

# Flexium and Flexium+ ENA

## Extended NCK Access

## Programming manual

M00026EN-03

Edition 09.2021

The screenshot displays the Flexium Tools IDE interface for the project 'GROND\_uff2\_1Ch\_V3200'. The 'POUs' pane on the left shows the project structure, including 'ExtComRead', 'ExtComWrite', 'Templates', 'Library Manager', 'Project Information', and 'Project Settings'. The 'Bibliotheksverwalter' (Library Manager) pane on the right lists various libraries and their namespaces, including 'NUM\_MP04 (NUM)', 'NUMENA (NUM)', 'Standard (System)', 'Util (System)', and 'CAA CIA 405 (CAA Technical Workgroup)'. The 'Devices' pane shows a list of devices, including 'Ripari (PRG)', 'ELG (PRG)', 'MFuncDec (PRG)', 'NCK\_1CH (PRG)', 'Puls\_MP04 (PRG)', 'Refrigerante (PRG)', and 'Shut\_Down (PRG)'. The 'Inputs/Outputs' pane shows the 'FUNCTION\_BLOCK\_R\_N' block with its inputs and outputs. The 'FUNCTION\_BLOCK\_R\_Drive\_Param EXTENDS dncbaseRead' block is highlighted, showing its inputs and outputs. The 'R\_Axis\_Meas\_ByAdr' block is also highlighted, showing its inputs and outputs. The code editor at the bottom shows the definition of the 'R\_Axis\_Meas\_ByAdr' function block, which extends 'dncbaseRead' and includes variables for 'byAdresse' and 'di\_Axis\_Meas'.

| Name         | Type                          | Inherited from | Address | Initial | Comment                   |
|--------------|-------------------------------|----------------|---------|---------|---------------------------|
| enable       | BOOL                          | dncbaseRead    |         |         | Request start             |
| CNCAdr       | POINTER TO fbCncBaseInterface | dncbaseRead    |         |         | CNC Address (Flexium_NCK) |
| uiResult     | UINT                          | dncbaseRead    |         |         | 0 = ok; >0 = fault        |
| uiReqResult  | UINT                          | dncbaseRead    |         |         | Return Code of Request    |
| uiRespResult | UINT                          | dncbaseRead    |         |         | Return Code of Response   |
| uiRespCode   | UINT                          | dncbaseRead    |         |         | Report Code of Response   |
| uiRespStatus | UINT                          | dncbaseRead    |         |         | Report Status of Response |
| ResBuf       | ARRAY [0..2048] OF BYTE       | dncbaseRead    |         |         | Buffer of Response        |

```
1 // Request of reading : Axis measurement E900xx by Address
2 FUNCTION_BLOCK R_Axis_Meas_ByAdr EXTENDS dncbaseRead
3 // Inherit the properties of dncbaseRead
4 VAR_INPUT
5     byAdresse:BYTE; //byAdresse 0-31
6 END_VAR
7 VAR_OUTPUT
8     di_Axis_Meas: DINT; // Axis Measure [internal Units]
9 END_VAR
```

---

Despite the care taken in the preparation of this document, NUM cannot guarantee the accuracy of the information it contains and cannot be held responsible for any errors therein, nor for any damage which might result from the use or application of the document.

The physical, technical and functional characteristics of the hardware and software products and the services described in this document are subject to modification and cannot under any circumstances be regarded as contractual.

The programming examples described in this manual are intended for guidance only. They must be specially adapted before they can be used in programs with an industrial application, according to the automated system used and the safety levels required.

**© Copyright NUM 2021**

All rights reserved. No part of this manual may be copied or reproduced in any form or by any means whatsoever, including photographic or magnetic processes. The transcription on an electronic machine of all or part of the contents is forbidden.

**© Copyright NUM 2021 software**

This software is the property of NUM. Each memorized copy of this software sold confers upon the purchaser a non-exclusive licence strictly limited to the use of the said copy. No copy or other form of duplication of this product is authorized.

# Summary

|          |                                                                                     |           |
|----------|-------------------------------------------------------------------------------------|-----------|
| <b>1</b> | <b>Foreword .....</b>                                                               | <b>9</b>  |
| 1.1      | Terminology .....                                                                   | 9         |
| 1.2      | Purpose of the ENA library .....                                                    | 9         |
| 1.3      | Use of the manual .....                                                             | 10        |
| 1.4      | Document revisions .....                                                            | 10        |
| 1.5      | Use of symbology in the manual.....                                                 | 11        |
| 1.6      | NUM Agencies.....                                                                   | 11        |
| <b>2</b> | <b>General.....</b>                                                                 | <b>15</b> |
| 2.1      | Basic functions.....                                                                | 15        |
| 2.2      | Format of the data returned .....                                                   | 17        |
| 2.3      | Error Codes.....                                                                    | 18        |
| <b>3</b> | <b>List of requests.....</b>                                                        | <b>23</b> |
| 3.1      | Read requests.....                                                                  | 23        |
| 3.1.1    | Common read requests .....                                                          | 23        |
| 3.1.2    | Flexium specific read requests .....                                                | 24        |
| 3.1.3    | Flexium+ specific read requests .....                                               | 25        |
| 3.2      | Write requests.....                                                                 | 26        |
| 3.2.1    | Common write requests.....                                                          | 26        |
| 3.2.2    | Flexium specific write requests.....                                                | 26        |
| 3.2.3    | Flexium+ specific write requests.....                                               | 27        |
| <b>4</b> | <b>Programming.....</b>                                                             | <b>31</b> |
| 4.1      | Program structure .....                                                             | 32        |
| <b>5</b> | <b>Read requests .....</b>                                                          | <b>37</b> |
| 5.1      | General requests .....                                                              | 37        |
| 5.1.1    | List of interpolated axes ▶ <a href="#">R_Axis_Drive</a> .....                      | 37        |
| 5.1.2    | Synchronization status in multi channels ▶ <a href="#">R_Channel_Syn</a> .....      | 38        |
| 5.1.3    | Reading of the drives error number ▶ <a href="#">R_Drive_Error</a> .....            | 39        |
| 5.1.4    | Reading of the value of a drive parameter ▶ <a href="#">R_Drive_Param</a> .....     | 40        |
| 5.1.5    | Reading of the content of a drive register ▶ <a href="#">R_Drive_Register</a> ..... | 41        |
| 5.1.6    | List of G codes active in the channel ▶ <a href="#">R_G_Funct</a> .....             | 42        |
| 5.1.7    | System identification ▶ <a href="#">R_Ident_Info</a> .....                          | 46        |
| 5.1.8    | List of L Parameters ▶ <a href="#">R_L_Vars</a> .....                               | 47        |
| 5.1.9    | Reading of a dialog (\$) message ▶ <a href="#">R_Message</a> .....                  | 48        |
| 5.1.10   | Reading of an NCK parameter ▶ <a href="#">R_Nck_Par</a> .....                       | 49        |
| 5.1.11   | Reading of the overrides values ▶ <a href="#">R_Override_All</a> .....              | 50        |
| 5.1.12   | Reading the part program memory size ▶ <a href="#">R_Program_Memory_Size</a> .....  | 51        |
| 5.1.13   | Reading the system serial number ▶ <a href="#">R_Serial_Number</a> .....            | 52        |
| 5.1.14   | Reading a solicited input ▶ <a href="#">R_Solicited_Input</a> .....                 | 53        |
| 5.1.15   | Reading the measure of a spindle ▶ <a href="#">R_Spindle_Meas</a> .....             | 54        |
| 5.1.16   | Reading the tool offset H dimension ▶ <a href="#">R_Tool_H</a> .....                | 55        |
| 5.1.17   | Reading a memory zone directory ▶ <a href="#">R_Directory</a> .....                 | 56        |
| 5.1.18   | Read a part program from NC memory ▶ <a href="#">R_Upload2_PP</a> .....             | 57        |

|          |                                                                                       |            |
|----------|---------------------------------------------------------------------------------------|------------|
| 5.2      | Flexium requests.....                                                                 | 58         |
| 5.2.1    | Reading an axis measure by address ▶ <a href="#">R_Axis_Meas_ByAdr</a> .....          | 58         |
| 5.2.2    | Reading the measure of all axes in a channel ▶ <a href="#">R_Axis_Meas</a> .....      | 59         |
| 5.2.3    | Reading machine data about axes in motion ▶ <a href="#">R_Axis_OM</a> .....           | 60         |
| 5.2.4    | Reading part origin data about axes in motion ▶ <a href="#">R_Axis_OP</a> .....       | 61         |
| 5.2.5    | Reading axes programmed position ▶ <a href="#">R_Axis_Prog</a> .....                  | 62         |
| 5.2.6    | Reading axes reference ▶ <a href="#">R_Axis_Ref</a> .....                             | 63         |
| 5.2.7    | Reading a part program block ▶ <a href="#">R_Block</a> .....                          | 64         |
| 5.2.8    | Reading an axis measure correction (E952xx) ▶ <a href="#">R_Corr_Dyn</a> .....        | 65         |
| 5.2.9    | Reading the part program block in execution ▶ <a href="#">R_Curr_Block</a> .....      | 66         |
| 5.2.10   | Reading current Dat1 value ▶ <a href="#">R_Dat1_Pref</a> .....                        | 67         |
| 5.2.11   | Reading current Dat2 value ▶ <a href="#">R_Dat2_Dec1</a> .....                        | 68         |
| 5.2.12   | Reading current Dat3 value ▶ <a href="#">R_Dat3_Dec3</a> .....                        | 69         |
| 5.2.13   | Reading Drives Test points values ▶ <a href="#">R_Drive_TPValue_V2</a> .....          | 70         |
| 5.2.14   | Reading a set of E8xyyy parameters ▶ <a href="#">R_E8xyyy</a> .....                   | 71         |
| 5.2.15   | Reading a program parameter ▶ <a href="#">R_E_Param</a> .....                         | 72         |
| 5.2.16   | Reading the Interpolation data ▶ <a href="#">R_Interpo</a> .....                      | 73         |
| 5.2.17   | Reading machine origin values ▶ <a href="#">R_Mach_Orig</a> .....                     | 74         |
| 5.2.18   | Reading Maximum Dynamic Travel values ▶ <a href="#">R_Max_Dyn_Travel</a> .....        | 75         |
| 5.2.19   | Reading Minimum Dynamic Travel values ▶ <a href="#">R_Min_Dyn_Travel</a> .....        | 76         |
| 5.2.20   | Reading Maximum Static Travel values ▶ <a href="#">R_Max_Stat_Travel</a> .....        | 77         |
| 5.2.21   | Reading Minimum Static Travel values ▶ <a href="#">R_Min_Stat_Travel</a> .....        | 78         |
| 5.2.22   | Reading NCK detected error ▶ <a href="#">R_Nck_Error</a> .....                        | 79         |
| 5.2.23   | Reading amount of NCK User memory free ▶ <a href="#">R_MemFree</a> .....              | 80         |
| 5.2.24   | Reading Part program execution tree ▶ <a href="#">R_Program_Tree</a> .....            | 81         |
| 5.2.25   | Reading Spindle information ▶ <a href="#">R_Spindle_Complete</a> .....                | 82         |
| 5.2.26   | Reading Spindle Speed ▶ <a href="#">R_Spindle_Speed</a> .....                         | 83         |
| 5.2.27   | Reading Tool offsets data ▶ <a href="#">R_Tool</a> .....                              | 84         |
| 5.2.28   | Read a part program from NC memory ▶ <a href="#">R_Upload_PP</a> .....                | 85         |
| 5.3      | Flexium+ requests .....                                                               | 86         |
| 5.3.1    | Reading an axis measure by address ▶ <a href="#">R_Axis_Meas_ByAdr_Plus</a> .....     | 86         |
| 5.3.2    | Reading the measure of all axes in a channel ▶ <a href="#">R_Axis_Meas_Plus</a> ..... | 87         |
| 5.3.3    | Reading machine data about axes in motion ▶ <a href="#">R_Axis_OM_Plus</a> .....      | 88         |
| 5.3.4    | Reading part origin data about axes in motion ▶ <a href="#">R_Axis_OP_Plus</a> .....  | 89         |
| 5.3.5    | Reading axes programmed position ▶ <a href="#">R_Axis_Prog_Plus</a> .....             | 90         |
| 5.3.6    | Reading axes reference ▶ <a href="#">R_Axis_Ref_Plus</a> .....                        | 91         |
| 5.3.7    | Reading a part program block ▶ <a href="#">R_Block_Plus</a> .....                     | 92         |
| 5.3.8    | Reading an axis measure correction (E952xx) ▶ <a href="#">R_Corr_Dyn_Plus</a> .....   | 93         |
| 5.3.9    | Reading the part program block in execution ▶ <a href="#">R_Curr_Block_Plus</a> ..... | 94         |
| 5.3.10   | Reading current Dat1 value ▶ <a href="#">R_Dat1_Pref_Plus</a> .....                   | 95         |
| 5.3.11   | Reading current Dat2 value ▶ <a href="#">R_Dat2_Dec1_Plus</a> .....                   | 96         |
| 5.3.12   | Reading current Dat3 value ▶ <a href="#">R_Dat3_Dec3_Plus</a> .....                   | 97         |
| 5.3.13   | Reading Drives Test points values ▶ <a href="#">R_Drive_TPValue_V2_Plus</a> .....     | 98         |
| 5.3.14   | Reading a set of E8xyyy parameters ▶ <a href="#">R_E8xyyy_Plus</a> .....              | 99         |
| 5.3.15   | Reading a program parameter ▶ <a href="#">R_E_Param_Plus</a> .....                    | 100        |
| 5.3.16   | Reading the Interpolation data ▶ <a href="#">R_Interpo_Plus</a> .....                 | 101        |
| 5.3.17   | Reading machine origin values ▶ <a href="#">R_Mach_Orig_Plus</a> .....                | 102        |
| 5.3.18   | Reading Maximum Dynamic Travel values ▶ <a href="#">R_Max_Dyn_Travel_Plus</a> .....   | 103        |
| 5.3.19   | Reading Minimum Dynamic Travel values ▶ <a href="#">R_Min_Dyn_Travel_Plus</a> .....   | 104        |
| 5.3.20   | Reading Maximum Static Travel values ▶ <a href="#">R_Max_Stat_Travel_Plus</a> .....   | 105        |
| 5.3.21   | Reading Minimum Static Travel values ▶ <a href="#">R_Min_Stat_Travel_Plus</a> .....   | 106        |
| 5.3.22   | Reading NCK detected error ▶ <a href="#">R_Nck_Error_Plus</a> .....                   | 107        |
| 5.3.23   | Reading Part program execution tree ▶ <a href="#">R_Program_Tree_Plus</a> .....       | 108        |
| 5.3.24   | Reading Spindle information ▶ <a href="#">R_Spindle_Complete_Plus</a> .....           | 109        |
| 5.3.25   | Reading Spindle information ▶ <a href="#">R_Spindle_Complete_Plus_V2</a> .....        | 110        |
| 5.3.26   | Reading Tool offsets data ▶ <a href="#">R_Tool_Plus</a> .....                         | 111        |
| <b>6</b> | <b>Write requests .....</b>                                                           | <b>115</b> |
| 6.1      | General requests .....                                                                | 115        |
| 6.1.1    | Downloading a part program ▶ <a href="#">W_Download2_PP</a> .....                     | 115        |

|          |                                                                                     |            |
|----------|-------------------------------------------------------------------------------------|------------|
| 6.1.2    | Downloading a part program in Drip Feed mode ▶ <a href="#">W_Download2_DripFeed</a> | 116        |
| 6.1.3    | Disable Drip Feed mode ▶ <a href="#">W_DripFeed_Disable</a>                         | 117        |
| 6.1.4    | Erasing an existing part program ▶ <a href="#">W_Delete_File</a>                    | 118        |
| 6.1.5    | Writing drive parameters in Flash memory ▶ <a href="#">W_Drive_Flash</a>            | 119        |
| 6.1.6    | Setting the drive in Halt ▶ <a href="#">W_Drive_Halt</a>                            | 120        |
| 6.1.7    | Writing drive parameters ▶ <a href="#">W_Drive_Param</a>                            | 121        |
| 6.1.8    | Writing drive parameters ▶ <a href="#">W_Drive_Param_Recursive</a>                  | 122        |
| 6.1.9    | Setting the drive in run ▶ <a href="#">W_Drive_Run</a>                              | 124        |
| 6.1.10   | Answering to a solicited input ▶ <a href="#">W_Solicited_Input</a>                  | 125        |
| 6.1.11   | Writing tool H data ▶ <a href="#">W_Tool_H</a>                                      | 126        |
| 6.2      | Flexium requests                                                                    | 127        |
| 6.2.1    | Writing a part program block ▶ <a href="#">W_Block</a>                              | 127        |
| 6.2.2    | Writing an axis measure correction ▶ <a href="#">W_Corr_Dyn</a>                     | 128        |
| 6.2.3    | Writing DAT1 values ▶ <a href="#">W_Dat1_Pref</a>                                   | 129        |
| 6.2.4    | Writing DAT2 values ▶ <a href="#">W_Dat2_Dec1</a>                                   | 130        |
| 6.2.5    | Writing DAT3 values ▶ <a href="#">W_Dat3_Dec3</a>                                   | 131        |
| 6.2.6    | Writing an E8... parameter value ▶ <a href="#">W_E8xyyy</a>                         | 132        |
| 6.2.7    | Writing an E5 or E6 parameter value ▶ <a href="#">W_EParam_5_6</a>                  | 133        |
| 6.2.8    | Writing the home position (P16) ▶ <a href="#">W_Mach_Orig</a>                       | 134        |
| 6.2.9    | Writing Maximum Dynamic Travel values ▶ <a href="#">W_Max_Dyn_Travel</a>            | 135        |
| 6.2.10   | Writing Maximum Dynamic Travel selected ▶ <a href="#">W_Max_Dyn_Travel_2</a>        | 136        |
| 6.2.11   | Writing Maximum Static Travel values ▶ <a href="#">W_Max_Stat_Travel</a>            | 137        |
| 6.2.12   | Writing Minimum Dynamic Travel values ▶ <a href="#">W_Min_Dyn_Travel</a>            | 138        |
| 6.2.13   | Writing Minimum Dynamic Travel selected ▶ <a href="#">W_Min_Dyn_Travel_2</a>        | 139        |
| 6.2.14   | Writing Minimum Static Travel values ▶ <a href="#">W_Min_Stat_Travel</a>            | 140        |
| 6.2.15   | Writing an NCK parameter ▶ <a href="#">W_NCK_Par</a>                                | 141        |
| 6.2.16   | Writing a tool offset ▶ <a href="#">W_Tool</a>                                      | 142        |
| 6.3      | Flexium+ requests                                                                   | 143        |
| 6.3.1    | Writing a part program block ▶ <a href="#">W_Block_Plus</a>                         | 143        |
| 6.3.2    | Writing an axis measure correction ▶ <a href="#">W_Corr_Dyn_Plus</a>                | 144        |
| 6.3.3    | Writing DAT1 values ▶ <a href="#">W_Dat1_Pref_Plus</a>                              | 145        |
| 6.3.4    | Writing DAT2 values ▶ <a href="#">W_Dat2_Dec1_Plus</a>                              | 146        |
| 6.3.5    | Writing DAT3 values ▶ <a href="#">W_Dat3_Dec3_Plus</a>                              | 147        |
| 6.3.6    | Writing an E8... parameter value ▶ <a href="#">W_E8xyyy_Plus</a>                    | 148        |
| 6.3.7    | Writing an E5 or E6 parameter value ▶ <a href="#">W_E_Param_5_6_Plus</a>            | 149        |
| 6.3.8    | Writing the home position (P16) ▶ <a href="#">W_Mach_Orig_Plus</a>                  | 150        |
| 6.3.9    | Writing Maximum Dynamic Travel values ▶ <a href="#">W_Max_Dyn_Travel_Plus</a>       | 151        |
| 6.3.10   | Writing Maximum Dynamic Travel selected ▶ <a href="#">W_Max_Dyn_Travel_Plus_2</a>   | 152        |
| 6.3.11   | Writing Maximum Static Travel values ▶ <a href="#">W_Max_Stat_Travel_Plus</a>       | 153        |
| 6.3.12   | Writing Minimum Dynamic Travel values ▶ <a href="#">W_Min_Dyn_Travel_Plus</a>       | 154        |
| 6.3.13   | Writing Minimum Dynamic Travel selected ▶ <a href="#">W_Min_Dyn_Travel_Plus_2</a>   | 155        |
| 6.3.14   | Writing Minimum Static Travel values ▶ <a href="#">W_Min_Stat_Travel_Plus</a>       | 156        |
| 6.3.15   | Execute motor brake test ▶ <a href="#">W_Motor_Brake_Test</a>                       | 157        |
| 6.3.16   | Writing an NCK parameter ▶ <a href="#">W_NCK_Par_Plus</a>                           | 158        |
| 6.3.17   | Writing a tool offset ▶ <a href="#">W_Tool_Plus</a>                                 | 159        |
| <b>A</b> | <b>Appendix</b>                                                                     | <b>163</b> |
| A.1      | Runtime Error Codes                                                                 | 163        |
| A.2      | Drive parameters specificities                                                      | 164        |
| A.2.1    | Case NUMDrive C Sw version 3.5x, 4.5x parameters conversions table                  | 165        |
| A.2.2    | Dependencies table                                                                  | 173        |
| A.2.3    | NUMDrive C parameters format and limits                                             | 175        |
| A.3      | NUMDrive X parameters Conversion rules, Dependencies, Limits                        | 183        |
| A.3.1    | Conversion rules table                                                              | 183        |
| A.3.2    | Dependencies table                                                                  | 194        |
| A.3.3    | NUMDrive X parameters format and limits                                             | 196        |
| A.4      | NCK parameters                                                                      | 205        |
| A.4.1    | NCK Parameters for Flexium+                                                         | 205        |
| A.4.2    | NCK Parameters for Flexium                                                          | 206        |

---

Page deliberately empty

# Summary

|          |                                     |          |
|----------|-------------------------------------|----------|
| <b>1</b> | <b>Foreword .....</b>               | <b>9</b> |
| 1.1      | Terminology .....                   | 9        |
| 1.2      | Purpose of the ENA library .....    | 9        |
| 1.3      | Use of the manual .....             | 10       |
| 1.4      | Document revisions .....            | 10       |
| 1.5      | Use of symbology in the manual..... | 11       |
| 1.6      | NUM Agencies .....                  | 11       |

1

2

3

4

5

6

A

---

Page deliberately empty

---

# 1 Foreword

---

## 1.1 Terminology

---

ENA stands for Extended NCK Access.

It extends the previous DNC1000 functionality allowing exchanges between PLC and NCK of numerous data which are not part of the standard exchange area.

It is provided as a library integrated into Flexium Tools.

## 1.2 Purpose of the ENA library

---

The aim of this built-in library is to provide an easy access to all functions without having to deal with request formatting.

This library is dedicated to the Flexium PLC and is managed via Flexium Tools.

This library is an option of the Flexium system:

- Option code: *FXSW282124 Extended NCK access*
- *# PLC licences*
- *# FXSW282124 Extended NCK access*

---

## 1.3 Use of the manual

---

This manual is intended to provide the user with all the information required to use the ENA library in a Flexium application.

Previous knowledge of PLC programming with Flexium Tools is required, in particular integration and use of a library.

## 1.4 Document revisions

---

| Date      | Index | Reason for revision                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                               |
|-----------|-------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 05 - 2011 | 00    | Document creation conform with Flexium Tools version 3.3.0.0                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                      |
| 12 - 2012 | 01    | - Chapter 6.3 Read NUMDrive test points: new procedure and user example.<br>- Deleted section Appendix 4.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                         |
| 11 - 2016 | 02    | - Introduced separated Read requests / Write requests informations for Flexium and Flexium+.<br>- Conform with Flexium Flexium system 4.0.30.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                     |
| 09 - 2021 | 03    | Conform with Flexium Flexium system 4.1.30.00<br><br>- Error codes - These codes are returned when invoking the function. Any value different of 0 (answer ready) or 99 (answer pending) must be managed. 2.3<br>- Image update chapter 3 - chapter 5<br>- Image and text update: ► <a href="#">R_Drive_Error</a> 5.1.3 - ► <a href="#">R_MemFree</a> 5.2.23 - ► <a href="#">R_Spindle_Speed</a> 5.2.26 - ► <a href="#">R_Upload_PP</a> - 5.2.28 - ► <a href="#">R_Spindle_Complete_Plus_V2</a> 5.3.25 ► <a href="#">R_Tool_Plus</a> 5.3.26<br>- All chapter 6 updated - Write request.<br>- A - Appendix - Update images, tables and information - a.2 - V107* - V108* - V115* - V128* - V129* - V240*- V241*- V256*- V257*- V348*- V349*- V481*<br>(*) Applicable to the Mono-Axis drives only. |

---

## 1.5 Use of symbology in the manual

---

| Symbol                                                                            | Meaning of the symbol                                                       |
|-----------------------------------------------------------------------------------|-----------------------------------------------------------------------------|
|  | Important notes on product use.                                             |
|  | You may find this symbol whenever a wrong operation may damage the product. |

---

## 1.6 NUM Agencies

---

The list of NUM agencies is given on <http://www.num.com>

---

Page deliberately empty

# Summary

|          |                                   |           |
|----------|-----------------------------------|-----------|
| <b>2</b> | <b>General.....</b>               | <b>15</b> |
| 2.1      | Basic functions.....              | 15        |
| 2.2      | Format of the data returned ..... | 17        |
| 2.3      | Error Codes.....                  | 18        |

1

2

3

4

5

6

A

---

Page deliberately empty

## 2 General

### 2.1 Basic functions

All ENA functions are defined in a PLC library. This library is not installed as standard and must therefore be included in your project should you need any of these functions.

An ENA action is done in three steps:

- Posting a request to the CNC
- Reading the answer when it becomes available
- Clearing the request.

Those steps are handled automatically by each function; there is no need to program them. As a general rule, an ENA function should be programmed in a specific task (50 to 100 ms).

An instance of the function is invoked and enabled during a particular scan and the answer checked at the next scan. If the answer is positive and without error, the function has been executed and the result is available; as soon as the function is posted the answer code is 0x99 (wait) meaning that the request is correctly taken into account.

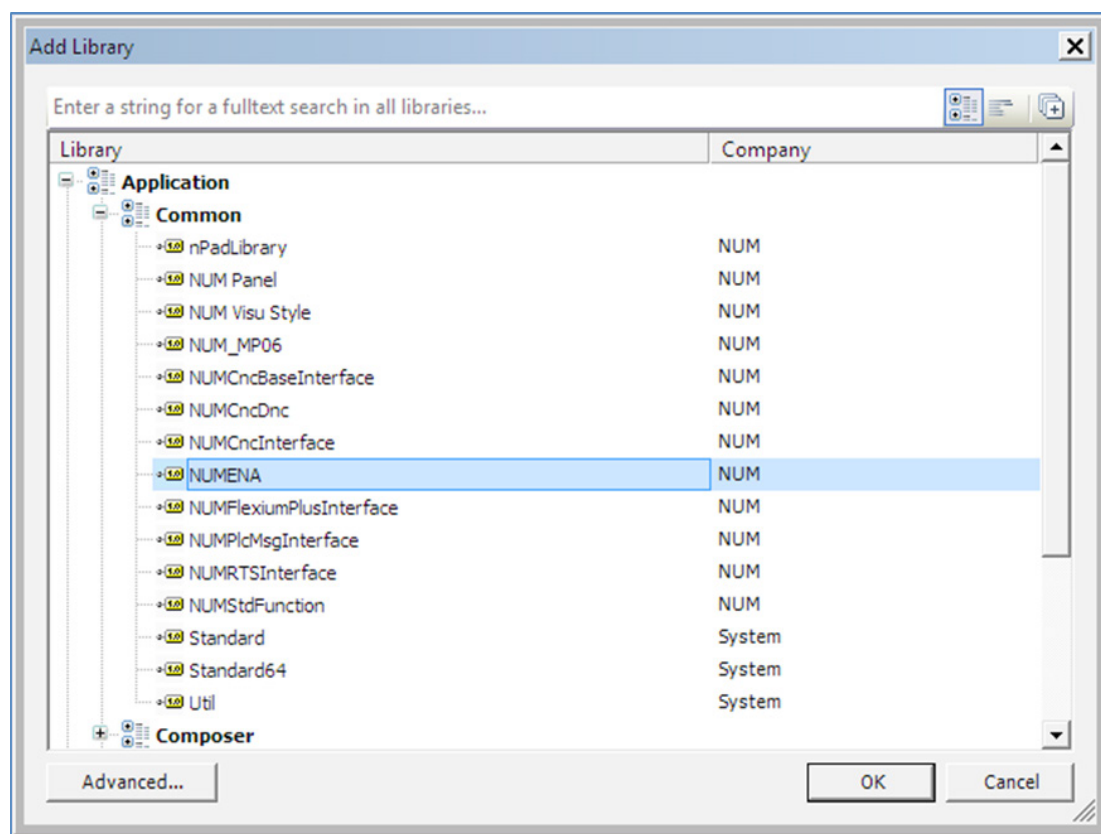


Figure 1 : ENA Library

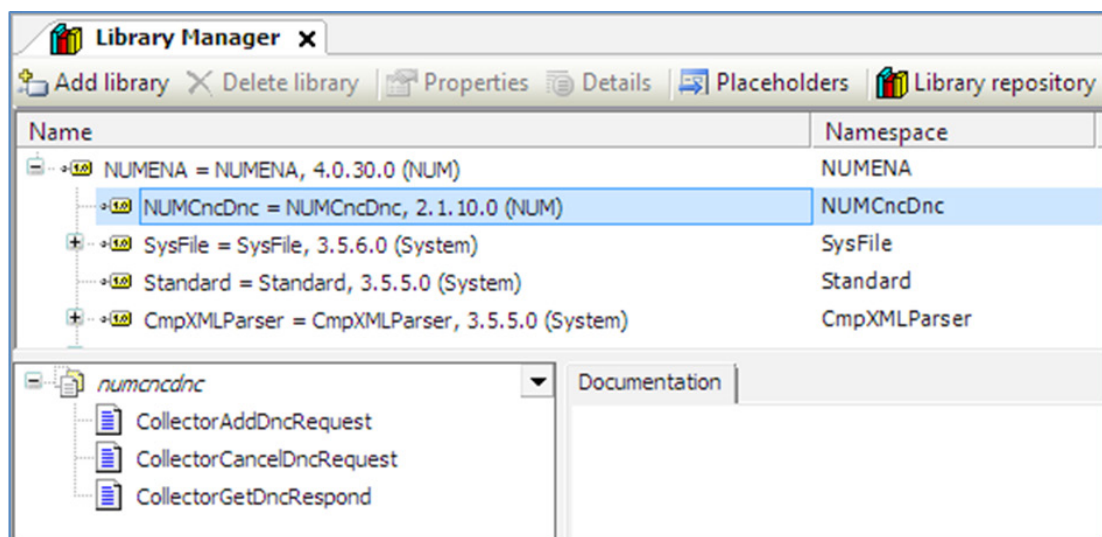


Figure 2 : The three steps of a function (for information)

## 2.2 Format of the data returned

| Type          | Lower limit            | Upper limit            | Size       | Prefix | Remark        |
|---------------|------------------------|------------------------|------------|--------|---------------|
| <b>BOOL</b>   | FALSE                  | TRUE                   | 1 (8) bits | bo     |               |
| <b>BYTE</b>   | 0                      | 255                    | 8 bits     | by     |               |
| <b>WORD</b>   | 0                      | 65535                  | 16 bits    | wo     |               |
| <b>DWORD</b>  | 0                      | 4294967295             | 32 bits    | dw     |               |
| <b>LWORD</b>  | 0                      | 2 <sup>64</sup>        | 64 bits    | lw     |               |
| <b>SINT</b>   | -128                   | 127                    | 8 bits     | si     |               |
| <b>USINT</b>  | 0                      | 255                    | 8 bits     | us     |               |
| <b>INT</b>    | -32768                 | 32767                  | 16 bits    | in     |               |
| <b>UINT</b>   | 0                      | 65535                  | 16 bits    | ui     |               |
| <b>DINT</b>   | -2147483648            | 2147483647             | 32 bits    | di     |               |
| <b>LINT</b>   | -2 <sup>63</sup>       | 2 <sup>63</sup> -1     | 64 bits    | li     |               |
| <b>UDINT</b>  | 0                      | 4294967295             | 32 bits    | ud     |               |
| <b>ULINT</b>  | 0                      | 2 <sup>64</sup>        | 64 bits    | ul     |               |
| <b>REAL</b>   | ±10 <sup>-37</sup>     | ±10 <sup>+38</sup>     | 32 bits    | re     |               |
| <b>LREAL</b>  | ±10 <sup>-307</sup>    | ±10 <sup>+308</sup>    | 64 bits    | lr     |               |
| <b>STRING</b> | 1                      | 255                    | bytes      | st     | String        |
| <b>TIME</b>   | T#0                    | T#12h34m15s10ms        | 32 bits    | ti     |               |
| <b>TOD</b>    | TOD#0                  | TOD#23:59:59           | 32 bits    | to     | Time_of_Day   |
| <b>DATE</b>   | DATE#1970-01-01        | DATE#1996-05-06        | 32 bits    | da     |               |
| <b>DT</b>     | DT#1970-01-01-00:00:00 | DT#1996-05-06-15:36:30 | 32 bits    | dt     | Date and Time |

## 2.3 Error Codes

These codes are returned when invoking the function. Any value different of 0 (answer ready) or 99 (answer pending) must be managed.

The error code `uiResult` allows for checking if any error is present,

```
IF uiResult = 0 THEN..
```

`uiResult` is made up of four return values:

```
uiResult:= iReqResult + (iRespResult * 10) + (Resbuf [2] * 100) + (Resbuf [4] * 1000);
```

If `uiResult` is not zero, the following variables can be used to find the type of error.

|                           |            |                             |                        |
|---------------------------|------------|-----------------------------|------------------------|
| <code>uiReqResult</code>  |            | Response on request sending | CollectorAddDncRequest |
| <code>uiRespResult</code> |            | Response on data return     | CollectorGetDncRespond |
| <code>uiRespCode</code>   | Resbuf [2] | Response Code               | Global answer          |
| <code>uiRespStatus</code> | Resbuf [4] | Status code                 | Set of answer          |

| <code>iReqResult</code> | Error name       | Error           | Remark                    |
|-------------------------|------------------|-----------------|---------------------------|
| 0                       | ERROK            | Transmission OK |                           |
| 1                       | BUFFERFULL       | Buffer full     | Resend the request        |
| 3                       | BUFFERERR        | Buffer error    | Message + fix application |
| 4                       | NCKNOTREADY      | CNC not ready   | Message + resend request  |
| 6                       | UNKNOWNCKADDRESS |                 | Error + fix application   |
| 9                       |                  | No answer       | Wait                      |
| 50                      |                  | No license      | Activate ENA option       |

| <code>iRespResult</code> | Error name            | Error                           | Remark                        |
|--------------------------|-----------------------|---------------------------------|-------------------------------|
| 0                        | ERROK                 | Read OK                         |                               |
| 2                        | RESPONDNOTREADY       | Answer not yet ready            | Read again the answer         |
| 4                        | NCKNOTREADY           | CNC not ready                   | Message + resend request      |
| 5                        | REQUESTNUMBERNOTREADY | Request not ready               | Cancel and resend the request |
| 6                        |                       | No message at this port number  |                               |
| 7                        |                       | Buffer too small for the answer |                               |
| 8                        |                       | Invalid Line Number             |                               |
| 9                        |                       | No answer                       | Wait                          |

| <code>iRespCode</code> | Error name | Error                    | Remark |
|------------------------|------------|--------------------------|--------|
| 0                      | ERROK      | Answer OK                |        |
| 1                      |            | Answer OK but partial    |        |
| 2                      |            | Variable access conflict |        |
| 3                      |            | Bad request parameter    |        |
| 9                      |            | No answer                | Wait   |

| <code>iRespStatus</code> | Error name | Error                 | Remark |
|--------------------------|------------|-----------------------|--------|
| 0                        | ERROK      | Transmission OK       |        |
| 1                        |            | Protected access      |        |
| 2                        |            | Non existing element  |        |
| 3                        |            | Limit error           |        |
| 4                        |            | Currently meaningless |        |
| 9                        |            | No answer             | Wait   |

Page deliberately empty

|   |
|---|
| 1 |
| 2 |
| 3 |
| 4 |
| 5 |
| 6 |
| A |

---

Page deliberately empty

## Summary

|          |                                       |           |
|----------|---------------------------------------|-----------|
| <b>3</b> | <b>List of requests.....</b>          | <b>23</b> |
| 3.1      | Read requests.....                    | 23        |
| 3.1.1    | Common read requests .....            | 23        |
| 3.1.2    | Flexium specific read requests .....  | 24        |
| 3.1.3    | Flexium* specific read requests ..... | 25        |
| 3.2      | Write requests.....                   | 26        |
| 3.2.1    | Common write requests.....            | 26        |
| 3.2.2    | Flexium specific write requests.....  | 26        |
| 3.2.3    | Flexium* specific write requests..... | 27        |

1

2

3

4

5

6

A

---

Page deliberately empty

## 3 List of requests

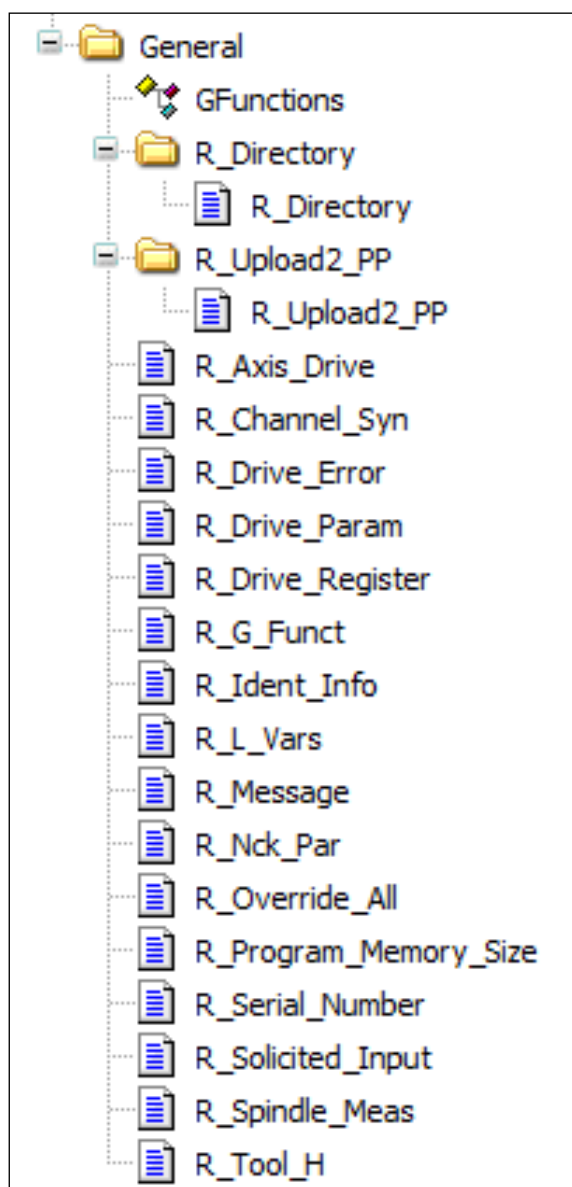
The library contains Read as well as Write requests grouped in three categories:

- Common requests
- Flexium specific requests
- Flexium+ specific requests.

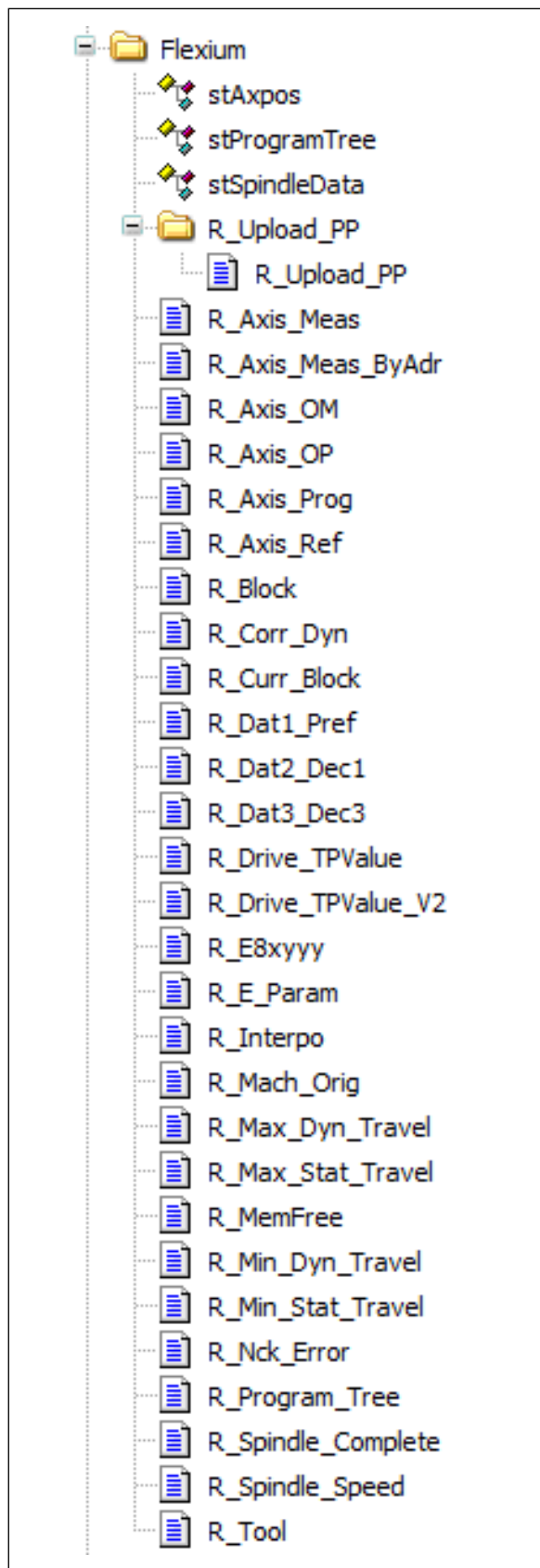
In addition several data structure are defined.

### 3.1 Read requests

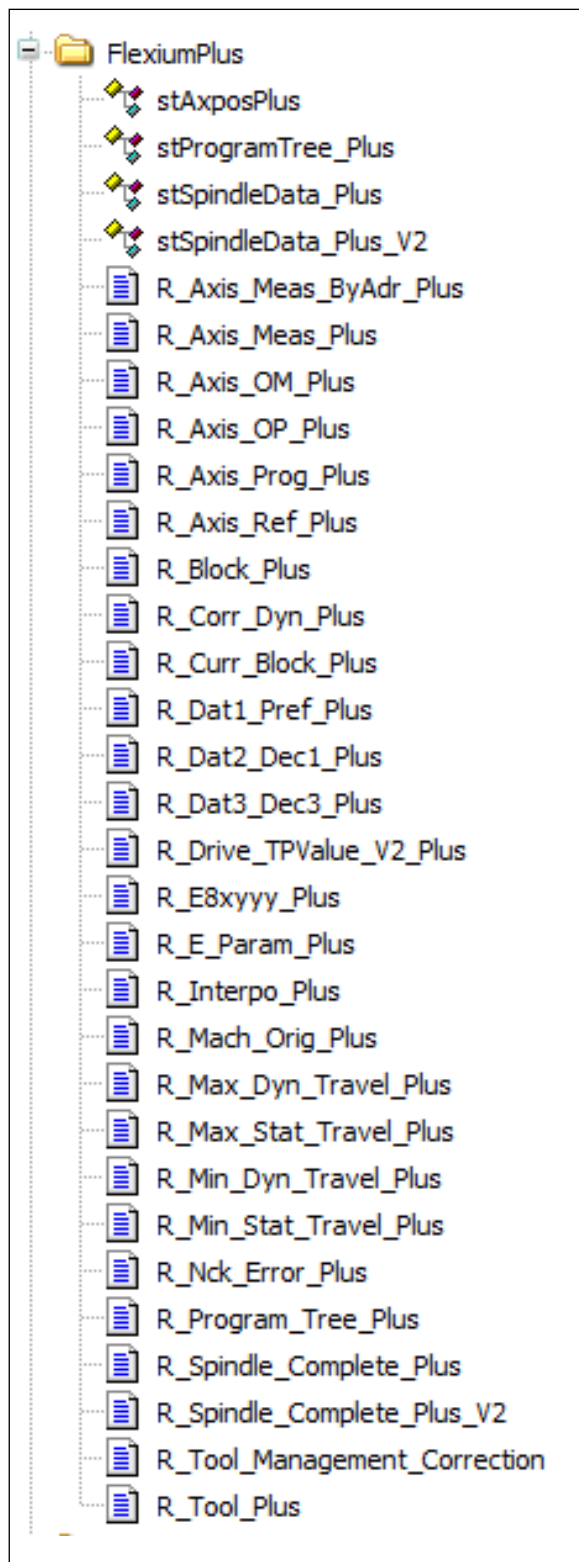
#### 3.1.1 Common read requests



### 3.1.2 Flexium specific read requests



### 3.1.3 Flexium+ specific read requests

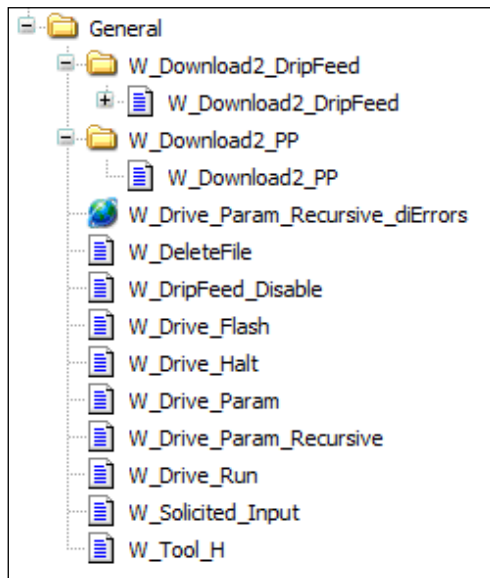


---

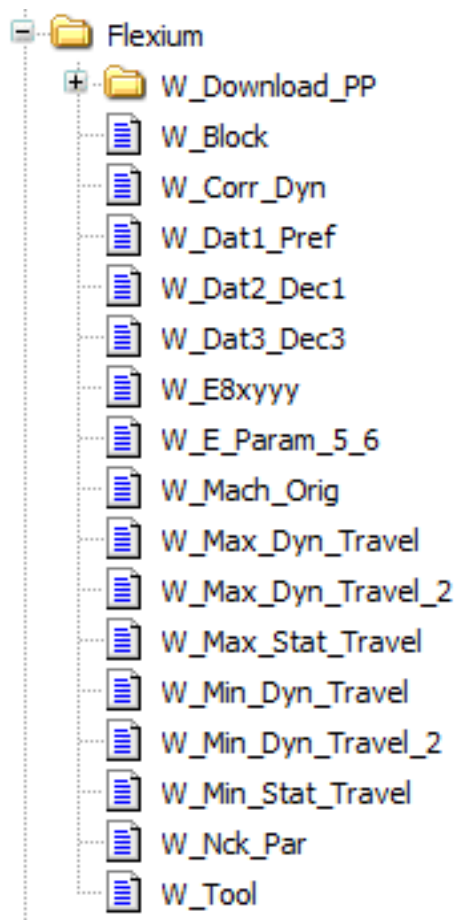
## 3.2 Write requests

---

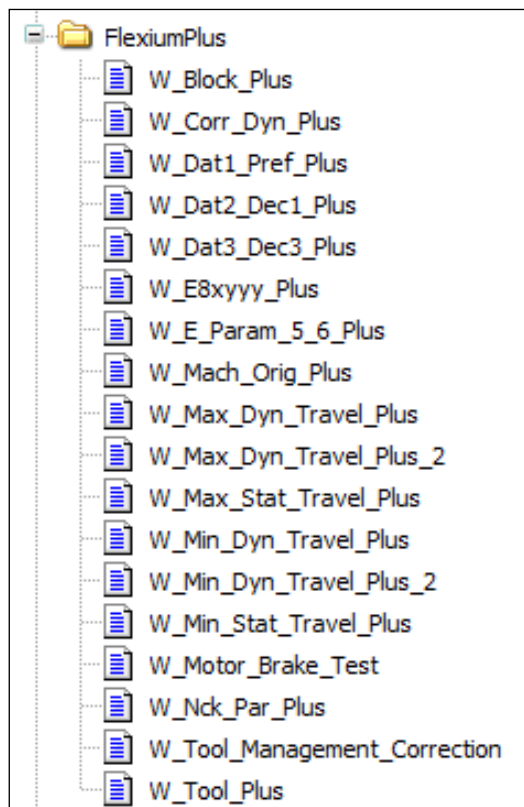
### 3.2.1 Common write requests



### 3.2.2 Flexium specific write requests



### 3.2.3 Flexium+ specific write requests



---

Page deliberately empty

# Summary

|          |                          |           |
|----------|--------------------------|-----------|
| <b>4</b> | <b>Programming .....</b> | <b>31</b> |
| 4.1      | Program structure .....  | 32        |

1

2

3

4

5

6

A

---

Page deliberately empty

## 4 Programming

The use of ENA is quite simple as all request are build on the same model.

This is generally done in the following steps:

- A specific task should be created for all ENA requests:  
It is in most cases not necessary to have a period lower than 50ms and this task priority should be lower than the PLC\_PRG main PLC tasks.
- In a POU of this task, declare an instance of the request to execute.
- Fill in the required parameters. Pay particular attention to the CNCAddress to avoid any exception.
- Use the enable flag to launch the request either one shot or permanently according to the context.
- Check the return code; handle the errors if they are some or manage the answer when this code is 0. Once the return code is 0 the request is terminated and the enable can be cleared.

P.S. Some request (super requests) require several cycles to be executed.

These requests are relative to the part program memory management. The number of cycles can vary according to the size of data transmitted. Explanations will be given when studying those super requests.



The following examples are given only to demonstrate the function. They are simplified on purpose and in particular the errors are not handled. In a real application it will be necessary to manage the possible errors in order to prevent a malfunction.

### Return codes

The different possibility of return codes are given at chapter 2.3.

In the following example the codes handled will be:

- 99 : Answer pending
- 0 : Answer available or no request.

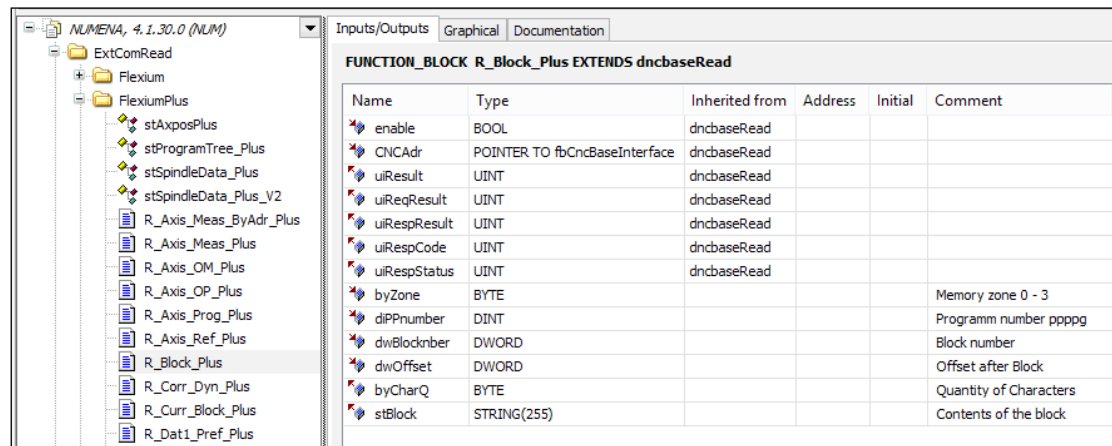
## 4.1 Program structure

The following requests were all launched one by one thanks to a specific task of 60ms period and a priority of 10. This task contains nothing else than the request to check.

In those examples, the flag *xEnable* is a Boolean which is SET by an action. The clearing of this flag is done in the example itself. As already stated the return code is not checked against an error.

This check must be made for real cases to prevent any lock-up or unwanted machine reaction. In general the *VAR* definition part of these examples only declares the mandatory variables in order to be able to run the said example. More variables may exist.

For the full list refer to the documentation tab of the request in the *Library Manager editor*.



The screenshot shows the Library Manager editor with the function block **R\_Block\_Plus** selected. The left pane displays a tree structure of the project, including **ExtComRead** and **Flexium**. The right pane shows the **FUNCTION\_BLOCK R\_Block\_Plus EXTENDS dncbaseRead** with a table of its variables.

| Name         | Type                          | Inherited from | Address | Initial | Comment                |
|--------------|-------------------------------|----------------|---------|---------|------------------------|
| enable       | BOOL                          | dncbaseRead    |         |         |                        |
| CNCAdr       | POINTER TO fbCncBaseInterface | dncbaseRead    |         |         |                        |
| uIResult     | UINT                          | dncbaseRead    |         |         |                        |
| uIReqResult  | UINT                          | dncbaseRead    |         |         |                        |
| uIRespResult | UINT                          | dncbaseRead    |         |         |                        |
| uIRespCode   | UINT                          | dncbaseRead    |         |         |                        |
| uIRespStatus | UINT                          | dncbaseRead    |         |         |                        |
| byZone       | BYTE                          |                |         |         | Memory zone 0 - 3      |
| dIPpNumber   | DINT                          |                |         |         | Programm number ppppg  |
| dwBlocknber  | DWORD                         |                |         |         | Block number           |
| dwOffset     | DWORD                         |                |         |         | Offset after Block     |
| byCharQ      | BYTE                          |                |         |         | Quantity of Characters |
| stBlock      | STRING(255)                   |                |         |         | Contents of the block  |

Page deliberately empty

|   |
|---|
| 1 |
| 2 |
| 3 |
| 4 |
| 5 |
| 6 |
| A |

---

Page deliberately empty

## Summary

|          |                                                                                     |           |
|----------|-------------------------------------------------------------------------------------|-----------|
| <b>5</b> | <b>Read requests .....</b>                                                          | <b>37</b> |
| 5.1      | General requests .....                                                              | 37        |
| 5.1.1    | List of interpolated axes ▶ <a href="#">R_Axis_Drive</a> .....                      | 37        |
| 5.1.2    | Synchronization status in multi channels ▶ <a href="#">R_Channel_Syn</a> .....      | 38        |
| 5.1.3    | Reading of the drives error number ▶ <a href="#">R_Drive_Error</a> .....            | 39        |
| 5.1.4    | Reading of the value of a drive parameter ▶ <a href="#">R_Drive_Param</a> .....     | 40        |
| 5.1.5    | Reading of the content of a drive register ▶ <a href="#">R_Drive_Register</a> ..... | 41        |
| 5.1.6    | List of G codes active in the channel ▶ <a href="#">R_G_Funct</a> .....             | 42        |
| 5.1.7    | System identification ▶ <a href="#">R_Ident_Info</a> .....                          | 46        |
| 5.1.8    | List of L Parameters ▶ <a href="#">R_L_Vars</a> .....                               | 47        |
| 5.1.9    | Reading of a dialog (\$) message ▶ <a href="#">R_Message</a> .....                  | 48        |
| 5.1.10   | Reading of an NCK parameter ▶ <a href="#">R_Nck_Par</a> .....                       | 49        |
| 5.1.11   | Reading of the overrides values ▶ <a href="#">R_Override_All</a> .....              | 50        |
| 5.1.12   | Reading the part program memory size ▶ <a href="#">R_Program_Memory_Size</a> .....  | 51        |
| 5.1.13   | Reading the system serial number ▶ <a href="#">R_Serial_Number</a> .....            | 52        |
| 5.1.14   | Reading a solicited input ▶ <a href="#">R_Solicited_Input</a> .....                 | 53        |
| 5.1.15   | Reading the measure of a spindle ▶ <a href="#">R_Spindle_Meas</a> .....             | 54        |
| 5.1.16   | Reading the tool offset H dimension ▶ <a href="#">R_Tool_H</a> .....                | 55        |
| 5.1.17   | Reading a memory zone directory ▶ <a href="#">R_Directory</a> .....                 | 56        |
| 5.1.18   | Read a part program from NC memory ▶ <a href="#">R_Upload2_PP</a> .....             | 57        |
| 5.2      | Flexium requests.....                                                               | 58        |
| 5.2.1    | Reading an axis measure by address ▶ <a href="#">R_Axis_Meas_ByAdr</a> .....        | 58        |
| 5.2.2    | Reading the measure of all axes in a channel ▶ <a href="#">R_Axis_Meas</a> .....    | 59        |
| 5.2.3    | Reading machine data about axes in motion ▶ <a href="#">R_Axis_OM</a> .....         | 60        |
| 5.2.4    | Reading part origin data about axes in motion ▶ <a href="#">R_Axis_OP</a> .....     | 61        |
| 5.2.5    | Reading axes programmed position ▶ <a href="#">R_Axis_Prog</a> .....                | 62        |
| 5.2.6    | Reading axes reference ▶ <a href="#">R_Axis_Ref</a> .....                           | 63        |
| 5.2.7    | Reading a part program block ▶ <a href="#">R_Block</a> .....                        | 64        |
| 5.2.8    | Reading an axis measure correction (E952xx) ▶ <a href="#">R_Corr_Dyn</a> .....      | 65        |
| 5.2.9    | Reading the part program block in execution ▶ <a href="#">R_Curr_Block</a> .....    | 66        |
| 5.2.10   | Reading current Dat1 value ▶ <a href="#">R_Dat1_Pref</a> .....                      | 67        |
| 5.2.11   | Reading current Dat2 value ▶ <a href="#">R_Dat2_Dec1</a> .....                      | 68        |
| 5.2.12   | Reading current Dat3 value ▶ <a href="#">R_Dat3_Dec3</a> .....                      | 69        |
| 5.2.13   | Reading Drives Test points values ▶ <a href="#">R_Drive_TPValue_V2</a> .....        | 70        |
| 5.2.14   | Reading a set of E8xyyy parameters ▶ <a href="#">R_E8xyyy</a> .....                 | 71        |
| 5.2.15   | Reading a program parameter ▶ <a href="#">R_E_Param</a> .....                       | 72        |
| 5.2.16   | Reading the Interpolation data ▶ <a href="#">R_Interpo</a> .....                    | 73        |
| 5.2.17   | Reading machine origin values ▶ <a href="#">R_Mach_Orig</a> .....                   | 74        |
| 5.2.18   | Reading Maximum Dynamic Travel values ▶ <a href="#">R_Max_Dyn_Travel</a> .....      | 75        |
| 5.2.19   | Reading Minimum Dynamic Travel values ▶ <a href="#">R_Min_Dyn_Travel</a> .....      | 76        |
| 5.2.20   | Reading Maximum Static Travel values ▶ <a href="#">R_Max_Stat_Travel</a> .....      | 77        |
| 5.2.21   | Reading Minimum Static Travel values ▶ <a href="#">R_Min_Stat_Travel</a> .....      | 78        |
| 5.2.22   | Reading NCK detected error ▶ <a href="#">R_Nck_Error</a> .....                      | 79        |
| 5.2.23   | Reading amount of NCK User memory free ▶ <a href="#">R_MemFree</a> .....            | 80        |
| 5.2.24   | Reading Part program execution tree ▶ <a href="#">R_Program_Tree</a> .....          | 81        |
| 5.2.25   | Reading Spindle information ▶ <a href="#">R_Spindle_Complete</a> .....              | 82        |
| 5.2.26   | Reading Spindle Speed ▶ <a href="#">R_Spindle_Speed</a> .....                       | 83        |
| 5.2.27   | Reading Tool offsets data ▶ <a href="#">R_Tool</a> .....                            | 84        |
| 5.2.28   | Read a part program from NC memory ▶ <a href="#">R_Upload_PP</a> .....              | 85        |
| 5.3      | Flexium* requests .....                                                             | 86        |

---

|        |                                               |                                              |     |
|--------|-----------------------------------------------|----------------------------------------------|-----|
| 5.3.1  | Reading an axis measure by address            | ▶ <a href="#">R_Axis_Meas_ByAdr_Plus</a>     | 86  |
| 5.3.2  | Reading the measure of all axes in a channel  | ▶ <a href="#">R_Axis_Meas_Plus</a>           | 87  |
| 5.3.3  | Reading machine data about axes in motion     | ▶ <a href="#">R_Axis_OM_Plus</a>             | 88  |
| 5.3.4  | Reading part origin data about axes in motion | ▶ <a href="#">R_Axis_OP_Plus</a>             | 89  |
| 5.3.5  | Reading axes programmed position              | ▶ <a href="#">R_Axis_Prog_Plus</a>           | 90  |
| 5.3.6  | Reading axes reference                        | ▶ <a href="#">R_Axis_Ref_Plus</a>            | 91  |
| 5.3.7  | Reading a part program block                  | ▶ <a href="#">R_Block_Plus</a>               | 92  |
| 5.3.8  | Reading an axis measure correction (E952xx)   | ▶ <a href="#">R_Corr_Dyn_Plus</a>            | 93  |
| 5.3.9  | Reading the part program block in execution   | ▶ <a href="#">R_Curr_Block_Plus</a>          | 94  |
| 5.3.10 | Reading current Dat1 value                    | ▶ <a href="#">R_Dat1_Pref_Plus</a>           | 95  |
| 5.3.11 | Reading current Dat2 value                    | ▶ <a href="#">R_Dat2_Dec1_Plus</a>           | 96  |
| 5.3.12 | Reading current Dat3 value                    | ▶ <a href="#">R_Dat3_Dec3_Plus</a>           | 97  |
| 5.3.13 | Reading Drives Test points values             | ▶ <a href="#">R_Drive_TPValue_V2_Plus</a>    | 98  |
| 5.3.14 | Reading a set of E8xyyy parameters            | ▶ <a href="#">R_E8xyyy_Plus</a>              | 99  |
| 5.3.15 | Reading a program parameter                   | ▶ <a href="#">R_E_Param_Plus</a>             | 100 |
| 5.3.16 | Reading the Interpolation data                | ▶ <a href="#">R_Interpo_Plus</a>             | 101 |
| 5.3.17 | Reading machine origin values                 | ▶ <a href="#">R_Mach_Orig_Plus</a>           | 102 |
| 5.3.18 | Reading Maximum Dynamic Travel values         | ▶ <a href="#">R_Max_Dyn_Travel_Plus</a>      | 103 |
| 5.3.19 | Reading Minimum Dynamic Travel values         | ▶ <a href="#">R_Min_Dyn_Travel_Plus</a>      | 104 |
| 5.3.20 | Reading Maximum Static Travel values          | ▶ <a href="#">R_Max_Stat_Travel_Plus</a>     | 105 |
| 5.3.21 | Reading Minimum Static Travel values          | ▶ <a href="#">R_Min_Stat_Travel_Plus</a>     | 106 |
| 5.3.22 | Reading NCK detected error                    | ▶ <a href="#">R_Nck_Error_Plus</a>           | 107 |
| 5.3.23 | Reading Part program execution tree           | ▶ <a href="#">R_Program_Tree_Plus</a>        | 108 |
| 5.3.24 | Reading Spindle information                   | ▶ <a href="#">R_Spindle_Complete_Plus</a>    | 109 |
| 5.3.25 | Reading Spindle information                   | ▶ <a href="#">R_Spindle_Complete_Plus_V2</a> | 110 |
| 5.3.26 | Reading Tool offsets data                     | ▶ <a href="#">R_Tool_Plus</a>                | 111 |

## 5 Read requests

### 5.1 General requests



Those requests are available for Flexium as well as Flexium+

#### 5.1.1 List of interpolated axes ► [R\\_Axis\\_Drive](#).

This request returns a DINT value.

