Lists 07/2005 Edition

SINUMERIK 840D sl/840D/810D/840Di
SIMODRIVE 611 digital/SINAMICS
Lists (2nd Book)

SIEMENS

SIEMENS

SINUMERIK 840D sl/
SINUMERIK 840D/840Di/810D
SIMODRIVE 611 digital/Sinamics

Lists (2nd Book)

Valid for

Controls	*Software version*
SINUMERIK 840D sl/840DE sl	1.2
SINUMERIK 840D powerline	7.3
SINUMERIK 840DE powerline (Export variant)	7.3
SINUMERIK 840Di	2.3
SINUMERIK 840DiE (Export variant)	2.3
SINUMERIK 810D powerline	7.3
SINUMERIK 810DE powerline (Export variant)	7.3
SINAMICS	2.3

07.05 Edition

SINUMERIK®-Dokumentation

Printing history

Brief details of this edition and previous editions are listed below.

The status of each edition is shown by the code in the "Remarks" column.

Status codes in the "Remarks" column.

A .... New documentation.
B .... Unrevised reprint with new Order No.
C .... Revised edition with new status.
If factual changes have been made on a page since the last edition, this is indicated by a new edition coding in the header on that page.

Edition	Order No.	Remarks
07.05	6FC5 397-3CP10-0BA0	**A**

Trademarks

SIMATIC©, SIMATIC HMI©, SIMATIC NET©, SIROTEC©, SINUMERIK© and SIMODRIVE© are trademarks of SIEMENS. Other product names used in this documentation may be trademarks which, if used by third parties, could infringe the rights of their owners.

Further information is available on the Internet under:
http://www.siemens.com/motioncontrol

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.

We have checked that the contents of this document correspond to the hardware and software described. Nonetheless, differences might exist. The information contained in this document is, however, reviewed regularly and any necessary changes will be included in the next edition. We welcome suggestions for improvement.

Subject to change without prior notice.

Order No. 6FC5 397-3CP10-0BA0
Printed in the Federal Republic of Germany

Siemens-Aktiengesellschaft.

Preface

With the present edition, the previous Lists with order number 6FC5 297-7AB70-.... will be subdivided into Lists (Volume 1) and Lists (Volume 2).

Volume 1 contains:

- Overview of functions
- Maschine data (Drive 611D, Hydraulics module, HMI, NCK, SD)
- **Sinamics** Parameters

The table of contents refers to the present Volume 2.

SINUMERIK Documentation

The SINUMERIK documentation is organized in three parts:

- General Documentation
- User Documentation
- Manufacturer / service documentation

Please contact your local Siemens office for more detailed information about other SINUMERIK 840D sl/840D/840Di/810D publications and publications that apply to all SINUMERIK controls (e.g. universal interface, measuring cycles etc.).

An overview of publications which is updated each month and shows the languages available is provided on the Internet at:
http://www.siemens.com/motioncontrol.
Click the menu items in the following order: "Support" → "Technical Documentation" → "Overview of Publications".

The Internet edition of DOConCD – DOConWEB - can be found at:
http://www.automation.siemens.com/doconweb

Target Group

This document is designed for machine tool manufacturers with SINUMERIK 840D sl/840D/840Di/810D controls and SIMODRIVE 611D or SINAMICS.

Standard Scope

This Programming Guide describes the functionality afforded by standard functions. Extensions or changes made by the machine tool manufacturer are documented by the machine tool 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.

Hotline

If you have any questions, please get in touch with our hotline:
A&D Technical Support
Tel.: +49 (0) 180 / 5050 - 222
Fax: +49 (0) 180 / 5050 - 223

E-Mail: mailto:adsupport@siemens.com
Internet: http://www.siemens.de/automation/support-request

Please send any queries about the documentation (suggestions or corrections) to the following fax number or e-mail address:
Fax: +49 (0) 9131 / 98 - 63315
E-Mail: motioncontrol.docu@siemens.com

Fax form: See the reply form at the end of the document.

Internet Address

http://www.siemens.com/motioncontrol

Safety Guidelines

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 to property damage only 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

with a safety alert symbol, indicates that minor personal injury can result if proper precautions are not taken.

Caution

without a safety alert symbol, indicates that property damage can result if proper precautions are not taken.

Notice

indicates that an unintended result or situation can occur if the corresponding information is not taken into account.

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 device/system may only be set up and used in conjunction with this documentation. Commissioning and operation of a device/system may only be performed by **qualified personnel**. Within the context of the safety notes in this documentation qualified persons are defined as persons who are authorized to commission, ground and label devices, systems and circuits in accordance with established safety practices and standards.

Prescribed Usage

Note the following:

Warning

This device may only be used for the applications described in the catalog or the technical description and only in connection with devices or components from other manufacturers which have been approved or recommended by Siemens. Correct, reliable operation of the product requires proper transport, storage, positioning and assembly as well as careful operation and maintenance.

Trademarks

IBM® is a registered trademark of the International Business Corporation. MS-DOS® and WINDOWS™ are registered trademarks of the Microsoft Corporation.

■

Contents

1 Variables

1

1.1 Introduction

This section describes the NCK variables that an MMC or the PLC can access via the operator panel interface. (Access is read and for some variables write also). The access methods of the various components are described in the following user documentation:

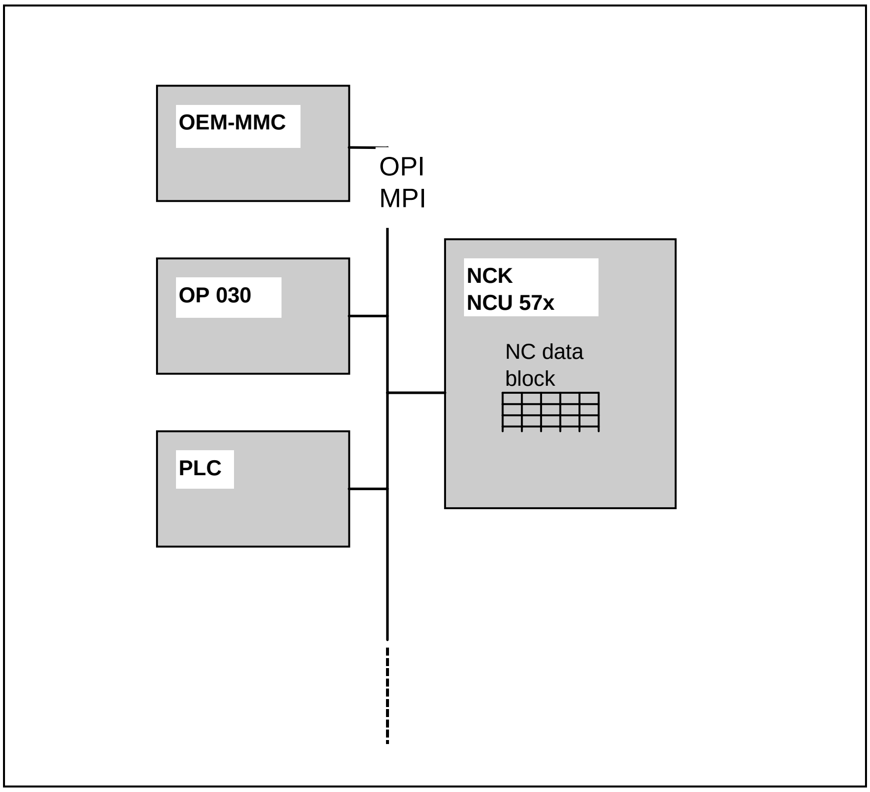
References: /FBO/, Configuring the OP030 Operator Interface

/PK/, Configuring kit MMC 100/Unit Operator Panel

Description of PLC access method in:

References: /FB/, P3, "Basic PLC Program"

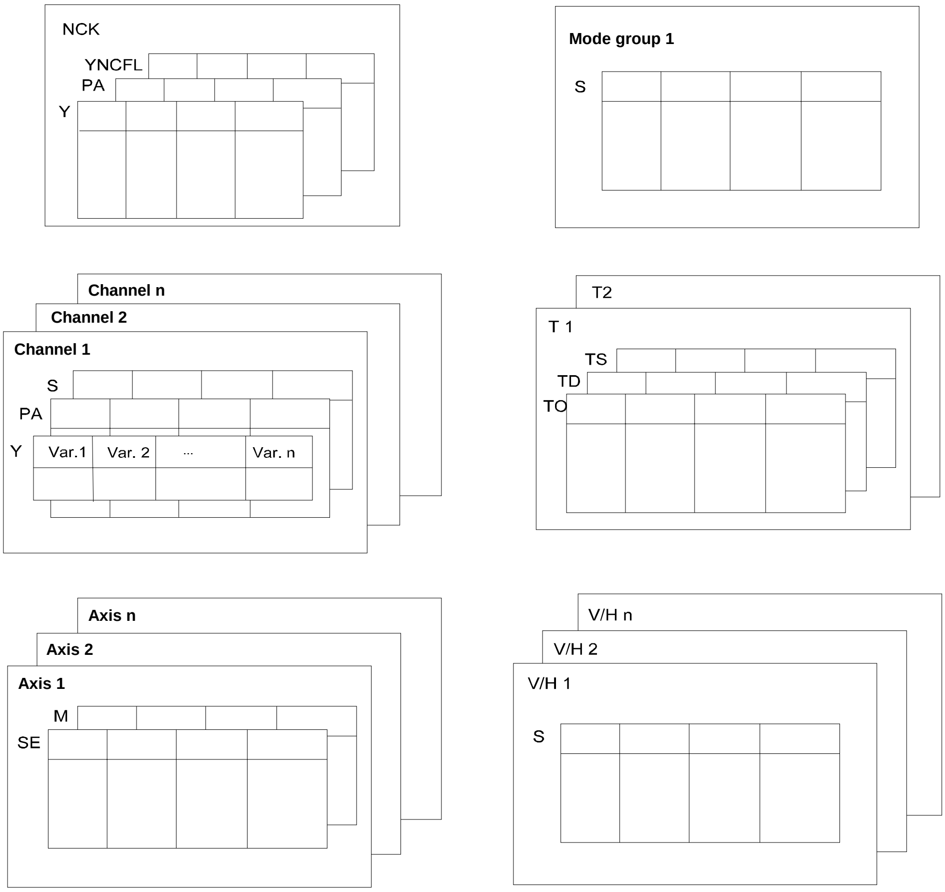
Description of the OEM-MMC access method in "OEM-MMC Description of Functions".



The components shown on the left-hand side of the diagram each have their own development environment which defines the syntax to be used. A variable is always addressed according to a defined pattern. All the information required for addressing the variables irrespective of the programming language chosen is summed up in the following lists.

1.1.1 General information

The NCK variables are stored in data modules that are assigned to the individual areas of the NCK as the figure below shows:



Variable areas of SINUMERIK 840D/810D/FM-NC

A distinction is made between the following areas:

- NCK (N)
- Mode group (B)
- Channel (C)
- Tool (T)
- Axis (A)
- Feed/main drive (V/H)

NCK Contains all the variables such as system data (Y), protection zones (PA), G groups (YNCFL) etc. that apply to the entire NCK.

Mode group Contains variables such as the status data (S) that apply to the mode group.

Channel Contains variables such as the system data (Y), protection zones (PA), global status data (S) etc. that apply to each channel.

Tool Contains variables such as the tool offset data (TO), general tool data (TD), tool monitoring data (TS) etc. that apply to the tools on the machine. Each tool area T is assigned to a channel.

Axis Contains the setting data and machine data that apply to each axis or spindle. For a description see Section "Axis-specific machine data".

Feed / main drive Contains machine data and machine data as the service values that apply to each drive. For a description see Section "Drive machine data".

1.1.2 Module types

The following table provides an overview of the modules for the variables of the NCK and how they are assigned to the individual areas.

Only the data modules whose variables can be read or written with direct access are contained in the list. Data modules whose variables can be defined by the programmer (e.g. global user data) are read by the MMC or PLC using other mechanisms. The documentation listed below describes the modules to which these mechanisms are applied:

References: /FBO/, Configuring the OP030 Operator Interface,

/FB/, P3, "Basic PLC Program" and

"OEM-MMC Description of Functions"

Module	Area						
	A	B	C	H	N	T	V
ETP			1				
ETPD					1		
DIAGN			1				
FA			1		1		
FB			1		1		
FE			1				
FU			1		1		
M	1				1		
NIB			1				
PA			1		1		
RP			1				

Module	Area						
continued	A	B	C	H	N	T	V
S		1	1	1	1		1
SALA					1		
SALAL					1		
SALAP					1		
SE	1		1		1		
SEGA			1				
SEMA			1		1		
SGA			1				
SINF			1				
SMA			1		1		
SNCF			1				
SPARP			1				
SPARPF			1				
SPARPI			1				
SPARPP			1				
SSP			1		1		
SSP2			1		1		
SSYNAC			1				
SYNACT			1				
TD						1	
TF						1	
TG						1	
TM						1	
TMC						1	
TMV						1	
TO						1	
TP						1	
TPM						1	
TS						1	
TT						1	
TU						1	
TUE						1	
TUM						1	
TUP						1	
TUS						1	
TV						1	
AD						1	
AEV						1	
TC						1	
TOE						1	
TOET						1	
TOS						1	
TOST						1	
TOT						1	
VSYN			1				
Y			1		1		
YNCFL					1		

1.1.3 Variable types

Within each area the variables are generally stored in the form of structures or in arrays of structures (tables). The following information must therefore be contained in an address when accessing a variable:

- Area + area number
- Module
- Variable name (or column number)
- Line number

It is generally possible to distinguish between three different variable types:

1. Variables that consist of 1 line
2. Variables that consist of several lines
3. Variables that consist of several columns and lines

Single-line variables

Each of these variables consists of a single value. The following information is required when accessing a variable of this type:

1. Area (and possibly area number)
2. Module
3. Variable name

numMachAxes					
Number of existing machine axes					
-				Word	r
Multi-line: no					

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

MMC102:

/Channel/Configuration/numMachAxes[u1]

MMC100/OP030:

P_C_Y_numMachAxes

PLC with NC-Var-Selector:

Area: C[.]

Module: Y

Variable: numMachAxes

Area No. = 1

Multi-line variables

These variables are defined as a one-dimensional field. When accessing a variable of this type the following information must be specified:

1. Area (and possibly area number)
2. Module
3. Variable name
4. Line number

actFeedRate	$AA_VACTB[x]				S5
Axial feedrate actual value (only if axis is a positioning axis "spec" = 1)					
%				Double	r
Multi-line:yes	Axis index		numMachAxes		

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

MMC102:

/Channel/MachineAxis/actFeedRate[u1, 3]

MMC100/OP030:

P_C_SEMA_actFeedRate

PLC with NC-Var-Selector:

Area: C[.]

Module: SEMA

Variable: actFeedRate[.]

Area No. = 1

Line = 3

Multi-line and multi-column variables

These variables are defined as a two-dimensional field. In order to access a variable of this type, the following information must be specified:

1. Area (and possibly area number)
2. Module
3. Variable name
4. Column number
5. Line number

In this case the entire data module only consists of this two-dimensional variable.

cuttEdgeParam	$TC_DPx[y,z]				
Offset value parameters for a cutting edge					
mm, inch or userdef	0			Double	wr
Multi-line:yes	(CuttEdgeNo 1) * numCuttEdgeParams + ParameterNo		numCuttEdgeParams * numCuttEdges		

Example for reading the current cutting edge data of cutting edge 3/parameter 1 of tool 3 in T area 1: (in this example it is assumed that each tool cutting edge has been defined with (numCuttEdgeParams =) 25 parameters).

MMC102:

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

MMC100/OP030:

P_T_TO_cuttEdgeParam

PLC with NC-Var-Selector:

Area: T[.]

Module: TO

Variable: cuttEdgeParam[.]

Area No. = 1

Column = 3

Line = 51

1.1.4 Data types

The following data types are used in this description:

Data type	Description
Bool	1 bit
Character	8 bits without sign
Byte	8 bits with sign
Word	16 bits without sign
Short Integer	16 bits with sign
Doubleword	32 bits without sign
Long Integer	32 bits with sign
Float	32 bits floating point
Double	64 bits floating point
String	String ending in zero

In the tables below the individual fields have the following meaning:

Variable name	Reference to assigned MD				Ref.
Variable brief description/ variable description « Description of value range »					
Physical unit	Default value	Lower limit	Upper limit	Format / field length	w / r
Multi-line:yes / no	Description of line index		Maximum line index		

Ref. Cross-reference to references

w / r w Variable can be overwritten

r Variable can be read

1.2 System data

1.2.1 Area N, Mod. Y: Global system data

OEM-MMC: Linkitem /Nck/Configuration/...

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					

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		

externCncSystem	$MN_EXTERN_LANGUAGE und $MN_EXTERN_CNC_SYSTEM				
CNC system whose part programs must be processed on the SINUMERIK control. 0: No external language defined 1: System ISO Dialect0 Milling 2: System ISO Dialect0 Turning 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: Edge parameter no. 26 valid (ISO Dialect Milling H No.) Bit 1: Edge parameter no. 27 valid Bit 2: Edge parameter no. 28 valid Bit 3: Edge parameter no. 29 valid Bit 4: Edge parameter no. 30 valid etc.					
-				UWord	r
Multi-line: yes	1		1		

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 SW 5.1 and later: 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				

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				

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 ??? NCK			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					

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				

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	wr
Multi-line: no			1		

nckType					
NCK type 0: 840D pl 1000: FM-NC (up to and including SW 6) 2000: 810D pl 3000: 802S (up to and including SW 6) 4000: 802D pl (up to and including SW 6) 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					
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 Not defined in SW 6.2. 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	UDoubleword	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					
Number of active drives					
-				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					

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	UDoubleword	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	UDoubleword	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 module TUM					
-	0	0	10	UWord	r
Multi-line: yes	1		1		

numMagPlaceParams	$TC_MPP1				
Number of parameters of a magazine location 8 in SW 5.1 and later					
-				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					

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

numParams_SC	$TC_SCPx; x=13,...21,....71				
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					

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 tool magazine management is not active, the value =0 will be returned.					
-	3	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	UDoubleword	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		

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: no			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)					
-				Double	r
Multi-line: no					

typeOfCuttingEdge					
Type of D-number programming see MD: MM_TYPE_OF_CUTTING_EDGE 0 no 'flat D-number management' active 1 D-numbers are programmed directly and absolutely 2 D-numbers are programmed indirectly and relatively					
-				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		

1.2.2 Area C, Mod. Y: Channel-specific system data

OEM-MMC: Linkitem /Channel/Configuration/...

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. ------------------------For test purposes only :--- 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.. ------------------------End For test purposes only :-- 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 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 MMC 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 MMC (e.g. for calling external subprograms)					
-				String[206]	r
Multi-line: yes	1		1		

mmcCmdQuit					
Acknowledgement from MMC for command from NCK to MMC 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 MMC for an NCK-preparation command to the MMC (e.g. for calling external subprograms)					
-				String[200]	wr
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					

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 parameters					
-				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		

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					

1.2.3 Area N, Mod. PA: Global protection zones

OEM-MMC: Linkitem /Nck/ProtectedArea/...

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	numContourInProtArea	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		

1.2.4 Area C, Mod. PA: Channel-specific protection zones

OEM-MMC: Linkitem /Channel/ProtectedArea/...

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		

1.2.5 Area N, Mod. YNCFL: NCK instruction groups

OEM-MMC: Linkitem /Nck/FunctionGrouping/...

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.

Function					
Instruction group Attention: This variable is called "Gruppe" in the non-Windows MMC and the PLC !					
-				String[16]	r
Multi-line: yes	Serial number		/N/YNCFL/GroupID		

GroupID					
Number of G functions in each G group Attention: This variable is called "Gruppe_NUM" in the non-Windows MMC and the PLC !					
-				UWord	r
Multi-line: yes	1		1		

Gruppe					
Instruction group Attention: This variable is called "Function" in the MMC102 !					
-				String[16]	r
Multi-line: yes	Serial number		/N/YNCFL/Gruppe_NUM		

Gruppe_NUM					
Number of G functions in each G group Attention: This variable is called "GroupID" in the MMC102 !					
-				UWord	r
Multi-line: yes	1		1		

1.3 State data of system

1.3.1 Area N, Mod. S: Global state data

OEM-MMC: Linkitem /Nck/State/...

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)

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	wr
Multi-line: no					

aDbb	$A_DBB[x] x = ByteNo				
Data byte from/to the PLC Can be written from SW 6.4.					
-				UWord	wr
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 Can be written from SW 6.4.					
-				Long Integer	wr
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 Can be written from SW 6.4.					
-				Double	wr
Multi-line: yes	Position offset within an I/O area				

aDbw	$A_DBW[x] x = Offset				
Data word (16 bits) from/to the PLC Can be written from SW 6.4.					
-				UWord	wr
Multi-line: yes	Position offset within an I/O area				

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

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

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

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

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

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	wr
Multi-line: yes	Number of analog output		numAnalogOutp		

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 = max. number of occupied slots in 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		

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		

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) Can be written from SW 6.4. Neg. values are also permitted in spite of TYPE_UWORD					
-				UWord	wr
Multi-line: yes	Position offset within the PLC input/output area				

aPbdIn	$A_PBD_IN[index]				
Data double word (32bits) in PLC input/output area IN					
-				UDoubleword	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) Can be written from SW 6.4.					
-				UDoubleword	wr
Multi-line: yes	Position offset within the 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) Can be written from SW 6.4.					
-				Double	wr
Multi-line: yes	Position offset within the 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) Can be written from SW 6.4. Neg. values are also permitted in spite of TYPE_UWORD					
-				UWord	wr
Multi-line: yes	Position offset within the PLC input/output area				

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 MMCs 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.					
-				UDoubleword	r
Multi-line: yes	1		1		

badMemFfs					
Number of bytes which are defective in the Flash File System (FFS)					
-	0			UDoubleword	r
Multi-line: yes	1		1		

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

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					
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	wr
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	1		1		

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		

freeMemDram2PassF					
Memory available in passive file system (DRAM no. 2) in bytes					
-	0	0		UDoubleword	r
Multi-line: yes	1		1		

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

freeMemFfs					
Number of bytes that are still available in the Flash File System (FFS)					
-	0			UDoubleword	r
Multi-line: yes	1		1		

freeMemSramPassF					
Memory available in passive file system (SRAM) in bytes					
-	0	0		UDoubleword	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		

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

hwProductSerialNr					
Unique hardware number of the NCU module					
-				String[16]	r
Multi-line: yes	1		1		

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

licenseStatus					
Licensing status 0: Licensed, 1: Insufficiently licensed 2: Not licensed					
-	0			UWord	r
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 MMC 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 MMC-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, by the client's "gloports": line index = 1 and gloports 0x20-0x2f --> DB10.DBX108 bit2 MMC1-CPU-Ready to MPI line index = 1 and gloports 0x10-0x1f --> DB10.DBX108 bit3 MMC1-CPU-Ready to OPI line index = 2 --> DB10.DBX108 bit1 MMC2-CPU-Ready Note: The related NCK-CPU-Ready signal is stored in DB10,DBX104.7.					
-				UWord	r
Multi-line: yes	MMC No.		2 (ab SW 5.2)		

ncuLinkActive					
Display indicating whether NCU link is activated (via machine data setting) Based on display, MMC 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					
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			UDoubleword	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		

protCnfgAutoLoad					
Log: Controls automatic loading of the trace session from the description file during the NCK start-up process 0: Automatic load is disabled 1: Load trace session from description file during NCK start-up 2: Same as (1), but cancel automatic load when stop trigger fires 3: Same as (1), but update the description file whenever the logging state changes					
-	0	0	3	UWord	wr
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]	wr
Multi-line: yes	User no. (1-10)		10		

protCnfgAutoSave					
Log: Automatically save the trace session into a description file 0: Automatic save is disabled 1: Save the trace session to a description file whenever data logging is terminated 2: Save the trace session and diagnostic information to a description file whenever data logging is terminated					
-	0	0	2	UWord	wr
Multi-line: yes	User no. (1-10)		10		

protCnfgCtl					
Log: Control word to manipulate the trace session description file 0: Do nothing 1: Save the trace session into a description file 2: Save the trace session with diagnostic information into a description file 3: Load the trace session from a description file and reset all active triggers to the armed state 4: Load the trace session from a description file with the saved trigger states 5: Delete the trace session description file					
-	0	0	5	UWord	wr
Multi-line: yes	User no. (1-10)		10		

protCnfgFilename					
Log: Directory path and file name of the session description file that is managed by protCnfgCtl					
-				String[64]	wr
Multi-line: yes	User no. (1-10)		10		

protCnfgStat					
Log: Result from the most recent save or load of a description file 0: No Error					
-	0			UWord	wr
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	wr
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	wr
Multi-line: yes	User no. (1-10)		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		UDoubleword	wr
Multi-line: yes	User no. (1-10)		10		

protocStrtMatchCount					
Logging: Specifies how often the comparison must match before the start trigger fires.					
-	0	0		UWord	wr
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	wr
Multi-line: yes	User no. (1-10)		10		

protocStrtRemMatchCount					
Logging: Specifies how often the comparison still has to match before the start trigger fires.					
-	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	wr
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	wr
Multi-line: yes	User no. (1-10)		10		

protocStrtValueInt16					
Logging: Integer 16 bit value with which the start trigger variable is to be compared					
-	0	0		UWord	wr
Multi-line: yes	User no. (1-10)		10		

protocStrtValueInt32					
Logging: Integer 32 bit value with which the start trigger variable is to be compared					
-	0	0		UDoubleword	wr
Multi-line: yes	User no. (1-10)		10		

protocStrtValueReal32					
Logging: Real 32 bit value with which the start trigger variable is to be compared					
-	0	0		Float	wr
Multi-line: yes	User no. (1-10)		10		

protocStrtValueReal64					
Logging: Real 64 bit value with which the start trigger variable is to be compared					
-	0	0		Double	wr
Multi-line: yes	User no. (1-10)		10		

protocStrtVarCol					
Logging: Variable which is to be monitored for the start triggering. Statement of the "Col"					
-	0	0		UWord	wr
Multi-line: yes	User no. (1-10)		10		

protocStrtVarRow					
Logging: Variable which is to be monitored for the start triggering. Statement of the "Row"					
-	0	0		UWord	wr
Multi-line: yes	User no. (1-10)		10		

protocStrtVarType					
Logging: Variable which is to be monitored for the start triggering. Statement of the "Type"					
-	0	0		UWord	wr
Multi-line: yes	User no. (1-10)		10		

protocStrtVarUnit					
Logging: Variable which is to be monitored for the start triggering. Statement of the "Unit".					
-	0	0		UWord	wr
Multi-line: yes	User no. (1-10)		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	wr
Multi-line: yes	User No. (1-10)		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		UDoubleword	wr
Multi-line: yes	User No. (1-10)		10		

protocTrigMatchCount					
Logging: Specifies how often the comparison must match before the trigger fires.					
-	0	0		UWord	wr
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	wr
Multi-line: yes	User No. (1-10)		10		

protocTrigRemMatchCount					
Logging: Specifies how often the comparison still has to match before the trigger fires.					
-	0	0		UWord	r
Multi-line: yes	User no. (1-10)		10		

protocTrigState					
Logging: Triggering status 0: Passive (trigger not active) 1: Active (trigger is active, but has not yet responded) 2: Delay (trigger has responded and is waiting for delay) 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	wr
Multi-line: yes	User No. (1-10)		10		

protocTrigType					
Logging: Triggering method 0: Monitor for equals 1: Monitor for greater than/equal to 2: Monitor for greater than 3: Monitor for less than/equal to 4: Monitor 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	wr
Multi-line: yes	User No. (1-10)		10		

protocTrigValueInt16					
Logging: Integer 16-bit value with which trigger variable must be compared					
-	0	0		UWord	wr
Multi-line: yes	User No. (1-10)		10		

protocTrigValueInt32					
Logging: Integer 32-bit value with which trigger variable must be compared					
-	0	0		UDoubleword	wr
Multi-line: yes	User No. (1-10)		10		

protocTrigValueReal32					
Logging: Real 32-bit value with which trigger variable must be compared					
-	0	0		Float	wr
Multi-line: yes	User No. (1-10)		10		

protocTrigValueReal64					
Logging: Real 64-bit value with which trigger variable must be compared					
-	0	0		Double	wr
Multi-line: yes	User No. (1-10)		10		

protocTrigVarArea					
Logging: Variable which is to be monitored for the start triggering. Specification of "Area".					
-	0	0		UWord	wr
Multi-line: yes	User No. (1-10)		10		

protocTrigVarCol					
Logging: Variable to be monitored for triggering. Specification of "Col".					
-	0	0		UWord	wr
Multi-line: yes	User No. (1-10)		10		

protocTrigVarRow					
Logging: Variable to be monitored for triggering. Specification of "Row".					
-	0	0		UWord	wr
Multi-line: yes	User No. (1-10)		10		

protocTrigVarType					
Logging: Variable to be monitored for triggering. Specification of "Type".					
-	0	0		UWord	wr
Multi-line: yes	User No. (1-10)		10		

protocTrigVarUnit					
Logging: Variable to be monitored for triggering. Specification of "Unit".					
-	0	0		UWord	wr
Multi-line: yes	User No. (1-10)		10		

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

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

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

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

protSessPrior					
Logging: Priority of the session					
-				String[32]	wr
Multi-line: yes	User no. (1-10)		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		64		

safeExtInpValNckWord	$A_INSED				
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_INSE[1]...[32] 2: image of the system variables $A_INSE[33]...[64]		2		

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		64		

safeExtInpValPlcWord	$A_INSEPD				
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_INSEP[1]...[32] 2: image of the system variables $A_INSEP[33]...[64]		2		

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		64		

safeExtOutpValNckWord	$A_OUTSED				
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_OUTSE[1]...[32] 2: image of the system variables $A_OUTSE[33]...[64]		2		

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		64		

safeExtOutpValPlcWord	$A_OUTSEPD				
Image of the external PLC outputs of the SI programmable logic					
-	0			Long Integer	r
Multi-line: yes	1: image of the system variable $A_OUTSEP[1]...[32] 2. mage of the system variable $A_OUTSEP[33]...[64]		2		

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		64		

safeIntInpValNckWord	$A_INSID				
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_INSI[1]...[32] 2: image of the system variables $A_INSI[33]...[64]		2		

safeIntInpValPlcBit	$A_INSIP[n]				
Internal PLC input of the SI programmable logic from the 611D's SI monitoring channel					
-	0	0	1	UWord	r
Multi-line: yes	Input number		64		

safeIntInpValPlcWord	$A_OUTSID				
Image of the internal PLC inputs of the SI programmable logic from the 611D's SI monitoring channel					
-	0			Long Integer	r
Multi-line: yes	1: image of the system variables $A_INSIP[1]...[32] 2: image of the system variables $A_INSIP[33]...[64]		2		

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		64		

safeIntOutpValNckWord	$A_OUTSID				
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 variable $A_OUTSI[1]...[32] 2: image of the system variable $A_OUTSI[33]...[64]		2		

safeIntOutpValPlcBit	$A_OUTSIP[n]				
Internal PLC output of the SI programmable logic to the 611D's SI monitoring channel					
-	0	0	1	UWord	r
Multi-line: yes	Output number		64		

safeIntOutpValPlcWord	$A_OUTSIPD				
Image of the internal PLC outputs of the SI programmable logic to the 611D's SI monitoring channel					
-	0			Long Integer	r
Multi-line: yes	1: image of the system variable $A_OUTSIP[1]...[32] 2: image of the system variable $A_OUTSIP[33]...[64]		2		

safeMarkerNck	$A_MARKERSI				
NCK flag for the SI programmable logic					
-	0	0	1	UWord	r
Multi-line: no			64		

safeMarkerPlc	$A_MARKERSIP				
Image of the PLC flag-variable for SI programmable logic					
-	0	0	1	UWord	r
Multi-line: no			64		

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[1...32]		32		

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[1...32]		32		

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: Drive runup status 4 reached, NCK is waiting for PLC to run up Bit 3: Drive runup status 4 reached, PLC has reached cyclic operating status. PLC can now 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: - Bit 9: NCK cross-checking has been started Bit10: PLC cross-checking has been started Bit11: Cyclic SPL checksum check active Bit12: All SPL protective mechanisms active					
-	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 MMCs 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		

swLicensePIN					
PIN for licensing					
-				String[128]	wr
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					
-	0	0		UDoubleword	r
Multi-line: yes	1		1		

sysTimeSinceStartup					
System run time in seconds since NCK ramp-up					
s, user defined	0	0		Double	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	1		1		

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		

totalMemDram2PassF					
Size of passive file system (DRAM No.2) in bytes					
-	0	0		UDoubleword	r
Multi-line: yes	1		1		

totalMemDramPassF					
Size of passive file system (DRAM No. 1) in bytes					
-	0	0		UDoubleword	r
Multi-line: yes	1		1		

totalMemFfs					
Number of bytes reserved on the PCMCIA card for the Flash File System (FFS)					
-	0			UDoubleword	r
Multi-line: yes	1		1		

totalMemSramPassF					
Size of passive file system (SRAM) in bytes					
-	0	0		UDoubleword	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 1 = Not active 2 = 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					
-	0	0	1	UWord	wr
Multi-line: yes	User no. (1-10)		10		

traceStopAction					
Logging: Actions on ending the recording Bit 0: Automatic restart 1: Disable					
-	0	0		UWord	wr
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	1		1		

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		

usedMemDram2PassF					
Memory used in passive file system (DRAM No.2) in bytes					
-	0	0		UDoubleword	r
Multi-line: yes	1		1		

usedMemDramPassF					
Memory used in passive file system (DRAM No. 1) in bytes					
-	0	0		UDoubleword	r
Multi-line: yes	1		1		

usedMemFfs					
Number of used bytes in the Flash File System (FFS)					
-	0			UDoubleword	r
Multi-line: yes	1		1		

usedMemSramPassF					
Memory used in passive file system (SRAM) in bytes					
-	0	0		UDoubleword	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		UDoubleword	r
Multi-line: yes	100 * axis index + word offset in the message frame		100 * numMachAxes + 19		

1.3.2 Area N, Mod. SALA: Alarms: List organized according to time, oldest alarm appears first

OEM-MMC: Linkitem /Nck/SequencedAlarms/...

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 MMC alarms. In order to read all alarms, the OEM-MMC 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 criteria for an alarm 1 = Power On 2 = Reset 3 = Cancel 4 = Alarm is cancelled by NCK-software (from SW 4.1) 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: no			1		

1.3.3 Area N, Mod. SALAP: Alarms: List organized according to priority

OEM-MMC: Linkitem /Nck/TopPrioAlarm/...

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.
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 SALAP only contains the alarms that are generated in the NCK. It contains neither PLC nor MMC alarms. In order to read all alarms, the OEM-MMC 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 criteria of an alarm 1 = Power On 2 = Reset 3 = Cancel 4 = Alarm is cancelled by NCK-software (from SW 4.1) 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: no					

1.3.4 Area N, Mod. SALAL: Alarms: Liste organized according to time, most recent alarm appears first

OEM-MMC: Linkitem /Nck/LastAlarm/...

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 MMC alarms. In order to read all alarms, the OEM-MMC user should use the alarm server functions and not read the SALA module directly.

<table>
<tr><td>alarmNo</td><td colspan="4"></td><td>DA</td></tr>
<tr><td colspan="6">Ordinal number of an alarm (how many alarms since control ON)
0 = unknown alarm</td></tr>
<tr><td>-</td><td></td><td></td><td></td><td>Long Integer</td><td>r</td></tr>
<tr><td>Multi-line: yes</td><td colspan="2">Alarm list index
The maximum alarm list index can be read via variable numAlarms in module S.</td><td colspan="3">16</td></tr>
</table>

<table>
<tr><td>clearInfo</td><td colspan="4"></td><td>DA</td></tr>
<tr><td colspan="6">Acknowledgement criteria for an alarm
1 = Power On
2 = Reset
3 = Cancel
4 = Alarm is cancelled by NCK-software (from SW 4.1)
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)</td></tr>
<tr><td>-</td><td></td><td></td><td></td><td>Long Integer</td><td>r</td></tr>
<tr><td>Multi-line: no</td><td colspan="2"></td><td colspan="3"></td></tr>
</table>

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: no					

1.3.5 Area N, Mod. SMA: State data: Machine axes

OEM-MMC: Linkitem /Nck/MachineAxis/...

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 (in this module).					
mm, inch, degree, user defined				Double	r
Multi-line: yes	Axis index		numMachAxes		

cmdToolBasePos					
Tool base position setpoint. 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 state 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		

1.3.6 Area N, Mod. SEMA: State data: Machine axes (extension of SMA)

OEM-MMC: Linkitem /Nck/MachineAxis/...

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.

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			UDoubleword	r
Multi-line: yes	Axis index		maxnumGlobMachAxes		

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 index		maxnumGlobMachAxes		

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 index		maxnumGlobMachAxes		

aaCoupAct					
Current coupling state of the slave spindle					
-				UWord	r
Multi-line: yes	Axis index		maxnumGlobMachAxes		

aaCoupOffs					
Position offset of the synchronous spindle desired value					
-				Double	r
Multi-line: yes	Axis index		maxnumGlobMachAxes		

aaCurr					
Actual value of the axis/spindle current in A (611D only)					
A				Double	r
Multi-line: yes	Axis index		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 index		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 index		maxnumGlobMachAxes		

aaDtepb					
Axis-specific distance-to-go of infeed during oscillation in the BCS (note: SYNACT only)					
-				Double	r
Multi-line: yes	Axis index		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 index		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 index		maxnumGlobMachAxes		

aaEsrTrigger	$AA_ESR_TRIGGER				
Activation of "NC-controlled ESR" for PLC-controlled axis					
-	0	0	1	UWord	r
Multi-line: yes	Axis index		maxnumGlobMachAxes		

aaIbnCorr	$AA_IBN_CORR[<Achse>]				
Current BZS setpoint value of an axis including override components					
-	0			Double	r
Multi-line: yes	Axis index		maxnumGlobMachAxes		

aaIenCorr	$AA_IEN_CORR[<Achse>]				
Current SZS setpoint value of an axis including override components					
-	0			Double	r
Multi-line: yes	Axis index		maxnumGlobMachAxes		

aaJerkCount	$AA_JERK_COUNT[Achse]				
Total traverse processes of an axis with jerk					
-		0		Double	r
Multi-line: yes	Axis index		maxnumGlobMachAxes		

aaJerkTime	$AA_JERK_TIME[Achse]				
Total traverse time of an axis with jerk					
-		0		Double	r
Multi-line: yes	Axis index		maxnumGlobMachAxes		

aaJerkTotal	$AA_JERK_TOT[Achse]				
Overall total of jerk of an axis					
-		0		Double	r
Multi-line: yes	Axis index		maxnumGlobMachAxes		

aaLeadP					
Actual lead value position					
-				Double	r
Multi-line: yes	Axis index		maxnumGlobMachAxes		

aaLeadPTurn	$AA_LEAD_P_TURN				
Current master value - position component lost as a result of modulo reduction					
-	0	0		UWord	r
Multi-line: yes	Axis index		maxnumGlobMachAxes		

aaLeadSp					
Simulated lead value - position					
-				Double	r
Multi-line: yes	Axis index		maxnumGlobMachAxes		

aaLeadSv					
Simulated leading value velocity					
-				Double	r
Multi-line: yes	Axis index		maxnumGlobMachAxes		

aaLeadTyp					
Source of the lead value 1: actual value 2: desired value 3: simulated value					
-				UWord	r
Multi-line: yes	Axis index		maxnumGlobMachAxes		

aaLeadV					
Actual lead value - velocity					
-				Double	r
Multi-line: yes	Axis index		maxnumGlobMachAxes		

aaLoad					
Drive load in % (611D only)					
%				Double	r
Multi-line: yes	Axis index		maxnumGlobMachAxes		

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	numGlobMachAxes	UWord	r
Multi-line: yes	Axis index		maxnumGlobMachAxes		

aaMm					
Latched probe position in the machine coordinate system					
-				Double	wr
Multi-line: yes	Axis index		maxnumGlobMachAxes		

aaMm1					
Access to measurement result of trigger event in the MCS					
-				Double	wr
Multi-line: yes	Axis index		maxnumGlobMachAxes		

aaMm2					
Access to measurement result of trigger event in the MCS					
-				Double	wr
Multi-line: yes	Axis index		maxnumGlobMachAxes		

aaMm3					
Access to measurement result of trigger event in the MCS					
-				Double	wr
Multi-line: yes	Axis index		maxnumGlobMachAxes		

aaMm4					
Access to measurement result of trigger event in the MCS					
-				Double	wr
Multi-line: yes	Axis index		maxnumGlobMachAxes		

aaOff					
Superimposed position offset from synchronous actions					
-				Double	r
Multi-line: yes	Axis index		maxnumGlobMachAxes		

aaOffLimit					
Limit for axial correction $AA_OFF reached (Note: for SYNACT only) 0: limit not reached 1: limit in positive axial direction reached 11: limit in negative axial direction reached					
-				UWord	r
Multi-line: yes	Axis index		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 index		maxnumGlobMachAxes		

aaOscillBreakPos1	$AA_OSCILL_BREAK_POS1[<Achse>]				
Oscillation interrupt position 1					
-				Double	r
Multi-line: yes	Axis index		maxnumGlobMachAxes		

aaOscillBreakPos2	$AA_OSCILL_BREAK_POS2[<Achse>]				
Oscillation interrupt position 2					
-				Double	r
Multi-line: yes	Axis index		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)					
-				Double	r
Multi-line: yes	Axis index		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)					
-				Double	r
Multi-line: yes	Axis index		maxnumGlobMachAxes		

aaOvr					
Axial override for synchronous actions					
-				Double	r
Multi-line: yes	Axis index		maxnumGlobMachAxes		

aaPlcOvr	$AA_PLC_OVR[Achse]				
Axial override specified by PLC for motion-synchronous actions					
-	100	0		Double	r
Multi-line: yes	Axis index		maxnumGlobMachAxes		

aaPolfa	$AA_POLFA				
The programmed retraction position of the single axis					
-				Double	r
Multi-line: yes	Axis index		maxnumGlobMachAxes		

aaPolfaValid	$AA_POLFA_VALID				
Indicates 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 index		maxnumGlobMachAxes		

aaPower					
Drive power in W (611D only)					
W				Double	r
Multi-line: yes	Axis index		maxnumGlobMachAxes		

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 index		maxnumGlobMachAxes		

aaSoftendn					
Software end position, negative direction					
-				Double	r
Multi-line: yes	Axis index		maxnumGlobMachAxes		

aaSoftendp					
Software end position, positive direction					
-				Double	r
Multi-line: yes	Axis index		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 index		maxnumGlobMachAxes		

aaSync					
Coupling state of the following axis with lead value coupling 0: not synchronized 1: synchronized coarse 2: synchronized fine 3: synchronized coarse and fine					
-				UWord	r
Multi-line: yes	Axis index		maxnumGlobMachAxes		

aaTorque					
Desired torque value in Nm (611D only)					
Nm				Double	r
Multi-line: yes	Axis index		maxnumGlobMachAxes		

aaTotalOvr	$AA_TOTAL_OVR[Achse]				
The total axial override for motion-synchronous actions					
-	100	0		Double	r
Multi-line: yes	Axis index		maxnumGlobMachAxes		

aaTravelCount	$AA_TRAVEL_COUNT[Achse]				
Total traverse processes of an axis					
-		0		Double	r
Multi-line: yes	Axis index		maxnumGlobMachAxes		

aaTravelCountHS	$AA_TRAVEL_COUNT_HS[Achse]				
Total traverse processes of an axis at high speed					
-		0		Double	r
Multi-line: yes	Axis index		maxnumGlobMachAxes		

aaTravelDist	$AA_TRAVEL_DIST[Achse]				
Total travel path of an axis in mm or degrees					
-		0		Double	r
Multi-line: yes	Axis index		maxnumGlobMachAxes		

aaTravelDistHS	$AA_TRAVEL_DIST_HS[Achse]				
Total travel path of an axis at high speed in mm or degrees					
-		0		Double	r
Multi-line: yes	Axis index		maxnumGlobMachAxes		

aaTravelTime	$AA_TRAVEL_TIME[Achse]				
Total traverse time of an axis in seconds					
-		0		Double	r
Multi-line: yes	Axis index		maxnumGlobMachAxes		

aaTravelTimeHS	$AA_TRAVEL_TIME_HS[Achse]				
Total traverse time of an axis at high speed in seconds					
-		0		Double	r
Multi-line: yes	Axis index		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 index		maxnumGlobMachAxes		

aaVactB					
Axis velocity in basic coordinate system					
mm/min, inch/min, user defined	0.0			Double	r
Multi-line: yes	Axis index		maxnumGlobMachAxes		

aaVactM					
Axis velocity in machine coordinate system					
mm/min, inch/min, user defined	0.0			Double	r
Multi-line: yes	Axis index		maxnumGlobMachAxes		

aaVc					
Additive correction value for path feed or axial feed					
-				Double	r
Multi-line: yes	Axis index		maxnumGlobMachAxes		

ackSafeMeasPos					
Confirmation of SI actual position 0 = not confirmed 0x00AC = confirmed					
-				UWord	wr
Multi-line: yes	Axis index		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 index		maxnumGlobMachAxes		

actFeedRate					S5
Actual value of axis-specific feedrate for positioning axes. Actual value of single axis feed for additional axes.					
-				Double	r
Multi-line: yes	Axis index		maxnumGlobMachAxes		

actIndexAxPosNo					
Current indexing position number 0 = no indexing position >0 = indexing position number					
-				UWord	r
Multi-line: yes	Axis index		maxnumGlobMachAxes		

actSpeedRel					
Actual value of rotary speed (referring to the maximum speed in %; for 611D in MD1401), for linear drives actual value of the velocity.					
%				Double	r
Multi-line: yes	Axis index		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 index		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 index		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 index		maxnumGlobMachAxes		

axisActiveInChan					
Flag indicating whether axis is active in this channel 0 = not active 1 = active					
-				UWord	r
Multi-line: yes	Axis index		maxnumGlobMachAxes		

axisFeedRateUnit					
Unit of the axis-specific feedrate 0 = mm/min 1 = inch/min 2 = degree/min					
-				UWord	r
Multi-line: yes	Axis index		maxnumGlobMachAxes		

chanAxisNoGap					
Display indicating whether axis exists, 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 index		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 index		maxnumGlobMachAxes		

cmdContrPos					
Desired value of position after fine interpolation					
mm, inch, degree, user defined				Double	r
Multi-line: yes	Axis index		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 index		maxnumGlobMachAxes		

cmdFeedRate					
Setpoint of axis-specific feedrate, if axis is a positioning axis. Single axis feedrate setpoint if the axis is an additional axis.					
-				Double	r
Multi-line: yes	Axis index		maxnumGlobMachAxes		

cmdSpeedRel					
Desired value of rotary speed. (referring to the max. speed in %; for 611D in MD 1401). For linear motors actual value of velocity.					
%				Double	r
Multi-line: yes	Axis index		maxnumGlobMachAxes		

contrConfirmActive					
Controller enable 0 = no controller enable 1 = controller enable					
-				UWord	r
Multi-line: yes	Axis index		maxnumGlobMachAxes		

contrMode					
Identifier for controller mode servo 0 = position control 1 = speed control 2 = stop 3 = park 4 = follow-up					
-				UWord	r
Multi-line: yes	Axis index		maxnumGlobMachAxes		

displayAxis					
Identifier indicating whether axis is displayed by MMC 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 index		maxnumGlobMachAxes		

distPerDriveRevol					
Distance per revolution. The physical unit is defined in measUnit (in this module).					
mm, inch, degree, user defined				Double	r
Multi-line: yes	Axis index		maxnumGlobMachAxes		

drive2ndTorqueLimit					
2nd torque limit. With linear motors: 2nd force limit 0 = not active 1 = active					
-				UWord	r
Multi-line: yes	Axis index		maxnumGlobMachAxes		

driveActMotorSwitch					
Actual motor wiring (star/delta) 0 = star 1 = delta					
-				UWord	r
Multi-line: yes	Axis index		maxnumGlobMachAxes		

driveActParamSet					
Number of the actual drive parameter set					
-		1	8	UWord	r
Multi-line: yes	Axis index		maxnumGlobMachAxes		

driveClass1Alarm					
Message ZK1 drive alarm 0 = no alarm set 1= alarm set (fatal error occured)					
-				UWord	r
Multi-line: yes	Axis index		maxnumGlobMachAxes		

driveContrMode					
Control mode of drive 0 = current control 1 = speed control					
-				UWord	r
Multi-line: yes	Axis index		maxnumGlobMachAxes		

driveCoolerTempWarn					
Heatsink temperature monitoring 0 = temperature OK 1 = overtemperature					
-				UWord	r
Multi-line: yes	Axis index		maxnumGlobMachAxes		

driveDesMotorSwitch					
Motor wiring selection (star/delta) 0 = star 1 = delta					
-				UWord	r
Multi-line: yes	Axis index		maxnumGlobMachAxes		

driveDesParamSet					
Desired parameter set of the drive					
-		1	8	UWord	r
Multi-line: yes	Axis index		maxnumGlobMachAxes		

driveFastStop					
Ramp-function generator rapid stop 0 = not stopped 1 = stopped					
-				UWord	r
Multi-line: yes	Axis index		maxnumGlobMachAxes		

driveFreqMode					
I/F mode					
-				UWord	r
Multi-line: yes	Axis index		maxnumGlobMachAxes		

driveImpulseEnabled					
Enable inverter impulse (checkback signal to impulseEnable) 0 = not enabled 1 = enabled					
-				UWord	r
Multi-line: yes	Axis index		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 index		maxnumGlobMachAxes		

driveIntegDisable					
Integrator disable 0 = not disabled 1 = disabled					
-				UWord	r
Multi-line: yes	Axis index		maxnumGlobMachAxes		

driveLinkVoltageOk					
State of the DC link voltage 0 = OK 1 = not OK					
-				UWord	r
Multi-line: yes	Axis index		maxnumGlobMachAxes		

driveMotorTempWarn					
Motor temperature warning 0 = temperature OK 1 = overtemperature					
-				UWord	r
Multi-line: yes	Axis index		maxnumGlobMachAxes		

driveNumCrcErrors					
CRC errors on the drive bus (transmission errors when writing data to the 611D; values may range up to FFFFH) 0 = no error					
-				UWord	r
Multi-line: yes	Axis index		maxnumGlobMachAxes		

driveParked					
Parking axis 0 = no parking axis 1 = parking axis					
-				UWord	r
Multi-line: yes	Axis index		maxnumGlobMachAxes		

drivePowerOn					
Drive switched on 0 = drive not switched on 1 = drive switched on					
-				UWord	r
Multi-line: yes	Axis index		maxnumGlobMachAxes		

driveProgMessages					
Configurable messages (via machine data)					
-				UWord	r
Multi-line: yes	Axis index		maxnumGlobMachAxes		

driveReady					
Drive ready 0 = drive not ready 1 = drive ready					
-				UWord	r
Multi-line: yes	Axis index		maxnumGlobMachAxes		

driveRunLevel					
Current state reached during the boot process (range: coarse state (0 to 5) * 100 + fine state (up to 22) Boot firmware ---> 0 XX Enter configuration ---> 1XX Hardware init, communication init Load, convert data ---> 2XX Change bus addressing ---> 3XX Prepare synchronization ---> 4XX Activate interrupt ---> 519 XX ==> fine state					
-				UWord	r
Multi-line: yes	Axis index		maxnumGlobMachAxes		

driveSetupMode					
Set-up mode 0 = not active 1 = active					
-				UWord	r
Multi-line: yes	Axis index		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 index		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 index		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 index		maxnumGlobMachAxes		

encChoice					
Active encoder 0 = does not exist 1 = encoder 1 2 = encoder 2					
-				UWord	r
Multi-line: yes	Axis index		maxnumGlobMachAxes		

fctGenState					
State of the function generator					
-				UWord	r
Multi-line: yes	Axis index		maxnumGlobMachAxes		

feedRateOvr					
Feedrate override (only if axis is a positioning axis)					
%				Double	r
Multi-line: yes	Axis index		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 index		maxnumGlobMachAxes		

fxsInfo	$VA_FXS_INFO[Achse]				
Additional information on travel to fixed stop if $VA_FXS[]=2, or OPI variable /N/SEMA/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 index		maxnumGlobMachAxes		

fxsStat					
State after travelling to fixed stop 0 = normal control 1 = fixed stop reached 2 = failed					
-				UWord	r
Multi-line: yes	Axis index		maxnumGlobMachAxes		

handwheelAss					
Number of handwheel assigned to an axis 0 = no handwheel assigned 1 to 3 = handwheel number					
-		0	3	UWord	r
Multi-line: yes	Axis index		maxnumGlobMachAxes		

impulseEnable					
Impulse enable for drive 0 = not enabled 1 = enabled					
-				UWord	r
Multi-line: yes	Axis index		maxnumGlobMachAxes		

index					
Absolute axis index referred to machine data					
-				UWord	r
Multi-line: yes	Axis index		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 index		maxnumGlobMachAxes		

kVFactor					
position control gain factor					
16.667 1/s				Double	r
Multi-line: yes	Axis index		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 index		maxnumGlobMachAxes		

logDriveNo					
Drive assignment (logical drive number) 0 = not available 1 to 15 = drive number					
-		0	15	UWord	r
Multi-line: yes	Axis index		maxnumGlobMachAxes		

measFctState					
State of the probing function					
-				UWord	r
Multi-line: yes	Axis index		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 index		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 index		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 index		maxnumGlobMachAxes		

measUnit					
Unit for service values of the drives 0 = mm 1 = inch 2 = grd					
-				UWord	r
Multi-line: yes	Axis index		maxnumGlobMachAxes		

paramSetNo					
Number of parameter set					
-		1	8	UWord	r
Multi-line: yes	Axis index		maxnumGlobMachAxes		

preContrFactTorque					
Feed forward control factor torque					
Nm				Double	r
Multi-line: yes	Axis index		maxnumGlobMachAxes		

preContrFactVel					
Feed forward control factor velocity					
-				Double	r
Multi-line: yes	Axis index		maxnumGlobMachAxes		

preContrMode					
Feed forward control mode 0 = inactive 1 = velocity feed forward 2 = torque feed forward					
-				UWord	r
Multi-line: yes	Axis index		maxnumGlobMachAxes		

