SIEMENS

SINUMERIK 840D sl, 840Di sl, SINAMICS S120

Lists, 2nd Book

Parameter Manual

Valid for

Control

SINUMERIK 840D sl/ 840DE sl
SINUMERIK 840Di sl/ 840DiE sl

Software	*Version*
NCU system software for 840D sl/ 840DE sl	1.5/ 2.5
system software for 840Di sl/ 840DiE sl	1.4

01/2008

SINUMERIK®-Documentation

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.

Edition	Order-No.	Remarks
07/2005	6FC5397-3CP10-0BA0	**A**
03/2006	6FC5397-3CP10-1BA0	**C**
11/2006	6FC5397-3CP10-2BA0	**C**
01/2008	6FC5397-3CP10-3BA0	**C**

Copyright © Siemens AG 1995 - 2008.
Orderl-No. 6FC5397-3CP10-3BA0

Siemens AG 2008.
Subject to change without prior notice

Preface

Structure of the documentation

The SINUMERIK documentation is organized in 3 parts:

- General documentation
- User documentation
- Manufacturer/service documentation

An overview of publications, which is updated on a monthly basis and provides information
about the language versions available, can be found on the Internet at:
http://www.siemens.com/motioncontrol

Follow menu items "Support" → "Technical Documentation" → "Ordering Documentation" →"Printed Documentation".

The Internet version of the DOConCD (DOConWEB) is available at:
http://www.automation.siemens.com/doconweb

Information about training courses and FAQs (Frequently Asked Questions) can be found at the following web site:
http://www.siemens.com/motioncontrol und dort unter Menüpunkt "Support"

Target group

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

Standard version

This Parameter Manual only describes the functionality of the standard version. 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.

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

Technical Support

If you have any technical questions, please contact our hotline:

Europe / Africa:
Phone: +49 (0) 180 / 5050 - 222
Fax: +49 (0) 180 / 5050 - 223
Internet: http://www.siemens.com/automation/support-request

America:
Phone: +1 423 262 2522
Fax: +1 423 262 2289
Email: mailto:techsupport.sea@siemens.com

Asien / Pazifik:
Phone: +86 1064 719 990
Fax: +86 1064 747 474
Email: mailto:adsupport.asia@siemens.com

Note

National phone numbers for technical support are provided under the following Internet adress:

http://www.siemens.com/automation/service&support
Calls are subject to charge, e.g. € 0.14/min on the German landline network. Tariffs of other hone providers may differ.

Questions about this documentation

If you have any queries (suggestions, corrections) in relation to this documentation, please send a fax or email to the following address:

Fax: 49 (0) 9131 / 98 - 2176
Email: mailto:docu.motioncontrol@siemens.com

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

SINUMERIK Internet address

http://www.siemens.com/sinumerik

Benefits

The Parameter Manual enables the intended target group to evaluate error and fault indications and to respond accordingly.
With the help of the Parameter Manual, the target group has an overview of the various diagnostic options and diagnostic tools.

With the present edition, the previous Lists will be subdivided into Lists (1st Book) and Lists (2nd Book).
1 st Book contains:
- Overview of functions
- Maschine data (HMI, NCK, SD)
- SINAMICS Parameters

The table of contents refers to the present 2nd Book.

Safety Instructions

This Manual contains information which you should carefully observe to ensure your own personal safety and the prevention of material damage. 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 The warnings appear in decreasing order of risk as given below.

Danger

Indicates an imminently hazardous situation which, if not avoided, will result in death or serious injury or in substantial property damage.

Warning

Indicates that death or severe personal injury will result if proper precautions are not taken.

Caution

with a warning triangle indicates that minor personal injury can result if proper precautions are not taken.

Caution

without a warning triangle indicates that property damage can result if proper precautions are not taken.

Notice

indicates a potential situation which, if not avoided, may result in an undesirable event or state.

If several hazards of different degrees occur, the hazard with the highest degree must always be given priority. A warning notice accompanied by a safety alert symbol indicating a risk of bodily injury can also indicate a risk of property damage.

Qualified Personnel

The associated device/system may only be set up and operated using this documentation. Commissioning and operation of a device/system may only be performed by qualified personnel. Qualified persons are defined as persons who are authorized to commission, to ground, and to tag circuits, equipment, and systems in accordance with established safety practices and standards.

■

Table of Contens

1

1 Variables

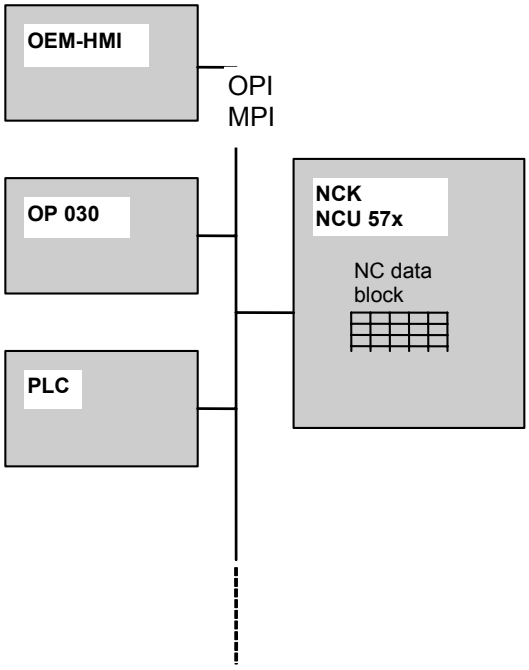
1.1 Introduction

This section describes the NCK variables that an HMI 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

Description of PLC access method in:

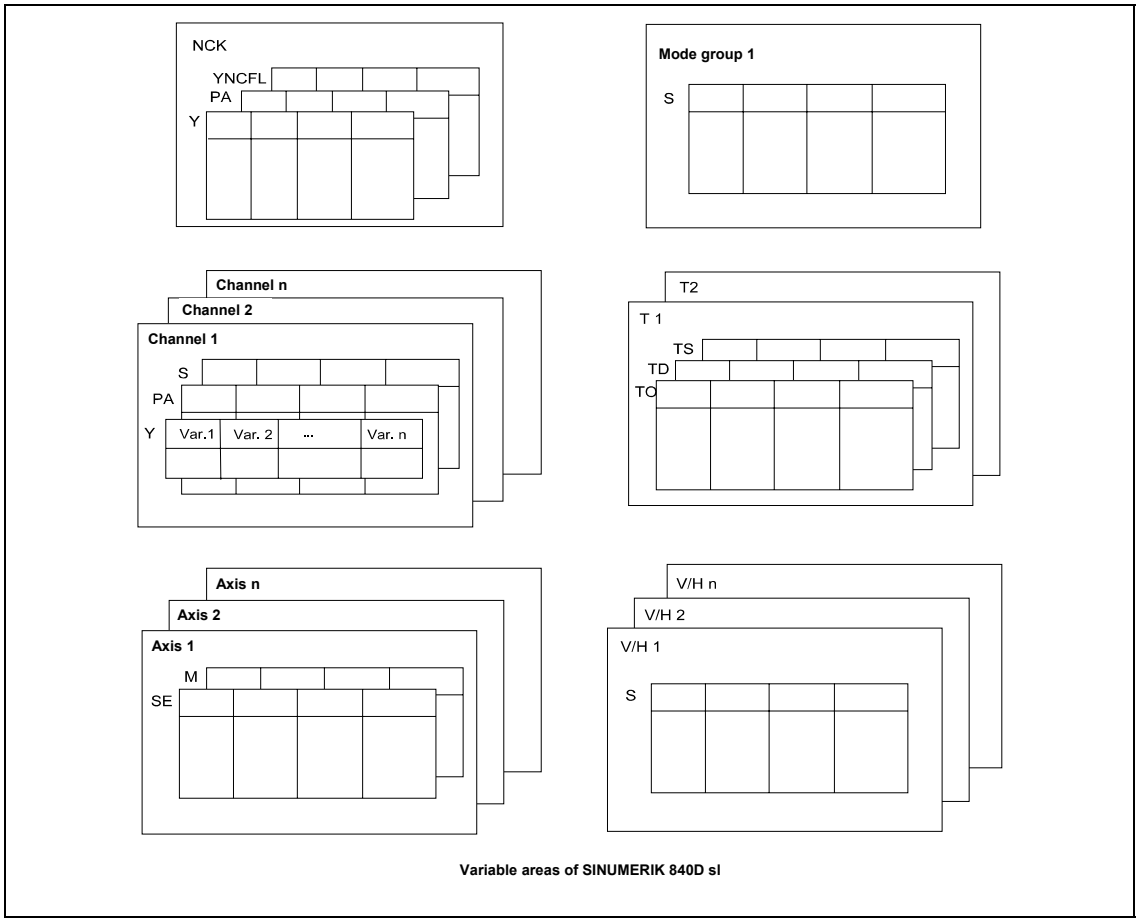
References: /FB/P3 "Basic PLC Program"



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 sl

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 HMI or PLC using other mechanisms. The documentation listed below describes the modules to which these mechanisms are applied:

References: /FBO/Configuring the OP 030 Operator Interface,
/FB1/P3/"Basic PLC Program"

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				
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:

HMI:
/Channel/Configuration/numMachAxes[u1]

HMI/OP 030:
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:

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

HMI/OP 030:
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	
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).

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

HMI/OP 030:
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

1.1.5 Reference

Reference is made to the following documents:

/FB1/	Function Manual of basic machines, supporting manuals: A2, A3, B1, B2, D1, F1, G2, H2, K1, K2, N2, P1, P3, R1, S1, V1, W1, Z1
/FB2/	Function Manual of expanded functions, supporting manuals: A4, B3, B4, H1, K3, K5, M1, M5, N3, N4, P2, P5, R2, S3, S7, T1, W3, W4, Z2
/FB3/	Function Manual of special functions, supporting manuals: F2, G1, G3, K6, M3, S9, T3, TE01, TE02, TE1, TE3, TE4, TE6, TE7, TE8, V2, W5, W6, Z3
/FBA/	Function manual of drive functions, supporting manuals: DB1, DD1, DD2, DE1, DF1, DG1, DL1, DM1, DS1, DÜ1
/FBU/	Description of Functions SIMODRIVE 611 universal
/FBSI/	Description of Functions Safety Integrated
/IAC/	810D Installation & Start-Up Guide
/IAD/	840D/611D Installation & Start-Up Guide
/POS3/	POSMO SI/CD/CA User Manual
/FBHLA/	Description of Functions HLA module
/IAM/	Commissioning CNC Part 2 (HMI), supporting manuals: BE1, HE1, IM2, IM4
/FBO/	Configuring OP 030 Operator Interface
/FBT/	Description of Functions ShopTurn
/FBSP/	Description of Functions ShopMill
/BAS/	Operating/Programming ShopMill
/BAD/	Operator's Guide HMI-Advanced
/BEM/	Operator's Guide HMI-Embedded
/FBW/	Description of Functions Tool Management
/FBMA/	Description of Functions ManualTurn
/FBFA/	Description of Functions ISO Dialects for SINUMERIK
/FBSY/	Description of Functions Synchronized Actions
/PGA/	Programming Manual Job Planning

1.2 System data

1.2.1 Area N, Module 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					
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: Cutting edge parameter no. 26 valid (ISO Dialect Milling H No.) Bit 1: Cutting edge parameter no. 27 valid (Orientation of the cutting edge) Bit 2: Cutting edge parameter no. 28 valid (L1 of the orientation of the cutting edge) Bit 3: Cutting edge parameter no. 29 valid (L2 of the orientation of the cutting edge) Bit 4: Cutting edge parameter no. 30 valid (L3 of the orientation of the cutting edge) Bit 5: Cutting edge parameter no. 31 valid (L1 of the orientation of the cutting edge normal) Bit 6: Cutting edge parameter no. 32 valid (L2 of the orientation of the cutting edge normal) Bit 7: Cutting edge parameter no. 33 valid (L3 of the orientation of the cutting edge normal) etc.					
-				UWord	r
Multi-line: yes	1		1		

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

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

maskToolManagement	$MN_MM_TOOL_MANAGEMENT_MASK				
Settings for NCK tool management function Activation of tool management memory with "0" means: The set tool management data do not occupy any memory space. Bit 0=1: Memory for TM-specific data is made available Bit 1=1: Memory for monitoring data is made available Bit 2=1: Memory for user data (CC data) is made available Bit 3=1: Memory for "Consider adjacent location" is made available 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				

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

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

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

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

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

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

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

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

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

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

maxnumEdgeSC	$MN_MAX_SUMCORR_PERCUTTING_EDGE				
Max. number of total offsets per edge 0 to 6					
-	0 ??? 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					

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	UWord	r
Multi-line: yes	1		1		

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

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

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

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

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

numDrives					
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	UWord	r
Multi-line: yes	1		1		

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

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

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

numMagPlaceParams					
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					

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

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

numParams_SC	$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	UWord	r
Multi-line: yes	1		1		

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

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

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

typeOfCuttingEdge					
Type of D-number programming see MD: MM_TYPE_OF_CUTTING_EDGE 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, Module 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 variables					
-				UWord	r
Multi-line: no					

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

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

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

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

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

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

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

stringIsFileId					
-				UWord	r
Multi-line: no					

stringIsMeaning					
-				UWord	r
Multi-line: no					

stringIsSymbolId					
-				UWord	r
Multi-line: no					

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

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

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

1.2.3 Area N, Module 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		

kinChain1stElem	$NK_1ST_ELEM				
First element of the kinematic chain					
-				String[32]	r
Multi-line: yes	Chain number		$MN_MM_MAXNUM_KIN_CHAINS		

kinChainName	$NK_CHAIN_NAME				
Name of the kinematic chain					
-				String[32]	r
Multi-line: yes	Chain number		$MN_MM_MAXNUM_KIN_CHAINS		

kinElemAngle	$NK_ANG				
Turning angle					
Degree				Double	r
Multi-line: yes	Number of the chain element		$MN_MM_MAXNUM_KIN_CHAIN_ELEM		

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

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

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

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

kinElemNextP	$NK_NEXT				
Reference to a branching kinematic chain					
-				String[32]	r
Multi-line: yes	Number of the chain element		$MN_MM_MAXNUM_KIN_CHAIN_ELEM		

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

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

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

kinElemRotAx0	$NK_ROT[0]				
Direction component in the X direction					
-				Double	r
Multi-line: yes	Number of the chain element		$MN_MM_MAXNUM_KIN_CHAIN_ELEM		

kinElemRotAx1	$NK_ROT[1]				
Direction component in the Y direction					
-				Double	r
Multi-line: yes	Number of the chain element		$MN_MM_MAXNUM_KIN_CHAIN_ELEM		

kinElemRotAx2	$NK_ROT[2]				
Direction component in the Z direction					
-				Double	r
Multi-line: yes	Number of the chain element		$MN_MM_MAXNUM_KIN_CHAIN_ELEM		

modelChangeCounter					
Modification counter of the machine model					
-				UWord	r
Multi-line: yes	1: Kinematic modification counter		4		

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

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

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

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

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

pa3DChainName	$NP_CHAIN_NAME				
Name of the kinematic chain carrying the protection zone					
-				String[32]	r
Multi-line: yes	Number of the protection zone		MN_MM_MAXNUM_3D_PROT_AREAS		

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

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

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

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

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

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

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

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

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

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

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

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

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

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

pa3DGroupAdd	$NP_ADD_GROUP				
Name of a protection zone group to be added					
-				String[32]	r
Multi-line: yes	Number of the protection zone group		$MN_MM_MAXNUM_3D_PROT_GROUPS		

pa3DGroupMember1	$NP_MEMBER[0]				
Name of 1st protection zone in this group					
-				String[32]	r
Multi-line: yes	Number of the protection zone group		$MN_MM_MAXNUM_3D_PROT_GROUPS		

pa3DGroupMember2	$NP_MEMBER[1]				
Name of 2nd protection zone in this group					
-				String[32]	r
Multi-line: yes	Number of the protection zone group		$MN_MM_MAXNUM_3D_PROT_GROUPS		

pa3DGroupMember3	$NP_MEMBER[2]				
Name of 3rd protection zone in this group					
-				String[32]	r
Multi-line: yes	Number of the protection zone group		$MN_MM_MAXNUM_3D_PROT_GROUPS		

pa3DGroupMember4	$NP_MEMBER[3]				
Name of 4th protection zone in this group					
-				String[32]	r
Multi-line: yes	Number of the protection zone group		$MN_MM_MAXNUM_3D_PROT_GROUPS		

pa3DGroupName	$NP_GROUP_NAME				
Name of the protection zone group					
-				String[32]	r
Multi-line: yes	Number of the protection zone group		$MN_MM_MAXNUM_3D_PROT_GROUPS		

pa3DInitStat	NP_INIT_STAT				
Initialization status of the protection zone ASCII code of the letters: "I", "A", "P"					
-				Character	r
Multi-line: yes	Number of the protection zone		MN_MM_MAXNUM_3D_PROT_AREAS		

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

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

pa3DSuppressed					
Collision monitoring suppressed by PLC					
-	0	0	1	UWord	r
Multi-line: no			1		

1.2.4 Area C, Module 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		

collisionAlarm	$ON_COLLCHECK_LEVEL				
Collision of two protection zones					
-				UWord	r
Multi-line: yes	1: 0=No collision in the channel, otherwise alarm number		3		

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

declarProtObjectReal					
Real parameters of the declaration of a variable protection zone					
-				Double	r
Multi-line: yes	Number of the real parameter. The number and significance		10		

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

1.2.5 Area N, Module 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.

Gruppe					
Instruction group column: G group number * 2					
-				String[16]	r
Multi-line: yes	Serial number		GroupID		

Gruppe_NUM					
Number of G functions in the relevant group column: (2 * G group number) - 1					
-				UWord	r
Multi-line: yes	1		1		

1.3 State data of system

1.3.1 Area N, Module 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)

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.				

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				

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

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

aDbw	$A_DBW[x] x = Offset				
Data word (16 bits) from/to the PLC 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					
-				Long Integer	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 data area				

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

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

aDpInLength	$A_DP_IN_LENGTH				
PROFIBUS length of input data area					
-	0	0	128	Long Integer	r
Multi-line: yes	RangeIndex		16		

aDpInState	$A_DP_IN_STATE				
PROFIBUS status of input data area					
-	0	0	3	Long Integer	r
Multi-line: yes	RangeIndex		16		

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

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

aDpOutLength	$A_DP_OUT_LENGTH				
PROFIBUS length of output data area					
-	0	0	128	Long Integer	r
Multi-line: yes	RangeIndex		16		

aDpOutState	$A_DP_OUT_STATE				
PROFIBUS status of output data area					
-	0	0	3	Long Integer	r
Multi-line: yes	RangeIndex		16		

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

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

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

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

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

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

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

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

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

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

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

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

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

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

aInsip	$A_INSIP				
Internal PLC-SPL input signal 0: Input not set 1: Input set					
-	0	0	1	UWord	r
Multi-line: yes	Bit number		64		

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

aPbbOut	$A_PBB_OUT[index]				
Data byte (8 bits) in PLC input/output area OUT (also available on 810D CCU2) 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 PLC input/output area				

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

aPbdOut	$A_PBD_OUT[index]				
Data double word (32 bits) in the PLC input/output area OUT (also available on 810D CCU2) Can be written from SW 6.4.					
-				Long Integer	wr
Multi-line: yes	Position offset within PLC input/output area				

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

aPbrOut	$A_PBR_OUT[index]				
Real data (32 bits) in the PLC input/output area OUT (also available on 810D CCU2) Can be written from SW 6.4.					
-				Double	wr
Multi-line: yes	Position offset within PLC input/output area				

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

aPbwOut	$A_PBW_OUT[index]				
Data word (16 bits) in the PLC input/output area OUT (also available on 810D CCU2) 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 PLC input/output area				

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

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

aXfaultsi	$A_XFAULTSI				
Information about Stop F for a safety axis: Bit 0 = 1: An actual value error has been discovered in the cross-check between NCK and 611D of an arbitrary safety axis. Bit 1 = 1: Some error has been discovered in the cross-check between NCK and 611D of an arbitrary safety axis and the waiting time until Stop B is triggered is running or has expired ($MA_SAFE_STOP_SWITCH_TIME_F)					
-	0	0		Long Integer	r
Multi-line: yes	1		1		

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

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

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

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

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

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

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

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

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

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

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

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

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

anRebootDelayTime	$AN_REBOOT_DELAY_TIME				
Time until reboot					
-	0	0		Double	r
Multi-line: yes	1		1		

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

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

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

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

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

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

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

anVModelStatus	$AN_VMODEL_STATUS				
System variable for the status of the VRML Model 1: MODIFIED_STATE: The model was internally modified This is the initial state. It could change, if machine parameters change, like protection areas. 2: COPIED_STATE: This state is generated outside the NCK, when the model file is ready to be displayed. 3: DISPLAYED_STATE: When the NCK send the order to display the model in the Viewer.					
-	1	1	3	Long Integer	wr
Multi-line: yes	1		1		

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

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

axisActivInNcu					
Display indicating whether the axis is active, i.e. whether it can be traversed via a channel of its own NCU or via another NCU (link axis). This data can be utilized by 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.					
-				Long Integer	r
Multi-line: yes	1		1		

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

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

checkSumForAcxData					
Current checksum of selected ACX data, in order to check relatively quickly whether ACX data have changed. The data required for the OPI access are at the start of the downloaded ACX file.					
-				String[32]	r
Multi-line: yes	1: _N_NC_TEA_ACX		6		

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)		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		

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

fsInfoChangeCounter					
First line enabled for modification in part program fsInfoPathName.					
-				UDoubleword	r
Multi-line: yes	No. of info object		20		

fsInfoChangeDateTime					
Change time of the file system object fsInfoPathName.					
-				String[13]	r
Multi-line: yes	No. of info object		20		

fsInfoChangeDateTimeSub					
Change time of the files contained in the fsInfoPathName directory.					
-				String[13]	r
Multi-line: yes	No. of info object		20		

fsInfoFileLength					
Length of the file system object fsInfoPathName.					
-				UDoubleword	r
Multi-line: yes	No. of info object		20		

fsInfoObjStatus					
Status of file system object fsInfoPathName. Bit-coded (may be added to later): Bit0 = 0: Object not loaded in the NCK. Bit0 = 1: Object loaded in the NCK. Bit1 = 0: Object is a file. Bit1 = 1: Object is a directory					
-		0	3	UWord	r
Multi-line: yes	No. of info object		20		

fsInfoPartition					
Partition and lifetime of the file fsInfoPathName SRP: SRAM persistent USV: DRAM User volatile USP: DRAM User persistent SIP: DRAM Siemens persistent MAV: DRAM Manufacturer volatile MAP: DRAM Manufacturer persistent TMV: DRAM Temporary volatile D1V: DRAM System 1 volatile EXV: DRAM External volatile EXP: DRAM External persistent					
-				String[160]	r
Multi-line: yes	No. of info object		20		

fsInfoPathName					
Name of a file or directory to be observed					
-				String[160]	wr
Multi-line: yes	No. of info object		20		

fsInfoRights					
Access protection and lifetime of file fsInfoPathName 0-7 ASCII-coded for read / write / execute / show / delete rights					
-	"77777"			String[6]	r
Multi-line: yes	No. of info object		20		

fsInfoSeekw					
First line enabled for modification in part program fsInfoPathName					
-				UDoubleword	r
Multi-line: yes	No. of info object		20		

fsInfoUsed					
Assignment of file system info object. The client first searches for a free info object by reading the list fsInfoUsed. This is assigned by writing fsInfoUsed with 1. If there is no error in the assignment, the desired file or directory can be selected by writing the fsInfoPathName. The information about this object can then be read from the other variables. 0: Info object free. It must be explicitly released by writing 0. 1: Info object assigned. If an assigned object is assigned again then this is acknowledged negatively.					
-		0	1	UWord	wr
Multi-line: yes	No. of info object		20		

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		

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

licenseStatus					
Licensing status 0: Licensed, 1: Insufficiently licensed 2: Not licensed					
-	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)		

nckMode					
Mode in which the NCK works. The mode can be set with the PI _N_NCKMOD. Bit0: NCK works accelerated in simulation mode. This mode is currently provided only for the VNCK. Bit1: NCK slowed in order to give the simulation more computation time on the same processor. NC start not possible.					
-	0	0	3	UWord	r
Multi-line: yes	1		1		

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

ncuLinkActive					
Display indicating whether NCU link is activated (via machine data setting) Based on display, 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			Long Integer	r
Multi-line: yes	1		1		

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

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

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

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

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

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

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		

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		

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		Long Integer	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		Long Integer	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		

protocStrtVarArea					
Logging: Variable which is to be monitored for start triggering.					
-	0	0		UWord	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		Long Integer	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 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		

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		Long Integer	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		

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		

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		

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		

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		

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

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

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

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

safeFrdpDiagNck	$A_FRDP_DIAG[n]				
Diagnostics data for F_RECVDP communication error 01H = Timeout detected 02H = Sequence number error detected 04H = CRC error detected					
-	0	0	0x7	UDoubleword	r
Multi-line: yes	3		3		

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

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

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

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

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

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

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

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

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

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

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

safeFsdpDiagNck	$A_FSDP_DIAG[n]				
Diagnostics data for F_SENDDP communication error 01H = Timeout detected 02H = Sequence number error detected 04H = CRC error detected					
-	0	0	0x7	UDoubleword	r
Multi-line: yes	3		3		

safeFsdpDriverStateNck					
Current state of the F_SENDDP driver 0 = Not parameterized 1 = Initialization 2 = F_SENDDP ready: waiting for F_RECVDP 3 = F_SENDDP and F_RECVDP ready: waiting for user acknowledgement from F_RECVDP 4 = Normal operation					
-	0	0	4	UWord	r
Multi-line: yes	3		3		

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

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

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

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

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

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

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

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		

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		

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		

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		

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

safeMarkerNckWord					
NCK flag words for the safe programmable logic					
-	0	0		Long Integer	r
Multi-line: yes	1: Image of the system variable $A_MARKERSID[1]		2		

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

safeMarkerPlcWord					
Image of the PLC flag words for the safe programmable logic					
-	0	0		Long Integer	r
Multi-line: yes	1: Image of the system variable $A_MARKERSIPD[1]		2		

safeNumActiveFrdp					
Number of active F_RECVDP connections					
-	0	0	3	UWord	r
Multi-line: yes	1		1		

safeNumActiveFsdp					
Number of active F_SENDDP connections					
-	0	0	3	UWord	r
Multi-line: yes	1		1		

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

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

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

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

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

stopCondNumNck					
Number of active stop states in the NCK Specifies the number of occupied lines in stopCond					
-				UWord	r
Multi-line: yes	1		1		

stopCondPar					
Stop state parameters in the NCK. More than one stop state can be active simultaneously. The highest priority stop state appears below the first line, this is followed by those with lower priorities.					
-				UWord	r
Multi-line: yes	High byte: No. of the active stop state				

stopCondParA					
Stop state parameters in the NCK. More than one stop state can be active simultaneously. The highest priority stop state appears below the first line, this is followed by those with lower priorities.					
-				String[32]	r
Multi-line: yes	High byte: No. of the active stop state				

stopCondTime					
BCD time stamp for stop state in the NCK. More than one stop state can be active simultaneously. The highest priority stop state appears below the first line, this is followed by those with lower priorities.					
-				Date+Time	r
Multi-line: yes	Number of the active stop state		stopCondNumNck		

swLicensePIN					
PIN for licensing					
-				String[128]	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		Long Integer	r
Multi-line: yes	1		1		

sysTimeNCSCatTraceStart					
Logging: NCSC time stamp for the trace start time in µs					
-	0	0		Long Integer	r
Multi-line: yes	User no. (1-10)		10		

sysTimeNCSCatTraceTrig					
Logging: NCSC time stamp for the trace start trigger time in µs					
-	0	0		Long Integer	r
Multi-line: yes	User no. (1-10)		10		

sysTimeNCSCdiffTraceStart					
Logging: Time difference for the trace start time in µs					
-	0	0		Long Integer	r
Multi-line: yes	User no. (1-10)		10		

sysTimeNCSCdiffTraceTrig					
Logging: time difference for the trace start trigger time in µs					
-	0	0		Long Integer	r
Multi-line: yes	User no. (1-10)		10		

sysTimeSinceStartup					
System run time in seconds since NCK ramp-up					
s, user defined	0	0		Double	r
Multi-line: yes	1		1		

tlkNr					
Unique copy number for the temporary license key					
-				String[32]	r
Multi-line: yes	1		1		

tlkPIN					
Temporary license key					
-				String[128]	r
Multi-line: yes	1		1		

tlkStatus					
Status of the temporary license key 0: Inactive 1: Set 2: Internally reset 3: Faulty input					
-	0			UWord	r
Multi-line: yes	1		1		

totalDirectorys					
Maximum number of directories which may be created see: $MN_MM_NUM_DIR_IN_FILESYSTEM					
-				UWord	r
Multi-line: yes	1		1		

totalFiles					
Maximum number of files which may be created (see: $MM_NUM_FILES_IN_FILESYSTEM)					
-				UWord	r
Multi-line: yes	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		

totalMemDramEPassF					
Size in bytes of the passive file system for executing from external drives					
-	0	0		Long Integer	r
Multi-line: yes	1		1		

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

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

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

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

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

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

totalMemISram					
Total internal SRAM in bytes					
-	0	0		Long Integer	r
Multi-line: yes	1		1		

totalMemSramPassF					
Size of passive file system (SRAM) in bytes					
-	0	0		Long Integer	r
Multi-line: yes	1		1		

totalProtokolFiles	$MM_PROTOC_NUM_FILES				
Logging: Maximum number of log files which may be created					
-	0	0	1	UWord	r
Multi-line: yes	User No. (1-10)		10		

traceProtocolActive	$A_PROTOC				
Logging: User status 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		

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

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

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

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

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

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

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

usedMemISram					
Occupied internal SRAM					
-	0	0		Long Integer	r
Multi-line: yes	1		1		

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

usedOptionsNotLicensed					
List of options which are not licensed					
-				String[200]	r
Multi-line: yes	1		1		

usedProtokolFiles					
Logging: Number of protocol files that have already been created					
-	0	0	1	UWord	r
Multi-line: yes	User No. (1-10)		10		

vaDpActTel	$VA_DP_ACT_TEL[n, Achse]				
Word for word image of the Profibus actual value message frames from drives on the Profibus/PROFIdrive					
-	0	0		Long Integer	r
Multi-line: yes	100 * axis index + word offset in the message frame		100 * numMachAxes + 19		

1.3.2 Area N, Module 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		16		

clearInfo					DA
Acknowledgement criterion 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		16		

fillText2					DA
Parameter 2 of the alarm					
-				String[32]	r
Multi-line: yes	Alarm list index		16		

fillText3					DA
Parameter 3 of the alarm					
-				String[32]	r
Multi-line: yes	Alarm list index		16		

fillText4					DA
Parameter 4 of the alarm					
-				String[32]	r
Multi-line: yes	Alarm list index		16		

textIndex					
Alarm number (actual alarm)					
-				Long Integer	r
Multi-line: yes	Alarm list index		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, Module 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		16		

clearInfo					DA
Acknowledgement criterion 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					

fillText1					DA
Parameter 1 of the alarm					
-				String[32]	r
Multi-line: yes	Alarm list index		16		

fillText2					DA
Parameter 2 of the alarm					
-				String[32]	r
Multi-line: yes	Alarm list index		16		

fillText3					DA
Parameter 3 of the alarm					
-				String[32]	r
Multi-line: yes	Alarm list index		16		

fillText4					DA
Parameter 4 of the alarm					
-				String[32]	r
Multi-line: yes	Alarm list index		16		

textIndex					
Alarm number (actual alarm)					
-				Long Integer	r
Multi-line: yes	Alarm list index		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, Module 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.

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		16		

clearInfo					DA
Acknowledgement criterion 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					

fillText1					DA
Parameter 1 of the alarm					
-				String[32]	r
Multi-line: yes	Alarm list index		16		

fillText2					DA
Parameter 2 of the alarm					
-				String[32]	r
Multi-line: yes	Alarm list index		16		

fillText3					DA
Parameter 3 of the alarm					
-				String[32]	r
Multi-line: yes	Alarm list index		16		

fillText4					DA
Parameter 4 of the alarm					
-				String[32]	r
Multi-line: yes	Alarm list index		16		

textIndex					
Alarm number (actual alarm)					
-				Long Integer	r
Multi-line: yes	Alarm list index		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, Module 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 (from this module)					
mm, inch, degree, user defined				Double	r
Multi-line: yes	Axis index		numMachAxes		

cmdToolBasePos					
Tool base position, desired value . Physical unit is defined in variable extUnit (in this module).					
mm, inch, degree, user defined				Double	r
Multi-line: yes	Axis index		numMachAxes		

extUnit					
Current physical unit of the axis position 0 = mm 1 = inch 2 = degree 3 = indexing position 4 = userdef					
-				UWord	r
Multi-line: yes	Axis index		numMachAxes		

name					
Axis name					
-				String[32]	r
Multi-line: yes	Axis index		numMachAxes		

status					
Axis status 0 = travel command in plus direction 1 = travel command in minus direction 2 = exact position coarse reached 3 = exact position fine reached					
-				UWord	r
Multi-line: yes	Axis index		numMachAxes		

toolBaseDistToGo					
Tool base distance-to-go. Physical unit is defined in the variable extUnit (in this module).					
mm, inch, degree, user defined				Double	r
Multi-line: yes	Axis index		numMachAxes		

toolBaseREPOS					
Tool base REPOS. Physical unit is defined in the variable extUnit (in this module).					
mm, inch, degree, user defined				Double	r
Multi-line: yes	Axis index		numMachAxes		

varIncrVal					
Settable value for INC_VAR. The physical value depends on whether the axis is linear or rotary. Linear axis: unit is 1 mm Rotary axis: unit is 1/1000 degrees					
mm, inch, degree, user defined				Double	r
Multi-line: yes	Axis index		numMachAxes		

1.3.6 Area N, Module 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.

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

PRESETVal					
The function PRESETON (...) programs a work offset for an axis. The value of the offset is stored in the variable 'PRESETVal'. The variable can be overwritten by the part program and by the MMC.					
mm, inch, user defined				Double	wr
Multi-line: yes	Axis index		maxnumGlobMachAxes		

aaAcc					
Current axial acceleration value					
m/s2, 1000 inch/ s2, rev/s2, user defined	0	0		Double	r
Multi-line: yes	Axis index		maxnumGlobMachAxes		

aaAccPercent					
Current acceleration value for single-axis interpolation in percent					
-	0	0		UWord	r
Multi-line: yes	Axis index		maxnumGlobMachAxes		

aaActIndexAxPosNo					
Current indexing position; the display depends on $MN_INDEX_AX_NO_MODE and the division (via table or equidistant)					
-	0			Long Integer	r
Multi-line: yes	Axis index		maxnumGlobMachAxes		

aaAlarmStat					
Display indicating whether alarms are active for a PLC-controlled axis. The relevant coded alarm reactions can be used as a source for the "Extended Stop and Retract" function. The data is bit-coded, allowing, where necessary, individual states to be masked or evaluated separately (bits not listed supply a value of 0) Bit2 = 1: NOREADY (active rapid deceleration + cancelation of servo enable) Bit6 = 1: STOPBYALARM (rampm stop in all channel axes) Bit9 = 1: SETVDI (VDI interface signal "Setting alarm") Bit13 = 1: FOLLOWUPBYALARM (Follow-up)					
-	0			UWord	r
Multi-line: yes	Axis index		maxnumGlobMachAxes		

aaAxChangeStat					
Axis status with respect to axis replacement 0: Axis can be replaced 1: Axis is linked to the channel, but can become the PLC, command or reciprocating axis 2: Axis cannot be replaced					
-	0	0	2	UWord	r
Multi-line: yes	Axis index		maxnumGlobMachAxes		

aaAxChangeTyp					
Axis type with respect to axis replacement 0: Axis assigned to the NC program 1: Axis assigned to the PLC or active as command axis or reciprocating axis 2: Other channel has interpolation right 3: Neutral axis 4: Neutral axis controlled from the PLC 5: Other channel has interpolation right; axis is requested for the NC program 6: Other channel has interpolation right; axis is requested as neutral axis 7: Axis is PLC axis or is active as command axis or reciprocating axis; axis is requested for the NC program 8: Axis is PLC axis or is active as command axis or reciprocating axis; axis is requested as neutral axis					
-	0	0	8	UWord	r
Multi-line: yes	Axis index		maxnumGlobMachAxes		

aaBcsOffset					
Sum of all axial offsets of an axis, such as DRF, online tool offset, $AA_OFF and ext. WO.					
-	0			Double	r
Multi-line: yes	Axis index		maxnumGlobMachAxes		

aaBrakeCondB					
Shows the pending braking requests (conditions) for the interpolator stop of the axis / spindle. A braking request consists of a collision direction relating to a coordinate axis in the BCS and a braking priority relating to the machining step. If the axis / spindle receives a current braking request on account of these requirement(s), bit 0 is set in $AA_BRAKE_STATE[X] (in the next IPO cycle). The braking priorities in the positive direction are indicated in bits 0 to 2: 0: No pending braking request 1: Priority 1 covers all positioning actions (G0, POS, SPOS) 2: Priority 2 covers DYNNORM and all priority 1 motions 3: Priority 3 covers DYNPOS and all priority 1 to 2 motions 4: Priority 4 covers DYNROUGH and all priority 1 to 3 motions 5: Priority 5 covers DYNSEMIFIN and all priority 1 to 4 motions 6: Priority 6 covers all motions (including DYNFINISH) 7: Priority 7 covers all motions. The request was triggered by the VDI interface signal DB31,..DBX4.3 "Feed stop / Spindle stop". Braking always takes place, irrespective of the direction of motion. The braking priorities in the negative direction are indicated in bits 16 to 18: 0 to 7: Same meaning as bits 0 to 2 All other bits are not set. If the value of the variable is shown in hexadecimal format, the fifth character from the right shows the braking word in the negative direction and the first character from the right shows it in the positive direction.					
-	0	0	0x70007	UDoubleword	r
Multi-line: yes	Axis index		maxnumGlobMachAxes		

aaBrakeCondM					
Shows the pending braking requests (conditions) for the interpolator stop of the axis / spindle. A braking request consists of a collision direction relating to a coordinate axis in the MCS and a braking priority relating to the machining step. The braking priorities in the positive direction are indicated in bits 0 to 2: 0: No pending braking request 1: Priority 1 covers all positioning actions (G0, POS, SPOS) 2: Priority 2 covers DYNNORM and all priority 1 motions 3: Priority 3 covers DYNPOS and all priority 1 to 2 motions 4: Priority 4 covers DYNROUGH and all priority 1 to 3 motions 5: Priority 5 covers DYNSEMIFIN and all priority 1 to 4 motions 6: Priority 6 covers all motions (including DYNFINISH) 7: Priority 7 covers all motions. The request was triggered by the VDI interface signal DB31,..DBX4.3 "Feed stop / Spindle stop". Braking always takes place, irrespective of the direction of motion. The braking priorities in the negative direction are indicated in bits 16 to 18: 0 to 7: Same meaning as bits 0 to 2 All other bits are not set. If the value of the variable is shown in hexadecimal format, the fifth character from the right shows the braking word in the negative direction and the first character from the right shows it in the positive direction.					
-	0	0	0x70007	UDoubleword	r
Multi-line: yes	Axis index		maxnumGlobMachAxes		

aaBrakeState					
Returns for the axis / spindle whether braking has been initiated on account of the request of $AA_BRAKE_CONDB[X] or a VDI interface signal DB31,..DBX4.3 "Feed stop / Spindle stop".					
-	0	0	1	UDoubleword	r
Multi-line: yes	Axis index		maxnumGlobMachAxes		

aaChanNo					
The variable supplies the number of the channel in which the axis is currently being interpolated. With value 0, the axis could not be assigned to any channel.					
-	0	0		UWord	r
Multi-line: yes	Axis index		maxnumGlobMachAxes		

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

aaCoupCorr	$AA_COUP_CORR[Achse]				
This variable is used to execute the function "Correct synchronism error". It returns the compensation value for the position offset for the generic coupling with CPFRS = "MCS". The actual values of this spindle are compared with the setpoints for the duration (MD 30455 MISC_FUNCTION_MASK, bit 7) of the activation of the VDI interface signal DB31..,DBX31.6 'Correct synchronism' for the following spindle with coupling active. The difference is the compensation value, which can be read with this variable.					
-	0			Double	r
Multi-line: yes	Axis index		maxnumGlobMachAxes		

aaCoupCorrDist	$AA_COUP_CORR_DIST[Achse]				
Generic coupling: path still to be retracted for aaCoupCorr					
-	0			Double	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		

aaDepAxO					
Dependency on other axes. Returns an axis code for the defined axis AX containing all the machine axes that have a mechanical dependency on the defined axis.					
-	0	0		Long Integer	r
Multi-line: yes	Axis 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		

aaDtbreb					
The total estimated distance until braking end, BCS					
-	0	0		Double	r
Multi-line: yes	Axis index		maxnumGlobMachAxes		

aaDtbrebCmd					
Commanded portion of the braking distance, BCS					
-	0	0		Double	r
Multi-line: yes	Axis index		maxnumGlobMachAxes		

aaDtbrebCorr					
Correction portion of the braking distance, BCS					
-	0	0		Double	r
Multi-line: yes	Axis index		maxnumGlobMachAxes		

aaDtbrebDep					
Dependent portion of the braking distance, BCS					
-	0	0		Double	r
Multi-line: yes	Axis index		maxnumGlobMachAxes		

aaDtbrem					
The total estimated distance until braking end, MCS					
-	0	0		Double	r
Multi-line: yes	Axis index		maxnumGlobMachAxes		

aaDtbremCmd					
Commanded portion of the braking distance, MCS					
-	0	0		Double	r
Multi-line: yes	Axis index		maxnumGlobMachAxes		

aaDtbremCorr					
Correction portion of the braking distance, MCS					
-	0	0		Double	r
Multi-line: yes	Axis index		maxnumGlobMachAxes		

aaDtbremDep					
Dependent portion of the braking distance, MCS					
-	0	0		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		

aaEnc1Active					
First measuring system is active 0: Measuring system is not active 1: Measuring system is active					
-	0	0	1	UWord	r
Multi-line: yes	Axis index		maxnumGlobMachAxes		

aaEnc1Ampl					
Enc1: Percentage gain factor of the amplitude control					
-	0	0		Double	r
Multi-line: yes	Axis index		maxnumGlobMachAxes		

aaEnc2Active					
Second measuring system is active 0: Measuring system is not active 1: Measuring system is active					
-	0	0	1	UWord	r
Multi-line: yes	Axis index		maxnumGlobMachAxes		

aaEnc2Ampl					
Enc2: Percentage gain factor of the amplitude control					
-	0	0		Double	r
Multi-line: yes	Axis index		maxnumGlobMachAxes		

aaEncActive					
Measuring system is active 0: Measuring system is not active 1: Measuring system is active					
-	0	0	1	UWord	r
Multi-line: yes	Axis 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					
Activation of "NC-controlled ESR" for PLC-controlled axis					
-	0	0	1	UWord	r
Multi-line: yes	Axis index		maxnumGlobMachAxes		

aaFixPointSelected					
Selected fixed point: Number of the fixed point that is to be approached					
-	0			UDoubleword	r
Multi-line: yes	Axis index		maxnumGlobMachAxes		

aaIbnCorr					
Current BZS setpoint value of an axis including override components					
-	0			Double	r
Multi-line: yes	Axis index		maxnumGlobMachAxes		

aaIenCorr					
Current SZS setpoint value of an axis including override components					
-	0			Double	r
Multi-line: yes	Axis index		maxnumGlobMachAxes		

aaInSync					
Synchronization status of the following axis with master value coupling and ELG 0: Synchronization is not running 1: Synchronization is running, i.e. following axis is being synchronized					
-	0	0	1	UWord	r
Multi-line: yes	Axis index		maxnumGlobMachAxes		

aaInposStat					
Status for the programmed position 0: No status available (axis/spindle is outside of the programmed position) 1: Travel motion pending 2: Position setpoint reached 3: Position reached with 'exact stop coarse' 4: Position reached with 'exact stop fine'					
-	0	0	4	UWord	r
Multi-line: yes	Axis index		maxnumGlobMachAxes		

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

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

aaJerkTotal					
Overall total jerk of an axis					
-		0		Double	r
Multi-line: yes	Axis index		maxnumGlobMachAxes		

aaJogPosAct					
Position reached for JOG to position					
-	0	0	1	UWord	r
Multi-line: yes	Axis index		maxnumGlobMachAxes		

aaJogPosSelected					
JOG to position is active					
-	0	0	1	UWord	r
Multi-line: yes	Axis index		maxnumGlobMachAxes		

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

aaLeadPTurn					
Current master value - position component lost as a result of modulo reduction					
-	0	0		Double	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		

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					
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		

aaMeaAct					
Axial measuring active 0: Measuring system is not active 1: Measuring system is active					
-	0	0	1	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 value not reached 1: Limit value reached in positive axis direction 11: Limit value reached in negative axis direction					
-				UWord	r
Multi-line: yes	Axis 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		

aaOnFixPoint					
Current fixed point, number of the fixed point at which the axis stands					
-	0			UDoubleword	r
Multi-line: yes	Axis index		maxnumGlobMachAxes		

aaOscillBreakPos1					
Oscillation interrupt position 1					
-				Double	r
Multi-line: yes	Axis index		maxnumGlobMachAxes		

aaOscillBreakPos2					
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					
Axial override specified by PLC for motion-synchronous actions					
-	100	0		Double	r
Multi-line: yes	Axis index		maxnumGlobMachAxes		

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

aaPolfaValid					
States whether the retraction of the single axis is programmed 0: No retraction programmed for the single axis 1: Retraction programmed as position 2: Retraction programmed as distance					
-	0	0	2	UWord	r
Multi-line: yes	Axis index		maxnumGlobMachAxes		

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

aaProgIndexAxPosNo					
Programmed indexing position 0: No indexing axis, therefore no indexing position available >0: Number of the programmed indexing position					
-	0	0		UWord	r
Multi-line: yes	Axis index		maxnumGlobMachAxes		

aaRef					
Axis is referenced 0: Axis is not referenced 1: Axis is referenced					
-	0	0	1	UWord	r
Multi-line: yes	Axis index		maxnumGlobMachAxes		

aaReposDelay					
REPOS suppression active 0: REPOS suppression is currently not active for this axis 1: REPOS suppression is currently active for this axis					
-	0	0	1	UWord	r
Multi-line: yes	Axis index		maxnumGlobMachAxes		

aaScPar					
Current setpoint parameter set					
-	0	0		Long Integer	r
Multi-line: yes	Axis index		maxnumGlobMachAxes		

aaSnglAxStat					
Display status of a PLC-controlled axis 0: Not a single axis 1: Reset 2: Ended 3: Interrupted 4: Active 5: Alarm					
-	0			UWord	r
Multi-line: yes	Axis 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 status of the following axis with master value coupling 0: No synchronism 1: Synchronism coarse 2: Synchronism fine 3: Synchronism coarse and fine					
-				UWord	r
Multi-line: yes	Axis index		maxnumGlobMachAxes		

aaSyncDiff					
Setpoint synchronism difference					
mm, inch, degree, user defined	0	0		Double	r
Multi-line: yes	Axis index		maxnumGlobMachAxes		

aaSyncDiffStat					
Status of the setpoint synchronism difference -4: No valid value in aaSyncDiff, coupled motion from part program -3: Reserved -2: Reserved -1: No valid value in aaSyncDiff 0: No valid value in aaSyncDiff, coupling not active 1: Valid value in aaSyncDiff					
-	0	-4	1	Long Integer	r
Multi-line: yes	Axis index		maxnumGlobMachAxes		

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

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

aaTravelCount					
Total traverse processes of an axis					
-		0		Double	r
Multi-line: yes	Axis index		maxnumGlobMachAxes		

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

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

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

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

aaTravelTimeHS					
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		

aaType					
Cross-channel axis type 0: Axis type cannot be determined 1: NC program axis 2: Neutral axis 3: PLC axis 4: Reciprocating axis 5: Neutral axis that is currently executing a JOG or homing motion 6: Following axis coupled to the master value 7: Coupled motion of the following axis, activated in a synchronized action 8: Command axis 9: Compile Cycle axis 10: Coupled slave axis (master-slave function.) 11: Program axis that is currently executing a JOG or homing motion					
-	0	0	11	UWord	r
Multi-line: yes	Axis 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		

acRpValid					
Reapproach position valid 0: Reapproach position not valid 1: Reapproach position valid					
-	0	0	1	UWord	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, if the axis is a positioning axis.					
-				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		

activeSvOverride					
Currently active SG override factor in the NCK					
-	-1	-1	100	Long Integer	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 = inactive 1 = active					
-				UWord	r
Multi-line: yes	Axis index		maxnumGlobMachAxes		

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

chanAxisNoGap					
Display of existing axis, i.e. no axis gap in channel. 0: Axis does not exist 1: Axis does exist					
-	0	0	1	UWord	r
Multi-line: yes	Axis 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					
Desired value of axis-specific feedrate for a positioning 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					
Rotary drive: Load-side path corresponding to one revolution of the drive. Is returned in the unit of the internal computational resolution INT_INCR_PER_MM (for linear axes) or INT_INCR_PER_DEG (for rotary axes / spindles) taking into account gear ratios etc. In the case of linear axes, the pitch of the ball screw is also included in the calculation. In the case of linear motors, a fixed value of "1mm" is used for the ball screw pitch instead of the non-existent ball screw.					
mm, inch, degree, user defined				Double	r
Multi-line: yes	Axis index		maxnumGlobMachAxes		

drfVal					
DRF value					
-	0			Double	r
Multi-line: yes	Axis index		maxnumGlobMachAxes		

drive2ndTorqueLimit					
2nd torque limit. With linear motors: 2nd force limit 0 = inactive 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) 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		maxnumGlobMachAxes		

driveSetupMode					
Set-up mode 0 = inactive 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					
Additional information on travel to fixed stop if $VA_FXS[]=2, or OPI variable fxsStat=2. 0 No additional information available 1 No approach motion programmed 2 Programmed end position reached, movement ended 3 Abort by NC RESET (Reset key) 4 Fixed stop window exited 5 Torque reduction was rejected by drive 6 PLC has canceled enable signals					
-	0	0	6	UWord	r
Multi-line: yes	Axis 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 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 MD					
-				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		

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: 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			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 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 between 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		

safeActiveCamTrack					
Status Safe cam track (active/inactive) Bit 0 = 1/0: Safe cam track 1 active/inactive Bit 1 = 1/0: Safe cam track 2 active/inactive Bit 2 = 1/0: Safe cam track 3 active/inactive Bit 3 = 1/0: Safe cam track 4 active/inactive					
-	0	0	0xF	UWord	r
Multi-line: no			maxnumGlobMachAxes		

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 = inactive 1 = active					
-				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		

safeOutputSigCam					
Results of the NCK safe cam evaluation					
-	0	0	3FFFFFFF	Long Integer	r
Multi-line: no			maxnumGlobMachAxes		

safeOutputSigCamDrive					
Results of the drive safe cam evaluation					
-	0	0	3FFFFFFF	Long Integer	r
Multi-line: no			maxnumGlobMachAxes		

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

safeOutputSigDrive2					
Safe output signals of the drive part 2					
-		0	0xffff	UWord	r
Multi-line: no			maxnumGlobMachAxes		

safePosCtrlActive					
Axis monitors the absolute position 0 = Axis does not monitor the absolute position (no SE/SN) 1 = Axis monitors the absolute position					
-	0	0	1	UWord	r
Multi-line: no			maxnumGlobMachAxes		

safeStopOtherAxis					
Stop on another axis 0: No stop on another axis 1: Stop on another axis					
-	0	0	1	UWord	r
Multi-line: yes	Axis 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		

vaAbsoluteEnc1DeltaInit	$VA_ABSOLUTE_ENC_DELTA_INIT[Achse]				
Enc1: Initial difference					
-	0	0		UDoubleword	r
Multi-line: yes	Axis index		maxnumGlobMachAxes		

vaAbsoluteEnc1ErrCnt					
Enc 1: Error counter for absolute encoder					
-	0	0		Long Integer	r
Multi-line: yes	Axis index		maxnumGlobMachAxes		

vaAbsoluteEnc1State					
Enc1: Status of absolute encoder interface Bit0: Interface is active Bit1: Error during parity check Bit2: Error bit alarm Bit3: Error bit CRC error Bit4: Start bit missing with EnDat transfer					
-	0	0		Long Integer	r
Multi-line: yes	Axis index		maxnumGlobMachAxes		

vaAbsoluteEnc1ZeroMonMax	$VA_ABSOLUTE_ENC_ZERO_MON_MAX[Achse]				
Enc1:Maximum of vaEnc1ZeroMonAct with absolute encoder					
-	0	0		UDoubleword	r
Multi-line: yes	Axis index		maxnumGlobMachAxes		

vaAbsoluteEnc2DeltaInit	$VA_ABSOLUTE_ENC_DELTA_INIT[Achse]				
Enc2: Initial difference					
-	0	0		UDoubleword	r
Multi-line: yes	Axis index		maxnumGlobMachAxes		

vaAbsoluteEnc2ErrCnt					
Enc 2: Error counter for absolute encoder					
-	0	0		Long Integer	r
Multi-line: yes	Axis index		maxnumGlobMachAxes		

vaAbsoluteEnc2State					
Enc2: Status of absolute encoder interface Bit0: Interface is active Bit1: Error during parity check Bit2: Error bit alarm Bit3: Error bit CRC error Bit4: Start bit missing with EnDat transfer					
-	0	0		Long Integer	r
Multi-line: yes	Axis index		maxnumGlobMachAxes		

vaAbsoluteEnc2ZeroMonMax	$VA_ABSOLUTE_ENC_ZERO_MON_MAX[Achse]				
Enc2:Maximum of vaEnc2ZeroMonAct with absolute encoder					
-	0	0		UDoubleword	r
Multi-line: yes	Axis index		maxnumGlobMachAxes		

vaCecCompVal					
Axial sag compensation value					
mm, inch, degree, user defined	0			Double	r
Multi-line: yes	Axis index		maxnumGlobMachAxes		

vaCurr					
Drive actual current value					
-	0			Double	r
Multi-line: yes	Axis index		maxnumGlobMachAxes		

