Chamfer/round: blockType = 2 2001: straight 2002: circle 3. SAR: blockType = 3 3001: Approach with straight 3002: Approach with quadrant 3003: Approach with semicircle 4. Tool offset: blockType = 4 4001: Approach block after STOPRE 4002: Link blocks if intersection not found 4003: Point circle on inside corners (only with TRACYL) 4004: Bypass circle (or conic) on outside corners 4005: Approach blocks for offset suppression 4006: Approach blocks for reactivation of TRC 4007: Blocks are split because curvature is too high 4008: Compensation blocks for 3D front milling (tool vector || plane vector) 5. Corner rounding: blockType = 5 5001: Rounding contour through G641 5002: Rounding contour through G642 5003: Rounding contour through G643 5004: Rounding contour through G644 6. TLIFT: blockType = 6 6001: TLIFT block with linear movement of the tangential axis but without retraction movement. 6002: TLIFT block with non-linear movement of the tangential axis (polynomial) but without retraction movement. 6003: TLIFT block with retraction movement. Tangential axis movement and retraction movement start simultaneously. 6004: TLIFT block with retraction movement. Tangential axis does not start until a specific retraction position is reached. 7. Path segmentation: blockType = 7 7001: Programmed path segmentation without punching or nibbling active. 7002: Programmed path segmentation with punching or nibbling active. 7003: Automatically generated path segmentation. 8. Compile cycles: blockType = 8 In this case, system variable $AC_BLOCKTYPEINFO includes the ID of the compile cycles application that generated the block.					
-	0	0		Long Integer	r
Multi-line: yes	1		1		

cIn	$C_IN[n]				
Signal from PLC to cycle (reserved for SIEMENS application, e.g. ShopMill/ManualTurn)					
-				UWord	r
Multi-line: yes	No. of input signal		16		

cOut	$C_OUT[n]				
Signal from cycle to PLC (reserved for SIEMENS application, e.g. ShopMill/ManualTurn)					
-				UWord	r
Multi-line: yes	No. of output signal		16		

chanAlarm	DB21-28, DBX36.6 und DBX36.7				A2
Code whether NCK alarm pending. 0 = no alarm in this channel 1 = alarm without stop 2 = alarm with stop					
-				UWord	r
Multi-line: no					

chanAxisNoGap					
Display of existing axes, i.e. no axis gap in channel. Bits 0-31 represent the axes of the channel. Bitn = 0: Axis does not exist. Bitn = 1: Axis does exist.					
-		0		Long Integer	r
Multi-line: yes	1		1		

chanStartLockState					
Status of the channel-specific start disable. Also see PI _N_STRTLK and _N_STRTUL. 0: No start disable 1: Start disable is activated					
-	0	0	1	UWord	r
Multi-line: yes	1		1		

chanStatus	DB21-28, DBX35.5-DBX35.7				K1
Channel status 0 = RESET 1 = active 2 = interrupted					
-				UWord	r
Multi-line: no					

changeAxConfCounter					
A counter which is incremented if the axes configuration has changed. This is the case, if e.g. geometry axes are switched or axes have been changed between channels. The counter is set to 0 at PowerOn and it might overflow. You cannot be sure, that the axes configuration actually has changed when the counter is incremented.					
-				UWord	r
Multi-line: yes	1		1		

cmdDwellTime					
Programmed dwell time See timeOrRevolDwell					
s, user defined	0	0		Double	r
Multi-line: yes	1		1		

cmdFeedRateIpo	$AC_F				
Desired feedrate of the interpolation feedrate. The physical unit is defined in the variable 'feedRateIpoUnit'					
mm/min, inch/min, user defined				Double	r
Multi-line: no					

cmdFeedRateIpoS					
Interpolation feed during search. The physical unit is in the variable feedRateIpoUnitS					
mm/min, inch/min, user defined	0	0		Double	r
Multi-line: yes	1		1		

cmdTrafoParS	$P_TRAFO_PAR[n]				
Supplies the value of parameter 'n' of the programmed transformation, e.g. the cylinder diameter in the case of TRACYL					
-				Double	r
Multi-line: yes	Number of the parameter (dependent on the transformation type)		8		

cmdTrafoParSetS	$P_TRAFO_PARSET				
The variable is '0' if no transformation is active. If a conventionally defined (i.e. not by kinematic chains) transformation is active, the variable contains the number of the current transformation data record during block search. If a transformation defined by kinematic chains is active, the variable contains the number of the $NT data record with an offset of 1000, i.e. the first transformation returns the value 1001 (during block search).					
-	0			UWord	r
Multi-line: yes	1		1		

cmdTrafoS	$P_TRAFO				
Code number of programmed transformation for block search Coding as for variable $AC_TRAFO					
-	0			UWord	r
Multi-line: yes	1		1		

contourDev					
Contour deviation					
mm, inch, user defined				Double	r
Multi-line: no					

corrBlActive					
Incorrect block has occurred (correction block) 0 = no incorrect block 1 = incorrect block					
-				UWord	r
Multi-line: no					

cycServRestricted					
Code whether restricted cyclic variable service is available. This is a privileged variable: Cyclic result acknowledgements for this variable are produced even if the cyclic services are no longer served by the NCK because of block cycle time problems. Caution: Privileged variables lose this characteristic if they are mixed with non-privileged variables in one request. -> Do not combine the variable cycServRestricted in a cluster with other variables! 0 = normal cycl. service 1 = no cyclic service (but acknowledgement)					
-				UWord	r
Multi-line: no					

delObjState					
Using PI _N_DELOBJ protection areas are deleted analogous to the language command DELOBJ(...). This OPI variable shows the status of the PI . 0 = PI successfully executed. -2 = Name of the object to be deleted not known. -3 = Index -1 not permitted -4 = Start index too large -5 = Illegal index when deleting a group (only -1 permitted) -6 = Start index smaller than end index -7 = End index too large					
-	0			Long Integer	r
Multi-line: yes	1		1		

delayFSt					
Delay Feed Stop, Stop is delayed in the current program area 0: Stop in the current program area is effective immediately 1: Stop in the current program area is effective after a delay 2: Stop in the current program area is effective immediately (same as 0), although a stop delay area was programmed in the parts program. (This means that the NCK could not switch on the stop delay area.)					
-	0	0	2	UWord	r
Multi-line: yes	1		1		

direction					
Traversing direction 0 = normal travel 1 = forward travel 2 = reverse travel 3 = reference point cycle 4 = stop state					
-				UWord	r
Multi-line: no					

enableOvrRapidFactor					
Activate additional rapid traverse override $SC_OVR_RAPID_FACTOR 0: not activated 1: activated					
-	0	0	1	UWord	rw
Multi-line: yes	1		1		

extProgActive	DB21-28, DBB32.0				
Flag indicating whether program execution from external is active. 0 = inactive 1 = active					
-				UWord	r
Multi-line: no					

feedRateIpoOvr					
Interpolation feedrate, override					
%				Double	r
Multi-line: no					

feedRateIpoUnit					
Interpolation feedrate, units 0 = mm/min 1 = mm/rev 2 = inch/min 3 = inch/rev					
-				UWord	r
Multi-line: no					

feedRateIpoUnitS					
Interpolation feed, units during search 0 = mm/min 1 = mm/rev 2 = inch/min 3 = inch/rev					
-	0	0		UWord	r
Multi-line: yes	1		1		

findBlActive	DB21-28, DBX33.4				K1
Code whether block search is active. 0 = not active 1 = active					
-				UWord	r
Multi-line: no					

gccState	$PC_GCC_STATE				
The variable shows the internal status of the G code converter. Status = 0 -> The G code converter is not selected. Status = 1 -> The G code converter is selected by HMI, but a trace has not yet been generated. Status = 2 -> The G code converter is active (after NC START), the trace is being output. Status = 3 -> The G code converter is active but interrupted by the language command GCCDISABLE, there is no output into the trace file.					
-	0	0	3	Long Integer	r
Multi-line: yes	1		1		

incoapB	$P_INCOAP_B				
Boolean supply and return parameter(s) of the COA application cutting generator					
-	0	0	1	UWord	rw
Multi-line: yes	Array index		incoapSize[1]		

incoapC	$P_INCOAP_C				
CHAR supply and return parameter(s) of the COA application cutting generator					
-	0	0	255	UWord	rw
Multi-line: yes	Array index		incoapSize[2]		

incoapI	$P_INCOAP_I				
INT supply and return parameter(s) of the COA application cutting generator					
-	0			UDoubleword	rw
Multi-line: yes	Array index		incoapSize[3]		

incoapR	$P_INCOAP_R				
DOUBLE supply and return parameter(s) of the COA application cutting generator					
-	0			Double	rw
Multi-line: yes	Array index		incoapSize[4]		

incoapS16	$P_INCOAP_S16[]				
CHAR16 supply and return parameter(s) of the COA application cutting generator					
-	0			String [16]	rw
Multi-line: yes	Array index		incoapSize[5]		

incoapS160	$P_INCOAP_S160[]				
CHAR160 supply and return parameter(s) of the COA application cutting generator					
-	0			String [160]	rw
Multi-line: yes	Array index		incoapSize[6]		

incoapS32	$P_INCOAP_S32[]				
CHAR32 supply and return parameter(s) of the COA application cutting generator					
-	0			String [32]	rw
Multi-line: yes	Array index		incoapSize[6]		

incoapSize	$P_INCOAP_SIZE[]				
Array size of the supply and return parameters of the COA application cutting generator					
-	0	0		UWord	rw
Multi-line: yes	1: Array size of $incoapB 2: Array size of $incoapC 3: Array size of $incoapI 4: Array size of $incoapR 5: Array size of $incoapS16 6: Array size of $incoapS32 7: Array size of $incoapS160		7		

isoActHDNo	$P_ISO2_HNO[n],$P_ISO2_DNO,$P_ISO3_NO				
The following applies to rows 1- 4: This value is only practical if ISO2 mode is permissible. This value contains the H number of the tool length offsets in the 3 geometry dimensions in rows 1 - 3 and the D number of the tool radius offset. If H99 is programmed, then all 3 geometry dimensions (=rows 1-3) have the value "-1" the radius (= row 4) has the value "-1" If H=D ($MN_EXTERN_TOOLPROG_MODE,Bit6=0), then these variables contain the last programmed D or H. If an offset D > 1 is selected in Siemens mode, all rows have the value "-2". If ISO2 mode cannot be activated ($MN_MM_EXTERN_CNC_SYSTEM != 4), the variable has the value=-3. The following applies to row 5: This value is only practicable if ISO3 mode is permissible. This value contains the current number of tool offsets in ISO3 mode. If an offset D > 1 is selected in Siemens mode, the value "-2" is returned. If ISO3 mode cannot be activated ($MN_MM_EXTERN_CNC_SYSTEM != 5), the variable value =-3. -3: ISO2 or ISO3 mode inactive -2: Siemens offset selected with D > 1. -1: H99 programmed in ISO mode, Siemens offset D1 active					
-	0			short Integer	r
Multi-line: yes	1: H number for L1 in ISO2 mode 2: H number for L2 in ISO2 mode 3: H number for L3 in ISO2 mode 4: H number for R in ISO2 mode 5: H number in ISO3 mode		5		

ludAccCounter					
Counter indicating that a new LUD ACC is available. If subprograms are called during an automatic program execution, a new set of LUDs becomes valid. In order to indicate to the HMI that it has to modify the display of the LUDs, respectively that the validity of the LUDs has changed, the variable 'ludAccCounter' is incremented. It is only necessary for the HMI to inquire a change of the variable's value, the value itself is of no importance.					
-				UWord	r
Multi-line: no					

machFunc	DB11, DBX7.0-DBX7.2				
Active channel machine function 0 = none 1 = REPOS 2 = TEACH IN 3 = REF 4 = TEACH-REPOS 5 = TEACH-REF					
-				UWord	r
Multi-line: no					

markActiveList					
Status array for the active marker in channel m. The first element (markActiveList[1]) of the array specifies the currently active marker number of this channel (channel m). The second element (markActiveList[2]) specifies bit-coded whether channel m is still waiting for the mark to be reached in the other channels (channel n), in short "waiting status". markActiveList[2] Bit-n == 1 Channel m is waiting for mark markActiveList[1] in channel n markActiveList[2] Bit-n == 0 Channel n has already reached mark markActiveList[1], or channel m is not waiting for mark markActiveList[1] at all markActiveList[1] == 0 Current channel m does not edit any wait marker markActiveList[1] == 1..99 Current channel m is positioned on the wait marker with markActiveList[1] markActiveList[2] Bit-n == 1 Channel m is waiting for mark markActiveList[1] in channel n markActiveList[2] Bit-n == 0 Channel n has already reached mark markActiveList[1], or channel m is not waiting for mark markActiveList[1] at all					
-	0	0	99	UWord	r
Multi-line: yes	1: Wait marker number 2: Bit-coded wait status for all channels		2		

nameIndex					
Using PI _N_NAMINT (NAMETOINT) string is searched in a 1-dimensional string field. In this OPI variable, the index of the found strings is returned to the string field. If the string is not found, the OPI variable is -1.					
-	0			Long Integer	r
Multi-line: yes	1		1		

ncProgEndCounter					
Counter which is incremented as soon as the NCK has processed an end of program.					
-	0	0		UWord	r
Multi-line: yes	1		1		

ncResetCounter					
Counter which is incremented with each 0->edge of the Reset key					
-	0	0		UWord	r
Multi-line: yes	1		1		

ncStartCounter					
Counter for the NC-start key. Pressing this key increments the variable 'ncStartCounter'. The value of the variable can be ignored, the HMI must just inquire the change of the variable to see whether the start-key has been pressed.					
-				UWord	r
Multi-line: no					

ncStartSignalCounter					
Counter that is incremented as soon as the channel-specific NC start signal has been activated in the VDI interface.					
-	0	0		UWord	r
Multi-line: yes	1		1		

numChanAlarms					
Number of present channel-specific alarms					
-				UWord	r
Multi-line: no					

numToolHolders	$P_MAGNS				
Number of tool holders/ spindles (buffer locations of the location type = spindle) from the magazine configuration of the TOA which are allocated to the channel. The number of tool holders / spindles is solely a function of the magazine configuration and does not change during an NC program execution. Value = 0, if there is no magazine configuration or the TMMG (tool management magazines) is not present in the NC.					
-	0	0	numMachAxes	UWord	r
Multi-line: no			1		

numTraceProtocEventType					
Logging: Number of standard event types					
-		0		UWord	r
Multi-line: yes	User No. (1-10)		10		

numTraceProtocOemEventType	$MM_PROTOC_NUM_ETP_OEM_TYP				
Logging: Number of OEM event types					
-		0		UWord	r
Multi-line: yes	User No. (1-10)		10		

oldProgNetTime	$AC_OLD_PROG_NET_TIME				
oldProgNetTime is the net runtime of the just correctly ended program. That is the program was not canceled with RESET, but terminated normally with M30. If a new program is started, oldProgNetTime remains unaffected until M30 is reached again. The implicit procedure of copying actProgNetTime to oldProgNetTime takes place only if progNetTimeTrigger is not written. oldProgNetTime s reset to zero with the PI "Select program". oldProgNetTime can be reset to zero by explicitly writing 0.0, other values may not be written. Seconds					
s, user defined	0	0		Double	rw
Multi-line: yes	1		1		

oldProgNetTimeCounter	$AC_OLD_PROG_NET_TIME_COUNT				
This is zero in the power ON status. oldProgNetTimeCounter is always increased when the NCK has newly written oldProgNetTime. This enables the user to ensure that oldProgNetTime has been written, that is, if the user cancels the current program with reset, oldProgNetTime and oldProgNetTimeCounter remain unchanged. Note: Two programs running consecutively can have identical runtimes and be correctly terminated. The user can then only detect this by the changed oldProgNetTimeCounter. Modification counter					
-	0	0		UWord	r
Multi-line: yes	1		1		

pCutInv	$AC_CUT_INV				
States that a turning tool is rotated against the machining plane (typically through 180 degrees around the C axis with G18) such that the direction of spindle rotation has to be inverted. FALSE, TRUE					
-	0	0	1	UWord	r
Multi-line: yes	1		1		

pCutInvS					
States that a turning tool is rotated against the machining plane (typically through 180 degrees around the C axis with G18) such that the direction of spindle rotation has to be inverted. For block search. FALSE, TRUE					
-	0	0	1	UWord	r
Multi-line: yes	1		1		

pCutMod	$AC_CUTMOD				
Reads the current valid value that was last programmed with the language command CUTMOD (number of the tool carrier for which the cutting edge data modification is to be activated). If the last programmed value was CUTMOD = -2 (activation with the currently active tool carrier with orientation capability), the value -2 is not returned but the number of the active tool carrier with orientation capability at the time of programming. -2, 999999					
-	0	0	1	Long Integer	r
Multi-line: yes	1		1		

pCutModK	$AC_CUTMODK				
Reads the currently valid value last programmed with the language command CUTMODK. (Name of the orientation transformation defined by kinematic chains for which the edge data modification is to be activated).					
-	"\0"			String [32]	r
Multi-line: yes	1		1		

pCutModKA	$AC_CUTMODKA				
Cutter position modification for a transformation defined with kinematic chains is active.					
-	0	0	1	UWord	r
Multi-line: yes	1		1		

pCutModKAS	$P_CUTMODKA				
Cutter position modification for a transformation defined with kinematic chains is active.					
-	0	0	1	UWord	r
Multi-line: yes	1		1		

pCutModKS					
Reads the currently valid value last programmed with the language command CUTMODK. (Name of the orientation transformation defined by kinematic chains for which the edge data modification is to be activated).					
-	"\0"			String [32]	r
Multi-line: yes	1		1		

pCutModS					
Reads the current valid value that was last programmed with the language command CUTMOD (number of the tool carrier for which the cutting edge data modification is to be activated). If the last programmed value was CUTMOD = -2 (activation with the currently active tool carrier with orientation capability), the value -2 is not returned but the number of the active tool carrier with orientation capability at the time of programming. For block search -2, 999999					
-	0	0	1	Long Integer	r
Multi-line: yes	1		1		

pEgBc	$P_EG_BC[a]				
Electronic gear: Block change criterion. Important for EGON, EGONSYN 0: NOC Block change is performed immediately 1: IPOSTOP Block change is performed with setpoint side synchronism 2: COARSE Block change is performed with "Synchronism coarse" 3: FINE Block change is performed with "Synchronism fine"					
-	3	0	3	UWord	r
Multi-line: yes	(Axis index of slave axis + 1)		numMachAxes		

pMthSDC	$P_MTHSDC				
Master tool holder no. or master spindle no. is determined with reference to the active tool for the next D offset selection. This is important if the master spindle changes after the last tool change. >0 Successful read access 0 No master tool holder or no master spindle available. The next D offset works with T0. -1 TMMG not available					
-	0	0	numMachAxes	Long Integer	r
Multi-line: no			1		

pOffn	$P_OFFN				
Last programmed offset normal					
-	0			Double	r
Multi-line: no					

pOriDiff0	$P_ORI_DIFF[0,n]				
The angle difference between the exact angles and those available in $P_ORI_ANG for the orientation axes of the first (or single) solution during orientation programming. The content of these variables can only be unequal to zero if the positions of the orientation axes are incremental (Hirth coupling).					
Degree	0			Double	r
Multi-line: yes	IndOriAchs		2		

pOriDiff1	$P_ORI_DIFF[1,n]				
The angle difference between the exact angles and those available in $P_ORI_ANG for the orientation axes of the second solution during orientation programming. The content of these variables can only be unequal to zero if the positions of the orientation axes are incremental (Hirth coupling).					
Degree	0			Double	r
Multi-line: yes	IndOriAchs		2		

pOriPos0	$P_ORI_POS[0,n]				
The angle of the orientation axes of the first (or single) solution during orientation programming.					
Degree	0			Double	r
Multi-line: yes	IndOriAchs		2		

pOriPos1	$P_ORI_POS[1,n]				
The angle of the orientation axes of the second solution during orientation programming.					
Degree	0			Double	r
Multi-line: yes	IndOriAchs		2		

pOriSol	$P_ORI_SOL				
Contains the number of solutions and additional status information for orientation programming. See also the documentation for the corresponding system variables.					
-	0			Long Integer	r
Multi-line: no					

pOriStat	$P_ORI_STAT				
Contains the status of the orientation axes during orientation programming. See also the documentation for the corresponding system variables.					
-	0			Long Integer	r
Multi-line: yes	IndOriAchs		2		

pTCutMod	$P_AD[2]				
Angle of rotation for modification of edge position and cutting direction Angle between 0 and 360 degrees					
Degree	0	0	360	Double	r
Multi-line: yes	1		1		

pTCutModS					
Angle of rotation for edge position and cutting direction for block search Angle between 0 and 360 degrees					
Degree	0	0	360	Double	r
Multi-line: yes	1		1		

pTc	$P_TC				
The active orientatable toolholder					
-	0	0		UWord	r
Multi-line: yes	1		1		

pTcAng	$P_TCANG[n]				
The current angles of the two axes of an orientation-capable toolholder					
Degree	0			Double	r
Multi-line: yes	Axis no. of toolholder		2		

pTcDiff	$P_TCDIFF[n]				
The difference between the exact and the actually used angles of the two axes of an orientation-capable toolholder					
Degree	0			Double	r
Multi-line: yes	Axis no. of toolholder		2		

pTcNum	$P_TCNUM				
Number of available orientable tool carriers in the channel					
-	0	0		UWord	r
Multi-line: yes	1		1		

pTcSol	$P_TCSOL				
Number of solutions (configuration options for rotary axes) on selection of an orientatable toolholder. The variable value can be between 0 and 2, where 0 to 2 means either none, 1 solution or 2 solutions.					
-	0	0		UWord	r
Multi-line: yes	1		1		

pTcStat	$P_TCSTAT				
Specifies the status of an orientable tool carrier. The variable is bit-coded with the following meanings: 0x0001 The first rotary axis is available 0x0002 The second rotary axis is available 0x0004 The angles used for the calculation come from an orientation in the frame direction 0x0008 The angles used for the calculation have been absolutely defined 0x0010 The polar axis angle is not defined in the case of orientation in the frame direction 0x1000 Only the tool can be rotated (kinematic type T) 0x2000 Only the workpiece can be rotated (kinematic type P) 0x4000 Tool and workpiece can be rotated (kinematic type M) The bits stated here are not currently assigned.					
-	0	0		Long Integer	r
Multi-line: yes	1		1		

pToolO	$P_TOOL_O				
Supplies the current tool orientation in various coordinate systems: Possible values of the line index: 1, 2, 3: Components of the vector in BCS 4, 5, 6: Components of the vector in PCS/WCS 7, 8, 9: Components of the vector in ENS The orientation vector is normalized, i.e. it has the value 1.					
-	0	-1	1	Double	r
Multi-line: yes	1: X component		9		

pToolRot	$P_TOOL_O_R				
Current tool rotation in various coordinate systems: Possible values of the line index: 1, 2, 3: Components of the vector in BCS 4, 5, 6: Components of the vector in PCS/WCS 7, 8, 9: Components of the vector in ENS The rotation vector is normalized, i.e. it has the absolute value 1.					
-	0	-1	1	Double	r
Multi-line: yes	1: X component		9		

paAccLimA	$PA_ACCLIMA[a]				
Axial acceleration override in run in 1-200					
-	100	1	200	UWord	r
Multi-line: yes	(Axis index)		numMachAxes		

paJerkLimA	$PA_JERKLIMA[a]				
Axial jerk override in run in 1-200					
-	100	1	200	UWord	r
Multi-line: yes	(Axis index)		numMachAxes		

paVeloLimA	$PA_VELOLIMA[a]				
Axial velocity override in run in 1-200					
-	100	1	200	UWord	r
Multi-line: yes	(Axis index)		numMachAxes		

pcTrafoRotChainIndex	$PC_TRAFO_ROT_CHAIN_INDEX				
Images the indices of the orientation axes in the array $NT_ROT_AX_NAME onto the internal orientation axis sequence. See also documentation of the associated system variables.					
-	0			Long Integer	r
Multi-line: yes	IndOriAchs		2		

pcTrafoRotChanAxEx	$PC_TRAFO_ROT_CHAN_AX_EX				
Determines the channel axis index of the i-th orientation axis, where i is the index of this axis in the external representation of the kinematic chain (index of the entry in the array$NT_ROT_AX_NAME[n, i]). See also documentation of the associated system variables.					
-	0			UWord	r
Multi-line: yes	IndOriAchs		2		

pcTrafoRotChanAxIn	$PC_TRAFO_ROT_CHAN_AX_IN				
Determines the channel axis index of the i-th orientation axis, where i is the index of this axis in the internal representation of the kinematic chain. See also documentation of the associated system variables.					
-	0			UWord	r
Multi-line: yes	IndOriAchs		2		

progDuploNumber					
Duplo number of programmed tool (does not yet have to be active)					
-	0			UWord	r
Multi-line: no			1		

progEvent					
Active prog. events The data is bit-coded, so that when required, also individual states can be masked or evaluated separately (bits that are not listed supply a value of 0) Bit0 = 1: start Bit1 = 1: M30 Bit2 = 1: Reset Bit3 = 1: Power-On Bit4 = 1: Search Bit5 = 1: Safety					
-				UWord	r
Multi-line: no					

progNetTimeTrigger	$AC_PROG_NET_TIME_TRIGGER				
Serves for the selective measurement of program sections, that is the time measurement can be switched on and off again by the program by writing progNetTimeTrigger. Certain values of progNetTimeTrigger are given a special function in order to fully exploit all trigger options: 0 Neutral: The trigger is not active, the value is taken from reset with the start key. 1 Terminate: Terminates the measurement and copies actProgNetTime -> oldProgNetTime. actProgNetTime is set to zero and then runs on again. 2 Start: Starts the measurement and sets actProgNetTime to zero. oldProgNetTime remains unchanged. 3 Stop: Stops the measurement. Does not change oldProgNetTime and holds actProgNetTime constant until resume. 4 Resume: Resumption of the measurement, that is a previously stopped measurement is resumed. actProgNetTime runs on. oldProgNetTime remains unchanged.					
-	0	0	4	UWord	r
Multi-line: yes	1		1		

progStatus	DB21-28, DBX35.0 - DBX35.4				K1
Program status 1 = interrupted 2 = stopped 3 = in progress 4 = waiting 5 = aborted					
-				UWord	r
Multi-line: no					

progTNumber					
Number of programmed tool					
-				UWord	r
Multi-line: no					

progTNumberLong					
Number of the programmed tool using flat D-numbers with up to 8 digits					
-	0			Long Integer	r
Multi-line: yes	1		1		

progToolIdent					
Identifier of programmed tool (does not yet have to be active)					
-	"\0"			String [32]	r
Multi-line: no			1		

progUsekt					
Programmed value of the command $P_USEKT. Bit-coded data for the programmed tool subgroup available for the tool change.					
-	0	0	0xF	Long Integer	r
Multi-line: no					

progWaitForEditUnlock					
The variable is used for two possible applications: 1. Notification to HMI that an NC program is to be processed for which a program execution delay was activated by means of the PI service _N_F_MODE. This is only possible for files of the passive file system of the NCK. 2. Notification to HMI that an NC program is to be processed for which a WRITE lock was set. This is only possible for files that are available on the CF, network drive or USB device and are processed in EES mode. The variable contains the complete path name in both cases.					
-	0			String [160]	r
Multi-line: yes	1		1		

protAreaCounter					
Counter is incremented by 1 every time a protection zone (block PA) is modified					
-				UWord	r
Multi-line: yes	1		1		

protocHmiEvent					
Logging: When writing, the defined event is activated during preprocessing. 49: HMI_TRIG_1 50: HMI_TRIG_2 51: HMI_TRIG_3					
-		0		UWord	rw
Multi-line: yes	1		1		

protocUserActive	$MM_PROTOC_USER_ACTIVE				
Logging: Displays active users 0: User inactive 1: User active					
-	0	0	1	UWord	r
Multi-line: yes	User No. (1-10)		10		

rapFeedRateOvr					
Rapid traverse override					
%				Double	r
Multi-line: no					

remainDwellTime					
Remaining dwell time See timeOrRevolDwell					
s, user defined	0	0		Double	r
Multi-line: yes	1		1		

reqParts	$AC_REQUIRED_PARTS				
Number of required workpieces (workpiece requirement): The workpiece count at which the number of current workpieces $AC_ACTUAL_PARTS is set to zero can be defined in this counter.					
-	0			Double	rw
Multi-line: no					

retractState					
Status information on the submode JOG retract Bit 0: 0: No retract data available; JOG retract cannot be activated 1: Retract data available; JOG retract can be activated with PI _N_RETRAC Bit 1: 0: JOG retract submode is not active 1: JOG retract submode is active Bit 3/2: 0: Function is not active 1: Retraction axis is 1st geometry axis according to $MC_AXCONF_GEOAX_ASSIGN_TAB 2: Retraction axis is 2nd geometry axis according to $MC_AXCONF_GEOAX_ASSIGN_TAB 3: Retraction axis is 3rd geometry axis according to $MC_AXCONF_GEOAX_ASSIGN_TAB Bit 4/5: 0: Default tool (milling cutter) 1: Tap (tapping with G33/G331/G332 is active) 2: Drilling tool from the 200 group Bit 6/7: Reserved Bit 8/11: 0: No problem known 1: No tool selected 2: No retract tool selected (turning or grinding tool) 3: Tool offset is not active 4: Retraction axis does not exist 5: G63 block 6: Path motion not in direction of tool axis 7: Retract data not persistent due to $MN_MM_ACTFILESYS_LOG_FILE_MEM[2]=0 8: Retract data are not consistent Bit 12/15: Reserved					
-	0			UWord	r
Multi-line: yes	1		1		

rotSys	$AC_ROT_SYS				
Reference system for orientation movements with cartesian manual traversal 0: Axis-specific manual traversal active 1: Cartesian manual traversal in basic coordinate system active 2: Cartesian manual traversal in workpiece coordinate system active 3: Cartesian manual traversal in tool coordinate system active					
-	0	0	3	UWord	r
Multi-line: yes	1		1		

searchRunMode					
Type of function in which search run has been integrated 1: Search run is used directly 2: Simulation search run 3: Execute program area The user can preselect a program area via HMI with "Execute program area", which he/she would like to execute "real" on the machine. To do this, the NCK uses an internal block search to approach the start of the program area (abbreviation: APb) correctly. Internal cancellation at the end of the program area (abbreviation: EPb) via reset. 0: Otherwise					
-	0	1	3	UWord	r
Multi-line: yes	Axis index of the following axis		numMachAxes		

searchRunStatus					
Status of the search run 1: activeSearchRun Simulation active, that is the NCK simulates the part program from the start to the agreed search target (or APb) in order to find, among other things, the correct starting position of the search target block. 2: targetFound The search target has been found and the NCK is waiting for the Start button. Simulation has finished. 3: activeAdaption After the start, the NCK outputs action blocks which set the machine to the search target (M± function output, spindle speeds) and, if applicable, starts an ASUB in which the user uses the ASUB program to adapt the machine to the part program situation in the target block. (For example, a programmed tool is read and a tool-changing cycle exchanges it with the current tool.) The NCK stops automatically after the action blocks or after the ASUB with alarm 10208. 4: finishedAdaption The NCK waits for the start. 5: activeStopRun After the adaptation, the REPOS function goes to the target block, and then the execution of the program is resumed. The NCK executes the program area after the target block, but is still within the function Execute program area. The blocks are scanned to see whether the end of the program area (EPb) could already have been reached. The program is cancelled at EPb with reset and searchRunStatus is cleared. 0: Otherwise					
-	0	1	5	UWord	r
Multi-line: yes	1		1		

seruproMasterChanNo					
The search type SERUPRO (search via program testing) may be started simultaneously in several channels in order to start a channel grouping correctly. A search target must be specified in one channel (master channel) in the grouping. The other channels do not need a search target, they wait until they have reached a stop condition and the master channel has reached the search target. These channels generally stop at WAIT marks. The variable seruproMasterChanNo defines the master channel.					
-	0	0	numChannels	UWord	rw
Multi-line: yes	1		1		

seruproMasterNcuNo						
The search type SERUPRO (search via program testing) may be started simultaneously in several channels in order to start a channel grouping correctly. A search target must be specified in one channel (master channel) in the grouping. The other channels do not need a search target, they wait until they have reached a stop condition and the master channel has reached the search target. These channels generally stop at WAIT marks. The variable seruproMasterChanNo defines the master channel. seruproMasterNcuNo specifies the master channel in more detail if it is not on the active NCU.						
-	0	0	$MN_MM_LINK_NUM_OF_MODULES	UWord		rw
Multi-line: yes	1		1			

simTolerance	keine					
The NCK simulation can run part programs at a higher speed (see PI _N_NCKMOD). Only then is simTolerance evaluated and it ONLY affects geometry blocks that have not been programmed as circles or straight lines. These blocks are slowed down so that two successive interpolation points can be connected by a straight line. This straight line deviates from the programmed contour by not more than 'simTolerance'.						
mm, inch, user defined	0			Double		rw
Multi-line: no						

simulationSupport						
Block information for the support of the JobShop simulation Bit0: Change of the transformation in the curr. block Bit1: Change of the frame in the curr. block Bit2: Curr. block in an action block Bit3: Curr. block in the last action block Bit4: Curr. block has PTP active (as of 510600)						
-	0	0		UWord		r
Multi-line: no			1			

simulationSupportS						
Block information for the support of the JobShop simulation during search Bit0: - Bit1: - Bit2: - Bit3: - Bit4: Curr. block has PTP active						
-	0	0		UWord		r
Multi-line: no			1			

specParts	$AC_SPECIAL_PARTS				
Number of current workpieces as defined by user: This counter enables the user to define his own workpiece count. The counter is reset to zero automatically only when the control system boots on defaults.					
-	0			Double	rw
Multi-line: no					

splitBlock	$AC_SPLITBLOCK				
Identifier of internally splitted blocks 0: A BLOCK programmed unchanged (a BLOCK generated by the compressor is regarded as programmed BLOCK): <>0: BLOCK was shortened or is an internally generated BLOCK; the following values are possible: 1: It is an internally generated BLOCK or a shortened original BLOCK 3: It is the last block in a chain of internally generated blocks or shortened original blocks.					
-	0	0	2	Long Integer	r
Multi-line: yes	1		1		

startLockCounter					
Counter that is incremented as soon as an NC start is activated with a set channel-specific start disable (see _N_STRTLK).					
-	0	0		UWord	r
Multi-line: yes	1		1		

startLockState					
Status of the global start disable. Also see PI _N_STRTLK and _N_STRTUL. 0: No start disable 1: Start disable is switched on and program is not running 2: Start disable is switched on and program is running nevertheless The NCK changes from 2->1 as soon as the program is stopped.					
-	0	0	2	UWord	r
Multi-line: yes	1		1		

startRejectCounter					
Counter that is incremented as soon as an NC start is rejected due to a global start disable (see _N_STRTLK), program-specific start disable (see _N_F_STLO), or channel-specific start disable (see _N_STRTLK).					
-	0	0		UWord	r
Multi-line: yes	1		1		

stopCond					
Replaced by stopCondNew					
-	0	0		UWord	r
Multi-line: yes	1		1		

stopCondChangeCounter					
Modification counter for stop states. Is incremented as soon as one of the stop states has changed.					
-				UWord	r
Multi-line: yes	1		1		

stopCondNew					
Number of the NC stop state More than one stop state can be active simultaneously. The highest priority stop state appears below the first line, this is followed by those with lower priorities. The documentation explains the meanings of the individual stop states.					
-	0	0		UWord	r
Multi-line: yes	Number of the active stop state		stopCondNum		

stopCondNum					
Number of active stop states. Specifies the number of occupied lines in stopCond					
-				UWord	r
Multi-line: yes	1		1		

stopCondPar					
Replaced by stopCondParNew					
-				UWord	r
Multi-line: yes	1				

stopCondParA					
Stop state parameter(s). More than one stop state can be active simultaneously. The highest priority stop state appears below the first line, this is followed by those with lower priorities.					
-				String [32]	r
Multi-line: yes	High byte: No. of the active stop state Low byte: No. of the parameter				

stopCondParNew					
Stop state parameter(s). More than one stop state can be active simultaneously. The highest priority stop state appears below the first line, this is followed by those with lower priorities.					
-				UWord	r
Multi-line: yes	High byte: No. of the active stop state Low byte: No. of the parameter				

stopCondTime					
BCD time stamp for stop state. More than one stop state can be active simultaneously. The highest priority stop state appears below the first line, this is followed by those with lower priorities.					
-				Date+Time	r
Multi-line: yes	Number of the stop state		stopCondNum		

stopRunActive					
Stop run active 0 = inactive 1 = active					
-	0	0	1	UWord	r
Multi-line: yes	1		1		

stopRunCounter					
Modification counter for stop run. This is always incremented when the NCK has stopped at a stop block.					
-	0	0		UWord	r
Multi-line: yes	1		1		

suppProgFunc					
Disabling of language commands Bit0 = 0: SBLOF command is active Bit0 = 1: SBLOF command is disabled					
-	Bit0 = 0			UWord	rw
Multi-line: yes	1		1		

syntaxCheckAlarmNo					
Alarm number in the case of a syntax error during the syntax check					
-	0	0		Long Integer	r
Multi-line: yes	1		1		

syntaxCheckAlarmPara1					
Parameter 1 for an alarm in the case of a syntax error during the syntax check					
-	0	0		String [32]	r
Multi-line: yes	1		1		

syntaxCheckAlarmPara2					
Parameter 2 for an alarm in the case of a syntax error during the syntax check					
-	0	0		String [32]	r
Multi-line: yes	1		1		

syntaxCheckAlarmPara3					
Parameter 3 for an alarm in the case of a syntax error during the syntax check					
-	0	0		String [32]	r
Multi-line: yes	1		1		

syntaxCheckAlarmPara4					
Parameter 4 for an alarm in the case of a syntax error during the syntax check					
-	0	0		String [32]	r
Multi-line: yes	1		1		

syntaxCheckSeek					
Line number of the faulty line in the syntax check					
-	0	0		Long Integer	r
Multi-line: yes	1		1		

syntaxCheckStatus					
Status of the "Syntax check" function 0: Syntax check not active (initialization status) 1: Syntax check selected 2: Syntax check active 3: Syntax check stopped with alarm on account of system error 4: Syntax check terminated 5: Syntax check canceled 6: Syntax check canceled on account of errors					
-	0	0	6	UWord	r
Multi-line: yes	1		1		

tOffL1L2L3	$AC_TOFFL				
Tool length offset programmed in coordinates of the tool length components L1/L2/L3.					
mm, inch, degree, user defined	0			Double	r
Multi-line: yes	1: Tool length offset programmed in the tool length component direction L1 2: Tool length offset programmed in the tool length component direction L2 3: Tool length offset programmed in the tool length component direction L3		3		

tOffLXYZ	$AC_TOFF				
Tool length offset TOFFL programmed in coordinates of the WCS.					
mm, inch, degree, user defined	0			Double	r
Multi-line: yes	1: Tool length offset programmed in the direction of the 1st geometry axis 2: Tool length offset programmed in the direction of the 2nd geometry axis 3: Tool length offset programmed in the direction of the 3rd geometry axis		3		

tOffR	$AC_TOFFR				
Programmed tool radius offset.					
mm, inch, degree, user defined	0			Double	r
Multi-line: yes	1		1		

threadPitch					
Current lead					
-	0	0		Double	r
Multi-line: yes	1		1		

threadPitchS					
Current lead during search run					
-	0	0		Double	r
Multi-line: yes	1		1		

timeOrRevolDwell					
Dwell time unit in seconds or spindle revolutions 0: cmdDwellTime and remainDwellTime in seconds 1: cmdDwellTime and remainDwellTime in spindle revolutions					
-	0	0	1	UWord	r
Multi-line: yes	1		1		

timeS	$AC_TIMES				
Time after a block change between programmed blocks in seconds Each programmed block can be divided up into a chain of part blocks that are processed one after the other. O n l y with the 1st cycle of the 1st block of the chain, timeS is set to zero and then counted up in seconds. Therefore, the variable enables time measurements throughout the entire block chain.					
s, user defined	0	0		Double	r
Multi-line: yes	1		1		

timeSC	$AC_TIMESC				
Time after a block change between programmed blocks in IPO cycles Each programmed block can be divided up into a chain of part blocks that are processed one after the other. Only (!) with the 1st cycle of the 1st block of the chain, timeSC is set to zero and then counted up in seconds. Therefore, the variable enables time measurements throughout the entire block chain.					
-	0	0		Double	r
Multi-line: yes	1		1		

toolCounter					
Counter of the changes of the tool data assigned to a channel. The counter is incremented each time a tool data is changed. All changes of tool data made by BTSS, part programs, INI files and by the Tool Management software are considered. Tool data are tool compensations, grinding-specific tool parameters, OEM tool parameters and Tool Managment data including magazine data. There is one exception: the present tool-in-use-time, since it is changed in each IPO cycle.					
-				UWord	r
Multi-line: yes	1		1		

toolCounterC					
Counter for modifications to tool offset data assigned to the channel (analog toolCounter).					
-				UWord	r
Multi-line: yes	1		1		

toolCounterIso	keine				
Each change of a tool offset value for ISO2.2 and ISO3.2 modes is counted. This is to enable the HMI to record data changes.					
-	0			Long Integer	r
Multi-line: yes	1		1		

toolCounterM					
Counter for modifications to magazine data assigned to the channel (analog toolCounter).					
-				UWord	r
Multi-line: yes	1		1		

toolFrameState					
toolFrameState provides bit-coded information about whether the PI service _N_SETUDT with the function designations 12 and 13 can be activated in its current state and also specifies any parameters that may be required: Bit 0 provides information about whether the NCK can generate a tool frame with PI service _N_SETUDT and function designation 12 in its current state. Once the bit is set, the NCK receives information about the current tool orientation, i.e. either a tool holder that can be oriented or an orientation transformation is active and the tool frame can be generated. Bit 1 provides information about whether, in its current state, the NCK has stored data for the restoration of the program environment (bit 1 = 1), which can be restored using PI service _N_SETUDT and function designation 13. When bit 0 is set, bit 2 provides information about whether the tool axis corresponds to a geometry axis of the current WCS: Bit 2 = 0: The tool axis does not correspond to any geometry axes of the current WCS. In this case, bit 3 / bit 4 provide the number of the geometry axis next to the tool axis. Bit 5 provides the retraction direction of this axis (plus/minus). This information can be displayed by the HMI as a suggestion or default setting for the retraction axis. Bit 2 = 1: The tool axis corresponds to a geometry axis of the current work. In this case bit 3 / bit 4 provide the number of this geometry axis, and bit 5 provides its retraction direction (plus/minus). Bit 0: 0: PI service _N_SETUDT with function designation 12 disabled 1: PI service _N_SETUDT with function designation 12 enabled Bit 1: 0: PI service _N_SETUDT with function designation 13 disabled 1: PI service _N_SETUDT with function designation 13 enabled Bit 2 0: Tool axis does not correspond to any geometry axes 1: Tool axis corresponds to a geometry axis Bit 3 / Bit 4:0: Function not active 1: Tool axis in the 1st geometry axis 2: Tool axis in the 2nd geometry axis 3: Tool axis in the 3rd geometry axis Bit 5: 0: Retraction direction plus 1: Retraction direction minus					
-	0	0	63	UWord	r
Multi-line: yes	1		1		

<table>
<tr><th>toolHolderData</th><th colspan="4">GETSELT, GETEXET</th><th></th></tr>
<tr><td colspan="6">Data for each tool holder/spindle from the magazine configuration of the TOA which is assigned to the channel.
There is a set of numToolHolderParams parameters for each tool holder.
Currently there are the 3 parameters P1, P2 and P3.
There are numToolHolders tool holders. The number of tool holders in this list is solely a function of the
magazine configuration, and it does not change while an NC program runs.
- P1: THNo ToolHolderNumber / SpindleNumber
(In the language commands of the NC program, corresponds
to the address extension <n> from T<n>=... or M<n>=6 with explicit
notation; in the magazine configuration, corresponds to the
location type index of the associated buffer location of the
location type = spindle.)
- P2: SelTno T number of the selected tool with
reference to the tool holder / spindle with the number of THNo
(The same TNo would also return the language command GETSELT.)
The value 0 indicates that no tool is selected with
reference to the tool holder. For further behavior see the description of GETSELT.
- P3: ExeTno TNumber of the tool to be loaded / loaded with
reference to the tool holder / the spindle with the number THNo from
the point of view of the NC program.
When working without M6, the same TNumber is in SelTno and ExeTno.
(The same TNumber would also return the language command GETEXET.)
The value 0 indicates that no tool is to be loaded / is loaded
with reference to the tool holder. For further behavior see the description of GETEXET.
- P4: SelTNoBeforeSearchRun
During the search: TNummer of the selected tool with reference to the tool holder / spindle before the search,
after reaching the search target and a T programming for this tool holder: the same value from P2.
- P5: ExeTNoBeforeSearchRun
During the search: TNummer of the tool loaded with reference to the tool holder / spindle before the search,
after reaching the search target and a tool change for this tool holder: the same value from P3.
An array access is possible to toolHolderData, with which the data of all numToolHolders tool holders can be read at one time.
If the flat D number is active, the value =0 is returned for all parameters.</td></tr>
<tr><td>-</td><td>0</td><td>0</td><td></td><td>Double</td><td>r</td></tr>
<tr><td>Multi-line: yes</td><td colspan="2">The line index addresses the parameters of the tool holder and the tool holder itself:
Line index = (ElementNo - 1) * numToolHolderParams + PNo
With: ElementNo 1 to numToolHolders;
The ElementNo is the list element no of the tool holder in this list.
PNo: Parameter number from 1 to numToolHolderParams
numToolHolderParams from range N, block Y, global system data</td><td colspan="3">numToolHolderParams * numToolHolders</td></tr>
</table>

toolholderOfDNo	$P_TH_OF_D				
Number of the tool holder or spindle on which the active tool is mounted, which contains the active D no.					
-	0	0		UWord	r
Multi-line: no			1		

totalParts	$AC_TOTAL_PARTS				
Total number of all machined workpieces: This counter specifies the number of workpieces machined since it was started. The counter is automatically set to zero only if the control system boots on defaults.					
-	0			Double	rw
Multi-line: no					

transSys	$AC_TRANS_SYS				
Reference system for translation with cartesian manual traversal 0: Axis-specific manual traversal active 1: Cartesian manual traversal in basic coordinate system active 2: Cartesian manual traversal in workpiece coordinate system active 3: Cartesian manual traversal in tool coordinate system active					
-	0	0	3	UWord	r
Multi-line: yes	1		1		

transfActive	DB21-28, DBX33.6				K1, M1
Transformation active 0 = inactive 1 = active					
-				UWord	r
Multi-line: no					

vaCcCompVal	$VA_CC_COMP_VAL[a,b]				
OA compensation value of the corresponding compile cycle					
mm, inch, degree, user defined	0			Double	r
Multi-line: yes	Low byte = axis number, high byte = index of compile cycle		numMachAxes		

vaEgSyncDiff	$VA_EG_SYNCDIFF[a]				
Electronic gear: Synchronism deviation (actual values). The comparison between this value and $MA_COUPLE_POS_TOL_... determines whether the appropriate "Synchronism" VDI signal is set.					
mm, inch, degree, user defined	0			Double	r
Multi-line: yes	(Axis index of slave axis + 1)		numMachAxes		

vaEgSyncDiffS	$VA_EG_SYNCDIFF_S[a]				
Electronic gear: Synchronous run difference (actual values) with sign. Whether the corresponding VDI signal "synchronous run" is set depends upon the comparison of this value with $MA_COUPLE_POS_TOL_....					
mm, inch, degree, user defined	0			Double	r
Multi-line: yes	(Axis index of the following axis)		numMachAxes		

vaSyncDiff	$VA_SYNCDIFF[]				
Actual value synchronism difference for all types of coupling					
mm, inch, degree, user defined	0			Double	r
Multi-line: yes	Axis index of the following axis		numMachAxes		

vaSyncDiffStat	$VA_SYNCDIFF_STAT[]				
Status of the actual value synchronism difference -4: Reserved -3: No valid value in $VA_SYNCDIFF, tangential control -2: No valid value in $VA_SYNCDIFF, master value coupling and simulated master value -1: No valid value in $VA_SYNCDIFF 0: No valid value in $VA_SYNCDIFF, coupling not active 1: Valid value in $VA_SYNCDIFF					
-	0	-4	1	Long Integer	r
Multi-line: yes	Axis index of the following axis		numMachAxes		

vcToolO	$VC_TOOL_O				
Supplies the actual value of the current tool orientation in various coordinate systems: Possible values of the line index: 1, 2, 3: Components of the vector in BCS 4, 5, 6: Components of the vector in PCS/WCS 7, 8, 9: Components of the vector in ENS The orientation vector is normalized, i.e. it has the value 1.					
-	0	-1	1	Double	r
Multi-line: yes	1: X component		9		

vcToolODiff	$VC_TOOL_O_DIFF				
Supplies the angle between the setpoint vector and actual vector of the tool orientation in various coordinate systems: Possible values of the line index: 1: Angle in BCS 2: Angle in PCS/WCS 3: Angle in ENS					
-	0	0	180	Double	r
Multi-line: yes	1		3		

vcToolOStat	$VC_TOOLO_STAT				
Supplies the status of the computation of the actual orientation					
-	0	-1	0	Long Integer	r
Multi-line: no					

vcToolR	$VC_TOOL_R				
Actual value of the tool rotation in various coordinate systems: Possible values of the line index: 1, 2, 3: Components of the vector in BCS 4, 5, 6: Components of the vector in PCS/WCS 7, 8, 9: Components of the vector in ENS The rotation vector is normalized, i.e. it has the value 1.					
-	0	-1	1	Double	r
Multi-line: yes	1: X component		9		

vcToolRDiff	$VC_TOOL_R_DIFF				
Angle between setpoint and actual tool rotation vectors in different coordinate systems: Possible values of the line index: 1: Angle in BCS 2: Angle in PCS/WCS 3: Angle in ENS					
Degree	0	0	180	Double	r
Multi-line: yes	1		3		

vcToolRStat	$VC_TOOLR_STAT				
Status of the computation of the actual rotation					
-	0	-1	0	Long Integer	r
Multi-line: yes	1		1		

workPnameSubstitution					
Path name defined by HMI on selection of a program or workpiece in /_N_EXT_DIR. It is only used for execute from external, and then denotes the data source of the download. HMI uses this path name in order to restore the program selection for execute from external after PowerOff. The string must be terminated with "\0". NCK does not use this path name. workPnameSubstitution is deleted on program selection in the passive file system or on an EES drive. Handling: workPnameSubstitution is written by the HMI on program selection for execute from external. NCK stores this information persistently. After PowerOff, the NCK deletes the reload buffer for execute from external, and selects _N_MPF0. HMI restores on the basis of the information: - _N_MPF0 is selected - workPnameSubstitution is set the selection for execute from external. With this program selection, the NCK does not remove the SPARPI interrupt pointer.					
-	0	0		String [128]	r
Multi-line: no					

3.4.3 Area C, Block SINF : Part-program-specific status data

OEM-MMC: Linkitem /ChannelProgramModification/...

During automatic execution of a part program different parameters can influence the type of machining. The current status data for the selected part program are combined in module SINF. The status data must only be changed via the PLC. interface.

DRFActive					
DRF active 0 = not active 1 = active					
-				UWord	r
Multi-line: no					

feedStopActive					
Feed disable 0 = inactive 1 = active					
-				UWord	r
Multi-line: no					

ipoBlocksOnly					
Display traversing blocks 0 = normal block transfer 1 = exclusively traversing blocks					
-				UWord	r
Multi-line: no					

optAssStopActive					
Associated M01 selected 0: Not selected 1: Selected					
-	0			UWord	r
Multi-line: yes	1		1		

optStopActive					
M01 selected 0 = not selected 1 = selected					
-				UWord	r
Multi-line: no					

progTestActive	DB21-28, DBX1.7				K1
Program test 0 = inactive 1 = active					
-				UWord	r
Multi-line: no					

rapFeedRateOvrActive					
ROV rapid traverse override 0 = inactive 1 = active					
-				UWord	r
Multi-line: no					

singleBlockActive					
Single block, SBL 0 = no single block 1 = SBL 1 2 = SBL 2					
-				UWord	r
Multi-line: no					

singleBlockType					
Single block mode 1 = interpolation single block 2 = decoder single block					
-				UWord	rw
Multi-line: no					

skipLevel0Active					
Info whether skip level /0 is activated. 0: Skip level /0 not active 1: Skip level /0 active					
-	0	0	1	UWord	r
Multi-line: no					

skipLevel1Active					
Info whether skip level /1 is activated 0: Skip level /1 not active 1: Skip level /1 active					
-	0	0	1	UWord	r
Multi-line: no					

skipLevel2Active					
Info whether skip level /2 is activated 0: Skip level /2 not active 1: Skip level /2 active					
-	0	0	1	UWord	r
Multi-line: no					

skipLevel3Active					
Info whether skip level /3 is activated 0: Skip level /3 not active 1: Skip level /3 active					
-	0	0	1	UWord	r
Multi-line: no					

skipLevel4Active					
Info whether skip level /4 is activated 0: Skip level /4 not active 1: Skip level /4 active					
-	0	0	1	UWord	r
Multi-line: no					

skipLevel5Active					
Info whether skip level /5 is activated 0: Skip level /5 not active 1: Skip level /5 active					
-	0	0	1	UWord	r
Multi-line: no					

skipLevel6Active					
Info whether skip level /6 is activated. 0: Skip level /6 not active 1: Skip level /6 active					
-	0	0	1	UWord	r
Multi-line: no					

skipLevel7Active					
Info whether skip level /7 is activated. 0: Skip level /7 not active 1: Skip level /7 active					
-	0	0	1	UWord	r
Multi-line: no					

skipLevel8Active					
Info whether skip level /8 is activated. 0: Skip level /8 not active 1: Skip level /8 active					
-	0	0	1	UWord	r
Multi-line: no					

skipLevel9Active					
Info whether skip level /9 is activated. 0: Skip level /9 not active 1: Skip level /9 active					
-	0	0	1	UWord	r
Multi-line: no					

trialRunActive	DB21-28, DBX0.6				V1
Dry run feedrate 0 = inactive 1 = active					
-				UWord	r
Multi-line: no					

3.4.4 Area C, Block SPARP : Part program information

OEM-MMC: Linkitem /ChannelProgramInfo/...

This module contains information on the currently active part programm in the respective channel.

absoluteBlockBufferName					
File name with path of upload buffer in which display blocks are stored Empty string: Function is deactivated					
-				String [128]	r
Multi-line: yes	1		1		

absoluteBlockBufferPreview					
Part of content of file absoluteBlockBufferName. The desired content of the variables is set by $MC_MM_ABSBLOCK_BUFFER_CONF. In principle, only complete parts program blocks are entered. If the desired number of previous blocks are not present, then an empty block ("LF") is entered in that place. If there is insufficient space for all parts program blocks, then the previous blocks are first replaced by empty blocks ("LF"), if this is still insufficient, the blocks at the end are also omitted.					
-				String [198]	r
Multi-line: yes	1		1		

absoluteBlockCounter					
Modification counter for display information in the upload buffer					
-	0	0		UWord	r
Multi-line: yes	1		1		

actBlock					
Current part program block. With DISPLOF the subroutine call is displayed.					
-				String [66]	r
Multi-line: yes	1		1		

actBlockA					
Current part program block. If search run is active, then search run block is displayed. Display is always made irrespective of DISPLOF.					
-				String [66]	r
Multi-line: yes	1		1		

actBlockI					
Current part program in the interpreter. Display is always made irrespective of DISPLOF.					
-				String [66]	r
Multi-line: yes	1		1		

actLineNumber					
Line number of the current NC instruction (starting at 1) 0: before program start -1: not available due to an error -2: not available because of DISPLOF					
-				Long Integer	r
Multi-line: yes	1		1		

actPartProgram					
Content of the current part program starting with the previous block. Blocks may be cut off at the end of the string. The line index determines the section within the program. An efficient current block display can be achieved with the aid of a cyclic variable service. If multiple lines are required, then it must be ensured that the client reads line 1 first, so that an NCK internal buffer is filled, which ensures that the following lines are returned accordingly. Note: In EES mode, only the current part program block is ever returned!					
-				String [200]	r
Multi-line: yes	If the index=1, the first data block is returned, for index=n the nth data block.		3		

block					
To display the currently active part programm, NCK supplies 3 ascii-blocks of the part programm in one single variable job (last, current and next block). That means the variable 'block' consists of a maximum of 3 lines: Line index 1: string of the last block Line index 2: string of the current block Line index 3: string of the next block To gain consistent information, all 3 array elements must be processed in one variable request. This is why the maximum string length of each array element is limited to 66 characters.					
-				String [66]	r
Multi-line: yes	Block index, 1 = last, 2 = current, 3 = next block		3		

blockNoStr					
Block number					
-				String [12]	r
Multi-line: no					

byteOffset					
Byte offset of the current NC block in the program workPandProgName					
-				Long Integer	r
Multi-line: no			1		

byteOffsetVL					
Byte offset of the current NC block in the preprocessing					
-				Long Integer	r
Multi-line: no			1		

circleCenter					
Center of the circle (WCS)					
-				Double	r
Multi-line: yes	Line index 1 - 3 for geometry axis 1 - 3 and only effective for G02 or G03		3		

circleCenterS					
Corresponds to circleCenter for search with calculation Attention: This variable is available for protocolling the block search events only, not for the Variable Service!					
-	0			Double	r
Multi-line: yes	No. of the geometry axis		3		

circlePlane					
The vector perpendicular to the circular plane (axial) is output to enable identification of the position of a circle in space					
-				Double	r
Multi-line: yes	No. of the geometry axis		3		

circlePlaneData					
To identify the position of a circle in space, the vector, which is vertical to the plane of the circle, is output (vector)					
-				Double	r
Multi-line: no			1		

circlePlaneDataNorm					
To identify the position of a circle in space, the vector, which is vertical to the plane of the circle, is output (normalized vector)					
-				UWord	r
Multi-line: no			1		

circlePlaneDataNormS					
To identify the position of a circle in space during the search, the vector, which is vertical to the plane of the circle, is output (normalized vector)					
-				UWord	r
Multi-line: no			1		

circlePlaneS					
The vector perpendicular to the circular plane (axial) is output to enable identification of the position of a circle in space					
-				Double	r
Multi-line: yes	No. of the geometry axis		3		

circleRadius					
Radius of the circle (only effective for G02/G03)					
-				Double	r
Multi-line: no					

circleRadiusS					
Corresponds to circleRadius for block search with calculation. Note: This variable is not available for the variable service, but only for logging in connection with block search events!					
-				Double	r
Multi-line: yes	1				

circleTurn					
Progr. number of additional circular passes with helical interpolation in curr. program					
-	0	0		Long Integer	r
Multi-line: yes	1		1		

circleTurnS					
Programmed number of additional circular passes with helical interpolation in the current program for search with calculation. Note: This variable is not available for the Variable Service, but only for logging of block search events					
-	0	0		Long Integer	r
Multi-line: yes	1		1		

cmdToolEdgeCenterCircleCenterEns					
Arc center in relation to WOS frame, i.e. with tool length but without tool radius					
-	0			Double	r
Multi-line: yes	No. of the geometry axis		3		

cmdToolEdgeCenterCircleCenterEnsS					
Corresponds to circleCenterWos for block search with calculation in relation to the WOS frame, i.e. with tool length but without tool radius Note: This variable is not available for the variable service, but only for logging in connection with block search events!					
-	0			Double	r
Multi-line: yes	No. of the geometry axis		3		

cmdToolEdgeCenterCircleDataEns					
Corresponds to cmdToolEdgeCenterCircleCenterEns for the three geometry axes as well as cmdToolEdgeCenterCircleRadiusEns The variable consists of four values of the DOUBLE type, i.e. is 32 bytes long.					
-				Double	r
Multi-line: yes	1		1		

cmdToolEdgeCenterCircleRadiusEns					
Arc radius in relation to WOS frame as center-point path, i.e. with tool length but without tool radius					
-	0			Double	r
Multi-line: yes	1		1		

cmdToolEdgeCenterCircleRadiusEnsS					
Corresponds to circleRadiusWos for block search with calculation in relation to WOS frame as center-point path. i.e. with tool length but without tool radius Note: This variable is not available for the variable service, but only for logging in connection with block search events!					
-	0			Double	r
Multi-line: yes	1		1		

displProgLevel					
Lowest program level to be displayed. The value 1 corresponds to the main program level.					
-				UWord	r
Multi-line: no			1		

displProgLevelVL					
Lowest preprocessing program level that shall be displayed. The value 1 corresponds to the main program level.					
-				UWord	r
Multi-line: no			1		

eesBufferEnd					
The value is only relevant with EED, when the part program is partially loaded into a buffer of the NCK. It indicates which NC block has been entered last in the buffer. The comparison with the value byteOffsetVL shows whether the reloading process is performed quickly enough in order to perform the preprocessing run without delay.					
-				Long Integer	r
Multi-line: no			1		

eesBufferFilling					
The value is only relevant with EES, when the part program is partially loaded into a buffer of the NCK. It indicates who many bytes are provided to the interpreter for processing in a the buffer (eesBufferEnd - byteOffsetVL). If the value approaches 0, this indicates that the reloading process is not performed quickly enough in order to supply the interpreter quickly enough with NC blocks.					
-				Long Integer	r
Multi-line: no			1		

eesBufferStart					
The value is only relevant with EES, when the part program is partially loaded into a buffer NCK. It indicates wthich NC block is entered first in the buffer.					
-				Long Integer	r
Multi-line: no			1		

eesBufferStatus					
Status of the EES buffer					
-				String [12]	r
Multi-line: no					

eesProgLevel					
Lowest program level of the EES reloading mode, which shall be displayed. The value 1 corresponds to the main program level.					
-				UWord	r
Multi-line: no			1		

extProgFlag					
Indicates whether programs are being executed externally 0: Program is being processed from NCK program memory 1: Program is being executed externally 2: Program is being executed in EES mode					
-				UWord	r
Multi-line: no			1		

lastBlockNoStr					
Indicates the last programmed block number, if $MN_DISPLAY_FUNCTION_MASK bit 0 is set. A block number is shown until either a new block number is programmed or the subroutine level which generated the block number has been left. Block numbers of masked blocks are not displayed. There is also no display if DISPLOF is active.					
-				String [12]	r
Multi-line: yes	1		1		

msg					PG
Messages from a part program can be programmed with the instruction 'MSG (...)'. The variable 'msg' contains the text of the current 'MSG(...)'-instruction until a new instrucion is processed or until the message is deleted with the instruction 'MSG ()'.					
-				String [128]	r
Multi-line: no			1		

progName					
Progam name of the currently active program (or subroutine)					
-				String [32]	r
Multi-line: no			1		

seekOffset					
Line number of the current NC block in the program workPandProgName					
-				Long Integer	r
Multi-line: no			1		

seekw					
First line enabled for modification in part program					
-	0	0		Long Integer	r
Multi-line: yes	1		1		

selectedWorkPProg					
Currently selected program, i.e. the program that has been selected with "Select". The variable also displays the program in the JOG and MDI modes. During the simulation, the simulation search temporarily deselects the selected program and selects the program to be simulated. This is hidden by selectedWorkPProg, i.e. during the simulation search, selectedWorkPProg remains unchanged.					
-				String [160]	r
Multi-line: yes	1		1		

singleBlock					
In most cases the variable 'block' is used to read the currently active blocks of the part program. Because this variable is limited to 66 characters per string, it might be necessary (for long blocks) to read longer strings. The variable 'singleBlock' can read complete blocks (up to strings with 198 charecters) . 3 lines can be addressed: Line index 1: last block Line index 2: current block Line index 3: next block It is not guaranteed for rapid block changes, that the information of 3 successive blocks is always consistent, because each block is read with a single variable request. This method is only safe, if the part program has stopped.					
-				String [198]	r
Multi-line: yes	Block index, 1 = last, 2 = current, 3 = next block		3		

stepEditorFormName					
Current module name for step editor is stored					
-				String [128]	r
Multi-line: yes	1		1		

workPName					
Name of the active workpiece					
-				String [32]	r
Multi-line: no			1		

workPNameLong					
Name of the active workpiece					
-				String [128]	r
Multi-line: no					

workPandProgName					
Workpiece name and name of current program.					
-				String [160]	r
Multi-line: yes	1		1		

workPandProgNameVL					
Workpiece name and program name of the current program in the preprocessing.					
-				String [160]	r
Multi-line: no			1		

3.4.5 Area C, Block SPARPP : Program pointer in automatic operation

OEM-MMC: Linkitem /ChannelProgramPointer/...

In automatic mode it is possible to branch to several subroutine levels from the main program level. The state of the program can be determined for every program level. Each variable of the module consists of 11 rows. This makes it possible to address the main program level and 11 subroutine levels (incl. ASUB levels).

The array indices (row indices) mean:

1 = main program level

2-18 = subroutine levels

actInvocCount					
Subroutine call counter, actual value. Specifies the number of subroutine passes. Is always set 1 for the main program and for asynchronous subroutines.					
-				UWord	r
Multi-line: yes	Index of program level		18		

actInvocCountVL					
Subroutine call counter, actual value in the preprocessing. Specifies the number of subroutine passes. It is always set to 1 for the main program and for asynchronous subroutines.					
-				UWord	r
Multi-line: yes	Index of program level		18		

blockLabel					
Block label					
-				String [32]	r
Multi-line: yes	Index of program level		18		

blockNoStr					
Block number [:][N]<No>					
-				String [12]	r
Multi-line: yes	Index of program level		18		

blockNoStrVL					
Block number in the preprocessing [:][N]<No>					
-				String [12]	r
Multi-line: yes	Index of program level		18		

byteOffset					
Byte offset of the current NC block					
-				Long Integer	r
Multi-line: yes	Index of program level		18		

byteOffsetVL					
Byte offset of the current NC block in the preprocessing					
-				Long Integer	r
Multi-line: yes	Index of program level		18		

cmdInvocCount					
Subroutine call counter, desired value. Specifies the number of subroutine passes. Is always set to 1 for the main program and for asynchronous subroutines.					
-				UWord	r
Multi-line: yes	Index of program level		18		

displayState					
Display state for block display. (Blocks should not be displayed automatically for program levels for which DISPLAY OFF has been programmed in the PROC instruction. This is valid also for the subroutine levels below). Value Meaning 0 = DISPLAY OFF for the program level 1 = DISPLAY ON for the program level					
-	0			UWord	r
Multi-line: yes	Index of program level		18		

eesBufferEnd					
The value is only relevant with EED, when the part program is partially loaded into a buffer of the NCK. It indicates which NC block has been entered last in the buffer. The comparison with the value byteOffsetVL shows whether the reloading process is performed quickly enough in order to perform the preprocessing run without delay.					
-				Long Integer	r
Multi-line: yes	Index of program level		18		

eesBufferFilling					
The value is only relevant with EES, when the part program is partially loaded into a buffer of the NCK. It indicates who many bytes are provided to the interpreter for processing in a the buffer (eesBufferEnd - byteOffsetVL). If the value approaches 0, this indicates that the reloading process is not performed quickly enough in order to supply the interpreter quickly enough with NC blocks.					
-				Long Integer	r
Multi-line: yes	Index of program level		18		

eesBufferStart					
The value is only relevant with EES, when the part program is partially loaded into a buffer NCK. It indicates wthich NC block is entered first in the buffer.					
-				Long Integer	r
Multi-line: yes	Index of program level		18		

eesBufferStatus					
Status of the EES buffer					
-				String [12]	r
Multi-line: yes	Index of program level		18		

extProgBufferName					
Name of FIFO buffer for execution from external source					
-				String [160]	rw
Multi-line: yes	Index of program level		18		

extProgFlag					
Indicates whether programs are being executed externally 0: Program is being processed from NCK program memory 1: Program is being executed externally 2: Program is being executed in EES mode					
-				UWord	r
Multi-line: yes	Index of program level		18		

lastBlockNoStr					
Returns the last programmed block number for each program level when $MN_DISPLAY_FUNCTION_MASK bit 0 is set. A block number is shown until either a new block number is programmed or the subroutine level which generated the block number has been left. Block numbers of masked blocks are not displayed. There is also no display if DISPLOF is active.					
-				String [12]	r
Multi-line: yes	Index of program level		18		

progName					
Program name					
-				String [32]	r
Multi-line: yes	Index of program level		18		

progNameVL					
Program name in the preprocessing					
-				String [32]	r
Multi-line: yes	Index of program level		18		

seekOffset					
Search pointer (block offset, each block consists of a string that ends with a line feed)					
-				Long Integer	r
Multi-line: yes	Index of program level		18		

seekOffsetVL					
Search pointer in the preprocessing (block offset, each block consists of a string that ends with a line feed)					
-				Long Integer	r
Multi-line: yes	Index of program level		18		

seekw					
First line enabled for modification in part program					
-	0	0		Long Integer	r
Multi-line: yes	Index of program level		18		

workPName					
Workpiece name = path name in the NC file structure					
-				String [32]	r
Multi-line: yes	Index of program level		18		

workPNameLong					
Workpiece name = path name in the NCK file structure Note: This variable is ignored when lines are accessed!					
-				String [128]	r
Multi-line: yes	Index of program level		18		

workPandProgName					
Workpiece name and name of current program.					
-				String [160]	r
Multi-line: yes	Index of program level		18		

workPandProgNameVL					
Workpiece name and program name of the current program in the preprocessing.					
-				String [160]	r
Multi-line: yes	Index of program level		18		

3.4.6 Area C, Block SPARPI : Program pointer on interruption

OEM-MMC: Linkitem /ChannelInterruptionSearch/...

In order to be able to continue at the point of interruption in a program, the current states of the main program and any subroutines must be stored. On a program interrupt the information is immediately updated in the NCK and reamins valid even after RESET.

This makes it possible to read the states of the main program level and the 11 subroutine levels (incl. ASUB levels).

The array indices (row indices) mean:

1 = main program level

2-18 = subroutine levels

byteOffset					
Search pointer (byte-oriented)					
-				Long Integer	r
Multi-line: yes	Index of program level		18		

displayState					
Display state for block display. (Blocks should not be displayed automatically for program levels for which DISPLAY OFF has been programmed in the PROC instruction. This is valid also for the subroutine levels below). Value Meaning 0 = DISPLAY OFF for the program level 1 = DISPLAY ON for the program level					
-	0			UWord	r
Multi-line: yes	Index of program level		18		

forward					
Search direction 2 = forwards					
-				UWord	r
Multi-line: yes	Index of program level		18		

haltBlock					
The following applies to the SPARPI: The interrupt pointer does not mark the block where the program was cancelled but a previous block (hold block), which enables a better resumption. The hold block is explicitly set with the part program commands IPTRLOCK and IPTRUNLOCK, or implicitly manipulated via $MC_AUTO_IPTR_LOCK. The following applies to the SPARPF: If SPARPI is copied completely, the value of the hold block which was set by the NCK is retained. This enables the NCK to recognize the situation, and it responds with the suppressed message alarm 16950. NOTE: For SPARPI and SPARPF this value is ONLY defined for program level 0.					
-	0	0	1	UWord	r
Multi-line: yes	Program levels (only defined for level 0)		1		

invocCount					
Actual value of the subroutine call counter. Is always 1 for the main program.					
-				UWord	r
Multi-line: yes	Index of program level		18		

plcStartReason					
Specifies for the SERUPRO function which channel has to be started by the PLC so that the current channel starts.					
-	0	0		UWord	r
Multi-line: yes	Index of program level		18		

progName					
Program name					
-				String [32]	r
Multi-line: yes	Index of program level		18		

searchString					
Search string (the first 64 characters of the NC block - corresponding to the search pointer)					
-				String [64]	r
Multi-line: yes	Index of program level		18		

searchType					
Search type 5 = search pointer block-oriented (searching for line feed characters)					
-				UWord	r
Multi-line: yes	Index of program level		18		

seekOffset					
Search pointer (block-oriented, searching for linefeed characters) 1fffffff HEX is returned if the value is invalid.					
-				Long Integer	r
Multi-line: yes	Index of program level		18		

status					
Informs about whether block SPARPI includes currently valid values, and provides the reason for the last update of the block, if available. Note: If an interruption occurs in a program range between the command IPTRLOCK and IPTRUNLOCK, the first block after IPTRLOCK will be provided in the SPARPI instead of the current block. The first interruption between IPTRLOCK and IPTRUNLOCK will set "status" and any additional interruption prior to IPTRUNLOCK will neither change "status" nor SPARPI. 0: Program is running, i.e. SPARPI variables are not up-to-date 1: Program selection, i.e. SPARPI has been reset 2: Block selection through PI service _N_SEL_BL 3: Reset (program abort) 4: Stop after program instruction, e.g. M0 5: Stop with STOP key 6: Stop caused by alarm					
-	1	0	6	UWord	r
Multi-line: yes	1		1		

workPName					
Workpiece name = path name in the NC file structure					
-				String [32]	r
Multi-line: yes	Index of program level		18		

workPNameL					
Workpiece name = path name in the NCK file structure Note: This variable is ignored when lines are accessed!					
-				String [160]	r
Multi-line: yes	Index of program level		18		

workPNameLong					
Workpiece name = path name in the NCK file structure Note: This variable is ignored when lines are accessed!					
-				String [128]	r
Multi-line: yes	Index of program level		18		

3.4.7 Area C, Block SPARPF : Program pointers for block search and stop run

OEM-MMC: Linkitem /ChannelSearch/...

To look for a particular block within a parts program the user can enter search criteria and start a block search. The variables to be entered are combined in the module SPARPF and must be written by the HMI (or another component on the MPI bus).

One main program level and 11 subroutine levels can be processed. These levels are the row indices of the individual variables. The search targets (seek pointer and search string) can only be used mutually exclusively in one level. If a collision occurs, a negative acknowledgement results when the block search is started.

Depending on the search type, the search string is either a block label, block number or any string.

If no path name is specified, the default search strategy for subroutine calls is used. The main program entered in the first program level must be selected for the block search; otherwise the search request is acknowledged negatively.

The array indices (row indices) mean:

1 = main program level for search run

2-18 = subroutine levels for search run

101 = main program level for stop run

102-118 = subroutine levels for stop run

byteOffset					
Search pointer (byte-oriented)					
-				Long Integer	r
Multi-line: yes	Index of program level		18		

displayState					
Display state for block display. (Blocks should not be displayed automatically for program levels for which DISPLAY OFF has been programmed in the PROC instruction. This is valid also for the subroutine levels below). Value Meaning 0 = DISPLAY OFF for the program level 1 = DISPLAY ON for the program level					
-	0			UWord	r
Multi-line: yes	1		18		

forward					
Search direction Search direction "backwards" is only possible in the mode without calculation 1 = backwards (without calculation) 2 = forwards					
-				UWord	rw
Multi-line: yes	Index of program level		18		

haltBlock					
The following applies to the SPARPI: The interrupt pointer does not mark the block where the program was cancelled but a previous block (hold block), which enables a better resumption. The hold block is explicitly set with the part program commands IPTRLOCK and IPTRUNLOCK, or implicitly manipulated via $MC_AUTO_IPTR_LOCK. The following applies to the SPARPF: If SPARPI is copied completely, the value of the hold block which was set by the NCK is retained. This enables the NCK to recognize the situation, and it responds with the suppressed message alarm 16950. NOTE: For SPARPI and SPARPF this value is ONLY defined for program level 0.					
-	0	0	1	UWord	r
Multi-line: yes	Program levels (only defined for level 0)		1		

invocCount					
Actual value of the subroutine call counter. Is always 1 for the main program.					
-				UWord	rw
Multi-line: yes	Index of program level		18		

plcStartReason					
Specifies for the SERUPRO function which channel has to be started by the PLC so that the current channel starts.					
-	0	0		UWord	rw
Multi-line: yes	Index of program level		112		

progName					
Program name. The main program that is used in the first main program level must be selected for the block search, otherwise the search request will be acknowledged negatively.					
-				String [32]	rw
Multi-line: yes	Index of program level		18		

searchString					
Search string (the first 64 characters of the NC block - corresponding to search pointer). Contents of the search string depends on the search type and are either: block label block number any string					
-				String [64]	rw
Multi-line: yes	Index of program level		18		

searchType					
Search type 1 = block number 2 = label 3 = string 4 = program level 5 = search pointer block-oriented (searching for line feeds)					
-				UWord	rw
Multi-line: yes	Index of program level		18		

seekOffset					
Search pointer (block-oriented, searching for line feeds). If the search pointer is used, a program name (progName) always must have been defined. The search pointer refers to this program.					
-				Long Integer	rw
Multi-line: yes	Index of program level		18		

status					
This variable is without function in block SPARPF. It has only been introduced to achieve the same structure of SPARPI and SPARPF.					
-	0	0	0	UWord	rw
Multi-line: yes	1		1		

workPName					
Workpiece name = path name in the NC file structure. If no path name is specified, the default search strategy for subroutine calls is used.					
-				String [32]	rw
Multi-line: yes	Index of program level		18		

workPNameL					
Workpiece name = path name in the NCK file structure. If no path name is specified, the default search strategy for subroutine calls is used. Note: This variable is ignored when lines are accessed!					
-				String [160]	rw
Multi-line: yes	Index of program level		18		

workPNameLong					
Workpiece name = path name in the NCK file structure. If no path name is specified, the default search strategy for subroutine calls is used. Note: This variable is ignored when lines are accessed!					
-				String [128]	rw
Multi-line: yes	Index of program level		18		

3.4.8 Area C, Block SSYNAC : Synchronous actions

OEM-MMC: Linkitem /ChannelSelectedFunctions/...

Several synchronous actions (M, H, S, E, F, T, D) can be active simultaneously in one channel. The module SSYNAC contains a list of all the synchronous actions programmed in the current block. This module consists of arrays of varying length because some types of synchronous actions might be programmed several times in a block. A synchronous action that is not assigned produces a negative number for the respective index.

For each synchronous action there is an address variable and a variable in which the value of the address is entered.

5 M functions

3 S functions

3 H functions

1 T function

1 D function

6 F functions

1 E function

can be programmed in each part program block, but no more than 10 synchronous actions must be programmed in a single block.

Dadr					
D-number. There is only one active D-number per channel.					
-				Long Integer	r
Multi-line: no			1		

Dval					
Value of the current D-number					
-				Long Integer	r
Multi-line: no			1		

Eadr					S5
Number of active E-function					
-				UWord	r
Multi-line: no			1		

Eval					S5
Value of the E-function					
mm/min, inch/min, user defined				Double	r
Multi-line: no			1		

Hadr					S5
Number of active auxiliary functions (H-functions). Up to three H-functions can be active simultaneously.					
-		0	99	UWord	r
Multi-line: yes	Serial number		3		

Hval					S5
Value of the H-function					
-		-99999,9999	99999,9999	Double	r
Multi-line: yes	Serial number		3		

Madr					S5
Number of the active M-function. Up to 5 M-functions can be active simultaneously.					
-		0	99	UWord	r
Multi-line: yes	Serial number		5		

Mval					S5
Value of the M-function					
-		0	99999999	Long Integer	r
Multi-line: yes	Serial number		5		

Sadr					S5
Number of active S-functions. Up to three S-functions can be active simultaneously.					
-		0	6	UWord	r
Multi-line: yes	Serial number		3		

Sval					S5
Value of the S-function. Specifies the spindle speed.					
rev/min , m/min		0	999999,999	Double	r
Multi-line: yes	Serial number		3		

TPreSelAdr					
Number of the preselected T-function					
-				UWord	r
Multi-line: no			1		

TPreSelVal					
Value of the preselected T-function					
-				Long Integer	r
Multi-line: no			1		

Tadr					
Active T-number. Only one T-number can be active at any a time.					
-				UWord	r
Multi-line: no			1		

Tval					
T-function value					
-				Long Integer	r
Multi-line: no			1		

3.4.9 Area C, Block SYNACT : Channel-specific synchronous actions

OEM-MMC: Linkitem /ChannelSelectedFunctions/...

This module contains information on the synchronous actions. The 1000 digit of the cell contains the user protection level (0-7) needed for displaying the corresponding synchronous action.

blockNoStrAct					
If a technology cycle is active: block number of the current action					
-				String [12]	r
Multi-line: yes	(Protection level) * 1000 + no. of the synchronous action		7 * 1000 + numSynAct		

blockNoStrProg					
Number of the block where the synchronous action has been programmed					
-				String [12]	r
Multi-line: yes	(Protection level) * 1000 + no. of the synchronous action		7 * 1000 + numSynAct		

id					
ID of the synchronous action; value 0 means that there is no ID defined (blockwise)					
-				UWord	r
Multi-line: yes	(Protection level) * 1000 + no. of the synchronous action		7 * 1000 + numSynAct		

numElem					
Number of occupied SYNACT elements					
-				UWord	r
Multi-line: yes	See module header				

numSynAct					
Number of synchronous actions					
-				UWord	r
Multi-line: yes	(protection level) * 1000 + 1		7 * 1000 + 1		

numVars					
Number of SYNACT variables					
-				UWord	r
Multi-line: yes	See module header				

progLineOffset					
SYNACT offset within the progPathName file					
-				Long Integer	r
Multi-line: yes	See module header				

progPathName					
Synchronized action file					
-				String [160]	r
Multi-line: yes	See module header				

selectIndex					
The HMI writes the corresponding ID of the synchronized action into lines 8000 or 10000. The variables are then read from this synchronized action and can be viewed via line 8000 in the case of a modal/static synchronized action or line 10000.					
-				UWord	rw
Multi-line: yes	See module header				

selectMask					
Masks some of the entries in the relevant SYNACT list. Only those SYNACTS are entered in the lists for which (selectMask-lowByte AND synActInfo-lowByte) AND (selectMask-higByte AND synActInfo-highByte) apply. The default value 0xFFFF generates completely unfiltered lists. Bit0: Area: User Bit1: Area: Manufacturer Bit2: Area: System Bit3: Area: Safety Bit8: Type: Static Bit9: Type: Modal					
-				UWord	rw
Multi-line: yes	See module header				

synActCounter					
Modification counter for the SYNACT entries in the relevant list.					
-				UWord	r
Multi-line: yes	See module header				

synActInfo					
Information on classification of the SYNACT Bit0: Area: User Bit1: Area: Manufacturer Bit2: Area: System Bit3: Area: Safety Bit8: Type: Static Bit9: Type: Modal					
-				UWord	r
Multi-line: yes	See module header				

synactBlock					
Current synchronized action block (short)					
-				String [66]	r
Multi-line: yes	See module header				

synactBlockL					
Current synchronized action block (long)					
-				String [198]	r
Multi-line: yes	See module header				

typStatus					
Type and status of the synchronized action Bit 0-7 describe the status: Bit 0: Active, i.e. condition fulfilled, action is being executed Bit 1: Lock, i.e. action is locked by PLC or Synact Bit 2: Lock nc, i.e. locked by another Synact Bit 3: Lock plc, i.e. locked by PLC Bit 4: Fire, i.e. condition is fulfilled Bit 5: Check condition, i.e. condition is checked Bit 6: Waiting, i.e. action is waiting to be executed Bit 7: Done, i.e. synchronized action has been completed Bit 8-15 describe the type: Bit 8: Static Bit 9: Modal Bit 10: Blockwise (recognized by id=0)					
-				UWord	r
Multi-line: yes	(Protection level) * 1000 + no. of the synchronous action		7 * 1000 + numSynAct		

varName					
Name of SYNACT variable					
-				String [32]	r
Multi-line: yes	See module header				

varTyp					
Data type of SYNACT variable. Coding according to ACX. 0: BOOL (2 bytes) 3: LONG 10: DOUBLE 12: CHAR[32]					
-				UWord	r
Multi-line: yes	See module header				

varValue					
Value of SYNACT variable					
-				String [32]	r
Multi-line: yes	See module header				

3.4.10 Area C, Block SNCF : Active G functions

OEM-MMC: Linkitem /ChannelSelectedFunctions/...