Each bit corresponds to an axis declared as servo controlled and interpolated.  
(P120 N2 for Flexium+ or P3 for Flexium)

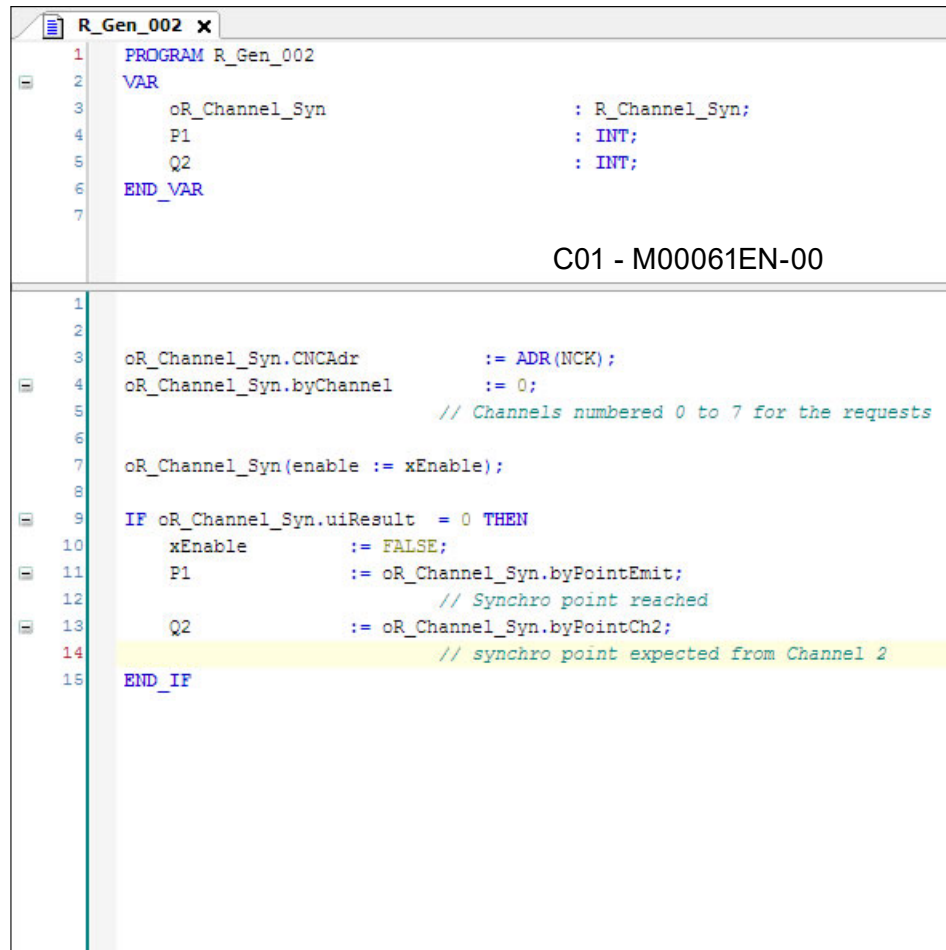
```

1  PROGRAM R_Gen_001
2  VAR
3
4      oR_Axis_Drive          : R_Axis_Drive;
5      answer                 : DINT;
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24
25
26
27
28
29
30
31
32
33
34
35
36
37
38
39
40
41
42
43
44
45
46
47
48
49
50
51
52
53
54
55
56
57
58
59
60
61
62
63
64
65
66
67
68
69
70
71
72
73
74
75
76
77
78
79
80
81
82
83
84
85
86
87
88
89
90
91
92
93
94
95
96
97
98
99
100
101
102
103
104
105
106
107
108
109
110
111
112
113
114
115
116
117
118
119
120
121
122
123
124
125
126
127
128
129
130
131
132
133
134
135
136
137
138
139
140
141
142
143
144
145
146
147
148
149
150
151
152
153
154
155
156
157
158
159
160
161
162
163
164
165
166
167
168
169
170
171
172
173
174
175
176
177
178
179
180
181
182
183
184
185
186
187
188
189
190
191
192
193
194
195
196
197
198
199
200
201
202
203
204
205
206
207
208
209
210
211
212
213
214
215
216
217
218
219
220
221
222
223
224
225
226
227
228
229
230
231
232
233
234
235
236
237
238
239
240
241
242
243
244
245
246
247
248
249
250
251
252
253
254
255
256
257
258
259
260
261
262
263
264
265
266
267
268
269
270
271
272
273
274
275
276
277
278
279
280
281
282
283
284
285
286
287
288
289
290
291
292
293
294
295
296
297
298
299
300
301
302
303
304
305
306
307
308
309
310
311
312
313
314
315
316
317
318
319
320
321
322
323
324
325
326
327
328
329
330
331
332
333
334
335
336
337
338
339
340
341
342
343
344
345
346
347
348
349
350
351
352
353
354
355
356
357
358
359
360
361
362
363
364
365
366
367
368
369
370
371
372
373
374
375
376
377
378
379
380
381
382
383
384
385
386
387
388
389
390
391
392
393
394
395
396
397
398
399
400
401
402
403
404
405
406
407
408
409
410
411
412
413
414
415
416
417
418
419
420
421
422
423
424
425
426
427
428
429
430
431
432
433
434
435
436
437
438
439
440
441
442
443
444
445
446
447
448
449
450
451
452
453
454
455
456
457
458
459
460
461
462
463
464
465
466
467
468
469
470
471
472
473
474
475
476
477
478
479
480
481
482
483
484
485
486
487
488
489
490
491
492
493
494
495
496
497
498
499
500
501
502
503
504
505
506
507
508
509
510
511
512
513
514
515
516
517
518
519
520
521
522
523
524
525
526
527
528
529
530
531
532
533
534
535
536
537
538
539
540
541
542
543
544
545
546
547
548
549
550
551
552
553
554
555
556
557
558
559
560
561
562
563
564
565
566
567
568
569
570
571
572
573
574
575
576
577
578
579
580
581
582
583
584
585
586
587
588
589
590
591
592
593
594
595
596
597
598
599
600
601
602
603
604
605
606
607
608
609
610
611
612
613
614
615
616
617
618
619
620
621
622
623
624
625
626
627
628
629
630
631
632
633
634
635
636
637
638
639
640
641
642
643
644
645
646
647
648
649
650
651
652
653
654
655
656
657
658
659
660
661
662
663
664
665
666
667
668
669
670
671
672
673
674
675
676
677
678
679
680
681
682
683
684
685
686
687
688
689
690
691
692
693
694
695
696
697
698
699
700
701
702
703
704
705
706
707
708
709
710
711
712
713
714
715
716
717
718
719
720
721
722
723
724
725
726
727
728
729
730
731
732
733
734
735
736
737
738
739
740
741
742
743
744
745
746
747
748
749
750
751
752
753
754
755
756
757
758
759
760
761
762
763
764
765
766
767
768
769
770
771
772
773
774
775
776
777
778
779
780
781
782
783
784
785
786
787
788
789
790
791
792
793
794
795
796
797
798
799
800
801
802
803
804
805
806
807
808
809
810
811
812
813
814
815
816
817
818
819
820
821
822
823
824
825
826
827
828
829
830
831
832
833
834
835
836
837
838
839
840
841
842
843
844
845
846
847
848
849
850
851
852
853
854
855
856
857
858
859
860
861
862
863
864
865
866
867
868
869
870
871
872
873
874
875
876
877
878
879
880
881
882
883
884
885
886
887
888
889
890
891
892
893
894
895
896
897
898
899
900
901
902
903
904
905
906
907
908
909
910
911
912
913
914
915
916
917
918
919
920
921
922
923
924
925
926
927
928
929
930
931
932
933
934
935
936
937
938
939
940
941
942
943
944
945
946
947
948
949
950
951
952
953
954
955
956
957
958
959
960
961
962
963
964
965
966
967
968
969
970
971
972
973
974
975
976
977
978
979
980
981
982
983
984
985
986
987
988
989
990
991
992
993
994
995
996
997
998
999
1000
1001
1002
1003
1004
1005
1006
1007
1008
1009
1010
1011
1012
1013
1014
1015
1016
1017
1018
1019
1020
1021
1022
1023
1024
1025
1026
1027
1028
1029
1030
1031
1032
1033
1034
1035
1036
1037
1038
1039
1040
1041
1042
1043
1044
1045
1046
1047
1048
1049
1050
1051
1052
1053
1054
1055
1056
1057
1058
1059
1060
1061
1062
1063
1064
1065
1066
1067
1068
1069
1070
1071
1072
1073
1074
1075
1076
1077
1078
1079
1080
1081
1082
1083
1084
1085
1086
1087
1088
1089
1090
1091
1092
1093
1094
1095
1096
1097
1098
1099
1100
1101
1102
1103
1104
1105
1106
1107
1108
1109
1110
1111
1112
1113
1114
1115
1116
1117
1118
1119
1120
1121
1122
1123
1124
1125
1126
1127
1128
1129
1130
1131
1132
1133
1134
1135
1136
1137
1138
1139
1140
1141
1142
1143
1144
1145
1146
1147
1148
1149
1150
1151
1152
1153
1154
1155
1156
1157
1158
1159
1160
1161
1162
1163
1164
1165
1166
1167
1168
1169
1170
1171
1172
1173
1174
1175
1176
1177
1178
1179
1180
1181
1182
1183
1184
1185
1186
1187
1188
1189
1190
1191
1192
1193
1194
1195
1196
1197
1198
1199
1200
1201
1202
1203
1204
1205
1206
1207
1208
1209
1210
1211
1212
1213
1214
1215
1216
1217
1218
1219
1220
1221
1222
1223
1224
1225
1226
1227
1228
1229
1230
1231
1232
1233
1234
1235
1236
1237
1238
1239
1240
1241
1242
1243
1244
1245
1246
1247
1248
1249
1250
1251
1252
1253
1254
1255
1256
1257
1258
1259
1260
1261
1262
1263
1264
1265
1266
1267
1268
1269
1270
1271
1272
1273
1274
1275
1276
1277
1278
1279
1280
1281
1282
1283
1284
1285
1286
1287
1288
1289
1290
1291
1292
1293
1294
1295
1296
1297
1298
1299
1300
1301
1302
1303
1304
1305
1306
1307
1308
1309
1310
1311
1312
1313
1314
1315
1316
1317
1318
1319
1320
1321
1322
1323
1324
1325
1326
1327
1328
1329
1330
1331
1332
1333
1334
1335
1336
1337
1338
1339
1340
1341
1342
1343
1344
1345
1346
1347
1348
1349
1350
1351
1352
1353
1354
1355
1356
1357
1358
1359
1360
1361
1362
1363
1364
1365
1366
1367
1368
1369
1370
1371
1372
1373
1374
1375
1376
1377
1378
1379
1380
1381
1382
1383
1384
1385
1386
1387
1388
1389
1390
1391
1392
1393
1394
1395
1396
1397
1398
1399
1400
1401
1402
1403
1404
1405
1406
1407
1408
1409
1410
1411
1412
1413
1414
1415
1416
1417
1418
1419
1420
1421
1422
1423
1424
1425
1426
1427
1428
1429
1430
1431
1432
1433
1434
1435
1436
1437
1438
1439
1440
1441
1442
1443
1444
1445
1446
1447
1448
1449
1450
1451
1452
1453
1454
1455
1456
1457
1458
1459
1460
1461
1462
1463
1464
1465
1466
1467
1468
1469
1470
1471
1472
1473
1474
1475
1476
1477
1478
1479
1480
1481
1482
1483
1484
1485
1486
1487
1488
1489
1490
1491
1492
1493
1494
1495
1496
1497
1498
1499
1500
1501
1502
1503
1504
1505
1506
1507
1508
1509
1510
1511
1512
1513
1514
1515
1516
1517
1518
1519
1520
1521
1522
1523
1524
1525
1526
1527
1528
1529
1530
1531
1532
1533
1534
1535
1536
1537
1538
1539
1540
1541
1542
1543
1544
1545
1546
1547
1548
1549
1550
1551
1552
1553
1554
1555
1556
1557
1558
1559
1560
1561
1562
1563
1564
1565
1566
1567
1568
1569
1570
1571
1572
1573
1574
1575
1576
1577
1578
1579
1580
1581
1582
1583
1584
1585
1586
1587
1588
1589
1590
1591
1592
1593
1594
1595
1596
1597
1598
1599
1600
1601
1602
1603
1604
1605
1606
1607
1608
1609
1610
1611
1612
1613
1614
1615
1616
1617
1618
1619
1620
1621
1622
1623
1624
1625
1626
1627
1628
1629
1630
1631
1632
1633
1634
1635
1636
1637
1638
1639
1640
1641
1642
1643
1644
1645
1646
1647
1648
1649
1650
1651
1652
1653
1654
1655
1656
1657
1658
1659
1660
1661
1662
1663
1664
1665
1666
1667
1668
1669
1670
1671
1672
1673
1674
1675
1676
1677
1678
1679
1680
1681
1682
1683
1684
1685
1686
1687
1688
1689
1690
1691
1692
1693
1694
1695
1696
1697
1698
1699
1700
1701
1702
1703
1704
1705
1706
1707
1708
1709
1710
1711
1712
1713
1714
1715
1716
1717
1718
1719
1720
1721
1722
1723
1724
1725
1726
1727
1728
1729
1730
1731
1732
1733
1734
1735
1736
1737
1738
1739
1740
1741
1742
1743
1744
1745
1746
1747
1748
1749
1750
1751
1752
1753
1754
1755
1756
1757
1758
1759
1760
1761
1762
1763
1764
1765
1766
1767
1768
1769
1770
1771
1772
1773
1774
1775
1776
1777
1778
1779
1780
1781
1782
1783
1784
1785
1786
1787
1788
1789
1790
1791
1792
1793
1794
1795
1796
1797
1798
1799
1800
1801
1802
1803
1804
1805
1806
1807
1808
1809
1810
1811
1812
1813
1814
1815
1816
1817
1818
1819
1820
1821
1822
1823
1824
1825
1826
1827
1828
1829
1830
1831
1832
1833
1834
1835
1836
1837
1838
1839
1840
1841
1842
1843
1844
1845
1846
1847
1848
1849
1850
1851
1852
1853
1854
1855
1856
1857
1858
1859
1860
1861
1862
1863
1864
1865
1866
1867
1868
1869
1870
1871
1872
1873
1874
1875
1876
1877
1878
1879
1880
1881
1882
1883
1884
1885
1886
1887
1888
1889
1890
1891
1892
1893
1894
1895
1896
1897
1898
1899
1900
1901
1902
1903
1904
1905
1906
1907
1908
1909
1910
1911
1912
1913
1914
1915
1916
1917
1918
1919
1920
1921
1922
1923
1924
1925
1926
1927
1928
1929
1930
1931
1932
1933
1934
1935
1936
1937
1938
1939
1940
1941
1942
1943
1944
1945
1946
1947
1948
1949
1950
1951
1952
1953
1954
1955
1956
1957
1958
1959
1960
1961
1962
1963
1964
1965
1966
1967
1968
1969
1970
1971
1972
1973
1974
1975
1976
1977
1978
1979
1980
1981
1982
1983
1984
1985
1986
1987
1988
1989
1990
1991
1992
1993
1994
1995
1996
1997
1998
1999
2000
2001
2002
2003
2004
2005
2006
2007
2008
2009
2010
2011
2012
2013
2014
2015
2016
2017
2018
2019
2020
2021
2022
2023
2024
2025
2026
2027
2028
2029
2030
2031
2032
2033
2034
2035
2036
2037
2038
2039
2040
2041
2042
2043
2044
2045
2046
2047
2048
2049
2050
2051
2052
2053
2054
2055
2056
2057
2058
2059
2060
2061
2062
2063
2064
2065
2066
2067
2068
2069
2070
2071
2072
2073
2074
2075
2076
2077
2078
2079
2080
2081
2082
2083
2084
2085
2086
2087
2088
2089
2090
2091
2092
2093
2094
2095
2096
2097
2098
2099
2100
2101
2102
2103
2104
2105
2106
2107
2108
2109
2110
2111
2112
2113
2114
2115
2116
2117
2118
2119
2120
2121
2122
2123
2124
2125
2126
2127
2128
2129
2130
2131
2132
2133
2134
2135
2136
2137
2138
2139
2140
2141
2142
2143
2144
2145
2146
2147
2148
2149
2150
2151
2152
2153
2154
2155
2156
2157
2158
2159
2160
2161
2162
2163
2164
2165
2166
2167
2168
2169
2170
2171
2172
2173
2174
2175
2176
2177
2178
2179
2180
2181
2182
2183
2184
2185
2186
2187
2188
2189
2190
2191
2192
2193
2194
2195
2196
2197
2198
2199
2200
2201
2202
2203
2204
2205
2206
2207
2208
2209
2210
2211
2212
2213
2214
2215
2216
2217
2218
2219
2220
2221
2222
2223
2224
2225
2226
2227
2228
2229
2230
2231
2232
2233
2234
2235
2236
2237
2238
2239
2240
2241
2242
2243
2244
2245
2246
2247
2248
2249
2250
2251
2252
2253
2254
2255
2256
2257
2258
2259
2260
2261
2262
2263
2264
2265
2266
2267
2268
2269
2270
2271
2272
2273
2274
2275
2276
2277
2278
2279
2280
2281
2282
2283
2284
2285
2286
2287
2288
2289
2290
2291
2292
2293
2294
2295
2296
2297
2298
2299
2300
2301
2302
2303
2304
2305
2306
2307
2308
2309
2310
2311
2312
2313
2314
2315
2316
2317
2318
2319
2320
2321
2322
2323
2324
2325
2326
2327
2328
2329
2330
2331
2332
2333
2334
2335
2336
2337
2338
2339
2340
2341
2342
2343
2344
2345
2346
2347
2348
2349
2350
2351
2352
2353
2354
2355
2356
2357
2358
2359
2360
2361
2362
2363
2364
2365
2366
2367
2368
2369
2370
2371
2372
2373
2374
2375
2376
2377
2378
2379
2380
2381
2382
2383
2384
2385
2386
2387
2388
2389
2390
2391
2392
2393
2394
2395
2396
2397
2398
2399
2400
2401
2402
2403
2404
2405
2406
2407
2408
2409
2410
2411
2412
2413
2414
2415
2416
2417
2418
2419
2420
2421
2422
2423
2424
2425
2426
2427
2428
2429
2430
2431
2432
2433
2434
2435
2436
2437
2438
2439
2440
2441
2442
2443
2444
2445
2446
2447
2448
2449
2450
2451
2452
2453
2454
2455
2456
2457
2458
2459
2460
2461
2462
2463
2464
2465
2466
2467
2468
2469
2470
2471
2472
2473
2474
2475
2476
2477
2478
2479
2480
2481
2482
2483
2484
2485
2486
2487
2488
2489
2490
2491
2492
2493
2494
2495
2496
2497
2498
2499
2500
2501
2502
2503
2504
2505
2506
2507
2508
2509
2510
2511
2512
2513
2514
2515
2516
2517
2518
2519
2520
2521
2522
2523
2524
2525
2526
2527
2528
2529
2530
2531
2532
2533
2534
2535
2536
2537
2538
2539
2540
2541
2542
2543
2544
2545
2546
2547
2548
2549
2550
2551
2552
2553
2554
2555
2556
2557
2558
2559
2560
2561
2562
2563
2564
2565
2566
2567
2568
2569
2570
2571
2572
2573
2574
2575
2576
2577
2578
2579
2580
2581
2582
2583
2584
2585
2586
2587
2588
2589
2590
2591
2592
2593
2594
2595
2596
2597
2598
2599
2600
2601
2602
2603
2604
2605
2606
2607
2608
2609
```

### 5.1.2 Synchronization status in multi channels ► [R\\_Channel\\_Syn](#)

This request invoked for a particular channel returns the status of the part program synchronization in a multi channel system (P and Q parameters of the G78 function).

Please note that like for any channel relative request, the channels are numbered from 0 to 7 contrary to the part program index which is from 1 to 8.



```
1 PROGRAM R_Gen_002
2 VAR
3     oR_Channel_Syn          : R_Channel_Syn;
4     P1                      : INT;
5     Q2                      : INT;
6 END_VAR
7
8
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24
25
26
27
28
29
30
31
32
33
34
35
36
37
38
39
40
41
42
43
44
45
46
47
48
49
50
51
52
53
54
55
56
57
58
59
60
61
62
63
64
65
66
67
68
69
70
71
72
73
74
75
76
77
78
79
80
81
82
83
84
85
86
87
88
89
90
91
92
93
94
95
96
97
98
99
100
101
102
103
104
105
106
107
108
109
110
111
112
113
114
115
116
117
118
119
120
121
122
123
124
125
126
127
128
129
130
131
132
133
134
135
136
137
138
139
140
141
142
143
144
145
146
147
148
149
150
151
152
153
154
155
156
157
158
159
160
161
162
163
164
165
166
167
168
169
170
171
172
173
174
175
176
177
178
179
180
181
182
183
184
185
186
187
188
189
190
191
192
193
194
195
196
197
198
199
200
201
202
203
204
205
206
207
208
209
210
211
212
213
214
215
216
217
218
219
220
221
222
223
224
225
226
227
228
229
230
231
232
233
234
235
236
237
238
239
240
241
242
243
244
245
246
247
248
249
250
251
252
253
254
255
256
257
258
259
260
261
262
263
264
265
266
267
268
269
270
271
272
273
274
275
276
277
278
279
280
281
282
283
284
285
286
287
288
289
290
291
292
293
294
295
296
297
298
299
300
301
302
303
304
305
306
307
308
309
310
311
312
313
314
315
316
317
318
319
320
321
322
323
324
325
326
327
328
329
330
331
332
333
334
335
336
337
338
339
340
341
342
343
344
345
346
347
348
349
350
351
352
353
354
355
356
357
358
359
360
361
362
363
364
365
366
367
368
369
370
371
372
373
374
375
376
377
378
379
380
381
382
383
384
385
386
387
388
389
390
391
392
393
394
395
396
397
398
399
400
401
402
403
404
405
406
407
408
409
410
411
412
413
414
415
416
417
418
419
420
421
422
423
424
425
426
427
428
429
430
431
432
433
434
435
436
437
438
439
440
441
442
443
444
445
446
447
448
449
450
451
452
453
454
455
456
457
458
459
460
461
462
463
464
465
466
467
468
469
470
471
472
473
474
475
476
477
478
479
480
481
482
483
484
485
486
487
488
489
490
491
492
493
494
495
496
497
498
499
500
501
502
503
504
505
506
507
508
509
510
511
512
513
514
515
516
517
518
519
520
521
522
523
524
525
526
527
528
529
530
531
532
533
534
535
536
537
538
539
540
541
542
543
544
545
546
547
548
549
550
551
552
553
554
555
556
557
558
559
560
561
562
563
564
565
566
567
568
569
570
571
572
573
574
575
576
577
578
579
580
581
582
583
584
585
586
587
588
589
590
591
592
593
594
595
596
597
598
599
600
601
602
603
604
605
606
607
608
609
610
611
612
613
614
615
616
617
618
619
620
621
622
623
624
625
626
627
628
629
630
631
632
633
634
635
636
637
638
639
640
641
642
643
644
645
646
647
648
649
650
651
652
653
654
655
656
657
658
659
660
661
662
663
664
665
666
667
668
669
670
671
672
673
674
675
676
677
678
679
680
681
682
683
684
685
686
687
688
689
690
691
692
693
694
695
696
697
698
699
700
701
702
703
704
705
706
707
708
709
710
711
712
713
714
715
716
717
718
719
720
721
722
723
724
725
726
727
728
729
730
731
732
733
734
735
736
737
738
739
740
741
742
743
744
745
746
747
748
749
750
751
752
753
754
755
756
757
758
759
760
761
762
763
764
765
766
767
768
769
770
771
772
773
774
775
776
777
778
779
780
781
782
783
784
785
786
787
788
789
790
791
792
793
794
795
796
797
798
799
800
801
802
803
804
805
806
807
808
809
810
811
812
813
814
815
816
817
818
819
820
821
822
823
824
825
826
827
828
829
830
831
832
833
834
835
836
837
838
839
840
841
842
843
844
845
846
847
848
849
850
851
852
853
854
855
856
857
858
859
860
861
862
863
864
865
866
867
868
869
870
871
872
873
874
875
876
877
878
879
880
881
882
883
884
885
886
887
888
889
890
891
892
893
894
895
896
897
898
899
900
901
902
903
904
905
906
907
908
909
910
911
912
913
914
915
916
917
918
919
920
921
922
923
924
925
926
927
928
929
930
931
932
933
934
935
936
937
938
939
940
941
942
943
944
945
946
947
948
949
950
951
952
953
954
955
956
957
958
959
960
961
962
963
964
965
966
967
968
969
970
971
972
973
974
975
976
977
978
979
980
981
982
983
984
985
986
987
988
989
990
991
992
993
994
995
996
997
998
999
1000
```

C01 - M00061EN-00

```
1
2
3 oR_Channel_Syn.CNCAdr      := ADR(NCK);
4 oR_Channel_Syn.byChannel   := 0;
5                               // Channels numbered 0 to 7 for the requests
6
7 oR_Channel_Syn(enable := xEnable);
8
9 IF oR_Channel_Syn.uiResult = 0 THEN
10     xEnable                := FALSE;
11     P1                     := oR_Channel_Syn.byPointEmit;
12                               // Synchro point reached
13     Q2                     := oR_Channel_Syn.byPointCh2;
14                               // synchro point expected from Channel 2
15 END_IF
```

1  
2  
3  
4  
5  
6  
A

1  
2  
3  
4  
5  
6  
A

1  
2  
3  
4  
5  
6  
A

#### 5.1.4 Reading of the value of a drive parameter ► R\_Drive\_Param

This request returns a particular drive parameter value in internal units (*diPdriveUI*) and in one of the other types according to the parameter structure. (see Annex A2)

```
R_GEN_003 X
1  PROGRAM R_GEN_003
2  VAR
3      oR_Drive_Param          :R_Drive_Param;
4      re_Answer               : REAL;
5      di_Answer               : DINT;
6      dw_Answer               : DWORD;
7      di_UI_Answer            : DINT;
8  END_VAR

1
2  oR_Drive_Param.CNCAdr       := ADR(NCK);
3  oR_Drive_Param.byDriveAdd    := 1;           // Reading drive 1
4  oR_Drive_Param.wopNumber     := 246;        // Reading V246
5
6  oR_Drive_Param(enable := xEnable);          // Trigger the request
7
8  IF oR_Drive_Param.uiResult = 0 THEN
9      xEnable                := FALSE;
10     re_Answer               := oR_Drive_Param.rePDrive;
11     di_Answer               := oR_Drive_Param.diPDrive;
12     dw_Answer               := oR_Drive_Param.dwPDrive;
13     di_UI_Answer            := oR_Drive_Param.diPDriveUI;
14 END_IF
```

### 5.1.5 Reading of the content of a drive register ► [R\\_Drive\\_Register](#)

This request is generally reserved for drive advanced diagnostic.

For Drive Registers definition and structure please refer to the drive parameters manuals (M00031).

```

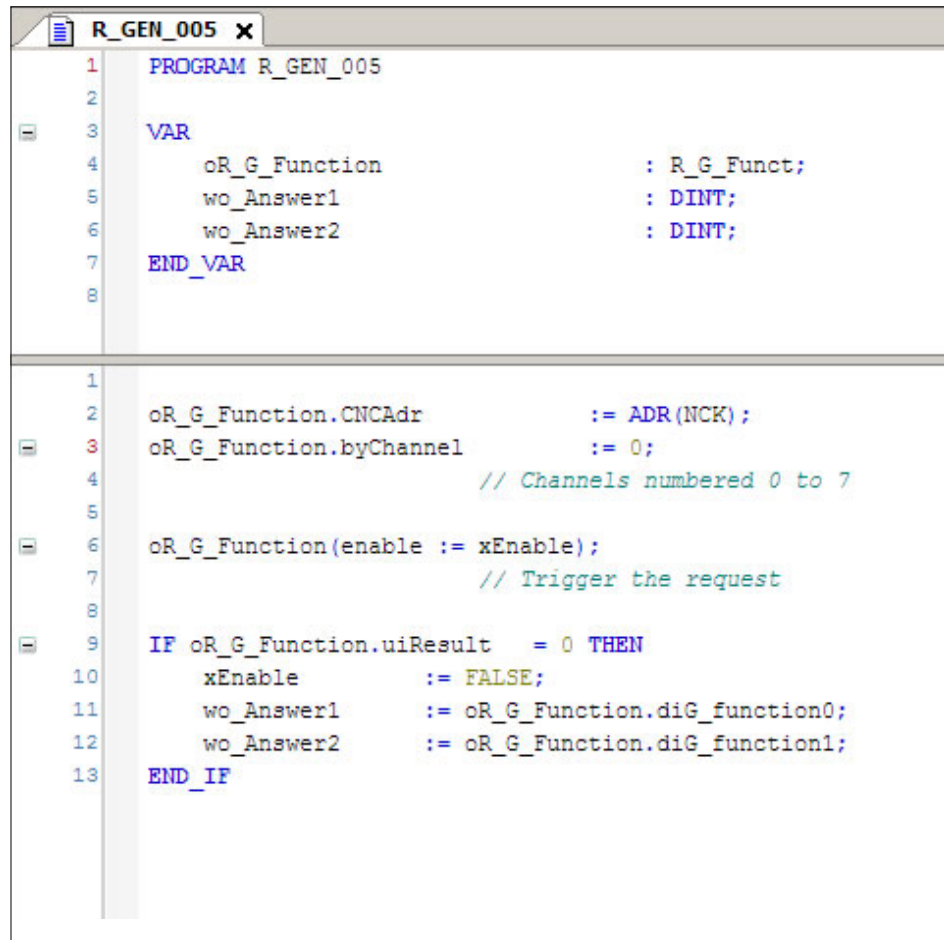
1  PROGRAM R_GEN_004
2
3  VAR
4      oR_Drive_Register      : R_Drive_Register;
5      wo_Answer              : WORD;
6  END_VAR
7
8
9
10
11  oR_Drive_Register.CNCAdr    := ADR(NCK);
12  oR_Drive_Register.byDriveAdd := 1;
13  // Reading drive 1
14  oR_Drive_Register.woRegister := 16#1AF1;
15  // Register address
16
17  oR_Drive_Register(enable := xEnable);
18  // Trigger the request
19
20  IF oR_Drive_Register.uiResult = 0 THEN
21      xEnable      := FALSE;
22      wo_Answer    := oR_Drive_Register.woRegValue;
23  END_IF

```

### 5.1.6 List of G codes active in the channel ▶ [R\\_G\\_Funct](#)

This channel specific request returns the list of all currently active G codes as two lists of bits.

The structure of this answer is given by the table below.



```
1  PROGRAM R_GEN_005
2
3  VAR
4      oR_G_Function          : R_G_Funct;
5      wo_Answer1             : DINT;
6      wo_Answer2             : DINT;
7  END_VAR
8
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24
25
26
27
28
29
30
31
32
33
34
35
36
37
38
39
40
41
42
43
44
45
46
47
48
49
50
51
52
53
54
55
56
57
58
59
60
61
62
63
64
65
66
67
68
69
70
71
72
73
74
75
76
77
78
79
80
81
82
83
84
85
86
87
88
89
90
91
92
93
94
95
96
97
98
99
100
101
102
103
104
105
106
107
108
109
110
111
112
113
114
115
116
117
118
119
120
121
122
123
124
125
126
127
128
129
130
131
132
133
134
135
136
137
138
139
140
141
142
143
144
145
146
147
148
149
150
151
152
153
154
155
156
157
158
159
160
161
162
163
164
165
166
167
168
169
170
171
172
173
174
175
176
177
178
179
180
181
182
183
184
185
186
187
188
189
190
191
192
193
194
195
196
197
198
199
200
201
202
203
204
205
206
207
208
209
210
211
212
213
214
215
216
217
218
219
220
221
222
223
224
225
226
227
228
229
230
231
232
233
234
235
236
237
238
239
240
241
242
243
244
245
246
247
248
249
250
251
252
253
254
255
256
257
258
259
260
261
262
263
264
265
266
267
268
269
270
271
272
273
274
275
276
277
278
279
280
281
282
283
284
285
286
287
288
289
290
291
292
293
294
295
296
297
298
299
300
301
302
303
304
305
306
307
308
309
310
311
312
313
314
315
316
317
318
319
320
321
322
323
324
325
326
327
328
329
330
331
332
333
334
335
336
337
338
339
340
341
342
343
344
345
346
347
348
349
350
351
352
353
354
355
356
357
358
359
360
361
362
363
364
365
366
367
368
369
370
371
372
373
374
375
376
377
378
379
380
381
382
383
384
385
386
387
388
389
390
391
392
393
394
395
396
397
398
399
400
401
402
403
404
405
406
407
408
409
410
411
412
413
414
415
416
417
418
419
420
421
422
423
424
425
426
427
428
429
430
431
432
433
434
435
436
437
438
439
440
441
442
443
444
445
446
447
448
449
450
451
452
453
454
455
456
457
458
459
460
461
462
463
464
465
466
467
468
469
470
471
472
473
474
475
476
477
478
479
480
481
482
483
484
485
486
487
488
489
490
491
492
493
494
495
496
497
498
499
500
501
502
503
504
505
506
507
508
509
510
511
512
513
514
515
516
517
518
519
520
521
522
523
524
525
526
527
528
529
530
531
532
533
534
535
536
537
538
539
540
541
542
543
544
545
546
547
548
549
550
551
552
553
554
555
556
557
558
559
560
561
562
563
564
565
566
567
568
569
570
571
572
573
574
575
576
577
578
579
580
581
582
583
584
585
586
587
588
589
590
591
592
593
594
595
596
597
598
599
600
601
602
603
604
605
606
607
608
609
610
611
612
613
614
615
616
617
618
619
620
621
622
623
624
625
626
627
628
629
630
631
632
633
634
635
636
637
638
639
640
641
642
643
644
645
646
647
648
649
650
651
652
653
654
655
656
657
658
659
660
661
662
663
664
665
666
667
668
669
670
671
672
673
674
675
676
677
678
679
680
681
682
683
684
685
686
687
688
689
690
691
692
693
694
695
696
697
698
699
700
701
702
703
704
705
706
707
708
709
710
711
712
713
714
715
716
717
718
719
720
721
722
723
724
725
726
727
728
729
730
731
732
733
734
735
736
737
738
739
740
741
742
743
744
745
746
747
748
749
750
751
752
753
754
755
756
757
758
759
760
761
762
763
764
765
766
767
768
769
770
771
772
773
774
775
776
777
778
779
780
781
782
783
784
785
786
787
788
789
790
791
792
793
794
795
796
797
798
799
800
801
802
803
804
805
806
807
808
809
810
811
812
813
814
815
816
817
818
819
820
821
822
823
824
825
826
827
828
829
830
831
832
833
834
835
836
837
838
839
840
841
842
843
844
845
846
847
848
849
850
851
852
853
854
855
856
857
858
859
860
861
862
863
864
865
866
867
868
869
870
871
872
873
874
875
876
877
878
879
880
881
882
883
884
885
886
887
888
889
890
891
892
893
894
895
896
897
898
899
900
901
902
903
904
905
906
907
908
909
910
911
912
913
914
915
916
917
918
919
920
921
922
923
924
925
926
927
928
929
930
931
932
933
934
935
936
937
938
939
940
941
942
943
944
945
946
947
948
949
950
951
952
953
954
955
956
957
958
959
960
961
962
963
964
965
966
967
968
969
970
971
972
973
974
975
976
977
978
979
980
981
982
983
984
985
986
987
988
989
990
991
992
993
994
995
996
997
998
999
1000
1001
1002
1003
1004
1005
1006
1007
1008
1009
1010
1011
1012
1013
1014
1015
1016
1017
1018
1019
1020
1021
1022
1023
1024
1025
1026
1027
1028
1029
1030
1031
1032
1033
1034
1035
1036
1037
1038
1039
1040
1041
1042
1043
1044
1045
1046
1047
1048
1049
1050
1051
1052
1053
1054
1055
1056
1057
1058
1059
1060
1061
1062
1063
1064
1065
1066
1067
1068
1069
1070
1071
1072
1073
1074
1075
1076
1077
1078
1079
1080
1081
1082
1083
1084
1085
1086
1087
1088
1089
1090
1091
1092
1093
1094
1095
1096
1097
1098
1099
1100
1101
1102
1103
1104
1105
1106
1107
1108
1109
1110
1111
1112
1113
1114
1115
1116
1117
1118
1119
1120
1121
1122
1123
1124
1125
1126
1127
1128
1129
1130
1131
1132
1133
1134
1135
1136
1137
1138
1139
1140
1141
1142
1143
1144
1145
1146
1147
1148
1149
1150
1151
1152
1153
1154
1155
1156
1157
1158
1159
1160
1161
1162
1163
1164
1165
1166
1167
1168
1169
1170
1171
1172
1173
1174
1175
1176
1177
1178
1179
1180
1181
1182
1183
1184
1185
1186
1187
1188
1189
1190
1191
1192
1193
1194
1195
1196
1197
1198
1199
1200
1201
1202
1203
1204
1205
1206
1207
1208
1209
1210
1211
1212
1213
1214
1215
1216
1217
1218
1219
1220
1221
1222
1223
1224
1225
1226
1227
1228
1229
1230
1231
1232
1233
1234
1235
1236
1237
1238
1239
1240
1241
1242
1243
1244
1245
1246
1247
1248
1249
1250
1251
1252
1253
1254
1255
1256
1257
1258
1259
1260
1261
1262
1263
1264
1265
1266
1267
1268
1269
1270
1271
1272
1273
1274
1275
1276
1277
1278
1279
1280
1281
1282
1283
1284
1285
1286
1287
1288
1289
1290
1291
1292
1293
1294
1295
1296
1297
1298
1299
1300
1301
1302
1303
1304
1305
1306
1307
1308
1309
1310
1311
1312
1313
1314
1315
1316
1317
1318
1319
1320
1321
1322
1323
1324
1325
1326
1327
1328
1329
1330
1331
1332
1333
1334
1335
1336
1337
1338
1339
1340
1341
1342
1343
1344
1345
1346
1347
1348
1349
1350
1351
1352
1353
1354
1355
1356
1357
1358
1359
1360
1361
1362
1363
1364
1365
1366
1367
1368
1369
1370
1371
1372
1373
1374
1375
1376
1377
1378
1379
1380
1381
1382
1383
1384
1385
1386
1387
1388
1389
1390
1391
1392
1393
1394
1395
1396
1397
1398
1399
1400
1401
1402
1403
1404
1405
1406
1407
1408
1409
1410
1411
1412
1413
1414
1415
1416
1417
1418
1419
1420
1421
1422
1423
1424
1425
1426
1427
1428
1429
1430
1431
1432
1433
1434
1435
1436
1437
1438
1439
1440
1441
1442
1443
1444
1445
1446
1447
1448
1449
1450
1451
1452
1453
1454
1455
1456
1457
1458
1459
1460
1461
1462
1463
1464
1465
1466
1467
1468
1469
1470
1471
1472
1473
1474
1475
1476
1477
1478
1479
1480
1481
1482
1483
1484
1485
1486
1487
1488
1489
1490
1491
1492
1493
1494
1495
1496
1497
1498
1499
1500
1501
1502
1503
1504
1505
1506
1507
1508
1509
1510
1511
1512
1513
1514
1515
1516
1517
1518
1519
1520
1521
1522
1523
1524
1525
1526
1527
1528
1529
1530
1531
1532
1533
1534
1535
1536
1537
1538
1539
1540
1541
1542
1543
1544
1545
1546
1547
1548
1549
1550
1551
1552
1553
1554
1555
1556
1557
1558
1559
1560
1561
1562
1563
1564
1565
1566
1567
1568
1569
1570
1571
1572
1573
1574
1575
1576
1577
1578
1579
1580
1581
1582
1583
1584
1585
1586
1587
1588
1589
1590
1591
1592
1593
1594
1595
1596
1597
1598
1599
1600
1601
1602
1603
1604
1605
1606
1607
1608
1609
1610
1611
1612
1613
1614
1615
1616
1617
1618
1619
1620
1621
1622
1623
1624
1625
1626
1627
1628
1629
1630
1631
1632
1633
1634
1635
1636
1637
1638
1639
1640
1641
1642
1643
1644
1645
1646
1647
1648
1649
1650
1651
1652
1653
1654
1655
1656
1657
1658
1659
1660
1661
1662
1663
1664
1665
1666
1667
1668
1669
1670
1671
1672
1673
1674
1675
1676
1677
1678
1679
1680
1681
1682
1683
1684
1685
1686
1687
1688
1689
1690
1691
1692
1693
1694
1695
1696
1697
1698
1699
1700
1701
1702
1703
1704
1705
1706
1707
1708
1709
1710
1711
1712
1713
1714
1715
1716
1717
1718
1719
1720
1721
1722
1723
1724
1725
1726
1727
1728
1729
1730
1731
1732
1733
1734
1735
1736
1737
1738
1739
1740
1741
1742
1743
1744
1745
1746
1747
1748
1749
1750
1751
1752
1753
1754
1755
1756
1757
1758
1759
1760
1761
1762
1763
1764
1765
1766
1767
1768
1769
1770
1771
1772
1773
1774
1775
1776
1777
1778
1779
1780
1781
1782
1783
1784
1785
1786
1787
1788
1789
1790
1791
1792
1793
1794
1795
1796
1797
1798
1799
1800
1801
1802
1803
1804
1805
1806
1807
1808
1809
1810
1811
1812
1813
1814
1815
1816
1817
1818
1819
1820
1821
1822
1823
1824
1825
1826
1827
1828
1829
1830
1831
1832
1833
1834
1835
1836
1837
1838
1839
1840
1841
1842
1843
1844
1845
1846
1847
1848
1849
1850
1851
1852
1853
1854
1855
1856
1857
1858
1859
1860
1861
1862
1863
1864
1865
1866
1867
1868
1869
1870
1871
1872
1873
1874
1875
1876
1877
1878
1879
1880
1881
1882
1883
1884
1885
1886
1887
1888
1889
1890
1891
1892
1893
1894
1895
1896
1897
1898
1899
1900
1901
1902
1903
1904
1905
1906
1907
1908
1909
1910
1911
1912
1913
1914
1915
1916
1917
1918
1919
1920
1921
1922
1923
1924
1925
1926
1927
1928
1929
1930
1931
1932
1933
1934
1935
1936
1937
1938
1939
1940
1941
1942
1943
1944
1945
1946
1947
1948
1949
1950
1951
1952
1953
1954
1955
1956
1957
1958
1959
1960
1961
1962
1963
1964
1965
1966
1967
1968
1969
1970
1971
1972
1973
1974
1975
1976
1977
1978
1979
1980
1981
1982
1983
1984
1985
1986
1987
1988
1989
1990
1991
1992
1993
1994
1995
1996
1997
1998
1999
2000
2001
2002
2003
2004
2005
2006
2007
2008
2009
2010
2011
2012
2013
2014
2015
2016
2017
2018
2019
2020
2021
2022
2023
2024
2025
2026
2027
2028
2029
2030
2031
2032
2033
2034
2035
2036
2037
2038
2039
2040
2041
2042
2043
2044
2045
2046
2047
2048
2049
2050
2051
2052
2053
2054
2055
2056
2057
2058
2059
2060
2061
2062
2063
2064
2065
2066
2067
2068
2069
2070
2071
2072
2073
2074
2075
2076
2077
2078
2079
2080
2081
2082
2083
2084
2085
2086
2087
2088
2089
2090
2091
2092
2093
2094
2095
2096
2097
2098
2099
2100
2101
2102
2103
2104
2105
2106
2107
2108
2109
2110
2111
2112
2113
2114
2115
2116
2117
2118
2119
2120
2121
2122
2123
2124
2125
2126
2127
2128
2129
2130
2131
2132
2133
2134
2135
2136
2137
2138
2139
2140
2141
2142
2143
2144
2145
2146
2147
2148
2149
2150
2151
2152
2153
2154
2155
2156
2157
2158
2159
2160
2161
2162
2163
2164
2165
2166
2167
2168
2169
2170
2171
2172
2173
2174
2175
2176
2177
2178
2179
2180
2181
2182
2183
2184
2185
2186
2187
2188
2189
2190
2191
2192
2193
2194
2195
2196
2197
2198
2199
2200
2201
2202
2203
2204
2205
2206
2207
2208
2209
2210
2211
2212
2213
2214
2215
2216
2217
2218
2219
2220
2221
2222
2223
2224
2225
2226
2227
2228
2229
2230
2231
2232
2233
2234
2235
2236
2237
2238
2239
2240
2241
2242
2243
2244
2245
2246
2247
2248
2249
2250
2251
2252
2253
2254
2255
2256
2257
2258
2259
2260
2261
2262
2263
2264
2265
2266
2267
2268
2269
2270
2271
2272
2273
2274
2275
2276
2277
2278
2279
2280
2281
2282
2283
2284
2285
2286
2287
2288
2289
2290
2291
2292
2293
2294
2295
2296
2297
2298
2299
2300
2301
2302
2303
2304
2305
2306
2307
2308
2309
2310
2311
2312
2313
2314
2315
2316
2317
2318
2319
2320
2321
2322
2323
2324
2325
2326
2327
2328
2329
2330
2331
2332
2333
2334
2335
2336
2337
2338
2339
2340
2341
2342
2343
2344
2345
2346
2347
2348
2349
2350
2351
2352
2353
2354
2355
2356
2357
2358
2359
2360
2361
2362
2363
2364
2365
2366
2367
2368
2369
2370
2371
2372
2373
2374
2375
2376
2377
2378
2379
2380
2381
2382
2383
2384
2385
2386
2387
2388
2389
2390
2391
2392
2393
2394
2395
2396
2397
2398
2399
2400
2401
2402
2403
2404
2405
2406
2407
2408
2409
2410
2411
2412
2413
2414
2415
2416
2417
2418
2419
2420
2421
2422
2423
2424
2425
2426
2427
2428
2429
2430
2431
2432
2433
2434
2435
2436
2437
2438
2439
2440
2441
2442
2443
2444
2445
2446
2447
2448
2449
2450
2451
2452
2453
2454
2455
2456
2457
2458
2459
2460
2461
2462
2463
2464
2465
2466
2467
2468
2469
2470
2471
2472
2473
2474
2475
2476
2477
2478
2479
2480
2481
2482
2483
2484
2485
2486
2487
2488
2489
2490
2491
2492
2493
2494
2495
2496
2497
2498
2499
2500
2501
2502
2503
2504
2505
2506
2507
2508
2509
2510
2511
2512
2513
2514
2515
2516
2517
2518
2519
2520
2521
2522
2523
2524
2525
2526
2527
2528
2529
2530
2531
2532
2533
2534
2535
2536
2537
2538
2539
2540
2541
2542
2543
2544
2545
2546
2547
2548
2549
2550
2551
2552
2553
2554
2555
2556
2557
2558
2559
2560
2561
2562
2563
2564
2565
2566
2567
2568
2569
2570
2571
2572
2573
2574
2575
2576
2577
2578
2579
2580
2581
2582
2583
2584
2585
2586
2587
2588
2589
2590
2591
2592
2593
2594
2595
2596
2597
2598
2599
2600
2601
2602
2603
2604
2605
2606
2607
2608
2609
2610
2611
2612
2613
2614
2615
2616
2617
2618
2619
2620
2621
2622
2623
2624
2625
2626
2627
2628
2629
2630
2631
2632
2633
2634
2635
2636
2637
2638
2639
2640

```

## List of G codes per set

| Index Set 1 | Fn                  | Description                                                      | Index Set 2 | Fn                  | Description                          |
|-------------|---------------------|------------------------------------------------------------------|-------------|---------------------|--------------------------------------|
| .0          | <b>G94</b>          | Feed rate expressed in millimetres, inches or degrees per minute | .0          | <b>G10</b>          | Activate interrupt stop              |
| .1          | <b>G95</b>          | Feed rate expressed in millimetres or inches per revolution      | .1          | <b>G16</b>          | Tool axis selection                  |
| .2          | <b>G38</b>          | Threading                                                        | .2          | <b>G51</b>          | Mirror function                      |
| .3          | <b>G96</b>          | Activate CSS                                                     | .3          | <b>G45</b>          | Pocket cycle                         |
| .4          | <b>G97</b>          | Cancel CSS                                                       | .4          | <b>G12</b>          | Overspeed by handwheel               |
| .5          | <b>G92</b>          | Tangential feed rate                                             | .5          | <i>NUM Reserved</i> |                                      |
| .6          | <b>G20</b>          | Polar programming (X-C) lathe                                    | .6          |                     |                                      |
| .7          | <b>G21</b>          | Cartesian programming (X-Y) lathe                                | .7          | <b>G46</b>          | Complex pocket milling               |
| .8          | <b>G40</b>          | Cancel radius correction                                         | .8          | <b>G26</b>          | Activate RTCP                        |
| .9          | <b>G41</b>          | Radius correction left                                           | .9          | <b>G73</b>          | Cancel scaling                       |
| .10         | <b>G42</b>          | Radius correction right                                          | .10         | <b>G74</b>          | Activate scaling                     |
| .11         | <b>G53</b>          | Inhibit DAT1                                                     | .11         | <b>G999</b>         | Block concatenation                  |
| .12         | <b>G54</b>          | Activate DAT1                                                    | .12         | <i>NUM Reserved</i> |                                      |
| .13         | <b>G29</b>          | 3D tool correction                                               | .13         |                     |                                      |
| .14         | <i>NUM Reserved</i> |                                                                  | .14         |                     |                                      |
| .15         | <b>G93</b>          | Feed in 1/T                                                      | .15         |                     |                                      |
| .16         | <b>G17</b>          | Interpolation XY                                                 | .16         |                     |                                      |
| .17         | <b>G18</b>          | Interpolation YZ                                                 | .17         |                     |                                      |
| .18         | <b>G19</b>          | Interpolation XZ                                                 | .18         |                     |                                      |
| .19         | <b>G90</b>          | Absolute programming                                             | .19         |                     |                                      |
| .20         | <b>G91</b>          | Incremental programming                                          | .20         |                     |                                      |
| .21         | <b>G70</b>          | Inches programming                                               | .21         |                     |                                      |
| .22         | <b>G52</b>          | Machine coordinates programming                                  | .22         |                     |                                      |
| .23         | <b>G22</b>          | Interpolation CZ                                                 | .23         | <b>G24</b>          | Activate inclined plane              |
| .24         | <b>G0</b>           | Rapid positioning                                                | .24         | <b>G80</b>          | Cancel machining cycles              |
| .25         | <b>G01</b>          | Linear interpolation                                             | .25         | <i>NUM Reserved</i> |                                      |
| .26         | <b>G02</b>          | Interpolation CW                                                 | .26         |                     |                                      |
| .27         | <b>G03</b>          | Interpolation CCW                                                | .27         |                     |                                      |
| .28         | <b>G04</b>          | Dwell                                                            | .28         | <b>G64</b>          | Roughing cycles                      |
| .29         | <i>NUM Reserved</i> |                                                                  | .29         | <b>G65</b>          | Cycle de gorge                       |
| .30         |                     |                                                                  | .30         | <i>NUM Reserved</i> |                                      |
| .31         | <b>G09</b>          | Precise stop                                                     | .31         | <b>G07</b>          | Inclined axes programming (grinders) |

---

### List of G codes per set

| Fn         | Description                          | Index | Set |
|------------|--------------------------------------|-------|-----|
| <b>G00</b> | Rapid positioning                    | .24   | 1   |
| <b>G01</b> | Linear interpolation                 | .25   | 1   |
| <b>G02</b> | Interpolation CW                     | .26   | 1   |
| <b>G03</b> | Interpolation CCW                    | .27   | 1   |
| <b>G04</b> | Dwell                                | .28   | 1   |
| <b>G07</b> | Inclined axes programming (grinders) | .31   | 2   |
| <b>G09</b> | Precise stop                         | .31   | 1   |
| <b>G10</b> | Activate interrupt stop              | .0    | 2   |
| <b>G12</b> | Overspeed by handwheel               | .4    | 2   |
| <b>G16</b> | Tool axis selection                  | .1    | 2   |
| <b>G17</b> | Interpolation XY                     | .16   | 1   |
| <b>G18</b> | Interpolation YZ                     | .17   | 1   |
| <b>G19</b> | Interpolation XZ                     | .18   | 1   |
| <b>G20</b> | Polar programming (X-C) lathe        | .6    | 1   |
| <b>G21</b> | Cartesian programming (X-Y) lathe    | .7    | 1   |
| <b>G22</b> | Interpolation CZ                     | .23   | 1   |
| <b>G24</b> | Activate inclined plane              | .23   | 2   |
| <b>G26</b> | Activate RTCP                        | .8    | 2   |
| <b>G29</b> | 3D tool correction                   | .13   | 1   |
| <b>G38</b> | Threading                            | .2    | 1   |
| <b>G40</b> | Cancel radius correction             | .8    | 1   |
| <b>G41</b> | Radius correction left               | .9    | 1   |
| <b>G42</b> | Radius correction right              | .10   | 1   |
| <b>G45</b> | Pocket cycle                         | .3    | 2   |
| <b>G46</b> | Complex pocket milling               | .7    | 2   |
| <b>G51</b> | Mirror function                      | .2    | 2   |
| <b>G52</b> | Machine coordinates programming      | .22   | 1   |
| <b>G53</b> | Inhibit DAT1                         | .11   | 1   |
| <b>G54</b> | Activate DAT1                        | .12   | 1   |
| <b>G64</b> | Roughing cycles                      | .28   | 2   |
| <b>G65</b> | Cycle de gorge                       | .29   | 2   |
| <b>G70</b> | Inches programming                   | .21   | 1   |
| <b>G73</b> | Cancel scaling                       | .9    | 2   |

|             |                         |            |          |
|-------------|-------------------------|------------|----------|
| <b>G74</b>  | Activate scaling        | <b>.10</b> | <b>2</b> |
| <b>G80</b>  | Cancel machining cycles | <b>.24</b> | <b>2</b> |
| <b>G90</b>  | Absolute programming    | <b>.19</b> | <b>1</b> |
| <b>G91</b>  | Incremental programming | <b>.20</b> | <b>1</b> |
| <b>G92</b>  | Tangential feed rate    | <b>.5</b>  | <b>1</b> |
| <b>G93</b>  | Feed in 1/T             | <b>.15</b> | <b>1</b> |
| <b>G94</b>  | Feed per mn             | <b>.0</b>  | <b>1</b> |
| <b>G95</b>  | Feed per rev.           | <b>.1</b>  | <b>1</b> |
| <b>G96</b>  | Activate CSS            | <b>.3</b>  | <b>1</b> |
| <b>G97</b>  | Cancel CSS              | <b>.4</b>  | <b>1</b> |
| <b>G999</b> | Block concatenation     | <b>.11</b> | <b>2</b> |

1

2

3

4

5

6

A

### 5.1.7 System identification ► R\_Ident\_Info

This request returns different information about the system configuration as shown in the online visualization below.

The serial number of the system is returned by another function.

The screenshot displays the 'R\_GEN\_006' application window. At the top, a table titled 'RTS.Application.R\_GEN\_006' lists system identification data. Below the table, a ladder logic program is shown with line numbers 1 through 13.

| Expression | Type         | Value        |
|------------|--------------|--------------|
| oR_Ident   | R_Ident_Info |              |
| st_Answer1 | STRING       | '2.2.1.0'    |
| st_Answer2 | STRING       | '1.1.1.0'    |
| st_Answer3 | STRING       | '4.0.91.15'  |
| st_Answer4 | STRING       | 'FLEXIUM+68' |

```
1  oR_Ident.CNCAdr 16#CD2DDB80 := ADR(NCK);
2
3
4  oR_Ident(enable FALSE := xEnable FALSE);
5  // Trigger the request
6
7  IF oR_Ident.uiResult 16#0063 = 0 THEN
8    xEnable FALSE := FALSE;
9    st_Answer1 2.2.1.0 := oR_Ident.stBootCNC 2.2.1.0;
10   st_Answer2 1.1.1.0 := oR_Ident.stFpgaCNC 1.1.1.0;
11   st_Answer3 4.0.91.15 := oR_Ident.stSoftCNC 4.0.91.15;
12   st_Answer4 FLEXIUM+68 := oR_Ident.stTypeCNC FLEXIUM+68;
13  END_IF RETURN
```

### 5.1.8 List of L Parameters ➤ R\_L\_Vars

This channel specific request will return the value of 10 consecutive L parameters.

The request will not work if the list passes the limit of L parameters:  
e.g. Request the reading starting at L102.

```

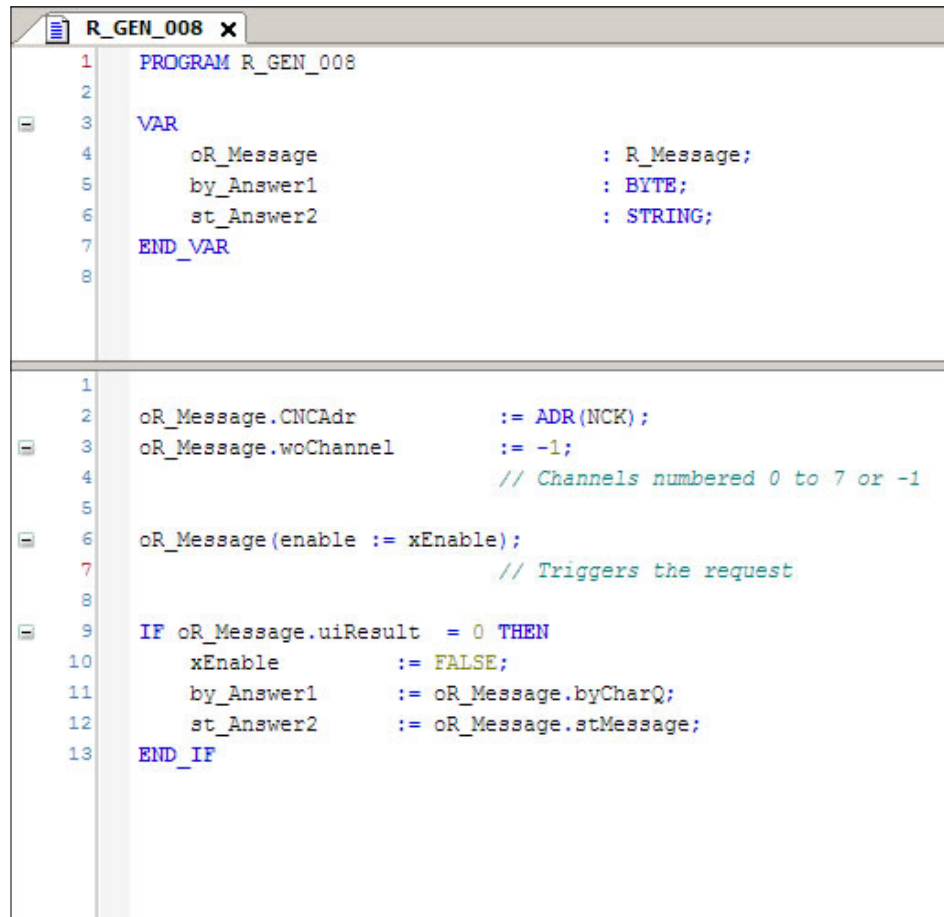
1  PROGRAM R_GEN_007
2
3  VAR
4      oR_L_Param          : R_L_Vars;
5      re_Answer1          : REAL;
6      re_Answer2          : REAL;
7      re_Answer3          : REAL;
8      re_Answer4          : REAL;
9      re_Answer5          : REAL;
10     re_Answer6          : REAL;
11     re_Answer7          : REAL;
12     re_Answer8          : REAL;
13     re_Answer9          : REAL;
14     re_Answer10         : REAL;
15
16
17     oR_L_Param.CNCAdr      := ADDR(NCK);
18     oR_L_Param.byChannel   := 0;
19     // Channels numbered 0 to 7
20     oR_L_Param.uiLnumber    := 102;
21     // Read from L102
22
23     oR_L_Param(enable := xEnable);
24     // Triggers the request
25
26     IF oR_L_Param.uiResult = 0 THEN
27         xEnable          := FALSE;
28         re_Answer1       := oR_L_Param.reLVars_0;
29         re_Answer2       := oR_L_Param.reLVars_1;
30         re_Answer3       := oR_L_Param.reLVars_2;
31         re_Answer4       := oR_L_Param.reLVars_3;
32         re_Answer5       := oR_L_Param.reLVars_4;
33         re_Answer6       := oR_L_Param.reLVars_5;
34         re_Answer7       := oR_L_Param.reLVars_6;
35         re_Answer8       := oR_L_Param.reLVars_7;
36         re_Answer9       := oR_L_Param.reLVars_8;
37         re_Answer10      := oR_L_Param.reLVars_9;
38     END_IF

```

### 5.1.9 Reading of a dialog (\$) message ► R\_Message

This channel specific request will return a string corresponding to a \$ message posted.

In addition to the string itself, the number of characters in the string is also reported.



```
1 PROGRAM R_GEN_008
2
3 VAR
4   oR_Message          : R_Message;
5   by_Answer1          : BYTE;
6   st_Answer2          : STRING;
7 END_VAR
8
9
10
11
12
13 oR_Message.CNCAdr      := ADR(NCK);
14 oR_Message.woChannel   := -1;
15                        // Channels numbered 0 to 7 or -1
16
17 oR_Message(enable := xEnable);
18                        // Triggers the request
19
20 IF oR_Message.uiResult = 0 THEN
21   xEnable              := FALSE;
22   by_Answer1           := oR_Message.byCharQ;
23   st_Answer2           := oR_Message.stMessage;
24 END_IF
```

### 5.1.10 Reading of an NCK parameter ► R\_Nck\_Par

This request will return the values of the different words of a parameter according to the parameter format (refer to the Flexium or Flexium+ Parameters manual M00021 or M00041).

Due to the high possible number of words it may be necessary to invoke the request several times by increasing the Offset attribute.

The flag *byByte\_4* indicates if the whole parameter has been read or not.



The offset value is given in Bytes, regardless of the type of parameters.

```

R_GEN_009 X
2
3  VAR
4      oR_NCK_param          : R_Nck_Par;
5      woAnswer0             : WORD;
6      woAnswer1             : WORD;
7      woAnswer2             : WORD;
8      woAnswer3             : WORD;
9      byAnswer4             : BYTE;
10     byAnswer5             : BYTE;
11     inAnswer6             : INT;
12     diAnswer7             : DINT;
13     siAnswer8             : SINT;
14     udAnswer9             : UDINT;
15     uiAnswer10            : UINT;
16     diAnswer11            : DINT;
17     stAnswer12            : STRING;
18     lrAnswer13            : LREAL;
19  END_VAR
20

1
2  oR_NCK_param.CNCAdr       := ADDR(NCK);
3  oR_NCK_param.uiPnumber    := 49;
4                             // Parameter to read
5  oR_NCK_param.uiPoffset    := 2;
6                             // Offset given in Bytes
7
8  oR_NCK_param(enable := xEnable);
9                             // Triggers the request
10
11 IF oR_NCK_param.uiResult = 0 THEN
12     xEnable                := FALSE;
13     woAnswer0              := oR_NCK_param.woWord_0;    // Parameter number
14     woAnswer1              := oR_NCK_param.woWord_1;    // Parameters type
15     woAnswer2              := oR_NCK_param.woWord_2;    // Parameter point
16     woAnswer3              := oR_NCK_param.woWord_3;    // Bytes read
17     byAnswer4              := oR_NCK_param.byByte_4;    // read complete or not
18     byAnswer5              := oR_NCK_param.byValue;     // for parameters type 0
19     inAnswer6              := oR_NCK_param.inValue;     // for parameters type 1
20     diAnswer7              := oR_NCK_param.diValue;     // for parameters type 2 and 6
21     siAnswer8              := oR_NCK_param.siValue;     // for parameters type 3
22     udAnswer9              := oR_NCK_param.udValue;     // for parameters type 4
23     uiAnswer10             := oR_NCK_param.uiValue;     // for parameters type 5 and 7
24     diAnswer11             := oR_NCK_param.diValue;     // for parameters type 2 and 6
25     stAnswer12             := oR_NCK_param.stValue;     // for parameters type 8
26     lrAnswer13             := oR_NCK_param.lrValue;     // for parameters type 9 and 10
27 END_IF
  
```

### 5.1.11 Reading of the overrides values ► R\_Override\_All

This request will return the value of the 8 feed overrides, regardless of the number of channels declared.

```
R_GEN_010 x
1 PROGRAM R_GEN_010
2
3 VAR
4     oR_Override           : R_Override_All;
5     di_Answer1            : DINT;
6     di_Answer2            : DINT;
7     di_Answer3            : DINT;
8     di_Answer4            : DINT;
9     di_Answer5            : DINT;
10    di_Answer6            : DINT;
11    di_Answer7            : DINT;
12    di_Answer8            : DINT;
13    diOverrideCh8         : DINT;
14 END_VAR
15
16
17 oR_Override.CNCAdr        := ADR(NCK);
18
19 oR_Override(enable := xEnable);
20                               // Trigger the request
21
22 IF oR_Override.uiResult = 0 THEN
23     xEnable                 := FALSE;
24     di_Answer1              := oR_Override.byOverrideCh1;
25     di_Answer2              := oR_Override.byOverrideCh2;
26     di_Answer3              := oR_Override.byOverrideCh3;
27     di_Answer4              := oR_Override.byOverrideCh4;
28     di_Answer5              := oR_Override.byOverrideCh5;
29     di_Answer6              := oR_Override.byOverrideCh6;
30     di_Answer7              := oR_Override.byOverrideCh7;
31     di_Answer8              := oR_Override.byOverrideCh8;
32 END_IF
```

### 5.1.12 Reading the part program memory size ▶ R\_Program\_Memory\_Size

This request returns the total as well as the available size of each segment of part program memory. In addition it also returns the stack size (for symbolic variables).

```

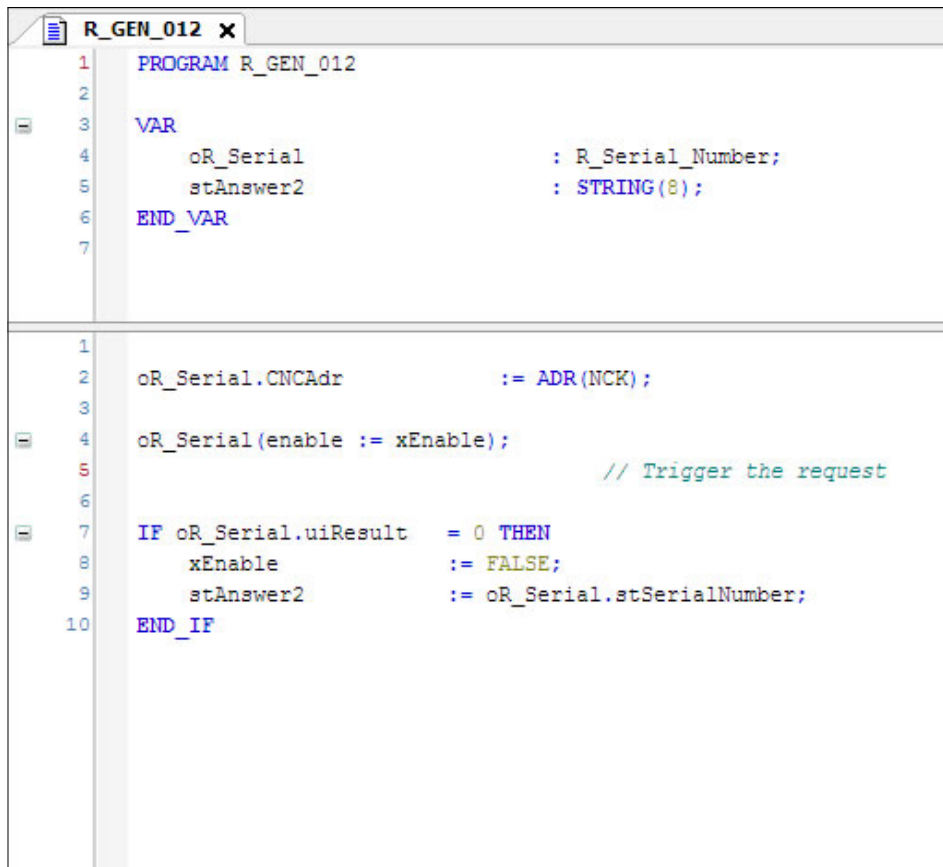
1  PROGRAM R_GEN_011
2
3  VAR
4      oR_Memory_Size          : R_Program_Memory_Size;
5      udAnswer1               : UDINT;
6      udAnswer2               : UDINT;
7      udAnswer3               : UDINT;
8      udAnswer4               : UDINT;
9      udAnswer5               : UDINT;
10     udAnswer6               : UDINT;
11     udAnswer7               : UDINT;
12     udAnswer8               : UDINT;
13     udAnswer9               : UDINT;
14  END_VAR

1
2  oR_Memory_Size.CNCAdr       := ADR(NCK);
3
4  oR_Memory_Size(enable := xEnable);
5                                     // Trigger the request
6
7  IF oR_Memory_Size.uiResult = 0 THEN
8      xEnable                 := FALSE;
9      udAnswer1               := oR_Memory_Size.udStackSize;
10     udAnswer2               := oR_Memory_Size.udTotalSize;
11     udAnswer3               := oR_Memory_Size.udZone0Free;
12     udAnswer4               := oR_Memory_Size.udZone1Free;
13     udAnswer5               := oR_Memory_Size.udZone1Size;
14     udAnswer6               := oR_Memory_Size.udZone2Free;
15     udAnswer7               := oR_Memory_Size.udZone2Size;
16     udAnswer8               := oR_Memory_Size.udZone3Free;
17     udAnswer9               := oR_Memory_Size.udZone3Size;
18  END_IF

```

### 5.1.13 Reading the system serial number ► [R\\_Serial\\_Number](#)

As indicated before, the system serial number is only returned by this request and not by System Identification.



```
1  PROGRAM R_GEN_012
2
3  VAR
4      oR_Serial          : R_Serial_Number;
5      stAnswer2          : STRING(8);
6  END_VAR
7
8
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24
25
26
27
28
29
30
31
32
33
34
35
36
37
38
39
40
41
42
43
44
45
46
47
48
49
50
51
52
53
54
55
56
57
58
59
60
61
62
63
64
65
66
67
68
69
70
71
72
73
74
75
76
77
78
79
80
81
82
83
84
85
86
87
88
89
90
91
92
93
94
95
96
97
98
99
100
101
102
103
104
105
106
107
108
109
110
111
112
113
114
115
116
117
118
119
120
121
122
123
124
125
126
127
128
129
130
131
132
133
134
135
136
137
138
139
140
141
142
143
144
145
146
147
148
149
150
151
152
153
154
155
156
157
158
159
160
161
162
163
164
165
166
167
168
169
170
171
172
173
174
175
176
177
178
179
180
181
182
183
184
185
186
187
188
189
190
191
192
193
194
195
196
197
198
199
200
201
202
203
204
205
206
207
208
209
210
211
212
213
214
215
216
217
218
219
220
221
222
223
224
225
226
227
228
229
230
231
232
233
234
235
236
237
238
239
240
241
242
243
244
245
246
247
248
249
250
251
252
253
254
255
256
257
258
259
260
261
262
263
264
265
266
267
268
269
270
271
272
273
274
275
276
277
278
279
280
281
282
283
284
285
286
287
288
289
290
291
292
293
294
295
296
297
298
299
300
301
302
303
304
305
306
307
308
309
310
311
312
313
314
315
316
317
318
319
320
321
322
323
324
325
326
327
328
329
330
331
332
333
334
335
336
337
338
339
340
341
342
343
344
345
346
347
348
349
350
351
352
353
354
355
356
357
358
359
360
361
362
363
364
365
366
367
368
369
370
371
372
373
374
375
376
377
378
379
380
381
382
383
384
385
386
387
388
389
390
391
392
393
394
395
396
397
398
399
400
401
402
403
404
405
406
407
408
409
410
411
412
413
414
415
416
417
418
419
420
421
422
423
424
425
426
427
428
429
430
431
432
433
434
435
436
437
438
439
440
441
442
443
444
445
446
447
448
449
450
451
452
453
454
455
456
457
458
459
460
461
462
463
464
465
466
467
468
469
470
471
472
473
474
475
476
477
478
479
480
481
482
483
484
485
486
487
488
489
490
491
492
493
494
495
496
497
498
499
500
501
502
503
504
505
506
507
508
509
510
511
512
513
514
515
516
517
518
519
520
521
522
523
524
525
526
527
528
529
530
531
532
533
534
535
536
537
538
539
540
541
542
543
544
545
546
547
548
549
550
551
552
553
554
555
556
557
558
559
560
561
562
563
564
565
566
567
568
569
570
571
572
573
574
575
576
577
578
579
580
581
582
583
584
585
586
587
588
589
590
591
592
593
594
595
596
597
598
599
600
601
602
603
604
605
606
607
608
609
610
611
612
613
614
615
616
617
618
619
620
621
622
623
624
625
626
627
628
629
630
631
632
633
634
635
636
637
638
639
640
641
642
643
644
645
646
647
648
649
650
651
652
653
654
655
656
657
658
659
660
661
662
663
664
665
666
667
668
669
670
671
672
673
674
675
676
677
678
679
680
681
682
683
684
685
686
687
688
689
690
691
692
693
694
695
696
697
698
699
700
701
702
703
704
705
706
707
708
709
710
711
712
713
714
715
716
717
718
719
720
721
722
723
724
725
726
727
728
729
730
731
732
733
734
735
736
737
738
739
740
741
742
743
744
745
746
747
748
749
750
751
752
753
754
755
756
757
758
759
760
761
762
763
764
765
766
767
768
769
770
771
772
773
774
775
776
777
778
779
780
781
782
783
784
785
786
787
788
789
790
791
792
793
794
795
796
797
798
799
800
801
802
803
804
805
806
807
808
809
810
811
812
813
814
815
816
817
818
819
820
821
822
823
824
825
826
827
828
829
830
831
832
833
834
835
836
837
838
839
840
841
842
843
844
845
846
847
848
849
850
851
852
853
854
855
856
857
858
859
860
861
862
863
864
865
866
867
868
869
870
871
872
873
874
875
876
877
878
879
880
881
882
883
884
885
886
887
888
889
890
891
892
893
894
895
896
897
898
899
900
901
902
903
904
905
906
907
908
909
910
911
912
913
914
915
916
917
918
919
920
921
922
923
924
925
926
927
928
929
930
931
932
933
934
935
936
937
938
939
940
941
942
943
944
945
946
947
948
949
950
951
952
953
954
955
956
957
958
959
960
961
962
963
964
965
966
967
968
969
970
971
972
973
974
975
976
977
978
979
980
981
982
983
984
985
986
987
988
989
990
991
992
993
994
995
996
997
998
999
1000
1001
1002
1003
1004
1005
1006
1007
1008
1009
1010
1011
1012
1013
1014
1015
1016
1017
1018
1019
1020
1021
1022
1023
1024
1025
1026
1027
1028
1029
1030
1031
1032
1033
1034
1035
1036
1037
1038
1039
1040
1041
1042
1043
1044
1045
1046
1047
1048
1049
1050
1051
1052
1053
1054
1055
1056
1057
1058
1059
1060
1061
1062
1063
1064
1065
1066
1067
1068
1069
1070
1071
1072
1073
1074
1075
1076
1077
1078
1079
1080
1081
1082
1083
1084
1085
1086
1087
1088
1089
1090
1091
1092
1093
1094
1095
1096
1097
1098
1099
1100
1101
1102
1103
1104
1105
1106
1107
1108
1109
1110
1111
1112
1113
1114
1115
1116
1117
1118
1119
1120
1121
1122
1123
1124
1125
1126
1127
1128
1129
1130
1131
1132
1133
1134
1135
1136
1137
1138
1139
1140
1141
1142
1143
1144
1145
1146
1147
1148
1149
1150
1151
1152
1153
1154
1155
1156
1157
1158
1159
1160
1161
1162
1163
1164
1165
1166
1167
1168
1169
1170
1171
1172
1173
1174
1175
1176
1177
1178
1179
1180
1181
1182
1183
1184
1185
1186
1187
1188
1189
1190
1191
1192
1193
1194
1195
1196
1197
1198
1199
1200
1201
1202
1203
1204
1205
1206
1207
1208
1209
1210
1211
1212
1213
1214
1215
1216
1217
1218
1219
1220
1221
1222
1223
1224
1225
1226
1227
1228
1229
1230
1231
1232
1233
1234
1235
1236
1237
1238
1239
1240
1241
1242
1243
1244
1245
1246
1247
1248
1249
1250
1251
1252
1253
1254
1255
1256
1257
1258
1259
1260
1261
1262
1263
1264
1265
1266
1267
1268
1269
1270
1271
1272
1273
1274
1275
1276
1277
1278
1279
1280
1281
1282
1283
1284
1285
1286
1287
1288
1289
1290
1291
1292
1293
1294
1295
1296
1297
1298
1299
1300
1301
1302
1303
1304
1305
1306
1307
1308
1309
1310
1311
1312
1313
1314
1315
1316
1317
1318
1319
1320
1321
1322
1323
1324
1325
1326
1327
1328
1329
1330
1331
1332
1333
1334
1335
1336
1337
1338
1339
1340
1341
1342
1343
1344
1345
1346
1347
1348
1349
1350
1351
1352
1353
1354
1355
1356
1357
1358
1359
1360
1361
1362
1363
1364
1365
1366
1367
1368
1369
1370
1371
1372
1373
1374
1375
1376
1377
1378
1379
1380
1381
1382
1383
1384
1385
1386
1387
1388
1389
1390
1391
1392
1393
1394
1395
1396
1397
1398
1399
1400
1401
1402
1403
1404
1405
1406
1407
1408
1409
1410
1411
1412
1413
1414
1415
1416
1417
1418
1419
1420
1421
1422
1423
1424
1425
1426
1427
1428
1429
1430
1431
1432
1433
1434
1435
1436
1437
1438
1439
1440
1441
1442
1443
1444
1445
1446
1447
1448
1449
1450
1451
1452
1453
1454
1455
1456
1457
1458
1459
1460
1461
1462
1463
1464
1465
1466
1467
1468
1469
1470
1471
1472
1473
1474
1475
1476
1477
1478
1479
1480
1481
1482
1483
1484
1485
1486
1487
1488
1489
1490
1491
1492
1493
1494
1495
1496
1497
1498
1499
1500
1501
1502
1503
1504
1505
1506
1507
1508
1509
1510
1511
1512
1513
1514
1515
1516
1517
1518
1519
1520
1521
1522
1523
1524
1525
1526
1527
1528
1529
1530
1531
1532
1533
1534
1535
1536
1537
1538
1539
1540
1541
1542
1543
1544
1545
1546
1547
1548
1549
1550
1551
1552
1553
1554
1555
1556
1557
1558
1559
1560
1561
1562
1563
1564
1565
1566
1567
1568
1569
1570
1571
1572
1573
1574
1575
1576
1577
1578
1579
1580
1581
1582
1583
1584
1585
1586
1587
1588
1589
1590
1591
1592
1593
1594
1595
1596
1597
1598
1599
1600
1601
1602
1603
1604
1605
1606
1607
1608
1609
1610
1611
1612
1613
1614
1615
1616
1617
1618
1619
1620
1621
1622
1623
1624
1625
1626
1627
1628
1629
1630
1631
1632
1633
1634
1635
1636
1637
1638
1639
1640
1641
1642
1643
1644
1645
1646
1647
1648
1649
1650
1651
1652
1653
1654
1655
1656
1657
1658
1659
1660
1661
1662
1663
1664
1665
1666
1667
1668
1669
1670
1671
1672
1673
1674
1675
1676
1677
1678
1679
1680
1681
1682
1683
1684
1685
1686
1687
1688
1689
1690
1691
1692
1693
1694
1695
1696
1697
1698
1699
1700
1701
1702
1703
1704
1705
1706
1707
1708
1709
1710
1711
1712
1713
1714
1715
1716
1717
1718
1719
1720
1721
1722
1723
1724
1725
1726
1727
1728
1729
1730
1731
1732
1733
1734
1735
1736
1737
1738
1739
1740
1741
1742
1743
1744
1745
1746
1747
1748
1749
1750
1751
1752
1753
1754
1755
1756
1757
1758
1759
1760
1761
1762
1763
1764
1765
1766
1767
1768
1769
1770
1771
1772
1773
1774
1775
1776
1777
1778
1779
1780
1781
1782
1783
1784
1785
1786
1787
1788
1789
1790
1791
1792
1793
1794
1795
1796
1797
1798
1799
1800
1801
1802
1803
1804
1805
1806
1807
1808
1809
1810
1811
1812
1813
1814
1815
1816
1817
1818
1819
1820
1821
1822
1823
1824
1825
1826
1827
1828
1829
1830
1831
1832
1833
1834
1835
1836
1837
1838
1839
1840
1841
1842
1843
1844
1845
1846
1847
1848
1849
1850
1851
1852
1853
1854
1855
1856
1857
1858
1859
1860
1861
1862
1863
1864
1865
1866
1867
1868
1869
1870
1871
1872
1873
1874
1875
1876
1877
1878
1879
1880
1881
1882
1883
1884
1885
1886
1887
1888
1889
1890
1891
1892
1893
1894
1895
1896
1897
1898
1899
1900
1901
1902
1903
1904
1905
1906
1907
1908
1909
1910
1911
1912
1913
1914
1915
1916
1917
1918
1919
1920
1921
1922
1923
1924
1925
1926
1927
1928
1929
1930
1931
1932
1933
1934
1935
1936
1937
1938
1939
1940
1941
1942
1943
1944
1945
1946
1947
1948
1949
1950
1951
1952
1953
1954
1955
1956
1957
1958
1959
1960
1961
1962
1963
1964
1965
1966
1967
1968
1969
1970
1971
1972
1973
1974
1975
1976
1977
1978
1979
1980
1981
1982
1983
1984
1985
1986
1987
1988
1989
1990
1991
1992
1993
1994
1995
1996
1997
1998
1999
2000
2001
2002
2003
2004
2005
2006
2007
2008
2009
2010
2011
2012
2013
2014
2015
2016
2017
2018
2019
2020
2021
2022
2023
2024
2025
2026
2027
2028
2029
2030
2031
2032
2033
2034
2035
2036
2037
2038
2039
2040
2041
2042
2043
2044
2045
2046
2047
2048
2049
2050
2051
2052
2053
2054
2055
2056
2057
2058
2059
2060
2061
2062
2063
2064
2065
2066
2067
2068
2069
2070
2071
2072
2073
2074
2075
2076
2077
2078
2079
2080
2081
2082
2083
2084
2085
2086
2087
2088
2089
2090
2091
2092
2093
2094
2095
2096
2097
2098
2099
2100
2101
2102
2103
2104
2105
2106
2107
2108
2109
2110
2111
2112
2113
2114
2115
2116
2117
2118
2119
2120
2121
2122
2123
2124
2125
2126
2127
2128
2129
2130
2131
2132
2133
2134
2135
2136
2137
2138
2139
2140
2141
2142
2143
2144
2145
2146
2147
2148
2149
2150
2151
2152
2153
2154
2155
2156
2157
2158
2159
2160
2161
2162
2163
2164
2165
2166
2167
2168
2169
2170
2171
2172
2173
2174
2175
2176
2177
2178
2179
2180
2181
2182
2183
2184
2185
2186
2187
2188
2189
2190
2191
2192
2193
2194
2195
2196
2197
2198
2199
2200
2201
2202
2203
2204
2205
2206
2207
2208
2209
2210
2211
2212
2213
2214
2215
2216
2217
2218
2219
2220
2221
2222
2223
2224
2225
2226
2227
2228
2229
2230
2231
2232
2233
2234
2235
2236
2237
2238
2239
2240
2241
2242
2243
2244
2245
2246
2247
2248
2249
2250
2251
2252
2253
2254
2255
2256
2257
2258
2259
2260
2261
2262
2263
2264
2265
2266
2267
2268
2269
2270
2271
2272
2273
2274
2275
2276
2277
2278
2279
2280
2281
2282
2283
2284
2285
2286
2287
2288
2289
2290
2291
2292
2293
2294
2295
2296
2297
2298
2299
2300
2301
2302
2303
2304
2305
2306
2307
2308
2309
2310
2311
2312
2313
2314
2315
2316
2317
2318
2319
2320
2321
2322
2323
2324
2325
2326
2327
2328
2329
2330
2331
2332
2333
2334
2335
2336
2337
2338
2339
2340
2341
2342
2343
2344
2345
2346
2347
2348
2349
2350
2351
2352
2353
2354
2355
2356
2357
2358
2359
2360
2361
2362
2363
2364
2365
2366
2367
2368
2369
2370
2371
2372
2373
2374
2375
2376
2377
2378
2379
2380
2381
2382
2383
2384
2385
2386
2387
2388
2389
2390
2391
2392
2393
2394
2395
2396
2397
2398
2399
2400
2401
2402
2403
2404
2405
2406
2407
2408
2409
2410
2411
2412
2413
2414
2415
2416
2417
2418
2419
2420
2421
2422
2423
2424
2425
2426
2427
2428
2429
2430
2431
2432
2433
2434
2435
2436
2437
2438
2439
2440
2441
2442
2443
2444
2445
2446
2447
2448
2449
2450
2451
2452
2453
2454
2455
2456
2457
2458
2459
2460
2461
2462
2463
2464
2465
2466
2467
2468
2469
2470
2471
2472
2473
2474
2475
2476
2477
2478
2479
2480
2481
2482
2483
2484
2485
2486
2487
2488
2489
2490
2491
2492
2493
2494
2495
2496
2497
2498
2499
2500
2501
2502
2503
2504
2505
2506
2507
2508
2509
2510
2511
2512
2513
2514
2515
2516
2517
2518
2519
2520
2521
2522
2523
2524
2525
2526
2527
2528
2529
2530
2531
2532
2533
2534
2535
2536
2537
2538
2539
2540
2541
2542
2543
2544
2545
2546
2547
2548
2549
2550
2551
2552
2553
2554
2555
2556
2557
2558
2559
2560
2561
2562
2563
2564
2565
2566
2567
2568
2569
2570
2571
2572
2573
2574
2575
2576
2577
2578
2579
2580
2581
2582
2583
2584
2585
2586
2587
2588
2589
2590
2591
2592
2593
2594
2595
2596
2597
2598
2599
2600
2601
2602
2603
2604
2605
2606
2607
2608
2609
2610
2611
2612
2613
2614
2615
2616
2617
2618
2619
2620
2621
2622
2623
2624
2625
2626
2627
2628
2629
2630
2631
2632
2633
2634
2635
2636
2637
2638
2639
2640
2641
2642
2643
2644
2645
2646
264
```

### 5.1.14 Reading a solicited input ► R\_Solicited\_Input

This request indicates if an operator input is requested by a \$ message. In addition it also returns the current CNC mode.

```

1  PROGRAM R_GEN_013
2
3  VAR
4      oR_Solicited_Input          : R_Solicited_Input;
5      xAnswer1                   : BOOL;
6      enumAnswer2                 : EnumMode;
7  END_VAR

1  oR_Solicited_Input.CNCAdr       := ADR(NCK);
2  oR_Solicited_Input.byChannel    := 0;
3                                  // Channels numbered 0 to 7
4  oR_Solicited_Input.byAxisNo     := 0;
5
6  oR_Solicited_Input(enable := xEnable);
7                                  // Trigger the request
8
9
10 IF oR_Solicited_Input.uiResult = 0 THEN
11     xEnable           := FALSE;
12     xAnswer1          := oR_Solicited_Input.boSolInput;
13     enumAnswer2       := oR_Solicited_Input.EnumMode;
14 END_IF

```

### 5.1.15 Reading the measure of a spindle ► R\_Spindle\_Meas

This request returns the current measure of a spindle according to the spindle physical address.

```
R_GEN_014 x
1 PROGRAM R_GEN_014
2
3 VAR
4     oR_Spindle_Meas          : R_Spindle_Meas;
5     xAnswer1                 : BOOL;
6     ui_Answer2               : UDINT;
7     xAnswer0                 : BOOL;
8 END_VAR
9
10
11 oR_Spindle_Meas.CNCAdr       := ADR(NCK);
12 oR_Spindle_Meas.bySpindleNo := 31;
13
14 oR_Spindle_Meas(enable := xEnable);
15                                     // Trigger the request
16
17 IF oR_Spindle_Meas.uiResult = 0 THEN
18     xEnable          := FALSE;
19     xAnswer1         := oR_Spindle_Meas.ok;
20     ui_Answer2       := oR_Spindle_Meas.uiMeas;
21     xAnswer0         := oR_Spindle_Meas.ok;
22 END_IF
```

### 5.1.16 Reading the tool offset H dimension ► R\_Tool\_H

This request will return the H value of 10 consecutive tool offsets.