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

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

vaEnc1CompVal					
Leadscrew error compensation (LEC) value encoder 1					
mm, inch, degree, user defined	0			Double	r
Multi-line: yes	Axis index		maxnumGlobMachAxes		

vaEnc1ZeroMonAccessCnt	$VA_ENC_ZERO_MON_ACCESS_CNT[Achse]				
Enc1: Update counter					
-	0	0		UDoubleword	r
Multi-line: yes	Axis index		maxnumGlobMachAxes		

vaEnc1ZeroMonAct	$VA_ENC_ZERO_MON_ACT[Achse]				
Enc1: Zero monitoring values					
-	0	0		UDoubleword	r
Multi-line: yes	Axis index		maxnumGlobMachAxes		

vaEnc1ZeroMonErrCnt					
Enc 1: Error counter for zero mark monitoring					
-	0	0		Long Integer	r
Multi-line: yes	Axis index		maxnumGlobMachAxes		

vaEnc1ZeroMonInit	$VA_ENC_ZERO_MON_INIT[Achse]				
Enc1:Hardware counter value of the basic zero mark					
-	0	0		UDoubleword	r
Multi-line: yes	Axis index		maxnumGlobMachAxes		

vaEnc2CompVal					
Leadscrew error compensation (LEC) value encoder 2					
mm, inch, degree, user defined	0			Double	r
Multi-line: yes	Axis index		maxnumGlobMachAxes		

vaEnc2ZeroMonAccessCnt	$VA_ENC_ZERO_MON_ACCESS_CNT[Achse]				
Enc2: Update counter					
-	0	0		UDoubleword	r
Multi-line: yes	Axis index		maxnumGlobMachAxes		

vaEnc2ZeroMonAct	$VA_ENC_ZERO_MON_ACT[Achse]				
Enc2: Zero monitoring values					
-	0	0		UDoubleword	r
Multi-line: yes	Axis index		maxnumGlobMachAxes		

vaEnc2ZeroMonErrCnt					
Enc 2: Error counter for zero mark monitoring					
-	0	0		Long Integer	r
Multi-line: yes	Axis index		maxnumGlobMachAxes		

vaEnc2ZeroMonInit	$VA_ENC_ZERO_MON_INIT[Achse]				
Enc2:Hardware counter value of the basic zero mark					
-	0	0		UDoubleword	r
Multi-line: yes	Axis index		maxnumGlobMachAxes		

vaFoc					
Actual status of "ForceControl" 0: ForceControl not active 1: Modal ForceControl active 2: Non-modal ForceControl active					
-	0	0	2	UWord	r
Multi-line: yes	Axis index		maxnumGlobMachAxes		

vaFxs					
Actual status of "Travel to fixed stop" 0: Axis not at fixed stop 1: Successful travel to fixed stop 2: Unsuccessful travel to fixed stop 3: Travel to fixed stop selection active 4: Fixed stop has been detected 5: Travel to fixed stop deselection active					
-	0	0	5	UWord	r
Multi-line: yes	Axis index		maxnumGlobMachAxes		

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

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

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

vaLagError					
Axis following error					
-	0			Double	r
Multi-line: yes	Axis index		maxnumGlobMachAxes		

vaLoad					
Drive utilization in %					
-	0	-100	100	Double	r
Multi-line: yes	Axis index		maxnumGlobMachAxes		

vaPosctrlMode					
Position controller mode" 0: Position control 1: Speed control 2: Holding 3: Parking 4: Tracking					
-	0	0	4	UWord	r
Multi-line: yes	Axis index		maxnumGlobMachAxes		

vaPower					
Active drive power					
-	0			Double	r
Multi-line: yes	Axis index		maxnumGlobMachAxes		

vaPressureA					
Pressure on A end of the cylinder in bar (only for 611D Hydraulic)					
-	0			Double	r
Multi-line: yes	Axis index		maxnumGlobMachAxes		

vaPressureB					
Pressure on B end of the cylinder in bar (only for 611D Hydraulic)					
-	0			Double	r
Multi-line: yes	Axis index		maxnumGlobMachAxes		

vaSce					
Status of speed controller enable					
-	0	0	1	UWord	r
Multi-line: yes	Axis index		maxnumGlobMachAxes		

vaStopSi					
Stop from Safety Integrated -1: No stop 0: Stop A 1: Stop B 2: Stop C 3: Stop D 4: Stop E 5: Stop F 10: Test stop of NC 11: Test of ext. pulse suppression					
-	0			Long Integer	r
Multi-line: yes	Axis index		maxnumGlobMachAxes		

vaSyncDiff					
Actual value synchronism difference					
mm, inch, degree, user defined	0			Long Integer	r
Multi-line: yes	Axis index		maxnumGlobMachAxes		

vaSyncDiffStat					
Status of the actual value synchronism difference -4: Reserved -3: No valid value in $VA_SYNCDIFF, tangential control -2: No valid value in $VA_SYNCDIFF, master value coupling and simulated master value -1: No valid value in $VA_SYNCDIFF 0: No valid value in $VA_SYNCDIFF, coupling not active 1: Valid value in $VA_SYNCDIFF					
mm, inch, degree, user defined	0	-4	1	Long Integer	r
Multi-line: yes	Axis index		maxnumGlobMachAxes		

vaTempCompVal					
Axial temperature compensation value					
mm, inch, degree, user defined	0			Double	r
Multi-line: yes	Axis index		maxnumGlobMachAxes		

vaTorque					
Drive torque setpoint					
-	0			Double	r
Multi-line: yes	Axis index		maxnumGlobMachAxes		

vaTorqueAtLimit					
Status "effective torque equals specified torque limit" 0: Effective torque lower than torque limit 1: Effective torque has reached torque limit					
-	0	0	1	UWord	r
Multi-line: yes	Axis index		maxnumGlobMachAxes		

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

vaValveLift					
Actual valve lift in mm (only for 611D Hydraulic)					
-	0			Double	r
Multi-line: yes	Axis index		maxnumGlobMachAxes		

vaXfaultSi					
Stop F through cross-checking error active Bit 0 set: An actual value error has been discovered in the cross-check between NCK and 611D Bit 1 set: Some error has been discovered in the cross-check between NCK and 611D and the waiting time until Stop B ($MA_SAFE_STOP_SWITCH_TIME_F) is triggered is running or has expired					
-	0			Long Integer	r
Multi-line: yes	Axis index		maxnumGlobMachAxes		

1.3.7 Area N, Module 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		

acSmaxAcc	$AC_SMAXACC[]				
Active acceleration of the spindle This variable returns the active acceleration of the spindle for spindle mode. Bit 7 of $AC_SPIND_STATE (spindle accelerating) is set for the duration of the acceleration to the defined setpoint speed. Bit 8 of $AC_SPIND_STATE (spindle braking) is set for the duration of the braking to the defined setpoint speed. Apart from that, the acceleration-determining machine and setting data can be determined with the system variable $AC_SMAXACC_INFO. If the spindle is in axis mode, then $AC_SMAXACC does not return the current acceleration, instead the machine data (MAX_AX_VELO, MAX_AX_ACCEL, ...) typical for axis mode are active.					
Rev/s2, user defined				Double	r
Multi-line: yes	Spindle index		numSpindles		

acSmaxAccInfo	$AC_SMAXACC_INFO[]				
Identifier for the active spindle acceleration data This system variable provides additional information to $AC_SMAXACC and returns the definitive machine data as identifier/index. The index can be used to determine the active acceleration data on the basis of the following table of existing spindle accelerations. The number range is oriented to the system variable $AC_SMAXVELO_INFO: 0: No acceleration limitation (SERUPRO) 1: Not used 2: Acceleration in speed control mode in the current gear stage without position control, MD 35200 GEAR_STEP_SPEEDCTRL_ACCEL * ACC * ACCFXS 3: Acceleration in the current gear stage with active position control (e.g. SPCON, SPOS) MD 35210 GEAR_STEP_POSCTRL_ACCEL * ACC * ACCFXS 4: Acceleration in the current gear stage with active position control while tapping (G331, G332 - only with corresponding configuration of the second data block) MD 35212 GEAR_STEP_POSCTRL_ACCEL2 5: Not used 6: Not used 7: Not used 8: Acceleration limited by the preparation for thread cutting and with synchronous spindle coupling of the leading spindle on account of limited acceleration of the following spindle. 9: Not used 10: Not used 11: Not used 12: Acceleration limited by tool parameter $TC_TP_MAX_ACCEL 13: Not used					
-				Long Integer	r
Multi-line: yes	Spindle index		numSpindles		

acSmaxVelo	$AC_SMAXVELO[]				
Maximum spindle speed This variable returns the maximum spindle speed for spindle mode. This is formed from the smallest active speed limitation, and cannot be exceeded by speed programming or override > 100%. A speed limitation is indicated by the VDI interface signal DB31..,DBX83.1 'Setpoint speed limited' and by $AC_SPIND_STATE, bit 5 (speed limitation active). The cause of the speed limitation (machine, setting data, G code, VDI interface signal etc.) can also be determined with the system variable $AC_SMAXVELO_INFO. If the spindle is in axis mode, then the speed is not limited by $AC_SMAXVELO but instead the machine data (MAX_AX_VELO, ...) typical for axis mode are active.					
rev/min, user defined				Double	r
Multi-line: yes	Spindle index		numSpindles		

acSmaxVeloInfo	$AC_SMAXVELO_INFO[]				
Identifier (index) for the speed-limiting data (machine/setting data, etc.) The system variable provides additional information about $AC_SMAXVELO, and returns the definitive data (machine, setting data, G code, VDI interface, etc.) as identifier/index. The speed-limiting data can be determined with the index from the following table of existing spindle speed limitations . 0: No limitation (SERUPRO) 1: Speed limited by clamping arrangement (chuck speed), MD 35100 SPIND_VELO_LIMIT; 2: Speed limitation (upper limit, that is maximum speed) in the current gear stage, MD 35130 GEAR_STEP_MAX_VELO_LIMIT * VELOLIMA (maximum speed factor from part program) 3: Speed limitation in the current gear stage with position control active MD 35132 GEAR_STEP_PC_MAX_VELO_LIMIT * VELOLIMA (among other items with SPCON, SPOS) 4: Speed limited to SD 43220 SPIND_MAX_VELO_G26 (part program instruction: G26 S.. or PI service from HMI) 5: Speed limited by set VDI interface signal DB31,...DBX3.6 to MD 35160 SPIND_EXTERN_VELO_LIMIT 6: Speed limited at constant cutting speed by part program instruction: LIMS (G96, G97, G961) SD 43230 SPIND_MAX_VELO_LIMS; 7: Speed limited by Safety Integrated to safe speed (SG) 8: Speed limited by the preparation for thread cutting and synchronous spindle coupling of the leading spindle on account of the speed limitation of the following spindle. 9: Speed limited to the maximum speed of the drive by the drive parameters, e.g. SINAMICS: p1082. 10: Speed limited to MD 36300 ENC_FREQ_LIMIT with functions that require a functioning measuring system, e.g. position control and for the master spindle with the G functions G95, G96, G97, G33..35. The limitation takes into account the encoder speed, the MS arrangement (direct/indirect), MS limiting frequency and the current parameter set. 11: Limitation of the overlaid movement of the following spindle to the dynamic remaining after the coupling. 12: Speed limited by tool parameter (in preparation) 13: Speed limited by tool carrier (in preparation)					
-				Long Integer	r
Multi-line: yes	Spindle index		numSpindles		

acSminVelo	$AC_SMINVELO[]				
Minimum spindle speed This variable returns the minimum spindle speed for speed control mode. This is formed from the highest active speed increase, and cannot be undershot by speed programming or override < 100%. A speed increase is indicated by the VDI interface signal DB31..,DBX83.2 'Setpoint speed increased' and by $AC_SPIND_STATE, bit 6 (speed increase active). The cause of the speed increase (machine, setting data, G code, VDI interface signal etc.) can also be determined with the system variable $AC_SMINVELO_INFO. If the spindle is in axis or positioning mode, then the speed is not increased by $AC_SMINVELO.					
rev/min, user defined				Double	r
Multi-line: yes	Spindle index		numSpindles		

acSminVeloInfo	$AC_SMINVELO_INFO[]				
Identifier (index) for the speed-limiting data (machine/setting data, etc.) The system variable provides additional information about $AC_SMAXVELO, and returns the definitive data (machine, setting data, G code, VDI interface, etc.) as identifier/index. The speed-limiting data can be determined with the index from the following table of existing spindle speed limitations. The system variable provides additional information about $AC_SMINVELO, and returns the speed increasing data (machine, setting data) as identifier/index. The speed-increasing data can be determined with the index from the following table of existing spindle speed increases.					
-				Long Integer	r
Multi-line: yes	Spindle index		numSpindles		

acSpindState	$AC_SPIND_STATE[]				
This variable returns the spindle status in speed control mode only (Bit0 =1). The variable $AA_INPOS_STAT is relevant in positioning and axis modes. $AC_SPIND_STATE returns no values in synchronous mode. Bit 0: "Speed control mode active" (condition for validity of the bits of the variable $AC_SPIND_STATE[) Bit 1: "Spindle stopped" (corresponds to VDI interface signal DB31..,DBX61.4) Bit 2: "Spindle programmed" (e.g. M3, M4 S.., FC18, ..) (corresponds to VDI interface signal DB31..,DBX64.4/5 or 6/7) Bit 3: "Spindle in set range" (corresponds to VDI interface signal DB31..,DBX83.5) Bit 4: Free Bit 5: "Limitation of setpoint speed active" remains as long as the programmed or overrride speed remains higher than the speed in the system variable $AC_SMAXVELO. Corresponds to VDI interface signal DB31..,DBX83.1 'Setpoint speed limited'. Bit 6: "Increasing setpoint speed active" remains as long as the programmed speed remains less than the speed in the system variable $AC_SMINVELO. Corresponds to VDI interface signal DB31..,DBX83.2 'Setpoint speed increasing' Bit 7: "Spindle accelerating" (setpoint side), remains as long as the spindle is accelerating to the defined setpoint speed. Bit 8: "Spindle braking" (setpoint side), remains as long as the spindle is braking to the defined setpoint speed or comes to a standstill. Bit 9: "Position control active"					
-				Long Integer	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		

dummy					
dummy					
-				UWord	r
Multi-line: yes	Spindle index		numSpindles		

gwpsActive	{$GWPS}				
SUG programming active (SUG is the function "constant perimeter speed of grinding wheel) 0 = inactive 1 = active					
-				UWord	r
Multi-line: yes	Spindle index		numSpindles		

index					
Absolute axis index referred to MD					
-				UWord	r
Multi-line: yes	Spindle index		numSpindles		

name					
Spindle name Note: If several logical spindles are referred to one physical spindle with active spindle conversion and access is made via area N of module SSP2, then the name of the first suitable logical spindle is output.					
-				String[32]	r
Multi-line: yes	Spindle index		numSpindles		

namePhys					
Name of assigned physical spindle, identical to "name" variable.					
-				String[32]	r
Multi-line: yes	Spindle index		numSpindles		

opMode					
Spindle mode 0 = spindle mode 1 = oscillation mode (gear step changeover) 2 = positioning mode 3 = synchronous mode 4 = axis mode					
-				UWord	r
Multi-line: yes	Spindle index		numSpindles		

pSMode					
Last programmed spindle mode 0: No spindle configured in channel or spindle is active in another channel or in use by the PLC (FC18) or by synchronized actions. 1: Speed control mode 2: Positioning mode 3: Synchronous mode 4: Axis mode					
-				UWord	r
Multi-line: yes	Spindle index		numSpindles		

pSModeS					
Last programmed spindle mode with block search 0: No spindle configured in channel or spindle is active in another channel or in use by the PLC (FC18) or by synchronized actions. 1: Speed control mode 2: Positioning mode 3: Synchronous mode 4: Axis mode					
-				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, Module 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		

acSmaxAcc	$AC_SMAXACC[]				
Active acceleration of the spindle This variable returns the active acceleration of the spindle for spindle mode. Bit 7 of $AC_SPIND_STATE (spindle accelerating) is set for the duration of the acceleration to the defined setpoint speed. Bit 8 of $AC_SPIND_STATE (spindle braking) is set for the duration of the braking to the defined setpoint speed. Apart from that, the acceleration-determining machine and setting data can be determined with the system variable $AC_SMAXACC_INFO. If the spindle is in axis mode, then $AC_SMAXACC does not return the current acceleration, instead the machine data (MAX_AX_VELO, MAX_AX_ACCEL, ...) typical for axis mode are active.					
Rev/s2, user defined				Double	r
Multi-line: yes	Logical spindle index		numSpindlesLog		

acSmaxAccInfo	$AC_SMAXACC_INFO[]				
Identifier for the active spindle acceleration data This system variable provides additional information to $AC_SMAXACC and returns the definitive machine data as identifier/index. The index can be used to determine the active acceleration data on the basis of the following table of existing spindle accelerations. The number range is oriented to the system variable $AC_SMAXVELO_INFO: 0: No acceleration limitation (SERUPRO) 1: Not used 2: Acceleration in speed control mode in the current gear stage without position control, MD 35200 GEAR_STEP_SPEEDCTRL_ACCEL * ACC * ACCFXS					

3: Acceleration in the current gear stage with active position control (e.g. SPCON, SPOS) MD 35210 GEAR_STEP_POSCTRL_ACCEL * ACC * ACCFXS 4: Acceleration in the current gear stage with active position control while tapping (G331, G332 - only with corresponding configuration of the second data block) MD 35212 GEAR_STEP_POSCTRL_ACCEL2 5: Not used 6: Not used 7: Not used 8: Acceleration limited by the preparation for thread cutting and with synchronous spindle coupling of the leading spindle on account of limited acceleration of the following spindle. 9: Not used 10: Not used 11: Not used 12: Acceleration limited by tool parameter $TC_TP_MAX_ACCEL 13: Not used

-				Long Integer	r
Multi-line: yes	Logical spindle index		numSpindlesLog		

acSmaxVelo	$AC_SMAXVELO[]		
Maximum spindle speed This variable returns the maximum spindle speed for spindle mode. This is formed from the smallest active speed limitation, and cannot be exceeded by speed programming or override > 100%. A speed limitation is indicated by the VDI interface signal DB31..,DBX83.1 'Setpoint speed limited' and by $AC_SPIND_STATE, bit 5 (speed limitation active). The cause of the speed limitation (machine, setting data, G code, VDI interface signal etc.) can also be determined with the system variable $AC_SMAXVELO_INFO. If the spindle is in axis mode, then the speed is not limited by $AC_SMAXVELO but instead the machine data (MAX_AX_VELO, ...) typical for axis mode are active.			
rev/min, user defined		Double	r
Multi-line: yes	Logical spindle index	numSpindlesLog	

acSmaxVeloInfo	$AC_SMAXVELO_INFO[]		
Identifier (index) for the speed-limiting data (machine/setting data, etc.) The system variable provides additional information about $AC_SMAXVELO, and returns the definitive data (machine, setting data, G code, VDI interface, etc.) as identifier/index. The speed-limiting data can be determined with the index from the following table of existing spindle speed limitations . 0: No limitation (SERUPRO) 1: Speed limited by clamping arrangement (chuck speed), MD 35100 SPIND_VELO_LIMIT; 2: Speed limitation (upper limit, that is maximum speed) in the current gear stage, MD 35130 GEAR_STEP_MAX_VELO_LIMIT * VELOLIMA (maximum speed factor from part program) 3: Speed limitation in the current gear stage with position control active MD 35132 GEAR_STEP_PC_MAX_VELO_LIMIT * VELOLIMA (among other items with SPCON, SPOS) 4: Speed limited to SD 43220 SPIND_MAX_VELO_G26 (part program instruction: G26 S.. or PI service from HMI) 5: Speed limited by set VDI interface signal DB31,...DBX3.6 to MD 35160 SPIND_EXTERN_VELO_LIMIT 6: Speed limited at constant cutting speed by part program instruction: LIMS (G96, G97, G961) SD 43230 SPIND_MAX_VELO_LIMS; 7: Speed limited by Safety Integrated to safe speed (SG) 8: Speed limited by the preparation for thread cutting and synchronous spindle coupling of the leading spindle on account of the speed limitation of the following spindle. 9: Speed limited to the maximum speed of the drive by the drive parameters, e.g. SINAMICS: p1082. 10: Speed limited to		

MD 36300 ENC_FREQ_LIMIT with functions that require a functioning measuring system, e.g. position control and for the master spindle with the G functions G95, G96, G97, G33..35. The limitation takes into account the encoder speed, the MS arrangement (direct/indirect), MS limiting frequency and the current parameter set. 11: Limitation of the overlaid movement of the following spindle to the dynamic remaining after the coupling. 12: Speed limited by tool parameter (in preparation) 13: Speed limited by tool carrier (in preparation)

-				Long Integer	r
Multi-line: yes	Logical spindle index		numSpindlesLog		

acSminVelo	$AC_SMINVELO[]				
Minimum spindle speed This variable returns the minimum spindle speed for speed control mode. This is formed from the highest active speed increase, and cannot be undershot by speed programming or override < 100%. A speed increase is indicated by the VDI interface signal DB31..,DBX83.2 'Setpoint speed increased' and by $AC_SPIND_STATE, bit 6 (speed increase active). The cause of the speed increase (machine, setting data, G code, VDI interface signal etc.) can also be determined with the system variable $AC_SMINVELO_INFO. If the spindle is in axis or positioning mode, then the speed is not increased by $AC_SMINVELO.					
rev/min, user defined				Double	r
Multi-line: yes	Logical spindle index		numSpindlesLog		

acSminVeloInfo	$AC_SMINVELO_INFO[]				
Identifier (index) for the speed-limiting data (machine/setting data, etc.) The system variable provides additional information about $AC_SMAXVELO, and returns the definitive data (machine, setting data, G code, VDI interface, etc.) as identifier/index. The speed-limiting data can be determined with the index from the following table of existing spindle speed limitations. The system variable provides additional information about $AC_SMINVELO, and returns the speed increasing data (machine, setting data) as identifier/index. The speed-increasing data can be determined with the index from the following table of existing spindle speed increases.					
-				Long Integer	r
Multi-line: yes	Logical spindle index		numSpindlesLog		

acSpindState	$AC_SPIND_STATE[]				
This variable returns the spindle status in speed control mode only (Bit0 =1). The variable $AA_INPOS_STAT is relevant in positioning and axis modes. $AC_SPIND_STATE returns no values in synchronous mode. Bit 0: "Speed control mode active" (condition for validity of the bits of the variable $AC_SPIND_STATE[) Bit 1: "Spindle stopped" (corresponds to VDI interface signal DB31..,DBX61.4) Bit 2: "Spindle programmed" (e.g. M3, M4 S.., FC18, ..) (corresponds to VDI interface signal DB31..,DBX64.4/5 or 6/7) Bit 3: "Spindle in set range" (corresponds to VDI interface signal DB31..,DBX83.5) Bit 4: Free Bit 5: "Limitation of setpoint speed active" remains as long as the programmed or overrride speed remains higher than the speed in the system variable $AC_SMAXVELO. Corresponds to VDI interface signal DB31..,DBX83.1 'Setpoint speed limited'. Bit 6: "Increasing setpoint speed active" remains as long as the programmed speed remains less than the speed in the system variable $AC_SMINVELO. Corresponds to VDI interface signal DB31..,DBX83.2 'Setpoint speed increasing' Bit 7: "Spindle accelerating" (setpoint side), remains as long as the spindle is accelerating to the defined setpoint speed. Bit 8: "Spindle braking" (setpoint side), remains as long as the spindle is braking to the defined setpoint speed or comes to a standstill. Bit 9: "Position control active"					
-				Long Integer	r
Multi-line: yes	Logical spindle index		numSpindlesLog		

actGearStage					
Actual gear stage of spindle					
-				UWord	r
Multi-line: yes	Logical spindle index		numSpindlesLog		

actSpeed					
Spindle speed actual value					
rev/min, user defined				Double	r
Multi-line: yes	Logical spindle index		numSpindlesLog		

channelNo					
Number of channel in which spindle is configured					
-				UWord	r
Multi-line: yes	Logical spindle index		numSpindlesLog		

cmdAngPos					
Spindle position (SPOS)					
Degree, user defined				Double	r
Multi-line: yes	Logical spindle index		numSpindlesLog		

cmdConstCutSpeed					
Constant cutting 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		

dummy					
dummy					
-				UWord	r
Multi-line: yes	Logical spindle index		numSpindlesLog		

gwpsActive					
SUG programming active (SUG is the function "constant perimeter speed of grinding wheel) 0 = inactive 1 = active					
-				UWord	r
Multi-line: yes	Logical spindle index		numSpindlesLog		

index					
Absolute axis index referred to MD					
-				UWord	r
Multi-line: yes	Logical spindle index		numSpindlesLog		

name					
Spindle name Note: If several logical spindles are referred to one physical spindle with active spindle conversion and access is made via area N of module SSP2, then the name of the first suitable logical spindle is output.					
-				String[32]	r
Multi-line: yes	Logical spindle index		numSpindlesLog		

namePhys					
Name of assigned physical spindle.					
-				String[32]	r
Multi-line: yes	Logical spindle index		numSpindlesLog		

opMode					
Spindle mode 0 = spindle mode 1 = oscillation mode (gear step changeover) 2 = positioning mode 3 = synchronous mode 4 = axis mode					
-				UWord	r
Multi-line: yes	Logical spindle index		numSpindlesLog		

pSMode					
Last programmed spindle mode 0: No spindle configured in channel or spindle is active in another channel or in use by the PLC (FC18) or by synchronized actions. 1: Speed control mode 2: Positioning mode 3: Synchronous mode 4: Axis mode					
-				UWord	r
Multi-line: yes	Logical spindle index		numSpindlesLog		

pSModeS					
Last programmed spindle mode with block search 0: No spindle configured in channel or spindle is active in another channel or in use by the PLC (FC18) or by synchronized actions. 1: Speed control mode 2: Positioning mode 3: Synchronous mode 4: Axis mode					
-				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, Module 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	diverse, siehe Bausteinbescheibung				PA
Translation of an active work offset (the physical unit is defined in basicLengthUnit in module Y in area N).					
mm, inch, user defined				Double	r
Multi-line: yes	Frame index * maxnumGlobMachAxes + axis number		6 * maxnumGlobMachAxes		

mirrorImgActive	diverse, siehe Bausteinbescheibung				PA
Mirroring enabled in an active work offset 0 = mirroring not active 1 = mirroring active					
-				UWord	r
Multi-line: yes	Frame index * maxnumGlobMachAxes + axis number		6 * maxnumGlobMachAxes		

rotation	diverse, siehe Bausteinbescheibung				PA
Rotation of an active work offset					
Degree				Double	r
Multi-line: yes	Frame index * maxnumGlobMachAxes + axis number		6 * maxnumGlobMachAxes		

scaleFact	diverse, siehe Bausteinbescheibung				PA
Scaling factor of an active work offset					
-				Double	r
Multi-line: yes	Frame index * maxnumGlobMachAxes + axis number		6 * maxnumGlobMachAxes		

1.3.10 Area N, Module 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 work 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 * 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 * maxnumGlobMachAxes + axis number		$MN_MM_NUM_GLOBAL_BASE_FRAMES * maxnumGlobMachAxes		

mirrorImgActive	$P_NCBFR[x ,MI] x=FrameNo, y=Axis				PA
Mirroring enabled in a settable work offset 0: Mirroring not active 1: Mirroring active					
-				UWord	wr
Multi-line: yes	Frame index * maxnumGlobMachAxes + axis number		$MN_MM_NUM_GLOBAL_BASE_FRAMES * maxnumGlobMachAxes		

rotation	$P_NCBFR[x,y,RT] x=FrameNo, y=Axis				PA
Rotation of a settable work offset					
Degree				Double	wr
Multi-line: yes	Frame index * 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 work offset					
-				Double	wr
Multi-line: yes	Frame index * maxnumGlobMachAxes + axis number		$MN_MM_NUM_GLOBAL_BASE_FRAMES * maxnumGlobMachAxes		

1.3.11 Area N, Module 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 work 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 * 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 * maxnumGlobMachAxes + axis number		$MN_MM_NUM_GLOBAL_USER_FRAMES * maxnumGlobMachAxes		

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

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

1.3.12 Area N, Module 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 column: Fanuc-G group number * 2					
-				String[16]	r
Multi-line: yes	Serial number		Gruppe_NUM		

Gruppe_NUM					
Number of Fanuc-G functions in the relevant group column: (2 * Fanuc-G group number) - 1					
-		0		UWord	r
Multi-line: yes	1		1		

1.3.13 Area B, Module 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)

autoJogState	$MC_AUTO_JOG_STATE				
Status of Automatic+JOG mode 1: Automatic is selected, $MN_JOG_MODE_MASK is set, and the mode group is in "Mode group reset". Thus can ??? by pressing the +/- buttons or the hand wheel in Auto. 2: This mode group has been switched internally to JOG on account of a JOG motion. VDI and OPI still show Automatic. 0: In all other cases					
-	0	0	2	UWord	r
Multi-line: yes	Mode group number		numBAGs		

ncAutoCounter					
Counter which is incremented with each 0->edge of the Auto key					
-	0	0		UWord	r
Multi-line: yes	Mode group number		numBAGs		

ncJogCounter					
Counter which is incremented with each 0->edge of the Jog key					
-	0	0		UWord	r
Multi-line: yes	Mode group number		numBAGs		

ncMDACounter					
Counter which is incremented with each 0->edge of the MDI key					
-	0	0		UWord	r
Multi-line: yes	Mode group number		numBAGs		

opMode	DB11, DBX6.0-6.2				
Active mode 0 = JOG 1 = MDI 2 = AUTO					
-				UWord	r
Multi-line: no					

readyActive	DB11, DBX6.3				
Code whether mode group is ready 0 = not ready 1 = ready					
-				UWord	r
Multi-line: no					

resetActive	DB11, DBX6.7				
Code whether all channels in mode group are in Reset 0 = not all channels in reset 1 = all channels in reset					
-				UWord	r
Multi-line: no					

1.3.14 Area N, Module SALAC: Alarm actions: List in rev. chronol. order, oldest alarm act. appears first

OEM-MMC: Linkitem

/Nck/AlarmEvent/...

In a given alarm, all values in the SALAC module are identical to the corresponding variables in the SALA, SALAP and SALAL modules with the exception of actionType and actionCount.
The same alarm can be found in the various modules by comparing the values of alarmNo.

A client is registered with an alarm server when cyclic reading of the SALAC module has been set.
If the operator panel sets cyclic reading when a data in the module changes and column index 0 has been specified, then the variable server sends the entire data block to the operator panel if the alarm server receives a new alarm action.

Another alarm server client is registered at each cyclic reading of the SALAC module which has been set. This mechanism therefore functions with more than one user interface connected. Registration is withdrawn when the corresponding cyclic reading is terminated. As cyclic reading only applies to registered clients,
each regular and noncyclic reading returns the default value for the requested variable.

actionCount					
A unique number assigned to the Alarm Action. It is reset to zero during NCK power on. It is incremented by one for each new Alarm Action.					
-	0			Long Integer	r
Multi-line: yes	1		1		

actionType					
Identifies whether the alarm is being cleared or activated. 0: No Alarm Action Pending 1: Alarm Set 2: Alarm Cleared					
-	0	0	2	Long Integer	r
Multi-line: yes	1		1		

alarmNo					
A unique number assigned to the alarm. It is incremented by one for each alarm that is reported.					
-	0			Long Integer	r
Multi-line: yes	1		1		

clearInfo	
Describes the delete criterion for the 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 mode groups (from SW 4.1)	

7: Alarm is cancelled by RESET in all channels of the NC (from SW 4.1)					
-	1	1	7	Long Integer	r
Multi-line: yes	1		1		

fillText1					
Parameter 1, ASCII text string that is inserted into the standard alarm text to supplement the alarm description.					
-	0			String[32]	r
Multi-line: yes	1		1		

fillText2					
Parameter 2, ASCII text string that is inserted into the standard alarm text to supplement the alarm description.					
-	0			String[32]	r
Multi-line: yes	1		1		

fillText3					
Parameter 3, ASCII text string that is inserted into the standard alarm text to supplement the alarm description.					
-	0			String[32]	r
Multi-line: yes	1		1		

fillText4					
Parameter 4, ASCII text string that is inserted into the standard alarm text to supplement the alarm description.					
-	0			String[32]	r
Multi-line: yes	1		1		

textIndex					
Identifies the text of the alarm description.					
-	0			Long Integer	r
Multi-line: yes	1		1		

timeBCD					
The date and time in BCD format for when the alarm occurred.					
-				Date+Time	r
Multi-line: yes	1		1		

1.4 State data of channel

1.4.1 Area C, Module M: Channel-specific machine data

OEM-MMC: Linkitem

/Channel/Drive/...

Channel-specific machine data

1.4.2 Area C, Module 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)

G0Mode	$AC_G0MODE				
G00 is active and $MC_G0_LINEAR_MODE is FALSE (Siemens mode) or $MC_EXTERN_G0_LINEAR_MODE is FALSE (ISO mode) and therefore non-linear interpolation is active with G0, i.e. the path axes are traversed as positioning axes. 0: G00 not active 1: G00 and linear interpolation active 2: G00 and non-linear interpolation active					
-	0	0	2	UWord	r
Multi-line: yes	1		1		

aGG	$A_GG				
active G function in synchronized action					
-	0	0		UWord	r
Multi-line: yes	Number of the G function group		Gruppe_NUM		

aLinkTransRate	$A_LINK_TRANS_RATE				
Link transfer rate Number of bytes that can still be transferred in the current IPO cycle via the NCU link communication.					
-		0		UWord	r
Multi-line: yes	1		1		

aMonifact	$A_MONIFACT				
Factor for tool life monitoring					
-	0	0		Double	r
Multi-line: yes	1		1		

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

aaAccLimA	$AA_ACCLIMA[a]				
Axial acceleration override in main run 1-200					
-	100	1	200	UWord	r
Multi-line: yes	(Axis index)		numMachAxes		

aaEgActive	$AA_EG_ACTIVE[a,b]				
Electronic gear: Link to the specified master axis is operative, i.e. activated. 0: Deactivated 1: Activated					
-	0	0	1	UWord	r
Multi-line: yes	(Axis index of the slave axis) * numMachAxes + (axis index of the master axis) + 1		numMachAxes * numMachAxes		

aaEgAx	$AA_EG_AX[n,a]				
Electronic gear: Axis number of nth master axis (1-n). (Axis index = axis number - 1) 1-numMachAxes					
-	0	1	numMachAxes	UWord	r
Multi-line: yes	(Axis index of slave axis) * 5 + (index of master axis) + 1		numMachAxes * 5		

aaEgDenom	$AA_EG_DENOM[a,b]				
Electronic gear: Denominator of link factor for the specified master axis. The link factor of the gear is the result of $AA_EG_NUMERA[a,b]/$AA_EG_DENOM[a,b].					
-	1			Double	r
Multi-line: yes	(Axis index of the slave axis) * numMachAxes + (axis index of the master axis) + 1		numMachAxes * numMachAxes		

aaEgNumLa	$AA_EG_NUM_LA[a]				
Electronic gear: Number of master axes specified with EGDEF. If the axis has not been specified with EGDEF as slave axis, the value is 0. 0-5					
-	0	0	5	UWord	r
Multi-line: yes	(Axis index of slave axis + 1)		numMachAxes		

aaEgNumera	$AA_EG_NUMERA[a,b]				
Electronic gear: Numerator of link factor for the specified master axis. The link factor of the gear is the result of $AA_EG_NUMERA[a,b]/$AA_EG_DENOM[a,b].					
-	0			Double	r
Multi-line: yes	(Axis index of the slave axis) * numMachAxes + (axis index of the master axis) + 1		numMachAxes * numMachAxes		

aaEgSyn	$AA_EG_SYN[a,b]				
Electronic gear: Synchronous position for the specified master axis.					
mm, inch, degree, user defined	0			Double	r
Multi-line: yes	(Axis index of the slave axis) * numMachAxes + (axis index of the master axis) + 1		numMachAxes * numMachAxes		

aaEgSynFa	$AA_EG_SYNFA[a]				
Electronic gear: Synchronous position for the slave axis.					
mm, inch, degree, user defined	0			Double	r
Multi-line: yes	(Axis index of slave axis + 1)		numMachAxes		

aaEgType	$AA_EG_TYPE[a,b]				
Electronic gear: Type of link for the specified master axis 0: Actual-value linkage 1: Setpoint linkage					
-	0	0	1	UWord	r
Multi-line: yes	(Axis index of the slave axis) * numMachAxes + (axis index of the master axis) + 1		numMachAxes * numMachAxes		

aaJerkLimA	$AA_JERKLIMA[a]				
Axial jerk override in run in 1-200					
-	100	1	200	UWord	r
Multi-line: yes	(Axis index)		numMachAxes		

aaMeasP1Valid	$AA_MEAS_P1_VALID				
Save axial measuring point P1 for workpiece and tool measurement 0: Clear axial measuring point 1: Write actual axial values to axial measuring point					
-	0	0	1	Long Integer	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	Long Integer	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	Long Integer	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	Long Integer	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	Long Integer	wr
Multi-line: yes	Axis index		numMachAxes		

aaSyncDiff	$AA_SYNCDIFF[]				
Setpoint synchronism difference for all types of coupling					
mm, inch, degree, user defined	0			Double	r
Multi-line: yes	Axis index of the following axis		numMachAxes		

aaSyncDiffStat	$VA_SYNCDIFF_STAT[]				
Status of setpoint synchronism difference -4: No valid value in $AA_SYNCDIFF, coupled motion from part program -3: Reserved -2: Reserved -1: No valid value in $AA_SYNCDIFF 0: No valid value in $AA_SYNCDIFF, coupling not active 1: Valid value in $AA_SYNCDIFF					
-	0	-4	1	Long Integer	r
Multi-line: yes	Axis index of the following axis		numMachAxes		

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		

acAsup	$AC_ASUP				
Code number for the reason for the activation of an ASUB The reasons are bit-coded. BIT0: Activation because of: User interrupt "ASUB with Blsync". BIT1: Activation because of: User interrupt "ASUB". BIT2: Activation because of: User interrupt "ASUB from channel state Ready". BIT3: Activation because of: User interrupt "ASUB in manual mode". BIT4: Activation because of: Activation because of: User interrupt "ASUB". BIT5: Activation because of: Abort of the subroutine repetition. BIT6: Activation because of: Activation of decoding single block. BIT7: Activation because of: Activation of DDTG. BIT8: Activation because of: Activation of axis synchronization. BIT9: Activation because of: Change of operating mode. BIT10: Activation because of: Program continuation under TeachIn or after TeachIn deactivation. BIT11: Activation because of: Selection overstore. BIT12: Activation because of: Alarm with reaction compensation block with REPOS (COMPBLOCKWITHREORG). BIT13: Activation because of: Retraction motion with G33 and Stop. BIT14: Activation because of: Activation of dry run feed. BIT15: Activation because of: Deactivation of dry run feed. BIT16: Activation because of: Activation of skip block. BIT17: Activation because of: Deactivation of skip block. BIT18: Activation because of: Activate machine data. BIT19: Activation because of: Activate tool offset. BIT20: Activation because of: System ASUB after search type SERUPRO has reached the search target.					
-	0	0		Long Integer	r
Multi-line: yes	1		1		

acAxCtSwA	$AC_AXCTSWA[CTn]				
Channel status of axis container rotation TRUE: The channel has enabled rotation for the axis container and rotation is still in progress. FALSE: Axis container rotation is already finished					
-	0	0	1	UWord	r
Multi-line: yes	Container no.		numContainer		

acConeAngle	$AC_CONE_ANGLE				
Currently effective taper angle for taper turning. The taper angle is specified by the setting data $SC_CONE_ANGLE and is effective in the operating mode JOG only.					
Degree	0	-90	90	Double	r
Multi-line: yes	1		1		

acDelt	$AC_DELT				
Stored distance-to-go of the path in the WCS after delete-distance-to-go of the path DELDTG for synchronous action (Note: for SYNACT only)					
-				Double	r
Multi-line: yes	1		1		

acDtbb	$AC_DTBB				
Distance from the beginning of the block in the BCS (Note: SYNACT only)					
-				Double	r
Multi-line: yes	1		1		

acDtbw	$AC_DTBW				
Distance from the beginning of the block in the WCS (Note: for SYNACT only)					
-				Double	r
Multi-line: yes	1		1		

acDteb	$AC_DTEB				
Distance to the end of the block in the BCS (Note: for SYNACT only)					
-				Double	r
Multi-line: yes	1		1		

acDtew	$AC_DTEW				
Distance to the end of the block in the WCS (Note: for SYNACT only)					
-				Double	r
Multi-line: yes	1		1		

acEsrTrigger	$AC_ESR_TRIGGER				
Activation of "NC-controlled ESR"					
-	0	0	1	UWord	r
Multi-line: yes	1		1		

acFGo	$AC_F_G0				
Max. rapid traverse rate in the block					
mm/min, inch/min, user defined	0	0		Double	r
Multi-line: yes	1		1		

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			Long Integer	r
Multi-line: yes	1		1		

acIwTu	$AC_IW_TU				
Current position of channel axes Bit-coded: Bit 0: Channel axis 1 position Bit 1: Channel axis 2 position Bit 2: Channel axis 3 position Bit 3: Channel axis 4 position ...					
-	0			Long Integer	r
Multi-line: yes	1		1		

acJogCircleSelected	$AC_JOG_CIRCLE_SELECTED				
JOG in circles has been selected					
-	0	0	1	UWord	r
Multi-line: yes	1		1		

acJogCoord	$AC_JOG_COORD				
Setting the coordinate system for the manual travel 0: Work 1: SZS					
-	0	0	1	Long Integer	wr
Multi-line: no					

acLiftFast	$AC_LIFTFAST				
Information about the execution of LIFTFAST. At the start of the LIFTFAST operation, the variable is set internally by the NC to the value "1". The variable must be reset to the initial state ($AC_LIFTFAST=0) by the evaluating program (if available), in order to be able to detect a subsequent LIFTFAST. 0: Initial state 1: LIFTFAST has been executed					
-	0	0	1	UWord	r
Multi-line: yes	1		1		

acMea	$AC_MEA				
Touch probe has switched No. of touch probe					
-	0	0	1	UWord	r
Multi-line: yes	No. of touch probe		2		

acMeasActPlane	$AC_MEAS_ACT_PLANE				
Plane setting for measurement calculation 0: G17, 1: G18, 2: G19					
-		0	2	Long Integer	wr
Multi-line: yes	1		1		

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

acMeasChsfr	$AC_MEAS_CHSFR				
System frame bit screen form for setting up the new frame					
-	0	0		Long Integer	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		

acMeasDNumber	$AC_MEAS_D_NUMBER				
Selected tool edge number					
-		0		Long Integer	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	Long Integer	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	Long Integer	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	Long Integer	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	Long Integer	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		Long Integer	wr
Multi-line: no					

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

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

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

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

acMeasPframe	$AC_MEAS_PFRAME				
Programmable frame is not included					
-	0	0	1	Long Integer	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		Long Integer	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	Long Integer	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		Long Integer	wr
Multi-line: no					

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

acMeasUifr	$AC_MEAS_UIFR				
Settable data management frame for setting up the new frame					
-	0	0	99	Long Integer	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		Long Integer	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		

acMonMin	$AC_MONMIN				
Ratio of the actual tool monitoring value to the setpoint					
-	0	0		Double	r
Multi-line: yes	1		1		

acMsNum	$AC_MSNUM				
Number of the master spindle 0: No spindle available 1..n: Number of the master spindle					
-	0	0		UWord	r
Multi-line: yes	1		1		

acMthNum	$AC_MTHNUM				
Number of the current master tool holder. Is only meaningful with active magazine management. 0: No master tool holder available 1..n: Number of the master tool holder					
-	0	0		UWord	r
Multi-line: yes	1		1		

acOvr	$AC_OVR				
Path override for synchronous actions (Note: for SYNACT only)					
-				Double	r
Multi-line: yes	1		1		

acPRTimeA					
For simulation: Estimation of program runtime in seconds - downtime					
s, user defined				Double	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		

acPathAcc	$AC_PATHACC				
Path acceleration for real-time events					
m/s2, 1000 inch/ s2, user defined	0	0		Double	r
Multi-line: yes	1		1		

acPathJerk	$AC_PATHJERK				
Path jerk for real-time events					
mm/s3, 1000 inch / s3, user defined	0	0		Double	r
Multi-line: yes	1		1		

acPathn	$AC_PATHN				
Normalized path parameter (Note: for SYNACT only)					
-				Double	r
Multi-line: yes	1		1		

acPlcOvr	$AC_PLC_OVR				
Path override for synchronized actions specified by the PLC					
-	100	0		Double	r
Multi-line: yes	1		1		

acPltbb	$AC_PLTBB				
Path length from the beginning of the block in the BCS (Note: for SYNACT only)					
-				Double	r
Multi-line: yes	1		1		

acPlteb	$AC_PLTEB				
Path length to the end of the block in the BCS (Note: for SYNACT only)					
-				Double	r
Multi-line: yes	1		1		

acProg	$AC_PROG				
Program status (identical to progStatus but with coding that corresponds to $AC_PROG) 0: aborted (reset) 1: halted (stop) 2: running (active) 3: waiting 4: interrupted					
-	0			UWord	r
Multi-line: yes	1		1		

acPtpSup					
Cartesian point-to-point travel (PTP) is supported by transformation 0: Cart. PTP travel is not supported 1: Cart. PTP travel is supported					
-	0	0	1	UWord	r
Multi-line: yes	1		1		

acSafeSynaMem	$AC_SAFE_SYNA_MEM				
Free Safety synchronized action elements The maximum number of elements is configured in $MC_MM_NUM_SAFE_SYNC_ELEMENTS					
-	0	0		UWord	r
Multi-line: yes	1		1		

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		

acSyncActLoad	$AC_SYNC_ACT_LOAD				
Current runtime for synchronized actions of the last IPO cycle in the channel					
-	0	0		Double	r
Multi-line: yes	1		1		

acSyncAverageLoad	$AC_SYNC_AVERAGE_LOAD				
Average runtime for synchronized actions of an IPO cycle in the channel					
-	0	0		Double	r
Multi-line: yes	1		1		

acSyncMaxLoad	$AC_SYNC_MAX_LOAD				
Longest runtime for synchronized actions of an IPO cycle in the channel					
-	0	0		Double	r
Multi-line: yes	1		1		

acTaneb	$AC_TANEB				
Tangent angle at the block end point					
-	0	0		Double	r
Multi-line: yes	1		1		

acTc	$AC_TC				
Active tool carrier					
-	0	0		Long Integer	r
Multi-line: yes	1		1		

acTcAckt	$AC_TC_ACKT				
Trigger variable ACKnowledgeTrigger always assumes a value of 1 for an IPO cycle when the PLC acknowledges a command of the tool management.					
-	0			Long Integer	r
Multi-line: yes	1		1		

acTcCmdt	$AC_TC_CMDT				
Trigger variable: CoMmadTrigger always assumes a value of 1 for an IPO cycle when a new command of the magazine management is output to the PLC.					
-	0			Long Integer	r
Multi-line: yes	1		1		

acThreadPitch	$AC_THREAD_PITCH				
Programmed lead					
-	0			Double	r
Multi-line: yes	1		1		

acThreadPitchAct	$AC_THREAD_PITCH_ACT				
Current lead					
-	0			Double	r
Multi-line: yes	1		1		

acThreadPitchInc	$AC_THREAD_PITCH_INC				
Current lead change					
-	0			Double	r
Multi-line: yes	1		1		

acTime	$AC_TIME				
Time from the beginning of the block in seconds (Note: for SYNACT only)					
s				Double	r
Multi-line: yes	1		1		

acTimec	$AC_TIMEC				
Time from the beginning of the block in interpolation cycles (Note: for SYNACT only)					
IPO cycle				Double	r
Multi-line: yes	1		1		

acTimer	$AC_TIMER[x] x = TimerNo				
Time variable in seconds (Note: for SYNACT only)					
s				Double	r
Multi-line: yes	Number of the time variable		$MN_MM_NUM_AC_TIMER		

acToolOAct	$AC_TOOLO_ACT				
Supplies the setpoint value of the current tool orientation. The orientation vector is normalized, i.e. it has the absolute value 1.					
-	0	-1		Double	r
Multi-line: no					

acToolODiff	$AC_TOOLO_DIFF				
Supplies the angle remaining between the current and end vectors of the tool orientation block					
-	0	0	180	Double	r
Multi-line: yes	1		1		

acToolOEnd	$AC_TOOLO_END				
Supplies the end orientation of the current block. The orientation vector is normalized, i.e. it has the absolute value 1.					
-	0	-1		Double	r
Multi-line: yes	1: X component		3		

acToolRAct	$AC_TOOLR_ACT				
Setpoint of the tool rotation. The orientation vector is normalized, i.e. it has the absolute value 1.					
-	0	-1	1	Double	r
Multi-line: yes	1: X component		3		

acToolRDiff	$AC_TOOLR_DIFF				
Angle remaining between the current and end rotation vectors of the tool orientation block					
-	0	0	180	Double	r
Multi-line: yes	1		1		

acToolREnd	$AC_TOOLR_END				
End rotation vector of the current block The orientation vector is normalized, i.e. it has the absolute value 1.					
-	0	-1	1	Double	r
Multi-line: yes	1: X component		3		

acTotalOvr	$AC_TOTAL_OVR				
Total path override for synchronized actions					
-	100	0		Double	r
Multi-line: yes	1		1		

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

acTrafoChain	$AC_TRAFO_CHAIN				
Active chained transformation Code numbers of the chained transformations of the active TRACON corresponding to machine data $MC_TRAFO_TYPE_m. 0: No master tool holder available 1..n: Number of the master tool holder					
-	0	0		UWord	r
Multi-line: yes	Index of the chained transformation		4		

acTrafoPar	$AC_TRAFO_PAR[n]				
Supplies the value of parameter 'n' of the current transformation, e.g. the cylinder diameter in the case of TRACYL					
-				Double	r
Multi-line: yes	Number of the parameter (dependent on the transformation type)		8		

acTrafoParSet	$AC_TRAFO_PARSET				
Number of current transformation data record. The variable is '0' if no transformation is active.					
-	0			UWord	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		

acVactBf	$AC_VACTBF				
Path velocity in the BCS. FGroup and FGREF are taken into account.					
mm/min, inch/min, user defined	0	0		Double	r
Multi-line: yes	1		1		

acVactWf	$AC_VACTWF				
Path velocity in the workpiece coordinate system. FGroup and FGREF are taken into account.					
mm/min, inch/min, user defined	0	0		Double	r
Multi-line: yes	1		1		

acVactw	$AC_VACTW				
Path velocity in the work piece coordinate system (Note: for SYNACT only)					
-				Double	r
Multi-line: yes	1		1		

acVc	$AC_VC				
Additive path feedrate correction value for synchronous actions (Note: for SYNACT only)					
-				Double	r
Multi-line: yes	1		1		

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

actDLNumberS	??				
Corresponds to actDLNumber for block search with calculation Caution: This variable is not available for the Variable Service, but only for logging in the case of block search events!.					
-				UWord	r
Multi-line: yes	1				

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

actDNumberFanuc					
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 work offset"). Frames 0 - 4 (corresponds to G500 ... G57) can be set in the standard version. The number of frames can be changed via machine data MM_NUM_USER_FRAMES. 0 = no frame selected 1 = G54 2 = G55 3 = G56 4 = G57 5 = G505 to 99 = G599					
-				UWord	r
Multi-line: no					