All G functions are organized in G groups. Only one function of each G group can be active at a time.

The module SNCF consists of a single variable that is organized as an array. The row index corresponds to the G group number.

ncFkt					
Active G-function of relevant group G <No>. If there is no function active within the corresponding G-group, the variable returns an empty string "\0".					
-				String [16]	r
Multi-line: yes	G group number		numGCodeGroups		

ncFktAct					
Active G function of relevant current group in current language mode. Depending on whether function has been programmed in Siemens or ISO Dialect mode, this is identical to ncFkt or ncFktFanuc.					
-				String [16]	r
Multi-line: yes	G group number or ISO Dialect G group number		numGCodeGroups bzw. numGCodeGroupsFanuc		

ncFktBin					
Active G-function of the correponding group					
-				UWord	r
Multi-line: yes	G group number		numGCodeGroups		

ncFktBinAct					
Active G function of relevant current group in current language mode. Depending on whether function has been programmed in Siemens or ISO Dialect mode, this is identical to ncFktBin or ncFktBinFanuc. (The value is the index of the active G function within the group)					
-				UWord	r
Multi-line: yes	G group number or ISO Dialect G group number		numGCodeGroups bzw. numGCodeGroupsFanuc		

ncFktBinFanuc					
Active G function of relevant ISO Dialect group (the value is the index of the active G function within the group)					
-				UWord	r
Multi-line: yes	ISO Dialect G group number		numGCodeGroupsFanuc		

ncFktBinS					
Index of the active G function of the corresponding group for block search with calculation Notice: This variable is available only for logging block search events, but not for the Variable Service.					
-				UWord	r
Multi-line: yes	G group number		numGCodeGroups		

ncFktFanuc					
Active G function of relevant ISO Dialect group					
-				String [16]	r
Multi-line: yes	ISO Dialect G group number		numGCodeGroupsFanuc		

ncFktS					
Name of the active G function of the corresponding group for block search with calculation Notice: This variable is available only for logging block search events, but not for the Variable Service.					
-				String [16]	r
Multi-line: yes	G group number		numGCodeGroups		

3.4.11 Area C, Block NIB : State data: Nibbling

OEM-MMC: Linkitem /ChannelNibbling/...

The module NIB contains technology-specific data for nibbling.

actPunchRate					N4
Strokes per minute					
-				UWord	r
Multi-line: no			1		

automCutSegment					N4
Identifier that indicates which type of automatic block division is active. The division is specified by the commands 'SPP' and 'SPN' in the part program. 0 = no block division 1 = number of segments per block ('SNP') 2 = segments of fixed length ('SPP')					
-				UWord	r
Multi-line: no			1		

numStrokes					N4
Number of strokes when the instruction 'SPN' divides the block into segments (variable 'automCutSegment' = 1).					
-				UWord	r
Multi-line: no			1		

partDistance					N4
If the block has been divided in segments with the instruction 'SPP' (variable 'automCutSegment' = 2) the variable specifies the length of the path between the punches.					
mm, inch, user defined				Double	r
Multi-line: no			1		

punchActive					N4
Identification of punching or nibbling active. The part program turns off/on punching and nibbling with 'SPOF', 'SON' and 'PON'. Rapid punching and nibbling are turned on/off with 'SONS' and 'PONS'. The variable 'punchActive' specified the present state. 0 = inactive 1 = punching active 2 = nibbling active 3 = rapid punching active 4 = rapid nibbling active					
-				UWord	r
Multi-line: no			1		

punchDelayActive					N4
Identifier that indicates whether punching with delay is active. The part program can turn on/off the delay with the instructions 'PDELAYON' and 'PDELAYOF'. The variable 'PunchDelayActive' indicates the present state. 0 = inactive 1 = active					
-				UWord	r
Multi-line: no			1		

punchDelayTime	SD 42400: PUNCH_DWELL_TIME				N4
Punching delay time					
ms				Double	r
Multi-line: no			1		

strokeNr					
Current stroke number					
-				UWord	r
Multi-line: no			1		

3.4.12 Area C, Block FB : Channel-specific base frames

OEM-MMC: Linkitem /ChannelBaseFrame/...

This only applies if $MC_MM_NUM_BASE_FRAMES > 0.

The maximum frame index is: $MC_MM_NUM_BASE_FRAMES - 1

linShift	$P_CHBFR[x,y,TR] x=FrameNo, y=Axis				PA
Translation of settable work offset (the physical unit is defined in basicLengthUnit in module Y in area N).					
mm, inch, user defined				Double	rw
Multi-line: yes	Frame index * (numGeoAxes + numAuxAxes) + axis number		$MC_MM_NUM_BASE_FRAMES * (numGeoAxes + numAuxAxes)		

linShiftFine	$P_CHBFR[x,y,SI] x=FrameNo, y=Axis				
Fine offset with frames, expansion of basic frames and settable frames					
mm, inch, user defined				Double	rw
Multi-line: yes	Frame index * (numGeoAxes + numAuxAxes) + axis number		$MC_MM_NUM_BASE_FRAMES * (numGeoAxes + numAuxAxes)		

mirrorImgActive	$P_CHBFR[x,y,MI] x=FrameNo, y=Axis				PA
Mirroring enabled in a settable work offset 0: Mirroring not active 1: Mirroring active					
-				UWord	rw
Multi-line: yes	Frame index * (numGeoAxes + numAuxAxes) + axis number		$MC_MM_NUM_BASE_FRAMES * (numGeoAxes + numAuxAxes)		

rotation	$P_CHBFR[x,y,RT] x=FrameNo, y=Axis				PA
Rotation of a settable work offset					
Degree				Double	rw
Multi-line: yes	Frame index * (numGeoAxes + numAuxAxes) + axis number		$MC_MM_NUM_BASE_FRAMES * (numGeoAxes + numAuxAxes)		

rotationCoordinate					
Rotation around coordinate of a channel base frame 1: Rotation around first non-existing geometry axis.					
Degree				Double	rw
Multi-line: yes	Frame index * (numGeoAxes + numAuxAxes) + 1		$MC_MM_NUM_BASE_FRAMES * (numGeoAxes + numAuxAxes)		

scaleFact	$P_CHBFR[x,y,SC] x=FrameNo, y=Axis				PA
Scaling factor of a settable work offset					
-				Double	rw
Multi-line: yes	Frame index * (numGeoAxes + numAuxAxes) + axis number		$MC_MM_NUM_BASE_FRAMES * (numGeoAxes + numAuxAxes)		

3.4.13 Area C, Block FS : Channel-specific system frames

OEM-MMC: Linkitem /ChannelSystemFrame/...

Those that exist are set by the bits in $MC_MM_SYSTEM_FRAME_MASK.

Consequently, there may be gaps between the active system frames.

The maximum frame index is:

3 up to but excluding SW $[[SW440000]].

5 from and including SW $[[SW440000]].

11 as from and including SW $[[SW660000]].

12 as from and including SW $[[SW700000]].

linShift	$P_SETFR[Achse, TR]				
Translation					
mm, inch, user defined	0			Double	rw
Multi-line: yes	Frame index * (numGeoAxes +numAuxAxes) + axno		12 * (numGeoAxes+numAuxAxes)		

linShiftFine	$P_SETFR[Achse, SI]				
Fine offset					
mm, inch, user defined	0			Double	rw
Multi-line: yes	Frame index * (numGeoAxes +numAuxAxes) + axno		12 * (numGeoAxes+numAuxAxes)		

mirrorImgActive	$P_SETFR[Achse, MI]				
Mirroring 0: Mirroring not active 1: Mirroring active					
-	0	0	1	UWord	rw
Multi-line: yes	Frame index * (numGeoAxes +numAuxAxes) + axno		12 * (numGeoAxes+numAuxAxes)		

rotation	$P_SETFR[Achse, RT]				
Rotation					
Degree	0			Double	rw
Multi-line: yes	Frame index * (numGeoAxes +numAuxAxes) + axno		12 * (numGeoAxes+numAuxAxes)		

rotationCoordinate					
Rotation around a coordinate of a system frame 1: Rotation around first non-existing geometry axis.					
Degree	0			Double	rw
Multi-line: yes	Frame index * (numGeoAxes +numAuxAxes) + 1		12 * (numGeoAxes+numAuxAxes)		

scaleFact	$P_SETFR[Achse, SC]				
Scaling factor					
-	0			Double	rw
Multi-line: yes	Frame index * (numGeoAxes +numAuxAxes) + axno		12 * (numGeoAxes+numAuxAxes)		

3.4.14 Area C, Block AUXFU : Auxiliary functions

OEM-MMC: Linkitem /ChannelAuxiliaryFunctions/...

The module includes the active auxiliary functions for each group.

In the line, the auxiliary function group (64 groups) and the

desired view are addressed:

Line 1001-1064: Active auxiliary function from the point of view of the NCK

Line 2001-2064: Collected auxiliary function (after search run) from the point of view of the NCK

Line 3001-3064: Active auxiliary function from the point of view of the PLC

Line 1-64: Summary of the above views

Only the values of lines 3001-3064 can be written.

When writing individual values, it must be taken care

that the status variable is written last.

The entire data block of an auxiliary function will not be

accepted before this variable is written.

acAuxfuMTick	$AC_AUXFU_M_TICK[groupIndex]				
The variable is used to read the time stamp of the last auxiliary function group collected (search function) or output for an auxiliary function group. If no auxiliary function has been output for a specified group yet, the variable provides value -1.					
-	-1	INT_MIN	INT_MAX	Long Integer	rw
Multi-line: yes	Group of auxiliary functions/view		3128		

acAuxfuPredefIndex	$AC_AUXFU_PREDEF_INDEX[groupIndex]				
The variable is used to read the pre-defined index of the last auxiliary function group collected (search function) or output for an auxiliary function group. If no auxiliary function has been output for a specified group yet, the variable provides value -1.					
-	-1	-1	INT_MAX	Long Integer	rw
Multi-line: yes	Group of auxiliary functions/view		3064		

acAuxfuSpec	$AC_AUXFU_SPEC[groupIndex]				
The variable is used to read the output specification of the last auxiliary function group collected (search function) or output for an auxiliary function group. If no auxiliary function has been output for a specified group yet, the variable provides value -1. The output specification is bit-coded: Bit 0 = 1 Acknowledgement "normal" after an OB1 cycle Bit 1 = 1 Acknowledgement "quick" with OB40 Bit 2 = 1 No pre-defined auxiliary function Bit 3 = 1 No output to the PLC Bit 4 = 1 Spindle reaction after acknowledgement by the PLC Bit 5 = 1 Output before movement Bit 6 = 1 Output during movement Bit 7 = 1 Output at block end Bit 8 = 1 No output after block search type 1,2,4 Bit 9 = 1 Collection during block search type 5 (SERUPRO) Bit 10 = 1 No output during block search type 5 (SERUPRO) Bit 11 = 1 Cross-channel auxiliary function (SERUPRO) Bit 12 = 1 Output made through synchronized action Bit 13 = 1 Implicit auxiliary function Bit 14 = 1 Active M01 Bit 15 = 1 No output during travel-in test run Bit 16 = 1 Nibbling OFF Bit 17 = 1 Nibbling ON Bit 18 = 1 Nibbling					
-	-1	INT_MIN	INT_MAX	Long Integer	rw
Multi-line: yes	Group of auxiliary functions/view		3064		

acAuxfuTickHifu	$AC_AUXFU_TICK[groupIndex,2]				
The variable is used to read the auxiliary function counter per package of the last auxiliary function group collected (search function) or output for an auxiliary function group.					
-	0	INT_MIN	INT_MAX	Long Integer	rw
Multi-line: yes	Group of auxiliary functions/view		3064		

acAuxfuTickPack	$AC_AUXFU_TICK[groupIndex,1]				
The variable is used to read the package counter per sequence of the last auxiliary function group collected (search function) or output for an auxiliary function group.					
-	0	INT_MIN	INT_MAX	Long Integer	rw
Multi-line: yes	Group of auxiliary functions/view		3064		

acAuxfuTickSeq	$AC_AUXFU_TICK[groupIndex,0]				
The variable is used to read the output sequence counter (all outputs within an IPO cycle) of the last auxiliary function group collected (search function) or output for an auxiliary function group.					
-	0	INT_MIN	INT_MAX	Long Integer	rw
Multi-line: yes	Group of auxiliary functions/view	3064			

extension	$AC_AUXFU_EXT[groupIndex]				
Extension of the auxiliary function					
-	0	0		UWord	rw
Multi-line: yes	Group of auxiliary functions/view	3128			

status					
Status of the auxiliary function Bit0 = 1: Auxiliary function has been collected (NCK view) Bit1 = 1: Auxiliary function has been output to PLC (NCK view) Bit2 = 1: Auxiliary function has been acknowledged by PLC (NCK view) Bit3 = 1: Auxiliary function has been acknowledged by PLC (PLC view) Bit4 = 1: Auxiliary function has been functionally completed (PLC view) Bit14 = 1: Value type is LONG Bit15 = 1: Value type is DOUBLE					
-	0	0		UWord	rw
Multi-line: yes	Group of auxiliary functions/view	3128			

type	$AC_AUXFU_TYPE[groupIndex]				
Type of the auxiliary function, e.g. "M", "S", "T", "D", "F", "H", "L".					
-				String [2]	rw
Multi-line: yes	Group of auxiliary functions/view	3128			

valueDo	$AC_AUXFU_VALUE[groupIndex]				
Value of the auxiliary function. This value will be supplied, if "status" Bit15 = 1					
-	0	0		Double	rw
Multi-line: yes	Group of auxiliary functions/view	3128			

valueLo	$AC_AUXFU_M_VALUE[groupIndex]				
Value of the auxiliary function. This value will be supplied, if "status" Bit14 = 1					
-	0	0		Long Integer	rw
Multi-line: yes	Group of auxiliary functions/view		3128		

3.5 Status data of the axes

3.5.1 Area C, Block SMA : State data: Machine axes

OEM-MMC: Linkitem /ChannelMachineAxis/...

All state data that are dependent on machine movement and are defined specifically for machine axes (geometry and special axes) are combined in module SMA. Supplementary information is to be found in module SEMA. The individual variables are defined as fields where the line index is the number of the machine axis (assigned to the current channel). The variable "name" in module SMA with the line index in question identifies the axis.

The assignment of the line indices in modules SMA and SEMA is identical.

actIncrVal	DB31-48, DBB5				H1
Active INC weighting of the axis 0 = INC_10000 1 = INC_1000 2 = INC_100 3 = INC_10 4 = INC_1 5 = INC_VAR 6 = INC_JOG_CONT 7 = no incremental mode set					
-				UWord	r
Multi-line: yes	Axis index		numMachAxes		

actToolBasePos	$AA_IM[x] x = Ax is				
Tool base position. Physical unit is defined in the variable extUnit (from this module)					
mm, inch, degree, user defined				Double	r
Multi-line: yes	Axis index		numMachAxes		

cmdToolBasePos					
Tool base position, desired value . Physical unit is defined in variable extUnit (in this module).					
mm, inch, degree, user defined				Double	r
Multi-line: yes	Axis index		numMachAxes		

extUnit					
Current physical unit of the axis position 0 = mm 1 = inch 2 = degree 3 = indexing position 4 = userdef					
-				UWord	r
Multi-line: yes	Axis index		numMachAxes		

name					
Axis name					
-				String [32]	r
Multi-line: yes	Axis index		numMachAxes		

status					
Axis status 0 = travel command in plus direction 1 = travel command in minus direction 2 = exact position coarse reached 3 = exact position fine reached					
-				UWord	r
Multi-line: yes	Axis index		numMachAxes		

toolBaseDistToGo					
Tool base distance-to-go. Physical unit is defined in the variable extUnit (in this module).					
mm, inch, degree, user defined				Double	r
Multi-line: yes	Axis index		numMachAxes		

toolBaseREPOS					
Tool base REPOS. Physical unit is defined in the variable extUnit (in this module).					
mm, inch, degree, user defined				Double	r
Multi-line: yes	Axis index		numMachAxes		

varIncrVal					
Settable value for INC_VAR. The physical value depends on whether the axis is linear or rotary. Linear axis: unit is 1 mm Rotary axis: unit is 1/1000 degrees					
mm, inch, degree, user defined				Double	r
Multi-line: yes	Axis index		numMachAxes		

3.5.2 Area C, Block SEMA : State data: Machine axes (extension of SMA)

OEM-MMC: Linkitem /ChannelMachineAxis/...

All state data that are dependent on machine movement and are defined specifically for machine axes (geometry and special axes) are combined in module SMA. Supplementary information is to be found in module SEMA. The individual variables are defined as fields where the line index is the number of the machine axis (assigned to the current channel). The variable "name" in module SMA with the line index in question identifies the axis.

The assignment of the line indices in modules SMA and SEMA is identical.

PRESETActive					
Preset state 0 = no preset active 1 = preset active					
-				UWord	r
Multi-line: yes	Axis Number		numMachAxes		

PRESETVal	$AC_PRESET[x] x = Axis				
The function PRESETON (...) programs a work offset for an axis. The value of the offset is stored in the variable 'PRESETVal'. The variable can be overwritten by the part program and by the HMI.					
mm, inch, user defined				Double	rw
Multi-line: yes	Axis Number		numMachAxes		

aaAcc	$AA_ACC[Achse]				
Current axial acceleration value					
m/s2, 1000 inch/ s2, rev/s2, user defined	0			Double	r
Multi-line: yes	Axis Number		numMachAxes		

aaAccPercent	$AA_ACC_PERCENT[Achse]				
Current acceleration value for single-axis interpolation in percent					
-	0	0		UWord	r
Multi-line: yes	Axis Number		numMachAxes		

aaActIndexAxPosNo	$AA_ACT_INDEX_AX_POS_NO[<Achse>]				
Current indexing position; the display depends on $MN_INDEX_AX_NO_MODE and the division (via table or equidistant)					
-	0			Long Integer	r
Multi-line: yes	Axis Number		numMachAxes		

aaAlarmStat	$AA_ALARM_STAT				
Display indicating whether alarms are active for a PLC-controlled axis. The relevant coded alarm reactions can be used as a source for the "Extended Stop and Retract" function. The data is bit-coded, allowing, where necessary, individual states to be masked or evaluated separately (bits not listed supply a value of 0) Bit2 = 1: NOREADY (active rapid deceleration + cancelation of servo enable) Bit6 = 1: STOPBYALARM (rampm stop in all channel axes) Bit9 = 1: SETVDI (VDI interface signal "Setting alarm") Bit13 = 1: FOLLOWUPBYALARM (Follow-up)					
-	0			UWord	r
Multi-line: yes	Axis Number		numMachAxes		

aaAxChangeStat	$AA_AXCHANGE_STAT[Achse]				
Axis status with respect to axis replacement 0: Axis can be replaced 1: Axis is linked to the channel, but can become the PLC, command or reciprocating axis 2: Axis cannot be replaced					
-	0	0	2	UWord	r
Multi-line: yes	Axis Number		numMachAxes		

aaAxChangeTyp	$AA_AXCHANGE_TYP[Achse]				
Axis type with respect to axis replacement 0: Axis assigned to the NC program 1: Axis assigned to the PLC or active as command axis or reciprocating axis 2: Other channel has interpolation right 3: Neutral axis 4: Neutral axis controlled from the PLC 5: Other channel has interpolation right; axis is requested for the NC program 6: Other channel has interpolation right; axis is requested as neutral axis 7: Axis is PLC axis or is active as command axis or reciprocating axis; axis is requested for the NC program 8: Axis is PLC axis or is active as command axis or reciprocating axis; axis is requested as neutral axis					
-	0	0	8	UWord	r
Multi-line: yes	Axis Number		numMachAxes		

aaAxDisable	$AA_AX_DISABLE[<Achse>]				
Resulting status of axis/spindle disable. 0: Axis/spindle disable not active. 1: Axis/spindle disable active.					
-	0			UDoubleword	r
Multi-line: yes	Axis Number		numMachAxes		

aaAxDisableSrc	$AA_AX_DISABLE_SRC[<Achse>]				
Bitmask which provides the status and source of the currently active axis/spindle disable. If bit 0 has been set, the axis/spindle disable is active. The data is bit-coded. Individual states can therefore be masked or evaluated separately if necessary: Bit0 = 1: Resulting state from all sources: axis/spindle disable active. Bit1 = 1: Axial signal 'Axis/spindle disable triggered by PLC' is active. Bit2 = 1: Channel-specific program test is active. Bit3 = 1: Axial suppression of 'Program test triggered by the PLC' is active. Bit4 = 1: Axial signal 'Program test (energy saving mode)' is active. Bit5 = 1: SERUPRO is active. Bit6 = 1: Coupling object 'Total state is axis/spindle disable' is active. Bit7 = 1: Coupling object 'Total state is real traveling' is active.					
-	0			UDoubleword	r
Multi-line: yes	Axis Number		numMachAxes		

aaBcsOffset	$AA_BCS_OFFSET[Achse]				
Sum of all axial offsets of an axis, such as DRF, online tool offset, $AA_OFF and ext. WO.					
-	0			Double	r
Multi-line: yes	Axis Number		numMachAxes		

aaBrakeCondB	$AA_BRAKE_CONDB[axis]				
Shows the pending braking requests (conditions) for the interpolator stop of the axis / spindle. A braking request consists of a collision direction relating to a coordinate axis in the BCS and a braking priority relating to the machining step. If the axis / spindle receives a current braking request on account of these requirement(s), bit 0 is set in $AA_BRAKE_STATE[X] (in the next IPO cycle).\ The highest deceleration priority in positive direction is shown in bits 0 to 3: 0x0: No pending deceleration request 0x1: Priority 1 covers all positioning actions (G0, POS, SPOS) 0x2: Priority 2 covers DYNNORM and all priority 1 motions 0x3: Priority 3 covers DYNPOS and all priority 1 to 2 motions 0x4: Priority 4 covers DYNROUGH and all priority 1 to 3 motions 0x5: Priority 5 covers DYNSEMIFIN and all priority 1 to 4 motions 0x6: Priority 6 covers all motions (including DYNFINISH). The request could also have been triggered by a CP SW limit stop. 0x7: Priority 7 covers all motions. The request was triggered by the VDI interface signal DB31,..DBX4.3 "Feed stop / Spindle stop". Deceleration always takes place, irrespective of the direction of motion. 0xD: Priority 13 covers all motions. Axial deceleration takes place with an emergency stop deceleration ramp. The highest deceleration priority in negative direction is shown in bits 16 to 19: 0x0 to 0xD: Same meaning as bits 0 to 3 All other bits are reserved and not set. If the value of the variable is shown in hexadecimal format, the fifth character from the right shows the deceleration priority in negative direction while the first number from the right shows the deceleration priority in positive direction.					
-	0	0	0xD000D	UDoubleword	r
Multi-line: yes	Axis Number		numMachAxes		

aaBrakeCondM	$AA_BRAKE_CONDM[axis]				
Shows the pending braking requests (conditions) for the interpolator stop of the axis / spindle. A braking request consists of a collision direction relating to a coordinate axis in the MCS and a braking priority relating to the machining step. The highest deceleration priority in positive direction is shown in bits 0 to 3: 0x0: No pending deceleration request 0x1: Priority 1 covers all positioning actions (G0, POS, SPOS) 0x2: Priority 2 covers DYNNORM and all priority 1 motions 0x3: Priority 3 covers DYNPOS and all priority 1 to 2 motions 0x4: Priority 4 covers DYNROUGH and all priority 1 to 3 motions 0x5: Priority 5 covers DYNSEMIFIN and all priority 1 to 4 motions 0x6: Priority 6 covers all motions (including DYNFINISH). The request could also have been triggered by a CP SW limit stop. 0x7: Priority 7 covers all motions. The request was triggered by the VDI interface signal DB31,..DBX4.3 "Feed stop / Spindle stop". Deceleration always takes place, irrespective of the direction of motion. 0xD: Priority 13 covers all motions. Axial deceleration takes place with an emergency stop deceleration ramp. The highest deceleration priority in negative direction is shown in bits 16 to 19: 0x0 to 0xD: Same meaning as bits 0 to 3 All other bits are reserved and not set. If the value of the variable is shown in hexadecimal format, the fifth character from the right shows the deceleration priority in negative direction while the first number from the right shows the deceleration priority in positive direction.					
-	0	0	0xD000D	UDoubleword	r
Multi-line: yes	Axis Number		numMachAxes		

aaBrakeState	$AA_BRAKE_STATE[axis]				
Returns for the axis / spindle whether an active deceleration request has been set on account of the request of aaBrakeCondB or a CP SW limit stop or a VDI interface signal DB31,..DBX4.3 "Feed stop / Spindle stop".					
-	0	0	1	UDoubleword	r
Multi-line: yes	Axis Number		numMachAxes		

aaChanNo	$AA_CHANNO[Achse]				
The variable supplies the number of the channel in which the axis is currently being interpolated. With value 0, the axis could not be assigned to any channel.					
-	0	0		UWord	r
Multi-line: yes	Axis Number		numMachAxes		

aaCollPos	$AA_COLLPOS[Achse]				
Position of a machine axis with risk of collision					
mm, inch, degree, user defined	0			Double	r
Multi-line: yes	Axis Number		numMachAxes		

aaCoupAct	$AA_COUP_ACT[x] x = Spindle following				
Current coupling state of the slave spindle					
-				UWord	r
Multi-line: yes	Axis Number		numMachAxes		

aaCoupCorr	$AA_COUP_CORR[Achse]				
This variable is used to execute the function "Correct synchronism error". It returns the compensation value for the position offset for the generic coupling with CPFRS = "MCS". The actual values of this spindle are compared with the setpoints for the duration (MD 30455 MISC_FUNCTION_MASK, bit 7) of the activation of the VDI interface signal DB31...,DBX31.6 'Correct synchronism' for the following spindle with coupling active. The difference is the compensation value, which can be read with this variable.					
-	0			Double	r
Multi-line: yes	Axis Number		numMachAxes		

aaCoupCorrDist	$AA_COUP_CORR_DIST[Achse]				
Generic coupling: path still to be retracted for aaCoupCorr					
-	0			Double	r
Multi-line: yes	Axis Number		numMachAxes		

aaCoupOffs	$AA_COUP_OFFS[x] x = Spindle				
Position offset of the synchronous spindle desired value					
-				Double	r
Multi-line: yes	Axis Number		numMachAxes		

aaCurr	$AA_CURR[x] x = Axis				
Actual value of the axis/spindle current in A (only available for PROFIdrive drives)					
A				Double	r
Multi-line: yes	Axis Number		numMachAxes		

aaDepAxO	$AA_DEPAXO[Achse]				
Dependency on other axes. Returns an axis code for the defined axis AX containing all the machine axes that have a mechanical dependency on the defined axis.					
-	0	0		Long Integer	r
Multi-line: yes	Axis Number		numMachAxes		

aaDtbb	$AA_DTBB[x] x = Axis				
Axis-specific distance from the beginning of the block in the BCS for positioning and synchronous axes used in synchronous actions (note: SYNACT only)					
-				Double	r
Multi-line: yes	Axis Number		numMachAxes		

aaDtbreb	$AA_DTBREB[axis]				
The estimated total distance until the end of deceleration is reached, BCS					
-	0	0		Double	r
Multi-line: yes	Axis Number		numMachAxes		

aaDtbrebCmd	$AA_DTBREB_CMD[axis]				
Command share of the overall deceleration distance of axis ax in the BCS. The value is the estimated deceleration distance of the axis up to standstill.					
-	0	0		Double	r
Multi-line: yes	Axis Number		numMachAxes		

aaDtbrebCorr	$AA_DTBREB_CORR[axis]				
Offset section of the deceleration distance, BCS					
-	0	0		Double	r
Multi-line: yes	Axis Number		numMachAxes		

aaDtbrebDep	$AA_DTBREB_DEP[axis]				
Dependent section of the decelaration distance, BCS					
-	0	0		Double	r
Multi-line: yes	Axis Number		numMachAxes		

aaDtbrem	$AA_DTBREM[axis]				
The estimated total distance until the end of deceleration is reached, MCS					
-	0	0		Double	r
Multi-line: yes	Axis Number		numMachAxes		

aaDtbremCmd	$AA_DTBREM_CMD[axis]				
Specified section of the decelaration distance, MCS					
-	0	0		Double	r
Multi-line: yes	Axis Number		numMachAxes		

aaDtbremCorr	$AA_DTBREM_CORR[axis]				
Offset section of the deceleration distance, MCS					
-	0	0		Double	r
Multi-line: yes	Axis Number		numMachAxes		

aaDtbremDep	$AA_DTBREM_DEP[axis]				
Dependent section of the decelaration distance, MCS					
-	0	0		Double	r
Multi-line: yes	Axis Number		numMachAxes		

aaDteb	$AA_DTEB[x] x = Axis				
Axis-specific distance to the end of the block in the BCS for positioning and synchronous axes used in synchronous actions (note: SYNACT only)					
-				Double	r
Multi-line: yes	Axis Number		numMachAxes		

aaDtepb	$AA_DTEPB[x] x = Axis				
Axis-specific distance-to-go of infeed during oscillation in the BCS (note: SYNACT only)					
-				Double	r
Multi-line: yes	Axis Number		numMachAxes		

aaEnc1Active	$AA_ENC1_ACTIVE[Achse]				
First measuring system is active 0: Measuring system is not active 1: Measuring system is active					
-	0	0	1	UWord	r
Multi-line: yes	Axis Number		numMachAxes		

aaEnc2Active	$AA_ENC2_ACTIVE[Achse]				
Second measuring system is active 0: Measuring system is not active 1: Measuring system is active					
-	0	0	1	UWord	r
Multi-line: yes	Axis Number		numMachAxes		

aaEncActive	$AA_ENC_ACTIVE[Achse]				
Measuring system is active 0: Measuring system is not active 1: Measuring system is active					
-	0	0	1	UWord	r
Multi-line: yes	Axis Number		numMachAxes		

aaEsrEnable	$AA_ESR_ENABLE[Achse]				
(Axial) enabling of reactions of "Extended Stop and Retract" function. The selected axial ESR reaction must be parameterized in MD $MA_ESR_REACTION. beforehand. The corresponding Stop or Retract reactions can be activated via $AN_ESR_TRIGGER (or for individual drives in the event of communications failure/ DC-link undervoltage), generator-mode operation is automatically activated in response to undervoltage conditions. 0: FALSE 1: TRUE					
-	0	0	1	UWord	r
Multi-line: yes	Axis Number		numMachAxes		

aaEsrStat	$AA_ESR_STAT[Achse]				
(Axial) status checkback signals of "Extended Stop and Retract" function, which can be applied as input signals for the gating logic of the ESR (synchronous actions). The data is bit-coded. Individual states can therefore be masked or evaluated separately if necessary: Bit0 = 1: Generator mode is activated Bit1 = 1: Retract operation is activated Bit2 = 1: Stop operation is activated Bit3 = 1: Risk of undervoltage (DC-link voltage monitoring, voltage has dropped below warning threshold) Bit4 = 1: Speed has dropped below minimum generator mode threshold (i.e. no more regenerative rotation energy is available).					
-	0			UWord	r
Multi-line: yes	Axis Number		numMachAxes		

aaEsrTrigger	$AA_ESR_TRIGGER				
Activation of "NC-controlled ESR" for PLC-controlled axis					
-	0	0	1	UWord	r
Multi-line: yes	Axis Number		numMachAxes		

aaFixPointSelected	$AA_FIX_POINT_SELECTED[<Achse>]				
Selected fixed point: Number of the fixed point that is to be approached					
-	0			UDoubleword	r
Multi-line: yes	Axis Number		numMachAxes		

aaIbnCorr	$AA_IBN_CORR[<Achse>]				
Current BZS setpoint value of an axis including override components					
-	0			Double	r
Multi-line: yes	Axis Number		numMachAxes		

aaIenCorr	$AA_IEN_CORR[<Achse>]				
Current SZS setpoint value of an axis including override components					
-	0			Double	r
Multi-line: yes	Axis Number		numMachAxes		

aaInSync	$AA_IN_SYNC[Achse]				
Synchronization status of the following axis with master value coupling and ELG 0: Synchronization is not running 1: Synchronization is running, i.e. following axis is being synchronized					
-	0	0	1	UWord	r
Multi-line: yes	Axis Number		numMachAxes		

aaInposStat	$AA_INPOS_STAT[Achse]				
Status for the programmed position 0: No status available (axis/spindle is outside of the programmed position) 1: Travel motion pending 2: Position setpoint reached 3: Position reached with 'exact stop coarse' 4: Position reached with 'exact stop fine'					
-	0	0	4	UWord	r
Multi-line: yes	Axis Number		numMachAxes		

aaIpoNcChanax	$AA_IPO_NC_CHANAX				
If the axis is currently interpolated to this NCU, the channel and channel axis number which define the interpolator of the axis are output. If the axis is currently interpolated to a different NCU, the NCU identifier of the interpolated NCU and the global axis number of the machine axis are output. This global axis number can then be used to transfer the interpolated channel and the channel axis number to the other NCU, with NCU ID 2, with anIpoChanAx[203]. The axis must be assigned to at least one channel on this NCU, otherwise 0 will be returned. The channel is output as from position 100, and the channel axis number is output as from position 1, e.g. 1005 - channel 10 channel axis 5. These values are always lower than 10000. The NCU is output as from position 10000, e.g. 20203: NCU 2 and the global axis number is 203.					
-	0	0		UDoubleword	r
Multi-line: yes	Axis Number		numMachAxes		

aaJerkCount	$AA_JERK_COUNT[Achse]				
Total traverse processes of an axis with jerk					
-		0		Double	r
Multi-line: yes	Axis Number		numMachAxes		

aaJerkTime	$AA_JERK_TIME[Achse]				
Total traverse time of an axis with jerk					
s, user defined		0		Double	r
Multi-line: yes	Axis Number		numMachAxes		

aaJerkTotal	$AA_JERK_TOT[Achse]				
Overall total jerk of an axis					
-		0		Double	r
Multi-line: yes	Axis Number		numMachAxes		

aaJogPosAct	$AA_JOG_POS_ACT[Achse]				
Position reached for JOG to position					
-	0	0	1	UWord	r
Multi-line: yes	Axis Number		maxnumGlobMachAxes		

aaJogPosSelected	$AA_JOG_POS_SELECTED[Achse]				
JOG to position is active					
-	0	0	1	UWord	r
Multi-line: yes	Axis Number		maxnumGlobMachAxes		

aaLeadP	$AA_LEAD_P[x] x = Axis				
Actual lead value position					
mm, inch, degree, user defined				Double	r
Multi-line: yes	Axis Number		numMachAxes		

aaLeadPTurn	$AA_LEAD_P_TURN				
Current master value - position component lost as a result of modulo reduction					
mm, inch, degree, user defined	0	0		Double	r
Multi-line: yes	Axis Number		numMachAxes		

aaLeadSp	$AA_LEAD_SP[x] x = Axis				
Simulated lead value - position					
mm, inch, degree, user defined				Double	r
Multi-line: yes	Axis Number		numMachAxes		

aaLeadSv	$AA_LEAD_SV[x] x = Axis				
Simulated leading value velocity					
mm, inch, degree, user defined				Double	r
Multi-line: yes	Axis Number		numMachAxes		

aaLeadV	$AA_LEAD_V[x] x = Axis				
Actual lead value - velocity					
mm/min, inch/min, user defined				Double	r
Multi-line: yes	Axis Number		numMachAxes		

aaLoad	$AA_LOAD[x] x = Axis				
Drive load in % (only available for PROFIdrive drives)					
%				Double	r
Multi-line: yes	Axis Number		numMachAxes		

aaLoadSmooth	$AA_LOAD_SMOOTH[Achse]				
Smoothed drive load in %					
%				Double	r
Multi-line: yes	Axis Number		maxnumGlobMachAxes		

aaMachax	$AA_MACHAX				
The NCU and machine axis are output for an axis, representing the physical image of the axis. The machine axis must be assigned to at least one channel on this NCU, otherwise 0 will be returned. Without an NCU link, i.e. if there is only one NCU, only the number of the machine axis will be output. In this case, the NCU ID is equal to zero. The NCU ID is output as from position 100, e.g. 20005: NCU 2 axis 5.					
-	0	0		UDoubleword	r
Multi-line: yes	Axis Number		numMachAxes		

aaMaslDef	$AA_MASL_DEF				
Each slave axis currently coupled via master-slave delivers the machine axis number of the corresponding master axis. Zero is displayed as default if the coupling is not configured. A master axis also shows default value zero. 0: No coupling for this axis configured, or axis is master axis, or no coupling active >0: Machine axis number of the master axis with which the slave axis is currently coupled					
-	0	0	numMachAxes	UWord	r
Multi-line: yes	Axis Number		numMachAxes		

aaMaslState	$AA_MASL_STAT				
Each slave axis currently coupled via master-slave delivers the machine axis number of the corresponding master axis. Zero is displayed as default for inactive coupling. A master axis also shows default value zero. 0: No coupling for this axis configured, or axis is master axis, or no coupling active >0: Machine axis number of the master axis with which the slave axis is currently coupled					
-	0	0	numMachAxes	UWord	r
Multi-line: yes	Axis Number		numMachAxes		

aaMeaAct	$AA_MEAACT[Achse]				
Axial measuring active 0: Measuring system is not active 1: Measuring system is active					
-	0	0	1	UWord	r
Multi-line: yes	Axis Number		numMachAxes		

aaMm	$AA_MM[x] x = Axis				
Latched probe position in the machine coordinate system					
mm, inch, degree, user defined				Double	rw
Multi-line: yes	Axis Number		numMachAxes		

aaMm1	$AA_MM1[x] x = Axis				
Access to measurement result of trigger event 1 in the MCS					
mm, inch, degree, user defined				Double	rw
Multi-line: yes	Axis Number		numMachAxes		

aaMm2	$AA_MM2[x] x = Axis				
Access to measurement result of trigger event 2 in the MCS					
mm, inch, degree, user defined				Double	rw
Multi-line: yes	Axis Number		numMachAxes		

aaMm3	$AA_MM3[x] x = Axis				
Access to measurement result of trigger event 3 in the MCS					
mm, inch, degree, user defined				Double	rw
Multi-line: yes	Axis Number		numMachAxes		

aaMm4	$AA_MM4[x] x = Axis				
Access to measurement result of trigger event 4 in the MCS					
mm, inch, degree, user defined				Double	rw
Multi-line: yes	Axis Number		numMachAxes		

aaOff	$AA_OFF[x] x = Axis				
Superimposed position offset from synchronous actions					
mm, inch, degree, user defined				Double	r
Multi-line: yes	Axis Number		numMachAxes		

aaOffLimit	$AA_OFF_LIMIT[x] x = Axis				
Limit for axial correction $AA_OFF reached (Note: for SYNACT only) 0: Limit value not reached 1: Limit value reached in positive axis direction 11: Limit value reached in negative axis direction					
-				UWord	r
Multi-line: yes	Axis Number		numMachAxes		

aaOffVal	$AA_OFF_VAL[x]				
Integrated value of overlaid motion for an axis. The negative value of this variable can be used to cancel an overlaid motion. e.g. $AA_OFF[axis] = -$AA_OFF_VAL[axis]					
-	0			Double	r
Multi-line: yes	Axis Number		numMachAxes		

aaOnFixPoint	$AA_FIX_ON_POINT[<Achse>]				
Number of the fixed point at which the axis stands					
-	0			UDoubleword	r
Multi-line: yes	Axis Number		numMachAxes		

aaOscillBreakPos1	$AA_OSCILL_BREAK_POS1[<Achse>]				
Oscillation interrupt position 1					
mm, inch, degree, user defined				Double	r
Multi-line: yes	Axis Number		numMachAxes		

aaOscillBreakPos2	$AA_OSCILL_BREAK_POS2[<Achse>]				
Oscillation interrupt position 2					
mm, inch, degree, user defined				Double	r
Multi-line: yes	Axis Number		numMachAxes		

aaOscillReversePos1	$AA_OSCILL_REVERSE_POS1[x] x = Axis				
Current reverse position 1 for oscillation in the BCS. For synchronous actions the value of the setting data $SA_OSCILL_REVERSE_POS1 is evaluated online; (note: SYNACT only)					
mm, inch, degree, user defined				Double	r
Multi-line: yes	Axis Number		numMachAxes		

aaOscillReversePos2	$AA_OSCILL_REVERSE_POS2[x] x = Axis				
Current reverse position 2 for oscillation in the BCS; For synchronous actions the value of the setting data $SA_OSCILL_REVERSE_POS1 is evaluated online; (note: SYNACT only)					
mm, inch, degree, user defined				Double	r
Multi-line: yes	Axis Number		numMachAxes		

aaOvr	$AA_OVR[x] x = Axis				
Axial override for synchronous actions					
-				Double	r
Multi-line: yes	Axis Number		numMachAxes		

aaPlcOvr	$AA_PLC_OVR[Achse]				
Axial override specified by PLC for motion-synchronous actions					
-	100	0		Double	r
Multi-line: yes	Axis Number		numMachAxes		

aaPolfa	$AA_POLFA				
The programmed retraction position of the single axis					
mm, inch, degree, user defined				Double	r
Multi-line: yes	Axis Number		numMachAxes		

aaPolfaValid	$AA_POLFA_VALID				
States whether the retraction of the single axis is programmed 0: No retraction programmed for the single axis 1: Retraction programmed as position 2: Retraction programmed as distance					
-	0	0	2	UWord	r
Multi-line: yes	Axis Number		numMachAxes		

aaPosRes	$AA_POSRES				
Axis status "Position restored". The value TRUE shows that the position of the axis has been restored after the voltage breakdown. ($MA_ENC_REFP_STATE[] = 3). After referencing of the axis, the value goes to FALSE. 1 = TRUE: Axis position not restored 0 = FALSE: Axis position restored					
-				UWord	r
Multi-line: yes	Axis Number		numMachAxes		

aaPower	$AA_POWER[x] x = Axis				
Drive power in W (only available for PROFIdrive drives)					
W				Double	r
Multi-line: yes	Axis Number		numMachAxes		

aaPowerSmooth	$AA_POWER_SMOOTH[Achse]				
Smoothed drive power in W (only available for PROFIdrive drives)					
W				Double	r
Multi-line: yes	Axis Number		maxnumGlobMachAxes		

aaProgIndexAxPosNo	$AA_PROG_INDEX_AX_POS_NO[Achse]				
Programmed indexing position 0: No indexing axis, therefore no indexing position available >0: Number of the programmed indexing position					
-	0	0		UWord	r
Multi-line: yes	Axis Number		numMachAxes		

aaRef	$AA_REF[Achse]				
Axis is referenced 0: Axis is not referenced 1: Axis is referenced					
-	0	0	1	UWord	r
Multi-line: yes	Axis Number		numMachAxes		

aaReposDelay	$AA_REPOS_DELAY[Achse]				
REPOS suppression active 0: REPOS suppression is currently not active for this axis 1: REPOS suppression is currently active for this axis					
-	0	0	1	UWord	r
Multi-line: yes	Axis Number		numMachAxes		

aaScPar	$AA_SCPAR[Achse]				
Current setpoint parameter set					
-	0	0		Long Integer	r
Multi-line: yes	Axis Number		numMachAxes		

aaSnglAxStat	$AA_SNGLAX_STAT				
Display status of a PLC-controlled axis 0: Not a single axis 1: Reset 2: Ended 3: Interrupted 4: Active 5: Alarm					
-	0			UWord	r
Multi-line: yes	Axis Number		numMachAxes		

aaSoftendn	$AA_SOFTENDN[x] x = Axis				
Software end position, negative direction					
-				Double	r
Multi-line: yes	Axis Number		numMachAxes		

aaSoftendp	$AA_SOFTENDP[x] x = Axis				
Software end position, positive direction					
-				Double	r
Multi-line: yes	Axis Number		numMachAxes		

aaStat	$AA_STAT[]				
Axis state 0: no axis state available 1: travel command is active 2: axis has reached the IPO end. only for channel axes 3: axis in position (exact stop coarse) for all axes 4: axis in position (exact stop fine) for all axes					
-				UWord	r
Multi-line: yes	Axis Number		numMachAxes		

aaSync	$AA_SYNC[x] x = Axis				
Coupling status of the following axis with master value coupling 0: No synchronism 1: Synchronism coarse 2: Synchronism fine 3: Synchronism coarse and fine					
-				UWord	r
Multi-line: yes	Axis Number		numMachAxes		

aaSyncDiff	$AA_SYNCDIFF[Achse]				
Setpoint synchronism difference					
mm, inch, degree, user defined	0			Double	r
Multi-line: yes	Axis Number		numMachAxes		

aaSyncDiffStat	$AA_SYNCDIFF_STAT[Achse]				
Status of the setpoint synchronism difference -4: No valid value in aaSyncDiff, coupled motion from part program -3: Reserved -2: Reserved -1: No valid value in aaSyncDiff 0: No valid value in aaSyncDiff, coupling not active 1: Valid value in aaSyncDiff					
-	0	-4	1	Long Integer	r
Multi-line: yes	Axis Number		numMachAxes		

aaTorque	$AA_TORQUE[x] x = Axis				
Drive torque setpoint in Nm (only available for PROFIdrive drives)					
Nm				Double	r
Multi-line: yes	Axis Number		numMachAxes		

aaTotalOvr	$AA_TOTAL_OVR[Achse]				
The total axial override for motion-synchronous actions					
-	100	0		Double	r
Multi-line: yes	Axis Number		numMachAxes		

aaTravelCount	$AA_TRAVEL_COUNT[Achse]				
Total traverse processes of an axis					
-		0		Double	r
Multi-line: yes	Axis Number		numMachAxes		

aaTravelCountHS	$AA_TRAVEL_COUNT_HS[Achse]				
Total traverse processes of an axis at high speed					
-		0		Double	r
Multi-line: yes	Axis Number		numMachAxes		

aaTravelDist	$AA_TRAVEL_DIST[Achse]				
Total travel path of an axis in mm or degrees					
mm, inch, degree, user defined		0		Double	r
Multi-line: yes	Axis Number		numMachAxes		

aaTravelDistHS	$AA_TRAVEL_DIST_HS[Achse]				
Total travel path of an axis at high speed in mm or degrees					
mm, inch, degree, user defined		0		Double	r
Multi-line: yes	Axis Number		numMachAxes		

aaTravelTime	$AA_TRAVEL_TIME[Achse]				
Total traverse time of an axis in seconds					
s, user defined		0		Double	r
Multi-line: yes	Axis Number		numMachAxes		

aaTravelTimeHS	$AA_TRAVEL_TIME_HS[Achse]				
Total traverse time of an axis at high speed in seconds					
s, user defined		0		Double	r
Multi-line: yes	Axis Number		numMachAxes		

aaTyp	$AA_TYP[x] x = Axis				
Axis type 0: axis in other channel 1: channel axis of same channel 2: neutral axis 3: PLC axis 4: reciprocating axis 5: neutral axis, currently traversing in JOG 6: slave axis coupled via master value 7: coupled motion slave axis 8: command axis 9: compile cycle axis					
-				UWord	r
Multi-line: yes	Axis Number		numMachAxes		

aaType	$AA_TYPE[Achse]				
Cross-channel axis type 0: Axis type cannot be determined 1: NC program axis 2: Neutral axis 3: PLC axis 4: Reciprocating axis 5: Neutral axis that is currently executing a JOG or homing motion 6: Following axis coupled to the master value 7: Coupled motion of the following axis, activated in a synchronized action 8: Command axis 9: Compile Cycle axis 10: Coupled slave axis (master-slave function.) 11: Program axis that is currently executing a JOG or homing motion					
-	0	0	11	UWord	r
Multi-line: yes	Axis Number		numMachAxes		

aaVactB	$AA_VACTB[X]				
Axis velocity in basic coordinate system					
mm/min, inch/min, user defined	0.0			Double	r
Multi-line: yes	Axis Number		numMachAxes		

aaVactM	$AA_VACTM[X]				
Axis velocity in machine coordinate system					
mm/min, inch/min, user defined	0.0			Double	r
Multi-line: yes	Axis Number		numMachAxes		

aaVc	$AA_VC[x] x = Axis				
Additive correction value for path feed or axial feed					
mm/min, inch/min, user defined				Double	r
Multi-line: yes	Axis Number		numMachAxes		

acRpValid	$AC_RPVALID[Achse]				
Reapproach position valid 0: Reapproach position not valid 1: Reapproach position valid					
-	0	0	1	UWord	r
Multi-line: yes	Axis Number		numMachAxes		

ackSafeMeasPos					
Confirmation of SI actual position 0 = not confirmed 0x00AC = confirmed					
-				UWord	rw
Multi-line: yes	Axis Number		numMachAxes		

actCouppPosOffset	$VA_COUP_OFFS[x] x = Axis				S3
Position offset of an axis to a leading axis / leading spindle (actual value)					
mm, inch, degree, user defined		0	360	Double	r
Multi-line: yes	Axis Number		numMachAxes		

actFeedRate					S5
Actual value of axis-specific feedrate, if the axis is a positioning axis.					
mm/min, inch/min, user defined				Double	r
Multi-line: yes	Axis Number		numMachAxes		

actIndexAxPosNo					
Current indexing position number 0 = no indexing position >0 = indexing position number					
-				UWord	r
Multi-line: yes	Axis Number		numMachAxes		

actSpeedRel					
Actual value of rotary speed (referring to the maximum speed in %), for linear drives actual value of the velocity.					
%				Double	r
Multi-line: yes	Axis Number		numMachAxes		

actValResol					
Actual value resolution. The physical unit is defined in measUnit (in this module)					
mm, inch, degree, user defined				Double	r
Multi-line: yes	Axis Number		numMachAxes		

activeSvOverride					
Currently active SG override factor in the NCK					
-	-1	-1	100	Long Integer	r
Multi-line: yes	Axis Number		numMachAxes		

amSetupState					
State variable of the PI Service Automatic set-up of an asynchronous motor 0 = inactive 1 = wait for PLC enable 2 = wait for key NC-start 3 = active 4 = stopped by Servo + fine code in the upper byte 5 = stopped by 611D + fine code in the upper byte 6 = stopped by NCK + fine code in the upper byte					
-	0	0	0xff06	UWord	r
Multi-line: yes	Axis Number		numMachAxes		

axComp					
Sum of axis-specific compensation values (CEC Cross Error compensation and temperature compensation). The physical unit is defined in measUnit (in this module).					
mm, inch, degree, user defined				Double	r
Multi-line: yes	Axis Number		numMachAxes		

axisActiveInChan					
Flag indicating whether axis is active in this channel 0 = inactive 1 = active					
-				UWord	r
Multi-line: yes	Axis Number		numMachAxes		

axisFeedRateUnit					
Unit of axial feedrate 0 = mm/min 1 = inch/min 2 = degree/min					
-				UWord	r
Multi-line: yes	Axis Number		numMachAxes		

chanAxisNoGap					
Display of existing axis, i.e. no axis gap in channel. 0: Axis does not exist 1: Axis does exist					
-	0	0	1	UWord	r
Multi-line: yes	Axis Number		numMachAxes		

chanNoAxisIsActive					
Channel number in which the channel axis is currently active 0 = axis is not assigned to any channel 1 to maxnumChannels (Area.:N / Module:Y) = channel number					
-				UWord	r
Multi-line: yes	Axis Number		numMachAxes		

clampStatus					
Axis is connected (VDI input signal) Bit 0 = 1: Axis is connected					
-	0	0	1	UWord	r
Multi-line: no			numMachAxes		

cmdContrPos					
Desired value of position after fine interpolation					
mm, inch, degree, user defined				Double	r
Multi-line: yes	Axis Number		numMachAxes		

cmdCouppPosOffset	$AA_COUP_OFFS[x] x = Axis				S3
Position offset of an axis referring to the leading axis / leading spindle (desired value)					
mm, inch, degree, user defined		0	360	Double	r
Multi-line: yes	Axis Number		numMachAxes		

cmdFeedRate					
Desired value of axis-specific feedrate for a positioning axis.					
mm/min, inch/min, user defined				Double	r
Multi-line: yes	Axis Number		numMachAxes		

cmdSpeedRel					
Speed setpoint (as % of the maximum speed), velocity setpoint in the case of linear motors.					
%				Double	r
Multi-line: yes	Axis Number		numMachAxes		

contrConfirmActive					
Controller enable 0 = no controller enable 1 = controller enable					
-				UWord	r
Multi-line: yes	Axis Number		numMachAxes		

contrMode					
Identifier for controller mode servo 0 = position control 1 = speed control 2 = stop 3 = park 4 = follow-up (set the mode through VDI interface and partly through part program)					
-				UWord	r
Multi-line: yes	Axis Number		numMachAxes		

displayAxis	$MC_DISPLAY_AXIS Bit16-31				
Identifier indicating whether axis is displayed by HMI as a machine axis. 0 = Do not display at all 0xFFFF = Always display everything bit 0 = Display in actual-value window bit 1 = Display in reference point window bit 2 = Display in Preset / Basic offset / Scratching bit 3 = Display in handwheel selection					
-	0xFFFF	0	0xFFFF	UWord	r
Multi-line: yes	Axis Number		numMachAxes		

distPerDriveRevol					
Rotary drive: Load-side path corresponding to one revolution of the drive. Is returned in the unit of the internal computational resolution INT_INCR_PER_MM (for linear axes) or INT_INCR_PER_DEG (for rotary axes / spindles) taking into account gear ratios etc. In the case of linear axes, the pitch of the ball screw is also included in the calculation. In the case of linear motors, a fixed value of "1mm" is used for the ball screw pitch instead of the non-existent ball screw.					
mm, inch, degree, user defined				Double	r
Multi-line: yes	Axis Number		numMachAxes		

drfVal					
DRF value					
-	0			Double	r
Multi-line: yes	Axis Number		numMachAxes		

drive2ndTorqueLimit					
2nd torque limit. With linear motors: 2nd force limit 0 = inactive 1 = active					
-				UWord	r
Multi-line: yes	Axis Number		numMachAxes		

driveActMotorSwitch					
Actual motor wiring (star/delta) 0 = star 1 = delta					
-				UWord	r
Multi-line: yes	Axis Number		numMachAxes		

driveActParamSet					
Number of the actual drive parameter set					
-		1	8	UWord	r
Multi-line: yes	Axis Number		numMachAxes		

driveClass1Alarm					
Message ZK1 drive alarm 0 = no alarm set 1= alarm set (fatal error occured)					
-				UWord	r
Multi-line: yes	Axis Number		numMachAxes		

driveContrMode					
Control mode of drive 0 = current control 1 = speed control					
-				UWord	r
Multi-line: yes	Axis Number		numMachAxes		

driveCoolerTempWarn					
Heatsink temperature monitoring 0 = temperature OK 1 = overtemperature					
-				UWord	r
Multi-line: yes	Axis Number		numMachAxes		

driveDdsPerMds					
Number of drive data sets assigned to a motor data set. Refer to the SINAMICS S120 Function Manual for more information about drive and motor data sets.					
-				UWord	r
Multi-line: yes	Axis Number		numMachAxes		

driveDesMotorSwitch					
Motor wiring selection (star/delta) 0 = star 1 = delta					
-				UWord	r
Multi-line: yes	Axis Number		numMachAxes		

driveDesParamSet					
Desired parameter set of the drive					
-		1	8	UWord	r
Multi-line: yes	Axis Number		numMachAxes		

driveFastStop					
Ramp-function generator rapid stop 0 = not stopped 1 = stopped					
-				UWord	r
Multi-line: yes	Axis Number		numMachAxes		

driveFreqMode					
I/F mode					
-				UWord	r
Multi-line: yes	Axis Number		numMachAxes		

driveImpulseEnabled					
Enable inverter impulse (checkback signal to impulseEnable) 0 = not enabled 1 = enabled					
-				UWord	r
Multi-line: yes	Axis Number		numMachAxes		

driveIndex					
Drive assignment (logical drive number) 0 = drive does not exist 1 to 15 = logical drive number					
-		0	15	UWord	r
Multi-line: yes	Axis Number		numMachAxes		

driveIntegDisable					
Integrator disable 0 = not disabled 1 = disabled					
-				UWord	r
Multi-line: yes	Axis Number		numMachAxes		

driveLinkVoltageOk					
State of the DC link voltage 0 = OK 1 = not OK					
-				UWord	r
Multi-line: yes	Axis Number		numMachAxes		

driveMotorTempWarn					
Motor temperature warning 0 = temperature OK 1 = overtemperature					
-				UWord	r
Multi-line: yes	Axis Number		numMachAxes		

driveNumCrcErrors					
CRC errors on the drive bus (Transmission errors when writing data to the drive; values may range up to FFFFH) 0 = no error					
-				UWord	r
Multi-line: yes	Axis Number		numMachAxes		

driveParked					
Parking axis 0 = no parking axis 1 = parking axis					
-				UWord	r
Multi-line: yes	Axis Number		numMachAxes		

drivePowerOn					
Drive switched on 0 = drive not switched on 1 = drive switched on					
-				UWord	r
Multi-line: yes	Axis Number		numMachAxes		

driveProgMessages					
Configurable messages (via machine data)					
-				UWord	r
Multi-line: yes	Axis Number		numMachAxes		

driveReady					
Drive ready 0 = drive not ready 1 = drive ready					
-				UWord	r
Multi-line: yes	Axis Number		numMachAxes		

driveRunLevel					
Current state reached during the boot process (range: coarse state (0 to 5) * 100 + fine state (up to 22) Booting the firmware ---> 0 XX entering the configuration ---> 1XX hardware-init, communication-init loading, converting data ---> 2XX changing bus addressing ---> 3XX preparing synchronization ---> 4XX activating interrupt ---> 519 XX ==> fine state					
-				UWord	r
Multi-line: yes	Axis Number		numMachAxes		

driveSetupMode					
Set-up mode 0 = inactive 1 = active					
-				UWord	r
Multi-line: yes	Axis Number		numMachAxes		

driveSpeedSmoothing					
Smoothing the desired value of the rotary speed, for linear drives: smoothing the desired value of the velocity 0 = no smoothing 1 = smoothing					
-				UWord	r
Multi-line: yes	Axis Number		numMachAxes		

effComp1					
Sum of the compensation values for encoder 1. The value results from: temperature compensation, backlash compensation, quadrant error compensation, beam sag compensation, leadscrew error compensation. The physical unit is defined in measUnit (in this module).					
mm, inch, degree, user defined				Double	r
Multi-line: yes	Axis Number		numMachAxes		

effComp2					
Sum of the compensation values for encoder 2. The value results from: temperature compensation, backlash compensation, quadrant error compensation, beam sag compensation, leadscrew error compensation. The physical unit is defined in measUnit (in this module).					
mm, inch, degree, user defined				Double	r
Multi-line: yes	Axis Number		numMachAxes		

enc1IsOn					
Operating status of position measuring system 1 0 = Position measuring system 1 parked (or is not configured), may be removed 1 = Position measuring system 1 is passive 2 = Position measuring system 1 is active (e.g. position control)					
-		0	2	UWord	r
Multi-line: yes	Axis Number		numMachAxes		

enc2IsOn					
Operating status of position measuring system 2 0 = Position measuring system 2 parked (or is not configured), may be removed 1 = Position measuring system 2 is passive 2 = Position measuring system 2 is active (e.g. position control)					
-		0	2	UWord	r
Multi-line: yes	Axis Number		numMachAxes		

encChoice					
Active encoder 0 = does not exist 1 = encoder 1 2 = encoder 2					
-				UWord	r
Multi-line: yes	Axis Number		numMachAxes		

fctGenState					
State of the function generator					
-				UWord	r
Multi-line: yes	Axis Number		numMachAxes		

feedRateOvr					
Feedrate override (only if axis is a positioning axis)					
%				Double	r
Multi-line: yes	Axis Number		numMachAxes		

focStat	$AA_FOC[x]				
Current status of "Travel with limited torque" function 0-2 0: FOC not active 1: FOC modal active (programming of FOCON[]) 2: FOC non-modal active (programming of FOC[])					
-	0	0	2	UWord	r
Multi-line: yes	Axis Number		numMachAxes		

fxsInfo	$VA_FXS_INFO[Achse]				
Additional information on travel to fixed stop if $VA_FXS[]=2, or OPI variable fxsStat=2. 0 No additional information available 1 No approach motion programmed 2 Programmed end position reached, movement ended 3 Abort by NC RESET (Reset key) 4 Fixed stop window exited 5 Torque reduction was rejected by drive 6 PLC has canceled enable signals					
-	0	0	6	UWord	r
Multi-line: yes	Axis Number		numMachAxes		

fxsStat	$AA_FXS[x] x = Axis				
State after travelling to fixed stop 0 = normal control, no clamping 1 = fixed stop reached, clamping active 2 = selection failed 3 = selection active 4 = stop detected 5 = deselection active					
-				UWord	r
Multi-line: yes	Axis Number		numMachAxes		

handwheelAss					
Number of handwheel assigned to axis 0 = no handwheel assigned 1 to 3 = handwheel number					
-		0	3	UWord	r
Multi-line: yes	Axis Number		numMachAxes		

impulseEnable					
Impulse enable for drive 0 = not enabled 1 = enabled					
-				UWord	r
Multi-line: yes	Axis Number		numMachAxes		

index					
Absolute axis index referred to MD					
-				UWord	r
Multi-line: yes	Axis Number		numMachAxes		

isDriveUsed					
One or more machine axes are assigned to each drive. The drive can only be controlled at any one time by one of these machine axes. The machine manufacturer makes the selection. The status of the drive control changes dynamically.					
-	0	0	1	UWord	r
Multi-line: yes	Axis Number		numMachAxes		

kVFactor					
position control gain factor					
16.667 1/s				Double	r
Multi-line: yes	Axis Number		numMachAxes		

lag					
Following error = desired value of position after fine interpolation - actual value of position. The physical unit is defined in measUnit (in this module).					
mm, inch, degree, user defined				Double	r
Multi-line: yes	Axis Number		numMachAxes		

logDriveNo					
Drive assignment (logical drive number) 0 = not available 1 to 15 = drive number					
-		0	15	UWord	r
Multi-line: yes	Axis Number		numMachAxes		

measFctState					
State of the probing function					
-				UWord	r
Multi-line: yes	Axis Number		numMachAxes		

measPos1					
Actual value of position for encoder 1. The physical unit is defined in measUnit (in this module).					
mm, inch, degree, user defined				Double	r
Multi-line: yes	Axis Number		numMachAxes		

measPos2					
Actual value of position for encoder 2. The physical unit is defined in measUnit (in this module).					
mm, inch, degree, user defined				Double	r
Multi-line: yes	Axis Number		numMachAxes		

measPosDev					
Actual position difference between the two encoders. The physical unit is defined in measUnit (in this module).					
mm, inch, degree, user defined				Double	r
Multi-line: yes	Axis Number		numMachAxes		

measUnit					
Unit for service values of the drives 0 = mm 1 = inch 2 = grd					
-				UWord	r
Multi-line: yes	Axis Number		numMachAxes		

paramSetNo					
Number of parameter set					
-		1	8	UWord	r
Multi-line: yes	Axis Number		numMachAxes		

preContrFactTorque					
Feed forward control factor torque					
Nm				Double	r
Multi-line: yes	Axis Number		numMachAxes		

preContrFactVel					
Feed forward control factor velocity					
-				Double	r
Multi-line: yes	Axis Number		numMachAxes		

preContrMode					
Feed forward control mode 0 = inactive 1 = velocity feed forward 2 = torque feed forward					
-				UWord	r
Multi-line: yes	Axis Number		numMachAxes		

progIndexAxPosNo					
Programmed indexing position number 0 = no indexing position >0 = indexing position number					
-				UWord	r
Multi-line: yes	Axis Number		numMachAxes		

qecLrnIsOn					
Quadrant error compensation learning active 0 = inactive 1 = Neuronal-QEC learning active 2 = Standard-QEC active 3 = Standard-QEC with adaptation of correction value active 4 = Neuronal-QEC active 5 = Neuronal-QEC with adaptation of measuring time active 6 = Neuronal-QEC with adaptation of decay time of correction value active 7 = Neuronal-QEC with adaptation of measuring time and decay time of correction value active					
-		0	7	UWord	r
Multi-line: yes	Axis Number		numMachAxes		

refPtBusy					
Axis is being referenced 0 = axis is not being referenced 1 = axis is being referenced					
-				UWord	r
Multi-line: yes	Axis Number		numMachAxes		

refPtCamNo					
Reference point cam 0 = no cam approached 1 = cam 1 2 = cam 2 3 = cam 3 4 = cam 4					
-				UWord	r
Multi-line: yes	Axis Number		numMachAxes		

refPtPhase					
Referencing phases 0 = False 1 = Phase 1 2 = Phase 2 3 = Phase 3 4 = Phase 4					
-				UWord	r
Multi-line: yes	Axis Number		numMachAxes		

refPtStatus					
Identifier indicating whether an axis requires referencing and is referenced. Note regarding exchange axes: An exchange axis need only ever be referenced in the channel to which it is currently assigned. A referenced exchange axis is thus logged onto the channel in which it is traversing with value "3" (requires referencing and referenced) and in other channels with value "1" (does not require referencing, but referenced). Set bits have the following meanings: Bit0: current measuring system has been referenced Bit1: current measuring system requires referencing (A busy signal affects the state)					
-	Achsindex			UWord	r
Multi-line: no			numMachAxes		

resolvStatus1					
Encoder status for measuring system 1 0 = Undefined 1 = Referenced 2 = Activated 3 = Limit frequency exceeded					
-				UWord	r
Multi-line: yes	Axis Number		numMachAxes		

resolvStatus2					
Encoder status for measuring system 2 0 = Undefined 1 = Referenced 2 = Activated 3 = Limit frequency exceeded					
-				UWord	r
Multi-line: yes	Axis Number		numMachAxes		

safeAcceptCheckPhase					
Flag for NCK-side acceptance test phase, the human-machine interface can determine which acceptance test phase is present on the NCK. 0: NCK has acceptance test phase inactive = 0 0ACH: NCK has acceptance test phase active					
-	0	0	0ACH	UWord	r
Multi-line: yes	Axis Number		numMachAxes		

safeAcceptTestMode					
SI PowerOn alarms can be acknowledged by Reset in acceptance test mode 0: Acceptance test mode: SI PowerOn alarms cannot be acknowledged by Reset 0ACH: Acceptance test mode: SI PowerOn alarms can be acknowledged by Reset					
-	0	0	0FFH	UWord	rw
Multi-line: yes	Axis Number		numMachAxes		

safeAcceptTestPhase					
Flag for acceptance test phase 0: Acceptance test Wizard not selected, activate NCK-side alarm suppression 0ACH: Dialogs for acceptance test support selected, deactivate NCK-side alarm suppression					
-	0	0	0FFH	UWord	rw
Multi-line: yes	Axis Number		numMachAxes		

safeAcceptTestSE					
Flag for NCK-side SE acceptance test. The human-machine interface starts checking the safe limit positions during the acceptance test 0: NCK has SE acceptance test inactive = 0. The single channel SW limit positions are activated. 0ACH: NCK is to activate SE acceptance test. The single channel SW limit positions are deactivated in this way.					
-	0	0	0ACH	UWord	r
Multi-line: yes	Axis Number		numMachAxes		

safeAcceptTestState					
Flag for acceptance test status, the human-machine interface can determine which acceptance test mode is present on the NCK. 0: NCK has acceptance test mode inactive 0CH: Acceptance test mode not activated because SI PowerOn alarms already present. The causes of the SI PowerOn alarms must be eliminated first. 0DH: Acceptance test mode not activated, the HMI writes invalid values in safeAcceptTestMode to the NCK. 0ACH: NCK has acceptance test mode active					
-	0	0	0FFH	UWord	r
Multi-line: yes	Axis Number		numMachAxes		

safeActPosDiff					
Current actual value difference betw. NCK and drive monitoring channels					
mm, inch, degree, user defined	0.0			Double	r
Multi-line: yes	Axis Number		numMachAxes		

safeActVeloDiff					
Current speed difference between NCK and drive monitoring channels					
mm/min, inch/min, user defined	0.0			Double	r
Multi-line: yes	Axis Number		numMachAxes		

safeActVeloLimit					
Safe limit of actual speed -1 => no actual speed limit active >= 0 => limit of actual speed is active					
mm, inch, degree, user defined		-1		Double	r
Multi-line: no			numMachAxes		

safeActiveCamTrack					
Status Safe cam track (active/inactive) Bit 0 = 1/0: Safe cam track 1 active/inactive Bit 1 = 1/0: Safe cam track 2 active/inactive Bit 2 = 1/0: Safe cam track 3 active/inactive Bit 3 = 1/0: Safe cam track 4 active/inactive					
-	0	0	0xF	UWord	r
Multi-line: no			numMachAxes		

safeAxisType					
Type of axial safety monitoring 0 = No SINUMERIK Safety Integrated active 1 = SINUMERIK Safety Integrated (SPL) active 2 = SINUMERIK Safety Integrated plus (F-PLC) active					
-				UWord	r
Multi-line: yes	Axis Number		numMachAxes		

safeDesVeloLimit					
Safe limit of desired speed -1 => no desired speed limit active >= 0 => desired speed limit is active					
mm, inch, degree, user defined		-1		Double	r
Multi-line: no			numMachAxes		

safeFctEnable					
Safe operation active (Safety Integrated / SPL) 0 = inactive >0 = active					
-				UWord	r
Multi-line: yes	Axis Number		numMachAxes		

safeInputSig					
Safe input signals of the axis					
-				UWord	r
Multi-line: yes	Axis Number		numMachAxes		

safeInputSig2					
Safe input signals part 2					
-		0	0xffff	UWord	r
Multi-line: no			numMachAxes		

safeInputSigDrive					
Safe input signals of the drive					
-				UWord	r
Multi-line: yes	Axis Number		numMachAxes		

safeInputSigDrive2					
Safe input signals of the drive part 2					
-		0	0xffff	UWord	r
Multi-line: no			numMachAxes		

safeMaxVeloDiff					
Maximum speed difference between NCK and drive monitoring channels since last NCK Reset					
mm/min, inch/min, user defined	0.0			Double	r
Multi-line: yes	Axis Number		numMachAxes		

safeMeasPos	$VA_IS[x] x = Axis				
Safe actual position of the axis. The physical unit is defined in the variable measUnit (in this module).					
mm, inch, degree, user defined				Double	r
Multi-line: yes	Axis Number		numMachAxes		

safeMeasPosDrive					
Safe actual position of drive. The physical unit is defined in measUnit (in this module).					
mm, inch, degree, user defined				Double	r
Multi-line: yes	Axis Number		numMachAxes		

safeOutputSig					
Safe output signals of the axis					
-				UWord	r
Multi-line: yes	Axis Number		numMachAxes		

safeOutputSig2					
Safe output signals part 2					
-		0	0xffff	UWord	r
Multi-line: no			numMachAxes		

safeOutputSigCam					
Results of the NCK safe cam evaluation					
-	0	0	3FFFFFFF	Long Integer	r
Multi-line: no			numMachAxes		

safeOutputSigCamDrive					
Results of the drive safe cam evaluation					
-	0	0	3FFFFFFF	Long Integer	r
Multi-line: no			numMachAxes		

safeOutputSigDrive					
Safe output signals of the drive					
-				UWord	r
Multi-line: yes	Axis Number		numMachAxes		

safeOutputSigDrive2					
Safe output signals of the drive part 2					
-		0	0xffff	UWord	r
Multi-line: no			numMachAxes		

safePosCtrlActive					
Axis monitors absolute position 0 = Axis does not monitor absolute position (no SE/SN) 1 = Axis monitors absolute position					
-	0	0	1	UWord	r
Multi-line: no			numMachAxes		

safeStopOtherAxis					
Stop on another axis 0: No stop on another axis 1: Stop on another axis					
-	0	0	1	UWord	r
Multi-line: yes	Axis Number		numMachAxes		

spec					
Axis specification 0 = path axis 1 = positioning axis					
-				UWord	r
Multi-line: yes	Axis Number		numMachAxes		

spindleModePiState					
Status of the spindle mode switchover for this machine axis by PI service _N_SPIMOD 0 = PI service not selected 10 = PI service active 50 = PI service ended successfully 101 = PI service rejected because axis/spindle is not known in the channel 102 = PI service rejected because axis/spindle is not available in the channel 104 = PI service rejected because axis/spindle is not defined as a spindle. 105 = PI service rejected because axis/spindle is a permanently assigned PLC axis/spindle 106 = PI service rejected because axis/spindle is an active following axis/spindle 107 = PI service rejected because axis/spindle is a transformed spindle/axis 108 = PI service rejected because axis/spindle is not available as a command axis 200 = PI service rejected because of an internal error					
-	0	0	999	UWord	r
Multi-line: yes	Axis Number		numMachAxes		

stateContrActive					
State controller (not available) 1 = TRUE 0 = FALSE					
-				UWord	r
Multi-line: yes	Axis Number		numMachAxes		

subSpec					T1
Subspecification 0 = normal axis 1 = indexing axis					
-				UWord	r
Multi-line: yes	Axis Number		numMachAxes		

torqLimit					
Torque limitation value (referring to the nominal value of the drive). For linear motors: force limitation value.					
%				Double	r
Multi-line: yes	Axis Number		numMachAxes		

traceState1					
State of trace channel 1 0 = idle state 1 = recording started 2 = trigger reached 3 = recording ended 4 = recording aborted					
-				UWord	r
Multi-line: yes	Axis Number		numMachAxes		

traceState2					
State of trace channel 2 0 = idle state 1 = recording started 2 = trigger reached 3 = recording ended 4 = recording aborted					
-				UWord	r
Multi-line: yes	Axis Number		numMachAxes		

traceState3					
State of trace channel 3 0 = idle state 1 = recording started 2 = trigger reached 3 = recording ended 4 = recording aborted					
-				UWord	r
Multi-line: yes	Axis Number		numMachAxes		

traceState4					
State of trace channel 4 0 = idle state 1 = recording started 2 = trigger reached 3 = recording ended 4 = recording aborted					
-				UWord	r
Multi-line: yes	Axis Number		numMachAxes		

trackErrContr					
Position controller difference (actual value / desired value of position)					
mm, inch, degree, user defined				Double	r
Multi-line: yes	Axis Number		numMachAxes		

trackErrDiff					
Contour deviation (difference actual value of position and calculated dynamical model)					
mm, inch, degree, user defined				Double	r
Multi-line: yes	Axis Number		numMachAxes		

type					
Axis type 0 = linear axis 1 = rotary axis 2 = spindle					
-				UWord	r
Multi-line: yes	Axis Number		numMachAxes		

vaAbsoluteEnc1DeltaInit	$VA_ABSOLUTE_ENC_DELTA_INIT[1,Achse]				
Enc1: Initial difference					
-	0	0		UDoubleword	r
Multi-line: yes	Axis Number		numMachAxes		

vaAbsoluteEnc1ErrCnt	$VA_ABSOLUTE_ENC_ERR_CNT[1,Achse]				
Enc 1: Error counter for absolute encoder					
-	0	0		Long Integer	r
Multi-line: yes	Axis Number		numMachAxes		

vaAbsoluteEnc1State	$VA_ABSOLUTE_ENC_STATE[1,Achse]				
Enc1: Status of absolute encoder interface Bit0: Interface is active Bit1: Error during parity check Bit2: Error bit alarm Bit3: Error bit CRC error Bit4: Start bit missing with EnDat transfer					
-	0	0		Long Integer	r
Multi-line: yes	Axis Number		numMachAxes		

vaAbsoluteEnc1ZeroMonMax	$VA_ABSOLUTE_ENC_ZERO_MON_MAX[1,Achse]				
Enc1:Maximum of vaEnc1ZeroMonAct with absolute encoder					
-	0	0		UDoubleword	r
Multi-line: yes	Axis Number		numMachAxes		

vaAbsoluteEnc2DeltaInit	$VA_ABSOLUTE_ENC_DELTA_INIT[2,Achse]				
Enc2: Initial difference					
-	0	0		UDoubleword	r
Multi-line: yes	Axis Number		numMachAxes		

vaAbsoluteEnc2ErrCnt	$VA_ABSOLUTE_ENC_ERR_CNT[2,Achse]				
Enc 2: Error counter for absolute encoder					
-	0	0		Long Integer	r
Multi-line: yes	Axis Number		numMachAxes		

vaAbsoluteEnc2State	$VA_ABSOLUTE_ENC_STATE[2,Achse]				
Enc2: Status of absolute encoder interface Bit0: Interface is active Bit1: Error during parity check Bit2: Error bit alarm Bit3: Error bit CRC error Bit4: Start bit missing with EnDat transfer					
-	0	0		Long Integer	r
Multi-line: yes	Axis Number		numMachAxes		

vaAbsoluteEnc2ZeroMonMax	$VA_ABSOLUTE_ENC_ZERO_MON_MAX[2,Achse]				
Enc2:Maximum of vaEnc2ZeroMonAct with absolute encoder					
-	0	0		UDoubleword	r
Multi-line: yes	Axis Number		numMachAxes		

vaCcCompValTotal	$VA_CC_COMP_VAL_TOTAL[Achse]				
Axial OA total compensation value via compile cycles					
mm, inch, degree, user defined	0			Double	r
Multi-line: yes	Axis Number		numMachAxes		

vaCecCompVal	$VA_CEC_COMP_VAL[Achse]				
Axial sag compensation value					
mm, inch, degree, user defined	0			Double	r
Multi-line: yes	Axis Number		numMachAxes		

vaCpSync2	$VA_CPSYNC2[a]				
Second synchronism monitoring of the following axis / spindle 0: Monitoring not active Bit 0 = 1: Monitoring 'Synchronism(2) coarse' active Bit 1 = 1: Synchronism(2) coarse available Bit 2 = 1: Monitoring 'Synchronism(2) fine' active Bit 3 = 1: Synchronism(2) fine available					
-				UWord	r
Multi-line: yes	Axis index of the following axis		numMachAxes		

vaCurr	$VA_CURR[Achse]				
Drive actual current value					
-	0			Double	r
Multi-line: yes	Axis Number		numMachAxes		

vaDistTorque	$VA_DIST_TORQUE[Achse]				
Disturbing torque/max. torque (motor end, York)					
%	0	-100	100	Double	r
Multi-line: yes	Axis Number		numMachAxes		

vaDpe	$VA_DPE[x1]				
Status of power enable of a machine axis 0 - 1					
-	0	0	1	UWord	r
Multi-line: yes	Axis Number		numMachAxes		

vaEnc1CompVal	$VA_ENC1_COMP_VAL[Achse]				
Leadscrew error compensation (LEC) value encoder 1					
mm, inch, degree, user defined	0			Double	r
Multi-line: yes	Axis Number		numMachAxes		

vaEnc1ZeroMonAccessCnt	$VA_ENC_ZERO_MON_ACCESS_CNT[1,Achse]				
Enc1: Update counter					
-	0	0		UDoubleword	r
Multi-line: yes	Axis Number		numMachAxes		

vaEnc1ZeroMonAct	$VA_ENC_ZERO_MON_ACT[1,Achse]				
Enc1: Zero monitoring values					
-	0	0		UDoubleword	r
Multi-line: yes	Axis Number		numMachAxes		

vaEnc1ZeroMonErrCnt	$VA_ENC_ZERO_MON_ERR_CNT[1,Achse]				
Enc 1: Error counter for zero mark monitoring					
-	0	0		Long Integer	r
Multi-line: yes	Axis Number		numMachAxes		

vaEnc1ZeroMonInit	$VA_ENC_ZERO_MON_INIT[1,Achse]				
Enc1:Hardware counter value of the basic zero mark					
-	0	0		UDoubleword	r
Multi-line: yes	Axis Number		numMachAxes		

vaEnc2CompVal	$VA_ENC2_COMP_VAL[Achse]				
Leadscrew error compensation (LEC) value encoder 2					
mm, inch, degree, user defined	0			Double	r
Multi-line: yes	Axis Number		numMachAxes		

vaEnc2ZeroMonAccessCnt	$VA_ENC_ZERO_MON_ACCESS_CNT[2,Achse]				
Enc2: Update counter					
-	0	0		UDoubleword	r
Multi-line: yes	Axis Number		numMachAxes		

vaEnc2ZeroMonAct	$VA_ENC_ZERO_MON_ACT[2,Achse]				
Enc2: Zero monitoring values					
-	0	0		UDoubleword	r
Multi-line: yes	Axis Number		numMachAxes		

vaEnc2ZeroMonErrCnt	$VA_ENC_ZERO_MON_ERR_CNT[2,Achse]				
Enc 2: Error counter for zero mark monitoring					
-	0	0		Long Integer	r
Multi-line: yes	Axis Number		numMachAxes		

vaEnc2ZeroMonInit	$VA_ENC_ZERO_MON_INIT[2,Achse]				
Enc2:Hardware counter value of the basic zero mark					
-	0	0		UDoubleword	r
Multi-line: yes	Axis Number		numMachAxes		

vaFoc	$VA_FOC[Achse]				
Actual status of "ForceControl" 0: ForceControl not active 1: Modal ForceControl active 2: Non-modal ForceControl active					
-	0	0	2	UWord	r
Multi-line: yes	Axis Number		numMachAxes		

vaFxs	$VA_FXS[Achse]				
Actual status of "Travel to fixed stop" 0: Axis not at fixed stop 1: Successful travel to fixed stop 2: Unsuccessful travel to fixed stop 3: Travel to fixed stop selection active 4: Fixed stop has been detected 5: Travel to fixed stop deselection active					
-	0	0	5	UWord	r
Multi-line: yes	Axis Number		numMachAxes		

vaIm	$VA_IM[x]				
Encoder actual value in the machine coordinate system (measured active measuring system)					
mm, inch, degree, user defined	0	0		Double	r
Multi-line: yes	Axis Number		numMachAxes		

vaIm1	$VA_IM1[x]				
Actual value in the machine coordinate system (measured encoder 1)					
mm, inch, degree, user defined	0	0		Double	r
Multi-line: yes	Axis Number		numMachAxes		

valm2	$VA_IM2[x]				
Actual value in the machine coordinate system (measured encoder 2)					
mm, inch, degree, user defined	0	0		Double	r
Multi-line: yes	Axis Number		numMachAxes		

valpoNcChanax	$VA_IPO_NC_CHANAX				
If the machine axis is currently interpolated to this NCU, the channel and channel number which define the interpolator of the axis are output. If the machine axis is currently interpolated to a different NCU, the NCU identifier of the interpolated NCU and the global axis number of the machine axis are output. This global axis number can then be used to transfer the interpolated channel and the channel axis number to the other NCU, with NCU ID 2, with anIpoChanAx[203]. The axis must be assigned to at least one channel on this NCU, otherwise 0 will be returned. The channel is output as from position 100, and the channel axis number is output as from position 1, e.g. 1005 - channel 10 channel axis 5. These values are always lower than 10000. The NCU is output as from position 10000, e.g. 20203: NCU 2 and the global axis number is 203.					
-	0	0		UDoubleword	r
Multi-line: yes	Axis Number		numMachAxes		

vaLagError	$VA_LAG_ERROR[Achse]				
Axis following error					
-	0			Double	r
Multi-line: yes	Axis Number		numMachAxes		

vaLoad	$VA_LOAD[Achse]				
Drive utilization in %					
-	0	-100	100	Double	r
Multi-line: yes	Axis Number		numMachAxes		

vaMotClampingState	$VA_MOT_CLAMPING_STATE				
Starting from the position of the draw-bar (value of the S1), this variable determines the clamping state. A maximum speed is assigned to each state. These are stored in the drive parameters p5043[0..6]. The following values are possible: 0: Sensor not present 1: Initial state, speed limit 0 rpm 2: Alarm, speed limit 0 rpm 3: Tool released / ejected, speed limit see drive parameter p5043[0] 4: Clamping (by spring force), speed limit see drive parameter p5043[1] 5: Releasing (by compressed air), speed limit see drive parameter p5043[2] 6: Releasing (by compressed air), speed limit see drive parameter p5043[3] 7: Clamped with tool, speed limit see drive parameter p5043[4] 8: Clamped with tool, speed limit see drive parameter p5043[4] 9: Further clamping (by spring force), speed limit see drive parameter p5043[5] 10: Clamped without tool, speed limit see drive parameter p5043[6] 11: Alarm, speed limit 0 rpm					
-	0	0		UDoubleword	r
Multi-line: yes	Axis Number		numMachAxes		

vaMotSensorAna	$VA_MOT_SENSOR_ANA				
This variable determines the analog measured value of sensor S1. At a resolution of 1 mV, the analog value 0 - 10 V is mapped by a maximum of +10000 increments.					
-	0	0		UDoubleword	r
Multi-line: yes	Axis Number		numMachAxes		

vaMotSensorConf	$VA_MOT_SENSOR_CONF				
The configuration of the motor sensors can be queried with this variable. The variable is bit-coded, and has the following meanings: Bit0 = 1: Sensor system present. Bit1 = 1: Sensor S1 present. Analog measured value for position of the draw-bar. Bit2 = 0: Bit3 = 0: Bit4 = 1: Sensor S4 present. Digital value for the piston end position Bit5 = 1: Sensor S5 present. Digital value for the angular position of the shaft.					
-	0	0		UDoubleword	r
Multi-line: yes	Axis Number		numMachAxes		

vaMotSensorDigi	$VA_MOT_SENSOR_DIGI				
This variable determines the states of the digital sensors S4 and S5. The variable is bit-coded, and has the following meanings: Bit0 = 0: Bit1 = 0: Bit2 = 0: Bit3 = 0: Bit4 = 1: Sensor S4 piston end position Bit5 = 1: Sensor S5 angular position of the shaft					
-	0	0		UDoubleword	r
Multi-line: yes	Axis Number		numMachAxes		

vaPosctrlMode	$VA_POSCTRL_MODE[Achse]				
Position controller mode 0: Position control 1: Speed control 2: Holding 3: Parking 4: Tracking					
-	0	0	4	UWord	r
Multi-line: yes	Axis Number		numMachAxes		

vaPower	$VA_POWER[Achse]				
Active drive power					
-	0			Double	r
Multi-line: yes	Axis Number		numMachAxes		

vaPressureA	$VA_PRESSURE_A[Achse]				
Pressure on A end of the cylinder in bar (only for Hydraulic)					
-	0			Double	r
Multi-line: yes	Axis Number		numMachAxes		

vaPressureB	$VA_PRESSURE_B[Achse]				
Pressure on B end of the cylinder in bar (only for Hydraulic)					
-	0			Double	r
Multi-line: yes	Axis Number		numMachAxes		

vaSce	$VA_SCE[Achse]				
Status of speed controller enable					
-	0	0	1	UWord	r
Multi-line: yes	Axis Number		numMachAxes		

vaStopSi	$VA_STOPSI[Achse]				
Stop from Safety Integrated -1: No stop 0: Stop A 1: Stop B 2: Stop C 3: Stop D 4: Stop E 5: Stop F 10: Test stop of NC					
-	0			Long Integer	r
Multi-line: yes	Axis Number		numMachAxes		

vaSyncDiff					
Actual value synchronism difference for all types of coupling					
mm, inch, degree, user defined	0			Double	r
Multi-line: yes	Axis Number		numMachAxes		

vaSyncDiffStat	$VA_SYNCDIFF_STAT[Achse]				
Status of the actual value synchronism difference -4: Reserved -3: No valid value in $VA_SYNCDIFF, tangential control -2: No valid value in $VA_SYNCDIFF, master value coupling and simulated master value -1: No valid value in $VA_SYNCDIFF 0: No valid value in $VA_SYNCDIFF, coupling not active 1: Valid value in $VA_SYNCDIFF					
-	0	-4	1	Long Integer	r
Multi-line: yes	Axis Number		numMachAxes		

vaTempCompVal	$VA_TEMP_COMP_VAL[Achse]				
Axial temperature compensation value					
mm, inch, degree, user defined	0			Double	r
Multi-line: yes	Axis Number		numMachAxes		

vaTorque	$VA_TORQUE[Achse]				
Drive torque setpoint					
-	0			Double	r
Multi-line: yes	Axis Number		numMachAxes		

vaTorqueAtLimit	$VA_TORQUE_AT_LIMIT[Achse]				
Status "effective torque equals specified torque limit" 0: Effective torque lower than torque limit 1: Effective torque has reached torque limit					
-	0	0	1	UWord	r
Multi-line: yes	Axis Number		numMachAxes		

vaVactm	$VA_VACTM[x] x = Axis				
Axis velocity actual value on the load side in the MCS					
mm/min, inch/min, user defined				Double	r
Multi-line: yes	Axis Number		numMachAxes		

vaValveLift	$VA_VALVELIFT[Achse]				
Actual valve lift in mm (only for Hydraulic)					
-	0			Double	r
Multi-line: yes	Axis Number		numMachAxes		

vaXfaultSi	$VA_XFAULTSI[Achse]				
Stop F through cross-checking error active Bit 0 set: An actual value error has been discovered in the cross-check between NCK and drive Bit 1 set: Some error has been discovered in the cross-check between NCK and drive and the waiting time until Stop B ($MA_SAFE_STOP_SWITCH_TIME_F) is running or has expired					
-	0			Long Integer	r
Multi-line: yes	Axis Number		numMachAxes		

3.5.3 Area C, Block SGA : State data: Geometry axes in tool offset memory

OEM-MMC: Linkitem /ChannelGeometricAxis/...