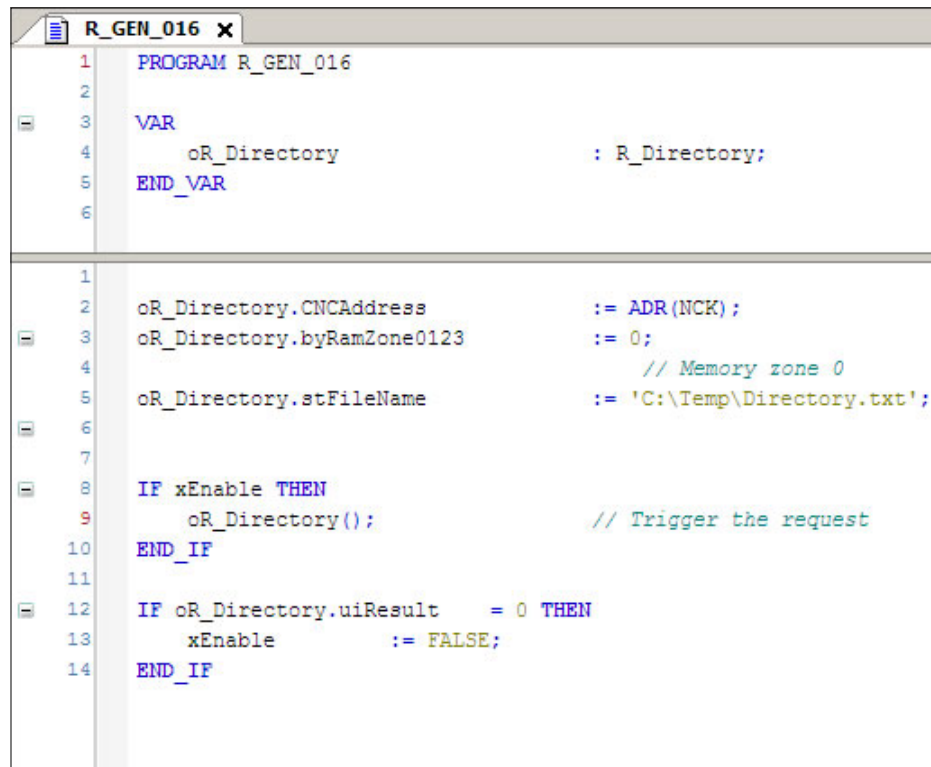
```

1  PROGRAM R_GEN_015
2
3  VAR
4      oR_Tool_H                : R_Tool_H;
5      wo_Answer1               : DINT;
6      wo_Answer2               : DINT;
7      wo_Answer3               : DINT;
8      wo_Answer4               : DINT;
9      wo_Answer5               : DINT;
10     wo_Answer6               : DINT;
11     wo_Answer7               : DINT;
12     wo_Answer8               : DINT;
13     wo_Answer9               : DINT;
14     wo_Answer10              : DINT;
15 END_VAR
16
17
18
19
20
21
22
23
24
25
26
27
28
29
30
31
32
33
34
35
36
37
38
39
40
41
42
43
44
45
46
47
48
49
50
51
52
53
54
55
56
57
58
59
60
61
62
63
64
65
66
67
68
69
70
71
72
73
74
75
76
77
78
79
80
81
82
83
84
85
86
87
88
89
90
91
92
93
94
95
96
97
98
99
100
101
102
103
104
105
106
107
108
109
110
111
112
113
114
115
116
117
118
119
120
121
122
123
124
125
126
127
128
129
130
131
132
133
134
135
136
137
138
139
140
141
142
143
144
145
146
147
148
149
150
151
152
153
154
155
156
157
158
159
160
161
162
163
164
165
166
167
168
169
170
171
172
173
174
175
176
177
178
179
180
181
182
183
184
185
186
187
188
189
190
191
192
193
194
195
196
197
198
199
200
201
202
203
204
205
206
207
208
209
210
211
212
213
214
215
216
217
218
219
220
221
222
223
224
225
226
227
228
229
230
231
232
233
234
235
236
237
238
239
240
241
242
243
244
245
246
247
248
249
250
251
252
253
254
255
256
257
258
259
260
261
262
263
264
265
266
267
268
269
270
271
272
273
274
275
276
277
278
279
280
281
282
283
284
285
286
287
288
289
290
291
292
293
294
295
296
297
298
299
300
301
302
303
304
305
306
307
308
309
310
311
312
313
314
315
316
317
318
319
320
321
322
323
324
325
326
327
328
329
330
331
332
333
334
335
336
337
338
339
340
341
342
343
344
345
346
347
348
349
350
351
352
353
354
355
356
357
358
359
360
361
362
363
364
365
366
367
368
369
370
371
372
373
374
375
376
377
378
379
380
381
382
383
384
385
386
387
388
389
390
391
392
393
394
395
396
397
398
399
400
401
402
403
404
405
406
407
408
409
410
411
412
413
414
415
416
417
418
419
420
421
422
423
424
425
426
427
428
429
430
431
432
433
434
435
436
437
438
439
440
441
442
443
444
445
446
447
448
449
450
451
452
453
454
455
456
457
458
459
460
461
462
463
464
465
466
467
468
469
470
471
472
473
474
475
476
477
478
479
480
481
482
483
484
485
486
487
488
489
490
491
492
493
494
495
496
497
498
499
500
501
502
503
504
505
506
507
508
509
510
511
512
513
514
515
516
517
518
519
520
521
522
523
524
525
526
527
528
529
530
531
532
533
534
535
536
537
538
539
540
541
542
543
544
545
546
547
548
549
550
551
552
553
554
555
556
557
558
559
560
561
562
563
564
565
566
567
568
569
570
571
572
573
574
575
576
577
578
579
580
581
582
583
584
585
586
587
588
589
590
591
592
593
594
595
596
597
598
599
600
601
602
603
604
605
606
607
608
609
610
611
612
613
614
615
616
617
618
619
620
621
622
623
624
625
626
627
628
629
630
631
632
633
634
635
636
637
638
639
640
641
642
643
644
645
646
647
648
649
650
651
652
653
654
655
656
657
658
659
660
661
662
663
664
665
666
667
668
669
670
671
672
673
674
675
676
677
678
679
680
681
682
683
684
685
686
687
688
689
690
691
692
693
694
695
696
697
698
699
700
701
702
703
704
705
706
707
708
709
710
711
712
713
714
715
716
717
718
719
720
721
722
723
724
725
726
727
728
729
730
731
732
733
734
735
736
737
738
739
740
741
742
743
744
745
746
747
748
749
750
751
752
753
754
755
756
757
758
759
760
761
762
763
764
765
766
767
768
769
770
771
772
773
774
775
776
777
778
779
780
781
782
783
784
785
786
787
788
789
790
791
792
793
794
795
796
797
798
799
800
801
802
803
804
805
806
807
808
809
810
811
812
813
814
815
816
817
818
819
820
821
822
823
824
825
826
827
828
829
830
831
832
833
834
835
836
837
838
839
840
841
842
843
844
845
846
847
848
849
850
851
852
853
854
855
856
857
858
859
860
861
862
863
864
865
866
867
868
869
870
871
872
873
874
875
876
877
878
879
880
881
882
883
884
885
886
887
888
889
890
891
892
893
894
895
896
897
898
899
900
901
902
903
904
905
906
907
908
909
910
911
912
913
914
915
916
917
918
919
920
921
922
923
924
925
926
927
928
929
930
931
932
933
934
935
936
937
938
939
940
941
942
943
944
945
946
947
948
949
950
951
952
953
954
955
956
957
958
959
960
961
962
963
964
965
966
967
968
969
970
971
972
973
974
975
976
977
978
979
980
981
982
983
984
985
986
987
988
989
990
991
992
993
994
995
996
997
998
999
1000
1001
1002
1003
1004
1005
1006
1007
1008
1009
1010
1011
1012
1013
1014
1015
1016
1017
1018
1019
1020
1021
1022
1023
1024
1025
1026
1027
1028
1029
1030
1031
1032
1033
1034
1035
1036
1037
1038
1039
1040
1041
1042
1043
1044
1045
1046
1047
1048
1049
1050
1051
1052
1053
1054
1055
1056
1057
1058
1059
1060
1061
1062
1063
1064
1065
1066
1067
1068
1069
1070
1071
1072
1073
1074
1075
1076
1077
1078
1079
1080
1081
1082
1083
1084
1085
1086
1087
1088
1089
1090
1091
1092
1093
1094
1095
1096
1097
1098
1099
1100
1101
1102
1103
1104
1105
1106
1107
1108
1109
1110
1111
1112
1113
1114
1115
1116
1117
1118
1119
1120
1121
1122
1123
1124
1125
1126
1127
1128
1129
1130
1131
1132
1133
1134
1135
1136
1137
1138
1139
1140
1141
1142
1143
1144
1145
1146
1147
1148
1149
1150
1151
1152
1153
1154
1155
1156
1157
1158
1159
1160
1161
1162
1163
1164
1165
1166
1167
1168
1169
1170
1171
1172
1173
1174
1175
1176
1177
1178
1179
1180
1181
1182
1183
1184
1185
1186
1187
1188
1189
1190
1191
1192
1193
1194
1195
1196
1197
1198
1199
1200
1201
1202
1203
1204
1205
1206
1207
1208
1209
1210
1211
1212
1213
1214
1215
1216
1217
1218
1219
1220
1221
1222
1223
1224
1225
1226
1227
1228
1229
1230
1231
1232
1233
1234
1235
1236
1237
1238
1239
1240
1241
1242
1243
1244
1245
1246
1247
1248
1249
1250
1251
1252
1253
1254
1255
1256
1257
1258
1259
1260
1261
1262
1263
1264
1265
1266
1267
1268
1269
1270
1271
1272
1273
1274
1275
1276
1277
1278
1279
1280
1281
1282
1283
1284
1285
1286
1287
1288
1289
1290
1291
1292
1293
1294
1295
1296
1297
1298
1299
1300
1301
1302
1303
1304
1305
1306
1307
1308
1309
1310
1311
1312
1313
1314
1315
1316
1317
1318
1319
1320
1321
1322
1323
1324
1325
1326
1327
1328
1329
1330
1331
1332
1333
1334
1335
1336
1337
1338
1339
1340
1341
1342
1343
1344
1345
1346
1347
1348
1349
1350
1351
1352
1353
1354
1355
1356
1357
1358
1359
1360
1361
1362
1363
1364
1365
1366
1367
1368
1369
1370
1371
1372
1373
1374
1375
1376
1377
1378
1379
1380
1381
1382
1383
1384
1385
1386
1387
1388
1389
1390
1391
1392
1393
1394
1395
1396
1397
1398
1399
1400
1401
1402
1403
1404
1405
1406
1407
1408
1409
1410
1411
1412
1413
1414
1415
1416
1417
1418
1419
1420
1421
1422
1423
1424
1425
1426
1427
1428
1429
1430
1431
1432
1433
1434
1435
1436
1437
1438
1439
1440
1441
1442
1443
1444
1445
1446
1447
1448
1449
1450
1451
1452
1453
1454
1455
1456
1457
1458
1459
1460
1461
1462
1463
1464
1465
1466
1467
1468
1469
1470
1471
1472
1473
1474
1475
1476
1477
1478
1479
1480
1481
1482
1483
1484
1485
1486
1487
1488
1489
1490
1491
1492
1493
1494
1495
1496
1497
1498
1499
1500
1501
1502
1503
1504
1505
1506
1507
1508
1509
1510
1511
1512
1513
1514
1515
1516
1517
1518
1519
1520
1521
1522
1523
1524
1525
1526
1527
1528
1529
1530
1531
1532
1533
1534
1535
1536
1537
1538
1539
1540
1541
1542
1543
1544
1545
1546
1547
1548
1549
1550
1551
1552
1553
1554
1555
1556
1557
1558
1559
1560
1561
1562
1563
1564
1565
1566
1567
1568
1569
1570
1571
1572
1573
1574
1575
1576
1577
1578
1579
1580
1581
1582
1583
1584
1585
1586
1587
1588
1589
1590
1591
1592
1593
1594
1595
1596
1597
1598
1599
1600
1601
1602
1603
1604
1605
1606
1607
1608
1609
1610
1611
1612
1613
1614
1615
1616
1617
1618
1619
1620
1621
1622
1623
1624
1625
1626
1627
1628
1629
1630
1631
1632
1633
1634
1635
1636
1637
1638
1639
1640
1641
1642
1643
1644
1645
1646
1647
1648
1649
1650
1651
1652
1653
1654
1655
1656
1657
1658
1659
1660
1661
1662
1663
1664
1665
1666
1667
1668
1669
1670
1671
1672
1673
1674
1675
1676
1677
1678
1679
1680
1681
1682
1683
1684
1685
1686
1687
1688
1689
1690
1691
1692
1693
1694
1695
1696
1697
1698
1699
1700
1701
1702
1703
1704
1705
1706
1707
1708
1709
1710
1711
1712
1713
1714
1715
1716
1717
1718
1719
1720
1721
1722
1723
1724
1725
1726
1727
1728
1729
1730
1731
1732
1733
1734
1735
1736
1737
1738
1739
1740
1741
1742
1743
1744
1745
1746
1747
1748
1749
1750
1751
1752
1753
1754
1755
1756
1757
1758
1759
1760
1761
1762
1763
1764
1765
1766
1767
1768
1769
1770
1771
1772
1773
1774
1775
1776
1777
1778
1779
1780
1781
1782
1783
1784
1785
1786
1787
1788
1789
1790
1791
1792
1793
1794
1795
1796
1797
1798
1799
1800
1801
1802
1803
1804
1805
1806
1807
1808
1809
1810
1811
1812
1813
1814
1815
1816
1817
1818
1819
1820
1821
1822
1823
1824
1825
1826
1827
1828
1829
1830
1831
1832
1833
1834
1835
1836
1837
1838
1839
1840
1841
1842
1843
1844
1845
1846
1847
1848
1849
1850
1851
1852
1853
1854
1855
1856
1857
1858
1859
1860
1861
1862
1863
1864
1865
1866
1867
1868
1869
1870
1871
1872
1873
1874
1875
1876
1877
1878
1879
1880
1881
1882
1883
1884
1885
1886
1887
1888
1889
1890
1891
1892
1893
1894
1895
1896
1897
1898
1899
1900
1901
1902
1903
1904
1905
1906
1907
1908
1909
1910
1911
1912
1913
1914
1915
1916
1917
1918
1919
1920
1921
1922
1923
1924
1925
1926
1927
1928
1929
1930
1931
1932
1933
1934
1935
1936
1937
1938
1939
1940
1941
1942
1943
1944
1945
1946
1947
1948
1949
1950
1951
1952
1953
1954
1955
1956
1957
1958
1959
1960
1961
1962
1963
1964
1965
1966
1967
1968
1969
1970
1971
1972
1973
1974
1975
1976
1977
1978
1979
1980
1981
1982
1983
1984
1985
1986
1987
1988
1989
1990
1991
1992
1993
1994
1995
1996
1997
1998
1999
2000
2001
2002
2003
2004
2005
2006
2007
2008
2009
2010
2011
2012
2013
2014
2015
2016
2017
2018
2019
2020
2021
2022
2023
2024
2025
2026
2027
2028
2029
2030
2031
2032
2033
2034
2035
2036
2037
2038
2039
2040
2041
2042
2043
2044
2045
2046
2047
2048
2049
2050
2051
2052
2053
2054
2055
2056
2057
2058
2059
2060
2061
2062
2063
2064
2065
2066
2067
2068
2069
2070
2071
2072
2073
2074
2075
2076
2077
2078
2079
2080
2081
2082
2083
2084
2085
2086
2087
2088
2089
2090
2091
2092
2093
2094
2095
2096
2097
2098
2099
2100
2101
2102
2103
2104
2105
2106
2107
2108
2109
2110
2111
2112
2113
2114
2115
2116
2117
2118
2119
2120
2121
2122
2123
2124
2125
2126
2127
2128
2129
2130
2131
2132
2133
2134
2135
2136
2137
2138
2139
2140
2141
2142
2143
2144
2145
2146
2147
2148
2149
2150
2151
2152
2153
2154
2155
2156
2157
2158
2159
2160
2161
2162
2163
2164
2165
2166
2167
2168
2169
2170
2171
2172
2173
2174
2175
2176
2177
2178
2179
2180
2181
2182
2183
2184
2185
2186
2187
2188
2189
2190
2191
2192
2193
2194
2195
2196
2197
2198
2199
2200
2201
2202
2203
2204
2205
2206
2207
2208
2209
2210
2211
2212
2213
2214
2215
2216
2217
2218
2219
2220
2221
2222
2223
2224
2225
2226
2227
2228
2229
2230
2231
2232
2233
2234
2235
2236
2237
2238
2239
2240
2241
2242
2243
2244
2245
2246
2247
2248
2249
2250
2251
2252
2253
2254
2255
2256
2257
2258
2259
2260
2261
2262
2263
2264
2265
2266
2267
2268
2269
2270
2271
2272
2273
2274
2275
2276
2277
2278
2279
2280
2281
2282
2283
2284
2285
2286
2287
2288
2289
2290
2291
2292
2293
2294
2295
2296
2297
2298
2299
2300
2301
2302
2303
2304
2305
2306
2307
2308
2309
2310
2311
2312
2313
2314
2315
2316
2317
2318
2319
2320
2321
2322
2323
2324
2325
2326
2327
2328
2329
2330
2331
2332
2333
2334
2335
2336
2337
2338
2339
2340
2341
2342
2343
2344
2345
2346
2347
2348
2349
2350
2351
2352
2353
2354
2355
2356
2357
2358
2359
2360
2361
2362
2363
2364
2365
2366
2367
2368
2369
2370
2371
2372
2373
2374
2375
2376
2377
2378
2379
2380
2381
2382
2383
2384
2385
2386
2387
2388
2389
2390
2391
2392
2393
2394
2395
2396
2397
2398
2399
2400
2401
2402
2403
2404
2405
2406
2407
2408
2409
2410
2411
2412
2413
2414
2415
2416
2417
2418
2419
2420
2421
2422
2423
2424
2425
2426
2427
2428
2429
2430
2431
2432
2433
2434
2435
2436
2437
2438
2439
2440
2441
2442
2443
2444
2445
2446
2447
2448
2449
2450
2451
2452
2453
2454
2455
2456
2457
2458
2459
2460
2461
2462
2463
2464
2465
2466
2467
2468
2469
2470
2471
2472
2473
2474
2475
2476
2477
2478
2479
2480
2481
2482
2483
2484
2485
2486
2487
2488
2489
2490
2491
2492
2493
2494
2495
2496
2497
2498
2499
2500
2501
2502
2503
2504
2505
2506
2507
2508
2509
2510
2511
2512
2513
2514
2515
2516
2517
2518
2519
2520
2521
2522
2523
2524
2525
2526
2527
2528
2529
2530
2531
2532
2533
2534
2535
2536
2537
2538
2539
2540
2541
2542
2543
2544
2545
2546
2547
2548
2549
2550
2551
2552
2553
2554
2555
2556
2557
2558
2559
2560
2561
2562
2563
2564
2565
2566
2567
2568
2569
2570
2571
2572
2573
2574
2575
2576
2577
2578
2579
2580
2581
2582
2583
2584
2585
2586
2587
2588
2589
2590
2591
2592
2593
2594
2595
2596
2597
2598
2599
2600
2601
2602
2603
2604
2605
2606
2607
2608
2609
2610
2611
2612
2613
2614
2615
2616
2617
2618
2619
2620
2621
2622
2623
2624
2625
2626
2627
2628
2629
2630
2631
2632
2633
2634
2635
2636
2637
2638
2639
264
```

### 5.1.17 Reading a memory zone directory ► [R\\_Directory](#)

This request will return the list of programs of a particular memory zone. The data are stored in a file on the system mass memory.

As this list can be long the request can take a certain time to get the complete list. The reading is complete when the *iuResult* flag is zero.



```
1 PROGRAM R_GEN_016
2
3 VAR
4   oR_Directory          : R_Directory;
5 END_VAR
6
7
8 IF xEnable THEN
9   oR_Directory();        // Trigger the request
10 END_IF
11
12 IF oR_Directory.iuResult = 0 THEN
13   xEnable                := FALSE;
14 END_IF
```

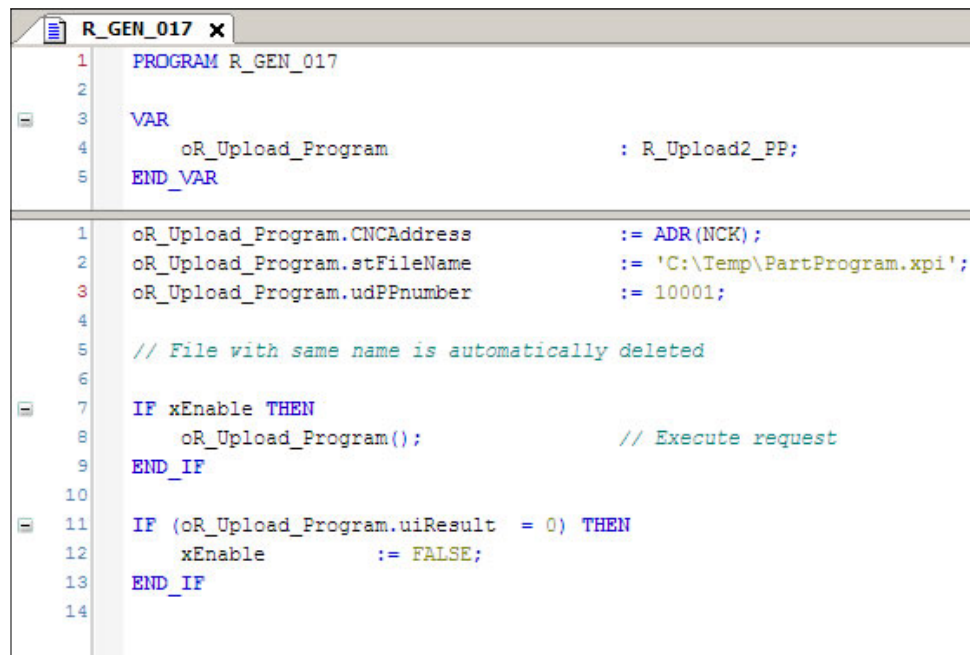
### 5.1.18 Read a part program from NC memory ► [R\\_Upload2\\_PP](#)

This request replaces the previous *R\_Upload\_PP*.  
It will read a part program in NC memory and store it as a file in the system mass memory.

The request requires the number of the part program in NC memory (zone 0) as well as the path name used to store it in mass memory.

The part program number passed to the request is separated in two elements. The last digit represents the index and the other digits the part program number itself.

The part program is always read from memory zone 0.



```

1  PROGRAM R_GEN_017
2
3  VAR
4      oR_Upload_Program          : R_Upload2_PP;
5  END_VAR

6
7  oR_Upload_Program.CNCAddress    := ADR(NCK);
8  oR_Upload_Program.stFileName    := 'C:\Temp\PartProgram.xpi';
9  oR_Upload_Program.udPPnumber    := 10001;
10
11 // File with same name is automatically deleted
12
13 IF xEnable THEN
14     oR_Upload_Program();          // Execute request
15 END_IF

16
17 IF (oR_Upload_Program.uiResult = 0) THEN
18     xEnable                      := FALSE;
19 END_IF
20

```

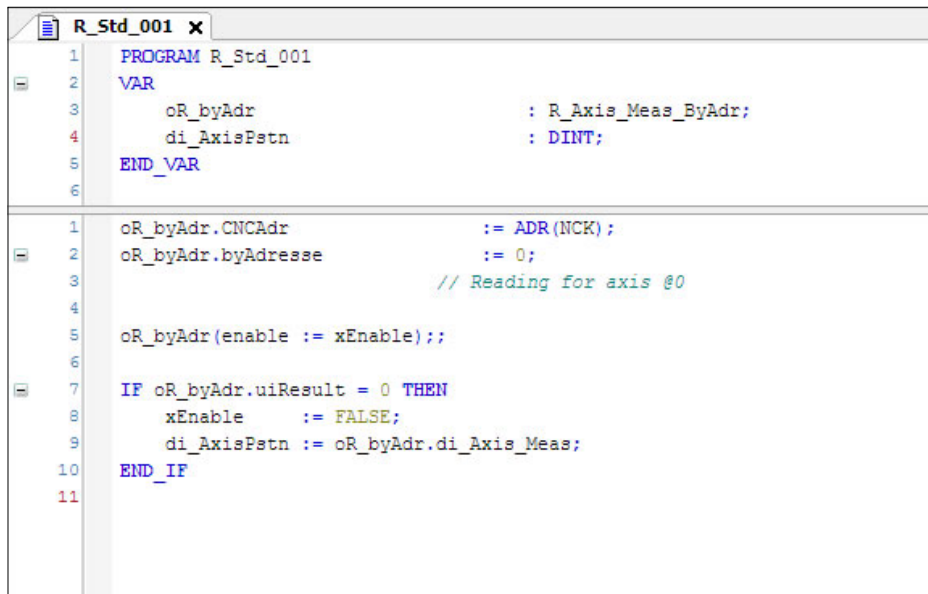
---

## 5.2 Flexium requests

### 5.2.1 Reading an axis measure by address ► [R\\_Axis\\_Meas\\_ByAdr](#)

This request will return an axis measure according to the axis physical address. It returns the data for one axis at a time.

To get the data for all axes of a channel, use *Read\_Axis\_Meas\_ByAdr*.



```
1  PROGRAM R_Std_001
2  VAR
3      oR_byAdr          : R_Axis_Meas_ByAdr;
4      di_AxisPstn       : DINT;
5  END_VAR
6
7  oR_byAdr.CNCAdr       := ADR(NCK);
8  oR_byAdr.byAdresse    := 0;
9                        // Reading for axis @0
10
11  oR_byAdr(enable := xEnable);;
12
13  IF oR_byAdr.uiResult = 0 THEN
14      xEnable          := FALSE;
15      di_AxisPstn := oR_byAdr.di_Axis_Meas;
16  END_IF
17
```

### 5.2.2 Reading the measure of all axes in a channel ▶ [R\\_Axis\\_Meas](#)

This request will return the measure of all axes in a particular channel.

Contrary to *R\_Axis\_Meas\_ByAdr* the data are sorted according to the axis name and not the physical address.

```

1  PROGRAM R_Std_002
2  VAR
3      oR_Meas          : R_Axis_Meas;
4      diMeas_X         : DINT;
5  END_VAR
6
7  oR_Meas.CNCAdr       := ADR(NCK);
8  oR_Meas.byChannel    := 0;
9                      // Channels numbered 0 to 7
10
11  oR_Meas(enable := xEnable);
12
13  IF oR_Meas.uiResult = 0 THEN
14      xEnable          := FALSE;
15      diMeas_X         := oR_Meas.diX_Axis;
16  END_IF
17

```

### 5.2.3 Reading machine data about axes in motion ► R\_Axis\_OM

This request returns Position in the machine referential as well as distance to go, lag and resolution about the axes of a particular channel.

The data are returned in a structure *Axis[i]* and are individually validated by a flag *xxxValid* e.g. *LagValid*.

```
1  PROGRAM R_Std_003
2  VAR
3      oR_Meas_OM          : R_Axis_OM;
4      Pos_0               : DINT;
5      ToGo_0              : DINT;
6      Lag_0               : DINT;
7      Res_0               : BYTE;
8  END_VAR
9
10 oR_Meas_OM.CNCAdr        := ADR(NCK);
11 oR_Meas_OM.byChannel     := 0;
12                          // Channels numbered 0 to 7
13
14 oR_Meas_OM(enable := xEnable);
15
16 IF oR_Meas_OM.uiResult = 0 THEN
17     xEnable               := FALSE;
18     IF oR_Meas_OM.Axis[0].PosValid THEN
19         Pos_0             := oR_Meas_OM.Axis[0].Position;
20     END_IF
21     IF oR_Meas_OM.Axis[0].ToGoValid THEN
22         ToGo_0            := oR_Meas_OM.Axis[0].ToGo;
23     END_IF
24     IF oR_Meas_OM.Axis[0].LagValid THEN
25         Lag_0             := oR_Meas_OM.Axis[0].Lag;
26     END_IF
27     Res_0                 := oR_Meas_OM.Axis[0].Resolution;
28 END_IF
29
```

## 5.2.4 Reading part origin data about axes in motion ▶ R\_Axis\_OP

This request returns the axes position in the part referential as well as distance to go, lag and resolution for a particular channel.

The data are returned in a structure *Axis[i]* and are individually validated by a flag *xxxValid* e.g. *LagValid*.

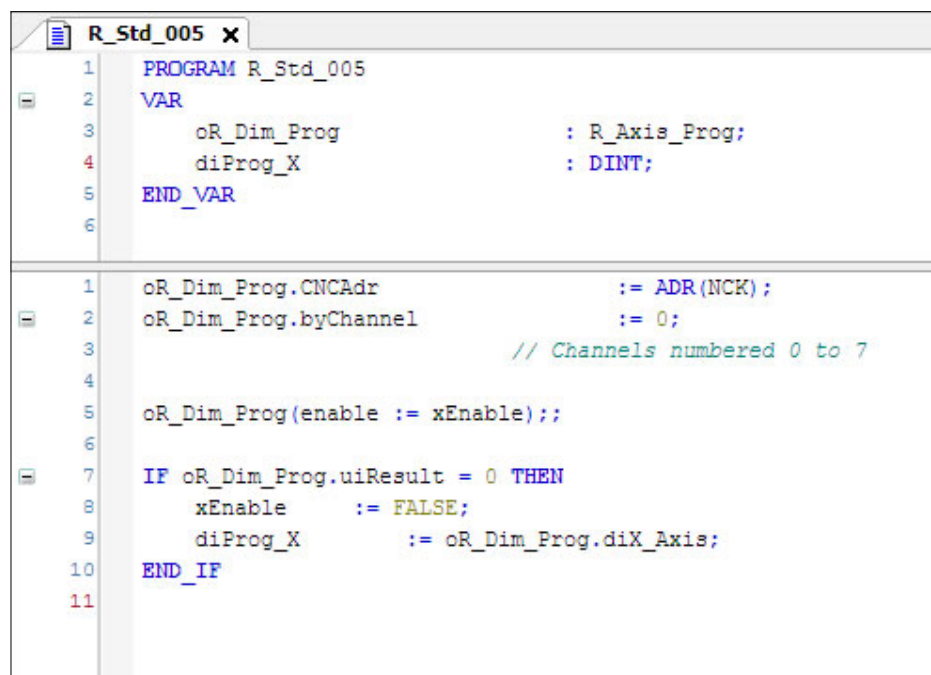
```

1  PROGRAM R_Std_004
2  VAR
3      oR_Meas_OP          : R_Axis_OP;
4      Pos_0               : DINT;
5      ToGo_0              : DINT;
6      Lag_0               : DINT;
7      Res_0               : BYTE;
8  END_VAR
9
10 oR_Meas_OP.CNCAdr        := ADR(NCK);
11 oR_Meas_OP.byChannel     := 0;
12                          // Channels numbered 0 to 7
13
14 oR_Meas_OP(enable := xEnable);
15
16 IF oR_Meas_OP.uiResult = 0 THEN
17     xEnable := FALSE;
18     IF oR_Meas_OP.Axis[0].PosValid THEN
19         Pos_0 := oR_Meas_OP.Axis[0].Position;
20     END_IF
21     IF oR_Meas_OP.Axis[0].ToGoValid THEN
22         ToGo_0 := oR_Meas_OP.Axis[0].ToGo;
23     END_IF
24     IF oR_Meas_OP.Axis[0].LagValid THEN
25         Lag_0 := oR_Meas_OP.Axis[0].Lag;
26     END_IF
27     Res_0 := oR_Meas_OP.Axis[0].Resolution;
28 END_IF

```

### 5.2.5 Reading axes programmed position ► [R\\_Axis\\_Prog](#)

This request returns the programmed position of all axes in a particular channel.  
The data are returned with respect to the axis name in the channel.



```
1 PROGRAM R_Std_005
2 VAR
3     oR_Dim_Prog      : R_Axis_Prog;
4     diProg_X         : DINT;
5 END_VAR
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24
25
26
27
28
29
30
31
32
33
34
35
36
37
38
39
40
41
42
43
44
45
46
47
48
49
50
51
52
53
54
55
56
57
58
59
60
61
62
63
64
65
66
67
68
69
70
71
72
73
74
75
76
77
78
79
80
81
82
83
84
85
86
87
88
89
90
91
92
93
94
95
96
97
98
99
100
101
102
103
104
105
106
107
108
109
110
111
112
113
114
115
116
117
118
119
120
121
122
123
124
125
126
127
128
129
130
131
132
133
134
135
136
137
138
139
140
141
142
143
144
145
146
147
148
149
150
151
152
153
154
155
156
157
158
159
160
161
162
163
164
165
166
167
168
169
170
171
172
173
174
175
176
177
178
179
180
181
182
183
184
185
186
187
188
189
190
191
192
193
194
195
196
197
198
199
200
201
202
203
204
205
206
207
208
209
210
211
212
213
214
215
216
217
218
219
220
221
222
223
224
225
226
227
228
229
230
231
232
233
234
235
236
237
238
239
240
241
242
243
244
245
246
247
248
249
250
251
252
253
254
255
256
257
258
259
260
261
262
263
264
265
266
267
268
269
270
271
272
273
274
275
276
277
278
279
280
281
282
283
284
285
286
287
288
289
290
291
292
293
294
295
296
297
298
299
300
301
302
303
304
305
306
307
308
309
310
311
312
313
314
315
316
317
318
319
320
321
322
323
324
325
326
327
328
329
330
331
332
333
334
335
336
337
338
339
340
341
342
343
344
345
346
347
348
349
350
351
352
353
354
355
356
357
358
359
360
361
362
363
364
365
366
367
368
369
370
371
372
373
374
375
376
377
378
379
380
381
382
383
384
385
386
387
388
389
390
391
392
393
394
395
396
397
398
399
400
401
402
403
404
405
406
407
408
409
410
411
412
413
414
415
416
417
418
419
420
421
422
423
424
425
426
427
428
429
430
431
432
433
434
435
436
437
438
439
440
441
442
443
444
445
446
447
448
449
450
451
452
453
454
455
456
457
458
459
460
461
462
463
464
465
466
467
468
469
470
471
472
473
474
475
476
477
478
479
480
481
482
483
484
485
486
487
488
489
490
491
492
493
494
495
496
497
498
499
500
501
502
503
504
505
506
507
508
509
510
511
512
513
514
515
516
517
518
519
520
521
522
523
524
525
526
527
528
529
530
531
532
533
534
535
536
537
538
539
540
541
542
543
544
545
546
547
548
549
550
551
552
553
554
555
556
557
558
559
560
561
562
563
564
565
566
567
568
569
570
571
572
573
574
575
576
577
578
579
580
581
582
583
584
585
586
587
588
589
590
591
592
593
594
595
596
597
598
599
600
601
602
603
604
605
606
607
608
609
610
611
612
613
614
615
616
617
618
619
620
621
622
623
624
625
626
627
628
629
630
631
632
633
634
635
636
637
638
639
640
641
642
643
644
645
646
647
648
649
650
651
652
653
654
655
656
657
658
659
660
661
662
663
664
665
666
667
668
669
670
671
672
673
674
675
676
677
678
679
680
681
682
683
684
685
686
687
688
689
690
691
692
693
694
695
696
697
698
699
700
701
702
703
704
705
706
707
708
709
710
711
712
713
714
715
716
717
718
719
720
721
722
723
724
725
726
727
728
729
730
731
732
733
734
735
736
737
738
739
740
741
742
743
744
745
746
747
748
749
750
751
752
753
754
755
756
757
758
759
760
761
762
763
764
765
766
767
768
769
770
771
772
773
774
775
776
777
778
779
780
781
782
783
784
785
786
787
788
789
790
791
792
793
794
795
796
797
798
799
800
801
802
803
804
805
806
807
808
809
810
811
812
813
814
815
816
817
818
819
820
821
822
823
824
825
826
827
828
829
830
831
832
833
834
835
836
837
838
839
840
841
842
843
844
845
846
847
848
849
850
851
852
853
854
855
856
857
858
859
860
861
862
863
864
865
866
867
868
869
870
871
872
873
874
875
876
877
878
879
880
881
882
883
884
885
886
887
888
889
890
891
892
893
894
895
896
897
898
899
900
901
902
903
904
905
906
907
908
909
910
911
912
913
914
915
916
917
918
919
920
921
922
923
924
925
926
927
928
929
930
931
932
933
934
935
936
937
938
939
940
941
942
943
944
945
946
947
948
949
950
951
952
953
954
955
956
957
958
959
960
961
962
963
964
965
966
967
968
969
970
971
972
973
974
975
976
977
978
979
980
981
982
983
984
985
986
987
988
989
990
991
992
993
994
995
996
997
998
999
1000
```

### 5.2.6 Reading axes reference ► [R\\_Axis\\_Ref](#)

This request returns the reference value (in the machine referential) of the axes in a particular channel. The data are returned with respect to the axis name.

```

1  PROGRAM R_Std_006
2  VAR
3      oR_Axis_Ref          : R_Axis_Ref;
4      diRef_X              : DINT;
5  END_VAR
6
7  oR_Axis_Ref.CNCAdr       := ADR(NCK);
8  oR_Axis_Ref.byChannel    := 0;
9                          // Channels numbered 0 to 7
10
11  oR_Axis_Ref(enable := xEnable);;
12
13  IF oR_Axis_Ref.uiResult = 0 THEN
14      xEnable              := FALSE;
15      diRef_X              := oR_Axis_Ref.diX_Axis;
16  END_IF

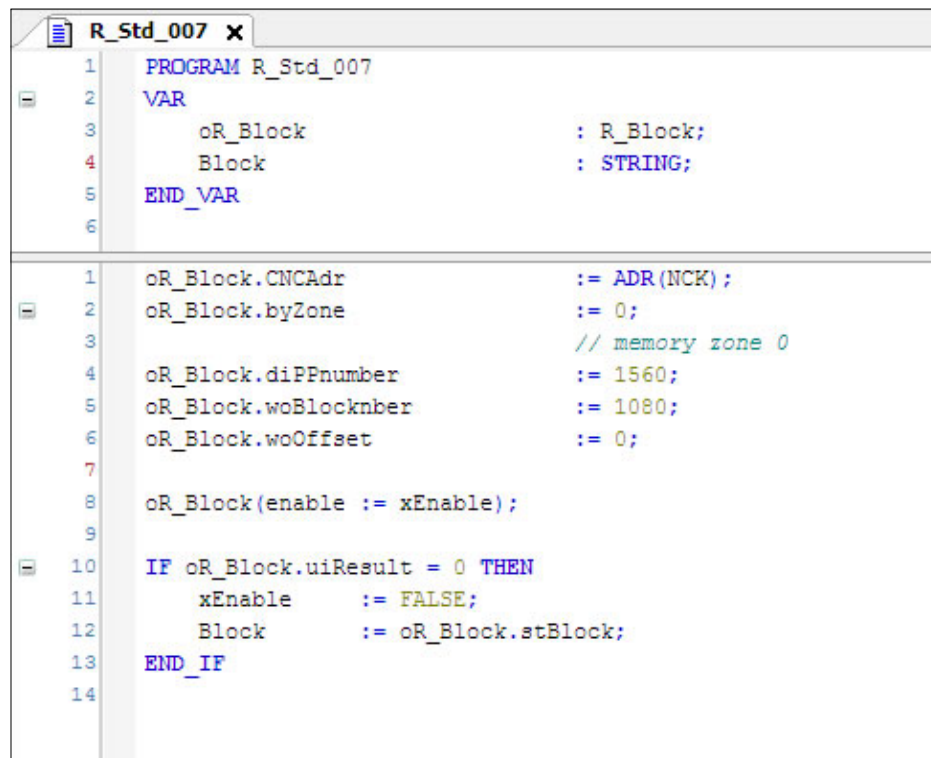
```

### 5.2.7 Reading a part program block ► R\_Block

This request returns a string corresponding to a particular part program block.

Input data will be the memory zone where the part program is searched for, the part program number (times 10 and including the index), the block number and the offset (in number of blocks) relative to this number. The string returned ends with \$N (meaning end of block).

In case of non numbered part program, the number 0 corresponds to the first line of program.



```
1 PROGRAM R_Std_007
2 VAR
3     oR_Block          : R_Block;
4     Block              : STRING;
5 END_VAR
6
7
8 oR_Block.CNCAdr        := ADR(NCK);
9 oR_Block.byZone        := 0;
10                        // memory zone 0
11 oR_Block.diPPnumber    := 1560;
12 oR_Block.woBlocknber   := 1080;
13 oR_Block.woOffset      := 0;
14
15 oR_Block(enable := xEnable);
16
17 IF oR_Block.uiResult = 0 THEN
18     xEnable            := FALSE;
19     Block               := oR_Block.stBlock;
20 END_IF
```

### 5.2.8 Reading an axis measure correction (E952xx) ► [R\\_Corr\\_Dyn](#)

This request returns the value of parameter E952xx (measure correction) of a particular axis.

```

1  PROGRAM R_Std_008
2  VAR
3      oR_Corr_Dyn          : R_Corr_Dyn;
4      diCorr_X             : DINT;
5  END_VAR
6
7  oR_Corr_Dyn.CNCAdr       := ADR(NCK) ;
8  oR_Corr_Dyn.byAxisNo    := 0;
9                          // axis 0
10 oR_Corr_Dyn(enable := xEnable);
11
12 IF oR_Corr_Dyn.uiResult = 0 THEN
13     xEnable      := FALSE;
14     dicorr_X     := oR_Corr_Dyn.diCorrDyn;
15 END_IF

```

1  
2  
3  
4  
5  
6  
A

## 5.2.9 Reading the part program block in execution ▶ R\_Curr\_Block

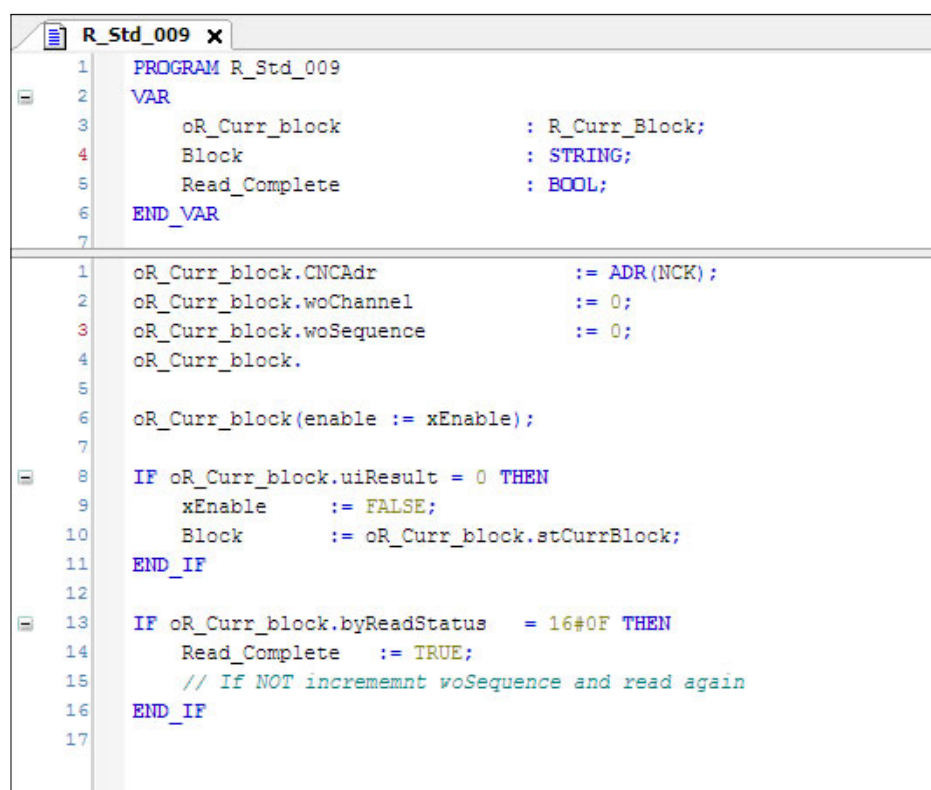
This request returns a string corresponding to the block in execution. As the answer can take several steps a Read\_Complete status flag is part of the answer.

Return code of byReadStatus:

- 0x06 : Block reading incomplete : Increment *woSequence* to get the rest of the block.
- 0x0F : Block read finished *woSequence* = 0 to read new block.
- 0x19 : Sequencing error.

The first sequence number should be 0.

If the block is not read entirely then read the next part after incrementing the sequence number : *woSequence* = *woSequence* + 1 and so on until *byReadStatus* = 0x0F.



```
1 PROGRAM R_Std_009
2 VAR
3     oR_Curr_block          : R_Curr_Block;
4     Block                  : STRING;
5     Read_Complete          : BOOL;
6 END_VAR
7
8 oR_Curr_block.CNCAdr       := ADR(NCK) ;
9 oR_Curr_block.woChannel    := 0;
10 oR_Curr_block.woSequence  := 0;
11 oR_Curr_block.
12
13 oR_Curr_block(enable := xEnable);
14
15 IF oR_Curr_block.uiResult = 0 THEN
16     xEnable      := FALSE;
17     Block        := oR_Curr_block.stCurrBlock;
18 END_IF
19
20 IF oR_Curr_block.byReadStatus = 16#0F THEN
21     Read_Complete := TRUE;
22     // If NOT increment woSequence and read again
23 END_IF
24
```

### 5.2.10 Reading current Dat1 value ► R\_Dat1\_Pref

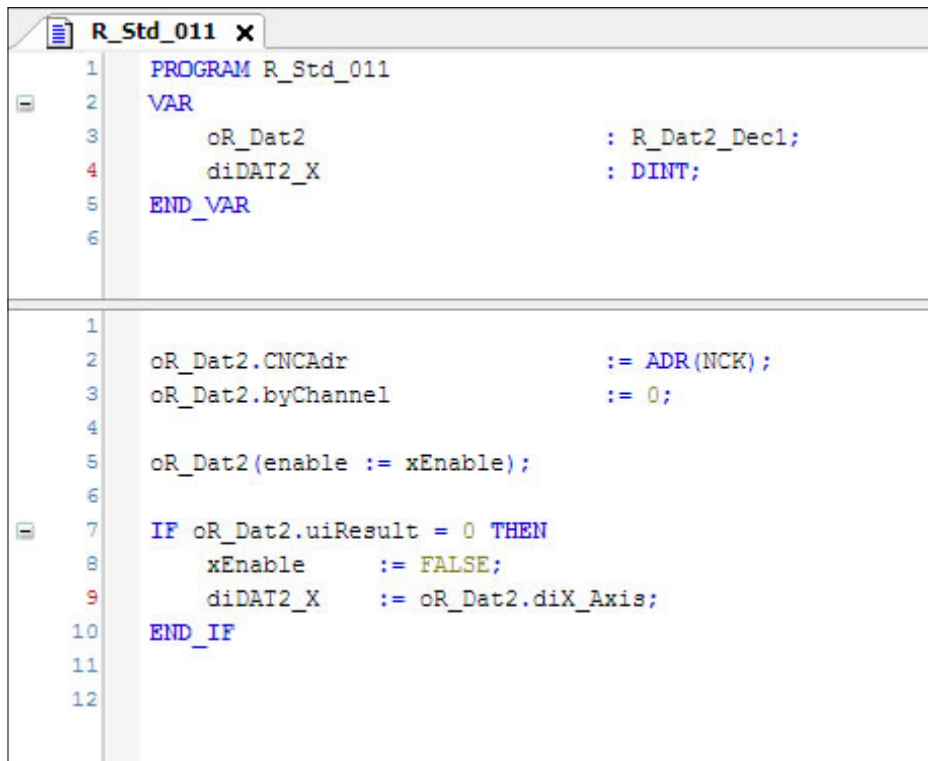
This request returns the current DAT1 set of values for a particular channel. The data are returned with respect to the axes names.

```
2  VAR
3      oR_Dat1          : R_Dat1_Pref;
4      diDAT1_X         : DINT;
5  END_VAR
6
1  oR_Dat1.CNCAdr       := ADR(NCK);
2  oR_Dat1.byChannel    := 0;
3
4  oR_Dat1(enable := xEnable);
5
6  IF oR_Dat1.uiResult = 0 THEN
7      xEnable          := FALSE;
8      diDAT1_X         := oR_Dat1.diX_Axis;
9  END_IF
10
11
12
```

---

### 5.2.11 Reading current Dat2 value ► [R\\_Dat2\\_Dec1](#)

This request returns the current DAT2 set of values for a particular channel.  
The data are returned with respect to the axes names.



```
1  PROGRAM R_Std_011
2  VAR
3      oR_Dat2                : R_Dat2_Dec1;
4      diDAT2_X               : DINT;
5  END_VAR
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24
25
26
27
28
29
30
31
32
33
34
35
36
37
38
39
40
41
42
43
44
45
46
47
48
49
50
51
52
53
54
55
56
57
58
59
60
61
62
63
64
65
66
67
68
69
70
71
72
73
74
75
76
77
78
79
80
81
82
83
84
85
86
87
88
89
90
91
92
93
94
95
96
97
98
99
100
101
102
103
104
105
106
107
108
109
110
111
112
113
114
115
116
117
118
119
120
121
122
123
124
125
126
127
128
129
130
131
132
133
134
135
136
137
138
139
140
141
142
143
144
145
146
147
148
149
150
151
152
153
154
155
156
157
158
159
160
161
162
163
164
165
166
167
168
169
170
171
172
173
174
175
176
177
178
179
180
181
182
183
184
185
186
187
188
189
190
191
192
193
194
195
196
197
198
199
200
201
202
203
204
205
206
207
208
209
210
211
212
213
214
215
216
217
218
219
220
221
222
223
224
225
226
227
228
229
230
231
232
233
234
235
236
237
238
239
240
241
242
243
244
245
246
247
248
249
250
251
252
253
254
255
256
257
258
259
260
261
262
263
264
265
266
267
268
269
270
271
272
273
274
275
276
277
278
279
280
281
282
283
284
285
286
287
288
289
290
291
292
293
294
295
296
297
298
299
300
301
302
303
304
305
306
307
308
309
310
311
312
313
314
315
316
317
318
319
320
321
322
323
324
325
326
327
328
329
330
331
332
333
334
335
336
337
338
339
340
341
342
343
344
345
346
347
348
349
350
351
352
353
354
355
356
357
358
359
360
361
362
363
364
365
366
367
368
369
370
371
372
373
374
375
376
377
378
379
380
381
382
383
384
385
386
387
388
389
390
391
392
393
394
395
396
397
398
399
400
401
402
403
404
405
406
407
408
409
410
411
412
413
414
415
416
417
418
419
420
421
422
423
424
425
426
427
428
429
430
431
432
433
434
435
436
437
438
439
440
441
442
443
444
445
446
447
448
449
450
451
452
453
454
455
456
457
458
459
460
461
462
463
464
465
466
467
468
469
470
471
472
473
474
475
476
477
478
479
480
481
482
483
484
485
486
487
488
489
490
491
492
493
494
495
496
497
498
499
500
501
502
503
504
505
506
507
508
509
510
511
512
513
514
515
516
517
518
519
520
521
522
523
524
525
526
527
528
529
530
531
532
533
534
535
536
537
538
539
540
541
542
543
544
545
546
547
548
549
550
551
552
553
554
555
556
557
558
559
560
561
562
563
564
565
566
567
568
569
570
571
572
573
574
575
576
577
578
579
580
581
582
583
584
585
586
587
588
589
590
591
592
593
594
595
596
597
598
599
600
601
602
603
604
605
606
607
608
609
610
611
612
613
614
615
616
617
618
619
620
621
622
623
624
625
626
627
628
629
630
631
632
633
634
635
636
637
638
639
640
641
642
643
644
645
646
647
648
649
650
651
652
653
654
655
656
657
658
659
660
661
662
663
664
665
666
667
668
669
670
671
672
673
674
675
676
677
678
679
680
681
682
683
684
685
686
687
688
689
690
691
692
693
694
695
696
697
698
699
700
701
702
703
704
705
706
707
708
709
710
711
712
713
714
715
716
717
718
719
720
721
722
723
724
725
726
727
728
729
730
731
732
733
734
735
736
737
738
739
740
741
742
743
744
745
746
747
748
749
750
751
752
753
754
755
756
757
758
759
760
761
762
763
764
765
766
767
768
769
770
771
772
773
774
775
776
777
778
779
780
781
782
783
784
785
786
787
788
789
790
791
792
793
794
795
796
797
798
799
800
801
802
803
804
805
806
807
808
809
810
811
812
813
814
815
816
817
818
819
820
821
822
823
824
825
826
827
828
829
830
831
832
833
834
835
836
837
838
839
840
841
842
843
844
845
846
847
848
849
850
851
852
853
854
855
856
857
858
859
860
861
862
863
864
865
866
867
868
869
870
871
872
873
874
875
876
877
878
879
880
881
882
883
884
885
886
887
888
889
890
891
892
893
894
895
896
897
898
899
900
901
902
903
904
905
906
907
908
909
910
911
912
913
914
915
916
917
918
919
920
921
922
923
924
925
926
927
928
929
930
931
932
933
934
935
936
937
938
939
940
941
942
943
944
945
946
947
948
949
950
951
952
953
954
955
956
957
958
959
960
961
962
963
964
965
966
967
968
969
970
971
972
973
974
975
976
977
978
979
980
981
982
983
984
985
986
987
988
989
990
991
992
993
994
995
996
997
998
999
1000
```

### 5.2.12 Reading current Dat3 value ► [R\\_Dat3\\_Dec3](#)

This request returns the current DAT3 set of values for a particular channel. The data are returned with respect to the axes names.

```

1  PROGRAM R_Std_012
2  VAR
3      oR_Dat3          : R_Dat3_Dec3;
4      diDAT3_X         : DINT;
5  END_VAR
6
7  oR_Dat3.CNCAdr       := ADR(NCK);
8  oR_Dat3.byChannel    := 0;
9
10 oR_Dat3(enable := xEnable);
11
12 IF oR_Dat3.uiResult = 0 THEN
13     xEnable          := FALSE;
14     diDAT3_X         := oR_Dat3.diX_Axis;
15 END_IF

```

### 5.2.13 Reading Drives Test points values ► [R\\_Drive\\_TPValue\\_V2](#)

This request returns up to eight test point values of one or several drives. The values are returned in Drive Internal units.

The test points configuration is defined in the drive parameters V130 to V133 and the values read are expressed in physical units.

```
R_Std_020 x
1 PROGRAM R_Std_020
2 VAR
3   oR_TP_Value          : R_Drive_TPValue_V2;
4   Drives                : ARRAY[1..8] OF BYTE
5                         := [0,0,0,0,1,1,1,1];
6   Points                : ARRAY[1..8] OF BYTE
7                         := [0,1,2,3,0,1,2,3];
8   Values                : ARRAY[1..8] OF LREAL;
9 END_VAR
10
11
12 oR_TP_Value.CNCAdr      := ADR(NCK);
13 oR_TP_Value.DriveAdresse := ADR(Drives);
14 oR_TP_Value.DriveTP     := ADR(Points);
15 oR_TP_Value.Number      := 4;
16
17 oR_TP_Value(enable := xEnable);
18
19 IF oR_TP_Value.uiResult = 0 THEN
20   xEnable      := FALSE;
21   Values[1]    := oR_TP_Value.Values[0];
22   Values[2]    := oR_TP_Value.Values[1];
23   Values[3]    := oR_TP_Value.Values[2];
24   Values[4]    := oR_TP_Value.Values[3];
25 END_IF
26
```

### 5.2.14 Reading a set of E8xyyy parameters ▶ R\_E8xyyy

This request returns a set of 10 E8.... parameters from the series E80, E81, E82 E83 or E84.

```

1  PROGRAM R_Std_030
2  VAR
3      oR_E8xyyy          : R_E8xyyy;
4      Para               : UDINT    := 80000;
5      Values              : ARRAY[1..10] OF DINT;
6  END_VAR
7
8  oR_E8xyyy.CNCAdr       := ADR(NCK);
9  oR_E8xyyy.udE8xyyyNo   := Para;
10
11  oR_E8xyyy(enable := xEnable);
12
13  IF oR_E8xyyy.uiResult = 0 THEN
14      xEnable           := FALSE;
15      Values[1]         := oR_E8xyyy.diE8_n0;
16      Values[2]         := oR_E8xyyy.diE8_n1;
17      Values[3]         := oR_E8xyyy.diE8_n2;
18      Values[4]         := oR_E8xyyy.diE8_n3;
19      Values[5]         := oR_E8xyyy.diE8_n4;
20      Values[6]         := oR_E8xyyy.diE8_n5;
21      Values[7]         := oR_E8xyyy.diE8_n6;
22      Values[8]         := oR_E8xyyy.diE8_n7;
23      Values[9]         := oR_E8xyyy.diE8_n8;
24      Values[10]        := oR_E8xyyy.diE8_n9;
25  END_IF

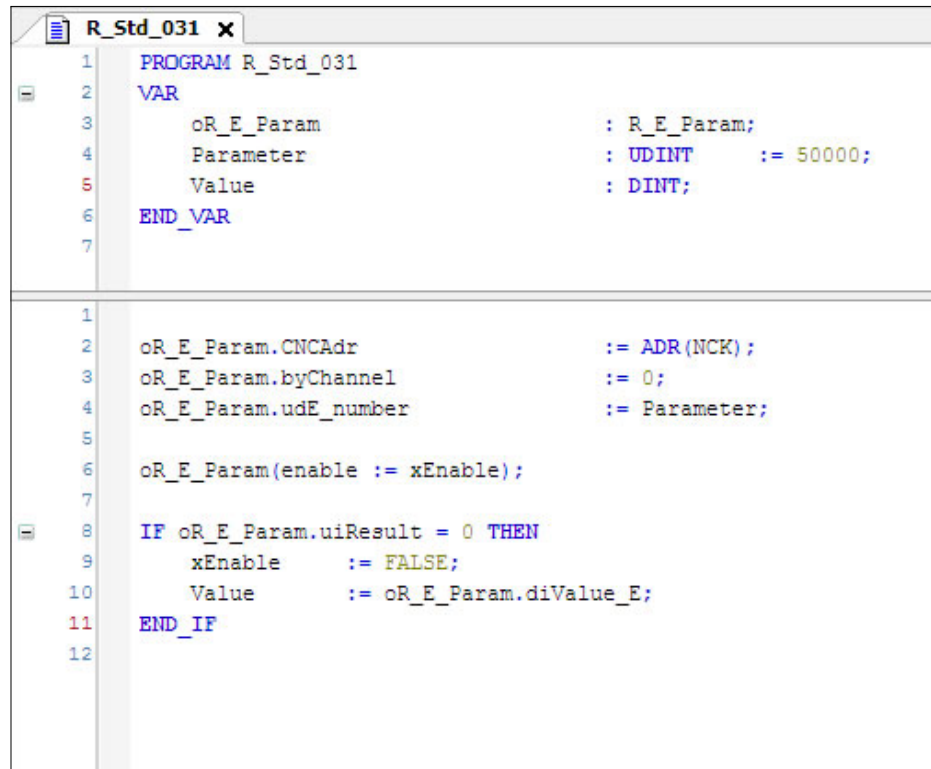
```

### 5.2.15 Reading a program parameter ► R\_E\_Param

This request returns the value of a particular E parameter.