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		

actIpoType					
Active interpolation mode used for the path motion. This date corresponds to a large degree to the SNCF:ncFktBin for the first G-group. The value differs for automatically generated intermediate blocks only. This is e.g. the case if two lines are connected with an arc by the command RND. The value is the index of the active G-code (analog with SNCF:ncFktBin)					
-				UWord	r
Multi-line: yes	1		1		

actIpoTypeS					
Active mode of interpolation applied during block searches. This data is very similar to SNCF:ncFktBinS for the 1st G group. Its value is different only in the case of automatically generated intermediate blocks, such as when, for example, two straight lines are connected to an arc by means of command RND. The value is the index of the active G function (analogous to SNCF:ncFktBinS).					
-				UWord	r
Multi-line: yes	1		1		

actLanguage					
Active language mode 0: Siemens 1: ISO mode 2: Reserviert for later language expansions					
-				UWord	r
Multi-line: yes	1				

actMasterToolHolderNo					
Active number of the master tool holder. Especially for $MC_RESET_MODE_MASK, Bit0=0, this is the value of SETMS or SETMTH last programmed in the RESET status of the NCK. Especially for $MC_RESET_MODE_MASK, Bit0=1, this is the value in the RESET status of the NCK for $MC_SPIND_DEF_MASTER_SPIND (if $MC_TOOL_MANAGEMENT_TOOLHOLDER=0); or $MC_TOOL_MANAGEMENT_TOOLHOLDER (if $MC_TOOL_MANAGEMENT_TOOLHOLDER > 0)					
-		1	max. Anzahl der Kanalachsen	UWord	r
Multi-line: yes	1		1		

actOriToolLength1					
X component in workpiece coordinate system (WCS) of active tool length, taking into account the tool orientation, incl. adapter data, mirroring and TCARR (orientable toolholder).					
-	0			Double	r
Multi-line: yes	1		1		

actOriToolLength2					
Y component in workpiece coordinate system (WCS) of active tool length, taking into account the tool orientation, incl. adapter data, mirroring and TCARR (orientation-capable toolholder).					
-	0			Double	r
Multi-line: yes	1		1		

actOriToolLength3					
Z component in workpiece coordinate system (WCS) of active tool length, taking into account the tool orientation, incl. adapter data, mirroring and TCARR (orientation-capable toolholder).					
-	0			Double	r
Multi-line: yes	1		1		

actParts	$AC_ACTUAL_PARTS				
Total number of workpieces machined in current run: This counter registers the number of workpieces machined since it started. When the required number is reached, the counter is set to zero automatically.					
-	0			Double	wr
Multi-line: no					

actProgNetTime	$AC_ACT_PROG_NET_TIME				
The current net runtime of the current program, that is the time in which the program was stopped, is subtracted. actProgNetTime is automatically reset to zero with the part program start in automatic mode, chanel status RESET. The net runtime does not include the time during which the program pauses on account of override=0. actProgNetTime can be further manipulated with progNetTimeTrigger. Seconds					
s, user defined	0	0		Double	r
Multi-line: yes	1		1		

actTNumber	$P_TOOLNO				W1
Number of active tool					
-		0	32000	UWord	r
Multi-line: no					

actTNumberLong					
Number of the active tool using flat D-numbers with up to 8 digits					
-				Long Integer	r
Multi-line: yes	1		1		

actTNumberS					
Corresponds to actTNumber for block search with calculation. Attention: This variable is available for protocolling the block search events only, not for the Variable Service!					
-				UWord	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		

actWaCSCoordSys	$AC_WORKAREA_CS_COORD_SYSTEM				
Coordinate system of the active coordinate system-specific working area limitation Identifier for the coordinate system in which the working area limitation is to apply. The following are valid: 0: Working area limitation in the WCS 3: Working area limitation in the SZS					
-	0			UWord	r
Multi-line: yes	1		1		

actWaCSLimitMinus	$AC_WORKAREA_CS_LIMIT_MINUS				
Position of the coordinate system-specific working area limitation in the negative direction for the addressed axis and working area group. Position of the working area limitation in the negative direction					
mm, inch, degree, user defined	0			Double	r
Multi-line: yes	Channel axis index		numMachAxes		

actWaCSLimitPlus	$AC_WORKAREA_CS_LIMIT_PLUS				
Position of the coordinate system-specific working area limitation in the positive direction for the addressed axis and working area group Position of the working area limitation in the positive direction					
mm, inch, degree, user defined	0			Double	r
Multi-line: yes	Channel axis index		numMachAxes		

actWaCSMinusEnable	$AC_WORKAREA_CS_MINUS_ENABLE				
The coordinate-specific working area limitation in the negative direction of actWaCSLimitMinus is valid. TRUE: The value in the variable actWaCSLimitMinus is valid for the axis. FALSE: There is no limitation in the coordinate-specific working area in the negative direction for this axis.					
-	0			UWord	r
Multi-line: yes	Channel axis index		numMachAxes		

actWaCSPlusEnable	$AC_WORKAREA_CS_PLUS_ENABLE				
The coordinate system-specific working area limitation in the positive direction of actWaCSLimitPlus is valid. TRUE: The value in the variable actWaCSLimitPlus is valid for the axis. FALSE: There is no limitation for this axis in the positive direction in the coordinate-specific working area.					
-	0			UWord	r
Multi-line: yes	Channel axis index		numMachAxes		

actWalimGroupNo	$AC_WORKAREA_CS_GROUP				
Active working area group in the IPO Writing is only possible if the channel has been aborted or stopped 0: Not activated n: $MC_MM_NUM_WORKAREA_CS_GROUPS					
-	0	0	10	UWord	wr
Multi-line: yes	1		1		

allAxesRefActive	DB21-28, DBX36.2				
Code specifying whether all axes are referenced 1 = all axes referenced 0 = at least 1 axis not referenced					
-				UWord	r
Multi-line: no					

allAxesStopped					
Code specifying whether axes are in exact stop 0 = at least one axis is not in exact stop 1 = All axes in exact stop					
-				UWord	r
Multi-line: no					

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

blockType	$AC_BLOCKTYPE				
Identifies the type of a block (programmed or generated internally) 0: No internally generated block 1: Internally generated block, but cannot be specified in detail 2: Block was generated on chamfering/rounding 3: Smooth approach and retraction (SAR) 4: Block was generated during tool offset 5: Block was generated on smoothing 6: Block was generated by TLIFT (tangential correction) 7: Block was generated during path segmentation 8: Block was generated by compile cycles					
-	0	0	8	Long Integer	r
Multi-line: yes	1		1		

blockTypeInfo	$AC_BLOCKTYPEINFO
Detailed information on block type The value range and the meaning of this variable depend on the current value of system variable blockType 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		Long Integer	r
Multi-line: yes	1		1		

cIn	$C_IN[n]				
Signal from PLC to cycle (reserved for SIEMENS application, e.g. ShopMill/ManualTurn)					
-				UWord	r
Multi-line: yes	No. of input signal		16		

cOut	$C_OUT[n]				
Signal from cycle to PLC (reserved for SIEMENS application, e.g. ShopMill/ManualTurn)					
-				UWord	r
Multi-line: yes	No. of output signal		16		

chanAlarm	DB21-28, DBX36.6 und DBX36.7				A2
Code whether NCK alarm pending. 0 = no alarm in this channel 1 = alarm without stop 2 = alarm with stop					
-				UWord	r
Multi-line: no					

chanAxisNoGap					
Display of existing axes, i.e. no axis gap in channel. Bits 0-31 represent the axes of the channel. Bitn = 0: Axis does not exist. Bitn = 1: Axis does exist.					
-		0		Long Integer	r
Multi-line: yes	1		1		

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

changeAxConfCounter					
A counter which is incremented if the axes configuration has changed. This is the case, if e.g. geometry axes are switched or axes have been changed between channels. The counter is set to 0 at PowerOn and it might overflow. You cannot be sure, that the axes configuration actually has changed when the counter is incremented.					
-				UWord	r
Multi-line: yes	1		1		

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

cmdFeedRateIpo	$AC_F				
Desired feedrate of the interpolation feedrate. The physical unit is defined in the variable 'feedRateIpoUnit'					
mm/min, inch/min, user defined				Double	r
Multi-line: no					

cmdFeedRateIpoS					
Interpolation feed during search. The physical unit is in the variable feedRateIpoUnitS					
mm/min, inch/min, user defined	0	0		Double	r
Multi-line: yes	1		1		

cmdTrafoParS	$P_TRAFO_PAR[n]				
Supplies the value of parameter 'n' of the programmed transformation, e.g. the cylinder diameter in the case of TRACYL					
-				Double	r
Multi-line: yes	Number of the parameter (dependent on the transformation type)		8		

cmdTrafoParSetS	$P_TRAFO_PARSET				
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					

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					

enableOvrRapidFactor					
Activate additional rapid traverse override $SC_OVR_RAPID_FACTOR 0: not activated 1: activated					
-	0	0	1	UWord	wr
Multi-line: yes	1		1		

extProgActive	DB21-28, DBB32.0				
Flag indicating whether program execution from external is active. 0 = inactive 1 = active					
-				UWord	r
Multi-line: no					

feedRateIpoOvr					
Interpolation feedrate, override					
%				Double	r
Multi-line: no					

feedRateIpoUnit					
Interpolation feedrate, units 0 = mm/min 1 = mm/rev 2 = inch/min 3 = inch/rev					
-				UWord	r
Multi-line: no					

feedRateIpoUnitS					
Interpolation feed, units during search 0 = mm/min 1 = mm/rev 2 = inch/min 3 = inch/rev					
-	0	0		UWord	r
Multi-line: yes	1		1		

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

incoapB	$P_INCOAP_B				
Boolean supply and return parameter(s) of the COA application cutting generator					
-	0	0	1	UWord	wr
Multi-line: yes	Array index		incoapSize[1]		

incoapC	$P_INCOAP_C				
CHAR supply and return parameter(s) of the COA application cutting generator					
-	0	0	255	UWord	wr
Multi-line: yes	Array index		incoapSize[2]		

incoapI	$P_INCOAP_I				
INT supply and return parameter(s) of the COA application cutting generator					
-	0			UDoubleword	wr
Multi-line: yes	Array index		incoapSize[3]		

incoapR	$P_INCOAP_R				
DOUBLE supply and return parameter(s) of the COA application cutting generator					
-	0			Double	wr
Multi-line: yes	Array index		incoapSize[4]		

incoapS16	$P_INCOAP_S16[]				
CHAR16 supply and return parameter(s) of the COA application cutting generator					
-	0			String[16]	wr
Multi-line: yes	Array index		incoapSize[5]		

incoapS160	$P_INCOAP_S160[]				
CHAR160 supply and return parameter(s) of the COA application cutting generator					
-	0			String[160]	wr
Multi-line: yes	Array index		incoapSize[6]		

incoapS32	$P_INCOAP_S32[]				
CHAR32 supply and return parameter(s) of the COA application cutting generator					
-	0			String[32]	wr
Multi-line: yes	Array index		incoapSize[6]		

incoapSize	$P_INCOAP_SIZE[]				
Array size of the supply and return parameters of the COA application cutting generator					
-	0	0		UWord	wr
Multi-line: yes	1: Array size of $incoapB		7		

isoActHDNo	$P_ISO2_HNO[n],$P_ISO2_DNO,$P_ISO3_NO				
The following applies to rows 1- 4: This value is only practical if ISO2 mode is permissible. This value contains the H number of the tool length offsets in the 3 geometry dimensions in rows 1 - 3 and the D number of the tool radius offset. If H99 is programmed, then all 3 geometry dimensions (=rows 1-3) have the value "-1" the radius (= row 4) has the value "-1" If H=D ($MN_EXTERN_TOOLPROG_MODE,Bit6=0), then these variables contain the last programmed D or H. If an offset D > 1 is selected in Siemens mode, all rows have the value "-2". If ISO2 mode cannot be activated ($MN_MM_EXTERN_CNC_SYSTEM != 4), the variable has the value=-3. The following applies to row 5: This value is only practicable if ISO3 mode is permissible. This value contains the current number of tool offsets in ISO3 mode. If an offset D > 1 is selected in Siemens mode, the value "-2" is returned. If ISO3 mode cannot be activated ($MN_MM_EXTERN_CNC_SYSTEM != 5), the variable value =-3. -3: ISO2 or ISO3 mode inactive -2: Siemens offset selected with D > 1. -1: H99 programmed in ISO mode, Siemens offset D1 active					
-	0			short Integer	r
Multi-line: yes	1: H number for L1 in ISO2 mode		5		

ludAccCounter					
Counter indicating that a new LUD ACC is available. If subprograms are called during an automatic program execution, a new set of LUDs becomes valid. In order to indicate to the 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		

ncProgEndCounter					
Counter which is incremented as soon as the NCK has processed an end of program.					
-	0	0		UWord	r
Multi-line: yes	1		1		

ncResetCounter					
Counter which is incremented with each 0->edge of the Reset key					
-	0	0		UWord	r
Multi-line: yes	1		1		

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

oldProgNetTime	$AC_OLD_PROG_NET_TIME				
oldProgNetTime is the net runtime of the just correctly ended program. That is the program was not canceled with RESET, but terminated normally with M30. If a new program is started, oldProgNetTime remains unaffected until M30 is reached again. The implicit procedure of copying actProgNetTime to oldProgNetTime takes place only if progNetTimeTrigger is not written. oldProgNetTime is reset to zero with the PI "Select program". Seconds					
s, user defined	0	0		Double	r
Multi-line: yes	1		1		

oldProgNetTimeCounter	$AC_OLD_PROG_NET_TIME_COUNT				
This is zero in the power ON status. oldProgNetTimeCounter is always increased when the NCK has newly written oldProgNetTime. This enables the user to ensure that oldProgNetTime has been written, that is, if the user cancels the current program with reset, oldProgNetTime and oldProgNetTimeCounter remain unchanged. Note: Two programs running consecutively can have identical runtimes and be correctly terminated. The user can then only detect this by the changed oldProgNetTimeCounter. Modification counter					
-	0	0		UWord	r
Multi-line: yes	1		1		

pCutInv	$AC_CUT_INV				
States that a turning tool is rotated against the machining plane (typically through 180 degrees around the C axis with G18) such that the direction of spindle rotation has to be inverted. FALSE, TRUE					
-	0	0	1	UWord	r
Multi-line: yes	1		1		

pCutInvS	$AC_CUT_INV				
States that a turning tool is rotated against the machining plane (typically through 180 degrees around the C axis with G18) such that the direction of spindle rotation has to be inverted. For block search. FALSE, TRUE					
-	0	0	1	UWord	r
Multi-line: yes	1		1		

pCutMod	$AC_CUTMOD				
Reads the current valid value that was last programmed with the language command CUTMOD (number of the tool carrier for which the cutting edge data modification is to be activated). If the last programmed value was CUTMOD = -2 (activation with the currently active tool carrier with orientation capability), the value -2 is not returned but the number of the active tool carrier with orientation capability at the time of programming. -2, 999999					
-	0	0	1	Long Integer	r
Multi-line: yes	1		1		

pCutModS	$AC_CUTMOD				
Reads the current valid value that was last programmed with the language command CUTMOD (number of the tool carrier for which the cutting edge data modification is to be activated). If the last programmed value was CUTMOD = -2 (activation with the currently active tool carrier with orientation capability), the value -2 is not returned but the number of the active tool carrier with orientation capability at the time of programming. For block search -2, 999999					
-	0	0	1	Long Integer	r
Multi-line: yes	1		1		

pEgBc	$P_EG_BC[a]				
Electronic gear: Block change criterion. Important for EGON, EGONSYN 0: NOC Block change is performed immediately 1: IPOSTOP Block change is performed with setpoint 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	Long Integer	r
Multi-line: no			1		

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

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		

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		Long Integer	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		3		

pToolRot	$P_TOOLROT				
Current tool rotation The rotation vector is normalized, i.e. it has the absolute value 1.					
-	0	-1	1	Double	r
Multi-line: yes	1: X component		3		

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		

progDuploNumber					
Duplo number of programmed tool (does not yet have to be active)					
-	0			UWord	r
Multi-line: no			1		

progNetTimeTrigger	$AC_PROG_NET_TIME_TRIGGER				
Serves for the selective measurement of program sections, that is the time measurement can be switched on and off again by the program by writing progNetTimeTrigger. Certain values of progNetTimeTrigger are given a special function in order to fully exploit all trigger options: 0 Neutral: The trigger is not active, the value is taken from reset with the start key. 1 Terminate: Terminates the measurement and copies actProgNetTime -> oldProgNetTime. actProgNetTime is set to zero and then runs on again. 2 Start: Starts the measurement and sets actProgNetTime to zero. oldProgNetTime remains unchanged. 3 Stop: Stops the measurement. Does not change oldProgNetTime and holds actProgNetTime constant until resume. 4 Resume: Resumption of the measurement, that is a previously stopped measurement is resumed. actProgNetTime runs on. oldProgNetTime remains unchanged.					
-	0	0	4	UWord	r
Multi-line: yes	1		1		

progStatus	DB21-28, DBX35.0 - DBX35.4				K1
Program status 1 = interrupted 2 = stopped 3 = in progress 4 = waiting 5 = aborted					
-				UWord	r
Multi-line: no					

progTNumber					
Number of programmed tool					
-				UWord	r
Multi-line: no					

progTNumberLong					
Number of the programmed tool using flat D-numbers with up to 8 digits					
-	0			Long Integer	r
Multi-line: yes	1		1		

progToolIdent					
Identifier of programmed tool (does not yet have to be active)					
-	"\0"			String[32]	r
Multi-line: no			1		

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		

protocHmiEvent					
Logging: When writing, the defined event is activated during preprocessing. 49: HMI_TRIG_1 50: HMI_TRIG_2 51: HMI_TRIG_3					
-		0		UWord	wr
Multi-line: yes	User No. (1-10)		10		

protocUserActive	$MM_PROTOC_USER_ACTIVE				
Logging: Displays active users 0: User inactive 1: User active					
-	0	0	1	UWord	r
Multi-line: yes	User No. (1-10)		10		

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

remainDwellTime					
Remaining dwell time See timeOrRevolDwell					
-	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		

searchRunMode					
Type of function in which search run has been integrated 1: Search run is used directly 2: Simulation search run 3: Execute program area The user can preselect a program area via HMI with "Execute program area", which he/she would like to execute "real" on the machine. To do this, the NCK uses an internal block search to approach the start of the program area (abbreviation: APb) correctly. Internal cancellation at the end of the program area (abbreviation: EPb) via reset. 0: Otherwise					
-	0	1	3	UWord	r
Multi-line: yes	Axis index of the following axis		numMachAxes		

searchRunStatus					
Status of the search run 1: activeSearchRun Simulation active, that is the NCK simulates the part program from the start to the agreed search target (or APb) in order to find, among other things, the correct starting position of the search target block. 2: targetFound The search target has been found and the NCK is waiting for the Start button. Simulation has finished. 3: activeAdaption After the start, the NCK outputs action blocks which set the machine to the search target (M± function output, spindle speeds) and, if applicable, starts an ASUB in which the user uses the ASUB program to adapt the machine to the part program situation in the target block. (For example, a programmed tool is read and a tool-changing cycle exchanges it with the current tool.) The NCK stops automatically after the action blocks or after the ASUB with alarm 10208. 4: finishedAdaption The NCK waits for the start. 5: activeStopRun After the adaptation, the REPOS function goes to the target block, and then the execution of the program is resumed. The NCK executes the program area after the target block, but is still within the function Execute program area. The blocks are scanned to see whether the end of the program area (EPb) could already have been reached. The program is cancelled at EPb with reset and searchRunStatus is cleared. 0: Otherwise					
-	0	1	5	UWord	r
Multi-line: yes	1		1		

seruproMasterChanNo					
The search type SERUPRO (search via program testing) may be started simultaneously in several channels in order to start a channel grouping correctly. A search target must be specified in one channel (master channel) in the grouping. The other channels do not need a search target, they wait until they have reached a stop condition and the master channel has reached the search target. These channels generally stop at WAIT marks. The variable seruproMasterChanNo defines the master channel.					
-	0	0	numChannels	UWord	wr
Multi-line: yes	1		1		

seruproMasterNcuNo					
The search type SERUPRO (search via program testing) may be started simultaneously in several channels in order to start a channel grouping correctly. A search target must be specified in one channel (master channel) in the grouping. The other channels do not need a search target, they wait until they have reached a stop condition and the master channel has reached the search target. These channels generally stop at WAIT marks. The variable seruproMasterChanNo defines the master channel. seruproMasterNcuNo specifies the master channel in more detail if it is not on the active NCU.					
-	0	0	$MN_MM_LINK_NUM_OF_MODULES	UWord	wr
Multi-line: yes	1		1		

simTolerance	keine				
The VNCK can simulate part programs at a higher speed (see PI _N_NCKMOD). Only then is simTolerance evaluated and it ONLY affects geometry blocks that have not been programmed as circles or straight lines. These blocks are slowed down so that two successive interpolation points can be connected by a straight line. This straight line deviates from the programmed contour by a maximum of the 'simTolerance'.					
mm, inch, user defined	0			Double	wr
Multi-line: no					

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	Long Integer	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_STLO).					
-	0	0		UWord	r
Multi-line: yes	1		1		

stopCond					
Replaced by stopCondNew					
-	0	0		UWord	r
Multi-line: yes	1		1		

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

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

stopCondNum					
Number of active stop states. Specifies the number of occupied lines in stopCond					
-				UWord	r
Multi-line: yes	1		1		

stopCondPar					
Replaced by stopCondParNew					
-				UWord	r
Multi-line: yes	1				

stopCondParA					
Stop state parameter(s). More than one stop state can be active simultaneously. The highest priority stop state appears below the first line, this is followed by those with lower priorities.					
-				String[32]	r
Multi-line: yes	High byte: No. of the active stop state				

stopCondParNew					
Stop state parameter(s). More than one stop state can be active simultaneously. The highest priority stop state appears below the first line, this is followed by those with lower priorities.					
-				UWord	r
Multi-line: yes	High byte: No. of the active stop state				

stopCondTime					
BCD time stamp for stop state. More than one stop state can be active simultaneously. The highest priority stop state appears below the first line, this is followed by those with lower priorities.					
-				Date+Time	r
Multi-line: yes	Number of the stop state		stopCondNum		

stopRunActive					
Stop run active 0 = inactive 1 = active					
-	0	0	1	UWord	r
Multi-line: yes	1		1		

stopRunCounter					
Modification counter for stop run. This is always incremented when the NCK has stopped at a stop block.					
-	0	0		UWord	r
Multi-line: yes	1		1		

suppProgFunc					
Disabling of language commands Bit0 = 0: SBLOF command is active Bit0 = 1: SBLOF command is disabled					
-	Bit0 = 0			UWord	wr
Multi-line: yes	1		1		

syntaxCheckAlarmNo					
Alarm number in the case of a syntax error during the syntax check					
-	0	0		Long Integer	r
Multi-line: yes	1		1		

syntaxCheckAlarmPara1					
Parameter 1 for an alarm in the case of a syntax error during the syntax check					
-	0	0		String[32]	r
Multi-line: yes	1		1		

syntaxCheckAlarmPara2					
Parameter 2 for an alarm in the case of a syntax error during the syntax check					
-	0	0		String[32]	r
Multi-line: yes	1		1		

syntaxCheckAlarmPara3					
Parameter 3 for an alarm in the case of a syntax error during the syntax check					
-	0	0		String[32]	r
Multi-line: yes	1		1		

syntaxCheckAlarmPara4					
Parameter 4 for an alarm in the case of a syntax error during the syntax check					
-	0	0		String[32]	r
Multi-line: yes	1		1		

syntaxCheckSeek					
Line number of the faulty line in the syntax check					
-	0	0		Long Integer	r
Multi-line: yes	1		1		

syntaxCheckStatus					
Status of the "Syntax check" function 0: Syntax check not active (initialization status) 1: Syntax check selected 2: Syntax check active 3: Syntax check stopped with alarm on account of system error 4: Syntax check terminated 5: Syntax check canceled 6: Syntax check canceled on account of errors					
-	0	0	6	UWord	r
Multi-line: yes	1		1		

tOffL1L2L3	$AC_TOFFL				
Tool length offset programmed in coordinates of the tool length components L1/L2/L3.					
mm, inch, degree, user defined	0			Double	r
Multi-line: yes	1: Tool length offset programmed in the tool length component direction L1		3		

tOffLXYZ	$AC_TOFF				
Tool length offset TOFFL programmed in coordinates of the WCS.					
mm, inch, degree, user defined	0			Double	r
Multi-line: yes	1: Tool length offset programmed in the direction of the 1st geometry axis		3		

tOffR	$AC_TOFFR				
Programmed tool radius offset.					
mm, inch, degree, user defined	0			Double	r
Multi-line: yes	1		1		

threadPitch					
Current lead					
-	0	0		Double	r
Multi-line: yes	1		1		

threadPitchS					
Current lead during search run					
-	0	0		Double	r
Multi-line: yes	1		1		

timeOrRevolDwell					
Dwell time unit in seconds or spindle revolutions 0: cmdDwellTime and remainDwellTime in seconds 1: cmdDwellTime and remainDwellTime in spindle revolutions					
-	0	0	1	UWord	r
Multi-line: yes	1		1		

timeS	$AC_TIMES				
Time after a block change between programmed blocks in seconds Each programmed block can be divided up into a chain of part blocks that are processed one after the other. O n l y with the 1st cycle of the 1st block of the chain, timeS is set to zero and then counted up in seconds. Therefore, the variable enables time measurements throughout the entire block chain.					
s, user defined	0	0		Double	r
Multi-line: yes	1		1		

timeSC	$AC_TIMESC				
Time after a block change between programmed blocks in IPO cycles Each programmed block can be divided up into a chain of part blocks that are processed one after the other. Only (!) with the 1st cycle of the 1st block of the chain, timeSC is set to zero and then counted up in seconds. Therefore, the variable enables time measurements throughout the entire block chain.					
-	0	0		Double	r
Multi-line: yes	1		1		

toolCounter					
Counter of the changes of the tool data assigned to a channel. The counter is incremented each time a tool data is changed. All changes of tool data made by BTSS, part programs, INI files and by the Tool Management software are considered. Tool data are tool compensations, grinding-specific tool parameters, OEM tool parameters and Tool Managment data including magazine data. There is one exception: the present tool-in-use-time, since it is changed in each IPO cycle.					
-				UWord	r
Multi-line: yes	1		1		

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

toolCounterIso	keine				
Each change of a tool offset value for ISO2.2 and ISO3.2 modes is counted. This is to enable the HMI to record data changes.					
-	98			Long Integer	r
Multi-line: yes	1		1		

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

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:		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					

transSys	$AC_TRANS_SYS				
Reference system for translation with cartesian manual traversal 0: Axis-specific manual traversal active 1: Cartesian manual traversal in basic coordinate system active 2: Cartesian manual traversal in workpiece coordinate system active 3: Cartesian manual traversal in tool coordinate system active					
-	0	0	3	UWord	r
Multi-line: yes	1		1		

transfActive	DB21-28, DBX33.6				K1, M1
Transformation active 0 = inactive 1 = active					
-				UWord	r
Multi-line: no					

vaEgSyncDiff	$VA_EG_SYNCDIFF[a]				
Electronic gear: Synchronism deviation (actual values). The comparison between this value and $MA_COUPLE_POS_TOL_... determines whether the appropriate "Synchronism" VDI signal is set.					
mm, inch, degree, user defined	0			Double	r
Multi-line: yes	(Axis index of slave axis + 1)		numMachAxes		

vaEgSyncDiffS	$VA_EG_SYNCDIFF_S[a]				
Electronic gear: Synchronous run difference (actual values) with sign. Whether the corresponding VDI signal "synchronous run" is set depends upon the comparison of this value with $MA_COUPLE_POS_TOL_....					
mm, inch, degree, user defined	0			Double	r
Multi-line: yes	(Axis index of the following axis)		numMachAxes		

vaSyncDiff	$VA_SYNCDIFF[]				
Actual value synchronism difference for all types of coupling					
mm, inch, degree, user defined	0			Double	r
Multi-line: yes	Axis index of the following axis		numMachAxes		

vaSyncDiffStat	$VA_SYNCDIFF_STAT[]				
Status of the actual value synchronism difference -4: Reserved -3: No valid value in $VA_SYNCDIFF, tangential control -2: No valid value in $VA_SYNCDIFF, master value coupling and simulated master value -1: No valid value in $VA_SYNCDIFF 0: No valid value in $VA_SYNCDIFF, coupling not active 1: Valid value in $VA_SYNCDIFF					
-	0	-4	1	Long Integer	r
Multi-line: yes	Axis index of the following axis		numMachAxes		

vcToolO	$VC_TOOLO				
Supplies the actual value of the current tool orientation. The orientation vector is normalized, i.e. it has the absolute value 1.					
-	0	-1	1	Double	r
Multi-line: yes	1: X component		3		

vcToolODiff	$VC_TOOLO_DIFF				
Supplies the angle between the setpoint and actual vectors of the tool orientation					
-	0	0	180	Double	r
Multi-line: yes	1		1		

vcToolOStat	$VC_TOOLO_STAT				
Supplies the status of the computation of the actual orientation					
-	0	-1	0	Long Integer	r
Multi-line: no					

vcToolR	$VC_TOOLR				
Actual value of tool rotation The rotation vector is normalized, i.e. it has the absolute value 1.					
-	0	-1	1	Double	r
Multi-line: yes	1: X component		3		

vcToolRDiff	$VC_TOOLR_DIFF				
Angle between setpoint and actual tool rotation vectors					
Degree	0	0	180	Double	r
Multi-line: yes	1		1		

vcToolRStat	$VC_TOOLR_STAT				
Status of the computation of the actual rotation					
-	0	-1	0	Long Integer	r
Multi-line: yes	1		1		

workPnameSubstitution					
Path name defined by HMI on selection of a progam or workpiece in /_N_EXT_DIR, which is displayed in the path variable of the OPI instead of the internal NCK path. The string must be terminated with "\0". The last character before the terminating zero is interpreted by the NCK as a separator between the path part and the file name (typically "\" or "/").					

Permitted character set: ASCII without control characters. Handling: workPnameSubstitution is written by the HMI before the program selection. NCK initially stores this information in a buffer. The information in the buffer becomes active on the program selection (PI _N_SELECT, _N_EXTERN, _N_EXTMOD) provided that the selection refers to the /_N_EXT_DIR. Otherwise workPnameSubstitution is deleted.					
-	0	0		String[128]	r
Multi-line: no					

1.4.3 Area C, Module 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 = inactive 1 = active					
-				UWord	r
Multi-line: no					

ipoBlocksOnly					
Display traversing blocks 0 = normal block transfer 1 = exclusively traversing blocks					
-				UWord	r
Multi-line: no					

optAssStopActive					
Associated M01 selected 0: Not selected 1: Selected					
-	0			UWord	r
Multi-line: yes	1		1		

optStopActive					
M01 selected 0 = not selected 1 = selected					
-				UWord	r
Multi-line: no					

progTestActive	DB21-28, DBX1.7				K1
Program test 0 = inactive 1 = active					
-				UWord	r
Multi-line: no					

rapFeedRateOvrActive					
ROV rapid traverse override 0 = inactive 1 = active					
-				UWord	r
Multi-line: no					

singleBlockActive					
Single block, SBL 0 = no single block 1 = SBL 1 2 = SBL 2					
-				UWord	r
Multi-line: no					

singleBlockType					
Single block mode 1 = interpolation single block 2 = decoder single block					
-				UWord	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 = inactive 1 = active					
-				UWord	r
Multi-line: no					

1.4.4 Area C, Module 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		

actPartProgram					
Content of the current part program starting with the previous block. Blocks may be cut off at the end of the string. The line index determines the section within the program. An efficient current block display can be achieved with the aid of a cyclical variable service. If multiple lines are required then it must be ensured that the client reads line 1 first so that an NCK internal buffer is filled which ensures that the following lines are returned accordingly.					
-				String[200]	r
Multi-line: yes	If the index=1, the first data block is returned, for index=n the nth data block.		3		

block					
To display the currently active part programm, NCK supplies 3 ascii-blocks of the part programm in one single variable job (last, current and next block). That means the variable 'block' consists of a maximum of 3 lines: Line index 1: string of the last block Line index 2: string of the current block Line index 3: string of the next block To gain consistent information, all 3 array elements must be processed in one variable request. This is why the maximum string length of each array element is limited to 66 characters.					
-				String[66]	r
Multi-line: yes	Block index, 1 = last, 2 = current, 3 = next block		3		

blockNoStr					
Block number					
-				String[12]	r
Multi-line: no					

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 the geometry 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 the geometry axis		3		

circleRadius					
Radius of the circle (only effective for G02/G03)					
-				Double	r
Multi-line: no					

circleRadiusS					
Corresponds to circleRadius for block search with calculation. Note: This variable is not available for the variable service, but only for logging in connection with block search events!					
-				Double	r
Multi-line: yes	1				

circleTurn					
Progr. number of additional circular passes with helical interpolation in curr. program					
-	0	0		Long Integer	r
Multi-line: yes	1		1		

circleTurnS					
Programmed number of additional circular passes with helical interpolation in the current program for search with calculation. Note: This variable is not available for the Variable Service, but only for logging of block search events					
-	0	0		Long Integer	r
Multi-line: yes	1		1		

cmdToolEdgeCenterCircleCenterEns					
Arc center in relation to WOS frame, i.e. with tool length but without tool radius					
-	0			Double	r
Multi-line: yes	No. of the geometry axis		3		

cmdToolEdgeCenterCircleCenterEnsS					
Corresponds to circleCenterWos for block search with calculation in relation to the WOS frame, i.e. with tool length but without tool radius Note: This variable is not available for the variable service, but only for logging in connection with block search events!					
-	0			Double	r
Multi-line: yes	No. of the geometry axis		3		

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		

seekOffset					
Line number of the current NC block in the program workPandProgName					
-				Long Integer	r
Multi-line: no			1		

seekw					
First line enabled for modification in part program					
-	0	0		Long Integer	r
Multi-line: yes	1		1		

selectedWorkPProg					
Currently selected program, i.e. the program that has been selected with "Select". The variable also displays the program in the JOG and MDI modes. During the simulation, the simulation search temporarily deselects the selected program and selects the program to be simulated. This is hidden by selectedWorkPProg, i.e. during the simulation search, selectedWorkPProg remains unchanged.					
-				String[160]	r
Multi-line: yes	1		1		

singleBlock					
In most cases the variable 'block' is used to read the currently active blocks of the part program. Because this variable is limited to 66 characters per string, it might be necessary (for long blocks) to read longer strings. The variable 'singleBlock' can read complete blocks (up to strings with 198 charecters) . 3 lines can be addressed: Line index 1: last block Line index 2: current block Line index 3: next block It is not guaranteed for rapid block changes, that the information of 3 successive blocks is always consistent, because each block is read with a single variable request. This method is only safe, if the part program has stopped.					
-				String[198]	r
Multi-line: yes	Block index, 1 = last, 2 = current, 3 = next block		3		

stepEditorFormName					
Current module name for step editor is stored					
-				String[128]	r
Multi-line: yes	1		1		

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

workPNameLong					
Name of the active workpiece					
-				String[128]	r
Multi-line: no					

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

1.4.5 Area C, Module 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 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		Long Integer	r
Multi-line: yes	Index of program level		12		

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		

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

1.4.6 Area C, Module 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		

haltBlock					
The following applies to the SPARPI: The interrupt pointer does not mark the block where the program was cancelled but a previous block (hold block), which enables a better resumption. The hold block is explicitly set with the part program commands IPTRLOCK and IPTRUNLOCK, or implicitly manipulated via $MC_AUTO_IPTR_LOCK. The following applies to the SPARPF: If SPARPI is copied completely, the value of the hold block which was set by the NCK is retained. This enables the NCK to recognize the situation, and it responds with the suppressed message alarm 16950. NOTE: For SPARPI and SPARPF this value is ONLY defined for program level 0.					
-	0	0	1	UWord	r
Multi-line: yes	Program levels (only defined for level 0)		1		

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

workPNameL					
Workpiece name = path name in the NCK file structure Note: This variable is ignored when lines are accessed!					
-				String[160]	r
Multi-line: yes	Index of program level		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, Module 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		

haltBlock					
The following applies to the SPARPI: The interrupt pointer does not mark the block where the program was cancelled but a previous block (hold block), which enables a better resumption. The hold block is explicitly set with the part program commands IPTRLOCK and IPTRUNLOCK, or implicitly manipulated via $MC_AUTO_IPTR_LOCK. The following applies to the SPARPF: If SPARPI is copied completely, the value of the hold block which was set by the NCK is retained. This enables the NCK to recognize the situation, and it responds with the suppressed message alarm 16950. NOTE: For SPARPI and SPARPF this value is ONLY defined for program level 0.					
-	0	0	1	UWord	r
Multi-line: yes	Program levels (only defined for level 0)		1		

invocCount					
Actual value of the subroutine call counter. Is always 1 for the main program.					
-				UWord	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	Index of program level		112		

progName					
Program name. The main program that is used in the first main program level must be selected for the block search, otherwise the search request will be acknowledged negatively.					
-				String[32]	wr
Multi-line: yes	Index of 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		

workPNameL					
Workpiece name = path name in the NCK file structure. If no path name is specified, the default search strategy for subroutine calls is used. Note: This variable is ignored when lines are accessed!					
-				String[160]	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, Module 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 respecitive index.
For each synchronous action there is an address variable and a variable in which the value of the address is entered.
5 M functions
3 S functions
3 H functions
1 T function
1 D function
6 F functions
1 E function
can be programmed in each part program block, but no more than 10 synchronous actions must be programmed in a single block.

Dadr					
D-number. There is only one active D-number per channel.					
-				Long Integer	r
Multi-line: no			1		

Dval					
Value of the current D-number					
-				Long Integer	r
Multi-line: no			1		

Eadr					S5
Number of active E-function					
-				UWord	r
Multi-line: no			1		

Eval					S5
Value of the E-function					
mm/min, inch/min, user defined				Double	r
Multi-line: no			1		

Hadr					S5
Number of active auxiliary functions (H-functions). Up to three H-functions can be active simultaneously.					
-		0	99	UWord	r
Multi-line: yes	Serial number		3		

Hval					S5
Value of the H-function					
-		-99999,9999	99999,9999	Double	r
Multi-line: yes	Serial number		3		

Madr					S5
Number of the active M-function. Up to 5 M-functions can be active simultaneously.					
-		0	99	UWord	r
Multi-line: yes	Serial number		5		

Mval					S5
Value of the M-function					
-		0	99999999	Long Integer	r
Multi-line: yes	Serial number		5		

Sadr					S5
Number of active S-functions. Up to three S-functions can be active simultaneously.					
-		0	6	UWord	r
Multi-line: yes	Serial number		3		

Sval					S5
Value of the S-function. Specifies the spindle speed.					
rev/min , m/min		0	999999,999	Double	r
Multi-line: yes	Serial number		3		

TPreSelAdr					
Number of the preselected T-function					
-				UWord	r
Multi-line: no			1		

TPreSelVal					
Value of the preselected T-function					
-				Long Integer	r
Multi-line: no			1		

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

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

1.4.9 Area C, Module 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 + numSynAct		

blockNoStrProg					
Number of the block where the synchronous action has been programmed					
-				String[12]	r
Multi-line: yes	(Protection level) * 1000 + no. of the synchronous action		7 * 1000 + numSynAct		

id					
ID of the synchronous action; value 0 means that there is no ID defined (blockwise)					
-				UWord	r
Multi-line: yes	(Protection level) * 1000 + no. of the synchronous action		7 * 1000 + numSynAct		

latchRow					
Consistent backup of a line in the module. When writing with a line number, the data in this line is consistently backed up. When written with the value 0, this back up is deleted.					
-				UWord	r
Multi-line: yes	See module header				

numElem					
Number of occupied SYNACT elements					
-				UWord	r
Multi-line: yes	See module header				

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

numVars					
Number of SYNACT variables					
-				UWord	r
Multi-line: yes	See module header				

progLineOffset					
SYNACT offset within the progPathName file					
-				Long Integer	r
Multi-line: yes	See module header				

progPathName					
Synchronized action file					
-				String[160]	r
Multi-line: yes	See module header				

selectMask					
Masks some of the entries in the relevant SYNACT list. Only those SYNACTS are entered in the lists for which (selectMask-lowByte AND synActInfo-lowByte) AND (selectMask-higByte AND synActInfo-highByte) apply. The default value 0xFFFF generates completely unfiltered lists. Bit0: Area: User Bit1: Area: Manufacturer Bit2: Area: System Bit3: Area: Safety Bit8: Type: Static Bit9: Type: Modal					
-				UWord	r
Multi-line: yes	See module header				

synActCounter					
Modification counter for the SYNACT entries in the relevant list.					
-				UWord	r
Multi-line: yes	See module header				

synActInfo					
Information on classification of the SYNACT Bit0: Area: User Bit1: Area: Manufacturer Bit2: Area: System Bit3: Area: Safety Bit8: Type: Static Bit9: Type: Modal					
-				UWord	r
Multi-line: yes	See module header				

synactBlock					
Current synchronized action block (short)					
-				String[66]	r
Multi-line: yes	See module header				

synactBlockL					
Current synchronized action block (long)					
-				String[198]	r
Multi-line: yes	See module header				

typStatus					
Type and 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 + numSynAct		

varName					
Name of SYNACT variable					
-				String[32]	r
Multi-line: yes	See module header				

varTyp					
Data type of SYNACT variable. Coding according to ACX. 0: BOOL (2 bytes) 3: LONG 10: DOUBLE 12: CHAR[32]					
-				UWord	r
Multi-line: yes	See module header				

varValue					
Value of SYNACT variable					
-				String[32]	r
Multi-line: yes	See module header				

1.4.10 Area C, Module 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, Module 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, Module 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 work 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 * (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 * (numGeoAxes + numAuxAxes) + axis number		$MC_MM_NUM_BASE_FRAMES * (numGeoAxes + numAuxAxes)		

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

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

1.4.13 Area C, Module FS: Channel-specific system frames

OEM-MMC: Linkitem

/Channel/SystemFrame/...

Those that exist are set by the bits in $MC_MM_SYSTEM_FRAME_MASK.
Consequently, there may be gaps between the active system frames.

The maximum frame index is:
3 up to but excluding SW 6.3.
5 from and including SW 6.3.
11 as from and including SW 7.4.
12 as from and including SW 7.5.

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 not active 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, Module AUXFU: Auxiliary functions

OEM-MMC: Linkitem

/Channel/AuxiliaryFunctions/...

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

1.5 State data of axis

1.5.1 Area C, Module 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 (from this module)					
mm, inch, degree, user defined				Double	r
Multi-line: yes	Axis index		numMachAxes		

cmdToolBasePos					
Tool base position, desired value . Physical unit is defined in variable extUnit (in this module).					
mm, inch, degree, user defined				Double	r
Multi-line: yes	Axis index		numMachAxes		

extUnit					
Current physical unit of the axis position 0 = mm 1 = inch 2 = degree 3 = indexing position 4 = userdef					
-				UWord	r
Multi-line: yes	Axis index		numMachAxes		

name					
Axis name					
-				String[32]	r
Multi-line: yes	Axis index		numMachAxes		

status					
Axis status 0 = travel command in plus direction 1 = travel command in minus direction 2 = exact position coarse reached 3 = exact position fine reached					
-				UWord	r
Multi-line: yes	Axis index		numMachAxes		

toolBaseDistToGo					
Tool base distance-to-go. Physical unit is defined in the variable extUnit (in this module).					
mm, inch, degree, user defined				Double	r
Multi-line: yes	Axis index		numMachAxes		

toolBaseREPOS					
Tool base REPOS. Physical unit is defined in the variable extUnit (in this module).					
mm, inch, degree, user defined				Double	r
Multi-line: yes	Axis index		numMachAxes		

varIncrVal					
Settable value for INC_VAR. The physical value depends on whether the axis is linear or rotary. Linear axis: unit is 1 mm Rotary axis: unit is 1/1000 degrees					
mm, inch, degree, user defined				Double	r
Multi-line: yes	Axis index		numMachAxes		

1.5.2 Area C, Module 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.

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 work offset for an axis. The value of the offset is stored in the variable 'PRESETVal'. The variable can be overwritten by the part program and by the MMC.					
mm, inch, user defined				Double	wr
Multi-line: yes	Axis index		numMachAxes		

aaAcc	$AA_ACC[Achse]				
Current axial acceleration value					
m/s2, 1000 inch/ s2, rev/s2, user defined	0			Double	r
Multi-line: yes	Axis index		numMachAxes		

aaAccPercent	$AA_ACC_PERCENT[Achse]				
Current acceleration value for single-axis interpolation in percent					
-	0	0		UWord	r
Multi-line: yes	Axis index		numMachAxes		

aaActIndexAxPosNo	$AA_ACT_INDEX_AX_POS_NO[<Achse>]				
Current indexing position; the display depends on $MN_INDEX_AX_NO_MODE and the division (via table or equidistant)					
-	0			Long Integer	r
Multi-line: yes	Axis 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		

aaAxChangeStat	$AA_AXCHANGE_STAT[Achse]				
Axis status with respect to axis replacement 0: Axis can be replaced 1: Axis is linked to the channel, but can become the PLC, command or reciprocating axis 2: Axis cannot be replaced					
-	0	0	2	UWord	r
Multi-line: yes	Axis index		numMachAxes		

aaAxChangeTyp	$AA_AXCHANGE_TYP[Achse]				
Axis type with respect to axis replacement 0: Axis assigned to the NC program 1: Axis assigned to the PLC or active as command axis or reciprocating axis 2: Other channel has interpolation right 3: Neutral axis 4: Neutral axis controlled from the PLC 5: Other channel has interpolation right; axis is requested for the NC program 6: Other channel has interpolation right; axis is requested as neutral axis 7: Axis is PLC axis or is active as command axis or reciprocating axis; axis is requested for the NC program 8: Axis is PLC axis or is active as command axis or reciprocating axis; axis is requested as neutral axis					
-	0	0	8	UWord	r
Multi-line: yes	Axis 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		

aaBrakeCondB	$AA_BRAKE_CONDB[axis]				
Shows the pending braking requests (conditions) for the interpolator stop of the axis / spindle. A braking request consists of a collision direction relating to a coordinate axis in the BCS and a braking priority relating to the machining step. If the axis / spindle receives a current braking request on account of these requirement(s), bit 0 is set in $AA_BRAKE_STATE[X] (in the next IPO cycle).\ The braking priorities in the positive direction are indicated in bits 0 to 2: 0: No pending braking request 1: Priority 1 covers all positioning actions (G0, POS, SPOS) 2: Priority 2 covers DYNNORM and all priority 1 motions 3: Priority 3 covers DYNPOS and all priority 1 to 2 motions 4: Priority 4 covers DYNROUGH and all priority 1 to 3 motions 5: Priority 5 covers DYNSEMIFIN and all priority 1 to 4 motions 6: Priority 6 covers all motions (including DYNFINISH) 7: Priority 7 covers all motions. The request was triggered by the VDI interface signal DB31,..DBX4.3 "Feed stop / Spindle stop". Braking always takes place, irrespective of the direction of motion. The braking priorities in the negative direction are indicated in bits 16 to 18: 0 to 7: Same meaning as bits 0 to 2 All other bits are not set. If the value of the variable is shown in hexadecimal format, the fifth character from the right shows the braking word in the negative direction and the first character from the right shows it in the positive direction.					
-	0	0	0x70007	UDoubleword	r
Multi-line: yes	Axis index		numMachAxes		

aaBrakeCondM	$AA_BRAKE_CONDM[axis]				
Shows the pending braking requests (conditions) for the interpolator stop of the axis / spindle. A braking request consists of a collision direction relating to a coordinate axis in the MCS and a braking priority relating to the machining step. The braking priorities in the positive direction are indicated in bits 0 to 2: 0: No pending braking request 1: Priority 1 covers all positioning actions (G0, POS, SPOS) 2: Priority 2 covers DYNNORM and all priority 1 motions					

3: Priority 3 covers DYNPOS and all priority 1 to 2 motions 4: Priority 4 covers DYNROUGH and all priority 1 to 3 motions 5: Priority 5 covers DYNSEMIFIN and all priority 1 to 4 motions 6: Priority 6 covers all motions (including DYNFINISH) 7: Priority 7 covers all motions. The request was triggered by the VDI interface signal DB31,..DBX4.3 "Feed stop / Spindle stop". Braking always takes place, irrespective of the direction of motion. The braking priorities in the negative direction are indicated in bits 16 to 18: 0 to 7: Same meaning as bits 0 to 2 All other bits are not set. If the value of the variable is shown in hexadecimal format, the fifth character from the right shows the braking word in the negative direction and the first character from the right shows it in the positive direction.