All status data that are dependent on machine movement and specified in the workpiece coordinate system are included in module SGA. Supplementary information can be found in module SEGA. The individual variables are defined as arrays where the line index is the number of the axis (assigned to the current channel).The variable "name" in module SGA with the line index in question identifies the axis.

The assignment of the line indices in modules SGA and SEGA is identical.

With SW 5.2 and later, OPI modules SGA and SEGA can be addressed via the geo-axis no. instead of via the channel axis no.:

Line index 1001: 1st geo-axis

Line index 1002: 2nd geo-axis

Line index 1003: 3rd geo-axis

The number of channel axes (geometry, special axes and spindles) can be found in "numMachAxes" in module Y in area C.

actIncrVal					
Active INC weighting of the axis 0 = INC_10000 1 = INC_1000 2 = INC_100 3 = INC_10 4 = INC_1 5 = INC_VAR 6 = INC_JOG_CONT 7 = no increment mode has been set					
-				UWord	r
Multi-line: yes	Axis index		numMachAxes		

actProgPos					
Programmed position, actual value. The physical unit is defined in the variable extUnit (in this module)					
mm, inch, degree, user defined				Double	r
Multi-line: yes	Axis index		numMachAxes		

actToolBasePos					
Tool base position. Physical unit is defined in the variable extUnit (from this module)					
mm, inch, degree, user defined				Double	r
Multi-line: yes	Axis index		numMachAxes		

actToolEdgeCenterPos	$AA_IW[x] x = Axis				
Center point of a cutting edge. Physical unit is defined in the variable extUnit (from this module)					
mm, inch, degree, user defined				Double	r
Multi-line: yes	Axis index		numMachAxes		

cmdProgPos					
Programmed position, desired value. Physical unit is defined in the variable extUnit (in this module)					
mm, inch, degree, user defined				Double	r
Multi-line: yes	Axis index		numMachAxes		

cmdToolBasePos					
Tool base position, desired value . Physical unit is defined in variable extUnit (in this module).					
mm, inch, degree, user defined				Double	r
Multi-line: yes	Axis index		numMachAxes		

cmdToolEdgeCenterPos					
Position of the cutting edge center point. Physical unit is defined in variable extUnit (in this module).					
mm, inch, degree, user defined				Double	r
Multi-line: yes	Axis index		numMachAxes		

extUnit					
Current physical unit of the related geometry axis or auxiliary axis 0 = mm 1 = inch 2 = degree 3 = indexing position 4 = userdef					
-				UWord	r
Multi-line: yes	Axis index		numMachAxes		

name					
Axis name					
-				String [32]	r
Multi-line: yes	Axis index		numMachAxes		

progDistToGo					
Programmed position, distance-to-go. The physical unit is defined in the variable extUnit (in this module).					
mm, inch, degree, user defined				Double	r
Multi-line: yes	Axis index		numMachAxes		

progREPOS					
Programmed position, REPOS. The physical unit is defined in the variable extUnit (in this module).					
mm, inch, degree, user defined				Double	r
Multi-line: yes	Axis index		numMachAxes		

status					
Axis status 0 = travel command in plus direction 1 = travel command in minus direction 2 = exact position coarse reached 3 = exact position fine reached					
-				UWord	r
Multi-line: yes	Axis index		numMachAxes		

subType					
Axis type geometry or auxiliary axis 0 = auxiliary axis 1 = geometry axis 2 = orientation axis					
-				UWord	r
Multi-line: yes	Axis index		numMachAxes		

toolBaseDistToGo					
Tool base distance-to-go. Physical unit is defined in the variable extUnit (in this module)					
mm, inch, degree, user defined				Double	r
Multi-line: yes	Axis index		numMachAxes		

toolBaseREPOS					
Tool base REPOS. Physical unit is defined in the variable extUnit (in this module).					
mm, inch, degree, user defined				Double	r
Multi-line: yes	Axis index		numMachAxes		

toolEdgeCenterDistToGo					
Center point of cutting edge distance-to-go. Physical unit results from the variable extUnit (in this module)					
mm, inch, degree, user defined				Double	r
Multi-line: yes	Axis index		numMachAxes		

toolEdgeCenterREPOS					
Center point of the cutting edge REPOS. Physical unit is defined in the variable extUnit (in this module).					
mm, inch, degree, user defined				Double	r
Multi-line: yes	Axis index		numMachAxes		

varIncrVal					
Setable value for INC_VAR. The physical unit depends on whether the axis is rotary or linear. Linear axes: 1mm rotary axes: 1/1000 degrees					
mm, inch, degree, user defined				Double	rw
Multi-line: yes	Axis index		numMachAxes		

3.5.4 Area C, Block SEGA : State data: Geometry axes in tool offset memory (extension of SGA)

OEM-MMC: Linkitem /ChannelGeometricAxis/...

All status data that are dependent on machine movement and specified in the workpiece coordinate system are combined in module SGA. Supplementary information can be found in module SEGA. The individual variables are defined as arrays where the line index is the number of the axis (assigned to the current channel).The variable "name" in module SGA with the line index in question identifies the axis.

The assignment of the line indices in modules SGA and SEGA is identical.

With SW 5.2 and later, OPI modules SGA and SEGA can be addressed via the geo-axis no. instead of via the channel axis no.:

Line index 1001: 1st geo-axis

Line index 1002: 2nd geo-axis

Line index 1003: 3rd geo-axis

The number of channel axes (geometry, special axes and spindles) can be found in "numMachAxes" in module Y in area C.

aaAcsRel	$AA_ACS_REL[Achse]				
The axial variable $AA_ACS_REL[ax] determines the current relative setpoint in the settable zero coordinate system (SZS) for the corresponding axis. The setpoint corresponds to $AA_IEN[ax], which is transformed by the current relative system frame $P_RELFRAME. The axial positions lie in the relative settable zero system.					
mm, inch, degree, user defined	0			Double	r
Multi-line: yes	Axis index		numMachAxes		

aaDelt	$AA_DELT[x] x = Axis				
Stored axial distance-to-go in the WCS after axial delete-distance-to-go DELDTG(axis) for synchronous actions (Note: for SYNACT only)					
mm, inch, degree, user defined				Double	r
Multi-line: yes	Axis index		numMachAxes		

aaDiamStat	$AA_DIAM_STAT[]				
Status of the diameter programming as a function of configuration and programming Bit 0=0: Diameter programming inactive Bit 0=1: Diameter programming active Bit 1=0: Channel-specific diameter programming					
-	0	0	15	UWord	r
Multi-line: yes	Axis index		numMachAxes		

aaDtbw	$AA_DTBW[x] x = Aaxis				
Axial distance from the beginning of the block in the WCS for positioning and synchronous axes for synchronous motion (Note: for SYNACT only)					
mm, inch, degree, user defined				Double	r
Multi-line: yes	Axis index		numMachAxes		

aaDtepw	$AA_DTEPW[x] x = Axis				
Axial distance-to-go for infeed during oscillation in the WCS (Note: for SYNACT only)					
mm, inch, degree, user defined				Double	r
Multi-line: yes	Axis index		numMachAxes		

aaDtew	$AA_DTEW[x] x = Axis				
Axial distance to the end of the block in the WCS for positioning and synchronous axes for synchronous actions (Note: for SYNACT only)					
mm, inch, degree, user defined				Double	r
Multi-line: yes	Axis index		numMachAxes		

aaDtsb	$AA_DTSB				
Path from the motion starting point in the BCS					
mm, inch, degree, user defined				Double	r
Multi-line: yes	Axis index		numMachAxes		

aaDtsw	$AA_DTSW				
Path from the motion starting point in the WCS					
mm, inch, degree, user defined				Double	r
Multi-line: yes	Axis index		numMachAxes		

aaIb	$AA_IB				
Current BCS setpoint of an axis					
mm, inch, degree, user defined				Double	r
Multi-line: yes	Axis index		numMachAxes		

aaIbCorr	$AA_IB_CORR				
Current BCS setpoint value of an axis including override components					
mm, inch, degree, user defined				Double	r
Multi-line: yes	Axis index		numMachAxes		

aaIbc	$AA_IBC[Achse]				
The axial variable $AA_IBC[ax] determines the setpoint of a Cartesian axis lying between BCS and MCS. Cartesian means that the axis is a linear axis, and it lies plane parallel to a coordinate axis in a clockwise coordinate system. If a geometry axis is still Cartesian at the output of the nth transformation, then this value is returned. The axis identifier used must represent a geometry axis in the BCS, otherwise the variable returns the value 0.					
mm, inch, degree, user defined	0			Double	r
Multi-line: yes	Axis index		numMachAxes		

aaItr1	$AA_ITR[Achse, 1]				
The axial variable determines the current setpoint of an axis at the output of the 1st chained transformation.					
mm, inch, degree, user defined	0			Double	r
Multi-line: yes	Axis index		numMachAxes		

aaItr2	$AA_ITR[Achse, 2]				
The axial variable determines the current setpoint of an axis at the output of the 2nd chained transformation.					
mm, inch, degree, user defined	0			Double	r
Multi-line: yes	Axis index		numMachAxes		

aaItr3	$AA_ITR[Achse, 3]				
The axial variable determines the current setpoint of an axis at the output of the 3rd chained transformation.					
mm, inch, degree, user defined	0			Double	r
Multi-line: yes	Axis index		numMachAxes		

aaIwCorr	$AA_IW_CORR				
Current WCS setpoint value of an axis including override components					
mm, inch, degree, user defined				Double	r
Multi-line: yes	Axis index		numMachAxes		

aaMw	$AA_MW[x] x = Axis				
Latched probe position retransformed in the WCS					
mm, inch, degree, user defined				Double	rw
Multi-line: yes	Axis index		numMachAxes		

aaMw1	$AA_MW1[Achse]				
Access to measurement result of trigger event 1 in the WCS					
mm, inch, degree, user defined				Double	rw
Multi-line: yes	Axis index		numMachAxes		

aaMw2	$AA_MW2[Achse]				
Access to measurement result of trigger event 2 in the WCS					
mm, inch, degree, user defined				Double	rw
Multi-line: yes	Axis index		numMachAxes		

aaMw3	$AA_MW3[Achse]				
Access to measurement result of trigger event 3 in the WCS					
mm, inch, degree, user defined				Double	rw
Multi-line: yes	Axis index		numMachAxes		

aaMw4	$AA_MW4[Achse]				
Access to measurement result of trigger event 4 in the WCS					
mm, inch, degree, user defined				Double	rw
Multi-line: yes	Axis index		numMachAxes		

aaPcsRel	$AA_PCS_REL[Achse]				
The axial variable $AA_PCS_REL[ax] determines the current relative setpoint of the corresponding axis in the workpiece coordinate system (WCS). The setpoint corresponds to $AA_IW[ax], which is transformed by the current relative system frame $P_RELFRAME. The axial positions lie in the relative workpiece coordinate system.					
mm, inch, degree, user defined	0			Double	r
Multi-line: yes	Axis index		numMachAxes		

aaSccStat	$AA_SCC_STAT[]				
Status of the G96/G961/G962 assignment as a function of configuration and programming Bit 0=0: Axis is not assigned to G96/G961/G962 Bit 0=1: Axis is assigned to G96/G961/G962					
-	0	0	1	UWord	r
Multi-line: yes	Axis index		numMachAxes		

aaTOff	$AA_TOFF[ ]				
Value of the superimposed motions which have been retracted in the individual tool directions via $AA_TOFF[ ]					
mm, inch, user defined	0			Double	rw
Multi-line: yes	1000 + geo axis number		1000 + numGeoAxes		

aaTOffLimit	$AA_TOFF_LIMIT[ ]				
Limiting value of the superimposed motion has been achieved in the tool direction via $AA_TOFF[ ] 0 : Limiting value not achieved 1 : Limiting value achieved in positive direction 11 : Limiting value achieved in negative direction					
-	0	0	11	UWord	r
Multi-line: yes	1000 + geo axis number		1000 + numGeoAxes		

aaTOffPrepDiff	$AA_TOFF_PREP_DIFF[ ]				
Difference between the current value of $AA_TOFF[] and the value as the current block was prepared.					
mm, inch, user defined	0			Double	r
Multi-line: yes	1000 + geo axis number		1000 + numGeoAxes		

aaTOffVal	$AA_TOFF_VAL[ ]				
Integrated value of the superimposed motions which have been retracted in the individual tool directions via $AA_TOFF[ ]					
mm, inch, user defined	0			Double	r
Multi-line: yes	1000 + geo axis number		1000 + numGeoAxes		

aaVactW	$AA_VACTW[X]				
Axis velocity in workpiece coordinate system					
mm/min, inch/min, user defined	0.0			Double	r
Multi-line: yes	Axis index		numMachAxes		

acRetpoint	$AC_RETPOINT[x] x = Axis				
Return point on the contour for repositioning					
mm, inch, degree, user defined				Double	r
Multi-line: yes	Axis index		numMachAxes		

actDistToGoEns					
Distance-to-go in the SZS based on the programmed position					
-				Double	r
Multi-line: yes	Axis index		numMachAxes		

actFeedRate					S5
Actual value of axis-specific feedrate, if the axis is a positioning axis.					
mm/min, inch/min, user defined				Double	r
Multi-line: yes	Axis index		numMachAxes		

actFeedRateIpo					
Corresponds to actFeedRate taking into account the revolutional feedrate. For geometry axes the value is reported in the MCS, i.e. related to the geometry axis and not to the machine axis. Associated unit see: axisFeedRateIpoUnit					
mm/min, inch/min, user defined				Double	r
Multi-line: yes	Axis index		numMachAxes		

actProgPosBKS					
Actual value of geometry and orientation axes in basic coordinate system					
mm, inch, degree, user defined	0.0			Double	r
Multi-line: yes	Axis index		numMachAxes		

actToolBasPosBN	$AA_IBN[x] x=Axis				
Actual tool base position in relation to basic zero point (SGA:actToolBasePos without progr. frame and without settable frames)					
mm, inch, degree, user defined				Double	r
Multi-line: yes	Axis index		numMachAxes		

actToolBasPosBNDiam					
Corresponds to actToolBasPosBN with diameter conversion					
mm, inch, degree, user defined				Double	r
Multi-line: yes	Axis index		numMachAxes		

actToolBasPosEN	$AA_IEN[x] x = Axis				
Base position of the active tool relative to the workpiece zero point (SGA:actToolBasePos without programmed frame)					
mm, inch, degree, user defined				Double	r
Multi-line: yes	Axis index		numMachAxes		

actToolBasPosENitc					
corresponds to actToolBasPosEN with $DISPLAY_MODE_POSITION=1					
mm, inch, degree, user defined				Double	r
Multi-line: yes	Axis index		numMachAxes		

actToolBasPosENjmp					
corresponds to actToolBasPosEN with $DISPLAY_MODE_POSITION=0					
mm, inch, degree, user defined				Double	r
Multi-line: yes	Axis index		numMachAxes		

actToolBasePosBasic					
Base position of the active tool in the base system (inch/metrical)					
mm, inch, degree, user defined	0.0			Double	r
Multi-line: yes	Axis index		numMachAxes		

actToolBasePosBasicDiam					
Corresponds to actToolBasePosBasic with diameter conversion					
mm, inch, degree, user defined				Double	r
Multi-line: yes	Axis index		numMachAxes		

actToolBasePosDiam					
Corresponds to actToolBasePos with diameter conversion					
mm, inch, degree, user defined				Double	r
Multi-line: yes	Axis index		numMachAxes		

actToolEdgeCenterPosEns					
Actual position value in relation to the WOS frame as center-point path, i.e. with tool length but without tool radius					
mm, inch, degree, user defined				Double	r
Multi-line: yes	Axis index		numMachAxes		

axisActiveInChan					
Flag indicating whether axis is active in this channel 0 = inactive 1 = active					
-				UWord	r
Multi-line: yes	Axis index		numMachAxes		

axisFeedRateIpoUnit					
Corresponds to axisFeedRateUnit with revolutional feedrate 0 = mm/min 1 = mm/rev. 2 = inch/min 3 = inch/rev. 4 = deg./min 5 = deg./rev.					
-				UWord	r
Multi-line: yes	Axis index		numMachAxes		

axisFeedRateUnit					
Unit of axial feedrate 0 = mm/min 1 = inch/min 2 = degree/min					
-				UWord	r
Multi-line: yes	Axis index		numMachAxes		

cmdFeedRate					
Desired value of axis-specific feedrate for a positioning axis.					
mm/min, inch/min, user defined				Double	r
Multi-line: yes	Axis index		numMachAxes		

cmdFeedRateIpo					
Corresponds to cmdFeedRate taking into account the revolutional feedrate. Associated unit see: axisFeedRateIpoUnit					
-				Double	r
Multi-line: yes	Axis index		numMachAxes		

cmdToolEdgeCenterPosEns					
Programmed SZS position in relation to the WOS frame as center-point path, i.e. with tool length but without tool radius					
mm, inch, degree, user defined				Double	r
Multi-line: yes	Axis index		numMachAxes		

cmdToolEdgeCenterPosEnsS					
Programmed SZS position for block search with calculation in relation to the WOS frame as center-point path, i.e. with tool length but without tool radius Notice! This variable is not available for the variable service, but only for logging in connection with block search events!					
mm, inch, degree, user defined				Double	r
Multi-line: yes	Axis index		numMachAxes		

diamonInfo					
Information whether position values are shown as diameter or radius values. This information is relevant for the following variables of the blocks SGA/SEGA: - cmdToolBasePos - toolBaseDistToGo - toolBaseREPOS - cmdToolEdgeCenterPos - actToolEdgeCenterPos - toolEdgeCenterDistToGo - toolEdgeCenterREPOS - cmdProgPos - actProgPos - progDistToGo - progREPOS - actToolBasPosEN - cmdToolEdgeCenterPosEnsS - actToolEdgeCenterPosEns - actToolBasPosBN - cmdToolBasPosENS - actProgPosBKS - actToolBasePosDiam - actToolBasePosBasicDiam - actToolBasPosBNDiam 0: Diameter programming inactive 1: Diameter programming active					
-	0	0	1	UWord	r
Multi-line: yes	Axis index		numMachAxes		

displayAxis	$MC_DISPLAY_AXIS Bit0-15				
Identifier indicating whether the axis is displayed by the HMI as a geometry or auxiliary axis. 0 = Do not display at all 0xFFFF = Always display everything bit 0 = Display in actual-value window bit 1 = Display in reference point window bit 2 = Display in Preset / Basic offset / Scratching bit 3 = Display in handwheel selection					
-	0xFFFF	0	0xFFFF	UWord	r
Multi-line: yes	Axis index		numMachAxes		

drfVal	$AC_DRF[x] x = Axis				
DRF value. The physical unit is defined in extUnit (in module SGA).					
mm, inch, degree, user defined				Double	r
Multi-line: yes	Axis index		numMachAxes		

effComp					
Sum of all length radius compensations. The physical unit is defined in extUnit (in the SGA block).					
mm, inch, degree, user defined				Double	r
Multi-line: yes	Axis index		numMachAxes		

feedRateOvr					
Feedrate override if axis is a positioning axis. Multiplying override component which is active in addition to the override factors programmed, set via handwheel or via PLC.					
%				Double	r
Multi-line: yes	Axis index		numMachAxes		

geoAxisNr					
Number of the geometry axis If the axis is a geometry axis: 1-3 If the axis is not a geometry axis: 0					
-				UWord	r
Multi-line: yes	Axis index		numMachAxes		

handwheelAss					
Number of handwheel assigned to axis 0 = no handwheel assigned 1 to 3 = handwheel number					
-		0	3	UWord	r
Multi-line: yes	Axis index		numMachAxes		

index					
Absolute axis index referred to MD					
-				UWord	r
Multi-line: yes	Axis index		numMachAxes		

motEnd	$AA_MOTEND				
Current motion end criterion for single-axis interpolation 1 = Motion end with exact stop FINE 2 = Motion end with exact stop COARSE 3 = Motion end with exact stop IPO Stop 4 = Block change in braking ramp of axis motion 5 = Block change in braking ramp of axis motion with tolerance window with reference to setpoint 6 = Block change in braking ramp of axis motion with tolerance window with reference to actual value					
-	1	1	6	UWord	r
Multi-line: yes	Axis index		numMachAxes		

spec					
Axis specification 0 = path axis 1 = positioning axis					
-				UWord	r
Multi-line: yes	Axis index		numMachAxes		

subSpec	MD 30500: INDEX_AX_ASSIGN_POS_TAB				T1
Subspecification, identifies whether an axis is an indexing axis 0 = normal axis 1 = indexing axis					
-				UWord	r
Multi-line: yes	Axis index		numMachAxes		

type					
Axis type 1 = linear axis 2 = rotary axis 3 = spindle					
-				UWord	r
Multi-line: yes	Axis index		numMachAxes		

vaIb	$VA_IB[Achse]	
The variable $VA_IB[ax] determines the encoder position of an axis retransformed into BCS. The BCS value contains all the axial override components (DRF, AA_OFF, external work offset, etc.) and compensation values (CEC, etc.). The positions are only computed		

vaIb	$VA_IB[Achse]				
once per IPO cycle for performance reasons. When a variable is read within one IPO cycle, the value of the variable does not change, although the actual value could have changed. If transformations are active, it must be taken into account that transforming the actual values into the BCS in the IPO cycle can be very time-consuming. In this case, an adequate IPO cycle must be set.					
mm, inch, degree, user defined	0			Double	r
Multi-line: yes	Axis index		numMachAxes		

vaIbc	$VA_IBC[Achse]				
The variable $VA_IBC[geo axis] determines the encoder position of a Cartesian axis lying between BCS and MCS. Cartesian means that the axis is a linear axis and lies plane parallel to a coordinate axis in a clockwise coordinate system. The axis identifier used can be a geometry, channel or machine axis identifier. This identifier must represent a geometry axis in the BCS, otherwise the variable returns the value 0.0. The positions are only computed once per IPO cycle for performance reasons. When a variable is read within one IPO cycle, the value of the variable does not change, although the actual value could have changed. If transformations are active, it must be taken into account that transforming the actual values into the BCS in the IPO cycle can be very time-consuming. In this case, an adequate IPO cycle must be set.					
mm, inch, degree, user defined	0			Double	r
Multi-line: yes	Axis index		numMachAxes		

vaItr1	$VA_ITR[Achse, 1]				
The axial variable determines the current encoder position of an axis at the output of the 1st chained transformation.					
mm, inch, degree, user defined	0			Double	r
Multi-line: yes	Axis index		numMachAxes		

vaItr2	$VA_ITR[Achse, 2]				
The axial variable determines the current encoder position of an axis at the output of the 2nd chained transformation.					
mm, inch, degree, user defined	0			Double	r
Multi-line: yes	Axis index		numMachAxes		

vaItr3	$VA_ITR[Achse, 3]				
The axial variable determines the current encoder position of an axis at the output of the 3rd chained transformation.					
mm, inch, degree, user defined	0			Double	r
Multi-line: yes	Axis index		numMachAxes		

vaIw	$VA_IW[Achse]				
The variable $VA_IW[ax] determines the encoder position of an axis retransformed into BCS. The BCS value contains all the axial override components (DRF, AA_OFF, external work offset, etc.) and compensation values (CEC, etc.). The positions are only computed					

valw	$VA_IW[Achse]				
once per IPO cycle for performance reasons. When a variable is read within one IPO cycle, the value of the variable does not change, although the actual value could have changed. If transformations are active, it must be taken into account that transforming the actual values into the BCS in the IPO cycle can be very time-consuming. In this case, an adequate IPO cycle must be set.					
mm, inch, degree, user defined	0			Double	r
Multi-line: yes	Axis index		numMachAxes		

3.5.5 Area C, Block SSP : State data: Spindle

OEM-MMC: Linkitem /ChannelSpindle/...

All status data that refer to the spindle are combined in the module SSP. The individual variables are defined as arrays where the row index is the number of the spindle (assigned to the current channel). The spindle can be identified by reading the variables "name" or "index" in the same module with the respective row index.

The number of spindles can be read from "numSpindles" in the module Y in the area C.

acConstCutS	$AC_CONSTCUT_S[n]				
Current constant cutting rate					
m/min, ft/min, user defined	0			Double	r
Multi-line: yes	Spindle index		numSpindles		

acSDir	$AC_SDIR[x] x = SpindleNo				
Current direction of spindle rotation in the sense of M3/M4/M5 in the part program, synchronized actions, PLC FC18, PLC DBB30. 3: Clockwise spindle rotation, 4: Counterclockwise spindle rotation, 5: Spindle stop					
-				UWord	r
Multi-line: yes	Spindle index		numSpindles		

acSMode	$AC_SMODE[x]				
Spindle mode 0: No spindle present in channel or spindle is active in another channel or is being used by PLC (FC18) or by synchronized actions. 1: Open-loop speed control mode 2: Positioning mode 3: Synchronous mode 4: Axis mode					
-	1	0	4	UWord	r
Multi-line: yes	Spindle index		numSpindles		

acSType	$AC_S_TYPE[x]				
Type of spindle programming Spindle programmed as: 0 Spindle not programmed 1 Spindle speed, S in rev/min 2 Cutting rate, SVC in m/min or ft/min 3 Constant cutting rate, S in m/min or ft/min 4 Constant grinding wheel peripheral speed, S in m/s or ft/s					
mm/min, inch/min, user defined	0	0		Double	r
Multi-line: no					

acSVC	$AC_SVC[x]				
Programmed, active cutting rate					
mm/min, inch/min, user defined	0	0		Double	r
Multi-line: no					

acSmaxAcc	$AC_SMAXACC[]				
Active acceleration of the spindle This variable returns the active acceleration of the spindle for spindle mode. Bit 14 of $AC_SPIND_STATE (spindle accelerating) is set for the duration of the acceleration to the defined setpoint speed. Bit 15 of $AC_SPIND_STATE (spindle braking) is set for the duration of the braking to the defined setpoint speed. Apart from that, the acceleration-determining machine and setting data can be determined with the system variable $AC_SMAXACC_INFO. If the spindle is in axis mode, then $AC_SMAXACC does not return the current acceleration, instead the machine data (MAX_AX_VELO, MAX_AX_ACCEL, ...) typical for axis mode are active.					
Rev/s2, user defined				Double	r
Multi-line: yes	Spindle index		numSpindles		

acSmaxAccInfo	$AC_SMAXACC_INFO[]				
Identifier for the active spindle acceleration data The system variable provides additional information about $AC_SMAXACC and returns the definitive machine data as identifier/index. The index can be used to determine the active acceleration data on the basis of the following table of existing spindle accelerations. The number range is oriented to the system variable $AC_SMAXVELO_INFO: 0 No acceleration limitation (SERUPRO) 1 Not used 2 Acceleration in speed control mode without position control in the current gear stage MD 35200 GEAR_STEP_SPEEDCTRL_ACCEL 3 Not used 4 Acceleration in the current gear stage based on position control MD 35210 GEAR_STEP_POSCTRL_ACCEL (SPCON, SPOS, poss. with COUPON,..) 5 Not used 6 Not used 7 Not used 8 Not used 9 Acceleration limited by preparation calculations 10 Not used 11 Not used 12 Acceleration limited by axis mode. In the case of a synchronous spindle, the axis mode is enforced by the leading spindle. 13 Acceleration of the superimposed motion of the following spindle limited to the residual dynamics remaining following the coupling 14 Acceleration of the leading spindle due to missing following spindle dynamics or a high transformation ratio 15 Acceleration of the master spindle MD 35212 GEAR_STEP_POSCTRL_ACCEL2 in the case of tapping with G331, G332 (only when the second data set is configured accordingly) 16 Acceleration limited by the configuration of ACC or ACCFXS (synchronized action) 17 Acceleration limited by tool parameter $TC_TP_MAX_ACCEL 18 Not used 19 Acceleration limited in JOG mode by MD 32301 MA_JOG_MAX_ACCEL 20 Acceleration limited due to NCU link 21 Not used 22 Acceleration limited by programming ACCLIMA 23 Not used In oscillation mode (gear stage change), the variable returns the value for spindle mode (speed control mode).					
-				Long Integer	r
Multi-line: yes	Spindle index		numSpindles		

acSmaxVelo	$AC_SMAXVELO[]				
Maximum spindle speed This variable returns the maximum spindle speed for spindle mode. This is formed from the smallest active speed limitation, and cannot be exceeded by speed programming or override > 100%. A speed limitation is indicated by the VDI interface signal DB31..,DBX83.1 'Setpoint speed limited' and by $AC_SPIND_STATE, bit 10 (setpoint speed limited). The cause of the speed limitation (machine, setting data, G code, VDI interface signal etc.) can also be determined with the system variable $AC_SMAXVELO_INFO. If the spindle is in axis mode, then the speed is not limited by $AC_SMAXVELO but instead the machine data (MAX_AX_VELO, ...) typical for axis mode are active.					
rev/min, user defined				Double	r
Multi-line: yes	Spindle index		numSpindles		

acSmaxVeloInfo	$AC_SMAXVELO_INFO[]				
Identifier (index) for the speed-limiting data (machine/setting data, etc.) The system variable provides additional information about $AC_SMAXVELO and returns the definitive data (machine, setting data, G code, VDI interface etc.) as identifier/index. The index can be used to determine the speed-limiting data on the basis of the following table of existing spindle speed limitations. 0 No limitation (SERUPRO) 1 Maximum speed (chuck speed) of spindle MD 35100 SPIND_VELO_LIMIT 2 Speed limited to maximum speed in the current gear stage MD 35130 GEAR_STEP_MAX_VELO_LIMIT 3 Speed limited due to position control to 90% of the minimum from MD 35100 and MD 35130 (SPCON, SPOS, poss. with COUPON,..) 4 Speed limited due to position control to MD 35132 GEAR_STEP_PC_MAX_VELO_LIMIT 5 Speed limited to SD 43220 SPIND_MAX_VELO_G26 (G26 S.. or specification from HMI) 6 Speed limited to MD 35160 SPIND_EXTERN_VELO_LIMIT based on the set VDI interface signal DB31,...DBX3.6 7 Speed limited to SD 43230 SPIND_MAX_VELO_LIMS at constant cutting speed (G96, G961, G962, G97, LIMS) 8 Speed limited to safe speed (SG) by Safety Integrated 9 Speed limited by preparation calculations 10 Limitation by drive parameter SINAMICS p1082 to maximum speed of the drive 11 Speed limitation to MD 36300 ENC_FREQ_LIMIT with functions that require a functioning measuring system, e.g. position control and G95, G96, G97, G973, G33, G34, G35 for the master spindle. The limitation takes into account the encoder speed, the MS arrangement (direct/indirect), MS limiting frequency and the current parameter set 12 Speed limited by axis mode. In the case of a synchronous spindle, axis mode is enforced by the leading spindle. 13 Speed of the superimposed motion of the following spindle limited to the residual dynamics remaining following the coupling. A larger proportion of the superimposed motion can be achieved by reducing the speed of the leading spindle, e.g. by programming G26 S, VELOLIM for the leading spindle or VELOLIMA for the following spindle. The coupling factor must be taken into account. 14 Speed of the leading spindle limited due to missing following spindle dynamics or a high transformation ratio 15 Speed of the master spindle limited to MD 35550 DRILL_VELO_LIMIT in the case of tapping with G331, G332 16 Speed limitation due to the programming of VELOLIM 17 Speed limitation by tool parameter $TC_TP_MAX_VELO 18 Not used 19 Not used 20 Speed limited due to NCU link 21 Speed limited by SD43235 SD_SPIND_USER_VELO_LIMIT, user-controlled speed limitation, e.g. tensioning device, chuck speed 22 Speed limited by the programming of VELOLIMA 23 Speed limited by the clamping state of the tool. In the case of a Weiss spindle, the clamping state can be read from $VA_MOT_CLAMPING_STATE[axn]. In oscillation mode (gear stage change), the variable returns the value for spindle mode (speed control mode).					
-				Long Integer	r
Multi-line: yes	Spindle index		numSpindles		

acSminVelo	$AC_SMINVELO[]				
Minimum spindle speed This variable returns the minimum spindle speed for speed control mode. This is formed from the highest active speed increase, and cannot be undershot by speed programming or override < 100%. A speed increase is indicated by the VDI interface signal DB31..,DBX83.2 'Setpoint speed increased' and by $AC_SPIND_STATE, bit 11 (setpoint speed increased). The cause of the speed increase (machine, setting data, G code, VDI interface signal etc.) can also be determined with the system variable $AC_SMINVELO_INFO. If the spindle is in axis or positioning mode, then the speed is not increased by $AC_SMINVELO.					
rev/min, user defined				Double	r
Multi-line: yes	Spindle index		numSpindles		

acSminVeloInfo	$AC_SMINVELO_INFO[]				
Identifier (index) for the speed-limiting data (machine/setting data, etc.) The system variable provides additional information about $AC_SMAXVELO, and returns the definitive data (machine, setting data, G code, VDI interface, etc.) as identifier/index. The speed-limiting data can be determined with the index from the following table of existing spindle speed limitations. The system variable provides additional information about $AC_SMINVELO, and returns the speed increasing data (machine, setting data) as identifier/index. The speed-increasing data can be determined with the index from the following table of existing spindle speed increases. 0 Not used 1 Not used 2 Lower speed limit (minimum speed) of the current gear stage MD 35140 GEAR_STEP_MIN_VELO_LIMIT 3 Not used 4 Not used 5 Lower speed limit (minimum speed) from SD 43210 SPIND_MIN_VELO_G25 (G25 S.. or specification from HMI) In oscillation mode (gear stage change) and axis mode, the variable returns the values from spindle mode.					
-				Long Integer	r
Multi-line: yes	Spindle index		numSpindles		

acSpindState	$AC_SPIND_STATE[]				
This variable returns the selected states of the spindle. For positioning and axis modes, the variable $AA_INPOS_STATE[Sn] can also be read. Bit 0: "Constant cutting speed active" (VDI interface signal DB31..,DBX84.0) Bit 1: "SUG active" (VDI interface signal DB31..,DBX84.1) Bit 2: "CLGON active" (VDI interface signal DB31..,DBX84.2) Bit 3: "Tapping without compensating chuck" (VDI interface signal DB31..,DBX84.3) Bit 4: "Synchronous mode" (following spindle with synchronous spindle coupling) (VDI interface signal DB31..,DBX84.4) Bit 5: "Positioning mode" (VDI interface signal DB31..,DBX84.5) Bit 6: "Oscillating mode" (gear stage change) (VDI interface signal DB31..,DBX84.6) Bit 7: "Speed control mode" (VDI interface signal DB31..,DBX84.7) Bit 8: "Spindle programmed" (e.g. M3, M4 S.., FC18, ..) (VDI interface signal DB31..,DBX64.4/5 or 6/7) Bit 9: "Speed limit exceeded" (VDI interface signal DB31..,DBX83.0) Bit 10: "Setpoint speed limited" (VDI interface signal DB31..,DBX83.1), active if the speed would be greater than the maximum speed as a result of programming or override ($AC_SMAXVELO) Bit 11: "Setpoint speed increased" (VDI interface signal DB31..,DBX83.2) active if the speed would be less than the minimum speed as a result of programming or override (system variable $AC_SMINVELO) Bit 12: "Spindle in setpoint range" (VDI interface signal DB31..,DBX83.5) Bit 13: "Actual direction of rotation right" (VDI interface signal DB31..,DBX83.7) Bit 14: "Spindle accelerating" remains active as long as the spindle is accelerating to the defined setpoint speed on the setpoint side. Bit 15: "Spindle braking" remains active as long as the spindle is braking to the defined setpoint speed or comes to a standstill on the setpoint side. Bit 16: "Spindle stopped" (VDI interface signal DB31..,DBX61.4) Bit 17: "Tool with dynamic limitation active" (VDI interface signal DB31..,DBX85.0) Bit 18: Reserved Bit 19: "Spindle in position" (VDI interface signal DB31..,DBX85.5) Bit 20: "Position control active" (VDI interface signal DB31..,DBX61.5) Bit 21: "Referenced/synchronized 1" (VDI interface signal DB31..,DBX60.4) Bit 22: "Referenced/synchronized 2" (VDI interface signal DB31..,DBX60.5) Bit 23: Direction of spindle rotation inverted by interface signal "Invert M3/M4" (DB31..,DBX17.6)					
-				Long Integer	r
Multi-line: yes	Spindle index		numSpindles		

actGearStage					
Actual gear stage of spindle					
-				UWord	r
Multi-line: yes	Spindle index		numSpindles		

actSpeed	$AA_S[x] x = SpindleNo				
Spindle speed actual value					
rev/min, user defined				Double	r
Multi-line: yes	Spindle index		numSpindles		

channelNo					
Number of channel in which spindle is configured					
-				UWord	r
Multi-line: yes	Spindle index		numSpindles		

cmdAngPos					
Spindle position (SPOS)					
Degree, user defined				Double	r
Multi-line: yes	Spindle index		numSpindles		

cmdConstCutSpeed					
Constant cutting rate of the master spindle. The requested value for the master spindle differs from SSP:cmdSpeed only if G96 is active. (For a certain OEM customer this variable is now available retroactively in software version 3.2)					
mm/min, inch/min, user defined	0.0			Double	r
Multi-line: yes	Spindle index		numSpindles		

cmdGearStage					
Requested gear stage					
-				UWord	r
Multi-line: yes	Spindle index		numSpindles		

cmdGwps					
Programmed SUG desired value (SUG is the function "constant perimeter speed of grinding wheel")					
m/s, ft/s				Double	r
Multi-line: yes	Spindle index		numSpindles		

cmdSpeed	$P_S[x] x = SpindleNo				
Spindle speed desired value					
rev/min , m/min				Double	r
Multi-line: yes	Spindle index		numSpindles		

driveLoad					
Load					
%				Double	r
Multi-line: yes	Spindle index		numSpindles		

gwpsActive	{$GWPS}				
SUG programming active (SUG is the function "constant perimeter speed of grinding wheel) 0 = inactive 1 = active					
-				UWord	r
Multi-line: yes	Spindle index		numSpindles		

index					
Absolute axis index referred to MD					
-				UWord	r
Multi-line: yes	Spindle index		numSpindles		

name					
Spindle name Note: If several logical spindles are referred to one physical spindle with active spindle conversion and access is made via area N of module SSP2, then the name of the first suitable logical spindle is output.					
-				String [32]	r
Multi-line: yes	Spindle index		numSpindles		

namePhys					
Name of assigned physical spindle, identical to "name" variable.					
-				String [32]	r
Multi-line: yes	Spindle index		numSpindles		

opMode					
Spindle mode 0 = spindle mode 1 = oscillation mode (gear step changeover) 2 = positioning mode 3 = synchronous mode 4 = axis mode					
-				UWord	r
Multi-line: yes	Spindle index		numSpindles		

pSMode	$P_SMODE				
Last programmed spindle mode 0: No spindle configured in channel or spindle is active in another channel or in use by the PLC (FC18) or by synchronized actions. 1: Speed control mode 2: Positioning mode 3: Synchronous mode 4: Axis mode					
-		0	4	UWord	r
Multi-line: yes	Spindle index		numSpindles		

pSModeS					
Last programmed spindle mode with block search 0: No spindle configured in channel or spindle is active in another channel or in use by the PLC (FC18) or by synchronized actions. 1: Speed control mode 2: Positioning mode 3: Synchronous mode 4: Axis mode					
-		0	4	UWord	r
Multi-line: yes	Spindle index		numSpindles		

psModePos					
If the spindle is in positioning mode (pSMode = 2) or axis mode (pSMode = 4), the value actToolEdgeCenterPosEns is returned, otherwise 0.					
-	0			Double	r
Multi-line: yes	Spindle index		numSpindles		

psModePosBKS					
If the spindle is in positioning mode (pSMode = 2) or axis mode (pSMode = 4), the value actProgPosBKS is returned, otherwise 0.					
-	0			Double	r
Multi-line: yes	Spindle index		numSpindles		

psModePosS					
If the spindle is in positioning mode (pSMode = 2) or axis mode (pSMode = 4), the value cmdToolEdgeCenterPosEnsS is returned, otherwise 0.					
-	0			Double	r
Multi-line: yes	Spindle index		numSpindles		

speedLimit					
Current speed limitation for spindle					
rev/min , m/min				Double	r
Multi-line: yes	Spindle index		numSpindles		

speedOvr					
Spindle override					
%				Double	r
Multi-line: yes	Spindle index		numSpindles		

spindleType					
Spindle type 0 = master spindle 1 = no master spindle					
-				UWord	r
Multi-line: yes	Spindle index		numSpindles		

status					
Spindle status Bit0 = following spindle Bit1 = leading spindle Bit2 = master spindle Bit3 = constant cutting rate (G96) active Bit0 = following spindle Bit1 = leading spindle					
-				UWord	r
Multi-line: yes	Spindle index		numSpindles		

turnState					
State of spindle rotation value range to be read via BTSS variable 0 = clockwise 1 = counter-clockwise 2 = stop value range to be read via $ variable 3 = clockwise 4 = counter-clockwise 5 = stop					
-				UWord	r
Multi-line: yes	Spindle index		numSpindles		

<table>
<tr><td>vcSGear</td><td colspan="4">$VC_SGEAR[spino]</td><td></td></tr>
<tr><td colspan="6">Variable $VC_SGEAR[spino] determines the currently active spindle gear stage. $AC_SGEAR[spino] determines the defined gear stage in the main run. During search the actual gear stage may differ from the defined gear stage, because during search the gear stages are not changed. Using $VC_SGEAR[spino] and $AC_SGEAR[spino] it can be checked whether a gear stage change is to be performed after a search.
The following values are possible:
1: 1st gear stage active
....
5: 5th gear stage active
1: 1st gear stage active
....
5: 5th gear stage active</td></tr>
<tr><td>-</td><td>0</td><td>0</td><td>5</td><td>short Integer</td><td>r</td></tr>
<tr><td>Multi-line: no</td><td colspan="2"></td><td colspan="3"></td></tr>
</table>

3.5.6 Area C, Block SSP2 : State data: Spindle

OEM-MMC: Linkitem /ChannelLogicalSpindle/...

All state data that refer to a spindle, if a spindle converter (logical spindles) is active

acConstCutS					
Current constant cutting rate					
m/min, ft/min, user defined	0			Double	r
Multi-line: yes	Logical spindle index		numSpindlesLog		

acSDir					
Programmed direction of spindle rotation in part program, synchronized actions, PLC FC18, PLC DBB30. 3: Clockwise spindle rotation, 4: Counterclockwise spindle rotation, 5: Spindle stop					
-				UWord	r
Multi-line: yes	Logical spindle index		numSpindlesLog		

acSMode					
Spindle mode 0: No spindle present in channel or spindle is active in another channel or is being used by PLC (FC18) or by synchronized actions. 1: Open-loop speed control mode 2: Positioning mode 3: Synchronous mode 4: Axis mode					
-	1	0	4	UWord	r
Multi-line: yes	Logical spindle index		numSpindlesLog		

acSType					
Type of spindle programming Spindle programmed as: 0 Spindle not programmed 1 Spindle speed, S in rev/min 2 Cutting rate, SVC in m/min or ft/min 3 Constant cutting rate, S in m/min or ft/min 4 Constant grinding wheel peripheral speed, S in m/s or ft/s					
mm/min, inch/min, user defined	0	0		Double	r
Multi-line: no					

acSVC					
Programmed, active cutting rate					
mm/min, inch/min, user defined	0	0		Double	r
Multi-line: no					

acSmaxAcc					
Active acceleration of the spindle This variable returns the active acceleration of the spindle for spindle mode. Bit 14 of $AC_SPIND_STATE (spindle accelerating) is set for the duration of the acceleration to the defined setpoint speed. Bit 15 of $AC_SPIND_STATE (spindle braking) is set for the duration of the braking to the defined setpoint speed. Apart from that, the acceleration-determining machine and setting data can be determined with the system variable $AC_SMAXACC_INFO. If the spindle is in axis mode, then $AC_SMAXACC does not return the current acceleration, instead the machine data (MAX_AX_VELO, MAX_AX_ACCEL, ...) typical for axis mode are active.					
Rev/s2, user defined				Double	r
Multi-line: yes	Logical spindle index		numSpindlesLog		

acSmaxAccInfo					
Identifier for the active spindle acceleration data The system variable provides additional information about $AC_SMAXACC and returns the definitive machine data as identifier/index. The index can be used to determine the active acceleration data on the basis of the following table of existing spindle accelerations. The number range is oriented to the system variable $AC_SMAXVELO_IDX: 0 No acceleration limitation (SERUPRO) 1 Not used 2 Acceleration in speed control mode without position control in the current gear stage MD 35200 GEAR_STEP_SPEEDCTRL_ACCEL 3 Not used 4 Acceleration in the current gear stage based on position control MD 35210 GEAR_STEP_POSCTRL_ACCEL (SPCON, SPOS, poss. with COUPON,..) 5 Not used 6 Not used 7 Not used 8 Not used 9 Acceleration limited by preparation calculations 10 Not used 11 Not used 12 Acceleration limited by axis mode. In the case of a synchronous spindle, the axis mode is enforced by the leading spindle. 13 Acceleration of the superimposed motion of the following spindle limited to the residual dynamics remaining following the coupling 14 Acceleration of the leading spindle due to missing following spindle dynamics or a high transformation ratio 15 Acceleration of the master spindle MD 35212 GEAR_STEP_POSCTRL_ACCEL2 in the case of tapping with G331, G332 (only when the second data set is configured accordingly) 16 Acceleration limited by the configuration of ACC or ACCFXS (synchronized action) 17 Acceleration limited by tool parameter $TC_TP_MAX_ACCEL 18 Not used 19 Acceleration limited in JOG mode by MD 32301 MA_JOG_MAX_ACCEL 20 Acceleration limited due to NCU link 21 Not used 22 Acceleration limited by programming ACCLIMA 23 Not used In oscillation mode (gear stage change), the variable returns the value for spindle mode (speed control mode).					
-				Long Integer	r
Multi-line: yes	Logical spindle index		numSpindlesLog		

acSmaxVelo					
Maximum spindle speed This variable returns the maximum spindle speed for spindle mode. This is formed from the smallest active speed limitation, and cannot be exceeded by speed programming or override > 100%. A speed limitation is indicated by the VDI interface signal DB31..,DBX83.1 'Setpoint speed limited' and by $AC_SPIND_STATE, bit 10 (setpoint speed limited). The cause of the speed limitation (machine, setting data, G code, VDI interface signal etc.) can also be determined with the system variable $AC_SMAXVELO_INFO. If the spindle is in axis mode, then the speed is not limited by $AC_SMAXVELO but instead the machine data (MAX_AX_VELO, ...) typical for axis mode are active.					
rev/min, user defined				Double	r
Multi-line: yes	Logical spindle index		numSpindlesLog		

acSmaxVeloInfo					
Identifier (index) for the speed-limiting data (machine/setting data, etc.) The system variable provides additional information about $AC_SMAXVELO and returns the definitive data (machine, setting data, G code, VDI interface etc.) as identifier/index. The index can be used to determine the speed-limiting data on the basis of the following table of existing spindle speed limitations. 0 No limitation (SERUPRO) 1 Maximum speed (chuck speed) of spindle MD 35100 SPIND_VELO_LIMIT 2 Speed limited to maximum speed in the current gear stage MD 35130 GEAR_STEP_MAX_VELO_LIMIT 3 Speed limited due to position control to 90% of the minimum from MD 35100 and MD 35130 (SPCON, SPOS, poss. with COUPON,..) 4 Speed limited due to position control to MD 35132 GEAR_STEP_PC_MAX_VELO_LIMIT 5 Speed limited to SD 43220 SPIND_MAX_VELO_G26 (G26 S.. or specification from HMI) 6 Speed limited to MD 35160 SPIND_EXTERN_VELO_LIMIT based on the set VDI interface signal DB31,...DBX3.6 7 Speed limited to SD 43230 SPIND_MAX_VELO_LIMS at constant cutting speed (G96, G961, G962, G97, LIMS) 8 Speed limited to safe speed (SG) by Safety Integrated 9 Speed limited by preparation calculations 10 Limitation by drive parameter SINAMICS p1082 to maximum speed of the drive 11 Speed limitation to MD 36300 ENC_FREQ_LIMIT with functions that require a functioning measuring system, e.g. position control and G95, G96, G97, G973, G33, G34, G35 for the master spindle. The limitation takes into account the encoder speed, the MS arrangement (direct/indirect), MS limiting frequency and the current parameter set 12 Speed limited by axis mode. In the case of a synchronous spindle, axis mode is enforced by the leading spindle. 13 Speed of the superimposed motion of the following spindle limited to the residual dynamics remaining following the coupling. A larger proportion of the superimposed motion can be achieved by reducing the speed of the leading spindle, e.g. by programming G26 S, VELOLIM for the leading spindle or VELOLIMA for the following spindle. The coupling factor must be taken into account. 14 Speed of the leading spindle limited due to missing following spindle dynamics or a high transformation ratio 15 Speed of the master spindle limited to MD 35550 DRILL_VELO_LIMIT in the case of tapping with G331, G332 16 Speed limitation due to the programming of VELOLIM 17 Speed limitation by tool parameter $TC_TP_MAX_VELO 18 Not used 19 Not used 20 Speed limited due to NCU link 21 Speed limited by SD43235 SD_SPIND_USER_VELO_LIMIT, user-controlled speed limitation, e.g. tensioning device, chuck speed 22 Speed limited by the programming of VELOLIMA 23 Speed limited by the clamping state of the tool. In the case of a Weiss spindle, the clamping state can be read from $VA_MOT_CLAMPING_STATE[axn]. In oscillation mode (gear stage change), the variable returns the value for spindle mode (speed control mode).					
-				Long Integer	r
Multi-line: yes	Logical spindle index		numSpindlesLog		

acSminVelo					
Minimum spindle speed This variable returns the minimum spindle speed for speed control mode. This is formed from the highest active speed increase, and cannot be undershot by speed programming or override < 100%. A speed increase is indicated by the VDI interface signal DB31..,DBX83.2 'Setpoint speed increased' and by $AC_SPIND_STATE, bit 11 (setpoint speed increased). The cause of the speed increase (machine, setting data, G code, VDI interface signal etc.) can also be determined with the system variable $AC_SMINVELO_INFO. If the spindle is in axis or positioning mode, then the speed is not increased by $AC_SMINVELO.					
rev/min, user defined				Double	r
Multi-line: yes	Logical spindle index		numSpindlesLog		

acSminVeloInfo					
Identifier (index) for the speed-limiting data (machine/setting data, etc.) The system variable provides additional information about $AC_SMAXVELO, and returns the definitive data (machine, setting data, G code, VDI interface, etc.) as identifier/index. The speed-limiting data can be determined with the index from the following table of existing spindle speed limitations. The system variable provides additional information about $AC_SMINVELO, and returns the speed increasing data (machine, setting data) as identifier/index. The speed-increasing data can be determined with the index from the following table of existing spindle speed increases. 0 Not used 1 Not used 2 Lower speed limit (minimum speed) of the current gear stage MD 35140 GEAR_STEP_MIN_VELO_LIMIT 3 Not used 4 Not used 5 Lower speed limit (minimum speed) from SD 43210 SPIND_MIN_VELO_G25 (G25 S.. or specification from HMI) In oscillation mode (gear stage change) and axis mode, the variable returns the values from spindle mode.					
-				Long Integer	r
Multi-line: yes	Logical spindle index		numSpindlesLog		

acSpindState					
This variable returns the selected states of the spindle. For positioning and axis modes, the variable $AA_INPOS_STATE[Sn] can also be read. Bit 0: "Constant cutting speed active" (VDI interface signal DB31..,DBX84.0) Bit 1: "SUG active" (VDI interface signal DB31..,DBX84.1) Bit 2: "CLGON active" (VDI interface signal DB31..,DBX84.2) Bit 3: "Tapping without compensating chuck" (VDI interface signal DB31..,DBX84.3) Bit 4: "Synchronous mode" (following spindle with synchronous spindle coupling) (VDI interface signal DB31..,DBX84.4) Bit 5: "Positioning mode" (VDI interface signal DB31..,DBX84.5) Bit 6: "Oscillating mode" (gear stage change) (VDI interface signal DB31..,DBX84.6) Bit 7: "Speed control mode" (VDI interface signal DB31..,DBX84.7) Bit 8: "Spindle programmed" (e.g. M3, M4 S.., FC18, ..) (VDI interface signal DB31..,DBX64.4/5 or 6/7) Bit 9: "Speed limit exceeded" (VDI interface signal DB31..,DBX83.0) Bit 10: "Setpoint speed limited" (VDI interface signal DB31..,DBX83.1), active if the speed would be greater than the maximum speed as a result of programming or override ($AC_SMAXVELO) Bit 11: "Setpoint speed increased" (VDI interface signal DB31..,DBX83.2) active if the speed would be less than the minimum speed as a result of programming or override (system variable $AC_SMINVELO) Bit 12: "Spindle in setpoint range" (VDI interface signal DB31..,DBX83.5) Bit 13: "Actual direction of rotation right" (VDI interface signal DB31..,DBX83.7) Bit 14: "Spindle accelerating" remains active as long as the spindle is accelerating to the defined setpoint speed on the setpoint side. Bit 15: "Spindle braking" remains active as long as the spindle is braking to the defined setpoint speed or comes to a standstill on the setpoint side. Bit 16: "Spindle stopped" (VDI interface signal DB31..,DBX61.4) Bit 17: "Tool with dynamic limitation active" (VDI interface signal DB31..,DBX85.0) Bit 18: Reserved Bit 19: "Spindle in position" (VDI interface signal DB31..,DBX85.5) Bit 20: "Position control active" (VDI interface signal DB31..,DBX61.5) Bit 21: "Referenced/synchronized 1" (VDI interface signal DB31..,DBX60.4) Bit 22: "Referenced/synchronized 2" (VDI interface signal DB31..,DBX60.5) Bit 23: Direction of spindle rotation inverted by interface signal "Invert M3/M4" (DB31..,DBX17.6)					
-				Long Integer	r
Multi-line: yes	Logical spindle index		numSpindlesLog		

actGearStage					
Actual gear stage of spindle					
-				UWord	r
Multi-line: yes	Logical spindle index		numSpindlesLog		

actSpeed					
Spindle speed actual value					
rev/min, user defined				Double	r
Multi-line: yes	Logical spindle index		numSpindlesLog		

channelNo					
Number of channel in which spindle is configured					
-				UWord	r
Multi-line: yes	Logical spindle index		numSpindlesLog		

cmdAngPos					
Spindle position (SPOS)					
Degree, user defined				Double	r
Multi-line: yes	Logical spindle index		numSpindlesLog		

cmdConstCutSpeed					
Constant cutting rate of the master spindle. The requested value for the master spindle differs from SSP:cmdSpeed only if G96 is active. (For a certain OEM customer this variable is now available retroactively in software version 3.2)					
mm/min, inch/min, user defined	0.0			Double	r
Multi-line: yes	Logical spindle index		numSpindlesLog		

cmdGearStage					
Requested gear stage					
-				UWord	r
Multi-line: yes	Logical spindle index		numSpindlesLog		

cmdGwps					
Programmed SUG desired value (SUG is the function "constant perimeter speed of grinding wheel")					
m/s, ft/s				Double	r
Multi-line: yes	Logical spindle index		numSpindlesLog		

cmdSpeed					
Spindle speed desired value					
rev/min , m/min				Double	r
Multi-line: yes	Logical spindle index		numSpindlesLog		

driveLoad					
Load					
%				Double	r
Multi-line: yes	Logical spindle index		numSpindlesLog		

gwpsActive					
SUG programming active (SUG is the function "constant perimeter speed of grinding wheel) 0 = inactive 1 = active					
-				UWord	r
Multi-line: yes	Logical spindle index		numSpindlesLog		

index					
Absolute axis index referred to MD					
-				UWord	r
Multi-line: yes	Logical spindle index		numSpindlesLog		

name					
Spindle name Note: If several logical spindles are referred to one physical spindle with active spindle conversion and access is made via area N of module SSP2, then the name of the first suitable logical spindle is output.					
-				String [32]	r
Multi-line: yes	Logical spindle index		numSpindlesLog		

namePhys					
Name of assigned physical spindle, identical to "name" variable.					
-				String [32]	r
Multi-line: yes	Logical spindle index		numSpindlesLog		

opMode					
Spindle mode 0 = spindle mode 1 = oscillation mode (gear step changeover) 2 = positioning mode 3 = synchronous mode 4 = axis mode					
-				UWord	r
Multi-line: yes	Logical spindle index		numSpindlesLog		

pSMode					
Last programmed spindle mode 0: No spindle configured in channel or spindle is active in another channel or in use by the PLC (FC18) or by synchronized actions. 1: Speed control mode 2: Positioning mode 3: Synchronous mode 4: Axis mode					
-		0	4	UWord	r
Multi-line: yes	Logical spindle index		numSpindlesLog		

pSModeS					
Last programmed spindle mode with block search 0: No spindle configured in channel or spindle is active in another channel or in use by the PLC (FC18) or by synchronized actions. 1: Speed control mode 2: Positioning mode 3: Synchronous mode 4: Axis mode					
-		0	4	UWord	r
Multi-line: yes	Logical spindle index		numSpindlesLog		

psModePos					
If the spindle is in positioning mode (pSMode = 2) or axis mode (pSMode = 4), the value actToolEdgeCenterPosEns is returned, otherwise 0.					
-	0			Double	r
Multi-line: yes	Logical spindle index		numSpindlesLog		

psModePosBKS					
If the spindle is in positioning mode (pSMode = 2) or axis mode (pSMode = 4), the value actProgPosBKS is returned, otherwise 0.					
-	0			Double	r
Multi-line: yes	Logical spindle index		numSpindlesLog		

psModePosS					
If the spindle is in positioning mode (pSMode = 2) or axis mode (pSMode = 4), the value cmdToolEdgeCenterPosEnsS is returned, otherwise 0.					
-	0			Double	r
Multi-line: yes	Logical spindle index		numSpindlesLog		

speedLimit					
Current speed limitation for spindle					
rev/min , m/min				Double	r
Multi-line: yes	Logical spindle index		numSpindlesLog		

speedOvr					
Spindle override					
%				Double	r
Multi-line: yes	Logical spindle index		numSpindlesLog		

spindleType					
Spindle type 0 = master spindle 1 = no master spindle					
-				UWord	r
Multi-line: yes	Logical spindle index		numSpindlesLog		

status					
Spindle status Bit0 = following spindle Bit1 = leading spindle Bit2 = master spindle Bit3 = constant cutting rate (G96) active Bit0 = following spindle Bit1 = leading spindle					
-				UWord	r
Multi-line: yes	Logical spindle index		numSpindlesLog		

turnState					
State of spindle rotation value range to be read via BTSS variable 0 = clockwise 1 = counter-clockwise 2 = stop value range to be read via $ variable 3 = clockwise 4 = counter-clockwise 5 = stop					
-				UWord	r
Multi-line: yes	Logical spindle index		numSpindlesLog		

vcSGear					
Variable $VC_SGEAR[spino] determines the currently active spindle gear stage. $AC_SGEAR[spino] determines the defined gear stage in the main run. During search the actual gear stage may differ from the defined gear stage, because during search the gear stages are not changed. Using $VC_SGEAR[spino] and $AC_SGEAR[spino] it can be checked whether a gear stage change is to be performed after a search. The following values are possible: 1: 1st gear stage active .... 5: 5th gear stage active 1: 1st gear stage active .... 5: 5th gear stage active					
-	0	0	5	short Integer	r
Multi-line: no					

3.5.7 Area C, Block FU : Channel-specific settable frames

OEM-MMC: Linkitem /ChannelUserFrame/...

This only applies if $MC_MM_NUM_USER_FRAMES > 0 and $MN_MM_NUM_GLOBAL_USER_FRAMES = 0, otherwise all settable frames have an NCU-global configuration.

The following frame indices are possible:

0 = G500

1 = G54

2 = G55

3 = G56

4 = G57

5 = G505

6 = G506

:

n = G5n

:

99 = G599

The maximum frame index is: $MC_MM_NUM_USER_FRAMES - 1

The PI service SETUFR has to be called in order to activate the settable frames.

linShift	$P_UIFR[x,y,TR] x=FrameNo,y=Axis				PA
Translation of settable work offset (the physical unit is defined in basicLengthUnit in module Y in area N).					
mm, inch, user defined				Double	rw
Multi-line: yes	Frame index * (numGeoAxes +numAuxAxes) + axis number		$MC_MM_NUM_USER_FRAMES * (numGeoAxes + numAuxAxes)		

linShiftFine	$P_UIFR[x,y,SI] x=FrameNo,y=Axis				
Fine offset with frames, expansion of basic frames and settable frames					
mm, inch, user defined				Double	rw
Multi-line: yes	Frame index * (numGeoAxes +numAuxAxes) + axis number		$MC_MM_NUM_USER_FRAMES * (numGeoAxes + numAuxAxes)		

mirrorImgActive	$P_UIFR[x,y,MI] x = FrameNo,y=Axis				PA
Mirroring enabled in a settable work offset 0 = mirroring not active 1 = mirroring active					
-				UWord	rw
Multi-line: yes	Frame index * (numGeoAxes +numAuxAxes) + axis number		$MC_MM_NUM_USER_FRAMES * (numGeoAxes + numAuxAxes)		

rotation	$P_UIFR[x,y,RT] x = FrameNo,y=Axis				PA
Rotation of a settable work offset					
Degree				Double	rw
Multi-line: yes	Frame index * (numGeoAxes +numAuxAxes) + axis number		$MC_MM_NUM_USER_FRAMES * (numGeoAxes + numAuxAxes)		

rotationCoordinate					
Rotation around a coordinate of a settable zero offset 1: Rotation around first non-existing geometry axis.					
Degree				Double	rw
Multi-line: yes	Frame index * (numGeoAxes +numAuxAxes) + 1		$MC_MM_NUM_USER_FRAMES * (numGeoAxes + numAuxAxes)		

scaleFact	$P_UIFR[x,y,SC] x = FrameNo,y=Axis				PA
Scaling factor of a settable work offset					
-				Double	rw
Multi-line: yes	Frame index * (numGeoAxes +numAuxAxes) + axis number		$MC_MM_NUM_USER_FRAMES * (numGeoAxes + numAuxAxes)		

3.5.8 Area C, Block FA : Active channel-specific frames

OEM-MMC: Linkitem /ChannelActualFrame/...

The following frame indices are available:

0: $P_ACTFRAME = current resulting work offset

1: $P_IFRAME = current settable work offset

2: $P_PFRAME = current programmable work offset

3: EXTFRAME = current external work offset

4: TOTFRAME = current total work offset = sum of ACTFRAME and EXTFRAME

5: $P_ACTBFRAME = current total base frame

6: $P_SETFRAME = current 1st system frame (set actual value, scratching)

7: $P_EXTSFRAME = current 2nd system frame (set actual value, scratching)

8: $P_PARTFRAME = current 3rd system frame (TCARR and PAROT with orientable tool carrier)

9: $P_TOOLFRAME = current 4th system frame (TOROT and TOFRAME)

10: $AC_MEASFRAME = result frame for workpiece and tool measurement

11: $P_WPFRAME = current 5th system frame (workpiece reference points) as from SW $[[SW440000]]

12: $P_CYCFRAME = current 6th system frame (cycles) as from SW $[[SW440000]]

13: $P_TRAFRAME = current 7th system frame (transformation) as from SW $[[SW520000]]

14: $P_ISO1FRAME = current ISO system frame for G51.1 mirroring as from SW $[[SW660000]]

15: $P_ISO2FRAME = current ISO system frame for G68 2DROT as from SW $[[SW660000]]

16: $P_ISO3FRAME = current ISO system frame for G68 3DROT as from SW $[[SW660000]]

17: $P_ISO4FRAME = current ISO system frame for G51 scale as from SW $[[SW660000]]

18: $P_ACSFRAME = current resulting frame for the SZS (ACS) as from SW $[[SW660000]]

19: $P_RELFRAME = current 12th system frame for relative coordinate systems as from SW $[[SW700000]]

20: $P_TRAFRAME_P = current frame of the workpiece component of an active kinematic (orientation) transformation as from SW $[[SW900000]]

21: $P_TRAFRAME_T = current frame of the tool component of an active kinematic (orientation) transformation as from SW $[[SW900000]]

The maximum frame index is 21.

linShift	diverse, siehe Bausteinbescheibung				PA
Translation of an active work offset (the physical unit is defined in basicLengthUnit in module Y in area N).					
mm, inch, user defined				Double	r
Multi-line: yes	Frame index * numMachAxes + axis number		20 * numMachAxes		

linShiftFine	diverse, siehe Bausteinbescheibung				
Fine offset of an active frame					
mm, inch, user defined				Double	rw
Multi-line: yes	Frame index * numMachAxes + axis number		20 * numMachAxes		

mirrorImgActive	diverse, siehe Bausteinbescheibung				PA
Mirroring enabled in an active work offset 0 = mirroring not active 1 = mirroring active					
-				UWord	r
Multi-line: yes	Frame index * numMachAxes + axis number		20 * numMachAxes		

rotation	diverse, siehe Bausteinbescheibung				PA
Rotation of an active work offset					
Degree				Double	r
Multi-line: yes	Frame index * numMachAxes + axis number		20 * numMachAxes		

rotationCoordinate					
Rotation around a coordinate of an active zero offset 1: Rotation around the first non-existing geometry axis.					
Degree				Double	r
Multi-line: yes	Frame index * numMachAxes + 1		20 * numMachAxes		

scaleFact	diverse, siehe Bausteinbescheibung				PA
Scaling factor of an active work offset					
-				Double	r
Multi-line: yes	Frame index * numMachAxes + axis number		20 * numMachAxes		

3.5.9 Area C, Block FE : Channel-specific external frame

OEM-MMC: Linkitem /ChannelExternFrame/...

There is exactly one external frame defined by the PLC.

The maximum frame index is: 0

linShift	$AA_ETRANS[x] x = FrameNo				PA
Translation of external work offset (the physical unit is defined in basicLengthUnit in module Y in area N).					
mm, inch, user defined				Double	rw
Multi-line: yes	Geo axis number		numGeoAxes		

linShiftFine	diverse, siehe Bausteinbescheibung				
Fine offset of external zero offset					
mm, inch, user defined				Double	rw
Multi-line: yes	Geo axis number		numGeoAxes		

mirrorImgActive	diverse, siehe Bausteinbescheibung				PA
Mirroring of an external work offset 0 = mirroring not active 1 = mirroring active					
-				UWord	r
Multi-line: yes	Frame index * numMachAxes + axis number		20 * numMachAxes		

rotation	diverse, siehe Bausteinbescheibung				PA
Rotation of an external work offset					
Degree				Double	r
Multi-line: yes	Frame index * numMachAxes + axis number		20 * numMachAxes		

rotationCoordinate					
Rotation around coordinate of an external zero offset 1: Rotation around first non-existing geometry axis.					
Degree				Double	r
Multi-line: yes	Frame index * numMachAxes + 1		20 * numMachAxes		

scaleFact	diverse, siehe Bausteinbescheibung				PA
Scaling factor of an external work offset					
-				Double	r
Multi-line: yes	Frame index * numMachAxes + axis number		20 * numMachAxes		

3.5.10 Area C, Block FG : Channel-specific frames for grinding applications

OEM-MMC: Linkitem /ChannelGrindingFrame/...

These occur only if $MC_MM_NUM_G_FRAMES > 0 and $MN_MM_NUM_GLOBAL_G_FRAMES = 0, otherwise all grinding frames are configured globally for the NCU.

The following frame indices are possible:

0: GRAME1

1: GRAME2

2: GRAME3

3: GRAME4

...

n: GRAMEn

...

99: GRAME100

The maximum frame index is: $MC_MM_NUM_G_FRAMES - 1

The PI service SETUFR has to be called to activate the grinding frames.

linShift	$P_GFR[x,y,TR] x=FrameNo,y=Axis				PA
Translation of grinding frame (the physical unit is defined in basicLengthUnit in block Y in area N).					
mm, inch, user defined				Double	rw
Multi-line: yes	Frame index * (numGeoAxes +numAuxAxes) + axis number		$MC_MM_NUM_G_FRAMES * (numGeoAxes + numAuxAxes)		

linShiftFine	$P_GFR[x,y,SI] x=FrameNo,y=Axis				
Fine offset with frames, expansion of basic frames and settable frames					
mm, inch, user defined				Double	rw
Multi-line: yes	Frame index * (numGeoAxes +numAuxAxes) + axis number		$MC_MM_NUM_G_FRAMES * (numGeoAxes + numAuxAxes)		

mirrorImgActive	$P_GFR[x,y,MI] x = FrameNo,y=Axis				PA
Mirroring 0 = mirroring not active 1 = mirroring active					
-				UWord	rw
Multi-line: yes	Frame index * (numGeoAxes +numAuxAxes) + axis number		$MC_MM_NUM_G_FRAMES * (numGeoAxes + numAuxAxes)		

rotation	$P_GFR[x,y,RT] x = FrameNo,y=Axis				PA
Rotation					
Degree				Double	rw
Multi-line: yes	Frame index * (numGeoAxes +numAuxAxes) + axis number		$MC_MM_NUM_G_FRAMES * (numGeoAxes + numAuxAxes)		

rotationCoordinate					
Rotation around a coordinate. 1: Rotation around the first non-existent geometry axis.					
Degree				Double	rw
Multi-line: yes	Frame index * (numGeoAxes +numAuxAxes) + 1		$MC_MM_NUM_G_FRAMES * (numGeoAxes + numAuxAxes)		

scaleFact	$P_GFR[x,y,SC] x = FrameNo,y=Axis				PA
Scaling factor					
-				Double	rw
Multi-line: yes	Frame index * (numGeoAxes +numAuxAxes) + axis number		$MC_MM_NUM_G_FRAMES * (numGeoAxes + numAuxAxes)		

3.5.11 Area N, Block FG : NCU global frames for grinding applications

OEM-MMC: Linkitem /NckGrindingFrame/...

These occur only if $MN_MM_NUM_GLOBAL_G_FRAMES > 0.

The following frame indices are possible:

0: GRAME1

1: GRAME2

2: GRAME3

3: GRAME4

...

n: GRAMEn

...

99: GRAME100

The maximum frame index is: $MN_MM_NUM_GLOBAL_G_FRAMES - 1

The PI service SETUFR has to be called to activate the grinding frames.

linShift					PA
Translation (the physical unit is defined in basicLengthUnit in block Y in area N).					
mm, inch, user defined				Double	rw
Multi-line: yes	Frame index * maxnumGlobMachAxes + axis number		$MN_MM_NUM_GLOBAL_G_FRAMES * maxnumGlobMachAxes		

linShiftFine					
Fine offset with frames, expansion of basic frames and settable frames					
mm, inch, user defined				Double	rw
Multi-line: yes	Frame index * maxnumGlobMachAxes + axis number		$MN_MM_NUM_GLOBAL_G_FRAMES * maxnumGlobMachAxes		

mirrorImgActive					PA
Mirroring 0 = mirroring not active 1 = mirroring active					
-				UWord	rw
Multi-line: yes	Frame index * maxnumGlobMachAxes + axis number		$MN_MM_NUM_GLOBAL_G_FRAMES * maxnumGlobMachAxes		

rotation					
Dummy variable, do not use					
-				Double	r
Multi-line: no					

rotationCoordinate					
Dummy variable, do not use					
-				Double	r
Multi-line: yes	Frame index * maxnumGlobMachAxes + 1		$MN_MM_NUM_GLOBAL_G_FRAMES * maxnumGlobMachAxes		

scaleFact					PA
Scaling factor					
-				Double	rw
Multi-line: yes	Frame index * maxnumGlobMachAxes + axis number		$MN_MM_NUM_GLOBAL_G_FRAMES * maxnumGlobMachAxes		

3.6 Status data of the drives

3.6.1 Area H, Block S : Drive-specific state data (MSD)

OEM-MMC: Linkitem	/DriveHsaState/...

During NC operation different internal states occur and system-specific data may change during operation. To distinguish those from system variables, they are classified as state data.

A distinction is made between:

- NCK-specific state data

- Mode-group-specific state data

- Channel-specific state data

- Drive-specific state data (FDD)

- Drive-specific state data (MSD)

Attention: The HS module cannot be addressed with MMC100/EBF/OP030

3.6.2 Area V, Block S : Drive-specific status data (FDD)

OEM-MMC: Linkitem	/DriveVsaState/...

During NC operation different internal states occur and system-specific data may change during operation. To distinguish those from system variables, they are classified as state data.

A distinction is made between:

- NCK-specific state data

- Mode-group-specific state data

- Channel-specific state data

- Drive-specific state data (FDD)

- Drive-specific state data (MSD)

No cyclic service may be set up on variables in this module. Only single variable access is permitted.

3.7 Tool and magazine data

3.7.1 Area C, Block TO : Tool data of the active tool

OEM-MMC: Linkitem /ChannelCompensation/...

Tool data of the active tool

cuttEdgeParam					
Parameter of the active tool cutting edge					
-	0			Double	r
Multi-line: yes	Number of the parameter: 1: Parameter 1 (tool type) 2: Parameter 2 (cutting edge position) 10: Parameter 10 (holder angle or lower limit angle for toroidal millers) 11: Parameter 11 (cutting direction or upper limit angle for toroidal millers) 15: Parameter 15 (wear on tool radius) 16: Parameter 16 (wear on rounding radius) 24: Parameter 24 (clearance angle)		24		

cuttEdgeParamMod					
Modified parameter of the active tool cutting edge. The rotation has been included in the calculation, so the value may not correspond with the original tool data.					
-	0			Double	r
Multi-line: yes	Number of the parameter: 1: Parameter 1 (tool type) 2: Parameter 2 (cutting edge position) 10: Parameter 10 (holder angle or lower limit angle for toroidal millers) 11: Parameter 11 (cutting direction or upper limit angle for toroidal millers) 15: Parameter 15 (wear on tool radius) 16: Parameter 16 (wear on rounding radius) 24: Parameter 24 (clearance angle)		24		

3.7.2 Area T, Block TO : Tool edge data: Offset data

OEM-MMC: Linkitem /ToolCompensation/...

The data module TO is organized as a 2-dimensional variable array.

The module contains the tool edge offset data for all tools. Each element can be addressed via a column and row index:

The column index is the tool number (T-number), i.e. the offset data for all cutting edges of a tool are located in one column. The assignment of a tool to a T-number is given in the module "Tool directory" (TV) in the related area T. If a non-existent tool number is entered for the column index the request is negatively acknowledged.

The number of rows is derived from the number of parameters per tool edge and the number of edges on a tool:

maxZeilenindex = numCuttEdgeParams * /T/TV/numCuttEdges (T-number)

The number of parameters per tool edge "numCuttEdgeParams" is given in module Y in area N. The number of cutting edges "/T/TV/numCuttEdges" is always tool-specific and is given in the module TV in associated area T.

If necessary, several rows can be addressed, so that in one request, for example, all tool edge offset data of a single tool can be read. The offset values of the tool edges are all of the same data type and have the same physical unit.

cuttEdgeParam	$TC_DPCEx[y,z] x = ParamNo y = ToolNo z = EdgeNo				
Replaced by edgeData The value for the tool type is stored internally as an integer.					
-	0			Double	rw
Multi-line: yes	See description edgeData		(numCuttEdgeParams + 1) * maxnumCuttEdges_Tool		

edgeData	$TC_DPx[y,z] x = ParamNo y = ToolNo z = EdgeNo	
Offset value parameters and cutting edge list with D numbers for a tool Part 1: Offset value parameters for a tool edge: Definition of a line index: (EdgeNo - 1) * numCuttEdgeParams + ParameterNo The meaning of each parameter depends on the corresponding tool type. At present, 35 parameters are reserved for each tool edge, however only some are allocated values. The valid parameters, some of which are only optional, can be found in the OPI variable		

edgeData	$TC_DPx[y,z] x = ParamNo y = ToolNo z = EdgeNo	

"extraCuttEdgeParams". To maintain flexibility for future extensions, the variable value 'numCuttEdgeParams' should be used for calculation rather than the fixed number of 35 parameters.
A detailed description of the tool parameters can be found in the 'Tool edge' chapter of the 'Tool Offset (W1)' documentation. The following list is a summary of the tool edge parameters:
Parameter 1: Geometry -- tool type ($TC_DP1)
Parameter 2: Geometry -- cutting edge position ($TC_DP2)
Parameter 3: Geometry -- length 1 ($TC_DP3)
Parameter 4: Geometry -- length 2 ($TC_DP4)
Parameter 5: Geometry -- length 3 ($TC_DP5)
Parameter 6: Geometry -- radius ($TC_DP6)
Parameter 7: Geometry -- corner radius (tool type 700; grooving saw) ($TC_DP7)
Parameter 8: Geometry -- length 4 (tool type 700; grooving saw) ($TC_DP8)
Parameter 9: Geometry -- length 5 ($TC_DP9)
Parameter 10: Geometry -- angle 1 ($TC_DP10)
Parameter 11: Geometry -- angle 2 for tapered milling tools ($TC_DP11)
Parameter 12: Wear -- length 1 ($TC_DP12)
Parameter 13: Wear -- length 2 ($TC_DP13)
Parameter 14: Wear -- length 3 ($TC_DP14)
Parameter 15: Wear -- radius ($TC_DP15)
Parameter 16: Wear -- groove width b / rounding radius ($TC_DP16)
Parameter 17: Wear -- proj. length k ($TC_DP17)
Parameter 18: Wear -- length 5 ($TC_DP18)
Parameter 19: Wear -- angle 1 ($TC_DP19)
Parameter 20: Wear -- angle 2 for tapered milling tools ($TC_DP20)
Parameter 21: Adapter -- length 1 ($TC_DP21)
Parameter 22: Adapter -- length 2 ($TC_DP22)
Parameter 23: Adapter -- length 3 ($TC_DP23)
Parameter 24: Tool clearance angle ($TC_DP24)
Parameter 25: Manual: Cutting speed ($TC_DP25)
ShopMill: Bit-coded value for different states of tools of type 1xx and 2xx ($TC_DP25)
Parameter 26: H number, with ISO mode
Parameter 27: Orientation -- Tool edge orientation
Parameter 28: Orientation -- L1 component of the tool edge orientation
Parameter 29: Orientation -- L2 component of the tool edge orientation
Parameter 30: Orientation -- L3 component of the tool edge orientation
Parameter 31: Orientation -- standardized L1 component of the tool edge orientation
Parameter 32: Orientation -- standardized L2 component of the tool edge orientation
Parameter 33: Orientation -- standardized L3 component of the tool edge orientation
Parameter 34: Number of teeth of a cutting edge
Parameter 35: Basic angle of rotation of the cutting edge
All unlisted parameters up to number 35 are reserved.
Part 2: edgeDNo, associated optional D numbers of cutting edges:
Definition of the line index: ((numCuttEdgeParams * maxnumCuttEdges_Tool) + EdgeNo)
Meaning of the values:
-1: No edge present
1 .. maxDNo: Edge present, associated D number, only when the "any D numbers" function is activated (maxnumCuttEdges_Tool < maxCuttingEdgeNo)
Edge no.: 1 to maxnumCuttEdges_Tool when edge is present, but when the "Assignment of any D numbers" function is not activated on the NC.
0: No D number assigned/assignment cancelled. (In this case, OPI deviates from the NCK variable $TC_DPCE.... $TC_DPCE.... $TC_DPCE = edge number, D = offset number D.

edgeData	$TC_DPx[y,z] x = ParamNo y = ToolNo z = EdgeNo				
If the D number of an edge (variable of module TO) has been set to invalid, the value $TC_DPCE remains unaffected. The edge number specified in the description of the row index matches the parameter $TC_DPCE. The variable D no. defined in the module matches the second index in the offset-specific parameters of type $TC_DPx[T,D],... and others; with x=1,....35.). Notice: This variable is called "cuttEdgeParam" in the non-Windows HMI and the PLC. The value for the tool type is stored internally as an integer.					
mm, inch, user defined	0			Double	rw
Multi-line: yes	See description		(numCuttEdgeParams + 1) * maxnumCuttEdges_Tool		

3.7.3 Area T, Block TD : Tool data: General data

OEM-MMC: Linkitem /ToolData/...