Other requests on parameters are:

- *R\_E\_8xyyy()* returns a set of 10 parameters
- *R\_Tool()* returns all data about a tool offset
- *R\_Tool\_H()* return H values by set of 10.



```
1 PROGRAM R_Std_031
2 VAR
3     oR_E_Param          : R_E_Param;
4     Parameter           : UDINT      := 50000;
5     Value               : DINT;
6 END_VAR
7
8
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24
25
26
27
28
29
30
31
32
33
34
35
36
37
38
39
40
41
42
43
44
45
46
47
48
49
50
51
52
53
54
55
56
57
58
59
60
61
62
63
64
65
66
67
68
69
70
71
72
73
74
75
76
77
78
79
80
81
82
83
84
85
86
87
88
89
90
91
92
93
94
95
96
97
98
99
100
101
102
103
104
105
106
107
108
109
110
111
112
113
114
115
116
117
118
119
120
121
122
123
124
125
126
127
128
129
130
131
132
133
134
135
136
137
138
139
140
141
142
143
144
145
146
147
148
149
150
151
152
153
154
155
156
157
158
159
160
161
162
163
164
165
166
167
168
169
170
171
172
173
174
175
176
177
178
179
180
181
182
183
184
185
186
187
188
189
190
191
192
193
194
195
196
197
198
199
200
201
202
203
204
205
206
207
208
209
210
211
212
213
214
215
216
217
218
219
220
221
222
223
224
225
226
227
228
229
230
231
232
233
234
235
236
237
238
239
240
241
242
243
244
245
246
247
248
249
250
251
252
253
254
255
256
257
258
259
260
261
262
263
264
265
266
267
268
269
270
271
272
273
274
275
276
277
278
279
280
281
282
283
284
285
286
287
288
289
290
291
292
293
294
295
296
297
298
299
300
301
302
303
304
305
306
307
308
309
310
311
312
313
314
315
316
317
318
319
320
321
322
323
324
325
326
327
328
329
330
331
332
333
334
335
336
337
338
339
340
341
342
343
344
345
346
347
348
349
350
351
352
353
354
355
356
357
358
359
360
361
362
363
364
365
366
367
368
369
370
371
372
373
374
375
376
377
378
379
380
381
382
383
384
385
386
387
388
389
390
391
392
393
394
395
396
397
398
399
400
401
402
403
404
405
406
407
408
409
410
411
412
413
414
415
416
417
418
419
420
421
422
423
424
425
426
427
428
429
430
431
432
433
434
435
436
437
438
439
440
441
442
443
444
445
446
447
448
449
450
451
452
453
454
455
456
457
458
459
460
461
462
463
464
465
466
467
468
469
470
471
472
473
474
475
476
477
478
479
480
481
482
483
484
485
486
487
488
489
490
491
492
493
494
495
496
497
498
499
500
501
502
503
504
505
506
507
508
509
510
511
512
513
514
515
516
517
518
519
520
521
522
523
524
525
526
527
528
529
530
531
532
533
534
535
536
537
538
539
540
541
542
543
544
545
546
547
548
549
550
551
552
553
554
555
556
557
558
559
560
561
562
563
564
565
566
567
568
569
570
571
572
573
574
575
576
577
578
579
580
581
582
583
584
585
586
587
588
589
590
591
592
593
594
595
596
597
598
599
600
601
602
603
604
605
606
607
608
609
610
611
612
613
614
615
616
617
618
619
620
621
622
623
624
625
626
627
628
629
630
631
632
633
634
635
636
637
638
639
640
641
642
643
644
645
646
647
648
649
650
651
652
653
654
655
656
657
658
659
660
661
662
663
664
665
666
667
668
669
670
671
672
673
674
675
676
677
678
679
680
681
682
683
684
685
686
687
688
689
690
691
692
693
694
695
696
697
698
699
700
701
702
703
704
705
706
707
708
709
710
711
712
713
714
715
716
717
718
719
720
721
722
723
724
725
726
727
728
729
730
731
732
733
734
735
736
737
738
739
740
741
742
743
744
745
746
747
748
749
750
751
752
753
754
755
756
757
758
759
760
761
762
763
764
765
766
767
768
769
770
771
772
773
774
775
776
777
778
779
780
781
782
783
784
785
786
787
788
789
790
791
792
793
794
795
796
797
798
799
800
801
802
803
804
805
806
807
808
809
810
811
812
813
814
815
816
817
818
819
820
821
822
823
824
825
826
827
828
829
830
831
832
833
834
835
836
837
838
839
840
841
842
843
844
845
846
847
848
849
850
851
852
853
854
855
856
857
858
859
860
861
862
863
864
865
866
867
868
869
870
871
872
873
874
875
876
877
878
879
880
881
882
883
884
885
886
887
888
889
890
891
892
893
894
895
896
897
898
899
900
901
902
903
904
905
906
907
908
909
910
911
912
913
914
915
916
917
918
919
920
921
922
923
924
925
926
927
928
929
930
931
932
933
934
935
936
937
938
939
940
941
942
943
944
945
946
947
948
949
950
951
952
953
954
955
956
957
958
959
960
961
962
963
964
965
966
967
968
969
970
971
972
973
974
975
976
977
978
979
980
981
982
983
984
985
986
987
988
989
990
991
992
993
994
995
996
997
998
999
1000
1001
1002
1003
1004
1005
1006
1007
1008
1009
1010
1011
1012
1013
1014
1015
1016
1017
1018
1019
1020
1021
1022
1023
1024
1025
1026
1027
1028
1029
1030
1031
1032
1033
1034
1035
1036
1037
1038
1039
1040
1041
1042
1043
1044
1045
1046
1047
1048
1049
1050
1051
1052
1053
1054
1055
1056
1057
1058
1059
1060
1061
1062
1063
1064
1065
1066
1067
1068
1069
1070
1071
1072
1073
1074
1075
1076
1077
1078
1079
1080
1081
1082
1083
1084
1085
1086
1087
1088
1089
1090
1091
1092
1093
1094
1095
1096
1097
1098
1099
1100
1101
1102
1103
1104
1105
1106
1107
1108
1109
1110
1111
1112
1113
1114
1115
1116
1117
1118
1119
1120
1121
1122
1123
1124
1125
1126
1127
1128
1129
1130
1131
1132
1133
1134
1135
1136
1137
1138
1139
1140
1141
1142
1143
1144
1145
1146
1147
1148
1149
1150
1151
1152
1153
1154
1155
1156
1157
1158
1159
1160
1161
1162
1163
1164
1165
1166
1167
1168
1169
1170
1171
1172
1173
1174
1175
1176
1177
1178
1179
1180
1181
1182
1183
1184
1185
1186
1187
1188
1189
1190
1191
1192
1193
1194
1195
1196
1197
1198
1199
1200
1201
1202
1203
1204
1205
1206
1207
1208
1209
1210
1211
1212
1213
1214
1215
1216
1217
1218
1219
1220
1221
1222
1223
1224
1225
1226
1227
1228
1229
1230
1231
1232
1233
1234
1235
1236
1237
1238
1239
1240
1241
1242
1243
1244
1245
1246
1247
1248
1249
1250
1251
1252
1253
1254
1255
1256
1257
1258
1259
1260
1261
1262
1263
1264
1265
1266
1267
1268
1269
1270
1271
1272
1273
1274
1275
1276
1277
1278
1279
1280
1281
1282
1283
1284
1285
1286
1287
1288
1289
1290
1291
1292
1293
1294
1295
1296
1297
1298
1299
1300
1301
1302
1303
1304
1305
1306
1307
1308
1309
1310
1311
1312
1313
1314
1315
1316
1317
1318
1319
1320
1321
1322
1323
1324
1325
1326
1327
1328
1329
1330
1331
1332
1333
1334
1335
1336
1337
1338
1339
1340
1341
1342
1343
1344
1345
1346
1347
1348
1349
1350
1351
1352
1353
1354
1355
1356
1357
1358
1359
1360
1361
1362
1363
1364
1365
1366
1367
1368
1369
1370
1371
1372
1373
1374
1375
1376
1377
1378
1379
1380
1381
1382
1383
1384
1385
1386
1387
1388
1389
1390
1391
1392
1393
1394
1395
1396
1397
1398
1399
1400
1401
1402
1403
1404
1405
1406
1407
1408
1409
1410
1411
1412
1413
1414
1415
1416
1417
1418
1419
1420
1421
1422
1423
1424
1425
1426
1427
1428
1429
1430
1431
1432
1433
1434
1435
1436
1437
1438
1439
1440
1441
1442
1443
1444
1445
1446
1447
1448
1449
1450
1451
1452
1453
1454
1455
1456
1457
1458
1459
1460
1461
1462
1463
1464
1465
1466
1467
1468
1469
1470
1471
1472
1473
1474
1475
1476
1477
1478
1479
1480
1481
1482
1483
1484
1485
1486
1487
1488
1489
1490
1491
1492
1493
1494
1495
1496
1497
1498
1499
1500
1501
1502
1503
1504
1505
1506
1507
1508
1509
1510
1511
1512
1513
1514
1515
1516
1517
1518
1519
1520
1521
1522
1523
1524
1525
1526
1527
1528
1529
1530
1531
1532
1533
1534
1535
1536
1537
1538
1539
1540
1541
1542
1543
1544
1545
1546
1547
1548
1549
1550
1551
1552
1553
1554
1555
1556
1557
1558
1559
1560
1561
1562
1563
1564
1565
1566
1567
1568
1569
1570
1571
1572
1573
1574
1575
1576
1577
1578
1579
1580
1581
1582
1583
1584
1585
1586
1587
1588
1589
1590
1591
1592
1593
1594
1595
1596
1597
1598
1599
1600
1601
1602
1603
1604
1605
1606
1607
1608
1609
1610
1611
1612
1613
1614
1615
1616
1617
1618
1619
1620
1621
1622
1623
1624
1625
1626
1627
1628
1629
1630
1631
1632
1633
1634
1635
1636
1637
1638
1639
1640
1641
1642
1643
1644
1645
1646
1647
1648
1649
1650
1651
1652
1653
1654
1655
1656
1657
1658
1659
1660
1661
1662
1663
1664
1665
1666
1667
1668
1669
1670
1671
1672
1673
1674
1675
1676
1677
1678
1679
1680
1681
1682
1683
1684
1685
1686
1687
1688
1689
1690
1691
1692
1693
1694
1695
1696
1697
1698
1699
1700
1701
1702
1703
1704
1705
1706
1707
1708
1709
1710
1711
1712
1713
1714
1715
1716
1717
1718
1719
1720
1721
1722
1723
1724
1725
1726
1727
1728
1729
1730
1731
1732
1733
1734
1735
1736
1737
1738
1739
1740
1741
1742
1743
1744
1745
1746
1747
1748
1749
1750
1751
1752
1753
1754
1755
1756
1757
1758
1759
1760
1761
1762
1763
1764
1765
1766
1767
1768
1769
1770
1771
1772
1773
1774
1775
1776
1777
1778
1779
1780
1781
1782
1783
1784
1785
1786
1787
1788
1789
1790
1791
1792
1793
1794
1795
1796
1797
1798
1799
1800
1801
1802
1803
1804
1805
1806
1807
1808
1809
1810
1811
1812
1813
1814
1815
1816
1817
1818
1819
1820
1821
1822
1823
1824
1825
1826
1827
1828
1829
1830
1831
1832
1833
1834
1835
1836
1837
1838
1839
1840
1841
1842
1843
1844
1845
1846
1847
1848
1849
1850
1851
1852
1853
1854
1855
1856
1857
1858
1859
1860
1861
1862
1863
1864
1865
1866
1867
1868
1869
1870
1871
1872
1873
1874
1875
1876
1877
1878
1879
1880
1881
1882
1883
1884
1885
1886
1887
1888
1889
1890
1891
1892
1893
1894
1895
1896
1897
1898
1899
1900
1901
1902
1903
1904
1905
1906
1907
1908
1909
1910
1911
1912
1913
1914
1915
1916
1917
1918
1919
1920
1921
1922
1923
1924
1925
1926
1927
1928
1929
1930
1931
1932
1933
1934
1935
1936
1937
1938
1939
1940
1941
1942
1943
1944
1945
1946
1947
1948
1949
1950
1951
1952
1953
1954
1955
1956
1957
1958
1959
1960
1961
1962
1963
1964
1965
1966
1967
1968
1969
1970
1971
1972
1973
1974
1975
1976
1977
1978
1979
1980
1981
1982
1983
1984
1985
1986
1987
1988
1989
1990
1991
1992
1993
1994
1995
1996
1997
1998
1999
2000
2001
2002
2003
2004
2005
2006
2007
2008
2009
2010
2011
2012
2013
2014
2015
2016
2017
2018
2019
2020
2021
2022
2023
2024
2025
2026
2027
2028
2029
2030
2031
2032
2033
2034
2035
2036
2037
2038
2039
2040
2041
2042
2043
2044
2045
2046
2047
2048
2049
2050
2051
2052
2053
2054
2055
2056
2057
2058
2059
2060
2061
2062
2063
2064
2065
2066
2067
2068
2069
2070
2071
2072
2073
2074
2075
2076
2077
2078
2079
2080
2081
2082
2083
2084
2085
2086
2087
2088
2089
2090
2091
2092
2093
2094
2095
2096
2097
2098
2099
2100
2101
2102
2103
2104
2105
2106
2107
2108
2109
2110
2111
2112
2113
2114
2115
2116
2117
2118
2119
2120
2121
2122
2123
2124
2125
2126
2127
2128
2129
2130
2131
2132
2133
2134
2135
2136
2137
2138
2139
2140
2141
2142
2143
2144
2145
2146
2147
2148
2149
2150
2151
2152
2153
2154
2155
2156
2157
2158
2159
2160
2161
2162
2163
2164
2165
2166
2167
2168
2169
2170
2171
2172
2173
2174
2175
2176
2177
2178
2179
2180
2181
2182
2183
2184
2185
2186
2187
2188
2189
2190
2191
2192
2193
2194
2195
2196
2197
2198
2199
2200
2201
2202
2203
2204
2205
2206
2207
2208
2209
2210
2211
2212
2213
2214
2215
2216
2217
2218
2219
2220
2221
2222
2223
2224
2225
2226
2227
2228
2229
2230
2231
2232
2233
2234
2235
2236
2237
2238
2239
2240
2241
2242
2243
2244
2245
2246
2247
2248
2249
2250
2251
2252
2253
2254
2255
2256
2257
2258
2259
2260
2261
2262
2263
2264
2265
2266
2267
2268
2269
2270
2271
2272
2273
2274
2275
2276
2277
2278
2279
2280
2281
2282
2283
2284
2285
2286
2287
2288
2289
2290
2291
2292
2293
2294
2295
2296
2297
2298
2299
2300
2301
2302
2303
2304
2305
2306
2307
2308
2309
2310
2311
2312
2313
2314
2315
2316
2317
2318
2319
2320
2321
2322
2323
2324
2325
2326
2327
2328
2329
2330
2331
2332
2333
2334
2335
2336
2337
2338
2339
2340
2341
2342
2343
2344
2345
2346
2347
2348
2349
2350
2351
2352
2353
2354
2355
2356
2357
2358
2359
2360
2361
2362
2363
2364
2365
2366
2367
2368
2369
2370
2371
2372
2373
2374
2375
2376
2377
2378
2379
2380
2381
2382
2383
2384
2385
2386
2387
2388
2389
2390
2391
2392
2393
2394
2395
2396
2397
2398
2399
2400
2401
2402
2403
2404
2405
2406
2407
2408
2409
2410
2411
2412
2413
2414
2415
2416
2417
2418
2419
2420
2421
2422
2423
2424
2425
2426
2427
2428
2429
2430
2431
2432
2433
2434
2435
2436
2437
2438
2439
2440
2441
2442
2443
2444
2445
2446
2447
2448
2449
2450
2451
2452
2453
2454
2455
2456
2457
2458
2459
2460
2461
2462
2463
2464
2465
2466
2467
2468
2469
2470
2471
2472
2473
2474
2475
2476
2477
2478
2479
2480
2481
2482
2483
2484
2485
2486
2487
2488
2489
2490
2491
2492
2493
2494
2495
2496
2497
2498
2499
2500
2501
2502
2503
2504
2505
2506
2507
2508
2509
2510
2511
2512
2513
2514
2515
2516
2517
2518
2519
2520
2521
2522
2523
2524
2525
2526
2527
2528
2529
2530
2531
2532
2533
2534
2535
2536
2537
2538
2539
2540
2541
2542
2543
2544
2545
2546
2547
2548
2549
2550
2551
2552
2553
2554
2555
2556
2557
2558
2559
2560
2561
2562
2563
2564
2565
2566
2567
2568
2569
2570
2571
2572
2573
2574
2575
2576
2577
2578
2579
2580
2581
2582
2583
2584
2585
2586
2587
2588
2589
2590
2591
2592
2593
2594
2595
2596
2597
2598
2599
2600
2601
2602
2603
260
```

## 5.2.16 Reading the Interpolation data ► R\_Interpo

This request returns current interpolation data of a particular channel.  
The data are curvilinear distance to go, feedrate as well as the override value.

```

1  PROGRAM R_Std_040
2  VAR
3      oR_Interpo          : R_Interpo;
4      di_Distance         : DINT;
5      di_Feed             : DINT;
6      di_Override         : DINT;
7  END_VAR

1  oR_Interpo.CNCAdr       := ADR(NCK);
2  oR_Interpo.byChannel    := 0;
3
4  oR_Interpo(enable := xEnable);
5
6
7  IF oR_Interpo.uiResult = 0 THEN
8      xEnable             := FALSE;
9      di_Distance         := oR_Interpo.diDistance;
10     di_Feed             := oR_Interpo.diFeedProg;
11     di_Override         := oR_Interpo.diOverride;
12 END_IF
13

```

### 5.2.17 Reading machine origin values ► [R\\_Mach\\_Orig](#)

This request returns the machine origin offset values (equivalent to parameter P16) according to the axis physical address. The values are returned for three consecutive addresses.

```
1  PROGRAM R_Std_041
2  VAR
3      oR_Mach_Origin          : R_Mach_Orig;
4      di_Axe1                 : DINT;
5      di_Axe2                 : DINT;
6      di_axe3                 : DINT;
7  END_VAR
8
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24
25
26
27
28
29
30
31
32
33
34
35
36
37
38
39
40
41
42
43
44
45
46
47
48
49
50
51
52
53
54
55
56
57
58
59
60
61
62
63
64
65
66
67
68
69
70
71
72
73
74
75
76
77
78
79
80
81
82
83
84
85
86
87
88
89
90
91
92
93
94
95
96
97
98
99
100
101
102
103
104
105
106
107
108
109
110
111
112
113
114
115
116
117
118
119
120
121
122
123
124
125
126
127
128
129
130
131
132
133
134
135
136
137
138
139
140
141
142
143
144
145
146
147
148
149
150
151
152
153
154
155
156
157
158
159
160
161
162
163
164
165
166
167
168
169
170
171
172
173
174
175
176
177
178
179
180
181
182
183
184
185
186
187
188
189
190
191
192
193
194
195
196
197
198
199
200
201
202
203
204
205
206
207
208
209
210
211
212
213
214
215
216
217
218
219
220
221
222
223
224
225
226
227
228
229
230
231
232
233
234
235
236
237
238
239
240
241
242
243
244
245
246
247
248
249
250
251
252
253
254
255
256
257
258
259
260
261
262
263
264
265
266
267
268
269
270
271
272
273
274
275
276
277
278
279
280
281
282
283
284
285
286
287
288
289
290
291
292
293
294
295
296
297
298
299
300
301
302
303
304
305
306
307
308
309
310
311
312
313
314
315
316
317
318
319
320
321
322
323
324
325
326
327
328
329
330
331
332
333
334
335
336
337
338
339
340
341
342
343
344
345
346
347
348
349
350
351
352
353
354
355
356
357
358
359
360
361
362
363
364
365
366
367
368
369
370
371
372
373
374
375
376
377
378
379
380
381
382
383
384
385
386
387
388
389
390
391
392
393
394
395
396
397
398
399
400
401
402
403
404
405
406
407
408
409
410
411
412
413
414
415
416
417
418
419
420
421
422
423
424
425
426
427
428
429
430
431
432
433
434
435
436
437
438
439
440
441
442
443
444
445
446
447
448
449
450
451
452
453
454
455
456
457
458
459
460
461
462
463
464
465
466
467
468
469
470
471
472
473
474
475
476
477
478
479
480
481
482
483
484
485
486
487
488
489
490
491
492
493
494
495
496
497
498
499
500
501
502
503
504
505
506
507
508
509
510
511
512
513
514
515
516
517
518
519
520
521
522
523
524
525
526
527
528
529
530
531
532
533
534
535
536
537
538
539
540
541
542
543
544
545
546
547
548
549
550
551
552
553
554
555
556
557
558
559
560
561
562
563
564
565
566
567
568
569
570
571
572
573
574
575
576
577
578
579
580
581
582
583
584
585
586
587
588
589
590
591
592
593
594
595
596
597
598
599
600
601
602
603
604
605
606
607
608
609
610
611
612
613
614
615
616
617
618
619
620
621
622
623
624
625
626
627
628
629
630
631
632
633
634
635
636
637
638
639
640
641
642
643
644
645
646
647
648
649
650
651
652
653
654
655
656
657
658
659
660
661
662
663
664
665
666
667
668
669
670
671
672
673
674
675
676
677
678
679
680
681
682
683
684
685
686
687
688
689
690
691
692
693
694
695
696
697
698
699
700
701
702
703
704
705
706
707
708
709
710
711
712
713
714
715
716
717
718
719
720
721
722
723
724
725
726
727
728
729
730
731
732
733
734
735
736
737
738
739
740
741
742
743
744
745
746
747
748
749
750
751
752
753
754
755
756
757
758
759
760
761
762
763
764
765
766
767
768
769
770
771
772
773
774
775
776
777
778
779
780
781
782
783
784
785
786
787
788
789
790
791
792
793
794
795
796
797
798
799
800
801
802
803
804
805
806
807
808
809
810
811
812
813
814
815
816
817
818
819
820
821
822
823
824
825
826
827
828
829
830
831
832
833
834
835
836
837
838
839
840
841
842
843
844
845
846
847
848
849
850
851
852
853
854
855
856
857
858
859
860
861
862
863
864
865
866
867
868
869
870
871
872
873
874
875
876
877
878
879
880
881
882
883
884
885
886
887
888
889
890
891
892
893
894
895
896
897
898
899
900
901
902
903
904
905
906
907
908
909
910
911
912
913
914
915
916
917
918
919
920
921
922
923
924
925
926
927
928
929
930
931
932
933
934
935
936
937
938
939
940
941
942
943
944
945
946
947
948
949
950
951
952
953
954
955
956
957
958
959
960
961
962
963
964
965
966
967
968
969
970
971
972
973
974
975
976
977
978
979
980
981
982
983
984
985
986
987
988
989
990
991
992
993
994
995
996
997
998
999
1000
```

## 5.2.18 Reading Maximum Dynamic Travel values ▶ R\_Max\_Dyn\_Travel

This request returns the maximum dynamic travels of the axes in a particular channel. The values are returned in reference to the axes names.

```

1  PROGRAM R_Std_042
2  VAR
3      oR_Max_Dyn_Travel          : R_Max_Dyn_Travel;
4      di_MaxDyn_A                : DINT;
5      di_MaxDyn_B                : DINT;
6      di_MaxDyn_C                : DINT;
7      di_MaxDyn_U                : DINT;
8      di_MaxDyn_V                : DINT;
9      di_MaxDyn_W                : DINT;
10     di_MaxDyn_X                : DINT;
11     di_MaxDyn_Y                : DINT;
12     di_MaxDyn_Z                : DINT;
13 END_VAR
14
15
16 oR_Max_Dyn_Travel.CNCAdr        := ADR(NCK);
17 oR_Max_Dyn_Travel.byChannel    := 0;
18
19 oR_Max_Dyn_Travel(enable := xEnable);
20
21 IF oR_Max_Dyn_Travel.uiResult = 0 THEN
22     xEnable := FALSE;
23     di_MaxDyn_A := oR_Max_Dyn_Travel.diA_Axis;
24     di_MaxDyn_B := oR_Max_Dyn_Travel.diB_Axis;
25     di_MaxDyn_C := oR_Max_Dyn_Travel.diC_Axis;
26     di_MaxDyn_U := oR_Max_Dyn_Travel.diU_Axis;
27     di_MaxDyn_V := oR_Max_Dyn_Travel.diV_Axis;
28     di_MaxDyn_W := oR_Max_Dyn_Travel.diW_Axis;
29     di_MaxDyn_X := oR_Max_Dyn_Travel.diX_Axis;
30     di_MaxDyn_Y := oR_Max_Dyn_Travel.diY_Axis;
31     di_MaxDyn_Z := oR_Max_Dyn_Travel.diZ_Axis;
32 END_IF
33

```

## 5.2.19 Reading Minimum Dynamic Travel values [▶ R\\_Min\\_Dyn\\_Travel](#)

This request returns the minimum dynamic travels of the axes in a particular channel. The values are returned in reference to the axes names.

```
R_Std_043 x
1 PROGRAM R_Std_043
2 VAR
3     oR_Min_Dyn_Travel          : R_Min_Dyn_Travel;
4     di_MinDyn_A                : DINT;
5     di_MinDyn_B                : DINT;
6     di_MinDyn_C                : DINT;
7     di_MinDyn_U                : DINT;
8     di_MinDyn_V                : DINT;
9     di_MinDyn_W                : DINT;
10    di_MinDyn_X                : DINT;
11    di_MinDyn_Y                : DINT;
12    di_MinDyn_Z                : DINT;
13 END_VAR
14
15
16
17
18
19
20
21
22
23
24
25
26
27
28
29
30
31
32
33
34
35
36
37
38
39
40
41
42
43
44
45
46
47
48
49
50
51
52
53
54
55
56
57
58
59
60
61
62
63
64
65
66
67
68
69
70
71
72
73
74
75
76
77
78
79
80
81
82
83
84
85
86
87
88
89
90
91
92
93
94
95
96
97
98
99
100
101
102
103
104
105
106
107
108
109
110
111
112
113
114
115
116
117
118
119
120
121
122
123
124
125
126
127
128
129
130
131
132
133
134
135
136
137
138
139
140
141
142
143
144
145
146
147
148
149
150
151
152
153
154
155
156
157
158
159
160
161
162
163
164
165
166
167
168
169
170
171
172
173
174
175
176
177
178
179
180
181
182
183
184
185
186
187
188
189
190
191
192
193
194
195
196
197
198
199
200
201
202
203
204
205
206
207
208
209
210
211
212
213
214
215
216
217
218
219
220
221
222
223
224
225
226
227
228
229
230
231
232
233
234
235
236
237
238
239
240
241
242
243
244
245
246
247
248
249
250
251
252
253
254
255
256
257
258
259
260
261
262
263
264
265
266
267
268
269
270
271
272
273
274
275
276
277
278
279
280
281
282
283
284
285
286
287
288
289
290
291
292
293
294
295
296
297
298
299
300
301
302
303
304
305
306
307
308
309
310
311
312
313
314
315
316
317
318
319
320
321
322
323
324
325
326
327
328
329
330
331
332
333
334
335
336
337
338
339
340
341
342
343
344
345
346
347
348
349
350
351
352
353
354
355
356
357
358
359
360
361
362
363
364
365
366
367
368
369
370
371
372
373
374
375
376
377
378
379
380
381
382
383
384
385
386
387
388
389
390
391
392
393
394
395
396
397
398
399
400
401
402
403
404
405
406
407
408
409
410
411
412
413
414
415
416
417
418
419
420
421
422
423
424
425
426
427
428
429
430
431
432
433
434
435
436
437
438
439
440
441
442
443
444
445
446
447
448
449
450
451
452
453
454
455
456
457
458
459
460
461
462
463
464
465
466
467
468
469
470
471
472
473
474
475
476
477
478
479
480
481
482
483
484
485
486
487
488
489
490
491
492
493
494
495
496
497
498
499
500
501
502
503
504
505
506
507
508
509
510
511
512
513
514
515
516
517
518
519
520
521
522
523
524
525
526
527
528
529
530
531
532
533
534
535
536
537
538
539
540
541
542
543
544
545
546
547
548
549
550
551
552
553
554
555
556
557
558
559
560
561
562
563
564
565
566
567
568
569
570
571
572
573
574
575
576
577
578
579
580
581
582
583
584
585
586
587
588
589
590
591
592
593
594
595
596
597
598
599
600
601
602
603
604
605
606
607
608
609
610
611
612
613
614
615
616
617
618
619
620
621
622
623
624
625
626
627
628
629
630
631
632
633
634
635
636
637
638
639
640
641
642
643
644
645
646
647
648
649
650
651
652
653
654
655
656
657
658
659
660
661
662
663
664
665
666
667
668
669
670
671
672
673
674
675
676
677
678
679
680
681
682
683
684
685
686
687
688
689
690
691
692
693
694
695
696
697
698
699
700
701
702
703
704
705
706
707
708
709
710
711
712
713
714
715
716
717
718
719
720
721
722
723
724
725
726
727
728
729
730
731
732
733
734
735
736
737
738
739
740
741
742
743
744
745
746
747
748
749
750
751
752
753
754
755
756
757
758
759
760
761
762
763
764
765
766
767
768
769
770
771
772
773
774
775
776
777
778
779
780
781
782
783
784
785
786
787
788
789
790
791
792
793
794
795
796
797
798
799
800
801
802
803
804
805
806
807
808
809
810
811
812
813
814
815
816
817
818
819
820
821
822
823
824
825
826
827
828
829
830
831
832
833
834
835
836
837
838
839
840
841
842
843
844
845
846
847
848
849
850
851
852
853
854
855
856
857
858
859
860
861
862
863
864
865
866
867
868
869
870
871
872
873
874
875
876
877
878
879
880
881
882
883
884
885
886
887
888
889
890
891
892
893
894
895
896
897
898
899
900
901
902
903
904
905
906
907
908
909
910
911
912
913
914
915
916
917
918
919
920
921
922
923
924
925
926
927
928
929
930
931
932
933
934
935
936
937
938
939
940
941
942
943
944
945
946
947
948
949
950
951
952
953
954
955
956
957
958
959
960
961
962
963
964
965
966
967
968
969
970
971
972
973
974
975
976
977
978
979
980
981
982
983
984
985
986
987
988
989
990
991
992
993
994
995
996
997
998
999
1000
```

### 5.2.20 Reading Maximum Static Travel values ► [R\\_Max\\_Stat\\_Travel](#)

This request returns the maximum static travels of axes.  
The values are given for three consecutive physical addresses.

```

1  PROGRAM R_Std_044
2  VAR
3      oR_Max_Stat_Travel      : R_Max_Stat_Travel;
4      di_MaxStat_0           : DINT;
5      di_MaxStat_1           : DINT;
6      di_MaxStat_2           : DINT;
7  END_VAR
8
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24
25
26
27
28
29
30
31
32
33
34
35
36
37
38
39
40
41
42
43
44
45
46
47
48
49
50
51
52
53
54
55
56
57
58
59
60
61
62
63
64
65
66
67
68
69
70
71
72
73
74
75
76
77
78
79
80
81
82
83
84
85
86
87
88
89
90
91
92
93
94
95
96
97
98
99
100
101
102
103
104
105
106
107
108
109
110
111
112
113
114
115
116
117
118
119
120
121
122
123
124
125
126
127
128
129
130
131
132
133
134
135
136
137
138
139
140
141
142
143
144
145
146
147
148
149
150
151
152
153
154
155
156
157
158
159
160
161
162
163
164
165
166
167
168
169
170
171
172
173
174
175
176
177
178
179
180
181
182
183
184
185
186
187
188
189
190
191
192
193
194
195
196
197
198
199
200
201
202
203
204
205
206
207
208
209
210
211
212
213
214
215
216
217
218
219
220
221
222
223
224
225
226
227
228
229
230
231
232
233
234
235
236
237
238
239
240
241
242
243
244
245
246
247
248
249
250
251
252
253
254
255
256
257
258
259
260
261
262
263
264
265
266
267
268
269
270
271
272
273
274
275
276
277
278
279
280
281
282
283
284
285
286
287
288
289
290
291
292
293
294
295
296
297
298
299
300
301
302
303
304
305
306
307
308
309
310
311
312
313
314
315
316
317
318
319
320
321
322
323
324
325
326
327
328
329
330
331
332
333
334
335
336
337
338
339
340
341
342
343
344
345
346
347
348
349
350
351
352
353
354
355
356
357
358
359
360
361
362
363
364
365
366
367
368
369
370
371
372
373
374
375
376
377
378
379
380
381
382
383
384
385
386
387
388
389
390
391
392
393
394
395
396
397
398
399
400
401
402
403
404
405
406
407
408
409
410
411
412
413
414
415
416
417
418
419
420
421
422
423
424
425
426
427
428
429
430
431
432
433
434
435
436
437
438
439
440
441
442
443
444
445
446
447
448
449
450
451
452
453
454
455
456
457
458
459
460
461
462
463
464
465
466
467
468
469
470
471
472
473
474
475
476
477
478
479
480
481
482
483
484
485
486
487
488
489
490
491
492
493
494
495
496
497
498
499
500
501
502
503
504
505
506
507
508
509
510
511
512
513
514
515
516
517
518
519
520
521
522
523
524
525
526
527
528
529
530
531
532
533
534
535
536
537
538
539
540
541
542
543
544
545
546
547
548
549
550
551
552
553
554
555
556
557
558
559
560
561
562
563
564
565
566
567
568
569
570
571
572
573
574
575
576
577
578
579
580
581
582
583
584
585
586
587
588
589
590
591
592
593
594
595
596
597
598
599
600
601
602
603
604
605
606
607
608
609
610
611
612
613
614
615
616
617
618
619
620
621
622
623
624
625
626
627
628
629
630
631
632
633
634
635
636
637
638
639
640
641
642
643
644
645
646
647
648
649
650
651
652
653
654
655
656
657
658
659
660
661
662
663
664
665
666
667
668
669
670
671
672
673
674
675
676
677
678
679
680
681
682
683
684
685
686
687
688
689
690
691
692
693
694
695
696
697
698
699
700
701
702
703
704
705
706
707
708
709
710
711
712
713
714
715
716
717
718
719
720
721
722
723
724
725
726
727
728
729
730
731
732
733
734
735
736
737
738
739
740
741
742
743
744
745
746
747
748
749
750
751
752
753
754
755
756
757
758
759
760
761
762
763
764
765
766
767
768
769
770
771
772
773
774
775
776
777
778
779
780
781
782
783
784
785
786
787
788
789
790
791
792
793
794
795
796
797
798
799
800
801
802
803
804
805
806
807
808
809
810
811
812
813
814
815
816
817
818
819
820
821
822
823
824
825
826
827
828
829
830
831
832
833
834
835
836
837
838
839
840
841
842
843
844
845
846
847
848
849
850
851
852
853
854
855
856
857
858
859
860
861
862
863
864
865
866
867
868
869
870
871
872
873
874
875
876
877
878
879
880
881
882
883
884
885
886
887
888
889
890
891
892
893
894
895
896
897
898
899
900
901
902
903
904
905
906
907
908
909
910
911
912
913
914
915
916
917
918
919
920
921
922
923
924
925
926
927
928
929
930
931
932
933
934
935
936
937
938
939
940
941
942
943
944
945
946
947
948
949
950
951
952
953
954
955
956
957
958
959
960
961
962
963
964
965
966
967
968
969
970
971
972
973
974
975
976
977
978
979
980
981
982
983
984
985
986
987
988
989
990
991
992
993
994
995
996
997
998
999
1000

```

## 5.2.21 Reading Minimum Static Travel values [► R\\_Min\\_Stat\\_Travel](#)

This request returns the maximum static travels of axes.  
The values are given for three consecutive physical addresses.

```
R_Std_045 x
1 PROGRAM R_Std_045
2 VAR
3     oR_Min_Stat_Travel      : R_Min_Stat_Travel;
4     di_MinStat_0            : DINT;
5     di_MinStat_1            : DINT;
6     di_MinStat_2            : DINT;
7 END_VAR
8
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24
25
26
27
28
29
30
31
32
33
34
35
36
37
38
39
40
41
42
43
44
45
46
47
48
49
50
51
52
53
54
55
56
57
58
59
60
61
62
63
64
65
66
67
68
69
70
71
72
73
74
75
76
77
78
79
80
81
82
83
84
85
86
87
88
89
90
91
92
93
94
95
96
97
98
99
100
101
102
103
104
105
106
107
108
109
110
111
112
113
114
115
116
117
118
119
120
121
122
123
124
125
126
127
128
129
130
131
132
133
134
135
136
137
138
139
140
141
142
143
144
145
146
147
148
149
150
151
152
153
154
155
156
157
158
159
160
161
162
163
164
165
166
167
168
169
170
171
172
173
174
175
176
177
178
179
180
181
182
183
184
185
186
187
188
189
190
191
192
193
194
195
196
197
198
199
200
201
202
203
204
205
206
207
208
209
210
211
212
213
214
215
216
217
218
219
220
221
222
223
224
225
226
227
228
229
230
231
232
233
234
235
236
237
238
239
240
241
242
243
244
245
246
247
248
249
250
251
252
253
254
255
256
257
258
259
260
261
262
263
264
265
266
267
268
269
270
271
272
273
274
275
276
277
278
279
280
281
282
283
284
285
286
287
288
289
290
291
292
293
294
295
296
297
298
299
300
301
302
303
304
305
306
307
308
309
310
311
312
313
314
315
316
317
318
319
320
321
322
323
324
325
326
327
328
329
330
331
332
333
334
335
336
337
338
339
340
341
342
343
344
345
346
347
348
349
350
351
352
353
354
355
356
357
358
359
360
361
362
363
364
365
366
367
368
369
370
371
372
373
374
375
376
377
378
379
380
381
382
383
384
385
386
387
388
389
390
391
392
393
394
395
396
397
398
399
400
401
402
403
404
405
406
407
408
409
410
411
412
413
414
415
416
417
418
419
420
421
422
423
424
425
426
427
428
429
430
431
432
433
434
435
436
437
438
439
440
441
442
443
444
445
446
447
448
449
450
451
452
453
454
455
456
457
458
459
460
461
462
463
464
465
466
467
468
469
470
471
472
473
474
475
476
477
478
479
480
481
482
483
484
485
486
487
488
489
490
491
492
493
494
495
496
497
498
499
500
501
502
503
504
505
506
507
508
509
510
511
512
513
514
515
516
517
518
519
520
521
522
523
524
525
526
527
528
529
530
531
532
533
534
535
536
537
538
539
540
541
542
543
544
545
546
547
548
549
550
551
552
553
554
555
556
557
558
559
560
561
562
563
564
565
566
567
568
569
570
571
572
573
574
575
576
577
578
579
580
581
582
583
584
585
586
587
588
589
590
591
592
593
594
595
596
597
598
599
600
601
602
603
604
605
606
607
608
609
610
611
612
613
614
615
616
617
618
619
620
621
622
623
624
625
626
627
628
629
630
631
632
633
634
635
636
637
638
639
640
641
642
643
644
645
646
647
648
649
650
651
652
653
654
655
656
657
658
659
660
661
662
663
664
665
666
667
668
669
670
671
672
673
674
675
676
677
678
679
680
681
682
683
684
685
686
687
688
689
690
691
692
693
694
695
696
697
698
699
700
701
702
703
704
705
706
707
708
709
710
711
712
713
714
715
716
717
718
719
720
721
722
723
724
725
726
727
728
729
730
731
732
733
734
735
736
737
738
739
740
741
742
743
744
745
746
747
748
749
750
751
752
753
754
755
756
757
758
759
760
761
762
763
764
765
766
767
768
769
770
771
772
773
774
775
776
777
778
779
780
781
782
783
784
785
786
787
788
789
790
791
792
793
794
795
796
797
798
799
800
801
802
803
804
805
806
807
808
809
810
811
812
813
814
815
816
817
818
819
820
821
822
823
824
825
826
827
828
829
830
831
832
833
834
835
836
837
838
839
840
841
842
843
844
845
846
847
848
849
850
851
852
853
854
855
856
857
858
859
860
861
862
863
864
865
866
867
868
869
870
871
872
873
874
875
876
877
878
879
880
881
882
883
884
885
886
887
888
889
890
891
892
893
894
895
896
897
898
899
900
901
902
903
904
905
906
907
908
909
910
911
912
913
914
915
916
917
918
919
920
921
922
923
924
925
926
927
928
929
930
931
932
933
934
935
936
937
938
939
940
941
942
943
944
945
946
947
948
949
950
951
952
953
954
955
956
957
958
959
960
961
962
963
964
965
966
967
968
969
970
971
972
973
974
975
976
977
978
979
980
981
982
983
984
985
986
987
988
989
990
991
992
993
994
995
996
997
998
999
1000
```

### 5.2.22 Reading NCK detected error ► [R\\_Nck\\_Error](#)

This request returns the information about an NCK detected error. The data are the block number, line number where the error has occurred as well as the error number.

```

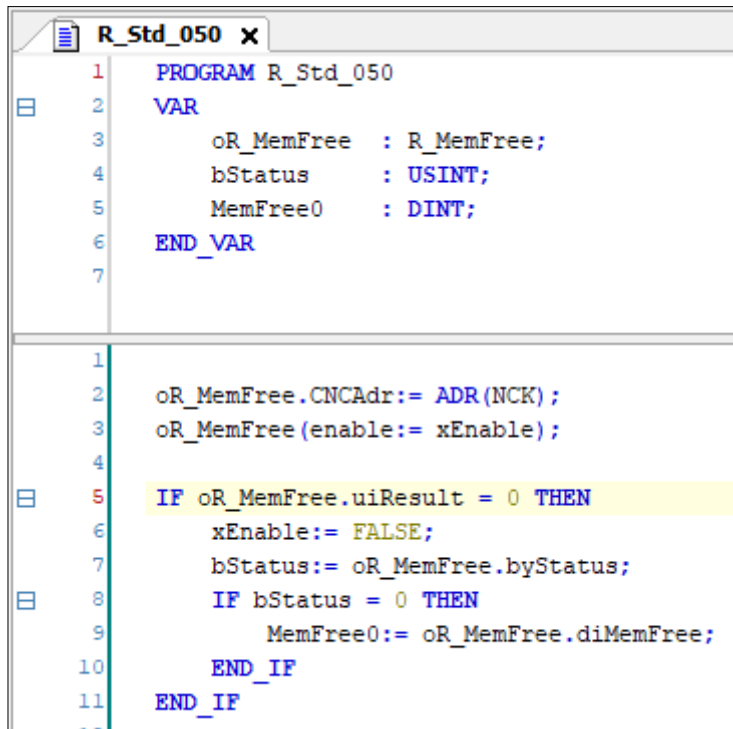
1  PROGRAM R_Std_046
2  VAR
3      oR_NckError          : R_Nck_Error;
4      wo_Block             : WORD;
5      wo_Line              : WORD;
6      wo_Error             : WORD;
7  END_VAR
8
9
10 oR_NckError.CNCAdr        := ADR(NCK);
11 oR_NckError.byChannel     := 0;
12
13 oR_NckError(enable := xEnable);
14
15 IF oR_NckError.uiResult = 0 THEN
16     xEnable                := FALSE;
17     wo_Block               := oR_NckError.woBlockNumber;
18     wo_Line                := oR_NckError.woLineNumber;
19     wo_Error               := oR_NckError.woErrorNumber;
20 END_IF

```

---

### 5.2.23 Reading amount of NCK User memory free ► [R\\_MemFree](#)

This request returns the information about the byte number of free User memory into the NCK. More information could be obtained using `R_Program_Memory_Size` function.

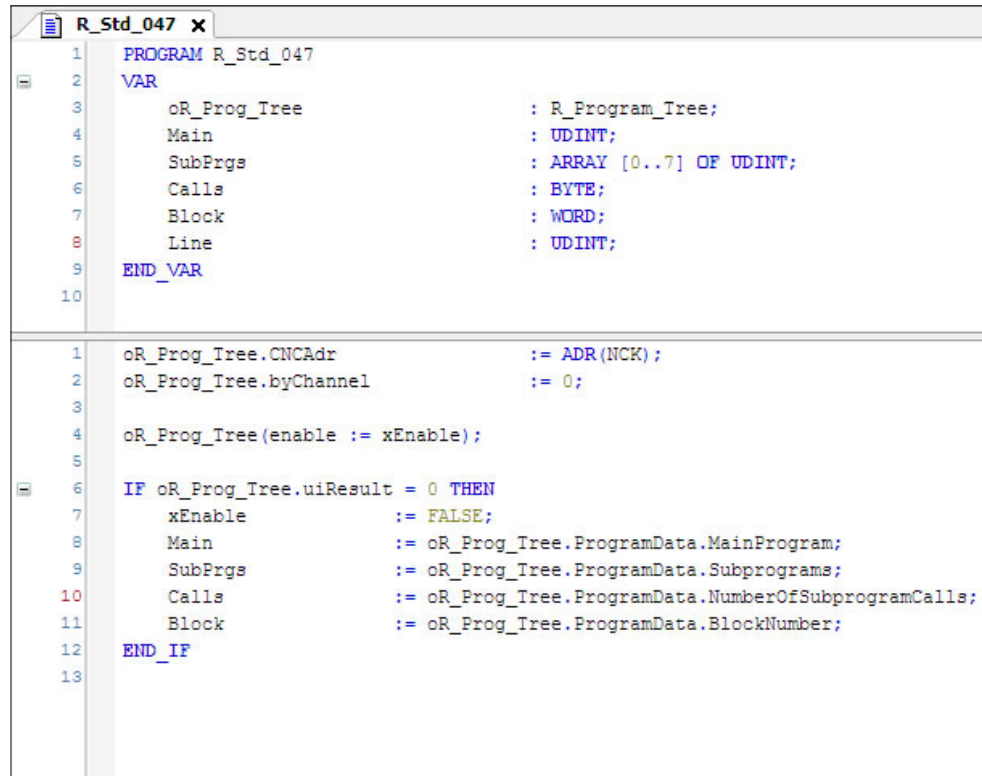


```
1  PROGRAM R_Std_050
2  VAR
3      oR_MemFree : R_MemFree;
4      bStatus    : USINT;
5      MemFree0   : DINT;
6  END_VAR
7
8
9
10
11  oR_MemFree.CNCAdr:= ADR(NCK);
12  oR_MemFree(enable:= xEnable);
13
14  IF oR_MemFree.uiResult = 0 THEN
15      xEnable:= FALSE;
16      bStatus:= oR_MemFree.byStatus;
17      IF bStatus = 0 THEN
18          MemFree0:= oR_MemFree.diMemFree;
19      END_IF
20  END_IF
```

### 5.2.24 Reading Part program execution tree ▶ [R\\_Program\\_Tree](#)

This request returns information about the part program in execution.

The information include block number, the current subroutine number with number of times it has been called as well as all the chain of calling programs (up to eight nesting level).



```

1  PROGRAM R_Std_047
2  VAR
3      oR_Prog_Tree          : R_Program_Tree;
4      Main                  : UDINT;
5      SubPrgs               : ARRAY [0..7] OF UDINT;
6      Calls                 : BYTE;
7      Block                 : WORD;
8      Line                  : UDINT;
9  END_VAR
10
11 oR_Prog_Tree.CNCAdr        := ADR(NCK);
12 oR_Prog_Tree.byChannel     := 0;
13
14 oR_Prog_Tree(enable := xEnable);
15
16 IF oR_Prog_Tree.uiResult = 0 THEN
17     xEnable                := FALSE;
18     Main                   := oR_Prog_Tree.ProgramData.MainProgram;
19     SubPrgs                := oR_Prog_Tree.ProgramData.Subprograms;
20     Calls                  := oR_Prog_Tree.ProgramData.NumberOfSubprogramCalls;
21     Block                  := oR_Prog_Tree.ProgramData.BlockNumber;
22 END_IF
  
```

## 5.2.25 Reading Spindle information ► R\_Spindle\_Complete

This request returns all relevant spindle information excluding the range currently active. The full list of data can be seen in the example below.

```
R_Std_048 x
1 PROGRAM R_Std_048
2 VAR
3     oR_Spindle           : R_Spindle_Complete;
4     bChannel             : BYTE;
5     bFormat              : BYTE;
6     xM19                 : BOOL;
7     xM3                  : BOOL;
8     xM4                  : BOOL;
9     xM5                  : BOOL;
10    uiOverride            : UINT;
11    lrProgSpeed           : LREAL;
12    lrRealSpeed           : LREAL;
13 END_VAR
14
15
16
17
18
19
20
21
22
23
24
25
26
27
28
29
30
31
32
33
34
35
36
37
38
39
40
41
42
43
44
45
46
47
48
49
50
51
52
53
54
55
56
57
58
59
60
61
62
63
64
65
66
67
68
69
70
71
72
73
74
75
76
77
78
79
80
81
82
83
84
85
86
87
88
89
90
91
92
93
94
95
96
97
98
99
100
101
102
103
104
105
106
107
108
109
110
111
112
113
114
115
116
117
118
119
120
121
122
123
124
125
126
127
128
129
130
131
132
133
134
135
136
137
138
139
140
141
142
143
144
145
146
147
148
149
150
151
152
153
154
155
156
157
158
159
160
161
162
163
164
165
166
167
168
169
170
171
172
173
174
175
176
177
178
179
180
181
182
183
184
185
186
187
188
189
190
191
192
193
194
195
196
197
198
199
200
201
202
203
204
205
206
207
208
209
210
211
212
213
214
215
216
217
218
219
220
221
222
223
224
225
226
227
228
229
230
231
232
233
234
235
236
237
238
239
240
241
242
243
244
245
246
247
248
249
250
251
252
253
254
255
256
257
258
259
260
261
262
263
264
265
266
267
268
269
270
271
272
273
274
275
276
277
278
279
280
281
282
283
284
285
286
287
288
289
290
291
292
293
294
295
296
297
298
299
300
301
302
303
304
305
306
307
308
309
310
311
312
313
314
315
316
317
318
319
320
321
322
323
324
325
326
327
328
329
330
331
332
333
334
335
336
337
338
339
340
341
342
343
344
345
346
347
348
349
350
351
352
353
354
355
356
357
358
359
360
361
362
363
364
365
366
367
368
369
370
371
372
373
374
375
376
377
378
379
380
381
382
383
384
385
386
387
388
389
390
391
392
393
394
395
396
397
398
399
400
401
402
403
404
405
406
407
408
409
410
411
412
413
414
415
416
417
418
419
420
421
422
423
424
425
426
427
428
429
430
431
432
433
434
435
436
437
438
439
440
441
442
443
444
445
446
447
448
449
450
451
452
453
454
455
456
457
458
459
460
461
462
463
464
465
466
467
468
469
470
471
472
473
474
475
476
477
478
479
480
481
482
483
484
485
486
487
488
489
490
491
492
493
494
495
496
497
498
499
500
501
502
503
504
505
506
507
508
509
510
511
512
513
514
515
516
517
518
519
520
521
522
523
524
525
526
527
528
529
530
531
532
533
534
535
536
537
538
539
540
541
542
543
544
545
546
547
548
549
550
551
552
553
554
555
556
557
558
559
560
561
562
563
564
565
566
567
568
569
570
571
572
573
574
575
576
577
578
579
580
581
582
583
584
585
586
587
588
589
590
591
592
593
594
595
596
597
598
599
600
601
602
603
604
605
606
607
608
609
610
611
612
613
614
615
616
617
618
619
620
621
622
623
624
625
626
627
628
629
630
631
632
633
634
635
636
637
638
639
640
641
642
643
644
645
646
647
648
649
650
651
652
653
654
655
656
657
658
659
660
661
662
663
664
665
666
667
668
669
670
671
672
673
674
675
676
677
678
679
680
681
682
683
684
685
686
687
688
689
690
691
692
693
694
695
696
697
698
699
700
701
702
703
704
705
706
707
708
709
710
711
712
713
714
715
716
717
718
719
720
721
722
723
724
725
726
727
728
729
730
731
732
733
734
735
736
737
738
739
740
741
742
743
744
745
746
747
748
749
750
751
752
753
754
755
756
757
758
759
760
761
762
763
764
765
766
767
768
769
770
771
772
773
774
775
776
777
778
779
780
781
782
783
784
785
786
787
788
789
790
791
792
793
794
795
796
797
798
799
800
801
802
803
804
805
806
807
808
809
810
811
812
813
814
815
816
817
818
819
820
821
822
823
824
825
826
827
828
829
830
831
832
833
834
835
836
837
838
839
840
841
842
843
844
845
846
847
848
849
850
851
852
853
854
855
856
857
858
859
860
861
862
863
864
865
866
867
868
869
870
871
872
873
874
875
876
877
878
879
880
881
882
883
884
885
886
887
888
889
890
891
892
893
894
895
896
897
898
899
900
901
902
903
904
905
906
907
908
909
910
911
912
913
914
915
916
917
918
919
920
921
922
923
924
925
926
927
928
929
930
931
932
933
934
935
936
937
938
939
940
941
942
943
944
945
946
947
948
949
950
951
952
953
954
955
956
957
958
959
960
961
962
963
964
965
966
967
968
969
970
971
972
973
974
975
976
977
978
979
980
981
982
983
984
985
986
987
988
989
990
991
992
993
994
995
996
997
998
999
1000
```

## 5.2.26 Reading Spindle Speed ► [R\\_Spindle\\_Speed](#)

This request returns the speed of the selected spindle (0...3).

```

1  PROGRAM R_Std_051
2  VAR
3      oR_Spindle_Speed : R_Spindle_Speed;
4      bSpindleN        : USINT;
5      SpindleSpeed      : DINT;
6  END_VAR
7
8  oR_Spindle_Speed.CNCAdr:= ADR(NCK);
9  oR_Spindle_Speed.bySpindleNo:= bSpindleN; // Spindle number 0...3
10 oR_Spindle_Speed(enable:= xEnable);
11
12 IF oR_Spindle_Speed.uiResult = 0 THEN
13     xEnable:= FALSE;
14     SpindleSpeed:= oR_Spindle_Speed.diSpeed;
15 END_IF

```

## 5.2.27 Reading Tool offsets data ► R\_Tool

This request returns all data about a particular tool offset, excluding the H value that can be obtained via the *Read\_E\_Param* request.

```

1  PROGRAM R_Std_049
2  VAR
3      oR_Tool          : R_Tool;
4      diToolPar1       : DINT;
5      diToolPar2       : DINT;
6      diToolPar3       : DINT;
7      diToolPar4       : DINT;
8      diToolPar5       : DINT;
9      diToolPar6       : DINT;
10     diToolPar7       : DINT;
11 END_VAR
12
13 oR_Tool.CNCAdr        := ADR(NCK);
14 oR_Tool.byCorrNo      := 1;      // Tool offset number
15
16 oR_Tool(enable := xEnable);
17
18 IF oR_Tool.uiResult = 0 THEN
19     xEnable            := FALSE;
20
21     diToolPar1         := oR_Tool.diE50xxx;
22     diToolPar2         := oR_Tool.diE51xxx;
23     diToolPar3         := oR_Tool.diE52xxx;
24     diToolPar4         := oR_Tool.diE53xxx;
25     diToolPar5         := oR_Tool.diE54xxx;
26     diToolPar6         := oR_Tool.diE55xxx;
27     diToolPar7         := oR_Tool.diE57xxx;
28 END_IF
29
```

### 5.2.28 Read a part program from NC memory ► [R\\_Upload\\_PP](#)

It will read a part program in NC memory and store it as a file in the system mass memory. The request requires the number of the part program in NC memory (zone 0) as well as the path name used to store it in mass memory.

The part program number passed to the request is separated in two elements. The last digit represents the index and the other digits the part program number itself. The part program is always read from memory zone 0.

This function is described for historical reason but the use of the improved function `R_Upload2_PP` is suggested.

```

1  PROGRAM R_Std_052
2  VAR
3      oR_Upload_PP      : R_Upload_PP;
4      sFileName         : STRING(255); // Path and name, file on the HD
5      PPNumber          : UDINT;      // PP Number*10 + channel
6      dError            : DINT;       // Error code
7  END_VAR
8
9
10 oR_Upload_PP.CNCAddress:= ADR(NCK);
11 sFileName:= 'C:/temp/UploadTest.xpi';
12 PPNumber:= 1000; // %100.0
13 oR_Upload_PP.stFileName:= sFileName;
14 IF xEnable THEN
15     oR_Upload_PP();
16     IF oR_Upload_PP.uiResult = 0 THEN
17         xEnable:= FALSE;
18         dError:= oR_Upload_PP.diError;
19     END_IF
20 END_IF

```

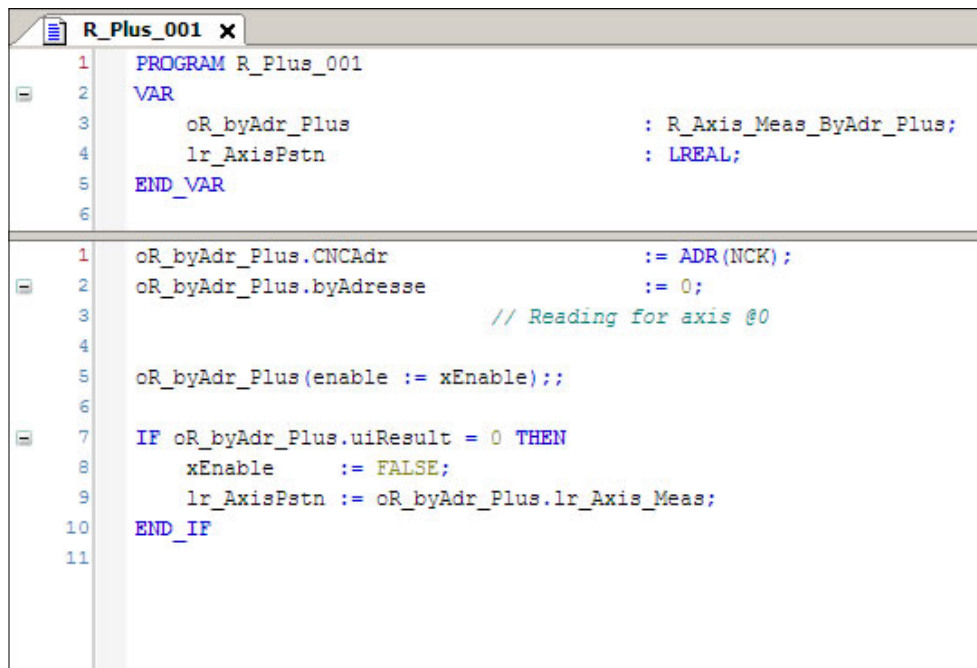
---

## 5.3 Flexium+ requests

### 5.3.1 Reading an axis measure by address ► [R\\_Axis\\_Meas\\_ByAdr\\_Plus](#)

This request will return an axis measure according to the axis physical address. It returns the data for one axis at a time.

To get the data for all axes of a channel, use R\_Axis\_Meas\_Plus (see chapter “5.3.2”).



```
1 PROGRAM R_Plus_001
2 VAR
3     oR_byAdr_Plus                : R_Axis_Meas_ByAdr_Plus;
4     lr_AxisPstn                  : LREAL;
5 END_VAR
6
7 oR_byAdr_Plus.CNCAdr              := ADR(NCK);
8 oR_byAdr_Plus.byAdresse           := 0;
9                                     // Reading for axis @0
10
11 oR_byAdr_Plus(enable := xEnable);
12
13 IF oR_byAdr_Plus.uiResult = 0 THEN
14     xEnable      := FALSE;
15     lr_AxisPstn := oR_byAdr_Plus.lr_Axis_Meas;
16 END_IF
```

### 5.3.2 Reading the measure of all axes in a channel ▶ **R\_Axis\_Meas\_Plus**

This request will return the measure of all axes in a particular channel.

Contrary to *R\_Axis\_Meas\_ByAdr* the data are sorted according to the axis name and not the physical address.

```

1  PROGRAM R_Plus_002
2  VAR
3      oR_Meas_Plus          : R_Axis_Meas_Plus;
4      lrMeas_X              : LREAL;
5  END_VAR

1  oR_Meas_Plus.CNCAdr       := ADR(NCK);
2  oR_Meas_Plus.byChannel    := 0;
3                               // Channels numbered 0 to 7
4
5  oR_Meas_Plus(enable := xEnable);;
6
7  IF oR_Meas_Plus.uiResult = 0 THEN
8      xEnable               := FALSE;
9      lrMeas_X              := oR_Meas_Plus.lrX_Axis;
10 END_IF
11

```

### 5.3.3 Reading machine data about axes in motion ► R\_Axis\_OM\_Plus

This request returns Position in the machine referential as well as distance to go, lag and resolution about the axes of a particular channel.

The data are returned in a structure *Axis[i]* and are individually validated by a flag *xxxValid* e.g. *LagValid*.

```

1  PROGRAM R_Plus_003
2  VAR
3      oR_Meas_OM_Plus          : R_Axis_OM_Plus;
4      Pos_0                    : LREAL;
5      ToGo_0                    : LREAL;
6      Lag_0                     : LREAL;
7      Res_0                     : LREAL;
8  END_VAR
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24
25
26
27
28
29
30
31
32
33
34
35
36
37
38
39
40
41
42
43
44
45
46
47
48
49
50
51
52
53
54
55
56
57
58
59
60
61
62
63
64
65
66
67
68
69
70
71
72
73
74
75
76
77
78
79
80
81
82
83
84
85
86
87
88
89
90
91
92
93
94
95
96
97
98
99
100
101
102
103
104
105
106
107
108
109
110
111
112
113
114
115
116
117
118
119
120
121
122
123
124
125
126
127
128
129
130
131
132
133
134
135
136
137
138
139
140
141
142
143
144
145
146
147
148
149
150
151
152
153
154
155
156
157
158
159
160
161
162
163
164
165
166
167
168
169
170
171
172
173
174
175
176
177
178
179
180
181
182
183
184
185
186
187
188
189
190
191
192
193
194
195
196
197
198
199
200
201
202
203
204
205
206
207
208
209
210
211
212
213
214
215
216
217
218
219
220
221
222
223
224
225
226
227
228
229
230
231
232
233
234
235
236
237
238
239
240
241
242
243
244
245
246
247
248
249
250
251
252
253
254
255
256
257
258
259
260
261
262
263
264
265
266
267
268
269
270
271
272
273
274
275
276
277
278
279
280
281
282
283
284
285
286
287
288
289
290
291
292
293
294
295
296
297
298
299
300
301
302
303
304
305
306
307
308
309
310
311
312
313
314
315
316
317
318
319
320
321
322
323
324
325
326
327
328
329
330
331
332
333
334
335
336
337
338
339
340
341
342
343
344
345
346
347
348
349
350
351
352
353
354
355
356
357
358
359
360
361
362
363
364
365
366
367
368
369
370
371
372
373
374
375
376
377
378
379
380
381
382
383
384
385
386
387
388
389
390
391
392
393
394
395
396
397
398
399
400
401
402
403
404
405
406
407
408
409
410
411
412
413
414
415
416
417
418
419
420
421
422
423
424
425
426
427
428
429
430
431
432
433
434
435
436
437
438
439
440
441
442
443
444
445
446
447
448
449
450
451
452
453
454
455
456
457
458
459
460
461
462
463
464
465
466
467
468
469
470
471
472
473
474
475
476
477
478
479
480
481
482
483
484
485
486
487
488
489
490
491
492
493
494
495
496
497
498
499
500
501
502
503
504
505
506
507
508
509
510
511
512
513
514
515
516
517
518
519
520
521
522
523
524
525
526
527
528
529
530
531
532
533
534
535
536
537
538
539
540
541
542
543
544
545
546
547
548
549
550
551
552
553
554
555
556
557
558
559
560
561
562
563
564
565
566
567
568
569
570
571
572
573
574
575
576
577
578
579
580
581
582
583
584
585
586
587
588
589
590
591
592
593
594
595
596
597
598
599
600
601
602
603
604
605
606
607
608
609
610
611
612
613
614
615
616
617
618
619
620
621
622
623
624
625
626
627
628
629
630
631
632
633
634
635
636
637
638
639
640
641
642
643
644
645
646
647
648
649
650
651
652
653
654
655
656
657
658
659
660
661
662
663
664
665
666
667
668
669
670
671
672
673
674
675
676
677
678
679
680
681
682
683
684
685
686
687
688
689
690
691
692
693
694
695
696
697
698
699
700
701
702
703
704
705
706
707
708
709
710
711
712
713
714
715
716
717
718
719
720
721
722
723
724
725
726
727
728
729
730
731
732
733
734
735
736
737
738
739
740
741
742
743
744
745
746
747
748
749
750
751
752
753
754
755
756
757
758
759
760
761
762
763
764
765
766
767
768
769
770
771
772
773
774
775
776
777
778
779
780
781
782
783
784
785
786
787
788
789
790
791
792
793
794
795
796
797
798
799
800
801
802
803
804
805
806
807
808
809
810
811
812
813
814
815
816
817
818
819
820
821
822
823
824
825
826
827
828
829
830
831
832
833
834
835
836
837
838
839
840
841
842
843
844
845
846
847
848
849
850
851
852
853
854
855
856
857
858
859
860
861
862
863
864
865
866
867
868
869
870
871
872
873
874
875
876
877
878
879
880
881
882
883
884
885
886
887
888
889
890
891
892
893
894
895
896
897
898
899
900
901
902
903
904
905
906
907
908
909
910
911
912
913
914
915
916
917
918
919
920
921
922
923
924
925
926
927
928
929
930
931
932
933
934
935
936
937
938
939
940
941
942
943
944
945
946
947
948
949
950
951
952
953
954
955
956
957
958
959
960
961
962
963
964
965
966
967
968
969
970
971
972
973
974
975
976
977
978
979
980
981
982
983
984
985
986
987
988
989
990
991
992
993
994
995
996
997
998
999
1000
1001
1002
1003
1004
1005
1006
1007
1008
1009
1010
1011
1012
1013
1014
1015
1016
1017
1018
1019
1020
1021
1022
1023
1024
1025
1026
1027
1028
1029
1030
1031
1032
1033
1034
1035
1036
1037
1038
1039
1040
1041
1042
1043
1044
1045
1046
1047
1048
1049
1050
1051
1052
1053
1054
1055
1056
1057
1058
1059
1060
1061
1062
1063
1064
1065
1066
1067
1068
1069
1070
1071
1072
1073
1074
1075
1076
1077
1078
1079
1080
1081
1082
1083
1084
1085
1086
1087
1088
1089
1090
1091
1092
1093
1094
1095
1096
1097
1098
1099
1100
1101
1102
1103
1104
1105
1106
1107
1108
1109
1110
1111
1112
1113
1114
1115
1116
1117
1118
1119
1120
1121
1122
1123
1124
1125
1126
1127
1128
1129
1130
1131
1132
1133
1134
1135
1136
1137
1138
1139
1140
1141
1142
1143
1144
1145
1146
1147
1148
1149
1150
1151
1152
1153
1154
1155
1156
1157
1158
1159
1160
1161
1162
1163
1164
1165
1166
1167
1168
1169
1170
1171
1172
1173
1174
1175
1176
1177
1178
1179
1180
1181
1182
1183
1184
1185
1186
1187
1188
1189
1190
1191
1192
1193
1194
1195
1196
1197
1198
1199
1200
1201
1202
1203
1204
1205
1206
1207
1208
1209
1210
1211
1212
1213
1214
1215
1216
1217
1218
1219
1220
1221
1222
1223
1224
1225
1226
1227
1228
1229
1230
1231
1232
1233
1234
1235
1236
1237
1238
1239
1240
1241
1242
1243
1244
1245
1246
1247
1248
1249
1250
1251
1252
1253
1254
1255
1256
1257
1258
1259
1260
1261
1262
1263
1264
1265
1266
1267
1268
1269
1270
1271
1272
1273
1274
1275
1276
1277
1278
1279
1280
1281
1282
1283
1284
1285
1286
1287
1288
1289
1290
1291
1292
1293
1294
1295
1296
1297
1298
1299
1300
1301
1302
1303
1304
1305
1306
1307
1308
1309
1310
1311
1312
1313
1314
1315
1316
1317
1318
1319
1320
1321
1322
1323
1324
1325
1326
1327
1328
1329
1330
1331
1332
1333
1334
1335
1336
1337
1338
1339
1340
1341
1342
1343
1344
1345
1346
1347
1348
1349
1350
1351
1352
1353
1354
1355
1356
1357
1358
1359
1360
1361
1362
1363
1364
1365
1366
1367
1368
1369
1370
1371
1372
1373
1374
1375
1376
1377
1378
1379
1380
1381
1382
1383
1384
1385
1386
1387
1388
1389
1390
1391
1392
1393
1394
1395
1396
1397
1398
1399
1400
1401
1402
1403
1404
1405
1406
1407
1408
1409
1410
1411
1412
1413
1414
1415
1416
1417
1418
1419
1420
1421
1422
1423
1424
1425
1426
1427
1428
1429
1430
1431
1432
1433
1434
1435
1436
1437
1438
1439
1440
1441
1442
1443
1444
1445
1446
1447
1448
1449
1450
1451
1452
1453
1454
1455
1456
1457
1458
1459
1460
1461
1462
1463
1464
1465
1466
1467
1468
1469
1470
1471
1472
1473
1474
1475
1476
1477
1478
1479
1480
1481
1482
1483
1484
1485
1486
1487
1488
1489
1490
1491
1492
1493
1494
1495
1496
1497
1498
1499
1500
1501
1502
1503
1504
1505
1506
1507
1508
1509
1510
1511
1512
1513
1514
1515
1516
1517
1518
1519
1520
1521
1522
1523
1524
1525
1526
1527
1528
1529
1530
1531
1532
1533
1534
1535
1536
1537
1538
1539
1540
1541
1542
1543
1544
1545
1546
1547
1548
1549
1550
1551
1552
1553
1554
1555
1556
1557
1558
1559
1560
1561
1562
1563
1564
1565
1566
1567
1568
1569
1570
1571
1572
1573
1574
1575
1576
1577
1578
1579
1580
1581
1582
1583
1584
1585
1586
1587
1588
1589
1590
1591
1592
1593
1594
1595
1596
1597
1598
1599
1600
1601
1602
1603
1604
1605
1606
1607
1608
1609
1610
1611
1612
1613
1614
1615
1616
1617
1618
1619
1620
1621
1622
1623
1624
1625
1626
1627
1628
1629
1630
1631
1632
1633
1634
1635
1636
1637
1638
1639
1640
1641
1642
1643
1644
1645
1646
1647
1648
1649
1650
1651
1652
1653
1654
1655
1656
1657
1658
1659
1660
1661
1662
1663
1664
1665
1666
1667
1668
1669
1670
1671
1672
1673
1674
1675
1676
1677
1678
1679
1680
1681
1682
1683
1684
1685
1686
1687
1688
1689
1690
1691
1692
1693
1694
1695
1696
1697
1698
1699
1700
1701
1702
1703
1704
1705
1706
1707
1708
1709
1710
1711
1712
1713
1714
1715
1716
1717
1718
1719
1720
1721
1722
1723
1724
1725
1726
1727
1728
1729
1730
1731
1732
1733
1734
1735
1736
1737
1738
1739
1740
1741
1742
1743
1744
1745
1746
1747
1748
1749
1750
1751
1752
1753
1754
1755
1756
1757
1758
1759
1760
1761
1762
1763
1764
1765
1766
1767
1768
1769
1770
1771
1772
1773
1774
1775
1776
1777
1778
1779
1780
1781
1782
1783
1784
1785
1786
1787
1788
1789
1790
1791
1792
1793
1794
1795
1796
1797
1798
1799
1800
1801
1802
1803
1804
1805
1806
1807
1808
1809
1810
1811
1812
1813
1814
1815
1816
1817
1818
1819
1820
1821
1822
1823
1824
1825
1826
1827
1828
1829
1830
1831
1832
1833
1834
1835
1836
1837
1838
1839
1840
1841
1842
1843
1844
1845
1846
1847
1848
1849
1850
1851
1852
1853
1854
1855
1856
1857
1858
1859
1860
1861
1862
1863
1864
1865
1866
1867
1868
1869
1870
1871
1872
1873
1874
1875
1876
1877
1878
1879
1880
1881
1882
1883
1884
1885
1886
1887
1888
1889
1890
1891
1892
1893
1894
1895
1896
1897
1898
1899
1900
1901
1902
1903
1904
1905
1906
1907
1908
1909
1910
1911
1912
1913
1914
1915
1916
1917
1918
1919
1920
1921
1922
1923
1924
1925
1926
1927
1928
1929
1930
1931
1932
1933
1934
1935
1936
1937
1938
1939
1940
1941
1942
1943
1944
1945
1946
1947
1948
1949
1950
1951
1952
1953
1954
1955
1956
1957
1958
1959
1960
1961
1962
1963
1964
1965
1966
1967
1968
1969
1970
1971
1972
1973
1974
1975
1976
1977
1978
1979
1980
1981
1982
1983
1984
1985
1986
1987
1988
1989
1990
1991
1992
1993
1994
1995
1996
1997
1998
1999
2000
2001
2002
2003
2004
2005
2006
2007
2008
2009
2010
2011
2012
2013
2014
2015
2016
2017
2018
2019
2020
2021
2022
2023
2024
2025
2026
2027
2028
2029
2030
2031
2032
2033
2034
2035
2036
2037
2038
2039
2040
2041
2042
2043
2044
2045
2046
2047
2048
2049
2050
2051
2052
2053
2054
2055
2056
2057
2058
2059
2060
2061
2062
2063
2064
2065
2066
2067
2068
2069
2070
2071
2072
2073
2074
2075
2076
2077
2078
2079
2080
2081
2082
2083
2084
2085
2086
2087
2088
2089
2090
2091
2092
2093
2094
2095
2096
2097
2098
2099
2100
2101
2102
2103
2104
2105
2106
2107
2108
2109
2110
2111
2112
2113
2114
2115
2116
2117
2118
2119
2120
2121
2122
2123
2124
2125
2126
2127
2128
2129
2130
2131
2132
2133
2134
2135
2136
2137
2138
2139
2140
2141
2142
2143
2144
2145
2146
2147
2148
2149
2150
2151
2152
2153
2154
2155
2156
2157
2158
2159
2160
2161
2162
2163
2164
2165
2166
2167
2168
2169
2170
2171
2172
2173
2174
2175
2176
2177
2178
2179
2180
2181
2182
2183
2184
2185
2186
2187
2188
2189
2190
2191
2192
2193
2194
2195
2196
2197
2198
2199
2200
2201
2202
2203
2204
2205
2206
2207
2208
2209
2210
2211
2212
2213
2214
2215
2216
2217
2218
2219
2220
2221
2222
2223
2224
2225
2226
2227
2228
2229
2230
2231
2232
2233
2234
2235
2236
2237
2238
2239
2240
2241
2242
2243
2244
2245
2246
2247
2248
2249
2250
2251
2252
2253
2254
2255
2256
2257
2258
2259
2260
2261
2262
2263
2264
2265
2266
2267
2268
2269
2270
2271
2272
2273
2274
2275
2276
2277
2278
2279
2280
2281
2282
2283
2284
2285
2286
2287
2288
2289
2290
2291
2292
2293
2294
2295
2296
2297
2298
2299
2300
2301
2302
2303
2304
2305
2306
2307
2308
2309
2310
2311
2312
2313
2314
2315
2316
2317
2318
2319
2320
2321
2322
2323
2324
2325
2326
2327
2328
2329
2330
2331
2332
2333
2334
2335
2336
2337
2338
2339
2340
2341
2342
2343
2344
2345
2346
2347
2348
2349
2350
2351
2352
2353
2354
2355
2356
2357
2358
2359
2360
2361
2362
2363
2364
2365
2366
2367
2368
2369
2370
2371
2372
2373
2374
2375
2376
2377
2378
2379
2380
2381
2382
2383
2384
2385
2386
2387
2388
2389
2390
2391
2392
2393
2394
2395
2396
2397
2398
2399
2400
2401
2402
2403
2404
2405
2406
2407
2408
2409
2410
2411
2412
2413
2414
2415
2416
2417
2418
2419
2420
2421
2422
2423
2424
2425
2426
2427
2428
2429
2430
2431
2432
2433
2434
2435
2436
2437
2438
2439
2440
2441
2442
2443
2444
2445
2446
2447
2448
2449
2450
2451
2452
2453
2454
2455
2456
2457
2458
2459
2460
2461
2462
2463
2464
2465
2466
2467
2468
2469
2470
2471
2472
2473
2474
2475
2476
2477
2478
2479
2480
2481
2482
2483
2484
2485
2486
2487
2488
2489
2490
2491
2492
2493
2494
2495
2496
2497
2498
2499
2500
2501
2502
2503
2504
2505
2506
2507
2508
2509
2510
2511
2512
2513
2514
2515
2516
2517
2518
2519
2520
2521
2522
2523
2524
2525
2526
2527
2528
2529
2530
2531
2532
2533
2534
2535
2536
2537
2538
2539
2540
2541
2542
2543
2544
2545
2546
2547
2548
2549
2550
2551
2552
2553
2554
2555
2556
2557
2558
2559
2560
2561
2562
2563
2564
2565
2566
2567
2568
2569
2570
2571
2572
2573
2574
2575
2576
2577
2578
2579
2580
2581
2582
2583
2584
2585
2586
2587
2588
2589
2590
2591
2592
2593
2594
2595
2596
2597
2598
2599
2600
2601
2602
2603
2604
2605
2606
2607
2608
2609
2610
2611
2612
2613
2614
2615
2616
2617
2618
2619
2620
2621
2622
2623
2624
2625
2626
2627
2628
2629
2630
2631
2632
2633
2634
2635
2636
2637
2638
2639
2640
2641
26
```

### 5.3.4 Reading part origin data about axes in motion ► R\_Axis\_OP\_Plus

This request returns Position in the part programming referential as well as distance to go, lag and resolution about the axes of a particular channel.

The data are returned in a structure *Axis[i]* and are individually validated by a flag *xxxValid* e.g. *LagValid*.