PRESETActive					
Preset state 0 = no preset active 1 = preset active					
-				UWord	r
Multi-line: yes	Axis index		maxnumGlobMachAxes		

PRESETVal					
The function PRESETON (...) programs a zero 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 MMC.					
mm, inch, user defined				Double	wr
Multi-line: yes	Axis index		maxnumGlobMachAxes		

progIndexAxPosNo					
Programmed indexing position number 0 = no indexing position >0 = indexing position number					
-				UWord	r
Multi-line: yes	Axis index		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 index		maxnumGlobMachAxes		

refPtBusy					
Axis is being referenced 0 = axis is not being referenced 1 = axis is being referenced					
-				UWord	r
Multi-line: yes	Axis index		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 index		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). A set bit means: SW release 3.1 and earlier: Bit0: at least 1 measuring system has been referenced Bit1: current measuring system requires referencing SW release 3.2 and later: Bit0: current measuring system has been referenced Bit1: current measuring system requires referencing (The busy signal effects the status)					
-	Achsindex			UWord	r
Multi-line: no			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 index		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	wr
Multi-line: yes	Axis index		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	wr
Multi-line: yes	Axis index		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 index		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 /N/SEMA/safeAcceptTestMode to the NCK. 0ACH: NCK has active acceptance test mode					
-	0	0	0FFH	UWord	r
Multi-line: yes	Axis index		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 index		maxnumGlobMachAxes		

safeActVeloDiff					
Current speed difference betw. NCK and drive monitoring channels					
mm/min, inch/min, user defined	0.0			Double	r
Multi-line: yes	Axis index		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		

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 0 = not activated 1 = activated					
-				UWord	r
Multi-line: yes	Axis index		maxnumGlobMachAxes		

safeInputSig					
Safe input signals of the axis					
-				UWord	r
Multi-line: yes	Axis index		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 index		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 index		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 index		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 index		maxnumGlobMachAxes		

safeOutputSig					
Safe output signals of the axis					
-				UWord	r
Multi-line: yes	Axis index		maxnumGlobMachAxes		

safeOutputSig2					
Safe output signals part 2					
-		0	0xffff	UWord	r
Multi-line: no			maxnumGlobMachAxes		

safeOutputSigDrive					
Safe output signals of the drive					
-				UWord	r
Multi-line: yes	Axis index		maxnumGlobMachAxes		

safeOutputSigDrive2					
Safe output signals of the drive part 2					
-		0	0xffff	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 index		maxnumGlobMachAxes		

spec					
Axis specification 0 = path axis 1 = positioning axis					
-				UWord	r
Multi-line: yes	Axis index		maxnumGlobMachAxes		

subSpec					T1
Subspecification 0 = normal axis 1 = indexing axis					
-				UWord	r
Multi-line: yes	Axis index		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 index		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 index		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 index		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 index		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 index		maxnumGlobMachAxes		

trackErrContr					
Position controller difference (actual value / desired value of position)					
mm, inch, degree, user defined				Double	r
Multi-line: yes	Axis index		maxnumGlobMachAxes		

trackErrDiff					
Contour deviation (difference actual value of position and calculated dynamical model)					
mm, inch, degree, user defined				Double	r
Multi-line: yes	Axis index		maxnumGlobMachAxes		

type					
Axis type 1 = linear axis 2 = rotary axis 3 = spindle					
-				UWord	r
Multi-line: yes	Axis index		maxnumGlobMachAxes		

vaDistTorque	$VA_DIST_TORQUE[Achse]				
Disturbing torque/max. torque (motor end, York)					
%	0	-100	100	Double	r
Multi-line: yes	Axis index		maxnumGlobMachAxes		

vaDpe	$VA_DPE[x1]				
Status of power enable of a machine axis 0 - 1					
-	0	0	1	UWord	r
Multi-line: yes	Axis index		maxnumGlobMachAxes		

vaIm	$VA_IM[x]				
Encoder actual value in the machine coordinate system (measured active measuring system)					
-	0	0		Double	r
Multi-line: yes	Axis index		maxnumGlobMachAxes		

vaIm1	$VA_IM1[x]				
Actual value in the machine coordinate system (measured encoder 1)					
-	0	0		Double	r
Multi-line: yes	Axis index		maxnumGlobMachAxes		

vaIm2	$VA_IM2[x]				
Actual value in the machine coordinate system (measured encoder 2)					
-	0	0		Double	r
Multi-line: yes	Axis index		maxnumGlobMachAxes		

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 index		maxnumGlobMachAxes		

vaVactm					
Axis velocity actual value on the load side in the MCS					
-				Double	r
Multi-line: yes	Axis index		maxnumGlobMachAxes		

1.3.7 Area N, Mod. SSP: State data: Spindle

OEM-MMC: Linkitem /Nck/Spindle/...

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	$AC_CONSTCUT_S[n]				
Current constant cutting rate					
m/min, ft/min, user defined	0			Double	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		

actGearStage					
Actual gear stage of spindle					
-				UWord	r
Multi-line: yes	Spindle index		numSpindles		

actSpeed					
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 speed of the master spindle. The requested value for the master spindle differs from SSP:cmdSpeed only if G96 is active					
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					
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 = not active 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 associated 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		

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 state 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		

1.3.8 Area N, Mod. SSP2: State data: Spindle

OEM-MMC: Linkitem /Nck/LogicalSpindle/...

All state data that refer to a spindle, if a spindle converter (logical spindles) is active

acConstCutS	$AC_CONSTCUT_S[n]				
Current constant cutting rate					
m/min, ft/min, user defined	0			Double	r
Multi-line: yes	Logical spindle index		numSpindlesLog		

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	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 speed of the master spindle. The requested value for the master spindle differs from SSP:cmdSpeed only if G96 is active					
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	{$GWPS}				
SUG programming active (SUG is the function "constant perimeter speed of grinding wheel) 0 = not active 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.					
-				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		

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 state 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		

1.3.9 Area N, Mod. FA: Active NCU global frames

OEM-MMC: Linkitem /Nck/ActualFrame/...

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	$P_PFRAME[x,TR] / $P_ACTFRAME / $P_IFRAME					PA
Translation of an active zero 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 - 1) * maxnumGlobMachAxes + axis number		6 * maxnumGlobMachAxes			

mirrorImgActive	$P_PFRAME[x,MI] / $P_ACTFRAME / $P_IFRAME					PA
Mirroring enabled in an active zero offset 0 = mirroring not active 1 = mirroring active						
-					UWord	r
Multi-line: yes	(Frame index - 1) * maxnumGlobMachAxes + axis number		6 * maxnumGlobMachAxes			

rotation	$P_PFRAME[x,RT] / $P_ACTFRAME / $P_IFRAME					PA
Rotation of an active zero offset						
Degree					Double	r
Multi-line: yes	(Frame index - 1) * maxnumGlobMachAxes + axis number		6 * maxnumGlobMachAxes			

scaleFact	$P_PFRAME[x,SC] / $P_ACTFRAME / $P_IFRAME					PA
Scaling factor of an active zero offset						
-					Double	r
Multi-line: yes	(Frame index - 1) * maxnumGlobMachAxes + axis number		6 * maxnumGlobMachAxes			

1.3.10 Area N, Mod. FB: NCU global base frames

OEM-MMC: Linkitem /Nck/BaseFrame/...

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 zero offset (the physical unit is defined in basicLengthUnit in module Y in area N).					
mm, inch, user defined				Double	wr
Multi-line: yes	(Frame index - 1) * 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	wr
Multi-line: yes	(Frame index - 1) * 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 zero offset 0: Mirroring not active 1: Mirroring active					
-				UWord	wr
Multi-line: yes	(Frame index - 1) * 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 zero offset					
Degree				Double	wr
Multi-line: yes	(Frame index - 1) * maxnumGlobMachAxes + axis number		$MN_MM_NUM_GLOBAL_BASE_FRAMES * maxnumGlobMachAxes		

scaleFact	$P_NCBFR[x,SC] x=FrameNo, y=Axis				PA
Scaling factor of a settable zero offset					
-				Double	wr
Multi-line: yes	(Frame index - 1) * maxnumGlobMachAxes + axis number		$MN_MM_NUM_GLOBAL_BASE_FRAMES * maxnumGlobMachAxes		

1.3.11 Area N, Mod. FU: NCU global settable frames

OEM-MMC: Linkitem /Nck/UserFrame/...
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	$P_UIFR[x,y,TR] x=FrameNo,y=Axis				PA
Translation of settable zero offset (the physical unit is defined in basicLengthUnit in module Y in area N).					
mm, inch, user defined				Double	wr
Multi-line: yes	(Frame index - 1) * maxnumGlobMachAxes + axis number		$MN_MM_NUM_GLOBAL_USER_FRAMES * maxnumGlobMachAxes		

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	wr
Multi-line: yes	(Frame index - 1) * maxnumGlobMachAxes + axis number		$MN_MM_NUM_GLOBAL_USER_FRAMES * maxnumGlobMachAxes		

mirrorImgActive	$P_UIFR[x,y,MI] x = FrameNo,y=Axis				PA
Mirroring enabled in settable zero offset 0 = mirroring not active 1 = mirroring active					
-				UWord	wr
Multi-line: yes	(Frame index - 1) * maxnumGlobMachAxes + axis number		$MN_MM_NUM_GLOBAL_USER_FRAMES * maxnumGlobMachAxes		

scaleFact	$P_UIFR[x,y,SC] x = FrameNo,y=Axis				PA
Scaling factor of settable zero offset					
-				Double	wr
Multi-line: yes	(Frame index - 1) * maxnumGlobMachAxes + axis number		$MN_MM_NUM_GLOBAL_USER_FRAMES * maxnumGlobMachAxes		

1.3.12 Area N, Mod. YFAFL: NCK instruction groups (Fanuc)

OEM-MMC: Linkitem /Nck/FunctionGroupingFanuc/...

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		/N/YFAFL/Gruppe_NUM		

Gruppe_NUM					
Number of Fanuc-G functions in respective group					
-		0		UWord	r
Multi-line: yes	1		1		

1.3.13 Area B, Mod. S: Mode-group-specific state data

OEM-MMC: Linkitem /Bag/State/...

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)

opMode	DB11, DBX6.0-6.2				
Active mode 0 = JOG 1 = MDA 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					

1.4 State data of channel

1.4.1 Area C, Mod. M: Channel-specific machine data

OEM-MMC: Linkitem /Channel/Drive/...

Channel-specific machine data

MDS_CHAN_NAME	MD 20000: CHAN_NAME				
Channel name					
-				String[16]	wr
Multi-line: no			1		

1.4.2 Area C, Mod. S: Channel-specific status data

OEM-MMC: Linkitem /Channel/State/...

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)

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) * 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		

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		

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		

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		

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	UDoubleword	wr
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	UDoubleword	wr
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	UDoubleword	wr
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	UDoubleword	wr
Multi-line: yes	Axis index		numMachAxes		

aaMeasPoint1	$AA_MEAS_POINT1				
1st measuring point for workpiece and tool measurement					
mm, inch, user defined	0			Double	wr
Multi-line: yes	Axis index		numMachAxes		

aaMeasPoint2	$AA_MEAS_POINT2				
2nd measuring point for workpiece and tool measurement					
mm, inch, user defined				Double	wr
Multi-line: yes	Axis index		numMachAxes		

aaMeasPoint3	$AA_MEAS_POINT2				
3rd measuring point for workpiece and tool measurement					
mm, inch, user defined				Double	wr
Multi-line: yes	Axis index		numMachAxes		

aaMeasPoint4	$AA_MEAS_POINT4				
4th measuring point for workpiece and tool measurement					
mm, inch, user defined				Double	wr
Multi-line: yes	Axis index		numMachAxes		

aaMeasSetangle	$AA_MEAS_SETANGLE				
Setpoint angle of an axis					
Degree, user defined				Double	wr
Multi-line: yes	Axis index				

aaMeasSetpoint	$AA_MEAS_SETPOINT				
Setpoint position of edge, corner or hole					
mm, inch, user defined				Double	wr
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	UDoubleword	wr
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		

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		

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		

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		

acDtepb	$AC_DTEPB				
Distance-to-go of infeed during oscillation in the BCS (Note: for SYNACT only)					
-				Double	r
Multi-line: yes	1		1		

acDtepw	$AC_DTEPW				
Distance-to-go of infeed during oscillation in the WCS (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		

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		

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			UDoubleword	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			UDoubleword	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	UDoubleword	wr
Multi-line: no					

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	UDoubleword	wr
Multi-line: yes	1		1		

acMeasChbfr	$AC_MEAS_CHBFR				
Channel basic frame screen form for setting up the new frame					
-	0	0		UDoubleword	wr
Multi-line: no					

acMeasChsfr	$AC_MEAS_CHSFR				
System frame bit screen form for setting up the new frame					
-	0	0		UDoubleword	wr
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	wr
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	UDoubleword	wr
Multi-line: yes	1		1		

acMeasDNumber	$AC_MEAS_D_NUMBER				
Selected tool edge number					
-		0		UDoubleword	wr
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	UDoubleword	wr
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	UDoubleword	wr
Multi-line: yes	1		1		

acMeasInput	$AC_MEAS_INPUT[n]				
Data for the workpiece and tool measurement					
-	0			Double	wr
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	UDoubleword	wr
Multi-line: yes	Measuring point no.		4		

acMeasNcbfr	$AC_MEAS_NCBFR				
Global basic frame screen form for setting up the new frame					
-	0	0		UDoubleword	wr
Multi-line: no					

acMeasP1Coord	$AC_MEAS_P1_COORD				
Coordinate system of the 1st measuring point 0: Work 1: BCS 2: MCS					
-	0	0		UDoubleword	wr
Multi-line: no					

acMeasP2Coord	$AC_MEAS_P2_COORD				
Coordinate system of the 2nd measuring point 0: Work 1: BCS 2: MCS					
-	0	0		UDoubleword	wr
Multi-line: no					

acMeasP3Coord	$AC_MEAS_P3_COORD				
Coordinate system of the 3rd measuring point 0: Work 1: BCS 2: MCS					
-	0	0		UDoubleword	wr
Multi-line: no					

acMeasP4Coord	$AC_MEAS_P4_COORD				
Coordinate system of the 4th measuring point 0: Work 1: BCS 2: MCS					
-	0	0		UDoubleword	wr
Multi-line: no					

acMeasPframe	$AC_MEAS_PFRAME				
Programmable frame is not included					
-	0	0	1	UDoubleword	wr
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		UDoubleword	wr
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	UDoubleword	wr
Multi-line: yes	1		1		

acMeasSetCoord	$AC_MEAS_SET_COORD				
Coordinate system of the set point 0: Work 1: BCS 2: MCS					
-	0	0		UDoubleword	wr
Multi-line: no					

acMeasTNumber	$AC_MEAS_T_NUMBER				
Selected tool number					
-		0		UDoubleword	wr
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		UDoubleword	wr
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	UDoubleword	wr
Multi-line: yes	1		1		

acMeasUifr	$AC_MEAS_UIFR				
Settable data management frame for setting up the new frame					
-	0	0	99	UDoubleword	wr
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		UDoubleword	wr
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	wr
Multi-line: yes	1		1		

acOvr	$AC_OVR				
Path override for synchronous actions (Note: for SYNACT only)					
-				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		

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		

acPRTimeA					
For simulation: Estimation of program runtime in seconds - downtime					
s, user defined				Double	wr
Multi-line: yes	1		1		

acPRTimeM					
For simulation: Estimation of program runtime in seconds - machining time					
s, user defined				Double	wr
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		

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		

actDLNumber	??				
Number of active total offset DL					
-				UWord	r
Multi-line: yes	1				

actDNumber	$P_TOOL				
Number of active tool edge					
-		0	9	UWord	r
Multi-line: no					

actDNumberFanuc					
With programming in ISO Dialect mode: Offset memory number radius. Assigned only in conjunction with ISO Dialect M external language.					
-				UWord	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	wr
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					

actFrameIndex	$P_UIFRNUM				
Index of the active set frame (index in G group 8 "Settable zero 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					

actHNumberFanuc					
With programming in ISO Dialect mode: Offset memory number length. Assigned only in conjunction with ISO Dialect M external language.					
-				UWord	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		

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		

actMasterToolHolderNo	$P_MTHNUM, $AC_MTHNUM bzw. $P_MSNUM, $AC_MSNUM				
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		

acTotalOvr	$AC_TOTAL_OVR				
Total path override for synchronized actions					
-	100	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	wr
Multi-line: no					

acTrafo	$AC_TRAFO				
Code number of the active transformation (encoded as for $AC_TRAFO)					
-				UWord	r
Multi-line: yes	1		1		

acTrafoParSet	$AC_TRAFO_PARSET				
Number of current transformation data record. The variable is '0' if no transformation is active.					
-	0			UWord	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	wr
Multi-line: yes	1		1		

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					

actTransform					
Active transformation					
-	\0			String[32]	r
Multi-line: yes	1		1		

acVactB	$AC_VACTB				
Path velocity in basic coordinate system					
mm/min, inch/min, user defined	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		

aLinkTransRate	$A_LINK_TRANS_RATE				
Link transfer rate Number of bytes that can still be transferred in the current IPO cycle via the NCU link communication.					
-		0		UWord	r
Multi-line: yes	Spindle no. or toolholder no.		max. Spindelnr oder WZ-Halter-Nr.		

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					

aTcAckC	$AC_TC_ACKC				
Counter variable: aTcAckC (AcknowledgeCounter) is incremented by 1 every time the PLC acknowledges a tool management command.					
-	0	0		UWord	wr
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	wr
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		

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		

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		

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	UDoubleword	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 With system variable blockTypeInfo, additional information on variable blockType can be requested. Depending on the value of system variable blockType, different values are 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 sets when intersection not found 4003: Pointed circle on the inner corners (with TRACYL only) 4004: Bypass circle (or conic) on outer corners 4005: Approach blocks for offset suppression 4006: Approach blocks for reactivation of TRC 4007: Block separation when 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 and without retraction movement. 6002: TLIFT block with non-linear tangential axis (polynomial) and 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 will only start, if certain retraction position has been reached. 7. Path segmentation: blockType = 7 7001: Programmed path segmentation without punching/nibbling to be active. 7002: Programmed path segmentation with active punching/nibbling. 7003: Automatic internally generated path segmentation. 8. Compile cycles: blockType = 8 In this case, system variable $AC_BLOCKTYPEINFO includes the ID of the compile cycles application that created the block.					
-	0	0		UDoubleword	r
Multi-line: yes	1		1		

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		UDoubleword	r
Multi-line: yes	1		1		

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		

chanStatus	DB21-28, DBX35.5-DBX35.7				K1
Channel status 0 = RESET 1 = active 2 = interrupted					
-				UWord	r
Multi-line: no					

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		

cmdDwellTime					
Programmed dwell time See timeOrRevolDwell					
-	0	0		Double	r
Multi-line: yes	1		1		

cmdFeedRateIpo					
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					

cmdTrafoParSetS	$P_TRAFO_PARSET				
Number of programmed transformation data record for block search The variable is '0' if no transformation is active.					
-	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		

corrBlActive					
Incorrect block has occurred (correction block) 0 = no incorrect block 1 = incorrect block					
-				UWord	r
Multi-line: no					

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		

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					

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					

extProgActive	DB21-28, DBB32.0				
Flag indicating whether program execution from external is active. 0 = not active 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					

findBlActive	DB21-28, DBX33.4				K1
Code whether block search is active. 0 = not active 1 = active					
-				UWord	r
Multi-line: no					

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		

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 MMC 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 MMC 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		

ncStartCounter					
Counter for the NC-start key. Pressing this key increments the variable 'ncStartCounter'. The value of the variable can be ignored, the MMC 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		

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		

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		

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 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 TO. -1 TMMG not available					
-	0	0	numMachAxes	UDoubleword	r
Multi-line: no			1		

pOffn	$P_OFFN				
Last programmed offset normal					
-	0			Double	r
Multi-line: no					

progDLNumberS	??				
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				

progDuploNumber					
Duplo number of programmed tool (does not yet have to be active)					
-	0			UWord	r
Multi-line: no			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		

progWaitForEditUnlock					
The interpreter is waiting until the editor has saved the specified parts program and has enabled it by means of _N_F_MODE					
-	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		

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		

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) 0x1000 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		UDoubleword	r
Multi-line: yes	1		1		

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	$P_AD[2]				
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		

pToolO	$P_TOOLO				
Supplies the current tool orientation The orientation vector is normalized, i.e. it has the value 1.					
-	0	-1	1	Double	r
Multi-line: yes	1: X component 2: Y component 3: Z component		3		

rapFeedRateOvr					
Rapid traverse override					
%				Double	r
Multi-line: no					

remainDwellTime					
Remaining dwell time See timeOrRevolDwell					
-	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	wr
Multi-line: no					

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		

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	wr
Multi-line: yes	1		1		

serupro MasterNcuNo					
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 serupro MasterChanNo defines the master channel. serupro MasterNcuNo specifies the master channel in more detail if it is not on the active NCU.					
-	0	0	$MN_MM_LINK_NUM_OF_MODULES	UWord	wr
Multi-line: yes	1		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	wr
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	UDoubleword	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) or a program-specific start disable (see _N_F_MODE).					
-	0	0		UWord	r
Multi-line: yes	1		1		

stopCond		

NC in stop state
NC stop state

0 = No stop state
1 = NC not ready
2 = Mode group not ready
3 = EMERGENCY STOP active
4 = Alarm with stop active
5 = M0 / M1 active
6 = Block ended in single block mode
7 = NC stop active
8 = Read-in enable missing
9 = Feed enable missing
10 = Dwell time active
11 = Aux. funct. acknowledgement missing
12 = Axis enable missing
13 = Exact stop not reached
14 = Wait for positioning axis
15 = Wait for spindle
16 = Wait for another channel
17 = Wait for feed override
18 = NC block faulty or user alarm
19 = Wait for NC blocks from external source
20 = Wait for synchronized action
21 = Block search active
22 = Spindle enable missing
23 = Axis feed override 0
24 = Wait for tool change acknowledgement
25 = Gear ratio change
26 = Wait for position control
27 = Wait for thread first cut
28 = Reserved
29 = Wait for punching
30 = Wait for safe operation
31 = No channel ready; from SW 4.1
32 = Oscillation active; from SW 4.1 to SW 6.3
32 = Reserved; from SW 6.3
33 = Axis replacement active; Block change prevented because axis replacement was active from SW 4.1
34 = Axis container rotation; from SW 4.4
35 = AXCT: Slave axis active; Axis container replacement from SW 5.2
36 = AXCT: Master axis active; Axis container replacement from SW 5.2
37 = AXCT: Follow-up active; Axis container replacement from SW 5.2
38 = AXCT: Internal status change; Axis container replacement from SW 5.2
The following internal status changes may cause this waiting status:
- Switch on the position controller
  - Request the zero mark
  - Reference point approach active
- Parameter block change active
  - Measuring cycle system change active
  - In-process measurement active
  - Servo enable removed

39= AXCT: Axis/spindle disable; Axis container replacement from SW 5.2
40 = AXCT: Corr. motion active; Axis container replacement: overlaid motion from SW 5.2
41 = AXCT: Axis replacement active; Axis container replacement from SW 5.2
42 = AXCT axis interpolator active; Axis container replacement from SW 5.2
43 = Wait for compile cycle; from SW 5.2
44 = Access to system variables; from SW 5.3
45 = Search target found; block search has found search target and NCK has stopped.
46 = Rapid retraction activated; from SW 6.2.
47 = AXCT: Wait for spindle stop; axis container replacement from SW 6.2.
48 = Machine data match; New config, from SW 6.2.
49 = Axis replacement: coupled axis; from SW 6.3
50 = Axis replacement: Liftfast active; from SW 6.3
51 = Axis replacement: New config active; from SW 6.3
52 = Axis replacement: AXCTSW active; from SW 6.3
53 = Axis replacement: Waitp active; from SW 6.3
54 = Axis in another channel; axis replacement from SW 6.3

55 = Axis replacement: axis PLC axis; from SW 6.3 56 = Axis replacement: axis reciprocating axis; from SW 6.3 57 = Axis replacement: axis jog axis; from SW 6.3 58 = Axis replacement: command axis; from SW 6.3 59 = Axis replacement: axis OEM axis; from SW 6.3 60 = Linked slave axis; axis replacement from SW 6.3 61 = Coupled-motion slave axis; axis replacement from SW 6.3 62 = Linked slave axis; axis replacement from SW 6.3 63 = Stop: associated M0 active; from SW 6.3 64 = Stop: associated M1 active; from SW 6.3 65 = Axis at limit stop; from SW 6.3 66 = Master-slave changeover active; from SW 6.3 67 = Axis replacement: axis single axis; from SW 6.3 68 = Stop run has reached the stop block; from SW 7.1 69 = Synchronous run: synchronous spindle; from SW 6.4 70 = Deactivation position of synchronous spindle; from SW 6.4 71 = Wait for release of transformation axis; from SW 7.1					
-				UWord	r
Multi-line: no					

stopCondPar					
Supplementary parameters for variable stopCond. stopCondPar has the default value 0. If stopCond takes one of the following values, variable stopCondPar contains supplementary information:					
-				UWord	r
Multi-line: no					

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	wr
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		

toolCounterM					
Counter for modifications to magazine data assigned to the channel (analog toolCounter).					
-				UWord	r
Multi-line: yes	1		1		

toolHolderData	GETSELT, GETEXET				
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 three 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 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. An array access is possible to toolHolderData, with which the data of all numToolHolders tool holders can be read at one time. 0 will be returned for line 1 only, if tool magazine management is not active.					
-	0	0		Double	r
Multi-line: yes	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		numToolHolderParams * numToolHolders		

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	wr
Multi-line: no					

transfActive	DB21-28, DBX33.6				K1, M1
Transformation active 0 = not active 1 = active					
-				UWord	r
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		

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		

1.4.3 Area C, Mod. SINF: Part-program-specific status data

OEM-MMC: Linkitem /Channel/ProgramModification/...

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 = not active 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 = not active 1 = active					
-				UWord	r
Multi-line: no					

rapFeedRateOvrActive					
ROV rapid traverse override 0 = not active 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	wr
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 = not active 1 = active					
-				UWord	r
Multi-line: no					

1.4.4 Area C, Mod. SPARP: Part program information

OEM-MMC: Linkitem /Channel/ProgramInfo/...

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. If search run is active, then search run block is displayed. With DISPLOF an empty string is returned; with search run the subroutine call.					
-				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		

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					

circleCenter					
Center of the circle (WCS)					
-				Double	r
Multi-line: yes	Line index 1 - 3 for geometry axis 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 geo axis		3		

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 geo 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		UDoubleword	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		UDoubleword	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 geo-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 geo-axis		3		

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		

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		

seekw					
First line enabled for modification in part program					
-	0	0		UWord	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		

workPandProgName					
Workpiece name and name of current program.					
-				String[160]	r
Multi-line: yes	1		1		

workPName					
Name of the active workpiece					
-				String[32]	r
Multi-line: no			1		

workPNameLong					
Workpiece name of active workpiece					
-				String[128]	r
Multi-line: no					

1.4.5 Area C, Mod. SPARPP: Program pointer in automatic operation

OEM-MMC: Linkitem /Channel/ProgramPointer/...

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 12 rows. This makes it possible to address the main program level and 11 subroutine levels (incl. ASUP levels).

The array indices (row indices) mean:

1 = Main program
2 = 1st subroutine level
3 = 2nd subroutine level
4 = 3rd subroutine level
5 = 4th subroutine level
6 = 5th subroutine level
7 = 6th subroutine level
8 = 7th subroutine level
9 = 1st asynchronous subroutine level
10 = 2nd asynchronous subroutine level
11 = 3rd asynchronous subroutine level
12 = 4th asynchronous subroutine level

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		12		

blockLabel					
Block label					
-				String[32]	r
Multi-line: yes	Index of program level		12		

blockNoStr					
Block number [:][N]<No>					
-				String[12]	r
Multi-line: yes	Index of program level		12		

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		12		

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). 0 = DISPLAY OFF for the program level 1 = DISPLAY ON for the program level					
-	0			UWord	r
Multi-line: yes	Index of the program level		12		

extProgBufferName					
Name of FIFO buffer for execution from external source					
-				String[160]	wr
Multi-line: yes	Index of program level		12		

extProgFlag					
Indicates whether programs are being executed externally 0: Program is being processed from NCK program memory 1: Program is being executed externally					
-				UWord	r
Multi-line: yes	Index of program level		12		

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		12		

progName					
Program name					
-				String[32]	r
Multi-line: yes	Index of program level		12		

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		12		

seekw					
First line enabled for modification in part program					
-	0	0		UWord	r
Multi-line: yes	Index of program level		12		

workPandProgName					
Workpiece name and name of current program.					
-				String[160]	r
Multi-line: yes	Index of program level		12		

workPName					
Workpiece name = path name in the NCK file structure					
-				String[32]	r
Multi-line: yes	Program level index		12		

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	Program level index		12		

1.4.6 Area C, Mod. SPARPI: Program pointer on interruption

OEM-MMC: Linkitem /Channel/InterruptionSearch/...

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. ASUP levels).
The array indices (row indices) mean:

1 = main program
2 = 1st subroutine level
3 = 2nd subroutine level
4 = 3rd subroutine level
5 = 4th subroutine level
6 = 5th subroutine level
7 = 6th subroutine level
8 = 7th subroutine level
9 = 1st asynchronous subroutine level
10 = 2nd asynchronous subroutine level
11 = 3rd asynchronous subroutine level
12 = 4th asynchronus subroutine level

forward					
Search direction 2 = forwards					
-				UWord	r
Multi-line: yes	Index of program level		12		

invocCount					
Actual value of the subroutine call counter. Is always 1 for the main program					
-				UWord	r
Multi-line: yes	Index of program level		12		

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	Program level index		12		

progName					
Program name					
-				String[32]	r
Multi-line: yes	Index of program level		12		

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		12		

searchType					
Search type 5 = search pointer block-oriented (searching for line feed characters)					
-				UWord	r
Multi-line: yes	Index of program level		12		

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		12		

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		12		

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		12		

1.4.7 Area C, Mod. SPARPF: Program pointers for block search and stop run

OEM-MMC: Linkitem /Channel/Search/...

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 MMC (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 = 1st subroutine level for search run
3 = 2nd subroutine level for search run
4 = 3rd subroutine level for search run
5 = 4th subroutine level for search run
6 = 5th subroutine level for search run
7 = 6th subroutine level for search run
8 = 7th subroutine level for search run
9 = 1st asynchronous subroutine level for search run
10 = 2nd asynchronous subroutine level for search run
11 = 3rd asynchronous subroutine level for search run
12 = 4th asynchronous subroutine level for search run

101 = main program level for stop run
102 = 1st subroutine level for stop run
103 = 2nd subroutine level for stop run
104 = 3rd subroutine level for stop run
105 = 4th subroutine level for stop run
106 = 5th subroutine level for stop run
107 = 6th subroutine level for stop run
108 = 7th subroutine level for stop run
109 = 1st asynchronous subroutine level for stop run
110 = 2nd asynchronous subroutine level for stop run
111 = 3rd asynchronous subroutine level for stop run
112 = 4th asynchronous subroutine level for stop run

forward					
Search direction Search direction "backwards" is only possible in the mode without calculation 1 = backwards (without calculation) 2 = forwards					
-				UWord	wr
Multi-line: yes	Index of program level		12		

invocCount					
Actual value of the subroutine call counter. Is always 1 for the main program.					
-				UWord	wr
Multi-line: yes	Index of program level		12		

plcStartReason					
Specifies for the SERUPRO function which channel has to be started by the PLC so that the current channel starts.					
-	0	0		UWord	wr
Multi-line: yes	Program level index		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]	wr
Multi-line: yes	Index of the program level		12		

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]	wr
Multi-line: yes	Index of program level		12		

searchType					
Search type 1 = block number 2 = label 3 = string 4 = program level 5 = search pointer block-oriented (searching for line feeds)					
-				UWord	wr
Multi-line: yes	Index of program level		12		

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	wr
Multi-line: yes	Index of program level		12		

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	wr
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]	wr
Multi-line: yes	Index of program level		12		

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]	wr
Multi-line: yes	Index of program level		12		

1.4.8 Area C, Mod. SSYNAC: Synchronous actions

OEM-MMC: Linkitem /Channel/SelectedFunctions/...

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.

blockNoStrAct					
Block number of the current block if synchronous actions are active					
-				String[12]	r
Multi-line: yes	No of the synchronous action		/C/SSYNAC/numSynAct		

blockNoStrProg					
Number of the block where the synchronous action has been programmed					
-				String[12]	r
Multi-line: yes	No. of the synchronous action		/C/SSYNAC/numSynAct		

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		

id					
ID of the synchronous action; value 0 means no ID defined					
-				UWord	r
Multi-line: yes	No. of the synchronous action		/C/SSYNAC/numSynAct		

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		

numSynAct					
Number of synchronous actions					
-				UWord	r
Multi-line: yes	1		1		

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		

Tadr					
Active T-number. Only one T-number can be active at any a time.					
-				UWord	r
Multi-line: no			1		

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		

Tval					
T-function value					
-				Long Integer	r
Multi-line: no			1		

typStatus					
Type and state of the synchronous action Bits0-7 describe the state: Bit0: active Bit1: lock Bits8-15 describe the type: Bit8: static Bit9: modal Bit10: blockwise					
-				UWord	r
Multi-line: yes	Number of the synchronous action		/C/SSYNAC/numSynAct		

1.4.9 Area C, Mod. SYNACT: Channel-specific synchronous actions

OEM-MMC: Linkitem /Channel/SelectedFunctions/...

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 + /C/SYNACT/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 + /C/SYNACT/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 + /C/SYNACT/numSynAct		

numSynAct					
Number of synchronous actions					
-				UWord	r
Multi-line: yes	(protection level) * 1000 + 1		7 * 1000 + 1		

typStatus					
Type and state of the synchronous action Bits0-7 describe the state: Bit0: active, i.e. condition fullfilled, action is being executed Bit1: lock, i.e. action is locked by PLC or Synact Bits8-15 describe the type: Bit8: static Bit9: modal Bit10: blockwise (to be recognized by id=0)					
-				UWord	r
Multi-line: yes	(Protection level) * 1000 + no. of the synchronous action		7 * 1000 + /C/SYNACT/numSynAct		

1.4.10 Area C, Mod. SNCF: Active G functions

OEM-MMC: Linkitem /Channel/SelectedFunctions/...

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					
Active G-function of the correponding group for block search with calculation (The value is the index of the active G-function within the group) Attention: This variable is available for protocolling block search events only, 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					
Active G-function of the correponding group for block search with calculation Attention: This variable is available for protocolling block search events only, but not for the Variable Service.					
-				String[16]	r
Multi-line: yes	G group number		numGCodeGroups		

1.4.11 Area C, Mod. NIB: State data: Nibbling

OEM-MMC: Linkitem /Channel/Nibbling/...

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 (PONS from SW 4.1) 4 = rapid nibbling active (SONS from SW 4.1)					
-				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		

1.4.12 Area C, Mod. FB: Channel-specific base frames

OEM-MMC: Linkitem /Channel/BaseFrame/...

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 zero offset (the physical unit is defined in basicLengthUnit in module Y in area N).					
mm, inch, user defined				Double	wr
Multi-line: yes	(Frame index - 1) * (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	wr
Multi-line: yes	(Frame index - 1) * (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 zero offset 0: Mirroring not active 1: Mirroring active					
-				UWord	wr
Multi-line: yes	(Frame index - 1) * (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 zero offset					
Degree				Double	wr
Multi-line: yes	(Frame index - 1) * (numGeoAxes + numAuxAxes) + axis number		$MC_MM_NUM_BASE_FRAMES * (numGeoAxes + numAuxAxes)		

scaleFact	$P_CHBFR[x,y,SC] x=FrameNo, y=Axis				PA
Scaling factor of a settable zero offset					
-				Double	wr
Multi-line: yes	(Frame index - 1) * (numGeoAxes + numAuxAxes) + axis number		$MC_MM_NUM_BASE_FRAMES * (numGeoAxes + numAuxAxes)		

1.4.13 Area C, Mod. FS: Channel-specific system frames

OEM-MMC: Linkitem /Channel/SystemFrame/...

Those that there are, 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 6.3.
5 from and including SW 6.3.

linShift	$P_SETFR[Achse, TR]				
Translation					
mm, inch, user defined	0			Double	wr
Multi-line: yes	Frameindex		4 * (numGeoAxes+numAuxAxes)		

linShiftFine	$P_SETFR[Achse, SI]				
Fine offset					
mm, inch, user defined	0			Double	wr
Multi-line: yes	Frameindex		4 * (numGeoAxes+numAuxAxes)		

mirrorImgActive	$P_SETFR[Achse, MI]				
Mirroring 0: Mirroring inactive 1: Mirroring active					
-	0	0	1	UWord	wr
Multi-line: yes	Frameindex		4 * (numGeoAxes+numAuxAxes)		

rotation	$P_SETFR[Achse, RT]				
Rotation					
Degree	0			Double	wr
Multi-line: yes	Frameindex		4 * (numGeoAxes+numAuxAxes)		

scaleFact	$P_SETFR[Achse, SC]				
Scaling factor					
-	0			Double	wr
Multi-line: yes	Frameindex		4 * (numGeoAxes+numAuxAxes)		

1.4.14 Area C, Mod. AUXFU: Auxiliary functions

OEM-MMC: Linkitem /Channel//...

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.

extension					
Extension of the auxiliary function					
-	0	0		UWord	wr
Multi-line: yes	Group of auxiliary functions/view		3064		

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	wr
Multi-line: yes	Group of auxiliary functions/view		3064		

type					
Type of the auxiliary function, e.g. "M", "S", "T", "D", "F", "H", "L".					
-				String[2]	wr
Multi-line: yes	Group of auxiliary functions/view		3064		

valueDo					
Value of the auxiliary function. This value will be supplied, if "status" Bit15 = 1					
-	0	0		Double	wr
Multi-line: yes	Group of auxiliary functions/view		3064		

valueLo					
Value of the auxiliary function. This value will be supplied, if "status" Bit14 = 1					
-	0	0		UDoubleword	wr
Multi-line: yes	Group of auxiliary functions/view		3064		

1.5 State data of axes

1.5.1 Area C, Mod. SMA: State data: Machine axes

OEM-MMC: Linkitem /Channel/MachineAxis/...

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 (in this module).					
mm, inch, degree, user defined				Double	r
Multi-line: yes	Axis index		numMachAxes		

cmdToolBasePos					
Tool base position. 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 state 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		

1.5.2 Area C, Mod. SEMA: State data: Machine axes (extension of SMA)

OEM-MMC: Linkitem /Channel/MachineAxis/...

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.