In addition to the tool offset data other tool characteristics are stored for managing the tools. The module TD contains the general data of the tools. The tool characteristics can be addressed via individual multiple-line variables. The variable line index corresponds to the T-number. If non-existent T-numbers are accessed, the request is acknowledged negatively. The module Tool directory (TV) in the associated T area shows which T-numbers are valid.

adaptNo					
Number of adapter defined by system parameter $TC_ADPx which is supporting the tool >0: adapter number 0: no adapter assigned					
-	0	0	numMagPlaces Max	UWord	r
Multi-line: yes	Tool number T		max. T-Nummer		

duploNo	$TC_TP1				FBW
Duplo number (number of replacement tool) In the tool management each tool is explicitly defined both by its identifier and its duplo number. This means that a T-area can only contain tool identifiers with different duplo numbers.					
-	T-Nummer			UWord	r
Multi-line: yes	Tool number T		32000		

numCuttEdges	$P_TOOLND[x] x = ToolNo				
Number of cutting edges of a tool					
-				UWord	r
Multi-line: no			1		

toolIdent	$TC_TP2				FBW
Tool identifier					
-	"<T-Nummer>"			String [32]	r
Multi-line: yes	Tool number T		32000		

toolInMag	$A_TOOLMN[x] x = ToolNo T				
Current magazine in which the tool is located					
-				UWord	r
Multi-line: yes	Tool number T		32000		

toolInMultitool	$A_TOOLMTN[x] x = ToolNo T				
Still to be defined					
-				UWord	r
Multi-line: yes	Tool number T		32000		

toolInMultitoolPlace	$A_TOOLMTLN[x] x = ToolNo T				
Still to be defined					
-				UWord	r
Multi-line: yes	Tool number T		32000		

toolInPlace	$A_TOOLMLN[x] x = ToolNo T				
Current location in which the tool is located					
-				UWord	r
Multi-line: yes	Tool number T		32000		

toolInfo	$TC_TP11				FBW
Tool information for HMI Not currently assigned					
-	0			UWord	rw
Multi-line: yes	Tool number T		32000		

toolMaxAcc	$TC_TP_MAX_ACC				
Maximum angular acceleration of the tool if the value is >0. There is no monitoring if no acceleration limit is defined (=0).					
Rev/s2, user defined				Double	rw
Multi-line: yes	Tool number T		32000		

toolMaxVelo	$TC_TP_MAX_VELO				
Maximum speed of the tool if the value is >0. There is no monitoring if no speed limit is defined (=0).					
rev/min, user defined				Double	rw
Multi-line: yes	Tool number T		32000		

toolMon	$TC_TP9				FBW
Type of tool monitoring 0: no tool monitoring 1: tool life monitoring 2: no. of workpieces monitoring 4: monitoring of edge wear parameters using wear limit 8: monitoring of total offset parameters using wear limit					
-	0			UWord	rw
Multi-line: yes	Tool number T		32000		

toolMyMag	$A_MYMN				
Owner magazine of the tool magazine from which the tool was loaded 0 = the tool is not loaded. If toolInMag is >0 at the same time, the T number will specify a manual tool, or TMMG is not active.					
-	-	0	max. Nummer eines def. Magazins	UWord	r
Multi-line: yes	Tool number T		max. T-Nummer		

toolMyMultitool	$A_MYMTN[x] x = ToolNo T				
Still to be defined					
-				UWord	r
Multi-line: yes	Tool number T		32000		

toolMyMultitoolPlace	$A_MYMTLN[x] x = ToolNo T				
Still to be defined					
-				UWord	r
Multi-line: yes	Tool number T		32000		

toolMyPlace	$A_MYMLN				
Owner magazine of the tool - Magazine location from which the tool was loaded 0 = the tool is not loaded. If toolInPlace is >0 at the same time, the T number will specify a manual tool, a valid magazine location number or TMMG is not active.					
-	-		max. Nummer def. Magazinplatz	UWord	r
Multi-line: yes	Tool number T		max. T-Nummer		

toolProtAreaFile					
Reserved, do not use!					
-				String [32]	r
Multi-line: no					

toolSearch	$TC_TP10				FBW
Type of tool search for replacement tools 0: no strategy 1: next duplo no. 2: shortest path					
-	0			UWord	rw
Multi-line: yes	Tool number T		32000		

toolState					FBW
Tool state 0x0000:0: Not enabled 0x0001:1: Active tool (A) 0x0002:2: Enabled (F) 0x0004:4: Disabled (G) 0x0008:8: Measured (M) 0x0010:16: Prewarning limit reached (V) 0x0020:32: Tool being changed (W) 0x0040:64: Fixed location coded (P) 0x0080:128: Tool was in use (E) 0x0100:256: Tool is in return transport (E) 0x0200:512: Ignore disabled state of tool 0x0400:1024: Tool must be unloaded (R) 0x0800:2048: Tool must be loaded (B) 0x1000:4096: Tool is a master tool (S) 0x2000:8192: Reserved. 0x4000:16384: Tool is involved in a tool change "new" for "old". 0x8000:32768: Tool is being used as a manual tool.					
-	0			UWord	rw
Multi-line: yes	Tool number T		32000		

toolStateL	$TC_TP8				FBW
Tool state large 0x0000: Not enabled 0x0001: Active tool (A) 0x0002: Enabled (F) 0x0004: Disabled (G) 0x0008: Measured (M) 0x0010: Prewarning limit reached (V) 0x0020: Tool being changed (W) 0x0040: Fixed location coded (P) 0x0080: Tool was in use (E) 0x0100: Tool is in return transport (E) 0x0200: Ignore disabled state of tool 0x0400: Tool must be unloaded (R) 0x0800: Tool must be loaded (B) 0x1000: Tool is a master tool (S) 0x2000: Reserved. 0x4000: Tool is involved in a tool change "new" for "old". 0x8000: Tool is being used as a manual tool. 0x10000: Reserved 0x20000: Tool is at a disabled magazine location					
-	0			UDoubleword	rw
Multi-line: yes	Tool number T		32000		

toolplace_spec	$TC_TP7				FBW
Magazine location type of tool					
-	9999			UWord	rw
Multi-line: yes	Tool number T		32000		

toolsize_down	$TC_TP6				FBW
Size downwards in half locations					
-	1			UWord	rw
Multi-line: yes	Tool number T		32000		

toolsize_left	$TC_TP3				FBW
Size to the left in half locations					
-	1			UWord	rw
Multi-line: yes	Tool number T		32000		

toolsize_right	$TC_TP4				FBW
Size to the right in half locations					
-	1			UWord	rw
Multi-line: yes	Tool number T		32000		

toolsize_upper	$TC_TP5				FBW
Size upwards in half locations					
-	1			UWord	rw
Multi-line: yes	Tool number T		32000		

3.7.4 Area T, Block TS : Tool edge data: Monitoring data

OEM-MMC: Linkitem	/ToolSupervision/...

The module TS is organized as a 2-dimensional variable array. The module contains the tool edge monitoring data for all tools. Each element can be addressed via a column and line index:

The column index is the tool number (T-number), i.e. one column contains the monitoring data for all tool edges of a tool. The assignment of a tool to a T-number is given in the module Tool directory (TV) in the associated area T. If a non-existent tool number is specified for the column index, the request is acknowledged negatively.

The number of lines is derived from the number of parameters per tool edge and from the number of tool edges of a tool:

maxZeilenanzahl = numCuttEdgeParams_ts * /T/TV/numCuttEdges (T-number)

The number of parameters per tool edge "numCuttEdgeParams_ts" is given in the module Y in area N. The number of tool edges "/T/TV/numCuttEdges" is always tool specific and can be found in the module TV in associated area T.

If necessary, several lines can be addressed, so that in one request, for example, all tool edge monitoring data of a single tool can be read. The monitoring data of the tool edges are all of the same data type and have the same physical unit.

New tool monitoring modes "Monitoring of wear values" and "Monitoring of total offsets":

3 new parameters are provided for these modes:

P7 = Wear prewarning limit (SW 5.1 and later) ($TC_MOP6)

P8 = Remaining wear (actual value) (SW 5.1 and later) ($TC_MOP5)

P9 = Wear setpoint (SW 5.1 and later) ($TC_MOP15)

data	$TC_MOPx[y,z] x=ParamNo,y=T-Number,z=Edge				
Notice: The variable is not documented for the user! Monitoring data per tool edge Important: This is a 2-dimensional variable. 9 parameters are available for each tool edge. The parameters have the following meaning: P1 = Prewarning limit service life in minutes ($TC_MOP1) P2 = Remaining service life in minutes ($TC_MOP2) P3 = Prewarning limit workpiece number ($TC_MOP3) P4 = Remaining workpiece number ($TC_MOP4) P5 = Desired service life ($TC_MOP11) P6 = Desired workpiece number ($TC_MOP13) P7 = Prewarning limit wear (prewarning limit) ($TC_MOP5) This parameter can only be set if bit 5 of machine data $MN_MM_TOOL_MANAGEMENT_MASK has been set correspondingly. P8 = Remaining wear (actual value) ($TC_MOP6) cannot be written P9 = Desired wear ($TC_MOP15) This parameter can only be set if Bit 5 of machine data $MN_MM_TOOL_MANAGEMENT_MASK has been set correspondingly.					
-	0			Double	rw
Multi-line: yes	(ToolEdgeNo - 1) * numCuttEdgeParams_ts + ParameterNo		numCuttEdgeParams_ts * maxnumCuttEdges_Tool		

3.7.5 Area T, Block TU : Tool data: User-defined data

OEM-MMC: Linkitem /ToolUser/...

(Previous designation: TUD)

The TUD data module is defined as a two-dimensional variable array. The module contains user-defined data for all tools. Each element can be addressed via a column and line index:

The column index is the number of the user-defined tool parameter. The number of tool parameters (columns) is given by the variable "numToolParams_tu" in module Y in area N.

The line index is the tool number. If non-existent tools are accessed, the request is acknowledged negatively.

The user-defined tool data are all of the same type.

data	$TC_TPCx[y] x = ParameterNo y = ToolNo				FBW
User-defined tool parameter. Important: This is a two-dimensional variable, the column index is the parameter number					
-				Double	rw
Multi-line: yes	Tool number T		32000		

3.7.6 Area T, Block TUE : Tool edge data: User-defined data

OEM-MMC: Linkitem /ToolUser/...

(Previous designation: TUO)

The data module TUE is organized as a two-dimensional variable field. The module contains user-defined edge data for all tools. Each element can be addressed via a column and line index:

The column index is the tool number (T number), i.e. the user-defined data for all edges of a tool can be found in one column. The assignment of a tool to a T number can be drawn from the module Tool Directory in te related area T. If a non-existant tool number is entered for the column index, then the job is acknowledged negatively.

The number of lines results from the number of parameters per edge and from the number of edges of a tool:

maxNumberof lines = numCuttEdgeParams_tu * /T/TV/numCuttEdges (T number)

The number of parameters per edge "numCuttEdgeParams_tu" can be drawn from module Y in area N. The number of edges "/T/TV/numCuttEdges" which can be tool-specific, can be drawn from module TV in the related area T.

If required, several lines may be addressed so that - for example - in one job all user-defined edge data of a tool can be read. The data are all of the same type.

edgeData	$TC_DPCx[y,z] x=ParamNo,y=ToolNo z=EdgeNo				FBW
User-defined cutting edge parameter. Important: This is a two-dimensional variable, the column index is the T number					
-				Double	rw
Multi-line: yes	(TooledgeNo - 1) * numCuttEdgeParams_tu + ParameterNo		numCuttEdgeParams_tu * maxnumCuttEdges_Tool		

3.7.7 Area T, Block TG : Tool data: Grinding-specific data

OEM-MMC: Linkitem /ToolGrindingData/...

Special tool data are required for grinding tools. These data are contained in the module TG. They can be addressed via several multiple-row variables. The row index corresponds to the T number. If a non-existent T-number is addressed negative acknowledgement is returned. The module tool directory (TV) in the associated area T shows which T-numbers are valid.

actToolWide	$TC_TPG5				W4
Current width of the grinding wheel					
mm, inch, user defined				Double	rw
Multi-line: yes	Tool number T		32000		

conntectPar	$TC_TPG2				W4
Chaining rule. This parameter (which is bitwise defined) specifies which tool parameters of cutting edge 2 and cutting edge 1 are chained. If the value of any chained parameter is altered, the value of the other chained parameter is automatically adapted. If the following bits are set, the corresponding parameters of D1 and D2 are chained: Bit0: tool type Bit2: geometry length1 Bit3: geometry length2 Bit4: geometry length3 Bit11: wear length1 Bit12: wear length2 Bit13: wear length3 Bit20: base dimension/adapter dimension length1 Bit21: base dimension/adapter dimension length2 Bit22: base dimension/adapter dimension length3 The value is stored internally as an integer.					
-				Double	rw
Multi-line: yes	Tool number T		32000		

drsPath	$TC_TPG_DRSPATH				
Path to the dressing program					
-				String [160]	rw
Multi-line: yes	Tool number T		32000		

drsProgname	$TC_TPG_DRSPROG				
Grinder dressing program name.					
-				String [32]	rw
Multi-line: yes	Tool number T		32000		

inclAngle	$TC_TPG8				W4
Angle of inclination of the inclined grinding wheel in the current plane					
Degree		-90	90	Double	rw
Multi-line: yes	Tool number T		32000		

maxRotSpeed	$TC_TPG6				W4
Maximum rotary speed of the grinding wheel					
rev/min , m/min				Double	rw
Multi-line: yes	Tool number T		32000		

maxTipSpeed	$TC_TPG7				W4
Maximum peripheral speed of the grinding wheel					
mm/min, inch/min, user defined				Double	rw
Multi-line: yes	Tool number T		32000		

minToolDia	$TC_TPG3				W4
Minimum diameter of the grinding wheel					
mm, inch, user defined				Double	rw
Multi-line: yes	Tool number T		32000		

minToolWide	$TC_TPG4				W4
Minimum width of the grinding wheel					
mm, inch, user defined				Double	rw
Multi-line: yes	Tool number T		32000		

paramNrCCV	$TC_TPG9				W4
Compensation parameters for the function SUG ("constant perimeter speed of grinding wheel"). These parameters define which compensation value is to be used for SUG, tool monitoring and centerless grinding. The value always refers to cutting edge D1. 3: length 1 4: length 2 5: length 3 6: radius The value is stored internally as an integer.					
-				Double	rw
Multi-line: yes	Tool number T		32000		

spinNoDress	$TC_TPG1				W4
Spindle number to which the monitoring data and the function SUG ("constant perimeter speed of grinding wheel") refer. The value is stored internally as an integer.					
-				Double	rw
Multi-line: yes	Tool number T		32000		

3.7.8 Area T, Block TMC : Magazine data: Configuration data

OEM-MMC: Linkitem /ToolMagazineConfiguration/...

Each tool magazine is configured with several parameters during start-up. These configuration data together with the state information are combined in the module TMC.

magBLMag					W4
Number of the internal load magazine					
-				UWord	r
Multi-line: no					

magCBCmd					W4
Command for magazine execution 1: Find_empty location_loading 2: Tool_MOVE					
-				UWord	r
Multi-line: no					

magCBCmdState					W4
Command state of the magazine (for magCBCmd) 1: started 2: running 3: end correct 4: end with error					
-				UWord	r
Multi-line: no					

magCBIdent	$TC_MAMP1				W4
Identifier of the magazine					
-				String [32]	r
Multi-line: no					

magCMCmdPar1					W4
Return variable for the command MagCBCmd In case of a succesfull return, the return value is the magazine number. If an error occurs, an error number is set.					
-				UWord	r
Multi-line: no					

magCMCmdPar2					W4
Return variable for the command MagCBCmd In case of a succesfull return, the return value is the place number. If an error occurs an error number is set.					
-				UWord	r
Multi-line: no					

magRPlaces					W4
Total number of real magazine locations (incl. buffer and loading locations)					
-				UWord	r
Multi-line: no					

magSearch	$TC_MAMP2				W4
Type of tool search. This variable is bitwise defined. A set bit has the following meaning: Bit0: search active tool Bit1: search tool by shortest path Bit8: begin search at first location (forwards) Bit9: begin search at current location forwards Bit10: begin search at last location (backwards) Bit11: begin search at current location backwards Bit12: begin search at current location symmetrically					
-				UWord	r
Multi-line: no					

magVPlaces					W4
Number of defined locations for the control block Number of virtual locations (without buffer and loading locations) for all real magazines in this area unit					
-				UWord	r
Multi-line: no					

magZWMag					W4
Number of internal buffer magazine					
-				UWord	r
Multi-line: no					

modeWearGroup	$TC_MAMP3				
Definition of strategies relating to wear group. The value is bit-coded. Default setting = 0. Effects on tool status Bit Value Meaning 0 0 When a wear group is activated internally, the status of the tools it contains remains unchanged. 1 When a wear group is activated internally, the status of the tools it contains changes. One tool from each tool group is set to the "active" state. 1 0 When a wear group is disabled internally, the status of the tools it contains remains unchanged. 1 When a wear group is disabled internally, the status of the tools it contains changes. The "active" status is cancelled for all tools. "Internally" in this instance means disabling or activation due to a tool change necessitating a change in the wear group. Activating/ disabling the appropriate tools after writing system parameters or via OPI. 2... Reserved ... Reserved 7... Reserved Search strategy for next wear group: Bit Value Meaning 8 0 Find the next possible wear group 1 Find the wear group with the next-higher group number which can be activated 9... Reserved ... Reserved 11... Reserved Search strategy within the wear group for the tool to be activated Bit Value Meaning 12 0 Lowest possible duplo number 1 Lowest possible magazine location number 13... Reserved ... Reserved 15... Reserved The active wear group can be disabled completely by negating the contents of $TC_MAP9. It is also possible to disable any selected wear group by negating $TC_MPP5 for a magazine location assigned to the relevant wear group. See also system parameter magWearCompoundNo / $TC_MAP9 (active wear group number) and wear group number of magazine location / $TC_MPP5.					
-				UWord	r
Multi-line: yes	1				

3.7.9 Area T, Block TMV : Magazine data: Directory

OEM-MMC: Linkitem /ToolMagazineCatalogue/...

The data module TMV can be used for the following purposes:

1. To display all magazines. The most important magazine information is combined in the module TMV. The existing magazines are sorted in ascending order according to the magazine number without gaps. This means that variables that are defined in this module as one-dimensional arrays contain all magazine information without any gaps. The row index with which a specific array can be addressed does not refer to the magazine number, it is merely a serial number. Inserting/deleting a magazine dynamically changes the contents of a row.

2. To access magazine data in the modules TM, TP and TPM. Before accessing an element in the above modules, the module TV should be consulted to determine which tools have actually been defined.

magVIdent					
Identifier of the magazine					
-				String [32]	r
Multi-line: yes	MagazineNo		numMagsMax		

magVNo					
Number of the magazine					
-				UWord	r
Multi-line: yes	MagazineNo		numMagsMax		

numActMags					
Number of magazines in the modules TMV and TM					
-			numMagsMax	UWord	r
Multi-line: no					

3.7.10 Area T, Block TM : Magazine data: General data

OEM-MMC: Linkitem /ToolMagazineDescription/...

This module contains the information for the available tool magazines.

magActPlace	$TC_MAP8				
Current magazine position Location number of tool change position					
-				UWord	rw
Multi-line: yes	Magazine number		numMagsMax		

magCmd					
Command for magazine execution 1: Find_empty location_loading 2: Tool_MOVE					
-				UWord	r
Multi-line: yes	Magazine number		numMagsMax		

magCmdPar1					
Command parameter of the magazine In case of a succesfull return, the return value is the magazine number. If an error occurs, an error number is set.					
-				UWord	r
Multi-line: yes	Magazine number		numMagsMax		

magCmdPar2					
Command parameter of the magazine In case of a succesfull return, the return value is the place number. If an error occurs an error number is set.					
-				UWord	r
Multi-line: yes	Magazine number		numMagsMax		

magCmdState					
Command state of the magazine 1: started 2: running 3: end correct 4: end with error					
-				UWord	r
Multi-line: yes	Magazine number		numMagsMax		

magDim	$TC_MAP6				FBW
Dimension of the magazine, number of magazine lines in the box magazine Applies to box magazines (magKind = 5) number of lines. For all other magazine types the value is 1.					
-				UWord	r
Multi-line: yes	Magazine number		numMagsMax		

magDim2	$TC_MAP7				
Dimension of the magazine, number of columns in the box magazine magDim * magDim2 = magNrPlaces					
-	1	1	600	UWord	r
Multi-line: yes	Magazine number		numMagsMax		

magIdent	$TC_MAP2				FBW
Identifier of the magazine					
-				String [32]	r
Multi-line: yes	Magazine number		numMagsMax		

magKind	$TC_MAP1				FBW
Type of the magazine 1 = chain 3 = revolver 5 = box magazine 7 = internal magazine tool buffer 9 = internal magazine loading stations					
-	0			UWord	r
Multi-line: yes	Magazine number		numMagsMax		

magLink1	$TC_MAP4				FBW
Chaining 1 of the magazine to the following magazine. Number to (next) background magazine. Can be used with chain, revolver and box magazines (magKind = 1,3 or 5)					
-	-1			UWord	r
Multi-line: yes	Magazine number		numMagsMax		

magLink2	$TC_MAP5				FBW
Chaining 2 of the magazine to the previous magazine. Backward chaining of background magazines. Can be used for chaining to chain, revolver and box magazines (magKind = 1, 3 or 5)					
-	-1			UWord	r
Multi-line: yes	Magazine number		numMagsMax		

magNo					
Number of the magazine					
-		1	numMagsMax	UWord	r
Multi-line: yes	Magazine number		numMagsMax		

magNrPlaces					
Number of real locations (in chain magazine) or number of slots (in box magazine)					
-				UWord	r
Multi-line: yes	Magazine number		numMagsMax		

magPlaceSearchStrat					
magPlaceSearchStrat					
-				UWord	r
Multi-line: no					

magPlaceUserDataNumLimit	entfaellt	BTSS-Baustein T/TUP
Readability of all OEM magazine location data The row number for access to the OEM magazine location data in the OPI block TUP is calculated as follows: numMagLocParams_u * magNrPlaces. ($MN_MM_NUM_CC_MAGLOC_PARAM * $TC_MAP6[magNo] * $TC_MAP7[magNo]). However, the maximum possible		

magPlaceUserDataNumLimit	entfaellt				BTSS-Baustein T/TUP
row number is 32767, this means that not all OEM location data can be addressed. The following bit-coded status codes are provided to indicate this state: Bit0=1: The product of the current values from the number of magazine locations ($TC_MAP6[magNo] * $TC_MAP7[magNo]) and the number of OEM magazine location parameters ($MN_MM_NUM_CC_MAGLOC_PARAM) exceeds the maximum row number. These values may not yet be effective. Bit1=1: The product of the effective values from the number of magazine locations ($TC_MAP6[magNo] * $TC_MAP7[magNo]) and the number of OEM magazine location parameters ($MN_MM_NUM_CC_MAGLOC_PARAM) exceeds the maximum row number. Therefore not all OEM data from all magazine locations of this magazine can be read on the OPI.					
-	0			UWord	r
Multi-line: yes	Magazine number		1		

magState	$TC_MAP3				FBW
State of the magazine 1 = current magazine 2 = disabled 4 = magazine in loading position 8 = motion is active 16 = enabled for loading					
-	2			UWord	rw
Multi-line: yes	Magazine number		numMagsMax		

magToolSearchStrat	$TC_MPAP10, Bits 0-7				
Tool search strategy during tool change					
-				UWord	r
Multi-line: yes	Magazine number		320000		

magWearCompoundNo	$TC_MAP9				
Each magazine has its own active wear group (wear group number). The number of this group is stored in OPI variables magWearCompoundNo: Meaning: Number of active wear group. =0: No wear group active. >0: Number of wear group in which tool search commences. (this is the number of the active wear group.) <0: Number of wear group in which tool search commences. However, this wear group is disabled which means that the next tool search is started in the next possible wear group. This system parameter can thus also be used to disable a wear group. See also wear group number of magazine location / $TC_MPP7 and modeWearGroup / $TC_MAMP3. Previous name: actWearGrInMag -32000, ..., -1, 0, 1, 2, ... 32000					
-	0			Long Integer	rw
Multi-line: yes	Magazine number		numMagsMax		

3.7.11 Area T, Block TP : Magazine data: Location data

OEM-MMC: Linkitem /ToolMagazine/...

The data module TP is organized as a 2-dimensional variable array. The module contains the state and assignment of all magazine locations of a T area. Each element can be addressed by a column and line index:

The column index is the magazine number, i.e. the configuration data for all locations of a magazine are contained in one column. The assignment of a magazine to a magazine number is stated in the associated area T of the associated Magazine directory module (TMV). If a non-existent magazine number is specified for the column index, the request is negatively acknowledged.

The number of lines is derived from the number of parameters per magazine location and from the number of magazine locations:

maxLineindex = numMagPlaceParams * magNrPlaces

The number of parameters per magazine location "numMagPlaceParams" is given in area N of module Y.

The line indices are based on the following scheme:

1: Location type ($TC_MPP1) (read only)

- 1: Magazine location
- 2: Spindle
- 3: Gripper
- 4: Loader
- 5: Transfer location
- 6: Loading station
- 7: Loading point

2: Location type ($TC_MPP2) (read only)

- >0: Location type for virtual location
- =0: "Match all" (buffer location)
- 9999: Undefined (not a virtual location)

3: T number of tool in this location ($TC_MPP6)

4: Consideration of adjacent location on / off ($TC_MPP3)

0: off

1: on

5: Location status ($TC_MPP4)

1: Disabled

2: Free (<> occupied)

4: Reserved for tool in buffer location

8: Reserved for tool to be loaded

16: Occupied in left half-location

32: Occupied in right half-location

64: Occupied in top half-location

128: Occupied in bottom half-location

6: Physical magazine reference (read only)

Magazine number of magazine to which location belongs

7: Type index ($TC_MPP5) (read only) and new: Wear group number as from SW 5.1

Type index/wear group number is read only in SW earlier than 5.1 but read/write as from SW 5.1 if it is assigned "Wear group" meaning.

Type index: The locations of a location type in a magazine are numbered in ascending order, e.g. type=2, type index=5; ==> Spindle5)

(previous meaning when location type = 1 before P5: Equals location number when location type=1)

Wear group number as from SW 5.1 ($TC_MPP5)

When location type = 1: Number of wear group to which this magazine location is assigned.

Value range: -32000, ..., -1, 0, 1, 2, ... 32000

=0: Not assigned to a wear group

>0: Number of assigned wear group, this wear group is enabled

<0: Number of assigned wear group, this wear group is disabled

By negating this system parameter, it is possible to disable or enable the whole assigned wear group.

See also magWearCompoundNo / $TC_MAP9 (active wear group number) and modeWearGroup / $TC_MAMP3 (general settings for wear grouping).

8: Adapter number as from SW 5.1 ($TC_MPP7)

Reference to adapter data set number.

Associated system data:

The number of parameters of this module changes accordingly:

N / Y, global system data, numMagPlaceParams = 8 as from SW 5.1

The number of magazine locations "magNrPlaces" is magazine specific and can be found in the associated area T of module TM.

The locations of the buffer magazine and the loading magazine are numbered in ascending order independently of the location type index.

If necessary, several lines can be addressed, so that, for example, all location data of a magazine can be read in a single request. The location data are all of the same type.

placeData	diverse, siehe Variablenbeschreibung				
P1: Place type (read-only access) ($TC_MPP1) P2: Place type (read-only access) ($TC_MPP2) P3: T number of the tool at this place ($TC_MPP6) P4: Consider adjacent place On/Off ($TC_MPP3) P5: Place state (bit array) ($TC_MPP4) P6: Physical magazine reference (read-only access) P7: Place type index (numbering of a place type) ($TC_MPP5) P8: Number of the adapter at the magazine place ($TC_MPP7) P9: Mag-Place-ToolNo-Reserver-For ($TC_MPP66) P10: Number of the spindle assigned to the buffer magazine place ($TC_MPP_SP) Only of significance if - tool holders are used ($MC_TOOLHOLDER_MANAGEMENT > 0) - the magazine place "m" belongs to a buffer magazine "n" - the magazine place describes a tool holder ($TC_MPP1[n,m]=2) In this case the system variable contains the spindle number, the speed of which is to be monitored for maximum tool speed. If no tool holders are used ($MC_TOOLHOLDER_MANAGEMENT = 0), the variable contains the value of the spindle index from $TC_MPP5 If magazine place "n,m" does not use a buffer magazine place for a spindle or tool holder, this variable contains the value = 0. P11: Type of T no. (tool or MT) ($P_TMNOIS) colIndex: Tool magazine number Notice! This variable is referred to as "dummy" in the non-Windows HMI and the PLC.					
-				UWord	rw
Multi-line: yes	(LocationNo - 1) * numMagPlaceParams + ParameterNo		numMagPlaceParams * magNrPlaces		

3.7.12 Area T, Block TPM : Magazine data: Multiple assignment of location data

OEM-MMC: Linkitem /ToolMagazine/...

The data module TPM is organized as a 2-dimensional variable array.

ParameterNo = 1: Specifies the magazine number with which a relationship exists.

ParameterNo = 2: Distance (in locations) between the internal location and the magazine change position (cf. magazine number for 1st parameter) with which a relationship will be established.

It contains information about possible multiple assignments. The column index is the magazine number.

For location P with location number p in magazine MP (= column index) numPlaceMulti times the multiple assignments to other magazines which are possible are stored with the associated distances to the change positions in each of the magazines. The offset for row index zi for a location number p is calculated according to the following rule: zi = (p-1) * numPlaceMulti * numPlaceMultiParams + ParameterNo.

Determining the distance between the load position and the change position:

The value 9999 (magazine no. load position) must be specified for the variable multiPlace in the column. The LocationNo (p) for the line is the number of the load position. The line for the first assignment is calculated with ParameterNo = 1. When reading the variable, the system can thus read the magazine number linked to the intended change position. If this magazine number is correct, it is possible to read the number of locations between the load position and the change position with the variable multiPlace with the next higher line number. If the magazine number read was incorrect, the following magazine assignment must be read with the line number increased by numPlaceMulti.

This procedure has to be repeated a max. of numPlaceMultiParams times until the desired relationship has been found.

multiPlace	diverse, siehe Variablenbeschreibung				
P1: Distance between the change position of magazine n and the location m of the first internal magazine (loading magazine, 9999) ($TC_MDP1) P2: Distance between the change position of magazine n and the location m of the second internal magazine (buffer magazine, 9998) ($TC_MDP2) colIndex: Tool magazine number					
-				UWord	r
Multi-line: yes	(LocationNo - 1) * numPlaceMulti * numPlaceMultiParams +ParameterNo In this case, numPlaceMulti and numPlaceMultiParams are other OPI variables from module Y.		numPlaceMulti * numPlaceMultiParams * magNrPlaces		

3.7.13 Area T, Block TT : Magazine data: Location types

OEM-MMC: Linkitem /ToolMagazine/...

The module TT is organized as a 2-dimensional array where the variable with index (1/1) contains the maximum number of columns (corresponds to the location hierarchies) in this module. Each element can be addressed via a column and row index:

The column index is the number of the location hierarchy + 1. The row index is the number of the location type + 1. Row 1 contains the current T-number of rows for a specific location hierarchy as special information.

If all location types are to be read out for a location hierarchy, this must be defined in two steps:

1. The 1st line of each location hierarchy contains the number of assigned location types for this hierarchy

2. Lines 2 ... n can be read out in a single request.

placeType					
Magazine location hierarchy Attention: This variable is called "dummy" in the non-Windows HMI and the PLC. colIndex: Number of the location hierarchy + 1					
-				UWord	r
Multi-line: yes	Number of the location type + 1		Wert aus Zeile 1		

3.7.14 Area T, Block TV : Tool data: Directory

OEM-MMC: Linkitem /ToolCatalogue/...

Data module TV can be used for the following purposes:

1. For displaying all tools of a magazine. The most important tool information is contained in module TV. Available tools are sorted consecutively in ascending order of T-number. This means that variables that are defined as one-dimensional arrays in this module contain all the tool information without any gaps. The line index with which a specific array is addressed has no connection with the tool number but is only a serial number. Inserting/deleting tools changes the contents of a line dynamically.

2. Access to tool data in modules TD, TG, TO, TS, TU and TUE. Before an element in one of the above modules is accessed, module TV should be consulted to ascertain which tools are actually defined.

SW 5.1 and later: Variable modeSpindleToolRevolver (module N/Y, global system data) defines for circular magazines (T / TM, magazine data, general data, MagKind=3) whether the tool in OPI modules "T / TP, magazine data, location data", "T / TD, tool data, general data", "T/TV, tool data, directory" and "T / AEV, working offsets, directory" remains (new functionality) in its circular magazine location during operation or changes to the buffer magazine (earlier behaviour).

Associated system data:

modeSpindleToolRevolver (module N / Y, global system data) with SW 5.1 and later.

TnumWZV					
Last assigned T-number for tool management The last assigned T number is the T number of the new tool last created in the NCK through an NC language command or the PI service.					
-				UWord	r
Multi-line: no					

nrDuplo					
Duplo number					
-				UWord	r
Multi-line: yes	Serial number		numTools		

numCuttEdges					
Number of cutting edges of a tool					
-			9	UWord	r
Multi-line: yes	Serial number		numTools		

numToolGroups					
numToolGroups					
-				UWord	r
Multi-line: no					

numTools					
Number of tools in the area TO					
-		0	MD MM_NUM_TOO L	UWord	r
Multi-line: no					

toolIdent					
Tool identifier					
-				String [32]	r
Multi-line: yes	Serial number		numTools		

toolInMag					
Current magazine in which the tool is located 0 = tool not loaded					
-				UWord	r
Multi-line: yes	Serial number		numTools		

toolInPlace					
Current location in which the tool is located 0 = tool not loaded					
-				UWord	r
Multi-line: yes	Serial number		numTools		

toolNo					
T number					
-				UWord	r
Multi-line: yes	Serial number		numTools		

3.7.15 Area T, Block TF : Parametrizing, return parameters of _N_TMGETT, _N_TSEARC

OEM-MMC: Linkitem /ToolFind/...

This module is used for parameterizing as well as for the return parameters of PI services _N_TMGETT and _N_TSEARC. Access to this module must be T area specific and exclusive. It is up to the clients to guarantee this by using the semaphore mechanism (PI service _N_MMCSEM) with the function number for _N_TMSEARCH.

With _N_TMGETT, NO parameterizing elements (input parameters) are relevant; the only relevant one is the result parameter resultToolNr

parDataTAD					
Parameterizing: For parameters with data type DOUBLE of the module TAD a value can be stored as a comparison value for a 'complex search' (_N_TSEARC). The comparison value is combined with the corresponding parameter in the module TAS according to parMasksTAD. The size of the column matches the lines in module TAO. See module TAD					
-				Double	rw
Multi-line: yes	Column index in the module TAD, i.e. the number of the user-defined tool parameter. The maximum line index thus equals the number of columns in the TAD module.		numToolParams_tad		

parDataTAO					
Parameterizing: For parameters with data type DOUBLE of the module TAO a value can be stored as a comparison value for a 'complex search' (_N_TSEARC). The comparison value is combined with the corresponding parameter in the module TAS according to parMasksTAO. The size of the column matches the lines in module TAO. See module TAO					
-				Double	rw
Multi-line: yes	Column index in the module TAO, i.e. tool number. The maximum line index thus equals the number of columns in the TAO module.		numCuttEdgeParams_tao		

parDataTAS					
Parameterizing: For parameters with data type DOUBLE of the module TAS a value can be stored as a comparison value for a 'complex search' (_N_TSEARC). The comparison value is combined with the corresponding parameter in the module TAS according to parMasksTAS. The size of the column matches the lines in module TAS. See module TAS					
-				Double	rw
Multi-line: yes	Column index in the module TAS, i.e. tool number. The maximum line index thus equals the number of columns in the TAS module.		numCuttEdgeParams_tas		

parDataTD					
Parameterizing: For parameters with data type UWORD of the module TD a value can be stored as a comparison value for a 'complex search' (_N_TSEARC). The comparison value is combined with the corresponding parameter in the module TD according to parMasksTD. The size of the column matches the lines in module TD. See module TD					
-				UWord	rw
Multi-line: yes	Index of the parameter (i.e. column index) in the TD module > 1. The maximum line index thus equals the number of columns in the TD module.		17		

parDataTO					
Parameterizing: For each parameter of the module TO, a value can be stored as a comparison value for the 'complex search' (_N_TSEARC). The comparison value is combined with the corresponding parameter in the module TO according to parMasksTO. The size of the column matches the data set of an edge in module TO. See module TO					
-				Double	rw
Multi-line: yes	Line index in the TO module, i.e. a cutting edge offset value parameter: (EdgeNo - 1) * numCuttEdgeParams + ParameterNo The maximum line index is thus the maximum cutting edge offset value parameter in the module TO.		numCuttEdgeParams * maxnumCuttEdges_Tool		

parDataTS					
Parameterizing: For each parameter of the module TS a value can be stored as a comparison value for a 'complex search' (_N_TSEARC). The comparison value is combined with the corresponding parameter in the module TS according to parMasksTS. The size of the column matches the data set of an edge in module TS. See module TS					
-				Double	rw
Multi-line: yes	Line index in the TS module: (EdgeNo - 1) * numCuttEdgeParams_ts + ParameterNo The maximum line index is thus the maximum cutting edge parameter in the module TS.		numCuttEdgeParams_ts * maxnumCuttEdges_Tool		

parDataTU					
Parameterizing: For each parameter of the module TU a value can be stored as a comparison value for a 'complex search' (_N_TSEARC). The comparison value is combined with the corresponding parameter in the module TU according to parMasksTU. The size of the column matches the lines in module TU. See module TU					
-				Double	rw
Multi-line: yes	Index of the parameter (i.e. column index) in the TU module is thus the number of the user-defined tool parameter. The maximum line index thus equals the number of columns in the TU module (numToolParams_tu).		numToolParams_tu		

parDataTUE					
Parameterizing: For each parameter of the module TUE a value can be stored as a comparison value for a 'complex search' (_N_TSEARC). The comparison value is combined with the corresponding parameter in the module TUE according to parMasksTUE. The size of the column matches the data set of an edge in module TUE. See module TUE					
-				Double	rw
Multi-line: yes	Line index in the TUE module: (EdgeNo - 1) * numCuttEdgeParams_tu + ParameterNo The maximum line index is thus the maximum cutting edge parameter in the module TUE.		numCuttEdgeParams_tu * maxnumCuttEdges_Tool		

parDataTUS					
Parameterizing: For each parameter of the module TUS a value can be stored as a comparison value for a 'complex search' (_N_TUSEARC). The comparison value is combined with the corresponding parameter in the module TUS according to parMasksTUS. The size of the column matches the data set of an edge in module TUS. See module TUS					
-				Double	rw
Multi-line: yes	Line index in the TUS module: Number of the user-defined parameter + (number of the tool cutting edge -1) * numCuttEdgeParams_tus. The maximum line index is thus the maximum cutting edge parameter in the module TUS.		numCuttEdgeParams_tus * maxnumCuttEdges_Tool		

parDataToolIdentTD					
Parameterizing: For the parameter with data type string[32] (tool identifier) of the module TD a value can be stored as a comparison value for a 'complex search' (_N_TSEARC). The comparison value is combined with the corresponding parameter in the module TD according to parMasksTD. See module TD					
-				String [32]	rw
Multi-line: no					

parMasksTAD					
Parameterizing: There is a mask for each parameter of the module TAD that indicates whether it is to serve as a search criterion for a 'complex search' (_N_TSEARC) and how it is to be combined. The corresponding comparison values are stored in parDataTAD. If more than one parameter (i.e. search criteron) has been selected (#0), they are logically combined with AND. Value 0 : Corresponding operand is not evaluated / Variable is not a criterion for comparison Value 1 : == (equal) Value 2 : < (less than) Value 3 : > (greater than) Value 4 : <= (less or equal) Value 5 : >= (greater or equal) Value 6 : && (bitwise AND, value only allowed for operands of the types WORD and DOUBLEWORD) For string operands "==" is the only operator allowed					
-	0	0	6	UWord	rw
Multi-line: yes	Column index in the module TAD, i.e. the number of the user-defined tool parameter. The maximum line index thus equals the number of columns in the TAD module.		numToolParams_tad		

parMasksTAO					
Parameterizing: There is a mask for each parameter of the module TAO that indicates whether it is to serve as a search criterion for a 'complex search' (_N_TSEARC) and how it is to be combined. The corresponding comparison values are stored in parDataTAO. If more than one parameter (i.e. search criteron) has been selected (#0), they are logically combined with AND. Value 0 : Corresponding operand is not evaluated / Variable is not a criterion for comparison Value 1 : == (equal) Value 2 : < (less than) Value 3 : > (greater than) Value 4 : <= (less or equal) Value 5 : >= (greater or equal) Value 6 : && (bitwise AND, value only allowed for operands of the types WORD and DOUBLEWORD) For string operands "==" is the only operator allowed					
-	0	0	6	UWord	rw
Multi-line: yes	Column index in the module TAO, i.e. tool number. The maximum line index thus equals the number of columns in the TAO module.		numCuttEdgeParams_tao		

parMasksTAS					
Parameterizing: There is a mask for each parameter of the module TAS that indicates whether it is to serve as a search criterion for a 'complex search' (_N_TSEARC) and how it is to be combined. The corresponding comparison values are stored in parDataTAS. If more than one parameter (i.e. search criteron) has been selected (#0), they are logically combined with AND. Value 0 : Corresponding operand is not evaluated / Variable is not a criterion for comparison Value 1 : == (equal) Value 2 : < (less than) Value 3 : > (greater than) Value 4 : <= (less or equal) Value 5 : >= (greater or equal) Value 6 : && (bitwise AND, value only allowed for operands of the types WORD and DOUBLEWORD) For string operands "==" is the only operator allowed					
-	0	0	6	UWord	rw
Multi-line: yes	Column index in the module TAS, i.e. tool number. The maximum line index thus equals the number of columns in the TAS module.		numCuttEdgeParams_tas		

parMasksTD					
Parameterizing: There is a mask for each parameter of the module TD that indicates whether it is to serve as a search criterion for a 'complex search' (_N_TSEARC) and how it is to be combined. The corresponding comparison values are stored in parDataTD. If more than one parameter (i.e. search criteron) has been selected (#0), they are logically combined with AND. Value 0 : Corresponding operand is not evaluated / Variable is not a criterion for comparison Value 1 : == (equal) Value 2 : < (less than) Value 3 : > (greater than) Value 4 : <= (less or equal) Value 5 : >= (greater or equal) Value 6 : && (bitwise AND, value only allowed for operands of the types WORD and DOUBLEWORD) For string operands "==" is the only operator allowed					
-	0	0	6	UWord	rw
Multi-line: yes	Index of the parameter (i.e. column index) in the TD module > 1. The maximum line index thus equals the number of columns in the TD module.		17		

parMasksTO					
Parameterizing: There is a mask for each parameter of the module TO that indicates whether it is to serve as a search criterion for a 'complex search' (_N_TSEARC) and how it is to be combined. The corresponding comparison values are stored in parDataTO. If more than one parameter (i.e. search criteron) has been selected (#0), they are logically combined with AND. Value 0 : Corresponding operand is not evaluated / Variable is not a criterion for comparison Value 1 : == (equal) Value 2 : < (less than) Value 3 : > (greater than) Value 4 : <= (less or equal) Value 5 : >= (greater or equal) Value 6 : && (bitwise AND, value only allowed for operands of the types WORD and DOUBLEWORD) For string operands "==" is the only operator allowed					
-	0	0	6	UWord	rw
Multi-line: yes	Line index in the TO module, i.e. a cutting edge offset value parameter: (EdgeNo - 1) * numCuttEdgeParams + ParameterNo The maximum line index is thus the maximum cutting edge offset value parameter in the module TO.		numCuttEdgeParams * maxnumCuttEdges_Tool		

parMasksTS					
Parameterizing: There is a mask for each parameter of the module TS that indicates whether it is to serve as a search criterion for a 'complex search' (_N_TSEARC) and how it is to be combined. The corresponding comparison values are stored in parDataTS. If more than one parameter (i.e. search criteron) has been selected (#0), they are logically combined with AND. Value 0 : Corresponding operand is not evaluated / Variable is not a criterion for comparison Value 1 : == (equal) Value 2 : < (less than) Value 3 : > (greater than) Value 4 : <= (less or equal) Value 5 : >= (greater or equal) Value 6 : && (bitwise AND, value only allowed for operands of the types WORD and DOUBLEWORD) For string operands "==" is the only operator allowed					
-	0	0	6	UWord	rw
Multi-line: yes	Line index in the TS module: (EdgeNo - 1) * numCuttEdgeParams_ts + ParameterNo The maximum line index is thus the maximum cutting edge parameter in the module TS.		numCuttEdgeParams_ts * maxnumCuttEdges_Tool		

parMasksTU					
Parameterizing: There is a mask for each parameter of the module TU that indicates whether it is to serve as a search criterion for a 'complex search' (_N_TSEARC) and how it is to be combined. The corresponding comparison values are stored in parDataTU. If more than one parameter (i.e. search criteron) has been selected (#0), they are logically combined with AND. Value 0 : Corresponding operand is not evaluated / Variable is not a criterion for comparison Value 1 : == (equal) Value 2 : < (less than) Value 3 : > (greater than) Value 4 : <= (less or equal) Value 5 : >= (greater or equal) Value 6 : && (bitwise AND, value only allowed for operands of the types WORD and DOUBLEWORD) For string operands "==" is the only operator allowed					
-	0	0	6	UWord	rw
Multi-line: yes	Index of the parameter (i.e. column index) in the TU module is thus the number of the user-defined tool parameter. The maximum line index thus equals the number of columns in the TU module (numToolParams_tu).		numToolParams_tu		

parMasksTUE					
Parameterizing: There is a mask for each parameter of the module TUE that indicates whether it is to serve as a search criterion for a 'complex search' (_N_TSEARC) and how it is to be combined. The corresponding comparison values are stored in parDataTUE. If more than one parameter (i.e. search criteron) has been selected (#0), they are logically combined with AND. Value 0 : Corresponding operand is not evaluated / Variable is not a criterion for comparison Value 1 : == (equal) Value 2 : < (less than) Value 3 : > (greater than) Value 4 : <= (less or equal) Value 5 : >= (greater or equal) Value 6 : && (bitwise AND, value only allowed for operands of the types WORD and DOUBLEWORD) For string operands "==" is the only operator allowed					
-	0	0	6	UWord	rw
Multi-line: yes	Line index in the TUE module: (EdgeNo - 1) * numCuttEdgeParams_tu + ParameterNo The maximum line index is thus the maximum cutting edge parameter in the module TUE.		numCuttEdgeParams_tu * maxnumCuttEdges_Tool		

parMasksTUS					
Parameterizing: There is a mask for each parameter of the module TUS that indicates whether it is to serve as a search criterion for a 'complex search' (_N_TUSEARC) and how it is to be combined. The corresponding comparison values are stored in parDataTUS. If more than one parameter (i.e. search criteron) has been selected (#0), they are logically combined with AND. Value 0 : Corresponding operand is not evaluated / Variable is not a criterion for comparison Value 1 : == (equal) Value 2 : < (less than) Value 3 : > (greater than) Value 4 : <= (less or equal) Value 5 : >= (greater or equal) Value 6 : && (bitwise AND, value only allowed for operands of the types WORD and DOUBLEWORD) For string operands "==" is the only operator allowed					
-	0	0	6	UWord	rw
Multi-line: yes	Line index in the TUS module: Number of the user-defined parameter + (number of the tool cutting edge -1) * numCuttEdgeParams_tus. The maximum line index is thus the maximum cutting edge parameter in the module TUS.		numCuttEdgeParams_tus * maxnumCuttEdges_Tool		

resultCuttingEdgeNrUsed	$A_USEDD				
D numbers of the cutting edges used since the last workpiece count, that have previously been used on the defined tool carrier via resultNrOfCutEdgesUsed. Various D offsets for a tool indicate multiple entries of the tool, that means a T number can be present more than once. The two variables are linked to each other. resultNrOfCutEdgesUsed has to be read first, and then the individual T numbers with resultToolNrUsed. See also $A_USEDND, $A_USEDT and SETPIECE command. 0 maximum number of cutting edges in NCK					
-	0	0	max. Anzahl Schneiden in NCK	Long Integer	r
Multi-line: yes	((i. tool carrier-1) * line2 of column3 (resultNrOfCutEdgesUsed)) + consecutive number of the tool used		Zeile 1 * Zeile 2 von resultNrOfCutEdgesUsed		

resultNrOfCutEdgesUsed	$A_USEDND				
Line 1: Number of tool carriers Line 2: Maximum number of entries of resultToolNrUsed or resultCuttingEdgeNrUsed per tool carrier Line i+2: Number of the i. tool carrier Line i+3: Number of cutting edges which have been used since the last workpiece count on the i. tool carrier. This corresponds to $A_USEDND. The T and D numbers of the cutting edges can be read with resultToolNrUsed and resultCuttingEdgeNrUsed respectively. If TOOLMAN is not active and $MC_T_M_ADDRESS_EXT_IS_SPINO = FALSE, then line 1 = 1, $MC_T_M_ADDRESS_EXT_IS_SPINO = TRUE, then line 1 = 32. If tool monitoring is not active, line 2 = 0. See also $A_USEDT, $A_USEDD and SETPIECE command 0 maximum number of cutting edges in NCK					
-	0	0	max. Anzahl Schneiden in NCK	Long Integer	r
Multi-line: yes	Meaning of the index: See description		2*max.Anz. der Distanzbez. zw.Mag. und WZ-Haltern + 2 = 66		

resultNrOfTools					
Result: Number of tools found In the case of _N_TMGETT, it is possible to find no tools (value=0) or exactly 1 tool (value 1); in the case of _N_TSEARC, the number of found tools can be any number > 0, limited by the number of tools in the NC or no tools at all (value=0).					
-	0	0	numTools	UWord	r
Multi-line: yes	1		1		

resultToolNr					
Result: T-numbers of the tools found The array elements contain the internal T- numbers of the tools found. The storing order is the order in which the tools have been found by the PI-Service.					
-	0	0	31999	UWord	r
Multi-line: no			resultNrOfTools		

resultToolNrUsed	$A_USEDT				
T numbers of the cutting edges used since the last workpiece count, that have previously been used on the defined tool carrier via resultNrOfCutEdgesUsed. Various D offsets for a tool indicate multiple entries of the tool, that means a T number can be present more than once. The two variables are linked to each other. resultNrOfCutEdgesUsed has to be read first, and then the individual T numbers with resultToolNrUsed. See also $A_USEDND, $A_USEDD and SETPIECE command. 0 maximum number of cutting edges in NCK					
-	0	0	max. Anzahl Schneiden in NCK	Long Integer	r
Multi-line: yes	((i. tool carrier-1) * line2 of column3 (resultNrOfCutEdgesUsed)) + consecutive number of the tool used		Zeile 1 * Zeile 2 von resultNrOfCutEdgesUsed		

3.7.16 Area T, Block TUM : Tool data: user magazine data

OEM-MMC: Linkitem	/ToolMagazineDescription/...

This block contains the user magazine data

The TUM block is not intended for new developments.

3.7.17 Area T, Block TUMD : Tool data: user magazine data

OEM-MMC: Linkitem /ToolMagazineDescription/...

This block contains the user magazine data (double)

userDataDouble	$TC_MAPCx[y] x = ParameterNo y = MagazineNo				
Magazine user data for a tool magazine. These parameters can only be used if the machine data $MN_MM_NUM_CC_MAGAZINE_PARAM and $MN_MM_TOOL_MANAGEMENT_MASK have been set accordingly. Replaces the obsolete block T / TUM (same access, the only data type there is "TYPE_DWORD")					
-	0			Double	rw
Multi-line: yes	Number of the user-defined parameter		numMagParams_u		

3.7.18 Area T, Block TUP : Tool data: user magatine place data

OEM-MMC: Linkitem	/ToolMagazine/...

This block contains the user magazine location data

The TUP block is not intended for new developments.

3.7.19 Area T, Block TUPD : Tool data: user magatine place data

OEM-MMC: Linkitem /ToolMagazine/...

This block contains the user magazine location data (double)

userPlaceDataDouble	$TC_MPPCx[y,z] x=ParamNo y=MagazineNo z=MagPlaceNo				
Magazine location user data for a tool magazine. These parameters can only be used if the machine data $MN_MM_NUM_CC_MAGLOC_PARAM and $MN_MM_TOOL_MANAGEMENT_MASK have been set accordingly. Replaces the obsolete block T / TUP (same access, the only data type there is "TYPE_DWORD")					
-	0			Double	rw
Multi-line: yes	Number of the user-defined parameter + numMagLocParams_u * (number of the magazine location - 1)		numMagLocParams_u * magNrPlaces		

3.7.20 Area T, Block TUS : Tool data: user monitoring data

OEM-MMC: Linkitem /ToolSupervision/...

This block contains the user monitoring data of the tool data

userData	$TC_MOPCx[y,z] x=ParamNo,y=T-Number,z=Edge				
User data for monitoring a cutting edge. These parameters can only be used if the machine data $MN_MM_NUM_CC_MON_PARAM and $MN_MM_TOOL_MANAGEMENT_MASK have been set accordingly.					
-	0			Double	rw
Multi-line: yes	Number of the user-defined parameter + (number of the cutting edge -1) * numCuttEdgeParams_tus		numCuttEdgeParams_tus * maxnumCuttEdges_Tool		

3.7.21 Area T, Block AD : Adapter data

OEM-MMC: Linkitem /ToolAdapter/...

Adapter data are used to define the dimensions of an adapter (L1, L2, L3) per magazine location and the direction (transformation) of loaded tools.

The transformation is applied when cutting edge data are processed in OPI modules TOT, TOST and TOET if the tool is loaded in a magazine location to which adapter data are assigned.

Adapter data exist independently of magazine location data. Magazine location data contain a reference (see module TP, placeData) to the adapter data.

adaptData					
Adapter data colIndex: AdaptNo					
mm, inch, user defined	0.0			Double	rw
Multi-line: yes	ParameterNo		numParams_Adapt		

3.7.22 Area T, Block AEV : Working offsets: Directory

OEM-MMC: Linkitem /ToolActiveCatalogue/...

The active tool edges are sorted in consecutive ascending D number sequence in the AEV module. This module also contains the essential tool data for each D number entered. "Active" in this case refers to the replacement tools.

(If the "unique D numbers" option is not activated in the NC, the edges are sorted according to ascending ToolIdent and DuploNumber. The D number variable is then set to 0 on all lines in this module.)

The D number assignment is not necessarily unique for active tools. For this reason, the same D number may be entered in several lines (successively).

The line number is a serial number which is not related to the D number.

The number of active tool edges is stored in numActDEdges (module AEV), e.g. example 10,

i.e. module AEV contains entries for 10 tool edges. These are sorted in ascending D number sequence. The tool edge with the lowest D number has index (serial number) 1, the next-higher D number index 2, etc. and the edge with the highest D number index 10.

When tools are activated/deactivated and D numbers re-assigned, the entries for a D number change line dynamically.

Module T / AEV is organized as a 1-dimensional variable array and can be used for the following purposes:

- Display all tool edges, including D numbers, of active tools.

- Display associated tool data

The module contains the following information which can be addressed via a column index:

- Single column, in 1st line only. Number of D numbers (lines, tool edges) in the current list

- The other columns apply to all lines, each line contains tool edge data with the following information:

  - D number

  - Internal T number of associated tool

- Tool edge number relative to tool

  - Tool identifier

  - Duplo number

  - Magazine number and

  - Location number of tool

Individual values cannot be altered via this module.

Re-assignment of D numbers and changes in allocation to tools (deactivate, activate replacement tools) and other modifications to data cause changes to toolCounter in "C / S Channel-specific status data".

Variable modeSpindleToolRevolver (module N/Y, global system data) defines for circular magazines (T / TM, magazine data, general data, MagKind=3) whether the tool in OPI modules "T / TP, magazine data, location data", "T / TD, tool data, general data", "T/TV, tool data, directory" and "T / AEV, working offsets, directory" remains (new functionality) in its circular magazine location during operation or changes to the buffer magazine (earlier behaviour)).

DNo					
D number Meaningful and defined only in connection with "unique D numbers" function.					
-				UWord	r
Multi-line: yes	Serial number of active edges		numActDEdges		

cuttEdgeNo					
Number of edge for this tool Meaningful and defined only in connection with "unique D numbers" function.					
-		1	maxnumCuttEdges_Tool	UWord	r
Multi-line: yes	Serial number of active edges		numActDEdges		

duploNo					
Duplo number Meaningful and defined only in connection with "unique D numbers" function.					
-				UWord	r
Multi-line: yes	Serial number of active edges		numActDEdges		

numActDEdges					
Number of D numbers in this list Meaningful and defined only in connection with "unique D numbers" function. When tool management function is active: Specifies the number of edges belonging to tools with "active" status (contained in the TO unit) When tool management function is not active: Specifies the number of all edges contained in the TO unit.					
-				UWord	r
Multi-line: yes	1		1		

toolIdent					
Tool identifier Meaningful and defined only in connection with "unique D numbers" function.					
-				String [32]	r
Multi-line: yes	Serial number of active edges		numActDEdges		

toolInMag					
Magazine in which tool is located Meaningful and defined only in connection with "unique D numbers" function.					
-				UWord	r
Multi-line: yes	Serial number of active edges		numActDEdges		

toolInPlace					
Magazine location of tool Meaningful and defined only in connection with "unique D numbers" function.					
-				UWord	r
Multi-line: yes	Serial number of active edges		numActDEdges		

toolNo					
Internal T number Meaningful and defined only in connection with "unique D numbers" function.					
-				UWord	r
Multi-line: yes	Serial number of active edges		numActDEdges		

3.7.23 Area T, Block TC : Toolholder parameters

OEM-MMC: Linkitem /ToolToolCarrier/...

Module TC contains the data which define an orientatable toolholder (offset vectors, axis directions, rotation angle, type information).

It is also possible to read the current positions of the toolholder axes and the differences between the current and programmed axis values for the active toolholder.

tcCarr1	$TC_CARR1				
x component of offset vector l1					
-	0			Double	rw
Multi-line: yes	No. of toolholder		$MN_MM_NUM_TOOL_CARRIER / numToBaust		

tcCarr10	$TC_CARR10				
x component of rotary axis v2					
-	0			Double	rw
Multi-line: yes	No. of toolholder		$MN_MM_NUM_TOOL_CARRIER / numToBaust		

tcCarr11	$TC_CARR11				
y component of rotary axis v2					
-	0			Double	rw
Multi-line: yes	No. of toolholder		$MN_MM_NUM_TOOL_CARRIER / numToBaust		

tcCarr12	$TC_CARR12				
z component of rotary axis v2					
-	0			Double	rw
Multi-line: yes	No. of toolholder		$MN_MM_NUM_TOOL_CARRIER / numToBaust		

tcCarr13	$TC_CARR13				
Angle of rotation alpha1 (in degrees)					
Degree	0			Double	rw
Multi-line: yes	No. of toolholder		$MN_MM_NUM_TOOL_CARRIER / numToBaust		

tcCarr14	$TC_CARR14				
Angle of rotation alpha2 (in degrees)					
Degree	0			Double	rw
Multi-line: yes	No. of toolholder		$MN_MM_NUM_TOOL_CARRIER / numToBaust		

tcCarr15	$TC_CARR15				
x component of offset vector l3					
-	0			Double	rw
Multi-line: yes	No. of toolholder		$MN_MM_NUM_TOOL_CARRIER / numToBaust		

tcCarr16	$TC_CARR16				
y component of offset vector l3					
-	0			Double	rw
Multi-line: yes	No. of toolholder		$MN_MM_NUM_TOOL_CARRIER / numToBaust		

tcCarr17	$TC_CARR17				
z component of offset vector l3					
-	0			Double	rw
Multi-line: yes	No. of toolholder		$MN_MM_NUM_TOOL_CARRIER / numToBaust		

tcCarr18	$TC_CARR18				
x component of offset vector l4					
-	0			Double	rw
Multi-line: yes	No. of toolholder		$MN_MM_NUM_TOOL_CARRIER / numToBaust		

tcCarr19	$TC_CARR19				
y component of offset vector l4					
-	0			Double	rw
Multi-line: yes	No. of toolholder		$MN_MM_NUM_TOOL_CARRIER / numToBaust		

tcCarr2	$TC_CARR2				
y component of offset vector l1					
-	0			Double	rw
Multi-line: yes	No. of toolholder		$MN_MM_NUM_TOOL_CARRIER / numToBaust		

tcCarr20	$TC_CARR20				
z component of offset vector l4					
-	0			Double	rw
Multi-line: yes	No. of toolholder		$MN_MM_NUM_TOOL_CARRIER / numToBaust		

tcCarr21	$TC_CARR21				
Axis identifier of 1st rotary axis					
-	0			String [32]	rw
Multi-line: yes	No. of toolholder		$MN_MM_NUM_TOOL_CARRIER / numToBaust		

tcCarr22	$TC_CARR22				
Axis identifier of 2nd rotary axis					
-	0			String [32]	rw
Multi-line: yes	No. of toolholder		$MN_MM_NUM_TOOL_CARRIER / numToBaust		

tcCarr23	$TC_CARR23				
Kinematic type Kinematic type: P: Rotatable workpiece (part) M: Rotatable tool and rotatable workpiece (mixed) T or any character except P and M: Rotatable tool					
-	0			String [32]	rw
Multi-line: yes	No. of toolholder		$MN_MM_NUM_TOOL_CARRIER / numToBaust		

tcCarr24	$TC_CARR24				
Offset of 1st rotary axis in degrees					
-	0			Double	rw
Multi-line: yes	No. of toolholder		$MN_MM_NUM_TOOL_CARRIER / numToBaust		

tcCarr25	$TC_CARR25				
Offset of 2nd rotary axis in degrees					
-	0			Double	rw
Multi-line: yes	No. of toolholder		$MN_MM_NUM_TOOL_CARRIER / numToBaust		

tcCarr26	$TC_CARR26				
Offset of Hirth tooth system in degrees of 1st rotary axis					
-	0			Double	rw
Multi-line: yes	No. of toolholder		$MN_MM_NUM_TOOL_CARRIER / numToBaust		

tcCarr27	$TC_CARR27				
Offset of Hirth tooth system in degrees of 2nd rotary axis					
-	0			Double	rw
Multi-line: yes	No. of toolholder		$MN_MM_NUM_TOOL_CARRIER / numToBaust		

tcCarr28	$TC_CARR28				
Increment of Hirth tooth system in degrees of 1st rotary axis					
-	0			Double	rw
Multi-line: yes	No. of toolholder		$MN_MM_NUM_TOOL_CARRIER / numToBaust		

tcCarr29	$TC_CARR29				
Increment of Hirth tooth system in degrees of 2nd rotary axis					
-	0			Double	rw
Multi-line: yes	No. of toolholder		$MN_MM_NUM_TOOL_CARRIER / numToBaust		

tcCarr3	$TC_CARR3				
z component of offset vector l1					
-	0			Double	rw
Multi-line: yes	No. of toolholder		$MN_MM_NUM_TOOL_CARRIER / numToBaust		

tcCarr30	$TC_CARR30				
Minimum position of 1st rotary axis					
-	0			Double	rw
Multi-line: yes	No. of toolholder		$MN_MM_NUM_TOOL_CARRIER / numToBaust		

tcCarr31	$TC_CARR31				
Minimum position of 2nd rotary axis					
-	0			Double	rw
Multi-line: yes	No. of toolholder		$MN_MM_NUM_TOOL_CARRIER / numToBaust		

tcCarr32	$TC_CARR32				
Maximum position of 1st rotary axis					
-	0			Double	rw
Multi-line: yes	No. of toolholder		$MN_MM_NUM_TOOL_CARRIER / numToBaust		

tcCarr33	$TC_CARR33				
Maximum position of 2nd rotary axis					
-	0			Double	rw
Multi-line: yes	No. of toolholder		$MN_MM_NUM_TOOL_CARRIER / numToBaust		

tcCarr34	$TC_CARR34				
Toolholder name Contains a freely definable string provided as a freely definable identifier for the orientatable toolholder. It has no meaning as yet within the NCK and is not evaluated either. The identifier should not be used for other purposes since a later expansion will allow an orientatable toolholder to be activated via a name as well as via numbers					
-				String [32]	rw
Multi-line: yes	No. of toolholder		$MN_MM_NUM_TOOL_CARRIER		

tcCarr35	$TC_CARR35				
Axis name 1 Contains a freely definable string provided as a free identifier for the first rotary axis. It has no meaning whatsoever within the NCK, neither is it evaluated. It can therefore be used for any other purposes.					
-	0			String [32]	rw
Multi-line: yes	No. of toolholder		$MN_MM_NUM_TOOL_CARRIER		

tcCarr36	$TC_CARR36				
Axis name 2 Contains a freely definable string provided as a free identifier for the second rotary axis. It has no meaning whatsoever within the NCK, neither is it evaluated. It can therefore be used for any other purposes.					
-				String [32]	rw
Multi-line: yes	No. of toolholder		$MN_MM_NUM_TOOL_CARRIER		

tcCarr37	$TC_CARR37				
Identifier Contains an integer number for identifying the toolholder. It has no meaning whatsoever within the NCK, neither is it evaluated.					
-	0			Long Integer	rw
Multi-line: yes	No. of toolholder		$MN_MM_NUM_TOOL_CARRIER		

tcCarr38	$TC_CARR38				
Position component X Contains a position (X component of return position). It has no meaning whatsoever within the NCK, neither is it evaluated.					
-	0			Double	rw
Multi-line: yes	No. of toolholder		$MN_MM_NUM_TOOL_CARRIER		

tcCarr39	$TC_CARR39				
Position component Y Contains a position (Y component of return position). It has no meaning whatsoever within the NCK, neither is it evaluated.					
-	0			Double	rw
Multi-line: yes	No. of toolholder		$MN_MM_NUM_TOOL_CARRIER		

tcCarr4	$TC_CARR4				
x component of offset vector l2					
-	0			Double	rw
Multi-line: yes	No. of toolholder		$MN_MM_NUM_TOOL_CARRIER / numToBaust		

tcCarr40	$TC_CARR40				
Position component Z Contains a position (Z component of return position). It has no meaning whatsoever within the NCK, neither is it evaluated.					
-	0			Double	rw
Multi-line: yes	No. of toolholder		$MN_MM_NUM_TOOL_CARRIER		

tcCarr41	$TC_CARR41				
x-component of the fine offset of the offset vector l1					
mm, inch, user defined	0	0		Double	rw
Multi-line: yes	Number of the tool carrier		$MN_MM_NUM_TOOL_CARRIER		

tcCarr42	$TC_CARR42				
y-component of the fine offset of the offset vector l1					
mm, inch, user defined	0	0		Double	rw
Multi-line: yes	Number of the tool carrier		$MN_MM_NUM_TOOL_CARRIER		

tcCarr43	$TC_CARR43				
z-component of the fine offset of the offset vector l1					
mm, inch, user defined	0	0		Double	rw
Multi-line: yes	Number of the tool carrier		$MN_MM_NUM_TOOL_CARRIER		

tcCarr44	$TC_CARR44				
x-component of the fine offset of the offset vector l2					
mm, inch, user defined	0	0		Double	rw
Multi-line: yes	Number of the tool carrier		$MN_MM_NUM_TOOL_CARRIER		

tcCarr45	$TC_CARR45				
y-component of the fine offset of the offset vector l2					
mm, inch, user defined	0	0		Double	rw
Multi-line: yes	Number of the tool carrier		$MN_MM_NUM_TOOL_CARRIER		

tcCarr46	$TC_CARR46				
z-component of the fine offset of the offset vector l2					
mm, inch, user defined	0	0		Double	rw
Multi-line: yes	Number of the tool carrier		$MN_MM_NUM_TOOL_CARRIER		

tcCarr5	$TC_CARR5				
y component of offset vector l2					
-	0			Double	rw
Multi-line: yes	No. of toolholder		$MN_MM_NUM_TOOL_CARRIER / numToBaust		

tcCarr55	$TC_CARR55				
x-component of the fine offset of the offset vector l3					
mm, inch, user defined	0	0		Double	rw
Multi-line: yes	Number of the tool carrier		$MN_MM_NUM_TOOL_CARRIER		

tcCarr56	$TC_CARR56				
y-component of the fine offset of the offset vector l3					
mm, inch, user defined	0	0		Double	rw
Multi-line: yes	Number of the tool carrier		$MN_MM_NUM_TOOL_CARRIER		

tcCarr57	$TC_CARR57				
z-component of the fine offset of the offset vector l3					
mm, inch, user defined	0	0		Double	rw
Multi-line: yes	Number of the tool carrier		$MN_MM_NUM_TOOL_CARRIER		

tcCarr58	$TC_CARR58				
x-component of the fine offset of the offset vector l4					
mm, inch, user defined	0	0		Double	rw
Multi-line: yes	Number of the tool carrier		$MN_MM_NUM_TOOL_CARRIER		

tcCarr59	$TC_CARR59				
y-component of the fine offset of the offset vector l4					
mm, inch, user defined	0	0		Double	rw
Multi-line: yes	Number of the tool carrier		$MN_MM_NUM_TOOL_CARRIER		

tcCarr6	$TC_CARR6				
z component of offset vector l2					
-	0			Double	rw
Multi-line: yes	No. of toolholder		$MN_MM_NUM_TOOL_CARRIER / numToBaust		

tcCarr60	$TC_CARR60				
z-component of the fine offset of the offset vector l4					
mm, inch, user defined	0	0		Double	rw
Multi-line: yes	Number of the tool carrier		$MN_MM_NUM_TOOL_CARRIER		

tcCarr64	$TC_CARR64				
Fine offset of the offset of the rotary axis v1					
Degree, user defined	0	0		Double	rw
Multi-line: yes	Number of the tool carrier		$MN_MM_NUM_TOOL_CARRIER		

tcCarr65	$TC_CARR65				
Fine offset of the offset of the rotary axis v2					
Degree, user defined	0	0		Double	rw
Multi-line: yes	Number of the tool carrier		$MN_MM_NUM_TOOL_CARRIER		

tcCarr7	$TC_CARR7				
x component of rotary axis v1					
-	0			Double	rw
Multi-line: yes	No. of toolholder		$MN_MM_NUM_TOOL_CARRIER / numToBaust		

tcCarr8	$TC_CARR8				
y component of rotary axis v1					
-	0			Double	rw
Multi-line: yes	No. of toolholder		$MN_MM_NUM_TOOL_CARRIER / numToBaust		

tcCarr9	$TC_CARR9				
z component of rotary axis v1					
-	0			Double	rw
Multi-line: yes	No. of toolholder		$MN_MM_NUM_TOOL_CARRIER / numToBaust		

tcCarr_KIN_PART_END	$TC_CARR_KIN_PART_END				
End element of the PART chain for parameterization from kinematic chains.					
-	0			String [32]	rw
Multi-line: yes	No. of toolholder		$MN_MM_NUM_TOOL_CARRIER / numToBaust		

tcCarr_KIN_PART_START	$TC_CARR_KIN_PART_START				
Start element of the PART chain for parameterization from kinematic chains.					
-	0			String [32]	rw
Multi-line: yes	No. of toolholder		$MN_MM_NUM_TOOL_CARRIER / numToBaust		

tcCarr_KIN_TOOL_END	$TC_CARR_KIN_TOOL_END				
Start element of the TOOL chain for parameterization from kinematic chains.					
-	0			String [32]	rw
Multi-line: yes	No. of toolholder		$MN_MM_NUM_TOOL_CARRIER / numToBaust		

tcCarr_KIN_TOOL_START	$TC_CARR_KIN_TOOL_START				
Start element of the TOOL chain for parameterization from kinematic chains.					
-	0			String [32]	rw
Multi-line: yes	No. of toolholder		$MN_MM_NUM_TOOL_CARRIER / numToBaust		

3.7.24 Area T, Block TOE : Edge-related coarse total offsets, setup offsets

OEM-MMC: Linkitem /ToolCompensation/...

One set of edge-related coarse total offsets, setup offsets, exists for each tool edge and operating location.

This module corresponds totally to module T / TOS, edge-related location-dependent fine total offsets.

edgeECData	$TC_ECPx[t,d]				
Location-dependent offsets, setup value					
mm, inch, user defined	0.0			Double	rw
Multi-line: yes	((EdgeNo-1) * (maxnumEdgeSC * numParams_SC)) + ((EdgeSC - 1)* numParams_SC) + ParameterNo		numParams_SC * maxnumEdgeSC * maxnumCuttEdges_Tool		

3.7.25 Area T, Block TOET : Edge-related coarse total offsets, transformed setup offsets

OEM-MMC: Linkitem /ToolCompTransfor/...

One set of edge-related transformed total offsets exists for each tool edge and operating location.

This module corresponds totally to module T / TOE.

<table>
<tr><th>edgeECData</th><th colspan="4"></th><th></th></tr>
<tr><td colspan="6">Transformed location-dependent offsets, setup value
colIndex: TNo</td></tr>
<tr><td>mm, inch, user defined</td><td>0.0</td><td></td><td></td><td>Double</td><td>rw</td></tr>
<tr><td>Multi-line: yes</td><td colspan="2">((EdgeNo-1) * (maxnumEdgeSC * numParams_SC)) + ((EdgeSC - 1)* numParams_SC) + ParameterNo</td><td colspan="3">numParams_SC * maxnumEdgeSC * maxnumCuttEdges_Tool</td></tr>
</table>

3.7.26 Area T, Block TOS : Edge-related location-dependent fine total offsets

OEM-MMC: Linkitem /ToolCompensation/...

One set of edge-related total offsets exists for each tool edge and operating location.

The maximum number of operating locations is identical for all tool edges and defined by the new variable maxnumEdgeSC ($MN_MAX_SUMCORR_PERCUTTING_EDGE) in "N / Y global system data".

numParams_SC (currently 9) offsets are provided (depending on location-independent wear values) for each total offset set: Length 1, length 2, length 3, radius and 5 others.

Each replacement tool has its own separate (different) data.

The NCK resets the data when the associated tool is activated if machine data ($MN_MM_KIND_OF_SUMCORR, bit 1 = 1) is used for activation.

The total offsets of a tool edge are accessible via the internal T number of the associated tool, edge number, total offset number ("operating location").

PI Services may exist for selective creation and deletion of tool edge total offsets.

The existence of total offsets can be controlled selectively via the new machine data $MN_MM_NUM_SUMCORR (OPI: maxNumSumCorr in N / Y).

The following applies:

When the MMC2 tool management function is in use, $MN_MM_NUM_SUMCORR = -1 must be set to ensure that the total offsets exist for all offset locations (number = maxnumEdgeSC) from creation of the tool edge until its deletion.

(The new PI Services for creation / deletion will not currently be used by the MMC2 tool management for turning applications). For the present, the new NC machine data $MN_MM_NUM_SUMCORR = -1 must be set to automatic creation / deletion.

The method of addressing in this module is analogous to accessing "Edge data / offsets" by column addressing with T number (using an array access operation to gain quick access to the total offsets of all tool edge operating locations or all edges of a tool).

The module contains the location-dependent total offsets for all tools. Each element is addressed via a column and line index:

The column index is the tool number (T number), i.e. all location-dependent total offsets of this tool (for all edges / locations) can be found in one column.

If a non-existent T number is specified as the column index, the request is acknowledged negatively.

The number of lines is determined by the number of total offset values, the number of operating locations and the maximum possible edge number of a tool:

maxZeilenindex = numParams_SC * maxnumEdgeSC * maxnumCuttEdges_Tool

These variables are stored in "N / Y global system data" and have the following meanings:

numParams_SC: No. of wear offsets per location (according to L1, L2, L3, radius and 5 others), currently 9

maxnumEdgeSC: Maximum number of locations (SC) per edge

maxnumCuttEdges_Tool: Max. permissible number of edges per tool

Several lines can be addressed simultaneously if necessary, allowing, for example, all location-dependent total offsets of all edges of one tool to be read in one request. The location-dependent total offsets of the tools are all of the same data type and have the same physical unit.

Module T / TOS has a 2-dimensional organization.

The following lines are provided for each T number (column index):

Edge 1, Location 1, L1

Edge 1, Location 1, L2

Edge 1, Location 1, L3

Edge 1, Location 1, Radius

Edge 1, Location 1, Par5

....

Edge 1, Location 1, Par numParams_SC

Edge 1, Location 2, L1

Edge 1, Location 2, L2

Edge 1, ...

Edge 1, Location maxnumEdgeSC, Par numParams_SC

Edge 2, Location 1, L1

....

Edge 2, Location maxnumEdgeSC, Par numParams_SC

....

Edge maxnumCuttEdges_Tool, Location maxnumEdgeSC, Par numParams_SC

Interrelationship between edge parameters, total offsets and variables:

Edge parameter DL1 DL2 ... DL4 ...

$TC_DP3 $TC_SCP13 $TC_SCP23 ... $TC_SCP43 ...

$TC_DP4 $TC_SCP14 $TC_SCP24 ... $TC_SCP44 ...

$TC_DP5 $TC_SCP15 $TC_SCP25 ... $TC_SCP45 ...

....

$TC_DP9 $TC_SCP19 $TC_SCP29 ... $TC_SCP49 ...

$TC_DP10 $TC_SCP20 $TC_SCP30 ... $TC_SCP50 ...

$TC_DP11 $TC_SCP21 $TC_SCP31 ... $TC_SCP51 ...

with DLx, TC_DPy, TC_SCPz

x from 1 to 6 (maxnumEdgeSC = $MN_MAX_SUMCORR_PERCUTTING_EDGE) and maximum = 6

y from 3 to 11

z = (10 * x) + y

edgeSCData	$TC_SCPx[t,d]				
Location-dependent offsets, wear colIndex: TNo					
mm, inch, user defined	0.0			Double	rw
Multi-line: yes	((EdgeNo-1) * (maxnumEdgeSC * numParams_SC)) + ((EdgeSC - 1)* numParams_SC) + ParameterNo		numParams_SC * maxnumEdgeSC * maxnumCuttEdges_Tool		

3.7.27 Area T, Block TOST : Edge-related location-dependent fine total offsets, transformed

OEM-MMC: Linkitem /ToolCompTransfor/...

One set of edge-related transformed total offsets exists for each tool edge and operating location.

This module corresponds totally to module T / TOS.

edgeSCData					
Transformed location-dependent offsets, wear colIndex: TNo					
mm, inch, user defined	0.0			Double	rw
Multi-line: yes	((EdgeNo-1) * (maxnumEdgeSC * numParams_SC)) + ((EdgeSC - 1)* numParams_SC) + ParameterNo		numParams_SC * maxnumEdgeSC * maxnumCuttEdges_Tool		

3.7.28 Area T, Block TOT : Edge data: Transformed offset data

OEM-MMC: Linkitem /ToolCompTransfor/...

The HMI must be capable of displaying and modifying the offset data of the tool edges as both transformed and untransformed data. The transformation refers to the adapter data (if programmed) of magazine locations. The HMI can display and modify both transformed and untransformed data (of the same tool if necessary) "simultaneously" (in different applications or different HMIs).

To provide access to transformed data, a new module, T / TOT (edge data: transformed offset data), is provided which is identical to the existing module T / TO (edge data: Offset data), except that it supplies transformed data instead of untransformed data.

The information edge DNo (D numbers assigned to edges) is included under the offset (numCuttEdgeParams * maxnumCuttEdges_Tool) in both the T / TOT and T / TO modules.

Both modules have a 2-dimensional organization.

The T number is the column index.

Line numbers are calculated by the following method:

(EdgeNo -1) * numCuttEdgeParams +parameter No.

numCuttEdgeParams = parameter per edge (currently 25) (from Y in N area)

EdgeNo = edge number for tool

Example: with numCuttEdgeParams = 25 and maxnumCuttEdges_Tool = 9

Column: T number

Lines:

1 edge 1, parameter 1

2 edge 1, parameter 2

...

25 edge 1, parameter numCuttEdgeParams

26 edge 2, parameter 1

27 edge 2, parameter 2

...

50 edge 2, parameter numCuttEdgeParams

...

225 edge maxnumCuttEdges_Tool, parameter numCuttEdgeParams

226 edge 1, D No assigned to edge 1

Untransformed data: /Tool/Compensation/edgeData[uToa,cTNr,line_from,line_to]

Transformed data: /Tool/CompTransfor/edgeData[uToa,cTNr,line_from,line_to]

Values which can be displayed as transformed data are the 9 geo-data (corresponding to L1, L2, L3, radius, and generally 5 other values), wear and total offsets.

If tools which are not located in a magazine location with adapter data are accessed via the module for transformed data, then the data are treated as if they were untransformed.

cuttEdgeParam					
Replaced by edgeData					
mm, inch, user defined	0.0			Double	rw
Multi-line: yes	See description edgeData		(numCuttEdgeParams + 1) * maxnumCuttEdges_Tool		

edgeData					
Adapter-transformed edge offset data and D number list NOTICE! This variable is called "cuttEdgeParam" in NonWindows HMI and PLC. Parameter number corresponds with numbering in T/TO module. The following data are transformed: Param2 (edge position) Param11 (cutting direction if tool type is grinding or turning tool) The following geometry data are exchanged amongst each other: Param3 - Param5 (length) Param12 - Param14 (wear) The other parameters are identical with the values in the T/TO module.					
mm, inch, user defined	0.0			Double	rw
Multi-line: yes	See description for T/TO module		(numCuttEdgeParams + 1) * maxnumCuttEdges_Tool		

3.7.29 Area T, Block TAD : Application-specific data

OEM-MMC: Linkitem /ToolData/...

Data module TAD is organized as a 2-dimensional variable array. This module contains application-specific data for all tools. Every element can be addressed via a column and row index:

The column index is the number of the user-defined tool parameter. The number of tool parameters (columns) can be found in variable numToolParams_tad in area N / module Y.