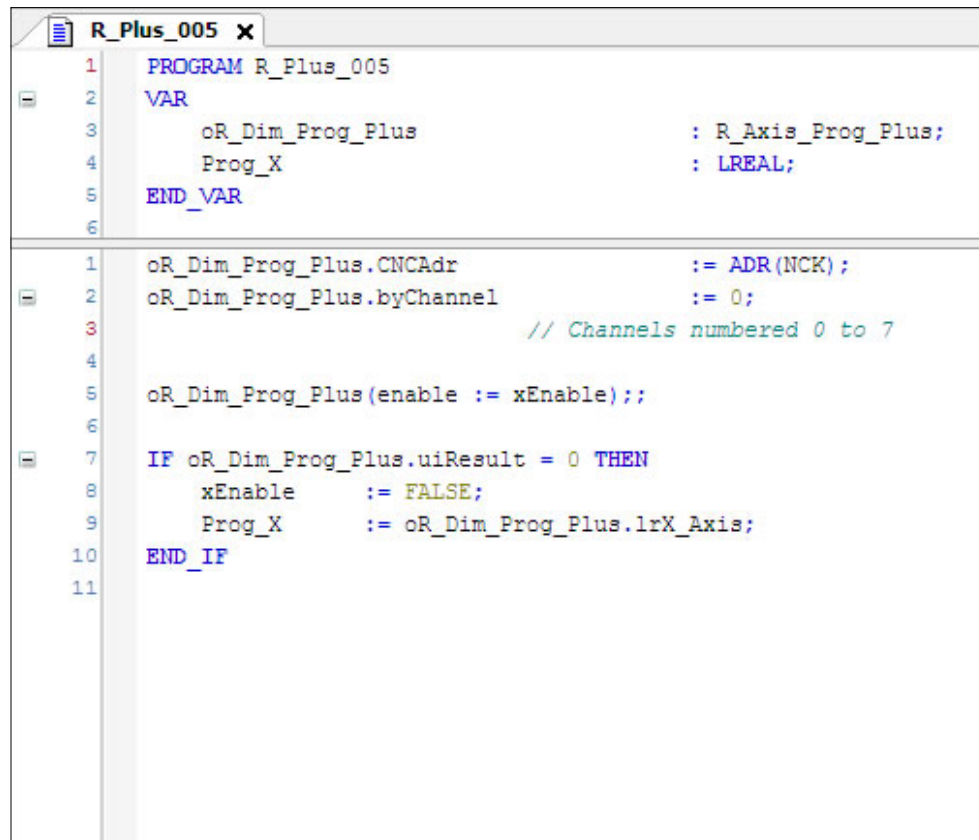
```

1  PROGRAM R_Plus_004
2  VAR
3      oR_Meas_OP_Plus          : R_Axis_OP_Plus;
4      Pos_0                    : LREAL;
5      ToGo_0                   : LREAL;
6      Lag_0                    : LREAL;
7      Res_0                    : LREAL;
8  END_VAR
9
10 oR_Meas_OP_Plus.CNCAdr        := ADR(NCK);
11 oR_Meas_OP_Plus.byChannel     := 0;
12                               // Channels numbered 0 to 7
13
14 oR_Meas_OP_Plus(enable := xEnable);
15
16 IF oR_Meas_OP_Plus.uiResult = 0 THEN
17     xEnable := FALSE;
18     IF oR_Meas_OP_Plus.Axis[0].PosValid THEN
19         Pos_0 := oR_Meas_OP_Plus.Axis[0].Position;
20     END_IF
21     IF oR_Meas_OP_Plus.Axis[0].ToGoValid THEN
22         ToGo_0 := oR_Meas_OP_Plus.Axis[0].ToGo;
23     END_IF
24     IF oR_Meas_OP_Plus.Axis[0].LagValid THEN
25         Lag_0 := oR_Meas_OP_Plus.Axis[0].Lag;
26     END_IF
27     Res_0 := oR_Meas_OP_Plus.Axis[0].Resolution;
28 END_IF
29

```

### 5.3.5 Reading axes programmed position ► [R\\_Axis\\_Prog\\_Plus](#)

This request returns the programmed position of all axes in a particular channel.  
The data are returned with respect to the axis name in the channel.



```
1 PROGRAM R_Plus_005
2 VAR
3     oR_Dim_Prog_Plus          : R_Axis_Prog_Plus;
4     Prog_X                    : LREAL;
5 END_VAR
6
7 oR_Dim_Prog_Plus.CNCAdr       := ADR(NCK);
8 oR_Dim_Prog_Plus.byChannel    := 0;
9                               // Channels numbered 0 to 7
10
11 oR_Dim_Prog_Plus(enable := xEnable);
12
13 IF oR_Dim_Prog_Plus.uiResult = 0 THEN
14     xEnable := FALSE;
15     Prog_X := oR_Dim_Prog_Plus.lrx_Axis;
16 END_IF
```

### 5.3.6 Reading axes reference ► R\_Axis\_Ref\_Plus

This request returns the reference value (in the machine referential) of the axes in a particular channel. The data are returned with respect to the axis name.

```

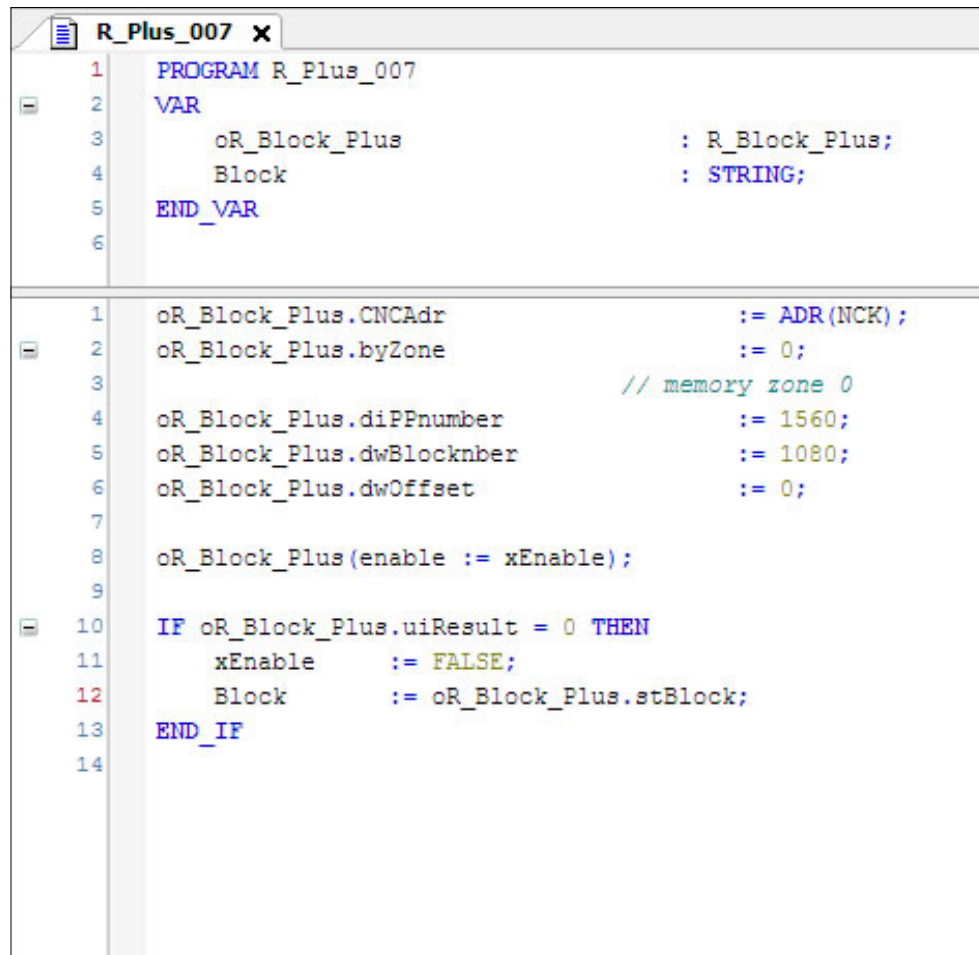
1  PROGRAM R_Plus_006
2  VAR
3      oR_Axis_Ref_Plus          : R_Axis_Ref_Plus;
4      Ref_X                    : LREAL;
5  END_VAR
6
7  oR_Axis_Ref_Plus.CNCAdr       := ADR(NCK);
8  oR_Axis_Ref_Plus.byChannel    := 0;
9                                // Channels numbered 0 to 7
10
11  oR_Axis_Ref_Plus(enable := xEnable);
12
13  IF oR_Axis_Ref_Plus.uiResult = 0 THEN
14      xEnable      := FALSE;
15      Ref_X        := oR_Axis_Ref_Plus.lrX_Axis;
16  END_IF

```

### 5.3.7 Reading a part program block ► R\_Block\_Plus

This request returns a string corresponding to a particular part program block.

Input data will be the memory zone where the part program is searched for, the part program number (times 10 and including the index), the block number and the offset (in number of blocks) relative to this number. In case of non numbered part program, the number 0 corresponds to the first line of program.



```
1 PROGRAM R_Plus_007
2 VAR
3     oR_Block_Plus          : R_Block_Plus;
4     Block                  : STRING;
5 END_VAR
6
7
8 oR_Block_Plus.CNCAdr        := ADR(NCK);
9 oR_Block_Plus.byZone        := 0;
10 // memory zone 0
11 oR_Block_Plus.diPPnumber    := 1560;
12 oR_Block_Plus.dwBlocknber   := 1080;
13 oR_Block_Plus.dwOffset      := 0;
14
15 oR_Block_Plus(enable := xEnable);
16
17 IF oR_Block_Plus.uiResult = 0 THEN
18     xEnable      := FALSE;
19     Block        := oR_Block_Plus.stBlock;
20 END_IF
```

### 5.3.8 Reading an axis measure correction (E952xx) ► [R\\_Corr\\_Dyn\\_Plus](#)

This request returns the value of parameter E952xx (measure correction) of a particular axis.

```

1  PROGRAM R_Plus_008
2  VAR
3      oR_Corr_Dyn_Plus          : R_Corr_Dyn_Plus;
4      Corr_X                    : LREAL;
5  END_VAR
6
7  oR_Corr_Dyn_Plus.CNCAdr        := ADR(NCK);
8  oR_Corr_Dyn_Plus.byAxisNo     := 0;
9                                  // axis 0
10
11  oR_Corr_Dyn_Plus(enable := xEnable);
12
13  IF oR_Corr_Dyn_Plus.uiResult = 0 THEN
14      xEnable := FALSE;
15      corr_X  := oR_Corr_Dyn_Plus.lrCorrDyn;
16  END_IF

```

### 5.3.9 Reading the part program block in execution ► R\_Curr\_Block\_Plus

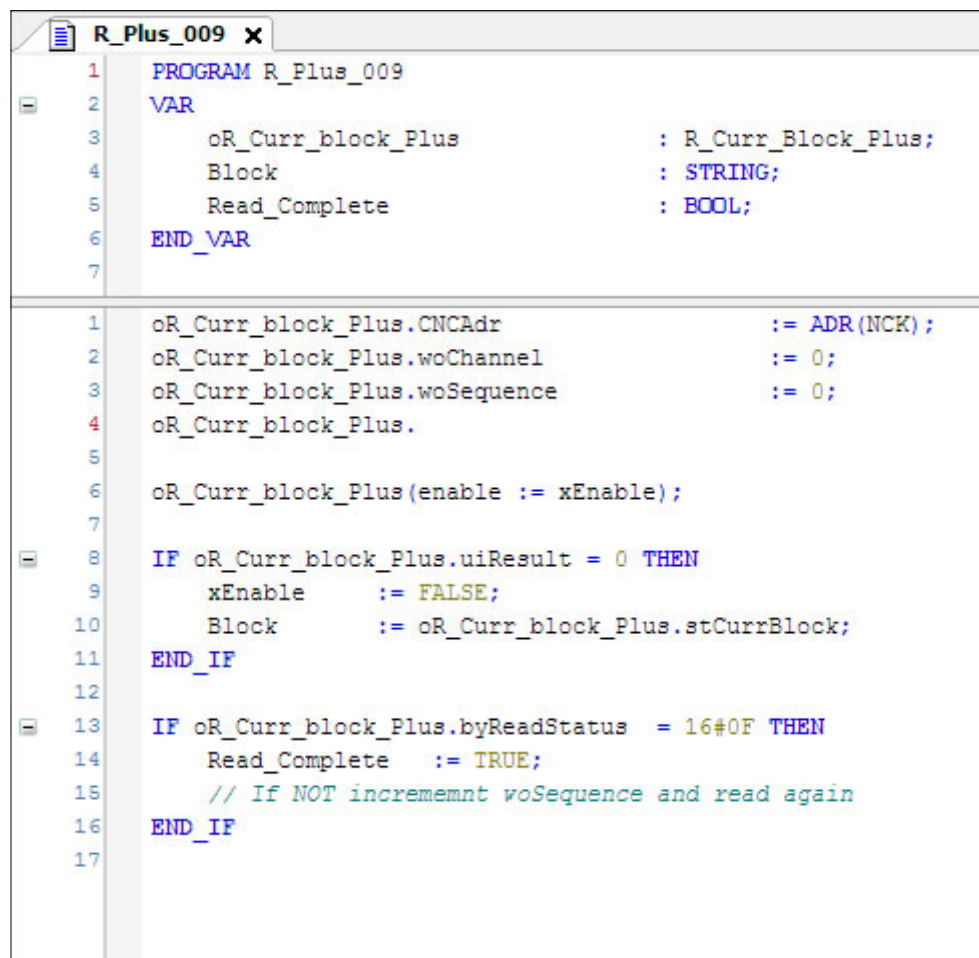
This request returns a string corresponding to the block in execution. As the answer can take several steps a *Read\_Complete* status flag is part of the answer.

Return code of *byReadStatus*:

- 0x06 : Block reading incomplete : Increment *woSequence* to get the rest of the block
- 0x0F : Block read finished *woSequence* = 0 to read new block
- 0x19 : Sequencing error.

The first sequence number should be 0.

If the block is not read entirely then read the next part after incrementing the sequence number: *woSequence* = *woSequence* + 1 and so on until *byReadStatus* = 0x0F.



```
1 PROGRAM R_Plus_009
2 VAR
3     oR_Curr_block_Plus      : R_Curr_Block_Plus;
4     Block                   : STRING;
5     Read_Complete           : BOOL;
6 END_VAR
7
8 oR_Curr_block_Plus.CNCAdr    := ADR(NCK);
9 oR_Curr_block_Plus.woChannel := 0;
10 oR_Curr_block_Plus.woSequence := 0;
11 oR_Curr_block_Plus.
12
13 oR_Curr_block_Plus(enable := xEnable);
14
15 IF oR_Curr_block_Plus.uiResult = 0 THEN
16     xEnable      := FALSE;
17     Block        := oR_Curr_block_Plus.stCurrBlock;
18 END_IF
19
20 IF oR_Curr_block_Plus.byReadStatus = 16#0F THEN
21     Read_Complete := TRUE;
22     // If NOT increment woSequence and read again
23 END_IF
```

### 5.3.10 Reading current Dat1 value ► R\_Dat1\_Pref\_Plus

This request returns the current DAT1 set of values for a particular channel.  
The data are returned with respect to the axes names.

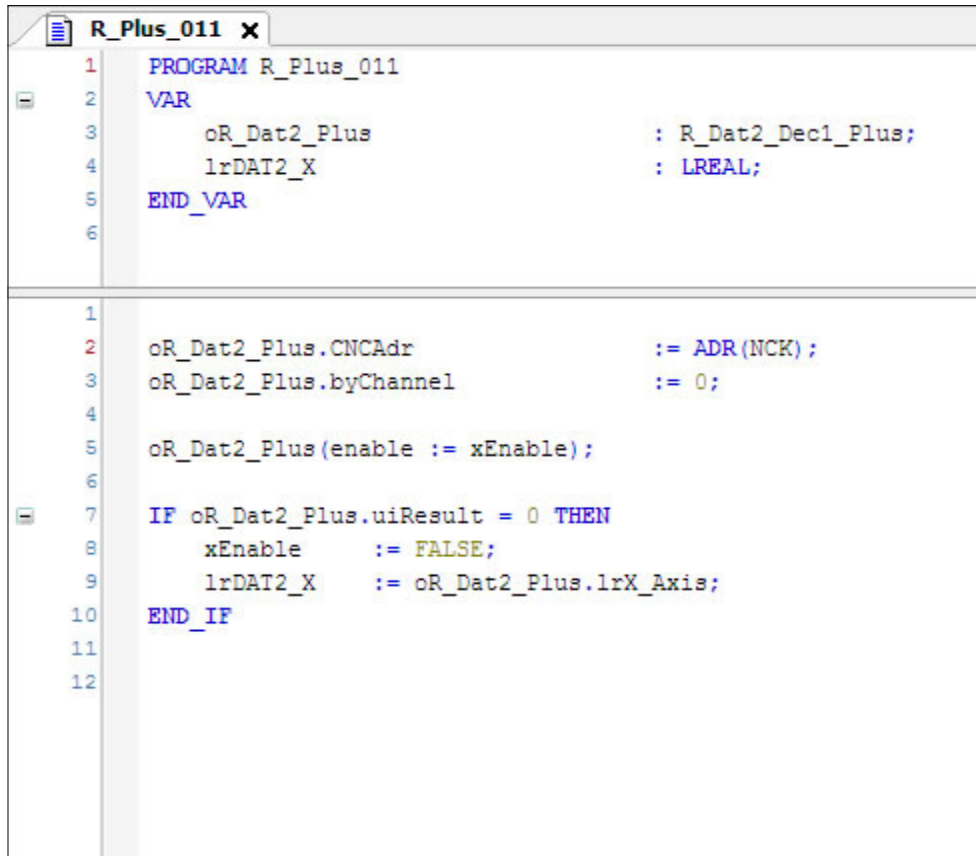
```

1  PROGRAM R_Plus_010
2  VAR
3      oR_Dat1_Plus          : R_Dat1_Pref_Plus;
4      lrDAT1_X              : LREAL;
5  END_VAR
6
7  oR_Dat1_Plus.CNCAdr       := ADR(NCK);
8  oR_Dat1_Plus.byChannel    := 0;
9
10 oR_Dat1_Plus(enable := xEnable);
11
12 IF oR_Dat1_Plus.uiResult = 0 THEN
13     xEnable      := FALSE;
14     lrDAT1_X     := oR_Dat1_Plus.lrX_Axis;
15 END_IF

```

### 5.3.11 Reading current Dat2 value ► [R\\_Dat2\\_Dec1\\_Plus](#)

This request returns the current DAT2 set of values for a particular channel.  
The data are returned with respect to the axes names.



```
1  PROGRAM R_Plus_011
2  VAR
3      oR_Dat2_Plus          : R_Dat2_Dec1_Plus;
4      lrDAT2_X              : LREAL;
5  END_VAR
6
7
8  oR_Dat2_Plus.CNCAdr       := ADR(NCK);
9  oR_Dat2_Plus.byChannel    := 0;
10
11  oR_Dat2_Plus(enable := xEnable);
12
13  IF oR_Dat2_Plus.uiResult = 0 THEN
14      xEnable      := FALSE;
15      lrDAT2_X     := oR_Dat2_Plus.lrX_Axis;
16  END_IF
```

### 5.3.12 Reading current Dat3 value ► [R\\_Dat3\\_Dec3\\_Plus](#)

This request returns the current DAT3 set of values for a particular channel. The data are returned with respect to the axes names.

NB : It is possible to enter a value for any of the linear axes of a channel, this request does not consider which axes are currently involved in DEC3. As well it does not change the current affectation.

```

1  PROGRAM R_Plus_012
2  VAR
3      oR_Dat3_Plus          : R_Dat3_Dec3_Plus;
4      lrDAT3_X              : LREAL;
5  END_VAR
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24
25
26
27
28
29
30
31
32
33
34
35
36
37
38
39
40
41
42
43
44
45
46
47
48
49
50
51
52
53
54
55
56
57
58
59
60
61
62
63
64
65
66
67
68
69
70
71
72
73
74
75
76
77
78
79
80
81
82
83
84
85
86
87
88
89
90
91
92
93
94
95
96
97
98
99
100
101
102
103
104
105
106
107
108
109
110
111
112
113
114
115
116
117
118
119
120
121
122
123
124
125
126
127
128
129
130
131
132
133
134
135
136
137
138
139
140
141
142
143
144
145
146
147
148
149
150
151
152
153
154
155
156
157
158
159
160
161
162
163
164
165
166
167
168
169
170
171
172
173
174
175
176
177
178
179
180
181
182
183
184
185
186
187
188
189
190
191
192
193
194
195
196
197
198
199
200
201
202
203
204
205
206
207
208
209
210
211
212
213
214
215
216
217
218
219
220
221
222
223
224
225
226
227
228
229
230
231
232
233
234
235
236
237
238
239
240
241
242
243
244
245
246
247
248
249
250
251
252
253
254
255
256
257
258
259
260
261
262
263
264
265
266
267
268
269
270
271
272
273
274
275
276
277
278
279
280
281
282
283
284
285
286
287
288
289
290
291
292
293
294
295
296
297
298
299
300
301
302
303
304
305
306
307
308
309
310
311
312
313
314
315
316
317
318
319
320
321
322
323
324
325
326
327
328
329
330
331
332
333
334
335
336
337
338
339
340
341
342
343
344
345
346
347
348
349
350
351
352
353
354
355
356
357
358
359
360
361
362
363
364
365
366
367
368
369
370
371
372
373
374
375
376
377
378
379
380
381
382
383
384
385
386
387
388
389
390
391
392
393
394
395
396
397
398
399
400
401
402
403
404
405
406
407
408
409
410
411
412
413
414
415
416
417
418
419
420
421
422
423
424
425
426
427
428
429
430
431
432
433
434
435
436
437
438
439
440
441
442
443
444
445
446
447
448
449
450
451
452
453
454
455
456
457
458
459
460
461
462
463
464
465
466
467
468
469
470
471
472
473
474
475
476
477
478
479
480
481
482
483
484
485
486
487
488
489
490
491
492
493
494
495
496
497
498
499
500
501
502
503
504
505
506
507
508
509
510
511
512
513
514
515
516
517
518
519
520
521
522
523
524
525
526
527
528
529
530
531
532
533
534
535
536
537
538
539
540
541
542
543
544
545
546
547
548
549
550
551
552
553
554
555
556
557
558
559
560
561
562
563
564
565
566
567
568
569
570
571
572
573
574
575
576
577
578
579
580
581
582
583
584
585
586
587
588
589
590
591
592
593
594
595
596
597
598
599
600
601
602
603
604
605
606
607
608
609
610
611
612
613
614
615
616
617
618
619
620
621
622
623
624
625
626
627
628
629
630
631
632
633
634
635
636
637
638
639
640
641
642
643
644
645
646
647
648
649
650
651
652
653
654
655
656
657
658
659
660
661
662
663
664
665
666
667
668
669
670
671
672
673
674
675
676
677
678
679
680
681
682
683
684
685
686
687
688
689
690
691
692
693
694
695
696
697
698
699
700
701
702
703
704
705
706
707
708
709
710
711
712
713
714
715
716
717
718
719
720
721
722
723
724
725
726
727
728
729
730
731
732
733
734
735
736
737
738
739
740
741
742
743
744
745
746
747
748
749
750
751
752
753
754
755
756
757
758
759
760
761
762
763
764
765
766
767
768
769
770
771
772
773
774
775
776
777
778
779
780
781
782
783
784
785
786
787
788
789
790
791
792
793
794
795
796
797
798
799
800
801
802
803
804
805
806
807
808
809
810
811
812
813
814
815
816
817
818
819
820
821
822
823
824
825
826
827
828
829
830
831
832
833
834
835
836
837
838
839
840
841
842
843
844
845
846
847
848
849
850
851
852
853
854
855
856
857
858
859
860
861
862
863
864
865
866
867
868
869
870
871
872
873
874
875
876
877
878
879
880
881
882
883
884
885
886
887
888
889
890
891
892
893
894
895
896
897
898
899
900
901
902
903
904
905
906
907
908
909
910
911
912
913
914
915
916
917
918
919
920
921
922
923
924
925
926
927
928
929
930
931
932
933
934
935
936
937
938
939
940
941
942
943
944
945
946
947
948
949
950
951
952
953
954
955
956
957
958
959
960
961
962
963
964
965
966
967
968
969
970
971
972
973
974
975
976
977
978
979
980
981
982
983
984
985
986
987
988
989
990
991
992
993
994
995
996
997
998
999
1000

```

### 5.3.13 Reading Drives Test points values ► [R\\_Drive\\_TPValue\\_V2\\_Plus](#)

This request returns up to eight test point values of one or several drives.

The test points configuration is defined in the drive parameters V130 to V133 and the values read are expressed in physical units.

```
R_Plus_020 x
1  PROGRAM R_Plus_020
2  VAR
3      oR_IP_Value          : R_Drive_TPValue_V2_Plus;
4      Drives               : ARRAY[1..8] OF BYTE
5                          := [0,0,0,0,1,1,1,1];
6      Points               : ARRAY[1..8] OF BYTE
7                          := [0,1,2,3,0,1,2,3];
8      Values               : ARRAY[1..8] OF LREAL;
9  END_VAR
10
11
12  oR_IP_Value.CNCAdr       := ADR(NCK);
13  oR_IP_Value.DriveAdresse := ADR(Drives);
14  oR_IP_Value.DriveTP      := ADR(Points);
15  oR_IP_Value.Number       := 4;
16
17  oR_IP_Value(enable := xEnable);
18
19  IF oR_IP_Value.uiResult = 0 THEN
20      xEnable := FALSE;
21      Values[1] := oR_IP_Value.Values[0];
22      Values[2] := oR_IP_Value.Values[1];
23      Values[3] := oR_IP_Value.Values[2];
24      Values[4] := oR_IP_Value.Values[3];
25  END_IF
26
```

### 5.3.14 Reading a set of E8xyyy parameters ▶ [R\\_E8xyyy\\_Plus](#)

This request returns a set of 10 E8.... parameters from the series E80, E81, E82 E83 or E84.

```

1  PROGRAM R_Plus_030
2  VAR
3      oR_E8xyyy          : R_E8xyyy_Plus;
4      Para               : UDINT := 80000;
5      Values             : ARRAY[1..10] OF LREAL;
6  END_VAR
7
8  oR_E8xyyy.CNCAdr       := ADR(NCK);
9  oR_E8xyyy.udE8xyyyNo  := Para;
10
11  oR_E8xyyy(enable := xEnable);
12
13  IF oR_E8xyyy.uiResult = 0 THEN
14      xEnable          := FALSE;
15      Values[1]        := oR_E8xyyy.lrE8_n0;
16      Values[2]        := oR_E8xyyy.lrE8_n1;
17      Values[3]        := oR_E8xyyy.lrE8_n2;
18      Values[4]        := oR_E8xyyy.lrE8_n3;
19      Values[5]        := oR_E8xyyy.lrE8_n4;
20      Values[6]        := oR_E8xyyy.lrE8_n5;
21      Values[7]        := oR_E8xyyy.lrE8_n6;
22      Values[8]        := oR_E8xyyy.lrE8_n7;
23      Values[9]        := oR_E8xyyy.lrE8_n8;
24      Values[10]       := oR_E8xyyy.lrE8_n9;
25  END_IF

```

### 5.3.15 Reading a program parameter ► R\_E\_Param\_Plus

This request returns the value of a particular E parameter as well as the type of parameter and the possibility of writing it or not.

```

1  PROGRAM R_Plus_031
2  VAR
3      oR_E_Param          : R_E_Param_Plus;
4      Parameter           : UDINT      := 50000;
5      Para_Type           : BYTE;
6      Writable            : BOOL;
7      pP1                 : POINTER TO USINT;
8      Type_1              : USINT;
9      pP8                 : POINTER TO BOOL;
10     Type_8               : BOOL;
11 END_VAR

1
2  oR_E_Param.CNCAdr       := ADR(NCK);
3  oR_E_Param.byChannel    := 0;
4  oR_E_Param.udE_number   := Parameter;
5
6  oR_E_Param(enable := xEnable);
7
8  IF oR_E_Param.uiResult = 0 THEN
9      xEnable             := FALSE;
10     Para_Type            := oR_E_Param.dataType;
11     Writable              := oR_E_Param.writeable;
12
13     CASE Para_Type OF
14         // Example for parameters type 1 and 8
15         1: pP1             := oR_E_Param.ptValue;
16            Type_1          := pP1^;
17
18         8: pP8             := oR_E_Param.ptValue;
19            Type_8          := pP8^;
20     ELSE
21         ;                 // To complete for other types
22     END_CASE
23 END_IF
24
```

### 5.3.16 Reading the Interpolation data ► [R\\_Interpo\\_Plus](#)

This request returns current interpolation data of a particular channel. The data are curvilinear distance to go, the feedrate as well as the override value.

```

1  PROGRAM R_Plus_040
2  VAR
3      oR_Interpo          : R_Interpo_Plus;
4      lr_Distance         : LREAL;
5      lr_Feed             : LREAL;
6      lr_Override         : LREAL;
7  END_VAR
8
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24
25
26
27
28
29
30
31
32
33
34
35
36
37
38
39
40
41
42
43
44
45
46
47
48
49
50
51
52
53
54
55
56
57
58
59
60
61
62
63
64
65
66
67
68
69
70
71
72
73
74
75
76
77
78
79
80
81
82
83
84
85
86
87
88
89
90
91
92
93
94
95
96
97
98
99
100
101
102
103
104
105
106
107
108
109
110
111
112
113
114
115
116
117
118
119
120
121
122
123
124
125
126
127
128
129
130
131
132
133
134
135
136
137
138
139
140
141
142
143
144
145
146
147
148
149
150
151
152
153
154
155
156
157
158
159
160
161
162
163
164
165
166
167
168
169
170
171
172
173
174
175
176
177
178
179
180
181
182
183
184
185
186
187
188
189
190
191
192
193
194
195
196
197
198
199
200
201
202
203
204
205
206
207
208
209
210
211
212
213
214
215
216
217
218
219
220
221
222
223
224
225
226
227
228
229
230
231
232
233
234
235
236
237
238
239
240
241
242
243
244
245
246
247
248
249
250
251
252
253
254
255
256
257
258
259
260
261
262
263
264
265
266
267
268
269
270
271
272
273
274
275
276
277
278
279
280
281
282
283
284
285
286
287
288
289
290
291
292
293
294
295
296
297
298
299
300
301
302
303
304
305
306
307
308
309
310
311
312
313
314
315
316
317
318
319
320
321
322
323
324
325
326
327
328
329
330
331
332
333
334
335
336
337
338
339
340
341
342
343
344
345
346
347
348
349
350
351
352
353
354
355
356
357
358
359
360
361
362
363
364
365
366
367
368
369
370
371
372
373
374
375
376
377
378
379
380
381
382
383
384
385
386
387
388
389
390
391
392
393
394
395
396
397
398
399
400
401
402
403
404
405
406
407
408
409
410
411
412
413
414
415
416
417
418
419
420
421
422
423
424
425
426
427
428
429
430
431
432
433
434
435
436
437
438
439
440
441
442
443
444
445
446
447
448
449
450
451
452
453
454
455
456
457
458
459
460
461
462
463
464
465
466
467
468
469
470
471
472
473
474
475
476
477
478
479
480
481
482
483
484
485
486
487
488
489
490
491
492
493
494
495
496
497
498
499
500
501
502
503
504
505
506
507
508
509
510
511
512
513
514
515
516
517
518
519
520
521
522
523
524
525
526
527
528
529
530
531
532
533
534
535
536
537
538
539
540
541
542
543
544
545
546
547
548
549
550
551
552
553
554
555
556
557
558
559
560
561
562
563
564
565
566
567
568
569
570
571
572
573
574
575
576
577
578
579
580
581
582
583
584
585
586
587
588
589
590
591
592
593
594
595
596
597
598
599
600
601
602
603
604
605
606
607
608
609
610
611
612
613
614
615
616
617
618
619
620
621
622
623
624
625
626
627
628
629
630
631
632
633
634
635
636
637
638
639
640
641
642
643
644
645
646
647
648
649
650
651
652
653
654
655
656
657
658
659
660
661
662
663
664
665
666
667
668
669
670
671
672
673
674
675
676
677
678
679
680
681
682
683
684
685
686
687
688
689
690
691
692
693
694
695
696
697
698
699
700
701
702
703
704
705
706
707
708
709
710
711
712
713
714
715
716
717
718
719
720
721
722
723
724
725
726
727
728
729
730
731
732
733
734
735
736
737
738
739
740
741
742
743
744
745
746
747
748
749
750
751
752
753
754
755
756
757
758
759
760
761
762
763
764
765
766
767
768
769
770
771
772
773
774
775
776
777
778
779
780
781
782
783
784
785
786
787
788
789
790
791
792
793
794
795
796
797
798
799
800
801
802
803
804
805
806
807
808
809
810
811
812
813
814
815
816
817
818
819
820
821
822
823
824
825
826
827
828
829
830
831
832
833
834
835
836
837
838
839
840
841
842
843
844
845
846
847
848
849
850
851
852
853
854
855
856
857
858
859
860
861
862
863
864
865
866
867
868
869
870
871
872
873
874
875
876
877
878
879
880
881
882
883
884
885
886
887
888
889
890
891
892
893
894
895
896
897
898
899
900
901
902
903
904
905
906
907
908
909
910
911
912
913
914
915
916
917
918
919
920
921
922
923
924
925
926
927
928
929
930
931
932
933
934
935
936
937
938
939
940
941
942
943
944
945
946
947
948
949
950
951
952
953
954
955
956
957
958
959
960
961
962
963
964
965
966
967
968
969
970
971
972
973
974
975
976
977
978
979
980
981
982
983
984
985
986
987
988
989
990
991
992
993
994
995
996
997
998
999
1000
1001
1002
1003
1004
1005
1006
1007
1008
1009
1010
1011
1012
1013
1014
1015
1016
1017
1018
1019
1020
1021
1022
1023
1024
1025
1026
1027
1028
1029
1030
1031
1032
1033
1034
1035
1036
1037
1038
1039
1040
1041
1042
1043
1044
1045
1046
1047
1048
1049
1050
1051
1052
1053
1054
1055
1056
1057
1058
1059
1060
1061
1062
1063
1064
1065
1066
1067
1068
1069
1070
1071
1072
1073
1074
1075
1076
1077
1078
1079
1080
1081
1082
1083
1084
1085
1086
1087
1088
1089
1090
1091
1092
1093
1094
1095
1096
1097
1098
1099
1100
1101
1102
1103
1104
1105
1106
1107
1108
1109
1110
1111
1112
1113
1114
1115
1116
1117
1118
1119
1120
1121
1122
1123
1124
1125
1126
1127
1128
1129
1130
1131
1132
1133
1134
1135
1136
1137
1138
1139
1140
1141
1142
1143
1144
1145
1146
1147
1148
1149
1150
1151
1152
1153
1154
1155
1156
1157
1158
1159
1160
1161
1162
1163
1164
1165
1166
1167
1168
1169
1170
1171
1172
1173
1174
1175
1176
1177
1178
1179
1180
1181
1182
1183
1184
1185
1186
1187
1188
1189
1190
1191
1192
1193
1194
1195
1196
1197
1198
1199
1200
1201
1202
1203
1204
1205
1206
1207
1208
1209
1210
1211
1212
1213
1214
1215
1216
1217
1218
1219
1220
1221
1222
1223
1224
1225
1226
1227
1228
1229
1230
1231
1232
1233
1234
1235
1236
1237
1238
1239
1240
1241
1242
1243
1244
1245
1246
1247
1248
1249
1250
1251
1252
1253
1254
1255
1256
1257
1258
1259
1260
1261
1262
1263
1264
1265
1266
1267
1268
1269
1270
1271
1272
1273
1274
1275
1276
1277
1278
1279
1280
1281
1282
1283
1284
1285
1286
1287
1288
1289
1290
1291
1292
1293
1294
1295
1296
1297
1298
1299
1300
1301
1302
1303
1304
1305
1306
1307
1308
1309
1310
1311
1312
1313
1314
1315
1316
1317
1318
1319
1320
1321
1322
1323
1324
1325
1326
1327
1328
1329
1330
1331
1332
1333
1334
1335
1336
1337
1338
1339
1340
1341
1342
1343
1344
1345
1346
1347
1348
1349
1350
1351
1352
1353
1354
1355
1356
1357
1358
1359
1360
1361
1362
1363
1364
1365
1366
1367
1368
1369
1370
1371
1372
1373
1374
1375
1376
1377
1378
1379
1380
1381
1382
1383
1384
1385
1386
1387
1388
1389
1390
1391
1392
1393
1394
1395
1396
1397
1398
1399
1400
1401
1402
1403
1404
1405
1406
1407
1408
1409
1410
1411
1412
1413
1414
1415
1416
1417
1418
1419
1420
1421
1422
1423
1424
1425
1426
1427
1428
1429
1430
1431
1432
1433
1434
1435
1436
1437
1438
1439
1440
1441
1442
1443
1444
1445
1446
1447
1448
1449
1450
1451
1452
1453
1454
1455
1456
1457
1458
1459
1460
1461
1462
1463
1464
1465
1466
1467
1468
1469
1470
1471
1472
1473
1474
1475
1476
1477
1478
1479
1480
1481
1482
1483
1484
1485
1486
1487
1488
1489
1490
1491
1492
1493
1494
1495
1496
1497
1498
1499
1500
1501
1502
1503
1504
1505
1506
1507
1508
1509
1510
1511
1512
1513
1514
1515
1516
1517
1518
1519
1520
1521
1522
1523
1524
1525
1526
1527
1528
1529
1530
1531
1532
1533
1534
1535
1536
1537
1538
1539
1540
1541
1542
1543
1544
1545
1546
1547
1548
1549
1550
1551
1552
1553
1554
1555
1556
1557
1558
1559
1560
1561
1562
1563
1564
1565
1566
1567
1568
1569
1570
1571
1572
1573
1574
1575
1576
1577
1578
1579
1580
1581
1582
1583
1584
1585
1586
1587
1588
1589
1590
1591
1592
1593
1594
1595
1596
1597
1598
1599
1600
1601
1602
1603
1604
1605
1606
1607
1608
1609
1610
1611
1612
1613
1614
1615
1616
1617
1618
1619
1620
1621
1622
1623
1624
1625
1626
1627
1628
1629
1630
1631
1632
1633
1634
1635
1636
1637
1638
1639
1640
1641
1642
1643
1644
1645
1646
1647
1648
1649
1650
1651
1652
1653
1654
1655
1656
1657
1658
1659
1660
1661
1662
1663
1664
1665
1666
1667
1668
1669
1670
1671
1672
1673
1674
1675
1676
1677
1678
1679
1680
1681
1682
1683
1684
1685
1686
1687
1688
1689
1690
1691
1692
1693
1694
1695
1696
1697
1698
1699
1700
1701
1702
1703
1704
1705
1706
1707
1708
1709
1710
1711
1712
1713
1714
1715
1716
1717
1718
1719
1720
1721
1722
1723
1724
1725
1726
1727
1728
1729
1730
1731
1732
1733
1734
1735
1736
1737
1738
1739
1740
1741
1742
1743
1744
1745
1746
1747
1748
1749
1750
1751
1752
1753
1754
1755
1756
1757
1758
1759
1760
1761
1762
1763
1764
1765
1766
1767
1768
1769
1770
1771
1772
1773
1774
1775
1776
1777
1778
1779
1780
1781
1782
1783
1784
1785
1786
1787
1788
1789
1790
1791
1792
1793
1794
1795
1796
1797
1798
1799
1800
1801
1802
1803
1804
1805
1806
1807
1808
1809
1810
1811
1812
1813
1814
1815
1816
1817
1818
1819
1820
1821
1822
1823
1824
1825
1826
1827
1828
1829
1830
1831
1832
1833
1834
1835
1836
1837
1838
1839
1840
1841
1842
1843
1844
1845
1846
1847
1848
1849
1850
1851
1852
1853
1854
1855
1856
1857
1858
1859
1860
1861
1862
1863
1864
1865
1866
1867
1868
1869
1870
1871
1872
1873
1874
1875
1876
1877
1878
1879
1880
1881
1882
1883
1884
1885
1886
1887
1888
1889
1890
1891
1892
1893
1894
1895
1896
1897
1898
1899
1900
1901
1902
1903
1904
1905
1906
1907
1908
1909
1910
1911
1912
1913
1914
1915
1916
1917
1918
1919
1920
1921
1922
1923
1924
1925
1926
1927
1928
1929
1930
1931
1932
1933
1934
1935
1936
1937
1938
1939
1940
1941
1942
1943
1944
1945
1946
1947
1948
1949
1950
1951
1952
1953
1954
1955
1956
1957
1958
1959
1960
1961
1962
1963
1964
1965
1966
1967
1968
1969
1970
1971
1972
1973
1974
1975
1976
1977
1978
1979
1980
1981
1982
1983
1984
1985
1986
1987
1988
1989
1990
1991
1992
1993
1994
1995
1996
1997
1998
1999
2000
2001
2002
2003
2004
2005
2006
2007
2008
2009
2010
2011
2012
2013
2014
2015
2016
2017
2018
2019
2020
2021
2022
2023
2024
2025
2026
2027
2028
2029
2030
2031
2032
2033
2034
2035
2036
2037
2038
2039
2040
2041
2042
2043
2044
2045
2046
2047
2048
2049
2050
2051
2052
2053
2054
2055
2056
2057
2058
2059
2060
2061
2062
2063
2064
2065
2066
2067
2068
2069
2070
2071
2072
2073
2074
2075
2076
2077
2078
2079
2080
2081
2082
2083
2084
2085
2086
2087
2088
2089
2090
2091
2092
2093
2094
2095
2096
2097
2098
2099
2100
2101
2102
2103
2104
2105
2106
2107
2108
2109
2110
2111
2112
2113
2114
2115
2116
2117
2118
2119
2120
2121
2122
2123
2124
2125
2126
2127
2128
2129
2130
2131
2132
2133
2134
2135
2136
2137
2138
2139
2140
2141
2142
2143
2144
2145
2146
2147
2148
2149
2150
2151
2152
2153
2154
2155
2156
2157
2158
2159
2160
2161
2162
2163
2164
2165
2166
2167
2168
2169
2170
2171
2172
2173
2174
2175
2176
2177
2178
2179
2180
2181
2182
2183
2184
2185
2186
2187
2188
2189
2190
2191
2192
2193
2194
2195
2196
2197
2198
2199
2200
2201
2202
2203
2204
2205
2206
2207
2208
2209
2210
2211
2212
2213
2214
2215
2216
2217
2218
2219
2220
2221
2222
2223
2224
2225
2226
2227
2228
2229
2230
2231
2232
2233
2234
2235
2236
2237
2238
2239
2240
2241
2242
2243
2244
2245
2246
2247
2248
2249
2250
2251
2252
2253
2254
2255
2256
2257
2258
2259
2260
2261
2262
2263
2264
2265
2266
2267
2268
2269
2270
2271
2272
2273
2274
2275
2276
2277
2278
2279
2280
2281
2282
2283
2284
2285
2286
2287
2288
2289
2290
2291
2292
2293
2294
2295
2296
2297
2298
2299
2300
2301
2302
2303
2304
2305
2306
2307
2308
2309
2310
2311
2312
2313
2314
2315
2316
2317
2318
2319
2320
2321
2322
2323
2324
2325
2326
2327
2328
2329
2330
2331
2332
2333
2334
2335
2336
2337
2338
2339
2340
2341
2342
2343
2344
2345
2346
2347
2348
2349
2350
2351
2352
2353
2354
2355
2356
2357
2358
2359
2360
2361
2362
2363
2364
2365
2366
2367
2368
2369
2370
2371
2372
2373
2374
2375
2376
2377
2378
2379
2380
2381
2382
2383
2384
2385
2386
2387
2388
2389
2390
2391
2392
2393
2394
2395
2396
2397
2398
2399
2400
2401
2402
2403
2404
2405
2406
2407
2408
2409
2410
2411
2412
2413
2414
2415
2416
2417
2418
2419
2420
2421
2422
2423
2424
2425
2426
2427
2428
2429
2430
2431
2432
2433
2434
2435
2436
2437
2438
2439
2440
2441
2442
2443
2444
2445
2446
2447
2448
2449
2450
2451
2452
2453
2454
2455
2456
2457
2458
2459
2460
2461
2462
2463
2464
2465
2466
2467
2468
2469
2470
2471
2472
2473
2474
2475
2476
2477
2478
2479
2480
2481
2482
2483
2484
2485
2486
2487
2488
2489
2490
2491
2492
2493
2494
2495
2496
2497
2498
2499
2500
2501
2502
2503
2504
2505
2506
2507
2508
2509
2510
2511
2512
2513
2514
2515
2516
2517
2518
2519
2520
2521
2522
2523
2524
2525
2526
2527
2528
2529
2530
2531
2532
2533
2534
2535
2536
2537
2538
2539
2540
2541
2542
2543
2544
2545
2546
2547
2548
2549
2550
2551
2552
2553
2554
2555
2556
2557
2558
2559
2560
2561
2562
2563
2564
2565
2566
2567
2568
2569
2570
2571
2572
2573
2574
2575
2576
2577
2578
2579
2580
2581
2582
2583
2584
2585
2586
2587
2588
2589
2590
2591
2592
2593
2594
2595
2596
2597
2598
2599
2600
2601
2602
2603
2604
2605
2606
2607
2608
2609
2610
2611
2612
2613
2614
2615
2616
2617
2618
2619
2620
2621
2622
2623
2624
2625
2626
2627
2628
2629
2630
2631
2632
2633
2634
2635
2636
2637
2638
2639
2640
2641
2642
2643
2644
2645
2646
2647
2648
2649

```

### 5.3.17 Reading machine origin values ► [R\\_Mach\\_Orig\\_Plus](#)

This request returns the machine origin offset values (equivalent to parameter P16) according to the axis physical address. The values are returned for three consecutive addresses.

```
1  PROGRAM R_Plus_041
2  VAR
3      oR_Mach_Origin          : R_Mach_Orig_Plus;
4      lr_Axe1                 : LREAL;
5      lr_Axe2                 : LREAL;
6      lr_axe3                 : LREAL;
7  END_VAR
8
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24
25
26
27
28
29
30
31
32
33
34
35
36
37
38
39
40
41
42
43
44
45
46
47
48
49
50
51
52
53
54
55
56
57
58
59
60
61
62
63
64
65
66
67
68
69
70
71
72
73
74
75
76
77
78
79
80
81
82
83
84
85
86
87
88
89
90
91
92
93
94
95
96
97
98
99
100
101
102
103
104
105
106
107
108
109
110
111
112
113
114
115
116
117
118
119
120
121
122
123
124
125
126
127
128
129
130
131
132
133
134
135
136
137
138
139
140
141
142
143
144
145
146
147
148
149
150
151
152
153
154
155
156
157
158
159
160
161
162
163
164
165
166
167
168
169
170
171
172
173
174
175
176
177
178
179
180
181
182
183
184
185
186
187
188
189
190
191
192
193
194
195
196
197
198
199
200
201
202
203
204
205
206
207
208
209
210
211
212
213
214
215
216
217
218
219
220
221
222
223
224
225
226
227
228
229
230
231
232
233
234
235
236
237
238
239
240
241
242
243
244
245
246
247
248
249
250
251
252
253
254
255
256
257
258
259
260
261
262
263
264
265
266
267
268
269
270
271
272
273
274
275
276
277
278
279
280
281
282
283
284
285
286
287
288
289
290
291
292
293
294
295
296
297
298
299
300
301
302
303
304
305
306
307
308
309
310
311
312
313
314
315
316
317
318
319
320
321
322
323
324
325
326
327
328
329
330
331
332
333
334
335
336
337
338
339
340
341
342
343
344
345
346
347
348
349
350
351
352
353
354
355
356
357
358
359
360
361
362
363
364
365
366
367
368
369
370
371
372
373
374
375
376
377
378
379
380
381
382
383
384
385
386
387
388
389
390
391
392
393
394
395
396
397
398
399
400
401
402
403
404
405
406
407
408
409
410
411
412
413
414
415
416
417
418
419
420
421
422
423
424
425
426
427
428
429
430
431
432
433
434
435
436
437
438
439
440
441
442
443
444
445
446
447
448
449
450
451
452
453
454
455
456
457
458
459
460
461
462
463
464
465
466
467
468
469
470
471
472
473
474
475
476
477
478
479
480
481
482
483
484
485
486
487
488
489
490
491
492
493
494
495
496
497
498
499
500
501
502
503
504
505
506
507
508
509
510
511
512
513
514
515
516
517
518
519
520
521
522
523
524
525
526
527
528
529
530
531
532
533
534
535
536
537
538
539
540
541
542
543
544
545
546
547
548
549
550
551
552
553
554
555
556
557
558
559
560
561
562
563
564
565
566
567
568
569
570
571
572
573
574
575
576
577
578
579
580
581
582
583
584
585
586
587
588
589
590
591
592
593
594
595
596
597
598
599
600
601
602
603
604
605
606
607
608
609
610
611
612
613
614
615
616
617
618
619
620
621
622
623
624
625
626
627
628
629
630
631
632
633
634
635
636
637
638
639
640
641
642
643
644
645
646
647
648
649
650
651
652
653
654
655
656
657
658
659
660
661
662
663
664
665
666
667
668
669
670
671
672
673
674
675
676
677
678
679
680
681
682
683
684
685
686
687
688
689
690
691
692
693
694
695
696
697
698
699
700
701
702
703
704
705
706
707
708
709
710
711
712
713
714
715
716
717
718
719
720
721
722
723
724
725
726
727
728
729
730
731
732
733
734
735
736
737
738
739
740
741
742
743
744
745
746
747
748
749
750
751
752
753
754
755
756
757
758
759
760
761
762
763
764
765
766
767
768
769
770
771
772
773
774
775
776
777
778
779
780
781
782
783
784
785
786
787
788
789
790
791
792
793
794
795
796
797
798
799
800
801
802
803
804
805
806
807
808
809
810
811
812
813
814
815
816
817
818
819
820
821
822
823
824
825
826
827
828
829
830
831
832
833
834
835
836
837
838
839
840
841
842
843
844
845
846
847
848
849
850
851
852
853
854
855
856
857
858
859
860
861
862
863
864
865
866
867
868
869
870
871
872
873
874
875
876
877
878
879
880
881
882
883
884
885
886
887
888
889
890
891
892
893
894
895
896
897
898
899
900
901
902
903
904
905
906
907
908
909
910
911
912
913
914
915
916
917
918
919
920
921
922
923
924
925
926
927
928
929
930
931
932
933
934
935
936
937
938
939
940
941
942
943
944
945
946
947
948
949
950
951
952
953
954
955
956
957
958
959
960
961
962
963
964
965
966
967
968
969
970
971
972
973
974
975
976
977
978
979
980
981
982
983
984
985
986
987
988
989
990
991
992
993
994
995
996
997
998
999
1000
```

### 5.3.18 Reading Maximum Dynamic Travel values ▶ R\_Max\_Dyn\_Travel\_Plus

This request returns the maximum dynamic travels of the axes in a particular channel. The values are returned in reference to the axes names.

```

1  PROGRAM R_Plus_042
2  VAR
3      oR_Max_Dyn_Travel          : R_Max_Dyn_Travel_Plus;
4      lr_MaxDyn_A                : LREAL;
5      lr_MaxDyn_B                : LREAL;
6      lr_MaxDyn_C                : LREAL;
7      lr_MaxDyn_U                : LREAL;
8      lr_MaxDyn_V                : LREAL;
9      lr_MaxDyn_W                : LREAL;
10     lr_MaxDyn_X                : LREAL;
11     lr_MaxDyn_Y                : LREAL;
12     lr_MaxDyn_Z                : LREAL;
13 END_VAR
14
15
16 oR_Max_Dyn_Travel.CNCAdr        := ADR(NCK);
17 oR_Max_Dyn_Travel.byChannel     := 0;
18
19 oR_Max_Dyn_Travel(enable := xEnable);
20
21 IF oR_Max_Dyn_Travel.uiResult = 0 THEN
22     xEnable                     := FALSE;
23     lr_MaxDyn_A                 := oR_Max_Dyn_Travel.lrA_Axis;
24     lr_MaxDyn_B                 := oR_Max_Dyn_Travel.lrB_Axis;
25     lr_MaxDyn_C                 := oR_Max_Dyn_Travel.lrC_Axis;;
26     lr_MaxDyn_U                 := oR_Max_Dyn_Travel.lrU_Axis;
27     lr_MaxDyn_V                 := oR_Max_Dyn_Travel.lrV_Axis;;
28     lr_MaxDyn_W                 := oR_Max_Dyn_Travel.lrW_Axis;;
29     lr_MaxDyn_X                 := oR_Max_Dyn_Travel.lrX_Axis;;
30     lr_MaxDyn_Y                 := oR_Max_Dyn_Travel.lrY_Axis;;
31     lr_MaxDyn_Z                 := oR_Max_Dyn_Travel.lrZ_Axis;;
32 END_IF
33

```

### 5.3.19 Reading Minimum Dynamic Travel values [▶ R\\_Min\\_Dyn\\_Travel\\_Plus](#)

This request returns the minimum dynamic travels of the axes in a particular channel. The values are returned in reference to the axes names.

```
R_Plus_043 x
1 PROGRAM R_Plus_043
2 VAR
3     oR_Min_Dyn_Travel          : R_Min_Dyn_Travel_Plus;
4     lr_MinDyn_A                : LREAL;
5     lr_MinDyn_B                : LREAL;
6     lr_MinDyn_C                : LREAL;
7     lr_MinDyn_U                : LREAL;
8     lr_MinDyn_V                : LREAL;
9     lr_MinDyn_W                : LREAL;
10    lr_MinDyn_X                : LREAL;
11    lr_MinDyn_Y                : LREAL;
12    lr_MinDyn_Z                : LREAL;
13 END_VAR
14
15 oR_Min_Dyn_Travel.CNCAdr      := ADR(NCK);
16 oR_Min_Dyn_Travel.byChannel   := 0;
17
18 oR_Min_Dyn_Travel(enable := xEnable);
19
20 IF oR_Min_Dyn_Travel.uiResult = 0 THEN
21     xEnable := FALSE;
22     lr_MinDyn_A := oR_Min_Dyn_Travel.lrA_Axis;
23     lr_MinDyn_B := oR_Min_Dyn_Travel.lrB_Axis;
24     lr_MinDyn_C := oR_Min_Dyn_Travel.lrC_Axis;;
25     lr_MinDyn_U := oR_Min_Dyn_Travel.lrU_Axis;
26     lr_MinDyn_V := oR_Min_Dyn_Travel.lrV_Axis;;
27     lr_MinDyn_W := oR_Min_Dyn_Travel.lrW_Axis;;
28     lr_MinDyn_X := oR_Min_Dyn_Travel.lrX_Axis;;
29     lr_MinDyn_Y := oR_Min_Dyn_Travel.lrY_Axis;;
30     lr_MinDyn_Z := oR_Min_Dyn_Travel.lrZ_Axis;;
31 END_IF
32
```

### 5.3.20 Reading Maximum Static Travel values ► [R\\_Max\\_Stat\\_Travel\\_Plus](#)

This request returns the maximum static travels of axes. The values are given for three consecutive physical addresses.

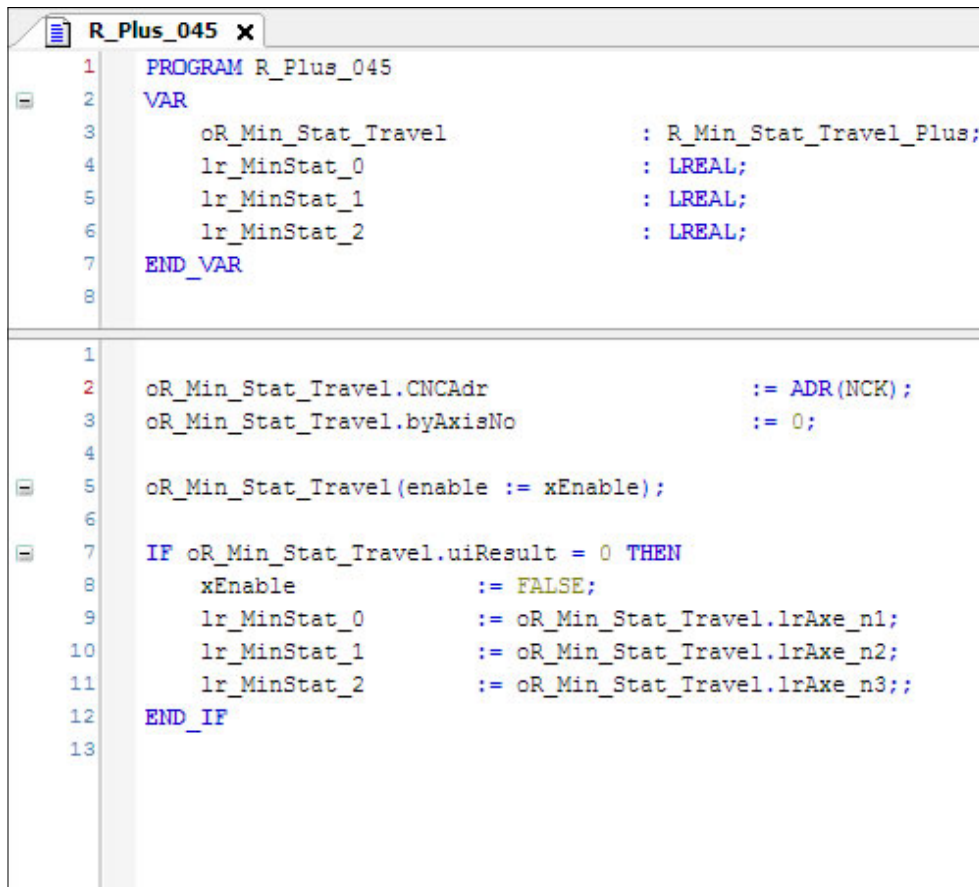
```

1  PROGRAM R_Plus_044
2  VAR
3      oR_Max_Stat_Travel      : R_Max_Stat_Travel_Plus;
4      lr_MaxStat_0            : LREAL;
5      lr_MaxStat_1            : LREAL;
6      lr_MaxStat_2            : LREAL;
7  END_VAR
8
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24
25
26
27
28
29
30
31
32
33
34
35
36
37
38
39
40
41
42
43
44
45
46
47
48
49
50
51
52
53
54
55
56
57
58
59
60
61
62
63
64
65
66
67
68
69
70
71
72
73
74
75
76
77
78
79
80
81
82
83
84
85
86
87
88
89
90
91
92
93
94
95
96
97
98
99
100
101
102
103
104
105
106
107
108
109
110
111
112
113
114
115
116
117
118
119
120
121
122
123
124
125
126
127
128
129
130
131
132
133
134
135
136
137
138
139
140
141
142
143
144
145
146
147
148
149
150
151
152
153
154
155
156
157
158
159
160
161
162
163
164
165
166
167
168
169
170
171
172
173
174
175
176
177
178
179
180
181
182
183
184
185
186
187
188
189
190
191
192
193
194
195
196
197
198
199
200
201
202
203
204
205
206
207
208
209
210
211
212
213
214
215
216
217
218
219
220
221
222
223
224
225
226
227
228
229
230
231
232
233
234
235
236
237
238
239
240
241
242
243
244
245
246
247
248
249
250
251
252
253
254
255
256
257
258
259
260
261
262
263
264
265
266
267
268
269
270
271
272
273
274
275
276
277
278
279
280
281
282
283
284
285
286
287
288
289
290
291
292
293
294
295
296
297
298
299
300
301
302
303
304
305
306
307
308
309
310
311
312
313
314
315
316
317
318
319
320
321
322
323
324
325
326
327
328
329
330
331
332
333
334
335
336
337
338
339
340
341
342
343
344
345
346
347
348
349
350
351
352
353
354
355
356
357
358
359
360
361
362
363
364
365
366
367
368
369
370
371
372
373
374
375
376
377
378
379
380
381
382
383
384
385
386
387
388
389
390
391
392
393
394
395
396
397
398
399
400
401
402
403
404
405
406
407
408
409
410
411
412
413
414
415
416
417
418
419
420
421
422
423
424
425
426
427
428
429
430
431
432
433
434
435
436
437
438
439
440
441
442
443
444
445
446
447
448
449
450
451
452
453
454
455
456
457
458
459
460
461
462
463
464
465
466
467
468
469
470
471
472
473
474
475
476
477
478
479
480
481
482
483
484
485
486
487
488
489
490
491
492
493
494
495
496
497
498
499
500
501
502
503
504
505
506
507
508
509
510
511
512
513
514
515
516
517
518
519
520
521
522
523
524
525
526
527
528
529
530
531
532
533
534
535
536
537
538
539
540
541
542
543
544
545
546
547
548
549
550
551
552
553
554
555
556
557
558
559
560
561
562
563
564
565
566
567
568
569
570
571
572
573
574
575
576
577
578
579
580
581
582
583
584
585
586
587
588
589
590
591
592
593
594
595
596
597
598
599
600
601
602
603
604
605
606
607
608
609
610
611
612
613
614
615
616
617
618
619
620
621
622
623
624
625
626
627
628
629
630
631
632
633
634
635
636
637
638
639
640
641
642
643
644
645
646
647
648
649
650
651
652
653
654
655
656
657
658
659
660
661
662
663
664
665
666
667
668
669
670
671
672
673
674
675
676
677
678
679
680
681
682
683
684
685
686
687
688
689
690
691
692
693
694
695
696
697
698
699
700
701
702
703
704
705
706
707
708
709
710
711
712
713
714
715
716
717
718
719
720
721
722
723
724
725
726
727
728
729
730
731
732
733
734
735
736
737
738
739
740
741
742
743
744
745
746
747
748
749
750
751
752
753
754
755
756
757
758
759
760
761
762
763
764
765
766
767
768
769
770
771
772
773
774
775
776
777
778
779
780
781
782
783
784
785
786
787
788
789
790
791
792
793
794
795
796
797
798
799
800
801
802
803
804
805
806
807
808
809
810
811
812
813
814
815
816
817
818
819
820
821
822
823
824
825
826
827
828
829
830
831
832
833
834
835
836
837
838
839
840
841
842
843
844
845
846
847
848
849
850
851
852
853
854
855
856
857
858
859
860
861
862
863
864
865
866
867
868
869
870
871
872
873
874
875
876
877
878
879
880
881
882
883
884
885
886
887
888
889
890
891
892
893
894
895
896
897
898
899
900
901
902
903
904
905
906
907
908
909
910
911
912
913
914
915
916
917
918
919
920
921
922
923
924
925
926
927
928
929
930
931
932
933
934
935
936
937
938
939
940
941
942
943
944
945
946
947
948
949
950
951
952
953
954
955
956
957
958
959
960
961
962
963
964
965
966
967
968
969
970
971
972
973
974
975
976
977
978
979
980
981
982
983
984
985
986
987
988
989
990
991
992
993
994
995
996
997
998
999
1000

```

### 5.3.21 Reading Minimum Static Travel values ► [R\\_Min\\_Stat\\_Travel\\_Plus](#)

This request returns the maximum static travels of axes. The values are given for three consecutive physical addresses.



```
1  PROGRAM R_Min_Stat_Travel_Plus
2  VAR
3      oR_Min_Stat_Travel          : R_Min_Stat_Travel_Plus;
4      lr_MinStat_0                : LREAL;
5      lr_MinStat_1                : LREAL;
6      lr_MinStat_2                : LREAL;
7  END_VAR
8
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24
25
26
27
28
29
30
31
32
33
34
35
36
37
38
39
40
41
42
43
44
45
46
47
48
49
50
51
52
53
54
55
56
57
58
59
60
61
62
63
64
65
66
67
68
69
70
71
72
73
74
75
76
77
78
79
80
81
82
83
84
85
86
87
88
89
90
91
92
93
94
95
96
97
98
99
100
101
102
103
104
105
106
107
108
109
110
111
112
113
114
115
116
117
118
119
120
121
122
123
124
125
126
127
128
129
130
131
132
133
134
135
136
137
138
139
140
141
142
143
144
145
146
147
148
149
150
151
152
153
154
155
156
157
158
159
160
161
162
163
164
165
166
167
168
169
170
171
172
173
174
175
176
177
178
179
180
181
182
183
184
185
186
187
188
189
190
191
192
193
194
195
196
197
198
199
200
201
202
203
204
205
206
207
208
209
210
211
212
213
214
215
216
217
218
219
220
221
222
223
224
225
226
227
228
229
230
231
232
233
234
235
236
237
238
239
240
241
242
243
244
245
246
247
248
249
250
251
252
253
254
255
256
257
258
259
260
261
262
263
264
265
266
267
268
269
270
271
272
273
274
275
276
277
278
279
280
281
282
283
284
285
286
287
288
289
290
291
292
293
294
295
296
297
298
299
300
301
302
303
304
305
306
307
308
309
310
311
312
313
314
315
316
317
318
319
320
321
322
323
324
325
326
327
328
329
330
331
332
333
334
335
336
337
338
339
340
341
342
343
344
345
346
347
348
349
350
351
352
353
354
355
356
357
358
359
360
361
362
363
364
365
366
367
368
369
370
371
372
373
374
375
376
377
378
379
380
381
382
383
384
385
386
387
388
389
390
391
392
393
394
395
396
397
398
399
400
401
402
403
404
405
406
407
408
409
410
411
412
413
414
415
416
417
418
419
420
421
422
423
424
425
426
427
428
429
430
431
432
433
434
435
436
437
438
439
440
441
442
443
444
445
446
447
448
449
450
451
452
453
454
455
456
457
458
459
460
461
462
463
464
465
466
467
468
469
470
471
472
473
474
475
476
477
478
479
480
481
482
483
484
485
486
487
488
489
490
491
492
493
494
495
496
497
498
499
500
501
502
503
504
505
506
507
508
509
510
511
512
513
514
515
516
517
518
519
520
521
522
523
524
525
526
527
528
529
530
531
532
533
534
535
536
537
538
539
540
541
542
543
544
545
546
547
548
549
550
551
552
553
554
555
556
557
558
559
560
561
562
563
564
565
566
567
568
569
570
571
572
573
574
575
576
577
578
579
580
581
582
583
584
585
586
587
588
589
590
591
592
593
594
595
596
597
598
599
600
601
602
603
604
605
606
607
608
609
610
611
612
613
614
615
616
617
618
619
620
621
622
623
624
625
626
627
628
629
630
631
632
633
634
635
636
637
638
639
640
641
642
643
644
645
646
647
648
649
650
651
652
653
654
655
656
657
658
659
660
661
662
663
664
665
666
667
668
669
670
671
672
673
674
675
676
677
678
679
680
681
682
683
684
685
686
687
688
689
690
691
692
693
694
695
696
697
698
699
700
701
702
703
704
705
706
707
708
709
710
711
712
713
714
715
716
717
718
719
720
721
722
723
724
725
726
727
728
729
730
731
732
733
734
735
736
737
738
739
740
741
742
743
744
745
746
747
748
749
750
751
752
753
754
755
756
757
758
759
760
761
762
763
764
765
766
767
768
769
770
771
772
773
774
775
776
777
778
779
780
781
782
783
784
785
786
787
788
789
790
791
792
793
794
795
796
797
798
799
800
801
802
803
804
805
806
807
808
809
810
811
812
813
814
815
816
817
818
819
820
821
822
823
824
825
826
827
828
829
830
831
832
833
834
835
836
837
838
839
840
841
842
843
844
845
846
847
848
849
850
851
852
853
854
855
856
857
858
859
860
861
862
863
864
865
866
867
868
869
870
871
872
873
874
875
876
877
878
879
880
881
882
883
884
885
886
887
888
889
890
891
892
893
894
895
896
897
898
899
900
901
902
903
904
905
906
907
908
909
910
911
912
913
914
915
916
917
918
919
920
921
922
923
924
925
926
927
928
929
930
931
932
933
934
935
936
937
938
939
940
941
942
943
944
945
946
947
948
949
950
951
952
953
954
955
956
957
958
959
960
961
962
963
964
965
966
967
968
969
970
971
972
973
974
975
976
977
978
979
980
981
982
983
984
985
986
987
988
989
990
991
992
993
994
995
996
997
998
999
1000
```

### 5.3.22 Reading NCK detected error ► [R\\_Nck\\_Error\\_Plus](#)

This request returns the information about an NCK detected error. The data are the block number and line number where the error has occurred as well as the error number and error extension number.