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			UDoubleword	r
Multi-line: yes	Axis index		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 index		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 index		numMachAxes		

aaCoupAct	$AA_COUP_ACT[x] x = Spindle following				
Current coupling state of the slave spindle					
-				UWord	r
Multi-line: yes	Axis index		numMachAxes		

aaCoupOffs	$AA_COUP_OFFS[x] x = Spindle				
Position offset of the synchronous spindle desired value					
-				Double	r
Multi-line: yes	Axis index		numMachAxes		

aaCurr	$AA_CURR[x] x = Axis				
Actual value of the axis/spindle current in A (611D only)					
A				Double	r
Multi-line: yes	Axis index		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 index		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 index		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 index		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 index		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 index		numMachAxes		

aaEsrTrigger	$AA_ESR_TRIGGER				
Activation of "NC-controlled ESR" for PLC-controlled axis					
-	0	0	1	UWord	r
Multi-line: yes	Axis index		numMachAxes		

aaIbnCorr	$AA_IBN_CORR[<Achse>]				
Current BZS setpoint value of an axis including override components					
-	0			Double	r
Multi-line: yes	Axis index		numMachAxes		

aaIenCorr	$AA_IEN_CORR[<Achse>]				
Current SZS setpoint value of an axis including override components					
-	0			Double	r
Multi-line: yes	Axis index		numMachAxes		

aaJerkCount	$AA_JERK_COUNT[Achse]				
Total traverse processes of an axis with jerk					
-		0		Double	r
Multi-line: yes	Axis index		numMachAxes		

aaJerkTime	$AA_JERK_TIME[Achse]				
Total traverse time of an axis with jerk					
-		0		Double	r
Multi-line: yes	Axis index		numMachAxes		

aaJerkTotal	$AA_JERK_TOT[Achse]				
Overall total jerk of an axis					
-		0		Double	r
Multi-line: yes	Axis index		numMachAxes		

aaLeadP	$AA_LEAD_P[x] x = Axis				
Actual lead value position					
-				Double	r
Multi-line: yes	Axis index		numMachAxes		

aaLeadPTurn	$AA_LEAD_P_TURN				
Current master value - position component lost as a result of modulo reduction					
-	0	0		UWord	r
Multi-line: yes	Axis index		numMachAxes		

aaLeadSp	$AA_LEAD_SP[x] x = Axis				
Simulated lead value - position					
-				Double	r
Multi-line: yes	Axis index		numMachAxes		

aaLeadSv	$AA_LEAD_SV[x] x = Axis				
Simulated leading value velocity					
-				Double	r
Multi-line: yes	Axis index		numMachAxes		

aaLeadTyp	$AA_LEAD_TYP[x] x = Axis				
Source of the lead value 1: actual value 2: desired value 3: simulated value					
-				UWord	r
Multi-line: yes	Axis index		numMachAxes		

aaLeadV	$AA_LEAD_V[x] x = Axis				
Actual lead value - velocity					
-				Double	r
Multi-line: yes	Axis index		numMachAxes		

aaLoad	$AA_LOAD[x] x = Axis				
Drive load in % (611D only)					
%				Double	r
Multi-line: yes	Axis index		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 index		numMachAxes		

aaMm	$AA_MM[x] x = Axis				
Latched probe position in the machine coordinate system					
-				Double	wr
Multi-line: yes	Axis index		numMachAxes		

aaMm1	$AA_MM1[x] x = Axis				
Access to measurement result of trigger event 1 in the MCS					
-				Double	wr
Multi-line: yes	Axis index		numMachAxes		

aaMm2	$AA_MM2[x] x = Axis				
Access to measurement result of trigger event 2 in the MCS					
-				Double	wr
Multi-line: yes	Axis index		numMachAxes		

aaMm3	$AA_MM3[x] x = Axis				
Access to measurement result of trigger event 3 in the MCS					
-				Double	wr
Multi-line: yes	Axis index		numMachAxes		

aaMm4	$AA_MM4[x] x = Axis				
Access to measurement result of trigger event 4 in the MCS					
-				Double	wr
Multi-line: yes	Axis index		numMachAxes		

aaOff	$AA_OFF[x] x = Axis				
Superimposed position offset from synchronous actions					
-				Double	r
Multi-line: yes	Axis index		numMachAxes		

aaOffLimit	$AA_OFF_LIMIT[x] x = Axis				
Limit for axial correction $AA_OFF reached (Note: for SYNACT only) 0: limit not reached 1: limit in positive axial direction reached 11: limit in negative axial direction reached					
-				UWord	r
Multi-line: yes	Axis index		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 index		numMachAxes		

aaOscillBreakPos1	$AA_OSCILL_BREAK_POS1[<Achse>]				
Oscillation interrupt position 1					
-				Double	r
Multi-line: yes	Axis index		numMachAxes		

aaOscillBreakPos2	$AA_OSCILL_BREAK_POS2[<Achse>]				
Oscillation interrupt position 2					
-				Double	r
Multi-line: yes	Axis index		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)					
-				Double	r
Multi-line: yes	Axis index		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)					
-				Double	r
Multi-line: yes	Axis index		numMachAxes		

aaOvr	$AA_OVR[x] x = Axis				
Axial override for synchronous actions					
-				Double	r
Multi-line: yes	Axis index		numMachAxes		

aaPlcOvr	$AA_PLC_OVR[Achse]				
Axial override specified by PLC for motion-synchronous actions					
-	100	0		Double	r
Multi-line: yes	Axis index		numMachAxes		

aaPolfa	$AA_POLFA				
The programmed retraction position of the single axis					
-				Double	r
Multi-line: yes	Axis index		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 index		numMachAxes		

aaPower	$AA_POWER[x] x = Axis				
Drive power in W (611D only)					
W				Double	r
Multi-line: yes	Axis index		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 index		numMachAxes		

aaSoftendn	$AA_SOFTENDN[x] x = Axis				
Software end position, negative direction					
-				Double	r
Multi-line: yes	Axis index		numMachAxes		

aaSoftendp	$AA_SOFTENDP[x] x = Axis				
Software end position, positive direction					
-				Double	r
Multi-line: yes	Axis index		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 index		numMachAxes		

aaSync	$AA_SYNC[x] x = Axis				
Coupling state of the following axis with lead value coupling 0: not synchronized 1: synchronized coarse 2: synchronized fine 3: synchronized coarse and fine					
-				UWord	r
Multi-line: yes	Axis index		numMachAxes		

aaTorque	$AA_TORQUE[x] x = Axis				
Desired torque value in Nm (611D only)					
Nm				Double	r
Multi-line: yes	Axis index		numMachAxes		

aaTotalOvr	$AA_TOTAL_OVR[Achse]				
The total axial override for motion-synchronous actions					
-	100	0		Double	r
Multi-line: yes	Axis index		numMachAxes		

aaTravelCount	$AA_TRAVEL_COUNT[Achse]				
Total traverse processes of an axis					
-		0		Double	r
Multi-line: yes	Axis index		numMachAxes		

aaTravelCountHS	$AA_TRAVEL_COUNT_HS[Achse]				
Total traverse processes of an axis at high speed					
-		0		Double	r
Multi-line: yes	Axis index		numMachAxes		

aaTravelDist	$AA_TRAVEL_DIST[Achse]				
Total travel path of an axis in mm or degrees					
-		0		Double	r
Multi-line: yes	Axis index		numMachAxes		

aaTravelDistHS	$AA_TRAVEL_DIST_HS[Achse]				
Total travel path of an axis at high speed in mm or degrees					
-		0		Double	r
Multi-line: yes	Axis index		numMachAxes		

aaTravelTime	$AA_TRAVEL_TIME[Achse]				
Total traverse time of an axis in seconds					
-		0		Double	r
Multi-line: yes	Axis index		numMachAxes		

aaTravelTimeHS	$AA_TRAVEL_TIME_HS[Achse]				
Total traverse time of an axis at high speed in seconds					
-		0		Double	r
Multi-line: yes	Axis index		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 index		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 index		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 index		numMachAxes		

aaVc	$AA_VC[x] x = Axis				
Additive correction value for path feed or axial feed					
-				Double	r
Multi-line: yes	Axis index		numMachAxes		

ackSafeMeasPos					
Confirmation of SI actual position 0 = not confirmed 0x00AC = confirmed					
-				UWord	wr
Multi-line: yes	Axis index		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 index		numMachAxes		

actFeedRate					S5
Actual value of axis-specific feedrate for positioning axes. Actual value of single axis feed for additional axes.					
-				Double	r
Multi-line: yes	Axis index		numMachAxes		

actIndexAxPosNo					
Current indexing position number 0 = no indexing position >0 = indexing position number					
-				UWord	r
Multi-line: yes	Axis index		numMachAxes		

actSpeedRel					
Actual value of rotary speed (referring to the maximum speed in %; for 611D in MD1401), for linear drives actual value of the velocity.					
%				Double	r
Multi-line: yes	Axis index		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 index		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 index		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 index		numMachAxes		

axisActiveInChan					
Flag indicating whether axis is active in this channel 0 = not active 1 = active					
-				UWord	r
Multi-line: yes	Axis index		numMachAxes		

axisFeedRateUnit					
Unit of the axis-specific feedrate 0 = mm/min 1 = inch/min 2 = degree/min					
-				UWord	r
Multi-line: yes	Axis index		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 index		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 index		numMachAxes		

cmdContrPos					
Desired value of position after fine interpolation					
mm, inch, degree, user defined				Double	r
Multi-line: yes	Axis index		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 index		numMachAxes		

cmdFeedRate					
Desired value of axis-specific feedrate, if axis is a positioning axis. Single axis feedrate if the axis is an additional axis.					
-				Double	r
Multi-line: yes	Axis index		numMachAxes		

cmdSpeedRel					
Desired value of rotary speed. (referring to the max. speed in %; for 611D in MD 1401). For linear motors actual value of velocity.					
%				Double	r
Multi-line: yes	Axis index		numMachAxes		

contrConfirmActive					
Controller enable 0 = no controller enable 1 = controller enable					
-				UWord	r
Multi-line: yes	Axis index		numMachAxes		

contrMode					
Identifier for controller mode servo 0 = position control 1 = speed control 2 = stop 3 = park 4 = follow-up					
-				UWord	r
Multi-line: yes	Axis index		numMachAxes		

displayAxis	$MC_DISPLAY_AXIS Bit16-31				
Identifier indicating whether axis is displayed by MMC 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 index		numMachAxes		

distPerDriveRevol					
Distance per revolution. The physical unit is defined in measUnit (in this module).					
mm, inch, degree, user defined				Double	r
Multi-line: yes	Axis index		numMachAxes		

drive2ndTorqueLimit					
2nd torque limit. With linear motors: 2nd force limit 0 = not active 1 = active					
-				UWord	r
Multi-line: yes	Axis index		numMachAxes		

driveActMotorSwitch					
Actual motor wiring (star/delta) 0 = star 1 = delta					
-				UWord	r
Multi-line: yes	Axis index		numMachAxes		

driveActParamSet					
Number of the actual drive parameter set					
-		1	8	UWord	r
Multi-line: yes	Axis index		numMachAxes		

driveClass1Alarm					
Message ZK1 drive alarm 0 = no alarm set 1= alarm set (fatal error occured)					
-				UWord	r
Multi-line: yes	Axis index		numMachAxes		

driveContrMode					
Control mode of drive 0 = current control 1 = speed control					
-				UWord	r
Multi-line: yes	Axis index		numMachAxes		

driveCoolerTempWarn					
Heatsink temperature monitoring 0 = temperature OK 1 = overtemperature					
-				UWord	r
Multi-line: yes	Axis index		numMachAxes		

driveDesMotorSwitch					
Motor wiring selection (star/delta) 0 = star 1 = delta					
-				UWord	r
Multi-line: yes	Axis index		numMachAxes		

driveDesParamSet					
Desired parameter set of the drive					
-		1	8	UWord	r
Multi-line: yes	Axis index		numMachAxes		

driveFastStop					
Ramp-function generator rapid stop 0 = not stopped 1 = stopped					
-				UWord	r
Multi-line: yes	Axis index		numMachAxes		

driveFreqMode					
I/F mode					
-				UWord	r
Multi-line: yes	Axis index		numMachAxes		

driveImpulseEnabled					
Enable inverter impulse (checkback signal to impulseEnable) 0 = not enabled 1 = enabled					
-				UWord	r
Multi-line: yes	Axis index		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 index		numMachAxes		

driveIntegDisable					
Integrator disable 0 = not disabled 1 = disabled					
-				UWord	r
Multi-line: yes	Axis index		numMachAxes		

driveLinkVoltageOk					
State of the DC link voltage 0 = OK 1 = not OK					
-				UWord	r
Multi-line: yes	Axis index		numMachAxes		

driveMotorTempWarn					
Motor temperature warning 0 = temperature OK 1 = overtemperature					
-				UWord	r
Multi-line: yes	Axis index		numMachAxes		

driveNumCrcErrors					
CRC errors on the drive bus (transmission errors when writing data to the 611D; values may range up to FFFFH) 0 = no error					
-				UWord	r
Multi-line: yes	Axis index		numMachAxes		

driveParked					
Parking axis 0 = no parking axis 1 = parking axis					
-				UWord	r
Multi-line: yes	Axis index		numMachAxes		

drivePowerOn					
Drive switched on 0 = drive not switched on 1 = drive switched on					
-				UWord	r
Multi-line: yes	Axis index		numMachAxes		

driveProgMessages					
Configurable messages (via machine data)					
-				UWord	r
Multi-line: yes	Axis index		numMachAxes		

driveReady					
Drive ready 0 = drive not ready 1 = drive ready					
-				UWord	r
Multi-line: yes	Axis index		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 index		numMachAxes		

driveSetupMode					
Set-up mode 0 = not active 1 = active					
-				UWord	r
Multi-line: yes	Axis index		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 index		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 index		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 index		numMachAxes		

encChoice					
Active encoder 0 = does not exist 1 = encoder 1 2 = encoder 2					
-				UWord	r
Multi-line: yes	Axis index		numMachAxes		

fctGenState					
State of the function generator					
-				UWord	r
Multi-line: yes	Axis index		numMachAxes		

feedRateOvr					
Feedrate override (only if axis is a positioning axis)					
%				Double	r
Multi-line: yes	Axis index		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 index		numMachAxes		

fxsInfo	$VA_FXS_INFO[Achse]				
Additional information on travel to fixed stop if $VA_FXS[]=2, or OPI variable /C/SEMA/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 index		numMachAxes		

fxsStat	$AA_FXS[x] x = Axis				
State after travelling to fixed stop 0 = normal control 1 = fixed stop reached 2 = failed					
-				UWord	r
Multi-line: yes	Axis index		numMachAxes		

handwheelAss					
Number of handwheel assigned to an axis 0 = no handwheel assigned 1 to 3 = handwheel number					
-		0	3	UWord	r
Multi-line: yes	Axis index		numMachAxes		

impulseEnable					
Impulse enable for drive 0 = not enabled 1 = enabled					
-				UWord	r
Multi-line: yes	Axis index		numMachAxes		

index					
Absolute axis index referring to machine data axis number					
-				UWord	r
Multi-line: yes	Axis index		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 index		numMachAxes		

kVFactor					
position control gain factor					
16.667 1/s				Double	r
Multi-line: yes	Axis index		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 index		numMachAxes		

logDriveNo					
Drive assignment (logical drive number) 0 = not available 1 to 15 = drive number					
-		0	15	UWord	r
Multi-line: yes	Axis index		numMachAxes		

measFctState					
State of the probing function					
-				UWord	r
Multi-line: yes	Axis index		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 index		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 index		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 index		numMachAxes		

measUnit					
Unit for service values of the drives 0 = mm 1 = inch 2 = grd					
-				UWord	r
Multi-line: yes	Axis index		numMachAxes		

paramSetNo					
Number of parameter set					
-		1	8	UWord	r
Multi-line: yes	Axis index		numMachAxes		

preContrFactTorque					
Feed forward control factor torque					
Nm				Double	r
Multi-line: yes	Axis index		numMachAxes		

preContrFactVel					
Feed forward control factor velocity					
-				Double	r
Multi-line: yes	Axis index		numMachAxes		

preContrMode					
Feed forward control mode 0 = inactive 1 = velocity feed forward 2 = torque feed forward					
-				UWord	r
Multi-line: yes	Axis index		numMachAxes		

PRESETActive					
Preset state 0 = no preset active 1 = preset active					
-				UWord	r
Multi-line: yes	Axis index		numMachAxes		

PRESETVal	$AC_PRESET[x] x = Axis				
The function PRESETON (...) programs a zero 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 MMC.					
mm, inch, user defined				Double	wr
Multi-line: yes	Axis index		numMachAxes		

progIndexAxPosNo					
Programmed indexing position number 0 = no indexing position >0 = indexing position number					
-				UWord	r
Multi-line: yes	Axis index		numMachAxes		

qecLrnIsOn					
Quadrant error compensation learning active 0 = inactive 1 = Neuronal-QEC learning active 2 = Standard-QEC active 3 = Standard-QEC with adaption of the correction value active 4 = Neuronal-QEC active 5 = Neuronal-QEC with adaption of the measuring time active 6 = Neuronal-QEC with adaption of the decay time of the correction value active 7 = Neuronal-QEC with adaption of the measuring time and the decay time of the correction value active					
-		0	7	UWord	r
Multi-line: yes	Axis index		numMachAxes		

refPtBusy					
Axis is being referenced 0 = axis is not being referenced 1 = axis is being referenced					
-				UWord	r
Multi-line: yes	Axis index		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 index		numMachAxes		

refPtStatus					
Identifier indicating whether an axis is liable for reference and actually is referenced. Note for changing axes into another channel: In general a changable axis is only liable for reference in the channel it is presently assigned to. Thus a referenced changable axis is announced to the channel it is presently being moved in with the value 3 (liable for reference and referenced) and to all other channels with the value 1 (not liable for reference but referenced). A set bit means: Until SW release 3.1: bit0: at least 1 measuring system has been referenced bit1: active measuring system is liable for reference From SW release 3.2: bit0: active measuring system has been referenced bit1: active measuring system is liable for reference (The busy signal effects the state)					
-	Achsindex			UWord	r
Multi-line: no			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 index		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	wr
Multi-line: yes	Axis index		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	wr
Multi-line: yes	Axis index		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 index		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 /C/SEMA/safeAcceptTestMode to the NCK. 0ACH: NCK has acceptance test mode active					
-	0	0	0FFH	UWord	r
Multi-line: yes	Axis index		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 index		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 index		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		

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 0 = not activated 1 = activated					
-				UWord	r
Multi-line: yes	Axis index		numMachAxes		

safeInputSig					
Safe input signals of the axis					
-				UWord	r
Multi-line: yes	Axis index		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 index		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 index		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 index		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 index		numMachAxes		

safeOutputSig					
Safe output signals of the axis					
-				UWord	r
Multi-line: yes	Axis index		numMachAxes		

safeOutputSig2					
Safe output signals part 2					
-		0	0xffff	UWord	r
Multi-line: no			numMachAxes		

safeOutputSigDrive					
Safe output signals of the drive					
-				UWord	r
Multi-line: yes	Axis index		numMachAxes		

safeOutputSigDrive2					
Safe output signals of the drive part 2					
-		0	0xffff	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 index		numMachAxes		

spec					
Axis specification 0 = path axis 1 = positioning axis					
-				UWord	r
Multi-line: yes	Axis index		numMachAxes		

subSpec					T1
Subspecification 0 = normal axis 1 = indexing axis					
-				UWord	r
Multi-line: yes	Axis index		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 index		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 index		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 index		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 index		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 index		numMachAxes		

trackErrContr					
Position controller difference (actual value / desired value of position)					
mm, inch, degree, user defined				Double	r
Multi-line: yes	Axis index		numMachAxes		

trackErrDiff					
Contour deviation (difference actual value of position and calculated dynamical model)					
mm, inch, degree, user defined				Double	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		

vaDistTorque	$VA_DIST_TORQUE[Achse]				
Disturbing torque/max. torque (motor end, York)					
%	0	-100	100	Double	r
Multi-line: yes	Axis index		numMachAxes		

vaDpe	$VA_DPE[x1]				
Status of power enable of a machine axis 0 - 1					
-	0	0	1	UWord	r
Multi-line: yes	Axis index		numMachAxes		

vaIm	$VA_IM[x]				
Encoder actual value in the machine coordinate system (measured active measuring system)					
-	0	0		Double	r
Multi-line: yes	Axis index		numMachAxes		

vaIm1	$VA_IM1[x]				
Actual value in the machine coordinate system (measured encoder 1)					
-	0	0		Double	r
Multi-line: yes	Axis index		numMachAxes		

vaIm2	$VA_IM2[x]				
Actual value in the machine coordinate system (measured encoder 2)					
-	0	0		Double	r
Multi-line: yes	Axis index		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 index		numMachAxes		

vaVactm	$VA_VACTM[x] x = Axis				
Axis velocity actual value on the load side in the MCS					
-				Double	r
Multi-line: yes	Axis index		numMachAxes		

1.5.3 Area C, Mod. SGA: State data: Geometry axes in tool offset memory

OEM-MMC: Linkitem /Channel/GeometricAxis/...

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	$AA_IW[x] x = Axis				
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					
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 state 0 = travel command in plus direction 1 = travel command in minus direction 2 = exact stop coarse reached 3 = exact stop 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	wr
Multi-line: yes	Axis index		numMachAxes		

1.5.4 Area C, Mod. SEGA: State data: Geometry axes in tool offset memory (extension of SGA)

OEM-MMC: Linkitem /Channel/GeometricAxis/...

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.

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)					
-				Double	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)					
-				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)					
-				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)					
-				Double	r
Multi-line: yes	Axis index		numMachAxes		

aaIbCorr	$AA_IB_CORR				
Current BCS setpoint value of an axis including override components					
-				Double	r
Multi-line: yes	Axis index		numMachAxes		

aaIwCorr	$AA_IW_CORR				
Current WCS setpoint value of an axis including override components					
-				Double	r
Multi-line: yes	Axis index		numMachAxes		

aaMw	$AA_MW[x] x = Axis				
Latched probe position retransformed in the WCS					
-				Double	wr
Multi-line: yes	Axis index		numMachAxes		

aaMw1					$AA_ MW1 [Achs e]
Access to measurement result of trigger event 1 in the WCS					
-				Double	wr
Multi-line: yes	Axis index		numMachAxes		

aaMw2					$AA_ MW2 [Achs e]
Access to measurement result of trigger event 2 in the WCS					
-				Double	wr
Multi-line: yes	Axis index		numMachAxes		

aaMw3					$AA_ MW3 [Achs e]
Access to measurement result of trigger event 3 in the WCS					
-				Double	wr
Multi-line: yes	Axis index		numMachAxes		

aaMw4					$AA_ MW4 [Achs e]
Access to measurement result of trigger event 4 in the WCS					
-				Double	wr
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	r
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					
-				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		

actProgPosBKS	$AA_IBORI				
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		

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					
-				Double	r
Multi-line: yes	Axis index		numMachAxes		

actToolBasePosDiam					
Corresponds to /C/SGA/actToolBasePos with diameter conversion					
-				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:/C/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					
-				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:/C/SGA/actToolBasePos without programmed frame)					
-				Double	r
Multi-line: yes	Axis index		numMachAxes		

actToolBasPosENitc					
corresponds to actToolBasPosEN with $DISPLAY_MODE_POSITION=1					
-				Double	r
Multi-line: yes	Axis index		numMachAxes		

actToolBasPosENjmp					
corresponds to actToolBasPosEN with $DISPLAY_MODE_POSITION=0					
-				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					
-				Double	r
Multi-line: yes	Axis index		numMachAxes		

axisActiveInChan					
Flag indicating whether axis is active in this channel 0 = not active 1 = active					
-				UWord	r
Multi-line: yes	Axis index		numMachAxes		

axisFeedRateUnit					
Unit of axial feedrate 0 = mm/min 1 = mm/rev 2 = inch/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		

cmdToolEdgeCenterPosEnsS					
Programmed 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 Note: 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: - /C/SGA/cmdToolBasePos - /C/SGA/toolBaseDistToGo - /C/SGA/toolBaseREPOS - cmdToolEdgeCenterPos - actToolEdgeCenterPos - toolEdgeCenterDistToGo - toolEdgeCenterREPOS - cmdProgPos - actProgPos - progDistToGo - progREPOS - actToolBasPosEN - cmdToolEdgeCenterPosEnsS - /C/SEGA/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 MMC 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 /C/SGA/extUnit (in module SGA).					
mm, inch, degree, user defined				Double	r
Multi-line: yes	Axis index		numMachAxes		

dummy					
Added for alignment only, row index may be used later					
-				UWord	
Multi-line: no					

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 1 - 3 for geometry axes 0 for non-geometry axes					
-				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 referring to machine data axis number					
-				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		

1.5.5 Area C, Mod. SSP: State data: Spindle

OEM-MMC: Linkitem /Channel/Spindle/...

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		

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		

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 speed of the master spindle. The requested value for the master spindle differs from SSP:cmdSpeed only if G96 is active					
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 = not active 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	$P_SMODE				
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		

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 state Bit0 = following spindle Bit1 = leading spindle					
-				UWord	r
Multi-line: yes	Spindle index		numSpindles		

turnState	$AC_SDIR[x] x = SpindleNo				
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		

1.5.6 Area C, Mod. FU: Channel-specific settable frames

OEM-MMC: Linkitem /Channel/UserFrame/...

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 zero offset (the physical unit is defined in basicLengthUnit in module Y in area N).					
mm, inch, user defined				Double	wr
Multi-line: yes	Frame index * numMachAxes + 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	wr
Multi-line: yes	Frame index * numMachAxes + axis number		$MC_MM_NUM_USER_FRAMES * (numGeoAxes + numAuxAxes)		

mirrorImgActive	$P_UIFR[x,y,MI] x = FrameNo,y=Axis				PA
Mirroring enabled in settable zero offset 0 = mirroring not active 1 = mirroring active					
-				UWord	wr
Multi-line: yes	Frame index * numMachAxes + axis number		$MC_MM_NUM_USER_FRAMES * (numGeoAxes + numAuxAxes)		

rotation	$P_UIFR[x,y,RT] x = FrameNo,y=Axis				PA
Rotation of settable zero offset					
Degree				Double	wr
Multi-line: yes	Frame index * numMachAxes + axis number		$MC_MM_NUM_USER_FRAMES * (numGeoAxes + numAuxAxes)		

scaleFact	$P_UIFR[x,y,SC] x = FrameNo,y=Axis				PA
Scaling factor of settable zero offset					
-				Double	wr
Multi-line: yes	Frame index * numMachAxes + axis number		$MC_MM_NUM_USER_FRAMES * (numGeoAxes + numAuxAxes)		

1.5.7 Area C, Mod. SSP2: State data: Spindle

OEM-MMC: Linkitem /Channel/LogicalSpindle/...

All state data that refer to a spindle, if a spindle converter (logical spindles) is active

acConstCutS	$AC_CONSTCUT_S[n]				
Current constant cutting rate					
m/min, ft/min, user defined	0			Double	r
Multi-line: yes	Logical spindle index		numSpindlesLog		

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	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 speed of the master spindle. The requested value for the master spindle differs from SSP:cmdSpeed only if G96 is active					
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	{$GWPS}				
SUG programming active (SUG is the function "constant perimeter speed of grinding wheel) 0 = not active 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	$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	Logical spindle index		numSpindlesLog		

pSModeS	$P_SMODE				
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		

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 state 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		

1.5.8 Area C, Mod. FA: Active channel-specific frames

OEM-MMC: Linkitem /Channel/ActualFrame/...

The following frame indices are available:

0: ACTFRAME = currently resulting work offset
1: IFRAME = current settable work offset
2: PFRAME = current programmable work offset
3: EXTFRAME = current external work offset
4: TOTFRAME = current total work offset = sum of ACTFRAME and EXTFRAME
5: ACTBFRAME = current total base frame
6: SETFRAME = current 1st system frame (set actual value, scratching)
7: EXTSFRAME = current 2nd system frame (set actual value, scratching)
8: PARTFRAME = current 3rd system frame (TCARR and PAROT with orientable toolholder)
9: TOOLFRAME = current 4th system frame (TOROT and TOFRAME)
10: MEASFRAME = result frame for workpiece and tool measurement
11: WPFRAME = current 5th system frame (workpiece reference points) as of SW 6.3
12: CYCFRAME = current 6th system frame (cycles) as of SW 6.3

The maximum frame index is 12.

linShift	$P_PFRAME[x,TR] / $P_ACTFRAME / $P_IFRAME				PA
Translation of an active zero 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		11 * numMachAxes		

mirrorImgActive	$P_PFRAME[x,MI] / $P_ACTFRAME / $P_IFRAME				PA
Mirroring enabled in an active zero offset 0 = mirroring not active 1 = mirroring active					
-				UWord	r
Multi-line: yes	Frame index * numMachAxes + axis number		11 * numMachAxes		

rotation	$P_PFRAME[x,RT] / $P_ACTFRAME / $P_IFRAME				PA
Rotation of an active zero offset					
Degree				Double	r
Multi-line: yes	Frame index * numMachAxes + axis number		11 * numMachAxes		

scaleFact	$P_PFRAME[x,SC] / $P_ACTFRAME / $P_IFRAME				PA
Scaling factor of an active zero offset					
-				Double	r
Multi-line: yes	Frameindex * numMachAxes + axis number		11 * numMachAxes		

1.5.9 Area C, Mod. FE: Channel-specific external frame

OEM-MMC: Linkitem /Channel/ExternFrame/...

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 zero offset (the physical unit is defined in basicLengthUnit in module Y in area N).					
mm, inch, user defined				Double	wr
Multi-line: yes	Geo axis number		numGeoAxes		

1.6 State data of drives

1.6.1 Area H, Mod. S: Drive-specific state data (MSD)

OEM-MMC: Linkitem /DriveHsa/State/...

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

actualCurrent	MD 1708: ACTUAL_CURRENT				IAD
Actual value of the smoothened current (referring to the maximum current of the drive) CAUTION: The variable cannot be configured by MMC100.					
%		-100000,0	10000,0	Float	r
Multi-line: no					

actualSpeed	MD 1701: ACTUAL_SPEED				IAD
Actual value of rotary speed (motor) CAUTION: The variable cannot be configured by MMC100.					
rev/min , m/min		-100000,0	100000,0	Float	r
Multi-line: no					

cl1PoImage	MD 1731: CL1_PO_IMAGE				IAD
Image of the ZK1PO register. The format is hexadecimal Attention: The variable cannot be configured by the MMC 100					
-		0x0	0x7FFF	UWord	r
Multi-line: no					

cl1ResImage	MD 1732: CL1_RES_IMAGE				IAD
Image of the ZK1RES register. The format is hexadecimal Attention: The variable cannot be configured by the MMC 100!					
-		0x0	0x7FFF	UWord	r
Multi-line: no					

crcErrorCount	MD 1720: CRC_DIAGNOSIS				IAD
CRC-parameter for diagnosis. Number is displayed in hexadecimal format. Attention: This variable cannot be configured by the MMC100!					
-		0x0	0x7FFF	UWord	r
Multi-line: no					

desiredSpeed	MD 1706: DESIRED_SPEED				IAD
Desired value of rotary speed. With linear motors: desired velocity Attention: The variable cannot be configured by the MMC100!					
rev/min , m/min		-100000,0	100000,0	Float	r
Multi-line: no					

encTypeDirect	MD 1791: ENC_TYPE_DIRECT				IAD
Direct mounted encoder used for reading in the actual values Attention: This variable cannot be configured by MMC100!					
-		0	32767	UWord	r
Multi-line: no					

encTypeMotor	MD 1790: ENC_TYPE_MPTOR				IAD
Motor mounted (indirect) encoder used for reading in the actual values. Attention: This variable cannot be configured by MMC100!					
-		0	32767	UWord	r
Multi-line: no					

firmwareDate	MD 1798: FIRMWARE_DATE				IAD
Firmware date Attention: This variable cannot be configured by the MMC100!					
-		0	32767	UWord	r
Multi-line: no					

firmwareVersion	MD 1799: FIRMWARE_VERSION				IAD
Firmware version Attention: This variable cannot be configured by the MMC100!					
-		0	32767	UWord	r
Multi-line: no					

linkVoltage	MD 1701: LINK_VOLTAGE				IAD
Link voltage Attention: This variable cannot be configured by the MMC100! 0 to 800					
V		0	32767	UWord	r
Multi-line: no					

load	MD 1722: LOAD				IAD
Load. Display is in hexadecimal format Attention: Variable cannot be configured by the MMC100!					
%		-100000,0	100000,0	Float	r
Multi-line: no					

motorTemperature	MD 1702: MOTOR_TEMPERATURE				IAD
Motor temperature Attention: The variable cannot be configured by the MMC100!					
C		0	32767	UWord	r
Multi-line: no					

operatingMode					IAD
Operating mode Attention: Variable cannot be configured By the MMC100! Bit0 = VSA Bit4 = HSA Bit8 = AM control Bit9 = AM closed loop control Bit12 = U/f-operation mode bits exclude one another (except bit 12)					
-				UWord	r
Multi-line: no					

pblVersion	MD 1797: PBL_VERSION				IAD
Data version Attention: The variable cannot be configured by the MMC100!					
-		0	32767	UWord	r
Multi-line: no					

safeStopFDiagnosis	MD 1395 : SAFE_STOP_F_DIAGNOSIS				
Drive error code for alarm 300911					
-	0			UWord	r
Multi-line: no					

terminalState	MD 1700: TERMINAL_STATE				IAD
State of the binary inputs (displayed in hexadecimal format) Attention: This variable cannot be configured by the MMC100!					
-		0x0	0x7FFF	UWord	r
Multi-line: no					

1.6.2 Area V, Mod. S: Drive-specific status data (FDD)

OEM-MMC: Linkitem /DriveVsa/State/...

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.

actualCurrent	MD 1708: ACTUAL_CURRENT				IAD
Actual value of the smoothened current (referring to the maximum current of the drive)					
%		-10000,0	10000,0	Float	r
Multi-line: no					

actualSpeed	MD 1707: ACTUAL_SPEED				IAD
Actual value of the rotary speed, actual velocity for linear drive (motor)					
rev/min , m/min		-100000,0	100000,0	Float	r
Multi-line: no					

cl1PoImage	MD 1731: CL1_PO_IMAGE				IAD
ZK1PO register image. Hexadecimal numerical representation					
-		0x0	0x7FFF	UWord	r
Multi-line: no					

cl1ResImage	MD 1732: CL1_RES_IMAGE				IAD
ZK1RES register image. Hexadecimal numerical representation					
-		0x0	0x7FFF	UWord	r
Multi-line: no					

crcErrorCount	MD 1720: CRC_DIAGNOSIS				IAD
CRC diagnostic parameter. Hexadecimal numerical representation					
-		0x0	0x7FFF	UWord	r
Multi-line: no					

desiredSpeed	MD 1706: DESIRED_SPEED				IAD
Speed setpoint					
rev/min , m/min		-100000,0	100000,0	Float	r
Multi-line: no					

encTypeDirect	MD 1791: ENC_TYPE_DIRECT				IAD
Measuring circuit type of direct measuring system					
-		0	32767	UWord	r
Multi-line: no					

encTypeMotor	MD 1790: ENC_TYPE_MOTOR				IAD
Measuring circuit type of indirect measuring system					
-		0	32767	UWord	r
Multi-line: no					

firmwareDate	MD 1798: FIRMWARE_DATE				IAD
Firmware date					
-		0	32767	UWord	r
Multi-line: no					

firmwareVersion	MD 1799: FIRMWARE_VERSION				IAD
Firmware version					
-		0	32767	UWord	r
Multi-line: no					

linkVoltage	MD 1701: LINK_VOLTAGE				IAD
DC-link voltage					
V		0	32767	UWord	r
Multi-line: no					

load	MD 1722: LOAD				IAD
Utilization: Represented in hexadecimal format					
%		-100000,0	100000,0	Float	r
Multi-line: no					

motorTemperature	MD 1702: MOTOR_TEMPERATURE				IAD
Motor temperature					
C		0	32767	UWord	r
Multi-line: no					

operatingMode					IAD
Operating mode Bit0 = FDD Bit4 = MSD Bit8 = Open-loop AM control Bit9 = Closed-loop AM control Bit12 = V/Hz mode					
-				UWord	r
Multi-line: no					

pblVersion	MD 1797: PBL_VERSION				IAD
Data version					
-		0	32767	UWord	r
Multi-line: no					

safeStopFDiagnosis	MD 1395 : SAFE_STOP_F_DIAGNOSIS				
Drive error code for alarm 300911					
-	0			UWord	r
Multi-line: no					

terminalState	MD 1700: TERMINAL_STATE				IAD
State of the binary inputs (in hexadecimal format)					
-		0x0	0x7FFF	UWord	r
Multi-line: yes	1				

1.7 Tool and magazine data

1.7.1 Area T, Mod. TO: Tool edge data: Offset data

OEM-MMC: Linkitem /Tool/Compensation/...

The data module TO is organized as a two-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.

<table>
<tr><td>cuttEdgeParam</td><td colspan="4">$TC_DPCEx[y,z] x = ParamNo y = ToolNo z = EdgeNo</td><td></td></tr>
<tr><td colspan="6">Offset value parameter and edge list with D numbers for a tool
1st section: Offset value parameter for a tool edge:
The meaning of each parameter depends on the tool type. At present 25 parameters are reserved for each tool edge (however only some of them are used). To be flexible for future extensions use the variable 'numCuttEdgeParams' rather than the fixed number 25 for the number of parameters.
A detailed description of tool parameters can be found in the documentation 'Tool Offset (W1)', Section "Tool edge".
The following list is a summary of the tool edge parameters:
Parameter 1: Geometry -- tool type ($TC_DP1)
Parameter 2: Geometry -- tool point direction ($TC_DP2)
Parameter 3: Geometry -- length 1 ($TC_DP3)
Parameter 4: Geometry -- length 2 ($TC_DP4)
Parameter 5: Geometry -- length 3 ($TC_DP5)
Parameter 6: Geometry -- radius ($TC_DP6)
Parameter 7: Geometry -- corner radius (tool type 700; slotting saw) ($TC_DP7)
Parameter 8: Geometry -- length 4 (tool type 700; slotting 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 -- slot 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: Relief angle ($TC_DP24)
Parameter 25: Manual Turn: Cutting rate ($TC_DP25)
Shopmill: Bit-coded value for different states of tools of type 1xx and 2xx ($TC_DP25)
All parameters up to 25, that are not listed, are reserved.

2nd section: edgeDNo (SW 5.1 and later), associated optional D numbers of edges:
-1: No edge
1 .. maxDNo: Edge exists, associated D number, only when "any D numbers" function is activated (maxnumCuttEdges_Tool < maxCuttingEdgeNo)
Edge No.:1 to maxnumCuttEdges_Tool, when edge exists, but when "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 NCK variable $TC_DPCE....
$TC_DPCE = edge number, D = offset number D.
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 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,....25.).

Important: This variable is called "edgeData" in the MMC102.
The value for the tool type is stored internally as an integer.</td></tr>
<tr><td>-</td><td>0</td><td></td><td></td><td>Double</td><td>wr</td></tr>
<tr><td>Multi-line: yes</td><td colspan="2">For edge offset value parameters:
(edgeNo - 1) * numCuttEdgeParams + ParameterNo
For D numbers:
((numCuttEdgeParams * maxnumCuttEdges_Tool) + EdgeNo)</td><td colspan="3">(numCuttEdgeParams + 1) * maxnumCuttEdges_Tool</td></tr>
</table>

<table>
<tr><td>edgeData</td><td colspan="4">$TC_DPx[y,z] x = ParamNo y = ToolNo z = EdgeNo</td><td></td></tr>
<tr><td colspan="6">Offset value parameter and edge list with D numbers for a tool
1st section: Offset value parameter for a tool edge:
The meaning of each parameter depends on the tool type. At present 25 parameters are reserved for each tool edge (however only some of them are used). To be flexible for future extensions use the variable 'numCuttEdgeParams' rather than the fixed number 25 for the number of parameters.
A detailed description of tool parameters can be found in the documentation Tool Offset (W1), Section "Tool edge".
The following list is a summary of the tool edge parameters:
Parameter 1: Geometry -- tool type ($TC_DP1)
Parameter 2: Geometry -- tool point direction ($TC_DP2)
Parameter 3: Geometry -- length 1 ($TC_DP3)
Parameter 4: Geometry -- length 2 ($TC_DP4)
Parameter 5: Geometry -- length 3 ($TC_DP5)
Parameter 6: Geometry -- radius ($TC_DP6)
Parameter 7: Geometry -- corner radius (tool type 700; slotting saw) ($TC_DP7)
Parameter 8: Geometry -- length 4 (tool type 700; slotting 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 -- slot 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: Relief angle ($TC_DP24)
Parameter 25: Manual Turn: Cutting rate ($TC_DP25)
Shopmill: Bit-coded value for different states of tools of type 1xx and 2xx ($TC_DP25)
All parameters up to 25, that are not listed, are reserved.

2nd section: edgeDNo (SW 5.1 and later), associated optional D numbers of edges:
-1: No edge
1 .. maxDNo: Edge exists, associated D number, only when "any D numbers" function is activated (maxnumCuttEdges_Tool < maxCuttingEdgeNo)
Edge No.:1 to maxnumCuttEdges_Tool, when edge exists, but when "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 NCK variable $TC_DPCE.... $TC_DPCE... contains a unique number > 32000 when a D number is not assigned.)
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 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,....25.).

Important: This variable is called "cuttEdgeParam" in the non-Windows-MMC and the PLC.

The value for the tool type is stored internally as an integer.</td></tr>
<tr><td>mm, inch, user defined</td><td>0</td><td></td><td></td><td>Double</td><td>wr</td></tr>
<tr><td>Multi-line: yes</td><td colspan="2">For edge offset value parameters:
(edgeNo - 1) * numCuttEdgeParams + ParameterNo
For D numbers:
((numCuttEdgeParams * maxnumCuttEdges_Tool) + EdgeNo)</td><td colspan="3">(numCuttEdgeParams + 1) * maxnumCuttEdges_Tool</td></tr>
</table>

1.7.2 Area T, Mod. TD: Tool data: General data

OEM-MMC: Linkitem /Tool/Data/...

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.

A new entry is provided for the tool monitoring mode:
Monitoring modes in toolMon:
0: No tool monitoring
1: Tool life monitoring
2: Workpiece number monitoring
4: Monitoring of edge wear parameters using wear limit (SW 5.1 and later)
8: Monitoring of total offset parameters (fine, $TC_SCP..., not setup offsets $TC_ECP...) using wear limit (SW 5.1 and later)

adaptNo					
Number of adapter defined by system parameter $TC_ADPx which is supporting the tool >0: adapter number 0: no adapter assigned					
-	0	0	numMagPlacesMax	UWord	r
Multi-line: yes	T number		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		

toolInfo	$TC_TP11				FBW
Tool information for MMC Not currently assigned					
-	0			UWord	wr
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		

toolInPlace	$A_TOOLMLN[x] x = ToolNo T				
Current location in which the tool is located					
-				UWord	r
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 (SW 5.1 and later) 8: monitoring of total offset parameters using wear limit (SW 5.1 and later)					
-	0			UWord	wr
Multi-line: yes	TNo		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	T number		max. T-Nummer		

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	T number		max. T-Nummer		

toolplace_spec	$TC_TP7				FBW
Magazine location type of tool					
-	9999			UWord	wr
Multi-line: yes	Tool number T		32000		

toolSearch	$TC_TP10				FBW
Type of tool search for replacement tools 0: no strategy 1: next duplo no. 2: shortest path					
-	0			UWord	wr
Multi-line: yes	Tool number T		32000		

toolsize_down	$TC_TP6				FBW
Size downwards in half locations					
-	1			UWord	wr
Multi-line: yes	Tool number T		32000		

toolsize_left	$TC_TP3				FBW
Size to the left in half locations					
-	1			UWord	wr
Multi-line: yes	Tool number T		32000		

toolsize_right	$TC_TP4				FBW
Size to the right in half locations					
-	1			UWord	wr
Multi-line: yes	Tool number t		32000		

toolsize_upper	$TC_TP5				FBW
Size upwards in half locations					
-	1			UWord	wr
Multi-line: yes	Tool number T		32000		

toolState	$TC_TP8				FBW
Tool state 0: not enabled 1: active tool (A) 2: enabled (F) 4: disabled (G) 8: measured (M) 16: prewarning limit reached (V) 32: tool being changed (W) 64: fixed location coded (P) 128: tool was in use (E)					
-	0			UWord	wr
Multi-line: yes	Tool number T		32000		

1.7.3 Area T, Mod. TS: Tool edge data: Monitoring data

OEM-MMC: Linkitem /Tool/Supervision/...

The module TS is organized as a two-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	$TC_MOP1[x,y] ...$TC_MOP15[x,y] x=ToolNo y=EdgeNo				
Monitoring data per tool edge Important: This is a two-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 (SW 5.1 and later) ($TC_MOP5) This parameter can only be set if bit 5 of machine data $MN_MM_TOOL_MANAGEMENT_MASK has been correspondingly set. P8 = Remaining wear (actual value) (SW 5.1 and later) ($TC_MOP6) cannot be written P9 = Desired wear (SW 5.1 and later) ($TC_MOP15) This parameter can only be set if bit 5 of machine data $MN_MM_TOOL_MANAGEMENT_MASK has been correspondingly set. Important: This variable is called "dummy" in the non-Windows-MMC and the PLC! The values for P3 to P9 are stored internally as integers.					
-	0			Double	wr
Multi-line: yes	(ToolEdgeNo - 1) * numCuttEdgeParams_ts + ParameterNo		numCuttEdgeParams_ts * maxnumCuttEdges_Tool		

1.7.4 Area T, Mod. TU: Tool data: User-defined data

OEM-MMC: Linkitem /Tool/User/...

The data module TU 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 row index:
The column index is the number of the user-defined tool parameter. The number of tool parameters (columns) is to be found in the variable "numToolParams_tu" in the module Y in area N.
The row 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 parameters. Important: This is a two-dimensional variable. The column index is the parameter number. Attention: This variable is called "dummy" in the non-Windows-MMC and the PLC !					
-				Double	wr
Multi-line: yes	Tool number T		32000		

1.7.5 Area T, Mod. TUE: Tool edge data: User-defined data

OEM-MMC: Linkitem /Tool/User/...

The data module TUE is organized as a two-dimensional variable array. The module contains user-defined tool edge 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 user-defined data for all tool edges are to be found in one column.The assignment of a tool to a T-number is to 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 rows is derived from the number of parameters per tool edge and the number of tool edges of a tool:

maxZeilenanzahl = numCuttEdgeParams_tu * /T/TV/numCuttEdges (T-number)

The number of parameters per tool edge "numCuttEdgeParams_tu" is given in the module Y in area N. The number of tool-specific tool edges "/T/TV/numCuttEdges" are contained in the module TV in the associated area T.
If neccessary, several lines can be addressed, so that in one request, for example, all user-defined tool edge data of a single tool can be read. The data are all of the same data 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 Caution: This variable is called "dummy" in the non-Windows-MMC and the PLC!					
-				Double	wr
Multi-line: yes	(TooledgeNo - 1) * numCuttEdgeParams_tu + ParameterNo		numCuttEdgeParams_tu * maxnumCuttEdges_Tool		

1.7.6 Area T, Mod. TG: Tool data: Grinding-specific data

OEM-MMC: Linkitem /Tool/GrindingData/...

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	wr
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	wr
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	wr
Multi-line: yes	Tool number T		32000		

maxRotSpeed	$TC_TPG6				W4
Maximum rotary speed of the grinding wheel					
rev/min , m/min				Double	wr
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	wr
Multi-line: yes	Tool number T		32000		

minToolDia	$TC_TPG3				W4
Minimum diameter of the grinding wheel					
mm, inch, user defined				Double	wr
Multi-line: yes	Tool number T		32000		

minToolWide	$TC_TPG4				W4
Minimum width of the grinding wheel					
mm, inch, user defined				Double	wr
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	wr
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	wr
Multi-line: yes	Tool-number T		32000		

1.7.7 Area T, Mod. TMC: Magazine data: Configuration data

OEM-MMC: Linkitem /Tool/MagazineConfiguration/...

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 variable is the magazine number. If an error occurs, an error number is set.					
-				UWord	r
Multi-line: no					

magCMCmdPar2					W4
Return value for 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 is described in Section ???. 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				

1.7.8 Area T, Mod. TMV: Magazine data: Directory

OEM-MMC: Linkitem /Tool/MagazineCatalogue/...

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	Magazine number		numMagsMax		

magVNo					
Number of the magazine					
-				UWord	r
Multi-line: yes	Magazine number		numMagsMax		

numActMags					
Number of magazines in the modules TMV and TM					
-			numMagsMax	UWord	r
Multi-line: no					

1.7.9 Area T, Mod. TM: Magazine data: General data

OEM-MMC: Linkitem /Tool/MagazineDescription/...

This module contains the information for the available tool magazines.

magActPlace	$TC_MAP8				
Current magazine position Location number of tool change position					
-				UWord	wr
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		

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		

magLocSearchStrat	$TC_MPAP10, Bits 8-15				
Empty location search strategy during tool change					
-				UWord	r
Multi-line: yes	Magazine number		320000		

magNo					
Number of the magazine					
-		1	numMagsMax	UWord	r
Multi-line: yes	Magazine number		numMagsMax		

magNrPlaces	$TC_MAP7 * $TC_MAP6				
Number of real locations (in chain magazine) or number of slots (in box magazine)					
-				UWord	r
Multi-line: yes	Magazine number		numMagsMax		

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	wr
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		

<table>
<tr><th>magWearCompoundNo</th><th colspan="4">$TC_MAP9</th><th></th></tr>
<tr><td colspan="6">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</td></tr>
<tr><td>-</td><td>0</td><td></td><td></td><td>Long Integer</td><td>wr</td></tr>
<tr><td>Multi-line: yes</td><td colspan="2">Magazine number</td><td colspan="3">numMagsMax</td></tr>
</table>

1.7.10 Area T, Mod. TP: Magazine data: Location data

OEM-MMC: Linkitem /Tool/Magazine/...

The data module TP is organized as a two-dimensional variable array. The module contains the state and assignment of all magazine locations of a T area. Each element can be addressed via a column or a line index:
The column index is the magazine number, i.e. the configuration data for all locations of a magazine are contained in a column. The assignment of a magazine to a magazine number is given in the associated module Magazine directory (TMV) in the associated area T. 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:

maxZeilenindex = numMagPlaceParams * magNrPlaces

The number of parameters per magazine location "numMagPlaceParams" is given in module Y in area N and is currently 7 (8 with SW 5.1 and later).
The row 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 ($TC_MPP5) (read only) and new: Wear group number from SW 5.1
Type index/wear group number is read only in SW earlier than 5.1 and read/write from SW 5.1 if it is assigned "Wear group" meaning.
Type index: The locations of a location type in a magazine are numbered in ascending sequence, e.g. type=2, type index=5; ==> Spindle5)
(previous meaning when location type = 1 before P5: Equals location number when location type=1)
Wear group number from SW 5.1 ($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 / $TC_MAMP3 (general settings for wear grouping).
8: Adapter number from SW 5.1 ($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 from SW 5.1

The number of magazine locations "magNrPlaces" is magazine specific and can be found in module TM in associated area T.

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.

<table>
<tr><th>placeData</th><th colspan="4">$TC_MPP1[n,m]...$TC_MPP7[n,m] n=MagNo m=SlotNo</th><th></th></tr>
<tr><td colspan="6">P1: Location type (read access only) ($TC_MPP1)
P2: Location type (read access only) ($TC_MPP2)
P3: T number of tool in this location ($TC_MPP6)
P4: Consider adjacent location on/off ($TC_MPP3)
P5: Location status (bit arry) ($TC_MPP4)
P6: Reference for physical magazine (read access only)
P7: Location type index (location type numbering) ($TC_MPP5)
P8: Number of adapter in magazine location ($TC_MPP7)

Attention: This variable is called "dummy" in the non-Windows-MMC and the PLC!</td></tr>
<tr><td>-</td><td></td><td></td><td></td><td>UWord</td><td>wr</td></tr>
<tr><td>Multi-line: yes</td><td colspan="2">(LocationNo - 1) * numMagPlaceParams + ParameterNo</td><td colspan="3">numMagPlaceParams * magNrPlaces</td></tr>
</table>

1.7.11 Area T, Mod. TPM: Magazine data: Multiple assignment of location data

OEM-MMC: Linkitem /Tool/Magazine/...

The data module TPM is organized as a two-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	$TC_MDP1[n,m]...$TC_MDP2[n,m] n=MagNo m=SlotNo				
P1: Distance between change position of magazine n and location m of 1st internal magazine (load magazine, 9999) ($TC_MDP1) P2: Distance between change position of magazine n and location m of 2nd internal magazine (buffer magazine, 9998) ($TC_MDP2) Attention: This variable is called "dummy" in the non-Windows-MMC and PLC !					
-				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		

1.7.12 Area T, Mod. TT: Magazine data: Location types

OEM-MMC: Linkitem /Tool/Magazine/...

The module TT is organized as a two-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	$TC_MPTH[n,m] n=0...7 Hierarchy m=0...7 SlotType				
Magazine location hierarchy Attention: This variable is called "dummy" in the non-Windows-MMC and the PLC!					
-				UWord	r
Multi-line: yes	Number of location type + 1		Wert aus Zeile 1		

1.7.13 Area T, Mod. TV: Tool data: Directory

OEM-MMC: Linkitem /Tool/Catalogue/...

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.

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		

numTools					
Number of tools in the area TO					
-		0	MD MM_NUM_TOOL	UWord	r
Multi-line: no					

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					

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		

1.7.14 Area T, Mod. TF: Parametrizing, return parameters of _N_TMGETT, _N_TSEARC

OEM-MMC: Linkitem /Tool/Find/...

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	wr
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	wr
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	wr
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	wr
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	wr
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		

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]	wr
Multi-line: no					

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	wr
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	wr
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	wr
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	wr
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		

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	wr
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	wr
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	wr
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	wr
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	wr
Multi-line: yes	Line index in the TO module is thus 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	wr
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		

parTasksTU					
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	wr
Multi-line: yes	Index of the parameter (i.e. column index) in the TU module, 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	wr
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	wr
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		

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			tfNrOfResults		

1.7.15 Area T, Mod. TUM: Tool data: user magazine data

OEM-MMC: Linkitem /Tool/MagazineDescription/...

userData	$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.					
-	0			Long Integer	wr
Multi-line: yes	Number of the user-defined parameter		numMagParams_u		

1.7.16 Area T, Mod. TUP: Tool data: user magatine place data

OEM-MMC: Linkitem /Tool/Magazine/...

userPlaceData	$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.					
-	0			Long Integer	wr
Multi-line: yes	Number of the user-defined parameter + numMagLocParams_u * (number of the magazine location - 1)		numMagLocParams_u * magNrPlaces		

1.7.17 Area T, Mod. TUS: Tool data: user monitoring data

OEM-MMC: Linkitem /Tool/Supervision/...

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	wr
Multi-line: yes	Number of the user-defined parameter + (number of the cutting edge -1) * numCuttEdgeParams_tus		numCuttEdgeParams_tus * maxnumCuttEdges_Tool		

1.7.18 Area T, Mod. AD: Adapter data

OEM-MMC: Linkitem /Tool/Adapter/...

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	$TC_ADPTx; x=1,...3, $TC_ADPTT				
Adapter data					
mm, inch, user defined	0.0			Double	wr
Multi-line: yes	ParameterNo		numParams_Adapt		

1.7.19 Area T, Mod. AEV: Working offsets: Directory

OEM-MMC: Linkitem /Tool/ActiveCatalogue/...

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)).

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		

DNo					
D number Meaningful and defined only in connection with "unique D numbers" function.					
-				UWord	r
Multi-line: yes	Current number of active tool 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		

1.7.20 Area T, Mod. TC: Toolholder parameters

OEM-MMC: Linkitem /Tool/ToolCarrier/...