The row index is the tool number. Attempts to access non-existent tools are negatively acknowledged.

Application-specific tool data are all of the same data type.

Application-specific tool data are reserved for SIEMENS applications.

siemData	$TC_TPCSx[y]				
Siemens application tool parameter Important: 2-dimensional variable. Column index corresponds to the parameter number. Reserved for SIEMENS applications.					
-	0			Double	rw
Multi-line: yes	Tool number T		32000		

3.7.30 Area T, Block TAM : Application-specific magazine data

OEM-MMC: Linkitem	/ToolMagazineDescription/...

The TAM block contains application-specific information about the available tool magazines.

The application-specific magazine data are reserved for SIEMENS applications.

The TAM block is not intended for new developments.

3.7.31 Area T, Block TAMD : Application-specific magazine data (double)

OEM-MMC: Linkitem /ToolMagazineDescription/...

The TAMD block contains application-specific information about the available tool magazines.

The application-specific magazine data are reserved for SIEMENS applications.

siemDataDouble	$TC_MAPCSx[y]				
Siemens application magazine data. These parameters can be used only if machine data $MN_MM_NUM_CCS_MAGAZINE_PARAM and $MN_MM_TOOL_MANAGEMENT_MASK are set accordingly. Reserved for SIEMENS applications. Replaces the obsolete block T / TAM (same access, the only data type there is "TYPE_DWORD")					
-	0			Double	rw
Multi-line: yes	Number of the application-specific parameter		numMagParams_tam		

3.7.32 Area T, Block TAO : Application-specific cutting edge data

OEM-MMC: Linkitem /ToolCompensation/...

Data module TAO is organized as a 2-dimensional variable array. This module contains application-specific cutting edge data for all tools. Every element can be addressed via a column and row index. The column index is the tool number (T number), i.e. one column contains the application-specific data for all the cutting edges of a tool.

The assignments between tools and T numbers are listed in the Tool Directory (TV) module in the relevant T area.

A request is negatively acknowledged if a non-existent tool number is entered as the column index.

The number of rows is determined by the number of parameters per cutting edge and the number of cutting edges of a tool:

Max. number of rows = numCuttEdgeParams_tao * /T/TV/numCuttEdges (T number)

The number of parameters per cutting edge numCuttEdgeParams_tao can be found in area N / module Y. The number of tool-specific cutting edges is specified in area T / module TV.

Several rows can be addressed where necessary which means, for example, that all application-specific edge data of a tool can be read in one request.

Application-specific edge data are all of the same data type.

Application-specific cutting edge data are reserved for SIEMENS applications.

siemEdgeData	$TC_DPCSx[y,z]				
Siemens application tool cutting edge parameter Important: 2-dimensional variable. Column index corresponds to the T number. Reserved for SIEMENS applications.					
-	0			Double	rw
Multi-line: yes	(EdgeNo-1) * numCuttEdgeParams_tao + ParameterNo		numCuttEdgeParams_tao * numCuttEdges		

3.7.33 Area T, Block TAP : Application-specific magazine location data

OEM-MMC: Linkitem	/ToolMagazine/...

The TAP data block is organized as a 2-dimensional variable array. This block contains application-specific data of a T area. Each element can be addressed via a column and row index:

The column index is the magazine number, i.e. one column contains the application-specific magazine location data for all the locations of one magazine. The assignments between magazines and magazine numbers are listed in the appropriate Magazine Directory (TMV) block in the relevant T area. A request is negatively acknowledged if a non-existent magazine number is entered as the column index.

The number of rows is determined by the number of parameters per magazine location and the number of magazine locations:

Max. number of rows = numMagLocParams_tap * magNrPlaces

Application-specific magazine location data are reserved for SIEMENS applications.

The TAP block is not intended for new applications.

3.7.34 Area T, Block TAPD : Application-specific magazine location data

OEM-MMC: Linkitem /ToolMagazine/...

The TAPD data block is organized as a 2-dimensional variable array. This block contains application-specific data of a T area. Each element can be addressed via a column and row index:

The column index is the magazine number, i.e. one column contains the application-specific magazine location data for all the locations of one magazine. The assignments between magazines and magazine numbers are listed in the appropriate Magazine Directory (TMV) block in the relevant T area. A request is negatively acknowledged if a non-existent magazine number is entered as the column index.

The number of rows is determined by the number of parameters per magazine location and the number of magazine locations:

Max. number of rows = numMagLocParams_tap * magNrPlaces

Application-specific magazine location data are reserved for SIEMENS applications.

siemPlaceDataDouble	$TC_MPPCSx[y,z]				
Siemens application magazine location data. These parameters can be used only if machine data $MN_MM_NUM_CCS_MAGLOC_PARAM and $MN_MM_TOOL_MANAGEMENT_MASK are set accordingly. Reserved for SIEMENS applications. Replaces the obsolete block T / TAP (same access, the only data type there is "TYPE_DWORD")					
-	0			Double	rw
Multi-line: yes	ParameterNumber + numMagLocParams_tap * MagazineLocationNumber-1		numMagLocParams_tap * magNrPlaces		

3.7.35 Area T, Block TAS : Application-specific monitoring data

OEM-MMC: Linkitem /ToolSupervision/...

Data module TAS is organized as a 2-dimensional variable array. This module contains application-specific monitoring data for all tools. Every element can be addressed via a column and row index:

The column index is the tool number (T number), i.e. one column contains the application-specific monitoring data for all the cutting edges of a tool. The assignments between tools and T numbers are listed in the Tool Directory (TV) module in the relevant T area. A request is negatively acknowledged if a non-existent tool number is entered as the column index.

The number of rows is determined by the number of parameters per cutting edge and the number of cutting edges of a tool:

Max. number of rows = numCuttEdgeParams_tas * /T/TV/numCuttEdges (T number)

The number of parameters per cutting edge numCuttEdgeParams_tas can be found in area N / module Y. The number of tool-specific cutting edges (/T/TV/numCuttEdges) is specified in area T / module TV.

Several rows can be addressed where necessary which means, for example, that all application-specific monitoring data of a tool can be read in one request.

Application-specific monitoring data are all of the same data type.

Application-specific monitoring data are reserved for SIEMENS applications.

siemData	$TC_MOPCSx[y,z]				
Siemens application monitoring data of a tool cutting edge. These parameters can be used only if machine data $MN_MM_NUM_CCS_MON_PARAM and $MN_MM_TOOL_MANAGEMENT_MASK are set accordingly. Reserved for SIEMENS applications.					
-	0			Double	rw
Multi-line: yes	ParameterNumber + (EdgeNo -1) * numCuttEdgeParams_tas		numCuttEdgeParams_tas * numCuttEdges		

3.8 Machine and setting data

3.8.1 Area N, Block M : Global machine data

OEM-MMC: Linkitem /NckDrive/...

Global machine data

MDCA_DRIVE_LOGIC_NR	MD 13010: DRIVE_LOGIC_NR[x] x = PlugplaceNo				
Logical drive number					
-		0	30	Character	rw
Multi-line: yes	Slot number in drive bus		14		

MDCA_DRIVE_MODULE_TYPE	MD 13030: DRIVE_MODULE_TYPE[x] x = PlugplaceNo				
Module identifier of relevant drive bus slot 1 = single-axis module 2 = two-axis module 9 = terminal block for dig. I/Os 10 = bit bus interface					
-				Character	rw
Multi-line: yes	Slot number in drive bus		14		

MDCA_DRIVE_TYPE	MD 13040: DRIVE_TYPE[x] x = PlugplaceNo				
Drive type identifier for each drive bus slot 1 = FDD 2 = MSD					
-				Character	rw
Multi-line: yes	Slot number in drive bus		14		

MDD_INT_INCR_PER_DEG	MD 10210: INT_INCR_PER_DEG				
Calculation resolution for angular position					
-		0,000001	1000	Double	rw
Multi-line: no			1		

MDD_INT_INCR_PER_MM	MD 10200: INT_INCR_PER_MM				
Calculation resolution for linear positions					
-		0,000001	1000	Double	rw
Multi-line: no			1		

MDD_SYSCLOCK_CYCLE_TIME	MD 10050: SYSCLOCK_CYCLE_TIME				
Basic system clock cycle. For possible assignment of values, see description of machine data SYSCLOCK_CYCLE_TIME.					
s		0,000125 s	0,032 s	Double	rw
Multi-line: no			1		

MDLA_DRIVE_INVERTER_CODE	MD 13020: DRIVE_INVERTER_CODE[x] x = PlugplaceNo				
Power section code of drive module					
-				Long Integer	rw
Multi-line: yes	Slot number of drive module		14		

MDL_POSCTRL_SYSCLOCK_TIME_RATIO	MD 10060: POSCTRL_SYSCLOCK_TIME_RATIO				
Position control cycle factor					
-		1	100	Long Integer	rw
Multi-line: no			1		

MDSA_AXCONF_MACHAX_NAME_TAB	MD 10000: AXCONF_MACHAX_NAME_TAB[x] x = Axis				
Machine axis name					
-				String [16]	rw
Multi-line: yes	Axis index from 0		7		

3.8.2 Area A, Block M : Axis-specific machine data

OEM-MMC: Linkitem /AxisDrive/...

Axis-specific machine data

MDCA_CTRLOUT_MODULE_NR	MD 30110: CTRLOUT_MODULE_NR				
Setpoint assignment: Drive number / module number					
-		1	15	Character	rw
Multi-line: no			1		

MDCA_CTRLOUT_TYPE	MD 30130: CTRLOUT_TYPE				
Type of setpoint output					
-		0	1	Character	rw
Multi-line: no			1		

MDCA_ENC_MODULE_NR	MD 30220: ENC_MODULE_NR[x] x = PlugplaceNo				
Actual value assignment: Drive number / measuring circuit number					
-		1	15	Character	rw
Multi-line: yes	Encoder number		2		

MDCA_ENC_TYPE	MD 30240: ENC_TYPE[x] x = PlugplaceNo				
Type of actual value sensing (actual position value) Encoder type: 0: Simulation 1: Raw signal generator (high resolution) 2: Square wave generator - only with available onboard hardware 3: Encoder for semi-servo - only with available onboard hardware 4: Absolute encoder, gen. (e.g. with EnDat interface) 5: reserved					
-		0	5	Character	rw
Multi-line: yes	Encoder number		2		

3.8.3 Area N, Block SE : Global setting data

OEM-MMC: Linkitem /NckSettings/...

This module contains all global setting data. The physical units depend on the variable "userScale" in module Y of area N.

MDB_JOG_CONT_MODE_LEVELTRIGGRD	SD 41050: $SN_MDB_JOG_CONT_MODE_LEVELTRIGGRD				
Jog mode					
-				Character	rw
Multi-line: no					

MDB_JOG_REV_IS_ACTIVE	SD 41100: $SN_MDB_JOG_REV_IS_ACTIVE				
JOG at revolutional feedrate 0 = G94 1 = G95					
-				Character	rw
Multi-line: no					

MDD_JOG_REV_SET_VELO	SD 41120: $SN_MDD_JOG_REV_SET_VELO				
JOG velocity for G95					
Degree, user defined				Double	rw
Multi-line: no					

MDD_JOG_SET_VELO	SD 41110: $SN_MDD_JOG_SET_VELO				
JOG velocity for G94					
mm, inch, user defined				Double	rw
Multi-line: no					

MDD_JOG_SPIND_SET_VELO	SD 41200: $SN_MDD_JOG_SPIND_SET_VELO				
JOG velocity for master spindle					
rev/min, user defined				Double	rw
Multi-line: no					

MDD_JOG_VAR_INCR_SIZE	SD 41010: $SN_MDD_JOG_VAR_INCR_SIZE				
Variable incremental value for JOG mode					
-				Double	rw
Multi-line: no					

3.8.4 Area C, Block SE : Channel-specific setting data

OEM-MMC: Linkitem /ChannelSettings/...

Channel-specific setting data

MDD_DRY_RUN_FEED	SD 42100: $SC_MDD_DRY_RUN_FEED				
Dry run feedrate					
mm/min, inch/min, user defined				Double	rw
Multi-line: no					

MDD_THREAD_START_ANGLE	SD 42000: $SC_MDD_THREAD_START_ANGLE				
Starting angle for thread					
Degree				Double	rw
Multi-line: no					

3.8.5 Area A, Block SE : Axis-specific setting data

OEM-MMC: Linkitem /AxisSettings/...

This block contains the axis-specific setting data

AA_OFF_LIMIT	SD 43350: $SA_AA_OFF_LIMIT				
Upper limit of compensation value which can be preset by means of synchronized actions via the system variable $AA_OFF. This limit value acts on the absolutely effective compensation value via $AA_OFF. It is possible to interrogate the compensation value for limit-range violation via the system variable $AA_OFF_LIMIT.					
-				Double	r
Multi-line: no					

MDB_WORKAREA_MINUS_ENABLE	SD 43410: $SA_MDB_WORKAREA_MINUS_ENABLE				
Working area limitation active in the negative direction 0 = inactive 1 = active					
-				Character	rw
Multi-line: yes	Number of machine axis		1		

MDB_WORKAREA_PLUS_ENABLE	SD 43400: $SA_MDB_WORKAREA_PLUS_ENABLE				
Working area limitation active in the positive direction 0 = inactive 1 = active					
-				Character	rw
Multi-line: yes	Number of machine axis		1		

MDD_SPIND_MAX_VELO_G26	SD 43220: $SA_MDD_SPIND_MAX_VELO_G26				
Maximum spindle speed at G26 (master spindle)					
rev/min, user defined				Double	rw
Multi-line: no			1		

MDD_SPIND_MAX_VELO_LIMS	SD 43230: $SA_MDD_SPIND_MAX_VELO_LIMS				
Spindle speed limitation (master spindle)					
rev/min, user defined				Double	rw
Multi-line: no			1		

MDD_SPIND_MIN_VELO_G25	SD 43210: $SA_MDD_SPIND_MIN_VELO_G25				
Minimum spindle speed at G25 (master spindle)					
rev/min, user defined				Double	rw
Multi-line: no			1		

MDD_WORKAREA_LIMIT_MINUS	SD 43430: $SA_MDD_WORKAREA_LIMIT_MINUS				
Working area limitation in the negative direction					
mm, inch, user defined				Double	rw
Multi-line: yes	Number of machine axis		1		

MDD_WORKAREA_LIMIT_PLUS	SD 43420: $SA_MDD_WORKAREA_LIMIT_PLUS				
Working area limitation in the positive direction					
mm, inch, user defined				Double	rw
Multi-line: yes	Number of machine axis		1		

3.9 Parameter data

3.9.1 Area N, Block RP : Arithmetic parameters

OEM-MMC: Linkitem /NckParameter/...

Arithmetic parameters are specific, predefined variables which are addressed with the address R and subsequent number. The contents and meaning of arithmetic parameters is specified by person programming the part program. The number of parameters can be set in the machine data 18156 (MM_NUM_R_PARAM_NCK).

RG	$RG[x] x = ParameterNo				PA
Global R variables					
-				Double	rw
Multi-line: yes	R number		$MN_MM_NUM_R_PARAM_NCK		

3.9.2 Area C, Block RP : Arithmetic parameters

OEM-MMC: Linkitem /ChannelParameter/...

Arithmetic parameters are special predefined variables which are addressed with the letter R followed by a number. The contents and meaning of an arithmetic parameter are defined by the programmer of a part program. 100 R variables are defined by default. The number of R variables can be set via machine data 28050 (MM_NUM_R_PARAM).

rpa	$R[x] x = ParameterNo				PA
R variables					
-				Double	rw
Multi-line: yes	R number + 1		MM_NUM_R_PARAM + 1		

3.9.3 Area C, Block VSYN : Channel-specific user variables for synchronous actions

OEM-MMC: Linkitem /ChannelSelectedFunctionData/...

This module contains channel-specific user variables for synchronous actions

acFifoN	$AC_FIFOx[y] , x = FIFONo (1-10) y = ParameterNo				
FIFO variable for synchronous actions (Note: SYNACT only) The number of columns depends on the number of FIFOs					
-				Double	r
Multi-line: yes	1=2: access to the first element read in 3: access to the last element read in 4: sum of all FIFO elements 5: number of elements available in FIFO 6: current write index in relation to start of FIFO 7 etc: FIFO contents		MD $MC_MM_LEN_AC_FIFO+6		

acMarker					
replaced by acMarkerL)					
-				UWord	r
Multi-line: yes	Number of the flag		MD $MC_MM_NUM_AC_MARKER		

acMarkerL	$AC_MARKER[n]				
Flag variable, counter for synchronous actions (Note: SYNACT only)					
-				Long Integer	rw
Multi-line: yes	Number of the flag		MD $MC_MM_NUM_AC_MARKER		

acParam	$AC_PARAM[x] x = ParameterNo				
Dynamic parameters for motion-synchronous actions (Note: only with SYNACT)					
-				Double	rw
Multi-line: yes	Number of the parameter		MD $MC_MM_NUM_AC_PARAM		

acSystemMarkerL					
Flag variable, counter for motion-synchronous actions (Note: only with SYNACT) Reserved for system.					
-				Long Integer	rw
Multi-line: yes	Number of the flag		MD $MC_MM_NUM_AC_MARKER		

acSystemParam					
Dynamic parameters for motion-synchronous actions (Note: only with SYNACT) Reserved for system.					
-				Double	rw
Multi-line: yes	Number of the parameter		MD $MC_MM_NUM_AC_PARAM		

3.10 Diagnostics data

3.10.1 Area N, Block DIAGN : Global diagnostic data

OEM-MMC: Linkitem /NckChannelDiagnose/...

This module contains information about global NC diagnostic data.

Net times: Time without interrupts by higher priority time levels.

Gross times: Time with interrupts by higher priority time levels.

Time levels in order of their priority: position controller, interpolator, block preparation.

actCycleTimeBrut					
Total of the current gross run times of all channels in ms.					
ms	0	0		Double	r
Multi-line: yes	Selects a specific SW task on the NCK: Line index 1: SERVO Line index 2: IPO Line index 3: VL Line index 4: PLC Line index 5: SYNACT Line index 6: COS Line index 7: DRIVE (low priority) Line index 8: EXCOM (domain services) Line index 9: Reserved Line index 10: Reserved Line index 11: INT (compile cycles in the interpreter) Line index 12: EES (EES - async. - subtask)		12		

actCycleTimeNet					
Total of the current net run times of all channels in ms.					
ms	0	0		Double	r
Multi-line: yes	Selects a specific SW task on the NCK: Line index 1: SERVO Line index 2: IPO Line index 3: VL Line index 4: PLC Line index 5: SYNACT Line index 6: COS Line index 7: DRIVE (low prior.) Line index 8: EXCOM (domain services) Line index 9: CYCLE (cyclic tasks SERVO+IPO+soft PLC times Line index 10: NCK (NCK in total Line index 11: INT (compile cycles in the interpreter) Line index 12: EES (EES - async. - subtask)		12		

actNckLoad					
NC load imposed by cyclic tasks (position controller, interpolator and possibly soft PLC). The load must not be too high, so that even low priority tasks, such as communication of the data for the display, can be performed. The value is based on actCycleTimeNet with line=CYCLE and brings this in relation to the taskCycleTime with line=CYCLE.					
%				Double	r
Multi-line: yes	1		1		

aveCycleTimeNet					
Average net run time in ms					
ms				Double	r
Multi-line: yes	Selects a specific SW task on the NCK: Line index 1: SERVO Line index 2: IPO Line index 3: VL Line index 4: PLC Line index 5: SYNACT Line index 6: COS Line index 7: DRIVE (low prior.) Line index 8: EXCOM (domain services) Line index 9: CYCLE (cyclic tasks SERVO+IPO+ soft PLC times Line index 10: NCK (NCK in total) Line index 11: INT (compile cycles in the interpreter) Line index 12: EES (EES - async. - subtask)		12		

aveNckLoad					
Average NC load imposed by cyclic tasks (position controller, interpolator and possibly soft PLC). The value is based on aveCycleTimeNet with line=CYCLE and brings this in relation to the taskCycleTime with line=CYCLE.					
%				Double	r
Multi-line: yes	1		1		

dp611USpecAccChangeCnt					
The counter is incremented if the NCK changes the available ACC information					
-	0			Long Integer	r
Multi-line: yes	1		1		

dp611USpecAccKey					
Version and type information about available ACC contents					
-	0			Long Integer	r
Multi-line: yes	Drive number		maxnumDrives		

dp611USpecAccMask					
Bit-coded screenform indicating the drives for which special ACC files are available Bit 0 == 1 -> A special ACC is available for drive with log. drive number 1.					
-	0			Long Integer	r
Multi-line: yes	1		1		

dp611USpecAccPath					
Path in which the ACC files are stored in the NCK file system. This path might be empty later on if the files are to be supplied from the active file system. Current equivalent value: /_N_VS_DIR					
-	0			String [32]	r
Multi-line: yes	1		1		

dpAxisCfgMachAxisNr					
Machine axis !!CAUTION NCU LINK!!					
-	0	0	INT32_MAX	Long Integer	r
Multi-line: yes	Axis number		dpAxisCfgNumAxes		

dpAxisCfgNumAxes					
Number of axes entered in the system					
-	0	0	INT32_MAX	Long Integer	r
Multi-line: yes	1		1		

dpAxisCfgValid					
Axis info is available 0=Information is not available 1=Information is available					
-	0	0	1	Long Integer	r
Multi-line: yes	1		1		

dpAxisStateCtrlout					
Status of output drivers. 0=no axis status assigned 1=axis status assigned 2=axis status is cyclical 3=axis status assigned and cyclical					
-	0	0	3	UWord	r
Multi-line: yes	Axis number		dpAxisCfgNumAxes		

dpAxisStateEnc1					
Status encoder 1 driver 0=no axis status assigned 1=axis status assigned 2=axis status is cyclical 3=axis status assigned and cyclical					
-	0			UWord	r
Multi-line: yes	Axis number		dpAxisCfgNumAxes		

dpAxisStateEnc2					
Status encoder 2 driver 0=no axis status assigned 1=axis status assigned 2=axis status is cyclical 3=axis status assigned and cyclical					
-	0			UWord	r
Multi-line: yes	Axis number		dpAxisCfgNumAxes		

dpAxisStateLifeCntErrCtrlout					
This data counts the number of position control cycles since failure of the sign-of-life signal 0 to n= number of position control cycles since failure of the sign-of-life signal					
-	0	0	INT32_MAX	Long Integer	r
Multi-line: yes	Axis number		dpAxisCfgNumAxes		

dpAxisStateLifeCntErrEnc1					
This data counts the number of position control cycles since failure of the sign-of-life signal 0 to n= number of position control cycles since failure of the sign-of-life signal					
-	0			Long Integer	r
Multi-line: yes	Axis number		dpAxisCfgNumAxes		

dpAxisStateLifeCntErrEnc2					
This data counts the number of position control cycles since failure of the sign-of-life signal 0 to n= number of position control cycles since failure of the sign-of-life signal					
-	0			Long Integer	r
Multi-line: yes	Axis number		dpAxisCfgNumAxes		

dpBusCfgBaudrate					
Baud rate on DP bus (bit/s) The permissible baud rates are determined by the Profibus standard (DIN19245 EN50170)					
Hz	0			Double	r
Multi-line: yes	Bus number		dpBusCfgNumBuses		

dpBusCfgBusNo					
Bus number of the bus; used for conversion of "Bus index"=1...dpBusCfgNumBuses to "Bus number" All permissible bus numbers are possible: 1 = 1.DP bus on the PLC 2 = 2.DP / MPI bus on the PLC 3 = Virtual PROFIBUS 4 = Isochronous real-time Ethernet (reserved)					
-	0	0	4	Long Integer	r
Multi-line: yes	1		1		

dpBusCfgCycleTime					
The time required by the master to scan all slaves once (request, response), until the cycle starts from the beginning again.					
s, user defined	0	0	DOUBLE_MAX	Double	r
Multi-line: yes	Bus number		dpBusCfgNumBuses		

dpBusCfgDataExTime					
Data exchange time in [s,s,userdef]					
s, user defined	0	0	DOUBLE_MAX	Double	r
Multi-line: yes	Bus number		dpBusCfgNumBuses		

dpBusCfgNumBuses					
Number of DP buses Currently only one bus standardized acc. to Profibus DP standard					
-	0	0	1	Long Integer	r
Multi-line: yes	1		1		

dpBusCfgValid					
Bus configuration data are available TRUE= data exist and are initialized FALSE= no data exist					
-	0	0	1	Long Integer	r
Multi-line: yes	1		1		

dpBusStateAccessDurationAct					
Current access time to communications buffer for DP master					
-	0			Long Integer	r
Multi-line: yes	Bus number		dpBusCfgNumBuses		

dpBusStateAccessDurationMax					
Maximum access time to communications buffer for DP master					
-	0			Long Integer	r
Multi-line: yes	Bus number		dpBusCfgNumBuses		

dpBusStateAccessDurationMin					
Minimum access time to communications buffer for DP master					
-	0			Long Integer	r
Multi-line: yes	Bus number		dpBusCfgNumBuses		

dpBusStateAccessErrCnt1					
Number of bus access errors of type 1 since NCK Start					
-	0			Long Integer	r
Multi-line: yes	Bus number		dpBusCfgNumBuses		

dpBusStateAccessErrCnt2					
Number of bus access errors of type 2 since NCK Start					
-	0			Long Integer	r
Multi-line: yes	Bus number		dpBusCfgNumBuses		

dpBusStateAvgCycleBetweenErr1					
Average number of cycles between two bus access errors of type 1					
-	0			Long Integer	r
Multi-line: yes	Bus number		dpBusCfgNumBuses		

dpBusStateAvgCycleBetweenErr2					
Average number of cycles between two bus access errors of type 2					
-	0			Long Integer	r
Multi-line: yes	Bus number		dpBusCfgNumBuses		

dpBusStateCycleCnt					
Number of bus cycles since NCK Start					
-	0			Long Integer	r
Multi-line: yes	Bus number		dpBusCfgNumBuses		

dpBusStateDpmAction					
Indicator for operating progress of DP M					
-	0			Long Integer	r
Multi-line: yes	Bus number		dpBusCfgNumBuses		

dpBusStateDpmActual					
Current status of DP M bus - controlled by DP M					
-	0			UWord	r
Multi-line: yes	Bus number		dpBusCfgNumBuses		

dpBusStateDpmCtrl					
Booting status of processor for DP Master dpcadmin					
-	0			UWord	r
Multi-line: yes	Bus number		dpBusCfgNumBuses		

dpBusStateDpmError					
Error on status transitions					
-	0			Long Integer	r
Multi-line: yes	Bus number		dpBusCfgNumBuses		

dpBusStateDpmPrjCnt					
Modification counter for new DP configurations. Suggested use: *) Read modification counter (1) *) Read out configuring data *) Read modification counter (2) *) If the modification counters in (1) and (2) are identical and both display "valid", the data read from HW-Config will be consistent. even values -> configuration invalid uneven values -> configuration valid					
-	0			UWord	r
Multi-line: yes	Bus number		dpBusCfgNumBuses		

dpBusStateDpmRequest					
Desired status of DP M bus - request from HOST					
-	0			UWord	r
Multi-line: yes	Bus number		dpBusCfgNumBuses		

dpBusStateNumActiveSlaves					
This data indicates how many slaves can currently be accessed via the bus. This value is updated in online operation. The number of slaves on the bus is determined by the Profibus standard (DIN19245 EN50170)					
-	0	0	125	Long Integer	r
Multi-line: yes	Bus number		dpBusCfgNumBuses		

dpClientCfgId					
Identification client NCK/PLC/3RD					
-	0			UWord	r
Multi-line: yes	Client number		dpClientCfgNumClnt		

dpClientCfgNumClnt					
Number of clients					
-	0	0	INT32_MAX	Long Integer	r
Multi-line: yes	1		1		

dpClientCfgValid					
Client information is available 0=no client information available 1=client information is available					
-	0	0	1	Long Integer	r
Multi-line: yes	1		1		

dpClientStateComm					
Client status incl. output release 0=None output enable 1=Client state output enable					
-	0			UWord	r
Multi-line: yes	Client number		dpClientCfgNumClnt		

dpSlaveCfgAssignBus					
Bus number of the slave					
-	0			UWord	r
Multi-line: yes	Slave number		dpSlaveCfgNumSlaves		

dpSlaveCfgBusAddr					
The address of the slave on the bus. In addition to its own address, every slave has a broadcast address via which all salves can be addressed. The broadcast address is not available for individually addressing a single slave. 127: Broadcast address					
-	0	0	127	UWord	r
Multi-line: yes	Slave number		dpSlaveCfgNumSlaves		

dpSlaveCfgDataExchangeTime					
Time for the end of cyclical data transfer See dpSlaveCfgMasterAppCycTime					
s, user defined	0			Double	r
Multi-line: yes	Slave number		dpSlaveCfgNumSlaves		

dpSlaveCfgInputTime					
Time for actual-value sensing See dpSlaveCfgMasterAppCycTime					
s, user defined	0			Double	r
Multi-line: yes	Slave number		dpSlaveCfgNumSlaves		

dpSlaveCfgIsochronModeSupport					
Gives information whether the slave has been configured for isochronous mode on the PROFIBUS. 0: Isochronous mode not configured 1: Isochronous mode configured					
-	0	0		Long Integer	r
Multi-line: yes	Slave number		dpSlaveCfgNumSlaves		

dpSlaveCfgMasterAppCycTime					
Position controller cycle. For a detailed description, please refer to PROFIDRIVE PROFIL ANTRIEBSTECHNIK (Edition: V1.2 Draft, April 1999) Section 7 See PROFIDRIVE PROFIL ANTRIEBSTECHNIK (Edition: V1.2 Draft, April 1999) Section 7					
s, user defined	0			Double	r
Multi-line: yes	Slave number		dpSlaveCfgNumSlaves		

dpSlaveCfgNumSlaves					
Number of slaves configured in SDB1xxx. This value may not match the actual number of slaves connected to the bus. The number of slaves which can be configured for bus connection is determined by Profibus standard (DIN19245 EN50170).					
-	0	0	125	Long Integer	r
Multi-line: yes	1		1		

dpSlaveCfgOutputTime					
Time for setpoint acceptance See dpSlaveCfgMasterAppCycTime					
s, user defined	0			Double	r
Multi-line: yes	Slave number		dpSlaveCfgNumSlaves		

dpSlaveCfgProfibusCycleTime					
Bus cycle time See dpSlaveCfgMasterAppCycTime					
s, user defined	0			Double	r
Multi-line: yes	Slave number		dpSlaveCfgNumSlaves		

dpSlaveCfgValid					
This data indicates whether the slave data structure has already been initialized. The structure is initialized when a slave configuration or status data is accessed. Scanning dpSlaveCfgValid also activates initialization of the structure. True: Slave data are available False: Slave data are not available					
-	0	0	1	Long Integer	r
Multi-line: yes	1		1		

dpSlaveIdentNo					
Ident number of the slave					
-	0			UWord	r
Multi-line: yes	Slave number		dpSlaveCfgNumSlaves		

dpSlaveIdentNoEx					
The extended ID no. of the PROFIBUS slave helps to identify the PROFIBUS slaves not officially classified as such and therefore lack specificiation dpSlaveIdentNo.					
-	0			UWord	r
Multi-line: yes	Slave number		dpSlaveCfgNumSlaves		

dpSlaveStateComm					
The slave is active on the bus once the drive assigned to the slave has successfully logged on to the bus. True: Slave on bus False: Slave not on bus					
-	0	0	1	UWord	r
Multi-line: yes	Slave number		dpSlaveCfgNumSlaves		

dpSlaveStateIncCnt					
The incarnation counter of the slave is increased by one each time the slave is included in the bus. If the slave drops out of the bus, this counter is not changed. After the first time it has gone into the bus (that is the first operational status of the slave), the value is 1. In case of an area overflow, the count restarts at 0. This only functions with slaves which contain at least one assigned NC axis. In the case of other slaves (pure I/O slaves, or axes controlled by the PLC), this values remains at 0. From 0 (starting value after Restart) to a maximum of 2147483647 (2^31-1).					
-	0	0	2147483647	Long Integer	r
Multi-line: yes	Slave number		dpSlaveCfgNumSlaves		

dpSlaveStateSync					
The drive linked to this slave is operating in cyclic mode. Slaves without a drive are defined as "non-cyclical". True: Cyclical False: Non-cyclical					
-	0	0	1	UWord	r
Multi-line: yes	Slave number		dpSlaveCfgNumSlaves		

dpSlaveVendorId					
PROFIBUS: Always returns 0 PROFINET: Manufacturer's number of the device					
-	0			UWord	r
Multi-line: yes	Slave number		dpSlaveCfgNumSlaves		

dpSlotCfgAssignAxis					
This data supplies the axis indices of the drive, encoder 1 and encoder 2 for access in the Axis-Assign-Table. The 32-bit value consists of 4 bytes with the following meaning: Byte0(bits 0-7) = axis index of axis Byte1(bits 8-15) = axis index, encoder 1 Byte2(bits 16-23)= axis index, encoder 2 Byte3(bits 24-31)= provided for future extensions. A byte with the value 0xFF indicates that no axis index is defined for the relevant slot.					
-	255	0	32	Long Integer	r
Multi-line: yes	Slot number (PROFINET: Subslot number)		dpSlotCfgNumSlots		

dpSlotCfgAssignBus					
Bus number assigned to this slot Since only one bus is currently supported by Profibus DP, there is only one bus to which all slots are assigned.					
-	0	0	1	Long Integer	r
Multi-line: yes	Slot number (PROFINET: Subslot number)		dpSlotCfgNumSlots		

dpSlotCfgAssignClient					
This data supplies the clientIndex for accessing the Client Assign table. 0=no assignment possible (this applies to diagnostic and PKW slots) >0 assignment exists					
-	0	0	2	Long Integer	r
Multi-line: yes	Slot number (PROFINET: Subslot number)		dpSlotCfgNumSlots		

dpSlotCfgAssignMaster					
Number of master to which this slot is assigned Since only one bus is currently supported by Profibus DP and only one Class 1 Master exists per bus, there is only one master to which all slots are assigned.					
-	0	0	1	Long Integer	r
Multi-line: yes	Slot number (PROFINET: Subslot number)		dpSlotCfgNumSlots		

dpSlotCfgAssignSlave					
This data contains the bus address of the slave belonging to the nth slot. All legal slave addresses can be specified					
-	0	0	125	Long Integer	r
Multi-line: yes	Slot number (PROFINET: Subslot number)		dpSlotCfgNumSlots		

dpSlotCfgIoType					
I/O identifier 0 = input slot 1 = output slot 2 = diagnosis slot					
-	0	0	2	UWord	r
Multi-line: yes	Slot number (PROFINET: Subslot number)		dpSlotCfgNumSlots		

dpSlotCfgLength					
Length in number of bytes					
-	0	0	32	Long Integer	r
Multi-line: yes	Slot number (PROFINET: Subslot number)		dpSlotCfgNumSlots		

dpSlotCfgLogBaseAddress					
The logical basic address of the slot is assigned during configuration. Although it is not needed on the bus for data transfer purposes, this address is the only means by which a unique link can be created between the NCK and bus nodes.					
-	0	0	UINT16_MAX	UWord	r
Multi-line: yes	Slot number (PROFINET: Subslot number)		dpSlotCfgNumSlots		

dpSlotCfgNumSlots					
The total number of all slots configured in the system is stored in this data. 0 (lower limit) up to INT32_MAX(upper limit); Note that a slave cannot support more than 256 slots.					
-	0	0	INT32_MAX	Long Integer	r
Multi-line: yes	1		1		

dpSlotCfgPNSlotNr					
PROFIBUS: Not used PROFINET: Slot number within the IO device					
-	0	0	255	UWord	r
Multi-line: yes	PROFINET: Subslot number		dpSlotCfgNumSlots		

dpSlotCfgSlaveAddress					
This data contains the bus address of the slave to which this slot is assigned. Several slots may have the same slave address. The number of available addresses on the bus is determined by the Profibus standard (DIN19245 EN50170).					
-	0	0	125	UWord	r
Multi-line: yes	Slot number (PROFINET: Subslot number)		dpSlotCfgNumSlots		

dpSlotCfgSlotNr					
PROFIBUS: Slot number within the slave PROFINET: Subslot number within the IO device A maximum total of 256 slots can be assigned to each slave. 0: Diagnostic slot 2: Diagnostic slot 4: 1st data slot					
-	0	0	255	UWord	r
Multi-line: yes	Slot number (PROFINET: Subslot number)		dpSlotCfgNumSlots		

dpSlotCfgValid					
The slot data structure (CcIdent) exists and is initialized. True: Data are valid False: Data are invalid or not initialized					
-	0	0	1	Long Integer	r
Multi-line: yes	1		1		

dpSlotStateComm					
Status of slots (ok, failed, not processed by the NCK) 0= no sign of life 1= sign of life 2= not processed by NCK					
-	0	0	1	UWord	r
Multi-line: yes	Slot number (PROFINET: Subslot number)		dpSlotCfgNumSlots		

dpSlotStateRecvTelegram					
Bit pattern of this slot received by the master in the form of a hexadecimal string					
-	0			String [198]	r
Multi-line: yes	Slot number (PROFINET: Subslot number)		dpSlotCfgNumSlots		

dpSlotStateSendTelegram					
Bit pattern of this slot sent to the slave in the form of a hexadecimal string Transmitted message frame					
-	0			String [198]	r
Multi-line: yes	Slot number (PROFINET: Subslot number)		dpSlotCfgNumSlots		

dpSlotStateTelegramType					
Message frame type of slot 0 = Message frame type unknown					
-	0	0	UINT16_MAX	UWord	r
Multi-line: yes	Slot number (PROFINET: Subslot number)		dpSlotCfgNumSlots		

dpSysCfgAvailable					
This data specifies whether the system has been generated with DP Adapter and/or DP Master 0= Neither DPA nor DPM available 1= DPA available 2= DPM available 3= DPA and DPM available					
-	0	0	3	UWord	r
Multi-line: yes	1		1		

dpSysCfgNumMaster					
Number of masters There is only one master per bus with DP. Since only 1 bus is currently permitted by the bus standard, there can only be a maximum of one master.					
-	0	0	1	Long Integer	r
Multi-line: yes	1		1		

dpSysCfgValid					
This data indicates whether the configuration data are valid and initialized. TRUE or FALSE					
-	0	0	1	Long Integer	r
Multi-line: yes	1		1		

dpSysCfgVersionDpm					
Version number of DP M SW as numerical value					
-	0			Double	r
Multi-line: yes	Number of the master		dpSysCfgNumMaster		

dpSysCfgVersionDpr					
Actual version Dpr (inaccessible in earlier SW)					
-	0			Double	r
Multi-line: yes	Number of the master		dpSysCfgNumMaster		

dpSysCfgVersionDprEx					
DPR_SS_VERSION is a version number stored in the NCK which can be read out via this variable.					
-	0			Double	r
Multi-line: yes	Number of the master		dpSysCfgNumMaster		

dpSysCfgVersionHost					
This data contains the version number of the host SW as a numerical value					
-	0	0	UINT16_MAX	Double	r
Multi-line: yes	Number of the master		dpSysCfgNumMaster		

dpSysStateDpmInit					
There are three different initialization states: REQUEST, ACKNOWLEDGE and ERROR					
-	0			UWord	r
Multi-line: yes	Number of the master		dpSysCfgNumMaster		

errCodeSetNrGen					
Selection of error code set to be used in the case of communication errors. The selection is client-specific, the client is identified by the sender address. 1: P1-compatible codes (default) 0-4: As 1 5: P5-compatible codes 6: Current codes (from P6) 7-100: Reserved					
-	0	0		UWord	rw
Multi-line: yes	1		1		

errCodeSetNrPi					
Selection of error code set to be used by PI Services in the case of communication errors. The selection is client-specific, the client is identified by the sender address. 0: P1-compatible code 5: P5-compatible code 6: P6-compatible code					
-	0	0		UWord	rw
Multi-line: yes	1		1		

isPersistencyOverflowIpo	$AN_PERSDIAG[row-1,11]				
Value=1: At the time of power-fail/power off, overflow of the power-fail buffer for changes of persistent data in synchronous actions was pending. The last data change before power-fail/power off has been lost.					
-	0	0		Long Integer	r
Multi-line: yes	1: Overflow of the synchronous action power-fail buffer at power-fail/power off		1		

isPersistencyOverflowPrep	$AN_PERSDIAG[row-1,9]				
Value=1: At the time of power-fail/power off, overflow of the power-fail buffer was pending in the preprocessing. The last data change before power-fail/power off has been lost.					
-	0	0		Long Integer	r
Multi-line: yes	1: Overflow of the preprocessing power-fail buffer at power-fail/ power off		1		

isPersistencyOverflowToolChange	$AN_PERSDIAG[row-1,10]				
Value=1: At the time of power-fail/power off, overflow of the power-fail buffer was pending in the preprocessing. The last tool/magazine data change before power-fail/power off has been lost.					
-	0	0		Long Integer	r
Multi-line: yes	1: Overflow of the tool change power-fail buffer at power-fail/ power off		1		

maxCycleTimeBrut					
Total of the maximum gross run times of all channels in ms.					
ms	0	0		Double	r
Multi-line: yes	Selects a specific SW task on the NCK: Line index 1: SERVO Line index 2: IPO Line index 3: VL Line index 4: PLC Line index 5: SYNACT Line index 6: COS Line index 7: DRIVE (low priority) Line index 8: EXCOM (domain services) Line index 9: Reserved Line index 10: Reserved Line index 11: INT (compile cycles in the interpreter) Line index 12: EES (EES - async. - subtask)		12		

maxCycleTimeNet					
Total of the maximum net run times of all channels in ms.					
ms	0	0		Double	r
Multi-line: yes	Selects a specific SW task on the NCK: Line index 1: SERVO Line index 2: IPO Line index 3: VL Line index 4: PLC Line index 5: SYNACT Line index 6: COS Line index 7: DRIVE (low prior.) Line index 8: EXCOM (domain services) Line index 9: CYCLE (cyclic tasks SERVO+IPO+soft PLC times Line index 10: NCK (NCK in total Line index 11: INT (compile cycles in the interpreter) Line index 12: EES (EES - async. - subtask)		12		

maxNckLoad					
Maximum NC load imposed by cyclic tasks (position controller, interpolator and possibly soft PLC). The value is based on maxCycleTimeNet with line=CYCLE and brings this in relation to the taskCycleTime with line=CYCLE.					
%				Double	r
Multi-line: yes	1		1		

minCycleTimeBrut					
Total of the minimum gross run times of all channels in ms.					
ms	0	0		Double	r
Multi-line: yes	Selects a specific SW task on the NCK: Line index 1: SERVO Line index 2: IPO Line index 3: VL Line index 4: PLC Line index 5: SYNACT Line index 6: COS Line index 7: DRIVE (low priority) Line index 8: EXCOM (domain services) Line index 9: Reserved Line index 10: Reserved Line index 11: INT (compile cycles in the interpreter) Line index 12: EES (EES - async. - subtask)		12		

minCycleTimeNet					
Total of the minimum net run times of all channels in ms.					
ms	0	0		Double	r
Multi-line: yes	Selects a specific SW task on the NCK: Line index 1: SERVO Line index 2: IPO Line index 3: VL Line index 4: PLC Line index 5: SYNACT Line index 6: COS Line index 7: DRIVE (low prior.) Line index 8: EXCOM (domain services) Line index 9: CYCLE (cyclic tasks SERVO+IPO+soft PLC times Line index 10: NCK (NCK in total Line index 11: INT (compile cycles in the interpreter) Line index 12: EES (EES - async. - subtask)		12		

minNckLoad					
Minimum NC load imposed by cyclic tasks (position controller, interpolator and possibly soft PLC). The value is based on minCycleTimeNet with line=CYCLE and brings this in relation to the taskCycleTime with line=CYCLE.					
%				Double	r
Multi-line: yes	1		1		

nckCapabilities					
Describes the NCK functionality Bit0=1: With Huffman algorithm compressed files can be transferred (this corresponds to instruction ";$COMPR=HUFFMAN1" during download) Bit1=1: The protocol of optimized upload is supported					
-	0	0		UWord	r
Multi-line: yes	1		1		

nckCompileSwitches					
Selected NCK compiler switches Bit0: NDEBUG Bit1: NOTRACES Bit2: EMBARGO Bit3: TARGET					
-				UWord	r
Multi-line: yes	1		1		

noOfPersistencyCollisions					
If a persistence operation (= flush) is triggered, although an asynchronous persistence operation with the same ID has not yet been executed, then the value of this variable is incremented.					
-	0	0		Long Integer	r
Multi-line: yes	1: Sum of the individual functions 2: Passive file system function 3: Active file system function 4: Machine data function		1		

noOfPersistencyEntriesIpo	$AN_PERSDIAG[row-1,14]				
Number of data entries in the power-fail buffer for changes to persistent data in synchronous actions					
-	0	0		Long Integer	r
Multi-line: yes	1: Number of data entries in the synchronous action power-fail buffer		1		

noOfPersistencyEntriesPrep	$AN_PERSDIAG[row-1,12]				
Number of data entries in the power-fail buffer in the preprocessing					
-	0	0		Long Integer	r
Multi-line: yes	1: Number of data entries in the preprocessing power-fail buffer		1		

noOfPersistencyEntriesToolChange	$AN_PERSDIAG[row-1,13]				
Number of data entries in the power-fail buffer for the tool change data changes in IPO					
-	0	0		Long Integer	r
Multi-line: yes	1: Number of data entries in the tool change power-fail buffer		1		

noOfPersistencyOverflowIpo	$AN_PERSDIAG[row-1,8]				
Number of overflows of the power-fail buffer for the changes to persistent data in synchronous actions (Value > 0 indicates that the buffer is too small -> increase $MN_MM_ACTFILESYS_LOG_FILE_MEM[2] if possible)					
-	0	0		Long Integer	r
Multi-line: yes	1: Number of overflows of the synchronous action power-fail buffer		1		

noOfPersistencyOverflowPrep	$AN_PERSDIAG[row-1,6]				
Number of overflows of the power-fail buffer in the preprocessing (Value > 0 indicates that the buffer is too small -> increase $MN_MM_ACTFILESYS_LOG_FILE_MEM[0] if possible)					
-	0	0		Long Integer	r
Multi-line: yes	1: Number of overflows of the preprocessing power-fail buffer		1		

noOfPersistencyOverflowToolChange	$AN_PERSDIAG[row-1,7]				
Number of overflows of the power-fail buffer for the tool change data changes (Value > 0 indicates that the buffer is too small -> increase $MN_MM_ACTFILESYS_LOG_FILE_MEM[1] if possible)					
-	0	0		Long Integer	r
Multi-line: yes	1: Number of overflows of the tool change power-fail buffer		1		

noOfPersistencyReq	$AN_PERSDIAG[row-1,0]				
Number of persistence operations					
-	0	0		Long Integer	r
Multi-line: yes	Synchronous flush calls 1: Sum of the individual functions 2: Passive file system function 3: Active file system function 4: Machine data function Asynchronous flush calls (blocking component) 11: Sum of the individual functions 12: Passive file system function 13: Active file system function 14: Machine data function Collisions during flush calls 21: Sum of the individual functions 22: Passive file system function 23: Active file system function 24: Machine data function 31: Reserved 32: Reserved 33: Reserved 34: Reserved		34		

noOfPersistencyReqFailed	$AN_PERSDIAG[row-1,1]				
Number of failed persistence operations					
-	0	0		Long Integer	r
Multi-line: yes	Synchronous flush calls 1: Sum of the individual functions 2: Passive file system function 3: Active file system function 4: Machine data function Asynchronous flush calls (blocking component) 11: Sum of the individual functions 12: Passive file system function 13: Active file system function 14: Machine data function The following indices are reserved 21: Reserved 22: Reserved 23: Reserved 24: Reserved 31: Reserved 32: Reserved 33: Reserved 34: Reserved		34		

persistencyTimeAverage	$AN_PERSDIAG[row-1,4]				
Average time for making data persistent					
s, user defined	0	0		Double	r
Multi-line: yes	Synchronous flush calls 1: Sum of the individual functions 2: Passive file system function 3: Active file system function 4: Machine data function Asynchronous flush calls (blocking component) 11: Sum of the individual functions 12: Passive file system function 13: Active file system function 14: Machine data function Collisions during flush calls 21: Sum of the individual functions 22: Passive file system function 23: Active file system function 24: Machine data function Asynchronous flush calls (total runtime) 31: Sum of the individual functions 32: Passive file system function 33: Active file system function 34: Machine data function		34		

persistencyTimeMaximal	$AN_PERSDIAG[row-1,5]				
Maximum time for making data persistent					
s, user defined	0	0		Double	r
Multi-line: yes	Synchronous flush calls 1: Sum of the individual functions 2: Passive file system function 3: Active file system function 4: Machine data function Asynchronous flush calls (blocking component) 11: Sum of the individual functions 12: Passive file system function 13: Active file system function 14: Machine data function Collisions during flush calls 21: Sum of the individual functions 22: Passive file system function 23: Active file system function 24: Machine data function Asynchronous flush calls (total runtime) 31: Sum of the individual functions 32: Passive file system function 33: Active file system function 34: Machine data function		34		

persistencyTimeMinimal	$AN_PERSDIAG[row-1,3]				
Minimum time for making data persistent					
s, user defined	0	0		Double	r
Multi-line: yes	Synchronous flush calls 1: Sum of the individual functions 2: Passive file system function 3: Active file system function 4: Machine data function Asynchronous flush calls (blocking component) 11: Sum of the individual functions 12: Passive file system function 13: Active file system function 14: Machine data function Collisions during flush calls 21: Sum of the individual functions 22: Passive file system function 23: Active file system function 24: Machine data function Asynchronous flush calls (total runtime) 31: Sum of the individual functions 32: Passive file system function 33: Active file system function 34: Machine data function		34		

poweronTime	$AN_POWERON_TIME				
Time since last normal boot (in minutes)					
s, user defined	0.0			Double	rw
Multi-line: yes	1		1		

setupTime	$AN_SETUP_TIME				
Time since last "control system boot on default values" (in minutes). The timer is automatically set to zero on every "control system boot on default values".					
s, user defined	0.0			Double	rw
Multi-line: yes	1		1		

sumCycleTimeNet					
Sum of net run times in s					
s				Double	r
Multi-line: yes	Selects a specific SW task on the NCK: Line index 1: SERVO Line index 2: IPO Line index 3: VL Line index 4: PLC Line index 5: SYNACT Line index 6: COS Line index 7: DRIVE (low prior.) Line index 8: EXCOM (domain services) Line index 9: CYCLE (cyclic tasks SERVO+IPO+soft PLC times Line index 10: NCK (NCK in total Line index 11: INT (compile cycles in the interpreter) Line index 12: EES (EES - async. - subtask)		12		

taskCycleTime					
Cycle time of the task in ms					
ms				Double	r
Multi-line: yes	Selects a specific SW task on the NCK: Line index 1: SERVO Line index 2: IPO Line index 4: PLC Line index 6: COS Line index 9: CYCLE (total cycle in the NCK within which all cyclic tasks repeat) Line index 10: NCK (see CYCLE)		12		

totalPersistencyTime	$AN_PERSDIAG[row-1,2]				
Summated time for making data persistent					
s, user defined	0	0		Double	r
Multi-line: yes	Synchronous flush calls 1: Sum of the individual functions 2: Passive file system function 3: Active file system function 4: Machine data function Asynchronous flush calls (blocking component) 11: Sum of the individual functions 12: Passive file system function 13: Active file system function 14: Machine data function Collisions during flush calls 21: Sum of the individual functions 22: Passive file system function 23: Active file system function 24: Machine data function Asynchronous flush calls (total runtime) 31: Sum of the individual functions 32: Passive file system function 33: Active file system function 34: Machine data function		34		

3.10.2 Area C, Block DIAGN : Channel-specific diagnosis data

OEM-MMC: Linkitem /ChannelChannelDiagnose/...

This module contains information about channel-specific NC diagnostic data.

Net times: Time without interrupts by higher priority time levels.

Gross times: Time with interrupts by higher priority time levels.

Time levels in order of their priority: position controller, interpolator, block preparation.

acIpoBuf	$AC_IPO_BUF				
Level of IPO buffer (number of blocks)					
-	0	0		UWord	r
Multi-line: yes	1		1		

actCycleTimeBrut					
Current gross run time in ms.					
ms				Double	r
Multi-line: yes	Selects a specific SW task on the NCK: Line index 1: SERVO Line index 2: IPO Line index 3: VL Line index 4: PLC Line index 5: SYNACT Line index 6: COS Line index 7: DRIVE (low prior.) Line index 8: EXCOM (domain services) Line index 9: Reserved Line index 10: Reserved Line index 11: INT (compile cycles in the interpreter) Line index 12: EES (EES async. - subtask)		12		

actCycleTimeNet					
Current net run time in ms.					
ms				Double	r
Multi-line: yes	Selects a specific SW task on the NCK: Line index 1: SERVO Line index 2: IPO Line index 3: VL Line index 4: PLC Line index 5: SYNACT Line index 6: COS Line index 7: DRIVE (low prior.) Line index 8: EXCOM (domain services) Line index 9: CYCLE (SERVO +IPO times related to an IPO cycle) Line index 10: NCK (NCK in total related to an IPO cycle) This value is only available for solution line systems. This time is restricted by MD $NCK_PCOS_TIME_RATIO to a portion of the IPO cycle. Line index 11: INT (compile cycles in the interpreter) Line index 12: EES (EES - async. - subtask)		12		

aveCycleTimeNet					
Average net run time in s					
ms				Double	r
Multi-line: yes	Selects a specific SW task on the NCK: Line index 1: SERVO Line index 2: IPO Line index 3: VL Line index 4: PLC Line index 5: SYNACT Line index 6: COS Line index 7: DRIVE (low prior.) Line index 8: EXCOM (domain services) Line index 9: CYCLE (SERVO +IPO times related to an IPO cycle) Line index 10: NCK (NCK in total related to an IPO cycle) This value is only available for solution line systems. This time is restricted by MD $NCK_PCOS_TIME_RATIO to a portion of the IPO cycle. Line index 11: INT (compile cycles in the interpreter) Line index 12: EES (EES - async. - subtask)		12		

cuttingTime	$AC_CUTTING_TIME				
Tool operating time (in seconds): The operating time of the path axes excluding active rapid traverse is measured in all NC programs between NC Start and Program End/NC Reset. The measurement is also interrupted during an active dwell time. The timer is automatically set to zero every time the control boots on default values.					
s, user defined	0.0			Double	rw
Multi-line: yes	1		1		

cycleTime	$AC_CYCLE_TIME				
Runtime of selected NC program (in seconds): The runtime between NC Start and Program End / NC Reset is measured in the selected NC program. The timer is cleared when a new NC program is started.					
s, user defined	0.0			Double	rw
Multi-line: yes	1		1		

ipoBufLevel					
Fill level of the IPO buffer (integer value in %)					
%		0	100	UWord	r
Multi-line: yes	1		1		

maxCycleTimeBrut					
Maximum gross run time in ms.					
ms				Double	rw
Multi-line: yes	Selects a specific SW task on the NCK: Line index 1: SERVO Line index 2: IPO Line index 3: VL Line index 4: PLC Line index 5: SYNACT Line index 6: COS Line index 7: DRIVE (low prior.) Line index 8: EXCOM (domain services) Line index 9: Reserved Line index 10: Reserved Line index 11: INT (compile cycles in the interpreter) Line index 100: ALL (all tasks, only for write access) Line index 12: EES (EES - async. - subtask)		100		

maxCycleTimeBrutPo					
Maximum gross run time since cold restart in ms.					
ms				Double	rw
Multi-line: yes	Selects a specific SW task on the NCK: Line index 1: SERVO Line index 2: IPO Line index 3: VL Line index 4: PLC Line index 5: SYNACT Line index 6: COS Line index 7: DRIVE (low prior.) Line index 8: EXCOM (domain services) Line index 9: CYCLE (SERVO +IPO times related to an IPO cycle) Line index 10: NCK (NCK in total related to an IPO cycle) This value is only available for solution line systems. This time is restricted by MD $NCK_PCOS_TIME_RATIO to a portion of the IPO cycle. Line index 11: INT (compile cycles in the interpreter) Line index 100: ALL (all task, only for write access) Line index 12: EES (EES - async. - subtask)		100		

maxCycleTimeNet					
Maximum net run time in ms.					
ms				Double	rw
Multi-line: yes	Selects a specific SW task on the NCK: Line index 1: SERVO Line index 2: IPO Line index 3: VL Line index 4: PLC Line index 5: SYNACT Line index 6: COS Line index 7: DRIVE (low prior.) Line index 8: EXCOM (domain services) Line index 9: Reserved Line index 10: Reserved Line index 11: INT (compile cycles in the interpreter) Line index 100: ALL (all tasks, only for write access) Line index 12: EES (EES - async. - subtask)		100		

maxCycleTimeNetPo					
Maximum net run time since cold restart in ms.					
ms				Double	rw
Multi-line: yes	Selects a specific SW task on the NCK: Line index 1: SERVO Line index 2: IPO Line index 3: VL Line index 4: PLC Line index 5: SYNACT Line index 6: COS Line index 7: DRIVE (low prior.) Line index 8: EXCOM (domain services) Line index 9: CYCLE (SERVO +IPO times related to an IPO cycle) Line index 10: NCK (NCK in total related to an IPO cycle) This value is only available for solution line systems. This time is restricted by MD $NCK_PCOS_TIME_RATIO to a portion of the IPO cycle. Line index 11: INT (compile cycles in the interpreter) Line index 100: ALL (all task, only for write access) Line index 12: EES (EES - async. - subtask)		100		

minCycleTimeBrut					
Minimum gross run time in ms.					
ms				Double	rw
Multi-line: yes	Selects a specific SW task on the NCK: Line index 1: SERVO Line index 2: IPO Line index 3: VL Line index 4: PLC Line index 5: SYNACT Line index 6: COS Line index 7: DRIVE (low prior.) Line index 8: EXCOM (domain services) Line index 9: Reserved Line index 10: Reserved Line index 11: INT (compile cycles in the interpreter) Line index 100: ALL (all tasks, only for write access) Line index 12: EES (EES - async. - subtask)		100		

minCycleTimeBrutPo					
Minimum gross run time since cold restart in ms.					
ms				Double	rw
Multi-line: yes	Selects a specific SW task on the NCK: Line index 1: SERVO Line index 2: IPO Line index 3: VL Line index 4: PLC Line index 5: SYNACT Line index 6: COS Line index 7: DRIVE (low prior.) Line index 8: EXCOM (domain services) Line index 9: CYCLE (SERVO +IPO times related to an IPO cycle) Line index 10: NCK (NCK in total related to an IPO cycle) This value is only available for solution line systems. This time is restricted by MD $NCK_PCOS_TIME_RATIO to a portion of the IPO cycle. Line index 11: INT (compile cycles in the interpreter) Line index 100: ALL (all task, only for write access) Line index 12: EES (EES - async. - subtask)		100		

minCycleTimeNet					
Minimum net run time in ms.					
ms				Double	rw
Multi-line: yes	Selects a specific SW task on the NCK: Line index 1: SERVO Line index 2: IPO Line index 3: VL Line index 4: PLC Line index 5: SYNACT Line index 6: COS Line index 7: DRIVE (low prior.) Line index 8: EXCOM (domain services) Line index 9: CYCLE (SERVO +IPO times related to an IPO cycle) Line index 10: NCK (NCK in total related to an IPO cycle) This value is only available for solution line systems. This time is restricted by MD $NCK_PCOS_TIME_RATIO to a portion of the IPO cycle. Line index 11: INT (compile cycles in the interpreter) Line index 100: ALL (all task, only for write access) Line index 12: EES (EES - async. - subtask)		100		

minCycleTimeNetPo					
Minimum net run time since cold restart in ms.					
ms				Double	rw
Multi-line: yes	Selects a specific SW task on the NCK: Line index 1: SERVO Line index 2: IPO Line index 3: VL Line index 4: PLC Line index 5: SYNACT Line index 6: COS Line index 7: DRIVE (low prior.) Line index 8: EXCOM (domain services) Line index 9: CYCLE (SERVO +IPO times related to an IPO cycle) Line index 10: NCK (NCK in total related to an IPO cycle) This value is only available for solution line systems. This time is restricted by MD $NCK_PCOS_TIME_RATIO to a portion of the IPO cycle. Line index 11: INT (compile cycles in the interpreter) Line index 100: ALL (all task, only for write access) Line index 12: EES (EES - async. - subtask)		100		

operatingTime	$AC_OPERATING_TIME				
Total runtime of NC programs in Automatic mode (in seconds): The runtimes of all programs are summed between NC Start and Program End/NC Reset. The timer is set to zero on every control boot.					
s, user defined	0.0			Double	rw
Multi-line: yes	1		1		

sumCycleTimeNet					
Sum of net run times in ms					
ms				Double	r
Multi-line: yes	Selects a specific SW task on the NCK: Line index 1: SERVO Line index 2: IPO Line index 3: VL Line index 4: PLC Line index 5: SYNACT Line index 6: COS Line index 7: DRIVE (low prior.) Line index 8: EXCOM (domain services) Line index 9: CYCLE (SERVO +IPO times related to an IPO cycle) Line index 10: NCK (NCK in total related to an IPO cycle) This value is only available for solution line systems. This time is restricted by MD $NCK_PCOS_TIME_RATIO to a portion of the IPO cycle. Line index 11: INT (compile cycles in the interpreter) Line index 12: EES (EES - async. - subtask)		12		

3.10.3 Area N, Block ETPD : Data lists for protocolling

OEM-MMC: Linkitem /NckProtocolData/...

Data lists for protocolling.This module allows to access several lines or rows at a time.

area					
Variable specification of nth OPI data in the list: area					
-				UWord	rw
Multi-line: yes	2 + 5 * (n-1)		2 + 5 * (numData- 1)		

col					
Variable specification of nth OPI data in list: col					
-				UWord	rw
Multi-line: yes	4 + 5 * (n-1)		4 + 5 * (numData- 1)		

numData					
Number of data in the list. <= maxnumTraceProtData					
-		0	maxnumTraceProtData	UWord	rw
Multi-line: yes	1		1		

row					
Variable specification of nth OPI data in list: row					
-				UWord	rw
Multi-line: yes	5 + 5 * (n-1)		5 + 5 * (numData- 1)		

type					
Low byte:Variable specification of nth OPI data in list: type (module type) High byte: Should be read more than a line so that the number of lines can be given here.					
-				UWord	rw
Multi-line: yes	6 + 5 * (n-1)		6 + 5 * (numData- 1)		

unit					
Variable specification of nth OPI data in list: unit					
-				UWord	rw
Multi-line: yes	3 + 5 * (n-1)		3 + 5 * (numData- 1)		

varSpecs					
Do not use this variable any more.					
-		0	maxnumTracePr otData	UWord	rw
Multi-line: yes	1		1		

3.10.4 Area C, Block ETP : Types of events

OEM-MMC: Linkitem /ChannelProtocolEvent/...

Description of logging event types.

It is permissible to access this module via several lines and columns.

The line index identifies a specific event.

Standard events: line index <= 10000:

OEM events: line index > 10000:

User index: is determined by the 1000s digit of the line index

Event type: is determined by the last three digits of the line index

Examples of the line index:

00001: Standard event of user 0 with the number 1 (IPO)

00006: Standard event of user 0 with the number 6 (NC start)

03006: Standard event of user 3 with the number 6 (NC start)

06006: Standard event of user 6 with the number 6 (NC start)

10001: OEM event of user 0 with the number 1

13002: OEM event of user 3 with the number 2

Standard event types:

Cyclic events:

1 = IPO and IPO cycle

15 = IPO2

47 = IPO3 (from SW $[[SW510400]])

48 = IPO4 (from SW $[[SW510400]])

Acyclic events related to axis motions:

2 = GEO_AXIS_START and Geo axis starts or changes the direction

18 = GEO_AXIS_STARTa see VDI interface NCK->PLC channel specific

DBB40 Bit6 and Bit7 (Bit6 = motion command+, Bit7 = motion command-)

Event occurs when a bit is reset.

3 = GEO_AXIS_STOP and Geo axis stops

19 = GEO_AXIS_STOPa, see VDI interface NCK->PLC channel specific

DBB40 Bit6 and Bit7 (Bit6 = motion command-, Bit7 = motion command+)

Event occurs when both bits are set to 0 and one of them was previously active.

4 = MA_AXIS_START, One machine axis of the channel starts or changes the direction

see VDI interface NCK->PLC axis-specific

DBB64 Bit6 and Bit7 (Bit6 = motion command-, Bit7 = motion command+)

Event occurs when a bit is reset.

5 = MA_AXIS_STOP, One machine axis stops

see VDI interface NCK->PLC axis-specific

DBB64 Bit6 and Bit7 (Bit6 = motion command-, Bit7 = motion command+)

Event occurs if both bits are set to 0 and one of them was previously active.

Acyclic events related to channel influence:

6 = NC_START NC start (if detected in NC)

7 = NC_STOP NC stop (if detected in NC, axes may still be traversed)

Acyclic events related to part program processing:

8 = BLOCK_BEG_1 Block start (first IPO cycle of a block) without intermediate blocks, all program levels

9 = BLOCK_BEG_2 and Block start (first IPO cycle of a block) with intermediate blocks, all program levels

20 = BLOCK_BEG_2a

10 = BLOCK_BEG_3 Block start (first IPO cycle of a block) without intermediate blocks, only main program level and MDA level

16 = BLOCK_BEG_S1 and Block start (search run with computation) with intermediate blocks, all program levels

22 = BLOCK_BEG_S1a

11 = BLOCK_END_1 Block end (first IPO cycle of a block) without intermediate blocks, all program levels

12 = BLOCK_END_2 and Block end (first IPO cycle of a block) with intermediate blocks, all program levels

21 = BLOCK_END_2a

13 = BLOCK_END_3 Block end (first IPO cycle of a block) without intermediate blocks, only main program level and MDA level

17 = BLOCK_END_S1 Block end (search run with computation) with intermediate blocks, all program levels

31 = BLOCK_END_P1 Block end (run in) (from SW $[[SW530000]])

32 = BLOCK_END_P1a Block end (run in) (from SW $[[SW530000]])

44 = BLOCK_END_I1 Block end (interpreter) (from SW $[[SW510300]])

43 = NC_LEVEL_CHG Level change during part program processing (from SW $[[SW510300]])

Acyclic events triggered by part programm command WRTPR

23 = PROT_TXT_REQ Logging a WRTPR text

24 = PROT_TXT_REQ_S1 Logging a WRTPR text (search run with computation)

33 = PROT_TXT_REQ_P1 Logging a WRTPR text (run in) (from SW $[[SW510300]])

Acyclic events triggered by the logging process itself

14 = PROT_FILE_BEG Start logging related to a log file.

29 = PROT_START_TRIG Start trigger has triggered (from SW $[[SW510300]])

30 = PROT_STOP_TRIG Stop trigger has triggered (from SW $[[SW510300]])

46 = PROT_START Start logging (from SW $[[SW510300]])

45 = PROT_STOP Stop logging (from SW $[[SW510300]])

Acyclic events triggered by buttons

42 = CANCEL_BUTTON The Cancel button was pressed (from SW $[[SW510300]])

Acyclic events triggered by alarms

41 = ALARM_REPORTED An alarm has occurred (from SW $[[SW510300]])

Acyclic events triggered by synchronized action

36 = SYNC_ACT_ACTIV Activating synchronized action (from SW $[[SW510300]])

37 = SYNC_ACT_DEACT Deactivating synchronized action (from SW $[[SW510300]])

38 = SYNC_ACT_FIRE Synchronized action triggers (from SW $[[SW510300]])

Acyclic events triggered by tool

25 = TOOL_CHANGE tool change (from SW $[[SW420000]])

27 = TOOL_CHANGE_S1 tool change (search run with computation) (from SW $[[SW440000]])

34 = TOOL_CHANGE_P1 tool change (run in) (from SW $[[SW510300]])

26 = CUTTEDGE_CHANGE cutting edge change (from SW $[[SW420000]])

28 = CUTTEDGE_CHANGE_S1 cutting edge change (search run with computation) (from SW $[[SW440000]])

35 = CUTTEDGE_CHANGE_P1 cutting edge change (run in) (from SW $[[SW510300]])

Acyclic events triggered by PLC

39 = PLC_OB_1 PLC OB1 started (from SW $[[SW510300]])

40 = PLC_OB40 PLC OB40 started (from SW $[[SW510300]])¶

asciiMode					
Data logging format 0: Data recorded in binary format with fixed alignment to 8 bytes 1: Data recorded in ASCII format 2: Data recorded in binary format with variable alignment 3: Data recorded in binary format with variable alignment and optimization of two consecutive data records of the same event. In this case, only the header is logged, not the actual data.					
-	0	0	3	UWord	rw
Multi-line: yes	Event (see module header)		siehe Bausteinkopf		

countActivated					
Number of times the event has occurred					
-	0			UWord	r
Multi-line: no					

countActivatedL					
Number of times the event has occurred.					
-	0			Long Integer	r
Multi-line: yes	Event (see module header)		siehe Bausteinkopf		

dataListIndex					
Index of data list to be used All valid columns in module ETPD - 1)					
-	0	0		UWord	rw
Multi-line: yes	Event (see module header)		siehe Bausteinkopf		

dataProtok					
Number of bytes entered in the Fifo file					
-	0			Long Integer	r
Multi-line: no					

dataUploaded					
Number of bytes already uploaded from the Fifo file					
-	0			Long Integer	r
Multi-line: no					

eventActive					
Event state 0: Not active 1: Active 2: Deactivate and release data set					
-	0	0	2	UWord	rw
Multi-line: yes	Event (see module header)		siehe Bausteinkopf		

eventActiveStatus					
For diagnosis: Event state 0: Activated 1: Not activated 2: Cannot be activated because the sum of the variable lengths is too large 3: Cannot be activated because the internal resources are not sufficient 4: Cannot be activated because the protocol file cannot be created 100-...- cannot be activated because the variable specification with the index (value - 100) is wrong					
-	0	0		UWord	r
Multi-line: yes	Event (see module header)		siehe Bausteinkopf		

headerType					

Type of header in the data record
0: No header
1: Short header with the following structure:
UDword dataStamp; // Data record identified by a consecutive number
UWord event; // Enter type of event that occurred
UWord protCount; // The number of times the event has been logged
2: Long header with the following structure:
UDword dataStamp; // Data record identified by a consecutive number
UWord event; // Enter type of event that occurred
UByte chan; // Channel in which the event occurred
UByte dummy1; // Still free
UDword protCount; // The number of times the event has been logged
UDword dummy2; // Still free
3: Mid-length header, non-aligned with the following structure:
UDword dataStamp; // Data record identified by a consecutive number
UWord event; // Enter type of event that occurred
UByte chan; // Channel in which the event occurred
UByte dummy1; // Still free
UDword protCount; // The number of times the event has been logged					

-	1	0	3	UWord	rw
Multi-line: yes	Event (see module header)		siehe Bausteinkopf		

maxElementsFastFifoUsed					
For diagnosis: Maximum number of entries in the FIFO buffer					
-	0	0		UWord	r
Multi-line: yes	Event (see module header)		siehe Bausteinkopf		

maxFileLength					
Maximum length of the log file. Values less than 1024 are interpreted as KB, larger values as bytes.					
-	0	0		UWord	rw
Multi-line: yes	Event (see module header)		siehe Bausteinkopf		

maxGrossFileLengthUsed					
For diagnosis: Maximum gross size of log file					
-	0	0		UWord	r
Multi-line: yes	Event (see module header)		siehe Bausteinkopf		

maxNetFileLengthTooSmall					
For diagnosis: Number of (net) bytes by which log file is undersized					
-	0	0		UWord	r
Multi-line: yes	Event (see module header)		siehe Bausteinkopf		

numElementsFastFifoTooSmall					
For diagnosis: Number of entries by which the Fifo buffer is undersized					
-	0	0		UWord	r
Multi-line: yes	Event (see module header)		siehe Bausteinkopf		

protocolFilename					
Name of the log file including the path					
-	0			String [64]	rw
Multi-line: yes	Event (see module header)		siehe Bausteinkopf		

resultPar1					
General result value, the significance is a function of the event. SYNC_ACT_ACTIVATE, SYNC_ACT_DEACTIVATE, and SYNC_ACT_FIRE: ID of the synchronous action. All non-stated events do not supply this result value.					
-	0			UWord	r
Multi-line: yes	Event (see module header)		siehe Bausteinkopf		

skip					
Number of events to be skipped					
-	0	0		UWord	rw
Multi-line: yes	Event (see module header)		siehe Bausteinkopf		

startTriggerLock					
Setting, whether the start trigger is not to be processed during this event. 0: Trigger is processed 1: Trigger is not processed					
-	0	0	1	UWord	rw
Multi-line: no					

stopTriggerLock					
Setting, whether the stop trigger is not to be processed during this event. 0: Trigger is processed 1: Trigger is not processed					
-	0	0	1	UWord	rw
Multi-line: no					

suppressProtLock					
Clears the effect of traceProtocolLock 0: The disable is active 1: The disable is canceled for this event					
-	0	0	1	UWord	rw
Multi-line: yes	1		1		

timePeriod					
Time base for cyclic event only					
ms	0	0		UWord	r
Multi-line: yes	Event (see module header)		siehe Bausteinkopf		

3.11 Status data of the HMI

3.11.1 Area M, Block S : Internal status data HMI

OEM-MMC: Linkitem /DriveState/...

Some internal status data of the HMI can be accessed via this module.

/Nck/Nck/ActApplication					
Current application for display in HMI					
-				String [32]	rw
Multi-line: no					

/Nck/Nck/ActBag					
Current operating mode for display in HMI					
-				Character	rw
Multi-line: no					

/Nck/Nck/Channel					
Current channel for display in HMI					
-				Character	rw
Multi-line: no					

/Nck/Nck/CoordSystem					
Coordinate system for display in HMI					
-				Character	rw
Multi-line: no					

3.12 User data

3.12.1 Area C, Block GD1 : GUD, channel-specific, area 1

OEM-MMC: Linkitem /Channel/...

Global user data, channel-specific, area 1.

The variables in this block are generated and deleted dynamically in the NCK.

The description and addressing of the existing variables are therefore not defined, and they have to be taken from the associated ACX file.

The variable is addressed via the column index (SymbolID).

The line index is only relevant for vectors and fields, it is calculated as follows:

Single data: 1

1-dim. fields: 1 + index1

2-dim. fields: 1 + index1 * maxdim2 +

index2

3-dim. fields: 1 + index1 * maxdim2 * maxdim3 +

index2 * maxdim3 +

index3

The formula for 3-dim. fields can be applied generally if, in the case of missing dimensions, the index is replaced by 0, and maxdim by 1.

Values ranges:

index1: 0 to (maxdim1-1)

index2: 0 to (maxdim2-1)

index3: 0 to (maxdim3-1)

Depending on the dimensioning of the fields, that is the size of maxdim1, maxdim2 and maxdim3, the value range of the line index (16 bit) may not be sufficient for addressing.

OPI access is not possible if this is the case.

3.12.2 Area C, Block GD2 : GUD, channel-specific, area 2

OEM-MMC: Linkitem /Channel/...

Global user data, channel-specific, area 2.

The variables in this block are generated and deleted dynamically in the NCK.

The description and addressing of the existing variables are therefore not defined, and they have to be taken from the associated ACX file.

The variable is addressed via the column index (SymbolID).

The line index is only relevant for vectors and fields, it is calculated as follows:

Single data: 1

1-dim. fields: 1 + index1

2-dim. fields: 1 + index1 * maxdim2 +

index2

3-dim. fields: 1 + index1 * maxdim2 * maxdim3 +

index2 * maxdim3 +

index3

The formula for 3-dim. fields can be applied generally if, in the case of missing dimensions, the index is replaced by 0, and maxdim by 1.

Values ranges:

index1: 0 to (maxdim1-1)

index2: 0 to (maxdim2-1)

index3: 0 to (maxdim3-1)

Depending on the dimensioning of the fields, that is the size of maxdim1, maxdim2 and maxdim3, the value range of the line index (16 bit) may not be sufficient for addressing.

OPI access is not possible if this is the case.

3.12.3 Area C, Block GD3 : GUD, channel-specific, area 3

OEM-MMC: Linkitem /Channel/...

Global user data, channel-specific, area 3.

The variables in this block are generated and deleted dynamically in the NCK.

The description and addressing of the existing variables are therefore not defined, and they have to be taken from the associated ACX file.

The variable is addressed via the column index (SymbolID).

The line index is only relevant for vectors and fields, it is calculated as follows:

Single data: 1

1-dim. fields: 1 + index1

2-dim. fields: 1 + index1 * maxdim2 +

index2

3-dim. fields: 1 + index1 * maxdim2 * maxdim3 +

index2 * maxdim3 +

index3

The formula for 3-dim. fields can be applied generally if, in the case of missing dimensions, the index is replaced by 0, and maxdim by 1.

Values ranges:

index1: 0 to (maxdim1-1)

index2: 0 to (maxdim2-1)

index3: 0 to (maxdim3-1)

Depending on the dimensioning of the fields, that is the size of maxdim1, maxdim2 and maxdim3, the value range of the line index (16 bit) may not be sufficient for addressing.

OPI access is not possible if this is the case.

3.12.4 Area C, Block GD4 : GUD, channel-specific, area 4

OEM-MMC: Linkitem /Channel/...

Global user data, channel-specific, area 4.

The variables in this block are generated and deleted dynamically in the NCK.

The description and addressing of the existing variables are therefore not defined, and they have to be taken from the associated ACX file.

The variable is addressed via the column index (SymbolID).

The line index is only relevant for vectors and fields, it is calculated as follows:

Single data: 1

1-dim. fields: 1 + index1

2-dim. fields: 1 + index1 * maxdim2 +

index2

3-dim. fields: 1 + index1 * maxdim2 * maxdim3 +

index2 * maxdim3 +

index3

The formula for 3-dim. fields can be applied generally if, in the case of missing dimensions, the index is replaced by 0, and maxdim by 1.

Values ranges:

index1: 0 to (maxdim1-1)

index2: 0 to (maxdim2-1)

index3: 0 to (maxdim3-1)

Depending on the dimensioning of the fields, that is the size of maxdim1, maxdim2 and maxdim3, the value range of the line index (16 bit) may not be sufficient for addressing.

OPI access is not possible if this is the case.

3.12.5 Area C, Block GD5 : GUD, channel-specific, area 5

OEM-MMC: Linkitem /Channel/...

Global user data, channel-specific, area 5.

The variables in this block are generated and deleted dynamically in the NCK.

The description and addressing of the existing variables are therefore not defined, and they have to be taken from the associated ACX file.

The variable is addressed via the column index (SymbolID).

The line index is only relevant for vectors and fields, it is calculated as follows:

Single data: 1

1-dim. fields: 1 + index1

2-dim. fields: 1 + index1 * maxdim2 +

index2

3-dim. fields: 1 + index1 * maxdim2 * maxdim3 +

index2 * maxdim3 +

index3

The formula for 3-dim. fields can be applied generally if, in the case of missing dimensions, the index is replaced by 0, and maxdim by 1.

Values ranges:

index1: 0 to (maxdim1-1)

index2: 0 to (maxdim2-1)

index3: 0 to (maxdim3-1)

Depending on the dimensioning of the fields, that is the size of maxdim1, maxdim2 and maxdim3, the value range of the line index (16 bit) may not be sufficient for addressing.

OPI access is not possible if this is the case.

3.12.6 Area C, Block GD6 : GUD, channel-specific, area 6

OEM-MMC: Linkitem /Channel/...

Global user data, channel-specific, area 6.

The variables in this block are generated and deleted dynamically in the NCK.

The description and addressing of the existing variables are therefore not defined, and they have to be taken from the associated ACX file.

The variable is addressed via the column index (SymbolID).

The line index is only relevant for vectors and fields, it is calculated as follows:

Single data: 1

1-dim. fields: 1 + index1

2-dim. fields: 1 + index1 * maxdim2 +

index2

3-dim. fields: 1 + index1 * maxdim2 * maxdim3 +

index2 * maxdim3 +

index3

The formula for 3-dim. fields can be applied generally if, in the case of missing dimensions, the index is replaced by 0, and maxdim by 1.

Values ranges:

index1: 0 to (maxdim1-1)

index2: 0 to (maxdim2-1)

index3: 0 to (maxdim3-1)

Depending on the dimensioning of the fields, that is the size of maxdim1, maxdim2 and maxdim3, the value range of the line index (16 bit) may not be sufficient for addressing.

OPI access is not possible if this is the case.

3.12.7 Area C, Block GD7 : GUD, channel-specific, area 7

OEM-MMC: Linkitem /Channel/...

Global user data, channel-specific, area 7.

The variables in this block are generated and deleted dynamically in the NCK.

The description and addressing of the existing variables are therefore not defined, and they have to be taken from the associated ACX file.

The variable is addressed via the column index (SymbolID).

The line index is only relevant for vectors and fields, it is calculated as follows:

Single data: 1

1-dim. fields: 1 + index1

2-dim. fields: 1 + index1 * maxdim2 +

index2

3-dim. fields: 1 + index1 * maxdim2 * maxdim3 +

index2 * maxdim3 +

index3

The formula for 3-dim. fields can be applied generally if, in the case of missing dimensions, the index is replaced by 0, and maxdim by 1.

Values ranges:

index1: 0 to (maxdim1-1)

index2: 0 to (maxdim2-1)

index3: 0 to (maxdim3-1)

Depending on the dimensioning of the fields, that is the size of maxdim1, maxdim2 and maxdim3, the value range of the line index (16 bit) may not be sufficient for addressing.

OPI access is not possible if this is the case.

3.12.8 Area C, Block GD8 : GUD, channel-specific, area 8

OEM-MMC: Linkitem /Channel/...

Global user data, channel-specific, area 8.

The variables in this block are generated and deleted dynamically in the NCK.

The description and addressing of the existing variables are therefore not defined, and they have to be taken from the associated ACX file.

The variable is addressed via the column index (SymbolID).

The line index is only relevant for vectors and fields, it is calculated as follows:

Single data: 1

1-dim. fields: 1 + index1

2-dim. fields: 1 + index1 * maxdim2 +

index2

3-dim. fields: 1 + index1 * maxdim2 * maxdim3 +

index2 * maxdim3 +

index3

The formula for 3-dim. fields can be applied generally if, in the case of missing dimensions, the index is replaced by 0, and maxdim by 1.

Values ranges:

index1: 0 to (maxdim1-1)

index2: 0 to (maxdim2-1)

index3: 0 to (maxdim3-1)

Depending on the dimensioning of the fields, that is the size of maxdim1, maxdim2 and maxdim3, the value range of the line index (16 bit) may not be sufficient for addressing.

OPI access is not possible if this is the case.

3.12.9 Area C, Block GD9 : GUD, channel-specific, area 9

OEM-MMC: Linkitem /Channel/...

Global user data, channel-specific, area 9.

The variables in this block are generated and deleted dynamically in the NCK.

The description and addressing of the existing variables are therefore not defined, and they have to be taken from the associated ACX file.

The variable is addressed via the column index (SymbolID).

The line index is only relevant for vectors and fields, it is calculated as follows:

Single data: 1

1-dim. fields: 1 + index1

2-dim. fields: 1 + index1 * maxdim2 +

index2

3-dim. fields: 1 + index1 * maxdim2 * maxdim3 +

index2 * maxdim3 +

index3

The formula for 3-dim. fields can be applied generally if, in the case of missing dimensions, the index is replaced by 0, and maxdim by 1.