-	0	0	0x70007	UDoubleword	r
Multi-line: yes	Axis index		numMachAxes		

aaBrakeState	$AA_BRAKE_STATE[axis]				
Returns for the axis / spindle whether braking has been initiated on account of the request of $AA_BRAKE_CONDB[X] or a VDI interface signal DB31,..DBX4.3 "Feed stop / Spindle stop".					
-	0	0	1	UDoubleword	r
Multi-line: yes	Axis index		numMachAxes		

aaChanNo	$AA_CHANNO[Achse]				
The variable supplies the number of the channel in which the axis is currently being interpolated. With value 0, the axis could not be assigned to any channel.					
-	0	0		UWord	r
Multi-line: yes	Axis 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		

aaCoupCorr	$AA_COUP_CORR[Achse]				
This variable is used to execute the function "Correct synchronism error". It returns the compensation value for the position offset for the generic coupling with CPFRS = "MCS". The actual values of this spindle are compared with the setpoints for the duration (MD 30455 MISC_FUNCTION_MASK, bit 7) of the activation of the VDI interface signal DB31..,DBX31.6 'Correct synchronism' for the following spindle with coupling active. The difference is the compensation value, which can be read with this variable.					
-	0			Double	r
Multi-line: yes	Axis index		numMachAxes		

aaCoupCorrDist	$AA_COUP_CORR_DIST[Achse]				
Generic coupling: path still to be retracted for aaCoupCorr					
-	0			Double	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		

aaDepAxO	$AA_DEPAXO[Achse]				
Dependency on other axes. Returns an axis code for the defined axis AX containing all the machine axes that have a mechanical dependency on the defined axis.					
-	0	0		Long Integer	r
Multi-line: yes	Axis 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		

aaDtbreb	$AA_DTBREB[axis]				
The total estimated distance until braking end, BCS					
-	0	0		Double	r
Multi-line: yes	Axis index		numMachAxes		

aaDtbrebCmd	$AA_DTBREB_CMD[axis]				
Commanded portion of the braking distance, BCS					
-	0	0		Double	r
Multi-line: yes	Axis index		numMachAxes		

aaDtbrebCorr	$AA_DTBREB_CORR[axis]				
Correction portion of the braking distance, BCS					
-	0	0		Double	r
Multi-line: yes	Axis index		numMachAxes		

aaDtbrebDep	$AA_DTBREB_DEP[axis]				
Dependent portion of the braking distance, BCS					
-	0	0		Double	r
Multi-line: yes	Axis index		numMachAxes		

aaDtbrem	$AA_DTBREM[axis]				
The total estimated distance until braking end, MCS					
-	0	0		Double	r
Multi-line: yes	Axis index		numMachAxes		

aaDtbremCmd	$AA_DTBREM_CMD[axis]				
Commanded portion of the braking distance, MCS					
-	0	0		Double	r
Multi-line: yes	Axis index		numMachAxes		

aaDtbremCorr	$AA_DTBREM_CORR[axis]				
Correction portion of the braking distance, MCS					
-	0	0		Double	r
Multi-line: yes	Axis index		numMachAxes		

aaDtbremDep	$AA_DTBREM_DEP[axis]				
Dependent portion of the braking distance, MCS					
-	0	0		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		

aaEnc1Active	$AA_ENC1_ACTIVE[Achse]				
First measuring system is active 0: Measuring system is not active 1: Measuring system is active					
-	0	0	1	UWord	r
Multi-line: yes	Axis index		numMachAxes		

aaEnc1Ampl	$AA_ENC_AMPL[1,ax]				
Enc1: Percentage gain factor of the amplitude control					
-	0	0		Double	r
Multi-line: yes	Axis index		numMachAxes		

aaEnc2Active	$AA_ENC2_ACTIVE[Achse]				
Second measuring system is active 0: Measuring system is not active 1: Measuring system is active					
-	0	0	1	UWord	r
Multi-line: yes	Axis index		numMachAxes		

aaEnc2Ampl	$AA_ENC_AMPL[2,ax]				
Enc2: Percentage gain factor of the amplitude control					
-	0	0		Double	r
Multi-line: yes	Axis index		numMachAxes		

aaEncActive	$AA_ENC_ACTIVE[Achse]				
Measuring system is active 0: Measuring system is not active 1: Measuring system is active					
-	0	0	1	UWord	r
Multi-line: yes	Axis 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		

aaFixPointSelected	$AA_FIX_POINT_SELECTED[<Achse>]				
Selected fixed point: Number of the fixed point that is to be approached					
-	0			UDoubleword	r
Multi-line: yes	Axis 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		

aaInSync	$AA_IN_SYNC[Achse]				
Synchronization status of the following axis with master value coupling and ELG 0: Synchronization is not running 1: Synchronization is running, i.e. following axis is being synchronized					
-	0	0	1	UWord	r
Multi-line: yes	Axis index		numMachAxes		

aaInposStat	$AA_INPOS_STAT[Achse]				
Status for the programmed position 0: No status available (axis/spindle is outside of the programmed position) 1: Travel motion pending 2: Position setpoint reached 3: Position reached with 'exact stop coarse' 4: Position reached with 'exact stop fine'					
-	0	0	4	UWord	r
Multi-line: yes	Axis 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		

aaJogPosAct	$AA_JOG_POS_ACT[Achse]				
Position reached for JOG to position					
-	0	0	1	UWord	r
Multi-line: yes	Axis index		maxnumGlobMachAxes		

aaJogPosSelected	$AA_JOG_POS_SELECTED[Achse]				
JOG to position is active					
-	0	0	1	UWord	r
Multi-line: yes	Axis index		maxnumGlobMachAxes		

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		Double	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		

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		

aaMeaAct	$AA_MEAACT[Achse]				
Axial measuring active 0: Measuring system is not active 1: Measuring system is active					
-	0	0	1	UWord	r
Multi-line: yes	Axis 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 value not reached 1: Limit value reached in positive axis direction 11: Limit value reached in negative axis direction					
-				UWord	r
Multi-line: yes	Axis 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		

aaOnFixPoint	$AA_FIX_ON_POINT[<Achse>]				
Number of the fixed point at which the axis stands					
-	0			UDoubleword	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		

aaProgIndexAxPosNo	$AA_PROG_INDEX_AX_POS_NO[Achse]				
Programmed indexing position 0: No indexing axis, therefore no indexing position available >0: Number of the programmed indexing position					
-	0	0		UWord	r
Multi-line: yes	Axis index		numMachAxes		

aaRef	$AA_REF[Achse]				
Axis is referenced 0: Axis is not referenced 1: Axis is referenced					
-	0	0	1	UWord	r
Multi-line: yes	Axis index		numMachAxes		

aaReposDelay	$AA_REPOS_DELAY[Achse]				
REPOS suppression active 0: REPOS suppression is currently not active for this axis 1: REPOS suppression is currently active for this axis					
-	0	0	1	UWord	r
Multi-line: yes	Axis index		numMachAxes		

aaScPar	$AA_SCPAR[Achse]				
Current setpoint parameter set					
-	0	0		Long Integer	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 status of the following axis with master value coupling 0: No synchronism 1: Synchronism coarse 2: Synchronism fine 3: Synchronism coarse and fine					
-				UWord	r
Multi-line: yes	Axis index		numMachAxes		

aaSyncDiff	$AA_SYNCDIFF[Achse]				
Setpoint synchronism difference					
mm, inch, degree, user defined	0			Double	r
Multi-line: yes	Axis index		numMachAxes		

aaSyncDiffStat	$AA_SYNCDIFF_STAT[Achse]				
Status of the setpoint synchronism difference -4: No valid value in aaSyncDiff, coupled motion from part program -3: Reserved -2: Reserved -1: No valid value in aaSyncDiff 0: No valid value in aaSyncDiff, coupling not active 1: Valid value in aaSyncDiff					
-	0	-4	1	Long Integer	r
Multi-line: yes	Axis 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		

aaType	$AA_TYPE[Achse]				
Cross-channel axis type 0: Axis type cannot be determined 1: NC program axis 2: Neutral axis 3: PLC axis 4: Reciprocating axis 5: Neutral axis that is currently executing a JOG or homing motion 6: Following axis coupled to the master value 7: Coupled motion of the following axis, activated in a synchronized action 8: Command axis 9: Compile Cycle axis 10: Coupled slave axis (master-slave function.) 11: Program axis that is currently executing a JOG or homing motion					
-	0	0	11	UWord	r
Multi-line: yes	Axis 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		

acRpValid	$AC_RPVALID[Achse]				
Reapproach position valid 0: Reapproach position not valid 1: Reapproach position valid					
-	0	0	1	UWord	r
Multi-line: yes	Axis 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, if the axis is a positioning axis.					
-				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		

activeSvOverride					
Currently active SG override factor in the NCK					
-	-1	-1	100	Long Integer	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 = inactive 1 = active					
-				UWord	r
Multi-line: yes	Axis index		numMachAxes		

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

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 for a positioning 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					
Rotary drive: Load-side path corresponding to one revolution of the drive. Is returned in the unit of the internal computational resolution INT_INCR_PER_MM (for linear axes) or INT_INCR_PER_DEG (for rotary axes / spindles) taking into account gear ratios etc. In the case of linear axes, the pitch of the ball screw is also included in the calculation. In the case of linear motors, a fixed value of "1mm" is used for the ball screw pitch instead of the non-existent ball screw.					
mm, inch, degree, user defined				Double	r
Multi-line: yes	Axis index		numMachAxes		

drfVal					
DRF value					
-	0			Double	r
Multi-line: yes	Axis index		numMachAxes		

drive2ndTorqueLimit					
2nd torque limit. With linear motors: 2nd force limit 0 = inactive 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 = inactive 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 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 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 referred to MD					
-				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		

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 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		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 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: 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 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		

safeActiveCamTrack					
Status Safe cam track (active/inactive) Bit 0 = 1/0: Safe cam track 1 active/inactive Bit 1 = 1/0: Safe cam track 2 active/inactive Bit 2 = 1/0: Safe cam track 3 active/inactive Bit 3 = 1/0: Safe cam track 4 active/inactive					
-	0	0	0xF	UWord	r
Multi-line: no			numMachAxes		

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 = inactive 1 = active					
-				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		

safeOutputSigCam					
Results of the NCK safe cam evaluation					
-	0	0	3FFFFFFF	Long Integer	r
Multi-line: no			numMachAxes		

safeOutputSigCamDrive					
Results of the drive safe cam evaluation					
-	0	0	3FFFFFFF	Long Integer	r
Multi-line: no			numMachAxes		

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

safeOutputSigDrive2					
Safe output signals of the drive part 2					
-		0	0xffff	UWord	r
Multi-line: no			numMachAxes		

safePosCtrlActive					
Axis monitors the absolute position 0 = Axis does not monitor the absolute position (no SE/SN) 1 = Axis monitors the absolute position					
-	0	0	1	UWord	r
Multi-line: no			numMachAxes		

safeStopOtherAxis					
Stop on another axis 0: No stop on another axis 1: Stop on another axis					
-	0	0	1	UWord	r
Multi-line: yes	Axis 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		

vaAbsoluteEnc1DeltaInit	$VA_ABSOLUTE_ENC_DELTA_INIT[Achse]				
Enc1: Initial difference					
-	0	0		UDoubleword	r
Multi-line: yes	Axis index		numMachAxes		

vaAbsoluteEnc1ErrCnt	$VA_ABSOLUTE_ENC_ERR_CNT[Achse]				
Enc 1: Error counter for absolute encoder					
-	0	0		Long Integer	r
Multi-line: yes	Axis index		numMachAxes		

vaAbsoluteEnc1State	$VA_ABSOLUTE_ENC_STATE[Achse]				
Enc1: Status of absolute encoder interface Bit0: Interface is active Bit1: Error during parity check Bit2: Error bit alarm Bit3: Error bit CRC error Bit4: Start bit missing with EnDat transfer					
-	0	0		Long Integer	r
Multi-line: yes	Axis index		numMachAxes		

vaAbsoluteEnc1ZeroMonMax	$VA_ABSOLUTE_ENC_ZERO_MON_MAX[Achse]				
Enc1:Maximum of vaEnc1ZeroMonAct with absolute encoder					
-	0	0		UDoubleword	r
Multi-line: yes	Axis index		numMachAxes		

vaAbsoluteEnc2DeltaInit	$VA_ABSOLUTE_ENC_DELTA_INIT[Achse]				
Enc2: Initial difference					
-	0	0		UDoubleword	r
Multi-line: yes	Axis index		numMachAxes		

vaAbsoluteEnc2ErrCnt	$VA_ABSOLUTE_ENC_ERR_CNT[Achse]				
Enc 2: Error counter for absolute encoder					
-	0	0		Long Integer	r
Multi-line: yes	Axis index		numMachAxes		

vaAbsoluteEnc2State	$VA_ABSOLUTE_ENC_STATE[Achse]				
Enc2: Status of absolute encoder interface Bit0: Interface is active Bit1: Error during parity check Bit2: Error bit alarm Bit3: Error bit CRC error Bit4: Start bit missing with EnDat transfer					
-	0	0		Long Integer	r
Multi-line: yes	Axis index		numMachAxes		

vaAbsoluteEnc2ZeroMonMax	$VA_ABSOLUTE_ENC_ZERO_MON_MAX[Achse]				
Enc2:Maximum of vaEnc2ZeroMonAct with absolute encoder					
-	0	0		UDoubleword	r
Multi-line: yes	Axis index		numMachAxes		

vaCecCompVal	$VA_CEC_COMP_VAL[Achse]				
Axial sag compensation value					
mm, inch, degree, user defined	0			Double	r
Multi-line: yes	Axis index		numMachAxes		

vaCurr	$VA_CURR[Achse]				
Drive actual current value					
-	0			Double	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		

vaEnc1CompVal	$VA_ENC1_COMP_VAL[Achse]				
Leadscrew error compensation (LEC) value encoder 1					
mm, inch, degree, user defined	0			Double	r
Multi-line: yes	Axis index		numMachAxes		

vaEnc1ZeroMonAccessCnt	$VA_ENC_ZERO_MON_ACCESS_CNT[Achse]				
Enc1: Update counter					
-	0	0		UDoubleword	r
Multi-line: yes	Axis index		numMachAxes		

vaEnc1ZeroMonAct	$VA_ENC_ZERO_MON_ACT[Achse]				
Enc1: Zero monitoring values					
-	0	0		UDoubleword	r
Multi-line: yes	Axis index		numMachAxes		

vaEnc1ZeroMonErrCnt	$VA_ENC_ZERO_MON_ERR_CNT[Achse]				
Enc 1: Error counter for zero mark monitoring					
-	0	0		Long Integer	r
Multi-line: yes	Axis index		numMachAxes		

vaEnc1ZeroMonInit	$VA_ENC_ZERO_MON_INIT[Achse]				
Enc1:Hardware counter value of the basic zero mark					
-	0	0		UDoubleword	r
Multi-line: yes	Axis index		numMachAxes		

vaEnc2CompVal	$VA_ENC2_COMP_VAL[Achse]				
Leadscrew error compensation (LEC) value encoder 2					
mm, inch, degree, user defined	0			Double	r
Multi-line: yes	Axis index		numMachAxes		

vaEnc2ZeroMonAccessCnt	$VA_ENC_ZERO_MON_ACCESS_CNT[Achse]				
Enc2: Update counter					
-	0	0		UDoubleword	r
Multi-line: yes	Axis index		numMachAxes		

vaEnc2ZeroMonAct	$VA_ENC_ZERO_MON_ACT[Achse]				
Enc2: Zero monitoring values					
-	0	0		UDoubleword	r
Multi-line: yes	Axis index		numMachAxes		

vaEnc2ZeroMonErrCnt	$VA_ENC_ZERO_MON_ERR_CNT[Achse]				
Enc 2: Error counter for zero mark monitoring					
-	0	0		Long Integer	r
Multi-line: yes	Axis index		numMachAxes		

vaEnc2ZeroMonInit	$VA_ENC_ZERO_MON_INIT[Achse]				
Enc2:Hardware counter value of the basic zero mark					
-	0	0		UDoubleword	r
Multi-line: yes	Axis index		numMachAxes		

vaFoc	$VA_FOC[Achse]				
Actual status of "ForceControl" 0: ForceControl not active 1: Modal ForceControl active 2: Non-modal ForceControl active					
-	0	0	2	UWord	r
Multi-line: yes	Axis index		numMachAxes		

vaFxs	$VA_FXS[Achse]				
Actual status of "Travel to fixed stop" 0: Axis not at fixed stop 1: Successful travel to fixed stop 2: Unsuccessful travel to fixed stop 3: Travel to fixed stop selection active 4: Fixed stop has been detected 5: Travel to fixed stop deselection active					
-	0	0	5	UWord	r
Multi-line: yes	Axis 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		

valm1	$VA_IM1[x]				
Actual value in the machine coordinate system (measured encoder 1)					
-	0	0		Double	r
Multi-line: yes	Axis index		numMachAxes		

valm2	$VA_IM2[x]				
Actual value in the machine coordinate system (measured encoder 2)					
-	0	0		Double	r
Multi-line: yes	Axis index		numMachAxes		

vaLagError	$VA_LAG_ERROR[Achse]				
Axis following error					
-	0			Double	r
Multi-line: yes	Axis index		numMachAxes		

vaLoad	$VA_LAG_ERROR[Achse]				
Drive utilization in %					
-	0	-100	100	Double	r
Multi-line: yes	Axis index		numMachAxes		

vaPosctrlMode	$VA_POSCTRL_MODE[Achse]				
Position controller mode 0: Position control 1: Speed control 2: Holding 3: Parking 4: Tracking					
-	0	0	4	UWord	r
Multi-line: yes	Axis index		numMachAxes		

vaPower	$VA_POWER[Achse]				
Active drive power					
-	0			Double	r
Multi-line: yes	Axis index		numMachAxes		

vaPressureA	$VA_PRESSURE_A[Achse]				
Pressure on A end of the cylinder in bar (only for 611D Hydraulic)					
-	0			Double	r
Multi-line: yes	Axis index		numMachAxes		

vaPressureB	$VA_PRESSURE_B[Achse]				
Pressure on B end of the cylinder in bar (only for 611D Hydraulic)					
-	0			Double	r
Multi-line: yes	Axis index		numMachAxes		

vaSce	$VA_SCE[Achse]				
Status of speed controller enable					
-	0	0	1	UWord	r
Multi-line: yes	Axis index		numMachAxes		

vaStopSi	$VA_STOPSI[Achse]				
Stop from Safety Integrated -1: No stop 0: Stop A 1: Stop B 2: Stop C 3: Stop D 4: Stop E 5: Stop F 10: Test stop of NC 11: Test of ext. pulse suppression					
-	0			Long Integer	r
Multi-line: yes	Axis index		numMachAxes		

vaSyncDiff	$VA_SYNCDISFF[Achse]				
Actual value synchronism difference					
mm, inch, degree, user defined	0			Long Integer	r
Multi-line: yes	Axis index		numMachAxes		

vaSyncDiffStat	$VA_SYNCDIFF_STAT[Achse]				
Status of the actual value synchronism difference -4: Reserved -3: No valid value in $VA_SYNCDIFF, tangential control -2: No valid value in $VA_SYNCDIFF, master value coupling and simulated master value -1: No valid value in $VA_SYNCDIFF 0: No valid value in $VA_SYNCDIFF, coupling not active 1: Valid value in $VA_SYNCDIFF					
-	0	-4	1	Long Integer	r
Multi-line: yes	Axis index		numMachAxes		

vaTempCompVal	$VA_TEMP_COMP_VAL[Achse]				
Axial temperature compensation value					
mm, inch, degree, user defined	0			Double	r
Multi-line: yes	Axis index		numMachAxes		

vaTorque	$VA_TORQUE[Achse]				
Drive torque setpoint					
-	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		

vaValveLift	$VA_VALVELIFT[Achse]				
Actual valve lift in mm (only for 611D Hydraulic)					
-	0			Double	r
Multi-line: yes	Axis index		numMachAxes		

vaXfaultSi	$VA_XFAULTSI[Achse]				
Stop F through cross-checking error active Bit 0 set: An actual value error has been discovered in the cross-check between NCK and 611D Bit 1 set: Some error has been discovered in the cross-check between NCK and 611D and the waiting time until Stop B ($MA_SAFE_STOP_SWITCH_TIME_F) is triggered is running or has expired					
-	0			Long Integer	r
Multi-line: yes	Axis index		numMachAxes		

1.5.3 Area C, Module 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					
Tool base position. Physical unit is defined in the variable extUnit (from this module)					
mm, inch, degree, user defined				Double	r
Multi-line: yes	Axis index		numMachAxes		

actToolEdgeCenterPos	$AA_IW[x] x = Axis				
Center point of a cutting edge. Physical unit is defined in the variable extUnit (from this module)					
mm, inch, degree, user defined				Double	r
Multi-line: yes	Axis index		numMachAxes		

cmdProgPos					
Programmed position, desired value. Physical unit is defined in the variable extUnit (in this module)					
mm, inch, degree, user defined				Double	r
Multi-line: yes	Axis index		numMachAxes		

cmdToolBasePos					
Tool base position, desired value . Physical unit is defined in variable extUnit (in this module).					
mm, inch, degree, user defined				Double	r
Multi-line: yes	Axis index		numMachAxes		

cmdToolEdgeCenterPos					
Position of the cutting edge center point. Physical unit is defined in variable extUnit (in this module).					
mm, inch, degree, user defined				Double	r
Multi-line: yes	Axis index		numMachAxes		

extUnit					
Current physical unit of the related geometry axis or auxiliary axis 0 = mm 1 = inch 2 = degree 3 = indexing position 4 = userdef					
-				UWord	r
Multi-line: yes	Axis index		numMachAxes		

name					
Axis name					
-				String[32]	r
Multi-line: yes	Axis index		numMachAxes		

progDistToGo					
Programmed position, distance-to-go. The physical unit is defined in the variable extUnit (in this module).					
mm, inch, degree, user defined				Double	r
Multi-line: yes	Axis index		numMachAxes		

progREPOS					
Programmed position, REPOS. The physical unit is defined in the variable extUnit (in this module).					
mm, inch, degree, user defined				Double	r
Multi-line: yes	Axis index		numMachAxes		

status					
Axis status 0 = travel command in plus direction 1 = travel command in minus direction 2 = exact position coarse reached 3 = exact position fine reached					
-				UWord	r
Multi-line: yes	Axis index		numMachAxes		

subType					
Axis type geometry or auxiliary axis 0 = auxiliary axis 1 = geometry axis 2 = orientation axis					
-				UWord	r
Multi-line: yes	Axis index		numMachAxes		

toolBaseDistToGo					
Tool base distance-to-go. Physical unit is defined in the variable extUnit (in this module)					
mm, inch, degree, user defined				Double	r
Multi-line: yes	Axis index		numMachAxes		

toolBaseREPOS					
Tool base REPOS. Physical unit is defined in the variable extUnit (in this module).					
mm, inch, degree, user defined				Double	r
Multi-line: yes	Axis index		numMachAxes		

toolEdgeCenterDistToGo					
Center point of cutting edge distance-to-go. Physical unit results from the variable extUnit (in this module)					
mm, inch, degree, user defined				Double	r
Multi-line: yes	Axis index		numMachAxes		

toolEdgeCenterREPOS					
Center point of the cutting edge REPOS. Physical unit is defined in the variable extUnit (in this module).					
mm, inch, degree, user defined				Double	r
Multi-line: yes	Axis index		numMachAxes		

varIncrVal					
Setable value for INC_VAR. The physical unit depends on whether the axis is rotary or linear. Linear axes: 1mm rotary axes: 1/1000 degrees					
mm, inch, degree, user defined				Double	wr
Multi-line: yes	Axis index		numMachAxes		

1.5.4 Area C, Module 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.

aaAcsRel	$AA_PCS_REL[Achse]				
The axial variable $AA_ACS_REL[ax] determines the current relative setpoint in the settable zero coordinate system (SZS) for the corresponding axis. The setpoint corresponds to $AA_IEN[ax], which is transformed by the current relative system frame $P_RELFRAME. The axial positions lie in the relative settable zero coordinate system.					
mm, inch, degree, user defined	0			Double	r
Multi-line: yes	Axis index		numMachAxes		

aaDelt	$AA_DELT[x] x = Axis				
Stored axial distance-to-go in the WCS after axial delete-distance-to-go DELDTG(axis) for synchronous actions (Note: for SYNACT only)					
-				Double	r
Multi-line: yes	Axis index		numMachAxes		

aaDiamStat	$AA_DIAM_STAT[]				
Status of the diameter programming as a function of configuration and programming Bit 0=0: Diameter programming inactive Bit 0=1: Diameter programming active Bit 1=0: Channel-specific diameter programming					
-	0	0	15	UWord	r
Multi-line: yes	Axis index		numMachAxes		

aaDtbw	$AA_DTBW[x] x = Aaxis				
Axial distance from the beginning of the block in the WCS for positioning and synchronous axes for synchronous motion (Note: for SYNACT only)					
-				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		

aaDtsb	$AA_DTSB				
Path from the motion starting point in the BCS					
mm, inch, degree, user defined				Double	r
Multi-line: yes	Axis index		numMachAxes		

aaDtsw	$AA_DTSW				
Path from the motion starting point in the WCS					
mm, inch, degree, user defined				Double	r
Multi-line: yes	Axis index		numMachAxes		

aaIb	$AA_IB				
Current BCS setpoint of an axis					
mm, inch, degree, user defined				Double	r
Multi-line: yes	Axis index		numMachAxes		

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

aaIbc	$AA_IBC[Achse]				
The axial variable $AA_IBC[ax] determines the setpoint of a Cartesian axis lying between BCS and MCS. Cartesian means that the axis is a linear axis, and it lies plane parallel to a coordinate axis in a clockwise coordinate system. If a geometry axis is still Cartesian at the output of the nth transformation, then this value is returned. The axis identifier used must represent a geometry axis in the BCS, otherwise the variable returns the value 0.					
mm, inch, degree, user defined	0			Double	r
Multi-line: yes	Axis index		numMachAxes		

aaItr1	$AA_ITR[Achse, 1]				
The axial variable determines the current setpoint of an axis at the output of the 1st chained transformation.					
mm, inch, degree, user defined	0			Double	r
Multi-line: yes	Axis index		numMachAxes		

aaItr2	$AA_ITR[Achse, 2]				
The axial variable determines the current setpoint of an axis at the output of the 2nd chained transformation.					
mm, inch, degree, user defined	0			Double	r
Multi-line: yes	Axis index		numMachAxes		

aaItr3	$AA_ITR[Achse, 3]				
The axial variable determines the current setpoint of an axis at the output of the 3rd chained transformation.					
mm, inch, degree, user defined	0			Double	r
Multi-line: yes	Axis index		numMachAxes		

aaIwCorr	$AA_IW_CORR				
Current WCS setpoint value of an axis including override components					
-				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[Achse]				
Access to measurement result of trigger event 1 in the WCS					
-				Double	wr
Multi-line: yes	Axis index		numMachAxes		

aaMw2	$AA_MW2[Achse]				
Access to measurement result of trigger event 2 in the WCS					
-				Double	wr
Multi-line: yes	Axis index		numMachAxes		

aaMw3	$AA_MW3[Achse]				
Access to measurement result of trigger event 3 in the WCS					
-				Double	wr
Multi-line: yes	Axis index		numMachAxes		

aaMw4	$AA_MW4[Achse]				
Access to measurement result of trigger event 4 in the WCS					
-				Double	wr
Multi-line: yes	Axis index		numMachAxes		

aaPcsRel	$AA_PCS_REL[Achse]				
The axial variable $AA_PCS_REL[ax] determines the current relative setpoint of the corresponding axis in the workpiece coordinate system (WCS). The setpoint corresponds to $AA_IW[ax], which is transformed by the current relative system frame $P_RELFRAME. The axial positions lie in the relative workpiece coordinate system.					
mm, inch, degree, user defined	0			Double	r
Multi-line: yes	Axis index		numMachAxes		

aaSccStat	$AA_SCC_STAT[]				
Status of the G96/G961/G962 assignment as a function of configuration and programming Bit 0=0: Axis is not assigned to G96/G961/G962 Bit 0=1: Axis is assigned to G96/G961/G962					
-	0	0	1	UWord	r
Multi-line: yes	Axis index		numMachAxes		

aaTOff	$AA_TOFF[ ]				
Value of the superimposed motions which have been retracted in the individual tool directions via $AA_TOFF[ ]					
mm, inch, user defined	0			Double	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		

actFeedRateIpo					
Corresponds to actFeedRate taking into account the revolutional feedrate. Associated unit see: axisFeedRateIpoUnit					
-				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		

actToolBasPosBN	$AA_IBN[x] x=Axis				
Actual tool base position in relation to basic zero point (SGA:actToolBasePos without progr. frame and without settable frames)					
mm, inch, degree, user defined				Double	r
Multi-line: yes	Axis index		numMachAxes		

actToolBasPosBNDiam					
Corresponds to actToolBasPosBN with diameter conversion					
-				Double	r
Multi-line: yes	Axis index		numMachAxes		

actToolBasPosEN	$AA_IEN[x] x = Axis				
Base position of the active tool relative to the workpiece zero point (SGA:actToolBasePos without programmed frame)					
-				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		

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 actToolBasePos with diameter conversion					
-				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 = inactive 1 = active					
-				UWord	r
Multi-line: yes	Axis index		numMachAxes		

axisFeedRateIpoUnit					
Corresponds to axisFeedRateUnit with revolutional feedrate 0 = mm/min 1 = mm/rev. 2 = inch/min 3 = inch/rev. 4 = deg./min 5 = deg./rev.					
-				UWord	r
Multi-line: yes	Axis index		numMachAxes		

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

cmdFeedRate					
Desired value of axis-specific feedrate for a positioning axis.					
mm/min, inch/min, user defined				Double	r
Multi-line: yes	Axis index		numMachAxes		

cmdFeedRateIpo					
Corresponds to cmdFeedRate taking into account the revolutional feedrate. Associated unit see: axisFeedRateIpoUnit					
-				Double	r
Multi-line: yes	Axis index		numMachAxes		

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: - cmdToolBasePos - toolBaseDistToGo - toolBaseREPOS - cmdToolEdgeCenterPos - actToolEdgeCenterPos - toolEdgeCenterDistToGo - toolEdgeCenterREPOS - cmdProgPos - actProgPos - progDistToGo

- progREPOS - actToolBasPosEN - cmdToolEdgeCenterPosEnsS - actToolEdgeCenterPosEns - actToolBasPosBN - cmdToolBasPosENS - actProgPosBKS - actToolBasePosDiam - actToolBasePosBasicDiam - actToolBasPosBNDiam 0: Diameter programming inactive 1: Diameter programming active					
-	0	0	1	UWord	r
Multi-line: yes	Axis index		numMachAxes		

displayAxis	$MC_DISPLAY_AXIS Bit0-15				
Identifier indicating whether the axis is displayed by the 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 extUnit (in module SGA).					
mm, inch, degree, user defined				Double	r
Multi-line: yes	Axis index		numMachAxes		

feedRateOvr					
Feedrate override if axis is a positioning axis. Multiplying override component which is active in addition to the override factors programmed, set via handwheel or via PLC.					
%				Double	r
Multi-line: yes	Axis index		numMachAxes		

geoAxisNr					
Number of the geometry axis If the axis is a geometry axis: 1-3 If the axis is not a geometry axis: 0					
-				UWord	r
Multi-line: yes	Axis index		numMachAxes		

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

index					
Absolute axis index referred to MD					
-				UWord	r
Multi-line: yes	Axis index		numMachAxes		

motEnd	$AA_MOTEND				
Current motion end criterion for single-axis interpolation 1 = Motion end with exact stop FINE 2 = Motion end with exact stop COARSE 3 = Motion end with exact stop IPO Stop 4 = Block change in braking ramp of axis motion 5 = Block change in braking ramp of axis motion with tolerance window with reference to setpoint 6 = Block change in braking ramp of axis motion with tolerance window with reference to actual value					
-	1	1	6	UWord	r
Multi-line: yes	Axis index		numMachAxes		

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

subSpec	MD 30500: INDEX_AX_ASSIGN_POS_TAB				T1
Subspecification, identifies whether an axis is an indexing axis 0 = normal axis 1 = indexing axis					
-				UWord	r
Multi-line: yes	Axis index		numMachAxes		

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

valb	$VA_IB[Achse]				
The variable $VA_IB[ax] determines the encoder position of an axis retransformed into BCS. The BCS value contains all the axial override components (DRF, AA_OFF, external work offset, etc.) and compensation values (CEC, etc.). The positions are only computed once per IPO cycle for performance reasons. When a variable is read within one IPO cycle, the value of the variable does not change, although the actual value could have changed. If transformations are active, it must be taken into account that transforming the actual values into the BCS in the IPO cycle can be very time-consuming. In this case, an adequate IPO cycle must be set.					
mm, inch, degree, user defined	0			Double	r
Multi-line: yes	Axis index		numMachAxes		

valbc	$VA_IBC[Achse]				
The variable $VA_IBC[geo axis] determines the encoder position of a Cartesian axis lying between BCS and MCS. Cartesian means that the axis is a linear axis and lies plane parallel to a coordinate axis in a clockwise coordinate system. The axis identifier used can be a geometry, channel or machine axis identifier. This identifier must represent a geometry axis in the BCS, otherwise the variable returns the value 0.0. The positions are only computed once per IPO cycle for performance reasons. When a variable is read within one IPO cycle, the value of the variable does not change, although the actual value could have changed. If transformations are active, it must be taken into account that transforming the actual values into the BCS in the IPO cycle can be very time-consuming. In this case, an adequate IPO cycle must be set.					
mm, inch, degree, user defined	0			Double	r
Multi-line: yes	Axis index		numMachAxes		

valtr1	$VA_ITR[Achse, 1]				
The axial variable determines the current encoder position of an axis at the output of the 1st chained transformation.					
mm, inch, degree, user defined	0			Double	r
Multi-line: yes	Axis index		numMachAxes		

valtr2	$VA_ITR[Achse, 2]				
The axial variable determines the current encoder position of an axis at the output of the 2nd chained transformation.					
mm, inch, degree, user defined	0			Double	r
Multi-line: yes	Axis index		numMachAxes		

valtr3	$VA_ITR[Achse, 3]				
The axial variable determines the current encoder position of an axis at the output of the 3rd chained transformation.					
mm, inch, degree, user defined	0			Double	r
Multi-line: yes	Axis index		numMachAxes		

valw	$VA_IW[Achse]				
The variable $VA_IW[ax] determines the encoder position of an axis retransformed into BCS. The BCS value contains all the axial override components (DRF, AA_OFF, external work offset, etc.) and compensation values (CEC, etc.). The positions are only computed once per IPO cycle for performance reasons. When a variable is read within one IPO cycle, the value of the variable does not change, although the actual value could have changed. If transformations are active, it must be taken into account that transforming the actual values into the BCS in the IPO cycle can be very time-consuming. In this case, an adequate IPO cycle must be set.					
mm, inch, degree, user defined	0			Double	r
Multi-line: yes	Axis index		numMachAxes		

1.5.5 Area C, Module 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		

acSmaxAcc	$AC_SMAXACC[]				
Active acceleration of the spindle This variable returns the active acceleration of the spindle for spindle mode. Bit 7 of $AC_SPIND_STATE (spindle accelerating) is set for the duration of the acceleration to the defined setpoint speed. Bit 8 of $AC_SPIND_STATE (spindle braking) is set for the duration of the braking to the defined setpoint speed. Apart from that, the acceleration-determining machine and setting data can be determined with the system variable $AC_SMAXACC_INFO. If the spindle is in axis mode, then $AC_SMAXACC does not return the current acceleration, instead the machine data (MAX_AX_VELO, MAX_AX_ACCEL, ...) typical for axis mode are active.					
Rev/s2, user defined				Double	r
Multi-line: yes	Spindle index		numSpindles		

acSmaxAccInfo	$AC_SMAXACC_INFO[]				
Identifier for the active spindle acceleration data This system variable provides additional information to $AC_SMAXACC and returns the definitive machine data as identifier/index. The index can be used to determine the active acceleration data on the basis of the following table of existing spindle accelerations. The number range is oriented to the system variable $AC_SMAXVELO_INFO: 0: No acceleration limitation (SERUPRO) 1: Not used 2: Acceleration in speed control mode in the current gear stage without position control, MD 35200 GEAR_STEP_SPEEDCTRL_ACCEL * ACC * ACCFXS 3: Acceleration in the current gear stage with active position control (e.g. SPCON, SPOS) MD 35210 GEAR_STEP_POSCTRL_ACCEL * ACC * ACCFXS 4: Acceleration in the current gear stage with active position control while tapping (G331, G332 - only with corresponding configuration of the second data block) MD 35212 GEAR_STEP_POSCTRL_ACCEL2 5: Not used 6: Not used 7: Not used 8: Acceleration limited by the preparation for thread cutting and with synchronous spindle coupling of the leading spindle on account of limited acceleration of the following spindle. 9: Not used 10: Not used 11: Not used 12: Acceleration limited by tool parameter $TC_TP_MAX_ACCEL 13: Not used					
-				Long Integer	r
Multi-line: yes	Spindle index		numSpindles		

acSmaxVelo	$AC_SMAXVELO[]				
Maximum spindle speed This variable returns the maximum spindle speed for spindle mode. This is formed from the smallest active speed limitation, and cannot be exceeded by speed programming or override > 100%. A speed limitation is indicated by the VDI interface signal DB31..,DBX83.1 'Setpoint speed limited' and by $AC_SPIND_STATE, bit 5 (speed limitation active). The cause of the speed limitation (machine, setting data, G code, VDI interface signal etc.) can also be determined with the system variable $AC_SMAXVELO_INFO. If the spindle is in axis mode, then the speed is not limited by $AC_SMAXVELO but instead the machine data (MAX_AX_VELO, ...) typical for axis mode are active.					
rev/min, user defined				Double	r
Multi-line: yes	Spindle index		numSpindles		

acSmaxVeloInfo	$AC_SMAXVELO_INFO[]				
Identifier (index) for the speed-limiting data (machine/setting data, etc.) The system variable provides additional information about $AC_SMAXVELO, and returns the definitive data (machine, setting data, G code, VDI interface, etc.) as identifier/index. The speed-limiting data can be determined with the index from the following table of existing spindle speed limitations . 0: No limitation (SERUPRO) 1: Speed limited by clamping arrangement (chuck speed), MD 35100 SPIND_VELO_LIMIT; 2: Speed limitation (upper limit, that is maximum speed) in the current gear stage, MD 35130 GEAR_STEP_MAX_VELO_LIMIT * VELOLIMA (maximum speed factor from part program) 3: Speed limitation in the current gear stage with position control active MD 35132 GEAR_STEP_PC_MAX_VELO_LIMIT * VELOLIMA (among other items with SPCON, SPOS) 4: Speed limited to SD 43220 SPIND_MAX_VELO_G26 (part program instruction: G26 S.. or PI service from HMI) 5: Speed limited by set VDI interface signal DB31,...DBX3.6 to MD 35160 SPIND_EXTERN_VELO_LIMIT 6: Speed limited at constant cutting speed by part program instruction: LIMS (G96, G97, G961) SD 43230 SPIND_MAX_VELO_LIMS; 7: Speed limited by Safety Integrated to safe speed (SG) 8: Speed limited by the preparation for thread cutting and synchronous spindle coupling of the leading spindle on account of the speed limitation of the following spindle. 9: Speed limited to the maximum speed of the drive by the drive parameters, e.g. SINAMICS: p1082. 10: Speed limited to MD 36300 ENC_FREQ_LIMIT with functions that require a functioning measuring system, e.g. position control and for the master spindle with the G functions G95, G96, G97, G33..35. The limitation takes into account the encoder speed, the MS arrangement (direct/indirect), MS limiting frequency and the current parameter set. 11: Limitation of the overlaid movement of the following spindle to the dynamic remaining after the coupling. 12: Speed limited by tool parameter (in preparation) 13: Speed limited by tool carrier (in preparation)					
-				Long Integer	r
Multi-line: yes	Spindle index		numSpindles		

acSminVelo	$AC_SMINVELO[]				
Minimum spindle speed This variable returns the minimum spindle speed for speed control mode. This is formed from the highest active speed increase, and cannot be undershot by speed programming or override < 100%. A speed increase is indicated by the VDI interface signal DB31..,DBX83.2 'Setpoint speed increased' and by $AC_SPIND_STATE, bit 6 (speed increase active). The cause of the speed increase (machine, setting data, G code, VDI interface signal etc.) can also be determined with the system variable $AC_SMINVELO_INFO. If the spindle is in axis or positioning mode, then the speed is not increased by $AC_SMINVELO.					
rev/min, user defined				Double	r
Multi-line: yes	Spindle index		numSpindles		

acSminVeloInfo	$AC_SMINVELO_INFO[]				
Identifier (index) for the speed-limiting data (machine/setting data, etc.) The system variable provides additional information about $AC_SMAXVELO, and returns the definitive data (machine, setting data, G code, VDI interface, etc.) as identifier/index. The speed-limiting data can be determined with the index from the following table of existing spindle speed limitations. The system variable provides additional information about $AC_SMINVELO, and returns the speed increasing data (machine, setting data) as identifier/index. The speed-increasing data can be determined with the index from the following table of existing spindle speed increases.					
-				Long Integer	r
Multi-line: yes	Spindle index		numSpindles		

acSpindState	$AC_SPIND_STATE[]				
This variable returns the spindle status in speed control mode only (Bit0 =1). The variable $AA_INPOS_STAT is relevant in positioning and axis modes. $AC_SPIND_STATE returns no values in synchronous mode. Bit 0: "Speed control mode active" (condition for validity of the bits of the variable $AC_SPIND_STATE[) Bit 1: "Spindle stopped" (corresponds to VDI interface signal DB31..,DBX61.4) Bit 2: "Spindle programmed" (e.g. M3, M4 S.., FC18, ..) (corresponds to VDI interface signal DB31..,DBX64.4/5 or 6/7) Bit 3: "Spindle in set range" (corresponds to VDI interface signal DB31..,DBX83.5) Bit 4: Free Bit 5: "Limitation of setpoint speed active" remains as long as the programmed or overrride speed remains higher than the speed in the system variable $AC_SMAXVELO. Corresponds to VDI interface signal DB31..,DBX83.1 'Setpoint speed limited'. Bit 6: "Increasing setpoint speed active" remains as long as the programmed speed remains less than the speed in the system variable $AC_SMINVELO. Corresponds to VDI interface signal DB31..,DBX83.2 'Setpoint speed increasing' Bit 7: "Spindle accelerating" (setpoint side), remains as long as the spindle is accelerating to the defined setpoint speed. Bit 8: "Spindle braking" (setpoint side), remains as long as the spindle is braking to the defined setpoint speed or comes to a standstill. Bit 9: "Position control active"					
-				Long Integer	r
Multi-line: yes	Spindle index		numSpindles		

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

actSpeed	$AA_S[x] x = SpindleNo				
Spindle speed actual value					
rev/min, user defined				Double	r
Multi-line: yes	Spindle index		numSpindles		

channelNo					
Number of channel in which spindle is configured					
-				UWord	r
Multi-line: yes	Spindle index		numSpindles		

cmdAngPos					
Spindle position (SPOS)					
Degree, user defined				Double	r
Multi-line: yes	Spindle index		numSpindles		

cmdConstCutSpeed					
Constant cutting 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		

dummy					
dummy					
-				UWord	r
Multi-line: yes	Spindle index		numSpindles		

gwpsActive	{$GWPS}				
SUG programming active (SUG is the function "constant perimeter speed of grinding wheel) 0 = inactive 1 = active					
-				UWord	r
Multi-line: yes	Spindle index		numSpindles		

index					
Absolute axis index referred to MD					
-				UWord	r
Multi-line: yes	Spindle index		numSpindles		

name					
Spindle name Note: If several logical spindles are referred to one physical spindle with active spindle conversion and access is made via area N of module SSP2, then the name of the first suitable logical spindle is output.					
-				String[32]	r
Multi-line: yes	Spindle index		numSpindles		

namePhys					
Name of assigned physical spindle, identical to "name" variable.					
-				String[32]	r
Multi-line: yes	Spindle index		numSpindles		

opMode					
Spindle mode 0 = spindle mode 1 = oscillation mode (gear step changeover) 2 = positioning mode 3 = synchronous mode 4 = axis mode					
-				UWord	r
Multi-line: yes	Spindle index		numSpindles		

pSMode	$P_SMODE				
Last programmed spindle mode 0: No spindle configured in channel or spindle is active in another channel or in use by the PLC (FC18) or by synchronized actions. 1: Speed control mode 2: Positioning mode 3: Synchronous mode 4: Axis mode					
-		0	4	UWord	r
Multi-line: yes	Spindle index		numSpindles		

pSModeS	$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, Module 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		

acSmaxAcc	$AC_SMAXACC[]				
Active acceleration of the spindle This variable returns the active acceleration of the spindle for spindle mode. Bit 7 of $AC_SPIND_STATE (spindle accelerating) is set for the duration of the acceleration to the defined setpoint speed. Bit 8 of $AC_SPIND_STATE (spindle braking) is set for the duration of the braking to the defined setpoint speed. Apart from that, the acceleration-determining machine and setting data can be determined with the system variable $AC_SMAXACC_INFO. If the spindle is in axis mode, then $AC_SMAXACC does not return the current acceleration, instead the machine data (MAX_AX_VELO, MAX_AX_ACCEL, ...) typical for axis mode are active.					
Rev/s2, user defined				Double	r
Multi-line: yes	Logical spindle index		numSpindlesLog		

acSmaxAccInfo	$AC_SMAXACC_INFO[]				
Identifier for the active spindle acceleration data This system variable provides additional information to $AC_SMAXACC and returns the definitive machine data as identifier/index. The index can be used to determine the active acceleration data on the basis of the following table of existing spindle accelerations. .The number range is oriented to the system variable $AC_SMAXVELO_IDX: 0: No acceleration limitation (SERUPRO) 1: Not used 2: Acceleration in speed control mode in the current gear stage without position control MD 35200 GEAR_STEP_SPEEDCTRL_ACCEL * ACC * ACCFXS 3: Acceleration in the current gear stage with active position control (e.g. SPCON, SPOS) MD 35210 GEAR_STEP_POSCTRL_ACCEL * ACC * ACCFXS 4: Acceleration in the current gear stage with active position control while tapping (G331, G332 - only with corresponding configuration of the second data block) MD 35212 GEAR_STEP_POSCTRL_ACCEL2 5: Not used 6: Not used 7: Not used 8: Acceleration limited by the preparation for thread cutting and the synchronous spindle coupling of the leading spindle on account of limited acceleration of the following spindle.					

9: Not used 10: Not used 11: Not used 12: Acceleration limited by tool parameter $TC_TP_MAX_ACCEL 13: Not used

-				Long Integer	r
Multi-line: yes	Logical spindle index		numSpindlesLog		

acSmaxVelo	$AC_SMAXVELO[]		
Maximum spindle speed This variable returns the maximum spindle speed for spindle mode. This is formed from the smallest active speed limitation, and cannot be exceeded by speed programming or override > 100%. A speed limitation is indicated by the VDI interface signal DB31..,DBX83.1 'Setpoint speed limited' and by $AC_SPIND_STATE, bit 5 (speed limitation active). The cause of the speed limitation (machine, setting data, G code, VDI interface signal etc.) can also be determined with the system variable $AC_SMAXVELO_INFO. If the spindle is in axis mode, then the speed is not limited by $AC_SMAXVELO but instead the machine data (MAX_AX_VELO, ...) typical for axis mode are active.			

rev/min, user defined				Double	r
Multi-line: yes	Logical spindle index		numSpindlesLog		

acSmaxVeloInfo	$AC_SMAXVELO_INFO[]		
Identifier (index) for the speed-limiting data (machine/setting data, etc.) The system variable provides additional information about $AC_SMAXVELO, and returns the definitive data (machine, setting data, G code, VDI interface, etc.) as identifier/index. The speed-limiting data can be determined with the index from the following table of existing spindle speed limitations . 0: No limitation (SERUPRO) 1: Speed limited by clamping arrangement (chuck speed), MD 35100 SPIND_VELO_LIMIT; 2: Speed limitation (upper limit, that is maximum speed) in the current gear stage, MD 35130 GEAR_STEP_MAX_VELO_LIMIT * VELOLIMA (maximum speed factor from part program) 3: Speed limitation in the current gear stage with position control active MD 35132 GEAR_STEP_PC_MAX_VELO_LIMIT * VELOLIMA (among other items with SPCON, SPOS) 4: Speed limited to SD 43220 SPIND_MAX_VELO_G26 (part program instruction: G26 S.. or PI service from HMI) 5: Speed limited by set VDI interface signal DB31,...DBX3.6 to MD 35160 SPIND_EXTERN_VELO_LIMIT 6: Speed limited at constant cutting speed by part program instruction: LIMS (G96, G97, G961) SD 43230 SPIND_MAX_VELO_LIMS; 7: Speed limited by Safety Integrated to safe speed (SG) 8: Speed limited by the preparation for thread cutting and synchronous spindle coupling of the leading spindle on account of the speed limitation of the following spindle. 9: Speed limited to the maximum speed of the drive by the drive parameters, e.g. SINAMICS: p1082. 10: Speed limited to MD 36300 ENC_FREQ_LIMIT with functions that require a functioning measuring system, e.g. position control and for the master spindle with the G functions G95, G96, G97, G33..35. The limitation takes into account the encoder speed, the MS arrangement (direct/indirect), MS limiting frequency and the current parameter set. 11: Limitation of the overlaid movement of the following spindle to the dynamic remaining after the coupling. 12: Speed limited by tool parameter (in preparation) 13: Speed limited by tool carrier (in preparation)		

-				Long Integer	r
Multi-line: yes	Logical spindle index		numSpindlesLog		

acSminVelo	$AC_SMINVELO[]				
Minimum spindle speed This variable returns the minimum spindle speed for speed control mode. This is formed from the highest active speed increase, and cannot be undershot by speed programming or override < 100%. A speed increase is indicated by the VDI interface signal DB31..,DBX83.2 'Setpoint speed increased' and by $AC_SPIND_STATE, bit 6 (speed increase active). The cause of the speed increase (machine, setting data, G code, VDI interface signal etc.) can also be determined with the system variable $AC_SMINVELO_INFO. If the spindle is in axis or positioning mode, then the speed is not increased by $AC_SMINVELO.					
rev/min, user defined				Double	r
Multi-line: yes	Logical spindle index		numSpindlesLog		

acSminVeloInfo	$AC_SMINVELO_INFO[]				
Identifier (index) for the speed-limiting data (machine/setting data, etc.) The system variable provides additional information about $AC_SMAXVELO, and returns the definitive data (machine, setting data, G code, VDI interface, etc.) as identifier/index. The speed-limiting data can be determined with the index from the following table of existing spindle speed limitations. The system variable provides additional information about $AC_SMINVELO, and returns the speed increasing data (machine, setting data) as identifier/index. The speed-increasing data can be determined with the index from the following table of existing spindle speed increases.					
-				Long Integer	r
Multi-line: yes	Logical spindle index		numSpindlesLog		

acSpindState	$AC_SPIND_STATE[]				
This variable returns the spindle status in speed control mode only (Bit0 =1). The variable $AA_INPOS_STAT is relevant in positioning and axis modes. $AC_SPIND_STATE returns no values in synchronous mode. Bit 0: "Speed control mode active" (condition for validity of the bits of the variable $AC_SPIND_STATE[) Bit 1: "Spindle stopped" (corresponds to VDI interface signal DB31..,DBX61.4) Bit 2: "Spindle programmed" (e.g. M3, M4 S.., FC18, ..) (corresponds to VDI interface signal DB31..,DBX64.4/5 or 6/7) Bit 3: "Spindle in set range" (corresponds to VDI interface signal DB31..,DBX83.5) Bit 4: Free Bit 5: "Limitation of setpoint speed active" remains as long as the programmed or overrride speed remains higher than the speed in the system variable $AC_SMAXVELO. Corresponds to VDI interface signal DB31..,DBX83.1 'Setpoint speed limited'. Bit 6: "Increasing setpoint speed active" remains as long as the programmed speed remains less than the speed in the system variable $AC_SMINVELO. Corresponds to VDI interface signal DB31..,DBX83.2 'Setpoint speed increasing' Bit 7: "Spindle accelerating" (setpoint side), remains as long as the spindle is accelerating to the defined setpoint speed. Bit 8: "Spindle braking" (setpoint side), remains as long as the spindle is braking to the defined setpoint speed or comes to a standstill. Bit 9: "Position control active"					
-				Long Integer	r
Multi-line: yes	Logical spindle index		numSpindlesLog		

actGearStage					
Actual gear stage of spindle					
-				UWord	r
Multi-line: yes	Logical spindle index		numSpindlesLog		

actSpeed					
Spindle speed actual value					
rev/min, user defined				Double	r
Multi-line: yes	Logical spindle index		numSpindlesLog		

channelNo					
Number of channel in which spindle is configured					
-				UWord	r
Multi-line: yes	Logical spindle index		numSpindlesLog		

cmdAngPos					
Spindle position (SPOS)					
Degree, user defined				Double	r
Multi-line: yes	Logical spindle index		numSpindlesLog		

cmdConstCutSpeed					
Constant cutting 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		

dummy					
dummy					
-				UWord	r
Multi-line: yes	Logical spindle index		numSpindlesLog		

gwpsActive	{$GWPS}				
SUG programming active (SUG is the function "constant perimeter speed of grinding wheel) 0 = inactive 1 = active					
-				UWord	r
Multi-line: yes	Logical spindle index		numSpindlesLog		

index					
Absolute axis index referred to MD					
-				UWord	r
Multi-line: yes	Logical spindle index		numSpindlesLog		

name					
Spindle name Note: If several logical spindles are referred to one physical spindle with active spindle conversion and access is made via area N of module SSP2, then the name of the first suitable logical spindle is output.					
-				String[32]	r
Multi-line: yes	Logical spindle index		numSpindlesLog		

namePhys					
Name of assigned physical spindle, identical to "name" variable.					
-				String[32]	r
Multi-line: yes	Logical spindle index		numSpindlesLog		

opMode					
Spindle mode 0 = spindle mode 1 = oscillation mode (gear step changeover) 2 = positioning mode 3 = synchronous mode 4 = axis mode					
-				UWord	r
Multi-line: yes	Logical spindle index		numSpindlesLog		

pSMode	$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		

1.5.7 Area C, Module 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 work 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 a settable work 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 a settable work 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 a settable work offset					
-				Double	wr
Multi-line: yes	Frame index * numMachAxes + axis number		$MC_MM_NUM_USER_FRAMES * (numGeoAxes + numAuxAxes)		

1.5.8 Area C, Module FA: Active channel-specific frames

OEM-MMC: Linkitem

/Channel/ActualFrame/...

The following frame indices are available:

0: $P_ACTFRAME = current resulting work offset
1: $P_IFRAME = current settable work offset
2: $P_PFRAME = current programmable work offset
3: EXTFRAME = current external work offset
4: TOTFRAME = current total work offset = sum of ACTFRAME and EXTFRAME
5: $P_ACTBFRAME = current total base frame
6: $P_SETFRAME = current 1st system frame (set actual value, scratching)
7: $P_EXTSFRAME = current 2nd system frame (set actual value, scratching)
8: $P_PARTFRAME = current 3rd system frame (TCARR and PAROT with orientable tool carrier)
9: $P_TOOLFRAME = current 4th system frame (TOROT and TOFRAME)
10: $AC_MEASFRAME = result frame for workpiece and tool measurement
11: $P_WPFRAME = current 5th system frame (workpiece reference points) as from SW 6.3
12: $P_CYCFRAME = current 6th system frame (cycles) as from SW 6.3
13: $P_TRAFRAME = current 7th system frame (transformation) as from SW 7.1
14: $P_ISO1FRAME = current ISO system frame for G51.1 mirroring as from SW 7.4
15: $P_ISO2FRAME = current ISO system frame for G68 2DROT as from SW 7.4
16: $P_ISO3FRAME = current ISO system frame for G68 3DROT as from SW 7.4
17: $P_ISO4FRAME = current ISO system frame for G51 scale as from SW 7.4
18: $P_ACSFRAME = current resulting frame for the SZS (ACS) as from SW 7.4
19: $P_RELFRAME = current 12th system frame for relative coordinate systems as from SW 7.5

The maximum frame index is 18.

linShift	diverse, siehe Bausteinbescheibung				PA
Translation of an active work offset (the physical unit is defined in basicLengthUnit in module Y in area N).					
mm, inch, user defined				Double	r
Multi-line: yes	Frame index * numMachAxes + axis number		11 * numMachAxes		

mirrorImgActive	diverse, siehe Bausteinbescheibung				PA
Mirroring enabled in an active work offset 0 = mirroring not active 1 = mirroring active					
-				UWord	r
Multi-line: yes	Frame index * numMachAxes + axis number		11 * numMachAxes		

rotation	diverse, siehe Bausteinbescheibung				PA
Rotation of an active work offset					
Degree				Double	r
Multi-line: yes	Frame index * numMachAxes + axis number		11 * numMachAxes		

scaleFact	diverse, siehe Bausteinbescheibung				PA
Scaling factor of an active work offset					
-				Double	r
Multi-line: yes	Frame index * numMachAxes + axis number		11 * numMachAxes		

1.5.9 Area C, Module 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 work 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		

mirrorImgActive	diverse, siehe Bausteinbescheibung				PA
Mirroring of an external work offset 0 = mirroring not active 1 = mirroring active					
-				UWord	r
Multi-line: yes	Frame index * numMachAxes + axis number		11 * numMachAxes		

rotation	diverse, siehe Bausteinbescheibung				PA
Rotation of an external work offset					
Degree				Double	r
Multi-line: yes	Frame index * numMachAxes + axis number		11 * numMachAxes		

scaleFact	diverse, siehe Bausteinbescheibung				PA
Scaling factor of an external work offset					
-				Double	r
Multi-line: yes	Frame index * numMachAxes + axis number		11 * numMachAxes		

1.6 State data of drives

1.6.1 Area H, Module 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, Module 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 C, Module TO: Tool data of the active tool

OEM-MMC: Linkitem

/Channel/Compensation/...