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	wr
Multi-line: yes	No. of toolholder		$MN_MM_NUM_TOOL_CARRIER / numToBaust		

tcCarr10	$TC_CARR10				
x component of rotary axis v2					
-	0			Double	wr
Multi-line: yes	No. of toolholder		$MN_MM_NUM_TOOL_CARRIER / numToBaust		

tcCarr11	$TC_CARR11				
y component of rotary axis v2					
-	0			Double	wr
Multi-line: yes	No. of toolholder		$MN_MM_NUM_TOOL_CARRIER / numToBaust		

tcCarr12	$TC_CARR12				
z component of rotary axis v2					
-	0			Double	wr
Multi-line: yes	No. of toolholder		$MN_MM_NUM_TOOL_CARRIER / numToBaust		

tcCarr13	$TC_CARR13				
Angle of rotation alpha1 (in degrees)					
Degree	0			Double	wr
Multi-line: yes	No. of toolholder		$MN_MM_NUM_TOOL_CARRIER / numToBaust		

tcCarr14	$TC_CARR14				
Angle of rotation alpha2 (in degrees)					
Degree	0			Double	wr
Multi-line: yes	No. of toolholder		$MN_MM_NUM_TOOL_CARRIER / numToBaust		

tcCarr15	$TC_CARR15				
x component of offset vector l3					
-	0			Double	wr
Multi-line: yes	No. of toolholder		$MN_MM_NUM_TOOL_CARRIER / numToBaust		

tcCarr16	$TC_CARR16				
y component of offset vector l3					
-	0			Double	wr
Multi-line: yes	No. of toolholder		$MN_MM_NUM_TOOL_CARRIER / numToBaust		

tcCarr17	$TC_CARR17				
z component of offset vector l3					
-	0			Double	wr
Multi-line: yes	No. of toolholder		$MN_MM_NUM_TOOL_CARRIER / numToBaust		

tcCarr18	$TC_CARR18				
x component of offset vector l4					
-	0			Double	wr
Multi-line: yes	No. of toolholder		$MN_MM_NUM_TOOL_CARRIER / numToBaust		

tcCarr19	$TC_CARR19				
y component of offset vector l4					
-	0			Double	wr
Multi-line: yes	No. of toolholder		$MN_MM_NUM_TOOL_CARRIER / numToBaust		

tcCarr2	$TC_CARR2				
y component of offset vector l1					
-	0			Double	wr
Multi-line: yes	No. of toolholder		$MN_MM_NUM_TOOL_CARRIER / numToBaust		

tcCarr20	$TC_CARR20				
z component of offset vector l4					
-	0			Double	wr
Multi-line: yes	No. of toolholder		$MN_MM_NUM_TOOL_CARRIER / numToBaust		

tcCarr21	$TC_CARR21				
Axis identifier of 1st rotary axis					
-	0			String[32]	wr
Multi-line: yes	No. of toolholder		$MN_MM_NUM_TOOL_CARRIER / numToBaust		

tcCarr22	$TC_CARR22				
Axis identifier of 2nd rotary axis					
-	0			String[32]	wr
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]	wr
Multi-line: yes	No. of toolholder		$MN_MM_NUM_TOOL_CARRIER / numToBaust		

tcCarr24	$TC_CARR24				
Offset of 1st rotary axis in degrees					
-	0			Double	wr
Multi-line: yes	No. of toolholder		$MN_MM_NUM_TOOL_CARRIER / numToBaust		

tcCarr25	$TC_CARR25				
Offset of 2nd rotary axis in degrees					
-	0			Double	wr
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	wr
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	wr
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	wr
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	wr
Multi-line: yes	No. of toolholder		$MN_MM_NUM_TOOL_CARRIER / numToBaust		

tcCarr3	$TC_CARR3				
z component of offset vector l1					
-	0			Double	wr
Multi-line: yes	No. of toolholder		$MN_MM_NUM_TOOL_CARRIER / numToBaust		

tcCarr30	$TC_CARR30				
Minimum position of 1st rotary axis					
-	0			Double	wr
Multi-line: yes	No. of toolholder		$MN_MM_NUM_TOOL_CARRIER / numToBaust		

tcCarr31	$TC_CARR31				
Minimum position of 2nd rotary axis					
-	0			Double	wr
Multi-line: yes	No. of toolholder		$MN_MM_NUM_TOOL_CARRIER / numToBaust		

tcCarr32	$TC_CARR32				
Maximum position of 1st rotary axis					
-	0			Double	wr
Multi-line: yes	No. of toolholder		$MN_MM_NUM_TOOL_CARRIER / numToBaust		

tcCarr33	$TC_CARR33				
Maximum position of 2nd rotary axis					
-	0			Double	wr
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]	wr
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]	wr
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]	wr
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			UDoubleword	wr
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	wr
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	wr
Multi-line: yes	No. of toolholder		$MN_MM_NUM_TOOL_CARRIER		

tcCarr4	$TC_CARR4				
x component of offset vector l2					
-	0			Double	wr
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	wr
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	wr
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	wr
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	wr
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	wr
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	wr
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	wr
Multi-line: yes	Number of the tool carrier		$MN_MM_NUM_TOOL_CARRIER		

tcCarr5	$TC_CARR5				
y component of offset vector l2					
-	0			Double	wr
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	wr
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	wr
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	wr
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	wr
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	wr
Multi-line: yes	Number of the tool carrier		$MN_MM_NUM_TOOL_CARRIER		

tcCarr6	$TC_CARR6				
z component of offset vector l2					
-	0			Double	wr
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	wr
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	wr
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	wr
Multi-line: yes	Number of the tool carrier		$MN_MM_NUM_TOOL_CARRIER		

tcCarr7	$TC_CARR7				
x component of rotary axis v1					
-	0			Double	wr
Multi-line: yes	No. of toolholder		$MN_MM_NUM_TOOL_CARRIER / numToBaust		

tcCarr8	$TC_CARR8				
y component of rotary axis v1					
-	0			Double	wr
Multi-line: yes	No. of toolholder		$MN_MM_NUM_TOOL_CARRIER / numToBaust		

tcCarr9	$TC_CARR9				
z component of rotary axis v1					
-	0			Double	wr
Multi-line: yes	No. of toolholder		$MN_MM_NUM_TOOL_CARRIER / numToBaust		

1.7.21 Area T, Mod. TOE: Edge-related coarse total offsets, setup offsets

OEM-MMC: Linkitem /Tool/Compensation/...

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	wr
Multi-line: yes	((EdgeNo-1) * (maxnumEdgeSC * numParams_SC)) + ((EdgeSC - 1)* numParams_SC) + ParameterNo		numParams_SC * maxnumEdgeSC * maxnumCuttEdges_Tool		

1.7.22 Area T, Mod. TOET: Edge-related coarse total offsets, transformed setup offsets

OEM-MMC: Linkitem /Tool/CompTransfor/...

One set of edge-related transformed total offsets exists for each tool edge and operating location.
This module corresponds totally to module T / TOE.

edgeECData					
Transformed location-dependent offsets, setup value					
mm, inch, user defined	0.0			Double	wr
Multi-line: yes	((EdgeNo-1) * (maxnumEdgeSC * numParams_SC)) + ((EdgeSC - 1)* numParams_SC) + ParameterNo		numParams_SC * maxnumEdgeSC * maxnumCuttEdges_Tool		

1.7.23 Area T, Mod. TOS: Edge-related location-dependent fine total offsets

OEM-MMC: Linkitem /Tool/Compensation/...

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) (P5??).

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.
For OPI, see Section OPI variables.

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					
mm, inch, user defined	0.0			Double	wr
Multi-line: yes	((EdgeNo-1) * (maxnumEdgeSC * numParams_SC)) + ((EdgeSC - 1)* numParams_SC) + ParameterNo		numParams_SC * maxnumEdgeSC * maxnumCuttEdges_Tool		

1.7.24 Area T, Mod. TOST: Edge-related location-dependent fine total offsets, transformed

OEM-MMC: Linkitem /Tool/CompTransfor/...

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					
mm, inch, user defined	0.0			Double	wr
Multi-line: yes	((EdgeNo-1) * (maxnumEdgeSC * numParams_SC)) + ((EdgeSC - 1)* numParams_SC) + ParameterNo		numParams_SC * maxnumEdgeSC * maxnumCuttEdges_Tool		

1.7.25 Area T, Mod. TOT: Edge data: Transformed offset data

OEM-MMC: Linkitem /Tool/CompTransfor/...

The MMC 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 MMC can display and modify both transformed and untransformed data (of the same tool if necessary) "simultaneously" (in different applications or different MMCs).

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.

For OPI, see Section OPI variables.

cuttEdgeParam					
Transformed edge offset data and D number list Important: This variable is called "edgeData" in the MMC102.					
mm, inch, user defined	0.0			Double	wr
Multi-line: yes	For edge offset value parameters: (edgeNo - 1) * numCuttEdgeParams + ParameterNo For D numbers: ((numCuttEdgeParams * maxnumCuttEdges_Tool) + EdgeNo)		(numCuttEdgeParams + 1) * maxnumCuttEdges_Tool		

edgeData					
Transformed edge offset data and D number list Important: This variable is called "cuttEdgeParam" in NonWindows MMC and PLC.					
mm, inch, user defined	0.0			Double	wr
Multi-line: yes	For edge offset value parameters: (edgeNo - 1) * numCuttEdgeParams + ParameterNo For D numbers: ((numCuttEdgeParams * maxnumCuttEdges_Tool) + EdgeNo)		(numCuttEdgeParams + 1) * maxnumCuttEdges_Tool		

1.7.26 Area T, Mod. TAD: Application-specific data

OEM-MMC: Linkitem /Tool//...

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 parameter number. Reserved for SIEMENS applications.					
-	0			Double	wr
Multi-line: yes	Tool number T		32000		

1.7.27 Area T, Mod. TAM: Application-specific magazine data

OEM-MMC: Linkitem /Tool//...

Module TAM contains application-specific information about tool magazines.

Application-specific magazine data are all of the same data type.

The application-specific magazine data are reserved for SIEMENS applications.

siemData	$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.					
-	0			UWord	wr
Multi-line: yes	Parameter number		numMagParams_tam		

1.7.28 Area T, Mod. TAO: Application-specific cutting edge data

OEM-MMC: Linkitem /Tool//...

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	wr
Multi-line: yes	(EdgeNo-1) * numCuttEdgeParams_tao + ParameterNo		numCuttEdgeParams_tao * /T/TV/numCuttEdges		

1.7.29 Area T, Mod. TAP: Application-specific magazine location data

OEM-MMC: Linkitem /Tool//...

Data module TAP is organized as a 2-dimensional variable array. This module contains application-specific data of a T area. Every 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) module 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 all of the same data type.

Application-specific magazine location data are reserved for SIEMENS applications.

siemPlaceData	$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.					
-	0			UWord	wr
Multi-line: yes	ParameterNumber + numMagLocParams_tap * MagazineLocationNumber-1		numMagLocParams_tap * magNrPlaces		

1.7.30 Area T, Mod. TAS: Application-specific monitoring data

OEM-MMC: Linkitem /Tool//...

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	wr
Multi-line: yes	ParameterNumber + (EdgeNo -1) * numCuttEdgeParams_tas		numCuttEdgeParams_tas * /T/TV/numCuttEdges		

1.8 Machine and setting data

1.8.1 Area N, Mod. M: Global machine data

OEM-MMC: Linkitem /Nck/Drive/...

Global machine data

MDBA_DRIVE_IS_ACTIVE	MD 13000: DRIVE_IS_ACTIVE[x] x = PlugplaceNo				
Activation of 611D drive / enable high-speed inputs/outputs 0 = inactive 1 = active					
-				Character	wr
Multi-line: yes	Slot number in drive bus		14		

MDCA_DRIVE_LOGIC_NR	MD 13010: DRIVE_LOGIC_NR[x] x = PlugplaceNo				
Logical drive number					
-		0	30	Character	wr
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	wr
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	wr
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	wr
Multi-line: no			1		

MDD_INT_INCR_PER_MM	MD 10200: INT_INCR_PER_MM				
Calculation resolution for linear positions					
-		0,000001	1000	Double	wr
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	wr
Multi-line: no			1		

MDL_POSCTRL_SYSCLOCK_TIME_RATIO	MD 10060: POSCTRL_SYSCLOCK_TIME_RATIO				
Position control cycle factor					
-		1	100	Long Integer	wr
Multi-line: no			1		

MDLA_DRIVE_INVERTER_CODE	MD 13020: DRIVE_INVERTER_CODE[x] x = PlugplaceNo				
Power section code of drive module					
-				Long Integer	wr
Multi-line: yes	Slot number of drive module		14		

MDSA_AXCONF_MACHAX_NAME_TAB	MD 10000: AXCONF_MACHAX_NAME_TAB[x] x = Axis				
Machine axis name					
-				String[16]	wr
Multi-line: yes	Axis index from 0		7		

1.8.2 Area A, Mod. M: Axis-specific machine data

OEM-MMC: Linkitem /Axis/Drive/...

Axis-specific machine data

MDCA_CTRLOUT_MODULE_NR	MD 30110: CTRLOUT_MODULE_NR				
Setpoint assignment: Drive number / module number					
-		1	15	Character	wr
Multi-line: no			1		

MDCA_CTRLOUT_TYPE	MD 30130: CTRLOUT_TYPE				
Type of setpoint output					
-		0	1	Character	wr
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	Byte	wr
Multi-line: yes	Encoder number		2		

MDCA_ENC_TYPE	MD 30240: ENC_TYPE[x] x = PlugplaceNo				
Type of actual value sensing (actual position value) 0 = Simulation 1 = Raw signal generator, high-resolution 2 = Square-wave generator, standard generator with pulse quadruplication 3 = Encoder for stepper motor 4 = Absolute encoder with EnDat interface 5 = Absolute encoder with SSI interface (FM-NC)					
-		0	5	Character	wr
Multi-line: yes	Encoder number		2		

1.8.3 Area N, Mod. SE: Global setting data

OEM-MMC: Linkitem /Nck/Settings/...

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	wr
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	wr
Multi-line: no					

MDD_JOG_REV_SET_VELO	SD 41120: $SN_MDD_JOG_REV_SET_VELO				
JOG velocity for G95					
Degree, user defined				Double	wr
Multi-line: no					

MDD_JOG_SET_VELO	SD 41110: $SN_MDD_JOG_SET_VELO				
JOG velocity for G94					
mm, inch, user defined				Double	wr
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	wr
Multi-line: no					

MDD_JOG_VAR_INCR_SIZE	SD 41010: $SN_MDD_JOG_VAR_INCR_SIZE				
Variable incremental value for JOG mode					
-				Double	wr
Multi-line: no					

1.8.4 Area C, Mod. SE: Channel-specific setting data

OEM-MMC: Linkitem /Channel/Settings/...

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	wr
Multi-line: no					

MDD_THREAD_START_ANGLE	SD 42000: $SC_MDD_THREAD_START_ANGLE				
Starting angle for thread					
Degree				Double	wr
Multi-line: no					

1.8.5 Area A, Mod. SE: Axis-specific setting data

OEM-MMC: Linkitem /Axis/Settings/...

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	wr
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	wr
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	wr
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	wr
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	wr
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	wr
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	wr
Multi-line: yes	Number of machine axis		1		

1.9 Parameters

1.9.1 Area C, Mod. RP: Arithmetic parameters

OEM-MMC: Linkitem /Channel/Parameter/...

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 the NC program. 100 R parameters are defined by default. The number of R parameters can be set via the channel-specific machine data 28050 (MM_NUM_R_PARAM). Up to 1000 R-Parameters can be set.

R	$R[x] x = ParameterNo				PA
R parameter (up to SW 3.2) Attention: This variable should be used for SW releases < 3.3. For later releases use the variable rpa ! Attention: For MMC102 the R number is used as row index !!!					
-				Double	wr
Multi-line: yes	R number		MM_NUM_R_PARAM		

rpa	R[x] x = ParameterNo				PA
R parameter (from SW 3.3) Attention: For MMC102 the R number is used as row index !!!					
-				Double	wr
Multi-line: yes	R number + 1		MM_NUM_R_PARAM + 1		

1.9.2 Area C, Mod. VSYN: Channel-specific user variables for synchronous actions

OEM-MMC: Linkitem /Channel/SelectedFunctionData/...

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	$AC_MARKER[x] x = MarkerNo				
Flag variable, counter for synchronous actions (Note: SYNACT only)					
-				UWord	r
Multi-line: yes	Number of the flag		MD $MC_MM_NUM_AC_MARKER		

acMarkerL	$AC_MARKER[n]				
Flag variable, counter for motion synchronous actions (Note: only with SYNACT)					
-				UDoubleword	r
Multi-line: yes	Flag number		MD $MC_MM_NUM_AC_MARKER		

acParam	$AC_PARAM[x] x = ParameterNo				
Dynamic parameters for synchronous actions (Note: SYNACT only)					
-				Double	r
Multi-line: yes	Number of the parameter		MD $MC_MM_NUM_AC_PARAM		

1.10 Servo

1.10.1 Area N, Mod. SD: Servo data

OEM-MMC: Linkitem /Nck/ServoData/...

The SD module makes servo data available.
These data can be accessed only via the cyc. variable service and the logging function (not individual variable service).

The row index is coded as follows:
The lower three places contain the NCK axis index
The fourth place contains the data format.

The column index is coded as follows:
The lower three places contain the signal ID
The fourth place contains the servo cycle

Data format coding:
0: 32 bit float
1: 64 bit float

Servo cycle coding:
0: The average value of all servo cycle values of one IPO cycle must be applied
61: The minimum value must be applied
62: The maximum value must be applied
n: The value of the nth servo cycle in the course of one IPO cycle must be applied
Maximum value of: IPO_SYSCLOCK_TIME_RATIO / POSCTRL_SYSCLOCK_TIME_RATIO
(1 <=n <=60)

Coding of signal ID:
1: Following error
2: Control deviation
3: Contour deviation
4: Actual position value, measuring system 1
5: Actual position value, measuring system 2
6: Position setpoint
7: Actual velocity value of active encoders (NCK)
8: Drive velocity setpoint (NCK)
9: Compensation value, measuring system 1
10: Compensation value, measuring system 2
11: Controller mode
12: Parameter set
13: Active measuring system
14: Position setpoint at controller input
15: Velocity setpoint at controller input
16: Acceleration setpoint at controller input
17: Velocity feedforward value (plus QEC)
18: Torque/force feedforward value
19: Torque/force limit value
20: Actual velocity, measuring system 1
21: Actual velocity, measuring system 2
22: Interpolation ended signal
23: Exact stop fine signal
24: Exact stop coarse signal
25: QEC learning criterion

26: QEC compensation value
50: Utilization
51: Active power
52: Torque/force setpoint
53: Actual current value (smoothed)
54: Actual speed/velocity motor
55: Valve lift setpoint
56: Actual valve lift
57: Actual pressure cylinder A end
58: Actual pressure cylinder B end
60: Safe actual position
61: Safe actual drive position
62: Safety-relevant input signal NCK
63: Safety-relevant output signal NCK
64: Safety-relevant input signal drive (from PLC)
65: Safety-relevant output signal drive (from PLC)
66: Reaction identifier for NCK
67: Reaction identifier for NCK/drive
68: Result list 1 NCK
69: Result list 1 drive
70: Result list 2 NCK
71: Result list 2 drive
72: Safety partial actual value
73: Actual velocity limit
74: Setpoint velocity limit
75: SI actual value difference
76: Current SI slip speed
77: Current SBR limit

servoDataFl32					
Servo data					
-	0			Float	r
Multi-line: yes	Axis index / data format (see module header)		siehe Bausteinkopf		

servoDataFl64					
Servo data					
-	0			Double	r
Multi-line: yes	Axis index / data format (see module header)		siehe Bausteinkopf		

1.11 Diagnosis data

1.11.1 Area N, Mod. DIAGN: Global diagnostic data

OEM-MMC: Linkitem /Nck/ChannelDiagnose/...

This module contains information about global NC diagnostic data.
The measured time variables are available on the destination hardware only.
The net time is calculated without interrupts by higher priority time levels, the gross time includes the interrupts.
The highest priority time level is the SERVO, followed by the IPO and finally the interpreter/ preparation. To obtain useful minimum and maximum time intervals, the corresponding variables must be initialized before the measurement.

actCycleTimeBrut					
Sum of current gross runtimes of all channels					
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 (from SW 7.1)		4		

actCycleTimeNet					
Sum of current net runtimes of all channels					
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 (from SW 7.1)		4		

compressAbility					
Describes whether the NCK supports the transfer of compressed files Bit0=1: With Huffman algorithm compressed files can be transferred (this corresponds to instruction ";$COMPR=HUFFMAN1" during download)					
-	0	0		UWord	r
Multi-line: yes	1		1		

dp611USpecAccChangeCnt					
The counter is incremented if the NCK changes the available ACC information					
-	0			UDoubleword	r
Multi-line: no			1		

dp611USpecAccKey					
Version and type information about available ACC contents					
-	0			UDoubleword	r
Multi-line: no			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			UDoubleword	r
Multi-line: no			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: no					

dpAxisCfgMachAxisNr					
Machine axis !!CAUTION NCU LINK!!					
-	0	0	INT32_MAX	UDoubleword	r
Multi-line: no			dpAxisCfgNumAxes		

dpAxisCfgNumAxes					
Number of axes entered in the system					
-	0	0	INT32_MAX	UDoubleword	r
Multi-line: no			1		

dpAxisCfgValid					
Axis info is available 0=Information is not available 1=Information is available					
-	0	0	1	UDoubleword	r
Multi-line: no			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: no			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: no			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: no			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	UDoubleword	r
Multi-line: no			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			UDoubleword	r
Multi-line: no			dpAxisCfgNumAxes		

dpAxisStateLifeCntErrEnc2					
This data counters 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			UDoubleword	r
Multi-line: no			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: no			dpBusCfgNumBuses		

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: no			dpBusCfgNumBuses		

dpBusCfgDataExTime					
Data exchange time in [s,s,userdef]					
s, user defined	0	0	DOUBLE_MAX	Double	r
Multi-line: no			dpBusCfgNumBuses		

dpBusCfgNumBuses					
Number of DP buses Currently only one bus standardized acc. to Profibus DP standard					
-	0	0	1	UDoubleword	r
Multi-line: no			1		

dpBusCfgValid					
Bus configuration data are available TRUE= data exist and are initialized FALSE= no data exist					
-	0	0	1	UDoubleword	r
Multi-line: no			1		

dpBusStateAccessDuration Act					
Current access time to communications buffer for DP master					
-	0			UDoubleword	r
Multi-line: no			dpBusCfgNumBuses		

dpBusStateAccessDuration Max					
Maximum access time to communications buffer for DP master					
-	0			UDoubleword	r
Multi-line: no			dpBusCfgNumBuses		

dpBusStateAccessDuration Min					
Minimum access time to communications buffer for DP master					
-	0			UDoubleword	r
Multi-line: no			dpBusCfgNumBuses		

dpBusStateAccessErrCnt1					
Number of bus access errors of type 1 since NCK Start					
-	0			UDoubleword	r
Multi-line: no			dpBusCfgNumBuses		

dpBusStateAccessErrCnt2					
Number of bus access errors of type 2 since NCK Start					
-	0			UDoubleword	r
Multi-line: no			dpBusCfgNumBuses		

dpBusStateAvgCycleBetwe enErr1					
Average number of cycles between two bus access errors of type 1					
-	0			UDoubleword	r
Multi-line: no			dpBusCfgNumBuses		

dpBusStateAvgCycleBetwe enErr2					
Average number of cycles between two bus access errors of type 2					
-	0			UDoubleword	r
Multi-line: no			dpBusCfgNumBuses		

dpBusStateCycleCnt					
Number of bus cycles since NCK Start					
-	0			UDoubleword	r
Multi-line: no			dpBusCfgNumBuses		

dpBusStateDpmAction					
Indicator for operating progress of DP M					
-	0			UDoubleword	r
Multi-line: no			dpBusCfgNumBuses		

dpBusStateDpmActual					
Current status of DP M bus - controlled by DP M					
-	0			UWord	r
Multi-line: no			dpBusCfgNumBuses		

dpBusStateDpmCtrl					
Booting status of processor for DP Master dpcadmin					
-	0			UWord	r
Multi-line: no			dpBusCfgNumBuses		

dpBusStateDpmError					
Error on status transitions					
-	0			UDoubleword	r
Multi-line: no			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: no			dpBusCfgNumBuses		

dpBusStateDpmRequest					
Desired status of DP M bus - request from HOST					
-	0			UWord	r
Multi-line: no			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	UDoubleword	r
Multi-line: no			dpBusCfgNumBuses		

dpClientCfgId					
Identification client NCK/PLC/3RD					
-	0			UWord	r
Multi-line: no			dpClientCfgNumClnt		

dpClientCfgNumClnt					
Number of clients					
-	0	0	INT32_MAX	UDoubleword	r
Multi-line: no			1		

dpClientCfgValid					
Client information is available 0=no client information available 1=client information is available					
-	0	0	1	UDoubleword	r
Multi-line: no			1		

dpClientStateComm					
Client status incl. output release 0=No output enable 1=Client state output enable					
-	0			UWord	r
Multi-line: no			dpClientCfgNumClnt		

dpSlaveCfgAssignBus					
Bus number of the slave					
-	0			UWord	r
Multi-line: no			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: no			dpSlaveCfgNumSlaves		

dpSlaveCfgInputTime					
Time for actual-value sensing See dpSlaveMasterAppCycTime					
s, user defined	0			Double	r
Multi-line: no			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: no			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	UDoubleword	r
Multi-line: no			1		

dpSlaveCfgOutputTime					
Time for setpoint acceptance See dpSlaveMasterAppCycTime					
s, user defined	0			Double	r
Multi-line: no			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	UDoubleword	r
Multi-line: no			1		

dpSlaveIdentNo					
Ident number of the slave					
-	0			UWord	r
Multi-line: no			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: no			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: no			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	UDoubleword	r
Multi-line: no			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: no			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	UDoubleword	r
Multi-line: no			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	UDoubleword	r
Multi-line: no			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	UDoubleword	r
Multi-line: no			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	UDoubleword	r
Multi-line: no			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	UDoubleword	r
Multi-line: no			dpSlotCfgNumSlots		

dpSlotCfgIoType					
I/O identifier 0 = input slot 1 = output slot 2 = diagnosis slot					
-	0	0	2	UWord	r
Multi-line: no			dpSlotCfgNumSlots		

dpSlotCfgLength					
Length in number of bytes					
-	0	0	32	UDoubleword	r
Multi-line: no			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: no			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	UDoubleword	r
Multi-line: no			1		

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: no			dpSlotCfgNumSlots		

dpSlotCfgSlotNr					
Slot number within the slave 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: no			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	UDoubleword	r
Multi-line: no			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: no			dpSlotCfgNumSlots		

dpSlotStateRecvTelegram					
Bit pattern of this slot received by the master in the form of a hexadecimal string					
-	0			String[198]	r
Multi-line: no			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: no			dpSlotCfgNumSlots		

dpSlotStateTelegramType					
Message frame type of slot 0 = Message frame type unknown					
-	0	0	UINT16_MAX	UWord	r
Multi-line: no			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: no			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	UDoubleword	r
Multi-line: no			1		

dpSysCfgValid					
This data indicates whether the configuration data are valid and initialized. TRUE or FALSE					
-	0	0	1	UDoubleword	r
Multi-line: no			1		

dpSysCfgVersionDpm					
Version number of DP M SW as numerical value					
-	0			Double	r
Multi-line: no			dpSysCfgNumMaster		

dpSysCfgVersionDpr					
Actual version Dpr (inaccessible in earlier SW)					
-	0			Double	r
Multi-line: no			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: no			dpSysCfgNumMaster		

dpSysCfgVersionHost					
This data contains the version number of the host SW as a numerical value					
-	0	0	UINT16_MAX	Double	r
Multi-line: no			dpSysCfgNumMaster		

dpSysStateDpmInit					
There are three different initialization states: REQUEST, ACKNOWLEDGE and ERROR					
-	0			UWord	r
Multi-line: no			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. 0: P1-compatible code 5: P5-compatible code 6: P6-compatible code					
-	0	0		UWord	wr
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	wr
Multi-line: yes	1		1		

maxCycleTimeBrut					
Sum of maximum gross runtime of all channels					
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 (from SW 7.1)		4		

maxCycleTimeNet					
Sum of maximum net runtime of all channels					
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 (from SW 7.1)		4		

minCycleTimeBrut					
Sum of minimum gross runtimes of all channel					
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 (from SW 7.1)		4		

minCycleTimeNet					
Sum of minimum net runtimes of all channels					
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 (from SW 7.1)		4		

nckCompileSwitches					
Selected NCK compiler switches Bit0: NDEBUG Bit1: NOTRACES Bit2: EMBARGO Bit3: TARGET					
-				UWord	r
Multi-line: yes	1		1		

pcmciaDataShotAct					
Current access to PCMCIA card: Transferred bytes Data pcmciaShotStatus, pcmciaDataShotSum and pcmciaDataShotAct can be used to implement a status display for PCMCIA card access operations.					
-	0	0		UDoubleword	r
Multi-line: yes	1		1		

pcmciaDataShotSum					
Current access to PCMCIAcard: Total length in bytes. Data pcmciaShotStatus, pcmciaDataShotSum and pcmciaDataShotAct can be used to implement a status display for PCMCIA card access operations.					
-	0	0		UDoubleword	r
Multi-line: yes	1		1		

pcmciaFfsLength					
Length of FFS on PCMCIA card in bytes					
-	0	0		UDoubleword	r
Multi-line: yes	1		1		

pcmciaShotStatus					
Current access to PCMCIA-Karte: Status Data pcmciaShotStatus, pcmciaDataShotSum and pcmciaDataShotAct can be used to implement a status display for PCMCIA card access operations. 0: Not active 1: Write active 2: Read active 3-: Reserved					
-	0	0		UWord	r
Multi-line: yes	1		1		

pcmciaStartFfsOffset					
Start offset of FFS at beginning of PCMCIA card in bytes					
-	0	0		UDoubleword	r
Multi-line: yes	1		1		

pcmciaStartShotOffset					
Current access to PCMCIA card: Start offset at beginning of PCMCIA card in bytes					
-	0	0		UDoubleword	r
Multi-line: yes	1		1		

poweronTime	$AN_POWERON_TIME				
Time since last normal boot (in minutes) Can be written from SW 6.3.					
s, user defined	0.0			Double	wr
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". Can be written from SW 6.3.					
s, user defined	0.0			Double	wr
Multi-line: yes	1		1		

1.11.2 Area C, Mod. DIAGN: Channel-specific diagnosis data

OEM-MMC: Linkitem /Channel/ChannelDiagnose/...

This module contains information about the channel-specific NC diagnostic data.
The measured time variables are available on the destination hardware only. The net time is calculated without interrupts by higher priority time levels, the gross time includes the interrupts. The highest priority time level is the SERVO, followed by the IPO and finally the interpreter/ preparation. To obtain useful minimum and maximum time intervals, the corresponding variables must be initialized before the measurement.

acIpoBuf	$AC_IPO_BUF				
Level of IPO buffer (number of blocks)					
-	0	0		UWord	r
Multi-line: yes	1		1		

actCycleTimeBrut					
Current gross cycle time line index 1: SERVO-task line index 2: IPO-task line index 3: interpreter/preparation-task line index 4: PLC line index 5: SYNACT (from SW 7.1)					
ms				Double	r
Multi-line: yes	1 - 3		3		

actCycleTimeNet					
Current net cycle time line index 1: SERVO-task line index 2: IPO-task line index 3: interpreter/preparation-task line index 4: PLC line index 5: SYNACT (from SW 7.1)					
ms				Double	r
Multi-line: yes	1 - 3		3		

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. Can be written as of SW 6.3.					
s, user defined	0.0			Double	wr
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. Can be written as of SW 6.3.					
s, user defined	0.0			Double	wr
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 cycle time line index 1: SERVO-task line index 2: IPO-task line index 3: interpreter/preparation task line index 4: PLC line index 5: SYNACT (from SW 7.1)					
ms				Double	wr
Multi-line: yes	1 - 3		3		

maxCycleTimeNet					
Maximum net cycle time line index 1: SERVO-task line index 2: IPO-task line index 3: interpreter/preparation-task line index 4: PLC line index 5: SYNACT (from SW 7.1)					
ms				Double	wr
Multi-line: yes	1 - 3		3		

minCycleTimeBrut					
Minimum gross cycle time line index 1: SERVO-task line index 2: IPO-task line index 3: interpreter/preparation task line index 4: PLC line index 5: SYNACT (from SW 7.1)					
ms				Double	wr
Multi-line: yes	1 - 3		3		

minCycleTimeNet					
Minimum net cycle time line index 1: SERVO-task line index 2: IPO-task line index 3: interpreter/preparation-task line index 4: PLC line index 5: SYNACT (from SW 7.1)					
ms				Double	wr
Multi-line: yes	1 - 3		3		

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. Can be written as of SW 6.3.					
s, user defined	0.0			Double	wr
Multi-line: yes	1		1		

1.11.3 Area N, Mod. ETPD: Data lists for protocolling

OEM-MMC: Linkitem /Nck/ProtocolData/...

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	wr
Multi-line: yes	2 + 5 * (n-1)		2 + 5 * (numData- 1)		

col					
Variable specification of nth OPI data in list: col					
-				UWord	wr
Multi-line: yes	4 + 5 * (n-1)		4 + 5 * (numData- 1)		

numData					
Number of data in the list <= maxnumTraceProtData					
-		0	maxnumTraceProtData	UWord	wr
Multi-line: yes	1		1		

row					
Variable specification of nth OPI data in list: row					
-				UWord	wr
Multi-line: yes	5 + 5 * (n-1)		5 + 5 * (numData- 1)		

type					
Variable specification of nth OPI data in list: type					
-				UWord	wr
Multi-line: yes	6 + 5 * (n-1)		6 + 5 * (numData- 1)		

unit					
Variable specification of nth OPI data in list: unit					
-				UWord	wr
Multi-line: yes	3 + 5 * (n-1)		3 + 5 * (numData- 1)		

1.11.4 Area C, Mod. ETP: Types of events

OEM-MMC: Linkitem /Channel/ProtocolEvent/...

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 6.4)
48 = IPO4 (from SW 6.4)

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 ?: not yet implemented)
32 = BLOCK_END_P1a Block end (run in) (from SW ?: not yet implemented)
44 = BLOCK_END_I1 Block end (interpreter) (from SW 6.4)
43 = NC_LEVEL_CHG Level change during part program processing (from SW 6.4)

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 6.4)
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 6.4)

30 = PROT_STOP_TRIG Stop trigger has triggered
(from SW 6.4)
46 = PROT_START Start logging
(from SW 6.4)
45 = PROT_STOP Stop logging
(from SW 6.4)

Acyclic events triggered by buttons
42 = CANCEL_BUTTON The Cancel button was pressed (from SW 6.4)
Acyclic events triggered by alarms
41 = ALARM_REPORTED An alarm has occurred (from SW 6.4)

Acyclic events triggered by synchronized action
36 = SYNC_ACT_ACTIV Activating synchronized action (from SW 6.4)
37 = SYNC_ACT_DEACT Deactivating synchronized action (from SW 6.4)
38 = SYNC_ACT_FIRE Synchronized action triggers (from SW 6.4)
Acyclic events triggered by tool
25 = TOOL_CHANGE tool change
(from SW 6.2)
27 = TOOL_CHANGE_S1 tool change (search run with computation) (from SW 6.3)
34 = TOOL_CHANGE_P1 tool change (run in) (from SW 6.4)
26 = CUTTEDGE_CHANGE cutting edge change
(from SW 6.2)
28 = CUTTEDGE_CHANGE_S1 cutting edge change (search run with computation) (from SW 6.3)
35 = CUTTEDGE_CHANGE_P1 cutting edge change (run in) (from SW 6.4)
Acyclic events triggered by PLC
39 = PLC_OB_1 PLC OB1 started
(from SW 6.4)
40 = PLC_OB40 PLC OB40 started
(from SW 6.4)

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	wr
Multi-line: yes	Event (see module header)		siehe Bausteinkopf		

countActivated					
Number of times the event has occurred					
-	0			UWord	r
Multi-line: no					

dataListIndex					
Index of data list to be used All valid columns in module ETPD - 1)					
-	0	0		UWord	wr
Multi-line: yes	Event (see module header)		siehe Bausteinkopf		

dataProtok					
Number of bytes entered in the Fifo file					
-	0			UWord	r
Multi-line: no					

dataUploaded					
Number of bytes already uploaded from the Fifo file					
-	0			UWord	r
Multi-line: no					

eventActive					
Event state 0: Not active 1: Active 2: Deactivate and release data set					
-	0	0	2	UWord	wr
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		

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 log file					
-	0	0		UWord	wr
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]	wr
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 block header)		siehe Bausteinkopf		

skip					
Number of events to be skipped					
-	0	0		UWord	wr
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	wr
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	wr
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	wr
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		

1.12 HMI / MMC State data

1.12.1 Area M, Mod. S: Internal status data MMC

OEM-MMC: Linkitem //State/...

Some internal status data of the MMC can be accessed via this module.

/Nck/Nck/ActApplication					
Current application for display in MMC					
-				String[32]	wr
Multi-line: no					

/Nck/Nck/ActBag					
Current operating mode for display in MMC					
-				Character	wr
Multi-line: no					

/Nck/Nck/Channel					
Current channel for display in MMC					
-				Character	wr
Multi-line: no					

/Nck/Nck/CoordSystem					
Coordinate system for display in MMC					
-				Character	wr
Multi-line: no					

■

Für Notizen

2

2 Interface Signals solution line

2.1 Data modules (DB) of the PLC application interface

Please find the descriptin of Data modules (DB) of the PLC application interface in chapter 5.3.

2.2 Interface signals of the PLC application interface

General

In the following list of interface signals, a reference to relevant documentation is provided for every signal.
This reference specifies the section number or the short designation of the description of functions, please refer to

References: /FB/, xx, ”yyy”

xx Short designation of individual description of functions (e.g.: /A2/)

yyy Name of description of functions (e.g.: ”Various interface signals” or title of the guide)

Inverse signals

Signals marked with a ”*” are so-called inverse signals. These signals initiate the appropriate function when a 0 signal appears rather than a 1 signal (e.g. MCP, byte n+2.0: *NC STOP).

Legend

- In STEP7, DBB means data module byte
- In STEP7, DBW means data module word (16 bits)
- In STEP7, DBD means data module double word (32 bits)

Hinweis

Please refere also to
SINAMICS S120 Installation and Start-UP Manual 6SL3097-2AF00-0BP3,

SINAMICS S List Manual 6SL3097-2AP00-0BP3

for **SINAMICS drives**.

2.2.1 Signals from/to machine control panel, M version

<table>
<tr><th colspan="9">Signals from machine control panel (keys)</th></tr>
<tr><th>Byte</th><th>Bit 7</th><th>Bit 6</th><th>Bit 5</th><th>Bit 4</th><th>Bit 3</th><th>Bit 2</th><th>Bit 1</th><th>Bit 0</th></tr>
<tr><td rowspan="2">IB n + 0</td><td colspan="4">Spindle speed override</td><td colspan="4">Operating mode</td></tr>
<tr><td>D</td><td>C</td><td>B</td><td>A</td><td>JOG</td><td>TEACH IN</td><td>MDA</td><td>AUTO</td></tr>
<tr><td rowspan="2">IB n + 1</td><td colspan="8">Machine function</td></tr>
<tr><td>REPOS</td><td>REF</td><td>var. INC</td><td>10000 INC</td><td>1000 INC</td><td>100 INC</td><td>10 INC</td><td>1 INC</td></tr>
<tr><td>IB n + 2</td><td>Key-switch position 0</td><td>Key-switch position 2</td><td>Spindle start</td><td>*Spindle stop</td><td>Feed start</td><td>*Feed stop</td><td>NC Start</td><td>*NC Stop</td></tr>
<tr><td rowspan="2">IB n + 3</td><td rowspan="2">Reset</td><td rowspan="2">Key-switch position 1</td><td rowspan="2">Single block</td><td colspan="5">Feedrate override</td></tr>
<tr><td>E</td><td>D</td><td>C</td><td>B</td><td>A</td></tr>
<tr><td rowspan="2">IB n + 4</td><td colspan="3">Direction keys</td><td rowspan="2">Keyswitch position 3</td><td colspan="4">Direction keys</td></tr>
<tr><td>+
R15</td><td>-
R13</td><td>Rapid traverse
R14</td><td>x
R1</td><td>4th axis
R4</td><td>7th axis
R7</td><td>R10</td></tr>
<tr><td rowspan="2">IB n + 5</td><td colspan="8">Axis selection</td></tr>
<tr><td>Y
R2</td><td>Z
R3</td><td>5th axis
R5</td><td>Traverse command MCS/WCS
R12</td><td>R11</td><td>R9</td><td>8th axis
R8</td><td>6th axis
R6</td></tr>
<tr><td rowspan="2">IB n + 6</td><td colspan="8">Unassigned customer keys</td></tr>
<tr><td>T9</td><td>T10</td><td>T11</td><td>T12</td><td>T13</td><td>T14</td><td>T15</td><td></td></tr>
<tr><td rowspan="2">IB n + 7</td><td colspan="8">Unassigned customer keys</td></tr>
<tr><td>T1</td><td>T2</td><td>T3</td><td>T4</td><td>T5</td><td>T6</td><td>T7</td><td>T8</td></tr>
</table>

Signals to machine control panel (LEDs)								
Byte	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
QB n + 0	Machine function				Operating mode			
	1000 INC	100 INC	10 INC	1 INC	JOG	TEACH IN	MDA	AUTO
QB n + 1	Feed start	*Feed stop	NC Start	*NC Stop	Machine function			
					REPOS	REF	var. INC	10000 INC
QB n + 2	Axis selection					Single block	Spindle start	*Spindle stop
	Direction key - R13	X R1	4th axis R4	7th axis R7	R10			
QB n + 3	Axis selection							
	Z R3	5th axis R5	Travel command MCS/WCS R12	R11	R9	8th axis R8	6th axis R6	Direction key + R15
QB n + 4	Unassigned customer keys							Y
	T9	T10	T11	T12	T13	T14	T15	R2
QB n + 5	Unassigned customer keys							
	T1	T2	T3	T4	T5	T6	T7	T8

Note

With the SINUMERIK 840D, the machine control panel is assigned to the input/output area by GP parameters; as a standard, initial address 0 is specified for the input and output areas.

With FM-NC, the initial address is set via the SDB 210. For the supplied SDB 210, initial address 120 is specified. If another initial address is desired, this must be specified via the STEP 7 Package Communication Configuration. Note that the GD parameters given automatically through Communication Configuration must be set on the machine control panel.

2.2.2 Signals from/to machine control panel, T version

Signals from machine control panel (keys)								
Byte	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
IB n + 0	Spindle speed override				Operating mode			
	D	C	B	A	JOG	TEACH IN	MDA	AUTO
IB n + 1	Machine function							
	REPOS	REF	var. INC	10000 INC	1000 INC	100 INC	10 INC	1 INC
IB n + 2	Keyswitch position 0	Keyswitch position 2	Spindle start	*Spindle stop	Feed start	*Feed stop	NC Start	*NC Stop
IB n + 3		Keyswitch position 1		Feed override				
	Reset		Single block	E	D	C	B	A
IB n + 4				Keyswitch position 3	Direction keys			
	R15	R13	R14		+Y R1	-Z R4	-C R7	R10
IB n + 5	Direction keys							
	+X R2	+C R3	Rapid traverse override R5	Travel command MCS/WCS R12	R11	-Y R9	-X R8	+Z R6
IB n + 6	Unassigned customer keys							
	T9	T10	T11	T12	T13	T14	T15	
IB n + 7	Unassigned customer keys							
	T1	T2	T3	T4	T5	T6	T7	T8

Signals to machine control panel (LEDs)								
Byte	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
QB n + 0	Machine function				Operating mode			
	1000 INC	100 INC	10 INC	1 INC	JOG	TEACH IN	MDA	AUTO
QB n + 1	Feed start	*Feed stop	NC Start	*NC Stop	Machine function			
					REPOS	REF	var. INC	10000 INC
QB n + 2	Direction keys					Single block	Spindle start	*Spindle stop
	R13	+Y R1	-Z R4	-C R7	R10			
QB n + 3	Direction keys							
	R3	R5	Travel command MCS/WCS	R11	-Y R9	-X R8	+Z R6	R15

QB n + 4	Unassigned customer keys							Direction key +X
	T9	T10	T11	T12	T13	T14	T15	R2
QB n + 5	Unassigned customer keys							
	T1	T2	T3	T4	T5	T6	T7	T8

2.2.3 Signals from/to slimline machine control panel

Signals from slimline machine control panel (keys and switches)								
Byte	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
IB n + 0	Spindle speed override				Operating mode			
	*NC Stop	SP -	SP 100%	SP +	SINGLEB	JOG	MDA	AUTOM.
IB n + 1	Spindle				Keyswitch	Machine function		
	NC Start	SP right	*SP Stop	SP left	SS 3	REF.	REPOS	Teach in
IB n + 2	Feedrate			Keyswitch	Machine functions			
	START	*STOP	var. INC	SS 0	1000 INC	100 INC	10 INC	1 INC
IB n + 3		Keyswitch		Feed override				
	RESET	SS 2	SS 1	E	D	C	B	A
IB n + 4	Direction keys			Optional customer keys				
	(+) R15	(-) R13	Rapid traverse R14	KT4	KT3	KT2	KT1	KT0
IB n + 5			Axis selection					
	T17	KT5	6	5	4	Z	Y	X
IB n + 6	Unassigned customer keys							
	T9	T10	T11	T12	T13	T14	T15	T16
IB n + 7	Unassigned customer keys							
	T1	T2	T3	T4	T5	T6	T7	T8

Signals to slimline machine control panel (LEDs)								
Byte	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
QB n + 0	Spindle speed override				Operating mode			
	NC Stop	SP -	SP 100 %	SP +	SINGLEB	JOG	MDA	AUTOM.
QB n + 1	Spindle					Machine function		
	NC Start	SP right	SP Stop	SP left	Unassigned	REF.	REPOS	Teach in
QB n + 2	Feedrate				Machine functions			
	START	STOP	var. INC	Unassigned	1000 INC	100 INC	10 INC	1 INC
QB n + 3	Unassigned							
	Unassigned	Unassigned	Unassigned	Unassigned	Unassigned	Unassigned	Unassigned	Unassigned
QB n + 4	Direction keys			Optional customer keys				
	(+) R15	(-) R13	Rapid traverse R14	KT4	KT3	KT2	KT1	KT0
QB n + 5			Axis selection					
	T17	KT5	6	5	4	Z	Y	X
QB n + 6	Unassigned customer keys							
	T9	T10	T11	T12	T13	T14	T15	T16
QB n + 7	Unassigned customer keys							
	T1	T2	T3	T4	T5	T6	T7	T8

2.2.4 Signals from/to handheld unit (HHU)

Signals from handheld unit (keys) (input display)								
Byte	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
IB n + 0	Reserved							
IB n + 1	Reserved							
IB n + 2								
	T9	T7	T6	T5	T4	T3	T2	T1
IB n + 3								
	T16	T15	T14	T13	T12	T11	T10	T9
IB n + 4								
	T24	T23	T22	T21				
IB n + 5	Acknowl-edgement	Keyswitch	Rapid traverse/feed override switch					
	Digital display		E	D	C	B	A	

Signals to handheld unit (LEDs) (Output display, LEDs)								
Byte	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
QB n + 0	always 1							
QB n + 1	New data for selected line							Line selection
QB n + 2	L8	L7	L6	L5	L4	L3	L2	L1
QB n + 3	L16	L15	L14	L13	L12	L11	L10	L9
HHU digital display								
QB n + 4	1st character (right) of the selected line							
QB n + 5	2nd character of the selected line							
QB ...								
QB n + 18	15th character of the selected line							
QB n + 19	16th character (left) of the selected line							

Note
The parameterization is described in the Installation and Start-Up Guide and in the Description of Functions P3 sl ”Basic PLC program”.