Values ranges:

index1: 0 to (maxdim1-1)

index2: 0 to (maxdim2-1)

index3: 0 to (maxdim3-1)

Depending on the dimensioning of the fields, that is the size of maxdim1, maxdim2 and maxdim3, the value range of the line index (16 bit) may not be sufficient for addressing.

OPI access is not possible if this is the case.

3.12.10 Area C, Block GUD : GUD, channel-specific, area 0

OEM-MMC: Linkitem /Channel/...

Global user data, channel-specific, area 0.

The variables in this block are generated and deleted dynamically in the NCK.

The description and addressing of the existing variables are therefore not defined, and they have to be taken from the associated ACX file.

The variable is addressed via the column index (SymbolID).

The line index is only relevant for vectors and fields, it is calculated as follows:

Single data: 1

1-dim. fields: 1 + index1

2-dim. fields: 1 + index1 * maxdim2 +

index2

3-dim. fields: 1 + index1 * maxdim2 * maxdim3 +

index2 * maxdim3 +

index3

The formula for 3-dim. fields can be applied generally if, in the case of missing dimensions, the index is replaced by 0, and maxdim by 1.

Values ranges:

index1: 0 to (maxdim1-1)

index2: 0 to (maxdim2-1)

index3: 0 to (maxdim3-1)

Depending on the dimensioning of the fields, that is the size of maxdim1, maxdim2 and maxdim3, the value range of the line index (16 bit) may not be sufficient for addressing.

OPI access is not possible if this is the case.

3.12.11 Area C, Block LUD : LUD, channel-specific

OEM-MMC: Linkitem /Channel/...

Local user data, channel-specific.

The variables in this block are dynamically generated and deleted in the NCK.

The description and addressing of the existing variables are therefore not specified, and must be taken from the associated ACX file.

The variable is addressed via the column index (SymbolID).

The line index is only relevant for vectors and fields, it is calculated as follows:

Single data: 1

1-dim. fields: 1 + index1

2-dim. fields: 1 + index1 * maxdim2 +

index2

3-dim. fields: 1 + index1 * maxdim2 * maxdim3 +

index2 * maxdim3 +

index3

The formula for 3-dim. fields can generally be applied if the missing dimension index is replaced by 0, and maxdim by 1.

Value ranges:

index1: 0 to (maxdim1-1)

index2: 0 to (maxdim2-1)

index3: 0 to (maxdim3-1)

Depending on the dimensioning of the fields, that is the size of maxdim1, maxdim2 and maxdim3, the value range of the line index (16 bits) may be inadequate for addressing.

OPI access is not possible if this is the case.

DUMMY					
Undefined					
-				Character	r
Multi-line: no			2		

3.12.12 Area N, Block GD1 : GUD, NCK-specific, area 1

OEM-MMC: Linkitem /Nck/...

Global user data, NCK-specific, area 1.

The variables in this block are generated and deleted dynamically in the NCK.

The description and addressing of the existing variables are therefore not defined, and they have to be taken from the associated ACX file.

The variable is addressed via the column index (SymbolID).

The line index is only relevant for vectors and fields, it is calculated as follows:

Single data: 1

1-dim. fields: 1 + index1

2-dim. fields: 1 + index1 * maxdim2 +

index2

3-dim. fields: 1 + index1 * maxdim2 * maxdim3 +

index2 * maxdim3 +

index3

The formula for 3-dim. fields can be applied generally if, in the case of missing dimensions, the index is replaced by 0, and maxdim by 1.

Values ranges:

index1: 0 to (maxdim1-1)

index2: 0 to (maxdim2-1)

index3: 0 to (maxdim3-1)

Depending on the dimensioning of the fields, that is the size of maxdim1, maxdim2 and maxdim3, the value range of the line index (16 bit) may not be sufficient for addressing.

OPI access is not possible if this is the case.

3.12.13 Area N, Block GD2 : GUD, NCK-specific, area 2

OEM-MMC: Linkitem /Nck/...

Global user data, NCK-specific, area 2.

The variables in this block are generated and deleted dynamically in the NCK.

The description and addressing of the existing variables are therefore not defined, and they have to be taken from the associated ACX file.

The variable is addressed via the column index (SymbolID).

The line index is only relevant for vectors and fields, it is calculated as follows:

Single data: 1

1-dim. fields: 1 + index1

2-dim. fields: 1 + index1 * maxdim2 +

index2

3-dim. fields: 1 + index1 * maxdim2 * maxdim3 +

index2 * maxdim3 +

index3

The formula for 3-dim. fields can be applied generally if, in the case of missing dimensions, the index is replaced by 0, and maxdim by 1.

Values ranges:

index1: 0 to (maxdim1-1)

index2: 0 to (maxdim2-1)

index3: 0 to (maxdim3-1)

Depending on the dimensioning of the fields, that is the size of maxdim1, maxdim2 and maxdim3, the value range of the line index (16 bit) may not be sufficient for addressing.

OPI access is not possible if this is the case.

3.12.14 Area N, Block GD3 : GUD, NCK-specific, area 3

OEM-MMC: Linkitem /Nck/...

Global user data, NCK-specific, area 3.

The variables in this block are generated and deleted dynamically in the NCK.

The description and addressing of the existing variables are therefore not defined, and they have to be taken from the associated ACX file.

The variable is addressed via the column index (SymbolID).

The line index is only relevant for vectors and fields, it is calculated as follows:

Single data: 1

1-dim. fields: 1 + index1

2-dim. fields: 1 + index1 * maxdim2 +

index2

3-dim. fields: 1 + index1 * maxdim2 * maxdim3 +

index2 * maxdim3 +

index3

The formula for 3-dim. fields can be applied generally if, in the case of missing dimensions, the index is replaced by 0, and maxdim by 1.

Values ranges:

index1: 0 to (maxdim1-1)

index2: 0 to (maxdim2-1)

index3: 0 to (maxdim3-1)

Depending on the dimensioning of the fields, that is the size of maxdim1, maxdim2 and maxdim3, the value range of the line index (16 bit) may not be sufficient for addressing.

OPI access is not possible if this is the case.

3.12.15 Area N, Block GD4 : GUD, NCK-specific, area 4

OEM-MMC: Linkitem /Nck/...

Global user data, NCK-specific, area 4.

The variables in this block are generated and deleted dynamically in the NCK.

The description and addressing of the existing variables are therefore not defined, and they have to be taken from the associated ACX file.

The variable is addressed via the column index (SymbolID).

The line index is only relevant for vectors and fields, it is calculated as follows:

Single data: 1

1-dim. fields: 1 + index1

2-dim. fields: 1 + index1 * maxdim2 +

index2

3-dim. fields: 1 + index1 * maxdim2 * maxdim3 +

index2 * maxdim3 +

index3

The formula for 3-dim. fields can be applied generally if, in the case of missing dimensions, the index is replaced by 0, and maxdim by 1.

Values ranges:

index1: 0 to (maxdim1-1)

index2: 0 to (maxdim2-1)

index3: 0 to (maxdim3-1)

Depending on the dimensioning of the fields, that is the size of maxdim1, maxdim2 and maxdim3, the value range of the line index (16 bit) may not be sufficient for addressing.

OPI access is not possible if this is the case.

3.12.16 Area N, Block GD5 : GUD, NCK-specific, area 5

OEM-MMC: Linkitem /Nck/...

Global user data, NCK-specific, area 5.

The variables in this block are generated and deleted dynamically in the NCK.

The description and addressing of the existing variables are therefore not defined, and they have to be taken from the associated ACX file.

The variable is addressed via the column index (SymbolID).

The line index is only relevant for vectors and fields, it is calculated as follows:

Single data: 1

1-dim. fields: 1 + index1

2-dim. fields: 1 + index1 * maxdim2 +

index2

3-dim. fields: 1 + index1 * maxdim2 * maxdim3 +

index2 * maxdim3 +

index3

The formula for 3-dim. fields can be applied generally if, in the case of missing dimensions, the index is replaced by 0, and maxdim by 1.

Values ranges:

index1: 0 to (maxdim1-1)

index2: 0 to (maxdim2-1)

index3: 0 to (maxdim3-1)

Depending on the dimensioning of the fields, that is the size of maxdim1, maxdim2 and maxdim3, the value range of the line index (16 bit) may not be sufficient for addressing.

OPI access is not possible if this is the case.

3.12.17 Area N, Block GD6 : GUD, NCK-specific, area 6

OEM-MMC: Linkitem /Nck/...

Global user data, NCK-specific, area 6.

The variables in this block are generated and deleted dynamically in the NCK.

The description and addressing of the existing variables are therefore not defined, and they have to be taken from the associated ACX file.

The variable is addressed via the column index (SymbolID).

The line index is only relevant for vectors and fields, it is calculated as follows:

Single data: 1

1-dim. fields: 1 + index1

2-dim. fields: 1 + index1 * maxdim2 +

index2

3-dim. fields: 1 + index1 * maxdim2 * maxdim3 +

index2 * maxdim3 +

index3

The formula for 3-dim. fields can be applied generally if, in the case of missing dimensions, the index is replaced by 0, and maxdim by 1.

Values ranges:

index1: 0 to (maxdim1-1)

index2: 0 to (maxdim2-1)

index3: 0 to (maxdim3-1)

Depending on the dimensioning of the fields, that is the size of maxdim1, maxdim2 and maxdim3, the value range of the line index (16 bit) may not be sufficient for addressing.

OPI access is not possible if this is the case.

3.12.18 Area N, Block GD7 : GUD, NCK-specific, area 7

OEM-MMC: Linkitem /Nck/...

Global user data, NCK-specific, area 7.

The variables in this block are generated and deleted dynamically in the NCK.

The description and addressing of the existing variables are therefore not defined, and they have to be taken from the associated ACX file.

The variable is addressed via the column index (SymbolID).

The line index is only relevant for vectors and fields, it is calculated as follows:

Single data: 1

1-dim. fields: 1 + index1

2-dim. fields: 1 + index1 * maxdim2 +

index2

3-dim. fields: 1 + index1 * maxdim2 * maxdim3 +

index2 * maxdim3 +

index3

The formula for 3-dim. fields can be applied generally if, in the case of missing dimensions, the index is replaced by 0, and maxdim by 1.

Values ranges:

index1: 0 to (maxdim1-1)

index2: 0 to (maxdim2-1)

index3: 0 to (maxdim3-1)

Depending on the dimensioning of the fields, that is the size of maxdim1, maxdim2 and maxdim3, the value range of the line index (16 bit) may not be sufficient for addressing.

OPI access is not possible if this is the case.

3.12.19 Area N, Block GD8 : GUD, NCK-specific, area 8

OEM-MMC: Linkitem /Nck/...

Global user data, NCK-specific, area 8.

The variables in this block are generated and deleted dynamically in the NCK.

The description and addressing of the existing variables are therefore not defined, and they have to be taken from the associated ACX file.

The variable is addressed via the column index (SymbolID).

The line index is only relevant for vectors and fields, it is calculated as follows:

Single data: 1

1-dim. fields: 1 + index1

2-dim. fields: 1 + index1 * maxdim2 +

index2

3-dim. fields: 1 + index1 * maxdim2 * maxdim3 +

index2 * maxdim3 +

index3

The formula for 3-dim. fields can be applied generally if, in the case of missing dimensions, the index is replaced by 0, and maxdim by 1.

Values ranges:

index1: 0 to (maxdim1-1)

index2: 0 to (maxdim2-1)

index3: 0 to (maxdim3-1)

Depending on the dimensioning of the fields, that is the size of maxdim1, maxdim2 and maxdim3, the value range of the line index (16 bit) may not be sufficient for addressing.

OPI access is not possible if this is the case.

3.12.20 Area N, Block GD9 : GUD, NCK-specific, area 9

OEM-MMC: Linkitem /Nck/...

Global user data, NCK-specific, area 9.

The variables in this block are generated and deleted dynamically in the NCK.

The description and addressing of the existing variables are therefore not defined, and they have to be taken from the associated ACX file.

The variable is addressed via the column index (SymbolID).

The line index is only relevant for vectors and fields, it is calculated as follows:

Single data: 1

1-dim. fields: 1 + index1

2-dim. fields: 1 + index1 * maxdim2 +

index2

3-dim. fields: 1 + index1 * maxdim2 * maxdim3 +

index2 * maxdim3 +

index3

The formula for 3-dim. fields can be applied generally if, in the case of missing dimensions, the index is replaced by 0, and maxdim by 1.

Values ranges:

index1: 0 to (maxdim1-1)

index2: 0 to (maxdim2-1)

index3: 0 to (maxdim3-1)

Depending on the dimensioning of the fields, that is the size of maxdim1, maxdim2 and maxdim3, the value range of the line index (16 bit) may not be sufficient for addressing.

OPI access is not possible if this is the case.

3.12.21 Area N, Block GUD : GUD, NCK-specific, area 0

OEM-MMC: Linkitem /Nck/...

Global user data, NCK-specific, area 0.

The variables in this block are generated and deleted dynamically in the NCK.

The description and addressing of the existing variables are therefore not defined, and they have to be taken from the associated ACX file.

The variable is addressed via the column index (SymbolID).

The line index is only relevant for vectors and fields, it is calculated as follows:

Single data: 1

1-dim. fields: 1 + index1

2-dim. fields: 1 + index1 * maxdim2 +

index2

3-dim. fields: 1 + index1 * maxdim2 * maxdim3 +

index2 * maxdim3 +

index3

The formula for 3-dim. fields can be applied generally if, in the case of missing dimensions, the index is replaced by 0, and maxdim by 1.

Values ranges:

index1: 0 to (maxdim1-1)

index2: 0 to (maxdim2-1)

index3: 0 to (maxdim3-1)

Depending on the dimensioning of the fields, that is the size of maxdim1, maxdim2 and maxdim3, the value range of the line index (16 bit) may not be sufficient for addressing.

OPI access is not possible if this is the case.

3.13 Generic coupling

3.13.1 Area N, Block CP : Generic coupling

OEM-MMC: Linkitem /NckGenericCoupling/...

The CP block contains the status data for the generic coupling.

The status of the axis couplings is structured in an NCK-specific and channel-specific area.

cpCtabExists					
Not zero, if the specified curve table exists					
-	0	0	1	UWord	r
Multi-line: yes	ID of the curve table				

cpCtabId					
ID no. of the nth curve table in the specified memory type					
-				Long Integer	r
Multi-line: yes	(n * 10) + memory type				

cpCtabIdNumLinSegDef					
Number of the linear segments defined for the specified curve table					
-	0			UWord	r
Multi-line: yes	ID of the curve table				

cpCtabIdNumPolDef					
Number of the polynomials defined for the specified curve table					
-	0			UWord	r
Multi-line: yes	ID of the curve table				

cpCtabIdNumPolySegDef					
Number of the polynomial segments defined for the specified curve table					
-	0			UWord	r
Multi-line: yes	ID of the curve table				

cpCtabIdNumSegDef					
Number of segments defined for the specified curve table					
-	0			UWord	r
Multi-line: yes	ID of the curve table				

cpCtabLocked					
Locking status, value > 0, if curve table is locked					
-		-1	3	Long Integer	r
Multi-line: yes	ID of the curve table				

cpCtabMemType					
Memory type in which the curve table is stored					
-		-1	2	Long Integer	r
Multi-line: yes	ID of the curve table				

cpCtabNumDef					
Total number of curve tables defined for the specified memory type					
-	0	0		UWord	r
Multi-line: yes	1=DRAM, 2=SRAM, 3=all memory types		3		

cpCtabNumFree					
Number of additional curve tables which can be defined in the specified memory type					
-	0	0		UWord	r
Multi-line: yes	1=DRAM, 2=SRAM, 3=all memory types		3		

cpCtabNumPolDef					
Total number of curve table polynomials defined in the specified memory type					
-	0			UWord	r
Multi-line: yes	1=DRAM, 2=SRAM, 3=all memory types		3		

cpCtabNumPolFree					
Number of additional curve table polynomials which can be defined in the specified memory type					
-	0			UWord	r
Multi-line: yes	1=DRAM, 2=SRAM, 3=all memory types		3		

cpCtabNumPolMax					
Maximum number of curve table polynomials permissible in the specified memory type					
-	0			UWord	r
Multi-line: yes	1=DRAM, 2=SRAM, 3=all memory types		3		

cpCtabNumSegDef					
Total number of curve table segments of the specified segment type defined in the specified memory type					
-	0			UWord	r
Multi-line: yes	(Segment type * 10) + memory type		23		

cpCtabNumSegFree					
Number of additional curve table segments of the specified segment type which can be defined in the specified memory type					
-	0			UWord	r
Multi-line: yes	(Segment type * 10) + memory type		23		

cpCtabNumSegMax					
Maximum number of curve table segments of the specified segment type permissible in the specified memory type					
-	0			UWord	r
Multi-line: yes	(Segment type * 10) + memory type		23		

cpCtabPeriodic					
Periodicity, value > 0, if curve table is periodic					
-		-1	2	Long Integer	r
Multi-line: yes	ID of the curve table				

3.13.2 Area C, Block CP : Generic coupling

OEM-MMC: Linkitem /ChannelGenericCoupling/...

This block contains the data of the generic coupling.

aaCpActFa	$AA_CPACTFA[ax,n]				
The axis index of the following axis of the nth coupling, in which the specified axis LAx is active as leading axis -1 = the following axis found is unknown in the channel or n == 0 or n > aaCpNumActFa (= number of active couplings of the axis as leading axis)					
-	-1	-1		UWord	r
Multi-line: yes	(low byte: axis index of leading axis (>= 1) high byte: serial number of following axis n (>= 1)				

aaCpActLa	$AA_CPACTLA[ax,n]				
The axis index of the nth leading axis which is active for the specified following axis -1 = the specified coupling is not active or n == 0 or n > aaCpNumActLa (= number of active leading axes of the following axis)					
-	-1	-1		UWord	r
Multi-line: yes	low byte: axis index of following axis (>= 1) high byte: serial number of leading axis n (>= 1)				

aaCpBlockChg	$AA_CPBC[a]				
The block change criterion indicates the condition that has to be fulfilled before one can continue with the next block of the NC program if a coupling has been activated for the stated following axis, FAx NONE - Block change is performed immediately FINE - Block change is performed with "Synchronism fine" COARSE - Block change is performed with "Synchronism coarse" IPOSTOP - Block change is performed with "Setpoint synchronism"					
-				String [32]	r
Multi-line: yes	Axis index of the following axis		numMachAxes		

aaCpDefLa	$AA_CPDEFLA[ax,n]				
The axis index of the nth leading axis which has been defined for the specified following axis -1 = the specified coupling is not defined or n == 0 or n > aaCpNumDefLa (= number of defined leading axes of the following axis)					
-	-1	-1		UWord	r
Multi-line: yes	low byte: axis index of following axis (>= 1) high byte: serial number of leading axis n (>= 1)				

aaCpMAlarm	$AA_CPMALARM[a]				
Behavior of coupling module regarding suppression of alarms					
-	0	0		UWord	r
Multi-line: yes	Axis index of the following axis		numMachAxes		

aaCpMReset	$AA_CPMRESET[a]				
Coupling mode through RESET NONE ON OFF DEL					
-				String [32]	r
Multi-line: yes	Axis index of the following axis		numMachAxes		

aaCpMStart	$AA_CPMSTART[a]				
Coupling mode through program start NONE ON OFF DEL					
-				String [32]	r
Multi-line: yes	Axis index of the following axis		numMachAxes		

aaCpMStartPrt	$AA_CPMSTARTPRT[a]				
Coupling mode through SERUPRO start NONE ON OFF DEL					
-				String [32]	r
Multi-line: yes	Axis index of the following axis		numMachAxes		

aaCpMVdi	$AA_CPMVDI[a]				
Behavior of the coupling module regarding VDI signals					
-	0	0		UWord	r
Multi-line: yes	Axis index of the following axis		numMachAxes		

aaCpNumActFa	$AA_CPNACTFA[ax]				
Number of couplings (following axes), in which the specified axis LAx is active as leading axis					
-	0	0		UWord	r
Multi-line: yes	Axis index of the leading axis		numMachAxes		

aaCpNumActLa	$AA_CPNACTLA[a]				
The number of leading axes which are active for the specified following axis					
-	0	0		UWord	r
Multi-line: yes	Axis index of the following axis		numMachAxes		

aaCpNumDefLa	$AA_CPNDEFLA[a]				
The number of leading axes which have been defined for the specified following axis					
-	0	0		UWord	r
Multi-line: yes	Axis index of the following axis		numMachAxes		

aaCpSetType	$AA_CPSETTYPE[a]				
Coupling set coupling type NONE TRAIL LEAD EG COUP					
-				String [32]	r
Multi-line: yes	Axis index of the following axis		numMachAxes		

aaCpSynCoPos	$AA_CPSYNCOP[a]				
Coarse positioning tolerance for coupling synchronization					
-				Double	r
Multi-line: yes	Axis index of the following axis		numMachAxes		

aaCpSynCoPos2	$AA_CPSYNCOP2[a]				
Second synchronism monitoring of the following axis / spindle: threshold value coarse					
-				Double	r
Multi-line: yes	Axis index of the following axis		numMachAxes		

aaCpSynCoVel	$AA_CPSYNCOV[a]				
Coarse velocity tolerance for coupling synchronization					
-				Double	r
Multi-line: yes	Axis index of the following axis		numMachAxes		

aaCpSynFiPos	$AA_CPSYNFIP[a]				
Fine positioning tolerance for coupling synchronization					
-				Double	r
Multi-line: yes	Axis index of the following axis		numMachAxes		

aaCpSynFiPos2	$AA_CPSYNFIP2[a]				
Second synchronism monitoring of the following axis / spindle: threshold value fine					
-				Double	r
Multi-line: yes	Axis index of the following axis		numMachAxes		

aaCpSynFiVel	$AA_CPSYNFIV[a]				
Fine velocity tolerance for coupling synchronization					
-				Double	r
Multi-line: yes	Axis index of the following axis		numMachAxes		

aaCpfAccelTotal	$AA_CPFACCT[a]				
Proportion of axis acceleration due to the coupled axes. The sum of the dependent proportion of the acceleration of all leading axes for the stated following axis, FAx.					
-				Double	r
Multi-line: yes	Axis index of the following axis		numMachAxes		

aaCpfActive	$AA_CPFACT[a]				
Bit-coded for identifying all types of coupling which are active for the stated following axis, FAx 0 = NONE - No active coupling to the following axis Bit 0 (0x0001) - TRAIL - Uses a coupling factor Bit 1 (0x0002) - LEAD - Uses a curve table Bit 2 (0x0004) - ELG - An electronic gearbox link Bit 3 (0x0008) - Reserved Bit 4 (0x0010) - COUP - Spindle/partial spindle coupling Bit 5 (0x0020) - GANTRY - Coupling of the split axes (axes mechanically bound) Bit 6 (0x0040) - TANG - Tangential coupling using a curve table Bit 7 (0x0080) - GEN_CP - Generic coupling					
-				UWord	r
Multi-line: yes	Axis index of the following axis		numMachAxes		

aaCpfCmdPosTotal	$AA_CPFCMDPT[a]				
Proportion of the axis position command due to the coupled axes. The sum of the dependent proportion of the position command for all leading axes for the stated following axis, FAx.					
-				Double	r
Multi-line: yes	Axis index of the following axis		numMachAxes		

aaCpfCmdVelTotal	$AA_CPFCMDVT[a]				
Proportion of the axis position command due to the coupled axes. The sum of the dependent proportion of the velocity command for all leading axes for the specified following axis.					
-				Double	r
Multi-line: yes	Axis index of the following axis		numMachAxes		

aaCpfMSOn	$AA_CPFMSON[a]				
Indicates the activation strategy of the following axis CNONE CFAST COARSE NTG ACN ACP DCT NTGP DCP					
-				String [32]	r
Multi-line: yes	Axis index of the following axis		numMachAxes		

aaCpfModeOff	$AA_CPFMOF[a]				
Identifies the behavior of the following axis if the coupling is deactivated STOP - Stop following axis/spindle CON - Continue movement with the current velocity ADD					
-				String [32]	r
Multi-line: yes	Axis index of the following axis		numMachAxes		

aaCpfModeOn	$AA_CPFMON[a]				
Indicates the behavior of the following axis, FAx, when the coupling is activated. STOP - Stop following axis/spindle CON - Continue motion of the FAx with the current velocity ADD					
-				String [32]	r
Multi-line: yes	Axis index of the following axis		numMachAxes		

aaCpfRS	$AA_CPFRS[a]				
The reference system indicates the point at which the coupling process is applied BCS - Basic coordinate system MCS - Machine coordinate system					
-				String [32]	r
Multi-line: yes	Axis index of the following axis		numMachAxes		

aaCpfReqVelocity	$AA_CPFREQV[a]				
Returns the speed requested by the active following axes/spindles.					
-				Double	r
Multi-line: yes	Axis index of the following axis		numMachAxes		

aaCplAccel	$AA_CPLACC[a,b]				
Acceleration proportion of the following axis caused by an active coupling to the specified leading axis					
-				Double	r
Multi-line: yes	(Axis index of the slave axis) * numMachAxes + (axis index of the master axis)		numMachAxes * numMachAxes		

aaCplCTabId	$AA_CPLCTID[a,b]				
ID number of the curve table which is used with the coupling of the stated axes.					
-				UWord	r
Multi-line: yes	(Axis index of the slave axis) * numMachAxes + (axis index of the master axis)		numMachAxes * numMachAxes		

aaCplCmdPos	$AA_CPLCMDP[a,b]				
The proportion of the axis position command assigned to the stated coupling.					
-				Double	r
Multi-line: yes	(Axis index of the slave axis) * numMachAxes + (axis index of the master axis)		numMachAxes * numMachAxes		

aaCplCmdVel	$AA_CPLCMDV[a,b]				
The proportion of the axis acceration command assigned to the stated coupling.					
-				Double	r
Multi-line: yes	(Axis index of the slave axis) * numMachAxes + (axis index of the master axis)		numMachAxes * numMachAxes		

aaCplDenominator	$AA_CPLDEN[a,b]				
Denominator of coupling factor					
-				Double	r
Multi-line: yes	(Axis index of the slave axis) * numMachAxes + (axis index of the master axis)		numMachAxes * numMachAxes		

aaCplInScale	$AA_CPLINSC[a,b]				
Input scaling factor of coupling factor					
-				Double	r
Multi-line: yes	(Axis index of the slave axis) * numMachAxes + (axis index of the master axis)		numMachAxes * numMachAxes		

aaCplInTrans	$AA_CPLINTR[a,b]				
Input transmission correction of coupling factor					
-				Double	r
Multi-line: yes	(Axis index of the slave axis) * numMachAxes + (axis index of the master axis)		numMachAxes * numMachAxes		

aaCplNumerator	$AA_CPLNUM[a,b]				
Counter of coupling factor					
-				Double	r
Multi-line: yes	(Axis index of the slave axis) * numMachAxes + (axis index of the master axis)		numMachAxes * numMachAxes		

aaCplOutScale	$AA_CPLOUTSC[a,b]				
Output scaling factor of coupling factor					
-				Double	r
Multi-line: yes	(Axis index of the slave axis) * numMachAxes + (axis index of the master axis)		numMachAxes * numMachAxes		

aaCplOutTrans	$AA_CPLOUTTR[a,b]				
Output transmission correction of coupling factor					
-				Double	r
Multi-line: yes	(Axis index of the slave axis) * numMachAxes + (axis index of the master axis)		numMachAxes * numMachAxes		

aaCplRS	$AA_CPLRS[a,b]				
Reference system for the specified coupling Reference system for the specified coupling DescriptionValue range: BCS - Basic coordinate system MCS - Machine coordinate system					
-				String [32]	r
Multi-line: yes	(Axis index of the slave axis) * numMachAxes + (axis index of the master axis)		numMachAxes * numMachAxes		

aaCplSetVal	$AA_CPLSETVAL[a,b]				
Indicates the type of defined value used for the coupling ACTPOS = Actual position CMDPOS = Setpoint position CMDVEL = Setpoint velocity					
-				String [32]	r
Multi-line: yes	(Axis index of the slave axis) * numMachAxes + (axis index of the master axis)		numMachAxes * numMachAxes		

aaCplState	$AA_CPLSTATE[a,b]				
A string which describes the actual status of the coupling DEF = Defined (but not yet activated) ON = Active OF = Deactivated					
-				String [32]	r
Multi-line: yes	(Axis index of the slave axis) * numMachAxes + (axis index of the master axis)		numMachAxes * numMachAxes		

aaCplType	$AA_CPLTYPE[a,b]				
Indicates the process that is used with the coupling of the stated following axis with the stated leading axis. 0 = NONE - No defined coupling with these axes Bit 0 (0x0001) - TRAIL - Uses a coupling factor Bit 1 (0x0002) - LEAD - Uses a curve table Bit 2 (0x0004) - ELG - An electronic gearbox link Bit 3 (0x0008) - Reserved Bit 4 (0x0010) - COUP - Spindle/partial spindle coupling Bit 5 (0x0020) - GANTRY - Coupling of the split axes (axes mechanically bound) Bit 6 (0x0040) - TANG - Tangential coupling with the aid of a curve table Bit 7 (0x0080) - GEN_CP - Generic coupling					
-				UWord	r
Multi-line: yes	(Axis index of the slave axis) * numMachAxes + (axis index of the master axis)		numMachAxes * numMachAxes		

3.13.3 Area C, Block WAL : Working area limitation

OEM-MMC: Linkitem /ChannelCoordSysWorkAreaLimits/...

This block contains the working area limitation data.

waCSCoordSys	$P_WORKAREA_CS_COORD_SYSTEM				
Coordinate system for working area limitation Identifier for the coordinate system in which the working area limitation is to apply. The following are valid: 0: Working area limitation in the WCS 3: Working area limitation in the SZS Special feature of line addressing: Any channel axis index can be selected as the channel axis index. Within a working area limitation group, the values are identical.					
-	0	0	3	UWord	rw
Multi-line: yes	Channel axis index + working area limitation group * numMachAxes		numMachAxes * $MC_MM_NUM_WORKAREA_CS_GROUPS		

waCSLimitMinus	$P_WORKAREA_CS_LIMIT_MINUS				
Position of the coordinate system-specific working area limitation in the negative direction for the addressed axis and working area group.					
-				Double	rw
Multi-line: yes	Channel axis index + working area limitation group * numMachAxes		numMachAxes * $MC_MM_NUM_WORKAREA_CS_GROUPS		

waCSLimitPlus	$P_WORKAREA_CS_LIMIT_PLUS				
Position of the coordinate system-specific working area limitation in the positive direction for the addressed axis and working area group.					
-				Double	rw
Multi-line: yes	Channel axis index + working area limitation group * numMachAxes		numMachAxes * $MC_MM_NUM_WORKAREA_CS_GROUPS		

waCSMinusEnable	$P_WORKAREA_CS_MINUS_ENABLE				
Coord.-specific working area limitation, negative valid TRUE: The limitation of waCSLimitMinus is valid.					
-	0	0	1	UWord	rw
Multi-line: yes	Channel axis index + working area limitation group * numMachAxes		numMachAxes * $MC_MM_NUM_WORKAREA_CS_GROUPS		

waCSPlusEnable	$P_WORKAREA_CS_PLUS_ENABLE				
Coord.-specific working area limitation, positive valid TRUE: The limitation of waCSLimitPlus is valid.					
-	0	0	1	UWord	rw
Multi-line: yes	Channel axis index + working area limitation group * numMachAxes		numMachAxes * $MC_MM_NUM_WORKAREA_CS_GROUPS		

3.13.4 Area N, Block VSYN : NCK-specific user variable for synchronous actions

OEM-MMC: Linkitem	/NckSelectedFunctionData/...

This block contains NCK-specific user variables for synchronous actions.

3.13.5 Area T, Block TDC : Tool parameters of the Siemens application

OEM-MMC: Linkitem /ToolTools/...

Tool parameters of the Siemens application

toolDataChangeInfo					
Siemens application tool parameter					
-				UWord	r
Multi-line: yes	TDC parameter no.				

3.13.6 Area T, Block TISO : ISO tool offset data

OEM-MMC: Linkitem /ToolIsoHDCompensation/...

This block contains the ISO tool offset data.

isoCorrParam	$TC_ISO_*				
This variable contains the offset values for the ISO2.2 and ISO3.2 modes. The column index contains the offset number.					
mm, inch, user defined	0			Double	rw
Multi-line: yes	1: Offset value for the geometry of the tool length in ISO2 mode. ($TC_ISO_H) 2: Offset value for the wear of the tool length in ISO2 mode. ($TC_ISO_HW) 3: Offset value for the geometry of the tool radius in ISO2 mode. ($TC_ISO_D) 4: Offset value for the wear of the tool radius in ISO2 mode. ($TC_ISO_DW) 5: Offset value for the geometry of the tool length L1 in ISO3 mode. ($TC_ISO_L1) 6: Offset value for the wear of the tool length L1 in ISO3 mode. ($TC_ISO_L1W) 7: Offset value for the geometry of the tool length L2 in ISO3 mode. ($TC_ISO_L2) 8: Offset value for the wear of the tool length L2 in ISO3 mode. ($TC_ISO_L2W) 9: Offset value for the geometry of the tool length L3 in ISO3 mode. ($TC_ISO_L3) 10: Offset value for the wear of the tool length L3 in ISO3 mode. ($TC_ISO_L3W) 11: Offset value for the geometry of the tool radius in ISO3 mode. ($TC_ISO_R) 12: Offset value for the wear of the tool radius in ISO3 mode. ($TC_ISO_RW) 13: Cutting edge position in ISO3 mode. ($TC_ISO_Q)		13		

3.14 Multitool status data

3.14.1 Area T, Block MTAD : Application-specific multi-tool data

OEM-MMC: Linkitem /ToolMT/...

This block contains application-specific multitool data.

siemData	$TC_MTPCSx[y] x=ParamNo y=MultitoolNo				
Siemens application multi-tool data Column index corresponds to parameter number. Reserved for SIEMENS applications.					
-	0.0			Double	rw
Multi-line: yes	Multi-tool number		32000		

3.14.2 Area T, Block MTAP : Application-specific multi-tool place data

OEM-MMC: Linkitem /ToolMTPlace/...

This block contains application-specific multitool location data.

siemPlaceData	$TC_MTPPCSx[y,z] x=ParamNo y=MtNo z=MtPlaceNo				
Siemens application multi-tool place data. These parameters can only be used if $MN_MM_NUM_CCS_MTLOC_PARAM and $MN_MM_TOOL_MANAGEMENT_MASK machine data have been set accordingly. Reserved for SIEMENS applications.					
-	0			Double	rw
Multi-line: yes	(MtLocNo-1)*numMultiToolPlaceParams_mtap+ParamNo		numMultiToolPlaceParams_mtap * maxNumPlacesPerMultitool		

3.14.3 Area T, Block MTD : Multi-tool data, general data

OEM-MMC: Linkitem /ToolMT/...

This block contains the general data of the multitool data.

multitoolIdent	$TC_MTP2				
MT identifier					
-				String [32]	rw
Multi-line: yes	Multi-tool number		32000		

multitoolInMag					
Number of the magazine on which the multi-tool is located					
-				UWord	r
Multi-line: yes	Multi-tool number		32000		

multitoolInPlace					
Number of the magazine place at which the multi-tool is located					
-				UWord	r
Multi-line: yes	Multi-tool number		32000		

multitoolKindOfDist	$TC_MTP_KD				
Type of distance coding 0: No multi-tool, or TMMG is not active 1: Multi-tool with place coding 2: Multi-tool with length coding 3: Multi-tool with angle coding 0: No multi-tool, or TMMG is not active 1: Multi-tool with place coding 2: Multi-tool with length coding 3: Multi-tool with angle coding					
-	1	0	3	UWord	r
Multi-line: yes	Multi-tool number		32000		

multitoolMyMag					
Owner magazine of the tool magazine from which the MT was loaded 0 = the MT has not been loaded. If multitoolInMag is >0, however, the MT number specifies a manual tool, or TMMG is not active					
-		0	max. Nummer eines def. Magazins	UWord	r
Multi-line: yes	Multi-tool number		32000		

multitoolMyPlace					
Owner magazine place of the MT - magazine place from which the multi-tool was loaded 0 = the MT is not loaded. If multitoolInPlace is >0, however, the MT number specifies a manual tool, a valid magazine place number, or TMMG is not active.					
-		0	max. Nummer def. Magazinplatz	UWord	r
Multi-line: yes	Multi-tool number		32000		

multitoolNumLoc	$TC_MTPN				
Number of places in MT					
-	0	0	$MN_MAX_TOOLS_PER_MULTITOOL	UWord	r
Multi-line: yes	Multi-tool number		32000		

multitoolPosition	$TC_MTP_POS				
MT position (number of the MT place)					
-	0	0	$MN_MAX_TOOLS_PER_MULTITOOL	UWord	rw
Multi-line: yes	Multi-tool number		32000		

multitoolProtAreaFile	$TC_MTP_PROTA				
Reserved, do not use!					
-				String [32]	rw
Multi-line: yes	Multi-tool number		32000		

multitoolStateL	$TC_MTP8				
Multi-tool status, significance of the bit values 0x0000: Not enabled 0x0001: Active MT 0x0002: Enabled 0x0004: Disabled 0x0008: Measured 0x0010: Prewarning limit reached 0x0020: MT being changed 0x0040: Fixed location coded 0x0080: MT was in use 0x0100: Autom. return 0x0200: Ignore disabled 0x0400: MT to be unloaded 0x0800: MT to be loaded 0x1000: Master tool 0x2000: Reserved 0x4000: Selected for 1:1 replacement 0x8000: Manual tool 0x10000: MT is disabled if a tool in the MT is disabled 0x20000: MT is at a disabled magazine location					
-	0			UDoubleword	rw
Multi-line: yes	Multi-tool number		32000		

multitoolplace_spec	$TC_MTP7				
MT magazine place type					
-				UWord	rw
Multi-line: yes	Multi-tool number		32000		

multitoolsize_down	$TC_MTP6				
MT size downward in half places					
-	1	1	7	UWord	rw
Multi-line: yes	Multi-tool number		32000		

multitoolsize_left	$TC_MTP3				
MT size to the left in half places					
-	1	1	7	UWord	rw
Multi-line: yes	Multi-tool number		32000		

multitoolsize_right	$TC_MTP4				
MT size to the right in half places					
-	1	1	7	UWord	rw
Multi-line: yes	Multi-tool number		32000		

multitoolsize_upper	$TC_MTP5				
MT size upward in half places					
-	1	1	7	UWord	rw
Multi-line: yes	Multi-tool number		32000		

3.14.4 Area T, Block MTP : Multi-tool data, place data

OEM-MMC: Linkitem /ToolMTPlace/...

This block contains the location data of the multitool data.

mtPlaceData	diverse, siehe Variablenbeschreibung				
P1: Place distance length ($TC_MTPPL) P2: Place distance angle ($TC_MTPPA) P3: Place type (read-only access) ($TC_MTPP2) P4: Place status (bit field) ($TC_MTPP4) P5: T number of the tool at this place ($TC_MTPP6) P6: Number of the adapter at this place ($TC_MTPP7)					
-				Double	rw
Multi-line: yes	(MtLocNo-1) * numMultiToolPlaceParams + ParamNo		numMultiToolPlaceParams * maxNumPlacesPerMultitool		

3.14.5 Area T, Block MTUD : Multi-tool data, user-defined data

OEM-MMC: Linkitem /ToolMT/...

This block contains the user-defined data of the multitool data.

userData	$TC_MTPCx[y] x=ParamNo y=MultitoolNo				
Multi-tool user data Column index corresponds to parameter number					
-	0.0			Double	rw
Multi-line: yes	Multi-tool number		32000		

3.14.6 Area T, Block MTUP : Multi-tool place user data

OEM-MMC: Linkitem /ToolMTPlace/...

This block contains the user multitool location data.

userPlaceData	$TC_MTPPCx[y,z] x=ParamNo y=MtNo z=MtPlaceNo				
Multi-tool place user data. These parameters can only be used if the machine data $MN_MM_NUM_CC_MTLOC_PARAM and $MN_MM_TOOL_MANAGEMENT_MASK have been set accordingly.					
-	0			Double	rw
Multi-line: yes	(MtLocNo-1)*numMultiToolPlaceParams_mtup+ParamNo		numMultiToolPlaceParams_mtup * maxNumPlacesPerMultitool		

3.14.7 Area T, Block MTV : Multi-tool data, directory

OEM-MMC: Linkitem /ToolMTCatalogue/...

This block contains the directory of the multitool data.

MTnumWZV					
Number of the last multi-tool to be created 0 = No multi-tools have been defined, or TMMG is not active					
-	0	0	32000	UWord	r
Multi-line: no					

multitoolIdent					
Multi-tool identifier "" = No multi-tool, or TMMG is not active					
-	""			String [32]	r
Multi-line: yes	Serial number, 1 - numMultiTools		$MN_MM_NUM_MULTITOOL		

multitoolInMag					
Number of the magazine on which the multi-tool is located 0 = The multi-tool is not loaded in a magazine, or TMMG is not active					
-	0	0	32000	UWord	r
Multi-line: yes	Serial number, 1 - numMultiTools		$MN_MM_NUM_MULTITOOL		

multitoolInPlace					
Number of the magazine place at which the multi-tool is located 0 = The multi-tool is not loaded in a magazine, or TMMG is not active					
-	0	0	32000	UWord	r
Multi-line: yes	Serial number, 1 - numMultiTools		$MN_MM_NUM_MULTITOOL		

multitoolKindOfDist					
Type of distance coding 0: No multi-tool, or TMMG is not active 1: Multi-tool with place coding 2: Multi-tool with length coding 3: Multi-tool with angle coding 0: No multi-tool, or TMMG is not active 1: Multi-tool with place coding 2: Multi-tool with length coding 3: Multi-tool with angle coding					
-	0	0	3	UWord	r
Multi-line: yes	Serial number, 1 - numMultiTools		$MN_MM_NUM_MULTITOOL		

multitoolNo	$P_MTOOLMT				
Number of the multi-tool. Array access to the column multitoolNo is possible in order to read all assigned MT numbers. 0 = No multi-tool, or TMMG is not active					
-	0	0	32000	UWord	r
Multi-line: yes	Serial number, 1 - numMultiTools		$MN_MM_NUM_MULTITOOL		

numLocations					
Number of places in the multi-tool					
-	0	0	$MN_MAX_TO OLS_PER_MUL TITOOL	UWord	r
Multi-line: yes	Serial number, 1 - numMultiTools		$MN_MM_NUM_MULTITOOL		

numMultiTools	$P_MTOOLN				
Number of defined multi-tools 0 = No multi-tools have been defined, or TMMG is not active					
-	0	0	32000	UWord	r
Multi-line: no					

4 Interface signals - overview

4.1 Overview of the PLC blocks

4.1.1 Organization blocks (OBs)

Table 4-1 Assignment of the organization blocks (OBs)

OB no.	Designation	Meaning	Package
1	CYCLE	Cyclic execution	GP
40	ALARM	Process alarms	GP
82	DIAGNOSTIC ALARM	Asynchronous error alarm	GP
86	RACK FAILURE	Asynchronous error alarm	GP
100	RESTART	Startup/restart	GP

4.1.2 Function blocks (FBs)

Table 4-2 Assignment of the function blocks (FBs)

Number		Designation	Meaning
0 - 29			Reserved for Siemens
	1	RUN_UP	Basic program start-up
	2	GET	Read NC variables
	3	PUT	Write NC variables
	4	PI_SERV	PI services
	5	GETGUD	Read GUD variable
	7	PI_SERV2	General PI services
	9	M2N	M to N transition block
	10	SI_relay	Safety Integrated relay
	11	SI_Braketest	Safety Integrated brake test
	29	Diagnostics	Signal recorder and data trigger diagnostics
30 - 999*			Free for user assignment
1000 - 1023			Reserved for Siemens
1024 - upper limit			Free for user assignment

* The actual upper limit of the block number depends on the PLC CPU contained in the selected NCU.

4.1.3 Function blocks (FCs)

Table 4-3 Assignment of the function blocks (FCs)

Number		Designation	Meaning
0 - 29			Reserved for Siemens
	2	GP_HP	Basic program cyclic part
	3	GP_PRAL	Basic program alarm-controlled part
	5	GP_DIAG	Basic program diagnostic alarm and module failure
	6	TM_TRANS2	Transfer block for tool management and multitool
	7	TM_REV	Transfer block for tool change with turret
	8	TM_TRANS	Transfer block for tool management
	9	ASUB	Asynchronous subprograms
	10	AL_MSG	Alarms/messages
	12	AUXFU	Call interface for user auxiliary functions
	13	BHG_DISP	Display control for the handheld unit
	17	YDelta	Star-delta changeover
	18	SpinCtrl	Spindle control for PLC
	19	MCP_IFM	Distribution of MCP and operating software signals at the interface (milling machine)
	21	Transfer	PLC-NC data exchange
	22	TM_DIR	Direction selection for tool management
	24	MCP_IFM2	Transmission of the MCP signals to the interface
	25	MCP_IFT	Transmission of the MCO/OP signals to the interface
	26	HPU_MCP	Transmission of the HT 8 signals to the interface
30 - 999*			Free for user assignment
1005		AG_SEND	Sends data to the Ethernet CP
1006		AG_RECV	Receives data from the Ethernet CP
1000 - 1023			Reserved for Siemens
1024 - upper limit			Free for user assignment

* The actual upper limit of the block number depends on the PLC CPU contained in the selected NCU.

4.1.4 Data blocks (DBs)

Note

The number of DBs to be set up depends on the parameterization made in the NC machine data.

Note

Data blocks of channels, axes/spindles and tool management that have not been activated are available for the user.

Table 4-4 Overview of the data blocks (DBs)

DB no.	Designation	Interface for
1		Reserved for Siemens
2 - 5	PLC-MELD	PLC messages
6 - 8		Basic program
9	NC-COMPILE	NC compile cycles
10	NC INTERFACE	Central NC
11	Mode group	Mode group
12		Computer link and transport system
13		Reserved for Hymnos
14		Reserved for basic program
15		Basic program
16		PI Service definitions
17		Version identifier
18		Reserved for basic program (SPL interface (Safety Integrated))
19		Operating software
20		PLC machine data
21 - 30	CHANNEL 1 ... CHANNEL 10	NC channels
31 - 61	AXIS 1 ... AXIS 31	Axes/spindle
62 - 70		Free for user assignment
71 - 74		User tool management
75 - 76		M group decoding
77		MCP, HHU signals (for SDB210)
78 - 80		Reserved for Siemens
81 - 127		Free for user assignment
1000		Ctrl-Energy
1001		SENTRON PAC
1002 … 1070		Reserved for Siemens
1071		Load/unload magazine (multitool)
1072		Spindle (multitool)
1073		Turret (multitool)
1074 - 1099		Reserved for Siemens

4.1.5 Timer block

Table 4-5 Assigned times

Timer no.	Meaning
0 - 512*	Free for user assignment

* The actual upper limit of the timer number (DB) depends on the PLC CPU contained in the selected NCU.

4.2 Signals from/to the machine control panel

4.2.1 M version, signals from the MCP: Input image

Table 4-6 M version, signals from the MCP: Input image

	Signals from the MCP (keys) (MCP → PLC)							
Byte	**Bit 7**	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
EB n + 0	Spindle override				Operating mode			
	D	C	B	A	JOG	TEACH IN	MDI	AUTO
EB n + 1	Machine function							
	REPOS	REF	INCvar	INC10000	INC1000	INC100	INC10	INC1
EB n + 2	Key-operated switch position 0	Key-operated switch position 2	Spindle start	*Spindle stop	Feedrate start	*Feedrate stop	NC start	*NC stop
EB n + 3		Key-operated switch position 1		Feedrate override				
	Reset		Single block	E	D	C	B	A
EB n + 4	Direction keys			Key-operated switch position 3	Axis selection			
	+R15	-R13	Rapid traverse R14		X R1	4. axis R4	7. axis R7	R10
EB n + 5	Axis selection							
	Y R2	Z R3	5. axis R5	Travel command MCS/WCS	R11	9. axis R9	8. axis R8	6. axis R6
EB n + 6	Freely assignable customer keys							
	T9	T10	T11	T12	T13	T14	T15	
EB n + 7	Freely assignable customer keys							
	T1	T2	T3	T4	T5	T6	T7	T8

4.2.2 M version, signals to the MCP: Output image

Table 4-7 M version, signals to the MCP: Output image

	Signals to the MCP (LED) (PLC → MCP)							
Byte	**Bit 7**	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
AB n + 0	Machine function				Operating mode			
	INC1000	INC100	INC10	INC1	JOG	TEACH IN	MDI	AUTO

	Signals to the MCP (LED) (PLC → MCP)							
Byte	**Bit 7**	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
AB n + 1	Feedrate start	*Feedrate stop	NC start	*NC stop	Machine function			
					REPOS	REF	INCvar	INC10000
AB n + 2	Direction key -R13	Axis selection				Single block	Spindle start	*Spindle stop
		X R1	4. Axis R4	7. axis R7	R10			
AB n + 3	Axis selection							Direction key +R15
	Z R3	5. axis R5	Travel command MCS/WCS R12	R11	9. axis R9	8. axis R8	6. axis R6	
AB n + 4	Freely assignable customer keys							Y R2
	T9	T10	T11	T12	T13	T14	T15	
AB n + 5	Freely assignable customer keys							
	T1	T2	T3	T4	T5	T6	T7	T8

4.2.3 T version, signals from the MCP: Input image

Table 4-8 T version, signals from the MCP: Input image

	Signals from the MCP (keys) (MCP → PLC)							
Byte	**Bit 7**	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
EB n + 0	Spindle override				Operating mode			
	D	C	B	A	JOG	TEACH IN	MDI	AUTO
EB n + 1	Machine function							
	REPOS	REF	INCvar	INC10000	INC1000	INC100	INC10	INC1
EB n + 2	Key-operated switch position 0	Key-operated switch position 2	Spindle start	*Spindle stop	Feedrate start	*Feedrate stop	NC start	*NC stop
EB n + 3		Key-operated switch position 1		Feedrate override				
	Reset		Single block	E	D	C	B	A
EB n + 4	Direction keys			Key-operated switch position 3	Direction keys			
	R15	R13	R14		+Y R1	-Z R4	-C R7	R10
EB n + 5	Direction keys							
	+X R2	+C R3	Rapid traverse override R5	Travel command MCS/ WCS R12	R11	-Y R9	-X R8	+Z R6
EB n + 6	Freely assignable customer keys							
	T9	T10	T11	T12	T13	T14	T15	

	Signals from the MCP (keys) (MCP → PLC)							
Byte	**Bit 7**	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
EB n + 7	Freely assignable customer keys							
	T1	T2	T3	T4	T5	T6	T7	T8

4.2.4 T version, signals to the MCP: Output image

Table 4-9 T version, signals to the MCP: Output image

	Signals to the MCP (LED) (PLC → MCP)							
Byte	**Bit 7**	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
AB n + 0	Machine function				Operating mode			
	INC1000	INC100	INC10	INC1	JOG	TEACH IN	MDI	AUTO
AB n + 1	Feedrate start	*Feedrate stop	NC start	*NC stop	Machine function			
					REPOS	REF	INCvar	INC10000
AB n + 2	Direction keys					Single block	Spindle start	*Spindle stop
	R13	+Y R1	-Z R4	-C R7	R10			
AB n + 3	Direction keys							
	R3	R5	Travel com-mand MCS/WCS	R11	-Y R9	-X R8	+Z R6	R15
AB n + 4	Freely assignable customer keys							Direction key +X R2
	T9	T10	T11	T12	T13	T14	T15	
AB n + 5	Freely assignable customer keys							
	T1	T2	T3	T4	T5	T6	T7	T8

4.2.5 Slimline version, signals from the MCP: Input image

Table 4-10 Slimline version, signals from the MCP: Input image

	Signals from the slimline MCP (switches and keys) (MCP → PLC)							
Byte	**Bit 7**	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
EB n + 0	Spindle override				Operating mode			
	*NC stop	SP -	SP 100%	SP +	SINGLEB	JOG	MDI	AUTO
EB n + 1	Spindle				Key-operat-ed switch	Machine function		
	NC start	SP right	*SP stop	SP left	INT 3	REF	REPOS	TEACH IN
EB n + 2	Feedrate			Key-operat-ed switch	Machine functions			
	Start	*Stop	INCvar	INT 0	INC1000	INC100	INC10	INC1

	Signals from the slimline MCP (switches and keys) (MCP → PLC)							
Byte	**Bit 7**	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
EB n + 3		Key-operated switch		Feedrate override				
	Reset	INT 2	INT 1	E	D	C	B	A
EB n + 4	Direction keys			Optional customer keys				
	+R15	-R13	Rapid traverse R14	KT4	KT3	KT2	KT1	KT0
EB n + 5			Axis selection					
	T17	KT5	6	5	4	Z	Y	X
EB n + 6	Freely assignable customer keys				MCS/WCS	Freely assignable customer keys		
	T9	T10	T11	T12		T14	T15	T16
EB n + 7	Freely assignable customer keys							
	T1	T2	T3	T4	T5	T6	T7	T8

4.2.6 Slimline version, signals to the MCP: Output image

Table 4-11 Slimline version, signals to the MCP: Output image

	Signals to the slimline MCP (LED) (PLC → MCP)							
Byte	**Bit 7**	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
EB n + 0	Spindle override				Operating mode			
	NC stop	SP -	SP 100%	SP +	SINGLEB	JOG	MDI	AUTO
EB n + 1	Spindle					Machine function		
	NC start	SP right	SP stop	SP left	Not assigned	REF	REPOS	TEACH IN
EB n + 2	Feedrate				Machine functions			
	Start	Stop	INCvar	Not assigned	INC1000	INC100	INC10	INC1
EB n + 3	Not assigned							
EB n + 4	Direction keys			Optional customer keys				
	+R15	-R13	Rapid traverse R14	KT4	KT3	KT2	KT1	KT0
EB n + 5			Axis selection					
	T17	KT5	6	5	4	Z	Y	X
EB n + 6	Freely assignable customer keys					Freely assignable customer keys		
	T9	T10	T11	T12	MCS/WCS	T14	T15	T16
EB n + 7	Freely assignable customer keys							
	T1	T2	T3	T4	T5	T6	T7	T8

4.3 Signals from/to the handheld unit HT 2

4.3.1 Signals from the handheld unit: Input image

Table 4-12 Signals from the handheld unit: Input image

	Signals from the handheld unit (keys) (HHU / HT 2 → PLC)							
Byte	**Bit 7**	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
EB n + 0	Reserved							
								Identifier HT 2
EB n + 1	Reserved							
EB n + 2								
	T9	T7	T6	T5	T4	T3	T2	T1
EB n + 3								
	T16	T15	T14	T13	T12	T11	T10	T9
EB n + 4								
	T24	T23	T22	T21				
EB n + 5	Acknowl-edgement		Rapid traverse/feedrate override switch					
	Digital dis-play	Key-operat-ed switch	E	D	C	B	A	

4.3.2 Signals to the handheld unit: Output image

Table 4-13 Signals to the handheld unit: Output image

	Signals to the handheld unit (LED) (PLC → HHU / HT 2)							
Byte	**Bit 7**	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
AB n + 0								
	Always 1							
AB n + 1	New data for selec-ted line						Selecting the line	
							Z3 and Z4	Z1 and Z2
AB n + 2								
	L8	L7	L6	L5	L4	L3	L2	L1
AB n + 3								
	L16	L15	L14	L13	L12	L11	L10	L9
	Digital display of the handheld unit							
AB n + 4	1. character (right) of the selected line							

	Signals to the handheld unit (LED) (PLC → HHU / HT 2)							
Byte	**Bit 7**	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
AB n + 5	2. character of the selected line							
AB ...	...							
AB n + 18	15. character of the selected line							
AB n + 19	16. character (left) of the selected line							

Note

The parameterization or configuration of the various MCP/HHU versions is described in:

References

- Manual, Operator Components and Networking
- Function Manual, Basic Functions: Basic PLC program (P3)

4.4 Signals from/to the handheld unit HT 8

4.4.1 Signals from the MCP simulation: Input image

Table 4-14 Signals from the MCP simulation: Input image

	Signals from the MCP simulation (HT 8 → PLC)							
Byte	**Bit 7**	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
EB n + 0	Function key block							
	REF	TEACH	AUTO	MDI	JOG	QUIT	Reset	Work/ Machine
EB n + 1	Function key block							
	CPF (U key)	U4	U3	BigFct	U2	U1	INC	REPOS
EB n + 2		Change-over axes (HMI Advanced only)	Traversing keys (JOG) positive direction					
		Ax7-Ax12 instead of Ax1-Ax6	Ax6	Ax5	Ax4	Ax3	Ax2	Ax1
EB n + 3	Traversing keys (JOG) negative direction							
			Ax6	Ax5	Ax4	Ax3	Ax2	Ax1
EB n + 4								
	U9	U10	U11	U12	U13	U14	U15	U16
EB n + 5								
		U8	U7	U6	U5	SBL		
EB n + 6			Start key block					
	Reserved	HT 8	SF2	SF1	SF4	SF3	Start	Stop
EB n + 7	Feedrate override							
				E	D	C	B	A

4.4.2 Signals to the MCP simulation: Output image

Table 4-15 Signals from the MCP simulation: Output image

	Signals to the MCP simulation (PLC → HT 8)							
Byte	**Bit 7**	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
AB n + 0	Function key block							
	REF	TEACH	AUTO	MDI	JOG	QUIT	Reset	Work/ Machine
AB n + 1	Function key block							
		U4	U3		U2	U1	INC	REPOS

	Signals to the MCP simulation (PLC → HT 8)							
Byte	**Bit 7**	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
AB n + 2		Axes 7 - n selected	Traversing keys (JOG) positive direction					
			Ax6	Ax5	Ax4	Ax3	Ax2	Ax1
AB n + 3	Traversing keys (JOG) negative direction							
	For WCS: no machine axes		Ax6	Ax5	Ax4	Ax3	Ax2	Ax1
AB n + 4								
	U9	U10	U11	U12	U13	U14	U15	U16
AB n + 5								
		U8	U7	U6	U5	SBL		
AB n + 6	Start key block							
	Display traversing keys		SF2	SF1	SF4	SF3	Start	Stop
AB n + 7								

4.5 PLC alarms/messages

4.5.1 FC 10 alarms in the DB2 (FB1: "ExtendAlMsg" == FALSE)

Message type

- **FM**: A fault message with the associated event number as fault number is triggered by the signal.
- **OM**: An operating message with the associated event number as message number is triggered by the signal.

References

A detailed description for error and operating messages can be found in the:
Function Manual, Basic Functions; Chapter "P3: Basic PLC program for SINUMERIK 840D sl", "Block descriptions", "FC10: AL_MSG - fault and operating messages"

Table 4-16 DB2, channel range 1

DB2	Signals for PLC events (PLC → HMI) FB1 parameter "ExtendAlMsg" == FALSE							
Byte (message type)	**Bit 7**	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
	Channel 1							
	Feed disable (event no.: 510000-510015)							
0 (FM)	510007	510006	510005	510004	510003	510002	510001	510000
1 (OM)	510015	510014	510013	510012	510011	510010	510009	510008
2 (FM)	Feed and read-in disable, byte 1 (event no.: 510100-510107)							
3 (FM)	Feed and read-in disable, byte 2 (event no.: 510108-510115)							
4 (OM)	Feed and read-in disable, byte 3 (event no.: 510116-510123)							
5 (OM)	Feed and read-in disable, byte 4 (event no.: 510124-510131)							
6 (FM)	Read-in disable, byte 1 (event no.: 510200-510207)							
7 (FM)	Read-in disable, byte 2 (event no.: 510208-510215)							
8 (OM)	Read-in disable, byte 3 (event no.: 510216-510223)							
9 (OM)	Read-in disable, byte 4 (event no.: 510224-510231)							
10 (FM)	NC start disable, byte 1 (event no.: 510300-510307)							
11 (OM)	NC start disable, byte 2 (event no.: 510308-510315)							
12 (FM)	Feed stop, geo axis 1, byte 1 (event no.: 511100-511107)							
13 (OM)	Feed stop, geo axis 1, byte 2 (event no.: 511108-511115)							
14 (FM)	Feed stop, geo axis 2, byte 1 (event no.: 511200-511207)							
15 (OM)	Feed stop, geo axis 2, byte 2 (event no.: 511208-511215)							
16 (FM)	Feed stop, geo axis 3, byte 1 (event no.: 511300-511307)							
17 (OM)	Feed stop, geo axis 3, byte 2 (event no.: 511308-511315)							

Table 4-17 DB2, channel range 2

DB2	Signals for PLC events (PLC → HMI) FB1 parameter "ExtendAIMsg" == FALSE							
Byte (message type)	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
	Channel 2							
	Feed disable (event no.: 520000-520015)							
18 (FM)	520007	520006	520005	520004	520003	520002	520001	520000
19 (OM)	520015	520014	520013	520012	520011	520010	520009	520008
20 (FM)	Feed and read-in disable, byte 1 (event no.: 520100-520107)							
21 (FM)	Feed and read-in disable, byte 2 (event no.: 520108-520115)							
22 (OM)	Feed and read-in disable, byte 3 (event no.: 520116-520123)							
23 (OM)	Feed and read-in disable, byte 4 (event no.: 520124-520131)							
24 (FM)	Read-in disable, byte 1 (event no.: 520200-520207)							
25 (FM)	Read-in disable, byte 2 (event no.: 520208-520215)							
26 (OM)	Read-in disable, byte 3 (event no.: 520216-520223)							
27 (OM)	Read-in disable, byte 4 (event no.: 520224-520231)							
28 (FM)	NC start disable, byte 1 (event no.: 520300-520307)							
29 (OM)	NC start disable, byte 2 (event no.: 520308-520315)							
30 (FM)	Feed stop, geo axis 1, byte 1 (event no.: 521100-521107)							
31 (OM)	Feed stop, geo axis 1, byte 2 (event no.: 521108-521115)							
32 (FM)	Feed stop, geo axis 2, byte 1 (event no.: 521200-521207)							
33 (OM)	Feed stop, geo axis 2, byte 2 (event no.: 521208-521215)							
34 (FM)	Feed stop, geo axis 3, byte 1 (event no.: 521300-521307)							
35 (OM)	Feed stop, geo axis 3, byte 2 (event no.: 521308-521315)							

Table 4-18 DB2, channel range 3

DB2	Signals for PLC events (PLC → HMI) FB1 parameter "ExtendAIMsg" == FALSE							
Byte (message type)	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
	Channel 3							
	Feed disable (event no.: 530000-530015)							
36 (FM)	530007	530006	530005	530004	530003	530002	530001	530000
37 (OM)	530015	530014	530013	530012	530011	530010	530009	530008
38 (FM)	Feed and read-in disable, byte 1 (event no.: 530100-530107)							
39 (FM)	Feed and read-in disable, byte 2 (event no.: 530108-530115)							
40 (OM)	Feed and read-in disable, byte 3 (event no.: 530116-530123)							
41 (OM)	Feed and read-in disable, byte 4 (event no.: 530124-530131)							
42 (FM)	Read-in disable, byte 1 (event no.: 530200-530207)							

DB2	Signals for PLC events (PLC → HMI) FB1 parameter "ExtendAlMsg" == FALSE						
Byte (message type)	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
43 (FM)	Read-in disable, byte 2 (event no.: 530208-530215)							
44 (OM)	Read-in disable, byte 3 (event no.: 530216-530223)							
45 (OM)	Read-in disable, byte 4 (event no.: 530224-530231)							
46 (FM)	NC start disable, byte 1 (event no.: 530300-530307)							
47 (OM)	NC start disable, byte 2 (event no.: 530308-530315)							
48 (FM)	Feed stop, geo axis 1, byte 1 (event no.: 531100-531107)							
49 (OM)	Feed stop, geo axis 1, byte 2 (event no.: 531108-531115)							
50 (FM)	Feed stop, geo axis 2, byte 1 (event no.: 531200-531207)							
51 (OM)	Feed stop, geo axis 2, byte 2 (event no.: 531208-531215)							
52 (FM)	Feed stop, geo axis 3, byte 1 (event no.: 531300-531307)							
53 (OM)	Feed stop, geo axis 3, byte 2 (event no.: 531308-531315)							

Table 4-19 DB2, channel range 4

DB2	Signals for PLC events (PLC → HMI) FB1 parameter "ExtendAlMsg" == FALSE							
Byte (message type)	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
	Channel 4							
	Feed disable (event no.: 540000-540015)							
54 (FM)	540007	540006	540005	540004	540003	540002	540001	540000
55 (OM)	540015	540014	540013	540012	540011	540010	540009	540008
56 (FM)	Feed and read-in disable, byte 1 (event no.: 540100-540107)							
57 (FM)	Feed and read-in disable, byte 2 (event no.: 540108-540115)							
58 (OM)	Feed and read-in disable, byte 3 (event no.: 540116-540123)							
59 (OM)	Feed and read-in disable, byte 4 (event no.: 540124-540131)							
60 (FM)	Read-in disable, byte 1 (event no.: 540200-540207)							
61 (FM)	Read-in disable, byte 2 (event no.: 540208-540215)							
62 (OM)	Read-in disable, byte 3 (event no.: 540216-540223)							
63 (OM)	Read-in disable, byte 4 (event no.: 540224-540231)							
64 (FM)	NC start disable, byte 1 (event no.: 540300-540307)							
65 (FM)	NC start disable, byte 2 (event no.: 540308-540315)							
66 (FM)	Feed stop, geo axis 1, byte 1 (event no.: 541100-541107)							
67 (OM)	Feed stop, geo axis 1, byte 2 (event no.: 541108-541115)							
68 (FM)	Feed stop, geo axis 2, byte 1 (event no.: 541200-541207)							
69 (OM)	Feed stop, geo axis 2, byte 2 (event no.: 541208-541215)							
70 (FM)	Feed stop, geo axis 3, byte 1 (event no.: 541300-541307)							
71 (OM)	Feed stop, geo axis 3, byte 2 (event no.: 541308-541315)							

Table 4-20 DB2, channel range 5

DB2	Signals for PLC events (PLC → HMI) FB1 parameter "ExtendAIMsg" == FALSE							
Byte (message type)	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
	Channel 5							
	Feed disable (event no.: 550000-550015)							
72 (FM)	550007	550006	550005	550004	550003	550002	550001	550000
73 (OM)	550015	550014	550013	550012	550011	550010	550009	550008
74 (FM)	Feed and read-in disable, byte 1 (event no.: 550100-550107)							
75 (OM)	Feed and read-in disable, byte 2 (event no.: 550108-550115)							
76 (OM)	Feed and read-in disable, byte 3 (event no.: 550116-550123)							
77 (OM)	Feed and read-in disable, byte 4 (event no.: 550124-550131)							
78 (FM)	Read-in disable, byte 1 (event no.: 550200-550207)							
79 (FM)	Read-in disable, byte 2 (event no.: 550208-550315)							
80 (OM)	Read-in disable, byte 3 (event no.: 550216-550223)							
81 (OM)	Read-in disable, byte 4 (event no.: 550224-550231)							
82 (FM)	NC start disable, byte 1 (event no.: 550300-550307)							
83 (OM)	NC start disable, byte 2 (event no.: 550308-550315)							
84 (FM)	Feed stop, geo axis 1, byte 1 (event no.: 551100-551107)							
85 (OM)	Feed stop, geo axis 1, byte 2 (event no.: 551108-551115)							
86 (FM)	Feed stop, geo axis 2, byte 1 (event no.: 551200-551207)							
87 (OM)	Feed stop, geo axis 2, byte 2 (event no.: 551208-551215)							
88 (FM)	Feed stop, geo axis 3, byte 1 (event no.: 551300-551307)							
89 (OM)	Feed stop, geo axis 3, byte 2 (event no.: 551308-551315)							

Table 4-21 DB2, channel range 6

DB2	Signals for PLC events (PLC → HMI) FB1 parameter "ExtendAIMsg" == FALSE							
Byte (message type)	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
	Channel 6							
	Feed disable (event no.: 560000-560015)							
90 (FM)	560007	560006	560005	560004	560003	560002	560001	560000
91 (OM)	560015	560014	560013	560012	560011	560010	560009	560008
92 (FM)	Feed and read-in disable, byte 1 (event no.: 560100-560107)							
93 (FM)	Feed and read-in disable, byte 2 (event no.: 560108-560115)							
94 (OM)	Feed and read-in disable, byte 3 (event no.: 560116-560123)							
95 (OM)	Feed and read-in disable, byte 4 (event no.: 560124-560131)							
96 (FM)	Read-in disable, byte 1 (event no.: 560200-560207)							

DB2	Signals for PLC events (PLC → HMI) FB1 parameter "ExtendAlMsg" == FALSE							
Byte (message type)	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
97 (FM)	Read-in disable, byte 2 (event no.: 560208-560315)							
98 (OM)	Read-in disable, byte 3 (event no.: 560216-560223)							
99 (OM)	Read-in disable, byte 4 (event no.: 560224-560231)							
100 (FM)	NC start disable, byte 1 (event no.: 560300-560307)							
101 (OM)	NC start disable, byte 2 (event no.: 560308-560315)							
102 (FM)	Feed stop, geo axis 1, byte 1 (event no.: 561100-561107)							
103 (OM)	Feed stop, geo axis 1, byte 2 (event no.: 561108-561115)							
104 (FM)	Feed stop, geo axis 2, byte 1 (event no.: 561200-561207)							
105 (OM)	Feed stop, geo axis 2, byte 2 (event no.: 561208-561215)							
106 (FM)	Feed stop, geo axis 3, byte 1 (event no.: 561300-561307)							
107 (OM)	Feed stop, geo axis 3, byte 2 (event no.: 561308-561315)							

Table 4-22 DB2, channel range 7

DB2	Signals for PLC events (PLC → HMI) FB1 parameter "ExtendAlMsg" == FALSE							
Byte (message type)	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
	Channel 7							
	Feed disable (event no.: 570000-570015)							
108 (FM)	570007	570006	570005	570004	570003	570002	570001	570000
109 (OM)	570015	570014	570013	570012	570011	570010	570009	570008
110 (FM)	Feed and read-in disable, byte 1 (event no.: 570100-570107)							
111 (FM)	Feed and read-in disable, byte 2 (event no.: 570108-570115)							
112 (OM)	Feed and read-in disable, byte 3 (event no.: 570116-570123)							
113 (OM)	Feed and read-in disable, byte 4 (event no.: 570124-570131)							
114 (FM)	Read-in disable, byte 1 (event no.: 570200-570207)							
115 (FM)	Read-in disable, byte 2 (event no.: 570208-570315)							
116 (OM)	Read-in disable, byte 3 (event no.: 570216-570223)							
117 (OM)	Read-in disable, byte 4 (event no.: 570224-570231)							
118 (FM)	NC start disable, byte 1 (event no.: 570300-570307)							
119 (OM)	NC start disable, byte 2 (event no.: 570308-570315)							
120 (FM)	Feed stop, geo axis 1, byte 1 (event no.: 571100-571107)							
121 (OM)	Feed stop, geo axis 1, byte 2 (event no.: 571108-571115)							
122 (FM)	Feed stop, geo axis 2, byte 1 (event no.: 571200-571207)							
123 (OM)	Feed stop, geo axis 2, byte 2 (event no.: 571208-571215)							
124 (FM)	Feed stop, geo axis 3, byte 1 (event no.: 571300-571307)							
125 (OM)	Feed stop, geo axis 3, byte 2 (event no.: 571308-571315)							

Table 4-23 DB2, channel range 8

DB2	Signals for PLC events (PLC → HMI) FB1 parameter "ExtendAIMsg" == FALSE							
Byte (message type)	**Bit 7**	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
	Channel 8							
	Feed disable (event no.: 580000-580015)							
126 (FM)	580007	580006	580005	580004	580003	580002	580001	580000
127 (OM)	580015	580014	580013	580012	580011	580010	580009	580008
128 (FM)	Feed and read-in disable, byte 1 (event no.: 580100-580107)							
129 (FM)	Feed and read-in disable, byte 2 (event no.: 580108-580115)							
130 (OM)	Feed and read-in disable, byte 3 (event no.: 580116-580123)							
131 (OM)	Feed and read-in disable, byte 4 (event no.: 580124-580131)							
132 (FM)	Read-in disable, byte 1 (event no.: 580200-580207)							
133 (FM)	Read-in disable, byte 2 (event no.: 580208-580315)							
134 (OM)	Read-in disable, byte 3 (event no.: 580216-580223)							
135 (OM)	Read-in disable, byte 4 (event no.: 580224-580231)							
136 (FM)	NC start disable, byte 1 (event no.: 580300-580307)							
137 (OM)	NC start disable, byte 2 (event no.: 580308-580315)							
138 (FM)	Feed stop, geo axis 1, byte 1 (event no.: 581100-581107)							
139 (OM)	Feed stop, geo axis 1, byte 2 (event no.: 581108-581115)							
140 (FM)	Feed stop, geo axis 2, byte 1 (event no.: 581200-581207)							
141 (OM)	Feed stop, geo axis 2, byte 2 (event no.: 581208-581215)							
142 (FM)	Feed stop, geo axis 3, byte 1 (event no.: 581300-581307)							
143 (OM)	Feed stop, geo axis 3, byte 2 (event no.: 581308-581315)							
	Channels 9 and **10** not implemented							

Table 4-24 DB2, axis ranges

DB2	Signals for PLC events (PLC → HMI) FB1 parameter "ExtendAIMsg" == FALSE							
Byte (message type)	**Bit 7**	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
	Axis/spindle							
	Feed stop / spindle stop for axis/spindle 1 (event no.: 600100-600115)							
144 (FM)	600107	600106	600105	600104	600103	600102	600101	600100
145 (OM)	600115	600114	600113	600112	600111	600110	600109	600108
146 (FM)	Feed stop / spindle stop for axis/spindle 2, byte 1 (event no.: 600200-600207)							
147 (OM)	Feed stop / spindle stop for axis/spindle 2, byte 2 (event no.: 600208-600215)							
148 (FM)	Feed stop / spindle stop for axis/spindle 3, byte 1 (event no.: 600300-600307)							
149 (OM)	Feed stop / spindle stop for axis/spindle 3, byte 2 (event no.: 600308-600315)							

DB2	Signals for PLC events (PLC → HMI) FB1 parameter "ExtendAlMsg" == FALSE							
Byte (message type)	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
150 (FM)	Feed stop / spindle stop for axis/spindle 4, byte 1 (event no.: 600400-600407)							
151 (OM)	Feed stop / spindle stop for axis/spindle 4, byte 2 (event no.: 600408-600415)							
152 (FM)	Feed stop / spindle stop for axis/spindle 5, byte 1 (event no.: 600500-600507)							
153 (OM)	Feed stop / spindle stop for axis/spindle 5, byte 2 (event no.: 600508-600515)							
154 (FM)	Feed stop / spindle stop for axis/spindle 6, byte 1 (event no.: 600600-600607)							
155 (OM)	Feed stop / spindle stop for axis/spindle 6, byte 2 (event no.: 600608-600615)							
156 (FM)	Feed stop / spindle stop for axis/spindle 7, byte 1 (event no.: 600700-600707)							
157 (OM)	Feed stop / spindle stop for axis/spindle 7, byte 2 (event no.: 600708-600715)							
158 (FM)	Feed stop / spindle stop for axis/spindle 8, byte 1 (event no.: 600800-600807)							
159 (OM)	Feed stop / spindle stop for axis/spindle 8, byte 2 (event no.: 600808-600815)							
160 (FM)	Feed stop / spindle stop for axis/spindle 9, byte 1 (event no.: 600900-600907)							
161 (OM)	Feed stop / spindle stop for axis/spindle 9, byte 2 (event no.: 600908-600915)							
162 (FM)	Feed stop / spindle stop for axis/spindle 10, byte 1 (event no.: 601000-601007)							
163 (OM)	Feed stop / spindle stop for axis/spindle 10, byte 2 (event no.: 601008-601015)							
164 (FM)	Feed stop / spindle stop for axis/spindle 11, byte 1 (event no.: 601100-601107)							
165 (OM)	Feed stop / spindle stop for axis/spindle 11, byte 2 (event no.: 601108-601115)							
166 (FM)	Feed stop / spindle stop for axis/spindle 12, byte 1 (event no.: 601200-601207)							
167 (OM)	Feed stop / spindle stop for axis/spindle 12, byte 2 (event no.: 601208-601215)							
168 (FM)	Feed stop / spindle stop for axis/spindle 13, byte 1 (event no.: 601300-601307)							
169 (OM)	Feed stop / spindle stop for axis/spindle 13, byte 2 (event no.: 601308-601315)							
170 (FM)	Feed stop / spindle stop for axis/spindle 14, byte 1 (event no.: 601400-601407)							
171 (OM)	Feed stop / spindle stop for axis/spindle 14, byte 2 (event no.: 601408-601415)							
172 (FM)	Feed stop / spindle stop for axis/spindle 15, byte 1 (event no.: 601500-601507)							
173 (OM)	Feed stop / spindle stop for axis/spindle 15, byte 2 (event no.: 601508-601515)							
174 (FM)	Feed stop / spindle stop for axis/spindle 16, byte 1 (event no.: 601600-601607)							
175 (OM)	Feed stop / spindle stop for axis/spindle 16, byte 2 (event no.: 601608-601615)							
176 (FM)	Feed stop / spindle stop for axis/spindle 17, byte 1 (event no.: 601700-601707)							
177 (OM)	Feed stop / spindle stop for axis/spindle 17, byte 2 (event no.: 601708-601715)							
178 (FM)	Feed stop / spindle stop for axis/spindle 18, byte 1 (event no.: 601800-601807)							
179 (OM)	Feed stop / spindle stop for axis/spindle 18, byte 2 (event no.: 601808-601815)							
	Axes 19 – 31 not realized							

Table 4-25 DB2, user ranges

DB2	Signals for PLC events (PLC → HMI) FB1 parameter "ExtendAlMsg" == FALSE							
Byte (message type)	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
	User ranges							
	User range 0 (event no.: 700000-700015)							
180 (FM)	700007	700006	700005	700004	700003	700002	700001	700000
181 (FM)	700015	700014	700013	700012	700011	700010	700009	700008
182 (FM)	User range 0, byte 3 (event no.: 700016-700023)							
183 (FM)	User range 0, byte 4 (event no.: 700024-700031)							
184 (OM)	User range 0, byte 5 (event no.: 700032-700039)							
185 (OM)	User range 0, byte 6 (event no.: 700040-700047)							
186 (OM)	User range 0, byte 7 (event no.: 700048-700055)							
187 (OM)	User range 0, byte 8 (event no.: 700056-700063)							
188 - 191 (FM)	User range 1, bytes 1 - 4 (event no.: 700100-700131)							
192 - 195 (OM)	User range 1, bytes 5 - 8 (event no.: 700132-700163)							
196 - 199 (FM)	User range 2, bytes 1 - 4 (event no.: 700200-700231)							
200 - 203 (OM)	User range 2, bytes 5 - 8 (event no.: 700232-700263)							
204 - 207 (FM)	User range 3, bytes 1 - 4 (event no.: 700300-700331)							
208 - 211 (OM)	User range 3, bytes 5 - 8 (event no.: 700332-700363)							
212 - 215 (FM)	User range 4, bytes 1 - 4 (event no.: 700400-700431)							
216 - 219 (OM)	User range 4, bytes 5 - 8 (event no.: 700432-700463)							
220 - 223 (FM)	User range 5, bytes 1 - 4 (event no.: 700500-700531)							
224 - 227 (OM)	User range 5, bytes 5 - 8 (event no.: 700532-700563)							
228 - 231 (FM)	User range 6, bytes 1 - 4 (event no.: 700600-700631)							
232 - 235 (OM)	User range 6, bytes 5 - 8 (event no.: 700632-700663)							
236 - 239 (FM)	User range 7, bytes 1 - 4 (event no.: 700700-700731)							
240 - 243 (OM)	User range 7, bytes 5 - 8 (event no.: 700732-700763)							
244 - 247 (FM)	User range 8, bytes 1 - 4 (event no.: 700800-700831)							
248 - 251 (OM)	User range 8, bytes 5 - 8 (event no.: 700832-700863)							
252 - 255 (FM)	User range 9, bytes 1 - 4 (event no.: 700900-700931)							
256 - 259 (OM)	User range 9, bytes 5 - 8 (event no.: 700932-700963)							
260 - 263 (FM)	User range 10, bytes 1 - 4 (event no.: 701000-701031)							
264 - 267 (OM)	User range 10, bytes 5 - 8 (event no.: 701032-701063)							
268 - 271 (FM)	User range 11, bytes 1 - 4 (event no.: 701100-701131)							
272 - 275 (OM)	User range 11, bytes 5 - 8 (event no.: 701132-701163)							
276 - 279 (FM)	User range 12, bytes 1 - 4 (event no.: 701200-701231)							
280 - 283 (OM)	User range 12, bytes 5 - 8 (event no.: 701232-701263)							
284 - 287 (FM)	User range 13, bytes 1 - 4 (event no.: 701300-701331)							
288 - 291 (OM)	User range 13, bytes 5 - 8 (event no.: 701332-701363)							
292 - 295 (FM)	User range 14, bytes 1 - 4 (event no.: 701400-701431)							

DB2	Signals for PLC events (PLC → HMI) FB1 parameter "ExtendAlMsg" == FALSE							
Byte (message type)	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
296 - 299 (OM)	User range 14, bytes 5 - 8 (event no.: 701432-701463)							
300 - 303 (FM)	User range 15, bytes 1 - 4 (event no.: 701500-701531)							
304 - 307 (OM)	User range 15, bytes 5 - 8 (event no.: 701532-701563)							
308 - 311 (FM)	User range 16, bytes 1 - 4 (event no.: 701600-701631)							
312 - 315 (OM)	User range 16, bytes 5 - 8 (event no.: 701632-701663)							
316 - 319 (FM)	User range 17, bytes 1 - 4 (event no.: 701700-701731)							
320 - 323 (OM)	User range 17, bytes 5 - 8 (event no.: 701732-701763)							
324 - 327 (FM)	User range 18, bytes 1 - 4 (event no.: 701800-701831)							
328 - 331 (OM)	User range 18, bytes 5 - 8 (event no.: 701832-701863)							
332 - 335 (FM)	User range 19, bytes 1 - 4 (event no.: 701900-701931)							
336 - 339 (OM)	User range 19, bytes 5 - 8 (event no.: 701932-701963)							
340 - 343 (FM)	User range 20, bytes 1 - 4 (event no.: 702000-702031)							
344 - 347 (OM)	User range 20, bytes 5 - 8 (event no.: 702032-702063)							
348 - 351 (FM)	User range 21, bytes 1 - 4 (event no.: 702100-702131)							
352 - 355 (OM)	User range 21, bytes 5 - 8 (event no.: 702132-702163)							
356 - 359 (FM)	User range 22, bytes 1 - 4 (event no.: 702200-702231)							
360 - 363 (OM)	User range 22, bytes 5 - 8 (event no.: 702232-702263)							
364 - 367 (FM)	User range 23, bytes 1 - 4 (event no.: 702300-702331)							
368 - 371 (OM)	User range 23, bytes 5 - 8 (event no.: 702332-702363)							
372 - 375 (FM)	User range 24, bytes 1 - 4 (event no.: 702400-702431)							
376 - 379 (OM)	User range 24, bytes 5 - 8 (event no.: 702432-702463)							
380 - 383 (FM)	User range 25, bytes 1 - 4 (event no.: 702500-702531)							
384 - 387 (OM)	User range 25, bytes 5 - 8 (event no.: 702532-702563)							
388 - 389 (FM)	User range 26, bytes 1 - 4 (event no.: 702600-702631)							
390 - 391 (OM)	User range 26, bytes 5 - 8 (event no.: 702632-702663)							
392 - 395 (FM)	User range 27, bytes 1 - 4 (event no.: 702700-702731)							
396 - 403 (OM)	User range 27, bytes 5 - 8 (event no.: 702732-702763)							
404 - 407 (FM)	User range 28, bytes 1 - 4 (event no.: 702800-702831)							
408 - 411 (OM)	User range 28, bytes 5 - 8 (event no.: 702832-702863)							
412 - 415 (FM)	User range 29, bytes 1 - 4 (event no.: 702900-702931)							
416 - 419 (OM)	User range 29, bytes 5 - 8 (event no.: 702932-702963)							
420 - 423 (FM)	User range 30, bytes 1 - 4 (event no.: 703000-703031)							
424 - 427 (OM)	User range 30, bytes 5 - 8 (event no.: 703032-703063)							
428 - 431 (FM)	User range 31, bytes 1 - 4 (event no.: 703100-703131)							
432 - 435 (OM)	User range 31, bytes 5 - 8 (event no.: 703132-703163)							

4.5.2 FC 10 alarms in the DB2 (FB1: "ExtendAlMsg" == TRUE)

Message type

- **FM**: A fault message with the associated event number as fault number is triggered by the signal.
- **OM**: An operating message with the associated event number as message number is triggered by the signal.