Tool data of the active tool

cuttEdgeParam					
Parameter of the active tool cutting edge					
-	0			Double	r
Multi-line: yes	Number of the parameter:		24		

cuttEdgeParamMod					
Modified parameter of the active tool cutting edge. The rotation has been included in the calculation, so the value may not correspond with the original tool data.					
-	0			Double	r
Multi-line: yes	Number of the parameter:		24		

1.7.2 Area T, Module 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.

cuttEdgeParam	$TC_DPCEx[y,z] x = ParamNo y = ToolNo z = EdgeNo				
Replaced by edgeData The value for the tool type is stored internally as an integer.					
-	0			Double	wr
Multi-line: yes	For edge offset value parameters:		(numCuttEdgeParams + 1) * maxnumCuttEdges_Tool		

edgeData	$TC_DPx[y,z] x = ParamNo y = ToolNo z = EdgeNo				
Offset value parameter and edge list with D numbers for a tool 1st section: Offset value parameter for a tool edge: The meaning of each parameter depends on the tool type. At present 25 parameters are reserved for each tool edge (however only some of them are used). To be flexible for future extensions use the variable 'numCuttEdgeParams' rather than the fixed number 25 for the number of parameters. A detailed description of tool parameters can be found in the documentation Tool Offset (W1), Section "Tool edge". The following list is a summary of the tool edge parameters: Parameter 1: Geometry -- tool type ($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.					
mm, inch, user defined	0			Double	wr
Multi-line: yes	For edge offset value parameters:		(numCuttEdgeParams + 1) * maxnumCuttEdges_Tool		

1.7.3 Area T, Module 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	Tool number T		max. T-Nummer		

duploNo	$TC_TP1				FBW
Duplo number (number of replacement tool) In the tool management each tool is explicitly defined both by its identifier and its duplo number. This means that a T-area can only contain tool identifiers with different duplo numbers.					
-	T-Nummer			UWord	r
Multi-line: yes	Tool number T		32000		

numCuttEdges	$P_TOOLND[x] x = ToolNo				
Number of cutting edges of a tool					
-				UWord	r
Multi-line: no			1		

toolIdent	$TC_TP2				FBW
Tool identifier					
-	"<T-Nummer>"			String[32]	r
Multi-line: yes	Tool number T		32000		

toolInMag	$A_TOOLMN[x] x = ToolNo T				
Current magazine in which the tool is located					
-				UWord	r
Multi-line: yes	Tool number T		32000		

toolInPlace	$A_TOOLMLN[x] x = ToolNo T				
Current location in which the tool is located					
-				UWord	r
Multi-line: yes	Tool number T		32000		

toolInfo	$TC_TP11				FBW
Tool information for MMC Not currently assigned					
-	0			UWord	wr
Multi-line: yes	Tool number T		32000		

toolMaxAcc					
Maximum angular acceleration of the tool if the value is >0. There is no monitoring if no acceleration limit is defined (=0).					
Rev/s2, user defined				Double	wr
Multi-line: yes	Tool number T		32000		

toolMaxVelo					
Maximum speed of the tool if the value is >0. There is no monitoring if no speed limit is defined (=0).					
rev/min, user defined				Double	wr
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	Tool number T		32000		

toolMyMag	$A_MYMN				
Owner magazine of the tool magazine from which the tool was loaded 0 = the tool is not loaded. If toolInMag is >0 at the same time, the T number will specify a manual tool, or TMMG is not active.					
-	-	0	max. Nummer eines def. Magazins	UWord	r
Multi-line: yes	Tool number T		max. T-Nummer		

toolMyPlace	$A_MYMLN				
Owner magazine of the tool - Magazine location from which the tool was loaded 0 = the tool is not loaded. If toolInPlace is >0 at the same time, the T number will specify a manual tool, a valid magazine location number or TMMG is not active.					
-	-		max. Nummer def. Magazinplatz	UWord	r
Multi-line: yes	Tool number T		max. T-Nummer		

toolProtAreaFile					
toolProtAreaFile					
-				String[32]	r
Multi-line: no					

toolSearch	$TC_TP10				FBW
Type of tool search for replacement tools 0: no strategy 1: next duplo no. 2: shortest path					
-	0			UWord	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		

toolplace_spec	$TC_TP7				FBW
Magazine location type of tool					
-	9999			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		

1.7.4 Area T, Module 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	diverse, siehe Variablenbeschreibung				
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. colIndex: TNo 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		

1.7.5 Area T, Module TUD: Tool data: User-defined data

OEM-MMC: Linkitem

/Tool/User/...

The TUD data module is defined as a two-dimensional variable array. The module contains user-defined data for all tools. Each element can be addressed via a column and row index:
The column index is the number of the user-defined tool parameter. The number of tool parameters (columns) is given by the variable "numToolParams_tu" in module Y in area N.
The 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					
Siemens application tool parameter					
-				Double	r
Multi-line: no					

1.7.6 Area T, Module TUO: Tool edge data: User-defined data

OEM-MMC: Linkitem

/Tool/User/...

The TUO data module is organized as a two-dimensional variable array. The module contains user-defined cutting 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 the cutting edges of a tool are to be found in one column.The assignment of a tool to a T-number is to be found in the Tool directory (TV) module in the associated area T. If a non-existent tool number is given for the column index, the request is acknowledged negatively.
The number of rows is derived from the number of parameters per cutting edge and the number of cutting edges of a tool:

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

The number of parameters per cutting edge "numCuttEdgeParams_tu" is given in module Y in area N. The number of tool-specific cutting edges "/T/TV/numCuttEdges" are to be found in module TV in the associated area T.
If neccessary, several rows can be addressed, so that, for example, all user-defined cutting edge data of a single tool can be read in one request. The data are all of the same data type.

edgeData					
Siemens application tool parameter					
-				Double	r
Multi-line: no					

1.7.7 Area T, Module 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		

1.7.8 Area T, Module 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 value is the magazine number. If an error occurs, an error number is set.					
-				UWord	r
Multi-line: no					

magCMCmdPar2					W4
Return variable for the command MagCBCmd In case of a succesfull return, the return value is the place number. If an error occurs an error number is set.					
-				UWord	r
Multi-line: no					

magRPlaces					W4
Total number of real magazine locations (incl. buffer and loading locations)					
-				UWord	r
Multi-line: no					

magSearch	$TC_MAMP2				W4
Type of tool search. This variable is bitwise defined. A set bit has the following meaning: Bit0: search active tool Bit1: search tool by shortest path Bit8: begin search at first location (forwards) Bit9: begin search at current location forwards Bit10: begin search at last location (backwards) Bit11: begin search at current location backwards Bit12: begin search at current location symmetrically					
-				UWord	r
Multi-line: no					

magVPlaces					W4
Number of defined locations for the control block Number of virtual locations (without buffer and loading locations) for all real magazines in this area unit					
-				UWord	r
Multi-line: no					

magZWMag					W4
Number of internal buffer magazine					
-				UWord	r
Multi-line: no					

modeWearGroup	$TC_MAMP3	
Definition of strategies relating to wear group. The value is bit-coded. Default setting = 0. Effects on tool status Bit Value Meaning 0 0 When a wear group is activated internally, the status of the tools it contains remains unchanged. 1 When a wear group is activated internally, the status of the tools it contains changes. One tool from each tool group is set to the "active" state. 1 0 When a wear group is disabled internally, the status of the tools it contains remains unchanged. 1 When a wear group is disabled internally, the status of the tools it contains changes. The "active" status is cancelled for all tools. "Internally" in this instance means disabling or activation due to a tool change necessitating a change in the wear group. Activating/disabling the appropriate tools after writing system parameters or via OPI 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				

1.7.9 Area T, Module 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	MagazineNo		numMagsMax		

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

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

1.7.10 Area T, Module 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		

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

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

magPlaceSearchStrat					
magPlaceSearchStrat					
-				UWord	r
Multi-line: no					

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		

magWearCompoundNo	$TC_MAP9				
Each magazine has its own active wear group (wear group number). The number of this group is stored in OPI variables magWearCompoundNo: Meaning: Number of active wear group. =0: No wear group active. >0: Number of wear group in which tool search commences. (this is the number of the active wear group.) <0: Number of wear group in which tool search commences. However, this wear group is disabled which means that the next tool search is started in the next possible wear group. This system parameter can thus also be used to disable a wear group. See also wear group number of magazine location / $TC_MPP7 and modeWearGroup / $TC_MAMP3. Previous name: actWearGrInMag -32000, ..., -1, 0, 1, 2, ... 32000					
-	0			Long Integer	wr
Multi-line: yes	Magazine number		numMagsMax		

1.7.11 Area T, Module 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 by a column and line index:
The column index is the magazine number, i.e. the configuration data for all locations of a magazine are contained in one column. The assignment of a magazine to a magazine number is stated in the associated area T of the associated Magazine directory module (TMV). If a non-existent magazine number is specified for the column index, the request is negatively acknowledged.
The number of lines is derived from the number of parameters per magazine location and from the number of magazine locations:

maxLineindex = numMagPlaceParams * magNrPlaces

The number of parameters per magazine location "numMagPlaceParams" is given in area N of module Y.
The line indices are based on the following scheme:
1: Location type ($TC_MPP1) (read only)
1: Magazine location
2: Spindle
3: Gripper
4: Loader
5: Transfer location
6: Loading station
7: Loading point
2: Location type ($TC_MPP2) (read only)
>0: Location type for virtual location
=0: "Match all" (buffer location)
9999: Undefined (not a virtual location)
3: T number of tool in this location ($TC_MPP6)
4: Consideration of adjacent location on / off ($TC_MPP3)
0: off
1: on
5: Location status ($TC_MPP4)
1: Disabled
2: Free (<> occupied)
4: Reserved for tool in buffer location
8: Reserved for tool to be loaded
16: Occupied in left half-location
32: Occupied in right half-location
64: Occupied in top half-location
128: Occupied in bottom half-location
6: Physical magazine reference (read only)
Magazine number of magazine to which location belongs
7: Type index ($TC_MPP5) (read only) and new: Wear group number as from SW 5.1
Type index/wear group number is read only in SW earlier than 5.1 but read/write as from SW 5.1 if it is assigned "Wear group" meaning.
Type index: The locations of a location type in a magazine are numbered in ascending order, e.g. type=2, type index=5; ==> Spindle5)
(previous meaning when location type = 1 before P5: Equals location number when location type=1)
Wear group number as from SW 5.1 ($TC_MPP5)
When location type = 1: Number of wear group to which this magazine location is assigned.
Value range: -32000, ..., -1, 0, 1, 2, ... 32000
=0: Not assigned to a wear group
>0: Number of assigned wear group, this wear group is enabled

<0: Number of assigned wear group, this wear group is disabled
By negating this system parameter, it is possible to disable or enable the whole assigned wear group.
See also magWearCompoundNo / $TC_MAP9 (active wear group number) and modeWearGroup / $TC_MAMP3 (general settings for wear grouping).
8: Adapter number as from SW 5.1 ($TC_MPP7)
Reference to adapter data set number.

Associated system data:
The number of parameters of this module changes accordingly:
N / Y, global system data, numMagPlaceParams = 8 as from SW 5.1

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

The locations of the buffer magazine and the loading magazine are numbered in ascending order independently of the location type index.

If necessary, several lines can be addressed, so that, for example, all location data of a magazine can be read in a single request. The location data are all of the same type.

placeData	diverse, siehe Variablenbeschreibung				
P1: 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 array) ($TC_MPP4) P6: Reference for physical magazine (read access only) P7: Location type index (location type numbering) ($TC_MPP5) P8: Number of adapter in magazine location ($TC_MPP7) P9: Number of the spindle assigned to the buffer magazine location ($TC_MPP_SP) colIndex: Tool magazine number Attention: This variable is called "dummy" in the non-Windows MMC and the PLC.					
-				UWord	wr
Multi-line: yes	(LocationNo - 1) * numMagPlaceParams + ParameterNo		numMagPlaceParams * magNrPlaces		

1.7.12 Area T, Module 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	diverse, siehe Variablenbeschreibung				
P1: Distance between the change position of magazine n and the location m of the first internal magazine (loading magazine, 9999) ($TC_MDP1) P2: Distance between the change position of magazine n and the location m of the second internal magazine (buffer magazine, 9998) ($TC_MDP2) colIndex: Tool magazine number					
-				UWord	r
Multi-line: yes	(LocationNo - 1) * numPlaceMulti * numPlaceMultiParams+ParameterNo		numPlaceMulti * numPlaceMultiParams * magNrPlaces		

1.7.13 Area T, Module 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. colIndex: Number of the location hierarchy + 1					
-				UWord	r
Multi-line: yes	Number of the location type + 1		Wert aus Zeile 1		

1.7.14 Area T, Module 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.

TnumWZV					
Last assigned T-number for tool management The last assigned T number is the T number of the new tool last created in the NCK through an NC language command or the PI service.					
-				UWord	r
Multi-line: no					

nrDuplo					
Duplo number					
-				UWord	r
Multi-line: yes	Serial number		numTools		

numCuttEdges					
Number of cutting edges of a tool					
-			9	UWord	r
Multi-line: yes	Serial number		numTools		

numToolGroups					
numToolGroups					
-				UWord	r
Multi-line: no					

numTools					
Number of tools in the area TO					
-		0	MD MM_NUM_TOOL	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		

1.7.15 Area T, Module 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.		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.		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.		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.		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:		numCuttEdgeParams * maxnumCuttEdges_Tool		

parDataTS					
Parameterizing: For each parameter of the module TS a value can be stored as a comparison value for a 'complex search' (_N_TSEARC). The comparison value is combined with the corresponding parameter in the module TS according to parMasksTS. The size of the column matches the data set of an edge in module TS. See module TS					
-				Double	wr
Multi-line: yes	Line index in the TS module:		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.		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:		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:		numCuttEdgeParams_tus * maxnumCuttEdges_Tool		

parDataToolIdentTD					
Parameterizing: For the parameter with data type string[32] (tool identifier) of the module TD a value can be stored as a comparison value for a 'complex search' (_N_TSEARC). The comparison value is combined with the corresponding parameter in the module TD according to parMasksTD. See module TD					
-				String[32]	wr
Multi-line: no					

parMasksTAD					
Parameterizing: There is a mask for each parameter of the module TAD that indicates whether it is to serve as a search criterion for a 'complex search' (_N_TSEARC) and how it is to be combined. The corresponding comparison values are stored in parDataTAD. If more than one parameter (i.e. search criteron) has been selected (#0), they are logically combined with AND. Value 0 : Corresponding operand is not evaluated / Variable is not a criterion for comparison Value 1 : == (equal) Value 2 : < (less than) Value 3 : > (greater than) Value 4 : <= (less or equal) Value 5 : >= (greater or equal) Value 6 : && (bitwise AND, value only allowed for operands of the types WORD and DOUBLEWORD) For string operands "==" is the only operator allowed					
-	0	0	6	UWord	wr
Multi-line: yes	Column index in the module TAD, i.e. the number of the user-defined tool parameter.		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.		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.		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.		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, i.e. a cutting edge offset value parameter:		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:		numCuttEdgeParams_ts * maxnumCuttEdges_Tool		

parMasksTU					
Parameterizing: There is a mask for each parameter of the module TU that indicates whether it is to serve as a search criterion for a 'complex search' (_N_TSEARC) and how it is to be combined. The corresponding comparison values are stored in parDataTU. If more than one parameter (i.e. search criteron) has been selected (#0), they are logically combined with AND. Value 0 : Corresponding operand is not evaluated / Variable is not a criterion for comparison Value 1 : == (equal) Value 2 : < (less than) Value 3 : > (greater than) Value 4 : <= (less or equal) Value 5 : >= (greater or equal) Value 6 : && (bitwise AND, value only allowed for operands of the types WORD and DOUBLEWORD) For string operands "==" is the only operator allowed					
-	0	0	6	UWord	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.		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:		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:		numCuttEdgeParams_tus * maxnumCuttEdges_Tool		

resultCuttingEdgeNrUsed	$A_USEDD				
D numbers of the cutting edges used since the last workpiece count, that have previously been used on the defined tool carrier via resultNrOfCutEdgesUsed. Various D offsets for a tool indicate multiple entries of the tool, that means a T number can be present more than once. The two variables are linked to each other. resultNrOfCutEdgesUsed has to be read first, and then the individual T numbers with resultToolNrUsed. See also $A_USEDND, $A_USEDT and SETPIECE command. 0 maximum number of cutting edges in NCK					
-	0	0	max. Anzahl Schneiden in NCK	Long Integer	r
Multi-line: yes	((i. tool carrier-1) * line2 of column3 (resultNrOfCutEdgesUsed)) + consecutive number of the tool used		Zeile 1 * Zeile 2 von resultNrOfCutEdgesUsed		

resultNrOfCutEdgesUsed	$A_USEDND				
Line 1: Number of tool carriers Line 2: Maximum number of entries of resultToolNrUsed or resultCuttingEdgeNrUsed per tool carrier Line i+2: Number of the i. tool carrier Line i+3: Number of cutting edges which have been used since the last workpiece count on the i. tool carrier. This corresponds to $A_USEDND. The T and D numbers of the cutting edges can be read with resultToolNrUsed and resultCuttingEdgeNrUsed respectively. If TOOLMAN is not active and $MC_T_M_ADDRESS_EXT_IS_SPINO = FALSE, then line 1 = 1, $MC_T_M_ADDRESS_EXT_IS_SPINO = TRUE, then line 1 = 32. If tool monitoring is not active, line 2 = 0. See also $A_USEDT, $A_USEDD and SETPIECE command 0 maximum number of cutting edges in NCK					
-	0	0	max. Anzahl Schneiden in NCK	Long Integer	r
Multi-line: yes	Meaning of the index: See description		2*max.Anz. der Distanzbez. zw.Mag. und WZ-Haltern + 2 = 66		

resultNrOfTools					
Result: Number of tools found In the case of _N_TMGETT, it is possible to find no tools (value=0) or exactly 1 tool (value 1); in the case of _N_TSEARC, the number of found tools can be any number > 0, limited by the number of tools in the NC or no tools at all (value=0).					
-	0	0	numTools	UWord	r
Multi-line: yes	1		1		

resultToolNr					
Result: T-numbers of the tools found The array elements contain the internal T- numbers of the tools found. The storing order is the order in which the tools have been found by the PI-Service.					
-	0	0	31999	UWord	r
Multi-line: no			resultNrOfTools		

resultToolNrUsed	$A_USEDT				
T numbers of the cutting edges used since the last workpiece count, that have previously been used on the defined tool carrier via resultNrOfCutEdgesUsed. Various D offsets for a tool indicate multiple entries of the tool, that means a T number can be present more than once. The two variables are linked to each other. resultNrOfCutEdgesUsed has to be read first, and then the individual T numbers with resultToolNrUsed. See also $A_USEDND, $A_USEDD and SETPIECE command. 0 maximum number of cutting edges in NCK					
-	0	0	max. Anzahl Schneiden in NCK	Long Integer	r
Multi-line: yes	((i. tool carrier-1) * line2 of column3 (resultNrOfCutEdgesUsed)) + consecutive number of the tool used		Zeile 1 * Zeile 2 von resultNrOfCutEdgesUsed		

1.7.16 Area T, Module 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. colIndex: Magazine number of magazine					
-	0			Long Integer	wr
Multi-line: yes	Number of the user-defined parameter		numMagParams_u		

1.7.17 Area T, Module 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. colIndex: Magazine number of the magazine					
-	0			Long Integer	wr
Multi-line: yes	Number of the user-defined parameter + numMagLocParams_u * (number of the magazine location - 1)		numMagLocParams_u * magNrPlaces		

1.7.18 Area T, Module 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. colIndex: TNo					
-	0			Double	wr
Multi-line: yes	Number of the user-defined parameter + (number of the cutting edge -1) * numCuttEdgeParams_tus		numCuttEdgeParams_tus * maxnumCuttEdges_Tool		

1.7.19 Area T, Module 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					
Adapter data colIndex: AdaptNo					
mm, inch, user defined	0.0			Double	wr
Multi-line: yes	ParameterNo		numParams_Adapt		

1.7.20 Area T, Module 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)).

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

cuttEdgeNo					
Number of edge for this tool Meaningful and defined only in connection with "unique D numbers" function.					
-		1	maxnumCuttEdges_Tool	UWord	r
Multi-line: yes	Serial number of active edges		numActDEdges		

duploNo					
Duplo number Meaningful and defined only in connection with "unique D numbers" function.					
-				UWord	r
Multi-line: yes	Serial number of active edges		numActDEdges		

numActDEdges					
Number of D numbers in this list Meaningful and defined only in connection with "unique D numbers" function. When tool management function is active: Specifies the number of edges belonging to tools with "active" status (contained in the TO unit) When tool management function is not active: Specifies the number of all edges contained in the TO unit.					
-				UWord	r
Multi-line: yes	1		1		

toolIdent					
Tool identifier Meaningful and defined only in connection with "unique D numbers" function.					
-				String[32]	r
Multi-line: yes	Serial number of active edges		numActDEdges		

toolInMag					
Magazine in which tool is located Meaningful and defined only in connection with "unique D numbers" function.					
-				UWord	r
Multi-line: yes	Serial number of active edges		numActDEdges		

toolInPlace					
Magazine location of tool Meaningful and defined only in connection with "unique D numbers" function.					
-				UWord	r
Multi-line: yes	Serial number of active edges		numActDEdges		

toolNo					
Internal T number Meaningful and defined only in connection with "unique D numbers" function.					
-				UWord	r
Multi-line: yes	Serial number of active edges		numActDEdges		

1.7.21 Area T, Module 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			Long Integer	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		

1.7.22 Area T, Module 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		

1.7.23 Area T, Module 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 colIndex: TNo					
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, Module 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 colIndex: TNo					
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, Module 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 colIndex: TNo					
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		

1.7.26 Area T, Module 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					
Replaced by edgeData					
mm, inch, user defined	0.0			Double	wr
Multi-line: yes	For edge offset value parameters:		(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:		(numCuttEdgeParams + 1) * maxnumCuttEdges_Tool		

1.7.27 Area T, Module TAD: Application-specific data

OEM-MMC: Linkitem

/Tool/Data/...

Data module TAD is organized as a 2-dimensional variable array. This module contains application-specific data for all tools. Every element can be addressed via a column and row index:

The column index is the number of the user-defined tool parameter. The number of tool parameters (columns) can be found in variable numToolParams_tad in area N / module Y.

The row index is the tool number. Attempts to access non-existent tools are negatively acknowledged.

Application-specific tool data are all of the same data type.

Application-specific tool data are reserved for SIEMENS applications.

siemData	$TC_TPCSx[y]				
Siemens application tool parameter Important: 2-dimensional variable. Column index corresponds to the parameter number. Reserved for SIEMENS applications. columnIndex: Number of the application-specific tool parameter					
-	0			Double	wr
Multi-line: yes	Tool number T		32000		

1.7.28 Area T, Module TAM: Application-specific magazine data

OEM-MMC: Linkitem

/Tool/MagazineDescription/...

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. colIndex: Magazine number					
-	0			UWord	wr
Multi-line: yes	Parameter number		numMagParams_tam		

1.7.29 Area T, Module TAO: Application-specific cutting edge data

OEM-MMC: Linkitem

/Tool/Compensation/...

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. colIndex: TNo					
-	0			Double	wr
Multi-line: yes	(EdgeNo-1) * numCuttEdgeParams_tao + ParameterNo		numCuttEdgeParams_tao * numCuttEdges		

1.7.30 Area T, Module TAP: Application-specific magazine location data

OEM-MMC: Linkitem

/Tool/Magazine/...

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. colIndex: Magazine number					
-	0			UWord	wr
Multi-line: yes	ParameterNumber + numMagLocParams_tap * MagazineLocationNumber-1		numMagLocParams_tap * magNrPlaces		

1.7.31 Area T, Module TAS: Application-specific monitoring data

OEM-MMC: Linkitem

/Tool/Supervision/...

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 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. colIndex: TNo					
-	0			Double	wr
Multi-line: yes	ParameterNumber + (EdgeNo -1) * numCuttEdgeParams_tas		numCuttEdgeParams_tas * numCuttEdges		

1.8 Machine and setting data

1.8.1 Area N, Module 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		

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		

MDL_POSCTRL_SYSCLOCK_TIME_RATIO	MD 10060: POSCTRL_SYSCLOCK_TIME_RATIO				
Position control cycle factor					
-		1	100	Long Integer	wr
Multi-line: no			1		

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, Module 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	Character	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, Module 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, Module 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, Module 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, Module 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 variables are defined by default. The number of R variables can be set via the channel-specific machine data 28050 (MM_NUM_R_PARAM). Up to 1000 R-variables can be set.

R	$R[x] x = ParameterNo				PA
R variables (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 variables (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, Module 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		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) Writeable as from SW 7.4.					
-				Long Integer	wr
Multi-line: yes	Number of the flag		MD $MC_MM_NUM_AC_MARKER		

acParam	$AC_PARAM[x] x = ParameterNo				
Dynamic parameters for motion-synchronous actions (Note: only with SYNACT) Writeable as from SW 7.4.					
-				Double	wr
Multi-line: yes	Number of the parameter		MD $MC_MM_NUM_AC_PARAM		

alignDummy					
dummy					
-				UWord	r
Multi-line: yes	1		1		

1.10 Servo

1.10.1 Area N, Module 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

servoDatFI32					
Servo data column: Signal ID / Servo cycle (see module header)					
-	0			Float	r
Multi-line: yes	Axis index / data format (see module header)		siehe Bausteinkopf		

servoDatFI64					
Servo data column: Signal ID / Servo cycle (see module header)					
-	0			Double	r
Multi-line: yes	1000+Axis index / data format (see module header)		siehe Bausteinkopf		

1.11 Diagnosis data

1.11.1 Area N, Module 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:		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:		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			Long Integer	r
Multi-line: yes	1		1		

dp611USpecAccKey					
Version and type information about available ACC contents					
-	0			Long Integer	r
Multi-line: yes	Drive number		maxnumDrives		

dp611USpecAccMask					
Bit-coded screenform indicating the drives for which special ACC files are available Bit 0 == 1 -> A special ACC is available for drive with log. drive number 1.					
-	0			Long Integer	r
Multi-line: yes	1		1		

dp611USpecAccPath					
Path in which the ACC files are stored in the NCK file system. This path might be empty later on if the files are to be supplied from the active file system. Current equivalent value: /_N_VS_DIR					
-	0			String[32]	r
Multi-line: yes	1		1		

dpAxisCfgMachAxisNr					
Machine axis !!CAUTION NCU LINK!!					
-	0	0	INT32_MAX	Long Integer	r
Multi-line: yes	Axis number		dpAxisCfgNumAxes		

dpAxisCfgNumAxes					
Number of axes entered in the system					
-	0	0	INT32_MAX	Long Integer	r
Multi-line: yes	1		1		

dpAxisCfgValid					
Axis info is available 0=Information is not available 1=Information is available					
-	0	0	1	Long Integer	r
Multi-line: yes	1		1		

dpAxisStateCtrlout					
Status of output drivers. 0=no axis status assigned 1=axis status assigned 2=axis status is cyclical 3=axis status assigned and cyclical					
-	0	0	3	UWord	r
Multi-line: yes	Axis number		dpAxisCfgNumAxes		

dpAxisStateEnc1					
Status encoder 1 driver 0=no axis status assigned 1=axis status assigned 2=axis status is cyclical 3=axis status assigned and cyclical					
-	0			UWord	r
Multi-line: yes	Axis number		dpAxisCfgNumAxes		

dpAxisStateEnc2					
Status encoder 2 driver 0=no axis status assigned 1=axis status assigned 2=axis status is cyclical 3=axis status assigned and cyclical					
-	0			UWord	r
Multi-line: yes	Axis number		dpAxisCfgNumAxes		

dpAxisStateLifeCntErrCtrlout					
This data counts the number of position control cycles since failure of the sign-of-life signal 0 to n= number of position control cycles since failure of the sign-of-life signal					
-	0	0	INT32_MAX	Long Integer	r
Multi-line: yes	Axis number		dpAxisCfgNumAxes		

dpAxisStateLifeCntErrEnc1					
This data counts the number of position control cycles since failure of the sign-of-life signal 0 to n= number of position control cycles since failure of the sign-of-life signal					
-	0			Long Integer	r
Multi-line: yes	Axis number		dpAxisCfgNumAxes		

dpAxisStateLifeCntErrEnc2					
This data counts the number of position control cycles since failure of the sign-of-life signal 0 to n= number of position control cycles since failure of the sign-of-life signal					
-	0			Long Integer	r
Multi-line: yes	Axis number		dpAxisCfgNumAxes		

dpBusCfgBaudrate					
Baud rate on DP bus (bit/s) The permissible baud rates are determined by the Profibus standard (DIN19245 EN50170)					
Hz	0			Double	r
Multi-line: yes	Bus number		dpBusCfgNumBuses		

dpBusCfgBusNo					
Bus number of the bus; used for conversion of "Bus index"=1...dpBusCfgNumBuses to "Bus number" All permissible bus numbers are possible: 1 = 1.DP bus on the PLC 2 = 2.DP / MPI bus on the PLC 3 = Virtual PROFIBUS 4 = Isochronous real-time Ethernet (reserved)					
-	0	0	4	Long Integer	r
Multi-line: yes	1		1		

dpBusCfgCycleTime					
The time required by the master to scan all slaves once (request, response), until the cycle starts from the beginning again.					
s, user defined	0	0	DOUBLE_MAX	Double	r
Multi-line: yes	Bus number		dpBusCfgNumBuses		

dpBusCfgDataExTime					
Data exchange time in [s,s,userdef]					
s, user defined	0	0	DOUBLE_MAX	Double	r
Multi-line: yes	Bus number		dpBusCfgNumBuses		

dpBusCfgNumBuses					
Number of DP buses Currently only one bus standardized acc. to Profibus DP standard					
-	0	0	1	Long Integer	r
Multi-line: yes	1		1		

dpBusCfgValid					
Bus configuration data are available TRUE= data exist and are initialized FALSE= no data exist					
-	0	0	1	Long Integer	r
Multi-line: yes	1		1		

dpBusStateAccessDurationAct					
Current access time to communications buffer for DP master					
-	0			Long Integer	r
Multi-line: yes	Bus number		dpBusCfgNumBuses		

dpBusStateAccessDurationMax					
Maximum access time to communications buffer for DP master					
-	0			Long Integer	r
Multi-line: yes	Bus number		dpBusCfgNumBuses		

dpBusStateAccessDurationMin					
Minimum access time to communications buffer for DP master					
-	0			Long Integer	r
Multi-line: yes	Bus number		dpBusCfgNumBuses		

dpBusStateAccessErrCnt1					
Number of bus access errors of type 1 since NCK Start					
-	0			Long Integer	r
Multi-line: yes	Bus number		dpBusCfgNumBuses		

dpBusStateAccessErrCnt2					
Number of bus access errors of type 2 since NCK Start					
-	0			Long Integer	r
Multi-line: yes	Bus number		dpBusCfgNumBuses		

dpBusStateAvgCycleBetweenErr1					
Average number of cycles between two bus access errors of type 1					
-	0			Long Integer	r
Multi-line: yes	Bus number		dpBusCfgNumBuses		

dpBusStateAvgCycleBetweenErr2					
Average number of cycles between two bus access errors of type 2					
-	0			Long Integer	r
Multi-line: yes	Bus number		dpBusCfgNumBuses		

dpBusStateCycleCnt					
Number of bus cycles since NCK Start					
-	0			Long Integer	r
Multi-line: yes	Bus number		dpBusCfgNumBuses		

dpBusStateDpmAction					
Indicator for operating progress of DP M					
-	0			Long Integer	r
Multi-line: yes	Bus number		dpBusCfgNumBuses		

dpBusStateDpmActual					
Current status of DP M bus - controlled by DP M					
-	0			UWord	r
Multi-line: yes	Bus number		dpBusCfgNumBuses		

dpBusStateDpmCtrl					
Booting status of processor for DP Master dpcadmin					
-	0			UWord	r
Multi-line: yes	Bus number		dpBusCfgNumBuses		

dpBusStateDpmError					
Error on status transitions					
-	0			Long Integer	r
Multi-line: yes	Bus number		dpBusCfgNumBuses		

dpBusStateDpmPrjCnt					
Modification counter for new DP configurations. Suggested use: *) Read modification counter (1) *) Read out configuring data *) Read modification counter (2) *) If the modification counters in (1) and (2) are identical and both display "valid", the data read from HW-Config will be consistent. even values -> configuration invalid uneven values -> configuration valid					
-	0			UWord	r
Multi-line: yes	Bus number		dpBusCfgNumBuses		

dpBusStateDpmRequest					
Desired status of DP M bus - request from HOST					
-	0			UWord	r
Multi-line: yes	Bus number		dpBusCfgNumBuses		

dpBusStateNumActiveSlaves					
This data indicates how many slaves can currently be accessed via the bus. This value is updated in online operation. The number of slaves on the bus is determined by the Profibus standard (DIN19245 EN50170)					
-	0	0	125	Long Integer	r
Multi-line: yes	Bus number		dpBusCfgNumBuses		

dpClientCfgId					
Identification client NCK/PLC/3RD					
-	0			UWord	r
Multi-line: yes	Client number		dpClientCfgNumClnt		

dpClientCfgNumClnt					
Number of clients					
-	0	0	INT32_MAX	Long Integer	r
Multi-line: yes	1		1		

dpClientCfgValid					
Client information is available 0=no client information available 1=client information is available					
-	0	0	1	Long Integer	r
Multi-line: yes	1		1		

dpClientStateComm					
Client status incl. output release 0=No output enable 1=Client state output enable					
-	0			UWord	r
Multi-line: yes	Client number		dpClientCfgNumClnt		

dpSlaveCfgAssignBus					
Bus number of the slave					
-	0			UWord	r
Multi-line: yes	Slave number		dpSlaveCfgNumSlaves		

dpSlaveCfgBusAddr					
The address of the slave on the bus. In addition to its own address, every slave has a broadcast address via which all salves can be addressed. The broadcast address is not available for individually addressing a single slave. 127: Broadcast address					
-	0	0	127	UWord	r
Multi-line: yes	Slave number		dpSlaveCfgNumSlaves		

dpSlaveCfgDataExchangeTime					
Time for the end of cyclical data transfer See dpSlaveCfgMasterAppCycTime					
s, user defined	0			Double	r
Multi-line: yes	Slave number		dpSlaveCfgNumSlaves		

dpSlaveCfgInputTime					
Time for actual-value sensing See dpSlaveCfgMasterAppCycTime					
s, user defined	0			Double	r
Multi-line: yes	Slave number		dpSlaveCfgNumSlaves		

dpSlaveCfgIsochronModeSupport					
Gives information whether the slave has been configured for isochronous mode on the PROFIBUS. 0: Isochronous mode not configured 1: Isochronous mode configured					
-	0	0		Long Integer	r
Multi-line: yes	Slave number		dpSlaveCfgNumSlaves		

dpSlaveCfgMasterAppCycTime					
Position controller cycle. For a detailed description, please refer to PROFIDRIVE PROFIL ANTRIEBSTECHNIK (Edition: V1.2 Draft, April 1999) Section 7 See PROFIDRIVE PROFIL ANTRIEBSTECHNIK (Edition: V1.2 Draft, April 1999) Section 7					
s, user defined	0			Double	r
Multi-line: yes	Slave number		dpSlaveCfgNumSlaves		

dpSlaveCfgNumSlaves					
Number of slaves configured in SDB1xxx. This value may not match the actual number of slaves connected to the bus. The number of slaves which can be configured for bus connection is determined by Profibus standard (DIN19245 EN50170).					
-	0	0	125	Long Integer	r
Multi-line: yes	1		1		

dpSlaveCfgOutputTime					
Time for setpoint acceptance See dpSlaveCfgMasterAppCycTime					
s, user defined	0			Double	r
Multi-line: yes	Slave number		dpSlaveCfgNumSlaves		

dpSlaveCfgProfibusCycleTime					
Bus cycle time See dpSlaveCfgMasterAppCycTime					
s, user defined	0			Double	r
Multi-line: yes	Slave number		dpSlaveCfgNumSlaves		

dpSlaveCfgValid					
This data indicates whether the slave data structure has already been initialized. The structure is initialized when a slave configuration or status data is accessed. Scanning dpSlaveCfgValid also activates initialization of the structure. True: Slave data are available False: Slave data are not available					
-	0	0	1	Long Integer	r
Multi-line: yes	1		1		

dpSlaveIdentNo					
Ident number of the slave					
-	0			UWord	r
Multi-line: yes	Slave number		dpSlaveCfgNumSlaves		

dpSlaveIdentNoEx					
The extended ID no. of the PROFIBUS slave helps to identify the PROFIBUS slaves not officially classified as such and therefore lack specificiation dpSlaveIdentNo.					
-	0			UWord	r
Multi-line: yes	Slave number		dpSlaveCfgNumSlaves		

dpSlaveStateComm					
The slave is active on the bus once the drive assigned to the slave has successfully logged on to the bus. True: Slave on bus False: Slave not on bus					
-	0	0	1	UWord	r
Multi-line: yes	Slave number		dpSlaveCfgNumSlaves		

dpSlaveStateIncCnt					
The incarnation counter of the slave is increased by one each time the slave is included in the bus. If the slave drops out of the bus, this counter is not changed. After the first time it has gone into the bus (that is the first operational status of the slave), the value is 1. In case of an area overflow, the count restarts at 0. This only functions with slaves which contain at least one assigned NC axis. In the case of other slaves (pure I/O slaves, or axes controlled by the PLC), this values remains at 0. From 0 (starting value after Restart) to a maximum of 2147483647 (2^31-1).					
-	0	0	2147483647	Long Integer	r
Multi-line: yes	Slave number		dpSlaveCfgNumSlaves		

dpSlaveStateSync					
The drive linked to this slave is operating in cyclic mode. Slaves without a drive are defined as "non-cyclical". True: Cyclical False: Non-cyclical					
-	0	0	1	UWord	r
Multi-line: yes	Slave number		dpSlaveCfgNumSlaves		

dpSlaveVendorId					
PROFIBUS: Always returns 0 PROFINET: Manufacturer's number of the device					
-	0			UWord	r
Multi-line: yes	Slave number		dpSlaveCfgNumSlaves		

dpSlotCfgAssignAxis					
This data supplies the axis indices of the drive, encoder 1 and encoder 2 for access in the Axis-Assign-Table. The 32-bit value consists of 4 bytes with the following meaning: Byte0(bits 0-7) = axis index of axis Byte1(bits 8-15) = axis index, encoder 1 Byte2(bits 16-23)= axis index, encoder 2 Byte3(bits 24-31)= provided for future extensions. A byte with the value 0xFF indicates that no axis index is defined for the relevant slot.					
-	255	0	32	Long Integer	r
Multi-line: yes	Slot number (PROFINET: Subslot number)		dpSlotCfgNumSlots		

dpSlotCfgAssignBus					
Bus number assigned to this slot Since only one bus is currently supported by Profibus DP, there is only one bus to which all slots are assigned.					
-	0	0	1	Long Integer	r
Multi-line: yes	Slot number (PROFINET: Subslot number)		dpSlotCfgNumSlots		

dpSlotCfgAssignClient					
This data supplies the clientIndex for accessing the Client Assign table. 0=no assignment possible (this applies to diagnostic and PKW slots) >0 assignment exists					
-	0	0	2	Long Integer	r
Multi-line: yes	Slot number (PROFINET: Subslot number)		dpSlotCfgNumSlots		

dpSlotCfgAssignMaster					
Number of master to which this slot is assigned Since only one bus is currently supported by Profibus DP and only one Class 1 Master exists per bus, there is only one master to which all slots are assigned.					
-	0	0	1	Long Integer	r
Multi-line: yes	Slot number (PROFINET: Subslot number)		dpSlotCfgNumSlots		

dpSlotCfgAssignSlave					
This data contains the bus address of the slave belonging to the nth slot. All legal slave addresses can be specified					
-	0	0	125	Long Integer	r
Multi-line: yes	Slot number (PROFINET: Subslot number)		dpSlotCfgNumSlots		

dpSlotCfgIoType					
I/O identifier 0 = input slot 1 = output slot 2 = diagnosis slot					
-	0	0	2	UWord	r
Multi-line: yes	Slot number (PROFINET: Subslot number)		dpSlotCfgNumSlots		

dpSlotCfgLength					
Length in number of bytes					
-	0	0	32	Long Integer	r
Multi-line: yes	Slot number (PROFINET: Subslot number)		dpSlotCfgNumSlots		

dpSlotCfgLogBaseAddress					
The logical basic address of the slot is assigned during configuration. Although it is not needed on the bus for data transfer purposes, this address is the only means by which a unique link can be created between the NCK and bus nodes.					
-	0	0	UINT16_MAX	UWord	r
Multi-line: yes	Slot number (PROFINET: Subslot number)		dpSlotCfgNumSlots		

dpSlotCfgNumSlots					
The total number of all slots configured in the system is stored in this data. 0 (lower limit) up to INT32_MAX(upper limit); Note that a slave cannot support more than 256 slots.					
-	0	0	INT32_MAX	Long Integer	r
Multi-line: yes	1		1		

dpSlotCfgPNSlotNr					
PROFIBUS: Not used PROFINET: Slot number within the IO device					
-	0	0	255	UWord	r
Multi-line: yes	PROFINET: Subslot number		dpSlotCfgNumSlots		

dpSlotCfgSlaveAddress					
This data contains the bus address of the slave to which this slot is assigned. Several slots may have the same slave address. The number of available addresses on the bus is determined by the Profibus standard (DIN19245 EN50170).					
-	0	0	125	UWord	r
Multi-line: yes	Slot number (PROFINET: Subslot number)		dpSlotCfgNumSlots		

dpSlotCfgSlotNr					
PROFIBUS: Slot number within the slave PROFINET: Subslot number within the IO device A maximum total of 256 slots can be assigned to each slave. 0: Diagnostic slot 2: Diagnostic slot 4: 1st data slot					
-	0	0	255	UWord	r
Multi-line: yes	Slot number (PROFINET: Subslot number)		dpSlotCfgNumSlots		

dpSlotCfgValid					
The slot data structure (CcIdent) exists and is initialized. True: Data are valid False: Data are invalid or not initialized					
-	0	0	1	Long Integer	r
Multi-line: yes	1		1		

dpSlotStateComm					
Status of slots (ok, failed, not processed by the NCK) 0= no sign of life 1= sign of life 2= not processed by NCK					
-	0	0	1	UWord	r
Multi-line: yes	Slot number (PROFINET: Subslot number)		dpSlotCfgNumSlots		

dpSlotStateRecvTelegram					
Bit pattern of this slot received by the master in the form of a hexadecimal string					
-	0			String[198]	r
Multi-line: yes	Slot number (PROFINET: Subslot number)		dpSlotCfgNumSlots		

dpSlotStateSendTelegram					
Bit pattern of this slot sent to the slave in the form of a hexadecimal string Transmitted message frame					
-	0			String[198]	r
Multi-line: yes	Slot number (PROFINET: Subslot number)		dpSlotCfgNumSlots		

dpSlotStateTelegramType					
Message frame type of slot 0 = Message frame type unknown					
-	0	0	UINT16_MAX	UWord	r
Multi-line: yes	Slot number (PROFINET: Subslot number)		dpSlotCfgNumSlots		

dpSysCfgAvailable					
This data specifies whether the system has been generated with DP Adapter and/or DP Master 0= Neither DPA nor DPM available 1= DPA available 2= DPM available 3= DPA and DPM available					
-	0	0	3	UWord	r
Multi-line: yes	1		1		

dpSysCfgNumMaster					
Number of masters There is only one master per bus with DP. Since only 1 bus is currently permitted by the bus standard, there can only be a maximum of one master.					
-	0	0	1	Long Integer	r
Multi-line: yes	1		1		

dpSysCfgValid					
This data indicates whether the configuration data are valid and initialized. TRUE or FALSE					
-	0	0	1	Long Integer	r
Multi-line: yes	1		1		

dpSysCfgVersionDpm					
Version number of DP M SW as numerical value					
-	0			Double	r
Multi-line: yes	Number of the master		dpSysCfgNumMaster		

dpSysCfgVersionDpr					
Actual version Dpr (inaccessible in earlier SW)					
-	0			Double	r
Multi-line: yes	Number of the master		dpSysCfgNumMaster		

dpSysCfgVersionDprEx					
DPR_SS_VERSION is a version number stored in the NCK which can be read out via this variable.					
-	0			Double	r
Multi-line: yes	Number of the master		dpSysCfgNumMaster		

dpSysCfgVersionHost					
This data contains the version number of the host SW as a numerical value					
-	0	0	UINT16_MAX	Double	r
Multi-line: yes	Number of the master		dpSysCfgNumMaster		

dpSysStateDpmInit					
There are three different initialization states: REQUEST, ACKNOWLEDGE and ERROR					
-	0			UWord	r
Multi-line: yes	Number of the master		dpSysCfgNumMaster		

errCodeSetNrGen					
Selection of error code set to be used in the case of communication errors. The selection is client-specific, the client is identified by the sender address. 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:		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:		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:		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:		4		

nckCompileSwitches					
Selected NCK compiler switches Bit0: NDEBUG Bit1: NOTRACES Bit2: EMBARGO Bit3: TARGET					
-				UWord	r
Multi-line: yes	1		1		

noOfPersistencyReq					
Number of persistence operations					
-	0	0		Long Integer	r
Multi-line: yes	1: Sum of the individual functions		1		

noOfPersistencyReqFailed					
Number of failed persistence operations					
-	0	0		Long Integer	r
Multi-line: yes	1: Sum of the individual functions		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		Long Integer	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		Long Integer	r
Multi-line: yes	1		1		

pcmciaFfsLength					
Length of FFS on PCMCIA card in bytes					
-	0	0		Long Integer	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		Long Integer	r
Multi-line: yes	1		1		

pcmciaStartShotOffset					
Current access to PCMCIA card: Start offset at beginning of PCMCIA card in bytes					
-	0	0		Long Integer	r
Multi-line: yes	1		1		

persistencyTimeAverage					
Average time for persistence creation					
-	0	0		Double	r
Multi-line: yes	1: Sum of the individual functions		1		

persistencyTimeMaximal					
Maximum time for persistence creation					
-	0	0		Double	r
Multi-line: yes	1: Sum of the individual functions		1		

persistencyTimeMinimal					
Minimum time for persistence creation					
-	0	0		Double	r
Multi-line: yes	1: Sum of the individual functions		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		

totalPersistencyTime					
Summated persistence creation time					
-	0	0		Double	r
Multi-line: yes	1: Sum of the individual functions		1		

1.11.2 Area C, Module 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, Module 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. column: 1000*UserIndex + ListNo., with OEM lists + 10000 <= 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)		

varSpecs					
Do not use this variable any more.					
-		0	maxnumTraceProtData	UWord	wr
Multi-line: yes	1		1		

1.11.4 Area C, Module 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					

countActivatedL					
Number of times the event has occurred.					
-	0			Long Integer	r
Multi-line: yes	Event (see module header)		siehe Bausteinkopf		

dataListIndex					
Index of data list to be used All valid columns in module ETPD - 1)					
-	0	0		UWord	wr
Multi-line: yes	Event (see module header)		siehe Bausteinkopf		

dataProtok					
Number of bytes entered in the Fifo file					
-	0			Long Integer	r
Multi-line: no					

dataUploaded					
Number of bytes already uploaded from the Fifo file					
-	0			Long Integer	r
Multi-line: no					

eventActive					
Event state 0: Not active 1: Active 2: Deactivate and release data set					
-	0	0	2	UWord	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		

headerType					

Type of header in the data record
0: No header
1: Short header with the following structure:
UDword dataStamp; // Data record identified by a consecutive number
UWord event; // Enter type of event that occurred
UWord protCount; // The number of times the event has been logged
2: Long header with the following structure:
UDword dataStamp; // Data record identified by a consecutive number
UWord event; // Enter type of event that occurred
UByte chan; // Channel in which the event occurred
UByte dummy1; // Still free
UDword protCount; // The number of times the event has been logged
UDword dummy2; // Still free
3: Mid-length header, non-aligned with the following structure:
UDword dataStamp; // Data record identified by a consecutive number
UWord event; // Enter type of event that occurred
UByte chan; // Channel in which the event occurred
UByte dummy1; // Still free
UDword protCount; // The number of times the event has been logged					

-	1	0	3	UWord	wr
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 module 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 MMC State data

1.12.1 Area M, Module S: Internal status data MMC

OEM-MMC: Linkitem

/Drive/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					

1.13 User data

1.13.1 Area C, Module GD1: C-GD1

OEM-MMC: Linkitem

/Channel//...

See description of the GUD block in area C

DUMMY					
Undefined					
-				Character	r
Multi-line: no			2		

1.13.2 Area C, Module GD2: C-GD2

OEM-MMC: Linkitem

/Channel//...

See description of the GUD block in area C

DUMMY					
Undefined					
-				Character	r
Multi-line: no			2		

1.13.3 Area C, Module GD3: C-GD3

OEM-MMC: Linkitem

/Channel//...

See description of the GUD block in area C

DUMMY					
Undefined					
-				Character	r
Multi-line: no			2		

1.13.4 Area C, Module GD4: C-GD4

OEM-MMC: Linkitem

/Channel//...

See description of the GUD block in area C

DUMMY					
Undefined					
-				Character	r
Multi-line: no			2		

1.13.5 Area C, Module GD5: C-GD5

OEM-MMC: Linkitem

/Channel//...

See description of the GUD block in area C

DUMMY					
Undefined					
-				Character	r
Multi-line: no			2		

1.13.6 Area C, Module GD6: C-GD6

OEM-MMC: Linkitem

/Channel//...

See description of the GUD block in area C

DUMMY					
Undefined					
-				Character	r
Multi-line: no			2		

1.13.7 Area C, Module GD7: C-GD7

OEM-MMC: Linkitem

/Channel//...

See description of the GUD block in area C

DUMMY					
Undefined					
-				Character	r
Multi-line: no			2		

1.13.8 Area C, Module GD8: C-GD8

OEM-MMC: Linkitem

/Channel//...

See description of the GUD block in area C

DUMMY					
Undefined					
-				Character	r
Multi-line: no			2		

1.13.9 Area C, Module GD9: C-GD9

OEM-MMC: Linkitem

/Channel//...

See description of the GUD block in area C

DUMMY					
Undefined					
-				Character	r
Multi-line: no			2		

1.13.10 Area C, Module GUD: C-GUD

OEM-MMC: Linkitem

/Channel//...

The variables in this block are dynamically generated and deleted in the NCK.
The description and addressing of the existing variables are therefore not specified, and must be taken from the associated ACX file.

The variable is addressed via the column index (SymbolID).

The line index is only relevant for vectors and fields, it is calculated as follows:
Single data: 1
1-dim. fields: 1 + index1
2-dim. fields: 1 + index1 * maxdim2 +
index2
3-dim. fields: 1 + index1 * maxdim2 * maxdim3 +
index2 * maxdim3 +
index3
The formula for 3-dim. fields can generally be applied if the missing dimension index is replaced by 0, and maxdim by 1.

Value ranges:
index1: 0 to (maxdim1-1)
index2: 0 to (maxdim2-1)
index3: 0 to (maxdim3-1)

Depending on the dimensioning of the fields, that is the size of maxdim1, maxdim2 and maxdim3, there is a possibility that the value range of the line index (16 bits) may be inadequate for addressing.
OPI access is not possible if this is the case.

DUMMY					
Undefined					
-				Character	r
Multi-line: no			2		

1.13.11 Area C, Module LUD: C-LUD

OEM-MMC: Linkitem

/Channel//...

See description of the GUD block in area C

DUMMY					
Undefined					
-				Character	r
Multi-line: no			2		

1.13.12 Area N, Module GD1: N-GD1

OEM-MMC: Linkitem

/Nck//...

See description of the GUD block in area C

DUMMY					
Undefined					
-				Character	r
Multi-line: no			2		

1.13.13 Area N, Module GD2: N-GD2

OEM-MMC: Linkitem

/Nck//...

See description of the GUD block in area C

DUMMY					
Undefined					
-				Character	r
Multi-line: no			2		

1.13.14 Area N, Module GD3: N-GD3

OEM-MMC: Linkitem

/Nck//...

See description of the GUD block in area C

DUMMY					
Undefined					
-				Character	r
Multi-line: no			2		

1.13.15 Area N, Module GD4: N-GD4

OEM-MMC: Linkitem

/Nck//...

See description of the GUD block in area C

DUMMY					
Undefined					
-				Character	r
Multi-line: no			2		

1.13.16 Area N, Module GD5: N-GD5

OEM-MMC: Linkitem

/Nck//...

See description of the GUD block in area C

DUMMY					
Undefined					
-				Character	r
Multi-line: no			2		

1.13.17 Area N, Module GD6: N-GD6

OEM-MMC: Linkitem

/Nck//...

See description of the GUD block in area C

DUMMY					
Undefined					
-				Character	r
Multi-line: no			2		

1.13.18 Area N, Module GD7: N-GD7

OEM-MMC: Linkitem

/Nck//...

See description of the GUD block in area C

DUMMY					
Undefined					
-				Character	r
Multi-line: no			2		

1.13.19 Area N, Module GD8: N-GD8

OEM-MMC: Linkitem

/Nck//...

See description of the GUD block in area C

DUMMY					
Undefined					
-				Character	r
Multi-line: no			2		

1.13.20 Area N, Module GD9: N-GD9

OEM-MMC: Linkitem

/Nck//...

See description of the GUD block in area C

DUMMY					
Undefined					
-				Character	r
Multi-line: no			2		

1.13.21 Area N, Module GUD: N-GUD

OEM-MMC: Linkitem

/Nck//...

See description of the GUD block in area C

DUMMY					
Undefined					
-				Character	r
Multi-line: no			2		

1.14 Generic coupling

1.14.1 Area N, Module CP: Generic coupling

OEM-MMC: Linkitem

/Nck/GenericCoupling/...