References: /BH/, ”Operator Components Manual”

2.2.5 Signals from/to handheld programming unit (HT6)

Signals from machine control panel simulation Interface HT6→PLC								
Byte	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
IB n + 0	Function key block							
	REF	TEACH	AUTO	MDA	JOG	QUIT	RESET	WCS/MCS
IB n + 1	Function key block							
		FCT15	FCT14	BigFct	FCT12	FCT11	INC	REPOS
IB n + 2	JOG keys positive direction							
		If 1: Ax1-Ax6= Ax7-Ax12	Ax6	Ax5	Ax4	Ax3	Ax2	Ax1
IB n + 3	JOG keys negative direction							
			Ax6	Ax5	Ax4	Ax3	Ax2	Ax1
IB n + 4	Shift keys							
	Signal	Diagno	Service	System	Param	Correct	Progr.	Mach.
IB n + 5	Shift keys							
	BF16	BF15	BF14	BF13	BF12	Step	Modify	Insert
IB n + 6	Start key block							
	Res.	HT 6	VAL+	VAL-	SF2	SF1	START	STOP
IB n + 7								
				E	D	C	B	A

Signals to machine control panel simulation Interface PLC→HT6								
Byte	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
QB n + 0	Function key block							
	REF	TEACH	AUTO	MDA	JOG	QUIT	RESET	WCS/MCS
QB n + 1	Function key block							
		FCT15	FCT14	BigFct	FCT12	FCT11	INC	REPOS
QB n + 2		Axes 7–12 selected	JOG keys positive direction					
QB n + 2								
			Ax6	Ax5	Ax4	Ax3	Ax2	Ax1
QB n + 3	JOG keys negative direction							
	For WCS: No MCS Ax4 to Ax6		Ax6	Ax5	Ax4	Ax3	Ax2	Ax1
QB n + 4	Shift keys							
	Signal	Diagno	Service	System	Param	Correct	Progr.	Mach.
QB n + 5	Shift keys							
	BF16	BF15	BF14	BF13	BF12	Step	Modify	Insert
QB n + 6	Start key block							
			VAL+	VAL-	SF2	SF1	START	STOP
QB n + 7								

2.2.6 PLC messages (DB 2)

DB2	Signals for PLC messages (PLC→HMI), /P3/							
Byte	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
	Channel 1							
0	510007	510006	510005	510004	510003	510002	510001	510000
	Feed disable (alarm no.: 510000–510015)							
1	510015	510014	510013	510012	510011	510010	510009	510008
2	Feed and read-in disable byte1 (alarm no.: 510100–510131)							
3	Feed and read-in disable byte 2 (alarm no.: 510108–510115)							
4	Feed and read-in disable byte 3 (alarm no.: 510116–510123)							
5	Feed and read-in disable byte 4 (alarm no.: 510124–510131)							
6	Read-in disable byte 1 (alarm no.: 510200–510207)							
7	Read-in disable byte 2 (alarm no.: 510208–510215)							

8	Read-in disable byte 3 (alarm no.: 510216–510223)							
9	Read-in disable byte 4 (alarm no.: 510224–510231)							
10	NC Start disable byte 1 (alarm no.: 510300–510307)							
11	NC Start disable byte 2 (alarm no.: 510308–510315)							
12	Feed stop GEOaxis 1 byte 1 (alarm no.: 511100–511107)							
13	Feed stop GEOaxis 1 byte 2 (alarm no.: 511108–511115)							
14	Feed stop GEOaxis 2 byte 1 (alarm no.: 511200–511207)							
15	Feed stop GEOaxis 2 byte 2 (alarm no.: 511208–511215)							
16	Feed stop GEOaxis 3 byte 1 (alarm no.: 511300–511307)							
17	Feed stop GEOaxis 3 byte 2 (alarm no.: 511308–511315)							
	Channel 2							
18	520007	520006	520005	520004	520003	520002	520001	520000
	Feed disable (alarm no.: 520000-520015)							
19	520015	520014	520013	520012	520011	520010	520009	520008
20-23	Feed and read-in disable byte 1-4 (alarm no.: 520100–520131)							
24-27	Read-in disable byte 1-4 (alarm no.: 520200–520231)							
28-29	NC Start disable byte 1-2 (alarm no.: 520300–520315)							
30-31	Feed stop GEOaxis 1 byte 1-2 (alarm no.: 521100-521115)							
32-33	Feed stop GEOaxis 2 byte 1-2 (alarm no.: 521200-521215)							
34-35	Feed stop GEOaxis 3 byte 1-2 (alarm no.: 521300-521315)							
36-143	As from channel 3, please refer to the following table							

Channel areas in DB2

Area	Address	Signal number
Channel 1, see above	DBX 0.0 - DBX 11.7	510.000 - 510.231
Channel 1, geo axes	DBX 12.0 - DBX 17.7	511.100 - 511.315
Channel 2, see above	DBX 18.0 - DBX 29.7	520.000 - 520.231
Channel 2, geo axes	DBX 30.0 - DBX 35.7	521.100 - 521.315
Channel 3	DBX 36.0 - DBX 47.7	530.000 - 530.231
Channel 3, geo axes	DBX 48.0 - DBX 53.7	531.100 - 531.315
Channel 4	DBX 54.0 - DBX 65.7	540.000 - 540.231
Channel 4, geo axes	DBX 66.0 - DBX 71.7	541.100 - 541.315
Channel 5	DBX 72.0 - DBX 83.7	550.000 - 550.231
Channel 5, geo axes	DBX 84.0 - DBX 89.7	551.100 - 551.315
Channel 6	DBX 90.0 - DBX 101.7	560.000 - 560.231
Channel 6, geo axes	DBX 102.0 - DBX 107.7	561.100 - 561.315
Channel 7	DBX 108.0 - DBX 119.7	570.000 - 570.231
Channel 7, geo axes	DBX 120.0 - DBX 125.7	571.100 - 571.315
Channel 8	DBX 126.0 - DBX 137.7	580.000 - 580.231
Channel 8, geo axes	DBX 138.0 - DBX 143.7	581.100 - 581.315
Channel 9, channel 10 not implemented		

Axis areas in DB2

	Axis/spindle							
144	600107	600106	600105	600104	600103	600102	600101	600100
	Feed stop/spindle stop (alarm no.: 600100-600015) for axis/spindle 1							
145	600115	600114	600113	600112	600111	600110	600109	600108
146-147	Feed stop/spindle stop (alarm no.: 600200-600215) for axis/spindle 2							
148-149	Feed stop/spindle stop (alarm no.: 600300-600315) for axis/spindle 3							
150-151	Feed stop/spindle stop (alarm no.: 600400-600415) for axis/spindle 4							
152-153	Feed stop/spindle stop (alarm no.: 600500-600515) for axis/spindle 5							
154-155	Feed stop/spindle stop (alarm no.: 600600-600615) for axis/spindle 6							
156-157	Feed stop/spindle stop (alarm no.: 600700-600715) for axis/spindle 7							
158-159	Feed stop/spindle stop (alarm no.: 600800-600815) for axis/spindle 8							
160-161	Feed stop/spindle stop (alarm no.: 600900-600915) for axis/spindle 9							
162-163	Feed stop/spindle stop (alarm no.: 601000-601015) for axis/spindle 10							
164-165	Feed stop/spindle stop (alarm no.: 601100-601115) for axis/spindle 11							
166-167	Feed stop/spindle stop (alarm no.: 601200-601215) for axis/spindle 12							
168-169	Feed stop/spindle stop (alarm no.: 601300-601315) for axis/spindle 13							
170-171	Feed stop/spindle stop (alarm no.: 601400-601415) for axis/spindle 14							
172-173	Feed stop/spindle stop (alarm no.: 601500-601515) for axis/spindle 15							
174-175	Feed stop/spindle stop (alarm no.: 601600-601615) for axis/spindle 16							
176-177	Feed stop/spindle stop (alarm no.: 601700-601715) for axis/spindle 17							
178-179	Feed stop/spindle stop (alarm no.: 601800-601815) for axis/spindle 18							
	Axes 19 -31 not implemented							

User areas

	User area 0 Bytes 1 - 8							
180	700007	700006	700005	700004	700003	700002	700001	700000
...	User area 0 (alarm no.: 700000-700063)							
187	700063	700062	700061	700060	700059	700058	700057	700056
188-195	User area 1 Bytes 1 - 8 (alarm no.: 700100-700163)							
...								
372-379	User area 24 Bytes 1 - 8 (alarm no.: 702400-702463)							

Note

In DB2, the assignment is made between message/alarm number, text and area identifier. All alarm or message bits are automatically transferred to the user interface (channel, axis/spindle) through appropriate parameter settings. If these parameter settings are not made, the bit transfer must be programmed in the user program. The user interface can be further influenced after the block for the error/operational messages has been called. Only signals of the channels and axes declared in the NC machine data can be transferred and texts displayed.

The user must acknowledge all error messages generated. Operational messages are displayed only for as long as the relevant condition prevails.

The number of user areas can be parameterized via FB 1.

DB2/DB3 must be deleted after changing the configuration (FB1: MsgUser).

Definition of error and operational messages /P3/							
Byte no. of DB2 / Error message EM or operational message OM							
7 / EM	6 / EM	5 / OM	4 / OM	3 / EM	2 / EM	1 / OM	0 / EM
15 / OM	14 / EM	13 / OM	12 / EM	11 / OM	10 / EM	9 / OM	8 / OM
23 / OM	22 / OM	21 / EM	20 / EM	19 / OM	18 / EM	17 / OM	16 / EM
31 / OM	30 / EM	29 / OM	28 / EM	27 / OM	26 / OM	25 / EM	24 / EM
				35 / OM	34 / EM	33 / OM	32 / EM
151 / OM	150 / EM	149 / OM	148 / EM	147 / OM	146 / EM	145 / OM	144 / EM
159 / OM	158 / EM	157 / OM	156 / EM	155 / OM	154 / EM	153 / OM	152 / EM
187 / OM	186 / OM	185 / OM	184 / OM	183 / EM	182 / EM	181 / EM	180 / EM
195 / OM	194 / OM	193 / OM	192 / OM	191 / EM	190 / EM	189 / EM	188 / EM

Example

The alarms numbered from 510200 to 510207 can be generated via DB2, DBB6 (read-in disable channel 1). These alarms are defined as error messages as standard.

2.2.7 Signals to NC (DB 10)

On-board input and output signals from NCK

DB10	Signals to NC (PLC→NC)							
Byte	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
DBB **0**	Disabling of digital NCK inputs /A2/							
	Digital inputs without hardware #)				On-board inputs §)			
	Input 8	Input 7	Input 6	Input 5	Input 4	Input 3	Input 2	Input 1
DBB **1**	Setting of digital NCK inputs from PLC							
	Digital inputs without hardware #)				On-board inputs §)			
	Input 8	Input 7	Input 6	Input 5	Input 4	Input 3	Input 2	Input 1
DBB **2, 3 unas-signed**								
DBB **4**	Disabling of digital NCK outputs /A2/							
	Digital outputs without hardware #)				On-board outputs §)			
	Output 8	Output 7	Output 6	Output 5	Output 4	Output 3	Output 2	Output 1
DBB **5**	Overwrite screenform of digital NCK /A2/ outputs							
	Digital outputs without hardware #)				On-board outputs §)			
	Output 8	Output 7	Output 6	Output 5	Output 4	Output 3	Output 2	Output 1
DBB **6**	Setting value of digital NCK outputs from PLC /A2/							
	Digital outputs without hardware #)				On-board outputs §)			
	Output 8	Output 7	Output 6	Output 5	Output 4	Output 3	Output 2	Output 1
DBB **7**	Input screenform of digital NCK outputs /A2/							
	Digital outputs without hardware #)				On-board outputs §)			
	Output 8	Output 7	Output 6	Output 5	Output 4	Output 3	Output 2	Output 1
DBB **8-29**	Machine axis number table for FC 19, 24, 25, 26 (1st MCP)							
DBB **30**	Upper limit of machine axis numbers for FC 19, 24 (1st MCP) With 0, the max. number of machine axis numbers applies							
DBB **32-53**	Machine axis number table for FC 19, 24, 25, 26 (2nd MCP)							
DBB **54**	Upper limit of machine axis numbers for FC 19, 24 (2nd MCP) With 0, the max. number of machine axis numbers applies							

Note

#) Bits 4-7 of the digital input and NCK outputs can be processed by the PLC even though there are no hardware I/Os available for this. Therefore, these bits can be used in addition to the information exchange between NCK and PLC.

§) On the 840D, the digital inputs and outputs 1 to 4 of the NCK are physically on-board. On the FM-NC, there are no hardware I/Os for bit 0 to bit 3. These can be processed by the PLC according to #).

General signals to NCK (DB10)

DB10	Signals to NC (PLC→NC)							
Byte	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
DBB **56**	Keyswitch /A2/ Position 3	Position 2	Position 1	Position 0		Acknowl. EMER-GENCY STOP /N2/	EMER-GENCY STOP /N2/	
DBB **57**					PC shutdown Only 840Di evaluated			INC inputs in mode group area active
DBB **58 - 59**								

2.2.8 Signals from/to NCK/HMI (DB 10)

On-board NCK inputs and outputs (DB 10)

DB10	Signals from (NCK -> PLC)							
Byte	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
DBB **60**					Actual value of the digital ON-BOARD inputs of the NCK On-board inputs §) Input 4	Input 3	Input 2	Input 1
DBB **61-63**								
DBB **64**	Setpoint for the digital outputs of the NCK without hardware Output 8	Output 7	Output 6	Output 5	Setpoint for the digital on-board outputs of the NCK Output 4	Output 3	Output 2	Output 1
DBB **65-67**				Unassigned				
DBB **68**	Handwheel 1 moved							
DBB **69**	Handwheel 2 moved							
DBB **70**	Handwheel 3 moved							

DBB **71**	Modification counter inch/metric system of units							
DBB **72** HT6	Status of the actual value display indicated (1st MCP)						Machine/ Work	Valid display
DBB **73** HT6	Status of the actual value display indicated (2nd MCP)						Machine/ Work	Valid display
DBB **74-79** HT6	Machine axis numbers of the displayed axes (1st MCP) MCP1AxisFromHMI							
DBB **80-85** HT6	Machine axis numbers of the displayed axes (2nd MCP) MCP2AxisFromHMI							
DBB **86**	Reserved							
DBB **88**	Reserved							

Note

#) Although no associated hardware I/Os exist, the PLC can process bits 4-7 of the digital inputs and NCK outputs. Consequently, these bits can also be used to transfer information between the NCK and the PLC.

§) The digital inputs and outputs 1 to 4 of the NCK exist as on-board hardware for the 840D. No hardware I/Os are available for bits 0-3 of the FM-NC. In accordance with #), these can be processed by the PLC.

Selection/status signals from HMI (DB 10)

DB 10	**Signals from NC (NCK→PLC)**							
Byte	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
DBB **90** ePS to PLC								
DBB **91** PLC to ePS								
DBB **92**	free							
DBB **93**	free							
DBB **94**	free							

DBB **95**	free							
DBB **96**	free							
DBB **97** HMI--> PLC					Channel number for handwheel 1 /H1/ D	C	B	A
DBB **98** HMI--> PLC					Channel number for handwheel 2 /H1/ D	C	B	A
DBB **99** HMI--> PLC					Channel number for handwheel 3 /H1/ D	C	B	A
DBB **100** HMI--> PLC	Machine axis	Handwheel selected	Contour handwheel	Axis number for handwheel 1 /H1/ E	D	C	B	A
DBB **101** HMI--> PLC	Machine axis	Handwheel selected	Contour handwheel	Axis number for handwheel 2 /H1/ E	D	C	B	A
DBB **102** HMI--> PLC	Machine axis	Handwheel selected	Contour handwheel	Axis number for handwheel 3 /H1/ E	D	C	B	A
DBB **103** HMI--> PLC	HMI-battery alarm	HMI tempera-ture limit	AT box ready	HMI fan monitoring	HMI HD monitoring			Remote diagnosis active /FBFE/

General signals from NCK (DB 10)

DB 10	Signals from NC (NCK→PLC)							
Byte	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
DBB **104**	NCK CPU ready /A2/	1 st OB1 cycle				HHU ready	MCP 2 ready	MCP 1 ready
DBB **105**								Too-lmanage-ment command cancellation
DBB **106**							EMERGEN-CY STOP active /N2/	
DBB **107**	Inch system	NCU-link active					Probe actuated /M4/	
							Probe 2	Probe 1
DBB **108**	NC ready /A2/	Drive ready /FBA/	Drives in cyclic operation		HMI-CPU Ready (HMI to OPI) /A2/	HMI CPU Ready (HMI to MPI) /A2/	HMI2 CPU ready E_HMI2 Ready	
DBB **109**	NCK battery alarm /A2/	Air temp. alarm /A2/	Heat sink temp. alarm NCU 573	PC operating system fault				NCK alarm present /A2/
DBB **110**	Software cams minus /N3/							
	7	6	5	4	3	2	1	0
DBB **111**	Software cams minus /N3/							
	15	14	13	12	11	10	9	8
DBB **112**	Software cams minus /N3/							
	23	22	21	20	19	18	17	16
DBB **113**	Software cams minus /N3/							
	31	30	29	28	27	26	25	24
DBB **114**	Software cams plus /N3/							
	7	6	5	4	3	2	1	0
DBB **115**	Software cams plus /N3/							
	15	14	13	12	11	10	9	8
DBB **116**	Software cams plus /N3/							
	23	22	21	20	19	18	17	16
DBB **117**	Software cams plus /N3/							
	31	30	29	28	27	26	25	24

External digital inputs of the NCK (DB10)

DB10	Signals to NC (PLC -> NCK)							
Byte	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
DBB **122**	Disable the external digital NCK inputs							
	Input 16	Input 15	Input 14	Input 13	Input 12	Input 11	Input 10	Input 9
DBB **123**	Values from the PLC for the external digital NCK inputs							
	Input 16	Input 15	Input 14	Input 13	Input 12	Input 11	Input 10	Input 9
DBB **124**	Disable the external digital NCK inputs							
	Input 24	Input 23	Input 22	Input 21	Input 20	Input 19	Input 18	Input 17
DBB **125**	Values from the PLC for the external digital NCK inputs							
	Input 24	Input 23	Input 22	Input 21	Input 20	Input 19	Input 18	Input 17
DBB **126**	Disable the external digital NCK inputs							
	Input 32	Input 31	Input 30	Input 29	Input 28	Input 27	Input 26	Input 25
DBB **127**	Values from the PLC for the external digital NCK inputs							
	Input 32	Input 31	Input 30	Input 29	Input 28	Input 27	Input 26	Input 25
DBB **128**	Disable the external digital NCK inputs							
	Input 40	Input 39	Input 38	Input 37	Input 36	Input 35	Input 34	Input 33
DBB **129**	Values from the PLC for the external digital NCK inputs							
	Input 40	Input 39	Input 38	Input 37	Input 36	Input 35	Input 34	Input 33

External digital outputs of the NCK (DB10)

DB10	Signals to NC (PLC -> NCK)							
Byte	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
DBB **130**	Disable the external digital NCK outputs							
	Output 16	Output 15	Output 14	Output 13	Output 12	Output 11	Output 10	Output 9
DBB **131**	Overwrite screenform for the external digital NCK outputs							
	Output 16	Output 15	Output 14	Output 13	Output 12	Output 11	Output 10	Output 9
DBB **132**	Value from the PLC for the external digital NCK outputs							
	Output 16	Output 15	Output 14	Output 13	Output 12	Output 11	Output 10	Output 9
DBB **133**	Default screenform for the external digital NCK outputs							
	Output 16	Output 15	Output 14	Output 13	Output 12	Output 11	Output 10	Output 9
DBB **134**	Disable the external digital NCK outputs							
	Output 24	Output 23	Output 22	Output 21	Output 20	Output 19	Output 18	Output 17
DBB **135**	Overwrite screenform for the external digital NCK outputs							
	Output 24	Output 23	Output 22	Output 21	Output 20	Output 19	Output 18	Output 17
DBB **136**	Value from the PLC for the external digital NCK outputs							
	Output 24	Output 23	Output 22	Output 21	Output 20	Output 19	Output 18	Output 17
DBB **137**	Default screenform for the external digital NCK outputs							
	Output 24	Output 23	Output 22	Output 21	Output 20	Output 19	Output 18	Output 17
DBB **138**	Disable the external digital NCK outputs							
	Output 32	Output 31	Output 30	Output 29	Output 28	Output 27	Output 26	Output 25
DBB **139**	Overwrite screenform for the external digital NCK outputs							
	Output 32	Output 31	Output 30	Output 29	Output 28	Output 27	Output 26	Output 25
DBB **140**	Value from the PLC for the external digital NCK outputs							
	Output 32	Output 31	Output 30	Output 29	Output 28	Output 27	Output 26	Output 25
DBB **141**	Default screenform for the external digital NCK outputs							
	Output 32	Output 31	Output 30	Output 29	Output 28	Output 27	Output 26	Output 25
DBB **142**	Disable the external digital NCK outputs							
	Output 40	Output 39	Output 38	Output 37	Output 36	Output 35	Output 34	Output 33
DBB **143**	Overwrite screenform for the external digital NCK outputs							
	Output 40	Output 39	Output 38	Output 37	Output 36	Output 35	Output 34	Output 33
DBB **144**	Value from the PLC for the external digital NCK outputs							
	Output 40	Output 39	Output 38	Output 37	Output 36	Output 35	Output 34	Output 33
DBB **145**	Default screenform for the external digital NCK outputs							
	Output 40	Output 39	Output 38	Output 37	Output 36	Output 35	Output 34	Output 33

Analog inputs of the NCK (external) (DB10)

DB10	Signals to NC (PLC -> NCK)							
Byte	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
DBB **146**	Disable the analog NCK inputs							
	Input 8	Input 7	Input 6	Input 5	Input 4	Input 3	Input 2	Input 1
DBB **147**	Specified analog value for NCK from PLC							
	Input 8	Input 7	Input 6	Input 5	Input 4	Input 3	Input 2	Input 1
DBW **148**	Setpoint from PLC for analog input 1 of NCK							
DBW **150**	Setpoint from PLC for analog input 2 of NCK							
DBW **152**	Setpoint from PLC for analog input 3 of NCK							
DBW **154**	Setpoint from PLC for analog input 4 of NCK							
DBW **156**	Setpoint from PLC for analog input 5 of NCK							
DBW **158**	Setpoint from PLC for analog input 6 of NCK							
DBW **160**	Setpoint from PLC for analog input 7 of NCK							
DBW **162**	Setpoint from PLC for analog input 8 of NCK							
DBB **164,165**	Unassigned							

Analog outputs of the NCK (external) (DB10)

DB10	Signals to NCK (PLC -> NCK)							
Byte	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
DBB **166**	Overwrite screenform for the analog NCK outputs							
	Output 8	Output 7	Output 6	Output 5	Output 4	Output 3	Output 2	Output 1
DBB **167**	Default screenform for the analog NCK outputs							
	Output 8	Output 7	Output 6	Output 5	Output 4	Output 3	Output 2	Output 1
DBB **168**	Disable the analog NCK outputs							
	Output 8	Output 7	Output 6	Output 5	Output 4	Output 3	Output 2	Output 1
DBB **169**	RESERVED							
DBW **170**	Setpoint from PLC for analog output 1 of NCK							
DBW **172**	Setpoint from PLC for analog output 2 of NCK							
DBW **174**	Setpoint from PLC for analog output 3 of NCK							
DBW **176**	Setpoint from PLC for analog output 4 of NCK							
DBW **178**	Setpoint from PLC for analog output 5 of NCK							
DBW **180**	Setpoint from PLC for analog output 6 of NCK							
DBW **182**	Setpoint from PLC for analog output 7 of NCK							
DBW **184**	Setpoint from PLC for analog output 8 of NCK							

Note

Concerning **NCK CPU Ready (DBX 104.7):**

This signal is the sign-of-life monitoring function for the NC. It must be included in the safety circuit of the machine.

Concerning **HMI CPU1 READY (DBX 108.3 and DBX 108.2):**

If the HMI is connected to the operator panel interface (X 101), bit 3 is set (default). When connecting to the PG MPI interface (X 122), bit 2 is set.

External digital input and output signals of the NCK (DB 10)

DB 10	Signals from NCK (NCK→PLC), /A2/							
Byte	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
DBB **186**	Actual value of external digital NCK inputs							
	Input 16	Input 15	Input 14	Input 13	Input 12	Input 11	Input 10	Input 9
DBB **187**	Actual value of external digital NCK inputs							
	Input 24	Input 23	Input 22	Input 21	Input 20	Input 19	Input 18	Input 17
DBB **188**	Actual value of external digital NCK inputs							
	Input 32	Input 31	Input 30	Input 29	Input 28	Input 27	Input 26	Input 25
DBB **189**	Actual value of external digital NCK inputs							
	Input 40	Input 39	Input 38	Input 37	Input 36	Input 35	Input 34	Input 33
DBB **190**	NCK setpoint for external digital NCK outputs							
	Output 16	Output 15	Output 14	Output 13	Output 12	Output 11	Output 10	Output 9
DBB **191**	NCK setpoint for external digital NCK outputs							
	Output 24	Output 23	Output 22	Output 21	Output 20	Output 19	Output 18	Output 17
DBB **192**	NCK setpoint for external digital NCK outputs							
	Output 32	Output 31	Output 30	Output 29	Output 28	Output 27	Output 26	Output 25
DBB **193**	NCK setpoint for external digital NCK outputs							
	Output 40	Output 39	Output 38	Output 37	Output 36	Output 35	Output 34	Output 33

Analog input and output signals of the NCK (DB 10)

DB 10	Signals from NCK (NCK→PLC), /A2/							
Byte	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
DBW **194**	Actual value for analog input 1 of the NCK							
DBW **196**	Actual value for analog input 2 of the NCK							
DBW **198**	Actual value for analog input 3 of the NCK							
DBW **200**	Actual value for analog input 4 of the NCK							
DBW **202**	Actual value for analog input 5 of the NCK							
DBW **204**	Actual value for analog input 6 of the NCK							
DBW **206**	Actual value for analog input 7 of the NCK							
DBW **208**	Actual value for analog input 8 of the NCK							
DBW **210**	Setpoint for analog output 1 of the NCK							
DBW **212**	Setpoint for analog output 2 of the NCK							
DBW **214**	Setpoint for analog output 3 of the NCK							
DBW **216**	Setpoint for analog output 4 of the NCK							
DBW **218**	Setpoint for analog output 5 of the NCK							
DBW **220**	Setpoint for analog output 6 of the NCK							
DBW **222**	Setpoint for analog output 7 of the NCK							
DBW **224**	Setpoint for analog output 8 of the NCK							

2.2.9 Signals from/to mode group (DB 11)

Mode group-specific signals (DB 11)

DB 11	Signals to mode group 1 (PLC→NCK) /K1/							
Byte	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
DBB **0**	Mode group reset	Mode group stop Axes plus spindle	Mode group stop	Mode change disable		Operating mode JOG	MDA	AUTO-MATIC
DBB **1**	Single block Type A	Type B				Machine function REF	REPOS	TEACH IN
DBB **2**	Machine function		var. INC	10000 INC	1000 INC	100 INC	10 INC	1 INC
DBB **3**								

Note

about **machine function**: machine function defined centrally when signal "INC inputs in mode group area active" (DB10.DBX57.0) is set.

DB 11	Signals from mode group 1 (NCK→PLC) /K1/							
Byte	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
DBB **4** HMI--> PLC						Strobe mode JOG	MDA	AUTOM.
DBB **5** HMI--> PLC						Strobe machine function REF	REPOS	TEACH IN
DBB **6**	All channels in reset state		NCK internal JOG active /K1/	Mode Group reseted /K1/	Mode group ready	Active operating mode JOG	MDA	AUTOM.
DBB **7**						Active machine function REF	REPOS	TEACH IN
DBB **8**	Machine function		var. INC	10000 INC	1000 INC	100 INC	10 INC	1 INC

DB 11	Signals to mode group 2 (PLC→NCK) /K1/							
Byte	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
DBB **20**	Mode group reset	Mode group stop Axes plus spindle	Mode group stop	Mode change disable		Operating mode JOG	MDA	AUTO-MATIC
DBB **21**	Single block Type A	Type B				Machine function REF	REPOS	TEACH IN
DBB **22**	Machine function		var. INC	10000 INC	1000 INC	100 INC	10 INC	1 INC
DBB **23**	Unassigned							

Note

about **machine function**: machine function defined centrally when signal "INC inputs in mode group area active" (DB10.DBX57.0) is set.

DB 11	Signals from mode group 2 (NCK→PLC) /K1/							
Byte	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
DBB **24** HMI--> PLC						Strobe mode JOG	MDA	AUTO-MATIC
DBB **25** HMI--> PLC						Strobe machine function REF	REPOS	TEACH IN
DBB **26**	All channels in reset state	NCK internal JOG active	Mode group reseted		Mode group ready	Active operating mode JOG	MDA	AUTO-MATIC
DBB **27**						Active machine function REF	REPOS	TEACH IN
DBB **28**	Machine functions		var. INC	10000 INC	1000 INC	100 INC	10 INC	1 INC

Note

The other mode groups (mode group 3 to mode group 10) are also located in DB 11 with the following initial bytes:

Mode group 3: DBB 40	Mode group 7: DBB 120
Mode group 4: DBB 60	Mode group 8: DBB 140
Mode group 5: DBB 80	Mode group 9: DBB 160
Mode group 6: DBB 100	Mode group 10: DBB 180

2.2.10 Signals for Safety SPL (safe programmable logic) (DB 18)

Parameterization section

References: /FBSI/, SINUMERIK Safety Integrated

DB 18	Signals for Safety SPL (PLC → PLC)							
Byte	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
DBB **0**	not relevant for solution line							
DBB **...**	not relevant for solution line							
DBW **34**	not relevant for solution line							
DBB **36**							Stop E	SPL READY
DBB **37**								

Data area / error

DB 18	Signals for Safety SPL (PLC ←→ NCK)							
Byte	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
	Data area for SPL inputs/outputs							
DBD **38**	SPL_DATA.INSEP [1..32]							
DBD **42**	SPL_DATA.INSEP [33..64]							
DBD **46**	SPL_DATA.OUTSEP [1..32]							
DBD **50**	SPL_DATA.OUTSEP [33..64]							

	Data area for user SPL							
DBD **54**	SPL_DATA.INSIP [1..32]							
DBD **58**	SPL_DATA.INSIP [33..64]							
DBD **62**	SPL_DATA.OUTSIP [1..32]							
DBD **66**	SPL_DATA.OUTSIP [33..64]							
DB 18	**Signals for Safety SPL (PLC ←→ NCK)**							
Byte	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
DBD **70**	SPL_DATA.MARKERSIP [1..32]							
DBD **74**	SPL_DATA.MARKERSIP [33..64]							

	Difference in level between NCK and PLC for diagnostics							
DBD **78**	SPL_DELTA.INSEP [1..32]							
DBD **82**	SPL_DELTA.INSEP [33..64]							
DBD **86**	SPL_DELTA.OUTSEP [1..32]							
DBD **90**	SPL_DELTA.OUTSEP [33..64]							
DBD **94**	SPL_DELTA.INSIP [1..32]							
DBD **98**	SPL_DELTA.INSIP [33..64]							
DBD **102**	SPL_DELTA.OUTSIP [1..32]							
DBD **106**	SPL_DELTA.OUTSIP [33..64]							
DBD **110**	SPL_DELTA.MARKERSIP [1..32]							
DBD **114**	SPL_DELTA.MARKERSIP [33..64]							
DBD **118**								CMDSI
DBD **119**			xxxxxxxx					
DBD **120**	Error number 0 = no error 1 - 320 = Signal number starting from SPL_DATA.INSEP [1]							
DBD **124**	Level indicator of cross-checking (diagnostics option: how many SPL signals currently differ in level)							

Supplementary data areas

DB 18	Signals for Safety SPL (PLC ←→ NCK)							
Byte	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
	Data area for single-channel inputs/outputs							
from NCK DBB **128**	PLCSIOUT [1 .. 8]							
from NCK DBB **129**	PLCSIOUT [9 .. 16]							
from NCK DBB **130**	PLCSIOUT [17 .. 24]							
from NCK DBB **131**	PLCSIOUT [25 .. 32]							
to NCK DBB **132**	PLCSIIN [1.. 8]							
to NCK DBB **133**	PLCSIIN [9 .. 16]							
to NCK DBB **134**	PLCSIIN [17 .. 24]							
to NCK DBB **135**	PLCSIIN [25 .. 32]							
DBB **136**	SPL status							
DBB **138**	PROFIsafe module(s) for 8th input byte	7th input byte	6th input byte	5th input byte	4th input byte	3rd input byte	2nd input byte	1st input byte
DBB **139**								
DBB **140**	PROFIsafe module(s) for 8th output byte	7th output byte	6th output byte	5th output byte	4th output byte	3rd output byte	2nd output byte	1st output byte
DBB **141**								
	reserved							
DBB **142** to DBB **188**								

2.2.11 Signals from/to operator panel (DB 19)

DB 19	Signals to operator panel (PLC→HMI)							
Byte	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
DBB **0**	Actual value in WCS 0=MCS /A2/	Back up travel recorder	HMI Adv shutdown (for OEM users)	Clear recall alarms HMI Adv	Clear cancel alarms HMI Adv	Key disable /A2/	Screen darkening /A2/	Screen bright /A2/
DBB **1**	Reserved							
DBW **2**								
DBW **4**								
DBB **6**	Analog spindle 1, capacity in percent							
DBB **7**	Analog spindle 2, capacity in percent							
DBB **8**	Channel number of machine control panel to HMI							
DBB **9**	Reserved for selection					Automatc tool measure-ment	OEM2	OEM1
DBB **10**	PLC Hardkeys (Values 1 ... 255, Default: 0)							
DBB **11**	Reserved for hardkey function expansions							
DBB **12**								
DBB **13**	Select /A2/	Load part program /A2/	Unload /A2/	Res.				Disable Teach transfer
DBB **14**	0=act. FS 1=pas. FS	RS-232 act. FS: Index of file to be transferred in the standard list. RS-232 pass. FS: Number of the control file for user file names.						
DBB **15**	RS-232 act. FS: Index that specifies the axis, channel or tool no. RS-232 pass. FS: Index of the file to be transferred in the user list							
DBB **16**	1=pas FS	Part program handling: Number of the control file for user file names.						
DBB **17**	Part program handling: Index of the file to be transferred in the user list							
DBB **18**								
DBB **19**	Reserved (signal counter), HMI --> PLC							

DB 19	Signals from operator panel (HMI → PLC)							
Byte	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
DBB **20**	MCS/WCS Change-over /A2/	Simulation active /A2/	Language 2 switched HMI Emb.	Recall alarm cleared HMI Adv /A2/	Cancel alarm cleared HMI Adv /A2/	Cancel key actuated /A2/	Screen is dark /A2/	
DBB **21**	Active HMI operating area							
DBB **22**	Displayed channel number from the HMI /A2/							
DBB **23**								
DBW **24**	actual mask number from JobShop							
DBB **26**	Part program handling status /A2/							
	Select	Load	Unload		Active	Error	OK	Error
DBB **27**	Error program handling /A2/							
DBW **28**	Mask number for "Extend user interface" /IAM/, BE1							
DBB **30**	Control bits PLC --> HMI							
							Exit mask	Request mask
DBB **31**	Control bits PLC --> HMI							
	Inactive bit			Error, Not possible to request mask	Mask exited	Mask active	Mask requested	Mask request accepted
DBB **32** PLC>HMI			FunctionSelectionNo. from PLC					
	Busy function	Strobe function						
DBB **33** PLC>HMI	Parameter 1 for FunctionSelectionNo. (function selection from DBB 32)							
DBB **34** PLC>HMI	Parameter 2 for FunctionSelectionNo. (function selection from DBB 32)							

DBB **35** PLC>HMI	Parameter 3 for FunctionSelectionNo. (function selection from DBB 32)							
DBB **36** HMI>PLC	Error code for FunctionSelectionNo. (function selection from DBB 32)							
DBB **37** HMI>PLC	Parameter 1 for FunctionSelectionNo. (function selection from DBB 48)							
DBB **38** HMI>PLC	Parameter 2 for FunctionSelectionNo. (function selection from DBB 48)							
DBB **39** HMI>PLC	Parameter 3 for FunctionSelectionNo. (function selection from DBB 48)							

DBB40-47 used internally

DBB **48** HMI>PLC	PLC-Busy function	HMI-Strobe function	FunctionSelectionNo. from HMI					
DBB **49** PLC>HMI	Error code for FunctionSelectionNo. (function selection from DBB48)							

Interface 2nd HMI								
DBB **50-99**	Assignment as for DBB 0 to DBB 49 Switchover interface to HMI							
	Knocking interface (HMI announces itself to NCU)							
DBW **100**	ONL_REQUEST /B3/ Online request from HMI HMI writes its client identification as online request (bit 8-15: bus type, bit 0-7: HMI bus address)							
DBW **102**	ONL_CONFIRM /B3/ Acknowledgment from PLC to online request PLC writes HMI client identification as acknowledgment (bus type, HMI bus address; as with DBW 100).							
DBW **104**	PAR_CLIENT_IDENT /B3/ HMI writes its client identification (bus type, HMI bus address; as with DBW 100).							
DBB **106**	PAR_HMI_TYP /B3/ Type of HMI as per NETNAMES.INI: Main / subordinate operator panel / server /...							
DBB **107**	PAR_MSTT_ADR /B3/ HMI writes address of MCP to be activated; 255, when no MCP activated							
DBB **108**	PAR_STATUS /B3/ PLC writes online enable for HMI.							

DBB **109**	PAR_Z_INFO /B3/ PLC writes additional info about status							
DBW **110**	M_TO_N_ALIVE Sign of life from PLC to HMI through M to N block							
DBB **112**	reserved Bustype MCP							
DBB **118**	TCU index 1st online interface							
DBB **119**	TCU index 2 nd online interface							
	Online interface HMI 1 (user)							
DBW **120**	MMC1_CLIENT_IDENT /B3/ PLC writes PAR_CLIENT_IDENT to MMCx_CLIENT_IDENT when HMI goes online.							
DBB **122**	MMC1_TYP /B3/ PLC writes PAR_MMC_TYP to MMCx_TYP when HMI goes online.							
DBB **123**	MMC1_MSTT_ADR /B3/ PLC writes PAR_MSTT_ADR to MMCx_MSTT_ADR when HMI goes online.							
DBB **124**	MMC1_STATUS /B3/ Connection status, HMI and PLC alternately write their requests/acknowledgments							
DBB **125**	MMC1_Z_INFO /B3/ Additional info connection status (pos./neg. acknowledgment, error messages...)							
DBB **126**	res.	TCU1_ SHIFT_ LOCK	MMC1_ CHANGE_ DENIED /B3/	MMC1_ ACTIVE_ CHANGED /B3/	MMC1_ ACTIVE_ PERM /B3/	MMC1_ ACTIVE_ REQ /B3/	MMC1_ MSTT_ SHIFT_ LOCK /B3/	MMC1 SHIFT LOCK /B3/
DBB **127**	Reserved Bustype MCP							
DBB **128**	Reserved Transline (Transline DB number)							

Online Interface HMI 2 (user)								
DBW **130**	MMC2_CLIENT_IDENT /B3/ PLC writes PAR_CLIENT_IDENT to MMCx_CLIENT_IDENT when HMI goes online.							
DBB **132**	MMC2_TYP /B3/ PLC writes PAR_MMC_TYP to MMCx_TYP when HMI goes online.							
DBB **133**	MMC2_MSTT_ADR /B3/ PLC writes PAR_MSTT_ADR to MMCx_MSTT_ADR when HMI goes online.							
DBB **134**	MMC2_STATUS /B3/ Connection status, MMC and PLC alternately write their requests/acknowledgments							
DBB **135**	MMC2_Z_INFO /B3/ Additional info connection status (pos./neg. acknowledgment, error messages...)							
DBB **136**	res.	TCU2_ SHIFT_ LOCK	MMC2_ CHANGE_ DENIED /B3/	MMC2_ ACTIVE_ CHANGED /B3/	MMC2_ ACTIVE_ PERM /B3/	MMC2_ ACTIVE_ REQ /B3/	MMC2_ MSTT_ SHIFT_ LOCK /B3/	MMC2_ SHIFT_ LOCK /B3/
DBB **137**	Reserved Bustype MCP							
DBB **138**	Reserved Transline (Transline DB number)							
DBB **140-197**	Code carrier input parameters Optional package SINTDC on HMI-Advanced required							
DBB **198-249**	Code carrier return parameters Optional package SINTDC on HMI-Advanced required							
DBB **250-255**	Commands Optional package SINTDC on HMI-Advanced required							
DBB **256-267**	Commands for Paramtm.exe Optional package SINTDC on HMI-Advanced required							
DBW 268	Traffic light status Optional package TPM on HMI Advanced required							
DBW 270 to 394	Counter[1 ... 32] Optional package TPM on HMI Advanced required							

2.2.12 PLC machine data (DB 20)

DB 20	PLC machine data (PLC→operator)							
Byte	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
DBW 0	INT values							
DBW								
DBW	INT values							
DBB	Bit arrays							
DBB								
DBB	Bit arrays							
DBD	REAL values							
DBD								
DBD	REAL values							

Note

The initial and end addresses of the PLC machine data areas depend on the respective length indications of the partial areas. In general, the integer values start with the data byte 0. The upper limit is determined by the corresponding length indication. In general, the following bit arrays (2-decade hexadecimal numbers on input) start with the following even address. The real values follow directly the bit arrays and also start with an even address.

2.2.13 Signals from/to NCK channel (DB 21–30)

DB 21 - 30	Signals to NCK channel (PLC→NCK)							
Byte	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
DBB **0**		Activate dry run feedrate /V1/	Activate M01 /K1/	Activate single block /K1/	Activate DRF /H1/			
DBB **1**	Activate program test /K1/	PLC action complete /K1/	CLC override /TE1/	CLC stop /TE1/	Time monitoring act. (tool manage-ment)	Synchron-ized action OFF /FBSY/	Enable protection zones /A3/	Activate referencing /R1/
DBB **2**	Skip block /K1/							
	/7	/6	/5	/4	/3	/2	/1	/0
DBB **3**	Nibbling and punching /N4/							
			Manual release of stroke 2	Stroke delayed /N4/	Stroke not operating /N4/	Stroke suppres-sion /N4/	Manual stroke enable /N4/	Stroke enable /N4/
DBB **4**	Feedrate override /V1/							
	H	G	F	E	D	C	B	A
DBB **5**	Rapid traverse override /V1/							
	H	G	F	E	D	C	B	A
DBB **6**	Feedrate override active /V1/	Rapid traverse override active /V1/		Program level abort /K1/	Delete subroutine no. of passes	Delete distance-to-go /A2/	Read-in disable /K1/	Feed disable /V1/
DBB **7**	Reset /K1/		Suppress Start Lock	NC Stop axes plus spindle /K1/	NC Stop /K1/	NC Stop to block limit /K1/	NC Start /K1/	NC Start disable /K1/
DBB **8**	Activate machine-related protection area /A3/							
	Area 8	Area 7	Area 6	Area 5	Area 4	Area 3	Area 2	Area 1
DBB **9**	Activate machine-related protection area /A3/							
							Area 10	Area 9
DBB **10**	Activate channel-specific protection area /A3/							
	Area 8	Area 7	Area 6	Area 5	Area 4	Area 3	Area 2	Area 1
	Activate channel-specific protection area /A3/							
DBB **11**							Area 10	Area 9

Note

on **Feedrate override active (DBX6.7)**
even if feedrate override is not active (= 100%), the setting 0% is effective.
on **Feedrate override (DBB 4)**
either 31 positions (Gray code) with 31 MD for % evaluation or 0-200% corresponding to the dual value in byte (201–255 ⇒ max. 200%).
on **Rapid traverse override (DBB 5)**
either 31 positions (Gray code) with 31 MD for % evaluation or 0-100% corresponding to the dual value in byte (101–255 ⇒ max. 100%).
on **Activate single block (DBX0.4)**
select variant via ”Write variable”.
on **Delete distance-to-go (DBX6.2)**
effects only path axes and not positioning axes

Control signals to geometry axes

DBB **12**	Geometry axis 1							
	Traversing keys /H1/		Rapid traverse override /H1/	Traversing key disable /H1/	Feed stop /V1/	Activate handwheel /H1/		
	+	-				3	2	1
DBB **13**	Geometry axis 1 machine function /H1/							
			Var. INC	10000 INC	1000 INC	100 INC	10 INC	1 INC
DBB **14**	OEM signals geometry axis 1							
DBB **15**	Geometry axis 1							
DBB **16**	Geometry axis 2							
	Traversing keys /H1/		Rapid traverse override /H1/	Traversing key disable /H1/	Feed stop /V1/	Activate handwheel /H1/		
	+	-				3	2	1
DBB **17**	Geometry axis 2 machine function /H1/							
			Var. INC	10000 INC	1000 INC	100 INC	10 INC	1 INC
DBB **18**	OEM signals geometry axis 2							
DBB **19**	Geometry axis 2							
DBB **20**	Geometry axis 3							
	Traversing keys /H1/		Rapid traverse override /H1/	Traversing key disable /H1/	Feed stop /V1/	Activate handwheel /H1/		
	+	-				3	2	1
DBB **21**	Geometry axis 3 machine function /H1/							
			Var. INC	10000 INC	1000 INC	100 INC	10 INC	1 INC

DBB **22**	OEM signals geometry axis 3							
DBB **23**	Geometry axis 3							

Note

about **machine function**: machine function only defined when signal "INC inputs in mode group area active" (DB10.DBX57.0) is not set.

Operating signals from HMI/status signals from NC channel

DB 21-30	**Signals from NCK channel (NCK→PLC, HMI→PLC, PLC→NCK)**							
Byte	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
DBB **24** HMI--> PLC		Dry run feedrate selected /V1/	M01 selected /K1/	Select NCK-related M01	DRF selected /H1/			
DBB **25** HMI--> PLC	Program test selected /K1/			REPOS MODE EDGE	Feedrate override for rapid traverse selected /V1/	REPOSPATHMODE 2	1	0
DBB **26** HMI--> PLC	Skip block selected /K1/ 7	6	5	4	3	2	1	0
DBB **27** HMI--> PLC							Skip block selected /K1/	Skip block selected /K1/
DBB **28** PLC→NCK	OEM channel signals							
DBB **29** PLC→NCK	Do not disable tool	Switch off wear monitoring	Switch off workpiece counter	Activate PTP motion	Activate fixed feed 4 /FBMA/, /V1/	Activate fixed feed 3 /FBMA/, /V1/	Activate fixed feed 2 /FBMA/, /V1/	Activate fixed feed 1 /FBMA/, /V1/

DBB **30** PLC→NCK						Activate contour handwheel		
	No tool change commands		Activate NCK-related M01 /H2/	Neg. direction simulation contour handwheel	Simulation contour handwheel on	Handwheel 3	Handwheel 2	Handwheel 1
DBB **31** PLC→NCK						REPOSPATHMODE		
	Skip block active /9	Skip block active /8		REPOS MODE EDGE		2	1	0
DBB **32** NCK--> PLC		Last action block active /K1/	M00/M01 active /K1/	Approach block active /K1/	Action block active /K1/			Execution from external source active
DBB **33** NCK--> PLC	Program test active /K1/	Transforma tion active /K1/M1	M02/M30 active /K1/	Block search active /K1/	Handwheel override active /H1/	Revolutio-nal feedrate active /V1/	Orientable toolholder active	Referencing active /R1/
DBB **34** NCK--> PLC	OEM channel signals feedback							
DBB **35** NCK--> PLC	Channel status /K1/			Program status /K1/				
	Reset	Interrupted	Active	Aborted	Interrupted	Stopped	Waiting	Running
DBB **36** NCK--> PLC	NCK alarm with processing stop present /A2/	Channel-specific NCK alarm present /A2/	Channel ready for operation	Interrupt processing active /K1/	All axes stationary /B1/	All axes requiring reference points are referenced /R1/		
DBB **37** NCK--> PLC	Stop	Read-in	CLC	CLC	CLC	Contour handwheel active		
	at block end with SBL is sup-pressed	enable is ignored	stopped upper limit /TE1/	stopped lower limit /TE1/	active /TE1/	Handwheel 3 /H1/	Handwheel 2 /H1/	Handwheel 1 /H1/
DBB **38** NCK--> PLC	Nibbling and punching /N4/							
							Acknowl. manual stroke enable /N4/	Stroke enable active /N4/
DBB **39** NCK--> PLC								Protection zones not guaranteed

Note

on **Feedrate override for rapid traverse selected (DBX25.3)**

Depending on this signal, the basic PLC program copies the feedrate override onto the rapid traverse override on the channel-specific interface.

On **Program test selected (DBX25.7)**

”Program test selected” means axis disable for all channel axes and spindles.

Status signals of geometry axes

DB 21 - 30	Signals from NCK channel (NCK→PLC)							
Byte	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
DBB **40**	Geometry axis 1							
	Traverse command /H1/		Travel requests			Handwheel active /H1/		
	plus	minus	plus	minus		3	2	1
DBB **41**	Geometry axis 1 active machine function /H1/							
			Var. INC	10000 INC	1000 INC	100 INC	10 INC	1 INC
DBB **42**	OEM signals geometry axis 1							
DBB **43**	Geometry axis 1							
DBB **44** HMI--> PLC								
DBB **46**	Geometry axis 2							
	Traverse command /H1/		Travel requests			Handwheel active /H1/		
	plus	minus	plus	minus		3	2	1
DBB **47**	Geometry axis 2 active machine function /H1/							
			Var. INC	10000 INC	1000 INC	100 INC	10 INC	1 INC
DBB **48**	OEM signals geometry axis 2							
DBB **49**	Geometry axis 2							
DBB **50** HMI--> PLC								
DBB **52**	Geometry axis 3							
	Traverse command /H1/		Travel requests			Handwheel active /H1/		
	plus	minus	plus	minus		3	2	1
DBB **53**	Geometry axis 3 active machine function /H1/							
			Var. INC	10000 INC	1000 INC	100 INC	10 INC	1 INC

DBB 54	OEM signals geometry axis 3							
DBB 55	Geometry axis 3							
DBB 56 HMI--> PLC								
DBB 57								

Change signals on auxiliary function transfer from NC channel

DB 21 - 30	Signals from NCK channel (NCK→PLC)							
Byte	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
DBB 58				M fct. 5 change /H2/	M fct. 4 change /H2/	M fct. 3 change /H2/	M fct. 2 change /H2/	M fct. 1 change /H2/
DBB 59				M fct. 5 not decoded	M fct. 4 not decoded	M fct. 3 not decoded	M fct. 2 not decoded	M fct. 1 not decoded
DBB 60		S fct. 3 quick	S fct. 2 quick	S fct. 1 quick		S fct. 3 change /H2/	S fct. 2 change /H2/	S fct. 1 change /H2/
DBB 61		T fct 3 quick	T fct. 2 quick	T fct. 1 quick		T fct. 3 change/H2/	T fct. 2 change/H2/	T fct. 1 change /H2/
DBB 62		D fct. 3 quick	D fct. 2 quick	D fct. 1 quick		D fct. 3 change/H2/	D fct. 2 change/H2/	D fct. 1 change /H2/
DBB 63				DL fct. quick				DL fct. change
DBB 64		H fct. 3 quick	H fct. 2 quick	H fct. 1 quick		H fct. 3 change /H2/	H fct. 2 change /H2/	H fct. 1 change /H2/
DBB 65			F fct. 6 change /H2/	F fct. 5 change /H2/	F fct. 4 change /H2/	F fct. 3 change /H2/	F fct. 2 change /H2/	F fct. 1 change /H2/
DBB 66				M fct. 5 quick	M fct. 4 quick	M fct. 3 quick	M fct. 2 quick	M fct. 1 quick
DBB 67			F fct. 6 quick	F fct. 5 quick	F fct. 4 quick	F fct. 3 quick	F fct. 2 quick	F fct. 1 quick

Note

For 10-decade T numbers, only the T fct. 1 change signal is available. For 5-decade D numbers, only the D fct. 1 change signal is available.

Transferred M/S functions

DB 21 - 30	Signals from NCK channel (NCK→PLC)							
Byte	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
DBW **68**	Extended address M function 1 (binary) /H2/							
DBD **70**	M function 1 (binary) /H2/							
DBW **74**	Extended address M function 2 (binary) /H2/							
DBD **76**	M function 2 (binary) /H2/							
DBW **80**	Extended address M function 3 (binary) /H2/							
DBD **82**	M function 3 (binary) /H2/							
DBW **86**	Extended address M function 4 (binary) /H2/							
DBD **88**	M function 4 (binary) /H2/							
DBW **92**	Extended address M function 5 (binary) /H2/							
DBD **94**	M function 5 (binary) /H2/							
DBW **98**	Extended address S function 1 (binary) /H2/							
DBD **100**	S function 1 (REAL format) /H2/							
DBW **104**	Extended address S function 2 (binary) /H2/							
DBD **106**	S function 2 (REAL format) /H2/							
DBW **110**	Extended address S function 3 (binary) /H2/							
DBD **112**	S function 3 (REAL format) /H2/							

Note

M functions are programmed in the part program in the INTEGER format (8 decades plus sign).

”REAL format” means: 24 bit mantissa and 8 bit exponent.