References

A detailed description of fault and operating messages can be found in the following manual: Function Manual, Basic Functions; Section "P3: Basic PLC program for SINUMERIK 840D sl", "Block descriptions", "FC10: AL_MSG - fault and operating messages"

Table 4-26 DB2, channel range 1

DB2	Signals for PLC events (PLC → HMI) FB1 parameter "ExtendAlMsg" == TRUE							
Byte (message type)	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
	Signals WITHOUT display of a fault/operating message (DBB0 - 309)							
	Channel 1							
0	Feed disable							
1	Feed disable							
2	Read-in disable							
3	Read-in disable							
4	Start disable							
5	Start disable							
6	Feed stop, geo axis 1, byte 1							
7	Feed stop, geo axis 1, byte 2							
8	Feed stop, geo axis 2, byte 1							
9	Feed stop, geo axis 2, byte 2							
10	Feed stop, geo axis 3, byte 1							
11	Feed stop, geo axis 3, byte 2							
12 - 119	**Channel 2 - channel 10,** see above "Channel 1"							
	Axis/spindle 1							
120	Feed stop / spindle stop, byte 1							
121	Feed stop / spindle stop, byte 2							
122 - 181	**Axis/spindle 2 - 31,** see above "Axis/spindle 1"							
	Additional values for user range 0							
182	Additional value for event number 700000							
184	Additional value for event number 700001							
...	...							
308	Additional value for event number 700063							

DB2	Signals for PLC events (PLC → HMI) FB1 parameter "ExtendAlMsg" == TRUE							
Byte (message type)	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
	Signals WITH display of a fault/operating message (as of DBB 310)							
	Channel 1							
	Feed disable (event no.: 510000-510015)							
310 (FM)	510007	510006	510005	510004	510003	510002	510001	510000
311 (OM)	510015	510014	510013	510012	510011	510010	510009	510008
312 (FM)	Feed and read-in disable, byte 1 (event no.: 510100-510107)							
313 (FM)	Feed and read-in disable, byte 2 (event no.: 510108-510115)							
314 (OM)	Feed and read-in disable, byte 3 (event no.: 510116-510123)							
315 (OM)	Feed and read-in disable, byte 4 (event no.: 510124-510131)							
316 (FM)	Read-in disable, byte 1 (event no.: 510200-510207)							
317 (FM)	Read-in disable, byte 2 (event no.: 510208-510215)							
318 (OM)	Read-in disable, byte 3 (event no.: 510216-510223)							
319 (OM)	Read-in disable, byte 4 (event no.: 510224-510231)							
320 (FM)	NC start disable, byte 1 (event no.: 510300-510307)							
321 (OM)	NC start disable, byte 2 (event no.: 510308-510315)							
322 (FM)	Feed stop, geo axis 1, byte 1 (event no.: 511100-511107)							
323 (OM)	Feed stop, geo axis 1, byte 2 (event no.: 511108-511115)							
324 (FM)	Feed stop, geo axis 2 byte 1 (event no.: 511200-511207)							
325 (OM)	Feed stop, geo axis 2 byte 2 (event no.: 511208-511215)							
326 (FM)	Feed stop, geo axis 3, byte 1 (event no.: 511300-511307)							
327 (OM)	Feed stop, geo axis 3, byte 2 (event no.: 511308-511315)							

Table 4-27 DB2, channel range 2

DB2	Signals for PLC events (PLC → HMI) FB1 parameter "ExtendAlMsg" == TRUE							
Byte (message type)	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
	Channel 2							
	Feed disable (event no.: 510000-520015)							
328 (FM)	520007	520006	520005	520004	520003	520002	520001	520000
329 (OM)	520015	520014	520013	520012	520011	520010	520009	520008
330 (FM)	Feed and read-in disable, byte 1 (event no.: 520100-520107)							
331 (FM)	Feed and read-in disable, byte 2 (event no.: 520108-520115)							
332 (OM)	Feed and read-in disable, byte 3 (event no.: 520116-520123)							
333 (OM)	Feed and read-in disable, byte 4 (event no.: 520124-520131)							
334 (FM)	Read-in disable, byte 1 (event no.: 520200-520207)							
335 (FM)	Read-in disable, byte 2 (event no.: 520208-520215)							

DB2	Signals for PLC events (PLC → HMI) FB1 parameter "ExtendAIMsg" == TRUE						
Byte (message type)	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
336 (OM)	Read-in disable, byte 3 (event no.: 520216-520223)							
337 (OM)	Read-in disable, byte 4 (event no.: 520224-520231)							
338 (FM)	NC start disable, byte 1 (event no.: 520300-520307)							
339 (OM)	NC start disable, byte 2 (event no.: 520308-520315)							
340 (FM)	Feed stop, geo axis 1, byte 1 (event no.: 521100-521107)							
341 (OM)	Feed stop, geo axis 1, byte 2 (event no.: 521108-521115)							
342 (FM)	Feed stop, geo axis 2, byte 1 (event no.: 521200-521207)							
343 (OM)	Feed stop, geo axis 2, byte 2 (event no.: 521208-521215)							
344 (FM)	Feed stop, geo axis 3, byte 1 (event no.: 521300-521307)							
345 (OM)	Feed stop, geo axis 3, byte 2 (event no.: 521308-521315)							

Table 4-28 DB2, channel range 3

DB2	Signals for PLC events (PLC → HMI) FB1 parameter "ExtendAIMsg" == TRUE							
Byte (message type)	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
	Channel 3							
	Feed disable (event no.: 530000-530015)							
346 (FM)	530007	530006	530005	530004	530003	530002	530001	530000
347 (OM)	530015	530014	530013	530012	530011	530010	530009	530008
348 (FM)	Feed and read-in disable, byte 1 (event no.: 530100-530107)							
349 (FM)	Feed and read-in disable, byte 2 (event no.: 530108-530115)							
350 (OM)	Feed and read-in disable, byte 2 (event no.: 530108-530115)							
351 (OM)	Feed and read-in disable, byte 4 (event no.: 530124-530131)							
352 (FM)	Read-in disable, byte 1 (event no.: 530200-530207)							
353 (FM)	Read-in disable, byte 2 (event no.: 530208-530215)							
354 (OM)	Read-in disable, byte 3 (event no.: 530216-530223)							
355 (OM)	Read-in disable, byte 4 (event no.: 530224-530231)							
356 (FM)	NC start disable, byte 1 (event no.: 530300-530307)							
357 (OM)	NC start disable, byte 2 (event no.: 530308-530315)							
358 (FM)	Feed stop, geo axis 1, byte 1 (event no.: 531100-531107)							
359 (OM)	Feed stop, geo axis 1, byte 2 (event no.: 531108-531115)							
360 (FM)	Feed stop, geo axis 2, byte 1 (event no.: 531200-531207)							
361 (OM)	Feed stop, geo axis 2, byte 2 (event no.: 531208-531215)							
362 (FM)	Feed stop, geo axis 3, byte 1 (event no.: 531300-531307)							
363 (OM)	Feed stop, geo axis 3, byte 2 (event no.: 531308-531315)							

Table 4-29 DB2, channel range 4

DB2	Signals for PLC events (PLC → HMI) FB1 parameter "ExtendAIMsg" == TRUE							
Byte (message type)	**Bit 7**	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
	Channel 4							
	Feed disable (event no.: 540000-540015)							
364 (FM)	540007	540006	540005	540004	540003	540002	540001	540000
365 (OM)	540015	540014	540013	540012	540011	540010	540009	540008
366 (FM)	Feed and read-in disable, byte 1 (event no.: 540100-540107)							
367 (FM)	Feed and read-in disable, byte 2 (event no.: 540108-540115)							
368 (OM)	Feed and read-in disable, byte 3 (event no.: 540116-540123)							
369 (OM)	Feed and read-in disable, byte 4 (event no.: 540124-540131)							
370 (FM)	Read-in disable, byte 1 (event no.: 540200-540207)							
371 (FM)	Read-in disable, byte 2 (event no.: 540208-540215)							
372 (OM)	Read-in disable, byte 3 (event no.: 540216-540223)							
373 (OM)	Read-in disable, byte 4 (event no.: 540224-540231)							
374 (FM)	NC start disable, byte 1 (event no.: 540300-540307)							
375 (OM)	NC start disable, byte 2 (event no.: 540308-540315)							
376 (FM)	Feed stop, geo axis 1, byte 1 (event no.: 541100-541107)							
377 (OM)	Feed stop, geo axis 1, byte 2 (event no.: 541108-541115)							
378 (FM)	Feed stop, geo axis 2, byte 1 (event no.: 541200-541207)							
379 (OM)	Feed stop, geo axis 2, byte 2 (event no.: 541208-541215)							
380 (FM)	Feed stop, geo axis 3, byte 1 (event no.: 541300-541307)							
381 (OM)	Feed stop, geo axis 3, byte 2 (event no.: 541308-541315)							

Table 4-30 DB2, channel range 5

DB2	Signals for PLC events (PLC → HMI) FB1 parameter "ExtendAIMsg" == TRUE							
Byte (message type)	**Bit 7**	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
	Channel 5							
	Feed disable (event no.: 550000-550015)							
382 (FM)	550007	550006	550005	550004	550003	550002	550001	550000
383 (OM)	550015	550014	550013	550012	550011	550010	550009	550008
384 (FM)	Feed and read-in disable, byte 1 (event no.: 550100-550107)							
385 (FM)	Feed and read-in disable, byte 2 (event no.: 550108-550115)							
386 (OM)	Feed and read-in disable, byte 3 (event no.: 550116-550123)							
387 (OM)	Feed and read-in disable, byte 4 (event no.: 550124-550131)							
388 (FM)	Read-in disable, byte 1 (event no.: 550200-550207)							

DB2	Signals for PLC events (PLC → HMI) FB1 parameter "ExtendAlMsg" == TRUE						
Byte (message type)	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
389 (FM)	Read-in disable, byte 2 (event no.: 550208-550215)							
390 (OM)	Read-in disable, byte 3 (event no.: 550216-550223)							
391 (OM)	Read-in disable, byte 4 (event no.: 550224-550231)							
392 (FM)	NC start disable, byte 1 (event no.: 550300-550307)							
393 (OM)	NC start disable, byte 2 (event no.: 550308-550315)							
394 (FM)	Feed stop, geo axis 1, byte 1 (event no.: 551100-551107)							
395 (OM)	Feed stop, geo axis 1, byte 2 (event no.: 551108-551115)							
396 (FM)	Feed stop, geo axis 2, byte 1 (event no.: 551200-551207)							
397 (OM)	Feed stop, geo axis 2, byte 2 (event no.: 551208-551215)							
398 (FM)	Feed stop, geo axis 3, byte 1 (event no.: 551300-551307)							
399 (OM)	Feed stop, geo axis 3, byte 2 (event no.: 551308-551315)							

Table 4-31 DB2, channel range 6

DB2	Signals for PLC events (PLC → HMI) FB1 parameter "ExtendAlMsg" == TRUE							
Byte (message type)	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
	Channel 6							
	Feed disable (event no.: 560000-560015)							
400 (FM)	560007	560006	560005	560004	560003	560002	560001	560000
401 (OM)	560015	560014	560013	560012	560011	560010	560009	560008
402 (FM)	Feed and read-in disable, byte 1 (event no.: 560100-560107)							
403 (FM)	Feed and read-in disable, byte 2 (event no.: 560108-560115)							
404 (OM)	Feed and read-in disable, byte 3 (event no.: 560116-560123)							
405 (OM)	Feed and read-in disable, byte 4 (event no.: 560124-560131)							
406 (FM)	Read-in disable, byte 1 (event no.: 560200-560207)							
407 (FM)	Read-in disable, byte 2 (event no.: 560208-560215)							
408 (OM)	Read-in disable, byte 3 (event no.: 560216-560223)							
409 (OM)	Read-in disable, byte 4 (event no.: 560224-560231)							
410 (FM)	NC start disable, byte 1 (event no.: 560300-560307)							
411 (OM)	NC start disable, byte 2 (event no.: 560308-560315)							
412 (FM)	Feed stop, geo axis 1, byte 1 (event no.: 561100-561107)							
413 (OM)	Feed stop, geo axis 1, byte 2 (event no.: 561108-561115)							
414 (FM)	Feed stop, geo axis 2, byte 1 (event no.: 561200-561207)							
415 (OM)	Feed stop, geo axis 2, byte 2 (event no.: 561208-561215)							
416 (FM)	Feed stop, geo axis 3, byte 1 (event no.: 561300-561307)							
417 (OM)	Feed stop, geo axis 3, byte 2 (event no.: 561308-561315)							

Table 4-32 DB2, channel range 7

DB2	Signals for PLC events (PLC → HMI) FB1 parameter "ExtendAlMsg" == TRUE							
Byte (message type)	**Bit 7**	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
	Channel 7							
	Feed disable (event no.: 570000-570015)							
418 (FM)	570007	570006	570005	570004	570003	570002	570001	570000
419 (OM)	570015	570014	570013	570012	570011	570010	570009	570008
420 (FM)	Feed and read-in disable, byte 1 (event no.: 570100-570107)							
421 (FM)	Feed and read-in disable, byte 2 (event no.: 570108-570115)							
422 (OM)	Feed and read-in disable, byte 3 (event no.: 570116-570123)							
423 (OM)	Feed and read-in disable, byte 4 (event no.: 570124-570131)							
424 (FM)	Read-in disable, byte 1 (event no.: 570200-570207)							
425 (FM)	Read-in disable, byte 2 (event no.: 570208-570215)							
426 (OM)	Read-in disable, byte 3 (event no.: 570216-570223)							
427 (OM)	Read-in disable, byte 4 (event no.: 570224-570231)							
428 (FM)	NC start disable, byte 1 (event no.: 570300-570307)							
429 (OM)	NC start disable, byte 2 (event no.: 570308-570315)							
430 (FM)	Feed stop, geo axis 1, byte 1 (event no.: 571100-571107)							
431 (OM)	Feed stop, geo axis 1, byte 2 (event no.: 571108-571115)							
432 (FM)	Feed stop, geo axis 2, byte 1 (event no.: 571200-571207)							
433 (OM)	Feed stop, geo axis 2, byte 2 (event no.: 571208-571215)							
434 (FM)	Feed stop, geo axis 3, byte 1 (event no.: 571300-571307)							
435 (OM)	Feed stop, geo axis 3, byte 2 (event no.: 571308-571315)							

Table 4-33 DB2, channel range 8

DB2	Signals for PLC events (PLC → HMI) FB1 parameter "ExtendAlMsg" == TRUE							
Byte (message type)	**Bit 7**	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
	Channel 8							
	Feed disable (event no.: 580000-580015)							
436 (FM)	580007	580006	580005	580004	580003	580002	580001	580000
437 (OM)	580015	580014	580013	580012	580011	580010	580009	580008
438 (FM)	Feed and read-in disable, byte 1 (event no.: 580100-580107)							
439 (FM)	Feed and read-in disable, byte 2 (event no.: 580108-580115)							
440 (OM)	Feed and read-in disable, byte 3 (event no.: 580116-580123)							
441 (OM)	Feed and read-in disable, byte 4 (event no.: 580124-580131)							
442 (FM)	Read-in disable, byte 1 (event no.: 580200-580207)							

DB2	Signals for PLC events (PLC → HMI) FB1 parameter "ExtendAlMsg" == TRUE							
Byte (message type)	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
443 (FM)	Read-in disable, byte 2 (event no.: 580208-580215)							
444 (OM)	Read-in disable, byte 3 (event no.: 580216-580223)							
445 (OM)	Read-in disable, byte 4 (event no.: 580224-580231)							
446 (FM)	NC start disable, byte 1 (event no.: 580300-580307)							
447 (OM)	NC start disable, byte 2 (event no.: 580308-580315)							
448 (FM)	Feed stop, geo axis 1, byte 1 (event no.: 581100-581107)							
449 (OM)	Feed stop, geo axis 1, byte 2 (event no.: 581108-581115)							
450 (FM)	Feed stop, geo axis 2, byte 1 (event no.: 581200-581207)							
451 (OM)	Feed stop, geo axis 2, byte 2 (event no.: 581208-581215)							
452 (FM)	Feed stop, geo axis 3, byte 1 (event no.: 581300-581307)							
453 (OM)	Feed stop, geo axis 3, byte 2 (event no.: 581308-581315)							

Table 4-34 DB2, channel range 9

DB2	Signals for PLC events (PLC → HMI) FB1 parameter "ExtendAlMsg" == TRUE							
Byte (message type)	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
	Channel 9							
	Feed disable (event no.: 590000-590015)							
454 (FM)	590007	590006	590005	590004	590003	590002	590001	590000
455 (OM)	590015	590014	590013	590012	590011	590010	590009	590008
456 (FM)	Feed and read-in disable, byte 1 (event no.: 590100-590107)							
457 (FM)	Feed and read-in disable, byte 2 (event no.: 590108-590115)							
458 (OM)	Feed and read-in disable, byte 3 (event no.: 590116-590123)							
459 (OM)	Feed and read-in disable, byte 4 (event no.: 590124-590131)							
460 (FM)	Read-in disable, byte 1 (event no.: 590200-590207)							
461 (FM)	Read-in disable, byte 2 (event no.: 590208-590215)							
462 (OM)	Read-in disable, byte 3 (event no.: 590216-590223)							
463 (OM)	Read-in disable, byte 4 (event no.: 590224-590231)							
464 (FM)	NC start disable, byte 1 (event no.: 590300-590307)							
465 (OM)	NC start disable, byte 2 (event no.: 590308-590315)							
466 (FM)	Feed stop, geo axis 1, byte 1 (event no.: 591100-591107)							
467 (OM)	Feed stop, geo axis 1, byte 2 (event no.: 591108-591115)							
468 (FM)	Feed stop, geo axis 2, byte 1 (event no.: 591200-591207)							
469 (OM)	Feed stop, geo axis 2, byte 2 (event no.: 591208-591215)							
470 (FM)	Feed stop, geo axis 3, byte 1 (event no.: 591300-591307)							
471 (OM)	Feed stop, geo axis 3, byte 2 (event no.: 591308-591315)							

Table 4-35 DB2, channel range 10

DB2	Signals for PLC events (PLC → HMI) FB1 parameter "ExtendAIMsg" == TRUE							
Byte (message type)	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
	Channel 10							
	Feed disable (event no.: 500000-500015)							
472 (FM)	500007	500006	500005	500004	500003	500002	500001	500000
473 (OM)	500015	500014	500013	500012	500011	500010	500009	500008
474 (FM)	Feed and read-in disable, byte 1 (event no.: 500100-500107)							
475 (FM)	Feed and read-in disable, byte 2 (event no.: 500108-500115)							
476 (OM)	Feed and read-in disable, byte 3 (event no.: 500116-500123)							
477 (OM)	Feed and read-in disable, byte 4 (event no.: 500124-500131)							
478 (FM)	Read-in disable, byte 1 (event no.: 500200-500207)							
479 (FM)	Read-in disable, byte 2 (event no.: 500208-500215)							
480 (OM)	Read-in disable, byte 3 (event no.: 500216-500223)							
481 (OM)	Read-in disable, byte 4 (event no.: 500224-500231)							
482 (FM)	NC start disable, byte 1 (event no.: 500300-500307)							
483 (OM)	NC start disable, byte 2 (event no.: 500308-500315)							
484 (FM)	Feed stop, geo axis 1, byte 1 (event no.: 501100-501107)							
485 (OM)	Feed stop, geo axis 1, byte 2 (event no.: 501108-501115)							
486 (FM)	Feed stop, geo axis 2, byte 1 (event no.: 501200-501207)							
487 (OM)	Feed stop, geo axis 2, byte 2 (event no.: 501208-501215)							
488 (FM)	Feed stop, geo axis 3, byte 1 (event no.: 501300-501307)							
489 (OM)	Feed stop, geo axis 3, byte 2 (event no.: 501308-501315)							

Table 4-36 DB2, axis ranges

DB2	Signals for PLC events (PLC → HMI) FB1 parameter "ExtendAIMsg" == TRUE							
Byte (message type)	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
	Axis/spindle							
	Feed stop / spindle stop for axis/spindle 1 (event no.: 600100-600015)							
490 (FM)	600107	600106	600105	600104	600103	600102	600101	600100
491 (OM)	600115	600114	600113	600112	600111	600110	600109	600108
492 (FM)	Feed stop / spindle stop for axis/spindle 2 (event no.: 600200-600207)							
493 (OM)	Feed stop / spindle stop for axis/spindle 2 (event no.: 600208-600215)							
494 (FM)	Feed stop / spindle stop for axis/spindle 3 (event no.: 600300-600307)							
495 (OM)	Feed stop / spindle stop for axis/spindle 3 (event no.: 600308-600315)							
496 (FM)	Feed stop / spindle stop for axis/spindle 4 (event no.: 600400-600407)							

DB2	Signals for PLC events (PLC → HMI) FB1 parameter "ExtendAIMsg" == TRUE							
Byte (message type)	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
497 (OM)	Feed stop / spindle stop for axis/spindle 4 (event no.: 600408-600415)							
498 (FM)	Feed stop / spindle stop for axis/spindle 5 (event no.: 600500-600507)							
499 (OM)	Feed stop / spindle stop for axis/spindle 5 (event no.: 600508-600515)							
500 (FM)	Feed stop / spindle stop for axis/spindle 6 (event no.: 600600-600607)							
501 (OM)	Feed stop / spindle stop for axis/spindle 6 (event no.: 600608-600615)							
502 (FM)	Feed stop / spindle stop for axis/spindle 7 (event no.: 600700-600707)							
503 (OM)	Feed stop / spindle stop for axis/spindle 7 (event no.: 600708-600715)							
504 (FM)	Feed stop / spindle stop for axis/spindle 8 (event no.: 600800-600807)							
505 (OM)	Feed stop / spindle stop for axis/spindle 8 (event no.: 600808-600815)							
506 (FM)	Feed stop / spindle stop for axis/spindle 9 (event no.: 600900-600907)							
507 (OM)	Feed stop / spindle stop for axis/spindle 9 (event no.: 600908-600915)							
508 (FM)	Feed stop / spindle stop for axis/spindle 10 (event no.: 601000-601007)							
509 (OM)	Feed stop / spindle stop for axis/spindle 10 (event no.: 601008-601015)							
510 (FM)	Feed stop / spindle stop for axis/spindle 11 (event no.: 601100-601107)							
511 (OM)	Feed stop / spindle stop for axis/spindle 11 (event no.: 601108-601115)							
512 (FM)	Feed stop / spindle stop for axis/spindle 12 (event no.: 601200-601207)							
513 (OM)	Feed stop / spindle stop for axis/spindle 12 (event no.: 601208-601215)							
514 (FM)	Feed stop / spindle stop for axis/spindle 13 (event no.: 601300-601307)							
515 (OM)	Feed stop / spindle stop for axis/spindle 13 (event no.: 601308-601315)							
516 (FM)	Feed stop / spindle stop for axis/spindle 14 (event no.: 601400-601407)							
517 (OM)	Feed stop / spindle stop for axis/spindle 14 (event no.: 601408-601415)							
518 (FM)	Feed stop / spindle stop for axis/spindle 15 (event no.: 601500-601507)							
519 (OM)	Feed stop / spindle stop for axis/spindle 15 (event no.: 601508-601515)							
520 (FM)	Feed stop / spindle stop for axis/spindle 16 (event no.: 601600-601607)							
521 (OM)	Feed stop / spindle stop for axis/spindle 16 (event no.: 601608-601615)							
522 (FM)	Feed stop / spindle stop for axis/spindle 17 (event no.: 601700-601707)							
523 (OM)	Feed stop / spindle stop for axis/spindle 17 (event no.: 601708-601715)							
524 (FM)	Feed stop / spindle stop for axis/spindle 18 (event no.: 601800-601807)							
525 (OM)	Feed stop / spindle stop for axis/spindle 18 (event no.: 601808-601815)							
526 (FM)	Feed stop / spindle stop for axis/spindle 19 (event no.: 601900-601907)							
527 (OM)	Feed stop / spindle stop for axis/spindle 19 (event no.: 601908-601915)							
528 (FM)	Feed stop / spindle stop for axis/spindle 20 (event no.: 602000-602007)							
529 (OM)	Feed stop / spindle stop for axis/spindle 20 (event no.: 602008-602015)							
530 (FM)	Feed stop / spindle stop for axis/spindle 21 (event no.: 602100-602107)							
531 (OM)	Feed stop / spindle stop for axis/spindle 21 (event no.: 602108-602115)							
532 (FM)	Feed stop / spindle stop for axis/spindle 22 (event no.: 602200-602207)							
533 (OM)	Feed stop / spindle stop for axis/spindle 22 (event no.: 602208-602215)							
534 (FM)	Feed stop / spindle stop for axis/spindle 23 (event no.: 602300-602307)							
535 (OM)	Feed stop / spindle stop for axis/spindle 23 (event no.: 602308-602315)							

DB2	Signals for PLC events (PLC → HMI) FB1 parameter "ExtendAIMsg" == TRUE							
Byte (message type)	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
536 (FM)	Feed stop / spindle stop for axis/spindle 24 (event no.: 602400-602407)							
537 (OM)	Feed stop / spindle stop for axis/spindle 24 (event no.: 602408-602415)							
538 (FM)	Feed stop / spindle stop for axis/spindle 25 (event no.: 602500-602507)							
539 (OM)	Feed stop / spindle stop for axis/spindle 25 (event no.: 602508-602515)							
540 (FM)	Feed stop / spindle stop for axis/spindle 26 (event no.: 602600-602607)							
541 (OM)	Feed stop / spindle stop for axis/spindle 26 (event no.: 602608-602615)							
542 (FM)	Feed stop / spindle stop for axis/spindle 27 (event no.: 602700-602707)							
543 (OM)	Feed stop / spindle stop for axis/spindle 27 (event no.: 602708-602715)							
544 (FM)	Feed stop / spindle stop for axis/spindle 28 (event no.: 602800-602807)							
545 (OM)	Feed stop / spindle stop for axis/spindle 28 (event no.: 602808-602815)							
546 (FM)	Feed stop / spindle stop for axis/spindle 29 (event no.: 602900-602907)							
547 (OM)	Feed stop / spindle stop for axis/spindle 29 (event no.: 602908-602915)							
548 (FM)	Feed stop / spindle stop for axis/spindle 30 (event no.: 603000-603007)							
549 (OM)	Feed stop / spindle stop for axis/spindle 30 (event no.: 603008-603015)							
550 (FM)	Feed stop / spindle stop for axis/spindle 31 (event no.: 603100-603107)							
551 (OM)	Feed stop / spindle stop for axis/spindle 31 (event no.: 603108-603115)							

Table 4-37 DB2, user ranges

DB2	Signals for PLC events (PLC → HMI) FB1 parameter "ExtendAIMsg" == TRUE							
Byte (message type)	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
	User ranges							
	User range 0 (event no.: 700000-700015)							
554 (FM)	700007	700006	700005	700004	700003	700002	700001	700000
555 (FM)	700015	700014	700013	700012	700011	700010	700009	700008
556 (FM)	User range 0, byte 3 (event no.: 700016-700023)							
557 (FM)	User range 0, byte 4 (event no.: 700024-700031)							
558 (OM)	User range 0, byte 5 (event no.: 700032-700039)							
559 (OM)	User range 0, byte 6 (event no.: 700040-700047)							
560 (OM)	User range 0, byte 7 (event no.: 700048-700055)							
561 (OM)	User range 0, byte 8 (event no.: 700056-700063)							
562 - 565 (FM)	User range 1, bytes 1 - 4 (event no.: 700100-700131)							
566 - 569 (OM)	User range 1, bytes 5 - 8 (event no.: 700132-700163)							
570 - 573 (FM)	User range 2, bytes 1 - 4 (event no.: 700200-700231)							
574 - 577 (OM)	User range 2, bytes 5 - 8 (event no.: 700232-700263)							
578 - 581 (FM)	User range 3, bytes 1 - 4 (event no.: 700300-700331)							
582 - 585 (OM)	User range 3, bytes 5 - 8 (event no.: 700332-700363)							

DB2	Signals for PLC events (PLC → HMI) FB1 parameter "ExtendAlMsg" == TRUE						
Byte (message type)	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
586 - 589 (FM)	User range 4, bytes 1 - 4 (event no.: 700400-700431)							
590 - 593 (OM)	User range 4, bytes 5 - 8 (event no.: 700432-700463)							
594 - 597 (FM)	User range 5, bytes 1 - 4 (event no.: 700500-700531)							
598 - 601 (OM)	User range 5, bytes 5 - 8 (event no.: 700532-700563)							
602 - 605 (FM)	User range 6, bytes 1 - 4 (event no.: 700600-700631)							
606 - 609 (OM)	User range 6, bytes 5 - 8 (event no.: 700632-700663)							
610 - 613 (FM)	User range 7, bytes 1 - 4 (event no.: 700700-700731)							
614 - 617 (OM)	User range 7, bytes 5 - 8 (event no.: 700732-700763)							
618 - 621 (FM)	User range 8, bytes 1 - 4 (event no.: 700800-700831)							
622 - 625 (OM)	User range 8, bytes 5 - 8 (event no.: 700832-700863)							
626 - 629 (FM)	User range 9, bytes 1 - 4 (event no.: 700900-700931)							
630 - 633 (OM)	User range 9, bytes 5 - 8 (event no.: 700932-700963)							
634 - 637 (FM)	User range 10, bytes 1 - 4 (event no.: 701000-701031)							
638 - 641 (OM)	User range 10, bytes 5 - 8 (event no.: 701032-701063)							
642 - 645 (FM)	User range 11, bytes 1 - 4 (event no.: 701100-701131)							
646 - 649 (OM)	User range 11, bytes 5 - 8 (event no.: 701132-701163)							
650 - 653 (FM)	User range 12, bytes 1 - 4 (event no.: 701200-701231)							
654 - 657 (OM)	User range 12, bytes 5 - 8 (event no.: 701232-701263)							
658 - 661 (FM)	User range 13, bytes 1 - 4 (event no.: 701300-701331)							
662 - 665 (OM)	User range 13, bytes 5 - 8 (event no.: 701332-701363)							
666 - 669 (FM)	User range 14, bytes 1 - 4 (event no.: 701400-701431)							
670 - 673 (OM)	User range 14, bytes 5 - 8 (event no.: 701432-701463)							
674 - 677 (FM)	User range 15, bytes 1 - 4 (event no.: 701500-701531)							
678 - 681 (OM)	User range 15, bytes 5 - 8 (event no.: 701532-701563)							
682 - 685 (FM)	User range 16, bytes 1 - 4 (event no.: 701600-701631)							
686 - 689 (OM)	User range 16, bytes 5 - 8 (event no.: 701632-701663)							
690 - 693 (FM)	User range 17, bytes 1 - 4 (event no.: 701700-701731)							
694 - 697 (OM)	User range 17, bytes 5 - 8 (event no.: 701732-701763)							
698 - 701 (FM)	User range 18, bytes 1 - 4 (event no.: 701800-701831)							
702 - 705 (OM)	User range 18, bytes 5 - 8 (event no.: 701832-701863)							
706 - 709 (FM)	User range 19, bytes 1 - 4 (event no.: 701900-701931)							
710 - 713 (OM)	User range 19, bytes 5 - 8 (event no.: 701932-701963)							
714 - 717 (FM)	User range 20, bytes 1 - 4 (event no.: 702000-702031)							
718 - 721 (OM)	User range 20, bytes 5 - 8 (event no.: 702032-702063)							
722 - 725 (FM)	User range 21, bytes 1 - 4 (event no.: 702100-702131)							
726 - 729 (OM)	User range 21, bytes 5 - 8 (event no.: 702132-702163)							
730 – 733 (FM)	User range 22, bytes 1 - 4 (event no.: 702200-702231)							
734 - 737 (OM)	User range 22, bytes 5 - 8 (event no.: 702232-702263)							
738 - 741 (FM)	User range 23, bytes 1 - 4 (event no.: 702300-702331)							
742 - 745 (OM)	User range 23, bytes 5 - 8 (event no.: 702332-702363)							

DB2	Signals for PLC events (PLC → HMI) FB1 parameter "ExtendAIMsg" == TRUE							
Byte (message type)	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
746 - 749 (FM)	User range 24, bytes 1 - 4 (event no.: 702400-702431)							
750 - 753 (OM)	User range 24, bytes 5 - 8 (event no.: 702432-702463)							
754 - 757 (FM)	User range 25, bytes 1 - 4 (event no.: 702500-702531)							
758 - 761 (OM)	User range 25, bytes 5 - 8 (event no.: 702532-702563)							
762 - 765 (FM)	User range 26, bytes 1 - 4 (event no.: 702600-702631)							
766 - 769 (OM)	User range 26, bytes 5 - 8 (event no.: 702632-702663)							
770 - 773 (FM)	User range 27, bytes 1 - 4 (event no.: 702700-702731)							
774 - 777 (OM)	User range 27, bytes 5 - 8 (event no.: 702732-702763)							
778 - 781 (FM)	User range 28, bytes 1 - 4 (event no.: 702800-702831)							
782 - 785 (OM)	User range 28, bytes 5 - 8 (event no.: 702832-702863)							
786 - 789 (FM)	User range 29, bytes 1 - 4 (event no.: 702900-702931)							
790 - 793 (OM)	User range 29, bytes 5 - 8 (event no.: 702932-702963)							
794 - 797 (FM)	User range 30, bytes 1 - 4 (event no.: 703000-703031)							
798 - 801 (OM)	User range 30, bytes 5 - 8 (event no.: 703032-703063)							
802 - 805 (FM)	User range 31, bytes 1 - 4 (event no.: 703100-703131)							
806 - 809 (OM)	User range 31, bytes 5 - 8 (event no.: 703132-703163)							

4.6 Signals from/to the NC, PLC and operating software

4.6.1 DB10, onboard inputs and outputs of the NC

Table 4-38 DB10, onboard inputs and outputs of the NC

DB10	Signals to the NC (PLC → NC)							
Byte	**Bit 7**	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
DBB0	Disable of the digital NC inputs /Z1-A2/							
	Input without hardware				Input onboard			
	8	7	6	5	4	3	2	1
DBB1	Setting of the digital NC inputs from the PLC							
	Input without hardware				Input onboard			
	8	7	6	5	4	3	2	1
DBB2 - DBB3	Not assigned							
DBB4	Disable of the digital NC outputs /Z2-A4/							
	Output without hardware				Output onboard			
	8	7	6	5	4	3	2	1
DBB5	Overwrite mask of the digital NC outputs /Z2-A4/							
	Output without hardware				Output onboard			
	8	7	6	5	4	3	2	1
DBB6	Setting value of the digital NC outputs from the PLC /Z2-A4/							
	Output without hardware				Output onboard			
	8	7	6	5	4	3	2	1
DBB7	Input mask of the digital NC outputs /Z2-A4/							
	Output without hardware				Output onboard			
	8	7	6	5	4	3	2	1
DBB8 - DBB29	Machine axis numbers table for FC 19, FC 24, FC 25, FC 26 (1st MCP)							
DBW30	Upper limit of the machine axis numbers for FC 19, FC 24 (1st MCP) With 0, the maximum number of machine axis numbers applies							
DBB32 - DBB53	Machine axis numbers table for FC 19, FB 24, FB 25, FB 26 (2nd MCP)							
DBW54	Upper limit of the machine axis numbers for FC 19, FC 24 (2nd MCP) With 0, the maximum number of machine axis numbers applies							

4.6.2 DB10, general signals to the NC

Table 4-39 DB10, general signals to the NC

DB10	Signals to the NC (PLC → NC)							
Byte	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
DBB56	Key-operated switch position /Z1-A2/					Acknowledge emergency stop /Z1-A2/	Emergency stop /Z1-A2/	
	3	2	1	0				
DBB57					Reserved			INC inputs in the mode group range active
DBB58	Collision avoidance: Deactivate protection area group							
	JOG				AUTO			
	Workpieces	Workholder	Tools	Machine	Workpieces	Workholder	Tools	Tools
DBB59								

4.6.3 DB10, onboard inputs and outputs from the NC/operating software

Table 4-40 DB10, onboard inputs and outputs from the NC/operating software

DB10	Signals from the NC (NC → PLC)							
Byte	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
DBB60	Freely available				Actual value of the digital onboard inputs of the NC /Z2-A4/			
					4	3	2	1
DBB61 - DBB63								
DBB64	Setpoint for the digital outputs of the NC without hardware /Z2-A4/				Setpoint for the digital onboard outputs of the NC /Z2-A4/			
	8	7	6	5	4	3	2	1
DBB65 - DBB67								
DBB68	Handwheel 1 is moved /FB2/H1/							
DBB69	Handwheel 2 is moved /FB2/H1/							
DBB70	Handwheel 3 is moved /FB2/H1/							

DB10	Signals from the NC (NC → PLC)							
Byte	**Bit 7**	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
DBB71	Change counter, inch/metric measuring system							
DBB72 HT 8 → operating software	Status of the displayed actual value screen (1st MCP)							
	Traversing keys shown						MCS/WCS	Display valid
DBB73 HT 8 → operating software	Status of the displayed actual value screen (2nd MCP)							
	Traversing keys shown						MCS/WCS	Display valid
DBB74 - DBB79 HT 8 → operating software	Machine axis numbers of the displayed axes (1st MCP) MSTT1AxisFromHMI							
DBB80 - DBB85 HT 8 → operating software	Machine axis numbers of the displayed axes (2nd MCP) MSTT2AxisFromHMI							
DBW86	Reserved							
DBB88	Reserved							

4.6.4 DB10, selection and status signals from the operating software

Table 4-41 DB10, selection and status signals from the operating software

DB10	Signals from the NC (NC → PLC)							
Byte	**Bit 7**	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
DBB90 ePS → PLC								
DBB91 PLC → SINUMERIK Integrate								
DBB92 GP → PLC		Suppress fault message in case of failure				Slave OK		
		PN bus	DP1 bus	MPI/DP bus		PN bus	DP1 bus	MPI/DP bus

DB10	Signals from the NC (NC → PLC)							
Byte	**Bit 7**	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
DBB93	Disable collision avoidance							
Operating soft- ware → PL C	JOG				AUTO			
	Workpieces	Workholder	Tools	Machine	Workpieces	Workholder	Tools	Machine
DBB94	Not assigned							
DBB95	Not assigned							
DBB96	Set language ID of the operating software							
Operating soft- ware → PL C								
DBB97					Channel number for handwheel 1 /Z2-H1/			
Operating soft- ware → PL C					D	C	B	A
DBB98					Channel number for handwheel 2 /Z2-H1/			
Operating soft- ware → PL C					D	C	B	A
DBB99					Channel number for handwheel 3 /Z2-H1/			
Operating soft- ware → PL C					D	C	B	A
DBB100				Axis number for handwheel 1 /Z2-H1/				
Operating soft- ware → PL C	Machine axis /Z2-H1/	Handwheel 1 selected /Z2-H1/	Define handwheel 1 as con- tour hand- wheel /Z2-H1/	E	D	C	B	A
DBB101				Axis number for handwheel 2 /Z2-H1/				
Operating soft- ware → PL C	Machine axis /Z2-H1/	Handwheel 2 selected /Z2-H1/	Define handwheel 2 as con- tour hand- wheel /Z2-H1/	E	D	C	B	A

DB10	Signals from the NC (NC → PLC)							
Byte	**Bit 7**	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
DBB102				Axis number for handwheel 3 /Z2-H1/				
Operating software → PLC	Machine axis /Z2-H1/	Handwheel 3 selected /Z2-H1/	Define handwheel 3 as contour handwheel /Z2-H1/	E	D	C	B	A
DBB103 Operating software → PLC	Operating software battery alarm /Z1-A2/	Operating software temperature limit /Z1-A2/	AT box ready /Z1-A2/	Operating software fan monitoring	Operating software monitor hard disk			Remote diagnostics active /Z1-A2/

4.6.5 DB10, general signals from the NC

Table 4-42 DB10, general signals from the NC

DB10	Signals from the NC (NC → PLC)							
Byte	**Bit 7**	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
DBB104 GP → PLC	NC CPU ready /Z1-A2/	1. OB1 cycle		Op2Key ready	Op1Key ready	HHU ready	MCP 2 ready	MCP 1 ready
DBB105 GP → PLC								TM Command Cancel
DBB106 NC → PLC								Collision avoidance Protection area monitoring active
DBB107	Inch measuring system /Z1-G2/	NCU link active /Z2-B3/					Probe actuated /Z2-M5/	
							Probe 2	Probe 1
DBB108	NC ready /Z1-A2/	Drive ready /Z1-A2/	Drives in cyclic operation /Z1-A2/		Operator panel at OPI: "ready" /Z1-A2/ /-B3/	Operator panel at MPI: "ready" /Z1-A2/ /-B3/	Operator panel 2: "ready" /Z1-A2/ /-B3/	

DB10	Signals from the NC (NC → PLC)							
Byte	**Bit 7**	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
DBB109	NC battery alarm /Z1-A2/	Air temperature alarm /Z1-A2/	NCU heat sink temp. alarm /Z1-A2/	PC operating system error				NC alarm is active /Z1-A2/
DBB110	Software cam minus /Z2-N3/							
	7	8	5	4	3	2	1	0
DBB111	Software cam minus /Z2-N3/							
	15	14	13	12	11	10	9	8
DBB112	Software cam minus /Z2-N3/							
	23	22	21	20	19	18	17	16
DBB113	Software cam minus /Z2-N3/							
	31	30	29	28	27	26	25	24
DBB114	Software cam plus /Z2-N3/							
	7	6	5	4	3	2	1	0
DBB115	Software cam plus /Z2-N3/							
	15	14	13	12	11	10	9	8
DBB116	Software cam plus /Z2-N3/							
	23	22	21	20	19	18	17	16
DBB117	Software cam plus /Z2-N3/							
	31	30	29	28	27	26	25	24
DBB118-DBB121 SINUMERIK Integrate → PLC	SINUMERIK Integrate data							

Note

DBX104.7 (NC CPU ready, sign-of-life of the NC)

The signal should be included in the machine safety circuit.

4.6.6 DB10, external digital NC inputs

Table 4-43 DB10, external digital NC inputs

DB10	Signals to the NC (PLC → NC)							
Byte	**Bit 7**	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
DBB122	Disable of the external digital NC inputs							
	16	15	14	13	12	11	10	9
DBB123	Values from the PLC for the external digital NC inputs							
	16	15	14	13	12	11	10	9

DB10	Signals to the NC (PLC → NC)							
Byte	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
DBB124	Disable of the external digital NC inputs							
	24	23	22	21	20	19	18	17
DBB125	Values from the PLC for the external digital NC inputs							
	24	23	22	21	20	19	18	17
DBB126	Disable of the external digital NC inputs							
	32	31	30	29	28	27	26	25
DBB127	Values from the PLC for the external digital NC inputs							
	32	31	30	29	28	27	26	25
DBB128	Disable of the external digital NC inputs							
	40	39	38	37	36	35	34	33
DBB129	Values from the PLC for the external digital NC inputs							
	40	39	38	37	36	35	34	33

4.6.7 DB10, external digital NC outputs

Table 4-44 DB10, external digital NC outputs

DB10	Signals to the NC (PLC → NC) /Z2-A4/							
Byte	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
DBB130	Disable of the external digital NC outputs							
	16	15	14	13	12	11	10	9
DBB131	Overwrite mask for the external digital NC outputs							
	16	15	14	13	12	11	10	9
DBB132	Value from the PLC for the external digital NC outputs							
	16	15	14	13	12	11	10	9
DBB133	Input mask for the external digital NC outputs							
	16	15	14	13	12	11	10	9
DBB134	Disable of the external digital NC outputs							
	24	23	22	21	20	19	18	17
DBB135	Overwrite mask for the external digital NC outputs							
	24	23	22	21	20	19	18	17
DBB136	Value from the PLC for the external digital NC outputs							
	24	23	22	21	20	19	18	17
DBB137	Input mask for the external digital NC outputs							
	24	23	22	21	20	19	18	17
DBB138	Disable of the external digital NC outputs							
	32	31	30	29	28	27	26	25
DBB139	Overwrite mask for the external digital NC outputs							
	32	31	30	29	28	27	26	25

DB10	Signals to the NC (PLC → NC) /Z2-A4/							
Byte	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
DBB140	Value from the PLC for the external digital NC outputs							
	32	31	30	29	28	27	26	25
DBB141	Input mask for the external digital NC outputs							
	32	31	30	29	28	27	26	25
DBB142	Disable of the external digital NC outputs							
	40	39	38	37	36	35	34	33
DBB143	Overwrite mask for the external digital NC outputs							
	40	39	38	37	36	35	34	33
DBB144	Value from the PLC for the external digital NC outputs							
	40	39	38	37	36	35	34	33
DBB145	Input mask for the external digital NC outputs							
	40	39	38	37	36	35	34	33

4.6.8 DB10, external analog NC inputs

Table 4-45 DB10, external analog NC inputs

DB10	Signals to the NC (PLC → NC) /Z2-A4/							
Byte	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
DBB146	Disable analog NC inputs							
	8	7	6	5	4	3	2	1
DBB147	Input analog value default for the NC from the PLC							
	8	7	6	5	4	3	2	1
DBW148	Setpoint from the PLC for analog input 1 of the NC							
DBW150	Setpoint from the PLC for analog input 2 of the NC							
DBW152	Setpoint from the PLC for analog input 3 of the NC							
DBW154	Setpoint from the PLC for analog input 4 of the NC							
DBW156	Setpoint from the PLC for analog input 5 of the NC							
DBW158	Setpoint from the PLC for analog input 6 of the NC							
DBW160	Setpoint from the PLC for analog input 7 of the NC							
DBW162	Setpoint from the PLC for analog input 8 of the NC							
DBB164 - DBB165	Not assigned							

4.6.9 DB10, external analog NC outputs

Table 4-46 DB10, external analog NC outputs

DB10	Signals to the NC (PLC → NC) /Z2-A4/							
Byte	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
DBB166	Overwrite mask for the analog NC outputs							
	8	7	6	5	4	3	2	1
DBB167	Input mask for the analog NC outputs							
	8	7	6	5	4	3	2	1
DBB168	Disable of the analog NC outputs							
	8	7	6	5	4	3	2	1
DBB169	Reserved							
DBW170	Setpoint from the PLC for analog output 1 of the NC							
DBW172	Setpoint from the PLC for analog output 2 of the NC							
DBW174	Setpoint from the PLC for analog output 3 of the NC							
DBW176	Setpoint from the PLC for analog output 4 of the NC							
DBW178	Setpoint from the PLC for analog output 5 of the NC							
DBW180	Setpoint from the PLC for analog output 6 of the NC							
DBW182	Setpoint from the PLC for analog output 7 of the NC							
DBW184	Setpoint from the PLC for analog output 8 of the NC							

4.6.10 DB10, external digital NC inputs and outputs

Table 4-47 DB10, external digital NC inputs and outputs

DB10	Signals from the NC (NC → PLC) /Z2-A4/							
Byte	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
DBB186	Actual value of the external digital NC inputs							
	16	15	14	13	12	11	10	9
DBB187	Actual value of the external digital NC inputs							
	24	23	22	21	20	19	18	17
DBB188	Actual value of the external digital NC inputs							
	32	31	30	29	28	27	26	25
DBB189	Actual value of the external digital NC inputs							
	40	39	38	37	36	35	34	33
DBB190	NC setpoint for the external digital NC outputs							
	16	15	14	13	12	11	10	9
DBB191	NC setpoint for the external digital NC outputs							
	24	23	22	21	20	19	18	17

DB10	Signals from the NC (NC → PLC) /Z2-A4/							
Byte	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
DBB192	NC setpoint for the external digital NC outputs							
	32	31	30	29	28	27	26	25
DBB193	NC setpoint for the external digital NC outputs							
	40	39	38	37	36	35	34	33

4.6.11 DB10, analog NC inputs and outputs

Table 4-48 DB10, analog NC inputs and outputs

DB10	Signals from the NC (NC → PLC) /Z2-A4/							
Byte	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
DBW194	Actual value of analog input 1 of the NC							
DBW196	Actual value of analog input 2 of the NC							
DBW198	Actual value of analog input 3 of the NC							
DBW200	Actual value of analog input 4 of the NC							
DBW202	Actual value of analog input 5 of the NC							
DBW204	Actual value of analog input 6 of the NC							
DBW206	Actual value of analog input 7 of the NC							
DBW208	Actual value of analog input 8 of the NC							
DBW210	Setpoint of analog output 1 of the NC							
DBW212	Setpoint of analog output 2 of the NC							
DBW214	Setpoint of analog output 3 of the NC							
DBW216	Setpoint of analog output 4 of the NC							
DBW218	Setpoint of analog output 5 of the NC							
DBW220	Setpoint of analog output 6 of the NC							
DBW222	Setpoint of analog output 7 of the NC							
DBW224	Setpoint of analog output 8 of the NC							

4.6.12 DB10, collision avoidance: Protection area active

Table 4-49 DB10, collision avoidance: Protection area active

DB10	Signals from the NC (NC → PLC)							
Byte	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
DBB226	Collision avoidance: Protection area active (bit)							
	7	6	5	4	3	2	1	0

DB10	Signals from the NC (NC → PLC)							
Byte	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
DBB227	Collision avoidance: Protection area active (bit)							
	15	14	13	12	11	10	9	8
DBB228	Collision avoidance: Protection area active (bit)							
	23	22	21	20	19	18	17	16
DBB229	Collision avoidance: Protection area active (bit)							
	31	30	29	28	27	26	25	24
DBB230	Collision avoidance: Protection area active (bit)							
	39	38	37	36	35	34	33	32
DBB231	Collision avoidance: Protection area active (bit)							
	47	46	45	44	43	42	41	40
DBB232	Collision avoidance: Protection area active (bit)							
	55	54	53	52	51	50	49	48
DBB233	Collision avoidance: Protection area active (bit)							
	63	62	61	60	59	58	57	56

4.6.13 DB10, collision avoidance: Activate protection area

Table 4-50 DB10, collision avoidance: Activate protection area

DB10	Signals from the NC (PLC → NC)							
Byte	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
DBB234	Collision avoidance: Activate protection area (bit)							
	7	6	5	4	3	2	1	0
DBB235	Collision avoidance: Activate protection area (bit)							
	15	14	13	12	11	10	9	8
DBB236	Collision avoidance: Activate protection area (bit)							
	23	22	21	20	19	18	17	16
DBB237	Collision avoidance: Activate protection area (bit)							
	31	30	29	28	27	26	25	24
DBB238	Collision avoidance: Activate protection area (bit)							
	39	38	37	36	35	34	33	32
DBB239	Collision avoidance: Activate protection area (bit)							
	47	46	45	44	43	42	41	40
DBB240	Collision avoidance: Activate protection area (bit)							
	55	54	53	52	51	50	49	48
DBB241	Collision avoidance: Activate protection area (bit)							
	63	62	61	60	59	58	57	56

4.6.14 DB10, extension, handwheel signals from the NC

Table 4-51 DB10, extension, handwheel signals from the NC

DB10 /FB2/H1/	Signals from the NC (NC → PLC)							
Byte	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
DBB242	Handwheel 4 is moved							
DBB243	Handwheel 5 is moved							
DBB244	Handwheel 6 is moved							
DBB245			Ethernet handwheel is stationary					
			Handwheel 6	Handwheel 5	Handwheel 4	Handwheel 3	Handwheel 2	Handwheel 1
DBB246	Reserved							

4.6.15 DB10, interface robot status.

Table 4-52 DB10, signals from the robot

DB10	Signals from the PLC (PLC → NC)							
Byte	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
DBB248	Robot status byte 0							
DBB249	Robot status byte 1							
DBB250	Robot status byte 2							
DBB251	Robot status byte 3							
DBB252	Robot status byte 4							
DBB253	Robot status byte 5							
DBB254	Robot status byte 6							
DBB255	Robot status byte 7							

4.6.16 DB10, interface robot status

Table 4-53 DB10, signals to the robot

DB10	Signals from the NC (NC → PLC)							
Byte	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
DBB256	Robot control byte 0							
DBB257	Robot control byte 1							
DBB258	Robot control byte 2							
DBB259	Robot control byte 3							
DBB260	Robot control byte 4							
DBB261	Robot control byte 5							

DB10	Signals from the NC (NC → PLC)							
Byte	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
DBB262	Robot control byte 6							
DBB263	Robot control byte 7							

4.7 Mode group-specific signals

4.7.1 DB11, mode signals to the NC

The start address of a mode group interface can be calculated as follows:

DBB((n-1) * 20), with n = mode group number = 1, 2, 3, etc.

Examples: Mode group 1: DBB0, mode group 2: DBB20, mode group 3: DBB60, etc.

Table 4-54 DB11, mode signals to the NC

DB11 /FB2/H1/	**Signals to mode group 1 (PLC → NC)**							
Byte	**Bit 7**	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
DBB0	Mode group reset Z1-K1/	Mode group stop axes plus spindle /Z1-K1/	Mode group stop /Z1-K1/	Mode change disable /Z1-K1/		Operating mode /Z1-K1/		
						JOG	MDI	AUTO
DBB1	Single block /Z1-K1/					Machine function /Z1-K1/		
	Type A	Type B				REF	REPOS	TEACH IN
DBB2	Machine function To use the machine function signals in DB, the signal DB10.DBB57.0 "INC inputs in the mode signal range active" must be set to "1".							
			INCvar	INC10000	INC1000	INC100	INC10	INC1
DBB3	Not assigned							

See also: Table 4-39 DB10, general signals to the NCDB10General signals to the NC (Page 752)

4.7.2 DB11, mode signals from the NC

Table 4-55 DB11, mode signals from the NC

DB11	**Signals from mode group 1 (NC → PLC)**							
Byte	**Bit 7**	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
DBB4 Operating software → PLC						Mode strobe /Z1-K1/		
						JOG	MDI	AUTO

DB11	Signals from mode group 1 (NC → PLC)							
Byte	**Bit 7**	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
DBB5 Operating software → PLC						Machine function strobe /Z1-K1/		
						REF	REPOS	TEACH IN
DBB6	All channels in the reset state		NC internal JOG active /K1/	Mode group reset performed /K1/	Mode group ready /Z1-K1/	Active operating mode /Z1-K1/		
						JOG	MDI	AUTO
DBB7						Active machine function /Z1-K1/		
						REF	REPOS	TEACH IN
DBB8	Machine functions							
			INCvar	INC10000	INC1000	INC100	INC10	INC1

4.8 SPL signals (Safety Integrated)

4.8.1 DB18, parameterization part

Table 4-56 DB18, parameterization part

DB18 /FBSIsl/	SPL signals (PLC → PLC)							
Byte	**Bit 7**	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
DBB0 - DBB34								
DBB36							Stop E	SPL ready
DBB37								

4.8.2 DB18, data area / errors

Table 4-57 DB18, data area / errors

DB18	SPL signals (PLC ←→ NC)							
Byte	**Bit 7**	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
	Data area of the SPL inputs/outputs							
DBB38 - DBB41	SPL_DATA.INSEP [1..32]							
DBB42 - DBB45	SPL_DATA.INSEP [33..64]							
DBB46 - DBB49	SPL_DATA.OUTSEP [1..32]							
DBB50 - DBB53	SPL_DATA.OUTSEP [33..64]							
	Data area for user SPL							
DBB54 - DBB57	SPL_DATA.INSIP [1..32]							
DBB58 - DBB61	SPL_DATA.INSIP [33..64]							
DBB62 - DBB65	SPL_DATA.OUTSIP [1..32]							
DBB66 - DBB69	SPL_DATA.OUTSIP [33..64]							
DBB70 - DBB73	SPL_DATA.MARKERSIP [1..32]							
DBB74 - DBB77	SPL_DATA.MARKERSIP [33..64]							

DB18	SPL signals (PLC ←→ NC)							
Byte	**Bit 7**	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
	Difference in signal level NC - PLC for diagnostics							
DBB78 - DBB81	-DBB81 SPL_DELTA.INSEP [1..32]							
DBB82 - DBB85	SPL_DELTA.INSEP [33..64]							
DBB86 - DBB89	SPL_DELTA.OUTSEP [1..32]							
DBB90 - DBB93	SPL_DELTA.OUTSEP [33..64]							
DBB94 - DBB97	SPL_DELTA.INSIP [1..32]							
DBB98 - DBB101	SPL_DELTA.INSIP [33..64]							
DBB102- DBB105	SPL_DELTA.OUTSIP [1..32]							
DBB106 - DBB109	SPL_DELTA.OUTSIP [33..64]							
DBB110 - DBB113	SPL_DELTA.MARKERSIP [1..32]							
DBB114 - DBB117	SPL_DELTA.MARKERSIP [33..64]							
DBB118								CMDSI
DBB119		NC signals a stop to the PLC	System error, crosswise data comparison	Crosswise data comparison error, SPL protection status		PROFIsafe communications error		
DBD120	Fault number 0 = no error 1 - 320 = signal number starting from SPL_DATA.INSEP[1]							
DBD124	CDC stack level display (diagnostics capability: How many SPL signals currently have a different level)							

4.8.3 DB18, additional data areas

Table 4-58 DB18, additional data areas

DB18	SPL signals (PLC ←→ NC)							
Byte	**Bit 7**	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
	Data area of the single-channel inputs/outputs							
DBB128	PLC_SI_OUT [1..8]							
NC → PLC								
DBB129	PLC_SI_OUT [9..16]							
NC → PLC								
DBB130	PLC_SI_OUT [17..24]							
NC → PLC								
DBB131	PLC_SI_OUT [25..32]							
NC → PLC								
DBB132	PLC_SI_IN [1..8]							
NC → PLC								
DBB133	PLC_SI_IN [9..16]							
NC → PLC								
DBB134	PLC_SI_IN [17..24]							
NC → PLC								
DBB135	PLC_SI_IN [25..32]							
NC → PLC								
DBB136-	SPL status							
DBB137								
DBB138	**PROFIsafe module(s) for input byte**							
	8	7	6	5	4	3	2	1
DBB139								
DBB140	**PROFIsafe module(s) for output byte**							
	8	7	6	5	4	3	2	1
DBB141								
DBB142 -								
DBB149								
DBB150 -								
DBB157								
DBB158 -								
DBB188								

4.8.4 DB18, F_SENDDP sender

Table 4-59 DB18, F_SENDDP sender

DB18	SPL signals (PLC ←→ NC)							
Byte	**Bit 7**	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
	1st F_SENDDP interface: FSDP[1]							
DBW190	ERR_REAC							
DBB192							SUBS_ON	ERROR
DBB193								
DBW194	DIAG							
DBW196	RETVAL14							
DBW198	RETVAL15							
	2nd F_SENDDP interface: FSDP[2]							
DBW200	ERR_REAC							
DBB202							SUBS_ON	ERROR
DBB203								
DBW204	DIAG							
DBW206	RETVAL14							
DBW208	RETVAL15							
	3rd F_SENDDP interface: FSDP[3]							
DBW210	ERR_REAC							
DBB212							SUBS_ON	ERROR
DBB213								
DBW214	DIAG							
DBW216	RETVAL14							
DBW218	RETVAL15							

4.8.5 DB18, F_SENDDP receiver

Table 4-60 DB18, F_SENDDP receiver

DB18	SPL signals (PLC ←→ NC)							
Byte	**Bit 7**	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
	1st F_RECVDP interface: FRDP[1] (SUBS)							
DBB220	7	6	5	4	3	2	1	0
DBB221	15	14	13	12	11	10	9	8
DBB222	REAC							
DBB224								ACK_REI
DBB225					SEND MODE	ACK_REQ	SUBS_ON	ERROR
DBW226	DIAG							

DB18	SPL signals (PLC ←→ NC)							
Byte	**Bit 7**	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
DBW228	RETVAL14							
DBW230	RETVAL15							
	2nd F_RECVDP interface: FRDP[2] (SUBS)							
DBB232	7	6	5	4	3	2	1	0
DBB233	15	14	13	12	11	10	9	8
DBW234								
DBB236								ACK_REI
DBB237					SEND MODE	ACK_REQ	SUBS_ON	ERROR
DBW238	DIAG							
DBW240	RETVAL14							
DBW242	RETVAL15							
	3rd F_RECVDP interface: FRDP[3] (SUBS)							
DBB244	7	6	5	4	3	2	1	0
DBB245	15	14	13	12	11	10	9	8
DBW246	REAC							
DBB248								ACK_REI
DBB249					SEND MODE	ACK_REQ	SUBS_ON	ERROR
DBW250	DIAG							
DBW252	RETVAL14							
DBW254	RETVAL15							

4.8.6 DB18, SPL user data

Table 4-61 DB18, SPL user data

DB18	SPL signals (PLC ←→ NC)							
Byte	**Bit 7**	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
DBD256	SPL_USER_DATA[0]							
DBD260	SPL_USER_DATA[1]							
DBD264	SPL_USER_DATA[2]							
DBD268	SPL_USER_DATA[3]							

4.8.7 DB18, data area / errors: Extended data area

Table 4-62 DB18, data area / errors: Extended data area

DB18	Signals for the safety SPL (PLC ←→ NC)							
Byte	**Bit 7**	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
	Data area of the SPL inputs/outputs							
DBD272	SPL_DATA_HF.INSEP [65..96]							
DBD276	SPL_DATA_HF.INSEP [97..128]							
DBD280	SPL_DATA_HF.INSEP [129..160]							
DBD284	SPL_DATA_HF.INSEP [161..192]							
DBD288	SPL_DATA_HF.OUTSEP [65..96]							
DBD292	SPL_DATA_HF.OUTSEP [97..128]							
DBD296	SPL_DATA_HF.OUTSEP [129..160]							
DBD300	SPL_DATA_HF.OUTSEP [161..192]							
	Data area for user SPL							
DBD304	SPL_DATA_HF.INSIP [65..96]							
DBD308	SPL_DATA_HF.INSIP [97..128]							
DBD312	SPL_DATA_HF.INSIP [129..160]							
DBD316	SPL_DATA_HF.INSIP [161..192]							
DBD320	SPL_DATA_HF.OUTSIP [65..96]							
DBD324	SPL_DATA_HF.OUTSIP [97..128]							
DBD328	SPL_DATA_HF.OUTSIP [129..160]							
DBD332	SPL_DATA_HF.OUTSIP [161..192]							
DBD336	SPL_DATA_HF. MARKERSIP [65..96]							
DBD340	SPL_DATA_HF. MARKERSIP [97..128]							

DB18	Signals for the safety SPL (PLC ←→ NC)							
Byte	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
DBD344	SPL_DATA_HF. MARKERSIP [129..160]							
DBD348	SPL_DATA_HF. MARKERSIP [161..192]							
	Difference in signal level NC - PLC for diagnostics							
DBD352	SPL_DELTA_HF.INSEP [65..96]							
DBD356	SPL_DELTA_HF.INSEP [97..128]							
DBD360	SPL_DELTA_HF.INSEP [129..160]							
DBD364	SPL_DELTA_HF.INSEP [161..192]							
DBD368	SPL_DELTA_HF.OUTSEP [65..96]							
DBD372	SPL_DELTA_HF.OUTSEP [97..128]							
DBD376	SPL_DELTA_HF.OUTSEP [129..160]							
DBD380	SPL_DELTA_HF.OUTSEP [161..192]							
DBD384	SPL_DELTA_HF. INSIP [65..96]							
DBD388	SPL_DELTA_HF.INSIP [97..128]							
DBD392	SPL_DELTA_HF.INSIP [129..160]							
DBD396	SPL_DATA_HF.INSIP [161..192]							
DBD400	SPL_DELTA_HF.OUTSIP [65..96]							
DBD404	SPL_DELTA_HF.OUTSIP [97..128]							
DBD408	SPL_DELTA_HF.OUTSIP [129..160]							
DBD412	SPL_DELTA_HF.OUTSIP [161..192]							
DBD416	SPL_DELTA_HF.MARKERSIP [65..96]							
DBD420	SPL_DELTA_HF. MARKERSIP [97..128]							

DB18	Signals for the safety SPL (PLC ←→ NC)							
Byte	**Bit 7**	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
DBD424	SPL_DELTA_HF. MARKERSIP [129..160]							
DBD428	SPL_DELTA_HF. MARKERSIP [161..192]							

4.8.8 DB18, additional data areas: Extended data area

Table 4-63 DB18, additional data areas: Extended data area

DB18	Signals for the safety SPL (PLC ←→ NC)							
Byte	**Bit 7**	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
	Data area of the single-channel inputs/outputs							
DBB432	PLCSIOUT_HF [33..40]							
from the NC								
DBB433	PLCSIOUT_HF [41..48]							
from the NC								
DBB434	PLCSIOUT_HF [49..56]							
from the NC								
DBB435	PLCSIOUT_HF [57..64]							
from the NC								
DBB436	PLCSIOUT_HF [65..72]							
from the NC								
DBB437	PLCSIOUT_HF [73..80]							
from the NC								
DBB438	PLCSIOUT_HF [81..88]							
from the NC								
DBB439	PLCSIOUT_HF [89..96]							
from the NC								
DBB440	PLCSIIN_HF [33..40]							
to the NC								
DBB441	PLCSIIN_HF [41..48]							
to the NC								
DBB442	PLCSIIN_HF [49..56]							
to the NC								

DB18	Signals for the safety SPL (PLC ←→ NC)							
Byte	**Bit 7**	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
DBB443 to the NC	PLCSIIN_HF [57..64]							
DBB444 to the NC	PLCSIIN_HF [65..72]							
DBB445 to the NC	PLCSIIN_HF [73..80]							
DBB446 to the NC	PLCSIIN_HF [81..88]							
DBB447 to the NC	PLCSIIN_HF [89..96]							

4.9 Signals from/to the operator panel (OP)

4.9.1 DB19, signals to the operator panel (OP)

Table 4-64 DB19, signals to the operator panel (OP)

DB19	Signals from the PLC to the OP (PLC → OP)							
Byte	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
DBB0	Interface of 1st control (DBB0 - DBB49)							
	Actual value in the WCS, (1) / MCS (0) /Z1-A2/	Save action log	HMI Advanced: Shutdown	Delete recall alarms	Delete cancel alarms	Key disable /Z1-A2/	Darken screen /Z1-A2/	Brighten screen /Z1-A2/
DBB1							Rights for the external viewer	External viewer
DBB2								
DBB4								
DBB6	Analog spindle 1: Utilization as a percentage							
DBB7	Analog spindle 2: Utilization as a percentage							
DBB8	Channel number of the machine control panel (MCP) at the control							
DBB9	Reserved selection					Autotool measure	OEM2	OEM1
DBB10	Hardkeys /FB1-P3/							
DBB11	Reserved							
DBB12								
DBB13	Part program /Z1-A2/			Reserved				Disable teach in transfer IHsl / IM9
	Selection	Loading	Unloading					
DBB14	Active (0) / passive (1) file system	V24 active file system: Index of the file to be transferred from the standard list. V24 passive file system: Number of the control file for user file names.						
DBB15	V24 active file system: Index which specifies the axis, channel or TO no. V24 passive file system: Index of the file to be transferred from the user list.							
DBB16	Active (0) / passive (1) file system	Program selection from the PLC: Index of the program list.						
DBB17	Program selection from the PLC: Program index in the program list.							
DBB18								
DBB19	Reserved (message counter)							

4.9.2 DB19, signals from the operator panel (OP)

Table 4-65 DB19, signals from the control

DB19	Signals from the OP to the PLC (OP → PLC)							
Byte	**Bit 7**	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
DBB20	Switch over machine coordinate system / workpiece coordinate system /Z1-A2/	Simulation active /Z1-A2/	Language 2 switched over /IHsl/IM9/	Recall alarm deleted /Z1-A2/	Cancel alarm deleted /Z1-A2/	Cancel key actuated /Z1-A2/	Screen is dark /Z1-A2/	
DBB21	Current number of the active operating area /FB1-P3/; /Z1-A2/							
DBB22	Current channel number /Z1-A2/							
DBB23						Control 1		
						Screen change active	Data transfer active	Keyboard has been used
DBW24	Current screen number /FB1-P3/							
DBB25								
DBB26	**Program selection from the PLC**: Status signals /Z1-A2/							
	Selection	Loading	Unloading		Active	Error	Job completed	Reserved
DBB27	**Program selection from the PLC**: Error identification /Z1-A2/							
DBW28	Screen number for "Supplement user interface" /IHsl-BE2/							
DBB30 PLC → OP	**Screen selection from the PLC:** Control signals							
							Screen deselection	Screen selection
DBB31 OP → PLC	**Screen selection from the PLC:** Status signals							
	Inactive			Error, screen selection not possible	Screen is deselected	Screen active	Screen is selected	Screen selection accepted
DBB32 PLC → OP	Busy function	Strobe function	Function selection no. from the PLC					
DBB33 PLC → OP	Parameter 1 for function selection no. (function selection from DBB32)							
DBB34 PLC → OP	Parameter 2 for function selection no. (function selection from DBB32)							
DBB35 PLC → OP	Parameter 3 for function selection no. (function selection from DBB32)							
DBB36 PLC → OP	Error code for function selection no. (function selection from DBB32)							

DB19	**Signals from the OP to the PLC (OP → PLC)**							
Byte	**Bit 7**	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
DBB37 PLC → OP	Parameter 1 for function selection no. (function selection from DBB48)							
DBB38 PLC → operating software	Parameter 2 for function selection no. (function selection from DBB48)							
DBB39 PLC → OP	Parameter 3 for function selection no. (function selection from DBB48)							
DBB40 - DBB47	Reserved							
DBB48 OP → PLC	PLC busy Function	Operating software strobe Function	Function selection No. from the operating software					
DBB49 PLC → OP	Error code for function selection no. (function selection from DBB48)							
DBB50 - DBB99	**Interface of 2nd control** (assignment the same as DBB0 - DBB49)							
DBB100	**Switchover interface to the operating software** **Call waiting interface (operating software announces itself to the NC)** ONL_REQUEST (online request from the operating software) /Z2-B3/ Operating software writes its client identification as online request (bits 8-15: bus type, bits 0-7: Bus address)							
DBB102	ONL_CONFIRM (acknowledgment from PLC regarding online request) /Z2-B3/ PLC writes as acknowledgement the operating software client identification (bus type, bus address; such as DBB100).							
DBB104	PAR_CLIENT_IDENT /Z2-B3/ Operating software writes its client identification (bus type, bus address; such as DBB100).							
DBB106	PAR_MMC_TYP /Z2-B3/ Type of operating software acc. to NETNAMES.INI: Main/secondary operator panel / server /...							
DBB107	PAR_MSTT_ADR /Z2-B3/ Operating software writes the address of the MCP to be activated; 255, if no MCP activation							
DBB108	PAR_STATUS /Z2-B3/ PLC writes the online enable for the operating software							
DBB109	PAR_Z_INFO /Z2-B3/ PLC writes supplementary information on the status							
DBB110	M_TO_N_ALIVE Sign-of-life from the PLC to the operating software using the M to N block							
DBB112	Res. bus type MCP							
DBB113	ParOpKeyAdr Direct key index call waiting interface							
DBB114	ParTcuIndex TCU index call waiting interface							
DBB115	ParHt2Index HT2 index log on interface							

DB19	Signals from the OP to the PLC (OP → PLC)							
Byte	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
DBB116	Direct key address of 1st online interface							
DBB117	Direct key address of 2nd online interface							
DBB118	TCU index of 1st online interface							
DBB119	TCU index of 2nd online interface							
DBB120	**Online interface OP 1 (user)** MMC1_CLIENT_IDENT /Z2-B3/ PLC writes PAR_CLIENT_IDENT to MMCx_CLIENT_IDENT, if operating software goes online.							
DBB122	MMC1_TYP /Z2-B3/ PLC writes PAR_MMC_TYP to MMCx_TYP, if operating software goes online.							
DBB123	MMC1_MSTT_ADR /Z2-B3/ PLC writes PAR_MSTT_ADR to MMCx_MSTT_ADR, if operating software goes online.							
DBB124	MMC1_STATUS /Z2-B3/ Connection state, operating software and PLC write alternating their requests/acknowledgements.							
DBB125	MMC1_Z_INFO /Z2-B3/ Additional information, connection state (pos./neg. acknowledgement, error messages, etc.)							
DBB126	Reserved	TCU1_ SHIFT_ LOCK	MMC1_ CHANGE_ DENIED /Z2-B3/	MMC1_ ACTIVE_ CHANGED /Z2-B3/	MMC1_ ACTIVE_ PERM /Z2-B3/	MMC1_ ACTIVE_ REQ /Z2-B3/	MMC1_ MSTT_ SHIFT- LOCK /Z2-B3/	MMC1_ SHIFT_ LOCK /Z2-B3/
DBB127	Reserved bus type MCP							
DBB128 - DBB129	Reserved Transline (Transline DB number)							
DBB130	**Online interface OP 2 (user)** MMC2_CLIENT_IDENT /Z2-B3/ PLC writes PAR_CLIENT_IDENT to MMCx_CLIENT_IDENT, if operating software goes online.							
DBB132	MMC2_TYP /Z2-B3/ PLC writes PAR_MMC_TYP to MMCx_TYP, if operating software goes online.							
DBB133	MMC2_MSTT_ADR /Z2-B3/ PLC writes PAR_MSTT_ADR to MMCx_MSTT_ADR, if operating software goes online.							
DBB134	MMC2_STATUS /Z2-B3/ Connection state, operating software and PLC write alternating their requests/acknowledgements.							
DBB135	MMC2_Z_INFO /Z2-B3/ Additional information, connection state (pos./neg. acknowledgement, error messages, etc.)							
DBB136	Reserved	TCU2_ SHIFT_ LOCK	MMC2_ CHANGE_ DENIED /Z2-B3/	MMC2_ ACTIVE_ CHANGED /Z2-B3/	MMC2_ ACTIVE_ PERM /Z2-B3/	MMC2_ ACTIVE_ REQ /Z2-B3/	MMC2_ MSTT_ SHIFT_ LOCK /Z2-B3/	MMC2_ SHIFT_ LOCK /Z2-B3/
DBB137	Reserved bus type MCP							

DB19	Signals from the OP to the PLC (OP → PLC)							
Byte	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
DBB138 - DBB139	Reserved Transline (Transline DB number)							
DBB140 - DBB197	Assignment of the PLC for transfer parameters These data blocks are reserved for the "Tool Ident Connection" option. /FBWsl/							
DBB198 - DBB249	Assignment of the PLC for return values These data blocks are reserved for the "Tool Ident Connection" option. /FBWsl/							
DBB250 - DBB255	Function call of the PLC interface These data blocks are reserved for the "Tool Ident Connection" option. /FBWsl/							
DBB256 - DBB267	Commands for Paramtm.exe							
DBB268	Traffic light status							
DBD270 - DBD394	Counter [1...32]							
DBB398	Handwheel number for simulation override							
DBW400	Simulation override							
DBW402	Simulation state							

4.10 Defining PLC alarms

4.10.1 DB20, NC machine data

Table 4-66 DB20, NC machine data

DB20	NC machine data (PLC → user)							
Byte	**Bit 7**	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
DBW0	INT values							
DBW								
DBW	INT values							
DBB	Bit arrays							
DBB								
DBB								
DBB	Bit arrays							
DBD	REAL values							
DBD								
DBD	REAL values							

Note

The start and end addresses of the PLC machine data areas are dependent on the particular length data of the partial areas. The range of INTEGER values always starts with data byte 0. The upper limit is defined by the associated length data. The range of the bit arrays starts after the range of the INTEGER values at the next even address. The range of REAL values starts after the range of the bit arrays at the next even address.

4.11 Channel-specific signals

4.11.1 DB21 - DB30, control signals to the channel (1)

Table 4-67 DB21 - DB30, control signals to the channel

DB21 - DB30	Signals to the channel (PLC → NC)							
Byte	**Bit 7**	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
DBB0		Activate dry run feed /Z1-V1/	Activate M01 /Z1-K1/	Activate single block /Z1-K1/	Activate DRF /Z2-H1/			
DBB1	Activate program test /Z1-K1/	PLC action completed /Z1-K1/	CLC correction /Z3-TE1/	CLC stop /Z3-TE1/	Time monitoring active (tool management)	Synchronized action off /FBSY/	Enable protection areas /Z1-A3/	Activate referencing /Z1-R1/
DBB2	Activate skip block /Z1-K1/							
	/7	/6	/5	/4	/3	/2	/1	/0
DBB3	Nibbling and punching /K2-N4/							
			Manual stroke release 2	Stroke inoperative /K2-N4	Delayed stroke /K2-N4/	Stroke suppression /K2-N4/	Manual stroke release /K2-N4/	Stroke enable /K2-N4/
DBB4	Feedrate override /Z1-V1/							
	H	G	F	E	D	C	B	A
DBB5	Rapid traverse override /Z1-V1/							
	H	G	F	E	D	C	B	A
DBB6	Feedrate override active /Z1-V1/	Rapid traverse override active /Z1-V1/		Program level cancel /Z1-K1/	Delete UP number of passes	Delete distance-to-go /Z1-A2/	Read-in disable /Z1-K1/	Feed disable /Z1-V1/
DBB7	Reset /Z1-K1		Suppress start lock	NC stop axes plus spindle /Z1-K1/	NC stop /Z1-K1/	NC stop at block limit /Z1-K1/	NC start /Z1-K1/	NC start disable /Z1-K1/
DBB8	Activate machine-related protection zone /Z1-A3/							
	8	7	6	5	4	3	2	1
DBB9	Activate machine-related protection zone /Z1-A3/							
							10	9
DBB10	Activate channel-specific protection zone /Z1-A3/							
	8	7	6	5	4	3	2	1
DBB11	Activate channel-specific protection zone /Z1-A3/							
							10	9

Note

- Feedrate override active: Even if the feedrate override is not active (=100%), the 0% position is still effective.
- Feedrate override: Either 31 positions (Gray code) with 31 MD for % evaluation, or 0% to 200% corresponding to the dual value in the byte (201 to 255 = max. 200%)
- Rapid traverse override: Either 31 positions (Gray code) with 31 MD for % evaluation, or 0% to 100% corresponding to the dual value in the byte (101 to 255 = max. 100%)
- Single block: Select the version using "Write variable"
- Delete distance-to-go: Is only active for path axes and not for positioning axes

4.11.2 DB21 - DB30, control signals to the geometry axes

Table 4-68 DB21 - DB30, control signals to the geometry axes

DB21 - DB30 /FB2/H1/	Signals to the channel (PLC → NC)							
Byte	**Bit 7**	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
	Geometry axis 1							
DBB12	Traversing keys		Rapid traverse override	Traversing key lock	Feedrate stop	Activate handwheel		
	Plus	Minus				C	B	A
DBB13	Requested machine function							
		Continuously	INCvar	INC10000	INC1000	INC100	INC10	INC1
DBB14	OEM signals							
DBB15								Invert handwheel direction of rotation
	Geometry axis 2							
DBB16	Traversing keys		Rapid traverse override	Traversing key lock	Feedrate stop	Activate handwheel		
	Plus	Minus				C	B	A
DBB17	Requested machine function							
		Continuously	INCvar	INC10000	INC1000	INC100	INC10	INC1
DBB18	OEM signals							

DB21 - DB30 /FB2/H1/	Signals to the channel (PLC → NC)						
Byte	**Bit 7**	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
DBB19								Invert handwheel direction of rotation
	Geometry axis 3							
DBB20	Traversing keys		Rapid traverse override	Traversing key lock	Feedrate stop	Activate handwheel		
	Plus	Minus				C	B	A
DBB21	Requested machine function							
		Continuously	INCvar	INC10000	INC1000	INC100	INC10	INC1
DBB22	OEM signals							
DBB23								Invert handwheel direction of rotation

Note

The NC only evaluates the machine function signals if the signal DB10.DBX57.0 "INC inputs in mode group area active" is not set.