The CP block contains the status data for the generic coupling.
The status of the axis couplings is structured in an NCK-specific and channel-specific area.

cpCtabExists					
Nonzero if the specified curve table exists					
-	0	0	1	UWord	r
Multi-line: yes	Curve Table ID				

cpCtabId					
Identification number of the nth curve table in the specified memory type.					
-				Long Integer	r
Multi-line: yes	(n * 10) + memType				

cpCtabIdNumLinSegDef					
The number of linear segments that are defined for the specified curve table					
-	0			UWord	r
Multi-line: yes	Curve Table ID				

cpCtabIdNumPolDef					
The number of polynomials that are defined for the specified curve table.					
-	0			UWord	r
Multi-line: yes	Curve Table ID				

cpCtabIdNumPolySegDef					
The number of polynomial segments that are defined for the specified curve table					
-	0			UWord	r
Multi-line: yes	Curve Table ID				

cpCtabIdNumSegDef					
The number of segments that are defined for the specified curve table.					
-	0			UWord	r
Multi-line: yes	Curve Table ID				

cpCtabLocked					
Locking state, value > 0 if curve table is locked					
-		-1	3	Long Integer	r
Multi-line: yes	Curve Table ID				

cpCtabMemType					
Memory type in which the curve table resides					
-		-1	2	Long Integer	r
Multi-line: yes	Curve Table ID				

cpCtabNumDef					
The total number of curve tables that are defined for the specified memory type.					
-	0	0		UWord	r
Multi-line: yes	1=DRAM, 2=SRAM, 3=All memory Types		3		

cpCtabNumFree					
The number of additional curve tables that may be defined in the specified memory type.					
-	0	0		UWord	r
Multi-line: yes	1=DRAM, 2=SRAM, 3=All memory Types		3		

cpCtabNumPolDef					
The total number of curve table polynomials that are defined in the specified memory type.					
-	0			UWord	r
Multi-line: yes	1=DRAM, 2=SRAM, 3=All memory Types		3		

cpCtabNumPolFree					
The number of additional curve table polynomials that may be defined in the specified memory type.					
-	0			UWord	r
Multi-line: yes	1=DRAM, 2=SRAM, 3=All memory Types		3		

cpCtabNumPolMax					
The maximum number of curve table polynomials that are allowed in the specified memory type.					
-	0			UWord	r
Multi-line: yes	1=DRAM, 2=SRAM, 3=All memory Types		3		

cpCtabNumSegDef					
The total number of curve table segments, of the specified segment type, that are defined in the specified memory type.					
-	0			UWord	r
Multi-line: yes	(segmentType * 10) + memType		23		

cpCtabNumSegFree					
The number of additional curve table segments, of the specified segment type, that may be defined in the specified memory type.					
-	0			UWord	r
Multi-line: yes	(segmentType * 10) + memType		23		

cpCtabNumSegMax					
The maximum number of curve table segments, of the specified segment type, that are allowed in the specified memory type.					
-	0			UWord	r
Multi-line: yes	(segmentType * 10) + memType		23		

cpCtabPeriodic					
Periodicity, value > 0 if curve table is periodic					
-		-1	2	Long Integer	r
Multi-line: yes	Curve Table ID				

1.14.2 Area C, Module CP:

OEM-MMC: Linkitem

/Channel/GenericCoupling/...

aaCpActFa	$AA_CPACTFA[a,b]				
The axis index of the nth leading axis which is active for the stated following axis, LAx. -1 = the stated coupling is not defined					
-	-1	-1		UWord	r
Multi-line: yes	(n*256) * axis index of the following axis				

aaCpActLa	$AA_CPACTLA[a,b]				
The axis index of the nth leading axis which is active for the stated following axis, FAx. -1 = the stated coupling is not defined					
-	-1	-1		UWord	r
Multi-line: yes	(n*256) * axis index of the following axis				

aaCpBlockChg	$AA_CPBC[a]				
The block change criterion indicates the condition that has to be fulfilled before one can continue with the next block of the NC program if a coupling has been activated for the stated following axis, FAx NONE - Block change is performed immediately FINE - Block change is performed with "Synchronism fine" COARSE - Block change is performed with "Synchronism coarse" IPOSTOP - Block change is performed with "Setpoint synchronism"					
-				String[32]	r
Multi-line: yes	Axis index of the following axis		numMachAxes		

aaCpDefLa	$AA_CPDEFLA[a,b]				
The axis index of the nth leading axis which has been defined for the stated following axis, FAx. -1 = the stated coupling is not defined					
-	-1	-1		UWord	r
Multi-line: yes	(n*256) * axis index of the following axis				

aaCpMReset	$AA_CPMRESET[a]				
Coupling mode through RESET ON OFF DEL					
-				String[32]	r
Multi-line: yes	Axis index of the following axis		numMachAxes		

aaCpMStart	$AA_CPMSTART[a]				
Coupling mode through program start ON OFF DEL					
-				String[32]	r
Multi-line: yes	Axis index of the following axis		numMachAxes		

aaCpMStartPrt	$AA_CPMSTARTPRT[a]				
Coupling mode through SERUPRO start ON OFF DEL					
-				String[32]	r
Multi-line: yes	Axis index of the following axis		numMachAxes		

aaCpNumActFa	$AA_CPNACTFA[a]				
The number of leading axes which are active for the stated following axis, LAx.					
-	0	0		UWord	r
Multi-line: yes	Axis index of the following axis		numMachAxes		

aaCpNumActLa	$AA_CPNACTLA[a]				
The number of leading axes which are active for the stated following axis, FAx.					
-	0	0		UWord	r
Multi-line: yes	Axis index of the following axis		numMachAxes		

aaCpNumDefLa	$AA_CPNDEFLA[a]				
The number of leading axes which have been defined for the stated following axis, FAx.					
-	0	0		UWord	r
Multi-line: yes	Axis index of the following axis		numMachAxes		

aaCpSetType	$AA_CPSETTYPE[a]				
Coupling set coupling type CNONE TRAIL LEAD EG COUP					
-				String[32]	r
Multi-line: yes	Axis index of the following axis		numMachAxes		

aaCpSynCoPos	$AA_CPSYNCOP[a]				
Coarse positioning tolerance for coupling synchronization					
-				Double	r
Multi-line: yes	Axis index of the following axis		numMachAxes		

aaCpSynCoVel	$AA_CPSYNCOV[a]				
Coarse velocity tolerance for coupling synchronization					
-				Double	r
Multi-line: yes	Axis index of the following axis		numMachAxes		

aaCpSynFiPos	$AA_CPSYNFIP[a]				
Fine positioning tolerance for coupling synchronization					
-				Double	r
Multi-line: yes	Axis index of the following axis		numMachAxes		

aaCpSynFiVel	$AA_CPSYNFIV[a]				
Fine velocity tolerance for coupling synchronization					
-				Double	r
Multi-line: yes	Axis index of the following axis		numMachAxes		

aaCpfAccelTotal	$AA_CPFACCT[a]				
Proportion of axis acceleration due to the coupled axes. The sum of the dependent proportion of the acceleration of all leading axes for the stated following axis, FAx.					
-				Double	r
Multi-line: yes	Axis index of the following axis		numMachAxes		

aaCpfActive	$AA_CPFACT[a]				
Bit-coded for identifying all types of coupling which are active for the stated following axis, FAx 0 = NONE - No active coupling to FAx Bit 0 (0x0001) - TRAIL - Uses a coupling factor Bit 1 (0x0002) - LEAD - Uses a curve table Bit 2 (0x0004) - ELG - An electronic gearbox link Bit 3 (0x0008) - Reserved Bit 4 (0x0010) - COUP - Spindle/partial spindle coupling Bit 5 (0x0020) - GANTRY - Coupling of the split axes (axes mechanically bound) Bit 6 (0x0040) - TANG - Tangential coupling with the aid of a curve table Bit 7 (0x0080) - GEN_CP - Generic coupling					
-				UWord	r
Multi-line: yes	Axis index of the following axis		numMachAxes		

aaCpfCmdPosTotal	$AA_CPFCMDPT[a]				
Proportion of the axis position command due to the coupled axes. The sum of the dependent proportion of the position command for all leading axes for the stated following axis, FAx.					
-				Double	r
Multi-line: yes	Axis index of the following axis		numMachAxes		

aaCpfCmdVelTotal	$AA_CPFCMDVT[a]				
Proportion of the axis position command due to the coupled axes. The sum of the dependent proportion of the velocity command for all leading axes for the stated following axis, FAx.					
-				Double	r
Multi-line: yes	Axis index of the following axis		numMachAxes		

aaCpfMSOn	$AA_CPFMSON[a]				
Indicates the activation strategy of the following axis CNONE CFAST COARSE NTG ACN ACP DCT NTGP DCP					
-				String[32]	r
Multi-line: yes	Axis index of the following axis		numMachAxes		

aaCpfModeOff	$AA_CPFMOF[a]				
Indicates the behavior of the following axis, FAx, when the coupling is deactivated. STOP - Stop following axis/spindle CON - Continue motion of the FAx with the current velocity ADD					
-				String[32]	r
Multi-line: yes	Axis index of the following axis		numMachAxes		

aaCpfModeOn	$AA_CPFMON[a]				
Indicates the behavior of the following axis, FAx, when the coupling is activated. STOP - Stop following axis/spindle CON - Continue motion of the FAx with the current velocity ADD					
-				String[32]	r
Multi-line: yes	Axis index of the following axis		numMachAxes		

aaCpfRS	$AA_CPFRS[a]				
The reference system indicates the point at which the coupling process is applied BCS - Basic coordinate system MCS - Machine coordinate system					
-				String[32]	r
Multi-line: yes	Axis index of the following axis		numMachAxes		

aaCpfReqVelocity	$AA_CPFREQV[a]				
The required velocity transferred to all active leading axes as a percentage of the maximum velocity of the stated following axis, FAx.					
-				Double	r
Multi-line: yes	Axis index of the following axis		numMachAxes		

aaCplAccel	$AA_CPLACC[a,b]				
The proportion of the axis acceleration assigned to the stated coupling.					
-				Double	r
Multi-line: yes	(Axis index of the slave axis) * numMachAxes + (axis index of the master axis)		numMachAxes * numMachAxes		

aaCplCTabId	$AA_CPLCTID[a,b]				
ID number of the curve table which is used with the coupling of the stated axes.					
-				UWord	r
Multi-line: yes	(Axis index of the slave axis) * numMachAxes + (axis index of the master axis)		numMachAxes * numMachAxes		

aaCplCmdPos	$AA_CPLCMDP[a,b]				
The proportion of the axis position command assigned to the stated coupling.					
-				Double	r
Multi-line: yes	(Axis index of the slave axis) * numMachAxes + (axis index of the master axis)		numMachAxes * numMachAxes		

aaCplCmdVel	$AA_CPLCMDV[a,b]				
The proportion of the axis acceration command assigned to the stated coupling.					
-				Double	r
Multi-line: yes	(Axis index of the slave axis) * numMachAxes + (axis index of the master axis)		numMachAxes * numMachAxes		

aaCplDenominator	$AA_CPLDEN[a,b]				
Denominator of coupling factor					
-				Double	r
Multi-line: yes	(Axis index of the slave axis) * numMachAxes + (axis index of the master axis)		numMachAxes * numMachAxes		

aaCplInScale	$AA_CPLINSC[a,b]				
Input scaling factor of coupling factor					
-				Double	r
Multi-line: yes	(Axis index of the slave axis) * numMachAxes + (axis index of the master axis)		numMachAxes * numMachAxes		

aaCplInTrans	$AA_CPLINTR[a,b]				
Input transmission correction of coupling factor					
-				Double	r
Multi-line: yes	(Axis index of the slave axis) * numMachAxes + (axis index of the master axis)		numMachAxes * numMachAxes		

aaCplNumerator	$AA_CPLNUM[a,b]				
Counter of coupling factor					
-				Double	r
Multi-line: yes	(Axis index of the slave axis) * numMachAxes + (axis index of the master axis)		numMachAxes * numMachAxes		

aaCplOutScale	$AA_CPLOUTSC[a,b]				
Output scaling factor of coupling factor					
-				Double	r
Multi-line: yes	(Axis index of the slave axis) * numMachAxes + (axis index of the master axis)		numMachAxes * numMachAxes		

aaCplOutTrans	$AA_CPLOUTTR[a,b]				
Output transmission correction of coupling factor					
-				Double	r
Multi-line: yes	(Axis index of the slave axis) * numMachAxes + (axis index of the master axis)		numMachAxes * numMachAxes		

aaCplRS	$AA_CPLRS[a,b]				
Reference system for the stated coupling Description Value range: BCS - Basic coordinate system MCS - Machine coordinate system					
-				String[32]	r
Multi-line: yes	(Axis index of the slave axis) * numMachAxes + (axis index of the master axis)		numMachAxes * numMachAxes		

aaCplSetVal	$AA_CPLSETVAL[a,b]				
Indicates the type of defined value used for the coupling ACTPOS = Actual position CMDPOS = Setpoint position CMDVEL = Setpoint velocity					
-				String[32]	r
Multi-line: yes	(Axis index of the slave axis) * numMachAxes + (axis index of the master axis)		numMachAxes * numMachAxes		

aaCplState	$AA_CPLSTATE[a,b]				
A string which describes the actual status of the coupling DEF = Defined (but not yet activated) ON = Active OF = Deactivated					
-				String[32]	r
Multi-line: yes	(Axis index of the slave axis) * numMachAxes + (axis index of the master axis)		numMachAxes * numMachAxes		

aaCplType	$AA_CPLTYPE[a,b]				
Indicates the process that is used with the coupling of the stated following axis with the stated leading axis. 0 = NONE - No defined coupling with these axes Bit 0 (0x0001) - TRAIL - Uses a coupling factor Bit 1 (0x0002) - LEAD - Uses a curve table Bit 2 (0x0004) - ELG - An electronic gearbox link Bit 3 (0x0008) - Reserved Bit 4 (0x0010) - COUP - Spindle/partial spindle coupling Bit 5 (0x0020) - GANTRY - Coupling of the split axes (axes mechanically bound) Bit 6 (0x0040) - TANG - Tangential coupling with the aid of a curve table Bit 7 (0x0080) - GEN_CP - Generic coupling					
-				UWord	r
Multi-line: yes	(Axis index of the slave axis) * numMachAxes + (axis index of the master axis)		numMachAxes * numMachAxes		

1.14.3 Area C, Module WAL: Working area limitation

OEM-MMC: Linkitem

/Channel/CoordSysWorkAreaLimits/...

Working area limitation data

waCSCoordSys	$P_WORKAREA_CS_COORD_SYSTEM				
Coordinate system for working area limitation Identifier for the coordinate system in which the working area limitation is to apply. The following are valid: 0: Working area limitation in the WCS 3: Working area limitation in the SZS Special feature of line addressing: Any channel axis index can be selected as the channel axis index. Within a working area limitation group, the values are identical.					
-	0	0	3	UWord	wr
Multi-line: yes	Channel axis index + working area limitation group * numMachAxes		numMachAxes * $MC_MM_NUM_WORKAREA_CS_GROUPS		

waCSLimitMinus	$P_WORKAREA_CS_LIMIT_MINUS				
Position of the coordinate system-specific working area limitation in the negative direction for the addressed axis and working area group.					
-				Double	wr
Multi-line: yes	Channel axis index + working area limitation group * numMachAxes		numMachAxes * $MC_MM_NUM_WORKAREA_CS_GROUPS		

waCSLimitPlus	$P_WORKAREA_CS_LIMIT_PLUS				
Position of the coordinate system-specific working area limitation in the positive direction for the addressed axis and working area group.					
-				Double	wr
Multi-line: yes	Channel axis index + working area limitation group * numMachAxes		numMachAxes * $MC_MM_NUM_WORKAREA_CS_GROUPS		

waCSMinusEnable	$P_WORKAREA_CS_MINUS_ENABLE				
Coord.-specific working area limitation, negative valid TRUE: The limitation of waCSLimitMinus is valid.					
-	0	0	1	UWord	wr
Multi-line: yes	Channel axis index + working area limitation group * numMachAxes		numMachAxes * $MC_MM_NUM_WORKAREA_CS_GROUPS		

waCSPlusEnable	$P_WORKAREA_CS_PLUS_ENABLE				
Coord.-specific working area limitation, positive valid TRUE: The limitation of waCSLimitPlus is valid.					
-	0	0	1	UWord	wr
Multi-line: yes	Channel axis index + working area limitation group * numMachAxes		numMachAxes * $MC_MM_NUM_WORKAREA_CS_GROUPS		

1.14.4 Area N, Module VSYN:

OEM-MMC: Linkitem

/Nck/SelectedFunctionData/...

anMarker					
Dummy					
-				UWord	r
Multi-line: yes	1		1		

dummy1					
dummy					
-				Double	r
Multi-line: yes	1		1		

1.14.5 Area T, Module TDC:

OEM-MMC: Linkitem

/Tool/Tools/...

toolDataChangeInfo					
Siemens application tool parameter					
-				UWord	r
Multi-line: no					

1.14.6 Area T, Module TISO:

OEM-MMC: Linkitem

/Tool/IsoHDCompensation/...

isoCorrParam	$TC_ISO_???,				
This variable contains the offset values for the ISO2.2 and ISO3.2 modes. The column index contains the offset number.					
mm, inch, user defined	0			Double	wr
Multi-line: yes	1: Offset value for the geometry of the tool length in ISO2 mode.($TC_ISO_H)		13		

■

2

2 Interface Signals

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 3.3.

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 several signals.
This reference specifies the section number or the short designation of the description of functions, please refer to

References:

/xxx/ or /xx-yy/

xxx,	Short designation, (e. g.: /FB1/)
xx	Short designation of individual description of functions (e. g.: /Z1/)
yyy	Name of the chapter (e. g.: -A1)

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)

Note

Please refere also to
SINAMICS S120 Installation and Start-UP Manual
SINAMICS S List Manual for SINAMICS drives.

2.2.1 Signals from/to machine control panel, M version

Signals from machine control panel (keys) Input display								
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	Reset	Key-switch position 1	Single block	Feedrate override 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	R11	9th axis 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) Output display, 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 key - R13	Axis selection				Single block	Spindle start	*Spindle stop
		X R1	4th axis R4	7th axis R7	R10			
QB n + 3	Axis selection							Direction key + R15
	Z R3	5th axis R5	Travel command MCS/WCS R12	R11	9th axis R9	8th axis R8	6th axis R6	
QB n + 4	Unassigned customer keys							Y R2
	T9	T10	T11	T12	T13	T14	T15	
QB n + 5	Unassigned customer keys							
	T1	T2	T3	T4	T5	T6	T7	T8

2.2.2 Signals from/to machine control panel, T version

Signals from machine control panel (keys) Input display								
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	Reset	Keyswitch position 1	Single block	Feed override				
				E	D	C	B	A
IB n + 4	Direction keys			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) Output display, 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				MCS/WCS	Unassigned customer keys		
	T9	T10	T11	T12		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					Unassigned customer keys		
	T9	T10	T11	T12	MCS/WCS	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, HT 2)

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							
								Identifier HT 2
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		Rapid traverse/feed override switch					
	Digital display	Keyswitch	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 following Documentation:

References: Equipment Manual: Operator Components Manual Networking
Function Manual, Basic Functions: Basic PLC Program (P3)

2.2.5 Signals from/to handheld programming unit (HT 8)

Signals from machine control panel simulation Interface HT8 → 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							
	Cntr Panel Func (CPF)	{U4}	{U3}	BigFct	{U2}	{U1}	(INC)	(REPOS)
IB n + 2		Change over axes	Travel keys (JOG) positive direction					
			Ax6	Ax5	Ax4	Ax3	Ax2	Ax1
IB n + 3			Travel keys (JOG) negative direction					
			Ax6	Ax5	Ax4	Ax3	Ax2	Ax1
IB n + 4								
	{U16} [BF8]	{U15} [BF7]	{U14} [BF6]	{U13} [BF5]	{U12} [BF4]	{U11} [BF3]	{U10} [BF2]	{U9} [BF1]
IB n + 5								
	[BF16]	{U8} [BF15]	{U7} [BF14]	{U6} [BF13]	{U5} [BF12]	(SBL) [BF10]	[BF10]	[BF9]
IB n + 6	Start key block							
	Reserved	HT 8	SF4 [VAL+]	SF3 [VAL-]	SF2	SF1	Start	Stop
IB n + 7	Feedrate override							
				E	D	C	B	A

Signals to machine control panel simulation Interface PLC → HT 8								
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 changed over	Travel keys (JOG) positive direction					
QB n + 2								
			Ax6	Ax5	Ax4	Ax3	Ax2	Ax1
QB n + 3	Travel keys (JOG) negative direction							
	For WCS: No MCS Ax4 to Ax6		Ax6	Ax5	Ax4	Ax3	Ax2	Ax1
QB n + 4								
QB n + 5								
QB n + 6	Start key block							
	Display travel keys		VAL+	VAL-	SF2	SF1	Start	Stop
QB n + 7								

2.3 PLC alarms / messages

The message/alarm numbers, text and area IDs are assigned in DB 2. When setting the relevant parameters, the alarm resp. message bits are automatically transferred to the user interface (channel, axis/spindle). If this is not the case, the transfer has to be programmed in the user program. The user interface can also be influenced after calling up the block for error/operational messages. Only signals from channels and axes which have been declared in the NC machine data can be transferred, only texts can be displayed.
All error messages have to be acknowledged by the user after being generated.
Operational messages are only displayed while the condition is pending.
The number of user areas (maximum 32) can be parameterized via FB1.

2.3.1 FC 10 alarms in DB 2 (FB1-Parameter "ExtendAlMsg"=False)

Channel areas in DB 2 (Parameter "ExtendAlMsg"=False)

DB 2	Signals for PLC messages (PLC → HMI)							
Byte (Message type)	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
	Channel 1							
0 (EM)	510007	510006	510005	510004	510003	510002	510001	510000
	Feed disable (alarm no.: 510000-510015)							
1 (OM)	510015	510014	510013	510012	510011	510010	510009	510008
2 (EM)	Feed and read-in disable byte 1 (alarm no.: 510100-510107)							
3 (EM)	Feed and read-in disable byte 2 (alarm no.: 510108-510115)							
4 (OM)	Feed and read-in disable byte 3 (alarm no.: 510116-510123)							
5 (OM)	Feed and read-in disable byte 4 (alarm no.: 510124-510131)							
6 (EM)	Read-in disable byte 1 (alarm no.: 510200-510207)							
7 (EM)	Read-in disable byte 2 (alarm no.: 510208-510215)							
8 (OM)	Read-in disable byte 3 (alarm no.: 510216-510223)							
9 (OM)	Read-in disable byte 4 (alarm no.: 510224-510231)							
10 (EM)	NC Start disable byte 1 (alarm no.: 510300-510307)							
11 (OM)	NC Start disable byte 2 (alarm no.: 510308-510315)							
12 (EM)	Feed stop GEOaxis 1 byte 1 (alarm no.: 511100-511107)							
13 (OM)	Feed stop GEOaxis 1 byte 2 (alarm no.: 511108-511115)							
14 (EM)	Feed stop GEOaxis 2 byte 1 (alarm no.: 511200-511207)							
15 (OM)	Feed stop GEOaxis 2 byte 2 (alarm no.: 511208-511215)							
16 (EM)	Feed stop GEOaxis 3 byte 1 (alarm no.: 511300-511307)							
17 (OM)	Feed stop GEOaxis 3 byte 2 (alarm no.: 511308-511315)							

Byte (Message type)	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
	Channel 2							
18 (EM)	520007	520006	520005	520004	520003	520002	520001	520000
	Feed disable (alarm no.: 520000-520015)							
19 (OM)	520015	520014	520013	520012	520011	520010	520009	520008
20 (EM)	Feed and read-in disable byte 1 (alarm no.: 520100-520107)							
21 (EM)	Feed and read-in disable byte 2 (alarm no.: 520108-520115)							
22 (OM)	Feed and read-in disable byte 3 (alarm no.: 520116-520123)							
23 (OM)	Feed and read-in disable byte 4 (alarm no.: 520124-520131)							
24 (EM)	Read-in disable byte 1 (alarm no.: 520200-520207)							
25 (EM)	Read-in disable byte 2 (alarm no.: 520208-520215)							
26 (OM)	Read-in disable byte 3 (alarm no.: 520216-520223)							
27 (OM)	Read-in disable byte 4 (alarm no.: 520224-520231)							
28 (EM)	NC Start disable byte 1 (alarm no.: 520300-520307)							
29 (OM)	NC Start disable byte 2 (alarm no.: 520308-520315)							
30 (EM)	Feed stop GEOaxis 1 byte 1 (alarm no.: 521100-521107)							
31 (OM)	Feed stop GEOaxis 1 byte 2 (alarm no.: 521108-521115)							
32 (EM)	Feed stop GEOaxis 2 byte 1 (alarm no.: 521200-521207)							
33 (OM)	Feed stop GEOaxis 2 byte 2 (alarm no.: 521208-521215)							
34 (EM)	Feed stop GEOaxis 3 byte 1 (alarm no.: 521300-521307)							
35 (OM)	Feed stop GEOaxis 3 byte 2 (alarm no.: 521308-521315)							

Byte (Message type)	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
	Channel 3							
36 (EM)	530007	530006	530005	530004	530003	530002	530001	530000
	Feed disable (alarm no.: 530000-530015)							
37 (OM)	530015	530014	530013	530012	530011	530010	530009	530008
38 (EM)	Feed and read-in disable byte 1 (alarm no.: 530100-530107)							
39 (EM)	Feed and read-in disable byte 2 (alarm no.: 530108-530115)							
40 (OM)	Feed and read-in disable byte 3 (alarm no.: 530116-530123)							
41 (OM)	Feed and read-in disable byte 4 (alarm no.: 530124-530131)							
42 (EM)	Read-in disable byte 1 (alarm no.: 530200-530207)							
43 (EM)	Read-in disable byte 2 (alarm no.: 530208-530215)							
44 (OM)	Read-in disable byte 3 (alarm no.: 530216-530223)							
45 (OM)	Read-in disable byte 4 (alarm no.: 530224-530231)							
46 (EM)	NC Start disable byte 1 (alarm no.: 530300-530307)							
47 (OM)	NC Start disable byte 2 (alarm no.: 530308-530315)							

Byte (Message type)	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
48 (EM)	Feed stop GEOaxis 1 byte 1 (alarm no.: 531100-531107)							
49 (OM)	Feed stop GEOaxis 1 byte 2 (alarm no.: 531108-531115)							
50 (EM)	Feed stop GEOaxis 2 byte 1 (alarm no.: 531200-531207)							
51 (OM)	Feed stop GEOaxis 2 byte 2 (alarm no.: 531208-531215)							
52 (EM)	Feed stop GEOaxis 3 byte 1 (alarm no.: 531300-531307)							
53 (OM)	Feed stop GEOaxis 3 byte 2 (alarm no.: 531308-531315)							

Byte (Message type)	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
	Channel 4							
54 (EM)	540007	540006	540005	540004	540003	540002	540001	540000
	Feed disable (alarm no.: 540000-540015)							
55 (OM)	540015	540014	540013	540012	540011	540010	540009	540008
56 (EM)	Feed and read-in disable byte 1 (alarm no.: 540100-540107)							
57 (EM)	Feed and read-in disable byte 2 (alarm no.: 540108-540115)							
58 (OM)	Feed and read-in disable byte 3 (alarm no.: 540116-540123)							
59 (OM)	Feed and read-in disable byte 4 (alarm no.: 540124-540131)							
60 (EM)	Read-in disable byte 1 (alarm no.: 540200-540207)							
61 (EM)	Read-in disable byte 2 (alarm no.: 540208-540215)							
62 (OM)	Read-in disable byte 3 (alarm no.: 540216-540223)							
63 (OM)	Read-in disable byte 4 (alarm no.: 540224-540231)							
64 (EM)	NC Start disable byte 1 (alarm no.: 540300-540307)							
65 (EM)	NC Start disable byte 2 (alarm no.: 540308-540315)							
66 (EM)	Feed stop GEOaxis 1 byte 1 (alarm no.: 541100-541107)							
67 (OM)	Feed stop GEOaxis 1 byte 2 (alarm no.: 541108-541115)							
68 (EM)	Feed stop GEOaxis 2 byte 1 (alarm no.: 541200-541207)							
69 (OM)	Feed stop GEOaxis 2 byte 2 (alarm no.: 541208-541215)							
70 (EM)	Feed stop GEOaxis 3 byte 1 (alarm no.: 541300-541307)							
71 (OM)	Feed stop GEOaxis 3 byte 2 (alarm no.: 541308-541315)							

Byte (Message type)	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
	Channel 5							
72 (EM)	550007	550006	550005	550004	550003	550002	550001	550000
	Feed disable (alarm no.: 550000-550015)							
73 (OM)	550015	550014	550013	550012	550011	550010	550009	550008
74 (EM)	Feed and read-in disable byte 1 (alarm no.: 550100-550107)							
75 /BM)	Feed and read-in disable byte 2 (alarm no.: 550108-550115)							
76 (OM)	Feed and read-in disable byte 3 (alarm no.: 550116-550123)							
77 (OM)	Feed and read-in disable byte 4 (alarm no.: 550124-550131)							
78 (EM)	Read-in disable byte 1 (alarm no.: 550200-550207)							
79 (EM)	Read-in disable byte 2 (alarm no.: 550208-550315)							
80 (OM)	Read-in disable byte 3 (alarm no.: 550216-550223)							
81 (OM)	Read-in disable byte 4 (alarm no.: 550224-550231)							
82 (EM)	NC Start disable byte 1 (alarm no.: 550300-550307)							
83 (OM)	NC Start disable byte 2 (alarm no.: 550308-550315)							
84 (EM)	Feed stop GEOaxis 1 byte 1 (alarm no.: 551100-551107)							
85 (OM)	Feed stop GEOaxis 1 byte 2 (alarm no.: 551108-551115)							
86 (EM)	Feed stop GEOaxis 2 byte 1 (alarm no.: 551200-551207)							
87 (OM)	Feed stop GEOaxis 2 byte 2 (alarm no.: 551208-551215)							
88 (EM)	Feed stop GEOaxis 3 byte 1 (alarm no.: 551300-551307)							
89 (OM)	Feed stop GEOaxis 3 byte 2 (alarm no.: 551308-551315)							

Byte (Message type)	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
	Channel 6							
90 (EM)	560007	560006	560005	560004	560003	560002	560001	560000
	Feed disable (alarm no.: 560000-560015)							
91 (OM)	560015	560014	560013	560012	560011	560010	560009	560008
92 (EM)	Feed and read-in disable byte 1 (alarm no.: 560100-560107)							
93 (EM)	Feed and read-in disable byte 2 (alarm no.: 560108-560115)							
94 (OM)	Feed and read-in disable byte 3 (alarm no.: 560116-560123)							
95 (OM)	Feed and read-in disable byte 4 (alarm no.: 560124-560131)							
96 (EM)	Read-in disable byte 1 (alarm no.: 560200-560207)							
97 (EM)	Read-in disable byte 2 (alarm no.: 560208-560315)							
98 (OM)	Read-in disable byte 3 (alarm no.: 560216-560223)							
99 (OM)	Read-in disable byte 4 (alarm no.: 560224-560231)							
100 (EM)	NC Start disable byte 1 (alarm no.: 560300-560307)							

Byte (Message type)	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
101 (OM)	NC Start disable byte 2 (alarm no.: 560308-560315)							
102 (EM)	Feed stop GEOaxis 1 byte 1 (alarm no.: 561100-561107)							
103 (OM)	Feed stop GEOaxis 1 byte 2 (alarm no.: 561108-561115)							
104 (EM)	Feed stop GEOaxis 2 byte 1 (alarm no.: 561200-561207)							
105 (OM)	Feed stop GEOaxis 2 byte 2 (alarm no.: 561208-561215)							
106 (EM)	Feed stop GEOaxis 3 byte 1 (alarm no.: 561300-561307)							
107 (OM)	Feed stop GEOaxis 3 byte 2 (alarm no.: 561308-561315)							

Byte (Message type)	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
	Channel 7							
108 (EM)	570007	570006	570005	570004	570003	570002	570001	570000
	Feed disable (alarm no.: 570000-570015)							
109 (OM)	570015	570014	570013	570012	570011	570010	570009	570008
110 (EM)	Feed and read-in disable byte 1 (alarm no.: 570100-570107)							
111 (EM)	Feed and read-in disable byte 2 (alarm no.: 570108-570115)							
112 (OM)	Feed and read-in disable byte 3 (alarm no.: 570116-570123)							
113 (OM)	Feed and read-in disable byte 4 (alarm no.: 570124-570131)							
114 (EM)	Read-in disable byte 1 (alarm no.: 570200-570207)							
115 (EM)	Read-in disable byte 2 (alarm no.: 570208-570315)							
116 (OM)	Read-in disable byte 3 (alarm no.: 570216-570223)							
117 (OM)	Read-in disable byte 4 (alarm no.: 570224-570231)							
118 (EM)	NC Start disable byte 1 (alarm no.: 570300-570307)							
119 (OM)	NC Start disable byte 2 (alarm no.: 570308-570315)							
120 (EM)	Feed stop GEOaxis 1 byte 1 (alarm no.: 571100-571107)							
121 (OM)	Feed stop GEOaxis 1 byte 2 (alarm no.: 571108-571115)							
122 (EM)	Feed stop GEOaxis 2 byte 1 (alarm no.: 571200-571207)							
123 (OM)	Feed stop GEOaxis 2 byte 2 (alarm no.: 571208-571215)							
124 (EM)	Feed stop GEOaxis 3 byte 1 (alarm no.: 571300-571307)							
125 (OM)	Feed stop GEOaxis 3 byte 2 (alarm no.: 571308-571315)							

Byte (Message type)	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
	Channel 8							
126 (EM)	580007	580006	580005	580004	580003	580002	580001	580000
	Feed disable (alarm no.: 580000-580015)							
127 (OM)	580015	580014	580013	580012	580011	580010	580009	580008
128 (EM)	Feed and read-in disable byte 1 (alarm no.: 580100-580107)							
129 (EM)	Feed and read-in disable byte 2 (alarm no.: 580108-580115)							
130 (OM)	Feed and read-in disable byte 3 (alarm no.: 580116-580123)							
131 (OM)	Feed and read-in disable byte 4 (alarm no.: 580124-580131)							
132 (EM)	Read-in disable byte 1 (alarm no.: 580200-580207)							
133 (EM)	Read-in disable byte 2 (alarm no.: 580208-580315)							
134 (OM)	Read-in disable byte 3 (alarm no.: 580216-580223)							
135 (OM)	Read-in disable byte 4 (alarm no.: 580224-580231)							
136 (EM)	NC Start disable byte 1 (alarm no.: 580300-580307)							
137 (OM)	NC Start disable byte 2 (alarm no.: 580308-580315)							
138 (EM)	Feed stop GEOaxis 1 byte 1 (alarm no.: 581100-581107)							
139 (OM)	Feed stop GEOaxis 1 byte 2 (alarm no.: 581108-581115)							
140 (EM)	Feed stop GEOaxis 2 byte 1 (alarm no.: 581200-581207)							
141 (OM)	Feed stop GEOaxis 2 byte 2 (alarm no.: 581208-581215)							
142 (EM)	Feed stop GEOaxis 3 byte 1 (alarm no.: 581300-581307)							
143 (OM)	Feed stop GEOaxis 3 byte 2 (alarm no.: 581308-581315)							

	Channel 9 and 10 unrealized

Axis areas in DB 2 (Parameter "ExtendAlMsg"=False)

Byte (Message type)	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
	axis/spindle							
144 (EM)	600107	600106	600105	600104	600103	600102	600101	600100
	Feed stop/spindle stop for axis/spindle 1 (alarm no.: 600100-600115)							
145 (OM)	600115	600114	600113	600112	600111	600110	600109	600108
146 (EM)	Feed stop/spindle stop for axis/spindle 2 byte 1 (alarm no.: 600200-600207)							
147 (OM)	Feed stop/spindle stop for axis/spindle 2 byte 2 (alarm no.: 600208-600215)							
148 (EM)	Feed stop/spindle stop for axis/spindle 3 byte 1 (alarm no.: 600300-600307)							
149 (OM)	Feed stop/spindle stop for axis/spindle 3 byte 2 (alarm no.: 600308-600315)							
150 (EM)	Feed stop/spindle stop for axis/spindle 4 byte 1 (alarm no.: 600400-600407)							

Byte (Message type)	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
151 (OM)	Feed stop/spindle stop for axis/spindle 4 byte 2 (alarm no.: 600408-600415)							
152 (EM)	Feed stop/spindle stop for axis/spindle 5 byte 1 (alarm no.: 600500-600507)							
153 (OM)	Feed stop/spindle stop for axis/spindle 5 byte 2 (alarm no.: 600508-600515)							
154 (EM)	Feed stop/spindle stop for axis/spindle 6 byte 1 (alarm no.: 600600-600607)							
155 (OM)	Feed stop/spindle stop for axis/spindle 6 byte 2 (alarm no.: 600608-600615)							
156 (EM)	Feed stop/spindle stop for axis/spindle 7 byte 1 (alarm no.: 600700-600707)							
157 (OM)	Feed stop/spindle stop for axis/spindle 7 byte 2 (alarm no.: 600708-600715)							
158 (EM)	Feed stop/spindle stop for axis/spindle 8 byte 1 (alarm no.: 600800-600807)							
159 (OM)	Feed stop/spindle stop for axis/spindle 8 byte 2 (alarm no.: 600808-600815)							
160 (EM)	Feed stop/spindle stop for axis/spindle 9 byte 1 (alarm no.: 600900-600907)							
161 (OM)	Feed stop/spindle stop for axis/spindle 9 byte 2 (alarm no.: 600908-600915)							
162 (EM)	Feed stop/spindle stop for axis/spindle 10 byte 1 (alarm no.: 601000-601007)							
163 (OM)	Feed stop/spindle stop for axis/spindle 10 byte 2 (alarm no.: 601008-601015)							
164 (EM)	Feed stop/spindle stop for axis/spindle 11 byte 1 (alarm no.: 601100-601107)							
165 (OM)	Feed stop/spindle stop for axis/spindle 11 byte 2 (alarm no.: 601108-601115)							
166 (EM)	Feed stop/spindle stop for axis/spindle 12 byte 1 (alarm no.: 601200-601207)							
167 (OM)	Feed stop/spindle stop for axis/spindle 12 byte 2 (alarm no.: 601208-601215)							
168 (EM)	Feed stop/spindle stop for axis/spindle 13 byte 1 (alarm no.: 601300-601307)							
169 (OM)	Feed stop/spindle stop for axis/spindle 13 byte 2 (alarm no.: 601308-601315)							
170 (EM)	Feed stop/spindle stop for axis/spindle 14 byte 1 (alarm no.: 601400-601407)							
171 (OM)	Feed stop/spindle stop for axis/spindle 14 byte 2 (alarm no.: 601408-601415)							
172 (EM)	Feed stop/spindle stop for axis/spindle 15 byte 1 (alarm no.: 601500-601507)							
173 (OM)	Feed stop/spindle stop for axis/spindle 15 byte 2 (alarm no.: 601508-601515)							
174 (EM)	Feed stop/spindle stop for axis/spindle 16 byte 1 (alarm no.: 601600-601607)							
175 (OM)	Feed stop/spindle stop for axis/spindle 16 byte 2 (alarm no.: 601608-601615)							
176 (EM)	Feed stop/spindle stop for axis/spindle 17 byte 1 (alarm no.: 601700-601707)							
177 (OM)	Feed stop/spindle stop for axis/spindle 17 byte 2 (alarm no.: 601708-601715)							
178 (EM)	Feed stop/spindle stop for axis/spindle 18 byte 1 (alarm no.: 601800-601807)							
179 (OM)	Feed stop/spindle stop for axis/spindle 18 byte 2 (alarm no.: 601808-601815)							
	axis 19 – 31 unrealized							

User areas im DB 2 (Parameter "ExtendAlMsg"=False)

Byte (Message type)	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
180 (EM)	700007	700006	700005	700004	700003	700002	700001	700000
	User area 0 (alarm no.: 700000-700015)							
181 (EM)	700015	700014	700013	700012	700011	700010	700009	700008

Byte (Message type)	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
182 (EM)	User area 0: byte 3 (alarm no.: 700016-700023)							
183 (EM)	User area 0: byte 4 (alarm no.: 700024-700031)							
184 (OM)	User area 0: byte 5 (alarm no.: 700032-700039)							
185 (OM)	User area 0: byte 6 (alarm no.: 700040-700047)							
186 (OM)	User area 0: byte 7 (alarm no.: 700048-700055)							
187 (OM)	User area 0: byte 8 (alarm no.: 700056-700063)							
188 - 191 (EM)	User area 1: bytes 1 - 4 (alarm no.: 700100-700131)							
192 – 195 (OM)	User area 1: bytes 5 - 8 (alarm no.: 700132-700163)							
196 – 199 (EM)	User area 2: bytes 1 - 4 (alarm no.: 700200-700231)							
200 – 203 (OM)	User area 2: bytes 5 - 8 (alarm no.: 700232-700263)							
204 – 207 (EM)	User area 3: bytes 1 - 4 (alarm no.: 700300-700331)							
208 – 211 (OM)	User area 3: bytes 5 - 8 (alarm no.: 700332-700363)							
212 – 215 (EM)	User area 4: bytes 1 - 4 (alarm no.: 700400-700431)							
216 – 219 (OM)	User area 4: bytes 5 - 8 (alarm no.: 700432-700463)							
220 – 223 (EM)	User area 5: bytes 1 - 4 (alarm no.: 700500-700531)							
224 – 227 (OM)	User area 5: bytes 5 - 8 (alarm no.: 700532-700563)							
228 – 231 (EM)	User area 6: bytes 1 - 4 (alarm no.: 700600-700631)							
232 – 235 (OM)	User area 6: bytes 5 - 8 (alarm no.: 700632-700663)							
236 – 239 (EM)	User area 7: bytes 1 - 4 (alarm no.: 700700-700731)							
240 – 243 (OM)	User area 7: bytes 5 - 8 (alarm no.: 700732-700763)							
244 – 247 (EM)	User area 8: bytes 1 - 4 (alarm no.: 700800-700831)							
248 – 251 (OM)	User area 8: bytes 5 - 8 (alarm no.: 700832-700863)							
252 – 255 (EM)	User area 9: bytes 1 - 4 (alarm no.: 700900-700931)							
256 – 259 (OM)	User area 9: bytes 5 - 8 (alarm no.: 700932-700963)							
260 – 263 (EM)	User area 10: bytes 1 - 4 (alarm no.: 701000-701031)							
264 – 267 (OM)	User area 10: bytes 5 - 8 (alarm no.: 701032-701063)							
268 – 271 (EM)	User area 11: bytes 1 - 4 (alarm no.: 701100-701131)							
272 – 275 (OM)	User area 11: bytes 5 - 8 (alarm no.: 701132-701163)							
276 – 279 (EM)	User area 12: bytes 1 - 4 (alarm no.: 701200-701231)							
280 – 283 (OM)	User area 12: bytes 5 - 8 (alarm no.: 701232-701263)							
284– 287 (EM)	User area 13: bytes 1 - 4 (alarm no.: 701300-701331)							
288 – 291 (OM)	User area 13: bytes 5 - 8 (alarm no.: 701332-701363)							
292– 295 (EM)	User area 14: bytes 1 - 4 (alarm no.: 701400-701431)							
296 – 299 (OM)	User area 14: bytes 5 - 8 (alarm no.: 701432-701463)							
300– 303 (EM)	User area 15: bytes 1 - 4 (alarm no.: 701500-701531)							
304 – 307 (OM)	User area 15: bytes 5 - 8 (alarm no.: 701532-701563)							
308– 311 (EM)	User area 16: bytes 1 - 4 (alarm no.: 701600-701631)							
312 – 315 (OM)	User area 16: bytes 5 - 8 (alarm no.: 701632-701663)							
316– 319 (EM)	User area 17: bytes 1 - 4 (alarm no.: 701700-701731)							

Byte (Message type)	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
320 – 323 (OM)	User area 17: bytes 5 - 8 (alarm no.: 701732-701763)							
324 – 327 (EM)	User area 18: bytes 1 - 4 (alarm no.: 701800-701831)							
328 – 331 (OM)	User area 18: bytes 5 - 8 (alarm no.: 701832-701863)							
332 – 335 (EM)	User area 19: bytes 1 - 4 (alarm no.: 701900-701931)							
336 – 339 (OM)	User area 19: bytes 5 - 8 (alarm no.: 701932-701963)							
340 – 343 (EM)	User area 20: bytes 1 - 4 (alarm no.: 702000-702031)							
344 – 347 (OM)	User area 20: bytes 5 - 8 (alarm no.: 702032-702063)							
348 – 351 (EM)	User area 21: bytes 1 - 4 (alarm no.: 702100-702131)							
352 – 355 (OM)	User area 21: bytes 5 - 8 (alarm no.: 702132-702163)							
356 – 359 (EM)	User area 22: bytes 1 - 4 (alarm no.: 702200-702231)							
360 – 363 (OM)	User area 22: bytes 5 - 8 (alarm no.: 702232-702263)							
364 – 367 (EM)	User area 23: bytes 1 - 4 (alarm no.: 702300-702331)							
368 – 371 (OM)	User area 23: bytes 5 - 8 (alarm no.: 702332-702363)							
372 – 375 (EM)	User area 24: bytes 1 - 4 (alarm no.: 702400-702431)							
376 – 379 (OM)	User area 24: bytes 5 - 8 (alarm no.: 702432-702463)							
380 – 383 (EM)	User area 25: bytes 1 - 4 (alarm no.: 702500-702531)							
384 – 387 (OM)	User area 25: bytes 5 - 8 (alarm no.: 702532-702563)							
388 – 389 (EM)	User area 26: bytes 1 - 4 (alarm no.: 702600-702631)							
390 – 391 (OM)	User area 26: bytes 5 - 8 (alarm no.: 702632-702663)							
392 – 395 (EM)	User area 27: bytes 1 - 4 (alarm no.: 702700-702731)							
396 – 403 (OM)	User area 27: bytes 5 - 8 (alarm no.: 702732-702763)							
404 – 407 (EM)	User area 28: bytes 1 - 4 (alarm no.: 702800-702831)							
408 – 411 (OM)	User area 28: bytes 5 - 8 (alarm no.: 702832-702863)							
412 – 415 (EM)	User area 29: bytes 1 - 4 (alarm no.: 702900-702931)							
416 – 419 (OM)	User area 29: bytes 5 - 8 (alarm no.: 702932-702963)							
420 – 423 (EM)	User area 30: bytes 1 - 4 (alarm no.: 703000-703031)							
424 – 427 (OM)	User area 30: bytes 5 - 8 (alarm no.: 703032-703063)							
428 – 431 (EM)	User area 31: bytes 1 - 4 (alarm no.: 703100-703131)							
432 – 435 (OM)	User area 31: bytes 5 - 8 (alarm no.: 703132-703163)							

2.3.2 FC10-Alarme im DB 2 (FB1-Parameter "ExtendAlMsg"=True)

Channel areas in DB 2 (Parameter "ExtendAlMsg"=True)

DB 2	Signals for PLC messsages (PLC → HMI)							
Byte (Message type)	**Bit 7**	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
	Channel 1							
0								
	Feed disable (without alarm no.)							
1								
2	Feed and read-in disable byte 1 (without alarm no.)							
3	Feed and read-in disable byte 2 (without alarm no.)							
4	Feed and read-in disable byte 3 (without alarm no.)							
5	Feed and read-in disable byte 4 (without alarm no.)							
6	Read-in disable byte 1 (without alarm no.)							
7	Read-in disable byte 2 (without alarm no.)							
8	Read-in disable byte 3 (without alarm no.)							
9	Read-in disable byte 4 (without alarm no.)							
10	NC Start disable byte 1 (without alarm no.)							
11	NC Start disable byte 2 (without alarm no.)							
12	Feed stop GEOaxis 1 byte 1 (without alarm no.)							
13	Feed stop GEOaxis 1 byte 2 (without alarm no.)							
14	Feed stop GEOaxis 2 byte 1 (without alarm no.)							
15	Feed stop GEOaxis 2 byte 2 (without alarm no.)							
16	Feed stop GEOaxis 3 byte 1 (without alarm no.)							
17	Feed stop GEOaxis 3 byte 2 (without alarm no.)							
18 - 119	**Channel 2 - Channel 10** (without alarm no.)							