Transferred T/D/DL functions

DB 21 - 30	Signals from NCK channel (NCK→PLC)							
Byte	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
DBW **116**	Extended address T function 1 (16 bit Int)							
DBW **118** DBD **118**	T function 1 (binary) /H2/ For 8-decade T nos., T function 1 (32 bit DINT) is used in DBD 118 (see note)							
DBW **120**	Extended address T function 2 (16 bit Int)							
DBW **122**	T function 2 (Int)							
DBW **124**	Extended address T function 3 (16 bit Int)							
DBW **126**	T function 3 (Int)							
DBB **128**								
DBB **129**	D function 1 (binary) /H2/							
DBW **130** DBB **130**	For 5-decade D nos., D function 1 (16 bit DINT) is used in DBD 130 (see note) Extended address D function 2 (8 bit Int)							
DBB **131**	D function 2 (8 bit Int)							
DBB **132**	Extended address D function 3 (8 bit Int)							
DBB **133**	D function 3 (8 bit Int)							
DBW **134**	Extended address DL function (16 bit Int)							
DBD **136**	DL function (REAL)							

Note

With active tool management, programmed T functions are **not** output to the PLC.

8-decade T nos. are only available as T function 1

Programmed D functions with names (e.g. D=CUTEDGE_1) **cannot** be output in ASCII format to the PLC.

5-decade D nos. are only available as D function 1

The REAL format corresponds to floating point representation in STEP 7 (24 bit mantissa and 8 bit exponent). This floating point format supplies a maximum of 7 valid places.

Transferred H/F functions

DB 21 - 30	Signals from NCK channel (NCK→PLC)							
Byte	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
DBW **140**	Extended address H function 1 (binary) /H2/							
DBD **142**	H function 1 (REAL or Dint) /H2/							
DBW **146**	Extended address H function 2 (binary) /H2/							
DBD **148**	H function 2 (REAL or Dint) /H2/							
DBW **152**	Extended address H function 3 (binary) /H2/							
DBD **154**	H function 3 (REAL or Dint) /H2/							
DBW **158**	Extended address F function 1 (binary) /H2/							
DBD **160**	F function 1 (REAL format) /H2/							
DBW **164**	Extended address F function 2 (binary) /H2/							
DBD **166**	F function 2 (REAL format) /H2/							
DBW **170**	Extended address F function 3 (binary) /H2/							
DBD **172**	F function 3 (REAL format) /H2/							
DBW **176**	Extended address F function 4 (binary) /H2/							
DBD **178**	F function 4 (REAL format) /H2/							
DBW **182**	Extended address F function 5 (binary) /H2/							
DBD **184**	F function 5 (REAL format) /H2/							
DBW **188**	Extended address F function 6 (binary) /H2/							
DBD **190**	F function 6 (REAL format) /H2/							

Note

F functions are programmed in the part program in the REAL format.

The extended address of the F function contains an identifier with the following meaning:
0 = path feed,
1-31 = machine axis number for feed with positioning axes.

The H function data type is dependent on MD 22110: AUXFU_H_TYPE_INT.

Decoded M signals (M0–M99)

DB 21 - 30	Signals from NCK channel (NCK→PLC)							
Byte	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
DBB **194**	Dynamic M functions							
	M07	M06	M05 *	M04 *	M03 *	M02	M01	M00
DBB **195**	Dynamic M functions /H2/							
	M15	M14	M13	M12	M11	M10	M09	M08
DBB **196**	Dynamic M functions /H2/							
	M23	M22	M21	M20	M19	M18	M17	M16
DBB **197**	Dynamic M functions /H2/							
	M31	M30	M29	M28	M27	M26	M25	M24
DBB **198**	Dynamic M functions /H2/							
	M39	M38	M37	M36	M35	M34	M33	M32
DBB **199**	Dynamic M functions /H2/							
	M47	M46	M45	M44	M43	M42	M41	M40
DBB **200**	Dynamic M functions /H2/							
	M55	M54	M53	M52	M51	M50	M49	M48
DBB **201**	Dynamic M functions /H2/							
	M63	M62	M61	M60	M59	M58	M57	M56
DBB **202**	Dynamic M functions /H2/							
	M71	M70 *	M69	M68	M67	M66	M65	M64
DBB **203**	Dynamic M functions /H2/							
	M79	M78	M77	M76	M75	M74	M73	M72
DBB **204**	Dynamic M functions /H2/							
	M87	M86	M85	M84	M83	M82	M81	M80
DBB **205**	Dynamic M functions /H2/							
	M95	M94	M93	M92	M91	M90	M89	M88
DBB **206**	Dynamic M functions /H2/							
					M99	M98	M97	M96
DBB **207**								

Note

M functions marked with * are not decoded in this bit array if a spindle is configured in the channel. In this case, these M functions are offered as extended M functions in DB21-30.DBB68 ff. and in the relevant axis DB DB31-61.DBB86 ff.

Dynamic M functions (M00 to M99) are decoded by the basic PLC program.

The PLC user must use dynamic M functions in order to generate static M functions.

Active G functions

DB 21 - 30	Signals from NCK channel (NCK→PLC)							
Byte	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
DBB **208**	Number of active G function of G function group 1 (binary) /K1/							
DBB **209**	Number of active G function of G function group 2 (binary) /K1/							
DBB **210**	Number of active G function of G function group 3 (binary) /K1/							
DBB **211**	Number of active G function of G function group 4 (binary) /K1/							
DBB **212**	Number of active G function of G function group 5 (binary) /K1/							
DBB **213**	Number of active G function of G function group 6 (binary) /K1/							
DBB **214**	Number of active G function of G function group 7 (binary) /K1/							
DBB **215**	Number of active G function of G function group 8 (binary) /K1/							
...								
DBB **270**	Number of active G function of G function group n-1 (binary) /K1/							
DBB **271**	Number of active G function of G function group n (binary) /K1/							

Note

The active G functions of the groups are updated each time a G function or a mnemonic identifier (e.g. SPLINE) is programmed.

G functions within a G group are output as binary value, starting with 1.

A G function with the value 0 means that no G function is active for this G group.

Signals for protection areas from NC channel

DB 21 - 30	Signals from NCK channel (NCK→PLC)							
Byte	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
DBB **272**	Machine-related protection area preactivated /A3/							
	Area 8	Area 7	Area 6	Area 5	Area 4	Area 3	Area 2	Area 1
DBB **273**	Machine-related protection area preactivated /A3/							
							Area 10	Area 9
DBB **274**	Channel-specific protection area preactivated /A3/							
	Area 8	Area 7	Area 6	Area 5	Area 4	Area 3	Area 2	Area 1
DBB **275**	Channel-specific protection area preactivated /A3/							
							Area 10	Area 9
DBB **276**	Machine-related protection area violated /A3/							
	Area 8	Area 7	Area 6	Area 5	Area 4	Area 3	Area 2	Area 1
DBB **277**	Machine-related protection area violated /A3/							
							Area 10	Area 9
DBB **278**	Channel-related protection area violated /A3/							
	Area 8	Area 7	Area 6	Area 5	Area 4	Area 3	Area 2	Area 1
DBB **279**	Channel-related protection area violated /A3/							
							Area 10	Area 9

Instruction-controlled signals to NC channel

DB 21 - 30	Signals to NCK channel (NCK→PLC)							
Byte	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
DBB **280**							Synch. action disable request	Reserved
DBB **281**							Synch. action disabled	
DBW **282**	Reserved							
DBW **284**	Reserved							
DBW **286**	Reserved							
DBW **288**	Reserved							
DBW **290**	Reserved							
DBW **292**	Reserved							
DBW **294**	Reserved							
DBW **296**	Reserved							
DBW **298**	Reserved							
DBB **300**	Disable synchronized actions /FBSY/							
	No. 8	No. 7	No. 6	No. 5	No. 4	No. 3	No. 2	No. 1
DBB **301**	Disable synchronized actions /FBSY/							
	No. 16	No. 15	No. 14	No. 13	No. 12	No. 11	No. 10	No. 9
DBB **302**	Disable synchronized actions /FBSY/							
	No. 24	No. 23	No. 22	No. 21	No. 20	No. 19	No. 18	No. 17
DBB **303**	Disable synchronized actions /FBSY/							
	No. 32	No. 31	No. 30	No. 29	No. 28	No. 27	No. 26	No. 25
DBB **304**	Disable synchronized actions /FBSY/							
	No. 40	No. 39	No. 38	No. 37	No. 36	No. 35	No. 34	No. 33
DBB **305**	Disable synchronized actions /FBSY/							
	No. 48	No. 47	No. 46	No. 45	No. 44	No. 43	No. 42	No. 41
DBB **306**	Disable synchronized actions /FBSY/							
	No. 56	No. 55	No. 54	No. 53	No. 52	No. 51	No. 50	No. 49
DBB **307**	Disable synchronized actions /FBSY/							
	No. 64	No. 63	No. 62	No. 61	No. 60	No. 59	No. 58	No. 57

Note

The request signals are set by the user and reset by the basic program after transmission of the corresponding data.

Instruction-controlled signals from NC channel

DB 21 - 30	Signals from NCK channel (NCK→PLC)							
Byte	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
DBB **308**	Synchronized actions can be disabled /FBSY/							
	Nr. 8	Nr. 7	Nr. 6	Nr. 5	Nr. 4	Nr. 3	Nr. 2	Nr.1
DBB **309**	Synchronized actions can be disabled /FBSY/							
	Nr. 16	Nr. 15	Nr. 14	Nr. 13	Nr. 12	Nr. 11	Nr.10	Nr.9
DBB **310**	Synchronized actions can be disabled /FBSY/							
	Nr. 24	Nr. 23	Nr. 22	Nr. 21	Nr. 20	Nr. 19	Nr.18	Nr.17
DBB **311**	Synchronized actions can be disabled /FBSY/							
	Nr. 32	Nr. 31	Nr. 30	Nr. 29	Nr. 28	Nr. 27	Nr. 26	Nr.25
DBB **312**	Synchronized actions can be disabled /FBSY/							
	Nr. 40	Nr. 39	Nr. 38	Nr. 37	Nr. 36	Nr. 35	Nr. 34	Nr. 33
DBB **313**	Synchronized actions can be disabled /FBSY/							
	Nr. 48	Nr. 47	Nr. 46	Nr. 45	Nr. 44	Nr. 43	Nr. 42	Nr.41
DBB **314**	Synchronized actions can be disabled /FBSY/							
	Nr. 56	Nr. 55	Nr. 54	Nr. 53	Nr. 52	Nr. 51	Nr. 50	Nr.49
DBB **315**	Synchronized actions can be disabled /FBSY/							
	Nr. 64	Nr. 63	Nr. 62	Nr. 61	Nr. 60	Nr. 59	Nr. 58	Nr.57
Cyclic Signals interface NCK → PLC								
DBB **316**	Active G functions							
							G0 PATH	G00 geo.
DBB **317**	Tool missing	PTP motion active	Travel request drive test				Workpiece setpoint reached	External language mode active
DBB **318**	Overstore active	Dry-run feedrate active /V1/	Associated M01 active /H2/	Stop delayed	TOFF movement active	TOFF active	Search active /K1/	ASUP stopped /K1/
DBB **319**	No tool change command active	Stop-delay-range not activated	Repos DEFERRA L Chan	Delay FTS	Repos Path Mode Ackn 2	Repos Path Mode Ackn 1	Repos Path Mode Ackn 0	REPOS MODE EDGE ACKN

Signals to orientation axes

DB 21 - 30	Signals to NCK channel (PLC→NCK)							
Byte	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
DBB **320**	Traversing keys		**Orientation axis 1**					
	+ –	– –	Rapid traverse override	Traversing key disable	Feed stop	Activate handwheel (bit value coding)		
DBB **321**	**Orientation axis 1**							
			var. INC	10000 INC	1000 INC	100 INC	10 INC	1 INC
DBB **322**	**OEM signals orientation axis 1**							
DBB **323**	**Orientation axis 1**							
			var. INC	10000 INC	1000 INC	100 INC	10 INC	1 INC
DBB **324**	Traversing keys		**Orientation axis 2**					
	+	–	Rapid traverse override	Traversing key disable	Feed stop	Activate handwheel (bit value coding)		
DBB **325**	**Orientation axis 2**							
			var. INC	10000 INC	1000 INC	100 INC	10 INC	1 INC
DBB **326**	**OEM signals orientation axis 2**							
DBB **327**	**Orientation axis 2**							
DBB **328**	Traversing keys		**Orientation axis 3**					
	+	–	Rapid traverse override	Traversing key disable	Feed stop	Activate handwheel (bit value coding)		
DBB **329**	**Orientation axis 3**							
DBB **330**	**OEM signals orientation axis 3**							
DBB **331**	**Orientation axis 3**							

Signals from orientation axes

DB 21 - 30	Signals from NCK channel (NCK→PLC)							
Byte	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
DBB **332**	**Orientation axis 1**							
	Travel command		Travel request			Handwheel active (bit value coding)		
	plus	minus	plus	minus				
DBB **333**	**Orientation axis 1** Active machine function							
			var. INC	10000 INC	1000 INC	100 INC	10 INC	1 INC
DBB **334**	**OEM signals orientation axis 1**							
DBB **335**	**Orientation axis 1**							
DBB **336**	**Orientation axis 2**							
	Travel command		Travel request			Handwheel active (bit value coding)		
	plus	minus	plus	minus				
DBB **337**	**Orientation axis 2** Active machine function							
			var. INC	10000 INC	1000 INC	100 INC	10 INC	1 INC
DBB **338**	**OEM signals orientation axis 2**							
DBB **339**	**Orientation axis 2**							
DBB **340**	**Orientation axis 3**							
	Travel command		Travel request			Handwheel active (bit value coding)		
	plus	minus	plus	minus				
DBB **341**	**Orientation axis 3** Active machine function							
			var. INC	10000 INC	1000 INC	100 INC	10 INC	1 INC
DBB **342**	**OEM signals orientation axis 3**							
DBB **343**	**Orientation axis 3**							

Tool management functions from NC channel

DB 21 - 30	Signals from NCK channel (NCK→PLC)							
Byte	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Modification signals tool management functions								
DBB **344**					Last replace-ment tool of tool group	Transfer to new replace-ment tool	Tool limit value reached	Tool pre-warning limit reached
DBB **345-347**								
Transferred tool management functions								
DBD **348**	T number for tool prewarning limit (DInt)							
DBD **352**	T number for tool limit value (DInt)							
DBD **356**	T number of new replacement tool (DInt)							
DBD **360**	T number of last replacement tool (DInt)							

Signals from/to NC channel

DB 21 - 30	Signals from NCK channel (NCK→PLC, PLC →NCK)							
Byte	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
	CH_CYCLES_SIG_IN (Bit 0 – 7)							
DBB **364**								
	CH_CYCLES_SIG_IN (Bit 8 – 15)							
DBB **365**								
	CH_CYCLES_SIG_OUT (Bit 0 – 7)							
DBB **366**								
	CH_CYCLES_SIG_OUT (Bit 8 – 15)							
DBB **367**								
	CH_OEM_TECHNO_SIG_IN (DBB368 - 371)							
DBB **368**								
DBB **369**								
DBB **370**								
DBB **371**								
DBB **372**	CH_OEM_TECHNO_SIG_OUT (DBB372 - 375)							
DBB **373**								
DBB **374**								
DBB **375**								
DBB **376**	ProgEventDisplay							
DBB **377**								
DBB **378**								
DBB **379**								
DBB **380**								
DBB **381**								
DBB **382**								
DBB **383**								

2.2.14 Signals from/to axis/spindle (PLC→NCK) (DB 31–DB 61)

DB 31 -61	Signals to axis/spindle (PLC→NCK)							
Byte	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
DBB **0** Axis and spindle	Feedrate override /V1/							
	H	G	F	E	D	C	B	A
DBB **1** Axis and spindle	Override active /V1/	Position measuring system 2 /A2/	Position measuring system 1 /A2/	Follow-up mode /A2/	Axis/spindle disable /A2/	Sensor fixed stop /F1/	Acknowl. fixed stop reached /F1/)	Drive test movement enable
DBB **2** Axis and spindle	Reference point value /R1/				Clamping in progress /A3/	Delete distance-to-go/ spindle reset /A2, S1/	Controller enable /A2/	Cam activation /N3/
	4	3	2	1				
DBB **3** Axis and spindle	Program test axis/ spindle release	Velocity/ spindle speed limitation /A3/	Activate fixed feed 4 /FBMA/, /V1/	Activate fixed feed 3 /FBMA/, /V1/	Activate fixed feed 2 /FBMA/, /V1/	Activate fixed feed 1 /FBMA/, /V1/	Enable travel to fixed stop /F1/	Accept external ZO /K2/
DBB **4** Axis and spindle	Traversing keys /H1/		Rapid traverse override /H1/	Traversing key disable /H1/	Feed stop/spindle stop /A2/	Activate handwheel /H1/		
	plus	minus				3	2	1
DBB **5** Axis and spindle	Machine function /H1/							
			Var. INC	10000 INC	1000 INC	100 INC	10 INC	1 INC
DBB **6** Axis and spindle	OEM axis signals							
DBB **7**								
DBB **8**	Request PLC axis/spindle /K5/			Activation signal with change of this byte /K5/	Allocate NC axis to channel /K5/			
					D	C	B	A

Note

DBX8.4: is automatically reset after assignment.

DB 31-61	Continuation: Signals to axis/spindle (PLC → NCK)							
Byte	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
DBB **9**					Lock parameter set definition from NC /A2/	Control parameter block /A2/ C	B	A
DBB **10**								REPOS DELAY
DBB **11**								Start brake test
DBB **12** Axis	Delay reference point approach /R1/			Modulo limit enabled	2nd software limit switch /A3/ plus	minus	Hardware limit switch /A3/ plus	minus
DBB **13** Axis								
DBB **14** Axis								
DBB **15** Axis								
DBB **16** Spindle	Delete S value /S1/	No n-monitoring when changing gear /S1/	Resyn-chronize spindle 1 /S1/	Resyn-chronize spindle 2 /S1/	Gear has changed over /S1/	Actual gear stage /S1/ C	B	A
DBB **17** Spindle		Invert M3/M4 /S1/	Resyn-chronize spindle at pos. 2 /S1/	Resyn-chronize spindle at pos. 1 /S1/				Feedrate override f. spindle valid /S1/
DBB **18** Spindle	Setpoint rot. direct. /S1/ CCW	CW	Oscillating speed /S1/	Oscillation via PLC /S1/				
DBB **19** Spindle	Spindle override /V1/ H	G	F	E	D	C	B	A

DB 31-61	Continuation: Signals to axis/spindle (PLC → NCK)							
Byte	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
DBB **20** Drive 1)	Not used	reserved	Release brake	Not used	Not used	Reserved Yaskawa	ramp generator quick stop /A2/	Not used
DBB **21** Drive 1)	Pulse enable /A2/	n controller integrator disable /A2/	Motor Selection done /A2/	Motor selection /A2/ B	A	Drive parameter set selection 0 … 7 /A2/ C	B	A
DBB **22** Safety Integr. 1)				Selection of Safe speed: bit value 1	bit value 0		Deselect safe standstill	Deselect safe velocity and standstill
DBB **23** Safety Integr.	Activate test stop			Activate end position pair 2		Trans-mission of bit value 2	Trans-mission of bit value 1	Trans-mission of bit value 0
DBB **24**	Master/ slave on	Bit value for CTRLOUT_changed: 1	0	Torque compensa-tion controller ON		CC_Slave axis Suppress link	Control Axis	Stepper motor Rotation monitoring
		Change setpoint output assignment (for compile cycles)						
DBB **25**								
DBB **26** Grinding	Enable ESR response			Enable slave axis overlay	Comp-ensation control ON			
DBB **27** Grinding		Stop				Resume		
	HIAxMove	Corr	DEPBCS	DEPMCS	HIAxMove	Corr	DEPBCS	DEPMCS
DBB **28** Oscillation	PLC checks axis /P5/	AxStop, stop	Stop at next reversal point /P5/	Change reversal point /P5/	Set reversal point /P5/	AXRESUM E	AXRESET	OscillAxExt Reversal
DBB **29** Grinding			Disable automatic synchroni-zation	Start gantry synchroni-zation Gantry				
DBB **30** (Technolog y)				Position spindle	Autom. gear step change	Start spindle Counter-clockwise	Start spindle Clockwise	Stop spindle

1) See note at the end of this subsection

DB 31-61	Continuation: Signals to axis/spindle (PLC → NCK)							
Byte	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
DBB **31** (Technology)		Track synchronism	Disable synchroni-zation	Resynchro-nize				
DBB **32** Safety Integr. 1)			Deselect external stop E	Deselect external stop D	Deselect external stop C	Deselect external stop A		
DBB **33** Safety Integr. 1)	Select override							
	Bit value 3	Bit value 2	Bit value 1	Bit value 0				
DBB **34**								
DBB ...								
DBB **56**						Spindle inside clamping	Spindle speed display	Seperate feed drive as C axis engaged
DBB **57**								
DBB **58**	Internal data for CF 18							
DBB **59**								

Note

The IS ”Delete distance-to-go” (DBX2.2) is effective only for position axes on an axis-specific basis; the IS ”Delete distance-to-go” (DB21-30, DB6.2) acts on a channel-specific basis. The IS ”Spindle reset” (DXB2.2) acts on a spindle-specific basis.

DB 31 -61	Signals from axis/spindle (NCK→PLC)							
Byte	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
DBB **60** Axis and spindle	Position reached /B1/ with exact stop fine	with exact stop coarse	References/ synchro-nizes 2 /R1/	References/ synchro-nizes 1 /R1/	Encoder limit frequency exceeded 2 /A3/	Encoder limit frequency exceeded 1 /A3/	NCU_Link Axis active /B3/	Spindle /no axis /S1/
DBB **61** Axis and spindle	Current controller active /A2/	Speed controller active /A2/	Position controller active /A2/	Axis/spindle stationary ($n < n_{min}$) /A2/	Follow-up mode active /A2/	Axis ready /B3/	Axial alarm	Travel request /F1/
DBB **62**	Axis container rotation active	Force fixed stop limited /F1/	Fixed stop reached /F1/	Activate travel to fixed stop /F1/	Measure-ment active /M5/	Revolutio-nal feedrate active	Handwheel overlay active /H1/	Software cams active /N3/
DBB **63**	Stop HIAxMove active	Corr active	DEPBCS active	DEPMCS active	Axis/ spindle disable active	Axis stop active /P2/	PLC-controlled axis /P2/	AXRESET DONE /P2/
DBB **64** Axis and spindle	Traverse command /H1/ plus	minus	Travel request plus	minus		Handwheel active /H1/ 3	2	1
65 Axis and spindle	Active machine function /H1/		Var. INC	10000 INC	1000 INC	100 INC	10 INC	1 INC

1) See note at the end of this subsection

DB 31-61	Continuation: Signals to axis/spindle (NCK → PLC)							
Byte	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
DBB **66** Axis and spindle	OEM axis signals (reserved)							Activate monitoring /TE6/
DBB **67**								
DBB **68**	PLC axis/ spindle /K5/	Neutral axis/ spindle /K5/	Axis replacement possible /K5/	New type requested by PLC /K5/	NC axis/spindle in channel /K5/ D	C	B	A
DBB **69**	NCU number in NCU link network					Control parameter block C	B	A
DBB **70**						Repos delay quit	Repos shift valid	Repos shift
DBB **71**								Brake test active
DBB **72**								REPOS DELAY
DBB **73**								

DBB **74**				Modulo limit enabled active				
DBB **75**								
DBB **76** Axis	Rounding axis in position	Indexing axis in position /T1/	Positioning axis /P2/	Path axis				Scratch pulse /A2/
DBB **77**								
DBB **78** Axis								
			F function (REAL format) for positioning axis /V1/					
DBB **82** Spindle					Gear change-over /S1/	Setpoint gear stage /S1/ C	B	A
DBB **83** Spindle	Actual rotat. direction CW /S1/	Speed monitoring /W1/	Spindle in setpoint range /S1/	Support area limits violated /S8/	Geometry monitoring /W1/	Set speed increased /S1/	Set speed limited /S1/	Speed limit exceeded /S1/

DB 31-61	Continuation: Signals to axis/spindle (NCK → PLC)							
Byte	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
DBB **84** Spindle	Active spindle operating mode /S1/ Control mode	Oscillation mode	Positioning mode	Synchro-nous mode	Tapping without compen-sating chuck /S1/	CLGON active /S8/	SUG active (grinding wheel surface speed)	Const. cutting speed active
DBB **85** Spindle			Spindle in position					
DBB **86** Spindle	M function (binary) for spindle /S1/							
DBD **88** Spindle	S function (floating-point) for spindle /S1/							
DBB **92** Drive 1)	Not used	reserved	Motor brake released	Not used	Not used	Not used	HLGSS active /A2/	Not used
DBB **93** Drive 1)	Enable pulses /A2/	n controller integrator disabled /A2/	Drive ready /A2/	Active motor /A2/ B	A	Active drive parameter set 0 … 7 /A2/ C	B	A
DBB **94** Drive 1)	reserved	$n_{act} = n_{set}$ /A2/	$\|n_{act}\| < n_x$ /A2/	$\|n_{act}\| < n_{min}$ /A2/	Md < Mdx /A2/	Ramp-up complete /A2/	Temperature prewarning /A2/ Heat sink	Motor
DBB **95** Drive 1)	Limitation of power section I2T			Speed threshold star/delta	Fall short of min. generator voltage	Generator active	Fall short of retract voltage	$U_{DC\text{-}link}$ < alarm threshold /A2/
DBB **96**	Master/ slave active /TE3/	Bit value for CTRLOUT_changed 1	0 Change setpoint output assignment (for compile cycles)	Master/ Slave Compen-sation controller activ	Master/ Slave coarse	Master/ Slave fine	Axis control active	(Stepper motor) error rotation monitoring /S6/
DBB **97**					Offset after turn-on point /TE6/	Activate mirroring /TE6/	Activate link /TE6/	Axis is slave axis /TE6/

1) See note at the end of this subsection

DB 31-61	Continuation: Signals to axis/spindle (NCK → PLC)							
Byte	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
DBB **98** Synchro-nous spindle	Emergency retraction active	Accel. warning threshold reached	Speed warning threshold reached	Overlaid motion /S3/		Actual value coupling /S3/	Synchronism /S3/ coarse	fine
DBB **99** Synchro-nous spindle	Emergency retraction enabled	Max. acce-leration reached	Max. speed reached	Synchro-nization running	Axis acce-lerating		Slave spindle active /S3/	Master spindle active /S3/
DBB **100** Grinding	Oscillation active /P5/	Oscillation motion active /P5/	Spark-out active /P5/	Error in oscillation /P5/	Oscillation cannot start /P5/	OscillAxExt Reversal active		
DBB **101** Gantry	Gantry axis /G1/	Gantry leading axis /G1/	Gantry grouping is synchro-nous /G1/	Gantry synchroni-zation run ready to start /G1/	Gantry warning limit exceeded /G1/	Gantry cut-off limit exceeded /G1/		
DBB **102,103**								
DBB **104** Grinding	Active infeed axis /P5/ Axis 8	Axis 7	Axis 6	Axis 5	Axis 4	Axis 3	Axis 2	Axis 1
DBB **105**	Active infeed axis /P5/ Axis 16							Axis 9
DBB **106**	Active infeed axis /P5/ Axis 24							Axis 17
DBB **107**	Active infeed axis /P5/	Axis 31	Axis 30					Axis 25
DBB **108** 1)	SINUMERIK Safety Integrated / / Axis safely referenced					Status pulses deleted		Safe operational stop / safe speed active

1) See note at the end of this subsection

DB 31-61	**Continuation: Signals from axis/spindle (NCK → PLC)**							
Byte	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
DBB **109** 1)	SINUMERIK Safety Integrated Actual position > cam position							
	SC 4-	SC 4+	SC 3-	SC 3+	SC 2-	SC 2+	SC 1-	SC 1+
DBB **110** 1)	SINUMERIK Safety Integrated							
			n < nx	Safe velocity active bit value 1	Safe velocity active bit value 0		Safe zero speed active	
DBB **111** 1)	Reserved for SINUMERIK Safety Integrated / /							
	Stop E active	Stop D active	Stop C active	Stop A/B active				
DBB **112 -114**	Reserved for SINUMERIK Safety Integrated / /							

Note

This note refers to the signal bytes marked with 1) in column 1 in the above table.

These signal bytes are directly transferred to the interface independently of any configured link communication.

2.2.15 Interface for loading/unloading magazine (DB 71)

DB 71	Interface for loading/unloading magazine (NCK→PLC)							
Byte	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
DBB **0**	Interface (I) active							
	I8	I7	I6	I5	I4	I3	I2	I1
DBB **1**								
	I16	I15	I14	I13	I12	I11	I10	I9
DBB **2,3**								
DBB **n**	res.	res.	res.	NC program positions magazine	Position at loading point	Reload	Unload	Load
DBB **n + 2**	Assigned channel (8 bit Int)							
DBB **n + 3**	Tool management no. (8 bit Int)							
DBD **n + 4**	Unassigned parameter 1 (D word)							
DBD **n + 8**	Unassigned parameter 2 (D word)							
DBD **n + 12**	Unassigned parameter 3 (D word)							
DBW **n + 16**	Identification for loading/unloading station (Int), (fixed value 9999)							
DBW **n + 18**	No. of loading station (Int)							
DBW **n + 20**	Magazine no. (source) for unloading/reloading/positioning (Int)							
DBW **n + 22**	Location no. (source) for unloading/reloading/positioning (Int)							
DBW **n + 24**	Magazine no. (target) for loading/reloading/positioning (Int)							
DBW **n + 26**	Location no. (target) for loading/reloading/positioning (Int)							
DBW **n + 28**	Reserved							

Initial addresses of the loading/unloading stations:
Loading/unloading station 1: n= 4 Loading/unloading station 3: n= 64
Loading/unloading station 2: n= 34 Loading/unloading station 4: n= 94

Load interface 1 is responsible for spindle loading and reloading of tools, for relocating tools and for positioning at any location (e.g. buffer).

References: /FBW/, ”Description of Functions Tool Management”

2.2.16 Interface for spindle as change position (DB 72)

DB 72	Signals from spindle (NCK→PLC)							
Byte	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
DBB **0**	Interface (I) active							
	I8	I7	I6	I5	I4	I3	I2	I1
DBB **1**	I16	I15	I14	I13	I12	I11	I10	I9
DBB **2,3**								
DBB **n**	res.	Replace manual tool	Replace manual tool.	OldT in buffer no. (n-42)	T0	Prepare change	Perform change (initiate: M06)	Compul-sory change
DBB **n + 1**	Unassigned							
DBB **n + 2**	Assigned channel (8 bit Int)							
DBB **n + 3**	Tool management no. (8 bit Int)							
DBD **n + 4**	Unassigned parameter 1 (D word)							
DBD **n + 8**	Unassigned parameter 2 (D word)							
DBD **n + 12**	Unassigned parameter 3 (D word)							
DBW **n + 16**	Buffer identification (Int), (fixed value 9998) (corresponds to "Target position for new tool")							
DBW **n + 18**	Relative location (target) in the buffer (Int)							
DBW **n + 20**	Magazine no. (source) for new tool (Int)							
DBW **n + 22**	Location no. (source) for new tool (Int)							
DBW **n + 24**	Magazine no. (target) for old tool (Int)							
DBW **n + 26**	Location no. (target) for old tool (Int)							
DBW **n + 28**	Tool new: location type (Int)							
DBW **n + 30**	Tool new: size left (Int)							
DBW **n + 32**	Tool new: size right (Int)							

DBW **n + 34**	Tool new: size top (Int)							
DBW **n + 36**	Tool new: size bottom (Int)							
DBW **n + 38**	Tool status for tool new							
	Tool was in use	Tool fixed location coded	Tool being changed	Prewarning limit reached	Tool measured	Tool disabled	Tool enabled	Active tool
				Bit 12 Master tool	Bit 11 to be loaded	Bit 10 to be unloaded	Bit 9 ignore disabled	Bit 8 ID for tools in buffer
DBW **n + 40**	Tool new: T no. (Int)							
DBW **n + 42**	If DBX (n+0.4) = 1, then buffer location of old tool is entered here.							
DBW **n + 44**	Original magazine of new tool							
DBW **n + 46**	Original location of new tool							
Initial addresses of the buffers: Spindle 1:n= 4 Spindle 2:n = 52								

References: /FBW/, ”Description of Functions, Tool Management”

2.2.17 Interface for circular magazine (DB 73)

DB73	Signals from circular magazine (NCK→PLC)							
Byte	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
DBB **0**	Interface (I) active							
	I8	I7	I6	I5	I4	I3	I2	I1
DBB **1**	I16	I15	I14	I13	I12	I11	I10	I9
DBB **2,3**								
DBB **n**	res.	res.	res.	res.	T0	res.	Perform change (initiation: T no.)	Obligatory change
DBB **n + 1**	Unassigned							
DBB **n + 2**	Assigned channel (8 bit Int)							
DBB **n + 3**	Tool management no. (8 bit Int)							
DBD **n + 4**	Unassigned parameter 1 (D word)							
DBD **n + 8**	Unassigned parameter 2 (D word)							
DBD **n + 12**	Unassigned parameter 3 (D word)							
DBW **n + 16**	Reserved							
DBW **n + 18**	Reserved							
DBW **n + 20**	Circular magazine no. (Int)							
DBW **n + 22**	Location no. for new tool (Int)							
DBW **n + 24**	Reserved							
DBW **n + 26**	Location no. for old tool (Int)							
DBW **n + 28**	Tool new: location type (Int)							

DBW **n + 30**	Tool new: size left (Int)							
DBW **n + 32**	Tool new: size right (Int)							
DBW **n + 34**	Tool new: size top (Int)							
DBW **n + 36**	Tool new: size bottom (Int)							
DBW **n + 38**	Tool status for tool new							
	Tool was in use	Tool fixed location coded	Tool being changed	Prewarning limit reached	Tool measured	Tool disabled	Tool enabled	Active tool
				Bit 12 Master tool	Bit 11 to be loaded	Bit 10 to be unloaded	Bit 9 ignore disabled	Bit 8 ID for tools in buffer
DBW **n + 40**	Tool new: T no. (Int)							
DBW **n + 42**	Original location of new tool in this circular magazine							
Initial addresses of the circular magazines: circular magazine 1: n = 4 2: n = 48								

References: /FBW/, "Description of Functions, Tool Management"

2.2.18 Signals to and from the machine control panel and HHU (840Di with MCI2 only) (DB 77)

DB77	**Signals to and from the machine control panel and HHU**							
Byte	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
DBB **0 to** DBB **7**	Input signals from MCP1 to PLC, MPI Bus (GD communication)							
DBB **8 to** DBB **15**	Output signals from MCP1 to PLC, MPI Bus (GD communication)							
DBD **16**	Status send MCP1, MPI bus (GD communication)							
DBD **20**	Status receive MCP1, MPI bus (GD communication)							
DBB **24 to** DBB **31**	Input signals from MCP2 to PLC, MPI bus (GD communication)							
DBB **32 to** DBB **39**	Output signals from MCP2 to PLC, MPI bus (GD communication)							
DBD **40**	Status send MCP2, MPI bus (GD communication)							

DB77	**Signals to and from the machine control panel and HHU**							
Byte	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
DBD **44**	Status receive MCP2, MPI bus (GD communication)							
DBB **48** to DBB **53**	Input signals from HHU to PLC, MPI bus (GD communication)							
DBB **60** to DBB **79**	Output signals from PLC to HHU, MPI bus (GD communication)							
DBD **80**	Status Send HHU, MPI bus (GD communication)							
DBD **84**	Status Receive HHU, MPI bus (GD communication)							

```
FB1- parameter:
MCPNum          :=1, //correct number of MCPs
MCP1In  :=P#DB77.DBX0.0,
MCP1Out         := P#DB77.DBX8.0,
MCP1StatSend    := P#DB77.DBX16.0,
MCP1StatRec     := P#DB77.DBX20.0,
MCP2In  :=P#DB77.DBX24.0,
MCP2Out         := P#DB77.DBX32.0,
MCP2StatSend    := P#DB77.DBX40.0,
MCP2StatRec     := P#DB77.DBX44.0,
MCPSDB210 := TRUE,

BHG:    :=1;    //handheld unit interface:
                //0 - no HHU
                //1 – HHU to MPI
                //2 – HHU to OPI
BHGIn   :=P#DB77.DBX48.0,       //transmitted data of handheld unit
BHGOut: :=P#DB77.DBX60.0,       //received data of handheld unit
BHGStatSend: :=P#DB77.DBX80.0, // status DW for transmitting handheld unit
BHGStatRec: :=P#DB77.DBX84.0,  // status DW for receiving HHU
```

■

Für Notizen

3

3 Interface Signals power line

3.1 Data modules (DB) of the PLC application interface

Please find the descriptin of Data modules (DB) of the PLC application interface in chapter 5.3.

3.2 Interface signals of the PLC application interface

General

In the following list of interface signals, a reference to relevant documentation is provided for every signal.
This reference specifies the section number or the short designation of the description of functions, please refer to

References: /FB/, xx, "yyy"

xx Short designation of individual description of functions (e.g.: /A2/)

yyy Name of description of functions (e.g.: "Various interface signals" or title of the guide)

Inverse signals

Signals marked with a "*" are so-called inverse signals. These signals initiate the appropriate function when a 0 signal appears rather than a 1 signal (e.g. MCP, byte n+2.0: *NC STOP).

Legend

- In STEP7, DBB means data module byte
- In STEP7, DBW means data module word (16 bits)
- In STEP7, DBD means data module double word (32 bits)

3.2.1 Signals from/to machine control panel, M version

Signals from machine control panel (keys)								
Byte	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
IB n + 0	Spindle speed override				Operating mode			
	D	C	B	A	JOG	TEACH IN	MDA	AUTO
IB n + 1	Machine function							
	REPOS	REF	var. INC	10000 INC	1000 INC	100 INC	10 INC	1 INC
IB n + 2	Key-switch position 0	Key-switch position 2	Spindle start	*Spindle stop	Feed start	*Feed stop	NC Start	*NC Stop
IB n + 3		Key-switch position 1		Feedrate override				
	Reset		Single block	E	D	C	B	A
IB n + 4	Direction keys			Keyswitch position 3	Direction keys			
	+ R15	- R13	Rapid traverse R14		x R1	4th axis R4	7th axis R7	R10
IB n + 5	Axis selection							
	Y R2	Z R3	5th axis R5	Traverse command MCS/WCS R12	R11	R9	8th axis R8	6th axis R6
IB n + 6	Unassigned customer keys							
	T9	T10	T11	T12	T13	T14	T15	
IB n + 7	Unassigned customer keys							
	T1	T2	T3	T4	T5	T6	T7	T8

Signals to machine control panel (LEDs)								
Byte	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
QB n + 0	Machine function				Operating mode			
	1000 INC	100 INC	10 INC	1 INC	JOG	TEACH IN	MDA	AUTO
QB n + 1	Feed start	*Feed stop	NC Start	*NC Stop	Machine function			
					REPOS	REF	var. INC	10000 INC
QB n + 2	Axis selection					Single block	Spindle start	*Spindle stop
	Direction key - R13	X R1	4th axis R4	7th axis R7	R10			
QB n + 3	Axis selection							
	Z R3	5th axis R5	Travel command MCS/WCS R12	R11	R9	8th axis R8	6th axis R6	Direction key + R15
QB n + 4	Unassigned customer keys							Y
	T9	T10	T11	T12	T13	T14	T15	R2
QB n + 5	Unassigned customer keys							
	T1	T2	T3	T4	T5	T6	T7	T8

Note

With the SINUMERIK 840D, the machine control panel is assigned to the input/output area by GP parameters; as a standard, initial address 0 is specified for the input and output areas.

With FM-NC, the initial address is set via the SDB 210. For the supplied SDB 210, initial address 120 is specified. If another initial address is desired, this must be specified via the STEP 7 Package Communication Configuration. Note that the GD parameters given automatically through Communication Configuration must be set on the machine control panel.

3.2.2 Signals from/to machine control panel, T version

Signals from machine control panel (keys)

Byte	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
IB n + 0	Spindle speed override				Operating mode			
	D	C	B	A	JOG	TEACH IN	MDA	AUTO
IB n + 1	Machine function							
	REPOS	REF	var. INC	10000 INC	1000 INC	100 INC	10 INC	1 INC
IB n + 2	Keyswitch position 0	Keyswitch position 2	Spindle start	*Spindle stop	Feed start	*Feed stop	NC Start	*NC Stop
IB n + 3		Keyswitch position 1		Feed override				
	Reset		Single block	E	D	C	B	A
IB n + 4				Keyswitch position 3	Direction keys			
	R15	R13	R14		+Y R1	-Z R4	-C R7	R10
IB n + 5	Direction keys							
	+X R2	+C R3	Rapid traverse override R5	Travel command MCS/WCS R12	R11	-Y R9	-X R8	+Z R6
IB n + 6	Unassigned customer keys							
	T9	T10	T11	T12	T13	T14	T15	
IB n + 7	Unassigned customer keys							
	T1	T2	T3	T4	T5	T6	T7	T8

Signals to machine control panel (LEDs)

Byte	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
QB n + 0	Machine function				Operating mode			
	1000 INC	100 INC	10 INC	1 INC	JOG	TEACH IN	MDA	AUTO
QB n + 1	Feed start	*Feed stop	NC Start	*NC Stop	Machine function			
					REPOS	REF	var. INC	10000 INC
QB n + 2	Direction keys					Single block	Spindle start	*Spindle stop
	R13	+Y R1	-Z R4	-C R7	R10			
QB n + 3	Direction keys							
	R3	R5	Travel command MCS/WCS	R11	-Y R9	-X R8	+Z R6	R15
QB n + 4	Unassigned customer keys							Direction key +X

	T9	T10	T11	T12	T13	T14	T15	R2
QB n + 5	Unassigned customer keys							
	T1	T2	T3	T4	T5	T6	T7	T8

3.2.3 Signals from/to slimline machine control panel

Signals from slimline machine control panel (keys and switches)								
Byte	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
IB n + 0	Spindle speed override				Operating mode			
	*NC Stop	SP -	SP 100%	SP +	SINGLEB	JOG	MDA	AUTOM.
IB n + 1	Spindle				Keyswitch	Machine function		
	NC Start	SP right	*SP Stop	SP left	SS 3	REF.	REPOS	Teach in
IB n + 2	Feedrate			Keyswitch	Machine functions			
	START	*STOP	var. INC	SS 0	1000 INC	100 INC	10 INC	1 INC
IB n + 3		Keyswitch		Feed override				
	RESET	SS 2	SS 1	E	D	C	B	A
IB n + 4	Direction keys			Optional customer keys				
	(+) R15	(-) R13	Rapid traverse R14	KT4	KT3	KT2	KT1	KT0
IB n + 5			Axis selection					
	T17	KT5	6	5	4	Z	Y	X
IB n + 6	Unassigned customer keys							
	T9	T10	T11	T12	T13	T14	T15	T16
IB n + 7	Unassigned customer keys							
	T1	T2	T3	T4	T5	T6	T7	T8

Signals to slimline machine control panel (LEDs)								
Byte	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
QB n + 0	Spindle speed override				Operating mode			
	NC Stop	SP -	SP 100 %	SP +	SINGLEB	JOG	MDA	AUTOM.
QB n + 1	Spindle					Machine function		
	NC Start	SP right	SP Stop	SP left	Unassigned	REF.	REPOS	Teach in
QB n + 2	Feedrate				Machine functions			
	START	STOP	var. INC	Unassigned	1000 INC	100 INC	10 INC	1 INC
QB n + 3	Unassigned							
	Unassigned	Unassigned	Unassigned	Unassigned	Unassigned	Unassigned	Unassigned	Unassigned
QB n + 4	Direction keys			Optional customer keys				
	(+) R15	(-) R13	Rapid traverse R14	KT4	KT3	KT2	KT1	KT0
QB n + 5			Axis selection					
	T17	KT5	6	5	4	Z	Y	X
QB n + 6	Unassigned customer keys							
	T9	T10	T11	T12	T13	T14	T15	T16
QB n + 7	Unassigned customer keys							
	T1	T2	T3	T4	T5	T6	T7	T8

3.2.4 Signals from/to handheld unit (HHU)

Signals from handheld unit (keys) (input display)								
Byte	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
IB n + 0	Reserved							
IB n + 1	Reserved							
IB n + 2								
	T9	T7	T6	T5	T4	T3	T2	T1
IB n + 3								
	T16	T15	T14	T13	T12	T11	T10	T9
IB n + 4								
	T24	T23	T22	T21				
IB n + 5	Acknowl-edgement	Keyswitch	Rapid traverse/feed override switch					
	Digital display		E	D	C	B	A	

Signals to handheld unit (LEDs) (Output display, LEDs)								
Byte	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
QB n + 0	always 1							
QB n + 1	New data for selected line							Line selection
QB n + 2	L8	L7	L6	L5	L4	L3	L2	L1
QB n + 3	L16	L15	L14	L13	L12	L11	L10	L9
HHU digital display								
QB n + 4	1st character (right) of the selected line							
QB n + 5	2nd character of the selected line							
QB ...								
QB n + 18	15th character of the selected line							
QB n + 19	16th character (left) of the selected line							

Note
With the SINUMERIK 840D, the handheld unit is connected to the OPI or MCP interface of the PLC.

The initial addresses of the input/output areas and the activation must be set via basic program parameter FB1. With the SINUMERIK 810D and FM-NC, the handheld unit is connected to the MPI interface of the PLC.

The initial addresses of the input/output areas as well as the number of bytes to be transferred must be specified via the STEP 7 Package Communication Configuration.

Note that the GD parameters given automatically through Communication Configuration must be set on the handheld unit.

The parameterization is described in the Installation and Start-Up Guide and in the Description of Functions P3 ”Basic PLC program”.

References: /BH/, ”Operator Components Manual”

3.2.5 Signals from/to handheld programming unit (HPU)

Signals from machine control panel simulation Interface HPU→PLC								
Byte	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
IB n + 0	Function key block							
	REF	TEACH	AUTO	MDA	JOG	QUIT	RESET	WCS/MCS
IB n + 1	Function key block							
		FCT15	FCT14	BigFct	FCT12	FCT11	INC	REPOS
IB n + 2	JOG keys positive direction							
		If 1: Ax1-Ax6= Ax7-Ax12	Ax6	Ax5	Ax4	Ax3	Ax2	Ax1
IB n + 3	JOG keys negative direction							
			Ax6	Ax5	Ax4	Ax3	Ax2	Ax1
IB n + 4	Shift keys							
	Signal	Diagno	Service	System	Param	Correct	Progr.	Mach.
IB n + 5	Shift keys							
	BF16	BF15	BF14	BF13	BF12	Step	Modify	Insert
IB n + 6	Start key block							
	Res.	HT 6	VAL+	VAL-	SF2	SF1	START	STOP
IB n + 7								

Signals to machine control panel simulation Interface PLC→HPU								
Byte	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
QB n + 0	Function key block							
	REF	TEACH	AUTO	MDA	JOG	QUIT	RESET	WCS/MCS
QB n + 1	Function key block							
		FCT15	FCT14	BigFct	FCT12	FCT11	INC	REPOS
QB n + 2		Axes 7–12 selected	JOG keys positive direction					
QB n + 2								
			Ax6	Ax5	Ax4	Ax3	Ax2	Ax1
QB n + 3	JOG keys negative direction							
	For WCS: No MCS Ax4 to Ax6		Ax6	Ax5	Ax4	Ax3	Ax2	Ax1
QB n + 4	Shift keys							
	Signal	Diagno	Service	System	Param	Correct	Progr.	Mach.
QB n + 5	Shift keys							
	BF16	BF15	BF14	BF13	BF12	Step	Modify	Insert
QB n + 6	Start key block							
			VAL+	VAL-	SF2	SF1	START	STOP
QB n + 7								

3.2.6 PLC messages (DB 2)

DB2	Signals for PLC messages (PLC→MMC), /P3/							
Byte	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
	Channel 1							
0	510007	510006	510005	510004	510003	510002	510001	510000
	Feed disable (alarm no.: 510000–510015)							
1	510015	510014	510013	510012	510011	510010	510009	510008
2	Feed and read-in disable byte1 (alarm no.: 510100–510131)							
3	Feed and read-in disable byte 2 (alarm no.: 510108–510115)							
4	Feed and read-in disable byte 3 (alarm no.: 510116–510123)							
5	Feed and read-in disable byte 4 (alarm no.: 510124–510131)							
6	Read-in disable byte 1 (alarm no.: 510200–510207)							
7	Read-in disable byte 2 (alarm no.: 510208–510215)							

8	Read-in disable byte 3 (alarm no.: 510216–510223)							
9	Read-in disable byte 4 (alarm no.: 510224–510231)							
10	NC Start disable byte 1 (alarm no.: 510300–510307)							
11	NC Start disable byte 2 (alarm no.: 510308–510315)							
12	Feed stop GEOaxis 1 byte 1 (alarm no.: 511100–511107)							
13	Feed stop GEOaxis 1 byte 2 (alarm no.: 511108–511115)							
14	Feed stop GEOaxis 2 byte 1 (alarm no.: 511200–511207)							
15	Feed stop GEOaxis 2 byte 2 (alarm no.: 511208–511215)							
16	Feed stop GEOaxis 3 byte 1 (alarm no.: 511300–511307)							
17	Feed stop GEOaxis 3 byte 2 (alarm no.: 511308–511315)							
	Channel 2							
18	520007	520006	520005	520004	520003	520002	520001	520000
	Feed disable (alarm no.: 520000-520015)							
19	520015	520014	520013	520012	520011	520010	520009	520008
20-23	Feed and read-in disable byte 1-4 (alarm no.: 520100–520131)							
24-27	Read-in disable byte 1-4 (alarm no.: 520200–520231)							
28-29	NC Start disable byte 1-2 (alarm no.: 520300–520315)							
30-31	Feed stop GEOaxis 1 byte 1-2 (alarm no.: 521100-521115)							
32-33	Feed stop GEOaxis 2 byte 1-2 (alarm no.: 521200-521215)							
34-35	Feed stop GEOaxis 3 byte 1-2 (alarm no.: 521300-521315)							
36-143	As from channel 3, please refer to the following table							

Channel areas in DB2

Area	**Address**	**Signal number**
Channel 1, see above	DBX 0.0 - DBX 11.7	510.000 - 510.231
Channel 1, geo axes	DBX 12.0 - DBX 17.7	511.100 - 511.315
Channel 2, see above	DBX 18.0 - DBX 29.7	520.000 - 520.231
Channel 2, geo axes	DBX 30.0 - DBX 35.7	521.100 - 521.315
Channel 3	DBX 36.0 - DBX 47.7	530.000 - 530.231
Channel 3, geo axes	DBX 48.0 - DBX 53.7	531.100 - 531.315
Channel 4	DBX 54.0 - DBX 65.7	540.000 - 540.231
Channel 4, geo axes	DBX 66.0 - DBX 71.7	541.100 - 541.315
Channel 5	DBX 72.0 - DBX 83.7	550.000 - 550.231
Channel 5, geo axes	DBX 84.0 - DBX 89.7	551.100 - 551.315
Channel 6	DBX 90.0 - DBX 101.7	560.000 - 560.231
Channel 6, geo axes	DBX 102.0 - DBX 107.7	561.100 - 561.315
Channel 7	DBX 108.0 - DBX 119.7	570.000 - 570.231
Channel 7, geo axes	DBX 120.0 - DBX 125.7	571.100 - 571.315
Channel 8	DBX 126.0 - DBX 137.7	580.000 - 580.231
Channel 8, geo axes	DBX 138.0 - DBX 143.7	581.100 - 581.315
Channel 9, channel 10 in SW 5 not implemented		

Axis areas in DB2

	Axis/spindle							
144	600107	600106	600105	600104	600103	600102	600101	600100
	Feed stop/spindle stop (alarm no.: 600100-600015) for axis/spindle 1							
145	600115	600114	600113	600112	600111	600110	600109	600108
146-147	Feed stop/spindle stop (alarm no.: 600200-600215) for axis/spindle 2							
148-149	Feed stop/spindle stop (alarm no.: 600300-600315) for axis/spindle 3							
150-151	Feed stop/spindle stop (alarm no.: 600400-600415) for axis/spindle 4							
152-153	Feed stop/spindle stop (alarm no.: 600500-600515) for axis/spindle 5							
154-155	Feed stop/spindle stop (alarm no.: 600600-600615) for axis/spindle 6							
156-157	Feed stop/spindle stop (alarm no.: 600700-600715) for axis/spindle 7							
158-159	Feed stop/spindle stop (alarm no.: 600800-600815) for axis/spindle 8							
160-161	Feed stop/spindle stop (alarm no.: 600900-600915) for axis/spindle 9							
162-163	Feed stop/spindle stop (alarm no.: 601000-601015) for axis/spindle 10							
164-165	Feed stop/spindle stop (alarm no.: 601100-601115) for axis/spindle 11							
166-167	Feed stop/spindle stop (alarm no.: 601200-601215) for axis/spindle 12							
168-169	Feed stop/spindle stop (alarm no.: 601300-601315) for axis/spindle 13							
170-171	Feed stop/spindle stop (alarm no.: 601400-601415) for axis/spindle 14							
172-173	Feed stop/spindle stop (alarm no.: 601500-601515) for axis/spindle 15							
174-175	Feed stop/spindle stop (alarm no.: 601600-601615) for axis/spindle 16							
176-177	Feed stop/spindle stop (alarm no.: 601700-601715) for axis/spindle 17							
178-179	Feed stop/spindle stop (alarm no.: 601800-601815) for axis/spindle 18							
	Axes 19 -31 not implemented							

User areas

	User area 0 Bytes 1 - 8							
180	700007	700006	700005	700004	700003	700002	700001	700000
...	User area 0 (alarm no.: 700000-700063)							
187	700063	700062	700061	700060	700059	700058	700057	700056
188-195	User area 1 Bytes 1 - 8 (alarm no.: 700100-700163)							
...								
372-379	User area 24 Bytes 1 - 8 (alarm no.: 702400-702463)							

Note

In DB2, the assignment is made between message/alarm number, text and area identifier. All alarm or message bits are automatically transferred to the user interface (channel, axis/spindle) through appropriate parameter settings. If these parameter settings are not made, the bit transfer must be programmed in the user program. The user interface can be further influenced after the block for the error/operational messages has been called. Only signals of the channels and axes declared in the NC machine data can be transferred and texts displayed.