See also: Table 4-39 DB10, general signals to the NCDB10General signals to the NC (Page 752)

4.11.3 DB21 - DB30, HMI signals to channel / OEM signals from/to channel

Table 4-69 DB21 - DB30, control signals from the operating software to the PLC, PLC to the NC and status signal from channel to the PLC

DB21 - DB30	**Signals from the channel/PLC/operating software (operating software → PLC, PLC → NC, NC → PLC)**						
Byte	**Bit 7**	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
DBB24 Operating software → PLC		Dry run feedrate selected /Z1-V1/	M01 selected /Z1-K1/	Select M01 associated with NC	DRF selected /Z2-H1/			SINUM. Integrate loads program
DBB25 Operating software → PLC	Program test selected /Z1-K1/			REPOS mode change /Z1-K1/	Feedrate override selected for rapid traverse /Z1-V1/	REPOS mode /Z1-K1/		
						C	B	A
DBB26 Operating software → PLC	Skip block selected, level /x /Z1-K1/							
	/7	/6	/5	/4	/3	/2	/1	/0
DBB27 Operating software → PLC							Skip block is selected, level/x	
							/9	/8
	OEM channel signals							
DBB28 PLC → NC								
DBB29 PLC → NC	Tool do not lock	Deactivate wear monitoring	Deactivate workpiece counter	Activate PTP traversal /Z3-F2/	Activate fixed feedrate /FBMA/, /Z1-V1/			
					4	3	2	1
DBB30 PLC → NC	No tool change commands	Jog circle	Activate M01 associated with NC /Z1-H2/	Contour handwheel simulation, negative direction /Z1-H2/	Contour handwheel simulation on /Z1-H2/	Activate contour handwheel /Z2-H1/		
						C	B	A
DBB31 PLC → NC	Skip block active		Invert contour handwheel direction of rotation /Z1-H2/	REPOS mode change /Z1-K1/		REPOS mode		
	/9	/8				C	B	A

DB21 - DB30	Signals from the channel/PLC/operating software (operating software → PLC, PLC → NC, NC → PLC)							
Byte	**Bit 7**	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
DBB32 NC → PLC		Last action block active /Z1-K1/	M00/M01 active /Z1-K1/	Approach block active /Z1-K1/	Action block active /Z1-K1/			Execution from external active
DBB33 NC → PLC	Program test active /Z1-K1/	Transformation active /Z1-K1 /K2-M1/ /Z3-F2/	M02/M30 active /Z1-K1/	Block search active /Z1-K1/	Handwheel override active /Z2-H1/	Revolutional feedrate active /Z1-V1/	Orientable toolholder active	Referencing active /Z1-R1/
DBB34	**OEM channel signals feedback signals**							
NC → PLC								
DBB35	Channel state /Z1-K1/			Program state /Z1-K1/				
NC → PLC	Reset	Interrupted	Active	Aborted	Interrupted	Stopped	Waiting	Running
DBB36 NC → PLC	NC alarm with processing stop present /Z1-A2/	Channel-specific NC alarm is active /Z1-A2/	Channel is ready /Z2-K3/	Interrupt processing active /Z1-K1/	All axes stationary /Z1-B1/	All axes that have to be referenced are referenced /Z1-R1/		
DBB37 NC → PLC	Stop at the end of block with SBL is suppressed /Z1-K1/	Read-in enable is ignored /Z1-K1/	CLC stopped Upper limit /Z3-TE1/	CLC stopped Lower limit /Z3-TE1/	CLC active /Z3-TE1/	Contour handwheel active /Z2-H1/		
						C	B	A
DBB38	Nibbling and punching /Z2-N4/							
NC → PLC							Acknowledgement of manual stroke release /K2-N4/	Stroke release active /Z2-N4/
DBB39 NC → PLC			Contour handwheel direction of rotation inverted					Protection areas not guaranteed

4.11.4 DB21 - DB30, control signals from the geometry axes

Table 4-70 DB21 - DB30, control signals from the geometry axes

DB21 - DB30 /FB2/H1/	Signals from the channel (NC → PLC)							
Byte	**Bit 7**	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
	Geometry axis 1							
DBB40	Travel command		Traversing requests			Handwheel active		
	Plus	Minus	Plus	Minus		C	B	A
DBB41	Active machine function							
		Continuous	INCvar	INC10000	INC1000	INC100	INC10	INC1
DBB42	OEM signals							
DBB43								Handwheel direction of rotation inverted
DBB44 Operating software → PLC								
	Geometry axis 2							
DBB46	Travel command		Traversing requests			Handwheel active		
	Plus	Minus	Plus	Minus		C	B	A
DBB47	Active machine function							
		Continuous	INCvar	INC10000	INC1000	INC100	INC10	INC1
DBB48	OEM signals							
DBB49								Handwheel direction of rotation inverted
DBB50 Operating software → PLC								
	Geometry axis 3							
DBB52	Travel command		Traversing requests			Handwheel active		
						C	B	A

DB21 - DB30 /FB2/H1/	Signals from the channel (NC → PLC)							
Byte	**Bit 7**	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
DBB53	Active machine function /FB2/H1/							
		Continuous	INCvar	INC10000	INC1000	INC100	INC10	INC1
DBB54	OEM signals							
DBB55								Handwheel direction of rotation inverted
DBB56 Operating software → PLC								

4.11.5 DB21 - DB30, change signals for auxiliary function transfer from the channel

Table 4-71 DB21 - DB30, change signals for auxiliary function transfer from the channel

DB21 - DB30	Signals from the channel (NC → PLC)							
Byte	**Bit 7**	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
DBB58				Change				
				M fct. 5 /Z1-H2/	M fct. 4 /Z1-H2/	M fct. 3 /Z1-H2/	M fct. 2 /Z1-H2/	M fct. 1 /Z1-H2/
DBB59				Not decoded				
				M fct. 5	M fct. 4	M fct. 3	M fct. 2	M fct. 1
DBB60		Quick				Change		
		S fct. 3	S fct. 2	S fct. 1		S fct. 3 /Z1-H2/	S fct. 2 /Z1-H2/	S fct. 1 /Z1-H2/
DBB61		Quick				Change		
		T fct. 3	T fct. 2	T fct. 1		T fct. 3 /Z1-H2/	T fct. 2 /Z1-H2/	T fct. 1 /Z1-H2/
DBB62		Quick				Change		
		D fct. 3	D fct. 2	D fct. 1		D fct. 3 /Z1-H2/	D fct. 2 /Z1-H2/	D fct. 1 /Z1-H2/
DBB63				DL fct. Quick				DL fct.change

DB21 - DB30	Signals from the channel (NC → PLC)							
Byte	**Bit 7**	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
DBB64		Quick				Change		
		H fct. 3	H fct. 2	H fct. 1		H fct. 3 /Z1-H2/	H fct. 2 /Z1-H2/	H fct. 1 /Z1-H2/
DBB65			Change					
			F fct. 6 /Z1-H2/	F fct. 5 /Z1-H2/	F fct. 4 /Z1-H2/	F fct. 3 /Z1-H2/	F fct. 2 /Z1-H2/	F fct. 1 /Z1-H2/
DBB66				Quick				
				M fct. 5	M fct. 4	M fct. 3	M fct. 2	M fct. 1
DBB67			Quick					
			F fct. 6	F fct. 5	F fct. 4	F fct. 3	F fct. 2	F fct. 1

Note

- For 10-decade T numbers, only the signal DBB61, DBX0 "T fct.1 change" is available.
- For 5-decade D numbers, only the signal DBB62, DBX0 "D fct.1 change" is available.

4.11.6 DB21 - DB30, transferred M and S functions:

Table 4-72 DB21 - DB30, transferred M and S functions:

DB21 - DB30	Signals from the channel (NC → PLC) /Z1-H2/							
Byte	**Bit 7**	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
DBW68	Extended address, M function 1 (16-bit INT)							
DBD70	M function 1 (DINT)							
DBW74	Extended address, M function 2 (16-bit INT)							
DBD76	M function 2 (DINT)							
DBW80	Extended address, M function 3 (16-bit INT)							
DBD82	M function 3 (DINT)							
DBW86	Extended address, M function 4 (16-bit INT)							
DBD88	M function 4 (DINT)							
DBW92	Extended address, M function 5 (16-bit INT)							
DBD94	M function 5 (DINT)							
DBW98	Extended address, S function 1 (16-bit INT)							
DBD100	S function 1 (REAL format)							
DBW104	Extended address, S function 2 (16-bit INT)							
DBD106	S function 2 (REAL format)							
DBW110	Extended address, S function 3 (16-bit INT)							
DBD112	S function 3 (REAL format)							

Note

M functions are programmed in the part program in the INTEGER format (8 decades plus leading sign).

4.11.7 DB21 - DB30, transferred T/D/DL functions

Table 4-73 DB21 - DB30, transferred T/D/DL functions

DB21 - DB30	Signals from the channel (NC → PLC)							
Byte	**Bit 7**	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
DBW116	Extended address, T function 1 (16-bit INT)							
DBW118 **DBD118**	T function 1 (dual) / for 8-decade T no., DBD118 is used as T function 1 (32-bit INT) (see note below) /Z1-H2/							
DBW120	Extended address, T function 2 (16-bit INT)							
DBW122	T function 2 (INT)							
DBW124	Extended address, T function 3 (16-bit INT)							
DBW126	T function 3 (INT)							
DBB128	Extended address D function 1 (8-bit INT)							
DBB129	D function 1 (dual) /Z1-H2/							
DBW130 **DBB130**	For 5-decade D no., DBW130 is used as D function 1 (16-bit INT) Extended address D function 2 (8-bit INT)							
DBB131	D function 2 (8-bit INT)							
DBB132	Extended address D function 3 (8-bit INT)							
DBB133	D function 3 (8-bit INT)							
DBW134	Extended address DL function (16-bit INT)							
DBD136	DL function (REAL)							

Note

- Programmed T functions are not output to the PLC when tool management is active.
- 8 decade T numbers are only available under DBD118 "T function 1".
- Programmed D functions with names (e.g. D=CUTTING EDGE_1) cannot be output to the PLC in ASCII format.
- 5-decade D numbers are only available as DBW130 "D function 1".
- The REAL format corresponds to the floating-point representation in STEP 7 (24-bit mantissa and 8-bit exponent). This floating point format supplies a maximum of seven valid places.

4.11.8 DB21 - DB30, transferred H/F functions

Table 4-74 DB21 - DB30, transferred H/F functions

DB21 - DB30	Signals from the channel (NC → PLC) /Z1-H2/							
Byte	**Bit 7**	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
DBW140	Extended address, H function 1 (16-bit INT)							
DBD142	H function 1 (REAL or DINT)							
DBW146	Extended address, H function 2 (16-bit INT)							
DBD148	H function 2 (REAL or DINT)							
DBW152	Extended address, H function 3 (16-bit INT)							
DBD154	H function 3 (REAL or DINT)							
DBW158	Extended address F function 1 (16-bit INT)							
DBD160	F function 1 (REAL format)							
DBW164	Extended address F function 2 (16-bit INT)							
DBD166	F function 2 (REAL format)							
DBW170	Extended address F function 3 (16-bit INT)							
DBD172	F function 3 (REAL format)							
DBW176	Extended address F function 4 (16-bit INT)							
DBD178	F function 4 (REAL format)							
DBW182	Extended address F function 5 (16-bit INT)							
DBD184	F function 5 (REAL format)							
DBW188	Extended address F function 6 (16-bit INT)							
DBD190	F function 6 (REAL format)							

Note

- F functions are programmed in the part program in the REAL data format.
- The extended address of the F function contains an identifier with the following meaning:
  - 0: Path feedrate
  - 1 - 31: Machine axis number for feedrate for positioning axes
- The data type of the H function depends on machine data:
  MD22110 $MC_AUXFU_H_TYPE_INT

4.11.9 DB21 - DB30, decoded M signals

Table 4-75 DB21 - DB30, decoded M signals

DB21 - DB30	Signals from the channel (M0 - M99) (NC → PLC) /Z1-H2/							
Byte	**Bit 7**	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
	Dynamic M functions							
DBB194								
	M07	M06	M05 #	M04 #	M03 #	M02	M01	M00
DBB195								
	M15	M14	M13	M12	M11	M10	M09	M08
DBB196								
	M23	M22	M21	M20	M19	M18	M17	M16
DBB197								
	M31	M30	M29	M28	M27	M26	M25	M24
DBB198								
	M39	M38	M37	M36	M35	M34	M33	M32
DBB199								
	M47	M46	M45	M44	M43	M42	M41	M40
DBB200								
	M55	M54	M53	M52	M51	M50	M49	M48
DBB201								
	M63	M62	M61	M60	M59	M58	M57	M56
DBB202								
	M71	M70 *	M69	M68	M67	M66	M65	M64
DBB203								
	M79	M78	M77	M76	M75	M74	M73	M72
DBB204								
	M87	M86	M85	M84	M83	M82	M81	M80
DBB205								
	M95	M94	M93	M92	M91	M90	M89	M88
DBB206								
					M99	M98	M97	M96
DBB207								

Note

- #: The M function is not displayed here if a spindle is parameterized in the channel. In this case, the M function is displayed as extended M function under DB21,... DBB68 ff.. and axially under DB31,... DBB86 ff.
- **Dynamic** M functions (M00 - M99) are decoded by the basic PLC program.
  Static M functions must be generated in the PLC user program from dynamic M functions.

4.11.10 DB21 - DB30, active G functions

Table 4-76 DB21 - DB30, active G functions

DB21 - DB30	Signals from the channel (NC → PLC) /Z1-K1/							
Byte	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
DBB208	Number of the active G function of G function group 1 (8-bit INT)							
DBB209	Number of the active G function of G function group 2 (8-bit INT)							
DBB210	Number of the active G function of G function group 3 (8-bit INT)							
DBB211	Number of the active G function of G function group 4 (8-bit INT)							
DBB212	Number of the active G function of G function group 5 (8-bit INT)							
DBB213	Number of the active G function of G function group 6 (8-bit INT)							
DBB214	Number of the active G function of G function group 7 (8-bit INT)							
DBB215	Number of the active G function of G function group 8 (8-bit INT)							
...	...							
DBB270	Number of the active G function of G function group n-1 (8-bit INT)							
DBB271	Number of the active G function of G function group n (8-bit INT)							

Note

- The active G functions of the group, for each programming of a G function or a mnemonic identifier (e.g. SPLINE), are updated.
- G functions within a G group are output as binary value, starting with 1. A G function with the value 0 means that for this G group, no G function is active.

4.11.11 DB21 - DB30, protection areas from the channel

Table 4-77 DB21 - DB30, signals for the protection areas from the channel

DB21 - DB30	Signals from the channel (NC → PLC) /Z1-A3/							
Byte	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
DBB272	Machine-related protection area preactivated							
	8	7	6	5	4	3	2	1
DBB273	Machine-related protection area preactivated							
							10	9
DBB274	Channel-specific protection area preactivated							
	8	7	6	5	4	3	2	1
DBB275	Channel-specific protection area preactivated							
							10	9
DBB276	Machine-related protection area violated							
	8	7	6	5	4	3	2	1

DB21 - DB30	Signals from the channel (NC → PLC) /Z1-A3/							
Byte	**Bit 7**	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
DBB277	Machine-related protection area violated							
							10	9
DBB278	Channel-specific protection zone violated							
	8	7	6	5	4	3	2	1
DBB279	Channel-specific protection zone violated							
							10	9

4.11.12 DB21 - DB30, synchronous actions, signals from/to the channel

Note

The request signals should be set in the PLC user program. After data transfer, they are reset by the basic PLC program.

Table 4-78 DB21 - DB30, job-controlled signals from/to the channel

DB21 - DB30	Signals to the channel (PLC ←→ NC)							
Byte	**Bit 7**	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
DBB280 PLC → NC	---	---	---	---	---	---	Request: Disable synchronized actions (DB300 - DB307) /FBSY/	Reserved
DBB281 NC → PLC	---	---	---	---	---	---	Acknowledgement: Synchronized actions disabled (DB300 - DB307) /FBSY/	---
DBW282 - DBW298	Reserved							
DBB300	Disable synchronized actions /FBSY/							
PLC → NC	8	7	6	5	4	3	2	1
DBB301	Disable synchronized actions /FBSY/							
PLC → NC	16	15	14	13	12	11	10	9

DB21 - DB30	Signals to the channel (PLC ←→ NC)							
Byte	**Bit 7**	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
DBB302	Disable synchronized actions /FBSY/							
PLC → NC	24	23	22	21	20	19	18	17
DBB303	Disable synchronized actions /FBSY/							
PLC → NC	32	31	30	29	28	27	26	25
DBB304	Disable synchronized actions /FBSY/							
PLC → NC	40	39	38	37	36	35	34	33
DBB305	Disable synchronized actions /FBSY/							
PLC → NC	48	47	46	45	44	43	42	41
DBB306	Disable synchronized actions /FBSY/							
PLC → NC	56	55	54	53	52	51	50	49
DBB307	Disable synchronized actions /FBSY/							
PLC → NC	64	63	62	61	60	59	58	57
DBB308	Synchronized actions can be disabled /FBSY/							
NC → PLC	8	7	6	5	4	3	2	1
DBB309	Synchronized actions can be disabled /FBSY/							
NC → PLC	16	15	14	13	12	11	10	9
DBB310	Synchronized actions can be disabled /FBSY/							
NC → PLC	24	23	22	21	20	19	18	17
DBB311	Synchronized actions can be disabled /FBSY/							
NC → PLC	32	31	30	29	28	27	26	25
DBB312	Synchronized actions can be disabled /FBSY/							
NC → PLC	40	39	38	37	36	35	34	33
DBB313	Synchronized actions can be disabled /FBSY/							
NC → PLC	48	47	46	45	44	43	42	41
DBB314	Synchronized actions can be disabled /FBSY/							
NC → PLC	56	55	54	53	52	51	50	49
DBB315	Synchronized actions can be disabled /FBSY/							
NC → PLC	64	63	62	61	60	59	58	57

4.11.13 DB21 - DB30, control signals from/to the channel

Note

The request signals should be set in the PLC user program. After data transfer, they are reset by the basic PLC program.

Table 4-79 DB21 - DB30, job-controlled signals from/to the channel

DB21 - DB30	**Signals from/to the channel (PLC ←→ NC)**							
Byte	**Bit 7**	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
DBB316	Active G functions							
	---	---	---	---	---	---		G00 geo.
DBB317	Tool missing	PTP travel active /Z3-F2/	Drive test travel request	---	---	---	Workpiece setpoint reached	External language mode active
DBB318	Overstore active /F1/A2	Dry run feedrate active /FB1/V1/	M01 associated with NC active /FB3/H2/	Delayed stop	TOFF motion active /FB3/F2 /Z3-F2/	TOFF active /FB1/F2 /Z3-F2/	Block search via program test, SERUPRO, is active	ASUB stopped /FB1/K1
DBB319	No tool change commands active	Stop delay area not accepted	REPOS DEFERAL channel /FB1-K1/	Delay FTS	Active REPOS mode			Acknowledgement of the REPOS mode change /FB1-K1/
					C	B	A	

4.11.14 DB21 - DB30, signals to the orientation axes

Table 4-80 DB21 - DB30, signals to the orientation axes

DB21 - DB30	**Signals to the channel (PLC → NC)**							
Byte	**Bit 7**	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
	Orientation axis 1 /Z2-H1/							
DBB320	Traversing keys		Rapid traverse override	Traversing key lock	Feed stop	Activate handwheel		
	Plus	Minus				C	B	A
DBB321			INCvar	INC10000	INC1000	INC100	INC10	INC1
DBB322	OEM signals							
DBB323								Handwheel direction of rotation inverted
	Orientation axis 2 /Z2-H1/							
DBB324	Traversing keys		Rapid traverse override	Traversing key lock	Feed stop	Activate handwheel		
	Plus	Minus				C	B	A
DBB325			INCvar	INC10000	INC1000	INC100	INC10	INC1

DB21 - DB30	Signals to the channel (PLC → NC)							
Byte	**Bit 7**	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
DBB326	OEM signals							
DBB327								Handwheel direction of rotation inverted
	Orientation axis 3 /Z2-H1/							
DBB328	Traversing keys		Rapid traverse override	Traversing key lock	Feed stop	Activate handwheel		
	Plus	Minus				C	B	A
DBB329			INCvar	INC10000	INC1000	INC100	INC10	INC1
DBB330	OEM signals							
DBB331								Handwheel direction of rotation inverted

4.11.15 DB21 - DB30, signals from the orientation axes

Table 4-81 DB21 - DB30, signals from the orientation axes

DB21 - DB30	Signals from the channel (NC → PLC)							
Byte	**Bit 7**	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
	Orientation axis 1 /Z2-H1/							
DBB332	Travel command		Travel request			Handwheel active		
	Plus	Minus	Plus	Minus		C	B	A
DBB333	Active machine function							
			INCvar	INC10000	INC1000	INC100	INC10	INC1
DBB334	OEM signals							
DBB335								Handwheel direction of rotation inversion active
	Orientation axis 2 /Z2-H1/							
DBB336	Travel command		Travel request			Handwheel active		
	Plus	Minus	Plus	Minus		C	B	A
DBB337	Active machine function							
			INCvar	INC10000	INC1000	INC100	INC10	INC1

DB21 - DB30	Signals from the channel (NC → PLC)							
Byte	**Bit 7**	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
DBB338	OEM signals							
DBB339								Handwheel direction of rotation inversion active
	Orientation axis 3 /Z2-H1/							
DBB340	Travel command		Travel request			Handwheel active		
	Plus	Minus	Plus	Minus		C	B	A
DBB341	Active machine function							
			INCvar	INC10000	INC1000	INC100	INC10	INC1
DBB342	OEM signals							
DBB343								Handwheel direction of rotation inversion active

4.11.16 DB21 - DB30, tool management functions from the channel

Table 4-82 DB21 - DB30, tool management functions from the channel

DB21 - DB30	Signals from the channel (NC → PLC)							
Byte	**Bit 7**	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
	Change signals, tool management functions							
DBB344					Last replacement tool of the tool group	Transition to new replacement tool	Tool limit value reached	Tool prewarning limit reached
DBB345-DBB347								
	Transferred tool management functions							
DBD348	T number for tool prewarning limit (DINT)							
DBD352	T number for tool limit value (DINT)							
DBD356	T number of new replacement tool (DINT)							
DBD360	T number of last replacement tool (DINT)							

4.11.17 DB21 - DB30, control signals from the channel (2)

Table 4-83 DB21 - DB30, signals from the channel

DB21 - DB30	Signals from the channel (NC → PLC)							
Byte	**Bit 7**	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
DBB364	CH_CYCLES_SIG_IN (1)							
	8	7	6	5	4	3	2	1
DBB365	CH_CYCLES_SIG_IN (2)							
	16	15	14	13	12	11	10	9
DBB366	CH_CYCLES_SIG_OUT (1)							
	8	7	6	5	4	3	2	1
DBB367	CH_CYCLES_SIG_OUT (2)							
	16	15	14	13	12	11	10	9
DBB368	CH_OEM_TECHNO_SIG_IN (1)							
	8	7	6	5	4	3	2	1
DBB369	CH_OEM_TECHNO_SIG_IN (2)							
	16	15	14	13	12	11	10	9
DBB370	CH_OEM_TECHNO_SIG_IN (3)							
	24	23	22	21	20	19	18	17
DBB371	CH_OEM_TECHNO_SIG_IN (4)							
	32	31	30	29	28	27	26	25
DBB372	CH_OEM_TECHNO_SIG_OUT (1)							
	8	7	6	5	4	3	2	1
DBB373	CH_OEM_TECHNO_SIG_OUT (2)							
	16	15	14	13	12	11	10	9
DBB374	CH_OEM_TECHNO_SIG_OUT (3)							
	24	23	22	21	20	19	18	17
DBB375	CH_OEM_TECHNO_SIG_OUT (4)							
	32	31	30	29	28	27	26	25
DBB376	ProgEventDisplay							
DBB377		Jog circle active	Retract data available	JOG retract active			Stop condition	Collision avoidance: Stop
DBB378							Still ASUB is active	ASUB is active
DBB379								
DBB380	Reserved ASUB							
DBB381	Reserved ASUB							
DBB382	Reserved ASUB							

DB21 - DB30	Signals from the channel (NC → PLC)							
Byte	**Bit 7**	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
DBB383	Reserved ASUB							

4.11.18 DB21 - DB30, control signals to the channel (2)

Table 4-84 DB21 - DB30, signals to the channel

DB21 - DB30	Signals to the channel (PLC → NC)							
Byte	**Bit 7**	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
DBX384	---	---	---	---	---	---	---	Control program branch (GOTOS) (Z1-K1/ PGAsl)
DBX385	Grinding: Input signals 1 ... 8 ($AC_IN_KEY_G[1 ... 8]) /LIS3sl/							
	8	7	6	5	4	3	2	1
DBX386	Grinding: Disable input signals 1 ... 8							
	8	7	6	5	4	3	2	1
DBX387	Grinding: Status of the grinding functions 1 ... 8 ($AC_IN_KEY_G_RUN_IN[1...8]) /LIS3sl/							
	8	7	6	5	4	3	2	1

4.11.19 DB21 - DB30, control signals from the channel (3)

Table 4-85 DB21 - DB30, signals from the channel

DB21 - DB30	Signals from the channel (NC → PLC)							
Byte	**Bit 7**	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
DBW388	Active transformation number							
DBB390	Grinding: Release status input signals (1 ... 8) ($AC_IN_KEY_G_ISENABLE[1...8])							
	8	7	6	5	4	3	2	1

DB21 - DB30	Signals from the channel (NC → PLC)							
Byte	**Bit 7**	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
DBB391	Grinding: Status of the grinding functions (1 ... 8) ($AC_IN_KEY_G_RUN_OUT[1...8])							
	8	7	6	5	4	3	2	1
DBB392	Coordinate system for Cartesian manual travel							
DBB393								
DBB394								
DBB395								

4.12 Axis/spindle signals

4.12.1 DB31 - DB61, signals to the axis/spindle

Table 4-86 DB31 - DB61, signals to the axis/spindle

<table>
<tr><td>DB31 - DB61</td><td colspan="8">Signals to the axis/spindle (PLC → NC)</td></tr>
<tr><td>Byte</td><td>Bit 7</td><td>Bit 6</td><td>Bit 5</td><td>Bit 4</td><td>Bit 3</td><td>Bit 2</td><td>Bit 1</td><td>Bit 0</td></tr>
<tr><td rowspan="2">DBB0
Axis and spindle</td><td colspan="8">Feedrate override request /Z1-V1/</td></tr>
<tr><td>H</td><td>G</td><td>F</td><td>E</td><td>D</td><td>C</td><td>B</td><td>A</td></tr>
<tr><td rowspan="2">DBB1
Axis and spindle</td><td rowspan="2">Override active
/Z1-V1/</td><td colspan="2">Position measuring system
/Z1-A2/</td><td rowspan="2">Follow-up mode
/Z1-A2/</td><td rowspan="2">Axis/spindle disable
/Z1-A2/</td><td rowspan="2">Sensor for fixed stop
/Z1-F1/</td><td rowspan="2">Acknowledge fixed stop reached
/Z1-F1/</td><td rowspan="2">Drive test travel enable
/Z1-A2/</td></tr>
<tr><td>2</td><td>1</td></tr>
<tr><td rowspan="2">DBB2
Axis and spindle</td><td colspan="4">Home position value /Z1-R1/</td><td rowspan="2">Clamping in progress
/Z1-A3/</td><td rowspan="2">Delete distance-to-go / spindle reset
/A2, S1/</td><td rowspan="2">Controller enable
/Z1-A2/</td><td rowspan="2">Output cam activation
/Z2-N3/</td></tr>
<tr><td>4</td><td>3</td><td>2</td><td>1</td></tr>
<tr><td rowspan="2">DBB3
Axis and spindle</td><td rowspan="2">Program test Axis/ Spindle Enable</td><td rowspan="2">Velocity / spindle speed limitation
/Z1-A3/</td><td colspan="4">Activate fixed feedrate
/Z1-V1/</td><td rowspan="2">Travel to fixed stop enabled
/Z1-F1/</td><td rowspan="2">Accept offset external WO
/Z1-K2/</td></tr>
<tr><td>4</td><td>3</td><td>2</td><td>1</td></tr>
<tr><td rowspan="2">DBB4
Axis and spindle</td><td colspan="2">Traversing keys
/Z2-H1/</td><td rowspan="2">Rapid traverse override
/Z2-H1/</td><td rowspan="2">Traversing key lock
/Z2-H1/</td><td rowspan="2">Feed stop / spindle stop
/Z1-V1/</td><td colspan="3">Activate handwheel /Z2-H1/</td></tr>
<tr><td>Plus</td><td>Minus</td><td>C</td><td>B</td><td>A</td></tr>
<tr><td rowspan="2">DBB5
Axis and spindle</td><td colspan="8">Machine function request /Z2-H1/</td></tr>
<tr><td></td><td>Continuous traversing</td><td>INCvar</td><td>INC10000</td><td>INC1000</td><td>INC100</td><td>INC10</td><td>INC1</td></tr>
<tr><td rowspan="2">DBB6</td><td colspan="8">OEM axis signals</td></tr>
<tr><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td></tr>
<tr><td rowspan="2">DBB7</td><td colspan="8">OEM axis signals</td></tr>
<tr><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td>Handwheel direction of rotation inverted
/Z1-H2/</td></tr>
<tr><td rowspan="2">DBB8</td><td rowspan="2">Request PLC axis/ spindle
/Z2-K5/</td><td rowspan="2"></td><td rowspan="2"></td><td rowspan="2">Channel assignment changed
/Z2-K5/</td><td colspan="4">NC axis / spindle channel assignment
/Z2-K5/</td></tr>
<tr><td>D</td><td>C</td><td>B</td><td>A</td></tr>
</table>

DB31 - DB61	Signals to the axis/spindle (PLC → NC)							
Byte	**Bit 7**	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
DBB9					Parameter set change inhibited /Z1-A2/	Controller parameter set /Z1-A2/		
						C	B	A
DBB10								REPOS DELAY
DBB11								Start brake test /FBSIsl/
DBB12 Axis	Delay reference point approach Z1-R1/			Modulo rotary axes: Activate traversing range limits	2nd software limit switch /Z1-A3/		Hardware limit switch /Z1-A3/	
					Plus	Minus	Plus	Minus
DBB13 Axis					JOG to position	JOG fixed point approach		
						2	1	0
DBB14 Axis							Activate program test	Suppress program test
DBB15 Axis								
DBB16 Spindle	S value Delete /Z1-S1/	No n-monitoring for gear change /Z1-S1/	Resynchronize spindle /Z1-S1/		Gear is changed over /Z1-S1/	Actual gear stage /Z1-S1/		
			Measuring system 2	Measuring system 1		C	B	A
DBB17 Spindle		Invert M3/M4 /Z1-S1/	Resynchronize spindle during positioning /Z1-S1/					Feedrate correction spindle valid /Z1-S1/
			Measuring system 2	Measuring system 1				
DBB18 Spindle	Oscillation rotation direction /Z1-S1/		Oscillation enable /Z1-S1/	Oscillation by PLC /Z1-S1/				
	Left	Right						
DBB19 Spindle	Spindle override /Z1-V1/							
	H	G	F	E	D	C	B	A
DBB20 Operating mechanism			Open motor holding brake /FBSIsl/				Ramp-function generator disable [1]	
DBB21 Operating mechanism	Pulse enable	Speed controller integrator disable /Z1-A2/	Motor being selected /Z1-A2/	Motor/drive data set: Request interface (interface definition: DB31, …DBX130.0 - 4)				
				E	D	C	B	A

DB31 - DB61	Signals to the axis/spindle (PLC → NC)							
Byte	**Bit 7**	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
DBB22 Safety Integrated				SG selection		Acknowledgement, communication failure	SBH deselection	SBH/SG deselection
				Bit 1	Bit 0			
DBB23 Safety Integrated	Test stop selection		Close SINAMICS brake /FBSIsl/	Activate SE 2		Gear ratio selection		
						Bit 2	Bit 1	Bit 0
DBB24	Master/ slave: On		Setpoint switchover: Switchover /Z3-S9/	Torque compensatory controller on	Switch on collision protection /Z3-TE6/	CC_Slave axis: Suppress coupling /Z3-TE6/	Control axis	Stepper motor: Rotation monitoring
DBB25								Activate dynam. backlash compensation /FB2/K3/
DBB26 Grinding				Following axis superimposition: Enable /Z3-M3/	Compensatory controller on			
DBB27 Grinding	Stop				Resume			
	HIAxMove	Corr.	DEPBCS	DEPMCS	HIAxMove	Corr.	DEPBCS	DEPMCS
DBB28 Reciprocating grinding	PLC controls axis /Z2-P5/ /FB2/P2/	Stop along braking ramp /Z2-P5/ /FB2/P2/	Stop at next reversal point /Z2-P5/	Alter reversal point /Z2-P5/	Set reversal point /Z2-P5/	Continue /Z2-P5/ /FB2/P2/	Reset /Z2-P5/ /FB2/P2/	External oscillation reversal /Z2-P5/
DBB29 Couplings			Lock automatic synchronization	Start gantry synchronization				
DBB30 Technology	Reserved							
DBB31 Technology	Delete synchronism correction /FB3/M3/	Track synchronism /FB3/M3/	Disable synchronization /FB3/M3/	Re-synchronization /FB3/M3/				
DBB32 Safety Integrated			Deselect external STOP_E	Deselect external STOP_D	Deselect external STOP_C	Deselect external STOP_A		

DB31 - DB61	Signals to the axis/spindle (PLC → NC)							
Byte	**Bit 7**	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
DBB33	SG correction selection / override							
Safety Integrated	Bit 3	Bit 2	Bit 1	Bit 0				
DBB34							Setpoint limitation	Setpoint limitation
DBB35								
DBB36 Technology								
DBB37								
DBB38								
DBB39								
DBB40 - DBB55								
DBB56 PLC → operating software						Spindle internal clamping	Spindle speed display	Separate feed drive coupled as C axis
DBB57								
DBB58	Reserved							
DBB59								

[1)] Only when cyclic interface between NC and drive is operated in the "611U compatibility mode".

Note

DBX8.4 is automatically reset after the assignment is executed

4.12.2 DB31 - DB61, signals from the axis/spindle

Table 4-87 DB31 - DB61, signals from the axis/spindle

DB31 - DB61	Signals from the axis/spindle (NC → PLC)							
Byte	**Bit 7**	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
DBB60 Axis and spindle	Position reached with exact stop /Z1-B1/		Referenced/synchronized position measuring system /Z1-A3, FBSIsl/		Encoder limit frequency exceeded /Z1-A3/		NCU_Link axis active /Z2-B3/	Spindle / no axis /Z1-S1/
	fine	coarse	2	1	2	1		

DB31 - DB61	Signals from the axis/spindle (NC → PLC)							
Byte	**Bit 7**	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
DBB61 Axis and spindle	Current controller active /Z1-A2/	Speed controller active /Z1-A3, FBSIsl/	Position controller active /Z1-A3, FBSIsl/	Axis/spindle stationary (n < n_{min})	Follow-up mode active /Z1-A2/	Axis ready /Z2-B3/	Axial alarm /Z2-P2/	Drive test travel request /Z1-A2/
DBB62	Axis container rotation active /Z2-B3/	Force of fixed stop limited /Z1-F1/	Fixed stop reached /Z1-F1, FBSIsl/	Activate travel to fixed stop /Z1-F1/	Measurement active /Z2-M5/	Revolutional feedrate active /Z1-V1/	Handwheel override active /Z2-H1/	Software cam active /Z2-N3/
DBB63	Stop				Axis/spindle disable active /	Axis stop active /Z2-P2/	PLC controls axis /Z2-P2, FBSIsl/	Reset executed /Z2-P2/
	HIAxMove active	Corr. active	DEPBCS active	DEPMCS active				
DBB64 Axis and spindle	Travel command /Z2-H1, FBSIsl/		Travel request			Handwheel active /Z2-H1/		
	Plus	Minus	Plus	Minus		C	B	A
DBB65	Active machine function /Z2-H1/							
Axis and spindle		Continuous traversing /Z2-A3/	INCvar	INC10000	INC1000	INC100	INC10	INC1
DBB66	Reserved OEM axis signals /Z3-TE6/							
Axis and spindle								
DBB67								Handwheel direction of rotation inversion active /Z2-H1/
DBB68	PLC axis/spindle /Z2-K5/	Neutral axis/spindle /Z2-K5/	Axis interchange possible /Z2-K5/	New type requested from PLC /Z2-K5/	Current channel assignment of the NC axis / spindle in channel /Z2-K5/			
					D	C	B	A
DBB69	NCU number in the NCU link group					Controller parameter set servo		
	E	D	C	B	A	C	B	A
DBB70			DRV Safety Integrated active with SIC/SCC	NC Safety Integrated active		REPOS delay acknowledgement	REPOS offset valid	REPOS offset
DBB71	PLC axis permanently assigned		Position restored					Brake test active
			Encoder 2	Encoder 1				

DB31 - DB61	Signals from the axis/spindle (NC → PLC)							
Byte	**Bit 7**	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
DBB72 Operating software → PLC								REPOS delay
DBB73 Operating software → PLC								
DBB74 Axis				Modulo rotary axes: Traversing range limits active				
DBB75 Axis	JOG position reached	JOG travel to position active	JOG approach fixed point reached /FB1/K2/			JOG approach fixed point active		
			2	1	0	2	1	0
DBB76 Axis	Rounding axis in position	Indexing axis in position /FB2/T1/	Positioning axis /FB2/P2/	Path axis /FB1/K1/				Lubrication pulse /FB1/A2/
DBB77 Axis								Collision avoidance: Velocity reduction
DBD78 Axis	F function (REAL) for positioning axis /Z1-V1/							
DBB82 Spindle					Change gear /Z1-S1/	Setpoint gear unit stage /Z1-S1/		
						C	B	A
DBB83 Spindle	Actual direction of rotation clockwise /FB1/S1/	Speed monitoring /FB1/V1/	Spindle in set range /FB1/S1/	Support range limits violated	Geometry monitoring /FB2/W4/	Setpoint speed /FB1/S1/		Speed limit exceeded /FB1/S1/
						Increased	limited	
DBB84 Spindle	Active spindle mode /FB1/S1/				Tapping with compensation chuck active /FB1/S1/	CLGON active /Z1-S8/	GWPS active /FB2/W4/	Constant cutting velocity active /FB1/S1/
	Control mode	Oscillation mode	Positioning mode	Synchronous mode				
DBB85 Spindle			Spindle actually reached in position /FB1S1/					Tool with dynamic response limitation

DB31 - DB61	Signals from the axis/spindle (NC → PLC)							
Byte	**Bit 7**	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
DBW86 Spindle	M function (INT) for spindle (M3, M4, M5, M19, M70 or defined via MD) /FB1/S1/							
DBD88 Spindle	S function (REAL) for spindle /FB1/S1/							
DBB92 Drive	Drive operation enabled		Motor holding brake opened	Drive-autonomous motion active[1]			Ramp-function-generator disable active /FB1/A2/	
DBB93 Drive	Pulses enabled /Z1-A3/	Speed controller integrator disabled /Z1-A2/	Drive ready /Z1-A2/	Motor/drive data set: Display interface (interface definition: DB31, …DBX130.0 - 4)				
				E	D	C	B	A
DBB94 Drive	Variable signaling function[2]	$n_{act} = n_{set}$ /Z1-A2/	$\|n_{act}\| < n_x$ /Z1-A2/	$\|n_{act}\| < n_{min}$ /Z1-A2/	$M_d < M_{dx}$ /Z1-A2/	Run-up completed	Temperature pre-alarm	
							Heat sink	Motor
DBB95 Drive	Warning of warning class C is pending				ESR: Regenerative operation speed lower than minimum (p2161) /Z3-R3/	ESR: Reaction triggered or generator operation active (r0887.12) /Z3-R3/	ESR: DC-link undervoltage (p1248) /Z3-R3/	
DBB96	Master/slave: Coupling active /Z3-TE3/		Setpoint switchover active /Z3-S9/	Master/slave:			Axis control active	Stepper motor: Rotation monitoring error
				Compensatory controller active	Coarse speed difference	Fine speed difference		
DBB97					OEM application			
					New offset	Activate mirroring /Z3-TE6/	MCS coupling active /Z3-TE6/	Slave axis /Z3-TE6/
DBB98 Synchronous spindle	ESR response initiated	Acceleration warning threshold reached /Z3-M3/	Velocity warning threshold reached /Z3-M3/	Overlaid movement /Z2-S3/		Actual value coupling /Z2-S3/	Synchronism /Z2-S3/	
							coarse	fine
DBB99 Synchronous spindle		Max. acceleration reached	Max. velocity reached	Synchronization in progress	Axis accelerating /Z3-M3/	Synchronism correction implemented	Following spindle/axis active /Z2-S3/	Leading spindle/axis active /Z2-S3/

DB31 - DB61	Signals from the axis/spindle (NC → PLC)							
Byte	**Bit 7**	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
DBB100 Reciprocating grinding	Oscillation active /Z2-P5/	Oscillation movement active /Z2-P55/	Sparking out active /Z2-P5/	Error during oscillation movement /Z2-P5/	Oscillation cannot be started /Z2-P5/	External oscillation reversal active /Z2-P5/		
DBB101 Gantry	Gantry axis /Z3-G1/	Gantry guide axis /Z3-G1/	Gantry grouping is synchronous /Z3-G1/	Gantry synchronization ready to start /Z3-G1/	Gantry alarm limit exceeded /Z3-G1/	Gantry trip limit exceeded /Z3-G1/		
DBB102		Position measuring system activated /Z1-A3, FBSIsl/			Clamping tolerance exceeded /FB1/A3/			Dynam. backlash compensation active
		2	1					
DBB103			Synchronous operation 2					Synchronism correction is taken into account
			coarse	fine				
DBB104 Grinding	Active infeed axis /Z2-P5/							
	8	7	6	5	4	3	2	1
DBB105 Grinding	Active infeed axis /Z2-P5/							
	16	15	14	13	12	11	10	9
DBB106 Grinding	Active infeed axis /Z2-P5/							
	24	23	22	21	20	19	18	17
DBB107 Grinding	Active infeed axis /Z2-P5/							
		31	30	29	28	27	26	25
DBB108 Safety Integrated	Axis safely referenced			Communication failure not acknowledged	CRC error	Status pulses have been cancelled	CRC or sign-of-life error	SBH/SG active
DBB109 Safety Integrated	Cam position							
	SN 4-	SN 4+	SN 3-	SN 3+	SN 2-	SN 2+	SN 1-	SN 1+
DBB110 Safety Integrated			n < nx	Active SG			SBH active	
				B	A			
DBB111 Safety Integrated	Stop E Active	Stop D Active	Stop C Active	Stop A/B Active				

DB31 - DB61	Signals from the axis/spindle (NC → PLC)							
Byte	**Bit 7**	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
DBB112	Cam range for cam track 1							
Safety Integrated								
DBB113	Cam range for cam track 2							
Safety Integrated								
DBB114	Cam range for cam track 3							
Safety Integrated								
DBB115	Cam range for cam track 4							
Safety Integrated								
DBB116	Reserved							
Safety Integrated								
DBB117	Reserved				Cam track			
Safety Integrated					4	3	2	1
DBB118	Cam range bit for cam track 1							
Safety Integrated	7	6	5	4	3	2	1	0
DBB119	Cam range bit for cam track 1							
Safety Integrated		14	13	12	11	10	9	8
DBB120	Cam range bit for cam track 2							
Safety Integrated	7	6	5	4	3	2	1	0
DBB121	Cam range bit for cam track 2							
Safety Integrated		14	13	12	11	10	9	8
DBB122	Cam range bit for cam track 3							
Safety Integrated	7	6	5	4	3	2	1	0
DBB123	Cam range bit for cam track 3							
Safety Integrated		14	13	12	11	10	9	8
DBB124	Cam range bit for cam track 4							
Safety Integrated	7	6	5	4	3	2	1	0
DBB125	Cam range bit for cam track 4							
Safety Integrated		14	13	12	11	10	9	8
DBB126								

DB31 - DB61	Signals from the axis/spindle (NC → PLC)							
Byte	**Bit 7**	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
DBB127								
DBB128							Program test	
Operating software → PLC							activate	suppress
DBB129								
DBB130	Motor/drive data set: Request and display interface valid (see bit 0 - 4) /FB1/A2/			Motor/drive data set: Formatting interface (Request: DB31, …DBX21.0 - 4, display: DBX93.0 - 4) /FB1/A2/				
				E	D	C	B	A
DBB131								
DBB132	Sensor configuration /FB1/S1/							
		Sensor S6 available	Sensor S5 available – angular position, motor shaft	Sensor S4 available piston end position			Sensor S1 available (clamped state)	Sensors available
DBB133	Sensor configuration /FB1/S1/							
						Status value is generated, speed limitation p5043 is active /FB1/S1/		
DBW134	Clamping state (sensor S1) /FB1/S1/							
DBW136	Analog value: Clamping state (sensor S1) /FB1/S1/							

DB31 - DB61	Signals from the axis/spindle (NC → PLC)							
Byte	**Bit 7**	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
DBB138	Status digital sensors /FB1/S1/							
			Sensor S5 angular position, motor shaft	Sensor S4, piston end position				
DBB139	Status digital sensors							

1) With SINAMICS valid for NC 62.07 and higher when using a 611U telegram type

2) With SINAMICS valid for SW2.6 and higher

4.12.3 DB31 - DB61, Safety Control Channel (SCC)

Table 4-88 DB31 - DB61, axis signals: Safety Control Channel (SCC)

DB31 - DB61 /FBSIsl/	Signals to the axis/spindle							
Byte	**Bit 7**	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
	SCC (PLC → drive)							
DBB140 Safety Integrated								Test stops for extended functions
DBB141 Safety Integrated								
DBB142 Safety Integrated								
DBB143 Safety Integrated			External brake closed	Test sequence 1 or 2	Direction of rotation	Test with brake 1 or 2	Start brake test	Select safe brake test
DBB144 ... DBB163 Safety Integrated								

4.12.4 DB31 - DB61, Safety Info Channel (SIC)

Table 4-89 DB31 - DB61, axis signals: Safety Info Channel (SIC)

DB31 - DB61 /FBSIsl/	Signals from axis/spindle							
Byte	**Bit 7**	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
	SIC (drive → PLC)							
DBB164 Safety Integrated	Safety error active	ESR requested				Bit 1 safely-limited speed limit value	Bit 0 safely-limited speed limit value	
DBB165 Safety Integrated	Safety error with Stop A	Safely-limited speed selected	Safe operating stop selected	Safely-limited speed active	Safe operating stop active	Safe stop 2	Safe stop 1	Safe torque off
DBB166 Safety Integrated			Test stop requested	Test stop active			Safe direction negative	Safe direction positive
DBB167 Safety Integrated	Safely limited position is selected			Bit 0 for safely-limited position range				
DBB168 Safety Integrated	Speed limit							
DBB172 Safety Integrated	Acceptance test Safely limited position is active	Acceptance test Safely limited position is selected				REPOS delay acknowledgement	REPOS offset valid	REPOS offset
DBB173 Safety Integrated	Load torque negative sign	Close external brake	Brake test					
			completed	OK	Active	with brake 2	Setpoint input during SBT in the drive	Safe Brake Test (SBT)
DBB174 ... DBB187								

4.13 Interface to the tool management

4.13.1 DB71, interface for loading/unloading the magazine

Table 4-90 DB71, interface for loading/unloading the magazine

DB71 /FBWsl/	Loading/unloading positions (NC → PLC)							
Byte	**Bit 7**	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
DBB0	Interface (SS) active							
	SS8	SS7	SS6	SS5	SS4	SS3	SS2	SS1
DBB1								
	SS16	SS15	SS14	SS13	SS12	SS11	SS10	SS9
DBB2	Standard end acknowledgement							
	SS8	SS7	SS6	SS5	SS4	SS3	SS2	SS1
DBB3								
	SS16	SS15	SS14	SS13	SS12	SS11	SS10	SS9
DBBn + 0	Reserved		Multitool positioning	NC progr. positions magazine	Magazine positioning	Relocating	Unloading	Loading
DBBn + 1	Data in extended area (DB1071)	Reserved						Acknowledgement status = 3
DBBn + 2	Assigned channel (8-bit INT)							
DBBn + 3	Tool management number (8-bit INT)							
DBBn + 4	Reserved (free parameter 1 (DWord))							
DBBn + 8	Reserved (free parameter 2 (DWord))							
DBDn + 12	Reserved (free parameter 3 (DWord))							
DBWn + 16	Identifier for loading/unloading point (INT), (fixed value 9999)							
DBWn + 18	Location no. of load/unload point (INT)							
DBWn + 20	Magazine no. (Source) for unloading/relocating/positioning (INT)							
DBWn + 22	Location no. (Source) for unloading/relocating/positioning (INT)							
DBWn + 24	Magazine no. (Target) for loading/relocating/positioning (INT)							
DBWn + 26	Location no. (Target) for loading/relocating/positioning (INT)							
DBWn + 28	Reserved							Loading/unloading without moving magazine
DBWn + 29	Reserved							

See also: Table 4-93 DB1071, interface for loading/unloading the magazine MultitoolDB1071Multitool magazine interface (Page 817)

Starting addresses of loading/unloading points:

Loading/unloading point 1: n = 4
Loading/unloading point 2: n = 34
Loading/unloading point 3: n = 64
Loading/unloading point 4: n = 94

Loading interface 1 is responsible for loading/unloading in (all) spindles/toolholders and for relocating tools and for positioning at any locations (e.g. buffer location).

Loading and unloading manual tools is always realized via loading interface 1.

4.13.2 DB72, interface for the spindle as change position

Table 4-91 DB72, interface for the spindle as change position

DB72 /FBWsl/	**Signals from the spindle (NC → PLC)**							
Byte	**Bit 7**	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
DBB0	Interface (SS) active							
	SS8	SS7	SS6	SS5	SS4	SS3	SS2	SS1
DBB1								
	SS16	SS15	SS14	SS13	SS12	SS11	SS10	SS9
DBB2	Standard end acknowledgement							
	SS8	SS7	SS6	SS5	SS4	SS3	SS2	SS1
DBB3								
	SS16	SS15	SS14	SS13	SS12	SS11	SS10	SS9
DBBn + 0	Spindle tool remains in the spindle	Replace manual tool	Insert manual tool	Old tool in BL no. (n + 42)	T0	Prepare change	Perform change (initiate: M06)	Obligatory change
DBBn + 1	Data in extended area (DB1072)	Reserved						Acknowledgement status = 3
DBBn + 2	Assigned channel (8-bit INT)							
DBBn + 3	Tool management no. (8-bit INT)							
DBDn + 4	User parameter 1 (DWord)							
DBDn + 8	User parameter 2 (DWord)							
DBDn + 12	User parameter 3 (DWord)							
DBWn + 16	Buffer identifier (INT), (fixed value 9998) (corresponds to "Target position for new tool")							
DBWn + 18	Relative location (target) in buffer magazine (INT)							
DBWn + 20	Magazine no. (Source) for new tool (INT)							
DBWn + 22	Location no. (Source) for new tool (INT)							
DBWn + 24	Magazine no. (Target) for old tool (INT)							
DBWn + 26	Location no. (Target) for old tool (INT)							
DBWn + 28	Tool new: Location type (INT)							

DB72 /FBWsl/	Signals from the spindle (NC → PLC)							
Byte	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
DBWn + 30	Tool new: Size left (INT)							
DBWn + 32	Tool new: Size right (INT)							
DBWn + 34	Tool new: Size top (INT)							
DBWn + 36	Tool new: Size bottom (INT)							
DBBn + 38	Tool status for new tool							
	Tool has been in use	Tool fixed-location-coded	Tool being changed	Prewarning limit reached	Measure tool	Tool disabled	Tool released	Active tool
DBBn + 39	Tool status for new tool							
	Manual tool	1:1 exchange	Reserved	Master tool	Tool to be loaded	Tool to be unloaded	Locked, but ignore	Identifier for tools
DBWn + 40	Tool new: Internal T no. (INT)							
DBWn + 42	If DBX (n + 0.4) = 1, the buffer location of the old tool is entered here.							
DBWn + 44	Original magazine of the new tool							
DBWn + 46	Original location of the new tool							

See also: Table 4-94 DB1072, interface for the spindle: MultitoolDB1072Multitool spindle interface (Page 817)

Start addresses of the buffer

Spindle 1: n = 4
Spindle 2: n = 52

4.13.3 DB73, interface for the turret

Table 4-92 DB73, interface for the turret

DB73 /FBWsl/	Signals for the turret (NC → PLC)							
Byte	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
DBB0	Interface (SS) active							
	SS8	SS7	SS6	SS5	SS4	SS3	SS2	SS1
DBB1								
	SS16	SS15	SS14	SS13	SS12	SS11	SS10	SS9
DBB2	Standard end acknowledgement							
	SS8	SS7	SS6	SS5	SS4	SS3	SS2	SS1
DBB3								
	SS16	SS15	SS14	SS13	SS12	SS11	SS10	SS9

DB73 /FBWsl/	Signals for the turret (NC → PLC)							
Byte	**Bit 7**	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
DBBn	Reserved	Unload manual tool	Reserved		T0	Reserved	Make change (initiation: T No.)	Obligatory change
DBBn + 1	Data in extended area (DB1073)	Reserved						Acknowledgement status = 3
DBBn + 2	Assigned channel (8-bit INT)							
DBBn + 3	Tool management no. (8-bit INT)							
DBDn + 4	User parameter 1 (DWord)							
DBDn + 8	User parameter 2 (DWord)							
DBDn + 12	User parameter 3 (DWord)							
DBWn + 16	Reserved							
DBWn + 18	Reserved							
DBWn + 20	Magazine no. of the turret (INT)							
DBWn + 22	Location no. of new tool (INT)							
DBWn + 24	Magazine no. of old tool							
DBWn + 26	Location no. of old tool (INT)							
DBWn + 28	Tool new: Location type (INT)							
DBWn + 30	Tool new: Size left (INT)							
DBWn + 32	Tool new: Size right (INT)							
DBWn + 34	Tool new: Size top (INT)							
DBWn + 36	Tool new: Size bottom (INT)							
DBBn + 38	Tool status for new tool							
	Manual tool	1:1 exchange		Master tool	Tool to be loaded	Tool to be unloaded	Locked, but ignore	Identifier for tools
DBBn + 39	Tool status for new tool							
	Tool has been in use	Tool fixed-location-coded	Tool being changed	Prewarning limit reached	Measure tool	Tool disabled	Tool released	Active tool
DBWn + 40	Tool new: Internal T no. (INT)							
DBWn + 42	Original location of new tool in this circular magazine							

See also: Table 4-95 DB1073, interface for the turret: MultitoolDB1073Multitool turret interface (Page 820)

Start addresses of the turret:

Turret, 1: n = 4
Turret, 2: n = 48

4.13.4 DB1071, interface for loading/unloading the magazine Multitool

Table 4-93 DB1071, interface for loading/unloading the magazine Multitool

DB1071 /FBWsl/	Loading/unloading positions (NC → PLC)							
Byte	**Bit 7**	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
DBWn + 0	Type of distance coding of the multitool (corresponds to $TC_MTP_KD)							
	1 = location number 2 = distance 3 = angle							
DBWn + 2	Multitool location number							
	Number of locations of the multitool.							
DBWn + 4	Multitool location distance							
	Distance of the MT location to be positioned from the reference location (real value), corresponding to the distance coding							
DBWn + 8	Multitool number							
	Internal T number of the multitool							
DBWn + 10	Multitool location number							
	Location number within the multitool to which the system positions to.							
DBWn + 12	Tool holder							
	Spindle or toolholder number							
DBWn + 14	Reserved							
DBWn + 16	Reserved							
DBWn + 18	Reserved							

4.13.5 DB1072, interface for the spindle: Multitool

Table 4-94 DB1072, interface for the spindle: Multitool

DB1072 /FBWsl/	Spindle (NC → PLC)							
Byte	**Bit 7**	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
DBWn + 0	Distance coding							
	Type of distance coding of the multitool (corresponds to $TC_MTP_KD) 1 = location number 2 = distance 3 = angle							
DBWn + 2	Multitool location number							
	Number of locations of the multitool.							

DB1072 /FBWsl/	Spindle (NC → PLC)							
Byte	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
DBWn + 4	Multitool location distance							
	Distance of the MT location to be positioned from the reference location (real value), corresponding to the distance coding							
DBWn + 8	Multitool number (new tool)							
	Internal T number of the new multitool							
DBWn + 10	Multitool location number (new tool)							
	Location number at which the new tool is located within the multitool							
DBWn + 12	Multitool number (old tool)							
	Internal T number of the old multitool. The T number is entered here if the preparation or the change to a tool is realized within the same multitool (which, due to a previous change, is located on the toolholder). It is identical to DB1072.DBW(n + 8)							
DBWn + 14	Multitool location number (old tool)							
	Location number at which the old tool is located within the multitool The location number at which the old tool is located is entered here if the preparation or the change to a tool is realized within the same multitool (which, due to a previous change, is located on the toolholder).							
DBWn + 16	Location type							
	Location type of the newly programmed tool in the multitool							
DBWn + 18	Tool new: Size left							
	Specification of the tool size to the left in half locations of the newly programmed tool in the multitool							
DBWn + 20	Tool new: Size right							
	Specification of the tool size to the right in half locations of the newly programmed tool in the multitool							
DBWn + 22	Tool new: Size top							
	Specification of the tool size to the top in half locations of the newly programmed tool in the multitool							
DBWn + 24	Tool new: Size bottom							
	Specification of the tool size to the bottom in half locations of the newly programmed tool in the multitool							

DB1072 /FBWsl/	Spindle (NC → PLC)							
Byte	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
DBWn + 26	Tool status for new tool							
	Tool status of the programmed tool in the multitool (corresponds to parameter $TC_TP8[T_No]) Bit 0 = active tool Bit 1: Tool enabled Bit 2: Tool disabled Bit 3: Measure tool Bit 4: Prewarning limit reached Bit 5: Tool is being changed Bit 6: Tool is fixed-location-coded Bit 7: Tool has been in use Bit 8: Tool in the buffer Bit 9: Ignore locked Bit 10: To be unloaded Bit 11: To be loaded Bit 12: Master tool Bit 13: Reserved Bit 14: 1:1 exchange Bit 15: Manual tool							
DBWn + 28	Tool new: Internal T number of the NC							
	Display of the internal T number of the NC for the new spindle tool of the programmed tool in the multitool							
DBWn + 30	Tool holder							
	Spindle or toolholder number to which the change refers							
DBWn + 32	Original magazine of the new tool							
	Owner magazine of the new tool (corresponds to NC variables $A_MYMN[T no.]) If the new tool is located in the magazine, then this value is identical with DB72.DBW (n + 20). If the new tool is located in the buffer (e.g. gripper), then the original magazine number is entered here. The new tool has the same owner location as the multitool onto which it is loaded.							
DBWn + 34	Original location of the new tool							
	Owner location of the new tool (corresponds to NC variables $A_MYMLN[T no.]) If the new tool is located in the magazine, then this value is identical with DB72.DBW (n + 22). If the new tool is located in the buffer (e.g. gripper), then the original magazine number is entered here. The new tool has the same owner location as the multitool onto which it is loaded.							
DBWn + 36 - DBWn + 48	Reserved							

See also: Table 4-91 DB72, interface for the spindle as change positionDB72Spindle interface (Page 814)

4.13.6 DB1073, interface for the turret: Multitool

Table 4-95 DB1073, interface for the turret: Multitool

DB1073	Turret (NC → PLC)							
Byte	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
DBWn + 0	Distance coding							
	Type of distance coding of the multitool (corresponds to $TC_MTP_KD) 1 = location number 2 = distance 3 = angle							
DBWn + 2	Multitool location number							
	Number of locations of the multitool.							
DBWn + 4	Multitool location distance							
	Distance of the MT location to be positioned from the reference location (real value), corresponding to the distance coding							
DBWn + 8	Multitool number (new tool)							
	Internal T number of the new multitool							
DBWn + 10	Multitool location number (new tool)							
	Location number at which the new tool is located within the multitool							
DBWn + 12	Multitool number (old tool)							
	Internal T number of the old multitool The T number is entered here if the preparation or the change to a tool is realized within the same multitool (which, due to a previous change, is located on the toolholder). It is identical to DB1073.DBW(n + 8)							
DBWn + 14	Multitool location number (old tool)							
	Location number at which the old tool is located within the multitool The location number at which the old tool is located is entered here if the preparation or the change to a tool is realized within the same multitool (which, due to a previous change, is located on the toolholder).							
DBWn + 16	Location type							
	Location type of the newly programmed tool in the multitool							
DBWn + 18	Tool new: Size left							
	Specification of the tool size to the left in half locations of the newly programmed tool in the multitool							
DBWn + 20	Tool new: Size right							
	Specification of the tool size to the right in half locations of the newly programmed tool in the multitool							
DBWn + 22	Tool new: Size top							
	Specification of the tool size to the top in half locations of the newly programmed tool in the multitool							
DBWn + 24	Tool new: Size bottom							
	Specification of the tool size to the bottom in half locations of the newly programmed tool in the multitool							

DB1073	Turret (NC → PLC)							
Byte	**Bit 7**	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
DBWn + 26	Tool status for new tool							
	Tool status of the programmed tool in the multitool (corresponds to parameter $TC_TP8[T_No]) Bit 0 = active tool Bit 1: Tool enabled Bit 2: Tool disabled Bit 3: Measure tool Bit 4: Prewarning limit reached Bit 5: Tool is being changed Bit 6: Tool is fixed-location-coded Bit 7: Tool has been in use Bit 8: Tool in the buffer Bit 9: Ignore locked Bit 10: To be unloaded Bit 11: To be loaded Bit 12: Master tool Bit 13: Reserved Bit 14: 1:1 exchange Bit 15: Manual tool							
DBWn + 28	Tool new: Internal T number of the NC							
	Display of the internal T number of the NC for the new spindle tool of the programmed tool in the multitool							
DBWn + 30	Tool holder							
	Spindle or toolholder number to which the change refers							
DBWn + 32	Original magazine of the new tool							
	Owner magazine of the new tool (corresponds to NC variables $A_MYMN[T no.]) If the new tool is located in the magazine, then this value is identical with DB73.DBW(n + 20). If the new tool is located in the buffer (e.g. gripper), then the original magazine number is entered here. The new tool has the same owner location as the multitool onto which it is loaded.							
DBWn + 34	Original location of the new tool							
	Owner location of the new tool (corresponds to NC variables $A_MYMLN[T no.]) If the new tool is located in the magazine, then this value is identical with DB73.DBW(n + 22). If the new tool is located in the buffer (e.g. gripper), then the original magazine number is entered here. The new tool has the same owner location as the multitool onto which it is loaded.							
DBWn + 36 - DBWn + 48	Reserved							

See also: Table 4-92 DB73, interface for the turretDB73Turret interface (Page 815)

4.14 Signals from/to the machine control panel and the handheld unit

4.14.1 DB77, signals from/to the MCP and the HHU

Table 4-96 DB77, signals from/to the MCP and the HHU

DB77	Signals from/to the MCP and the HHU (GD communication)							
Byte	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
DBB0 - DBB7	Input signals from MCP 1 to the PLC, MPI bus							
DBB8 - DBB15	Output signals from MCP 1 to the PLC, MPI bus							
DBD16	Status, send MCP 1, MPI bus							
DBD20	Status, receive MCP 1, MPI bus							
DBB24 - DBB31	Input signals from MCP 2 to the PLC, MPI bus							
DBB32 - DBB39	Output signals from MCP 2 to the PLC, MPI bus							
DBD40	Status, send MCP 2, MPI bus							
DBD44	Status, receive MCP 2, MPI bus							
DBB48 - DBB53	Input signals from the HHU to the PLC, MPI bus							
DBB60 - DBB79	Output signals from the HHU to the PLC, MPI bus							
DBD80	Status, send HHU, MPI bus							
DBD84	Status, receive HHU, MPI bus							

4.15 Signals for Ctrl-Energy

4.15.1 DB1000, energy-saving profiles

Table 4-97 DB1000, energy-saving profiles

DB1000	Ctrl-Energy (operating software → PLC)							
Byte	**Bit 7**	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
DBBn + 0	Control signals							
							Set time to prewarning limit	Immediately activate energy-saving profile
DBBn + 1	Control signals							
								Directly activate energy-saving profile
DBBn + 2	Signals to check/test the energy-saving profile							
							PLC user signal	Master computer signal
DBBn + 3	Reserved							
DBBn + 4	Status signal							
							Activation time T1 has expired	Energy-saving profile active
DBBn + 5	Reserved							
DBWn + 6	Actual value: Actual value T1							
DBWn + 8	Actual value: Actual value T2							
DBBn + 10	Effectiveness, profile							
							Disable energy-saving profile	Energy-saving profile configured
	State conditions							
DBBn + 11								
						Screen change active	Data transfer active	Keyboard has been used

DB1000	Ctrl-Energy (operating software → PLC)							
Byte	**Bit 7**	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
DBBn + 12								
								MCP operated
DBBn + 13	NC channel in reset							
	8	7	6	5	4	3	2	1
DBBn + 14	NC channel in reset							
							10	9
DBBn + 15								
							PLC user signal	Master computer signal
DBWn + 16	Activation time T1							
DBWn + 18	Pre-warning time T2							

Additional profile instances

Energy profile 2: DB1000.DBB20 … DBB39
Energy profile 3: DB1000.DBB40 … DBB59
Energy profile 4: DB1000.DBB60 … DBB79
Energy profile 5: DB1000.DBB80 … DBB99
Energy profile 6: DB1000.DBB100 … DBB119
Energy profile 7: DB1000.DBB120 … DBB139
Energy profile 8: DB1000.DBB140 … DBB159

4.16 SENTRON PAC

4.16.1 DB1001, SENTRON PAC

Table 4-98 DB1001, signals for SENTRON PAC

DB1001 /SCE/	SENTRON PAC							
Byte	**Bit 7**	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
DBB0 PLC → operating software			Regenerative energy: Integration using FW	Regenerative energy: Integration using FW	Regenerative energy is read by SENTRON PAC	Energy fed in is read by SENTRON PAC	SENTRON PAC represents the machine	Display manual value
DBB1 GP								GP should perform measurement
DBB2 Operating software → PLC								
								Measurement in progress
DBB3 PLC → operating software								Power display on
DBD4 PLC → operating software	Manual value (REAL) to operating software							
DBD8 PLC → operating software	Total active power (REAL) to operating software							
DBD12 PLC → operating software	Measured drawn active energy in kWh (REAL) to the operating software							
DBD16 PLC → operating software	Measured supplied active energy in kWh (REAL) to the operating software							
DBD20 GP	Total active power in watts (REAL) from SENTRON							
DBD24 GP	Drawn active energy at tariff 1 (F) in Wh (REAL) from SENTRON							

DB1001 /SCE/	SENTRON PAC							
Byte	**Bit 7**	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
DBD28	Supplied active energy at tariff 1 (F) in Wh (REAL) from SENTRON							
GP								
DBD32	Drawn active energy in kWh per day (REAL) to the operating software							
DBD36	Supplied active energy in kWh day (REAL) to the operating software							
DBD40	Drawn active energy in kWh previous day (REAL) to the operating software							
DBD44	Supplied active energy in kWh previous day (REAL) to the operating software							
DBD48	Drawn active energy in kWh month (REAL) to the operating software							
DBD52	Supplied active energy in kWh month (REAL) to the operating software							
DBD56	Drawn active energy in kWh previous month (REAL) to the operating software							
DBD60	Supplied active energy in kWh previous month (REAL) to the operating software							
DBD64	Drawn active energy in kWh year (REAL) to the operating software							
DBD68	Supplied active energy in kWh year (REAL) to the operating software							
DBD72	Drawn active energy in kWh previous year (REAL) to the operating software							
DBD76	Supplied active energy in kWh previous year (REAL) to the operating software							
DBB80 ... DBB95	Reserved							
DBB96	ProductionAct							
PLC→GP/ operating software								
DBB97						Values invalid in		
GP						DBD28	DBD24	DBD20
DBB98	Values invalid in							
GP	DBD384	DBD344	DBD304	DBD264	DBD224	DBD184	DBD144	DBD104
DBB99							Values invalid in	
GP							DBD464	DBD424

4.16.2 DB1001, SENTRON PAC, auxiliary devices

Table 4-99 DB1001, signals for SENTRON PAC

DB1001 /SCE/	SENTRON PAC							
Byte	**Bit 7**	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
DBBn+100	Control/command bits							
PLC → GP operating software	Reset data structure			Input mode: (energy or power)	Different values when measuring energy	Energy measurement running	Read energy actual value update	Process device
DBBn+101	Reserved							
DBBn+102	Reserved							
DBBn+104	Active power or active energy of the auxiliary unit [kW] or [kWh]							
PLC → GP operating software								
DBDn+108	Drawn active energy of the auxiliary unit [kWh]							
PLC → operating software								
DBDn+112	Active energy supplied by the auxiliary unit [kWh]							
GP → operating software								
DBDn+116	Active energy drawn by the auxiliary unit at the measurement start [kWh]							
GP → operating software								
DBDn+120	Active energy supplied by the auxiliary unit at the measurement start [kWh]							
GP → operating software								
DBDn+124	Active energy drawn by the auxiliary unit at the measurement end [kWh]							
GP → operating software								
DBDn+128	Active energy supplied by the auxiliary unit at the measurement end [kWh]							
GP → operating software								
DBDn+132	Reserved							
DBDn+136	Reserved							

Auxiliary unit instances:

Auxiliary unit 1	(n=0):	DB1001.DBB100 … DBB139
Auxiliary unit 2	(n=40):	DB1001.DBB140 … DBB179
Auxiliary unit 3	(n=80):	DB1001.DBB180 … DBB219
Auxiliary unit 4	(n=120):	DB1001.DBB220 … DBB259
Auxiliary unit 5	(n=160):	DB1001.DBB260 … DBB299
Auxiliary unit 6	(n=200):	DB1001.DBB300 … DBB339
Auxiliary unit 7	(n=240):	DB1001.DBB340 … DBB379
Auxiliary unit 8	(n=280):	DB1001.DBB380 … DBB419
Auxiliary unit 9	(n=320):	DB1001.DBB420 … DBB459
Auxiliary unit 10	(n=360):	DB1001.DBB460 … DBB499

4.17 Spindle temperature sensor

4.17.1 DB1002: spindle temperature sensors

Table 4-100 DB1002, signals for **spindle temperature sensors**

DB1002 /SCE/	SENTRON PAC							
Byte	**Bit 7**	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
DBWn + 0	Sensor n Sensor installation location							
DBWn + 2	Reserved							
DBDn + 4	Sensor n Temperature sensor actual value [°C]							
DBDn + 8	Sensor n Temperature sensor warning threshold value [°C]							
DBWn + 12	Sensor n Number of alarm limit value violations							
DBBn + 14	Sensor n Last alarm limit value violation: Year total active power (REAL) to the operating software							
DBBn + 15	Sensor n Last alarm limit value violation: Month							
DBBn + 16	Sensor n Last alarm limit value violation: Day							
DBBn + 17	Sensor n Last alarm limit value violation: Hour							
DBBn + 18	Sensor n Last alarm limit value violation: Minute							
DBBn + 19	Sensor n Last alarm limit value violation: Seconds							

DB1002 /SCE/	SENTRON PAC							
Byte	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
DBBn + 20	Sensor n Duration of the warning limit value violations							
DBBn + 24	Sensor n Temperature sensor fault threshold value [°C]							
DBBn + 28	Sensor n Number of fault limit value violations							
DBBn + 30	Sensor n Last alarm limit value violation: Year							
DBBn + 31	Sensor n Last alarm limit value violation: Month							
DBBn + 32	Sensor n Last alarm limit value violation: Day							
DBBn + 33	Sensor n Last alarm limit value violation: Hour							
DBBn + 34	Sensor n Last alarm limit value violation: Minute							
DBBn + 35	Sensor n Last alarm limit value violation: Seconds							
DBDn + 36	Sensor n Duration of the fault limit value violation							

Spindle/temperature sensor instances

Spindle_1, Tempera-ture_sensor_1	(n=0):	DB1002.DBB00…DBB39
Spindle_1, Tempera-ture_sensor_2	(n=40):	DB1002.DBB40…DBB79
Spindle_1, Tempera-ture_sensor_3	(n=80):	DB1002.DBB80…DBB119
Spindle_1, Tempera-ture_sensor_4	(n=120):	DB1002.DBB120…DBB159

Spindle_1, Temperature_sensor_5	(n=160):	DB1002.DBB160…DBB199
Spindle_1, Temperature_sensor_6	(n=200):	DB1002.DBB200…DBB239
Spindle_2, Temperature_sensor_1	(n=240):	DB1002.DBB240…DBB279
Spindle_2, Temperature_sensor_2	(n=280):	DB1002.DBB280…DBB319
Spindle_2, Temperature_sensor_3	(n=320):	DB1002.DBB320…DBB359
Spindle_2, Temperature_sensor_4	(n=360):	DB1002.DBB360…DBB399
Spindle_2, Temperature_sensor_5	(n=400):	DB1002.DBB400…DBB439
Spindle_2, Temperature_sensor_6	(n=440):	DB1002.DBB440…DBB479

Appendix A

A

A.1 List of abbreviations

Abbreviation	Source of abbreviation	Meaning
ADI4	Analog Drive Interface for 4 axes	
AC	Adaptive Control	
ALM	Active Line Module	Infeed module for drives
UP	User Program	
AS	Automation System	
ASCII	American Standard Code for Information Interchange	American coding standard for the exchange of information
ASIC	Application Specific Integrated Circuit	User switching circuit
ASUP	Asynchronous subprogram	
AUTO		Operating mode "Automatic"
AUXFU	Auxiliary Function:	Auxiliary functions
STL	Statement List	
BA	Operating mode	
Mode group	Mode group	
BERO	Proximity limit switch with feedback oscillator	
BI	Binector Input	
HHU	Handheld unit	
BICO	Binector Connector	Interconnection technology for the drive
BIN	Binary files	Binary files
BIOS	Basic Input Output System	
BCS	Basic Coordinate System	
BO	Binector Output	
OPI	Operator Panel Interface	
CAD	Computer-Aided Design	
CAM	Computer-Aided Manufacturing	
CC	Compile Cycle	Compile cycles
CI	Connector Input	
CF Card	Compact Flash Card	
CNC	Computerized Numerical Control	Computer-Supported Numerical Control
CO	Connector Output	
COM board	Communication Board	
CP	Communications Processor	
CPU	Central Processing Unit	Central processing unit
CR	Carriage Return	
CRC	Cyclic Redundancy Check	Checksum test
CRT	Cathode Ray Tube	picture tube

Abbreviation	Source of abbreviation	Meaning
CSB	Central Service Board	PLC module
CTS	Clear To Send	Signal from serial data interfaces
CUTCOM	Cutter radius compensation	Tool radius compensation
DB	Data block	Data block in the PLC
DBB	Data-block byte	Data block-byte in the PLC
DBW	Data-block word	Data-block word in the PLC
DBX	Data-block bit	Data-block bit in the PLC
DDE	Dynamic Data Exchange	Dynamic Data Exchange
DDS	Drive Data Set	Drive data set
DIN	Deutsche Industrie Norm (German Industry Standard)	
DIR	Directory	Directory
DLL	Dynamic Link Library	
DO	Drive Object	Drive object
DPM	Dual-Port Memory	
DRAM	Dynamic Random Access Memory	Dynamic memory block
DRF	Differential Resolver Function	Differential resolver function (handwheel)
DRIVE-CLiQ	Drive Component Link with IQ	
DRY	DRY run	DRY run feedrate
DSB	Decoding Single Block	Decoding single block
DSC	Dynamic Servo Control / Dynamic Stiffness Control	
DSR	Data Send Ready	Signals that data is ready to be sent from the serial data interfaces
DW	Data word	
DWORD	Double Word (currently 32 bits)	
E	Input	
I/O	Input/Output	
ENC	Encoder	Actual value encoder
EPROM	Erasable Programmable Read Only Memory	Erasable, electronically programmable read-only memory
ePS Network Services		Services for Internet-based remote machine maintenance
EQN		Designation for an absolute encoder with 2048 sine signals per revolution
ESR	Extended stop and retract	
ETC	ETC key	Expansion of the softkey bar in the same menu
FB	Function block	
FBS	Slimline screen	
FC	Function call	Function block in the PLC
FEPROM	Flash EPROM	Read and write memory
FIFO	First In - First Out	Method of storing and retrieving data in a memory
FIPO	Fine InterPOlator	
FM	Function Module	

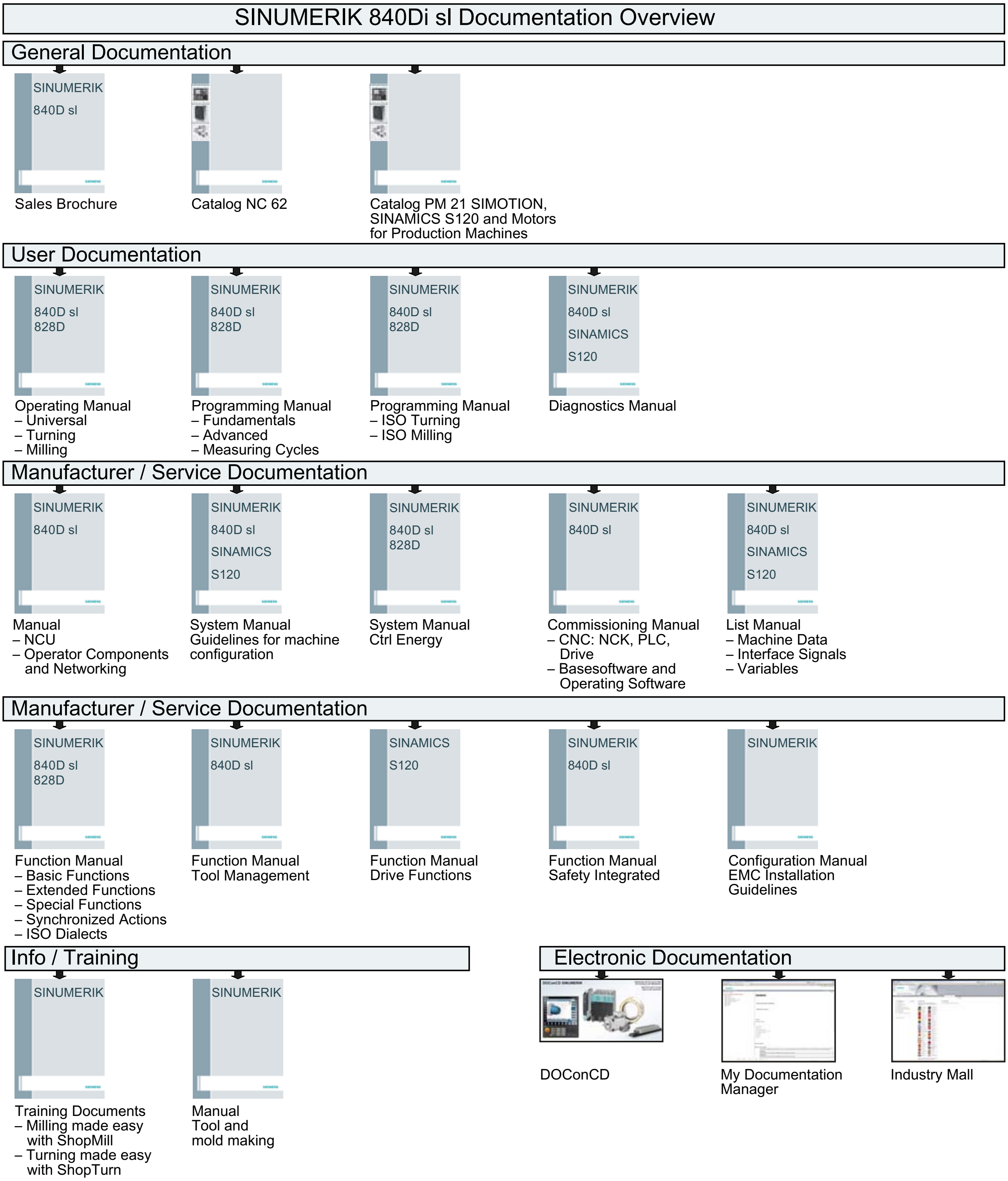
Abbreviation	Source of abbreviation	Meaning
FM-NC	Function Module Numerical Control	Numerical control
FPU	Floating-Point Unit	Floating-point unit
FRA	Frame block	
FRAME	Data set	Coordinate conversion with the components work offset, rotation, scaling, mirroring
CRC	Cutter Radius Compensation	
FST	Feed Stop	Feedrate stop
CSF	Function plan (PLC programming method)	
FW	Firmware	
GC	Global control	PROFIBUS: Broadcast telegram
GD	Global data	
GEO	Geometry, e.g. geometry axis	
GP	Basic program	
GS	Gear stage	
GUD	Global User Data	Global user data
HD	Hard Disk	Hard disk
HEX	Abbreviation for hexadecimal number	
AuxF	Auxiliary function	
HMI	Human Machine Interface	SINUMERIK user interface
MSD	Main Spindle Drive	
HT	Handheld Terminal	Handheld unit
HW	Hardware	
IBN	Startup	
IF	Drive module pulse enable	
IK (GD)	Implicit communication (global data)	
IKA	Interpolative Compensation	Interpolatory compensation
IM	Interface module	Interconnection module
INC	Increment	Increment
INI	Initializing Data	Initializing data
IGBT	Insulated Gate Bipolar Transistor	
IPO	Interpolator	
ISO	International Standardization Organization	International Standards Organization
JOG	"Jogging" operating mode	
COR	Coordinate rotation	
KDV	Crosswise data comparison	Crosswise data comparison between the NC and PLC
K_V	Servo-gain factor	Gain factor of control loop
LAD	Ladder diagram	PLC programming method
LCD	Liquid Crystal Display	Liquid crystal display
LED	Light Emitting Diode	Light Emitting Diode
LF	Line Feed	
PMS		
LSB	Least Significant Bit	Least significant bit

Abbreviation	Source of abbreviation	Meaning
LUD	Local User Data	User data
MAC	Media Access Control	
MAIN	Main program	Main program (OB1, PLC)
MB	Megabyte	
MCI	Motion Control Interface	
MCIS	Motion Control Information System	
MCP	Machine control panel	Machine control panel
MD	Machine data	
MDI	"Manual Data Automatic" operating mode	Manual input
MCS	Machine coordinate system	
MPF	Main Program File	Main program (NC part program)
MPI	Multi-Point Interface	Multi-point interface
MCP	Machine control panel	
NC	Numerical Control	Numerical Control
NCK	Numerical Control Kernel	Numerical control kernel
NCU	Numerical Control Unit	The NC hardware unit
IS	Interfaces	Interface signal
WO	Zero offset	
NX	Numerical Extension	Axis expansion board
OB	Organization block in the PLC	
OEM	Original Equipment Manufacturer	
OP	Operator panel	Operator panel
OPI	Operator Panel Interface	Interface for connection to the operator panel
OSI	Open Systems Interconnection	Standard for computer communications
OPT	Options	Options
PIQ	Process Image Output	
PII	Process Image Input	
P bus	Peripheral Bus	
PC	Personal Computer	
PCMCIA	Personal Computer Memory Card International Association	Standard for plug-in memory cards
PCU	Programmable Control Unit	
PI	Program Instance	
PG	Programming device	
PLC	Programmable logic controller	Programmable Logic Controller
PN	PROFINET	
PO	POWER ON	
POU	Program Organization Unit	Unit in the PLC user program
PPU	Panel Processing Unit	Panel-based control
PTP	Point-to-point	Point-to-Point
PZD	Process data for drives	
QEC	Quadrant Error Compensation	Quadrant error compensation
QEC	Quadrant error compensation	

Abbreviation	Source of abbreviation	Meaning
RAM	Random Access Memory	Program memory that can be read and written to
REF POINT		Function "Reference point approach" in JOG mode
REPOS		Function "Repositioning" in JOG mode
RPA	R parameter Active	Memory area on the NC for R parameter numbers
RPY	Roll Pitch Yaw	Rotation type of a coordinate system
RTC	Real-Time Clock	Real-time clock
RTS	Request To Send	RTS, control signal of serial data interfaces
SBL	Single Block	Single block
SBR	Subroutine	Subroutine (PLC)
SBT	Safe brake test	
SCC	Safety Control Channel	
SD	Setting Data	
SDB	System Data Block	
SEA	Setting Data Active	Identifier (file type) for setting data
SERUPRO	SEarch RUn by PROgram test	Search run by program test
SFC	System Function Call	
SGE	Safety-related input	
SGA	Safety-related output	
SH	Safe standstill	
SIC	Safety Info Channel	
SK	Softkey	
SKP	Skip	Skip block
SLM	Smart Line Module	
SM	Stepper Motor	
SPF	Subprogram file	Subprogram (NC)
SPL	Safe programmable logic	
PLC	Programmable Logic Controller	
SRAM	Static Random Access Memory	Static memory block
TNRC	Tool Nose Radius Compensation	
LEC	Leadscrew error compensation	
SSI	Serial Synchronous Interface	Synchronous serial interface
STW	Control word	
GWPS	Grinding Wheel Peripheral Speed	
SW	Software	
SYF	System Files	System files
SYNACT	SYNACT synchronized action	Synchronized Action
TB	Terminal Board (SINAMICS)	
TEA	Testing Data Active	Identifier for machine data
TCP	Tool Center Point	Tool tip
TCU	Thin Client Unit	
TEA	Testing Data Active	Identifier for machine data
TM	Terminal Module (SINAMICS)	
TO	Tool offset	Tool offset

Abbreviation	Source of abbreviation	Meaning
TOA	Tool Offset Active	Identifier (file type) for tool offsets
TRANSMIT	Transform Milling Into Turning	Coordinate conversion on turning machine for milling operations
TTL	Transistor-transistor logic	Interface type
UFR	User frame	Work offset
SR	Subroutine	
USB	Universal Serial Bus	
UPS	Uninterruptible Power Supply	
VDI		Internal communication interface between NC and PLC
FDD	Feed Drive	
VPM	Voltage Protection Module	
VSM	Voltage Sensing Module	
WAB		Function "Smooth Approach and Retraction"
Work	Workpiece coordinate system	
T	Tool	
TLC	Tool length compensation	
WPD	Workpiece Directory	Workpiece directory
T	Tool	
TM	Tool management	
TC	Tool change	
ZWS		Buffer location
ZOA	Work Offset Active	Identifier (file type) for zero offset data
SW	Status word (of drive)	

A.2 Documentation overview



Index

A

C

D

F

H

I

M

N

O

R

T