Byte (Message type)	**Bit 7**	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
	Channel 1							
310 (EM)	510007	510006	510005	510004	510003	510002	510001	510000
	Feed disable (alarm no.: 510000-510015)							
311 (OM)	510015	510014	510013	510012	510011	510010	510009	510008
312 (EM)	Feed and read-in disable byte 1 (alarm no.: 510100-510107)							
313 (EM)	Feed and read-in disable byte 2 (alarm no.: 510108-510115)							
314 (OM)	Feed and read-in disable byte 3 (alarm no.: 510116-510123)							
315 (OM)	Feed and read-in disable byte 4 (alarm no.: 510124-510131)							

316 (EM)	Read-in disable byte 1 (alarm no.: 510200-510207)
317 (EM)	Read-in disable byte 2 (alarm no.: 510208-510215)
318 (OM)	Read-in disable byte 3 (alarm no.: 510216-510223)
319 (OM)	Read-in disable byte 4 (alarm no.: 510224-510231)
320 (EM)	NC Start disable byte 1 (alarm no.: 510300-510307)
321 (OM)	NC Start disable byte 2 (alarm no.: 510308-510315)
322 (EM)	Feed stop GEOaxis 1 byte 1 (alarm no.: 511100-511107)
323 (OM)	Feed stop GEOaxis 1 byte 2 (alarm no.: 511108-511115)
324 (EM)	Feed stop GEOaxis 2 byte 1 (alarm no.: 511200-511207)
325 (OM)	Feed stop GEOaxis 2 byte 2 (alarm no.: 511208-511215)
326 (EM)	Feed stop GEOaxis 3 byte 1 (alarm no.: 511300-511307)
327 (OM)	Feed stop GEOaxis 3 byte 2 (alarm no.: 511308-511315)

Byte (Message type)	**Bit 7**	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
	Channel 2							
328 (EM)	520007	520006	520005	520004	520003	520002	520001	520000
	Feed disable (alarm no.: 510000-520015)							
329 (OM)	520015	520014	520013	520012	520011	520010	520009	520008
330 (EM)	Feed and read-in disable byte 1 (alarm no.: 520100-520107)							
331 (EM)	Feed and read-in disable byte 2 (alarm no.: 520108-520115)							
332 (OM)	Feed and read-in disable byte 3 (alarm no.: 520116-520123)							
333 (OM)	Feed and read-in disable byte 4 (alarm no.: 520124-520131)							
334 (EM)	Read-in disable byte 1 (alarm no.: 520200-520207)							
335 (EM)	Read-in disable byte 2 (alarm no.: 520208-520215)							
336 (OM)	Read-in disable byte 3 (alarm no.: 520216-520223)							
337 (OM)	Read-in disable byte 4 (alarm no.: 520224-520231)							
338 (EM)	NC Start disable byte 1 (alarm no.: 520300-520307)							
339 (OM)	NC Start disable byte 2 (alarm no.: 520308-520315)							
340 (EM)	Feed stop GEOaxis 1 byte 1 (alarm no.: 521100-521107)							
341 (OM)	Feed stop GEOaxis 1 byte 2 (alarm no.: 521108-521115)							
342 (EM)	Feed stop GEOaxis 2 byte 1 (alarm no.: 521200-521207)							
343 (OM)	Feed stop GEOaxis 2 byte 2 (alarm no.: 521208-521215)							
344 (EM)	Feed stop GEOaxis 3 byte 1 (alarm no.: 521300-521307)							
345 (OM)	Feed stop GEOaxis 3 byte 2 (alarm no.: 521308-521315)							

Byte (Message type)	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
	Channel 3							
346 (EM)	530007	530006	530005	530004	530003	530002	530001	530000
	Feed disable (alarm no.: 530000-530015)							
347 (OM)	530015	530014	530013	530012	530011	530010	530009	530008
348 (EM)	Feed and read-in disable byte 1 (alarm no.: 530100-530107)							
349 (EM)	Feed and read-in disable byte 2 (alarm no.: 530108-530115)							
350 (OM)	Feed and read-in disable byte 3 (alarm no.: 530116-530123)							
351 (OM)	Feed and read-in disable byte 4 (alarm no.: 530124-530131)							
352 (EM)	Read-in disable byte 1 (alarm no.: 530200-530207)							
353 (EM)	Read-in disable byte 2 (alarm no.: 530208-530215)							
354 (OM)	Read-in disable byte 3 (alarm no.: 530216-530223)							
355 (OM)	Read-in disable byte 4 (alarm no.: 530224-530231)							
356 (EM)	NC Start disable byte 1 (alarm no.: 530300-530307)							
357 (OM)	NC Start disable byte 2 (alarm no.: 530308-530315)							
358 (EM)	Feed stop GEOaxis 1 byte 1 (alarm no.: 531100-531107)							
359 (OM)	Feed stop GEOaxis 1 byte 2 (alarm no.: 531108-531115)							
360 (EM)	Feed stop GEOaxis 2 byte 1 (alarm no.: 531200-531207)							
361 (OM)	Feed stop GEOaxis 2 byte 2 (alarm no.: 531208-531215)							
362 (EM)	Feed stop GEOaxis 3 byte 1 (alarm no.: 531300-531307)							
363 (OM)	Feed stop GEOaxis 3 byte 2 (alarm no.: 531308-531315)							

Byte (Message type)	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
	Channel 4							
364 (EM)	540007	540006	540005	540004	540003	540002	540001	540000
	Feed disable (alarm no.: 540000-540015)							
365 (OM)	540015	540014	540013	540012	540011	540010	540009	540008
366 (EM)	Feed and read-in disable byte 1 (alarm no.: 540100-540107)							
367 (EM)	Feed and read-in disable byte 2 (alarm no.: 540108-540115)							
368 (OM)	Feed and read-in disable byte 3 (alarm no.: 540116-540123)							
369 (OM)	Feed and read-in disable byte 4 (alarm no.: 540124-540131)							
370 (EM)	Read-in disable byte 1 (alarm no.: 540200-540207)							
371 (EM)	Read-in disable byte 2 (alarm no.: 540208-540215)							
372 (OM)	Read-in disable byte 3 (alarm no.: 540216-540223)							
373 (OM)	Read-in disable byte 4 (alarm no.: 540224-540231)							
374 (EM)	NC Start disable byte 1 (alarm no.: 540300-540307)							

375 (OM)	NC Start disable byte 2 (alarm no.: 540308-540315)
376 (EM)	Feed stop GEOaxis 1 byte 1 (alarm no.: 541100-541107)
377 (OM)	Feed stop GEOaxis 1 byte 2 (alarm no.: 541108-541115)
378 (EM)	Feed stop GEOaxis 2 byte 1 (alarm no.: 541200-541207)
379 (OM)	Feed stop GEOaxis 2 byte 2 (alarm no.: 541208-541215)
380 (EM)	Feed stop GEOaxis 3 byte 1 (alarm no.: 541300-541307)
381 (OM)	Feed stop GEOaxis 3 byte 2 (alarm no.: 541308-541315)

Byte (Message type)	**Bit 7**	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
	Channel 5							
382 (EM)	550007	550006	550005	550004	550003	550002	550001	550000
	Feed disable (alarm no.: 550000-550015)							
383 (OM)	550015	550014	550013	550012	550011	550010	550009	550008
384 (EM)	Feed and read-in disable byte 1 (alarm no.: 550100-550107)							
385 (EM)	Feed and read-in disable byte 2 (alarm no.: 550108-550115)							
386 (OM)	Feed and read-in disable byte 3 (alarm no.: 550116-550123)							
387 (OM)	Feed and read-in disable byte 4 (alarm no.: 550124-550131)							
388 (EM)	Read-in disable byte 1 (alarm no.: 550200-550207)							
389 (EM)	Read-in disable byte 2 (alarm no.: 550208-550215)							
390 (OM)	Read-in disable byte 3 (alarm no.: 550216-550223)							
391 (OM)	Read-in disable byte 4 (alarm no.: 550224-550231)							
392 (EM)	NC Start disable byte 1 (alarm no.: 550300-550307)							
393 (OM)	NC Start disable byte 2 (alarm no.: 550308-550315)							
394 (EM)	Feed stop GEOaxis 1 byte 1 (alarm no.: 551100-551107)							
395 (OM)	Feed stop GEOaxis 1 byte 2 (alarm no.: 551108-551115)							
396 (EM)	Feed stop GEOaxis 2 byte 1 (alarm no.: 551200-551207)							
397 (OM)	Feed stop GEOaxis 2 byte 2 (alarm no.: 551208-551215)							
398 (EM)	Feed stop GEOaxis 3 byte 1 (alarm no.: 551300-551307)							
399 (OM)	Feed stop GEOaxis 3 byte 2 (alarm no.: 551308-551315)							

Byte (Message type)	**Bit 7**	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
	Channel 6							
400 (EM)	560007	560006	560005	560004	560003	560002	560001	560000
	Feed disable (alarm no.: 560000-560015)							
401 (OM)	560015	560014	560013	560012	560011	560010	560009	560008
402 (EM)	Feed and read-in disable byte 1 (alarm no.: 560100-560107)							

403 (EM)	Feed and read-in disable byte 2 (alarm no.: 560108-560115)
404 (OM)	Feed and read-in disable byte 3 (alarm no.: 560116-560123)
405 (OM)	Feed and read-in disable byte 4 (alarm no.: 560124-560131)
406 (EM)	Read-in disable byte 1 (alarm no.: 560200-560207)
407(EM)	Read-in disable byte 2 (alarm no.: 560208-560215)
408 (OM)	Read-in disable byte 3 (alarm no.: 560216-560223)
409 (OM)	Read-in disable byte 4 (alarm no.: 560224-560231)
410 (EM)	NC Start disable byte 1 (alarm no.: 560300-560307)
411 (OM)	NC Start disable byte 2 (alarm no.: 560308-560315)
412 (EM)	Feed stop GEOaxis 1 byte 1 (alarm no.: 561100-561107)
413 (OM)	Feed stop GEOaxis 1 byte 2 (alarm no.: 561108-561115)
414 (EM)	Feed stop GEOaxis 2 byte 1 (alarm no.: 561200-561207)
415 (OM)	Feed stop GEOaxis 2 byte 2 (alarm no.: 561208-561215)
416 (EM)	Feed stop GEOaxis 3 byte 1 (alarm no.: 561300-561307)
417 (OM)	Feed stop GEOaxis 3 byte 2 (alarm no.: 561308-561315)

Byte (Message type)	**Bit 7**	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
	Channel 7							
418 (EM)	570007	570006	570005	570004	570003	570002	570001	570000
	Feed disable (alarm no.: 570000-570015)							
419 (OM)	570015	570014	570013	570012	570011	570010	570009	570008
420 (EM)	Feed and read-in disable byte 1 (alarm no.: 570100-570107)							
421 (EM)	Feed and read-in disable byte 2 (alarm no.: 570108-570115)							
422 (OM)	Feed and read-in disable byte 3 (alarm no.: 570116-570123)							
423 (OM)	Feed and read-in disable byte 4 (alarm no.: 570124-570131)							
424 (EM)	Read-in disable byte 1 (alarm no.: 570200-570207)							
425 (EM)	Read-in disable byte 2 (alarm no.: 570208-570215)							
426 (OM)	Read-in disable byte 3 (alarm no.: 570216-570223)							
427 (OM)	Read-in disable byte 4 (alarm no.: 570224-570231)							
428 (EM)	NC Start disable byte 1 (alarm no.: 570300-570307)							
429 (OM)	NC Start disable byte 2 (alarm no.: 570308-570315)							
430 (EM)	Feed stop GEOaxis 1 byte 1 (alarm no.: 571100-571107)							
431 (OM)	Feed stop GEOaxis 1 byte 2 (alarm no.: 571108-571115)							
432 (EM)	Feed stop GEOaxis 2 byte 1 (alarm no.: 571200-571207)							
433 (OM)	Feed stop GEOaxis 2 byte 2 (alarm no.: 571208-571215)							
434 (EM)	Feed stop GEOaxis 3 byte 1 (alarm no.: 571300-571307)							
435 (OM)	Feed stop GEOaxis 3 byte 2 (alarm no.: 571308-571315)							

Byte (Message type)	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
	Channel 8							
436 (EM)	580007	580006	580005	580004	580003	580002	580001	580000
	Feed disable (alarm no.: 580000-580015)							
437 (OM)	580015	580014	580013	580012	580011	580010	580009	580008
438 (EM)	Feed and read-in disable byte 1 (alarm no.: 580100-580107)							
439 (EM)	Feed and read-in disable byte 2 (alarm no.: 580108-580115)							
440 (OM)	Feed and read-in disable byte 3 (alarm no.: 580116-580123)							
441 (OM)	Feed and read-in disable byte 4 (alarm no.: 580124-580131)							
442 (EM)	Read-in disable byte 1 (alarm no.: 580200-580207)							
443 (EM)	Read-in disable byte 2 (alarm no.: 580208-580215)							
444 (OM)	Read-in disable byte 3 (alarm no.: 580216-580223)							
445 (OM)	Read-in disable byte 4 (alarm no.: 580224-580231)							
446 (EM)	NC Start disable byte 1 (alarm no.: 580300-580307)							
447 (OM)	NC Start disable byte 2 (alarm no.: 580308-580315)							
448 (EM)	Feed stop GEOaxis 1 byte 1 (alarm no.: 581100-581107)							
449 (OM)	Feed stop GEOaxis 1 byte 2 (alarm no.: 581108-581115)							
450 (EM)	Feed stop GEOaxis 2 byte 1 (alarm no.: 581200-581207)							
451 (OM)	Feed stop GEOaxis 2 byte 2 (alarm no.: 581208-581215)							
452 (EM)	Feed stop GEOaxis 3 byte 1 (alarm no.: 581300-581307)							
453 (OM)	Feed stop GEOaxis 3 byte 2 (alarm no.: 581308-581315)							

Byte (Message type)	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
	Channel 9							
454 (EM)	590007	590006	590005	590004	590003	590002	590001	590000
	Feed disable (alarm no.: 590000-590015)							
455 (OM)	590015	590014	590013	590012	590011	590010	590009	590008
456 (EM)	Feed and read-in disable byte 1 (alarm no.: 590100-590107)							
457 (EM)	Feed and read-in disable byte 2 (alarm no.: 590108-590115)							
458 (OM)	Feed and read-in disable byte 3 (alarm no.: 590116-590123)							
459 (OM)	Feed and read-in disable byte 4 (alarm no.: 590124-590131)							
460 (EM)	Read-in disable byte 1 (alarm no.: 590200-590207)							
461(EM)	Read-in disable byte 2 (alarm no.: 590208-590215)							
462 (OM)	Read-in disable byte 3 (alarm no.: 590216-590223)							
463 (OM)	Read-in disable byte 4 (alarm no.: 590224-590231)							
464 (EM)	NC Start disable byte 1 (alarm no.: 590300-590307)							
465 (OM)	NC Start disable byte 2 (alarm no.: 590308-590315)							

466 (EM)	Feed stop GEOaxis 1 byte 1 (alarm no.: 591100-591107)
467 (OM)	Feed stop GEOaxis 1 byte 2 (alarm no.: 591108-591115)
468 (EM)	Feed stop GEOaxis 2 byte 1 (alarm no.: 591200-591207)
469 (OM)	Feed stop GEOaxis 2 byte 2 (alarm no.: 591208-591215)
470 (EM)	Feed stop GEOaxis 3 byte 1 (alarm no.: 591300-591307)
471 (OM)	Feed stop GEOaxis 3 byte 2 (alarm no.: 591308-591315)

Byte (Message type)	**Bit 7**	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
	Channel 10							
472 (EM)	500007	500006	500005	500004	500003	500002	500001	500000
	Feed disable (alarm no.: 500000-500015)							
473 (OM)	500015	500014	500013	500012	500011	500010	500009	500008
474 (EM)	Feed and read-in disable byte 1 (alarm no.: 500100-500107)							
475 (EM)	Feed and read-in disable byte 2 (alarm no.: 500108-500115)							
476 (OM)	Feed and read-in disable byte 3 (alarm no.: 500116-500123)							
477 (OM)	Feed and read-in disable byte 4 (alarm no.: 500124-500131)							
478 (EM)	Read-in disable byte 1 (alarm no.: 500200-500207)							
479(EM)	Read-in disable byte 2 (alarm no.: 500208-500215)							
480 (OM)	Read-in disable byte 3 (alarm no.: 500216-500223)							
481 (OM)	Read-in disable byte 4 (alarm no.: 500224-500231)							
482 (EM)	NC Start disable byte 1 (alarm no.: 500300-500307)							
483 (OM)	NC Start disable byte 2 (alarm no.: 500308-500315)							
484 (EM)	Feed stop GEOaxis 1 byte 1 (alarm no.: 501100-501107)							
485 (OM)	Feed stop GEOaxis 1 byte 2 (alarm no.: 501108-501115)							
486 (EM)	Feed stop GEOaxis 2 byte 1 (alarm no.: 501200-501207)							
487 (OM)	Feed stop GEOaxis 2 byte 2 (alarm no.: 501208-501215)							
488 (EM)	Feed stop GEOaxis 3 byte 1 (alarm no.: 501300-501307)							
489 (OM)	Feed stop GEOaxis 3 byte 2 (alarm no.: 501308-501315)							

Axis areas in DB 2 (Parameter "ExtendAlMsg" = True)

Byte (Message type)	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
	axis/spindle							
120	Feed stop/spindle stop for axis/spindle 1 (without alarm no.)							
121								
122 - 181	Feed stop/spindle stop for axis/spindle 2 - Feed stop/spindle stop for axis/spindle 31 (without alarm no.)							

Byte (Message type)	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
	axis/spindle							
490 (EM)	600107	600106	600105	600104	600103	600102	600101	600100
	Feed stop/spindle stop for axis/spindle 1 (alarm no.: 600100-600015)							
491 (OM)	600115	600114	600113	600112	600111	600110	600109	600108
492 (EM)	Feed stop/spindle stop for axis/spindle 2 (alarm no.: 600200-600207)							
493 (OM)	Feed stop/spindle stop for axis/spindle 2 (alarm no.: 600208-600215)							
494 (EM)	Feed stop/spindle stop for axis/spindle 3 (alarm no.: 600300-600307)							
495 (OM)	Feed stop/spindle stop for axis/spindle 3 (alarm no.: 600308-600315)							
496 (EM)	Feed stop/spindle stop for axis/spindle 4 (alarm no.: 600400-600407)							
497 (OM)	Feed stop/spindle stop for axis/spindle 4 (alarm no.: 600408-600415)							
498 (EM)	Feed stop/spindle stop for axis/spindle 5 (alarm no.: 600500-600507)							
499 (OM)	Feed stop/spindle stop for axis/spindle 5 (alarm no.: 600508-600515)							
500 (EM)	Feed stop/spindle stop for axis/spindle 6 (alarm no.: 600600-600607)							
501 (OM)	Feed stop/spindle stop for axis/spindle 6 (alarm no.: 600608-600615)							
502 (EM)	Feed stop/spindle stop for axis/spindle 7 (alarm no.: 600700-600707)							
503 (OM)	Feed stop/spindle stop for axis/spindle 7(alarm no.: 600708-600715)							
504 (EM)	Feed stop/spindle stop for axis/spindle 8 (alarm no.: 600800-600807)							
505 (OM)	Feed stop/spindle stop for axis/spindle 8 (alarm no.: 600808-600815)							
506 (EM)	Feed stop/spindle stop for axis/spindle 9 (alarm no.: 600900-600907)							
507 (OM)	Feed stop/spindle stop for axis/spindle 9 (alarm no.: 600908-600915)							
508 (EM)	Feed stop/spindle stop for axis/spindle 10 (alarm no.: 601000-601007)							
509 (OM)	Feed stop/spindle stop for axis/spindle 10 (alarm no.: 601008-601015)							
510 (EM)	Feed stop/spindle stop for axis/spindle 11 (alarm no.: 601100-601107)							
511 (OM)	Feed stop/spindle stop for axis/spindle 11 (alarm no.: 601108-601115)							
512 (EM)	Feed stop/spindle stop for axis/spindle 12 (alarm no.: 601200-601207)							
513 (OM)	Feed stop/spindle stop for axis/spindle 12 (alarm no.: 601208-601215)							
514 (EM)	Feed stop/spindle stop for axis/spindle 13 (alarm no.: 601300-601307)							

Byte (Message type)	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
515 (OM)	Feed stop/spindle stop for axis/spindle 13 (alarm no.: 601308-601315)							
516 (EM)	Feed stop/spindle stop for axis/spindle 14 (alarm no.: 601400-601407)							
517 (OM)	Feed stop/spindle stop for axis/spindle 14 (alarm no.: 601408-601415)							
518 (EM)	Feed stop/spindle stop for axis/spindle 15 (alarm no.: 601500-601507)							
519 (OM)	Feed stop/spindle stop for axis/spindle 15 (alarm no.: 601508-601515)							
520 (EM)	Feed stop/spindle stop for axis/spindle 16 (alarm no.: 601600-601607)							
521 (OM)	Feed stop/spindle stop for axis/spindle 16 (alarm no.: 601608-601615)							
522 (EM)	Feed stop/spindle stop for axis/spindle 17 (alarm no.: 601700-601707)							
523 (OM)	Feed stop/spindle stop for axis/spindle 17 (alarm no.: 601708-601715)							
524 (EM)	Feed stop/spindle stop for axis/spindle 18 (alarm no.: 601800-601807)							
525 (OM)	Feed stop/spindle stop for axis/spindle 18 (alarm no.: 601808-601815)							
526 (EM)	Feed stop/spindle stop for axis/spindle 19 (alarm no.: 601900-601907)							
527 (OM)	Feed stop/spindle stop for axis/spindle 19 (alarm no.: 601908-601915)							
528 (EM)	Feed stop/spindle stop for axis/spindle 20 (alarm no.: 602000-602007)							
529 (OM)	Feed stop/spindle stop for axis/spindle 20 (alarm no.: 602008-602015)							
530 (EM)	Feed stop/spindle stop for axis/spindle 21 (alarm no.: 602100-602107)							
531 (OM)	Feed stop/spindle stop for axis/spindle 21 (alarm no.: 602108-602115)							
532 (EM)	Feed stop/spindle stop for axis/spindle 22 (alarm no.: 602200-602207)							
533 (OM)	Feed stop/spindle stop for axis/spindle 22 (alarm no.: 602208-602215)							
534 (EM)	Feed stop/spindle stop for axis/spindle 23 (alarm no.: 602300-602307)							
535 (OM)	Feed stop/spindle stop for axis/spindle 23 (alarm no.: 602308-602315)							
536 (EM)	Feed stop/spindle stop for axis/spindle 24 (alarm no.: 602400-602407)							
537 (OM)	Feed stop/spindle stop for axis/spindle 24 (alarm no.: 602408-602415)							
538 (EM)	Feed stop/spindle stop for axis/spindle 25 (alarm no.: 602500-602507)							
539 (OM)	Feed stop/spindle stop for axis/spindle 25 (alarm no.: 602508-602515)							
540 (EM)	Feed stop/spindle stop for axis/spindle 26 (alarm no.: 602600-602607)							
541 (OM)	Feed stop/spindle stop for axis/spindle 26 (alarm no.: 602608-602615)							
542 (EM)	Feed stop/spindle stop for axis/spindle 27 (alarm no.: 602700-602707)							
543 (OM)	Feed stop/spindle stop for axis/spindle 27 (alarm no.: 602708-602715)							
544 (EM)	Feed stop/spindle stop for axis/spindle 28 (alarm no.: 602800-602807)							
545 (OM)	Feed stop/spindle stop for axis/spindle 28 (alarm no.: 602808-602815)							
546 (EM)	Feed stop/spindle stop for axis/spindle 29 (alarm no.: 602900-602907)							
547 (OM)	Feed stop/spindle stop for axis/spindle 29 (alarm no.: 602908-602915)							
548 (EM)	Feed stop/spindle stop for axis/spindle 30 (alarm no.: 603000-603007)							
549 (OM)	Feed stop/spindle stop for axis/spindle 30 (alarm no.: 603008-603015)							
550 (EM)	Feed stop/spindle stop for axis/spindle 31 (alarm no.: 603100-603107)							
551 (OM)	Feed stop/spindle stop for axis/spindle 31 (alarm no.: 603108-603115)							

User areas in DB 2 (Parameter "ExtendAlMsg" = True)

Byte (Message type)	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
182								
	User area 0: byte 0 (without alarm no)							
183								
184 - 308	User area 0: byte 1 - User area 0: byte 63 (without alarm no)							

Byte (Message type)	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
554 (EM)	700007	700006	700005	700004	700003	700002	700001	700000
	User area 0 (alarm no.: 700000-700015)							
555 (EM)	700015	700014	700013	700012	700011	700010	700009	700008
556 (EM)	User area 0: byte 3 (alarm no.: 700016-700023)							
557 (EM)	User area 0: byte 4 (alarm no.: 700024-700031)							
558 (OM)	User area 0: byte 5 (alarm no.: 700032-700039)							
559 (OM)	User area 0: byte 6 (alarm no.: 700040-700047)							
560 (OM)	User area 0: byte 7 (alarm no.: 700048-700055)							
561 (OM)	User area 0: byte 8 (alarm no.: 700056-700063)							
562 - 565 (EM)	User area 1 bytes 1 - 4 (alarm no.: 700100-700131)							
566 - 569 (OM)	User area 1 bytes 5 - 8 (alarm no.: 700132-700163)							
570 - 573 (EM)	User area 2 bytes 1 - 4 (alarm no.: 700200-700231)							
574 - 577 (OM)	User area 2 bytes 5 - 8 (alarm no.: 700232-700263)							
578 - 581 (EM)	User area 3 bytes 1 - 4 (alarm no.: 700300-700331)							
582 - 585 (OM)	User area 3 bytes 5 - 8 (alarm no.: 700332-700363)							
586 - 589 (EM)	User area 4 bytes 1 - 4 (alarm no.: 700400-700431)							
590 - 593 (OM)	User area 4 bytes 5 - 8 (alarm no.: 700432-700463)							
594 - 597 (EM)	User area 5 bytes 1 - 4 (alarm no.: 700500-700531)							
598 - 601 (OM)	User area 5 bytes 5 - 8 (alarm no.: 700532-700563)							
602 - 605 (EM)	User area 6 bytes 1 - 4 (alarm no.: 700600-700631)							
606 - 609 (OM)	User area 6 bytes 5 - 8 (alarm no.: 700632-700663)							
610 - 613 (EM)	User area 7 bytes 1 - 4 (alarm no.: 700700-700731)							
614 - 617 (OM)	User area 7 bytes 5 - 8 (alarm no.: 700732-700763)							
618 - 621 (EM)	User area 8 bytes 1 - 4 (alarm no.: 700800-700831)							
622 - 625 (OM)	User area 8 bytes 5 - 8 (alarm no.: 700832-700863)							
626 - 629 (EM)	User area 9 bytes 1 - 4 (alarm no.: 700900-700931)							
630 - 633 (OM)	User area 9 bytes 5 - 8 (alarm no.: 700932 - 700963)							
634 - 637 (EM)	User area 10 bytes 1 - 4 (alarm no.: 701000-701031)							
638 - 641 (OM)	User area 10 bytes 5 - 8 (alarm no.: 701032-701063)							

Byte (Message type)	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
642 - 645 (EM)	User area 11 bytes 1 - 4 (alarm no.: 701100-701131)							
646 - 649 (OM)	User area 11 bytes 5 - 8 (alarm no.: 701132-701163)							
650 - 653 (EM)	User area 12 bytes 1 - 4 (alarm no.: 701200-701231)							
654 - 657 (OM)	User area 12 bytes 5 - 8 (alarm no.: 701232-701263)							
658 - 661 (EM)	User area 13 bytes 1 - 4 (alarm no.: 701300-701331)							
662 - 665 (OM)	User area 13 bytes 5 - 8 (alarm no.: 701332-701363)							
666 - 669 (EM)	User area 14 bytes 1 - 4 (alarm no.: 701400-701431)							
670 - 673 (OM)	User area 14 bytes 5 - 8 (alarm no.: 701432-701463)							
674 - 677 (EM)	User area 15 bytes 1 - 4 (alarm no.: 701500-701531)							
678 - 681 (OM)	User area 15 bytes 5 - 8 (alarm no.: 701532-701563)							
682 - 685 (EM)	User area 16 bytes 1 - 4 (alarm no.: 701600-701631)							
686 - 689 (EM)	User area 16 bytes 5 - 8 (alarm no.: 701632-701663)							
690 - 693 (EM)	User area 17 bytes 1 - 4 (alarm no.: 701700-701731)							
694 - 697 (OM)	User area 17 bytes 5 - 8 (alarm no.: 701732-701763)							
698 - 701 (EM)	User area 18 bytes 1 - 4 (alarm no.: 701800-701831)							
702 - 705 (OM)	User area 18 bytes 5 - 8 (alarm no.: 701832-701863)							
706 - 709 (EM)	User area 19 bytes 1 - 4 (alarm no.: 701900-701931)							
710 - 713 (OM)	User area 19 bytes 5 - 8 (alarm no.: 701932-701963)							
714 - 717 (EM)	User area 20 bytes 1 - 4 (alarm no.: 702000-702031)							
718 - 721 (OM)	User area 20 bytes 5 - 8 (alarm no.: 702032-702063)							
722 - 725 (EM)	User area 21 bytes 1 - 4 (alarm no.: 702100-702131)							
726 - 729 (OM)	User area 21 bytes 5 - 8 (alarm no.: 702132-702163)							
730 - 733 (EM)	User area 22 bytes 1 - 4 (alarm no.: 702200-702231)							
734 - 737 (OM)	User area 22 bytes 5 - 8 (alarm no.: 702232-702263)							
738 - 741 (EM)	User area 23 bytes 1 - 4 (alarm no.: 702300-702331)							
742 - 745 (OM)	User area 23 bytes 5 - 8 (alarm no.: 702332-702363)							
746 - 749 (EM)	User area 24 bytes 1 - 4 (alarm no.: 702400-702431)							
750 - 753 (OM)	User area 24 bytes 5 - 8 (alarm no.: 702432-702463)							
754 - 757 (EM)	User area 25 bytes 1 - 4 (alarm no.: 702500-702531)							
758 - 761 (OM)	User area 25 bytes 5 - 8 (alarm no.: 702532-702563)							
762 - 765 (EM)	User area 26 bytes 1 - 4 (alarm no.: 702600-702631)							
766 - 769 (OM)	User area 26 bytes 5 - 8 (alarm no.: 702632-702663)							
770 - 773 (EM)	User area 27 bytes 1 - 4 (alarm no.: 702700-702731)							
774 - 777 (OM)	User area 27 bytes 5 - 8 (alarm no.: 702732-702763)							
778 - 781 (EM)	User area 28 bytes 1 - 4 (alarm no.: 702800-702831)							
782 - 785 (OM)	User area 28 bytes 5 - 8 (alarm no.: 702832-702863)							
786 - 789 (EM)	User area 29 bytes 1 - 4 (alarm no.: 702900-702931)							
790 - 793 (OM)	User area 29 bytes 5 - 8 (alarm no.: 702932-702963)							

Byte (Message type)	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
794 - 797 (EM)	User area 30 bytes 1 - 4 (alarm no.: 703000-703031)							
798 - 801 (OM)	User area 30 bytes 5 - 8 (alarm no.: 703032-703063)							
802 - 805 (EM)	User area 31 bytes 1 - 4 (alarm no.: 703100-703131)							
806 - 809 (OM)	User area 31 bytes 5 - 8 (alarm no.: 703132-703163)							

2.3.3 Signals to NC (DB 10)

On-board NCK input and outputs

DB 10	Signals to NC (PLC → NC)							
Byte	**Bit 7**	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
DBB0	Disabling of digital NCK inputs /Z1-A2/							
	Digital inputs without hardware #)				On-board inputs §)			
	Input 8	Input 7	Input 6	Input 5	Input 4	Input 3	Input 2	Input 1
DBB1	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
DBB2 DBB3 Unas-signed								
DBB4	Disabling of digital NCK outputs /Z2-A4/							
	Digital outputs without hardware #)				On-board outputs §)			
	Output 8	Output 7	Output 6	Output 5	Output 4	Output 3	Output 2	Output 1
DBB5	Overwrite screenform of digital NCK outputs /Z2-A4/							
	Digital outputs without hardware #)				On-board outputs §)			
	Output 8	Output 7	Output 6	Output 5	Output 4	Output 3	Output 2	Output 1
DBB6	Setting value of digital NCK outputs from PLC /Z2-A4/							
	Digital outputs without hardware #)				On-board outputs §)			
	Output 8	Output 7	Output 6	Output 5	Output 4	Output 3	Output 2	Output 1
DBB7	Input screenform of digital NCK outputs /Z2-A4/							
	Digital outputs without hardware #)				On-board outputs §)			
	Output 8	Output 7	Output 6	Output 5	Output 4	Output 3	Output 2	Output 1
DBB8 - DBB29	Machine axis number table for FC 19, FC 24, FC 25, FC 26 (1st MCP)							
DBB30	Upper limit of machine axis numbers for FC 19, FC 24 (1st MCP) With 0, the max. number of machine axis numbers applies							
DBB32 - DBB53	Machine axis number table for FC 19, FC 24, FC 25, FC 26 (2nd MCP)							
DBB54	Upper limit of machine axis numbers for FC 19, FC 24 (2nd MCP) With 0, the max. number of machine axis numbers applies							

General signals to NCK (DB 10)

DB 10	Signals to NC (PLC → NC)							
Byte	**Bit 7**	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
DBB56	Keyswitch /Z1-A2/					Acknowl. Emergancy Stop /Z1-A2/	Emergancy Stop /Z1-A2/	
	Position 3	Position 2	Position 1	Position 0				
DBB57					PC shutdown Only 840Di evaluated			INC inputs in mode group area active
DBB58								Collision detection off
DBB59								

2.3.4 Signals from/to NCK/HMI (DB 10)

On board NCK inputs and outputs (DB 10)

DB 10	Signals from (NCK -> PLC)							
Byte	**Bit 7**	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
DBB60					Actual value of the digital ON-BOARD inputs of the NCK On-board inputs §) /Z2-A4/			
					Input 4	Input 3	Input 2	Input 1
DBB61 - DBB63								
DBB64	Setpoint for the digital outputs of the NCK without hardware /Z2-A4/				Setpoint for the digital on-board outputs of the NCK /Z2-A4/			
	Output 8	Output 7	Output 6	Output 5	Output 4	Output 3	Output 2	Output 1
DBB65 - DBB67				unassigned				
DBB68	Handwheel 1 moved							
DBB69	Handwheel 2 moved							
DBB70	Handwheel 3 moved							
DBB71	Modification counter inch/metric system of units							

	Signals from (HMI → PLC)							
Byte	**Bit 7**	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
DBB72	Status of the actual value display indicated (1st MCP)							
HT 8	Travel keys displayed						MCS/ WCS	Valid display
DBB73	Status of the actual value display indicated (2nd MCP)							
HT 8	Travel keys displayed						MCS/ WCS	Valid display
DBB74 - DBB79	Machine axis numbers of the displayed axes (1st MCP) MCP1AxisFromHMI							
HT 8								
DBB80 - DBB85	Machine axis numbers of the displayed axes (2nd MCP) MCP2AxisFromHMI							
DBB86	Reserved							
DBB88	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 sl.

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**
DBB90 ePS to PLC								
DBB91 PLC to ePS								
DBB92 GP → PLC							DP1 Bus Slaves OK	MPI/DP Bus Slaves OK
DBB93	unassigned							
DBB94	unassigned							
DBB95	unassigned							
DBB96	unassigned							
DBB97 HMI → PLC					Channel number for handwheel 1 /Z2-H1			
					D	C	B	A
DBB98 HMI → PLC					Channel number for handwheel 2 /Z2-H1			
					D	C	B	A
DBB99 HMI → PLC					Channel number for handwheel 3 /Z2-H1			
					D	C	B	A
DBB100 HMI → PLC				Axis number for handwheel 1 /Z2-H1				
	Machine axis /Z2-H1/	Handwheel selected /Z2-H1/	Contour handwheel	E	D	C	B	A
DBB101 HMI → PLC				Axis number for handwheel 2 /Z2-H1				
	Machine axis /Z2-H1/	Handwheel selected /Z2-H1/	Contour handwheel	E	D	C	B	A
DBB102 HMI → PLC				Axis number for handwheel 3 /Z2-H1/				
	Machine axis /Z2-H1/	Handwheel selected /Z2-H1/	Contour handwheel	E	D	C	B	A

DB 10	Signals from NC (NCK → PLC)							
Byte	**Bit 7**	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
DBB103 HMI → PLC	HMI battery alarm /Z1-A2/	HMI tempera-ture limit /Z1-A2/	AT box ready /Z1-A2/	HMI fan monitoring	HMI HD monitoring			Remote diagnosis active /FBFE/ /Z1-A2/

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**
DBB104 GP → PLC	NCK CPU ready /Z1-A2/	1 st OB1 cycle		Op2Key ready	Op1Key ready	HHU ready	MCP 2 ready	MCP 1 ready
DBB105 GP → PLC								Too-lmanage-ment command cancellation
DBB106							EMER-GENCY Stop active /Z1-N2/	Collision detection off /Z2-M5/
DBB107	Inch system /Z1-G2/	NCU-link active /Z2-B3/					Probe actuated	
							Probe 2	Probe 1
DBB108	NC ready /Z1-A2/	Drive ready /Z1-A2/	Drives in cyclic operation		HMI 1 CPU Ready (HMI to OPI) /Z1-A2/	HMI 2 CPU Ready (HMI to MPI) /Z1-A2/	HMI2 CPU ready E_HMI2 ready /Z1-A2/	
DBB109	NCK battery alarm /Z1-A2/	Air temp. alarm /Z1-A2/	Heat sink temp. alarm NCU 573 /Z1-A2/	PC operating system fault /HBI/				NCK alarm present /Z1-A2/
DBB110	Software cams minus /Z2-N3/							
	7	6	5	4	3	2	1	0
DBB111	Software cams minus /Z2-N3/							
	15	14	13	12	11	10	9	8
DBB112	Software cams minus /Z2-N3/							
	23	22	21	20	19	18	17	16
DBB113	Software cams minus /Z2-N3/							
	31	30	29	28	27	26	25	24
DBB114	Software cams plus /Z2-N3/							
	7	6	5	4	3	2	1	0
DBB115	Software cams plus /Z2-N3/							
	15	14	13	12	11	10	9	8

DB 10	Signals from NC (NCK → PLC)							
Byte	**Bit 7**	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
DBB116	Software cams plus /Z2-N3/							
	23	22	21	20	19	18	17	16
DBB117	Software cams plus /Z2-N3/							
	31	30	29	28	27	26	25	24
DBB118 eps to PLC	Eps-Daten							
DBB119 eps to PLC	Eps-Daten							
DBB120 eps to PLC	Eps-Daten							
DBB121 eps to PLC	Eps-Daten							

Note

regarding NCK-CPU Ready (DBX104.7):
This signal represents the NC sign-of-life monitoring. The signal has to be entered into the machine safety circuit.

External digital inputs of the NCK (DB 10)

DB 10	Signals to NC (PLC → NCK)							
Byte	**Bit 7**	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
DBB122	Disable the external digital NCK inputs							
	Input 16	Input 15	Input 14	Input 13	Input 12	Input 11	Input 10	Input 9
DBB123	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
DBB124	Disable the external digital NCK inputs							
	Input 24	Input 23	Input 22	Input 21	Input 20	Input 19	Input 18	Input 17
DBB125	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
DBB126	Disable the external digital NCK inputs							
	Input 32	Input 31	Input 30	Input 29	Input 28	Input 27	Input 26	Input 25
DBB127	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
DBB128	Disable the external digital NCK inputs							
	Input 40	Input 39	Input 38	Input 37	Input 36	Input 35	Input 34	Input 33
DBB129	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 (DB 10)

DB 10	Signals to NC (PLC → NCK)							
Byte	**Bit 7**	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
DBB130	Disable the external digital NCK outputs /Z2-A4/							
	Output 16	Output 15	Output 14	Output 13	Output 12	Output 11	Output 10	Output 9
DBB131	Overwrite screenform for the external digital NCK outputs /Z2-A4/							
	Output 16	Output 15	Output 14	Output 13	Output 12	Output 11	Output 10	Output 9
DBB132	Value from the PLC for the external digital NCK outputs /Z2-A4/							
	Output 16	Output 15	Output 14	Output 13	Output 12	Output 11	Output 10	Output 9
DBB133	Default screenform for the external digital NCK outputs /Z2-A4/							
	Output 16	Output 15	Output 14	Output 13	Output 12	Output 11	Output 10	Output 9
DBB134	Disable the external digital NCK outputs /Z2-A4/							
	Output 24	Output 23	Output 22	Output 21	Output 20	Output 19	Output 18	Output 17
DBB135	Overwrite screenform for the external digital NCK outputs /Z2-A4/							
	Output 24	Output 23	Output 22	Output 21	Output 20	Output 19	Output 18	Output 17
DBB136	Value from the PLC for the external digital NCK outputs /Z2-A4/							
	Output 24	Output 23	Output 22	Output 21	Output 20	Output 19	Output 18	Output 17
DBB137	Default screenform for the external digital NCK outputs /Z2-A4/							
	Output 24	Output 23	Output 22	Output 21	Output 20	Output 19	Output 18	Output 17
DBB138	Disable the external digital NCK outputs /Z2-A4/							
	Output 32	Output 31	Output 30	Output 29	Output 28	Output 27	Output 26	Output 25
DBB139	Overwrite screenform for the external digital NCK outputs /Z2-A4/							
	Output 32	Output 31	Output 30	Output 29	Output 28	Output 27	Output 26	Output 25
DBB140	Value from the PLC for the external digital NCK outputs /Z2-A4/							
	Output 32	Output 31	Output 30	Output 29	Output 28	Output 27	Output 26	Output 25
DBB141	Default screenform for the external digital NCK outputs /Z2-A4/							
	Output 32	Output 31	Output 30	Output 29	Output 28	Output 27	Output 26	Output 25
DBB142	Disable the external digital NCK outputs /Z2-A4/							
	Output 40	Output 39	Output 38	Output 37	Output 36	Output 35	Output 34	Output 33
DBB143	Overwrite screenform for the external digital NCK outputs /Z2-A4/							
	Output 40	Output 39	Output 38	Output 37	Output 36	Output 35	Output 34	Output 33
DBB144	Value from the PLC for the external digital NCK outputs /Z2-A4/							
	Output 40	Output 39	Output 38	Output 37	Output 36	Output 35	Output 34	Output 33
DBB145	Default screenform for the external digital NCK outputs /Z2-A4/							
	Output 40	Output 39	Output 38	Output 37	Output 36	Output 35	Output 34	Output 33

Analog inputs of the NCK (external) (DB 10)

DB 10	Signals to NCK (PLC → NCK)							
Byte	**Bit 7**	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
DBB146	Disable the analog NCK inputs /Z2-A4/							
	Input 8	Input 7	Input 6	Input 5	Input 4	Input 3	Input 2	Input 1
DBB147	Specified analog value for NCK from PLC /Z2-A4/							
	Input 8	Input 7	Input 6	Input 5	Input 4	Input 3	Input 2	Input 1
DBW148	Setpoint from PLC for analog input 1 of NCK /Z2-A4/							
DBW150	Setpoint from PLC for analog input 2 of NCK /Z2-A4/							
DBW152	Setpoint from PLC for analog input 3 of NCK /Z2-A4/							
DBW154	Setpoint from PLC for analog input 4 of NCK /Z2-A4/							
DBW156	Setpoint from PLC for analog input 5 of NCK /Z2-A4/							
DBW158	Setpoint from PLC for analog input 6 of NCK /Z2-A4/							
DBW160	Setpoint from PLC for analog input 7 of NCK /Z2-A4/							
DBW162	Setpoint from PLC for analog input 8 of NCK /Z2-A4/							
DBB164 DBB165	unassigned							

Analog outputs of the NCK (external) (DB 10)

DB 10	Signals to NCK (PLC → NCK)							
Byte	**Bit 7**	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
DBB166	Overwrite screenform for the analog NCK outputs /Z2-A4/							
	Output 8	Output 7	Output 6	Output 5	Output 4	Output 3	Output 2	Output 1
DBB167	Default screenform for the analog NCK outputs /Z2-A4/							
	Output 8	Output 7	Output 6	Output 5	Output 4	Output 3	Output 2	Output 1
DBB168	Disable the analog NCK outputs /Z2-A4/							
	Output 8	Output 7	Output 6	Output 5	Output 4	Output 3	Output 2	Output 1
DBB169	Reserved							
DBW170	Setpoint from PLC for analog output 1 of NCK /Z2-A4/							
DBW172	Setpoint from PLC for analog output 2 of NCK /Z2-A4/							
DBW174	Setpoint from PLC for analog output 3 of NCK /Z2-A4/							
DBW176	Setpoint from PLC for analog output 4 of NCK /Z2-A4/							
DBW178	Setpoint from PLC for analog output 5 of NCK /Z2-A4/							
DBW180	Setpoint from PLC for analog output 6 of NCK /Z2-A4/							
DBW182	Setpoint from PLC for analog output 7 of NCK /Z2-A4/							
DBW184	Setpoint from PLC for analog output 8 of NCK /Z2-A4/							

External digital input and output signals of the NCK (DB 10)

DB 10	Signals from NCK (NCK → PLC)							
Byte	**Bit 7**	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
DBB186	Actual value of external digital NCK inputs /Z2-A4/							
	Input 16	Input 15	Input 14	Input 13	Input 12	Input 11	Input 10	Input 9
DBB187	Actual value of external digital NCK inputs /Z2-A4/							
	Input 24	Input 23	Input 22	Input 21	Input 20	Input 19	Input 18	Input 17
DBB188	Actual value of external digital NCK inputs /Z2-A4/							
	Input 32	Input 31	Input 30	Input 29	Input 28	Input 27	Input 26	Input 25
DBB189	Actual value of external digital NCK inputs /Z2-A4/							
	Input 40	Input 39	Input 38	Input 37	Input 36	Input 35	Input 34	Input 33
DBB190	NCK setpoint for external digital NCK outputs /Z2-A4/							
	Output 16	Output 15	Output 14	Output 13	Output 12	Output 11	Output 10	Output 9
DBB191	NCK setpoint for external digital NCK outputs /Z2-A4/							
	Output 24	Output 23	Output 22	Output 21	Output 20	Output 19	Output 18	Output 17
DBB192	NCK setpoint for external digital NCK outputs /Z2-A4/							
	Output 32	Output 31	Output 30	Output 29	Output 28	Output 27	Output 26	Output 25
DBB193	NCK setpoint for external digital NCK outputs /Z2-A4/							
	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)							
Byte	**Bit 7**	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
DBW194	Actual value for analog input 1 of the NCK /Z2-A4/							
DBW196	Actual value for analog input 2 of the NCK /Z2-A4/							
DBW198	Actual value for analog input 3 of the NCK /Z2-A4/							
DBW200	Actual value for analog input 4 of the NCK /Z2-A4/							
DBW202	Actual value for analog input 5 of the NCK /Z2-A4/							
DBW204	Actual value for analog input 6 of the NCK /Z2-A4/							
DBW206	Actual value for analog input 7 of the NCK /Z2-A4/							
DBW208	Actual value for analog input 8 of the NCK /Z2-A4/							
DBW210	Setpoint for analog output 1 of the NCK /Z2-A4/							
DBW212	Setpoint for analog output 2 of the NCK /Z2-A4/							
DBW214	Setpoint for analog output 3 of the NCK /Z2-A4/							
DBW216	Setpoint for analog output 4 of the NCK /Z2-A4/							
DBW218	Setpoint for analog output 5 of the NCK /Z2-A4/							
DBW220	Setpoint for analog output 6 of the NCK /Z2-A4/							
DBW222	Setpoint for analog output 7 of the NCK /Z2-A4/							
DBW224	Setpoint for analog output 8 of the NCK /Z2-A4/							

DB 10	Signals from NCK (NCK → PLC)							
Byte	**Bit 7**	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
DBW226 - DBW233	Collision detection 8 bytes = 64 bits (NCK → PLC)							
DBW234 - DBW241	Collision detection 8 bytes = 64 bits (PLC → NCK)							

2.3.5 Signals from/to mode group (DB 11)

Mode group-specific signals (DB 11)

DB 11	Signals to mode group 1 (PLC → NCK)							
Byte	**Bit 7**	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
DBB0	Mode group reset /Z1-K1/	BAG-Stopp Mode group stop Axes plus spindle /Z1-K1/	Mode group stop /Z1-K1/	Mode change disable /Z1-K1/		Operating mode /Z1-K1/		
						JOG	MDA	AUTO-MATIC
DBB1	Single block /Z1-K1/					Machine function /Z1-K1/		
	Type A	Type B				REF	REPOS	TEACH IN
DBB2	Machine function							
			var. INC	10000 INC	1000 INC	100 INC	10 INC	1 INC
DBB3								

Note

about **machine function**: machine function defined centrally when signal "INC inputs in mode group area active" (DB 10.DBX57.0) is set.

DB 11	Signals from mode group 1 (NCK → PLC)							
Byte	**Bit 7**	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
DBB4 HMI → PLC						Strobe mode /Z1-K1/		
						JOG	MDA	AUTOM.
DBB5 HMI → PLC						Strobe machine function /Z1-K1/		
						REF	REPOS	TEACH IN
DBB6	All channels in reset state		NCK internal JOG active /Z1-K1/	Mode Group reseted /Z1-K1/	Mode group ready /Z1-K1/	Active operating mode /Z1-K1/		
						JOG	MDA	AUTOM.
DBB7						Active machine function /Z1-K1/		
						REF	REPOS	TEACH IN
DBB8	Machine function							
			var. INC	10000 INC	1000 INC	100 INC	10 INC	1 INC

DB 11	Signals to mode group 2 (PLC→NCK)							
Byte	**Bit 7**	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
DBB20	Mode group reset /Z1-K1/	Mode group stop Axes plus spindle /Z1-K1/	Mode group stop /Z1-K1/	Mode change disable /Z1-K1/		Operating mode /Z1-K1/		
						JOG	MDA	AUTO-MATIC
DBB21	Single block					Machine function /Z1-K1/		
	Type A	Type B				REF	REPOS	TEACH IN
DBB22	Machine function							
			var. INC	10000 INC	1000 INC	100 INC	10 INC	1 INC
DBB23	unassigned							

Note

about **machine function**: machine function defined centrally when signal "INC inputs in mode group area active" (DB 10.DBX57.0) is set.

DB 11	Signals from mode group 2 (NCK → PLC)							
Byte	**Bit 7**	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
DBB24 HMI → PLC						Strobe mode /Z1-K1/		
						JOG	MDA	AUTO-MATIC
DBB25 HMI → PLC						Strobe machine function /Z1-K1/		
						REF	REPOS	TEACH IN
DBB26	All channels in reset state /Z1-K1/	NCK internal JOG active /Z1-K1/	Mode group reseted /Z1-K1/		Mode group ready /Z1-K1/	Active operating mode /Z1-K1/		
						JOG	MDA	AUTO-MATIC
DBB27						Active machine function /Z1-K1/		
						REF	REPOS	TEACH IN
DBB28	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: DBB40	Mode group 7: DBB120
Mode group 4: DBB60	Mode group 8: DBB140
Mode group 5: DBB80	Mode group 9: DBB160
Mode group 6: DBB100	Mode group 10: DBB180

2.3.6 Signals for Safety SPL (safe programmable logic) (DB 18)

Parameterization section

References: 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**
DBB0	not relevant for solution line							
DBB3								
DBW34	not relevant for solution line							
DBB36							Stop E	SPL READY
DBB37								

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							
DBD38	SPL_DATA.INSEP [1..32]							
DBD42	SPL_DATA.INSEP [33..64]							
DBD46	SPL_DATA.OUTSEP [1..32]							
DBD50	SPL_DATA.OUTSEP [33..64]							
	Data area for user SPL							
DBD54	SPL_DATA.INSIP [1..32]							
DBD58	SPL_DATA.INSIP [33..64]							
DBD62	SPL_DATA.OUTSIP [1..32]							
DBD66	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**
DBD70	SPL_DATA.MARKERSIP [1..32]							
DBD74	SPL_DATA.MARKERSIP [33..64]							
	Difference in level between NCK and PLC for diagnostics							
DBD78	SPL_DELTA.INSEP [1..32]							
DBD82	SPL_DELTA.INSEP [33..64]							
DBD86	SPL_DELTA.OUTSEP [1..32]							
DBD90	SPL_DELTA.OUTSEP [33..64]							
DBD94	SPL_DELTA.INSIP [1..32]							
DBD98	SPL_DELTA.INSIP [33..64]							
DBD102	SPL_DELTA.OUTSIP [1..32]							
DBD106	SPL_DELTA.OUTSIP [33..64]							
DBD110	SPL_DELTA.MARKERSIP [1..32]							
DBD114	SPL_DELTA.MARKERSIP [33..64]							
DBD118								CMDSI
DBD119			Systemfehler cross-checking					
DBD120	Error number 0 = no error 1 - 320 = Signal number starting from SPL_DATA.INSEP [1]							
DBD124	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 **DBB128**	PLCSIOUT [1 .. 8]							
from NCK **DBB129**	PLCSIOUT [9 .. 16]							
from NCK **DBB130**	PLCSIOUT [17 .. 24]							
from NCK **DBB131**	PLCSIOUT [25 .. 32]							
to NCK **DBB132**	PLCSIIN [1.. 8]							
to NCK **DBB133**	PLCSIIN [9 .. 16]							
to NCK **DBB134**	PLCSIIN [17 .. 24]							
to NCK **DBB135**	PLCSIIN [25 .. 32]							
DBB136	SPL status							
DBB138	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
DBB139								
DBB140	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
DBB141								
DBB142 - 149								
DBB150 - DBB157								
DBB158 - DBB188								

2.3.7 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**
DBB0	Actual value in WCS 0=MCS /Z1-A2/	Back up travel recorder	HMI Adv shutdown (for OEM users)	Clear recall alarms HMI Adv	Clear cancel alarms HMI Adv	Key disable /Z1-A2/	Screen darkening /Z1-A2/	Screen bright /Z1-A2/
DBB1								
DBW2								
DBW4								
DBB6	Analog spindle 1, capacity in percent							
DBB7	Analog spindle 2, capacity in percent							
DBB8	Channel number of machine control panel to HMI							
DBB9	Reserved for selection					Automatc tool measure-ment	OEM2	OEM1
DBB10	PLC Hardkeys (Values 1 ... 255, Default: 0)							
DBB11	Reserved for hardkey function expansions							
DBB12								
DBB13	Select /Z1-A2/	Load part program /Z1-A2/	Unload /Z1-A2/	Reserved				Disable Teach transfer
DBB14	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.						
DBB15	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							
DBB16	1=pas FS	Part program handling: Number of the control file for user file names.						
DBB17	Part program handling: Index of the file to be transferred in the user list							
DBB18								
DBB19	Reserved (signal counter)							

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**
DBB20	MCS/WCS Change-over /Z1-A2/	Simulation active /Z1-A2/	Language 2 switched HMI Emb.	Recall alarm cleared HMI Adv /Z1-A2/	Cancel alarm cleared HMI Adv /Z1-A2/	Cancel key actuated /Z1-A2/	Screen is dark /Z1-A2/	
DBB21	Active HMI operating area							
DBB22	Displayed channel number from the HMI /Z1-A2/							
DBB23								
DBW24	actual mask number from JobShop							
DBB26	Part program handling status /Z1-A2/							
	Select	Load	Unload		Active	Error	OK	Reserved
DBB27	Error program handling /Z1-A2/							
DBW28	Mask number for "Extend user interface" /Z1-A2/							
DBB30	Control bits PLC --> HMI							
							Exit mask	Request mask
DBB31	Control bits PLC --> HMI							
	Inactive bit			Error, Not possible to request mask	Mask exited	Mask active	Mask requested	Mask request accepted
DBB32	FunctionSelectionNo. from PLC							
PLC → HMI	Busy function	Strobe function						
DBB33	Parameter 1 for FunctionSelectionNo. (function selection from DBB32)							
PLC → HMI								
DBB34	Parameter 2 for FunctionSelectionNo. (function selection from DBB32)							
PLC → HMI								
DBB35	Parameter 3 for FunctionSelectionNo. (function selection from DBB32)							
PLC → HMI								
DBB36	Error code for FunctionSelectionNo. (function selection from DBB32)							
HMI → PLC								
DBB37	Parameter 1 for FunctionSelectionNo. (function selection from DBB48)							
HMI → PLC								
DBB38	Parameter 2 for FunctionSelectionNo. (function selection from DBB48)							
HMI → PLC								
DBB39	Parameter 3 for FunctionSelectionNo. (function selection from DBB48)							
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**
DBB40 - DBB47	Reserved							
DBB48 HMI → PLC	PLC busy function	HMI strobe function	FunctionSelectionNo. from HMI					
DBB49 PLC → HMI	Error code for FunctionSelectionNo. (function selection from DBB48)							
Interface 2nd HMI								
DBB50 - DBB99	Assignment as for DBB 0 to DBB 49							
	Switchover interface to HMI							
	Knocking interface (HMI announces itself to NCU)							
DBW100	ONL_REQUEST /Z2-B3/ Online request from HMI HMI writes its client identification as online request (bit 8-15: bus type, bit 0-7: HMI bus address)							
DBW102	ONL_CONFIRM /Z2-B3/ Acknowledgment from PLC to online request PLC writes HMI client identification as acknowledgment (bus type, HMI bus address; as with DBW 100).							
DBW104	PAR_CLIENT_IDENT /Z2-B3/ HMI writes its client identification (bus type, HMI bus address; as with DBW 100).							
DBB106	PAR_HMI_TYP /Z2-B3/ Type of HMI as per NETNAMES.INI: Main / subordinate operator panel / server /...							
DBB107	PAR_MSTT_ADR /Z2-B3/ HMI writes address of MCP to be activated; 255, when no MCP activated							
DBB108	PAR_STATUS /Z2-B3/ PLC writes online enable for HMI.							
DBB109	PAR_Z_INFO /Z2-B3/ PLC writes additional info about status							

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**
DBW110	M_TO_N_ALIVE Sign of life from PLC to HMI through M to N block							
DBB112	Reserved bus type MCP							
DBB113	ParOpKeyAdr Direct key index knocking interface							
DBB114	ParTcuIndex TCU index knocking interface							
DBB115	ParHt2Index Ht2 index login interface							
DBB116	Direct key address 1st online interface							
DBB117	Direct key address 2nd online interface							
DBB118	TCU index 1st online interface							
DBB119	TCU index 2 nd online interface							
	Online interface HMI 1 (user)							
DBW120	MMC1_CLIENT_IDENT /Z2-B3/ PLC writes PAR_CLIENT_IDENT to MMCx_CLIENT_IDENT when HMI goes online.							
DBB122	MMC1_TYP /Z2-B3/ PLC writes PAR_MMC_TYP to MMCx_TYP when HMI goes online.							
DBB123	MMC1_MSTT_ADR /Z2-B3/ PLC writes PAR_MSTT_ADR to MMCx_MSTT_ADR when HMI goes online.							
DBB124	MMC1_STATUS /Z2-B3/ Connection status, HMI and PLC alternately write their requests/acknowledgments							
DBB125	MMC1_Z_INFO /Z2-B3/ Additional info connection status (pos./neg. acknowledgment, error messages...)							