The user must acknowledge all error messages generated. Operational messages are displayed only for as long as the relevant condition prevails.

The number of user areas can be parameterized via FB 1.

DB2/DB3 must be deleted after changing the configuration (FB1: MsgUser).

Definition of error and operational messages /P3/							
Byte no. of DB2 / Error message EM or operational message OM							
7 / EM	6 / EM	5 / OM	4 / OM	3 / EM	2 / EM	1 / OM	0 / EM
15 / OM	14 / EM	13 / OM	12 / EM	11 / OM	10 / EM	9 / OM	8 / OM
23 / OM	22 / OM	21 / EM	20 / EM	19 / OM	18 / EM	17 / OM	16 / EM
31 / OM	30 / EM	29 / OM	28 / EM	27 / OM	26 / OM	25 / EM	24 / EM
				35 / OM	34 / EM	33 / OM	32 / EM
151 / OM	150 / EM	149 / OM	148 / EM	147 / OM	146 / EM	145 / OM	144 / EM
159 / OM	158 / EM	157 / OM	156 / EM	155 / OM	154 / EM	153 / OM	152 / EM
187 / OM	186 / OM	185 / OM	184 / OM	183 / EM	182 / EM	181 / EM	180 / EM
195 / OM	194 / OM	193 / OM	192 / OM	191 / EM	190 / EM	189 / EM	188 / EM

Example

The alarms numbered from 510200 to 510207 can be generated via DB2, DBB6 (read-in disable channel 1). These alarms are defined as error messages as standard.

3.2.7 Signals to NC (DB 10)

On-board input and output signals from NCK

DB10	Signals to NC (PLC→NC)							
Byte	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
DBB **0**	Disabling of digital NCK inputs /A2/ (SW 2 and higher) Digital inputs without hardware #)				On-board inputs §)			
	Input 8	Input 7	Input 6	Input 5	Input 4	Input 3	Input 2	Input 1
DBB **1**	Setting of digital NCK inputs from PLC (SW 2 and higher) Digital inputs without hardware #)				On-board inputs §)			
	Input 8	Input 7	Input 6	Input 5	Input 4	Input 3	Input 2	Input 1
DBB **2, 3 unas-signed**								
DBB **4**	Disabling of digital NCK outputs /A2/ (SW 2 and higher) Digital outputs without hardware #)				On-board outputs §)			
	Output 8	Output 7	Output 6	Output 5	Output 4	Output 3	Output 2	Output 1
DBB **5**	Overwrite screenform of digital NCK /A2/ outputs (SW 2 and higher) Digital outputs without hardware #)				On-board outputs §)			
	Output 8	Output 7	Output 6	Output 5	Output 4	Output 3	Output 2	Output 1
DBB **6**	Setting value of digital NCK outputs from PLC /A2/ (SW 2 and higher) Digital outputs without hardware #)				On-board outputs §)			
	Output 8	Output 7	Output 6	Output 5	Output 4	Output 3	Output 2	Output 1
DBB **7**	Input screenform of digital NCK outputs /A2/ (SW 2 and higher) Digital outputs without hardware #)				On-board outputs §)			
	Output 8	Output 7	Output 6	Output 5	Output 4	Output 3	Output 2	Output 1
DBB **8-29**	Machine axis number table for FC 19, 24, 25, 26 (1st MCP)							
DBB **30** from SW 6	Upper limit of machine axis numbers for FC 19, 24 (1st MCP) With 0, the max. number of machine axis numbers applies							
DBB **32-53** from SW 6	Machine axis number table for FC 19, 24, 25, 26 (2nd MCP)							
DBB **54** from SW 6	Upper limit of machine axis numbers for FC 19, 24 (2nd MCP) With 0, the max. number of machine axis numbers applies							

Note

#) Bits 4-7 of the digital input and NCK outputs can be processed by the PLC even though there are no hardware I/Os available for this. Therefore, these bits can be used in addition to the information exchange between NCK and PLC.

§) On the 840D, the digital inputs and outputs 1 to 4 of the NCK are physically on-board. On the FM-NC, there are no hardware I/Os for bit 0 to bit 3. These can be processed by the PLC according to #).

The external I/O signals from the NCK have been shifted to the range starting with DBB122.

General signals to NCK (DB10)

DB10	Signals to NC (PLC→NC)							
Byte	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
DBB **56**	Keyswitch /A2/ Position 3	Position 2	Position 1	Position 0		Acknowl. EMER-GENCY STOP /N2/	EMER-GENCY STOP /N2/	
DBB **57**					PC shutdown Only 840Di evaluated			INC inputs in mode group area active
DBB **58 - 59**								

External digital inputs of the NCK (DB10)

DB10	Signals to NC (PLC -> NCK)							
Byte	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
DBB **122**	Disable the external digital NCK inputs (SW 2 and higher) Input 16	Input 15	Input 14	Input 13	Input 12	Input 11	Input 10	Input 9
DBB **123**	Values from the PLC for the external digital NCK inputs (SW 2 and higher) Input 16	Input 15	Input 14	Input 13	Input 12	Input 11	Input 10	Input 9
DBB **124**	Disable the external digital NCK inputs (SW 2 and higher) Input 24	Input 23	Input 22	Input 21	Input 20	Input 19	Input 18	Input 17
DBB **125**	Values from the PLC for the external digital NCK inputs (SW 2 and higher) Input 24	Input 23	Input 22	Input 21	Input 20	Input 19	Input 18	Input 17
DBB **126**	Disable the external digital NCK inputs (SW 2 and higher) Input 32	Input 31	Input 30	Input 29	Input 28	Input 27	Input 26	Input 25
DBB **127**	Values from the PLC for the external digital NCK inputs (SW 2 and higher) Input 32	Input 31	Input 30	Input 29	Input 28	Input 27	Input 26	Input 25

DBB **128**	Disable the external digital NCK inputs (SW 2 and higher)							
	Input 40	Input 39	Input 38	Input 37	Input 36	Input 35	Input 34	Input 33
DBB **129**	Values from the PLC for the external digital NCK inputs (SW 2 and higher)							
	Input 40	Input 39	Input 38	Input 37	Input 36	Input 35	Input 34	Input 33

External digital outputs of the NCK (DB10)

DB10	**Signals to NC (PLC -> NCK)**							
Byte	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
DBB **130**	Disable the external digital NCK outputs (SW 2 and higher)							
	Output 16	Output 15	Output 14	Output 13	Output 12	Output 11	Output 10	Output 9
DBB **131**	Overwrite screenform for the external digital NCK outputs (SW 2 and higher)							
	Output 16	Output 15	Output 14	Output 13	Output 12	Output 11	Output 10	Output 9
DBB **132**	Value from the PLC for the external digital NCK outputs (SW 2 and higher)							
	Output 16	Output 15	Output 14	Output 13	Output 12	Output 11	Output 10	Output 9
DBB **133**	Default screenform for the external digital NCK outputs (SW 2 and higher)							
	Output 16	Output 15	Output 14	Output 13	Output 12	Output 11	Output 10	Output 9
DBB **134**	Disable the external digital NCK outputs (SW 2 and higher)							
	Output 24	Output 23	Output 22	Output 21	Output 20	Output 19	Output 18	Output 17
DBB **135**	Overwrite screenform for the external digital NCK outputs (SW 2 and higher)							
	Output 24	Output 23	Output 22	Output 21	Output 20	Output 19	Output 18	Output 17
DBB **136**	Value from the PLC for the external digital NCK outputs (SW 2 and higher)							
	Output 24	Output 23	Output 22	Output 21	Output 20	Output 19	Output 18	Output 17
DBB **137**	Default screenform for the external digital NCK outputs (SW 2 and higher)							
	Output 24	Output 23	Output 22	Output 21	Output 20	Output 19	Output 18	Output 17
DBB **138**	Disable the external digital NCK outputs (SW 2 and higher)							
	Output 32	Output 31	Output 30	Output 29	Output 28	Output 27	Output 26	Output 25
DBB **139**	Overwrite screenform for the external digital NCK outputs (SW 2 and higher)							
	Output 32	Output 31	Output 30	Output 29	Output 28	Output 27	Output 26	Output 25

DBB **140**	Value from the PLC for the external digital NCK outputs (SW 2 and higher)							
	Output 32	Output 31	Output 30	Output 29	Output 28	Output 27	Output 26	Output 25
DBB **141**	Default screenform for the external digital NCK outputs (SW 2 and higher)							
	Output 32	Output 31	Output 30	Output 29	Output 28	Output 27	Output 26	Output 25
DBB **142**	Disable the external digital NCK outputs (SW 2 and higher)							
	Output 40	Output 39	Output 38	Output 37	Output 36	Output 35	Output 34	Output 33
DBB **143**	Overwrite screenform for the external digital NCK outputs (SW 2 and higher)							
	Output 40	Output 39	Output 38	Output 37	Output 36	Output 35	Output 34	Output 33
DBB **144**	Value from the PLC for the external digital NCK outputs (SW 2 and higher)							
	Output 40	Output 39	Output 38	Output 37	Output 36	Output 35	Output 34	Output 33
DBB **145**	Default screenform for the external digital NCK outputs (SW 2 and higher)							
	Output 40	Output 39	Output 38	Output 37	Output 36	Output 35	Output 34	Output 33

Analog inputs of the NCK (external) (DB10)

DB10	**Signals to NC (PLC -> NCK)**							
Byte	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
DBB **146**	Disable the analog NCK inputs							
	Input 8	Input 7	Input 6	Input 5	Input 4	Input 3	Input 2	Input 1
DBB **147**	Specified analog value for NCK from PLC							
	Input 8	Input 7	Input 6	Input 5	Input 4	Input 3	Input 2	Input 1
DBW **148**	Setpoint from PLC for analog input 1 of NCK							
DBW **150**	Setpoint from PLC for analog input 2 of NCK							
DBW **152**	Setpoint from PLC for analog input 3 of NCK							
DBW **154**	Setpoint from PLC for analog input 4 of NCK							
DBW **156**	Setpoint from PLC for analog input 5 of NCK							
DBW **158**	Setpoint from PLC for analog input 6 of NCK							
DBW **160**	Setpoint from PLC for analog input 7 of NCK							
DBW **162**	Setpoint from PLC for analog input 8 of NCK							
DBB **164,165**	Unassigned							

Analog outputs of the NCK (external) (DB10)

DB10	Signals to NCK (PLC -> NCK)							
Byte	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
DBB **166**	Overwrite screenform for the analog NCK outputs							
	Output 8	Output 7	Output 6	Output 5	Output 4	Output 3	Output 2	Output 1
DBB **167**	Default screenform for the analog NCK outputs							
	Output 8	Output 7	Output 6	Output 5	Output 4	Output 3	Output 2	Output 1
DBB **168**	Disable the analog NCK outputs							
	Output 8	Output 7	Output 6	Output 5	Output 4	Output 3	Output 2	Output 1
DBB **169**	RESERVED							
DBW **170**	Setpoint from PLC for analog output 1 of NCK							
DBW **172**	Setpoint from PLC for analog output 2 of NCK							
DBW **174**	Setpoint from PLC for analog output 3 of NCK							
DBW **176**	Setpoint from PLC for analog output 4 of NCK							
DBW **178**	Setpoint from PLC for analog output 5 of NCK							
DBW **180**	Setpoint from PLC for analog output 6 of NCK							
DBW **182**	Setpoint from PLC for analog output 7 of NCK							
DBW **184**	Setpoint from PLC for analog output 8 of NCK							

3.2.8 Signals from/to NCK/MMC (DB 10)

On-board NCK inputs and outputs (DB 10)

DB10	Signals from (NCK -> PLC)							
Byte	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
DBB **60**					Actual value of the digital ON-BOARD inputs of the NCK (SW 2 and higher) On-board inputs §)			
					Input 4	Input 3	Input 2	Input 1
DBB **61-63**								
DBB **64**	Setpoint for the digital outputs of the NCK without hardware				Setpoint for the digital on-board outputs of the NCK			
	Output 8	Output 7	Output 6	Output 5	Output 4	Output 3	Output 2	Output 1
DBB **65-67**				Unassigned				
DBB **68**	Handwheel 1 moved							
DBB **69**	Handwheel 2 moved							
DBB **70**	Handwheel 3 moved							
DBB **71**	Modification counter inch/metric system of units							
DBB **72** HT6 from SW 6.1.51	Status of the actual value display indicated (1st MCP)							
							Machine/ Work	Valid display
DBB **73** HT6 from SW 6.1.51	Status of the actual value display indicated (2nd MCP)							
							Machine/ Work	Valid display
DBB **74-79** HT6 from SW 6.1.51	Machine axis numbers of the displayed axes (1st MCP) MCP1AxisFromHMI							
DBB **80-85** HT6 from SW 6.1.51	Machine axis numbers of the displayed axes (2nd MCP) MCP2AxisFromHMI							
DBB **86**	Reserved							
DBB **88**	Reserved							

Note

#) Although no associated hardware I/Os exist, the PLC can process bits 4-7 of the digital inputs and NCK outputs. Consequently, these bits can also be used to transfer information between the NCK and the PLC.

§) The digital inputs and outputs 1 to 4 of the NCK exist as on-board hardware for the 840D. No hardware I/Os are available for bits 0-3 of the FM-NC. In accordance with #), these can be processed by the PLC.

Selection/status signals from MMC (DB 10)

DB 10	Signals from NC (NCK→PLC)							
Byte	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
DBB **90**								
DBB **91**								
DBB **92**	free							
DBB **93**	free							
DBB **94**	free							
DBB **95**	free							
DBB **96**	free							
DBB **97** MMC--> PLC					Channel number for handwheel 1 /H1/ (SW 2 and higher)			
					D	C	B	A
DBB **98** MMC--> PLC					Channel number for handwheel 2 /H1/ (SW 2 and higher)			
					D	C	B	A
DBB **99** MMC--> PLC					Channel number for handwheel 3 /H1/ (SW 4.1 and higher)			
					D	C	B	A
DBB **100** MMC--> PLC				Axis number for handwheel 1 /H1/ (SW 2 and higher)				
	Machine axis	Handwheel selected	Contour handwheel	E	D	C	B	A
DBB **101** MMC--> PLC				Axis number for handwheel 2 /H1/ (SW 2 and higher)				
	Machine axis	Handwheel selected	Contour handwheel	E	D	C	B	A
DBB **102**				Axis number for handwheel 3 /H1/ (SW 4.1 and higher)				

MMC--> PLC	Machine axis	Handwheel selected	Contour handwheel	E	D	C	B	A
DBB **103** MMC--> PLC	MMC-101/102 battery alarm	MMC tempera-ture limit	AT box ready					Remote diagnosis active /FBFE/

General signals from NCK (DB 10)

DB 10	Signals from NC (NCK→PLC)							
Byte	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
DBB **104**	NCK CPU ready /A2/					HHU ready	MCP 2 ready	MCP 1 ready
DBB **105**	Unassigned							
DBB **106**							EMERGEN-CY STOP active /N2/	
DBB **107**	Inch system	NCU-link active					Probe actuated /M4/ Probe 2	Probe 1
DBB **108**	NC ready /A2/	Drive ready /FBA/	Drives in cyclic operation		MMC-CPU Ready (MMC to OPI) /A2/	MMC CPU Ready (MMC to MPI) /A2/	MMC2 CPU ready E_MMC2 Ready	
DBB **109**	NCK battery alarm /A2/	Air temp. alarm /A2/	Heat sink temp. alarm NCU 573	PC operating system fault				NCK alarm present /A2/
DBB **110**	Software cams minus (SW 2 and higher) /N3/							
	7	6	5	4	3	2	1	0
DBB **111**	Software cams minus (SW 2 and higher) /N3/							
	15	14	13	12	11	10	9	8
DBB **112**	Software cams minus (SW 4.1 and higher) /N3/							
	23	22	21	20	19	18	17	16
DBB **113**	Software cams minus (SW 4.1 and higher) /N3/							
	31	30	29	28	27	26	25	24
DBB **114**	Software cams plus (SW 2 and higher) /N3/							
	7	6	5	4	3	2	1	0
DBB **115**	Software cams plus (SW 2 and higher) /N3/							
	15	14	13	12	11	10	9	8
DBB **116**	Software cams plus (SW 4.1 and higher) /N3/							
	23	22	21	20	19	18	17	16
DBB **117**	Software cams plus (SW 4.1 and higher) /N3/							
	31	30	29	28	27	26	25	24

Note

Concerning **NCK CPU Ready (DBX 104.7):**

This signal is the sign-of-life monitoring function for the NC. It must be included in the safety circuit of the machine.

Concerning **MMC CPU1 READY (DBX 108.3 and DBX 108.2):**

If the MMC is connected to the operator panel interface (X 101), bit 3 is set (default). When connecting to the PG MPI interface (X 122), bit 2 is set.

External digital input and output signals of the NCK (DB 10)

DB 10	**Signals from NCK (NCK→PLC), /A2/ (SW2 and higher)**							
Byte	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
DBB **186**	Actual value of external digital NCK inputs							
	Input 16	Input 15	Input 14	Input 13	Input 12	Input 11	Input 10	Input 9
DBB **187**	Actual value of external digital NCK inputs							
	Input 24	Input 23	Input 22	Input 21	Input 20	Input 19	Input 18	Input 17
DBB **188**	Actual value of external digital NCK inputs							
	Input 32	Input 31	Input 30	Input 29	Input 28	Input 27	Input 26	Input 25
DBB **189**	Actual value of external digital NCK inputs							
	Input 40	Input 39	Input 38	Input 37	Input 36	Input 35	Input 34	Input 33
DBB **190**	NCK setpoint for external digital NCK outputs							
	Output 16	Output 15	Output 14	Output 13	Output 12	Output 11	Output 10	Output 9
DBB **191**	NCK setpoint for external digital NCK outputs							
	Output 24	Output 23	Output 22	Output 21	Output 20	Output 19	Output 18	Output 17
DBB **192**	NCK setpoint for external digital NCK outputs							
	Output 32	Output 31	Output 30	Output 29	Output 28	Output 27	Output 26	Output 25
DBB **193**	NCK setpoint for external digital NCK outputs							
	Output 40	Output 39	Output 38	Output 37	Output 36	Output 35	Output 34	Output 33

Analog input and output signals of the NCK (DB 10)

DB 10	Signals from NCK (NCK→PLC), /A2/ (SW2 and higher)							
Byte	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
DBW **194**	Actual value for analog input 1 of the NCK							
DBW **196**	Actual value for analog input 2 of the NCK							
DBW **198**	Actual value for analog input 3 of the NCK							
DBW **200**	Actual value for analog input 4 of the NCK							
DBW **202**	Actual value for analog input 5 of the NCK							
DBW **204**	Actual value for analog input 6 of the NCK							
DBW **206**	Actual value for analog input 7 of the NCK							
DBW **208**	Actual value for analog input 8 of the NCK							
DBW **210**	Setpoint for analog output 1 of the NCK							
DBW **212**	Setpoint for analog output 2 of the NCK							
DBW **214**	Setpoint for analog output 3 of the NCK							
DBW **216**	Setpoint for analog output 4 of the NCK							
DBW **218**	Setpoint for analog output 5 of the NCK							
DBW **220**	Setpoint for analog output 6 of the NCK							
DBW **222**	Setpoint for analog output 7 of the NCK							
DBW **224**	Setpoint for analog output 8 of the NCK							

3.2.9 Signals from/to mode group (DB 11)

Mode group-specific signals (DB 11)

DB 11	Signals to mode group 1 (PLC→NCK) /K1/							
Byte	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
DBB **0**	Mode group reset	Mode group stop Axes plus spindle	Mode group stop	Mode change disable		Operating mode JOG	MDA	AUTO-MATIC
DBB **1**	Single block Type A	Type B				Machine function REF	REPOS	TEACH IN
DBB **2**	Machine function		var. INC	10000 INC	1000 INC	100 INC	10 INC	1 INC
DBB **3**								

Note

about **machine function**: machine function defined centrally when signal "INC inputs in mode group area active" (DB10.DBX57.0) is set.

DB 11	Signals from mode group 1 (NCK→PLC) /K1/							
Byte	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
DBB **4** MMC--> PLC						Strobe mode JOG	MDA	AUTOM.
DBB **5** MMC--> PLC						Strobe machine function REF	REPOS	TEACH IN
DBB **6**	All channels in reset state				Mode group ready	Active operating mode JOG	MDA	AUTOM.
DBB **7**					Digitizing	Active machine function REF	REPOS	TEACH IN

DB 11	Signals to mode group 2 (PLC→NCK) /K1/							
Byte	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
DBB **20**	Mode group reset	Mode group stop Axes plus spindle	Mode group stop	Mode change disable		Operating mode JOG	MDA	AUTO-MATIC
DBB **21**	Single block Type A	Type B				Machine function REF	REPOS	TEACH IN
DBB **22**	Machine function		var. INC	10000 INC	1000 INC	100 INC	10 INC	1 INC
DBB **23**	Unassigned							

Note

about **machine function**: machine function defined centrally when signal "INC inputs in mode group area active" (DB10.DBX57.0) is set.

DB 11	Signals from mode group 2 (NCK→PLC) /K1/							
Byte	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
DBB **24** MMC--> PLC						Strobe mode JOG	MDA	AUTO-MATIC
DBB **25** MMC--> PLC						Strobe machine function REF	REPOS	TEACH IN
DBB **26**	All channels in reset state				Mode group ready	Active operating mode JOG	MDA	AUTO-MATIC
DBB **27**					Digitizing /FBD/ (SW 2 and higher)	Active machine function REF	REPOS	TEACH IN
DBB **28**	Machine functions		var. INC	10000 INC	1000 INC	100 INC	10 INC	1 INC

Note

The other mode groups (mode group 3 to mode group 10) are also located in DB 11 with the following initial bytes:

Mode group 3: DBB 40	Mode group 7: DBB 120
Mode group 4: DBB 60	Mode group 8: DBB 140
Mode group 5: DBB 80	Mode group 9: DBB 160
Mode group 6: DBB 100	Mode group 10: DBB 180

3.2.10 Signals for Safety SPL (safe programmable logic) (DB 18)

Parameterization section

References: /FBSI/, SINUMERIK Safety Integrated

DB 18	Signals for Safety SPL (PLC → PLC)							
Byte	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
DBB **0**	INSEP valid (valid bit)							
	8th input byte	7th input byte	6th input byte	5th input byte	4th input byte	3rd input byte	2nd input byte	1st input byte
DBB **1**								
DBB **2**	OUTSEP valid (valid bit)							
	8th output byte	7th output byte	6th output byte	5th output byte	4th output byte	3rd output byte	2nd output byte	1st output byte
DBB **3**								
DBW **4**	INSEP_ADDR (address 1st input byte)							
DBW **6**	INSEP_ADDR (address 2nd input byte)							
DBW **8**	INSEP_ADDR (address 3rd input byte)							
DBW **10**	INSEP_ADDR (address 4th input byte)							
DBW **12**	INSEP_ADDR (address 5th input byte)							
DBW **14**	INSEP_ADDR (address 6th input byte)							
DBW **16**	INSEP_ADDR (address 7th input byte)							
DBW **18**	INSEP_ADDR (address 8th input byte)							

DBW **20**	OUTSEP_ADDR (address 1st output byte)							
DBW **22**	OUTSEP_ADDR (address 2nd output byte)							
DBW **24**	OUTSEP_ADDR (address 3rd output byte)							
DBW **26**	OUTSEP_ADDR (address 4th output byte)							
DBW **28**	OUTSEP_ADDR (address 5th output byte)							
DBW **30**	OUTSEP_ADDR (address 6th output byte)							
DBW **32**	OUTSEP_ADDR (address 7th output byte)							
DBW **34**	OUTSEP_ADDR (address 8th output byte)							
DBB **36**							Stop E	SPL READY
DBB **37**								

Data area / error

DB 18	**Signals for Safety SPL (PLC ←→ NCK)**							
Byte	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
	Data area for SPL inputs/outputs							
DBD **38**	SPL_DATA.INSEP [1..32]							
DBD **42**	SPL_DATA.INSEP [33..64]							
DBD **46**	SPL_DATA.OUTSEP [1..32]							
DBD **50**	SPL_DATA.OUTSEP [33..64]							
	Data area for user SPL							
DBD **54**	SPL_DATA.INSIP [1..32]							
DBD **58**	SPL_DATA.INSIP [33..64]							
DBD **62**	SPL_DATA.OUTSIP [1..32]							
DBD **66**	SPL_DATA.OUTSIP [33..64]							

DB 18	Signals for Safety SPL (PLC ←→ NCK)							
Byte	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
DBD **70**	SPL_DATA.MARKERSIP [1..32]							
DBD **74**	SPL_DATA.MARKERSIP [33..64]							
	Difference in level between NCK and PLC for diagnostics							
DBD **78**	SPL_DELTA.INSEP [1..32]							
DBD **82**	SPL_DELTA.INSEP [33..64]							
DBD **86**	SPL_DELTA.OUTSEP [1..32]							
DBD **90**	SPL_DELTA.OUTSEP [33..64]							
DBD **94**	SPL_DELTA.INSIP [1..32]							
DBD **98**	SPL_DELTA.INSIP [33..64]							
DBD **102**	SPL_DELTA.OUTSIP [1..32]							
DBD **106**	SPL_DELTA.OUTSIP [33..64]							
DBD **110**	SPL_DELTA.MARKERSIP [1..32]							
DBD **114**	SPL_DELTA.MARKERSIP [33..64]							
DBD **118**								CMDSI
DBD **119**			xxxxxxxx					
DBD **120**	Error number 0 = no error 1 - 320 = Signal number starting from SPL_DATA.INSEP [1]							
DBD **124**	Level indicator of cross-checking (diagnostics option: how many SPL signals currently differ in level)							

Supplementary data areas

DB 18	Signals for Safety SPL (PLC ←→ NCK)							
Byte	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
	Data area for single-channel inputs/outputs							
DBB **128**	$A_PLCSIOUT [1 .. 8]							
DBB **129**	$A_PLCSIOUT [9 .. 16]							
DBB **130**	$A_PLCSIOUT [17 .. 24]							
DBB **131**	$A_PLCSIOUT [25 .. 32]							
DBB **132**	$A_PLCSIIN [1.. 8]							
DBB **133**	$A_PLCSIIN [9 .. 16]							
DBB **134**	$A_PLCSIIN [17 .. 24]							
DBB **135**	$A_ PLCSIIN [25 .. 32]							
DBB **136**	SPL status							
DBB **138**	PROFIsafe module(s) for 8th input byte	7th input byte	6th input byte	5th input byte	4th input byte	3rd input byte	2nd input byte	1st input byte
DBB **139**								
DBB **140**	PROFIsafe module(s) for 8th output byte	7th output byte	6th output byte	5th output byte	4th output byte	3rd output byte	2nd output byte	1st output byte
DBB **141**								
	Teststop data							
DBB **142** to DBB **149**	Number of axes per TeststopBlock 1 (NoOfAxisPerBlock[1]) Number of axes per TeststopBlock 8 (NoOfAxisPerBlock[8])							
DBB **150** to DBB **157**	Pointer onto axis table 1 (BlockPointer[1]) Pointer onto axis table 8 (BlockPointer[8])							
DBB **158** to DBB **188**	Safety axis table (AxisTable[1]) 1st axis Safety axis table (AxisTable[31]) 31st axis							

3.2.11 Signals from/to operator panel (DB 19)

DB 19	Signals to operator panel (PLC→MMC)							
Byte	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
DBB **0**	Actual value in WCS 0=MCS /A2/	Back up travel recorder	MMC shutdown (for OEM users)	Clear recall alarms	Clear cancel alarms	Key disable /A2/	Screen darkening /A2/	Screen bright /A2/
DBB **1**	Reserved							
DBW **2**	Higraph first error display							
DBW **4**	Higraph first error display							
DBB **6**	Analog spindle 1, capacity in percent							
DBB **7**	Analog spindle 2, capacity in percent							
DBB **8**	Channel number of machine control panel to MMC							
DBB **9**	Reserved for selection					Automatc tool measure-ment	OEM2	OEM1
DBB **10**	PLC Hardkeys (Values 1 ... 255, Default: 0)							
DBB **11**	Reserved for hardkey function expansions							
DBB **12**	RS-232 On /A2/	RS-232 Off /A2/	RS-232 External /A2/	RS-232 Stop /A2/	COM1 /A2/	COM2 /A2/	Res.	Res.
DBB **13**	Select /A2/	Load part program /A2/	Unload /A2/	Res.				Disable Teach transfer
DBB **14**	0=act. FS 1=pas. FS	RS-232 act. FS: Index of file to be transferred in the standard list. RS-232 pass. FS: Number of the control file for user file names.						
DBB **15**	RS-232 act. FS: Index that specifies the axis, channel or tool no. RS-232 pass. FS: Index of the file to be transferred in the user list							
DBB **16**	1=pas FS	Part program handling: Number of the control file for user file names.						
DBB **17**	Part program handling: Index of the file to be transferred in the user list							
DBB **18**								TO comp.
DBB **19**	Reserved (signal counter)							

DB 19	Signals from operator panel (MMC → PLC)							
Byte	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
DBB **20**	MCS/WCS Change-over /A2/	Simulation active /A2/	Language 2 switched HMI Emb.	Recall alarm cleared MMC 103 /A2/	Cancel alarm cleared MMC 103 /A2/	Cancel key actuated /A2/	Screen is dark /A2/	
DBB **21**	Active MMC operating area							
DBB **22**	Displayed channel number from the MMC /A2/							
DBB **23**							Counter spindle internal voltage	Main spindle internal voltage
DBB **24**	RS-232 status from PLC							
	RS-232 On /A2/	RS-232 Off /A2/	RS-232 External /A2/	RS-232 Stop /A2/	Com1 active /A2/	Com2 active /A2/	OK /A2/	Error /A2/
DBB **25**	Error RS-232 /A2/							
DBB **26**	Part program handling status /A2/							
	Select	Load	Unload		Active	Error MMC 5.3 and higher; 6.1	OK	Error
DBB **27**	Error program handling /A2/							
DBW **28**	Mask number for "Extend user interface" /IAM/, BE1							
DBB **30**	Control bits PLC --> MMC							
							Exit mask	Request mask
DBB **31**	Control bits PLC --> MMC							
	Inactive bit			Error, Not possible to request mask	Mask exited	Mask active	Mask requested	Mask request accepted
DBB **32** PLC>MMC			FunctionSelectionNo. from PLC					
	Busy function	Strobe function						
DBB **33** PLC>MMC	Parameter 1 for FunctionSelectionNo. (function selection from DBB 32)							
DBB **34** PLC>MMC	Parameter 2 for FunctionSelectionNo. (function selection from DBB 32)							

DBB **35** PLC>MMC	Parameter 3 for FunctionSelectionNo. (function selection from DBB 32)							
DBB **36** MMC>PLC	Error code for FunctionSelectionNo. (function selection from DBB 32)							
DBB **37** MMC>PLC	Parameter 1 for FunctionSelectionNo. (function selection from DBB 48)							
DBB **38** MMC>PLC	Parameter 2 for FunctionSelectionNo. (function selection from DBB 48)							
DBB **39** MMC>PLC	Parameter 3 for FunctionSelectionNo. (function selection from DBB 48)							

DBB40-47 used internally

DBB **48** MMC>PLC	PLC-Busy function	HMI-Strobe function	FunctionSelectionNo. from MMC					
DBB **49** PLC>MMC	Error code for FunctionSelectionNo. (function selection from DBB48)							

Interface 2nd MMC	
DBB **50-99**	Assignment as for DBB 0 to DBB 49 Switchover interface to MMC
	Knocking interface (MMC announces itself to NCU)
DBW **100**	ONL_REQUEST /B3/ Online request from MMC MMC writes its client identification as online request (bit 8-15: bus type, bit 0-7: MMC bus address)

DBW **102**	ONL_CONFIRM /B3/ Acknowledgment from PLC to online request PLC writes MMC client identification as acknowledgment (bus type, MMC bus address; as with DBW 100).
DBW **104**	PAR_CLIENT_IDENT /B3/ MMC writes its client identification (bus type, MMC bus address; as with DBW 100).
DBB **106**	PAR_MMC_TYP /B3/ Type of MMC as per NETNAMES.INI: Main / subordinate operator panel / server /...
DBB **107**	PAR_MSTT_ADR /B3/ MMC writes address of MCP to be activated; 255, when no MCP activated
DBB **108**	PAR_STATUS /B3/ PLC writes online enable for MMC.

DBB **109**	PAR_Z_INFO /B3/ PLC writes additional info about status							
DBB **110**	M_TO_N_ALIVE Sign of life from PLC to MMC through M to N block							

	Online interface MMC 1 (user)							
DBW **120**	MMC1_CLIENT_IDENT /B3/ PLC writes PAR_CLIENT_IDENT to MMCx_CLIENT_IDENT when MMC goes online.							
DBB **122**	MMC1_TYP /B3/ PLC writes PAR_MMC_TYP to MMCx_TYP when MMC goes online.							
DBB **123**	MMC1_MSTT_ADR /B3/ PLC writes PAR_MSTT_ADR to MMCx_MSTT_ADR when MMC goes online.							
DBB **124**	MMC1_STATUS /B3/ Connection status, MMC and PLC alternately write their requests/acknowledgments							
DBB **125**	MMC1_Z_INFO /B3/ Additional info connection status (pos./neg. acknowledgment, error messages...)							
DBB **126**			MMC1 ACTIVE DENIED /B3/	MMC1 ACTIVE CHANGED /B3/	MMC1 ACTIVE PERM /B3/	MMC1 ACTIVE REQ /B3/	MMC1 MCP SHIFT LOCK /B3/	MMC1 SHIFT LOCK /B3/
DBB **127-129**	Reserved							

Online Interface MMC 2 (user)								
DBW **130**	MMC2_CLIENT_IDENT /B3/ PLC writes PAR_CLIENT_IDENT to MMCx_CLIENT_IDENT when MMC goes online.							
DBB **132**	MMC2_TYP /B3/ PLC writes PAR_MMC_TYP to MMCx_TYP when MMC goes online.							
DBB **133**	MMC2_MSTT_ADR /B3/ PLC writes PAR_MSTT_ADR to MMCx_MSTT_ADR when MMC goes online.							
DBB **134**	MMC2_STATUS /B3/ Connection status, MMC and PLC alternately write their requests/acknowledgments							
DBB **135**	MMC2_Z_INFO /B3/ Additional info connection status (pos./neg. acknowledgment, error messages...)							
DBB **136**			MMC2 ACTIVE DENIED /B3/	MMC2 ACTIVE CHANGED /B3/	MMC2 ACTIVE PERM /B3/	MMC2 ACTIVE REQ /B3/	MMC2 MCP SHIFT LOCK /B3/	MMC2 SHIFT LOCK /B3/
DBB **137-139**	Reserved							
DBB **140-197**	Code carrier input parameters Optional package SINTDC on HMI-Advanced required							
DBB **198-249**	Code carrier return parameters Optional package SINTDC on HMI-Advanced required							
DBB **250-255**	Commands Optional package SINTDC on HMI-Advanced required							
DBB **256-267**	Commands for Paramtm.exe Optional package SINTDC on HMI-Advanced required							
DBW 268	Traffic light status Optional package TPM on HMI Advanced required							
DBW 270 to 394	Counter[1 ... 32] Optional package TPM on HMI Advanced required							

3.2.12 PLC machine data (DB 20)

DB 20	PLC machine data (PLC→operator)							
Byte	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
DBW 0	INT values							
DBW								
DBW	INT values							
DBB	Bit arrays							
DBB								
DBB	Bit arrays							
DBD	REAL values							
DBD								
DBD	REAL values							

Note

The initial and end addresses of the PLC machine data areas depend on the respective length indications of the partial areas. In general, the integer values start with the data byte 0. The upper limit is determined by the corresponding length indication. In general, the following bit arrays (2-decade hexadecimal numbers on input) start with the following even address. The real values follow directly the bit arrays and also start with an even address.

3.2.13 Signals from/to NCK channel (DB 21–30)

DB 21 - 30	Signals to NCK channel (PLC→NCK)							
Byte	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
DBB **0**		Activate dry run feedrate /V1/	Activate M01 /K1/	Activate single block /K1/	Activate DRF /H1/			
DBB **1**	Activate program test /K1/	PLC action complete /K1/	CLC override /TE1/	CLC stop /TE1/	Time monitoring act. (tool manage-ment)	Synchron-ized action OFF /FBSY/	Enable protection zones /A3/	Activate referencing /R1/
DBB **2**	Skip block /K1/							
	/7	/6	/5	/4	/3	/2	/1	/0
DBB **3**	Nibbling and punching /N4/							
			Manual release of stroke 2	Stroke delayed /N4/	Stroke not operating /N4/	Stroke suppres-sion /N4/	Manual stroke enable /N4/	Stroke enable /N4/
DBB **4**	Feedrate override /V1/							
	H	G	F	E	D	C	B	A
DBB **5**	Rapid traverse override /V1/							
	H	G	F	E	D	C	B	A
DBB **6**	Feedrate override active /V1/	Rapid traverse override active /V1/		Program level abort /K1/	Delete subroutine no. of passes	Delete distance-to-go /A2/	Read-in disable /K1/	Feed disable /V1/
DBB **7**	Reset /K1/			NC Stop axes plus spindle /K1/	NC Stop /K1/	NC Stop to block limit /K1/	NC Start /K1/	NC Start disable /K1/
DBB **8**	Activate machine-related protection area /A3/ (SW 2 and higher)							
	Area 8	Area 7	Area 6	Area 5	Area 4	Area 3	Area 2	Area 1
DBB **9**	Activate machine-related protection area /A3/ (SW 2 and higher)							
							Area 10	Area 9
DBB **10**	Activate channel-specific protection area /A3/ (SW 2 and higher)							
	Area 8	Area 7	Area 6	Area 5	Area 4	Area 3	Area 2	Area 1
	Activate channel-specific protection area /A3/ (SW 2 and higher)							
DBB **11**							Area 10	Area 9

Note

on **Feedrate override active (DBX6.7)**
even if feedrate override is not active (= 100%), the setting 0% is effective.
on **Feedrate override (DBB 4)**
either 31 positions (Gray code) with 31 MD for % evaluation or 0-200% corresponding to the dual value in byte (201–255 ⇒ max. 200%).
on **Rapid traverse override (DBB 5)**
either 31 positions (Gray code) with 31 MD for % evaluation or 0-100% corresponding to the dual value in byte (101–255 ⇒ max. 100%).
on **Activate single block (DBX0.4)**
select variant via ”Write variable”.
on **Delete distance-to-go (DBX6.2)**
effects only path axes and not positioning axes

Control signals to geometry axes

DBB **12**	Geometry axis 1							
	Traversing keys /H1/		Rapid traverse override /H1/	Traversing key disable /H1/	Feed stop /V1/	Activate handwheel /H1/		
	+	-				3	2	1
DBB **13**	Geometry axis 1 machine function /H1/							
			Var. INC	10000 INC	1000 INC	100 INC	10 INC	1 INC
DBB **14**	OEM signals geometry axis 1							
DBB **15**	Geometry axis 1							
DBB **16**	Geometry axis 2							
	Traversing keys /H1/		Rapid traverse override /H1/	Traversing key disable /H1/	Feed stop /V1/	Activate handwheel /H1/		
	+	-				3	2	1
DBB **17**	Geometry axis 2 machine function /H1/							
			Var. INC	10000 INC	1000 INC	100 INC	10 INC	1 INC
DBB **18**	OEM signals geometry axis 2							
DBB **19**	Geometry axis 2							
DBB **20**	Geometry axis 3							
	Traversing keys /H1/		Rapid traverse override /H1/	Traversing key disable /H1/	Feed stop /V1/	Activate handwheel /H1/		
	+	-				3	2	1
DBB **21**	Geometry axis 3 machine function /H1/							
			Var. INC	10000 INC	1000 INC	100 INC	10 INC	1 INC

DBB 22	OEM signals geometry axis 3							
DBB 23	Geometry axis 3							

Note

about **machine function**: machine function only defined when signal "INC inputs in mode group area active" (DB10.DBX57.0) is not set.

Operating signals from MMC/status signals from NC channel

DB 21-30	**Signals from NCK channel (NCK→PLC, MMC→PLC, PLC→NCK)**							
Byte	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
DBB 24 MMC--> PLC		Dry run feedrate selected /V1/	M01 selected /K1/	Select NCK-related M01	DRF selected /H1/			
DBB 25 MMC--> PLC	Program test selected /K1/			REPOS MODE EDGE	Feedrate override for rapid traverse selected /V1/	REPOSPATHMODE 2	1	0
DBB 26 MMC--> PLC	Skip block selected /K1/ (SW 2 and higher) 7	6	5	4	3	2	1	0
DBB 27 MMC--> PLC							Skip block selected /K1/ (SW 2 and higher)	Skip block selected /K1/ (SW 2 and higher)
DBB 28 PLC→NCK	OEM channel signals							
DBB 29 PLC→NCK	Do not disable tool	Switch off wear monitoring	Switch off workpiece counter	Activate PTP motion	Activate fixed feed 4 /FBMA/, /V1/ (SW 4 and higher)	Activate fixed feed 3 /FBMA/, /V1/ (SW 4 and higher)	Activate fixed feed 2 /FBMA/, /V1/ (SW 4 and higher)	Activate fixed feed 1 /FBMA/, /V1/ (SW 4 and higher)

DBB **30** PLC→NCK						Activate contour handwheel		
	No tool change commands		Activate NCK-related M01 /H2/	Neg. direction simulation contour handwheel	Simulation contour handwheel on	Handwheel 3	Handwheel 2	Handwheel 1
DBB **31** PLC→NCK							REPOSPATHMODE	
	Skip block active /9	Skip block active /8		REPOS MODE EDGE		2	1	0
DBB **32** NCK--> PLC		Last action block active /K1/	M00/M01 active /K1/	Approach block active /K1/	Action block active /K1/			Execution from external source active
DBB **33** NCK--> PLC	Program test active /K1/	Transforma tion active /K1/M1	M02/M30 active /K1/	Block search active /K1/	Handwheel override active (SW 2 and higher) /H1/	Revolutio-nal feedrate active /V1/	Orientable toolholder active	Referencing active /R1/
DBB **34** NCK--> PLC				OEM channel signals feedback				
DBB **35** NCK--> PLC		Channel status /K1/				Program status /K1/		
	Reset	Interrupted	Active	Aborted	Interrupted	Stopped	Waiting	Running
DBB **36** NCK--> PLC	NCK alarm with processing stop present /A2/	Channel-specific NCK alarm present /A2/	Channel ready for operation in SW 4 and higher	Interrupt processing active /K1/	All axes stationary /B1/	All axes requiring reference points are referenced /R1/		
DBB **37** NCK--> PLC	Stop	Read-in	CLC	CLC	CLC		Contour handwheel active	
	at block end with SBL is sup-pressed	enable is ignored	stopped upper limit /TE1/	stopped lower limit /TE1/	active /TE1/	Handwheel 3 /H1/	Handwheel 2 /H1/	Handwheel 1 /H1/
DBB **38** NCK--> PLC				Nibbling and punching /N4/				
							Acknowl. manual stroke enable /N4/	Stroke enable active /N4/
DBB **39** NCK--> PLC								Protection zones not guaranteed

Note

on **Feedrate override for rapid traverse selected (DBX25.3)**

Depending on this signal, the basic PLC program copies the feedrate override onto the rapid traverse override on the channel-specific interface.

On **Program test selected (DBX25.7)**

"Program test selected" means axis disable for all channel axes and spindles.

Status signals of geometry axes

DB 21 - 30	Signals from NCK channel (NCK→PLC)							
Byte	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
DBB **40**	Geometry axis 1							
	Traverse command /H1/		Travel requests			Handwheel active /H1/		
	plus	minus	plus	minus		3	2	1
DBB **41**	Geometry axis 1 active machine function /H1/							
			Var. INC	10000 INC	1000 INC	100 INC	10 INC	1 INC
DBB **42**	OEM signals geometry axis 1							
DBB **43**	Geometry axis 1							
DBB **44** MMC--> PLC								
DBB **46**	Geometry axis 2							
	Traverse command /H1/		Travel requests			Handwheel active /H1/		
	plus	minus	plus	minus		3	2	1
DBB **47**	Geometry axis 2 active machine function /H1/							
			Var. INC	10000 INC	1000 INC	100 INC	10 INC	1 INC
DBB **48**	OEM signals geometry axis 2							
DBB **49**	Geometry axis 2							
DBB **50** MMC--> PLC								
DBB **52**	Geometry axis 3							
	Traverse command /H1/		Travel requests			Handwheel active /H1/		
	plus	minus	plus	minus		3	2	1
DBB **53**	Geometry axis 3 active machine function /H1/							
			Var. INC	10000 INC	1000 INC	100 INC	10 INC	1 INC

DBB **54**	OEM signals geometry axis 3							
DBB **55**	Geometry axis 3							
DBB **56** **MMC--> PLC**								
DBB **57**								

Change signals on auxiliary function transfer from NC channel

DB 21 - 30	**Signals from NCK channel (NCK→PLC)**							
Byte	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
DBB **58**				M fct. 5 change /H2/	M fct. 4 change /H2/	M fct. 3 change /H2/	M fct. 2 change /H2/	M fct. 1 change /H2/
DBB **59**				M fct. 5 not decoded	M fct. 4 not decoded	M fct. 3 not decoded	M fct. 2 not decoded	M fct. 1 not decoded
DBB **60**		S fct. 3 quick	S fct. 2 quick	S fct. 1 quick		S fct. 3 change /H2/	S fct. 2 change /H2/	S fct. 1 change /H2/
DBB **61**		T fct 3 quick	T fct. 2 quick	T fct. 1 quick		T fct. 3 change/H2/ (SW 2 and higher)	T fct. 2 change/H2/ (SW 2 and higher)	T fct. 1 change /H2/
DBB **62**		D fct. 3 quick	D fct. 2 quick	D fct. 1 quick		D fct. 3 change/H2/ (SW 2 and higher)	D fct. 2 change/H2/ (SW 2 and higher)	D fct. 1 change /H2/
DBB **63**				DL fct. quick				DL fct. change
DBB **64**		H fct. 3 quick	H fct. 2 quick	H fct. 1 quick		H fct. 3 change /H2/	H fct. 2 change /H2/	H fct. 1 change /H2/
DBB **65**			F fct. 6 change /H2/	F fct. 5 change /H2/	F fct. 4 change /H2/	F fct. 3 change /H2/	F fct. 2 change /H2/	F fct. 1 change /H2/
DBB **66**				M fct. 5 quick	M fct. 4 quick	M fct. 3 quick	M fct. 2 quick	M fct. 1 quick
DBB **67**			F fct. 6 quick	F fct. 5 quick	F fct. 4 quick	F fct. 3 quick	F fct. 2 quick	F fct. 1 quick

Note

For 10-decade T numbers, only the T fct. 1 change signal is available. For 5-decade D numbers, only the D fct. 1 change signal is available.