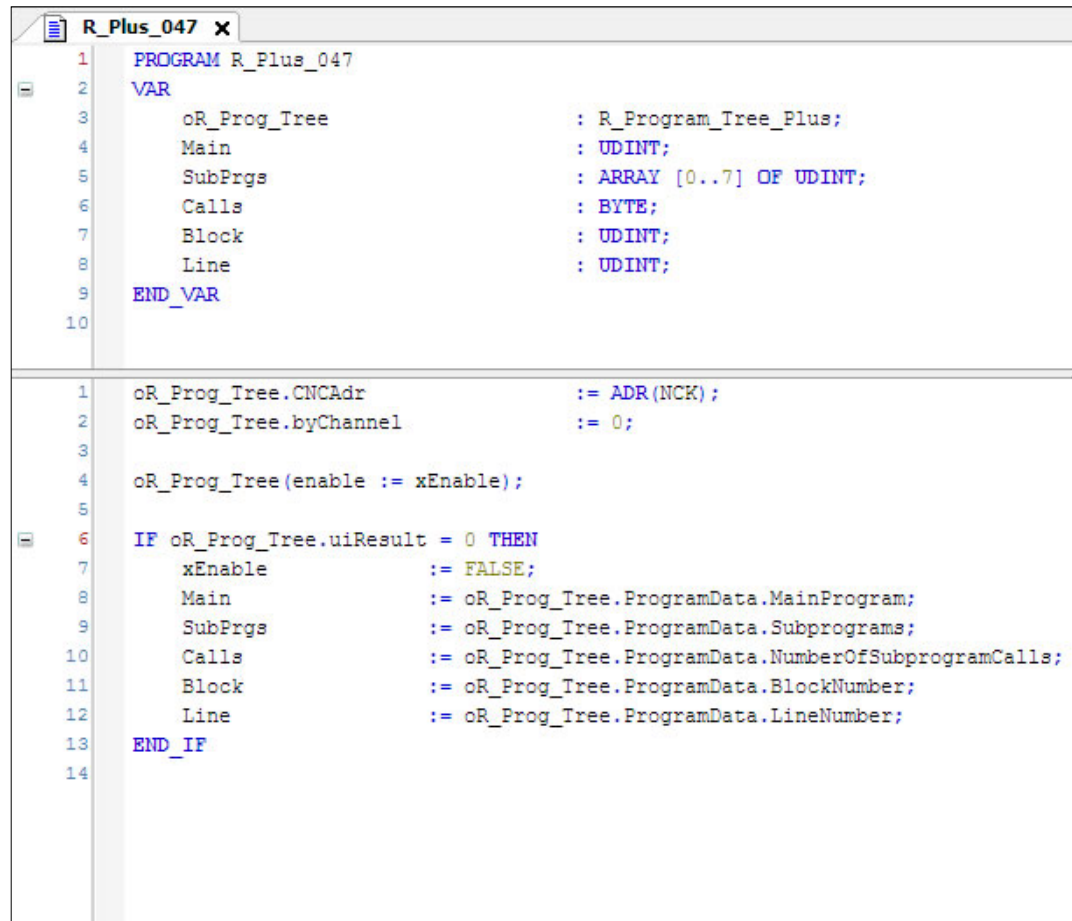
```

1  PROGRAM R_Plus_046
2  VAR
3      oR_NckError          : R_Nck_Error_Plus;
4      ud_Block             : UDINT;
5      ud_Line              : UDINT;
6      wo_Error             : WORD;
7      wo_Extension         : WORD;
8  END_VAR
9
10 oR_NckError.CNCAdr        := ADR(NCK);
11 oR_NckError.byChannel     := 0;
12
13 oR_NckError(enable := xEnable);
14
15 IF oR_NckError.uiResult = 0 THEN
16     xEnable                := FALSE;
17     ud_Block               := oR_NckError.udBlockNumber;
18     ud_Line                := oR_NckError.udLineNumber;
19     wo_Error               := oR_NckError.woErrorNumber;
20     wo_Extension           := oR_NckError.woExtendedErrorNumber;
21 END_IF
22

```

### 5.3.23 Reading Part program execution tree ► [R\\_Program\\_Tree\\_Plus](#)

This request returns information about the part program in execution. The information include block number as well as line number, the current subroutine number with number of times it has been called as well as all the chain of calling programs (up to eight nesting level).



```
1 PROGRAM R_Plus_047
2 VAR
3     oR_Prog_Tree           : R_Program_Tree_Plus;
4     Main                   : UDINT;
5     SubPrgs                 : ARRAY [0..7] OF UDINT;
6     Calls                   : BYTE;
7     Block                   : UDINT;
8     Line                    : UDINT;
9 END_VAR
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24
25
26
27
28
29
30
31
32
33
34
35
36
37
38
39
40
41
42
43
44
45
46
47
48
49
50
51
52
53
54
55
56
57
58
59
60
61
62
63
64
65
66
67
68
69
70
71
72
73
74
75
76
77
78
79
80
81
82
83
84
85
86
87
88
89
90
91
92
93
94
95
96
97
98
99
100
101
102
103
104
105
106
107
108
109
110
111
112
113
114
115
116
117
118
119
120
121
122
123
124
125
126
127
128
129
130
131
132
133
134
135
136
137
138
139
140
141
142
143
144
145
146
147
148
149
150
151
152
153
154
155
156
157
158
159
160
161
162
163
164
165
166
167
168
169
170
171
172
173
174
175
176
177
178
179
180
181
182
183
184
185
186
187
188
189
190
191
192
193
194
195
196
197
198
199
200
201
202
203
204
205
206
207
208
209
210
211
212
213
214
215
216
217
218
219
220
221
222
223
224
225
226
227
228
229
230
231
232
233
234
235
236
237
238
239
240
241
242
243
244
245
246
247
248
249
250
251
252
253
254
255
256
257
258
259
260
261
262
263
264
265
266
267
268
269
270
271
272
273
274
275
276
277
278
279
280
281
282
283
284
285
286
287
288
289
290
291
292
293
294
295
296
297
298
299
300
301
302
303
304
305
306
307
308
309
310
311
312
313
314
315
316
317
318
319
320
321
322
323
324
325
326
327
328
329
330
331
332
333
334
335
336
337
338
339
340
341
342
343
344
345
346
347
348
349
350
351
352
353
354
355
356
357
358
359
360
361
362
363
364
365
366
367
368
369
370
371
372
373
374
375
376
377
378
379
380
381
382
383
384
385
386
387
388
389
390
391
392
393
394
395
396
397
398
399
400
401
402
403
404
405
406
407
408
409
410
411
412
413
414
415
416
417
418
419
420
421
422
423
424
425
426
427
428
429
430
431
432
433
434
435
436
437
438
439
440
441
442
443
444
445
446
447
448
449
450
451
452
453
454
455
456
457
458
459
460
461
462
463
464
465
466
467
468
469
470
471
472
473
474
475
476
477
478
479
480
481
482
483
484
485
486
487
488
489
490
491
492
493
494
495
496
497
498
499
500
501
502
503
504
505
506
507
508
509
510
511
512
513
514
515
516
517
518
519
520
521
522
523
524
525
526
527
528
529
530
531
532
533
534
535
536
537
538
539
540
541
542
543
544
545
546
547
548
549
550
551
552
553
554
555
556
557
558
559
560
561
562
563
564
565
566
567
568
569
570
571
572
573
574
575
576
577
578
579
580
581
582
583
584
585
586
587
588
589
590
591
592
593
594
595
596
597
598
599
600
601
602
603
604
605
606
607
608
609
610
611
612
613
614
615
616
617
618
619
620
621
622
623
624
625
626
627
628
629
630
631
632
633
634
635
636
637
638
639
640
641
642
643
644
645
646
647
648
649
650
651
652
653
654
655
656
657
658
659
660
661
662
663
664
665
666
667
668
669
670
671
672
673
674
675
676
677
678
679
680
681
682
683
684
685
686
687
688
689
690
691
692
693
694
695
696
697
698
699
700
701
702
703
704
705
706
707
708
709
710
711
712
713
714
715
716
717
718
719
720
721
722
723
724
725
726
727
728
729
730
731
732
733
734
735
736
737
738
739
740
741
742
743
744
745
746
747
748
749
750
751
752
753
754
755
756
757
758
759
760
761
762
763
764
765
766
767
768
769
770
771
772
773
774
775
776
777
778
779
780
781
782
783
784
785
786
787
788
789
790
791
792
793
794
795
796
797
798
799
800
801
802
803
804
805
806
807
808
809
810
811
812
813
814
815
816
817
818
819
820
821
822
823
824
825
826
827
828
829
830
831
832
833
834
835
836
837
838
839
840
841
842
843
844
845
846
847
848
849
850
851
852
853
854
855
856
857
858
859
860
861
862
863
864
865
866
867
868
869
870
871
872
873
874
875
876
877
878
879
880
881
882
883
884
885
886
887
888
889
890
891
892
893
894
895
896
897
898
899
900
901
902
903
904
905
906
907
908
909
910
911
912
913
914
915
916
917
918
919
920
921
922
923
924
925
926
927
928
929
930
931
932
933
934
935
936
937
938
939
940
941
942
943
944
945
946
947
948
949
950
951
952
953
954
955
956
957
958
959
960
961
962
963
964
965
966
967
968
969
970
971
972
973
974
975
976
977
978
979
980
981
982
983
984
985
986
987
988
989
990
991
992
993
994
995
996
997
998
999
1000
1001
1002
1003
1004
1005
1006
1007
1008
1009
1010
1011
1012
1013
1014
1015
1016
1017
1018
1019
1020
1021
1022
1023
1024
1025
1026
1027
1028
1029
1030
1031
1032
1033
1034
1035
1036
1037
1038
1039
1040
1041
1042
1043
1044
1045
1046
1047
1048
1049
1050
1051
1052
1053
1054
1055
1056
1057
1058
1059
1060
1061
1062
1063
1064
1065
1066
1067
1068
1069
1070
1071
1072
1073
1074
1075
1076
1077
1078
1079
1080
1081
1082
1083
1084
1085
1086
1087
1088
1089
1090
1091
1092
1093
1094
1095
1096
1097
1098
1099
1100
1101
1102
1103
1104
1105
1106
1107
1108
1109
1110
1111
1112
1113
1114
1115
1116
1117
1118
1119
1120
1121
1122
1123
1124
1125
1126
1127
1128
1129
1130
1131
1132
1133
1134
1135
1136
1137
1138
1139
1140
1141
1142
1143
1144
1145
1146
1147
1148
1149
1150
1151
1152
1153
1154
1155
1156
1157
1158
1159
1160
1161
1162
1163
1164
1165
1166
1167
1168
1169
1170
1171
1172
1173
1174
1175
1176
1177
1178
1179
1180
1181
1182
1183
1184
1185
1186
1187
1188
1189
1190
1191
1192
1193
1194
1195
1196
1197
1198
1199
1200
1201
1202
1203
1204
1205
1206
1207
1208
1209
1210
1211
1212
1213
1214
1215
1216
1217
1218
1219
1220
1221
1222
1223
1224
1225
1226
1227
1228
1229
1230
1231
1232
1233
1234
1235
1236
1237
1238
1239
1240
1241
1242
1243
1244
1245
1246
1247
1248
1249
1250
1251
1252
1253
1254
1255
1256
1257
1258
1259
1260
1261
1262
1263
1264
1265
1266
1267
1268
1269
1270
1271
1272
1273
1274
1275
1276
1277
1278
1279
1280
1281
1282
1283
1284
1285
1286
1287
1288
1289
1290
1291
1292
1293
1294
1295
1296
1297
1298
1299
1300
1301
1302
1303
1304
1305
1306
1307
1308
1309
1310
1311
1312
1313
1314
1315
1316
1317
1318
1319
1320
1321
1322
1323
1324
1325
1326
1327
1328
1329
1330
1331
1332
1333
1334
1335
1336
1337
1338
1339
1340
1341
1342
1343
1344
1345
1346
1347
1348
1349
1350
1351
1352
1353
1354
1355
1356
1357
1358
1359
1360
1361
1362
1363
1364
1365
1366
1367
1368
1369
1370
1371
1372
1373
1374
1375
1376
1377
1378
1379
1380
1381
1382
1383
1384
1385
1386
1387
1388
1389
1390
1391
1392
1393
1394
1395
1396
1397
1398
1399
1400
1401
1402
1403
1404
1405
1406
1407
1408
1409
1410
1411
1412
1413
1414
1415
1416
1417
1418
1419
1420
1421
1422
1423
1424
1425
1426
1427
1428
1429
1430
1431
1432
1433
1434
1435
1436
1437
1438
1439
1440
1441
1442
1443
1444
1445
1446
1447
1448
1449
1450
1451
1452
1453
1454
1455
1456
1457
1458
1459
1460
1461
1462
1463
1464
1465
1466
1467
1468
1469
1470
1471
1472
1473
1474
1475
1476
1477
1478
1479
1480
1481
1482
1483
1484
1485
1486
1487
1488
1489
1490
1491
1492
1493
1494
1495
1496
1497
1498
1499
1500
1501
1502
1503
1504
1505
1506
1507
1508
1509
1510
1511
1512
1513
1514
1515
1516
1517
1518
1519
1520
1521
1522
1523
1524
1525
1526
1527
1528
1529
1530
1531
1532
1533
1534
1535
1536
1537
1538
1539
1540
1541
1542
1543
1544
1545
1546
1547
1548
1549
1550
1551
1552
1553
1554
1555
1556
1557
1558
1559
1560
1561
1562
1563
1564
1565
1566
1567
1568
1569
1570
1571
1572
1573
1574
1575
1576
1577
1578
1579
1580
1581
1582
1583
1584
1585
1586
1587
1588
1589
1590
1591
1592
1593
1594
1595
1596
1597
1598
1599
1600
1601
1602
1603
1604
1605
1606
1607
1608
1609
1610
1611
1612
1613
1614
1615
1616
1617
1618
1619
1620
1621
1622
1623
1624
1625
1626
1627
1628
1629
1630
1631
1632
1633
1634
1635
1636
1637
1638
1639
1640
1641
1642
1643
1644
1645
1646
1647
1648
1649
1650
1651
1652
1653
1654
1655
1656
1657
1658
1659
1660
1661
1662
1663
1664
1665
1666
1667
1668
1669
1670
1671
1672
1673
1674
1675
1676
1677
1678
1679
1680
1681
1682
1683
1684
1685
1686
1687
1688
1689
1690
1691
1692
1693
1694
1695
1696
1697
1698
1699
1700
1701
1702
1703
1704
1705
1706
1707
1708
1709
1710
1711
1712
1713
1714
1715
1716
1717
1718
1719
1720
1721
1722
1723
1724
1725
1726
1727
1728
1729
1730
1731
1732
1733
1734
1735
1736
1737
1738
1739
1740
1741
1742
1743
1744
1745
1746
1747
1748
1749
1750
1751
1752
1753
1754
1755
1756
1757
1758
1759
1760
1761
1762
1763
1764
1765
1766
1767
1768
1769
1770
1771
1772
1773
1774
1775
1776
1777
1778
1779
1780
1781
1782
1783
1784
1785
1786
1787
1788
1789
1790
1791
1792
1793
1794
1795
1796
1797
1798
1799
1800
1801
1802
1803
1804
1805
1806
1807
1808
1809
1810
1811
1812
1813
1814
1815
1816
1817
1818
1819
1820
1821
1822
1823
1824
1825
1826
1827
1828
1829
1830
1831
1832
1833
1834
1835
1836
1837
1838
1839
1840
1841
1842
1843
1844
1845
1846
1847
1848
1849
1850
1851
1852
1853
1854
1855
1856
1857
1858
1859
1860
1861
1862
1863
1864
1865
1866
1867
1868
1869
1870
1871
1872
1873
1874
1875
1876
1877
1878
1879
1880
1881
1882
1883
1884
1885
1886
1887
1888
1889
1890
1891
1892
1893
1894
1895
1896
1897
1898
1899
1900
1901
1902
1903
1904
1905
1906
1907
1908
1909
1910
1911
1912
1913
1914
1915
1916
1917
1918
1919
1920
1921
1922
1923
1924
1925
1926
1927
1928
1929
1930
1931
1932
1933
1934
1935
1936
1937
1938
1939
1940
1941
1942
1943
1944
1945
1946
1947
1948
1949
1950
1951
1952
1953
1954
1955
1956
1957
1958
1959
1960
1961
1962
1963
1964
1965
1966
1967
1968
1969
1970
1971
1972
1973
1974
1975
1976
1977
1978
1979
1980
1981
1982
1983
1984
1985
1986
1987
1988
1989
1990
1991
1992
1993
1994
1995
1996
1997
1998
1999
2000
2001
2002
2003
2004
2005
2006
2007
2008
2009
2010
2011
2012
2013
2014
2015
2016
2017
2018
2019
2020
2021
2022
2023
2024
2025
2026
2027
2028
2029
2030
2031
2032
2033
2034
2035
2036
2037
2038
2039
2040
2041
2042
2043
2044
2045
2046
2047
2048
2049
2050
2051
2052
2053
2054
2055
2056
2057
2058
2059
2060
2061
2062
2063
2064
2065
2066
2067
2068
2069
2070
2071
2072
2073
2074
2075
2076
2077
2078
2079
2080
2081
2082
2083
2084
2085
2086
2087
2088
2089
2090
2091
2092
2093
2094
2095
2096
2097
2098
2099
2100
2101
2102
2103
2104
2105
2106
2107
2108
2109
2110
2111
2112
2113
2114
2115
2116
2117
2118
2119
2120
2121
2122
2123
2124
2125
2126
2127
2128
2129
2130
2131
2132
2133
2134
2135
2136
2137
2138
2139
2140
2141
2142
2143
2144
2145
2146
2147
2148
2149
2150
2151
2152
2153
2154
2155
2156
2157
2158
2159
2160
2161
2162
2163
2164
2165
2166
2167
2168
2169
2170
2171
2172
2173
2174
2175
2176
2177
2178
2179
2180
2181
2182
2183
2184
2185
2186
2187
2188
2189
2190
2191
2192
2193
2194
2195
2196
2197
2198
2199
2200
2201
2202
2203
2204
2205
2206
2207
2208
2209
2210
2211
2212
2213
2214
2215
2216
2217
2218
2219
2220
2221
2222
2223
2224
2225
2226
2227
2228
2229
2230
2231
2232
2233
2234
2235
2236
2237
2238
2239
2240
2241
2242
2243
2244
2245
2246
2247
2248
2249
2250
2251
2252
2253
2254
2255
2256
2257
2258
2259
2260
2261
2262
2263
2264
2265
2266
2267
2268
2269
2270
2271
2272
2273
2274
2275
2276
2277
2278
2279
2280
2281
2282
2283
2284
2285
2286
2287
2288
2289
2290
2291
2292
2293
2294
2295
2296
2297
2298
2299
2300
2301
2302
2303
2304
2305
2306
2307
2308
2309
2310
2311
2312
2313
2314
2315
2316
2317
2318
2319
2320
2321
2322
2323
2324
2325
2326
2327
2328
2329
2330
2331
2332
2333
2334
2335
2336
2337
2338
2339
2340
2341
2342
2343
2344
2345
2346
2347
2348
2349
2350
2351
2352
2353
2354
2355
2356
2357
2358
2359
2360
2361
2362
2363
2364
2365
2366
2367
2368
2369
2370
2371
2372
2373
2374
2375
2376
2377
2378
2379
2380
2381
2382
2383
2384
2385
2386
2387
2388
2389
2390
2391
2392
2393
2394
2395
2396
2397
2398
2399
2400
2401
2402
2403
2404
2405
2406
2407
2408
2409
2410
2411
2412
2413
2414
2415
2416
2417
2418
2419
2420
2421
2422
2423
2424
2425
2426
2427
2428
2429
2430
2431
2432
2433
2434
2435
2436
2437
2438
2439
2440
2441
2442
2443
2444
2445
2446
2447
2448
2449
2450
2451
2452
2453
2454
2455
2456
2457
2458
2459
2460
2461
2462
2463
2464
2465
2466
2467
2468
2469
2470
2471
2472
2473
2474
2475
2476
2477
2478
2479
2480
2481
2482
2483
2484
2485
2486
2487
2488
2489
2490
2491
2492
2493
2494
2495
2496
2497
2498
2499
2500
2501
2502
2503
2504
2505
2506
2507
2508
2509
2510
2511
2512
2513
2514
2515
2516
2517
2518
2519
2520
2521
2522
2523
2524
2525
2526
2527
2528
2529
2530
2531
2532
2533
2534
2535
2536
2537
2538
2539
2540
2541
2542
2543
2544
2545
2546
2547
2548
2549
2550
2551
2552
2553
2554
2555
2556
2557
2558
2559
2560
2561
2562
2563
2564
2565
2566
2567
2568
2569
2570
2571
2572
2573
2574
2575
2576
2577
2578
2579
2580
2581
2582
2583
2584
2585
2586
2587
2588
2589
2590
2591
2592
2593
2594
2595
2596
2597
2598
2599
26
```

### 5.3.24 Reading Spindle information ► R\_Spindle\_Complete\_Plus

This request returns all relevant spindle information excluding the range currently active. The full list of data can be seen in the example below.

```

1  PROGRAM R_Plus_048
2  VAR
3      oR_Spindle                : R_Spindle_Complete_Plus;
4      bNumberOfSpindles         : BYTE;
5      bChannel                   : BYTE;
6      bFormat                   : BYTE;
7      bLogicalAddress           : BYTE;
8      bRank                     : BYTE; // Main or Auxiliary
9      xM19                      : BOOL;
10     xM3                       : BOOL;
11     xM4                       : BOOL;
12     xM5                       : BOOL;
13     xM62                      : BOOL;
14     xM66                      : BOOL;
15     uiOverride                 : UINT;
16     bPhysicalAddress           : BYTE;
17     lrProgSpeed               : LREAL;
18     lrRealSpeed               : LREAL;
19
20     oR_Spindle.CNCAdr          := ADR(NCK);
21
22     oR_Spindle(enable := xEnable);
23
24     IF oR_Spindle.uiResult = 0 THEN
25         xEnable                := FALSE;
26         bNumberOfSpindles       := oR_Spindle.nrOfSpindles;
27         bChannel                := oR_Spindle.SpindleData[0].Channel;
28         bFormat                := oR_Spindle.SpindleData[0].Format;
29         bLogicalAddress         := oR_Spindle.SpindleData[0].LogicalIndexInChannel;
30         bRank                  := oR_Spindle.SpindleData[0].LogicalIndexInSystem;
31         xM19                   := oR_Spindle.SpindleData[0].M19;
32         xM3                    := oR_Spindle.SpindleData[0].M3;
33         xM4                    := oR_Spindle.SpindleData[0].M4;
34         xM5                    := oR_Spindle.SpindleData[0].M5;
35         xM62                   := oR_Spindle.SpindleData[0].M62;
36         xM66                   := oR_Spindle.SpindleData[0].M66;
37         uiOverride              := oR_Spindle.SpindleData[0].Override;
38         bPhysicalAddress        := oR_Spindle.SpindleData[0].PhysicalAddress;
39         lrProgSpeed             := oR_Spindle.SpindleData[0].SpeedProgrammed;
40         lrRealSpeed             := oR_Spindle.SpindleData[0].SpeedReal;
41     END_IF

```

### 5.3.25 Reading Spindle information ► R\_Spindle\_Complete\_Plus\_V2

This request returns all relevant spindle information excluding the range currently active, but including Speed Limit G92 value respect the previous request **R\_Spindle\_Complete\_Plus**.

The full list of data can be seen in the example below.

The spindle selection is obtained writing the drive address x(0...31) into the SpindleData[x] index.....

```
R_Plus_050 x
1  PROGRAM R_Plus_050
2  VAR
3      oR_Spindle_Complete      : R_Spindle_Complete_Plus_V2;
4      bNumberOfSpindles        : BYTE;
5      bChannel                  : BYTE;
6      bFormat                   : BYTE;
7      bLogicalAddress           : BYTE;
8      bRank                     : BYTE; // Main or Auxiliary
9      xM19                      : BOOL;
10     xM3                       : BOOL;
11     xM4                       : BOOL;
12     xM5                       : BOOL;
13     xM62                      : BOOL;
14     xM66                      : BOOL;
15     uiOverride                 : UINT;
16     bPhysicalAddress           : BYTE;
17     lrProgSpeed                : LREAL; // RPM
18     lrRealSpeed                : LREAL; // RPM
19     lrSpeedLimitG92            : LREAL; // RPM
20 END_VAR

1
2  oR_Spindle_Complete.CNCAdr := ADR(NCK);
3
4  oR_Spindle_Complete(enable:= xEnable);
5
6  IF oR_Spindle_Complete.uiResult = 0 THEN
7      xEnable                := FALSE;
8      bNumberOfSpindles      := oR_Spindle_Complete.nrOfSpindles;
9      bChannel                := oR_Spindle_Complete.SpindleData[0].Channel;
10     bFormat                 := oR_Spindle_Complete.SpindleData[0].Format;
11     bLogicalAddress          := oR_Spindle_Complete.SpindleData[0].LogicalIndexInChannel;
12     bRank                   := oR_Spindle_Complete.SpindleData[0].LogicalIndexInSystem;
13     xM19                    := oR_Spindle_Complete.SpindleData[0].M19;
14     xM3                     := oR_Spindle_Complete.SpindleData[0].M3;
15     xM4                     := oR_Spindle_Complete.SpindleData[0].M4;
16     xM5                     := oR_Spindle_Complete.SpindleData[0].M5;
17     xM62                    := oR_Spindle_Complete.SpindleData[0].M62;
18     xM66                    := oR_Spindle_Complete.SpindleData[0].M66;
19     uiOverride               := oR_Spindle_Complete.SpindleData[0].Override;
20     bPhysicalAddress         := oR_Spindle_Complete.SpindleData[0].PhysicalAddress;
21     lrProgSpeed              := oR_Spindle_Complete.SpindleData[0].SpeedProgrammed;
22     lrRealSpeed              := oR_Spindle_Complete.SpindleData[0].SpeedReal;
23     lrSpeedLimitG92          := oR_Spindle_Complete.SpindleData[0].SpeedLimitG92;
24 END_IF
```

### 5.3.26 Reading Tool offsets data ► R\_Tool\_Plus

This request returns all data about a particular tool offset, excluding the H value that can be obtained via the *Read\_E\_Param* request.

```

1  PROGRAM R_Plus_049
2  VAR
3      oR_Tool          : R_Tool_Plus;
4      ToolPar1         : LREAL;
5      ToolPar2         : LREAL;
6      ToolPar3         : LREAL;
7      ToolPar4         : LREAL;
8      ToolPar5         : LREAL;
9      ToolPar6         : DINT;
10     ToolPar7         : DINT;
11 END_VAR
12
13
14
15
16
17
18
19
20
21
22
23
24
25
26
27
28
29
30
31
32
33
34
35
36
37
38
39
40
41
42
43
44
45
46
47
48
49
50
51
52
53
54
55
56
57
58
59
60
61
62
63
64
65
66
67
68
69
70
71
72
73
74
75
76
77
78
79
80
81
82
83
84
85
86
87
88
89
90
91
92
93
94
95
96
97
98
99
100
101
102
103
104
105
106
107
108
109
110
111
112
113
114
115
116
117
118
119
120
121
122
123
124
125
126
127
128
129
130
131
132
133
134
135
136
137
138
139
140
141
142
143
144
145
146
147
148
149
150
151
152
153
154
155
156
157
158
159
160
161
162
163
164
165
166
167
168
169
170
171
172
173
174
175
176
177
178
179
180
181
182
183
184
185
186
187
188
189
190
191
192
193
194
195
196
197
198
199
200
201
202
203
204
205
206
207
208
209
210
211
212
213
214
215
216
217
218
219
220
221
222
223
224
225
226
227
228
229
230
231
232
233
234
235
236
237
238
239
240
241
242
243
244
245
246
247
248
249
250
251
252
253
254
255
256
257
258
259
260
261
262
263
264
265
266
267
268
269
270
271
272
273
274
275
276
277
278
279
280
281
282
283
284
285
286
287
288
289
290
291
292
293
294
295
296
297
298
299
300
301
302
303
304
305
306
307
308
309
310
311
312
313
314
315
316
317
318
319
320
321
322
323
324
325
326
327
328
329
330
331
332
333
334
335
336
337
338
339
340
341
342
343
344
345
346
347
348
349
350
351
352
353
354
355
356
357
358
359
360
361
362
363
364
365
366
367
368
369
370
371
372
373
374
375
376
377
378
379
380
381
382
383
384
385
386
387
388
389
390
391
392
393
394
395
396
397
398
399
400
401
402
403
404
405
406
407
408
409
410
411
412
413
414
415
416
417
418
419
420
421
422
423
424
425
426
427
428
429
430
431
432
433
434
435
436
437
438
439
440
441
442
443
444
445
446
447
448
449
450
451
452
453
454
455
456
457
458
459
460
461
462
463
464
465
466
467
468
469
470
471
472
473
474
475
476
477
478
479
480
481
482
483
484
485
486
487
488
489
490
491
492
493
494
495
496
497
498
499
500
501
502
503
504
505
506
507
508
509
510
511
512
513
514
515
516
517
518
519
520
521
522
523
524
525
526
527
528
529
530
531
532
533
534
535
536
537
538
539
540
541
542
543
544
545
546
547
548
549
550
551
552
553
554
555
556
557
558
559
560
561
562
563
564
565
566
567
568
569
570
571
572
573
574
575
576
577
578
579
580
581
582
583
584
585
586
587
588
589
590
591
592
593
594
595
596
597
598
599
600
601
602
603
604
605
606
607
608
609
610
611
612
613
614
615
616
617
618
619
620
621
622
623
624
625
626
627
628
629
630
631
632
633
634
635
636
637
638
639
640
641
642
643
644
645
646
647
648
649
650
651
652
653
654
655
656
657
658
659
660
661
662
663
664
665
666
667
668
669
670
671
672
673
674
675
676
677
678
679
680
681
682
683
684
685
686
687
688
689
690
691
692
693
694
695
696
697
698
699
700
701
702
703
704
705
706
707
708
709
710
711
712
713
714
715
716
717
718
719
720
721
722
723
724
725
726
727
728
729
730
731
732
733
734
735
736
737
738
739
740
741
742
743
744
745
746
747
748
749
750
751
752
753
754
755
756
757
758
759
760
761
762
763
764
765
766
767
768
769
770
771
772
773
774
775
776
777
778
779
780
781
782
783
784
785
786
787
788
789
790
791
792
793
794
795
796
797
798
799
800
801
802
803
804
805
806
807
808
809
810
811
812
813
814
815
816
817
818
819
820
821
822
823
824
825
826
827
828
829
830
831
832
833
834
835
836
837
838
839
840
841
842
843
844
845
846
847
848
849
850
851
852
853
854
855
856
857
858
859
860
861
862
863
864
865
866
867
868
869
870
871
872
873
874
875
876
877
878
879
880
881
882
883
884
885
886
887
888
889
890
891
892
893
894
895
896
897
898
899
900
901
902
903
904
905
906
907
908
909
910
911
912
913
914
915
916
917
918
919
920
921
922
923
924
925
926
927
928
929
930
931
932
933
934
935
936
937
938
939
940
941
942
943
944
945
946
947
948
949
950
951
952
953
954
955
956
957
958
959
960
961
962
963
964
965
966
967
968
969
970
971
972
973
974
975
976
977
978
979
980
981
982
983
984
985
986
987
988
989
990
991
992
993
994
995
996
997
998
999
1000
1001
1002
1003
1004
1005
1006
1007
1008
1009
1010
1011
1012
1013
1014
1015
1016
1017
1018
1019
1020
1021
1022
1023
1024
1025
1026
1027
1028
1029
1030
1031
1032
1033
1034
1035
1036
1037
1038
1039
1040
1041
1042
1043
1044
1045
1046
1047
1048
1049
1050
1051
1052
1053
1054
1055
1056
1057
1058
1059
1060
1061
1062
1063
1064
1065
1066
1067
1068
1069
1070
1071
1072
1073
1074
1075
1076
1077
1078
1079
1080
1081
1082
1083
1084
1085
1086
1087
1088
1089
1090
1091
1092
1093
1094
1095
1096
1097
1098
1099
1100
1101
1102
1103
1104
1105
1106
1107
1108
1109
1110
1111
1112
1113
1114
1115
1116
1117
1118
1119
1120
1121
1122
1123
1124
1125
1126
1127
1128
1129
1130
1131
1132
1133
1134
1135
1136
1137
1138
1139
1140
1141
1142
1143
1144
1145
1146
1147
1148
1149
1150
1151
1152
1153
1154
1155
1156
1157
1158
1159
1160
1161
1162
1163
1164
1165
1166
1167
1168
1169
1170
1171
1172
1173
1174
1175
1176
1177
1178
1179
1180
1181
1182
1183
1184
1185
1186
1187
1188
1189
1190
1191
1192
1193
1194
1195
1196
1197
1198
1199
1200
1201
1202
1203
1204
1205
1206
1207
1208
1209
1210
1211
1212
1213
1214
1215
1216
1217
1218
1219
1220
1221
1222
1223
1224
1225
1226
1227
1228
1229
1230
1231
1232
1233
1234
1235
1236
1237
1238
1239
1240
1241
1242
1243
1244
1245
1246
1247
1248
1249
1250
1251
1252
1253
1254
1255
1256
1257
1258
1259
1260
1261
1262
1263
1264
1265
1266
1267
1268
1269
1270
1271
1272
1273
1274
1275
1276
1277
1278
1279
1280
1281
1282
1283
1284
1285
1286
1287
1288
1289
1290
1291
1292
1293
1294
1295
1296
1297
1298
1299
1300
1301
1302
1303
1304
1305
1306
1307
1308
1309
1310
1311
1312
1313
1314
1315
1316
1317
1318
1319
1320
1321
1322
1323
1324
1325
1326
1327
1328
1329
1330
1331
1332
1333
1334
1335
1336
1337
1338
1339
1340
1341
1342
1343
1344
1345
1346
1347
1348
1349
1350
1351
1352
1353
1354
1355
1356
1357
1358
1359
1360
1361
1362
1363
1364
1365
1366
1367
1368
1369
1370
1371
1372
1373
1374
1375
1376
1377
1378
1379
1380
1381
1382
1383
1384
1385
1386
1387
1388
1389
1390
1391
1392
1393
1394
1395
1396
1397
1398
1399
1400
1401
1402
1403
1404
1405
1406
1407
1408
1409
1410
1411
1412
1413
1414
1415
1416
1417
1418
1419
1420
1421
1422
1423
1424
1425
1426
1427
1428
1429
1430
1431
1432
1433
1434
1435
1436
1437
1438
1439
1440
1441
1442
1443
1444
1445
1446
1447
1448
1449
1450
1451
1452
1453
1454
1455
1456
1457
1458
1459
1460
1461
1462
1463
1464
1465
1466
1467
1468
1469
1470
1471
1472
1473
1474
1475
1476
1477
1478
1479
1480
1481
1482
1483
1484
1485
1486
1487
1488
1489
1490
1491
1492
1493
1494
1495
1496
1497
1498
1499
1500
1501
1502
1503
1504
1505
1506
1507
1508
1509
1510
1511
1512
1513
1514
1515
1516
1517
1518
1519
1520
1521
1522
1523
1524
1525
1526
1527
1528
1529
1530
1531
1532
1533
1534
1535
1536
1537
1538
1539
1540
1541
1542
1543
1544
1545
1546
1547
1548
1549
1550
1551
1552
1553
1554
1555
1556
1557
1558
1559
1560
1561
1562
1563
1564
1565
1566
1567
1568
1569
1570
1571
1572
1573
1574
1575
1576
1577
1578
1579
1580
1581
1582
1583
1584
1585
1586
1587
1588
1589
1590
1591
1592
1593
1594
1595
1596
1597
1598
1599
1600
1601
1602
1603
1604
1605
1606
1607
1608
1609
1610
1611
1612
1613
1614
1615
1616
1617
1618
1619
1620
1621
1622
1623
1624
1625
1626
1627
1628
1629
1630
1631
1632
1633
1634
1635
1636
1637
1638
1639
1640
1641
1642
1643
1644
1645
1646
1647
1648
1649
1650
1651
1652
1653
1654
1655
1656
1657
1658
1659
1660
1661
1662
1663
1664
1665
1666
1667
1668
1669
1670
1671
1672
1673
1674
1675
1676
1677
1678
1679
1680
1681
1682
1683
1684
1685
1686
1687
1688
1689
1690
1691
1692
1693
1694
1695
1696
1697
1698
1699
1700
1701
1702
1703
1704
1705
1706
1707
1708
1709
1710
1711
1712
1713
1714
1715
1716
1717
1718
1719
1720
1721
1722
1723
1724
1725
1726
1727
1728
1729
1730
1731
1732
1733
1734
1735
1736
1737
1738
1739
1740
1741
1742
1743
1744
1745
1746
1747
1748
1749
1750
1751
1752
1753
1754
1755
1756
1757
1758
1759
1760
1761
1762
1763
1764
1765
1766
1767
1768
1769
1770
1771
1772
1773
1774
1775
1776
1777
1778
1779
1780
1781
1782
1783
1784
1785
1786
1787
1788
1789
1790
1791
1792
1793
1794
1795
1796
1797
1798
1799
1800
1801
1802
1803
1804
1805
1806
1807
1808
1809
1810
1811
1812
1813
1814
1815
1816
1817
1818
1819
1820
1821
1822
1823
1824
1825
1826
1827
1828
1829
1830
1831
1832
1833
1834
1835
1836
1837
1838
1839
1840
1841
1842
1843
1844
1845
1846
1847
1848
1849
1850
1851
1852
1853
1854
1855
1856
1857
1858
1859
1860
1861
1862
1863
1864
1865
1866
1867
1868
1869
1870
1871
1872
1873
1874
1875
1876
1877
1878
1879
1880
1881
1882
1883
1884
1885
1886
1887
1888
1889
1890
1891
1892
1893
1894
1895
1896
1897
1898
1899
1900
1901
1902
1903
1904
1905
1906
1907
1908
1909
1910
1911
1912
1913
1914
1915
1916
1917
1918
1919
1920
1921
1922
1923
1924
1925
1926
1927
1928
1929
1930
1931
1932
1933
1934
1935
1936
1937
1938
1939
1940
1941
1942
1943
1944
1945
1946
1947
1948
1949
1950
1951
1952
1953
1954
1955
1956
1957
1958
1959
1960
1961
1962
1963
1964
1965
1966
1967
1968
1969
1970
1971
1972
1973
1974
1975
1976
1977
1978
1979
1980
1981
1982
1983
1984
1985
1986
1987
1988
1989
1990
1991
1992
1993
1994
1995
1996
1997
1998
1999
2000
2001
2002
2003
2004
2005
2006
2007
2008
2009
2010
2011
2012
2013
2014
2015
2016
2017
2018
2019
2020
2021
2022
2023
2024
2025
2026
2027
2028
2029
2030
2031
2032
2033
2034
2035
2036
2037
2038
2039
2040
2041
2042
2043
2044
2045
2046
2047
2048
2049
2050
2051
2052
2053
2054
2055
2056
2057
2058
2059
2060
2061
2062
2063
2064
2065
2066
2067
2068
2069
2070
2071
2072
2073
2074
2075
2076
2077
2078
2079
2080
2081
2082
2083
2084
2085
2086
2087
2088
2089
2090
2091
2092
2093
2094
2095
2096
2097
2098
2099
2100
2101
2102
2103
2104
2105
2106
2107
2108
2109
2110
2111
2112
2113
2114
2115
2116
2117
2118
2119
2120
2121
2122
2123
2124
2125
2126
2127
2128
2129
2130
2131
2132
2133
2134
2135
2136
2137
2138
2139
2140
2141
2142
2143
2144
2145
2146
2147
2148
2149
2150
2151
2152
2153
2154
2155
2156
2157
2158
2159
2160
2161
2162
2163
2164
2165
2166
2167
2168
2169
2170
2171
2172
2173
2174
2175
2176
2177
2178
2179
2180
2181
2182
2183
2184
2185
2186
2187
2188
2189
2190
2191
2192
2193
2194
2195
2196
2197
2198
2199
2200
2201
2202
2203
2204
2205
2206
2207
2208
2209
2210
2211
2212
2213
2214
2215
2216
2217
2218
2219
2220
2221
2222
2223
2224
2225
2226
2227
2228
2229
2230
2231
2232
2233
2234
2235
2236
2237
2238
2239
2240
2241
2242
2243
2244
2245
2246
2247
2248
2249
2250
2251
2252
2253
2254
2255
2256
2257
2258
2259
2260
2261
2262
2263
2264
2265
2266
2267
2268
2269
2270
2271
2272
2273
2274
2275
2276
2277
2278
2279
2280
2281
2282
2283
2284
2285
2286
2287
2288
2289
2290
2291
2292
2293
2294
2295
2296
2297
2298
2299
2300
2301
2302
2303
2304
2305
2306
2307
2308
2309
2310
2311
2312
2313
2314
2315
2316
2317
2318
2319
2320
2321
2322
2323
2324
2325
2326
2327
2328
2329
2330
2331
2332
2333
2334
2335
2336
2337
2338
2339
2340
2341
2342
2343
2344
2345
2346
2347
2348
2349
2350
2351
2352
2353
2354
2355
2356
2357
2358
2359
2360
2361
2362
2363
2364
2365
2366
2367
2368
2369
2370
2371
2372
2373
2374
2375
2376
2377
2378
2379
2380
2381
2382
2383
2384
2385
2386
2387
2388
2389
2390
2391
2392
2393
2394
2395
2396
2397
2398
2399
2400
2401
2402
2403
2404
2405
2406
2407
2408
2409
2410
2411
2412
2413
2414
2415
2416
2417
2418
2419
2420
2421
2422
2423
2424
2425
2426
2427
2428
2429
2430
2431
2432
2433
2434
2435
2436
2437
2438
2439
2440
2441
2442
2443
2444
2445
2446
2447
2448
2449
2450
2451
2452
2453
2454
2455
2456
2457
2458
2459
2460
2461
2462
2463
2464
2465
2466
2467
2468
2469
2470
2471
2472
2473
2474
2475
2476
2477
2478
2479
2480
2481
2482
2483
2484
2485
2486
2487
2488
2489
2490
2491
2492
2493
2494
2495
2496
2497
2498
2499
2500
2501
2502
2503
2504
2505
2506
2507
2508
2509
2510
2511
2512
2513
2514
2515
2516
2517
2518
2519
2520
2521
2522
2523
2524
2525
2526
2527
2528
2529
2530
2531
2532
2533
2534
2535
2536
2537
2538
2539
2540
2541
2542
2543
2544
2545
2546
2547
2548
2549
2550
2551
2552
2553
2554
2555
2556
2557
2558
2559
2560
2561
2562
2563
2564
2565
2566
2567
2568
2569
2570
2571
2572
2573
2574
2575
2576
2577
2578
2579
2580
2581
2582
2583
2584
2585
2586
2587
2588
2589
2590
2591
2592
2593
2594
2595
2596
2597
2598
2599
2600
2601
2602
2603
2604
2605
2606
2607
2608
2609
2610
2611
2612
2613
2614
2615
2616
2617
2618
2619
2620
2621
2622
2623
2624
2625
2626
2627
2628
2629
2630
2631
2632
2633
2634
2635
2636
2637
2638
2639
2640
2641
2642
2643
2644
```

---

Page deliberately empty

## Summary

|          |                                                                                           |            |
|----------|-------------------------------------------------------------------------------------------|------------|
| <b>6</b> | <b>Write requests .....</b>                                                               | <b>115</b> |
| 6.1      | General requests .....                                                                    | 115        |
| 6.1.1    | Downloading a part program ▶ <a href="#">W_Download2_PP</a> .....                         | 115        |
| 6.1.2    | Downloading a part program in Drip Feed mode ▶ <a href="#">W_Download2_DripFeed</a> ..... | 116        |
| 6.1.3    | Disable Drip Feed mode ▶ <a href="#">W_DripFeed_Disable</a> .....                         | 117        |
| 6.1.4    | Erasing an existing part program ▶ <a href="#">W_Delete_File</a> .....                    | 118        |
| 6.1.5    | Writing drive parameters in Flash memory ▶ <a href="#">W_Drive_Flash</a> .....            | 119        |
| 6.1.6    | Setting the drive in Halt ▶ <a href="#">W_Drive_Halt</a> .....                            | 120        |
| 6.1.7    | Writing drive parameters ▶ <a href="#">W_Drive_Param</a> .....                            | 121        |
| 6.1.8    | Writing drive parameters ▶ <a href="#">W_Drive_Param_Recursive</a> .....                  | 122        |
| 6.1.9    | Setting the drive in run ▶ <a href="#">W_Drive_Run</a> .....                              | 124        |
| 6.1.10   | Answering to a solicited input ▶ <a href="#">W_Solicited_Input</a> .....                  | 125        |
| 6.1.11   | Writing tool H data ▶ <a href="#">W_Tool_H</a> .....                                      | 126        |
| 6.2      | Flexium requests .....                                                                    | 127        |
| 6.2.1    | Writing a part program block ▶ <a href="#">W_Block</a> .....                              | 127        |
| 6.2.2    | Writing an axis measure correction ▶ <a href="#">W_Corr_Dyn</a> .....                     | 128        |
| 6.2.3    | Writing DAT1 values ▶ <a href="#">W_Dat1_Pref</a> .....                                   | 129        |
| 6.2.4    | Writing DAT2 values ▶ <a href="#">W_Dat2_Dec1</a> .....                                   | 130        |
| 6.2.5    | Writing DAT3 values ▶ <a href="#">W_Dat3_Dec3</a> .....                                   | 131        |
| 6.2.6    | Writing an E8... parameter value ▶ <a href="#">W_E8xyyy</a> .....                         | 132        |
| 6.2.7    | Writing an E5 or E6 parameter value ▶ <a href="#">W_EParam_5_6</a> .....                  | 133        |
| 6.2.8    | Writing the home position (P16) ▶ <a href="#">W_Mach_Orig</a> .....                       | 134        |
| 6.2.9    | Writing Maximum Dynamic Travel values ▶ <a href="#">W_Max_Dyn_Travel</a> .....            | 135        |
| 6.2.10   | Writing Maximum Dynamic Travel selected ▶ <a href="#">W_Max_Dyn_Travel_2</a> .....        | 136        |
| 6.2.11   | Writing Maximum Static Travel values ▶ <a href="#">W_Max_Stat_Travel</a> .....            | 137        |
| 6.2.12   | Writing Minimum Dynamic Travel values ▶ <a href="#">W_Min_Dyn_Travel</a> .....            | 138        |
| 6.2.13   | Writing Minimum Dynamic Travel selected ▶ <a href="#">W_Min_Dyn_Travel_2</a> .....        | 139        |
| 6.2.14   | Writing Minimum Static Travel values ▶ <a href="#">W_Min_Stat_Travel</a> .....            | 140        |
| 6.2.15   | Writing an NCK parameter ▶ <a href="#">W_NCK_Par</a> .....                                | 141        |
| 6.2.16   | Writing a tool offset ▶ <a href="#">W_Tool</a> .....                                      | 142        |
| 6.3      | Flexium+ requests .....                                                                   | 143        |
| 6.3.1    | Writing a part program block ▶ <a href="#">W_Block_Plus</a> .....                         | 143        |
| 6.3.2    | Writing an axis measure correction ▶ <a href="#">W_Corr_Dyn_Plus</a> .....                | 144        |
| 6.3.3    | Writing DAT1 values ▶ <a href="#">W_Dat1_Pref_Plus</a> .....                              | 145        |
| 6.3.4    | Writing DAT2 values ▶ <a href="#">W_Dat2_Dec1_Plus</a> .....                              | 146        |
| 6.3.5    | Writing DAT3 values ▶ <a href="#">W_Dat3_Dec3_Plus</a> .....                              | 147        |
| 6.3.6    | Writing an E8... parameter value ▶ <a href="#">W_E8xyyy_Plus</a> .....                    | 148        |
| 6.3.7    | Writing an E5 or E6 parameter value ▶ <a href="#">W_E_Param_5_6_Plus</a> .....            | 149        |
| 6.3.8    | Writing the home position (P16) ▶ <a href="#">W_Mach_Orig_Plus</a> .....                  | 150        |
| 6.3.9    | Writing Maximum Dynamic Travel values ▶ <a href="#">W_Max_Dyn_Travel_Plus</a> .....       | 151        |
| 6.3.10   | Writing Maximum Dynamic Travel selected ▶ <a href="#">W_Max_Dyn_Travel_Plus_2</a> .....   | 152        |
| 6.3.11   | Writing Maximum Static Travel values ▶ <a href="#">W_Max_Stat_Travel_Plus</a> .....       | 153        |
| 6.3.12   | Writing Minimum Dynamic Travel values ▶ <a href="#">W_Min_Dyn_Travel_Plus</a> .....       | 154        |
| 6.3.13   | Writing Minimum Dynamic Travel selected ▶ <a href="#">W_Min_Dyn_Travel_Plus_2</a> .....   | 155        |
| 6.3.14   | Writing Minimum Static Travel values ▶ <a href="#">W_Min_Stat_Travel_Plus</a> .....       | 156        |
| 6.3.15   | Execute motor brake test ▶ <a href="#">W_Motor_Brake_Test</a> .....                       | 157        |
| 6.3.16   | Writing an NCK parameter ▶ <a href="#">W_NCK_Par_Plus</a> .....                           | 158        |
| 6.3.17   | Writing a tool offset ▶ <a href="#">W_Tool_Plus</a> .....                                 | 159        |

---

Page deliberately empty

## 6 Write requests

### 6.1 General requests

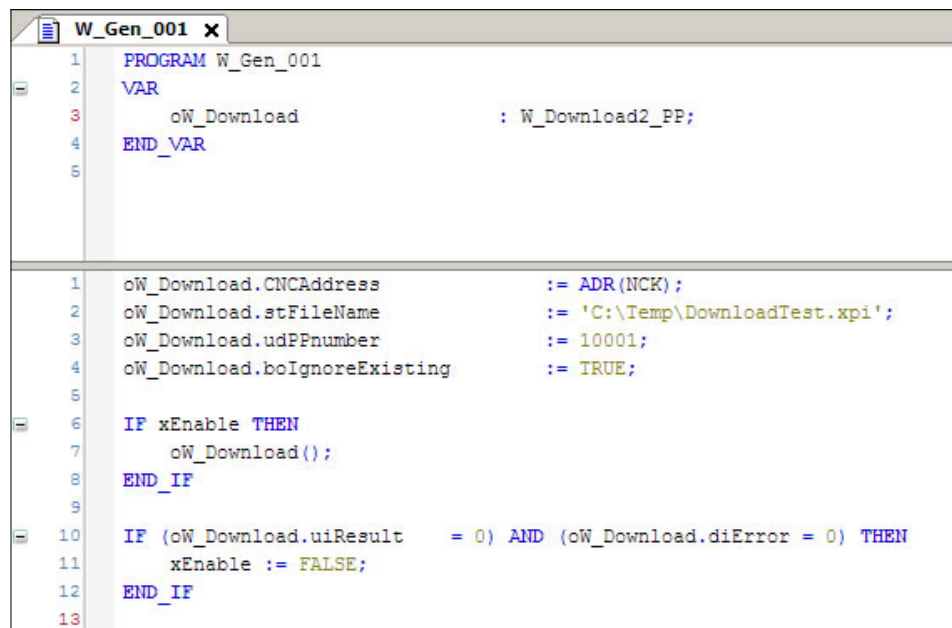
#### 6.1.1 Downloading a part program ► [W\\_Download2\\_PP](#)

This request replaces the older request *W\_Download\_PP* (see M00026EN-00 manual).

The part program to load is taken from the mass memory. Be careful to maintain the request until the end of the transfer.

A program of the same number already in memory must be erased first before the download.

The flag *boIgnoreExisting* is used to authorize or cancel the transfer if the program to download already exists in one of the protected zones (1 to 3). The transfer is cancelled when the flag is False.



```

1  PROGRAM W_Gen_001
2  VAR
3      oW_Download          : W_Download2_PP;
4  END_VAR
5
6  oW_Download.CNCAddress   := ADR(NCK);
7  oW_Download.stFileName   := 'C:\Temp\DownloadTest.xpi';
8  oW_Download.udPPnumber   := 10001;
9  oW_Download.boIgnoreExisting := TRUE;
10
11 IF xEnable THEN
12     oW_Download();
13 END_IF
14
15 IF (oW_Download.uiResult = 0) AND (oW_Download.diError = 0) THEN
16     xEnable := FALSE;
17 END_IF
18

```

### 6.1.2 Downloading a part program in Drip Feed mode ► [W\\_Download2\\_DripFeed](#)

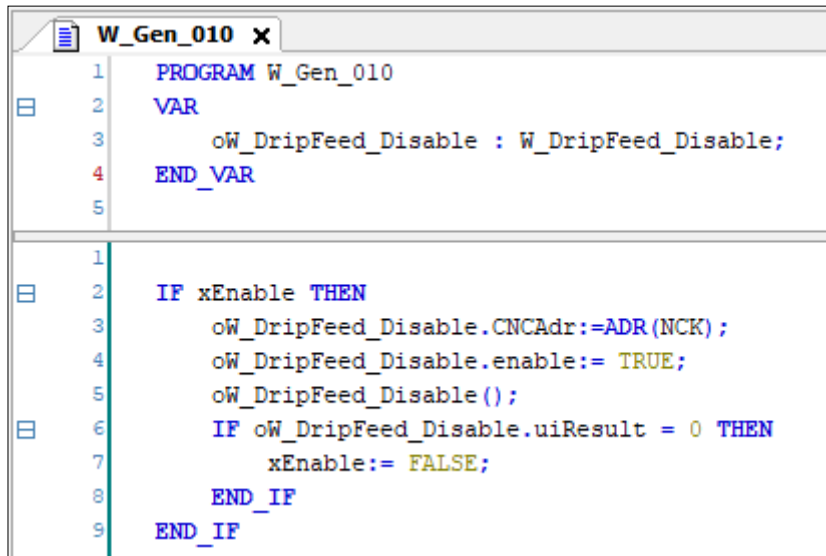
The part program to load is taken from the mass memory and executed into the channel defined in the function call. Be careful to maintain the request until the end of the transfer.

```
W_Gen_009 x
1  PROGRAM W_Gen_009
2  VAR
3      oW_Download2_DripFeed  : W_Download2_DripFeed;
4      stPath                  : STRING;
5      byPPIndex               : BYTE;
6      dError                  : DINT;           // Error code
7  END_VAR

1  stPath:= 'C:/temp/DownloadTestPPP.xpi';
2  byPPIndex:= 1; // Channel 1-8: 1-8; 0 Channel: 0
3
4  oW_Download2_DripFeed.cancel:=FALSE;
5  oW_Download2_DripFeed.useFile:=TRUE;
6  oW_Download2_DripFeed.CNCAdr:=ADR(NCK);
7  oW_Download2_DripFeed.stFileName:=stPath;
8  oW_Download2_DripFeed.byProgramIndex:=byPPIndex;
9
10 IF xEnable THEN
11     oW_Download2_DripFeed();
12     IF oW_Download2_DripFeed.uiResult = 0 THEN
13         xEnable:= FALSE;
14         dError:= oW_Download2_DripFeed.diError;
15     END_IF
16 END_IF
```

### 6.1.3 Disable Drip Feed mode ► [W\\_DripFeed\\_Disable](#)

This request is used to disable the drip feed mode.



```
1  PROGRAM W_Gen_010
2  VAR
3      oW_DripFeed_Disable : W_DripFeed_Disable;
4  END_VAR
5
6  IF xEnable THEN
7      oW_DripFeed_Disable.CNCAdr:=ADR(NCK);
8      oW_DripFeed_Disable.enable:= TRUE;
9      oW_DripFeed_Disable();
10     IF oW_DripFeed_Disable.uiResult = 0 THEN
11         xEnable:= FALSE;
12     END_IF
13 END_IF
```

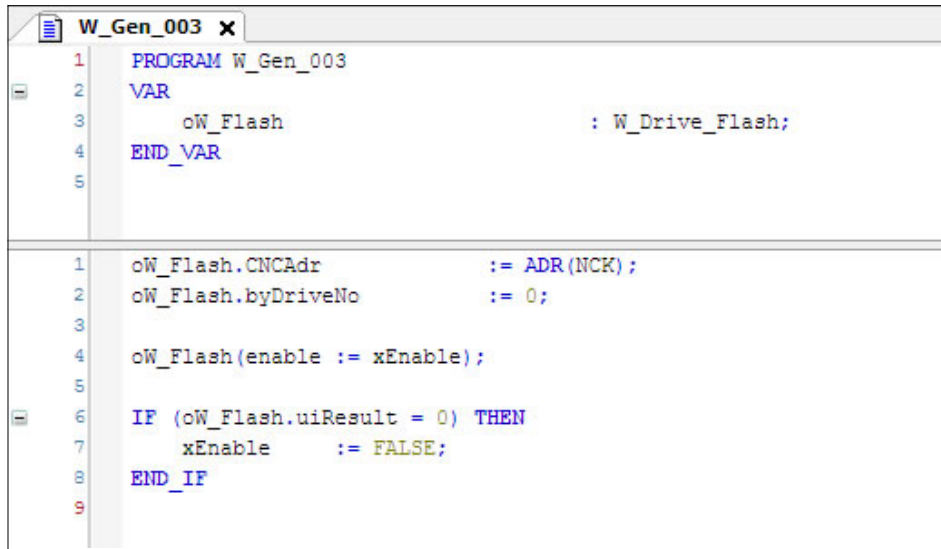
#### 6.1.4 Erasing an existing part program ► [W\\_Delete\\_File](#)

This request is intended to erase a part program in NC memory generally to permit downloading a program with the same number.

```
W_Gen_002 x
1  PROGRAM W_Gen_002
2  VAR
3      oW_DeleteFile          : W_DeleteFile;
4      byStatus               : BYTE;
5  END_VAR
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24
25
26
27
28
29
30
31
32
33
34
35
36
37
38
39
40
41
42
43
44
45
46
47
48
49
50
51
52
53
54
55
56
57
58
59
60
61
62
63
64
65
66
67
68
69
70
71
72
73
74
75
76
77
78
79
80
81
82
83
84
85
86
87
88
89
90
91
92
93
94
95
96
97
98
99
100
101
102
103
104
105
106
107
108
109
110
111
112
113
114
115
116
117
118
119
120
121
122
123
124
125
126
127
128
129
130
131
132
133
134
135
136
137
138
139
140
141
142
143
144
145
146
147
148
149
150
151
152
153
154
155
156
157
158
159
160
161
162
163
164
165
166
167
168
169
170
171
172
173
174
175
176
177
178
179
180
181
182
183
184
185
186
187
188
189
190
191
192
193
194
195
196
197
198
199
200
201
202
203
204
205
206
207
208
209
210
211
212
213
214
215
216
217
218
219
220
221
222
223
224
225
226
227
228
229
230
231
232
233
234
235
236
237
238
239
240
241
242
243
244
245
246
247
248
249
250
251
252
253
254
255
256
257
258
259
260
261
262
263
264
265
266
267
268
269
270
271
272
273
274
275
276
277
278
279
280
281
282
283
284
285
286
287
288
289
290
291
292
293
294
295
296
297
298
299
300
301
302
303
304
305
306
307
308
309
310
311
312
313
314
315
316
317
318
319
320
321
322
323
324
325
326
327
328
329
330
331
332
333
334
335
336
337
338
339
340
341
342
343
344
345
346
347
348
349
350
351
352
353
354
355
356
357
358
359
360
361
362
363
364
365
366
367
368
369
370
371
372
373
374
375
376
377
378
379
380
381
382
383
384
385
386
387
388
389
390
391
392
393
394
395
396
397
398
399
400
401
402
403
404
405
406
407
408
409
410
411
412
413
414
415
416
417
418
419
420
421
422
423
424
425
426
427
428
429
430
431
432
433
434
435
436
437
438
439
440
441
442
443
444
445
446
447
448
449
450
451
452
453
454
455
456
457
458
459
460
461
462
463
464
465
466
467
468
469
470
471
472
473
474
475
476
477
478
479
480
481
482
483
484
485
486
487
488
489
490
491
492
493
494
495
496
497
498
499
500
501
502
503
504
505
506
507
508
509
510
511
512
513
514
515
516
517
518
519
520
521
522
523
524
525
526
527
528
529
530
531
532
533
534
535
536
537
538
539
540
541
542
543
544
545
546
547
548
549
550
551
552
553
554
555
556
557
558
559
560
561
562
563
564
565
566
567
568
569
570
571
572
573
574
575
576
577
578
579
580
581
582
583
584
585
586
587
588
589
590
591
592
593
594
595
596
597
598
599
600
601
602
603
604
605
606
607
608
609
610
611
612
613
614
615
616
617
618
619
620
621
622
623
624
625
626
627
628
629
630
631
632
633
634
635
636
637
638
639
640
641
642
643
644
645
646
647
648
649
650
651
652
653
654
655
656
657
658
659
660
661
662
663
664
665
666
667
668
669
670
671
672
673
674
675
676
677
678
679
680
681
682
683
684
685
686
687
688
689
690
691
692
693
694
695
696
697
698
699
700
701
702
703
704
705
706
707
708
709
710
711
712
713
714
715
716
717
718
719
720
721
722
723
724
725
726
727
728
729
730
731
732
733
734
735
736
737
738
739
740
741
742
743
744
745
746
747
748
749
750
751
752
753
754
755
756
757
758
759
760
761
762
763
764
765
766
767
768
769
770
771
772
773
774
775
776
777
778
779
780
781
782
783
784
785
786
787
788
789
790
791
792
793
794
795
796
797
798
799
800
801
802
803
804
805
806
807
808
809
810
811
812
813
814
815
816
817
818
819
820
821
822
823
824
825
826
827
828
829
830
831
832
833
834
835
836
837
838
839
840
841
842
843
844
845
846
847
848
849
850
851
852
853
854
855
856
857
858
859
860
861
862
863
864
865
866
867
868
869
870
871
872
873
874
875
876
877
878
879
880
881
882
883
884
885
886
887
888
889
890
891
892
893
894
895
896
897
898
899
900
901
902
903
904
905
906
907
908
909
910
911
912
913
914
915
916
917
918
919
920
921
922
923
924
925
926
927
928
929
930
931
932
933
934
935
936
937
938
939
940
941
942
943
944
945
946
947
948
949
950
951
952
953
954
955
956
957
958
959
960
961
962
963
964
965
966
967
968
969
970
971
972
973
974
975
976
977
978
979
980
981
982
983
984
985
986
987
988
989
990
991
992
993
994
995
996
997
998
999
1000
1001
1002
1003
1004
1005
1006
1007
1008
1009
1010
1011
1012
1013
1014
1015
1016
1017
1018
1019
1020
1021
1022
1023
1024
1025
1026
1027
1028
1029
1030
1031
1032
1033
1034
1035
1036
1037
1038
1039
1040
1041
1042
1043
1044
1045
1046
1047
1048
1049
1050
1051
1052
1053
1054
1055
1056
1057
1058
1059
1060
1061
1062
1063
1064
1065
1066
1067
1068
1069
1070
1071
1072
1073
1074
1075
1076
1077
1078
1079
1080
1081
1082
1083
1084
1085
1086
1087
1088
1089
1090
1091
1092
1093
1094
1095
1096
1097
1098
1099
1100
1101
1102
1103
1104
1105
1106
1107
1108
1109
1110
1111
1112
1113
1114
1115
1116
1117
1118
1119
1120
1121
1122
1123
1124
1125
1126
1127
1128
1129
1130
1131
1132
1133
1134
1135
1136
1137
1138
1139
1140
1141
1142
1143
1144
1145
1146
1147
1148
1149
1150
1151
1152
1153
1154
1155
1156
1157
1158
1159
1160
1161
1162
1163
1164
1165
1166
1167
1168
1169
1170
1171
1172
1173
1174
1175
1176
1177
1178
1179
1180
1181
1182
1183
1184
1185
1186
1187
1188
1189
1190
1191
1192
1193
1194
1195
1196
1197
1198
1199
1200
1201
1202
1203
1204
1205
1206
1207
1208
1209
1210
1211
1212
1213
1214
1215
1216
1217
1218
1219
1220
1221
1222
1223
1224
1225
1226
1227
1228
1229
1230
1231
1232
1233
1234
1235
1236
1237
1238
1239
1240
1241
1242
1243
1244
1245
1246
1247
1248
1249
1250
1251
1252
1253
1254
1255
1256
1257
1258
1259
1260
1261
1262
1263
1264
1265
1266
1267
1268
1269
1270
1271
1272
1273
1274
1275
1276
1277
1278
1279
1280
1281
1282
1283
1284
1285
1286
1287
1288
1289
1290
1291
1292
1293
1294
1295
1296
1297
1298
1299
1300
1301
1302
1303
1304
1305
1306
1307
1308
1309
1310
1311
1312
1313
1314
1315
1316
1317
1318
1319
1320
1321
1322
1323
1324
1325
1326
1327
1328
1329
1330
1331
1332
1333
1334
1335
1336
1337
1338
1339
1340
1341
1342
1343
1344
1345
1346
1347
1348
1349
1350
1351
1352
1353
1354
1355
1356
1357
1358
1359
1360
1361
1362
1363
1364
1365
1366
1367
1368
1369
1370
1371
1372
1373
1374
1375
1376
1377
1378
1379
1380
1381
1382
1383
1384
1385
1386
1387
1388
1389
1390
1391
1392
1393
1394
1395
1396
1397
1398
1399
1400
1401
1402
1403
1404
1405
1406
1407
1408
1409
1410
1411
1412
1413
1414
1415
1416
1417
1418
1419
1420
1421
1422
1423
1424
1425
1426
1427
1428
1429
1430
1431
1432
1433
1434
1435
1436
1437
1438
1439
1440
1441
1442
1443
1444
1445
1446
1447
1448
1449
1450
1451
1452
1453
1454
1455
1456
1457
1458
1459
1460
1461
1462
1463
1464
1465
1466
1467
1468
1469
1470
1471
1472
1473
1474
1475
1476
1477
1478
1479
1480
1481
1482
1483
1484
1485
1486
1487
1488
1489
1490
1491
1492
1493
1494
1495
1496
1497
1498
1499
1500
1501
1502
1503
1504
1505
1506
1507
1508
1509
1510
1511
1512
1513
1514
1515
1516
1517
1518
1519
1520
1521
1522
1523
1524
1525
1526
1527
1528
1529
1530
1531
1532
1533
1534
1535
1536
1537
1538
1539
1540
1541
1542
1543
1544
1545
1546
1547
1548
1549
1550
1551
1552
1553
1554
1555
1556
1557
1558
1559
1560
1561
1562
1563
1564
1565
1566
1567
1568
1569
1570
1571
1572
1573
1574
1575
1576
1577
1578
1579
1580
1581
1582
1583
1584
1585
1586
1587
1588
1589
1590
1591
1592
1593
1594
1595
1596
1597
1598
1599
1600
1601
1602
1603
1604
1605
1606
1607
1608
1609
1610
1611
1612
1613
1614
1615
1616
1617
1618
1619
1620
1621
1622
1623
1624
1625
1626
1627
1628
1629
1630
1631
1632
1633
1634
1635
1636
1637
1638
1639
1640
1641
1642
1643
1644
1645
1646
1647
1648
1649
1650
1651
1652
1653
1654
1655
1656
1657
1658
1659
1660
1661
1662
1663
1664
1665
1666
1667
1668
1669
1670
1671
1672
1673
1674
1675
1676
1677
1678
1679
1680
1681
1682
1683
1684
1685
1686
1687
1688
1689
1690
1691
1692
1693
1694
1695
1696
1697
1698
1699
1700
1701
1702
1703
1704
1705
1706
1707
1708
1709
1710
1711
1712
1713
1714
1715
1716
1717
1718
1719
1720
1721
1722
1723
1724
1725
1726
1727
1728
1729
1730
1731
1732
1733
1734
1735
1736
1737
1738
1739
1740
1741
1742
1743
1744
1745
1746
1747
1748
1749
1750
1751
1752
1753
1754
1755
1756
1757
1758
1759
1760
1761
1762
1763
1764
1765
1766
1767
1768
1769
1770
1771
1772
1773
1774
1775
1776
1777
1778
1779
1780
1781
1782
1783
1784
1785
1786
1787
1788
1789
1790
1791
1792
1793
1794
1795
1796
1797
1798
1799
1800
1801
1802
1803
1804
1805
1806
1807
1808
1809
1810
1811
1812
1813
1814
1815
1816
1817
1818
1819
1820
1821
1822
1823
1824
1825
1826
1827
1828
1829
1830
1831
1832
1833
1834
1835
1836
1837
1838
1839
1840
1841
1842
1843
1844
1845
1846
1847
1848
1849
1850
1851
1852
1853
1854
1855
1856
1857
1858
1859
1860
1861
1862
1863
1864
1865
1866
1867
1868
1869
1870
1871
1872
1873
1874
1875
1876
1877
1878
1879
1880
1881
1882
1883
1884
1885
1886
1887
1888
1889
1890
1891
1892
1893
1894
1895
1896
1897
1898
1899
1900
1901
1902
1903
1904
1905
1906
1907
1908
1909
1910
1911
1912
1913
1914
1915
1916
1917
1918
1919
1920
1921
1922
1923
1924
1925
1926
1927
1928
1929
1930
1931
1932
1933
1934
1935
1936
1937
1938
1939
1940
1941
1942
1943
1944
1945
1946
1947
1948
1949
1950
1951
1952
1953
1954
1955
1956
1957
1958
1959
1960
1961
1962
1963
1964
1965
1966
1967
1968
1969
1970
1971
1972
1973
1974
1975
1976
1977
1978
1979
1980
1981
1982
1983
1984
1985
1986
1987
1988
1989
1990
1991
1992
1993
1994
1995
1996
1997
1998
1999
2000
2001
2002
2003
2004
2005
2006
2007
2008
2009
2010
2011
2012
2013
2014
2015
2016
2017
2018
2019
2020
2021
2022
2023
2024
2025
2026
2027
2028
2029
2030
2031
2032
2033
2034
2035
2036
2037
2038
2039
2040
2041
2042
2043
2044
2045
2046
2047
2048
2049
2050
2051
2052
2053
2054
2055
2056
2057
2058
2059
2060
2061
2062
2063
2064
2065
2066
2067
2068
2069
2070
2071
2072
2073
2074
2075
2076
2077
2078
2079
2080
2081
2082
2083
2084
2085
2086
2087
2088
2089
2090
2091
2092
2093
2094
2095
2096
2097
2098
2099
2100
2101
2102
2103
2104
2105
2106
2107
2108
2109
2110
2111
2112
2113
2114
2115
2116
2117
2118
2119
2120
2121
2122
2123
2124
2125
2126
2127
2128
2129
2130
2131
2132
2133
2134
2135
2136
2137
2138
2139
2140
2141
2142
2143
2144
2145
2146
2147
2148
2149
2150
2151
2152
2153
2154
2155
2156
2157
2158
2159
2160
2161
2162
2163
2164
2165
2166
2167
2168
2169
2170
2171
2172
2173
2174
2175
2176
2177
2178
2179
2180
2181
2182
2183
2184
2185
2186
2187
2188
2189
2190
2191
2192
2193
2194
2195
2196
2197
2198
2199
2200
2201
2202
2203
2204
2205
2206
2207
2208
2209
2210
2211
2212
2213
2214
2215
2216
2217
2218
2219
2220
2221
2222
2223
2224
2225
2226
2227
2228
2229
2230
2231
2232
2233
2234
2235
2236
2237
2238
2239
2240
2241
2242
2243
2244
2245
2246
2247
2248
2249
2250
2251
2252
2253
2254
2255
2256
2257
2258
2259
2260
2261
2262
2263
2264
2265
2266
2267
2268
2269
2270
2271
2272
2273
2274
2275
2276
2277
2278
2279
2280
2281
2282
2283
2284
2285
2286
2287
2288
2289
2290
2291
2292
2293
2294
2295
2296
2297
2298
2299
2300
2301
2302
2303
2304
2305
2306
2307
2308
2309
2310
2311
2312
2313
2314
2315
2316
2317
2318
2319
2320
2321
2322
2323
2324
2325
2326
2327
2328
2329
2330
2331
2332
2333
2334
2335
2336
2337
2338
2339
2340
2341
2342
2343
2344
2345
2346
2347
2348
2349
2350
2351
2352
2353
2354
2355
2356
2357
2358
2359
2360
2361
2362
2363
2364
2365
2366
2367
2368
2369
2370
2371
2372
2373
2374
2375
2376
2377
2378
2379
2380
2381
2382
2383
2384
2385
2386
2387
2388
2389
2390
2391
2392
2393
2394
2395
2396
2397
2398
2399
2400
2401
2402
2403
2404
2405
2406
2407
2408
2409
2410
2411
2412
2413
2414
2415
2416
2417
2418
2419
2420
2421
2422
2423
2424
2425
2426
2427
2428
2429
2430
2431
2432
2433
2434
2435
2436
2437
2438
2439
2440
2441
2442
2443
2444
2445
2446
2447
2448
2449
2450
2451
2452
2453
2454
2455
2456
2457
2458
2459
2460
2461
2462
2463
2464
2465
2466
2467
2468
2469
2470
2471
2472
2473
2474
2475
2476
2477
2478
2479
2480
2481
2482
2483
2484
2485
2486
2487
2488
2489
2490
2491
2492
2493
2494
2495
2496
2497
2498
2499
2500
2501
2502
2503
2504
2505
2506
2507
2508
2509
2510
2511
2512
2513
2514
2515
2516
2517
2518
2519
2520
2521
2522
2523
2524
2525
2526
2527
2528
2529
2530
2531
2532
2533
2534
2535
2536
2537
2538
2539
2540
2541
2542
2543
2544
2545
2546
2547
2548
2549
2550
2551
2552
2553
2554
2555
2556
2557
2558
2559
2560
2561
2562
2563
2564
2565
2566
2567
2568
2569
2570
2571
2572
2573
2574
2575
2576
2577
2578
2579
2580
2581
2582
2583
2584
2585
2586
2587
2588
2589
2590
2591
2592
2593
2594
2595
2596
2597
2598
2599
2600
2601
2602
2603
2604
2605
2606
2607
2608
2609
2610
2611
2612
2613
2614
2615
2616
2617
2618
2619
2620
2621
2622
2623
2624
2625
2626
```

### 6.1.5 Writing drive parameters in Flash memory ► [W\\_Drive\\_Flash](#)

This request is used to make permanent the changes that have been made by *W\_Drive\_Param*.