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**
DBB126	Reserved	TCU1_ SHIFT_ LOCK	MMC1_ CHANGE_ DENIED /Z2-B3/	MMC1_ ACTIVE_ CHANGED /Z2-B3/	MMC1_ ACTIVE_ PERM /Z2-B3/	MMC1_ ACTIVE_ REQ /Z2-B3/	MMC1_ MSTT_ SHIFT_ LOCK /Z2-B3/	MMC1 SHIFT LOCK /Z2-B3/
DBB127	Reserved Bus type MCP							
DBB128 - DBB129	Reserved Transline (Transline DB number)							
Online interface HMI 2 (user)								
DBW130	MMC2_CLIENT_IDENT /Z2-B3/ PLC writes PAR_CLIENT_IDENT to MMCx_CLIENT_IDENT when HMI goes online.							
DBB132	MMC2_TYP /Z2-B3/ PLC writes PAR_MMC_TYP to MMCx_TYP when HMI goes online.							
DBB133	MMC2_MSTT_ADR /Z2-B3/ PLC writes PAR_MSTT_ADR to MMCx_MSTT_ADR when HMI goes online.							
DBB134	MMC2_STATUS /Z2-B3/ Connection status, MMC and PLC alternately write their requests/acknowledgments							
DBB135	MMC2_Z_INFO /Z2-B3/ Additional info connection status (pos./neg. acknowledgment, error messages...)							
DBB136	Reserved	TCU2_ SHIFT_ LOCK	MMC2_ CHANGE_ DENIED /Z2-B3/	MMC2_ ACTIVE_ CHANGED /Z2-B3/	MMC2_ ACTIVE_ PERM /Z2-B3/	MMC2_ ACTIVE_ REQ /Z2-B3/	MMC2_ MSTT_ SHIFT_ LOCK /Z2-B3/	MMC2_ SHIFT_ LOCK /Z2-B3/
DBB137	Reserved Bus type MCP							
DBB138 – DBB139	Reserved Transline (Transline DB number)							
DBB140 - DBB197	Code carrier input parameters Optional package SINTDC on HMI-Advanced required							
DBB198 - DBB249	Code carrier return parameters Optional package SINTDC on HMI-Advanced required							
DBB250 - DBB255	Commands Optional package SINTDC on HMI-Advanced required							
DBB256 - DBB267	Commands for Paramtm.exe Optional package SINTDC on HMI-Advanced required							
DBW268	Traffic light status Optional package TPM on HMI Advanced required							

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**
DBW270 to DBW394	Counter[1 ... 32] Optional package TPM on HMI Advanced required							

2.3.8 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**
DBW0	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.

2.3.9 Signals from/to NCK channel (DB 21 – DB 30)

DB 21 - DB 30	Signals to NCK channel (PLC→NCK)							
Byte	**Bit 7**	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
DBB0		Activate dry run feedrate /Z1-V1/	Activate M01 /Z1-K1/	Activate single block /Z1-K1/	Activate DRF /Z2-H1/			
DBB1	Activate program test /Z1-K1/	PLC action complete /Z1-K1/	CLC override /Z3-TE1/	CLC stop /Z3-TE1/	Time monitoring act. (tool manage-ment)	Synchron-ized action OFF /FBSY/	Enable protection zones /Z1-A3/	Activate referencing /Z1-R1/
DBB2	Skip block /Z1-K1/							
	/7	/6	/5	/4	/3	/2	/1	/0
DBB3	Nibbling and punching /K2-N4/							
			Manual release of stroke 2	Stroke not operating /K2-N4/	Stroke delayed /K2-N4/	Stroke suppres-sion /K2-N4/	Manual stroke enable /K2-N4/	Stroke enable /K2-N4/
DBB4	Feedrate override /Z1-V1/							
	H	G	F	E	D	C	B	A
DBB5	Rapid traverse override /Z1-V1/							
	H	G	F	E	D	C	B	A
DBB6	Feedrate override active override /Z1-V1/	Rapid traverse override active override /Z1-V1/		Program level abort /Z1-K1/	Delete subroutine no. of passes	Delete distance-to-go /Z1-A2/	Read-in disable /Z1-K1/	Feed disable override /Z1-V1/
DBB7	Reset /Z1-K1//		Suppress Start Lock	NC Stop axes plus spindle /Z1-K1/	NC Stop /Z1-K1/	NC Stop to block limit /Z1-K1/	NC Start /Z1-K1/	NC Start disable /Z1-K1/
DBB8	Activate machine-related protection area /Z1-A3/							
	Area 8	Area 7	Area 6	Area 5	Area 4	Area 3	Area 2	Area 1
DBB9	Activate machine-related protection area /Z1-A3/							
							Area 10	Area 9
DBB10	Activate channel-specific protection area /Z1-A3/							
	Area 8	Area 7	Area 6	Area 5	Area 4	Area 3	Area 2	Area 1
DBB11	Activate channel-specific protection area /Z1-A3/							
							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 (DBB4)
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 (DBB5)
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

DB 21 - DB 30	Signals to NCK channel (PLC → NCK)							
Byte	**Bit 7**	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
DBB12	Geometry axis 1							
	Traversing keys /Z2-H1/		Rapid traverse override /Z2-H1/	Traversing key disable /Z2-H1/	Feed stop /Z1-V1/	Activate handwheel /Z2-H1/		
	+	-				3	2	1
DBB13	Geometry axis 1 machine function /Z2-H1/							
			Var. INC	10000 INC	1000 INC	100 INC	10 INC	1 INC
DBB14	OEM signals geometry axis 1							
DBB15	Geometry axis 1							
DBB16	Geometry axis 2							
	Traversing keys /Z2-H1/		Rapid traverse override /Z2-H1/	Traversing key disable /Z2-H1/	Feed stop /Z1-V1/	Activate handwheel /Z2-H1/		
	+	-				3	2	1
DBB17	Geometry axis 2 machine function /Z2-H1/							
			Var. INC	10000 INC	1000 INC	100 INC	10 INC	1 INC
DBB18	OEM signals geometry axis 2							
DBB19	Geometry axis 2							

DB 21 - DB 30	Signals to NCK channel (PLC → NCK)							
Byte	**Bit 7**	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
DBB20	Geometry axis 3							
	Traversing keys /Z2-H1/		Rapid traverse override /Z2-H1/	Traversing key disable /Z2-H1/	Feed stop /Z1-V1/	Activate handwheel /Z2-H1/		
	+	-				3	2	1
DBB21	Geometry axis 3 machine function /Z2-H1/							
			Var. INC	10000 INC	1000 INC	100 INC	10 INC	1 INC
DBB22	OEM signals geometry axis 3							
DBB23	Geometry axis 3							

Note

about **machine function**: machine function only defined when signal "INC inputs in mode group area active" (DB 10.DBX57.0) is not set.

Operating signals from HMI/status signals from NC channel

DB 21 – DB 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**
DBB24 HMI → PLC		Dry run feedrate selected /Z1-V1/	M01 selected /Z1-K1/	Select NCK-related M01	DRF selected /Z2-H1/			
DBB25 HMI → PLC	Program test selected /Z1-K1/			REPOS MODE EDGE	Feedrate override for rapid traverse selected /Z1-V1/	REPOSPATHMODE		
						2	1	0
DBB26	Skip block selected /Z1-K1/							
HMI → PLC	7	6	5	4	3	2	1	0
DBB27 HMI → PLC							Skip block selected /Z1-K1/	Skip block selected /Z1-K1/
DBB28	OEM channel signals							
PLC → NCK								
DBB29 PLC → NCK	Do not disable tool	Switch off wear monitoring	Switch off workpiece counter	Activate PTP motion	Activate fixed feed 4 /FBMA/, /Z1-V1/	Activate fixed feed 3 /FBMA/, /Z1-V1/	Activate fixed feed 2 /FBMA/, /Z1-V1/	Activate fixed feed 1 /FBMA/, /Z1-V1/

DB 21 – DB 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**
DBB30 PLC → NCK	Activate contour handwheel							
	No tool change commands		Activate NCK-related M01 /Z1-H2/	Neg. direction simulation contour handwheel	Simulation contour handwheel on	Handwheel 3	Handwheel 2	Handwheel 1
DBB31 PLC → NCK						REPOSPATHMODE		
	Skip block active /9	Skip block active /8		REPOS MODE EDGE		2	1	0
DBB32 NCK → PLC		Last action block active /Z1-K1/	M00/M01 active /Z1-K1/	Approach block active /Z1-K1/	Action block active /Z1-K1/			Execution from external source active
DBB33 NCK → PLC	Program test active /Z1-K1/	Transformation active /Z1-K1/ /K2-M1/	M02/M30 active /Z1-K1/	Block search active /Z1-K1/	Handwheel override active /Z2-H1/	Revolutio-nal feedrate active /Z1-V1/	Orientable toolholder active	Referencing active /Z1-R1/
DBB34 NCK → PLC	OEM channel signals feedback							
DBB35 NCK → PLC	Channel status /Z1-K1/			Program status /Z1-K1/				
	Reset	Interrupted	Active	Aborted	Interrupted	Stopped	Waiting	Running
DBB36 NCK → PLC	NCK alarm with processing stop present /Z1-A2/	Channel-specific NCK alarm present /Z1-A2/	Channel ready for operation	Interrupt processing active /Z1-K1/	All axes stationary /Z1-B1/	All axes requiring reference points are referenced /Z1-R1/		
DBB37 NCK → PLC	Stop	Read-in	CLC	CLC	CLC	Contour handwheel active		
	at block end with SBL is sup-pressed /Z1-K1/	enable is ignored /Z1-K1/	stopped upper limit /Z3-TE1/	stopped lower limit /Z3-TE1/	active /Z3-TE1/	Handwheel 3 /Z2-H1/	Handwheel 2 /Z2-H1/	Handwhe el 1 /Z2-H1/
DBB38 NCK → PLC	Nibbling and punching /Z2-N4/							
							Acknowl. manual stroke enable /Z2-N4/	Stroke enable active /Z2-N4/
DBB39 NCK → PLC								Protection zones not guaran-teed

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 – DB 30	Signals from NCK channel (NCK → PLC)							
Byte	**Bit 7**	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
DBB40	Geometry axis 1							
	Traverse command /Z2-H1/		Travel requests			Handwheel active /Z2-H1/		
	plus	minus	plus	minus		3	2	1
DBB41	Geometry axis 1 active machine function /Z2-H1/							
			Var. INC	10000 INC	1000 INC	100 INC	10 INC	1 INC
DBB42	OEM signals geometry axis 1							
DBB43	Geometry axis 1							
DBB44 HMI → PLC								
DBB 46	Geometry axis 2							
	Traverse command /Z2-H1/		Travel requests			Handwheel active /Z2-H1/		
	plus	minus	plus	minus		3	2	1
DBB 47	Geometry axis 2 active machine function /Z2-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 /Z2-H1/		Travel requests			Handwheel active /Z2-H1/		
	plus	minus	plus	minus		3	2	1
DBB 53	Geometry axis 3 active machine function /Z2-H1/							
			Var. INC	10000 INC	1000 INC	100 INC	10 INC	1 INC
DBB54	OEM signals geometry axis 3							

DB 21 – DB 30	Signals from NCK channel (NCK → PLC)							
Byte	**Bit 7**	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
DBB55	Geometry axis 3							
DBB56 HMI → PLC								
DBB57								

Change signals on auxiliary function transfer from NC channel

DB 21 – DB 30	Signals from NCK channel (NCK→PLC)							
Byte	**Bit 7**	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
DBB58				M fct. 5 change /Z1-H2/	M fct. 4 change /Z1-H2/	M fct. 3 change /Z1-H2/	M fct. 2 change /Z1-H2/	M fct. 1 change /Z1-H2/
DBB59				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
DBB60		S fct. 3 quick	S fct. 2 quick	S fct. 1 quick		S fct. 3 change /Z1-H2/	S fct. 2 change /Z1-H2/	S fct. 1 change /Z1-H2/
DBB61		T fct 3 quick	T fct. 2 quick	T fct. 1 quick		T fct. 3 change/Z1-H2/	T fct. 2 change/Z1-H2/	T fct. 1 change /Z1-H2/
DBB62		D fct. 3 quick	D fct. 2 quick	D fct. 1 quick		D fct. 3 change /Z1-H2/	D fct. 2 change/ Z1-H2/	D fct. 1 change /Z1-H2/
DBB63				DL fct. quick				DL fct. change
DBB64		H fct. 3 quick	H fct. 2 quick	H fct. 1 quick		H fct. 3 change /Z1-H2/	H fct. 2 change /Z1-H2/	H fct. 1 change /Z1-H2/
DBB65			F fct. 6 change /Z1-H2/	F fct. 5 change /Z1-H2/	F fct. 4 change /Z1-H2/	F fct. 3 change /Z1-H2/	F fct. 2 change /Z1-H2/	F fct. 1 change /Z1-H2/
DBB66				M fct. 5 quick	M fct. 4 quick	M fct. 3 quick	M fct. 2 quick	M fct. 1 quick
DBB67			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 – DB 30	Signals from NCK channel (NCK→PLC)							
Byte	**Bit 7**	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
DBW68	Extended address M function 1 (binary) /Z1-H2/							
DBD70								
		M function 1 (binary) /Z1-H2/						
DBW74	Extended address M function 2 (binary) /Z1-H2/							
DBD76	M function 2 (binary) /Z1-H2/							
DBW80	Extended address M function 3 (binary) /Z1-H2/							
DBD82	M function 3 (binary) /Z1-H2/							
DBW86	Extended address M function 4 (binary) /Z1-H2/							
DBD88	M function 4 (binary) /Z1-H2/							
DBW92	Extended address M function 5 (binary) /Z1-H2/							
DBD94	M function 5 (binary) /Z1-H2/							
DBW98	Extended address S function 1 (binary) /Z1-H2/							
DBD100	S function 1 (REAL format) /Z1-H2/							
DBW104	Extended address S function 2 (binary) /Z1-H2/							
DBD106	S function 2 (REAL format) /Z1-H2/							
DBW110	Extended address S function 3 (binary) /Z1-H2/							
DBD112	S function 3 (REAL format) /Z1-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 - DB 30	Signals from NCK channel (NCK→PLC)							
Byte	**Bit 7**	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
DBW116	Extended address T function 1 (16 bit Int)							
DBW118 **DBD118**	T function 1 (binary) /Z1-H2/ For 8-decade T nos., T function 1 (32 bit DINT) is used in DBD 118 (see note)							
DBW120	Extended address T function 2 (16 bit Int)							
DBW122	T function 2 (Int)							
DBW124	Extended address T function 3 (16 bit Int)							
DBW126	T function 3 (Int)							
DBB128								
DBB129	D function 1 (binary) /Z1-H2/							
DBW130 **DBB130**	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)							
DBB131	D function 2 (8 bit Int)							
DBB132	Extended address D function 3 (8 bit Int)							
DBB133	D function 3 (8 bit Int)							
DBW134	Extended address DL function (16 bit Int)							
DBD136	DL function (REAL)							

Note

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 - DB 30	Signals from NCK channel (NCK→PLC)							
Byte	**Bit 7**	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
DBW140	Extended address H function 1 (binary) /Z1-H2/							
DBD142	H function 1 (REAL or Dint) /Z1-H2/							
DBW146	Extended address H function 2 (binary) /Z1-H2/							
DBD148	H function 2 (REAL or Dint) /Z1-H2/							
DBW152	Extended address H function 3 (binary) /Z1-H2/							
DBD154	H function 3 (REAL or Dint) /Z1-H2/							
DBW158	Extended address F function 1 (binary) /Z1-H2/							
DBD160	F function 1 (REAL format) /Z1-H2/							
DBW164	Extended address F function 2 (binary) /Z1-H2/							
DBD166	F function 2 (REAL format) /Z1-H2/							
DBW170	Extended address F function 3 (binary) /Z1-H2/							
DBD172	F function 3 (REAL format) / Z1-H2/							
DBW176	Extended address F function 4 (binary) / Z1-H2/							
DBD178	F function 4 (REAL format) /Z1-H2/							
DBW182	Extended address F function 5 (binary) / Z1-H2/							
DBD184	F function 5 (REAL format) /Z1-H2/							
DBW188	Extended address F function 6 (binary) /Z1-H2/							
DBD190	F function 6 (REAL format) /Z1-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:
  = 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 – DB 30	Signals from NCK channel (NCK→PLC)							
Byte	**Bit 7**	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
DBB194	Dynamic M functions							
	M07	M06	M05 *	M04 *	M03 *	M02	M01	M00
DBB195	Dynamic M functions /Z1-H2/							
	M15	M14	M13	M12	M11	M10	M09	M08
DBB196	Dynamic M functions /Z1-H2/							
	M23	M22	M21	M20	M19	M18	M17	M16
DBB197	Dynamic M functions /Z1-H2/							
	M31	M30	M29	M28	M27	M26	M25	M24
DBB198	Dynamic M functions /Z1-H2/							
	M39	M38	M37	M36	M35	M34	M33	M32
DBB199	Dynamic M functions /Z1-H2/							
	M47	M46	M45	M44	M43	M42	M41	M40
DBB200	Dynamic M functions /Z1-H2/							
	M55	M54	M53	M52	M51	M50	M49	M48
DBB201	Dynamic M functions /Z1-H2/							
	M63	M62	M61	M60	M59	M58	M57	M56
DBB202	Dynamic M functions / Z1-H2/							
	M71	M70 *	M69	M68	M67	M66	M65	M64
DBB203	Dynamic M functions /Z1-H2/							
	M79	M78	M77	M76	M75	M74	M73	M72
DBB204	Dynamic M functions /Z1-H2/							
	M87	M86	M85	M84	M83	M82	M81	M80
DBB205	Dynamic M functions /Z1-H2/							
	M95	M94	M93	M92	M91	M90	M89	M88
DBB206	Dynamic M functions /Z1-H2/							
					M99	M98	M97	M96
DBB207								

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 DB 21-30.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 - DB 30	Signals from NCK channel (NCK→PLC)							
Byte	**Bit 7**	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
DBB208	Number of active G function of G function group 1 (binary) /Z1-K1/							
DBB209	Number of active G function of G function group 2 (binary) /Z1-K1/							
DBB210	Number of active G function of G function group 3 (binary) /Z1-K1/							
DBB211	Number of active G function of G function group 4 (binary) /Z1-K1/							
DBB212	Number of active G function of G function group 5 (binary) /Z1-K1/							
DBB213	Number of active G function of G function group 6 (binary) /Z1-K1/							
DBB214	Number of active G function of G function group 7 (binary) /Z1-K1/							
DBB215	Number of active G function of G function group 8 (binary) /Z1-K1/							
...								
DBB270	Number of active G function of G function group n-1 (binary) /Z1-K1/							
DBB271	Number of active G function of G function group n (binary) /Z1-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 - DB 30	Signals from NCK channel (NCK→PLC)							
Byte	**Bit 7**	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
DBB272	Machine-related protection area preactivated /Z1-A3/							
	Area 8	Area 7	Area 6	Area 5	Area 4	Area 3	Area 2	Area 1
DBB273	Machine-related protection area preactivated /Z1-A3/							
							Area 10	Area 9
DBB274	Channel-specific protection area preactivated /Z1-A3/							
	Area 8	Area 7	Area 6	Area 5	Area 4	Area 3	Area 2	Area 1
DBB275	Channel-specific protection area preactivated /Z1-A3/							
							Area 10	Area 9
DBB276	Machine-related protection area violated /Z1-A3/							
	Area 8	Area 7	Area 6	Area 5	Area 4	Area 3	Area 2	Area 1
DBB277	Machine-related protection area violated /Z1-A3/							
							Area 10	Area 9
DBB278	Channel-related protection area violated /Z1-A3/							
	Area 8	Area 7	Area 6	Area 5	Area 4	Area 3	Area 2	Area 1
DBB279	Channel-related protection area violated /Z1-A3/							
							Area 10	Area 9

Instruction-controlled signals to NC channel

DB 21 – DB 30	Signals to NCK channel (NCK→PLC)							
Byte	**Bit 7**	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
DBB280							Synch. action disable request	Reserved
DBB281							Synch. action disabled	
DBW282	Reserved							
DBW284	Reserved							
DBW286	Reserved							
DBW288	Reserved							
DBW290	Reserved							
DBW292	Reserved							
DBW294	Reserved							
DBW296	Reserved							
DBW298	Reserved							
DBB300	Disable synchronized actions /FBSY/							
	No. 8	No. 7	No. 6	No. 5	No. 4	No. 3	No. 2	No. 1
DBB301	Disable synchronized actions /FBSY/							
	No. 16	No. 15	No. 14	No. 13	No. 12	No. 11	No. 10	No. 9
DBB302	Disable synchronized actions /FBSY/							
	No. 24	No. 23	No. 22	No. 21	No. 20	No. 19	No. 18	No. 17
DBB303	Disable synchronized actions /FBSY/							
	No. 32	No. 31	No. 30	No. 29	No. 28	No. 27	No. 26	No. 25
DBB304	Disable synchronized actions /FBSY/							
	No. 40	No. 39	No. 38	No. 37	No. 36	No. 35	No. 34	No. 33
DBB305	Disable synchronized actions /FBSY/							
	No. 48	No. 47	No. 46	No. 45	No. 44	No. 43	No. 42	No. 41
DBB306	Disable synchronized actions /FBSY/							
	No. 56	No. 55	No. 54	No. 53	No. 52	No. 51	No. 50	No. 49
DBB307	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 - DB 30	Signals from NCK channel (NCK→PLC)							
Byte	**Bit 7**	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
DBB308	Synchronized actions can be disabled /FBSY/							
	Nr. 8	Nr. 7	Nr. 6	Nr. 5	Nr. 4	Nr. 3	Nr. 2	Nr.1
DBB309	Synchronized actions can be disabled /FBSY/							
	Nr. 16	Nr. 15	Nr. 14	Nr. 13	Nr. 12	Nr. 11	Nr.10	Nr.9
DBB310	Synchronized actions can be disabled /FBSY/							
	Nr. 24	Nr. 23	Nr. 22	Nr. 21	Nr. 20	Nr. 19	Nr.18	Nr.17
DBB311	Synchronized actions can be disabled /FBSY/							
	Nr. 32	Nr. 31	Nr. 30	Nr. 29	Nr. 28	Nr. 27	Nr. 26	Nr.25
DBB312	Synchronized actions can be disabled /FBSY/							
	Nr. 40	Nr. 39	Nr. 38	Nr. 37	Nr. 36	Nr. 35	Nr. 34	Nr. 33
DBB313	Synchronized actions can be disabled /FBSY/							
	Nr. 48	Nr. 47	Nr. 46	Nr. 45	Nr. 44	Nr. 43	Nr. 42	Nr.41
DBB314	Synchronized actions can be disabled /FBSY/							
	Nr. 56	Nr. 55	Nr. 54	Nr. 53	Nr. 52	Nr. 51	Nr. 50	Nr.49
DBB315	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								
DBB316	Active G functions							
								G00 geo.
DBB317	Tool missing	PTP motion active /FB3/F2/	Travel request drive test				Workpiece setpoint reached	External language mode active
DBB318	Overstore active /Z1/A2	Dry-run feedrate active /FB1/V1/	Associated M01 active /FB3/H2/	Stop delayed	TOFF movement active /FB3/F2/	TOFF active /FB1/F2	Search active	ASUP stopped /FB1/K1/
DBB319	No tool change command active	Stop-delay-range not activated	Repos DEFERRAL Chan FB1/K1	Delay FTS	Repos Path Mode Ackn 2 /FB1/K1/	Repos Path Mode Ackn 1 /FB1/K1/	Repos Path Mode Ackn 0 /FB1/K1/	REPOS MODE EDGE ACKN /FB1/K1/

Signals to orientation axes

DB 21 - DB 30	Signals to NCK channel (PLC→NCK)							
Byte	**Bit 7**	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
DBB320	Traversing keys		Orientation axis 1					
	+	–	Rapid traverse override	Traversing key disable	Feed stop	Activate handwheel (bit value coding)		
	–	–						
DBB321	Orientation axis 1							
			var. INC	10000 INC	1000 INC	100 INC	10 INC	1 INC
DBB322	OEM signals orientation axis 1							
DBB323	Orientation axis 1							
			var. INC	10000 INC	1000 INC	100 INC	10 INC	1 INC
DBB324	Traversing keys		Orientation axis 2					
	+	–	Rapid traverse override	Traversing key disable	Feed stop	Activate handwheel (bit value coding)		
DBB325	Orientation axis 2							
			var. INC	10000 INC	1000 INC	100 INC	10 INC	1 INC
DBB326	OEM signals orientation axis 2							
DBB327	Orientation axis 2							
DBB328	Traversing keys		Orientation axis 3					
	+	–	Rapid traverse override	Traversing key disable	Feed stop	Activate handwheel (bit value coding)		
DBB329	Orientation axis 3							
			var. INC	10000 INC	1000 INC	100 INC	10 INC	1 INC
DBB330	OEM signals orientation axis 3							
DBB331	Orientation axis 3							

Signals from orientation axes

DB 21 - DB 30	Signals from NCK channel (NCK→PLC)							
Byte	**Bit 7**	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
DBB332	Orientation axis 1 /Z2-H1/							
	Travel command		Travel request			Handwheel active		
	plus	minus	plus	minus		(bit value coding)		
DBB333	Orientation axis 1							
	Active machine function							
			var. INC	10000 INC	1000 INC	100 INC	10 INC	1 INC
DBB334	OEM signals orientation axis 1							
DBB335	Orientation axis 1							
DBB336	Orientation axis 2 /Z2-H1/							
	Travel command		Travel request			Handwheel active (bit value coding)		
	plus	minus	plus	minus				
DBB337	Orientation axis 2							
	Active machine function							
			var. INC	10000 INC	1000 INC	100 INC	10 INC	1 INC
DBB338	OEM signals orientation axis 2							
DBB339	Orientation axis 2							
DBB340	Orientation axis 3 /Z2-H1/							
	Travel command		Travel request			Handwheel active		
	plus	minus	plus	minus		(bit value coding)		
DBB341	Orientation axis 3							
	Active machine function							
			var. INC	10000 INC	1000 INC	100 INC	10 INC	1 INC
DBB342	OEM signals orientation axis 3							
DBB343	Orientation axis 3							

Tool management functions from NC channel

DB 21 - DB 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								
DBB344					Last replacement tool of tool group	Transfer to new replacement tool	Tool limit value reached	Tool pre-warning limit reached
DBB345 - DBB347								
Transferred tool management functions								
DBD348	T number for tool prewarning limit (DInt)							
DBD352	T number for tool limit value (DInt)							
DBD356	T number of new replacement tool (DInt)							
DBD360	T number of last replacement tool (DInt)							

Signals from NC channel

DB 21 – DB 30	Signals from NCK channel (NCK→PLC)							
Byte	**Bit 7**	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
CH_CYCLES_SIG_IN (Bit 0 - 7)								
DBB364								
CH_CYCLES_SIG_IN (Bit 8 - 15)								
DBB365								
CH_CYCLES_SIG_OUT (Bit 0 - 7)								
DBB366								
CH_CYCLES_SIG_OUT (Bit 8 - 15)								
DBB367								
CH_OEM_TECHNO_SIG_IN (DBB368 - DBB371)								
DBB368								
DBB369								
DBB370								
DBB371								
DBB372	CH_OEM_TECHNO_SIG_OUT (DBB372 - DBB375)							
DBB373								
DBB374								
DBB375								
DBB376	ProgEventDisplay							

DBB377							Stop condition	Stop following collision detection.
DBB378							Silence ASUP aktive	ASUP aktive
DBB379								
DBB380								
DBB381								
DBB382								
DBB383								

Signals to NCK channel

DB 21 – DB 30	**Signals to NCK channel (NCK→PLC)**							
Byte	**Bit 7**	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
DBB384								Control program branch
DBB385								
DBB386								
DBB387								

2.3.10 Signals from/to axis/spindle (PLC → NCK) (DB 31 – DB 61)

DB 31 - DB 61	Signals to axis/spindle (PLC → NCK)							
Byte	**Bit 7**	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
DBB0 Axis and spindle	Feedrate override /Z1-V1/							
	H	G	F	E	D	C	B	A
DBB1 Axis and spindle	Override active /Z1-V1/	Position measuring system 2 /Z1-A2/	Position measuring system 1 /Z1-A2/	Follow-up mode /Z1-A2/	Axis/spindle disable /Z1-A2/	Sensor fixed stop /Z1-F1/	Acknowl. fixed stop reached /Z1-F1/	Drive test movement enable /Z1-A2/
DBB2 Axis and spindle	Reference point value /Z1-R1/				Clamping in progress /Z1-A3/	Delete distance-to-go/ spindle reset /Z1-S1/	Controller enable /Z1-A2/	Cam activation /Z1-N3/
	4	3	2	1				
DBB3 Axis and spindle	Program test axis/ spindle release	Velocity/ spindle speed limitation /Z1-A3/	Activate fixed feed 4 /FBMA/, /Z1-V1/	Activate fixed feed 3 /FBMA/, /Z1-V1/	Activate fixed feed 2 /FBMA/, /Z1-V1/	Activate fixed feed 1 /FBMA/, /Z1-V1/	Enable travel to fixed stop /Z1-F1/	Accept external ZO /Z1-K2/
DBB4 Axis and spindle	Traversing keys /Z2-H1/		Rapid traverse override /Z2-H1/	Traversing key disable /Z2-H1/	Feed stop/spindle stop /Z1-V1/	Activate handwheel /Z2-H1/		
	plus	minus				3	2	1
DBB5	Machine function /Z2-H1/							
Axis and spindle			Var. INC	10000 INC	1000 INC	100 INC	10 INC	1 INC
DBB6	OEM axis signals							
Axis and spindle								
DBB7								
DBB8	Request PLC axis/spindle /Z2-K5/			Activation signal with change of this byte /Z2-K5/	Allocate NC axis to channel /Z2-K5/			
					D	C	B	A
	Note DBX8.4: is automatically reset after assignment.							
DBB9					Lock parameter set definition from NC /Z1-A2/	Control parameter block /Z1-A2/		
						C	B	A
DBB10								REPOS DELAY
DBB11								Start brake test

DB 31 - DB 61	Signals to axis/spindle (PLC → NCK)							
Byte	**Bit 7**	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
DBB12 Axis	Delay reference point approach /Z1-R1/			Modulo limit enabled	2nd software limit switch /Z1-A3/		Hardware limit switch /Z1-A3/	
					plus	minus	plus	minus
DBB13 Axis								
DBB14 Axis					Joggen auf Position	JOG Fest-Festpkt-anfahren 2	JOG Fest-Festpkt-anfahren 1	JOG Fest-Festpkt-anfahren 0
DBB15 Axis								
DBB16 Spindle	Delete S value /Z1-S1/	No n-monitoring when changing gear /Z1-S1/	Resyn-chronize spindle 1 /Z1-S1/	Resyn-chronize spindle 2 /Z1-S1/	Gear has changed over /Z1-S1/	Actual gear stage /Z1-S1/		
						C	B	A
DBB17 Spindle		Invert M3/M4 /Z1-S1/	Resyn-chronize spindle at pos. 2 /Z1-S1/	Resyn-chronize spindle at pos. 1 /Z1-S1/				Feedrate override f. spindle valid /Z1-S1/
DBB18 Spindle	Setpoint rot. direct. /Z1-S1/		Oscillating speed /Z1-S1/	Oscillation via PLC /Z1-S1/				
	CCW	CW						
DBB19 Spindle	Spindle override /Z1-V1/							
	H	G	F	E	D	C	B	A
DBB20 Drive	Not used	Reserved	Release brake	Not used	Not used	Reserved Yaskawa	ramp generator quick stop /Z1-A2/	Not used
DBB21 Drive	Pulse enable /Z1-A2/	n controller integrator disable /Z1-A2/	Motor Selection done /Z1-A2/	Motor selection /Z1-A2/		Drive parameter set selection 0 … 7 /Z1-A2/		
				B	A	C	B	A
DBB22 Safety Integrated				Selection of Safe speed			Deselect safe standstill	Deselect safe velocity and standstill
				bit value 1	bit value 0			
DBB23 Safety Integrated	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

DB 31 - DB 61	Signals to axis/spindle (PLC → NCK)							
Byte	**Bit 7**	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
DBB24	Master/ slave on	Bit value for CTRLOUT_changed: 1 0 Change setpoint output assignment (for compile cycles)		Torque compensa-tion controller ON		CC_Slave axis Suppress link	Control Axis	Stepper motor Rotation monitoring
DBB25								
DBB26 Grinding	Enable ESR response			Enable slave axis overlay	Compen-sation control ON			
DBB27 Grinding	Stop				Resume			
	HIAxMove	Corr	DEPBCS	DEPMCS	HIAxMove	Corr	DEPBCS	DEPMCS
DBB28 Oscillation	PLC checks axis /Z2-P5/	AxStop, stop /Z2-P5/	Stop at next reversal point /Z2-P5/	Change reversal point /Z2-P5/	Set reversal point /Z2-P5/	AXRESUME /Z2-P2/	AXRESET /Z2-P2/	OscillAxExt Reversal
DBB29 Grinding			Disable automatic synchroni-zation	Start gantry synchroni-zation Gantry				
DBB30 Technology	Reserved							
DBB31 Technology		Track synchronism	Disable synchroni-zation	Resynchro-nize				
DBB32 Safety Integrated			Deselect external stop E	Deselect external stop D	Deselect external stop C	Deselect external stop A		
DBB33 Safety Integrated	Select override							
	Bit value 3	Bit value 2	Bit value 1	Bit value 0				
DBB34								
DBB35 - DBB55								
DBB56 PLC to HMI						Spindle inside clamping	Spindle speed display	Seperate feed drive as C axis engaged
DBB57								

DB 31 - DB 61	Signals to axis/spindle (PLC → NCK)							
Byte	**Bit 7**	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
DBB58	Reserved							
DBB59								
	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" (DB 21-30, DB 6.2) acts on a channel-specific basis. The IS "Spindle reset" (DXB2.2) acts on a spindle-specific basis.							
DBB60 Axis and spindle	Position reached /Z1-B1/ with exact stop fine	Position reached /Z1-B1/ with exact stop coarse	References/ synchro-nizes 2 /Z1-R1/	References/ synchro-nizes 1 /Z1-R1/	Encoder limit frequency exceeded 2 /Z1-A3/	Encoder limit frequency exceeded 1 /Z1-A3/	NCU_Link Axis active /Z2-B3/	Spindle /no axis /Z1-S1/
DBB61 Axis and spindle	Current controller active /Z1-A2/	Speed controller active /Z1-A2/	Position controller active /Z1-A2/	Axis/spindle stationary ($n < n_{min}$) /Z1-A2/	Follow-up mode active /Z1-A2/	Axis ready /Z2-B3/	Axial alarm	Travel request /Z1-F1/
DBB62	Axis container rotation active /Z2-B3/	Force fixed stop limited /Z1-F1/	Fixed stop reached /Z1-F1/	Activate travel to fixed stop /Z1-F1/	Measure-ment active /Z2-M5/	Revolutio-nal feedrate active /Z1-V1/	Handwheel overlay active /Z2-H1/	Software cams active /Z1-N3/
DBB63	Stop HIAxMove active	Stop Corr active	DEPBCS active	DEPMCS active	Axis/ spindle disable active	Axis stop active /Z2-P2/	PLC-controlled axis /Z2-P2/	AXRESET DONE /Z2-P2/
DBB64 Axis and spindle	Traverse command /Z2-H1/ plus	Traverse command /Z2-H1/ minus	Travel request plus	Travel request minus		Handwheel active /Z2-H1/ 3	Handwheel active /Z2-H1/ 2	Handwheel active /Z2-H1/ 1
DBB65 Axis and spindle	Active machine function /Z2-H1/		Var. INC	10000 INC	1000 INC	100 INC	10 INC	1 INC

DB 31 – DB 61	Signals to axis/spindle (NCK → PLC)							
Byte	**Bit 7**	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
DBB66 Axis and spindle	OEM axis signals (Reserved)							Activate monitoring /Z3-TE6/
DBB67								
DBB68	PLC axis/ spindle /Z2-K5/	Neutral axis/ spindle /Z2-K5/	Axis replacement possible /Z2-K5/	New type requested by PLC /Z2-K5/	NC axis/spindle in channel /Z2-K5/ D	C	B	A
DBB69	NCU number in NCU link network					Control parameter block C	B	A
DBB70						Repos delay quit	Repos shift valid	Repos shift
DBB71	PLC axis permanently assigned							Brake test active
DBB72								REPOS DELAY
DBB73								
DBB74				Modulo limit enabled active				
DBB75	JOG-position reached	Position JOG active	JOG fixed-point approach 2 reached	JOG fxed-point approach 1 reached	JOG fixed-point approach 0 reached	JOG fixed-point approach 2 active	JOG fixed-point approach 1 active	JOG fixed-point approach 0 active
DBB76 Axis	Rounding axis in position	Indexing axis in position /Z2-T1/	Positioning axis /Z2-P2/	Path axis				Scratch pulse /Z1-A2/
DBB77								Reduced-speed colision
DBB78 Axis	F function (REAL format) for positioning axis /Z1-V1/							
DBB82 Spindle					Gear change-over /Z1-S1/	Setpoint gear stage /Z1-S1/ C	B	A
DBB83 Spindle	Actual rotat. direction CW /Z1-S1/	Speed monitoring /Z1-V1/	Spindle in setpoint range /Z1-S1/	Support area limits violated	Geometry monitoring /Z1-V1/	Set speed increased /Z1-S1/	Set speed limited /Z1-S1/	Speed limit exceeded /Z1-S1/

DB 31 – DB 61	Signals to axis/spindle (NCK → PLC)							
Byte	**Bit 7**	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
DBB84 Spindle	Active spindle operating mode /Z1-S1/ Control mode	Oscillation mode	Position-ing mode	Synchronous mode	Tapping without compen-sating chuck /Z1-S1/	CLGON active /Z1-S8/	SUG active (grinding wheel surface speed)	Const. cutting speed active
DBB85 Spindle			Spindle in position					Werkzeug mit Dynamik-limitierung
DBB86 Spindle	M function (binary) for spindle /Z1-S1/							
DBD88 Spindle	S function (floating-point) for spindle /Z1-S1/							
DBB92 Drive			Motor brake released					
DBB93 Drive	Enable pulses /Z1-A2/	n controller integrator disabled /Z1-A2/	Drive ready /Z1-A2/	Active motor /Z1-A2/ B	A	Active drive parameter set 0 … 7 /Z1-A2/ C	B	A
DBB4 Drive		$n_{act} = n_{set}$ /Z1-A2/	$\|n_{act}\| < n_x$ /Z1-A2/	$\|n_{act}\| < n_{min}$ /Z1-A2/	Md < Mdx /Z1-A2/	Ramp-up complete /Z1-A2/	Temperature prewarning /Z1-A2/ Heat sink	Motor
DBB95 Drive								
DBB96	Master/ slave active /Z3-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	(Stepper motor) Axis control active /Z3-TE6/	Error rotation monitoring
DBB97					Offset after turn-on point /Z3-TE6/	Activate mirroring /Z3-TE6/	Coupling active /Z3-TE6/	Axis is slave axis /Z3-TE6/
DBB98 Synchro-nous spindle	Emergency retraction active	Accel. warning threshold reached	Speed warning threshold reached	Overlaid motion /Z2-S3/		Actual value coupling /Z2-S3/	Synchronism /Z2-S3/ coarse	fine

DB 31 – DB 61	Signals to axis/spindle (NCK → PLC)							
Byte	**Bit 7**	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
DBB99 Synchro-nous spindle	Emergency retraction enabled	Max. acce-leration reached	Max. speed reached	Synchro-nization running	Axis acce-lerating		Slave spindle active /Z2-S3/	Master spindle active /Z2-S3/
DBB100 Grinding	Oscillation active /Z2-P5/	Oscillation motion active /Z2-P5/	Spark-out active /Z2-P5/	Error in oscillation /Z2-P5/	Oscillation cannot start /Z2-P5/	OscillAxExtReversal active		
DBB101 Gantry	Gantry axis /Z2-G1/	Gantry leading axis /Z2-G1/	Gantry grouping is synchro-nous /Z2-G1/	Gantry synchroni-zation run ready to start /Z2-G1/	Gantry warning limit exceeded /Z2-G1/	Gantry cut-off limit exceeded /Z2-G1/		
DBB102 **DBB103**								
DBB104 Grinding	Active infeed axis /Z2-P5/							
	Axis 8	Axis 7	Axis 6	Axis 5	Axis 4	Axis 3	Axis 2	Axis 1
DBB105	Active infeed axis /Z2-P5/							
	Axis 16							Axis 9
DBB106	Active infeed axis /Z2-P5/							
	Axis 24							Axis 17
DBB107	Active infeed axis /Z2-P5/							
		Axis 31	Axis 30					Axis 25
DBB108	SINUMERIK Safety Integrated /FBSY/							
	Axis safely referenced					Status pulses deleted		Safe operational stop / safe speed active

DB 31 – DB 61	Signals from axis/spindle (NCK → PLC)							
Byte	**Bit 7**	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
DBB109	SINUMERIK Safety Integrated Actual position > cam position							
	SC 4-	SC 4+	SC 3-	SC 3+	SC 2-	SC 2+	SC 1-	SC 1+
DBB110	SINUMERIK Safety Integrated							
			n < nx	Safe velocity active bit value 1	Safe velocity active bit value 0		Safe zero speed active	
DBB111	Reserved for SINUMERIK Safety Integrated /FBSI/							
	Stop E active	Stop D active	Stop C active	Stop A/B active				
DBB112	SINUMERIK Safety Integrated /FBSI/							
DBB113	SINUMERIK Safety Integrated /FBSI/							
DBB114	SINUMERIK Safety Integrated /FBSI/							
DBB115	SINUMERIK Safety Integrated /FBSI/							

2.3.11 Tool management interface

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**
DBB0	Interface (I) active							
	I8	I7	I6	I5	I4	I3	I2	I1
DBB1								
	I16	I15	I14	I13	I12	I11	I10	I9
DBB2	Standard end-of-acknowledgement							
	I 8	I 7	I 6	I 5	I 4	I 3	I 2	I 1
DBB3								
	I 16	I 15	I 14	I 13	I 12	I 11	I 10	I 9
DBBn	Reserved	Reserved	Reserved	NC program positions magazine	Position at loading point	Reload	Unload	Load
DBBn + 1	Reserved	Reserved	Reserved	Reserved	Reserved	Reserved	Reserved	Acknowledgement Status = 3
DBBn + 2	Assigned channel (8 bit Int)							

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**
DBBn + 3	Tool management no. (8 bit Int)							
DBDn + 4	Unassigned parameter 1 (D word)							
DBDn + 8	Unassigned parameter 2 (D word)							
DBDn + 12	Unassigned parameter 3 (D word)							
DBWn + 16	Identification for loading/unloading station (Int), (fixed value 9999)							
DBWn + 18	No. of loading station (Int)							
DBWn + 20	Magazine no. (source) for unloading/reloading/positioning (Int)							
DBWn + 22	Location no. (source) for unloading/reloading/positioning (Int)							
DBWn + 24	Magazine no. (target) for loading/reloading/positioning (Int)							
DBWn + 26	Location no. (target) for loading/reloading/positioning (Int)							
DBBn + 28	Reserved							Loading/ unloading without magazine movement
DBBn + 29	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

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**
DBB0	Interface (I) active							
	I8	I7	I6	I5	I4	I3	I2	I1
DBB1								
	I16	I15	I14	I13	I12	I11	I10	I9
DBB2	Standard end-of-acknowledgement							
	I 8	I 7	I 6	I 5	I 4	I 3	I 2	I 1
DBB3								
	I 16	I 15	I 14	I 13	I 12	I 11	I 10	I 9
DBBn								
	Spindel WZ bleibt in Spindel	Replace manual tool	Replace manual tool.	OldT in buffer no. (n-42)	T0	Prepare change	Perform change (initiate: M06)	Compul-sory change
DBBn + 1	Reserve d	Reserved	Reserved	Reserved	Reserved	Reserved	Reserved	Acknowled gement Status = 3
DBBn + 1	unassigned							
DBBn + 2	Assigned channel (8 bit Int)							
DBBn + 3	Tool management no. (8 bit Int)							
DBDn + 4	Unassigned parameter 1 (D word)							
DBDn + 8	Unassigned parameter 2 (D word)							
DBDn + 12	Unassigned parameter 3 (D word)							
DBWn + 16	Buffer identification (Int), (fixed value 9998) (corresponds to ”Target position for new tool”)							
DBWn + 18	Relative location (target) in the buffer (Int)							
DBWn + 20	Magazine no. (source) for new tool (Int)							
DBWn + 22	Location no. (source) for new tool (Int)							
DBWn + 24	Magazine no. (target) for old tool (Int)							
DBWn + 26	Location no. (target) for old tool (Int)							
DBWn + 28	Tool new: location type (Int)							
DBWn + 30	Tool new: size left (Int)							
DBWn + 32	Tool new: size right (Int)							
DBWn + 34	Tool new: size top (Int)							
DBWn + 36	Tool new: size bottom (Int)							

DB 72	Signals from spindle (NCK→PLC)							
Byte	**Bit 7**	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
DBWn + 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
DBWn + 40	Tool new: T no. (Int)							
DBWn + 42	If DBX (n+0.4) = 1, then buffer location of old tool is entered here.							
DBWn + 44	Original magazine of new tool							
DBWn + 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

Interface for circular magazine (DB 73)

DB 73	Signals from circular magazine (NCK → PLC)							
Byte	**Bit 7**	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
DBB0	Interface (I) active							
	I8	I7	I6	I5	I4	I3	I2	I1
DBB1								
	I16	I15	I14	I13	I12	I11	I10	I9
DBB2	Standard end-of-acknowledgement							
	I 8	I 7	I 6	I 5	I 4	I 3	I 2	I 1
DBB3								
	I 16	I 15	I 14	I 13	I 12	I 11	I 10	I 9
DBBn								
	Reserved	Replace manual tool	Reserved	Reserved	T0	Reserved	Perform change (initiation: T no.)	Obligatory change
DBBn + 1	Reserved	Reserved	Reserved	Reserved	Reserved	Reserved	Reserved	Acknowledgement Status = 3
DBBn + 2	Assigned channel (8 bit Int)							
DBBn + 3	Tool management no. (8 bit Int)							
DBDn + 4	Unassigned parameter 1 (D word)							
DBDn + 8	Unassigned parameter 2 (D word)							
DBDn + 12	Unassigned parameter 3 (D word)							
DBWn + 16	Reserved							

DB 73	Signals from circular magazine (NCK → PLC)							
Byte	**Bit 7**	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
DBWn + 18	Reserved							
DBWn + 20	Circular magazine no. (Int)							
DBWn + 22	Location no. for new tool (Int)							
DBWn + 24	Magazine no. of the old tool							
DBWn + 26	Location no. for old tool (Int)							
DBWn + 28	Tool new: location type (Int)							
DBWn + 30	Tool new: size left (Int)							
DBWn + 32	Tool new: size right (Int)							
DBWn + 34	Tool new: size top (Int)							
DBWn + 36	Tool new: size bottom (Int)							
DBWn + 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
DBWn + 40	Tool new: T no. (Int)							
DBWn + 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

2.3.12 Signals to/from the machine control panel and HHU

DB 77	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**
DBB0 - DBB7	Input signals from MCP1 to PLC, MPI Bus (GD communication)							
DBB8 - DBB15	Output signals from MCP1 to PLC, MPI Bus (GD communication)							
DBD16	Status send MCP1, MPI bus (GD communication)							
DBD20	Status receive MCP1, MPI bus (GD communication)							
DBB24 - DBB31	Input signals from MCP2 to PLC, MPI bus (GD communication)							
DBB32 - DBB39	Output signals from MCP2 to PLC, MPI bus (GD communication)							
DBD40	Status send MCP2, MPI bus (GD communication)							
DBD44	Status receive MCP2, MPI bus (GD communication)							
DBB48 - DBB53	Input signals from HHU to PLC, MPI bus (GD communication)							
DBB60 - DBB79	Output signals from PLC to HHU, MPI bus (GD communication)							
DBD80	Status Send HHU, MPI bus (GD communication)							
DBD84	Status Receive HHU, MPI bus (GD communication)							

```
FB 1- 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

3 PLC-Blocks

3.1 Overview of organization blocks

Table 3-1: 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

3.2 Overview of function blocks

Table 3-2: 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	GP
30 - 35		Assigned if ManualTurn, ShopMill or ShopTurn are installed	GP
36 - 255		User assignable	

Table 3-3: 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
9	M2N	M : N changeover block	GP
10	SI_Relais	Safety Integrated relay	GP
11	SI_Braketest	Safety Integrated brake service	GP
29		Diagnostics for signal recorder and data trigger	GP
36 - 127		User assignable	
36 - 255		User assignable	

3.3 Assignment of data blocks

Note

Only so many DBs are created as are necessary according to NC-MD.

Tabelle 3-4: Overview of data blocks

DB no.	Designation	Meaning	Kit
1		Reserved for Siemens	GP
2 – 5	PLC MSG	PLC messages	GP
6 - 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 (for SDB210)	
78 - 80		Reserved for Siemens	
81 - 127		User assignable	
1000 - 1099		Reserved for Siemens	

Note

Data blocks of inactivated channels, axes/spindles, C programming, tool management can be assigned by the user.

3.4 Assigned timers

Tabelle 3-5: Assigned timers

Timer no.	Meaning
1 - 9	Reserved
10 - 127	User assignable

■

I Index

R

S

T

V

An
Siemens AG

I DT MC MS1
P. O. Box 3180

D-91050 Erlangen

Federal Republic of Germany

Fax +49 (0) 9131 / 98 - 2176 [Documentation]

mailto:docu.motioncontrol@siemens.com

Suggestions

Corrections

for Publication/Manual
SINUMERIK 840D sl, 840Di sl, SINAMICS S120

Lists, 2nd Book

Manufacturer Documentation

From
Name:
Company/Dept.

Adress:

_____

_____ _____

Telephone: _____ / _____

Telefax: _____ / _____

Email: _____ -

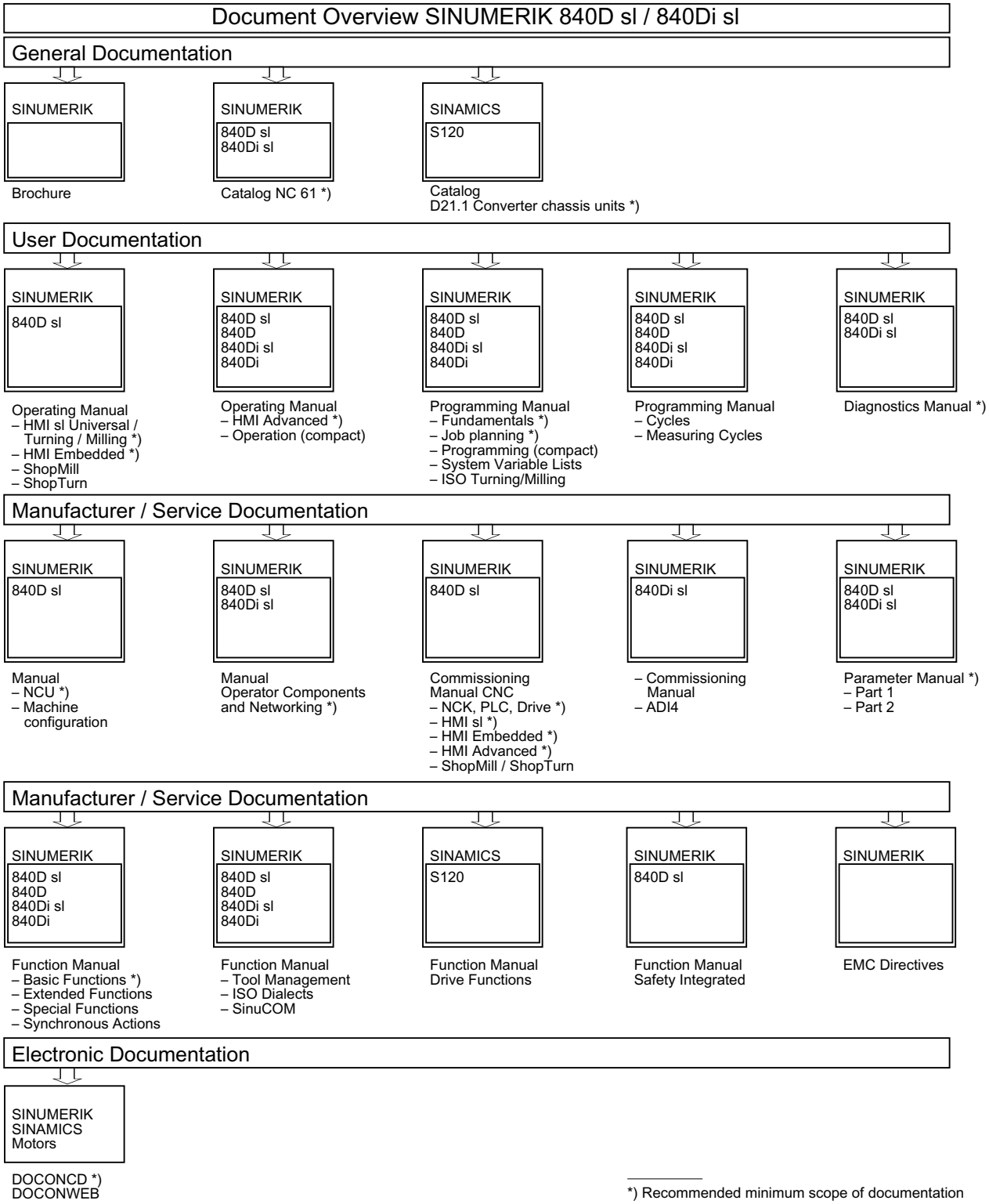
Parameter Manual

Order -No.: 6FC5397-3CP10-3BA0
01/2008

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



*) Recommended minimum scope of documentation