Transferred M/S functions

DB 21 - 30	Signals from NCK channel (NCK→PLC)							
Byte	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
DBW **68**	Extended address M function 1 (binary) /H2/							
DBD **70**	M function 1 (binary) /H2/							
DBW **74**	Extended address M function 2 (binary) /H2/							
DBD **76**	M function 2 (binary) /H2/							
DBW **80**	Extended address M function 3 (binary) /H2/							
DBD **82**	M function 3 (binary) /H2/							
DBW **86**	Extended address M function 4 (binary) /H2/							
DBD **88**	M function 4 (binary) /H2/							
DBW **92**	Extended address M function 5 (binary) /H2/							
DBD **94**	M function 5 (binary) /H2/							
DBW **98**	Extended address S function 1 (binary) /H2/							
DBD **100**	S function 1 (REAL format) /H2/							
DBW **104**	Extended address S function 2 (binary) /H2/							
DBD **106**	S function 2 (REAL format) /H2/							
DBW **110**	Extended address S function 3 (binary) /H2/							
DBD **112**	S function 3 (REAL format) /H2/							

Note

M functions are programmed in the part program in the INTEGER format (8 decades plus sign).

”REAL format” means: 24 bit mantissa and 8 bit exponent.

Transferred T/D/DL functions

DB 21 - 30	Signals from NCK channel (NCK→PLC)							
Byte	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
DBW **116**	Extended address T function 1 (16 bit Int)							
DBW **118** DBD **118**	T function 1 (binary) /H2/ For 8-decade T nos., T function 1 (32 bit DINT) is used in DBD 118 (see note)							
DBW **120**	Extended address T function 2 (16 bit Int)							
DBW **122**	T function 2 (Int)							
DBW **124**	Extended address T function 3 (16 bit Int)							
DBW **126**	T function 3 (Int)							
DBB **128**								
DBB **129**	D function 1 (binary) /H2/							
DBW **130** DBB **130**	For 5-decade D nos., D function 1 (16 bit DINT) is used in DBD 130 (see note) Extended address D function 2 (8 bit Int)							
DBB **131**	D function 2 (8 bit Int)							
DBB **132**	Extended address D function 3 (8 bit Int)							
DBB **133**	D function 3 (8 bit Int)							
DBW **134**	Extended address DL function (16 bit Int)							
DBD **136**	DL function (REAL)							

Note

With active tool management, programmed T functions are **not** output to the PLC.

8-decade T nos. are only available as T function 1

Programmed D functions with names (e.g. D=CUTEDGE_1) **cannot** be output in ASCII format to the PLC.

5-decade D nos. are only available as D function 1

The REAL format corresponds to floating point representation in STEP 7 (24 bit mantissa and 8 bit exponent). This floating point format supplies a maximum of 7 valid places.

Transferred H/F functions

DB 21 - 30	Signals from NCK channel (NCK→PLC)							
Byte	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
DBW **140**	Extended address H function 1 (binary) /H2/							
DBD **142**	H function 1 (REAL or Dint) /H2/							
DBW **146**	Extended address H function 2 (binary) /H2/							
DBD **148**	H function 2 (REAL or Dint) /H2/							
DBW **152**	Extended address H function 3 (binary) /H2/							
DBD **154**	H function 3 (REAL or Dint) /H2/							
DBW **158**	Extended address F function 1 (binary) /H2/							
DBD **160**	F function 1 (REAL format) /H2/							
DBW **164**	Extended address F function 2 (binary) /H2/							
DBD **166**	F function 2 (REAL format) /H2/							
DBW **170**	Extended address F function 3 (binary) /H2/							
DBD **172**	F function 3 (REAL format) /H2/							
DBW **176**	Extended address F function 4 (binary) /H2/							
DBD **178**	F function 4 (REAL format) /H2/							
DBW **182**	Extended address F function 5 (binary) /H2/							
DBD **184**	F function 5 (REAL format) /H2/							
DBW **188**	Extended address F function 6 (binary) /H2/							
DBD **190**	F function 6 (REAL format) /H2/							

Note

F functions are programmed in the part program in the REAL format.

The extended address of the F function contains an identifier with the following meaning:
0 = path feed,
1-31 = machine axis number for feed with positioning axes.

The H function data type is dependent on MD 22110: AUXFU_H_TYPE_INT.

Decoded M signals (M0–M99)

DB 21 - 30	Signals from NCK channel (NCK→PLC)							
Byte	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
DBB **194**	Dynamic M functions							
	M07	M06	M05 *	M04 *	M03 *	M02	M01	M00
DBB **195**	Dynamic M functions /H2/							
	M15	M14	M13	M12	M11	M10	M09	M08
DBB **196**	Dynamic M functions /H2/							
	M23	M22	M21	M20	M19	M18	M17	M16
DBB **197**	Dynamic M functions /H2/							
	M31	M30	M29	M28	M27	M26	M25	M24
DBB **198**	Dynamic M functions /H2/							
	M39	M38	M37	M36	M35	M34	M33	M32
DBB **199**	Dynamic M functions /H2/							
	M47	M46	M45	M44	M43	M42	M41	M40
DBB **200**	Dynamic M functions /H2/							
	M55	M54	M53	M52	M51	M50	M49	M48
DBB **201**	Dynamic M functions /H2/							
	M63	M62	M61	M60	M59	M58	M57	M56
DBB **202**	Dynamic M functions /H2/							
	M71	M70 *	M69	M68	M67	M66	M65	M64
DBB **203**	Dynamic M functions /H2/							
	M79	M78	M77	M76	M75	M74	M73	M72
DBB **204**	Dynamic M functions /H2/							
	M87	M86	M85	M84	M83	M82	M81	M80
DBB **205**	Dynamic M functions /H2/							
	M95	M94	M93	M92	M91	M90	M89	M88
DBB **206**	Dynamic M functions /H2/							
					M99	M98	M97	M96
DBB **207**								

Note

M functions marked with * are not decoded in this bit array if a spindle is configured in the channel. In this case, these M functions are offered as extended M functions in DB21-30.DBB68 ff. and in the relevant axis DB DB31-61.DBB86 ff.

Dynamic M functions (M00 to M99) are decoded by the basic PLC program.

The PLC user must use dynamic M functions in order to generate static M functions.

Active G functions

DB 21 - 30	Signals from NCK channel (NCK→PLC)							
Byte	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
DBB **208**	Number of active G function of G function group 1 (binary) /K1/							
DBB **209**	Number of active G function of G function group 2 (binary) /K1/							
DBB **210**	Number of active G function of G function group 3 (binary) /K1/							
DBB **211**	Number of active G function of G function group 4 (binary) /K1/							
DBB **212**	Number of active G function of G function group 5 (binary) /K1/							
DBB **213**	Number of active G function of G function group 6 (binary) /K1/							
DBB **214**	Number of active G function of G function group 7 (binary) /K1/							
DBB **215**	Number of active G function of G function group 8 (binary) /K1/							
...								
DBB **270**	Number of active G function of G function group n-1 (binary) /K1/							
DBB **271**	Number of active G function of G function group n (binary) /K1/							

Note

The active G functions of the groups are updated each time a G function or a mnemonic identifier (e.g. SPLINE) is programmed.

G functions within a G group are output as binary value, starting with 1.

A G function with the value 0 means that no G function is active for this G group.

Signals for protection areas from NC channel

DB 21 - 30	Signals from NCK channel (NCK→PLC) (SW 2 and higher)							
Byte	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
DBB **272**	Machine-related protection area preactivated /A3/							
	Area 8	Area 7	Area 6	Area 5	Area 4	Area 3	Area 2	Area 1
DBB **273**	Machine-related protection area preactivated /A3/							
							Area 10	Area 9
DBB **274**	Channel-specific protection area preactivated /A3/							
	Area 8	Area 7	Area 6	Area 5	Area 4	Area 3	Area 2	Area 1
DBB **275**	Channel-specific protection area preactivated /A3/							
							Area 10	Area 9
DBB **276**	Machine-related protection area violated /A3/							
	Area 8	Area 7	Area 6	Area 5	Area 4	Area 3	Area 2	Area 1
DBB **277**	Machine-related protection area violated /A3/							
							Area 10	Area 9
DBB **278**	Channel-related protection area violated /A3/							
	Area 8	Area 7	Area 6	Area 5	Area 4	Area 3	Area 2	Area 1
DBB **279**	Channel-related protection area violated /A3/							
							Area 10	Area 9

Instruction-controlled signals to NC channel

DB 21 - 30	Signals to NCK channel (NCK→PLC) (SW 4 and higher)							
Byte	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
DBB **280**							Synch. action disable request	Reserved
DBB **281**							Synch. action disabled	
DBW **282**	Reserved							
DBW **284**	Reserved							
DBW **286**	Reserved							
DBW **288**	Reserved							
DBW **290**	Reserved							
DBW **292**	Reserved							
DBW **294**	Reserved							
DBW **296**	Reserved							
DBW **298**	Reserved							
DBB **300**	Disable synchronized actions /FBSY/							
	No. 8	No. 7	No. 6	No. 5	No. 4	No. 3	No. 2	No. 1
DBB **301**	Disable synchronized actions /FBSY/							
	No. 16	No. 15	No. 14	No. 13	No. 12	No. 11	No. 10	No. 9
DBB **302**	Disable synchronized actions /FBSY/							
	No. 24	No. 23	No. 22	No. 21	No. 20	No. 19	No. 18	No. 17
DBB **303**	Disable synchronized actions /FBSY/							
	No. 32	No. 31	No. 30	No. 29	No. 28	No. 27	No. 26	No. 25
DBB **304**	Disable synchronized actions /FBSY/							
	No. 40	No. 39	No. 38	No. 37	No. 36	No. 35	No. 34	No. 33
DBB **305**	Disable synchronized actions /FBSY/							
	No. 48	No. 47	No. 46	No. 45	No. 44	No. 43	No. 42	No. 41
DBB **306**	Disable synchronized actions /FBSY/							
	No. 56	No. 55	No. 54	No. 53	No. 52	No. 51	No. 50	No. 49
DBB **307**	Disable synchronized actions /FBSY/							
	No. 64	No. 63	No. 62	No. 61	No. 60	No. 59	No. 58	No. 57

Note

The request signals are set by the user and reset by the basic program after transmission of the corresponding data.

Instruction-controlled signals from NC channel

DB 21 - 30	Signals from NCK channel (NCK→PLC) (SW 4 and higher)							
Byte	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
DBB **308**	Synchronized actions can be disabled /FBSY/							
	Nr. 8	Nr. 7	Nr. 6	Nr. 5	Nr. 4	Nr. 3	Nr. 2	Nr.1
DBB **309**	Synchronized actions can be disabled /FBSY/							
	Nr. 16	Nr. 15	Nr. 14	Nr. 13	Nr. 12	Nr. 11	Nr.10	Nr.9
DBB **310**	Synchronized actions can be disabled /FBSY/							
	Nr. 24	Nr. 23	Nr. 22	Nr. 21	Nr. 20	Nr. 19	Nr.18	Nr.17
DBB **311**	Synchronized actions can be disabled /FBSY/							
	Nr. 32	Nr. 31	Nr. 30	Nr. 29	Nr. 28	Nr. 27	Nr. 26	Nr.25
DBB **312**	Synchronized actions can be disabled /FBSY/							
	Nr. 40	Nr. 39	Nr. 38	Nr. 37	Nr. 36	Nr. 35	Nr. 34	Nr. 33
DBB **313**	Synchronized actions can be disabled /FBSY/							
	Nr. 48	Nr. 47	Nr. 46	Nr. 45	Nr. 44	Nr. 43	Nr. 42	Nr.41
DBB **314**	Synchronized actions can be disabled /FBSY/							
	Nr. 56	Nr. 55	Nr. 54	Nr. 53	Nr. 52	Nr. 51	Nr. 50	Nr.49
DBB **315**	Synchronized actions can be disabled /FBSY/							
	Nr. 64	Nr. 63	Nr. 62	Nr. 61	Nr. 60	Nr. 59	Nr. 58	Nr.57
Cyclic Signals interface NCK → PLC								
DBB **316**	Active G functions							
							G0 PATH	G00 geo.
DBB **317**	Tool missing	PTP motion active	Travel request drive test				Workpiece setpoint reached	External language mode active
DBB **318**	Overstore active	Dry-run feedrate active /V1/	Associated M01 active /H2/	Stop delayed	TOFF movement active	TOFF active	Search active /K1/	ASUP stopped /K1/
DBB **319**	No tool change command active	Stop-delay-range not activated	Repos DEFERRA L Chan	Delay FTS	Repos Path Mode Ackn 2	Repos Path Mode Ackn 1	Repos Path Mode Ackn 0	REPOS MODE EDGE ACKN

Signals to orientation axes

DB 21 - 30	Signals to NCK channel (PLC→NCK)							
Byte	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
DBB **320**	Traversing keys		**Orientation axis 1**					
	+	–	Rapid traverse override	Traversing key disable	Feed stop	Activate handwheel (bit value coding)		
DBB **321**	**Orientation axis 1**							
			var. INC	10000 INC	1000 INC	100 INC	10 INC	1 INC
DBB **322**	**OEM signals orientation axis 1**							
DBB **323**	**Orientation axis 1**							
			var. INC	10000 INC	1000 INC	100 INC	10 INC	1 INC
DBB **324**	Traversing keys		**Orientation axis 2**					
	+	–	Rapid traverse override	Traversing key disable	Feed stop	Activate handwheel (bit value coding)		
DBB **325**	**Orientation axis 2**							
			var. INC	10000 INC	1000 INC	100 INC	10 INC	1 INC
DBB **326**	**OEM signals orientation axis 2**							
DBB **327**	**Orientation axis 2**							
DBB **328**	Traversing keys		**Orientation axis 3**					
	+	–	Rapid traverse override	Traversing key disable	Feed stop	Activate handwheel (bit value coding)		
DBB **329**	**Orientation axis 3**							
DBB **330**	**OEM signals orientation axis 3**							
DBB **331**	**Orientation axis 3**							

Signals from orientation axes

DB 21 - 30	Signals from NCK channel (NCK→PLC)							
Byte	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
DBB **332**	**Orientation axis 1**							
	Travel command		Travel request			Handwheel active (bit value coding)		
	plus	minus	plus	minus				
DBB **333**	**Orientation axis 1** Active machine function							
			var. INC	10000 INC	1000 INC	100 INC	10 INC	1 INC
DBB **334**	**OEM signals orientation axis 1**							
DBB **335**	**Orientation axis 1**							
DBB **336**	**Orientation axis 2**							
	Travel command		Travel request			Handwheel active (bit value coding)		
	plus	minus	plus	minus				
DBB **337**	**Orientation axis 2** Active machine function							
			var. INC	10000 INC	1000 INC	100 INC	10 INC	1 INC
DBB **338**	**OEM signals orientation axis 2**							
DBB **339**	**Orientation axis 2**							
DBB **340**	**Orientation axis 3**							
	Travel command		Travel request			Handwheel active (bit value coding)		
	plus	minus	plus	minus				
DBB **341**	**Orientation axis 3** Active machine function							
			var. INC	10000 INC	1000 INC	100 INC	10 INC	1 INC
DBB **342**	**OEM signals orientation axis 3**							
DBB **343**	**Orientation axis 3**							

Tool management functions from NC channel

DB 21 - 30	Signals from NCK channel (NCK→PLC)							
Byte	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Modification signals tool management functions								
DBB **344**					Last replacement tool of tool group	Transfer to new replacement tool	Tool limit value reached	Tool prewarning limit reached
DBB **345-347**								
Transferred tool management functions								
DBD **348**	T number for tool prewarning limit (DInt)							
DBD **352**	T number for tool limit value (DInt)							
DBD **356**	T number of new replacement tool (DInt)							
DBD **360**	T number of last replacement tool (DInt)							

Signals from/to NC channel

DB 21 - 30	Signals from NCK channel (NCK→PLC, PLC →NCK)							
Byte	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
	CH_CYCLES_SIG_IN (Bit 0 – 7)							
DBB **364**								
	CH_CYCLES_SIG_IN (Bit 8 – 15)							
DBB **365**								
	CH_CYCLES_SIG_OUT (Bit 0 – 7)							
DBB **366**								
	CH_CYCLES_SIG_OUT (Bit 8 – 15)							
DBB **367**								
	CH_OEM_TECHNO_SIG_IN (DBB368 - 371)							
DBB **368**								
DBB **369**								
DBB **370**								
DBB **371**								
DBB **372**	CH_OEM_TECHNO_SIG_OUT (DBB372 - 375)							
DBB **373**								
DBB **374**								
DBB **375**								
DBB **376**	ProgEventDisplay							
DBB **377**								
DBB **378**								
DBB **379**								
DBB **380**								
DBB **381**								
DBB **382**								
DBB **383**								

3.2.14 Signals from/to axis/spindle (PLC→NCK) (DB 31–DB 61)

DB 31 -61	Signals to axis/spindle (PLC→NCK)							
Byte	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
DBB **0** Axis and spindle	Feedrate override /V1/ H	G	F	E	D	C	B	A
DBB **1** Axis and spindle	Override active /V1/	Position measuring system 2 /A2/	Position measuring system 1 /A2/	Follow-up mode /A2/	Axis/spindle disable /A2/	Sensor fixed stop /F1/ (SW 2 and higher)	Acknowl. fixed stop reached /F1/ (SW 2 and higher)	Drive test movement enable
DBB **2** Axis and spindle	Reference point value /R1/ 4	3	2	1	Clamping in progress /A3/	Delete distance-to-go/ spindle reset /A2, S1/	Controller enable /A2/	Cam activation /N3/ (SW 2 and higher)
DBB **3** Axis and spindle	Program test axis/ spindle release	Velocity/ spindle speed limitation /A3/	Activate fixed feed 4 /FBMA/, /V1/ (SW 4 and higher)	Activate fixed feed 3 /FBMA/, /V1/ (SW 4 and higher)	Activate fixed feed 2 /FBMA/, /V1/ (SW 4 and higher)	Activate fixed feed 1 /FBMA/, /V1/ (SW 4 and higher)	Enable travel to fixed stop /F1/ SW 2 and higher)	Accept external ZO /K2/ (SW 2 and higher)
DBB **4** Axis and spindle	Traversing keys /H1/ plus	minus	Rapid traverse override /H1/	Traversing key disable /H1/	Feed stop/spindle stop /A2/	Activate handwheel /H1/ 3	2	1
DBB **5** Axis and spindle	Machine function /H1/		Var. INC	10000 INC	1000 INC	100 INC	10 INC	1 INC
DBB **6** Axis and spindle	OEM axis signals							
DBB **7**								
DBB **8**	Request PLC axis/spindle /K5/			Activation signal with change of this byte /K5/	Allocate NC axis to channel /K5/ D	C	B	A

Note

DBX8.4: is automatically reset after assignment (SW 3.7, 4.2 and higher). For previous SW versions, the activation signal must be applied until the assignment is made (DBB68).

DB 31-61	Continuation: Signals to axis/spindle (PLC → NCK)							
Byte	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
DBB **9**					Lock parameter set definition from NC /A2/	Control parameter block (SW4 and higher) /A2/ C	B	A
DBB **10**								REPOS DELAY
DBB **11**								Start brake test
DBB **12** Axis	Delay reference point approach /R1/			Modulo limit enabled	2nd software limit switch /A3/ plus	minus	Hardware limit switch /A3/ plus	minus
DBB **13** Axis								
DBB **14** Axis								
DBB **15** Axis								
DBB **16** Spindle	Delete S value /S1/	No n-monitoring when changing gear /S1/	Resyn-chronize spindle 1 /S1/	Resyn-chronize spindle 2 /S1/	Gear has changed over /S1/	Actual gear stage /S1/ C	B	A
DBB **17** Spindle		Invert M3/M4 /S1/	Resyn-chronize spindle at pos. 2 /S1/	Resyn-chronize spindle at pos. 1 /S1/				Feedrate override f. spindle valid /S1/
DBB **18** Spindle	Setpoint rot. direct. /S1/ CCW	CW	Oscillating speed /S1/	Oscillation via PLC /S1/				
DBB **19** Spindle	Spindle override /V1/ H	G	F	E	D	C	B	A

DB 31-61	Continuation: Signals to axis/spindle (PLC → NCK)							
Byte	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
DBB **20** 611D 1)					Speed setpoint smoothing /A2/	Torque limit 2 /A2/	Ramp-function generator interface /A2/	Runup change-over U/f mode /DE1/
DBB **21** 611D 1)	Pulse enable /A2/	n controller integrator disable /A2/	Selecting motor /A2/	Motor selection /A2/ B	A	Drive parameter set selection /A2/ C	B	A
DBB **22** Safety Integr. 1)				Selection of Safe speed bit value 1	bit value 0		Deselect safe standstill	Deselect safe velocity and standstill
DBB **23** Safety Integr.	Activate test stop			Activate end position pair 2		Trans-mission of bit value 2	Trans-mission of bit value 1	Trans-mission of bit value 0
DBB **24**	Master/ slave on	Bit value for CTRLOUT_changed: 1 Change setpoint output assignment (for compile cycles)	0	Torque compensa-tion controller ON		CC_Slave axis Suppress link	Control Axis	Stepper motor Rotation monitoring
DBB **25**								
DBB **26** Grinding	Enable ESR response			Enable slave axis overlay	Comp-ensation control ON			
DBB **27** Grinding	HIAxMove	Stop Corr	DEPBCS	DEPMCS	HIAxMove	Resume Corr	DEPBCS	DEPMCS
DBB **28** Oscillation	PLC checks axis /P5/ (SW 2 and higher)	AxStop, stop	Stop at next reversal point /P5/ (SW 2 and higher)	Change reversal point /P5/ (SW 2 and higher)	Set reversal point /P5/ (SW 2 and higher)	AXRESUM E	AXRESET	
DBB **29** Grinding			Disable automatic synchroni-zation	Start gantry synchroni-zation Gantry				
DBB **30** (Technolog y)				Position spindle	Autom. gear step change	Start spindle Counter-clockwise	Start spindle Clockwise	Stop spindle

1) See note at the end of this subsection

DB 31-61	Continuation: Signals to axis/spindle (PLC → NCK)							
Byte	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
DBB **31** (Technology)		Track synchronism	Disable synchroni-zation	Resynchro-nize				
DBB **32** Safety Integr. 1)			Deselect external stop E	Deselect external stop D	Deselect external stop C	Deselect external stop A		
DBB **33** Safety Integr. 1)	Select override Bit value 3	Bit value 2	Bit value 1	Bit value 0				
DBB **34**								
DBB ...								
DBB **59**								

Note

The IS ”Delete distance-to-go” (DBX2.2) is effective only for position axes on an axis-specific basis; the IS ”Delete distance-to-go” (DB21-30, DB6.2) acts on a channel-specific basis. The IS ”Spindle reset” (DXB2.2) acts on a spindle-specific basis.

DB 31 -61	Signals from axis/spindle (NCK→PLC)							
Byte	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
DBB **60** Axis and spindle	Position reached /B1/ with exact stop fine	with exact stop coarse	References/ synchro-nizes 2 /R1/	References/ synchro-nizes 1 /R1/	Encoder limit frequency exceeded 2 /A3/	Encoder limit frequency exceeded 1 /A3/	NCU_Link Axis active /B3/	Spindle /no axis /S1/
DBB **61** Axis and spindle	Current controller active /A2/	Speed controller active /A2/	Position controller active /A2/	Axis/spindle stationary ($n < n_{min}$) /A2/	Follow-up mode active /A2/	Axis ready /B3/	Axial alarm	Travel request /F1/
DBB **62**	Axis container rotation active	Force fixed stop limited /F1/ (SW 5.2)	Fixed stop reached /F1/ (SW 2 and higher)	Activate travel to fixed stop /F1/ (SW 2 and higher)	Measure-ment active /M5/	Revolutio-nal feedrate active	Handwheel overlay active /H1/ (SW 2 and higher)	Software cams active /N3/ (SW 2 and higher)
DBB **63**	Stop HIAxMove active	Corr active	DEPBCS active	DEPMCS active	Axis/ spindle disable active	Axis stop active /P2/	PLC-controlled axis /P2/	AXRESET DONE /P2/
DBB **64** Axis and spindle	Traverse command /H1/ plus	minus	Travel request plus	minus		Handwheel active /H1/ 3	2	1
65 Axis and spindle	Active machine function /H1/		Var. INC	10000 INC	1000 INC	100 INC	10 INC	1 INC

1) See note at the end of this subsection

DB 31-61	Continuation: Signals to axis/spindle (NCK → PLC)							
Byte	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
DBB **66** Axis and spindle	OEM axis signals (reserved)							Activate monitoring /TE6/
DBB **67**								
DBB **68**	PLC axis/ spindle /K5/	Neutral axis/ spindle /K5/	Axis replacement possible /K5/	New type requested by PLC /K5/	NC axis/spindle in channel /K5/ D	C	B	A
DBB **69**	NCU number in NCU link network					Control parameter block C	B	A
DBB **70**						Repos delay quit	Repos shift valid	Repos shift
DBB **71**								Brake test active
DBB **72**								REPOS DELAY
DBB **73**								
DBB **74**				Modulo limit enabled active				
DBB **75**								
DBB **76** Axis	Rounding axis in position	Indexing axis in position /T1/	Positioning axis /P2/	Path axis				Scratch pulse /A2/
DBB **77**								
DBB **78** Axis	F function (REAL format) for positioning axis /V1/							
DBB **82** Spindle					Gear change-over /S1/	Setpoint gear stage /S1/ C	B	A
DBB **83** Spindle	Actual rotat. direction CW /S1/	Speed monitoring /W1/ (SW 2 and higher)	Spindle in setpoint range /S1/	Support area limits violated /S8/ (SW 2 and higher)	Geometry monitoring /W1/ (SW 2 and higher)	Set speed increased /S1/	Set speed limited /S1/	Speed limit exceeded /S1/

DB 31-61	**Continuation: Signals to axis/spindle (NCK → PLC)**							
Byte	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
DBB **84** Spindle	Active spindle operating mode /S1/				Tapping without compen-sating chuck /S1/	CLGON active /S8/ (SW 2 and higher)	SUG active (grinding wheel surface speed)	Const. cutting speed active
	Control mode	Oscillation mode	Positioning mode	Synchro-nous mode				
DBB **85** Spindle			Spindle in position					
DBB **86** Spindle	M function (binary) for spindle /S1/							
DBD **88** Spindle	S function (floating-point) for spindle /S1/							
DBB **92** 611D 1)				RLI active	Speed setpoint smoothing active /A2/	Torque limit 2 active /A2/	HLGSS active /A2/	Set-up mode active /A2/
DBB **93** 611D 1)	Enable pulses /A2/	n controller integrator disabled /A2/	Drive ready /A2/	Active motor /A2/		Active drive parameter set /A2/		
				B	A	C	B	A
DBB **94** 611D 1)	Variable signaling fct. /A2/	$n_{act} = n_{set}$ /A2/	$\lvert n_{act} \rvert < n_x$ /A2/	$\lvert n_{act} \rvert < n_{min}$ /A2/	Md < Mdx /A2/	Ramp-up complete /A2/	Temperature prewarning /A2/	
							Heat sink	Motor
DBB **95** 611D 1)	Limitation of power section I2T							$U_{DC\text{-}link}$ < alarm threshold /A2/
DBB **96**	Master/ slave active /TE3/	Bit value for CTRLOUT_changed		Master/ Slave Compen-sation controller activ	Master/ Slave coarse	Master/ Slave fine	Axis control active	(Stepper motor) error rotation monitoring /S6/
		1	0					
		Change setpoint output assignment (for compile cycles)						
DBB **97**					Offset after turn-on point /TE6/	Activate mirroring /TE6/	Activate link /TE6/	Axis is slave axis /TE6/

1) See note at the end of this subsection

DB 31-61	Continuation: Signals to axis/spindle (NCK → PLC)							
Byte	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
DBB **98** Synchro-nous spindle	Emergency retraction active	Accel. warning threshold reached	Speed warning threshold reached	Overlaid motion /S3/ (SW2 and higher)		Actual value coupling /S3/ (SW2 and higher)	Synchronism (SW 2 and higher) /S3/ coarse	fine
DBB **99** Synchro-nous spindle	Emergency retraction enabled	Max. acce-leration reached	Max. speed reached	Synchro-nization running	Axis acce-lerating		Slave spindle active (SW 2 and higher) /S3/	Master spindle active (SW 2 and higher) /S3/
DBB **100** Grinding (SW 2 and higher)	Oscillation active /P5/	Oscillation motion active /P5/	Spark-out active /P5/	Error in oscillation /P5/	Oscillation cannot start /P5/			
DBB **101** Gantry (SW 2 and higher)	Gantry axis /G1/	Gantry leading axis /G1/	Gantry grouping is synchro-nous /G1/	Gantry synchroni-zation run ready to start /G1/	Gantry warning limit exceeded /G1/	Gantry cut-off limit exceeded /G1/		
DBB **102,103**								
DBB **104** Grinding (SW2 and higher)	Active infeed axis /P5/ Axis 8	Axis 7	Axis 6	Axis 5	Axis 4	Axis 3	Axis 2	Axis 1
DBB **105**	Active infeed axis /P5/ Axis 16							Axis 9
DBB **106**	Active infeed axis /P5/ Axis 24							Axis 17
DBB **107**	Active infeed axis /P5/	Axis 31	Axis 30					Axis 25
DBB **108** 1)	SINUMERIK Safety Integrated / / Axis safely referenced					Status pulses deleted		Safe operational stop / safe speed active

1) See note at the end of this subsection

DB 31-61	Continuation: Signals from axis/spindle (NCK → PLC)							
Byte	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
DBB **109** 1)	SINUMERIK Safety Integrated Actual position > cam position SC 4-	SC 4+	SC 3-	SC 3+	SC 2-	SC 2+	SC 1-	SC 1+
DBB **110** 1)	SINUMERIK Safety Integrated		n < nx	Safe velocity active bit	Safe velocity active bit		Safe zero speed	

				value 1	value 0		active	
DBB **111** 1)	Reserved for SINUMERIK Safety Integrated / /							
	Stop E active	Stop D active	Stop C active	Stop A/B active				

Note

This note is refers to the signal bytes marked with 1) in column 1 in the above table.

These signal bytes are directly transferred to the interface independently of any configured link communication.

3.2.15 Interface for loading/unloading magazine (DB 71)

DB 71	Interface for loading/unloading magazine (NCK→PLC)							
Byte	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
DBB **0**	Interface (I) active							
	I8	I7	I6	I5	I4	I3	I2	I1
DBB **1**	I16	I15	I14	I13	I12	I11	I10	I9
DBB **2,3**								
DBB **n**	res.	res.	res.	NC program positions magazine	Position at loading point	Reload	Unload	Load
DBB **n + 2**	Assigned channel (8 bit Int)							
DBB **n + 3**	Tool management no. (8 bit Int)							
DBD **n + 4**	Unassigned parameter 1 (D word)							
DBD **n + 8**	Unassigned parameter 2 (D word)							
DBD **n + 12**	Unassigned parameter 3 (D word)							
DBW **n + 16**	Identification for loading/unloading station (Int), (fixed value 9999)							
DBW **n + 18**	No. of loading station (Int)							
DBW **n + 20**	Magazine no. (source) for unloading/reloading/positioning (Int)							
DBW **n + 22**	Location no. (source) for unloading/reloading/positioning (Int)							
DBW **n + 24**	Magazine no. (target) for loading/reloading/positioning (Int)							
DBW **n + 26**	Location no. (target) for loading/reloading/positioning (Int)							
DBW **n + 28**	Reserved							

Initial addresses of the loading/unloading stations:

Loading/unloading station 1: n= 4 Loading/unloading station 3: n= 64
Loading/unloading station 2: n= 34 Loading/unloading station 4: n= 94

Load interface 1 is responsible for spindle loading and reloading of tools, for relocating tools and for positioning at any location (e.g. buffer).

References: /FBW/, "Description of Functions Tool Management"

3.2.16 Interface for spindle as change position (DB 72)

DB 72	Signals from spindle (NCK→PLC)							
Byte	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
DBB **0**	Interface (I) active							
	I8	I7	I6	I5	I4	I3	I2	I1
DBB **1**	I16	I15	I14	I13	I12	I11	I10	I9
DBB **2,3**								
DBB **n**	res.	Replace manual tool	Replace manual tool.	OldT in buffer no. (n-42)	T0	Prepare change	Perform change (initiate: M06)	Compul-sory change
DBB **n + 1**	Unassigned							
DBB **n + 2**	Assigned channel (8 bit Int)							
DBB **n + 3**	Tool management no. (8 bit Int)							
DBD **n + 4**	Unassigned parameter 1 (D word)							
DBD **n + 8**	Unassigned parameter 2 (D word)							
DBD **n + 12**	Unassigned parameter 3 (D word)							
DBW **n + 16**	Buffer identification (Int), (fixed value 9998) (corresponds to "Target position for new tool")							
DBW **n + 18**	Relative location (target) in the buffer (Int)							
DBW **n + 20**	Magazine no. (source) for new tool (Int)							
DBW **n + 22**	Location no. (source) for new tool (Int)							
DBW **n + 24**	Magazine no. (target) for old tool (Int)							
DBW **n + 26**	Location no. (target) for old tool (Int)							
DBW **n + 28**	Tool new: location type (Int)							
DBW **n + 30**	Tool new: size left (Int)							
DBW **n + 32**	Tool new: size right (Int)							
DBW	Tool new: size top (Int)							

n + 34								
DBW **n + 36**	Tool new: size bottom (Int)							
DBW **n + 38**	Tool status for tool new							
	Tool was in use	Tool fixed location coded	Tool being changed	Prewarning limit reached	Tool measured	Tool disabled	Tool enabled	Active tool
				Bit 12 Master tool	Bit 11 to be loaded	Bit 10 to be unloaded	Bit 9 ignore disabled	Bit 8 ID for tools in buffer
DBW **n + 40**	Tool new: T no. (Int)							
DBW **n + 42**	If DBX (n+0.4) = 1, then buffer location of old tool is entered here.							
DBW **n + 44**	Original magazine of new tool (SW 6.4 and higher)							
DBW **n + 46**	Original location of new tool (SW 6.4 and higher)							
Initial addresses of the buffers: Spindle 1:n= 4 Spindle 2:n = 52								

References: /FBW/, ”Description of Functions, Tool Management”

3.2.17 Interface for circular magazine (DB 73)

DB73	Signals from circular magazine (NCK→PLC)							
Byte	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
DBB **0**	Interface (I) active							
	I8	I7	I6	I5	I4	I3	I2	I1
DBB **1**	I16	I15	I14	I13	I12	I11	I10	I9
DBB **2,3**								
DBB **n**	res.	res.	res.	res.	T0	res.	Perform change (initiation: T no.)	Obligatory change
DBB **n + 1**	Unassigned							
DBB **n + 2**	Assigned channel (8 bit Int)							
DBB **n + 3**	Tool management no. (8 bit Int)							
DBD **n + 4**	Unassigned parameter 1 (D word)							
DBD **n + 8**	Unassigned parameter 2 (D word)							
DBD **n + 12**	Unassigned parameter 3 (D word)							
DBW **n + 16**	Reserved							
DBW **n + 18**	Reserved							
DBW **n + 20**	Circular magazine no. (Int)							
DBW **n + 22**	Location no. for new tool (Int)							
DBW **n + 24**	Reserved							
DBW **n + 26**	Location no. for old tool (Int)							
DBW **n + 28**	Tool new: location type (Int)							

DBW **n + 30**	Tool new: size left (Int)							
DBW **n + 32**	Tool new: size right (Int)							
DBW **n + 34**	Tool new: size top (Int)							
DBW **n + 36**	Tool new: size bottom (Int)							
DBW **n + 38**	Tool status for tool new							
	Tool was in use	Tool fixed location coded	Tool being changed	Prewarning limit reached	Tool measured	Tool disabled	Tool enabled	Active tool
				Bit 12 Master tool	Bit 11 to be loaded	Bit 10 to be unloaded	Bit 9 ignore disabled	Bit 8 ID for tools in buffer
DBW **n + 40**	Tool new: T no. (Int)							
DBW **n + 42**	Original location of new tool in this circular magazine (SW 6.3 and higher)							
Initial addresses of the circular magazines: circular magazine 1: n = 4 2: n = 48								

References: /FBW/, "Description of Functions, Tool Management"

3.2.18 Signals to and from the machine control panel and HHU (840Di with MCI2 only) (DB 77)

DB77	**Signals to and from the machine control panel and HHU**							
Byte	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
DBB **0 to** DBB **7**	Input signals from MCP1 to PLC, MPI Bus (GD communication)							
DBB **8 to** DBB **15**	Output signals from MCP1 to PLC, MPI Bus (GD communication)							
DBD **16**	Status send MCP1, MPI bus (GD communication)							
DBD **20**	Status receive MCP1, MPI bus (GD communication)							
DBB **24 to** DBB **31**	Input signals from MCP2 to PLC, MPI bus (GD communication)							
DBB **32 to** DBB **39**	Output signals from MCP2 to PLC, MPI bus (GD communication)							
DBD **40**	Status send MCP2, MPI bus (GD communication)							

DB77	Signals to and from the machine control panel and HHU							
Byte	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
DBD **44**	Status receive MCP2, MPI bus (GD communication)							
DBB **48** to DBB **53**	Input signals from HHU to PLC, MPI bus (GD communication)							
DBB **60** to DBB **79**	Output signals from PLC to HHU, MPI bus (GD communication)							
DBD **80**	Status Send HHU, MPI bus (GD communication)							
DBD **84**	Status Receive HHU, MPI bus (GD communication)							

```
FB1- parameter:
MCPNum          :=1, //correct number of MCPs
MCP1In  :=P#DB77.DBX0.0,
MCP1Out         := P#DB77.DBX8.0,
MCP1StatSend    := P#DB77.DBX16.0,
MCP1StatRec     := P#DB77.DBX20.0,
MCP2In  :=P#DB77.DBX24.0,
MCP2Out         := P#DB77.DBX32.0,
MCP2StatSend    := P#DB77.DBX40.0,
MCP2StatRec     := P#DB77.DBX44.0,
MCPSDB210 := TRUE,

BHG:    :=1;    //handheld unit interface:
                //0 - no HHU
                //1 – HHU to MPI
                //2 – HHU to OPI
BHGIn   :=P#DB77.DBX48.0,       //transmitted data of handheld unit
BHGOut: :=P#DB77.DBX60.0,       //received data of handheld unit
BHGStatSend: :=P#DB77.DBX80.0, // status DW for transmitting handheld unit
BHGStatRec: :=P#DB77.DBX84.0,  // status DW for receiving HHU
```

3.2.19 Signals to/from ManualTurn, ShopMill, ShopTurn (DB 82)

A list of the signals of DB 82 can be found in:
References: /FBMA/, Description of Functions ManualTurn
/FBSP/, Description of Functions ShopMill
/FBT/, Description of Functions ShopTurn

of Functions ShopTurn

■

Für Notizen

4 PLC-Blocks

4

4.1 Overview of organization blocks

Table	Overview of organization blocks (OBs)		
OB no.	**Designation**	**Meaning**	**Kit**
1	ZYKLUS	Cyclic processing	GP
40	ALARM	Process alarms	GP
100	NEUSTART	Beginning of restart	GP

4.2 Overview of function blocks

Table Overview of **function blocks (FCs)**

FC no.	**Designation**	**Meaning**	**Kit**
0	-	Reserved for Siemens	
2	GP_HP	Basic program, cyclic part	GP
3	GP_PRAL	Basic program, alarm-controlled part	GP
5	GP_DIAG	Basic program, diagnostic alarm (FM-NC)	GP
7	TM_REV	Transfer block for tool change with circular magazine	GP
8	TM_TRANS	Transfer block for tool management	GP
9	ASUP	Asynchronous subprograms	GP
10	AL_MSG	Alarms/messages	GP
12	AUXFU	Call interface for user auxiliary functions	GP
13	BHG_DISP	Display control for handheld unit	GP
15	POS_AX	Positioning axis	GP
16	PART_AX	Indexing axis	GP
17		Y-D switchover	GP
18	SpinCtrl	Spindle control from PLC	GP
19	MCP_IFM	Distribution of machine control panel and MMC signals to interface (milling machine)	GP
21		Transfer data exchange PLC-NCK	GP
22	TM_DIR	Selection of direction	GP
24	MCP_IFM2	Transfer of MCP signals to interface	GP
25	MCP_IFT	Distribution of machine control panel and MMC signals to interface	GP
26	HPU_MCP	Distribution of HPU signals to interface	
30 - 35		Assigned if ManualTurn, ShopMill or ShopTurn are installed	
36 - 127		**User assignable with FM-NC, 810DE**	
36 - 255		**User assignable with 810D, 840DE, 840D**	

Table	Overview of **function blocks (FBs)**		
FB no.	**Designation**	**Meaning**	**Kit**
0 - 29		Reserved for Siemens	
1	RUN_UP	Basic program, booting	GP
2	GET	Read NC variables	GP
3	PUT	Write NC variables	GP
4	PI_SERV	PI services	GP
5	GETGUD	Read GUD variable	GP
7	PI_SERV2	General PI services	GP
29		Diagnostics for signal recorder and data trigger	GP
36 - 127		**User assignable with FM-NC, 810DE**	
36 - 255		**User assignable with 810D, 840DE, 840D**	

4.3 Assignment of data blocks

Note

Only so many DBs are created as are necessary according to NC-MD.

Table **Fehler! Formatvorlage nicht definiert.**-1 Overview of data blocks

DB no.	Designation	Meaning	Kit
1		Reserved for Siemens	GP
2 - 4	PLC MSG	PLC messages	GP
5 - 8		Basic program	
9	NC COMPILE	Interface for NC compile cycles	GP
10	NC INTERFACE	Central NC interface	GP
11	BAG 1	Mode group interface	GP
12		Computer link and transport system	
13-14		Reserved (Hymnos, basic program)	
15		Basic program	
16		PI service definitions	
17		Version code	
18		SPL interface (Safety Integrated)	
19		MMC interface	
20		PLC machine data	
21 - 30	CHANNEL 1	NC channel interface	GP
31 - 61	AXIS 1,...	Reserved for interface axis/spindle no. 1 to 31	GP
62 - 70		**User assignable**	
71 - 74		User tool management	GP
75 - 76		M group decoding	GP
77		Tool management buffer	
78 - 80		Reserved for Siemens	
81 - 89		Assigned if ManualTurn, ShopMill or ShopTurn are installed	
(81)90 - 127		**User assignable FM-NC, 810DE, see below**	
(81)90 - 399		**User assignable 810D, 840DE, 840D, see below**	

Note

Data blocks of inactivated channels, axes/spindles, C programming, tool management can be assigned by the user.

4.4 Assigned timers

Timer no.	Meaning
1 - 9	Reserved
10 - 127	User assignable

■

A

A Appendix

References

You will find a list that is updated monthly of the documentation available in each language in the Internet at:

http://www.siemens.com/motioncontrol

Follow the menu items → “Support” → “Technical documentation” → “Overview of publications” or “DOConWEB”.

■

Notes

I

I Index

I.1 Index

$

/

1

2

3

4

6

A

B

C

E

F

G

H

I

J

K

L

M

N

O

P

Q

R

S

T

U

V

W

X

Y

Z

■

Notes

Siemens AG

A&D MC BMS
P. O. Box 3180

D-91050 Erlangen

Federal Republic of Germany

Tel. +49 180 / 5050 – 222 [Hotline]

Fax +49 9131 / 98 – 63315[Documentation]

E-Mail: mailto:motioncontrol.docu@siemens.com

Suggestions

Corrections

For Publication/Manual:

SINUMERIK 840D sl/840D/840Di/810D
SIMODRIVE 611 digital/Sinamics

Lists

Manufacturer/Service documentation

From

Name:

Company/Dept.

Address: ______________________________

______________ ______________________________

Telephone: ___________ / ______________________________

Telefax: ________ / ______________________________

Lists (2nd Book)

Order No.: 6FC5 397-3CP10-0BA0
07.05 Edition

Should you come across any printing errors when reading this publication, please notify us on this sheet. Suggestions for improvements are also welcome.

Suggestions and/or corrections

Overview of documentation SINUMERIK 840D sl (07/2005)

General Documentation

- *SINUMERIK* — Brochure
- *SINUMERIK* 840D sl — Catalog NC 61 ***)**
- *SINAMICS* S120 — Catalog D21.2 Servo Control ***)**

User Documentation

- *SINUMERIK* 840D sl — Operator's Guide
  – **HMI Embedded *)**
  – **ShopMill**
  – **ShopTurn**
- *SINUMERIK* 840D sl, 840D, 840Di, 810D — Operator's Guide
  – **HMI Advanced *)**
  – Short Guide
- *SINUMERIK* 840D sl, 840D, 840Di, 810D — **Program. Guide**
  – Fundamentals ***)**
  – Advance ***)**
  – Short Guide
  – Lists System Variable
  – ISO Turning/Milling
- *SINUMERIK* 840D sl, 840D, 840Di, 810D — Program. Guide
  – **Cycles**
  – **Measuring cycles**
- *SINUMERIK* 840D sl, 840D, 840Di, 810D — **Diagnostics–Guide *)**

Manufacturer/Service Documentation

- *SINUMERIK* 840D sl — Manual **NCU *)**
- *SINUMERIK* 840D sl — Manual **Operator–Components *)**
- *SINUMERIK* 840D sl — **Installation & Start–up Guide** CNC ***)**
  – Part 1 NCK, PLC, drive
  – Part 2 HMI
  – Part 3 ShopMill
  – Part 4 ShopTurn
  – Part 5 Basesoftware
- *SINUMERIK* 840D sl, 840D, 840Di, 810D — **Lists *)**
  – Part 1
  – Part 2

Manufacturer/Service Documentation

- *SINUMERIK* 840D sl, 840D, 840Di, 810D — **Description of Functions**
  – Basic Machine ***)**
  – Extended Functions
  – Spacial Functions
- *SINUMERIK* 840D sl, 840D, 840Di, 810D — Description of Functions
  _ **Synchronized Actions**
  – **ISO–Dialects**
- *SINAMICS* S120 — Description of Fuctions **Drive Function**
- *SINUMERIK* 840D sl — Description of Functions **Safety Integrated**
- *SINUMERIK* 840D sl, 840D, 840Di, 810D — **EMV Guidelines**

Electronic Documentation

- *SINUMERIK* *SINAMICS* *Motoren* — DOCONCD ***)** DOCONWEB

***)** These Documents are a minimum requirement

Overview of documentation SINUMERIK 840D/840D/810D (07/2005)

General Documentation

- *SINUMERIK* — Brochure
- *SINUMERIK* 840D, 840Di, 810D — Catalog NC 60 ***)**
- *Safety Integrated* — Safety Integrated Application Manual

Anwender–Dokumentation

- *SINUMERIK* 840D, 840Di, 810D — Operator's Guide
  - **HMI Embedded *)**
  - **ShopMill**
  - **ShopTurn**
  - HT6
- *SINUMERIK* 840D sl, 840D, 840Di, 810D — Operator's Guide
  - **HMI Advanced *)**
  - Short Guide
- *SINUMERIK* 840D sl, 840D, 840Di, 810D — **Program. Guide**
  - Fundamentals ***)**
  - **Advance*)**
  - Short Guide
  - Lists System Variable
  - ISO Turning/Milling
- *SINUMERIK* 840D sl, 840D, 840Di, 810D — Program. Guide
  - **Cycles**
  - **Measuring Cycles**
- *SINUMERIK* 840D sl, 840D, 840Di, 810D — **Diagnostics–guide *)**
- *SINUMERIK* 840Di — **Systemoverview**

Manufacturer/Service Documentation

- *SINUMERIK* — **Configuring** (HW) ***)**
  - 840D
  - 810D
- *SINUMERIK* 840D, 840Di, 810D — Manual Operator–**Components *)**
- *SINUMERIK* — **Installation & Start–up Guide *)**
  - 840D
  - 810D
  - HMI
- *SINUMERIK* 840D, 840Di, 810D — Description of Functions
  - **ShopMill**
  - **ShopTurn**
- *SINUMERIK* 840Di — **Installation & Start–up Guide**
- *SINUMERIK* 840D sl, 840D, 840Di, 810D — **Lists *)**
  - Part 1
  - Part 2

Manufacturer/Service Documentation

- *SINUMERIK* 840D sl, 840D, 840Di, 810D — **Description of Functions**
  - Basic Machine ***)**
  - Extended Functions
  - Special Functions
  - **Synchronized Actions**
  - **ISO–Dialects**
  - **EMV–Guidelines**
- *SINUMERIK* 840D, 840Di, 810D — Description of Functions
  - **Linear motor**
  - **Tool Mgmt.**
  - **Hydraulics Modules**
  - **Analog Modules**
- *SINUMERIK* 840D, 840Di, 810D — Description of Functions
  - **Remote Diagnosis**
  - **@Event**
- *SINUMERIK* 840D — Description of Functions **Safety Integrated**
- *SINUMERIK* 840D, 810D — Configuring Kit
  - **Operator Interface OP030**
  - **HMI Embedded**
- *SINUMERIK* 840D, 810D — **MCIS**
  - Computer Link
  - Tool Data Information System
  - NC Data Mgmt.
  - NC Data Transfer
  - Tool Data Communication

Electronic Documentation

- *SINUMERIK SIMODRIVE Motoren* — DOCONCD ***)**, DOCONWEB

***)** This documents are a minimum requirement