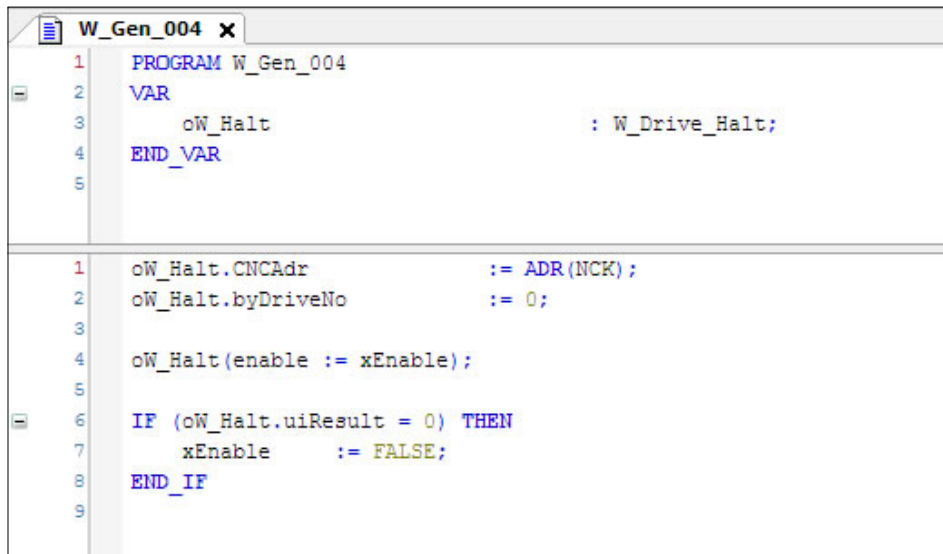


```
1  PROGRAM W_Gen_003
2  VAR
3      oW_Flash                : W_Drive_Flash;
4  END_VAR
5
6  oW_Flash.CNCAdr             := ADR(NCK) ;
7  oW_Flash.byDriveNo          := 0;
8
9  oW_Flash(enable := xEnable);
10
11 IF (oW_Flash.uiResult = 0) THEN
12     xEnable                 := FALSE;
13 END_IF
```

---

### 6.1.6 Setting the drive in Halt ► [W\\_Drive\\_Halt](#)

Some drive parameters can only be changed when the drive is in Halt. Before launching this request the programmer must take all precautions to prevent any incident or accident.



```
1  PROGRAM W_Gen_004
2  VAR
3      oW_Halt                : W_Drive_Halt;
4  END_VAR
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24
25
26
27
28
29
30
31
32
33
34
35
36
37
38
39
40
41
42
43
44
45
46
47
48
49
50
51
52
53
54
55
56
57
58
59
60
61
62
63
64
65
66
67
68
69
70
71
72
73
74
75
76
77
78
79
80
81
82
83
84
85
86
87
88
89
90
91
92
93
94
95
96
97
98
99
100
101
102
103
104
105
106
107
108
109
110
111
112
113
114
115
116
117
118
119
120
121
122
123
124
125
126
127
128
129
130
131
132
133
134
135
136
137
138
139
140
141
142
143
144
145
146
147
148
149
150
151
152
153
154
155
156
157
158
159
160
161
162
163
164
165
166
167
168
169
170
171
172
173
174
175
176
177
178
179
180
181
182
183
184
185
186
187
188
189
190
191
192
193
194
195
196
197
198
199
200
201
202
203
204
205
206
207
208
209
210
211
212
213
214
215
216
217
218
219
220
221
222
223
224
225
226
227
228
229
230
231
232
233
234
235
236
237
238
239
240
241
242
243
244
245
246
247
248
249
250
251
252
253
254
255
256
257
258
259
260
261
262
263
264
265
266
267
268
269
270
271
272
273
274
275
276
277
278
279
280
281
282
283
284
285
286
287
288
289
290
291
292
293
294
295
296
297
298
299
300
301
302
303
304
305
306
307
308
309
310
311
312
313
314
315
316
317
318
319
320
321
322
323
324
325
326
327
328
329
330
331
332
333
334
335
336
337
338
339
340
341
342
343
344
345
346
347
348
349
350
351
352
353
354
355
356
357
358
359
360
361
362
363
364
365
366
367
368
369
370
371
372
373
374
375
376
377
378
379
380
381
382
383
384
385
386
387
388
389
390
391
392
393
394
395
396
397
398
399
400
401
402
403
404
405
406
407
408
409
410
411
412
413
414
415
416
417
418
419
420
421
422
423
424
425
426
427
428
429
430
431
432
433
434
435
436
437
438
439
440
441
442
443
444
445
446
447
448
449
450
451
452
453
454
455
456
457
458
459
460
461
462
463
464
465
466
467
468
469
470
471
472
473
474
475
476
477
478
479
480
481
482
483
484
485
486
487
488
489
490
491
492
493
494
495
496
497
498
499
500
501
502
503
504
505
506
507
508
509
510
511
512
513
514
515
516
517
518
519
520
521
522
523
524
525
526
527
528
529
530
531
532
533
534
535
536
537
538
539
540
541
542
543
544
545
546
547
548
549
550
551
552
553
554
555
556
557
558
559
560
561
562
563
564
565
566
567
568
569
570
571
572
573
574
575
576
577
578
579
580
581
582
583
584
585
586
587
588
589
590
591
592
593
594
595
596
597
598
599
600
601
602
603
604
605
606
607
608
609
610
611
612
613
614
615
616
617
618
619
620
621
622
623
624
625
626
627
628
629
630
631
632
633
634
635
636
637
638
639
640
641
642
643
644
645
646
647
648
649
650
651
652
653
654
655
656
657
658
659
660
661
662
663
664
665
666
667
668
669
670
671
672
673
674
675
676
677
678
679
680
681
682
683
684
685
686
687
688
689
690
691
692
693
694
695
696
697
698
699
700
701
702
703
704
705
706
707
708
709
710
711
712
713
714
715
716
717
718
719
720
721
722
723
724
725
726
727
728
729
730
731
732
733
734
735
736
737
738
739
740
741
742
743
744
745
746
747
748
749
750
751
752
753
754
755
756
757
758
759
760
761
762
763
764
765
766
767
768
769
770
771
772
773
774
775
776
777
778
779
780
781
782
783
784
785
786
787
788
789
790
791
792
793
794
795
796
797
798
799
800
801
802
803
804
805
806
807
808
809
810
811
812
813
814
815
816
817
818
819
820
821
822
823
824
825
826
827
828
829
830
831
832
833
834
835
836
837
838
839
840
841
842
843
844
845
846
847
848
849
850
851
852
853
854
855
856
857
858
859
860
861
862
863
864
865
866
867
868
869
870
871
872
873
874
875
876
877
878
879
880
881
882
883
884
885
886
887
888
889
890
891
892
893
894
895
896
897
898
899
900
901
902
903
904
905
906
907
908
909
910
911
912
913
914
915
916
917
918
919
920
921
922
923
924
925
926
927
928
929
930
931
932
933
934
935
936
937
938
939
940
941
942
943
944
945
946
947
948
949
950
951
952
953
954
955
956
957
958
959
960
961
962
963
964
965
966
967
968
969
970
971
972
973
974
975
976
977
978
979
980
981
982
983
984
985
986
987
988
989
990
991
992
993
994
995
996
997
998
999
1000
1001
1002
1003
1004
1005
1006
1007
1008
1009
1010
1011
1012
1013
1014
1015
1016
1017
1018
1019
1020
1021
1022
1023
1024
1025
1026
1027
1028
1029
1030
1031
1032
1033
1034
1035
1036
1037
1038
1039
1040
1041
1042
1043
1044
1045
1046
1047
1048
1049
1050
1051
1052
1053
1054
1055
1056
1057
1058
1059
1060
1061
1062
1063
1064
1065
1066
1067
1068
1069
1070
1071
1072
1073
1074
1075
1076
1077
1078
1079
1080
1081
1082
1083
1084
1085
1086
1087
1088
1089
1090
1091
1092
1093
1094
1095
1096
1097
1098
1099
1100
1101
1102
1103
1104
1105
1106
1107
1108
1109
1110
1111
1112
1113
1114
1115
1116
1117
1118
1119
1120
1121
1122
1123
1124
1125
1126
1127
1128
1129
1130
1131
1132
1133
1134
1135
1136
1137
1138
1139
1140
1141
1142
1143
1144
1145
1146
1147
1148
1149
1150
1151
1152
1153
1154
1155
1156
1157
1158
1159
1160
1161
1162
1163
1164
1165
1166
1167
1168
1169
1170
1171
1172
1173
1174
1175
1176
1177
1178
1179
1180
1181
1182
1183
1184
1185
1186
1187
1188
1189
1190
1191
1192
1193
1194
1195
1196
1197
1198
1199
1200
1201
1202
1203
1204
1205
1206
1207
1208
1209
1210
1211
1212
1213
1214
1215
1216
1217
1218
1219
1220
1221
1222
1223
1224
1225
1226
1227
1228
1229
1230
1231
1232
1233
1234
1235
1236
1237
1238
1239
1240
1241
1242
1243
1244
1245
1246
1247
1248
1249
1250
1251
1252
1253
1254
1255
1256
1257
1258
1259
1260
1261
1262
1263
1264
1265
1266
1267
1268
1269
1270
1271
1272
1273
1274
1275
1276
1277
1278
1279
1280
1281
1282
1283
1284
1285
1286
1287
1288
1289
1290
1291
1292
1293
1294
1295
1296
1297
1298
1299
1300
1301
1302
1303
1304
1305
1306
1307
1308
1309
1310
1311
1312
1313
1314
1315
1316
1317
1318
1319
1320
1321
1322
1323
1324
1325
1326
1327
1328
1329
1330
1331
1332
1333
1334
1335
1336
1337
1338
1339
1340
1341
1342
1343
1344
1345
1346
1347
1348
1349
1350
1351
1352
1353
1354
1355
1356
1357
1358
1359
1360
1361
1362
1363
1364
1365
1366
1367
1368
1369
1370
1371
1372
1373
1374
1375
1376
1377
1378
1379
1380
1381
1382
1383
1384
1385
1386
1387
1388
1389
1390
1391
1392
1393
1394
1395
1396
1397
1398
1399
1400
1401
1402
1403
1404
1405
1406
1407
1408
1409
1410
1411
1412
1413
1414
1415
1416
1417
1418
1419
1420
1421
1422
1423
1424
1425
1426
1427
1428
1429
1430
1431
1432
1433
1434
1435
1436
1437
1438
1439
1440
1441
1442
1443
1444
1445
1446
1447
1448
1449
1450
1451
1452
1453
1454
1455
1456
1457
1458
1459
1460
1461
1462
1463
1464
1465
1466
1467
1468
1469
1470
1471
1472
1473
1474
1475
1476
1477
1478
1479
1480
1481
1482
1483
1484
1485
1486
1487
1488
1489
1490
1491
1492
1493
1494
1495
1496
1497
1498
1499
1500
1501
1502
1503
1504
1505
1506
1507
1508
1509
1510
1511
1512
1513
1514
1515
1516
1517
1518
1519
1520
1521
1522
1523
1524
1525
1526
1527
1528
1529
1530
1531
1532
1533
1534
1535
1536
1537
1538
1539
1540
1541
1542
1543
1544
1545
1546
1547
1548
1549
1550
1551
1552
1553
1554
1555
1556
1557
1558
1559
1560
1561
1562
1563
1564
1565
1566
1567
1568
1569
1570
1571
1572
1573
1574
1575
1576
1577
1578
1579
1580
1581
1582
1583
1584
1585
1586
1587
1588
1589
1590
1591
1592
1593
1594
1595
1596
1597
1598
1599
1600
1601
1602
1603
1604
1605
1606
1607
1608
1609
1610
1611
1612
1613
1614
1615
1616
1617
1618
1619
1620
1621
1622
1623
1624
1625
1626
1627
1628
1629
1630
1631
1632
1633
1634
1635
1636
1637
1638
1639
1640
1641
1642
1643
1644
1645
1646
1647
1648
1649
1650
1651
1652
1653
1654
1655
1656
1657
1658
1659
1660
1661
1662
1663
1664
1665
1666
1667
1668
1669
1670
1671
1672
1673
1674
1675
1676
1677
1678
1679
1680
1681
1682
1683
1684
1685
1686
1687
1688
1689
1690
1691
1692
1693
1694
1695
1696
1697
1698
1699
1700
1701
1702
1703
1704
1705
1706
1707
1708
1709
1710
1711
1712
1713
1714
1715
1716
1717
1718
1719
1720
1721
1722
1723
1724
1725
1726
1727
1728
1729
1730
1731
1732
1733
1734
1735
1736
1737
1738
1739
1740
1741
1742
1743
1744
1745
1746
1747
1748
1749
1750
1751
1752
1753
1754
1755
1756
1757
1758
1759
1760
1761
1762
1763
1764
1765
1766
1767
1768
1769
1770
1771
1772
1773
1774
1775
1776
1777
1778
1779
1780
1781
1782
1783
1784
1785
1786
1787
1788
1789
1790
1791
1792
1793
1794
1795
1796
1797
1798
1799
1800
1801
1802
1803
1804
1805
1806
1807
1808
1809
1810
1811
1812
1813
1814
1815
1816
1817
1818
1819
1820
1821
1822
1823
1824
1825
1826
1827
1828
1829
1830
1831
1832
1833
1834
1835
1836
1837
1838
1839
1840
1841
1842
1843
1844
1845
1846
1847
1848
1849
1850
1851
1852
1853
1854
1855
1856
1857
1858
1859
1860
1861
1862
1863
1864
1865
1866
1867
1868
1869
1870
1871
1872
1873
1874
1875
1876
1877
1878
1879
1880
1881
1882
1883
1884
1885
1886
1887
1888
1889
1890
1891
1892
1893
1894
1895
1896
1897
1898
1899
1900
1901
1902
1903
1904
1905
1906
1907
1908
1909
1910
1911
1912
1913
1914
1915
1916
1917
1918
1919
1920
1921
1922
1923
1924
1925
1926
1927
1928
1929
1930
1931
1932
1933
1934
1935
1936
1937
1938
1939
1940
1941
1942
1943
1944
1945
1946
1947
1948
1949
1950
1951
1952
1953
1954
1955
1956
1957
1958
1959
1960
1961
1962
1963
1964
1965
1966
1967
1968
1969
1970
1971
1972
1973
1974
1975
1976
1977
1978
1979
1980
1981
1982
1983
1984
1985
1986
1987
1988
1989
1990
1991
1992
1993
1994
1995
1996
1997
1998
1999
2000
2001
2002
2003
2004
2005
2006
2007
2008
2009
2010
2011
2012
2013
2014
2015
2016
2017
2018
2019
2020
2021
2022
2023
2024
2025
2026
2027
2028
2029
2030
2031
2032
2033
2034
2035
2036
2037
2038
2039
2040
2041
2042
2043
2044
2045
2046
2047
2048
2049
2050
2051
2052
2053
2054
2055
2056
2057
2058
2059
2060
2061
2062
2063
2064
2065
2066
2067
2068
2069
2070
2071
2072
2073
2074
2075
2076
2077
2078
2079
2080
2081
2082
2083
2084
2085
2086
2087
2088
2089
2090
2091
2092
2093
2094
2095
2096
2097
2098
2099
2100
2101
2102
2103
2104
2105
2106
2107
2108
2109
2110
2111
2112
2113
2114
2115
2116
2117
2118
2119
2120
2121
2122
2123
2124
2125
2126
2127
2128
2129
2130
2131
2132
2133
2134
2135
2136
2137
2138
2139
2140
2141
2142
2143
2144
2145
2146
2147
2148
2149
2150
2151
2152
2153
2154
2155
2156
2157
2158
2159
2160
2161
2162
2163
2164
2165
2166
2167
2168
2169
2170
2171
2172
2173
2174
2175
2176
2177
2178
2179
2180
2181
2182
2183
2184
2185
2186
2187
2188
2189
2190
2191
2192
2193
2194
2195
2196
2197
2198
2199
2200
2201
2202
2203
2204
2205
2206
2207
2208
2209
2210
2211
2212
2213
2214
2215
2216
2217
2218
2219
2220
2221
2222
2223
2224
2225
2226
2227
2228
2229
2230
2231
2232
2233
2234
2235
2236
2237
2238
2239
2240
2241
2242
2243
2244
2245
2246
2247
2248
2249
2250
2251
2252
2253
2254
2255
2256
2257
2258
2259
2260
2261
2262
2263
2264
2265
2266
2267
2268
2269
2270
2271
2272
2273
2274
2275
2276
2277
2278
2279
2280
2281
2282
2283
2284
2285
2286
2287
2288
2289
2290
2291
2292
2293
2294
2295
2296
2297
2298
2299
2300
2301
2302
2303
2304
2305
2306
2307
2308
2309
2310
2311
2312
2313
2314
2315
2316
2317
2318
2319
2320
2321
2322
2323
2324
2325
2326
2327
2328
2329
2330
2331
2332
2333
2334
2335
2336
2337
2338
2339
2340
2341
2342
2343
2344
2345
2346
2347
2348
2349
2350
2351
2352
2353
2354
2355
2356
2357
2358
2359
2360
2361
2362
2363
2364
2365
2366
2367
2368
2369
2370
2371
2372
2373
2374
2375
2376
2377
2378
2379
2380
2381
2382
2383
2384
2385
2386
2387
2388
2389
2390
2391
2392
2393
2394
2395
2396
2397
2398
2399
2400
2401
2402
2403
2404
2405
2406
2407
2408
2409
2410
2411
2412
2413
2414
2415
2416
2417
2418
2419
2420
2421
2422
2423
2424
2425
2426
2427
2428
2429
2430
2431
2432
2433
2434
2435
2436
2437
2438
2439
2440
2441
2442
2443
2444
2445
2446
2447
2448
2449
2450
2451
2452
2453
2454
2455
2456
2457
2458
2459
2460
2461
2462
2463
2464
2465
2466
2467
2468
2469
2470
2471
2472
2473
2474
2475
2476
2477
2478
2479
2480
2481
2482
2483
2484
2485
2486
2487
2488
2489
2490
2491
2492
2493
2494
2495
2496
2497
2498
2499
2500
2501
2502
2503
2504
2505
2506
2507
2508
2509
2510
2511
2512
2513
2514
2515
2516
2517
2518
2519
2520
2521
2522
2523
2524
2525
2526
2527
2528
2529
2530
2531
2532
2533
2534
2535
2536
2537
2538
2539
2540
2541
2542
2543
2544
2545
2546
2547
2548
2549
2550
2551
2552
2553
2554
2555
2556
2557
2558
2559
2560
2561
2562
2563
2564
2565
2566
2567
2568
2569
2570
2571
2572
2573
2574
2575
2576
2577
2578
2579
2580
2581
2582
2583
2584
2585
2586
2587
2588
2589
2590
2591
2592
2593
2594
2595
2596
2597
2598
2599
2600
2601
2602
2603
2604
2605
2606
2607
2608
2609
2610
2611
2612
2613
2614
2615
2616
2617
2618
2619
2620
2621
2622
2623
2624
2625
2626
2627
2628
2629
2630
2631
2632
2633
2634
```

### 6.1.7 Writing drive parameters ► W\_Drive\_Param

Some drive parameters may only be changed with Torque Off or Drive in Halt. Moreover the new values must be entered in both Physical values and Drive internal values.

The table in annex gives the relevant information.

```

W_Gen_005 x
1  PROGRAM W_Gen_005
2  VAR
3      oW_Drive_Param          : W_Drive_Param;
4  END_VAR
5
6
7
8
9
10
11  oW_Drive_Param.CNCAdr      := ADR(NCK);
12  oW_Drive_Param.byDriveAdd  := 0;
13
14  oW_Drive_Param.diPinValue   := 0;
15  oW_Drive_Param.diPphValue   := 0;
16  oW_Drive_Param.dwPphValue   := 0;
17  oW_Drive_Param.rePphValue   := 0;
18  oW_Drive_Param.woPnumber    := 0;
19
20  oW_Drive_Param(enable := xEnable);
21
22  IF (oW_Drive_Param.uiResult = 0) THEN
23      xEnable := FALSE;
24  END_IF
25

```

## 6.1.8 Writing drive parameters ► W\_Drive\_Param\_Recursive

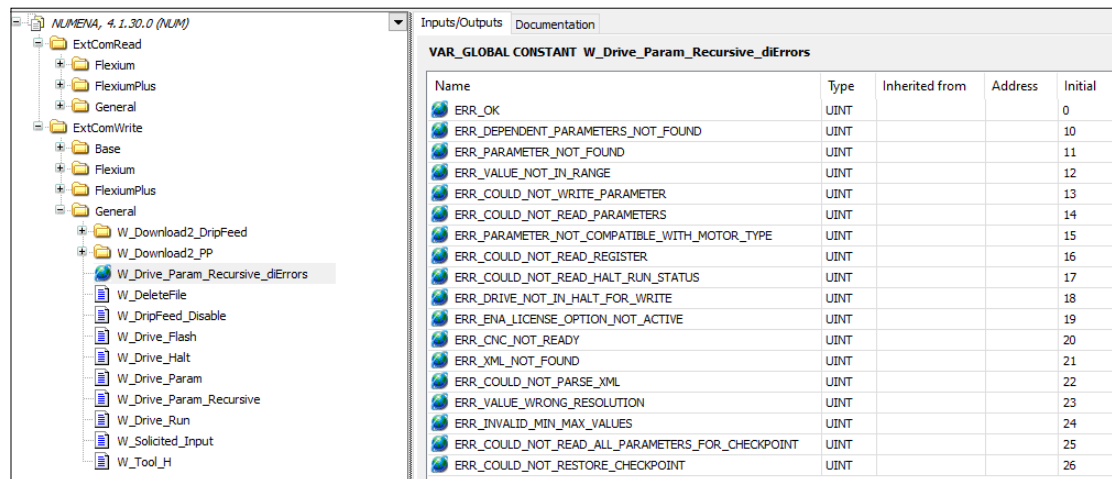
This request is an improved version of the previous W\_Drive\_Param.

The relevant differences are:

- The parameter value is required in Physical unit only. The request itself computes the conversion in Drive internal unit, as described in appendix A.3.1
- The request checks if the drive is in the status (Halt, Run) needed to accept the parameter value change related to the drive parameter number, as described in appendix A.3.1
- The request changes the values of the parameters depending from the parameter selected into **.Pnumber** input variable. as described in appendix A.3.2
- The request checks if the new parameter values of all the parameters involved full fill the limits as described in appendix A.3.3. If one or more value will go out from that limits the request will automatically adapt them.

In case of error a value will be put into the out variable **.diError**

The meaning of such error is described into the **VAR\_GLOBAL\_CONSTANT W\_Drive\_Param\_Recursive\_diErrors** section.



| VAR_GLOBAL_CONSTANT W_Drive_Param_Recursive_diErrors |      |                |         |         |  |
|------------------------------------------------------|------|----------------|---------|---------|--|
| Name                                                 | Type | Inherited from | Address | Initial |  |
| ERR_OK                                               | UINT |                |         | 0       |  |
| ERR_DEPENDENT_PARAMETERS_NOT_FOUND                   | UINT |                |         | 10      |  |
| ERR_PARAMETER_NOT_FOUND                              | UINT |                |         | 11      |  |
| ERR_VALUE_NOT_IN_RANGE                               | UINT |                |         | 12      |  |
| ERR_COULD_NOT_WRITE_PARAMETER                        | UINT |                |         | 13      |  |
| ERR_COULD_NOT_READ_PARAMETERS                        | UINT |                |         | 14      |  |
| ERR_PARAMETER_NOT_COMPATIBLE_WITH_MOTOR_TYPE         | UINT |                |         | 15      |  |
| ERR_COULD_NOT_READ_REGISTER                          | UINT |                |         | 16      |  |
| ERR_COULD_NOT_READ_HALT_RUN_STATUS                   | UINT |                |         | 17      |  |
| ERR_DRIVE_NOT_IN_HALT_FOR_WRITE                      | UINT |                |         | 18      |  |
| ERR_ENA_LICENSE_OPTION_NOT_ACTIVE                    | UINT |                |         | 19      |  |
| ERR_CNC_NOT_READY                                    | UINT |                |         | 20      |  |
| ERR_XML_NOT_FOUND                                    | UINT |                |         | 21      |  |
| ERR_COULD_NOT_PARSE_XML                              | UINT |                |         | 22      |  |
| ERR_VALUE_WRONG_RESOLUTION                           | UINT |                |         | 23      |  |
| ERR_INVALID_MIN_MAX_VALUES                           | UINT |                |         | 24      |  |
| ERR_COULD_NOT_READ_ALL_PARAMETERS_FOR_CHECKPOINT     | UINT |                |         | 25      |  |
| ERR_COULD_NOT_RESTORE_CHECKPOINT                     | UINT |                |         | 26      |  |

```

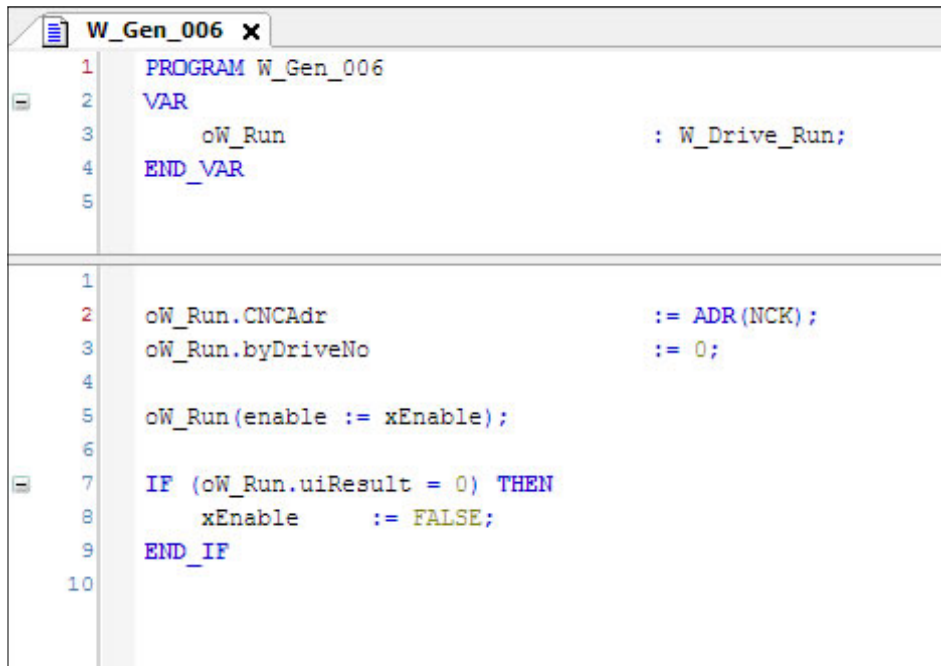
1  PROGRAM W_Gen_011
2  VAR
3      oW_Drive_Param_Recursive   : W_Drive_Param_Recursive;
4      DriveAddress               : BYTE;
5      DriveParameterNumber       : WORD;
6      ParameterValue             : LREAL;
7      Error                     : DINT;
8  END VAR

1
2  DriveAddress:= 0;
3  DriveParameterNumber:= 91;
4  ParameterValue:= 300.0;
5
6  IF xEnable THEN
7      oW_Drive_Param_Recursive(CNCAdr:=ADR(NCK),
8          byDriveAdd:= DriveAddress,
9          woPnumber:= DriveParameterNumber,
10         value:= ParameterValue,
11         boIsDriveX:= TRUE);
12  IF oW_Drive_Param_Recursive.uiResult = 0 THEN
13      Error:= oW_Drive_Param_Recursive.diError;
14      xEnable:= FALSE;
15  END_IF
16  END_IF

```

### 6.1.9 Setting the drive in run ► W\_Drive\_Run

This request is intended to restart a drive that was put in Halt.  
The programmer must take all precautions to prevent any incident or accident.



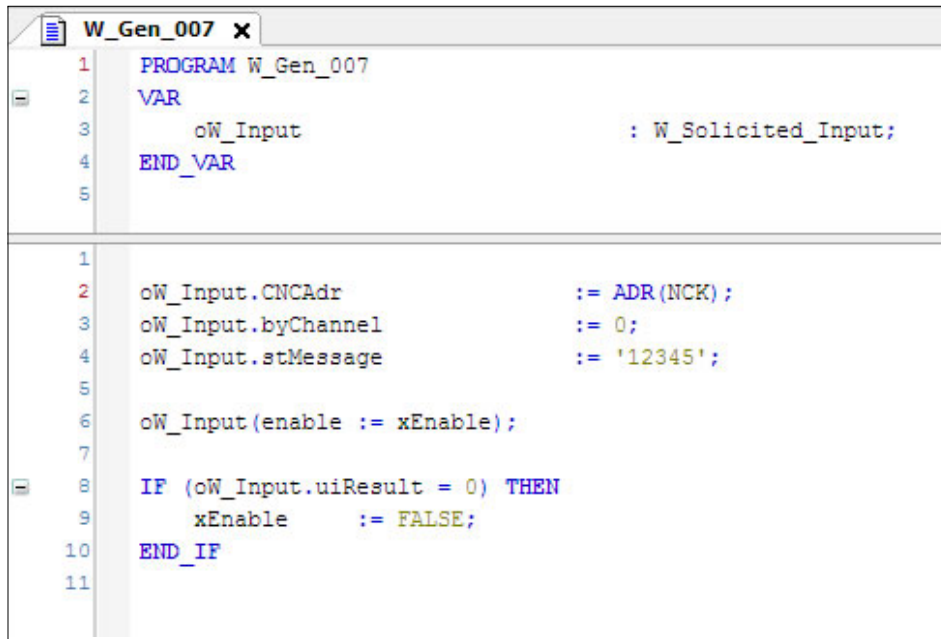
```
1  PROGRAM W_Gen_006
2  VAR
3      oW_Run                               : W_Drive_Run;
4  END_VAR
5
6
7  oW_Run.CNCAdr                           := ADR(NCK);
8  oW_Run.byDriveNo                         := 0;
9
10 oW_Run(enable := xEnable);
11
12 IF (oW_Run.uiResult = 0) THEN
13     xEnable := FALSE;
14 END_IF
```

### 6.1.10 Answering to a solicited input ► **W\_Solicited\_Input**

This request allows for sending a text in the Operator's dialog line or answering a part program request (\$0).

Writing *FF* in the channel number accesses the currently displayed channel.  
In Sequence number search and common groups, the channel number is not used.

Important: In *MDI* the request *R\_Solicited\_Input* should be sent first.



```

1  PROGRAM W_Gen_007
2  VAR
3      oW_Input                : W_Solicited_Input;
4  END_VAR
5
6  oW_Input.CNCAdr             := ADR(NCK);
7  oW_Input.byChannel          := 0;
8  oW_Input.stMessage          := '12345';
9
10 oW_Input(enable := xEnable);
11
12 IF (oW_Input.uiResult = 0) THEN
13     xEnable := FALSE;
14 END_IF

```

### 6.1.11 Writing tool H data ► W\_Tool\_H

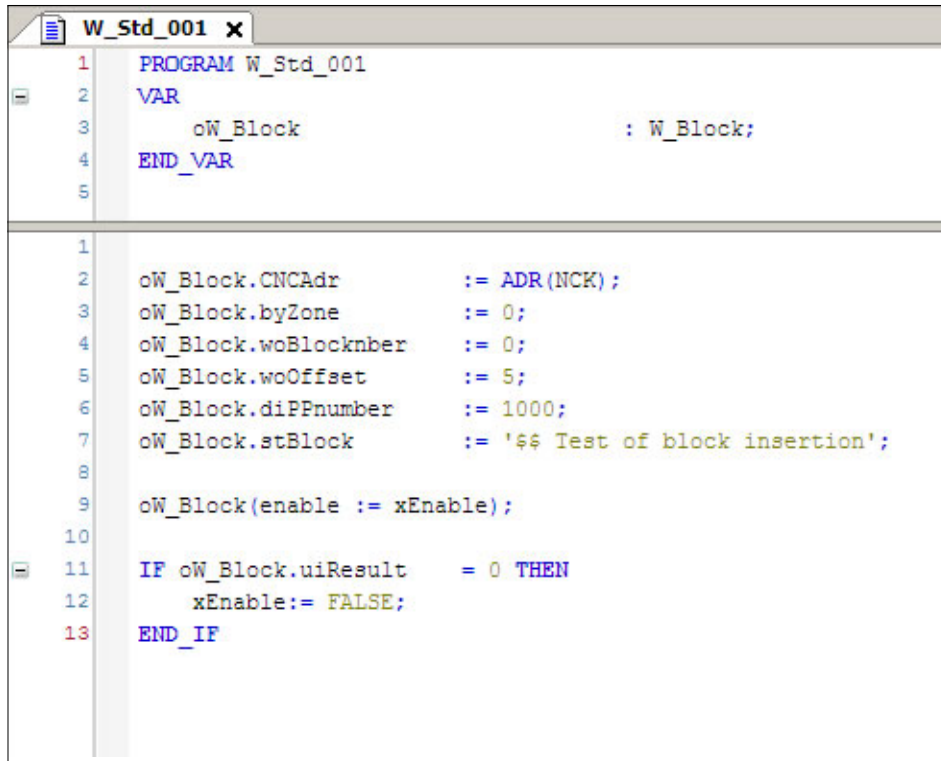
This request will enter the H value of up to 10 tool offsets. The data are NCK in internal units.

```
W_Gen_008 x
1  PROGRAM W_Gen_008
2  VAR
3      oW_Tool_H                : W_Tool_H;
4  END_VAR
5
6
7
8
9
10
11
12
13
14
15  oW_Tool_H.CNCAdr             := ADR(NCK);
16
17  oW_Tool_H.byH_No             := 1;
18
19  oW_Tool_H.diE56_n0           := 100;
20
21  oW_Tool_H.diE56_n1           := 200;
22
23  oW_Tool_H.diE56_n2           := 300;
24
25  oW_Tool_H.diE56_n3           := 400;
26
27  oW_Tool_H.diE56_n4           := 500;
28
29  oW_Tool_H.diE56_n5           := 600;
30
31  oW_Tool_H.diE56_n6           := 700;
32
33  oW_Tool_H.diE56_n7           := 800;
34
35  oW_Tool_H.diE56_n8           := 900;
36
37  oW_Tool_H.diE56_n9           := 999;
38
39
40  oW_Tool_H(enable := xEnable);
41
42  IF (oW_Tool_H.uiResult = 0) THEN
43      xEnable := FALSE;
44  END_IF
```

## 6.2 Flexium requests

### 6.2.1 Writing a part program block ► [W\\_Block](#)

This request allows for writing an NC block. It does not insert a new block just replaces the existing values.



```
1 PROGRAM W_Std_001
2 VAR
3     oW_Block                : W_Block;
4 END_VAR
5
6
7
8
9
10
11 oW_Block.CNCAdr             := ADR(NCK);
12 oW_Block.byZone             := 0;
13 oW_Block.wcBlocknber        := 0;
14 oW_Block.wcOffset           := 5;
15 oW_Block.diPPnumber         := 1000;
16 oW_Block.stBlock            := '$$ Test of block insertion';
17
18 oW_Block(enable := xEnable);
19
20
21 IF oW_Block.uiResult         = 0 THEN
22     xEnable:= FALSE;
23 END_IF
```

## 6.2.2 Writing an axis measure correction ▶ [W\\_Corr\\_Dyn](#)

This request allows for writing into the parameter E952xx that can be only reached by Dynamic operator or ENA.

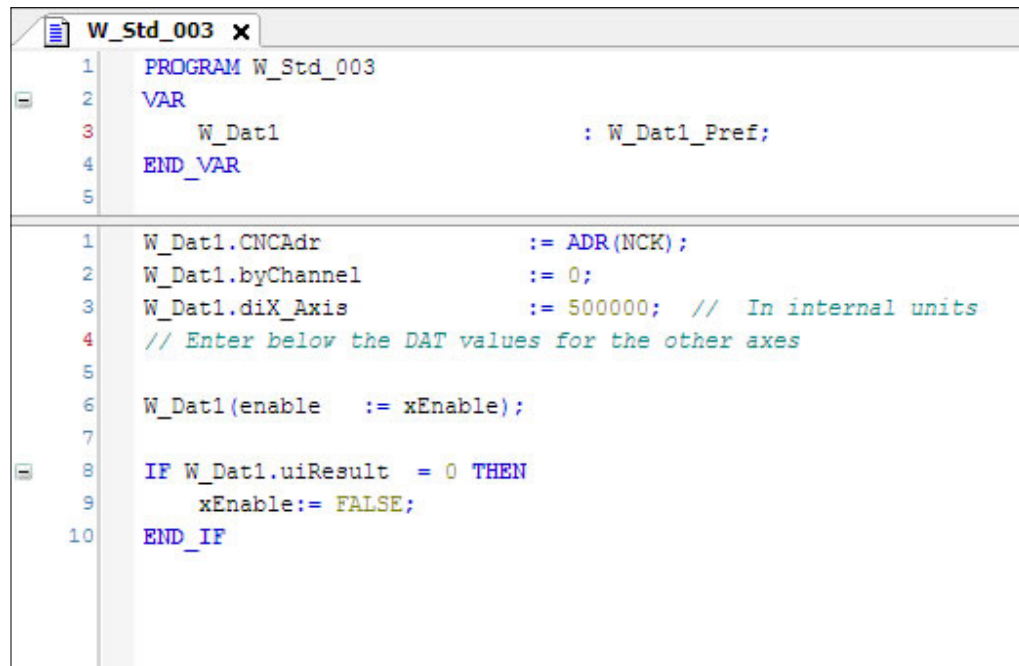
```
W_Std_002 x
1  PROGRAM W_Std_002
2  VAR
3      oW_Corr_Dyn          : W_Corr_Dyn;  // Write into E952xx
4  END_VAR
5
6
7
8
9
10 oW_Corr_Dyn.CNCAdr      := ADR(NCK);
11 oW_Corr_Dyn.byAxisNo    := 0;
12 oW_Corr_Dyn.diCorrDyn   := 123;
13
14 oW_Corr_Dyn(enable := xEnable);
15
16 IF oW_Corr_Dyn.uiResult = 0 THEN
17     xEnable:= FALSE;
18 END_IF
```

### 6.2.3 Writing DAT1 values ► [W\\_Dat1\\_Pref](#)

This request allows for writing the values of the current DAT1 of a particular channel.

It is strongly recommended to run this request in Reset status as the new values will not be taken into account immediately to prevent an incident but after an M0 / M2 or a block holding preparation for example M1 (validated or not) or a writing operation into an E parameter.

As this request writes the values for all axes of the channel, be sure to fill in the data for all these axes. Failure to do this will enter erroneous values for the non defined addresses.



```

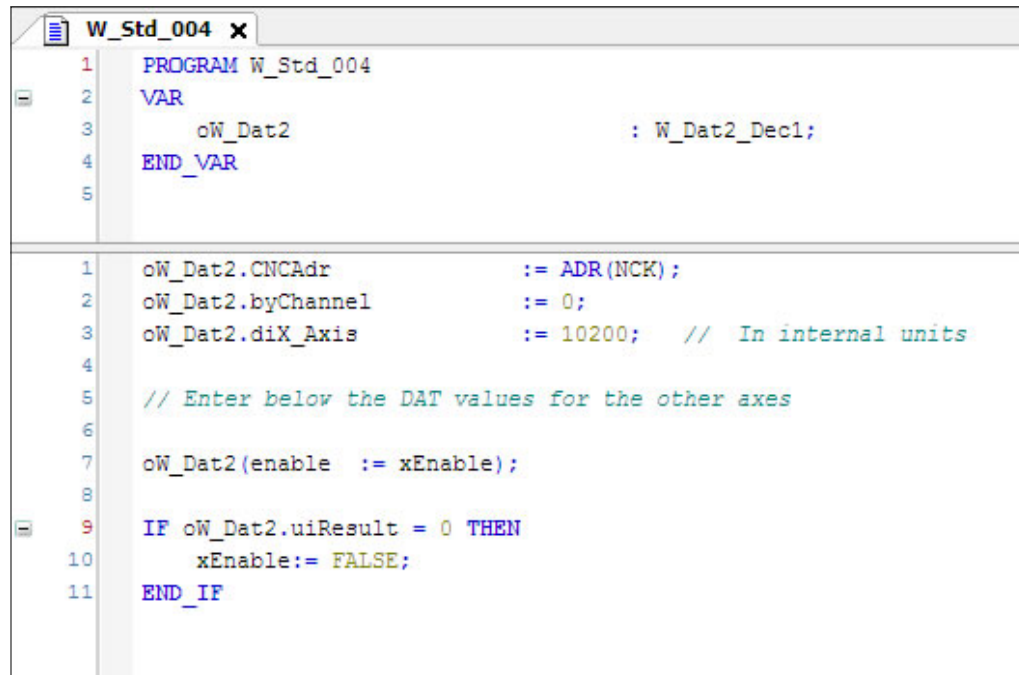
1  PROGRAM W_Std_003
2  VAR
3      W_Dat1                      : W_Dat1_Pref;
4  END_VAR
5
6  W_Dat1.CNCAdr                   := ADR(NCK);
7  W_Dat1.byChannel                 := 0;
8  W_Dat1.diX_Axis                 := 500000; // In internal units
9  // Enter below the DAT values for the other axes
10
11  W_Dat1(enable := xEnable);
12
13  IF W_Dat1.uiResult = 0 THEN
14      xEnable := FALSE;
15  END_IF
  
```

## 6.2.4 Writing DAT2 values ► W\_Dat2\_Dec1

This request allows for writing the values of the current DAT2 of a particular channel.

It is strongly recommended to run this request in Reset status as the new values will not be taken into account immediately but at the next M0 / M1 M2 or E parameter write operation.

As this request writes the values for all axes of the channel, be sure to fill in the data for all these axes. Failure to do this will enter erroneous values for the non defined addresses.



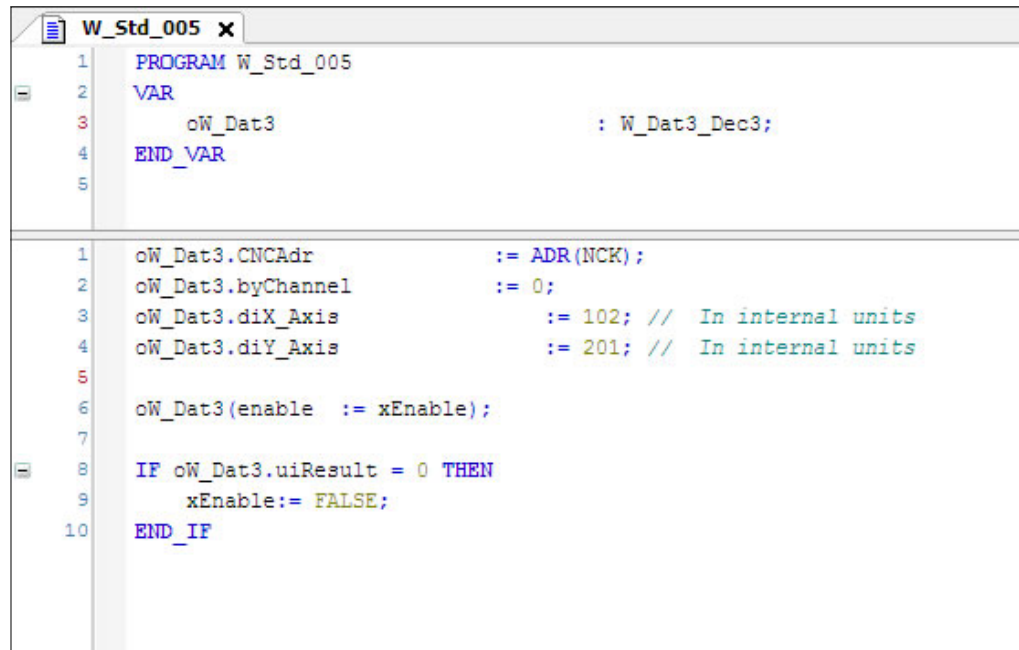
```
1 PROGRAM W_Std_004
2 VAR
3     oW_Dat2                               : W_Dat2_Dec1;
4 END_VAR
5
6
7 oW_Dat2.CNCAdr      := ADR(NCK);
8 oW_Dat2.byChannel   := 0;
9 oW_Dat2.diX_Axis    := 10200; // In internal units
10
11 // Enter below the DAT values for the other axes
12
13 oW_Dat2(enable := xEnable);
14
15 IF oW_Dat2.uiResult = 0 THEN
16     xEnable := FALSE;
17 END_IF
```

### 6.2.5 Writing DAT3 values ► W\_Dat3\_Dec3

This request allows for writing the values of the current DAT3 of a particular channel.

It is strongly recommended to run this request in Reset status as the new values will not be taken into account immediately but at the next M0 / M1 M2 or E parameter write operation.

As this request writes the values for all axes of the channel, be sure to fill in the data for all these axes. Failure to do this will enter erroneous values for the non defined addresses.



```

1  PROGRAM W_Std_005
2  VAR
3      oW_Dat3                      : W_Dat3_Dec3;
4  END_VAR
5
6  oW_Dat3.CNCAdr                  := ADR(NCK);
7  oW_Dat3.byChannel                := 0;
8  oW_Dat3.diX_Axis                 := 102; // In internal units
9  oW_Dat3.diY_Axis                 := 201; // In internal units
10
11  oW_Dat3(enable := xEnable);
12
13  IF oW_Dat3.uiResult = 0 THEN
14      xEnable:= FALSE;
15  END_IF

```

## Writing an E8... parameter value ▶ W\_E8xyyy

This request allows for writing one E8xyyy parameter.  
The integer input argument will be written in the parameter.

```

1  PROGRAM W_Std_006
2  VAR
3      oW_E8xyyy                : W_E8xyyy;
4  END_VAR
5
6
7
8  oW_E8xyyy.CNCAdr              := ADR(NCK);
9  oW_E8xyyy.udE8xyyyNo         := 83004;
10 oW_E8xyyy.diE8_n0            := 1250;
11
12 oW_E8xyyy(enable := xEnable);
13
14 IF oW_E8xyyy.uiResult = 0 THEN
15     xEnable:= FALSE;
16 END_IF

```

### 6.2.7 Writing an E5 or E6 parameter value ► [W\\_EParam\\_5\\_6](#)

This request allows for writing one parameter of the series E5 or E6. Only the integer part of the input argument will be taken into account as internal units.

```

1  PROGRAM W_Std_007
2  VAR
3      oW_E_5_6                : W_E_Param_5_6;
4      Value                   : LREAL;
5  END_VAR
6
7
8  oW_E_5_6.CNCAdr             := ADR(NCK);
9  oW_E_5_6.byChannel          := 0;
10
11  oW_E_5_6.udE_number         := 50001;
12  oW_E_5_6.diValue_E         := 25000;
13
14  IF oW_E_5_6.uiResult        = 0 THEN
15      xEnable:= FALSE;
16  END_IF

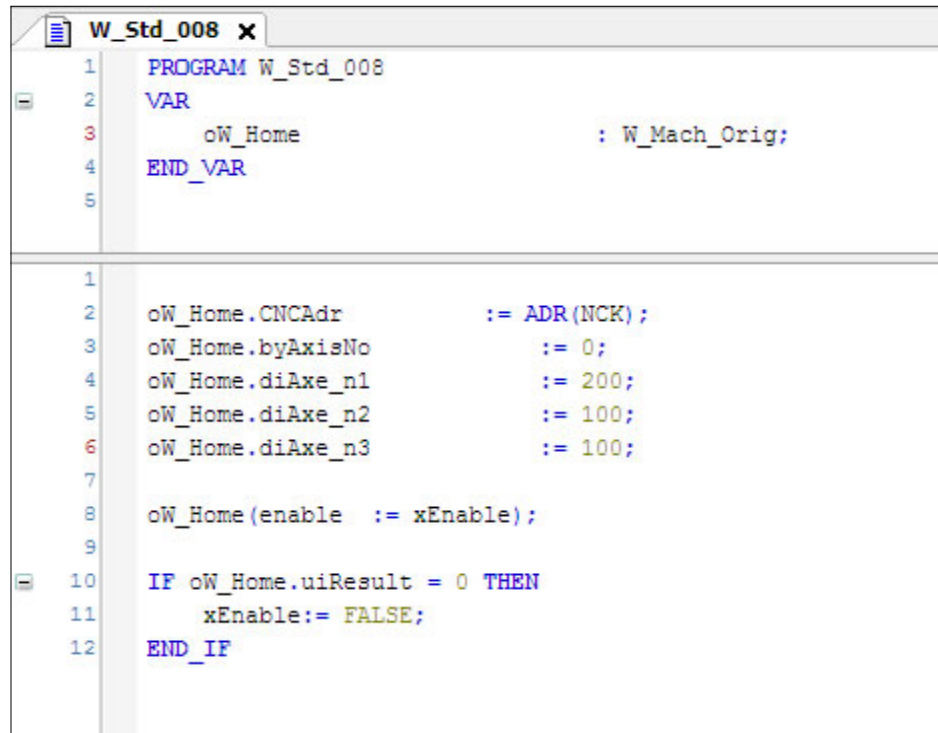
```

### 6.2.8 Writing the home position (P16) ► [W\\_Mach\\_Orig](#)

This request allows for writing the home position for an axis (equivalent of P16).

Please note that this value is not written permanently in the parameter. The value indicated in the project will be taken at the next system start-up.

The request writes the values at three consecutive axes addresses. Therefore be sure to enter these three values.

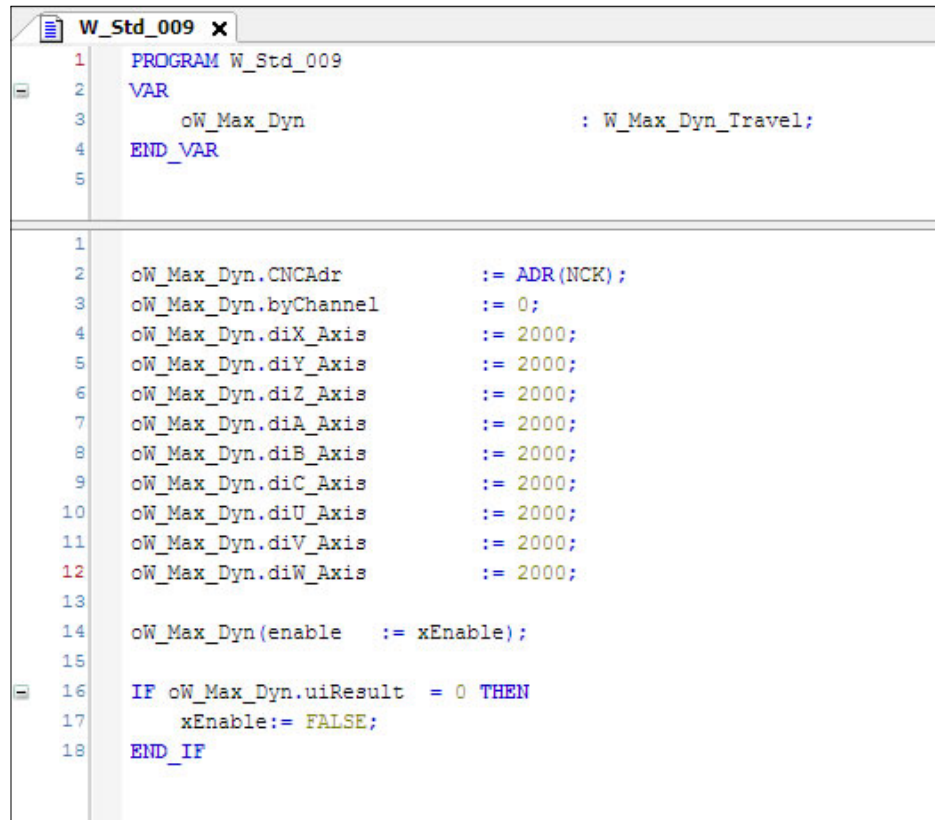


```
1  PROGRAM W_Std_008
2  VAR
3      oW_Home                : W_Mach_Orig;
4  END_VAR
5
6
7
8
9
10 oW_Home.CNCAdr              := ADR(NCK);
11 oW_Home.byAxisNo            := 0;
12 oW_Home.diAxe_n1            := 200;
13 oW_Home.diAxe_n2            := 100;
14 oW_Home.diAxe_n3            := 100;
15
16 oW_Home(enable := xEnable);
17
18 IF oW_Home.uiResult = 0 THEN
19     xEnable := FALSE;
20 END_IF
```

## 6.2.9 Writing Maximum Dynamic Travel values ► [W\\_Max\\_Dyn\\_Travel](#)

This request allows for writing the maximum dynamic travels of the axes in a particular channel. The values are given in reference to the axes names.

As this request writes the values for all axes of the channel, be sure to fill in the data for all these axes. Failure to do this will enter erroneous values for the non defined addresses.



```

1  PROGRAM W_Std_009
2  VAR
3      oW_Max_Dyn                : W_Max_Dyn_Travel;
4  END_VAR
5
6
7
8
9
10
11
12
13
14  oW_Max_Dyn.CNCAdr             := ADR(NCK);
15
16  oW_Max_Dyn.byChannel           := 0;
17  oW_Max_Dyn.diX_Axis            := 2000;
18  oW_Max_Dyn.diY_Axis            := 2000;
19  oW_Max_Dyn.diZ_Axis            := 2000;
20  oW_Max_Dyn.diA_Axis            := 2000;
21  oW_Max_Dyn.diB_Axis            := 2000;
22  oW_Max_Dyn.diC_Axis            := 2000;
23  oW_Max_Dyn.diU_Axis            := 2000;
24  oW_Max_Dyn.diV_Axis            := 2000;
25  oW_Max_Dyn.diW_Axis            := 2000;
26
27  oW_Max_Dyn(enable := xEnable);
28
29  IF oW_Max_Dyn.uiResult = 0 THEN
30      xEnable:= FALSE;
31  END_IF

```

## 6.2.10 Writing Maximum Dynamic Travel selected ► W\_Max\_Dyn\_Travel\_2

This request allows for writing the maximum dynamic travels of the axes in a particular channel. The values are given in reference to the axes names.

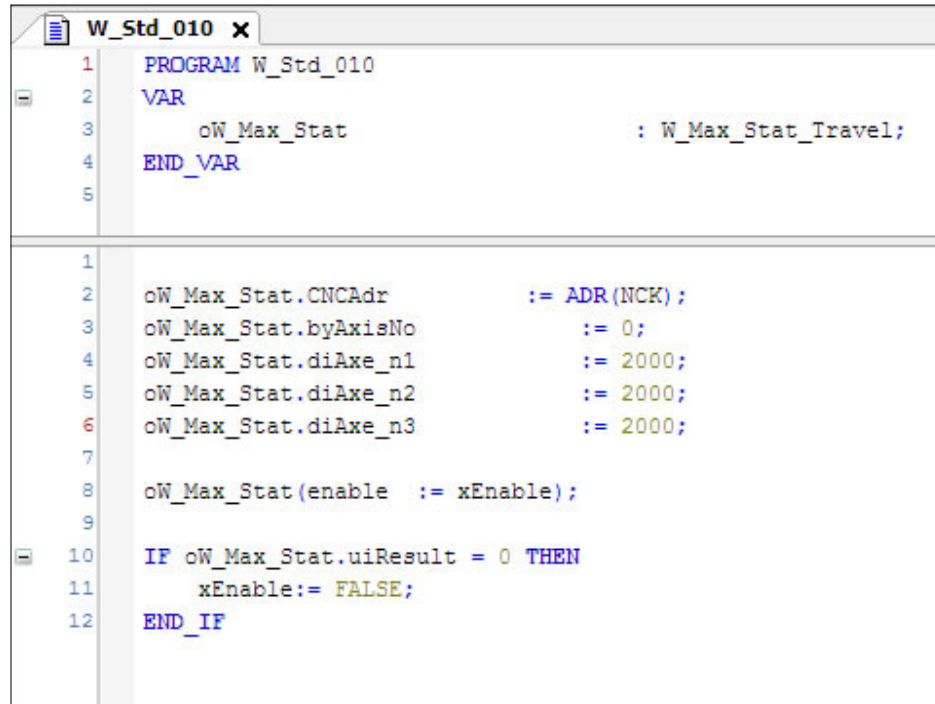
Respect to the previous **W\_Max\_Dyn\_Travel** request, this request writes the values for the axes selected with the input variable **.boUpdate\_n\_Axis:=TRUE**, so only the needed change can be written.

```
W_Std_015 x
1  PROGRAM W_Std_015
2  VAR
3      oW_Max_Dyn_Travel_2 : W_Max_Dyn_Travel_2;
4      byChannel            : BYTE;
5      diY_Axis             : DINT:= -20000;
6      diU_Axis             : DINT:= -40000;
7      diA_Axis             : DINT:= -70000;
8  END_VAR
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23  byChannel:= 0; // 0...7
24
25  oW_Max_Dyn_Travel_2.CNCAdr:= ADR(NCK);
26  oW_Max_Dyn_Travel_2.byChannel:= byChannel;
27
28  oW_Max_Dyn_Travel_2.diY_Axis:= diY_Axis;
29  oW_Max_Dyn_Travel_2.boUpdate_Y_Axis:= TRUE;
30  oW_Max_Dyn_Travel_2.diU_Axis:= diU_Axis;
31  oW_Max_Dyn_Travel_2.boUpdate_U_Axis:= TRUE;
32  oW_Max_Dyn_Travel_2.diA_Axis:= diA_Axis;
33  oW_Max_Dyn_Travel_2.boUpdate_A_Axis:= TRUE;
34
35  oW_Max_Dyn_Travel_2.boUpdate_X_Axis:= FALSE;
36  oW_Max_Dyn_Travel_2.boUpdate_Z_Axis:= FALSE;
37  oW_Max_Dyn_Travel_2.boUpdate_V_Axis:= FALSE;
38  oW_Max_Dyn_Travel_2.boUpdate_W_Axis:= FALSE;
39  oW_Max_Dyn_Travel_2.boUpdate_B_Axis:= FALSE;
40  oW_Max_Dyn_Travel_2.boUpdate_C_Axis:= FALSE;
41
42  oW_Max_Dyn_Travel_2(enable:= xEnable);
43
44  IF oW_Max_Dyn_Travel_2.uiResult = 0 THEN
45      xEnable:= FALSE;
46  END_IF
```

### 6.2.11 Writing Maximum Static Travel values ► [W\\_Max\\_Stat\\_Travel](#)

This request allows for writing the maximum static travels of axes.

The values are given for three consecutive axes addresses. Therefore be sure to enter these three values.



```

1  PROGRAM W_Max_Stat_Travel
2  VAR
3      oW_Max_Stat          : W_Max_Stat_Travel;
4  END_VAR
5
6
7
8
9
10 oW_Max_Stat.CNCAdr      := ADR(NCK);
11 oW_Max_Stat.byAxisNo    := 0;
12 oW_Max_Stat.diAxe_n1    := 2000;
13 oW_Max_Stat.diAxe_n2    := 2000;
14 oW_Max_Stat.diAxe_n3    := 2000;
15
16 oW_Max_Stat(enable := xEnable);
17
18 IF oW_Max_Stat.uiResult = 0 THEN
19     xEnable := FALSE;
20 END_IF
  
```

## 6.2.12 Writing Minimum Dynamic Travel values ► [W\\_Min\\_Dyn\\_Travel](#)

This request allows for writing the minimum dynamic travels of the axes in a particular channel. The values are given in reference to the axes names.

As this request writes the values for all axes of the channel, be sure to fill in the data for all these axes. Failure to do this will enter erroneous values for the non defined addresses.

```
W_Std_011 x
1 PROGRAM W_Std_011
2 VAR
3     oW_Min_Dyn                : W_Min_Dyn_Travel;
4 END_VAR
5
6
7
8
9
10
11
12
13
14 oW_Min_Dyn.CNCAdr             := ADR(NCK);
15 oW_Min_Dyn.byChannel          := 0;
16 oW_Min_Dyn.diX_Axis           := 2000;
17 oW_Min_Dyn.diY_Axis           := 2000;
18 oW_Min_Dyn.diZ_Axis           := 2000;
19 oW_Min_Dyn.diA_Axis           := 2000;
20 oW_Min_Dyn.diB_Axis           := 2000;
21 oW_Min_Dyn.diC_Axis           := 2000;
22 oW_Min_Dyn.diU_Axis           := 2000;
23 oW_Min_Dyn.diV_Axis           := 2000;
24 oW_Min_Dyn.diW_Axis           := 2000;
25
26 oW_Min_Dyn(enable := xEnable);
27
28 IF oW_Min_Dyn.uiResult = 0 THEN
29     xEnable:= FALSE;
30 END_IF
```

## 6.2.13 Writing Minimum Dynamic Travel selected ► [W\\_Min\\_Dyn\\_Travel\\_2](#)

This request allows for writing the minimum dynamic travels of the axes in a particular channel. The values are given in reference to the axes names.

Respect to the previous [W\\_Min\\_Dyn\\_Travel](#) request, this request writes the values for the axes selected with the input variable **.boUpdate\_n\_Axis:=TRUE**, so only the needed change can be written.

```

1  PROGRAM W_Std_016
2  VAR
3      oW_Min_Dyn_Travel_2 : W_Min_Dyn_Travel_2;
4      byChannel            : BYTE;
5      diY_Axis            : DINT:=-20000;
6      diU_Axis            : DINT:=-40000;
7      diA_Axis            : DINT:=-70000;
8  END_VAR
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24
25
26

```

```

1
2  byChannel:= 0; // 0...7
3
4  oW_Min_Dyn_Travel_2.CNCAdr:= ADR(NCK);
5  oW_Min_Dyn_Travel_2.byChannel:= byChannel;
6
7  oW_Min_Dyn_Travel_2.diY_Axis:= diY_Axis;
8  oW_Min_Dyn_Travel_2.boUpdate_Y_Axis:= TRUE;
9  oW_Min_Dyn_Travel_2.diU_Axis:= diU_Axis;
10 oW_Min_Dyn_Travel_2.boUpdate_U_Axis:= TRUE;
11 oW_Min_Dyn_Travel_2.diA_Axis:= diA_Axis;
12 oW_Min_Dyn_Travel_2.boUpdate_A_Axis:= TRUE;
13
14 oW_Min_Dyn_Travel_2.boUpdate_X_Axis:= FALSE;
15 oW_Min_Dyn_Travel_2.boUpdate_Z_Axis:= FALSE;
16 oW_Min_Dyn_Travel_2.boUpdate_V_Axis:= FALSE;
17 oW_Min_Dyn_Travel_2.boUpdate_W_Axis:= FALSE;
18 oW_Min_Dyn_Travel_2.boUpdate_B_Axis:= FALSE;
19 oW_Min_Dyn_Travel_2.boUpdate_C_Axis:= FALSE;
20
21 oW_Min_Dyn_Travel_2(enable:= xEnable);
22
23 IF oW_Min_Dyn_Travel_2.uiResult = 0 THEN
24     xEnable:= FALSE;
25 END_IF
26

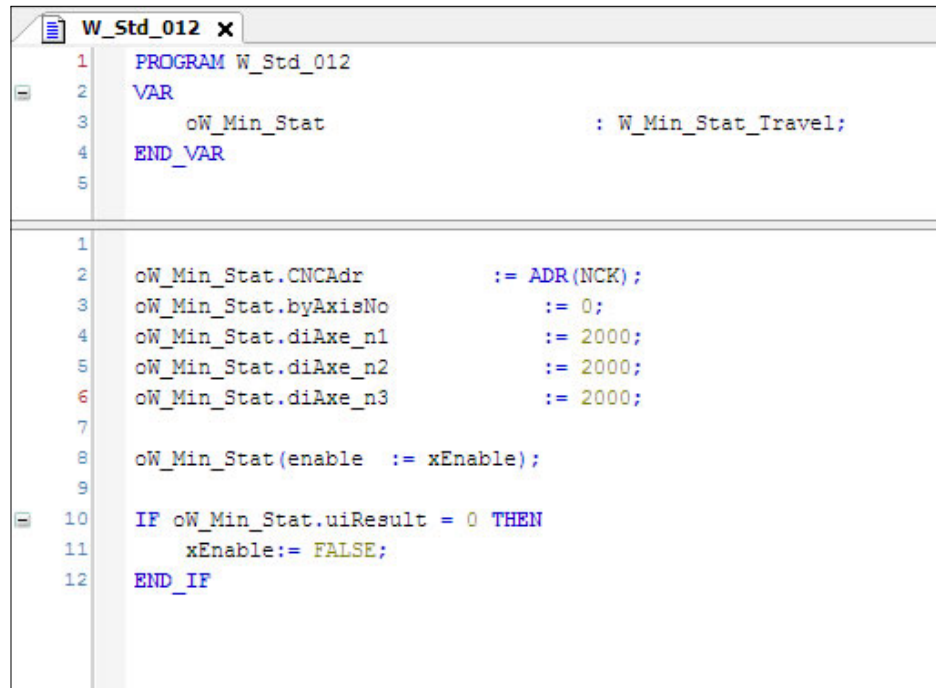
```

---

## 6.2.14 Writing Minimum Static Travel values ► [W\\_Min\\_Stat\\_Travel](#)

This request allows for writing the minimum static travels of axes.

The values are given for three consecutive axes addresses. Therefore be sure to enter all these three values.



```
W_Min_Stat x
1  PROGRAM W_Min_Stat
2  VAR
3      oW_Min_Stat          : W_Min_Stat_Travel;
4  END_VAR
5
6
7
8
9
10 oW_Min_Stat.CNCAdr      := ADR(NCK);
11 oW_Min_Stat.byAxisNo    := 0;
12 oW_Min_Stat.diAxe_n1    := 2000;
13 oW_Min_Stat.diAxe_n2    := 2000;
14 oW_Min_Stat.diAxe_n3    := 2000;
15
16 oW_Min_Stat(enable := xEnable);
17
18 IF oW_Min_Stat.uiResult = 0 THEN
19     xEnable := FALSE;
20 END_IF
```

### 6.2.15 Writing an NCK parameter ► W\_NCK\_Par

This request allows for writing a word of a particular machine parameter. The word number is given by the property *uiPoffset*.

Please refer to the annex A4 for the list of parameters that are taken into account immediately.

The change of other parameters has no effect as a restart will use the parameters value defined in the application. Be cautious when modifying a parameter as some others may depend on it.

```

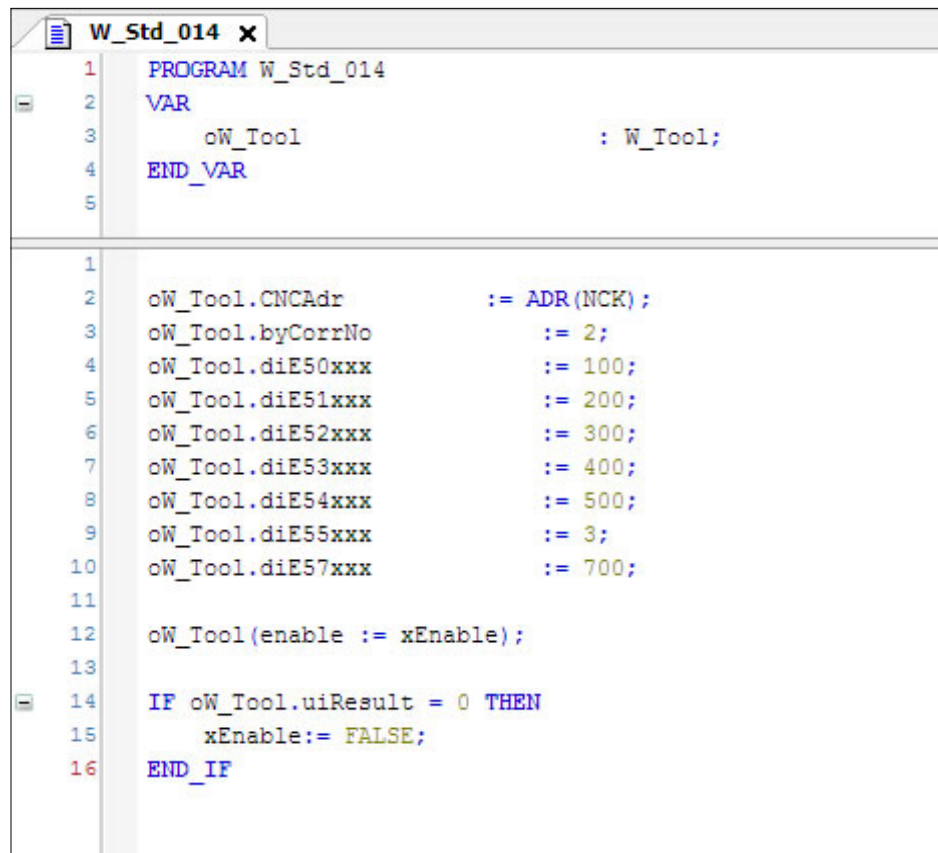
1  PROGRAM W_Std_013
2  VAR
3      oW_Nck_Par                : W_Nck_Par;
4  END_VAR
5
6
7  oW_Nck_Par.CNCAdr             := ADR(NCK);
8  oW_Nck_Par.uiPnumber          := 30;
9  oW_Nck_Par.uiPoffset          := 5;
10 oW_Nck_Par.diPvalue           := 5500;
11
12 oW_Nck_Par(enable := xEnable);
13
14 IF oW_Nck_Par.uiResult = 0 THEN
15     xEnable:= FALSE;
16 END_IF

```

## 6.2.16 Writing a tool offset ► W\_Tool

This request allows for writing a particular tool offset.

All components are written with the exception of the H value which is given by a separate request.

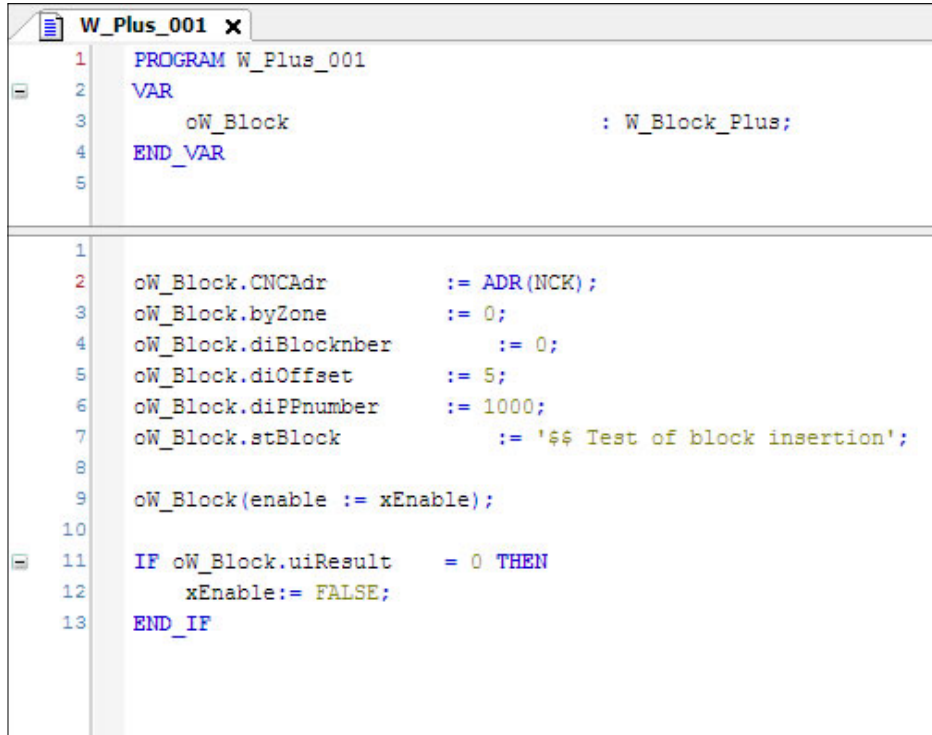


```
1  PROGRAM W_Std_014
2  VAR
3      oW_Tool          : W_Tool;
4  END_VAR
5
6
7
8
9
10
11
12  oW_Tool.CNCAdr      := ADR(NCK);
13
14  oW_Tool.byCorrNo    := 2;
15  oW_Tool.diE50xxx    := 100;
16  oW_Tool.diE51xxx    := 200;
17  oW_Tool.diE52xxx    := 300;
18  oW_Tool.diE53xxx    := 400;
19  oW_Tool.diE54xxx    := 500;
20  oW_Tool.diE55xxx    := 3;
21  oW_Tool.diE57xxx    := 700;
22
23  oW_Tool(enable := xEnable);
24
25  IF oW_Tool.uiResult = 0 THEN
26      xEnable:= FALSE;
27  END_IF
```

## 6.3 Flexium+ requests

### 6.3.1 Writing a part program block ► [W\\_Block\\_Plus](#)

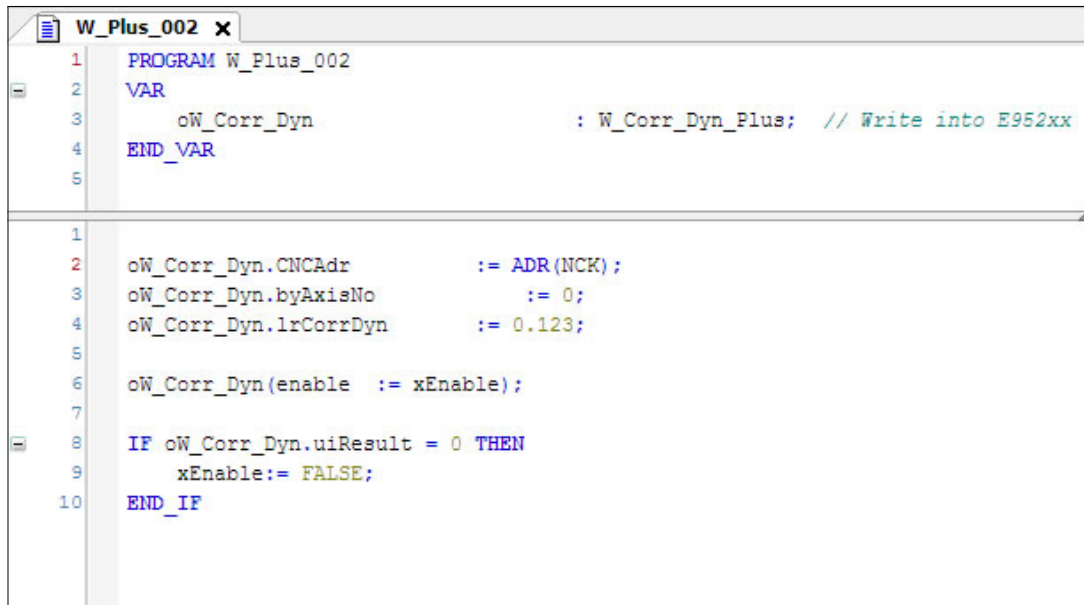
This request allows for writing an NC block.  
It does not insert a new block just replaces the existing values.



```
1 PROGRAM W_Plus_001
2 VAR
3     oW_Block : W_Block_Plus;
4 END_VAR
5
6
7
8
9
10
11 oW_Block.CNCAdr := ADR(NCK);
12 oW_Block.byZone := 0;
13 oW_Block.diBlocknber := 0;
14 oW_Block.diOffset := 5;
15 oW_Block.diPPnumber := 1000;
16 oW_Block.stBlock := '$$ Test of block insertion';
17
18 oW_Block(enable := xEnable);
19
20 IF oW_Block.uiResult = 0 THEN
21     xEnable:= FALSE;
22 END_IF
```

### 6.3.2 Writing an axis measure correction ► [W\\_Corr\\_Dyn\\_Plus](#)

This request allows for writing into the parameter E952xx that can be only reached by Dynamic operator or ENA.



```
W_Plus_002 x
1  PROGRAM W_Plus_002
2  VAR
3      oW_Corr_Dyn          : W_Corr_Dyn_Plus; // Write into E952xx
4  END_VAR
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24
25
26
27
28
29
30
31
32
33
34
35
36
37
38
39
40
41
42
43
44
45
46
47
48
49
50
51
52
53
54
55
56
57
58
59
60
61
62
63
64
65
66
67
68
69
70
71
72
73
74
75
76
77
78
79
80
81
82
83
84
85
86
87
88
89
90
91
92
93
94
95
96
97
98
99
100
101
102
103
104
105
106
107
108
109
110
111
112
113
114
115
116
117
118
119
120
121
122
123
124
125
126
127
128
129
130
131
132
133
134
135
136
137
138
139
140
141
142
143
144
145
146
147
148
149
150
151
152
153
154
155
156
157
158
159
160
161
162
163
164
165
166
167
168
169
170
171
172
173
174
175
176
177
178
179
180
181
182
183
184
185
186
187
188
189
190
191
192
193
194
195
196
197
198
199
200
201
202
203
204
205
206
207
208
209
210
211
212
213
214
215
216
217
218
219
220
221
222
223
224
225
226
227
228
229
230
231
232
233
234
235
236
237
238
239
240
241
242
243
244
245
246
247
248
249
250
251
252
253
254
255
256
257
258
259
260
261
262
263
264
265
266
267
268
269
270
271
272
273
274
275
276
277
278
279
280
281
282
283
284
285
286
287
288
289
290
291
292
293
294
295
296
297
298
299
300
301
302
303
304
305
306
307
308
309
310
311
312
313
314
315
316
317
318
319
320
321
322
323
324
325
326
327
328
329
330
331
332
333
334
335
336
337
338
339
340
341
342
343
344
345
346
347
348
349
350
351
352
353
354
355
356
357
358
359
360
361
362
363
364
365
366
367
368
369
370
371
372
373
374
375
376
377
378
379
380
381
382
383
384
385
386
387
388
389
390
391
392
393
394
395
396
397
398
399
400
401
402
403
404
405
406
407
408
409
410
411
412
413
414
415
416
417
418
419
420
421
422
423
424
425
426
427
428
429
430
431
432
433
434
435
436
437
438
439
440
441
442
443
444
445
446
447
448
449
450
451
452
453
454
455
456
457
458
459
460
461
462
463
464
465
466
467
468
469
470
471
472
473
474
475
476
477
478
479
480
481
482
483
484
485
486
487
488
489
490
491
492
493
494
495
496
497
498
499
500
501
502
503
504
505
506
507
508
509
510
511
512
513
514
515
516
517
518
519
520
521
522
523
524
525
526
527
528
529
530
531
532
533
534
535
536
537
538
539
540
541
542
543
544
545
546
547
548
549
550
551
552
553
554
555
556
557
558
559
560
561
562
563
564
565
566
567
568
569
570
571
572
573
574
575
576
577
578
579
580
581
582
583
584
585
586
587
588
589
590
591
592
593
594
595
596
597
598
599
600
601
602
603
604
605
606
607
608
609
610
611
612
613
614
615
616
617
618
619
620
621
622
623
624
625
626
627
628
629
630
631
632
633
634
635
636
637
638
639
640
641
642
643
644
645
646
647
648
649
650
651
652
653
654
655
656
657
658
659
660
661
662
663
664
665
666
667
668
669
670
671
672
673
674
675
676
677
678
679
680
681
682
683
684
685
686
687
688
689
690
691
692
693
694
695
696
697
698
699
700
701
702
703
704
705
706
707
708
709
710
711
712
713
714
715
716
717
718
719
720
721
722
723
724
725
726
727
728
729
730
731
732
733
734
735
736
737
738
739
740
741
742
743
744
745
746
747
748
749
750
751
752
753
754
755
756
757
758
759
760
761
762
763
764
765
766
767
768
769
770
771
772
773
774
775
776
777
778
779
780
781
782
783
784
785
786
787
788
789
790
791
792
793
794
795
796
797
798
799
800
801
802
803
804
805
806
807
808
809
810
811
812
813
814
815
816
817
818
819
820
821
822
823
824
825
826
827
828
829
830
831
832
833
834
835
836
837
838
839
840
841
842
843
844
845
846
847
848
849
850
851
852
853
854
855
856
857
858
859
860
861
862
863
864
865
866
867
868
869
870
871
872
873
874
875
876
877
878
879
880
881
882
883
884
885
886
887
888
889
890
891
892
893
894
895
896
897
898
899
900
901
902
903
904
905
906
907
908
909
910
911
912
913
914
915
916
917
918
919
920
921
922
923
924
925
926
927
928
929
930
931
932
933
934
935
936
937
938
939
940
941
942
943
944
945
946
947
948
949
950
951
952
953
954
955
956
957
958
959
960
961
962
963
964
965
966
967
968
969
970
971
972
973
974
975
976
977
978
979
980
981
982
983
984
985
986
987
988
989
990
991
992
993
994
995
996
997
998
999
1000
1001
1002
1003
1004
1005
1006
1007
1008
1009
1010
1011
1012
1013
1014
1015
1016
1017
1018
1019
1020
1021
1022
1023
1024
1025
1026
1027
1028
1029
1030
1031
1032
1033
1034
1035
1036
1037
1038
1039
1040
1041
1042
1043
1044
1045
1046
1047
1048
1049
1050
1051
1052
1053
1054
1055
1056
1057
1058
1059
1060
1061
1062
1063
1064
1065
1066
1067
1068
1069
1070
1071
1072
1073
1074
1075
1076
1077
1078
1079
1080
1081
1082
1083
1084
1085
1086
1087
1088
1089
1090
1091
1092
1093
1094
1095
1096
1097
1098
1099
1100
1101
1102
1103
1104
1105
1106
1107
1108
1109
1110
1111
1112
1113
1114
1115
1116
1117
1118
1119
1120
1121
1122
1123
1124
1125
1126
1127
1128
1129
1130
1131
1132
1133
1134
1135
1136
1137
1138
1139
1140
1141
1142
1143
1144
1145
1146
1147
1148
1149
1150
1151
1152
1153
1154
1155
1156
1157
1158
1159
1160
1161
1162
1163
1164
1165
1166
1167
1168
1169
1170
1171
1172
1173
1174
1175
1176
1177
1178
1179
1180
1181
1182
1183
1184
1185
1186
1187
1188
1189
1190
1191
1192
1193
1194
1195
1196
1197
1198
1199
1200
1201
1202
1203
1204
1205
1206
1207
1208
1209
1210
1211
1212
1213
1214
1215
1216
1217
1218
1219
1220
1221
1222
1223
1224
1225
1226
1227
1228
1229
1230
1231
1232
1233
1234
1235
1236
1237
1238
1239
1240
1241
1242
1243
1244
1245
1246
1247
1248
1249
1250
1251
1252
1253
1254
1255
1256
1257
1258
1259
1260
1261
1262
1263
1264
1265
1266
1267
1268
1269
1270
1271
1272
1273
1274
1275
1276
1277
1278
1279
1280
1281
1282
1283
1284
1285
1286
1287
1288
1289
1290
1291
1292
1293
1294
1295
1296
1297
1298
1299
1300
1301
1302
1303
1304
1305
1306
1307
1308
1309
1310
1311
1312
1313
1314
1315
1316
1317
1318
1319
1320
1321
1322
1323
1324
1325
1326
1327
1328
1329
1330
1331
1332
1333
1334
1335
1336
1337
1338
1339
1340
1341
1342
1343
1344
1345
1346
1347
1348
1349
1350
1351
1352
1353
1354
1355
1356
1357
1358
1359
1360
1361
1362
1363
1364
1365
1366
1367
1368
1369
1370
1371
1372
1373
1374
1375
1376
1377
1378
1379
1380
1381
1382
1383
1384
1385
1386
1387
1388
1389
1390
1391
1392
1393
1394
1395
1396
1397
1398
1399
1400
1401
1402
1403
1404
1405
1406
1407
1408
1409
1410
1411
1412
1413
1414
1415
1416
1417
1418
1419
1420
1421
1422
1423
1424
1425
1426
1427
1428
1429
1430
1431
1432
1433
1434
1435
1436
1437
1438
1439
1440
1441
1442
1443
1444
1445
1446
1447
1448
1449
1450
1451
1452
1453
1454
1455
1456
1457
1458
1459
1460
1461
1462
1463
1464
1465
1466
1467
1468
1469
1470
1471
1472
1473
1474
1475
1476
1477
1478
1479
1480
1481
1482
1483
1484
1485
1486
1487
1488
1489
1490
1491
1492
1493
1494
1495
1496
1497
1498
1499
1500
1501
1502
1503
1504
1505
1506
1507
1508
1509
1510
1511
1512
1513
1514
1515
1516
1517
1518
1519
1520
1521
1522
1523
1524
1525
1526
1527
1528
1529
1530
1531
1532
1533
1534
1535
1536
1537
1538
1539
1540
1541
1542
1543
1544
1545
1546
1547
1548
1549
1550
1551
1552
1553
1554
1555
1556
1557
1558
1559
1560
1561
1562
1563
1564
1565
1566
1567
1568
1569
1570
1571
1572
1573
1574
1575
1576
1577
1578
1579
1580
1581
1582
1583
1584
1585
1586
1587
1588
1589
1590
1591
1592
1593
1594
1595
1596
1597
1598
1599
1600
1601
1602
1603
1604
1605
1606
1607
1608
1609
1610
1611
1612
1613
1614
1615
1616
1617
1618
1619
1620
1621
1622
1623
1624
1625
1626
1627
1628
1629
1630
1631
1632
1633
1634
1635
1636
1637
1638
1639
1640
1641
1642
1643
1644
1645
1646
1647
1648
1649
1650
1651
1652
1653
1654
1655
1656
1657
1658
1659
1660
1661
1662
1663
1664
1665
1666
1667
1668
1669
1670
1671
1672
1673
1674
1675
1676
1677
1678
1679
1680
1681
1682
1683
1684
1685
1686
1687
1688
1689
1690
1691
1692
1693
1694
1695
1696
1697
1698
1699
1700
1701
1702
1703
1704
1705
1706
1707
1708
1709
1710
1711
1712
1713
1714
1715
1716
1717
1718
1719
1720
1721
1722
1723
1724
1725
1726
1727
1728
1729
1730
1731
1732
1733
1734
1735
1736
1737
1738
1739
1740
1741
1742
1743
1744
1745
1746
1747
1748
1749
1750
1751
1752
1753
1754
1755
1756
1757
1758
1759
1760
1761
1762
1763
1764
1765
1766
1767
1768
1769
1770
1771
1772
1773
1774
1775
1776
1777
1778
1779
1780
1781
1782
1783
1784
1785
1786
1787
1788
1789
1790
1791
1792
1793
1794
1795
1796
1797
1798
1799
1800
1801
1802
1803
1804
1805
1806
1807
1808
1809
1810
1811
1812
1813
1814
1815
1816
1817
1818
1819
1820
1821
1822
1823
1824
1825
1826
1827
1828
1829
1830
1831
1832
1833
1834
1835
1836
1837
1838
1839
1840
1841
1842
1843
1844
1845
1846
1847
1848
1849
1850
1851
1852
1853
1854
1855
1856
1857
1858
1859
1860
1861
1862
1863
1864
1865
1866
1867
1868
1869
1870
1871
1872
1873
1874
1875
1876
1877
1878
1879
1880
1881
1882
1883
1884
1885
1886
1887
1888
1889
1890
1891
1892
1893
1894
1895
1896
1897
1898
1899
1900
1901
1902
1903
1904
1905
1906
1907
1908
1909
1910
1911
1912
1913
1914
1915
1916
1917
1918
1919
1920
1921
1922
1923
1924
1925
1926
1927
1928
1929
1930
1931
1932
1933
1934
1935
1936
1937
1938
1939
1940
1941
1942
1943
1944
1945
1946
1947
1948
1949
1950
1951
1952
1953
1954
1955
1956
1957
1958
1959
1960
1961
1962
1963
1964
1965
1966
1967
1968
1969
1970
1971
1972
1973
1974
1975
1976
1977
1978
1979
1980
1981
1982
1983
1984
1985
1986
1987
1988
1989
1990
1991
1992
1993
1994
1995
1996
1997
1998
1999
2000
2001
2002
2003
2004
2005
2006
2007
2008
2009
2010
2011
2012
2013
2014
2015
2016
2017
2018
2019
2020
2021
2022
2023
2024
2025
2026
2027
2028
2029
2030
2031
2032
2033
2034
2035
2036
2037
2038
2039
2040
2041
2042
2043
2044
2045
2046
2047
2048
2049
2050
2051
2052
2053
2054
2055
2056
2057
2058
2059
2060
2061
2062
2063
2064
2065
2066
2067
2068
2069
2070
2071
2072
2073
2074
2075
2076
2077
2078
2079
2080
2081
2082
2083
2084
2085
2086
2087
2088
2089
2090
2091
2092
2093
2094
2095
2096
2097
2098
2099
2100
2101
2102
2103
2104
2105
2106
2107
2108
2109
2110
2111
2112
2113
2114
2115
2116
2117
2118
2119
2120
2121
2122
2123
2124
2125
2126
2127
2128
2129
2130
2131
2132
2133
2134
2135
2136
2137
2138
2139
2140
2141
2142
2143
2144
2145
2146
2147
2148
2149
2150
2151
2152
2153
2154
2155
2156
2157
2158
2159
2160
2161
2162
2163
2164
2165
2166
2167
2168
2169
2170
2171
2172
2173
2174
2175
2176
2177
2178
2179
2180
2181
2182
2183
2184
2185
2186
2187
2188
2189
2190
2191
2192
2193
2194
2195
2196
2197
2198
2199
2200
2201
2202
2203
2204
2205
2206
2207
2208
2209
2210
2211
2212
2213
2214
2215
2216
2217
2218
2219
2220
2221
2222
2223
2224
2225
2226
2227
2228
2229
2230
2231
2232
2233
2234
2235
2236
2237
2238
2239
2240
2241
2242
2243
2244
2245
2246
2247
2248
2249
2250
2251
2252
2253
2254
2255
2256
2257
2258
2259
2260
2261
2262
2263
2264
2265
2266
2267
2268
2269
2270
2271
2272
2273
2274
2275
2276
2277
2278
2279
2280
2281
2282
2283
2284
2285
2286
2287
2288
2289
2290
2291
2292
2293
2294
2295
2296
2297
2298
2299
2300
2301
2302
2303
2304
2305
2306
2307
2308
2309
2310
2311
2312
2313
2314
2315
2316
2317
2318
2319
2320
2321
2322
2323
2324
2325
2326
2327
2328
2329
2330
2331
2332
2333
2334
2335
2336
2337
2338
2339
2340
2341
2342
2343
2344
2345
2346
2347
2348
2349
2350
2351
2352
2353
2354
2355
2356
2357
2358
2359
2360
2361
2362
2363
2364
2365
2366
2367
2368
2369
2370
2371
2372
2373
2374
2375
2376
2377
2378
2379
2380
2381
2382
2383
2384
2385
2386
2387
2388
2389
2390
2391
2392
2393
2394
2395
2396
2397
2398
2399
2400
2401
2402
2403
2404
2405
2406
2407
2408
2409
2410
2411
2412
2413
2414
2415
2416
2417
2418
2419
2420
2421
2422
2423
2424
2425
2426
2427
2428
2429
2430
2431
2432
2433
2434
2435
2436
2437
2438
2439
2440
2441
2442
2443
2444
2445
2446
2447
2448
2449
2450
2451
2452
2453
2454
2455
2456
2457
2458
2459
2460
2461
2462
2463
2464
2465
2466
2467
2468
2469
2470
2471
2472
2473
2474
2475
2476
2477
2478
2479
2480
2481
2482
2483
2484
2485
2486
2487
2488
2489
2490
2491
2492
2493
2494
2495
2496
2497
2498
2499
2500
2501
2502
2503
2504
2505
2506
2507
2508
2509
2510
2511
2512
2513
2514
2515
2516
2517
2518
2519
2520
2521
2522
2523
2524
2525
2526
2527
2528
2529
2530
2531
2532
2533
2534
2535
2536
2537
2538
2539
2540
2541
2542
2543
2544
2545
2546
2547
2548
2549
2550
2551
2552
2553
2554
2555
2556
2557
2558
2559
2560
2561
2562
2563
2564
2565
2566
2567
2568
2569
2570
2571
2572
2573
2574
2575
2576
2577
2578
2579
2580
2581
2582
2583
2584
2585
2586
2587
2588
2589
2590
2591
2592
2593
2594
2595
2596
2597
2598
2599
2600
2601
2602
2603
2604
2605
2606
2607
2608
2609
2610
2611
2612
2613
2614
2615
2616
2617
2618
2619
2620
2621
2622
2623
2624
2625
2626
2627
2628
2629
2630
2631
2632
2633
2634
2635
2636
2637
2638
2639
2640
2641
2642
2643
2644
2645
2646
2
```

### 6.3.3 Writing DAT1 values ► W\_Dat1\_Pref\_Plus

This request allows for writing the values of the current DAT1 of a particular channel.

It is strongly recommended to run this request in Reset status as the new values will not be taken into account immediately to prevent an incident but after an M0 / M2 or a block holding preparation for example M1 (validated or not) or a writing operation into an E parameter.

As this request write the values for all axes of the channel be sure to fill-in the data for all these axes.

```

1  PROGRAM W_Plus_003
2  VAR
3      W_Dat1                : W_Dat1_Pref_Plus;
4  END_VAR
5
6  W_Dat1.CNCAdr             := ADR(NCK);
7  W_Dat1.byChannel          := 0;
8  W_Dat1.lrx_Axis           := 102; // In internal units
9  // You can also enter DAT values for the other axes
10 W_Dat1(enable := xEnable);
11
12 IF W_Dat1.uiResult = 0 THEN
13     xEnable:= FALSE;
14 END_IF

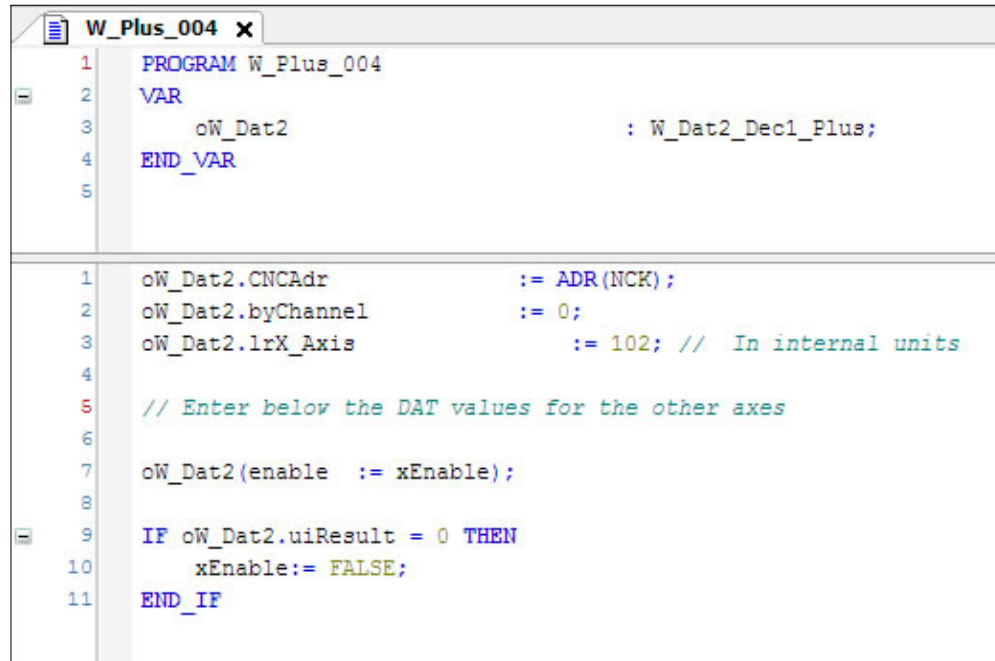
```

### 6.3.4 Writing DAT2 values ► [W\\_Dat2\\_Dec1\\_Plus](#)

This request allows for writing the values of the current DAT2 of a particular channel.

It is strongly recommended to run this request in Reset status as the new values will not be taken into account immediately to prevent an incident but after an M0 / M2 or a block holding preparation for example M1 (validated or not) or a writing operation into an E parameter.

As this request write the values for all axes of the channel be sure to fill-in the data for all these axes.



```
1 PROGRAM W_Plus_004
2 VAR
3     oW_Dat2                : W_Dat2_Dec1_Plus;
4 END_VAR
5
6
7 oW_Dat2.CNCAdr              := ADR(NCK);
8 oW_Dat2.byChannel          := 0;
9 oW_Dat2.lrX_Axis           := 102; // In internal units
10
11 // Enter below the DAT values for the other axes
12
13 oW_Dat2(enable := xEnable);
14
15 IF oW_Dat2.uiResult = 0 THEN
16     xEnable:= FALSE;
17 END_IF
```

### 6.3.5 Writing DAT3 values ► W\_Dat3\_Dec3\_Plus

This request allows for writing the values of the current DAT3 of a particular channel.

It is strongly recommended to run this request in Reset status as the new values will not be taken into account immediately to prevent an incident but after an M0 / M2 or a block holding preparation for example M1 (validated or not) or a writing operation into an E parameter.

As this request write the values for all axes of the channel be sure to fill-in the data for all these axes.

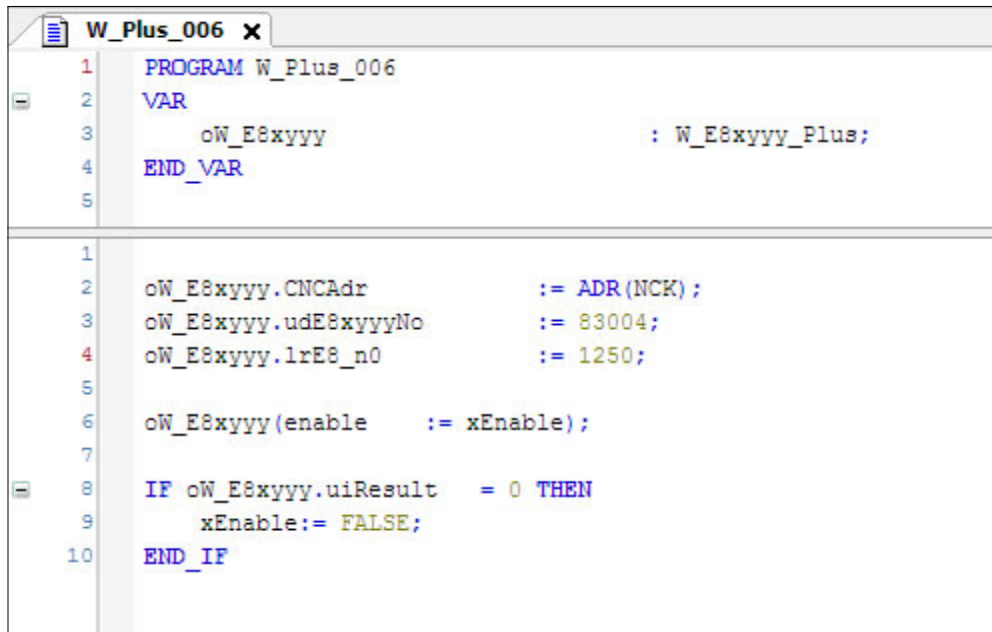
```

1  PROGRAM W_Plus_005
2  VAR
3      oW_Dat3                      : W_Dat3_Dec3_Plus;
4  END_VAR
5
6
7  oW_Dat3.CNCAdr                  := ADR(NCK);
8  oW_Dat3.byChannel                := 0;
9  oW_Dat3.lrX_Axis                  := 102; // In internal units
10 oW_Dat3.lrY_Axis                  := 201; // In internal units
11 // You can also enter DAT values for the other axes
12
13 oW_Dat3(enable := xEnable);
14
15 IF oW_Dat3.uiResult = 0 THEN
16     xEnable := FALSE;
17 END_IF

```

### 6.3.6 Writing an E8... parameter value ► [W\\_E8xyyy\\_Plus](#)

This request allows for writing one E8xyyy parameter.  
The integer part of the input argument will be written in the parameter.



```
1 PROGRAM W_Plus_006
2 VAR
3     oW_E8xyyy                : W_E8xyyy_Plus;
4 END_VAR
5
6
7
8 oW_E8xyyy.CNCAdr             := ADR(NCK);
9 oW_E8xyyy.udE8xyyyNo        := 83004;
10 oW_E8xyyy.lrE8_n0           := 1250;
11
12 oW_E8xyyy(enable := xEnable);
13
14 IF oW_E8xyyy.uiResult = 0 THEN
15     xEnable:= FALSE;
16 END_IF
```

### 6.3.7 Writing an E5 or E6 parameter value ► [W\\_E\\_Param\\_5\\_6\\_Plus](#)

This request allows for writing one parameter of the series E5 or E6.  
Only the integer part of the input argument will be taken into account as internal units.

```

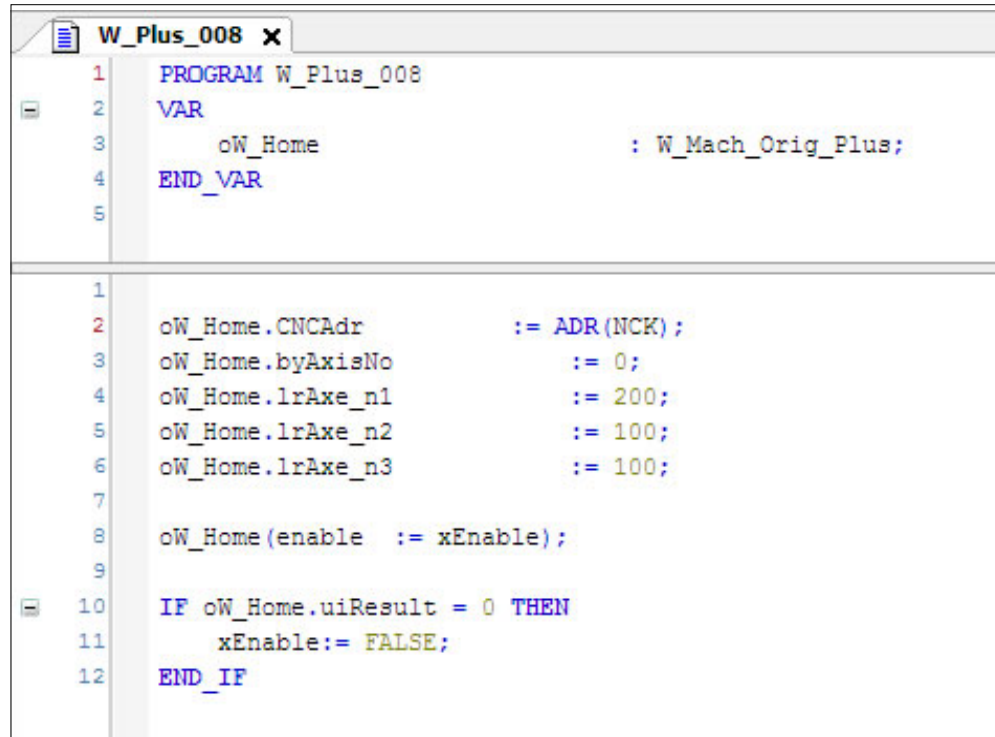
1  PROGRAM W_Plus_007
2  VAR
3      oW_E_5_6          : W_E_Param_5_6_Plus;
4      Value              : LREAL;
5  END_VAR
6
7
8  oW_E_5_6.CNCAdr       := ADR(NCK);
9  oW_E_5_6.byChannel    := 0;
10
11  oW_E_5_6.udE_number   := 70001;
12  oW_E_5_6.dataType     := 6;
13
14  Value                 := 123.456;
15  oW_E_5_6.ptValue      := ADR(Value);
16
17  oW_E_5_6(enable := xEnable);
18
19  IF oW_E_5_6.uiResult   = 0 THEN
20      xEnable := FALSE;
21  END_IF

```

### 6.3.8 Writing the home position (P16) ► W\_Mach\_Orig\_Plus

This request allows for writing the home position for an axis (equivalent of P16). Please note that this value is not written permanently in the parameter.

The value indicated in the project will be taken at the next system start-up. The request write the values at three consecutive axes addresses. **Therefore be sure to enter these three values.**

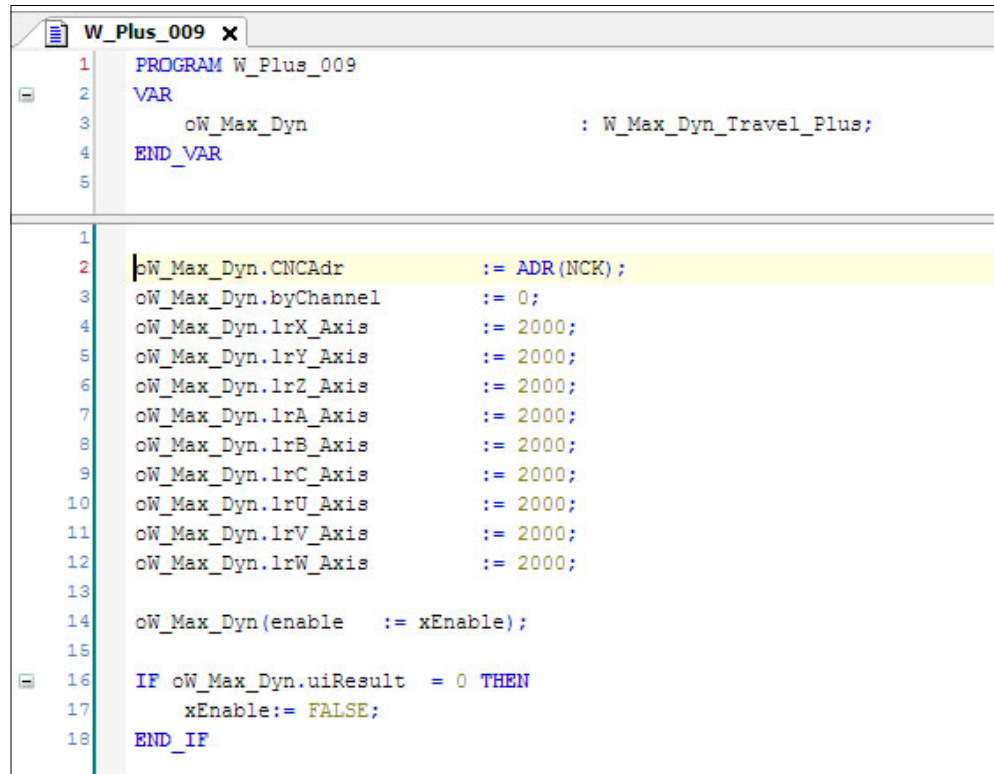


```
1  PROGRAM W_Plus_008
2  VAR
3      oW_Home                : W_Mach_Orig_Plus;
4  END_VAR
5
6
7
8
9
10 oW_Home.CNCAdr              := ADR(NCK) ;
11 oW_Home.byAxisNo            := 0;
12 oW_Home.lrAxe_n1            := 200;
13 oW_Home.lrAxe_n2            := 100;
14 oW_Home.lrAxe_n3            := 100;
15
16 oW_Home(enable := xEnable);
17
18
19 IF oW_Home.uiResult = 0 THEN
20     xEnable:= FALSE;
21 END_IF
```

### 6.3.9 Writing Maximum Dynamic Travel values ► [W\\_Max\\_Dyn\\_Travel\\_Plus](#)

This request allows for writing the maximum dynamic travels of the axes in a particular channel. The values are given in reference to the axes names.

As this request write the values for all axes of the channel be sure to fill-in the data for all these axes.



```

1  PROGRAM W_Max_Dyn_Travel_Plus
2  VAR
3      oW_Max_Dyn                : W_Max_Dyn_Travel_Plus;
4  END_VAR
5
6  oW_Max_Dyn.CNCAdr             := ADR(NCK);
7  oW_Max_Dyn.byChannel          := 0;
8  oW_Max_Dyn.lrX_Axis           := 2000;
9  oW_Max_Dyn.lrY_Axis           := 2000;
10 oW_Max_Dyn.lrZ_Axis           := 2000;
11 oW_Max_Dyn.lrA_Axis           := 2000;
12 oW_Max_Dyn.lrB_Axis           := 2000;
13 oW_Max_Dyn.lrC_Axis           := 2000;
14 oW_Max_Dyn.lrU_Axis           := 2000;
15 oW_Max_Dyn.lrV_Axis           := 2000;
16 oW_Max_Dyn.lrW_Axis           := 2000;
17
18 oW_Max_Dyn(enable := xEnable);
19
20 IF oW_Max_Dyn.uiResult = 0 THEN
21     xEnable := FALSE;
22 END_IF
  
```

### 6.3.10 Writing Maximum Dynamic Travel selected ► [W\\_Max\\_Dyn\\_Travel\\_Plus\\_2](#)

This request allows for writing the maximum dynamic travels of the axes in a particular channel. The values are given in reference to the axes names.

Respect to the previous [W\\_Max\\_Dyn\\_Travel\\_Plus](#) request, this request writes the values for the axes selected with the input variable `.boUpdate_n_Axis:= TRUE`, so only the needed change can be written.

```
W_Plus_015 x
1 PROGRAM W_Plus_015
2 VAR
3     oW_Max_Dyn_Travel_Plus_2 : W_Max_Dyn_Travel_Plus_2;
4     byChannel                : BYTE;
5     lrY_Axis                 : LREAL:=-20000;
6     lrU_Axis                 : LREAL:=-40000;
7     lrA_Axis                 : LREAL:=-70000;
8 END_VAR
9
10
11 byChannel:= 0; // 0...7
12
13 oW_Max_Dyn_Travel_Plus_2.CNCAdr:= ADR(NCK);
14 oW_Max_Dyn_Travel_Plus_2.byChannel:= byChannel;
15
16 oW_Max_Dyn_Travel_Plus_2.lrY_Axis:= lrY_Axis;
17 oW_Max_Dyn_Travel_Plus_2.boUpdate_Y_Axis:= TRUE;
18 oW_Max_Dyn_Travel_Plus_2.lrU_Axis:= lrU_Axis;
19 oW_Max_Dyn_Travel_Plus_2.boUpdate_U_Axis:= TRUE;
20 oW_Max_Dyn_Travel_Plus_2.lrA_Axis:= lrA_Axis;
21 oW_Max_Dyn_Travel_Plus_2.boUpdate_A_Axis:= TRUE;
22
23 oW_Max_Dyn_Travel_Plus_2.boUpdate_X_Axis:= FALSE;
24 oW_Max_Dyn_Travel_Plus_2.boUpdate_Z_Axis:= FALSE;
25 oW_Max_Dyn_Travel_Plus_2.boUpdate_V_Axis:= FALSE;
26 oW_Max_Dyn_Travel_Plus_2.boUpdate_W_Axis:= FALSE;
27 oW_Max_Dyn_Travel_Plus_2.boUpdate_B_Axis:= FALSE;
28 oW_Max_Dyn_Travel_Plus_2.boUpdate_C_Axis:= FALSE;
29
30 oW_Max_Dyn_Travel_Plus_2(enable:= xEnable);
31
32 IF oW_Max_Dyn_Travel_Plus_2.uiResult = 0 THEN
33     xEnable:= FALSE;
34 END_IF
```

### 6.3.11 Writing Maximum Static Travel values ► [W\\_Max\\_Stat\\_Travel\\_Plus](#)

This request allows for writing the maximum static travels of axes.  
The values are given for three consecutive axes addresses.

As this request write the values for three consecutive axes addresses be sure to fill-in the data for all these axes.

```

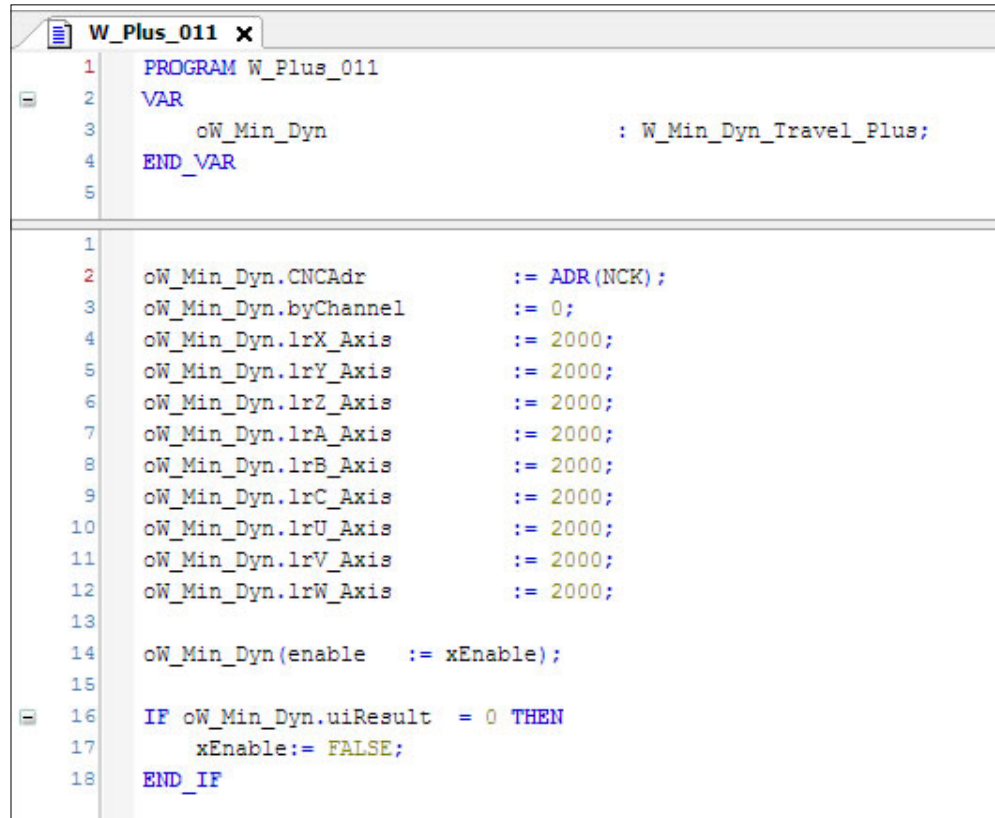
1  PROGRAM W_Plus_010
2  VAR
3      oW_Max_Stat          : W_Max_Stat_Travel_Plus;
4  END_VAR
5
6
7
8  oW_Max_Stat.CNCAdr      := ADR(NCK) ;
9  oW_Max_Stat.byAxisNo    := 0;
10 oW_Max_Stat.lrAxe_n1    := 2000;
11 oW_Max_Stat.lrAxe_n2    := 2000;
12 oW_Max_Stat.lrAxe_n3    := 2000;
13
14 oW_Max_Stat(enable := xEnable);
15
16 IF oW_Max_Stat.uiResult = 0 THEN
17     xEnable:= FALSE;
18 END_IF

```

### 6.3.12 Writing Minimum Dynamic Travel values ► [W\\_Min\\_Dyn\\_Travel\\_Plus](#)

This request allows for writing the minimum dynamic travels of the axes in a particular channel. The values are given in reference to the axes names.

As this request write the values for all axes of the channel be sure to fill-in the data for all these axes.



```
1  PROGRAM W_Min_Dyn_Travel_Plus
2  VAR
3      oW_Min_Dyn          : W_Min_Dyn_Travel_Plus;
4  END_VAR
5
6
7  oW_Min_Dyn.CNCAdr      := ADR(NCK);
8  oW_Min_Dyn.byChannel   := 0;
9  oW_Min_Dyn.lrX_Axis    := 2000;
10 oW_Min_Dyn.lrY_Axis    := 2000;
11 oW_Min_Dyn.lrZ_Axis    := 2000;
12 oW_Min_Dyn.lrA_Axis    := 2000;
13 oW_Min_Dyn.lrB_Axis    := 2000;
14 oW_Min_Dyn.lrC_Axis    := 2000;
15 oW_Min_Dyn.lrU_Axis    := 2000;
16 oW_Min_Dyn.lrV_Axis    := 2000;
17 oW_Min_Dyn.lrW_Axis    := 2000;
18
19 oW_Min_Dyn(enable := xEnable);
20
21 IF oW_Min_Dyn.uiResult = 0 THEN
22     xEnable:= FALSE;
23 END_IF
```

### 6.3.13 Writing Minimum Dynamic Travel selected ► [W\\_Min\\_Dyn\\_Travel\\_Plus\\_2](#)

This request allows for writing the maximum dynamic travels of the axes in a particular channel. The values are given in reference to the axes names.

Respect to the previous [W\\_Min\\_Dyn\\_Travel\\_Plus](#) request, this request writes the values for the axes selected with the input variable **.boUpdate\_n\_Axis:= TRUE**, so only the needed change can be written.

```

1 | PROGRAM W_Plus_016
2 | VAR
3 |     oW_Min_Dyn_Travel_Plus_2    : W_Min_Dyn_Travel_Plus_2;
4 |     byChannel                   : BYTE;
5 |     lrY_Axis                    : LREAL:= -20000;
6 |     lrU_Axis                    : LREAL:= -40000;
7 |     lrA_Axis                    : LREAL:= -70000;
8 | END_VAR

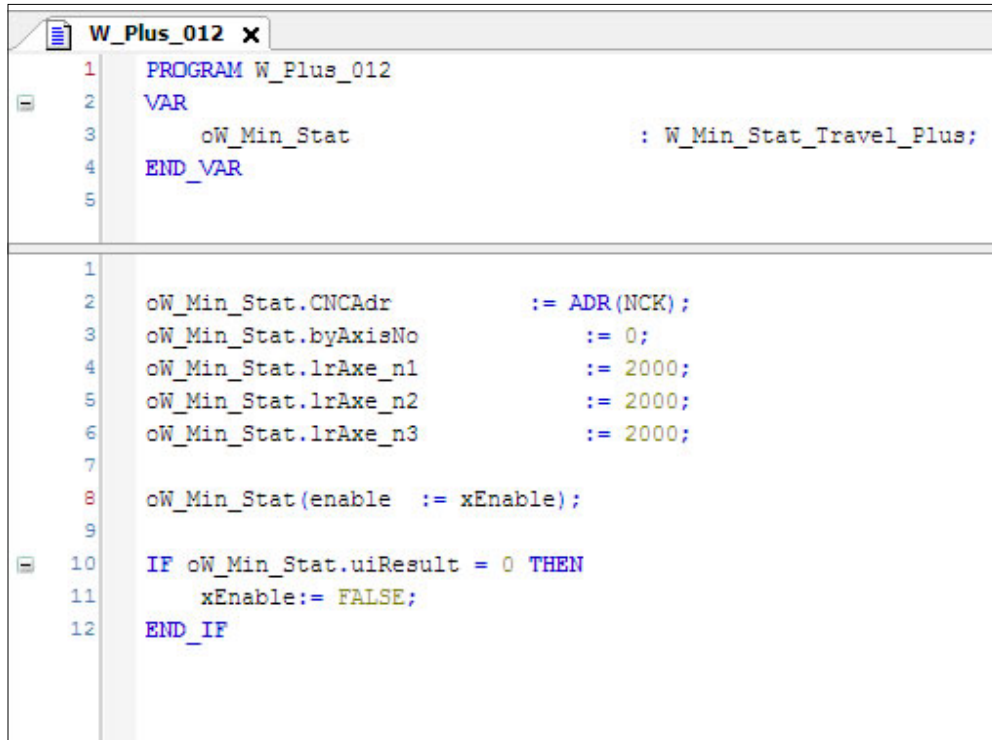
1 |
2 |     byChannel:= 0; // 0...7
3 |
4 |     oW_Min_Dyn_Travel_Plus_2.CNCAdr:= ADR(NCK);
5 |     oW_Min_Dyn_Travel_Plus_2.byChannel:= byChannel;
6 |
7 |     oW_Min_Dyn_Travel_Plus_2.lrY_Axis:= lrY_Axis;
8 |     oW_Min_Dyn_Travel_Plus_2.boUpdate_Y_Axis:= TRUE;
9 |     oW_Min_Dyn_Travel_Plus_2.lrU_Axis:= lrU_Axis;
10 |    oW_Min_Dyn_Travel_Plus_2.boUpdate_U_Axis:= TRUE;
11 |    oW_Min_Dyn_Travel_Plus_2.lrA_Axis:= lrA_Axis;
12 |    oW_Min_Dyn_Travel_Plus_2.boUpdate_A_Axis:= TRUE;
13 |
14 |    oW_Min_Dyn_Travel_Plus_2.boUpdate_X_Axis:= FALSE;
15 |    oW_Min_Dyn_Travel_Plus_2.boUpdate_Z_Axis:= FALSE;
16 |    oW_Min_Dyn_Travel_Plus_2.boUpdate_V_Axis:= FALSE;
17 |    oW_Min_Dyn_Travel_Plus_2.boUpdate_W_Axis:= FALSE;
18 |    oW_Min_Dyn_Travel_Plus_2.boUpdate_B_Axis:= FALSE;
19 |    oW_Min_Dyn_Travel_Plus_2.boUpdate_C_Axis:= FALSE;
20 |
21 |    oW_Min_Dyn_Travel_Plus_2(enable:= xEnable);
22 |
23 | IF oW_Min_Dyn_Travel_Plus_2.uiResult = 0 THEN
24 |     xEnable:= FALSE;
25 | END_IF
26 |

```

### 6.3.14 Writing Minimum Static Travel values ► [W\\_Min\\_Stat\\_Travel\\_Plus](#)

This request allows for writing the minimum static travels of axes.  
The values are given for three consecutive axes addresses.

As this request write the values for three consecutive axes addresses, be sure to fill-in the data for all these axes.



```
1 PROGRAM W_Plus_012
2 VAR
3     oW_Min_Stat                : W_Min_Stat_Travel_Plus;
4 END_VAR
5
6
7
8 oW_Min_Stat.CNCAdr             := ADR(NCK);
9
10 oW_Min_Stat.byAxisNo          := 0;
11 oW_Min_Stat.lrAxe_n1          := 2000;
12 oW_Min_Stat.lrAxe_n2          := 2000;
13 oW_Min_Stat.lrAxe_n3          := 2000;
14
15 oW_Min_Stat(enable := xEnable);
16
17 IF oW_Min_Stat.uiResult = 0 THEN
18     xEnable:= FALSE;
19 END_IF
```

### 6.3.15 Execute motor brake test ► W\_Motor\_Brake\_Test

#### Requirement

This request has been introduced from the FlexiumTool version 4.1.30.00:  
It needs:

CNC software version >= 4.1.30.00

Drive software version >= 1.30.0000

PLC ENA library version >= 1.30.0000 where the request W\_Motor\_Brake\_Test, to be instanced, is resident.

CNC option FXSO000463 otherwise error 2 will be received by the PLC request call. No ENA CNC option is required

The PLC variable Flexium.RCNC.Axis[x].ExtendStatus.BrakeTestRunning has been added to show the drive condition. Refer to M00031-EN manual for a detailed example.

```

W_Plus_017 x
1  PROGRAM W_Plus_017
2  VAR
3      oW_Motor_Brake_Test : W_Motor_Brake_Test;
4      byDriveAdd           : BYTE;
5      DirPos               : LREAL;
6      DirNeg               : LREAL;
7      Error                : UINT;
8  END_VAR

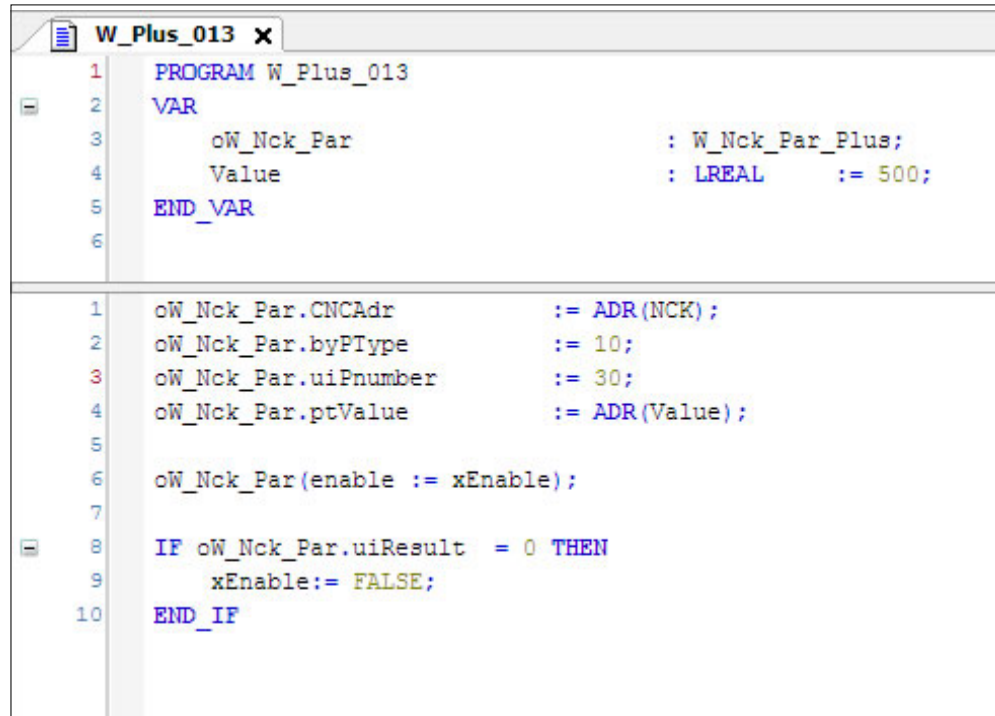
1
2  DirPos:= 2000; // CNC increment
3  DirNeg:= -1000; // CNC increment
4  byDriveAdd:= 0; // 0...31
5
6  oW_Motor_Brake_Test.CNCAdr:= ADR(NCK);
7  oW_Motor_Brake_Test.byDriveAdd:= byDriveAdd;
8  oW_Motor_Brake_Test.lNegAmp:= DirNeg;
9  oW_Motor_Brake_Test.lPosAmp:= DirPos;
10
11  oW_Motor_Brake_Test(enable:= xEnable);
12
13  IF oW_Motor_Brake_Test.uiResult = 0 THEN
14      xEnable:= FALSE;
15      Error:= oW_Motor_Brake_Test.uiRespStatus;
16  END_IF
17

```

### 6.3.16 Writing an NCK parameter ► W\_NCK\_Par\_Plus

This request allows for writing a word of a particular machine parameter.  
The word number is given by the property *uiPoffset* (not shown in example).

The type of parameter must be given, the possibility of writing and type of parameters is given in annex.

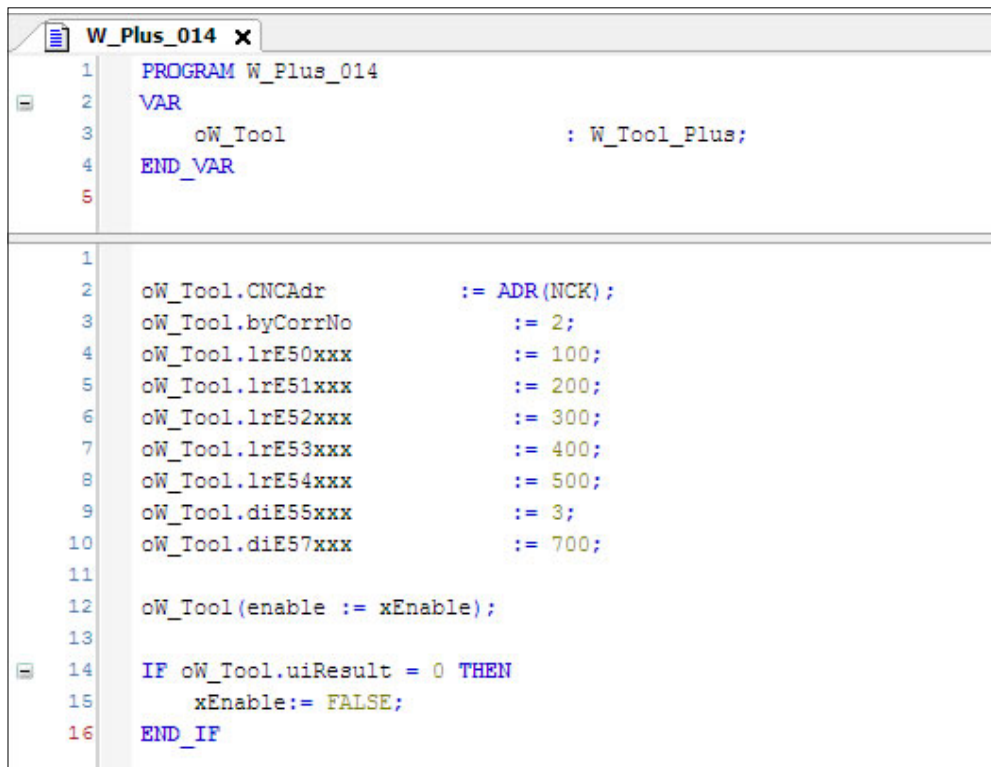


```
1 PROGRAM W_Plus_013
2 VAR
3     oW_Nck_Par          : W_Nck_Par_Plus;
4     Value               : LREAL      := 500;
5 END_VAR
6
7
8 oW_Nck_Par.CNCAdr       := ADR(NCK);
9 oW_Nck_Par.byPType     := 10;
10 oW_Nck_Par.uiPnumber   := 30;
11 oW_Nck_Par.ptValue     := ADR(Value);
12
13 oW_Nck_Par(enable := xEnable);
14
15 IF oW_Nck_Par.uiResult = 0 THEN
16     xEnable:= FALSE;
17 END_IF
```

### 6.3.17 Writing a tool offset ▶ W\_Tool\_Plus

This request allows for writing a particular tool offset.

All components are written with the exception of the H value which is given by a separate request.



```

1  PROGRAM W_Plus_014
2  VAR
3      oW_Tool                : W_Tool_Plus;
4  END_VAR
5
6
7
8
9
10
11
12  oW_Tool.CNCAdr             := ADR(NCK);
13  oW_Tool.byCorrNo           := 2;
14  oW_Tool.lrE50xxx           := 100;
15  oW_Tool.lrE51xxx           := 200;
16  oW_Tool.lrE52xxx           := 300;
17  oW_Tool.lrE53xxx           := 400;
18  oW_Tool.lrE54xxx           := 500;
19  oW_Tool.diE55xxx           := 3;
20  oW_Tool.diE57xxx           := 700;
21
22  oW_Tool(enable := xEnable);
23
24  IF oW_Tool.uiResult = 0 THEN
25      xEnable:= FALSE;
26  END_IF
  
```

---

Page deliberately empty

## Summary new

|          |                                                                          |            |
|----------|--------------------------------------------------------------------------|------------|
| <b>A</b> | <b>Appendix .....</b>                                                    | <b>163</b> |
| A.1      | Runtime Error Codes .....                                                | 163        |
| A.2      | Drive parameters specificities .....                                     | 164        |
| A.2.1    | Case NUMDrive C Sw version 3.5x, 4.5x parameters conversions table ..... | 165        |
| A.2.2    | Dependencies table .....                                                 | 173        |
| A.2.3    | NUMDrive C parameters format and limits.....                             | 175        |
| A.3      | NUMDrive X parameters Conversion rules, Dependencies, Limits .....       | 183        |
| A.3.1    | Conversion rules table .....                                             | 183        |
| A.3.2    | Dependencies table .....                                                 | 194        |
| A.3.3    | NUMDrive X parameters format and limits.....                             | 196        |
| A.4      | NCK parameters .....                                                     | 205        |
| A.4.1    | NCK Parameters for Flexium+ .....                                        | 205        |
| A.4.2    | NCK Parameters for Flexium .....                                         | 206        |

1

2

3

4

5

6

A

---

Page deliberately empty

## A Appendix

### A.1 Runtime Error Codes

| N° | Variable name                 | Value Description                                                                                        |
|----|-------------------------------|----------------------------------------------------------------------------------------------------------|
| 0  | <b>ERR_OK</b>                 | No error                                                                                                 |
| 1  | <b>ERR_FAILED</b>             | General error - to be used only for internal errors                                                      |
| 2  | <b>ERR_PARAMETER</b>          | An invalid parameter was passed to a function.                                                           |
| 3  | <b>ERR_NOTINITIALIZED</b>     | Function cannot be executed, since component has not been initialized yet.<br>It may work later, though. |
| 4  | <b>ERR_VERSION</b>            | Version conflict                                                                                         |
| 5  | <b>ERR_TIMEOUT</b>            | Operation timed out                                                                                      |
| 6  | <b>ERR_NOBUFFER</b>           | Insufficient memory to carry out the request                                                             |
| 10 | <b>ERR_PENDING</b>            | For asynchronous calls: call not yet complete                                                            |
| 11 | <b>ERR_NUMPENDING</b>         | Too many pending calls. Try later                                                                        |
| 12 | <b>ERR_NOTIMPLEMENTED</b>     | The function is not implemented                                                                          |
| 13 | <b>ERR_INVALIDID</b>          | No object with the provided id found                                                                     |
| 14 | <b>ERR_OVERFLOW</b>           | Integer overflow                                                                                         |
| 15 | <b>ERR_BUFFERSIZE</b>         | The size of a buffer is too small or invalid                                                             |
| 16 | <b>ERR_NO_OBJECT</b>          | No object with this specified name available                                                             |
| 17 | <b>ERR_NOMEMORY</b>           | No heap memory available                                                                                 |
| 18 | <b>ERR_DUPLICATE</b>          | An object with the same name is still available                                                          |
| 19 | <b>ERR_MEMORY_OVERWRITE</b>   | Heap memory was written out of bounds!<br>Memory overwrite error                                         |
| 20 | <b>ERR_INVALID_HANDLE</b>     | Invalid handle to an object                                                                              |
| 21 | <b>ERR_END_OF_OBJECT</b>      | End of object reached                                                                                    |
| 22 | <b>ERR_NO_CHANGE</b>          | No changes done                                                                                          |
| 23 | <b>ERR_INVALID_INTERFACE</b>  | Invalid or unknown interface                                                                             |
| 24 | <b>ERR_NOT_SUPPORTED</b>      | Functionality not supported                                                                              |
| 25 | <b>ERR_NO_ACCESS_RIGHTS</b>   | No access rights for this operation                                                                      |
| 26 | <b>ERR_OUT_OF_LIMITS</b>      | Specified limits of a resource exceeded                                                                  |
| 27 | <b>ERR_ENTRIES_REMAINING</b>  | Remaining entries that could not be transmitted because of buffer limitation                             |
| 28 | <b>ERR_INVALID_SESSION_ID</b> | Invalid online session id                                                                                |
| 29 | <b>ERR_EXCEPTION</b>          | Exception occurred                                                                                       |
| 30 | <b>ERR_SIGNATURE_MISMATCH</b> | Signature mismatch of an API function                                                                    |
| 31 | <b>ERR_VERSION_MISMATCH</b>   | Version mismatch                                                                                         |
| 32 | <b>ERR_TYPE_MISMATCH</b>      | Type mismatch                                                                                            |
| 33 | <b>ERR_ID_MISMATCH</b>        | ID mismatch                                                                                              |
| 34 | <b>ERR_NO_CONSISTENCY</b>     | Consistency error                                                                                        |

## A.2 Drive parameters specificities

The drive parameters must be expressed in internal units (for rapidity) as well as in physical units (for coherence)

The relation between the physical units and internal units can be different according to the parameter and are described in the two following tables (For NUMDriveC and NUMDriveX)

As an example for V50 the relation is:

$$diPinValue = trunc \left( \frac{\text{V50 in Physical Unit} \times \text{According to internal unit conversion for V50}}{\text{Module size (Ipeak) i.e MDLU3014}} \right) = 7944$$

### Ipeak and Inom drive current table



For more detailed informations on the NUMDrive parameters please refer to NUMDrive C Parameters manual AMOMAN007.

| Module size | Ipeak (A) | Inom (Arms) |         |
|-------------|-----------|-------------|---------|
|             |           | V271=100    | V271=50 |
| MDLU3007N   | 7.00      | 2.00        | 2.00    |
| MDLU3014N   | 14.00     | 4.00        | 4.00    |
| MDLU3021N   | 21.00     | 7.00        | 7.00    |
| MDLU3034N   | 34.00     | 14.00       | 14.00   |
| MDLU3050N   | 50.00     | 20.00       | 20.00   |
| MDLU3075N   | 75.00     | 35.00       | 35.00   |
| MDLU3100N   | 100.00    | 45.00       | 45.00   |
| MDLU3150N   | 150.00    | 60.00       | 60.00   |
| MDLU3014A   | 14.00     | 8.90        | 6.00    |
| MDLU3021A   | 21.00     | 13.00       | 8.00    |
| MDLU3034A   | 34.00     | 13.00       | 8.00    |
| MDLU3050A   | 50.00     | 28.28       | 18.38   |
| MDLU3075A   | 75.00     | 34.00       | 23.00   |
| MDLU3130A   | 130.00    | 60.00       | 42.00   |
| MDLU3200A   | 200.00    | 100.00      | 72.00   |
| MDLU3400A   | 400.00    | 200.00      | 130.00  |
| MDLU3014B   | 14.00     | 8.90        | 6.00    |
| MDLU3021B   | 21.00     | 8.90        | 6.00    |
| MDLU3050B   | 50.00     | 28.28       | 18.38   |

| Module size | Ipeak (A) | Inom (Arms) |         |
|-------------|-----------|-------------|---------|
|             |           | V271=100    | V271=50 |
| MDLUX007A   | 7.00      | 4.40        | 4.40    |
| MDLUX014A   | 14.00     | 8.90        | 6.00    |
| MDLUX021A   | 21.00     | 13.00       | 8.00    |
| MDLUX034A   | 34.00     | 13.00       | 8.00    |
| MDLUX050A   | 50.00     | 28.28       | 18.38   |
| MDLUX075A   | 75.00     | 34.00       | 23.00   |
| MDLUX100A   | 100.00    | 45.00       | 34.00   |
| MDLUX150A   | 150.00    | 60.00       | 40.00   |
| MDLUX200A   | 200.00    | 100.00      | 70.00   |
| MDLUX400A   | 400.00    | 200.00      | 130.00  |
| MDLUX007B   | 7.00      | 4.40        | 4.40    |
| MDLUX014B   | 14.00     | 8.90        | 6.00    |
| MDLUX021B   | 21.00     | 8.90        | 6.00    |
| MDLUX050B   | 50.00     | 28.28       | 18.38   |
| MDLUX075B   | 75.00     | 41.00       | 28.28   |

## A.2.1 Case NUMDrive C Sw version 3.5x, 4.5x parameters conversions table



The conversions table may be different for other drive Software versions.

Therefore, please consult NUM for the latest informations.

The parameter value in internal unit [IU] is obtained by applying the conversion rule to the value in physical units [PU].

Example:  $V37 [IU] = V37 [PU] * 1000$

| Param. N° | Type  | Halt | [PU]              | Resolution | Conversion rule from Vxx in physical unit to Vxx in internal unit | Description                                            |
|-----------|-------|------|-------------------|------------|-------------------------------------------------------------------|--------------------------------------------------------|
| V20       | DWORD | Yes  | [ ]               | 1          | V20                                                               | Sensor connectors X3, X4 configuration                 |
| V21       | DWORD | No   | [ ]               | 1          | V21                                                               | TP1/2 output configuration                             |
| V22       | DWORD | Yes  | [ ]               | 1          | V22                                                               | PROGO1/2 output configuration                          |
| V23       | DINT  | Yes  | [ ]               | 1          | V23                                                               | Drives addresses confirm and check                     |
| V30       | DWORD | Yes  | [ ]               | 1          | V30                                                               | General Drive Configuration                            |
| V32       | DWORD | Yes  | [ ]               | 1          | V32                                                               | Config. Braking following alarms [1=Brake]             |
| V33       | DWORD | Yes  | [ ]               | 1          | V33                                                               | Alarm reset configuration [1=Reset]                    |
| V34       | DWORD | Yes  | [ ]               | 1          | V34                                                               | Configuration DF1/DF2 relay [1=Open]                   |
| V35       | DWORD | Yes  | [ ]               | 1          | V35                                                               | SW functions [0=Not available][1=Available]            |
| V36       | DINT  | Yes  | [ ]               | 1          | V36                                                               | SAM scaling factor                                     |
| V37       | REAL  | Yes  | s                 | 0.01       | $V37 * 1000$                                                      | Delay time of power detection                          |
| V40       | REAL  | No   | %                 | 0.01       | $V40 * 65536 / 100$                                               | Speed reached dynamic threshold                        |
| V41       | REAL  | No   | mm/min            | 0.01       | $V41 * 163.840$                                                   | Motor zero speed threshold, enable, brake              |
| V42       | REAL  | No   | Rpm               | 0.01       | $V42 * 1342.18$                                                   | Motor zero speed threshold, enable, brake              |
| V43       | REAL  | Yes  | s                 | 0.01       | $V43 * 1000$                                                      | Braking timeout on config. V30 V32                     |
| V44       | DINT  | No   | %                 | 1          | $V44 * 1024 / 100$                                                | Braking current on config. V30 V32                     |
| V45       | REAL  | No   | mm/min            | 0.01       | $V45 * 163.840$                                                   | Speed dynamic check [0]                                |
| V46       | REAL  | No   | Rpm               | 0.01       | $V46 * 1342.18$                                                   | Speed dynamic check [0]                                |
| V50       | REAL  | No   | Arms              | 0.01       | $V50 * 1853609.716 / (I_{peak} * 100)$                            | Torque current limit [0]                               |
| V51       | REAL  | No   | mA/Rpm            | 0.01       | $V51 * 5656.854 / (I_{peak} * 100)$                               | Proportional speed loop gain [0]                       |
| V52       | REAL  | No   | ms                | 0.01       | $32767 * V264 / (V52 * 1000)$                                     | Integral speed loop gain [0]                           |
| V53       | REAL  | No   | ms                | 0.01       | $V264 * 32767 / (V53 * 1000)$                                     | AP10: 2 <sup>Δ</sup> Integral speed loop gain [0]      |
| V57       | REAL  | No   | Arms              | 0.01       | $V57 * 1853609.716 / (I_{peak} * 100)$                            | Torque current limit h.r. [0]                          |
| V58       | REAL  | No   | mA/Rpm            | 0.01       | $V58 * 5656.854 / (I_{peak} * 100)$                               | Proportional speed loop gain h.r. [0]                  |
| V59       | REAL  | No   | ms                | 0.01       | $32767 * V264 / (V59 * 1000)$                                     | Integral speed loop gain h.r. [0]                      |
| V60       | REAL  | No   | Arms/m/min        | 0.01       | $V60 * 46340.95 / (I_{peak} * 100)$                               | Proportional speed loop gain [0]                       |
| V75       | REAL  | No   | ms                | 0.01       | $V264 * 32767 / (V75 * 1000)$                                     | AP10: 2 <sup>Δ</sup> Integral speed loop gain h.r. [0] |
| V80       | REAL  | No   | [ ]               | 0.01       | $32767.5 * V80$                                                   | Set Point Weight speed loop gain                       |
| V81       | REAL  | No   | [ ]               | 0.01       | $16383.5 * V81$                                                   | Anti-windup speed loop gain                            |
| V91       | DINT  | No   | Rpm/s             | 1          | $V91 * V265 * 1.34218 / 1000$                                     | Acceleration/deceleration ramp [0]                     |
| V92       | DINT  | No   | Rpm/s             | 1          | $V92 * V265 * 1.34218 / 1000$                                     | Acceleration/deceleration ramp h.r.[0]                 |
| V93       | DINT  | No   | mm/s <sup>2</sup> | 1          | $V93 * 60 * V265 * 0.163840 / 1000$                               | Acceleration/deceleration ramp [0]                     |
| V94       | DINT  | No   | %                 | 1          | $V94 * 163840 * V95 / 100$                                        | Speed reference Limit [% maximum speed] [0]            |
| V95       | DINT  | No   | m/min             | 1          | $V95 * 163840$                                                    | Maximum motor speed [0]                                |
| V96       | DINT  | No   | m/min             | 1          | $V96 * 163840$                                                    | Threshold for overspeed alarm [0]                      |
| V97       | DINT  | No   | %                 | 1          | $V97 * 1342.18 * V98 / 100$                                       | Speed reference limit [% maximum speed] [0]            |
| V98       | DINT  | No   | Rpm               | 1          | $V98 * 1342.18$                                                   | Maximum motor speed [0]                                |
| V99       | DINT  | No   | Rpm               | 1          | $V99 * 1342.18$                                                   | Overspeed alarm threshold [0]                          |

| Param. N° | Type  | Halt | [PU]      | Resolution | Conversion rule from Vxx in physical unit to Vxx in internal unit                  | Description                                    |
|-----------|-------|------|-----------|------------|------------------------------------------------------------------------------------|------------------------------------------------|
| V100      | REAL  | No   | Arms      | 0.01       | $V100 \cdot 1853609.716 / (I_{peak} \cdot 100)$                                    | Deliverable current limit                      |
| V101      | DINT  | No   | V         | 1          | $V101 \cdot 8$                                                                     | Voltage limits Vd and Vq                       |
| V102      | REAL  | No   | V/Arms    | 0.01       | $(V102 \cdot (I_{peak} \cdot 100)) / 28.28427 / 4$                                 | Proportional current loop gain                 |
| V103      | REAL  | No   | ms        | 0.01       | $32767 \cdot V271 / (1000 \cdot V103)$                                             | Integral current loop gain                     |
| V104      | REAL  | No   | mVrms/Rpm | 0.01       | $V104 \cdot 16.237976 \cdot 2 / V257$                                              | Motor EMF phase compensation                   |
| V105      | REAL  | No   | mH        | 0.01       | $(V105 + V408) \cdot (I_{peak} \cdot 100) \cdot 0.01503011016$                     | Compensation of motor Lq inductance            |
| V106      | REAL  | No   | mH        | 0.01       | $(V106 + V408) \cdot (I_{peak} \cdot 100) \cdot 0.01503011016$                     | Compensation of motor Ld inductance            |
| V107*     | DINT  | Yes  | Hz        | 1          | $(0.102943708 \cdot V271 \cdot V107) / (1 + 0.00000314159 \cdot V271 \cdot V107)$  | Lowpass Filter electrical speed                |
| V108*     | REAL  | Yes  | Arms      | 0.01       | $V108 \cdot 1853609.716 / (I_{peak} \cdot 100)$                                    | Rated motor current [I <sup>2</sup> t]         |
| V109      | REAL  | No   | Arms      | 0.01       | $V109 \cdot 1853609.716 / (I_{peak} \cdot 100)$                                    | Vertical load compensation current             |
| V110      | REAL  | No   | Arms      | 0.01       | $V110 \cdot 1853609.716 / (I_{peak} \cdot 100)$                                    | Deliverable current limit h.r.                 |
| V111      | REAL  | No   | V/Arms    | 0.01       | $(V111 \cdot (I_{peak} \cdot 100)) / 28.28427 / 4$                                 | Proportional current gain h.r.                 |
| V112      | REAL  | No   | ms        | 0.01       | $32767 \cdot V271 / (1000 \cdot V112)$                                             | Integral current gain h.r.                     |
| V114      | REAL  | No   | mH        | 0.01       | $((V105 \cdot 1.00 / V407) + V408) \cdot (I_{peak} \cdot 100) \cdot 0.01503011016$ | Motor Lq inductance compensation h.r.          |
| V115*     | REAL  | Yes  | Arms      | 0.01       | $V115 \cdot 1853609.716 / (I_{peak} \cdot 100)$                                    | Rated motor current [I <sup>2</sup> t] h.r.    |
| V116      | REAL  | No   | mH        | 0.01       | $V116 \cdot (I_{peak} \cdot 100) \cdot 0.01503011016$                              | Lm (for compensation) l.r.                     |
| V117      | REAL  | No   | mH        | 0.01       | $V116 \cdot (I_{peak} \cdot 100) \cdot 0.01503011016 / V407$                       | Lm (for compensation) h.r.                     |
| V118      | REAL  | No   | mH        | 0.01       | $((V106 \cdot 1.00 / V407) + V408) \cdot (I_{peak} \cdot 100) \cdot 0.01503011016$ | Motor Ld inductance compensation h.r.          |
| V119      | REAL  | No   | cycles    | 0.01       | $V119 \cdot (V271 / V263) \cdot 1024$                                              | Phase lead                                     |
| V120      | REAL  | No   | Vrms/m/s  | 0.01       | $V120 \cdot V259 \cdot 54.12659 / (50 \cdot V257)$                                 | Linear Motor EMF phase compensation            |
| V121      | REAL  | No   | [ ]       | 0.01       | $32767.5 \cdot V121$                                                               | Set Point Weight current loop gain             |
| V122      | REAL  | No   | [ ]       | 0.01       | $16383.5 \cdot V122$                                                               | Anti-windup speed current gain                 |
| V123      | DINT  | No   | [ ]       | 1          | V123                                                                               | Step of Iq ref. interpolation                  |
| V125      | DINT  | No   | %         | 1          | $V125 \cdot V108 \cdot 18536.09716 / (I_{peak} \cdot 100)$                         | Current percentage V108 for EPSM               |
| V128*     | DINT  | Yes  | %         | 1          | $V128 \cdot V108 \cdot 18536.09716 / (I_{peak} \cdot 100)$                         | Locked rotor current I <sup>2</sup> t (% V108) |
| V129*     | DINT  | Yes  | %         | 1          | $V129 \cdot V115 \cdot 18536.09716 / (I_{peak} \cdot 100)$                         | Locked rotor current I <sup>2</sup> t (% V115) |
| V130      | DINT  | No   | [ ]       | 1          | V130                                                                               | Associate MP with TP1                          |
| V131      | DINT  | No   | [ ]       | 1          | V131                                                                               | Associate MP with TP2                          |
| V132      | DINT  | No   | [ ]       | 1          | V132                                                                               | Associate MP with TP3                          |
| V133      | DINT  | No   | [ ]       | 1          | V133                                                                               | Associate MP with TP4                          |
| V134      | DINT  | No   | [ ]       | 1          | V134                                                                               | Multiplication factor for TP1                  |
| V135      | DINT  | No   | [ ]       | 1          | V135                                                                               | Multiplication factor for TP2                  |
| V136      | DINT  | No   | [ ]       | 1          | V136                                                                               | Offset for analog TP1                          |
| V137      | DINT  | No   | [ ]       | 1          | V137                                                                               | Offset for analog TP2                          |
| V138      | DWORD | No   | [ ]       | 1          | V138                                                                               | Configuration of analog TPs                    |
| V139      | DWORD | Yes  | [ ]       | 1          | V139                                                                               | Configuration of Programmable Relays           |
| V140      | DINT  | No   | [ ]       | 1          | V140                                                                               | Associate MP with programmable relay 1         |
| V141      | REAL  | No   | [ ]       | 0.01       | $V141 \cdot V143$                                                                  | Programmable relay 1. threshold 1              |
| V142      | REAL  | No   | [ ]       | 0.01       | $V142 \cdot V143$                                                                  | Programmable relay 1. threshold 2              |
| V143      | REAL  | No   | [ ]       | 0.01       | V143                                                                               | Programmable relay 1. MP scale factor          |
| V144      | DINT  | No   | ms        | 1          | $V144 / (V263 / 1000)$                                                             | Programm. relay 1. tripping time constant      |
| V150      | DINT  | No   | [ ]       | 1          | V150                                                                               | Associate MP with programmable relay 2         |
| V151      | REAL  | No   | [ ]       | 0.01       | $V151 \cdot V153$                                                                  | Programmable relay 2. threshold 1              |

| Param. N° | Type  | Halt | [PU]        | Resolution | Conversion rule from Vxx in physical unit to Vxx in internal unit     | Description                                 |
|-----------|-------|------|-------------|------------|-----------------------------------------------------------------------|---------------------------------------------|
| V152      | REAL  | No   | [ ]         | 0.01       | V152*V153                                                             | Programmable relay 2. threshold 2           |
| V153      | REAL  | No   | [ ]         | 0.01       | V153                                                                  | Programmable relay 2. MP scale factor       |
| V154      | DINT  | No   | ms          | 1          | V154/(V263/1000)                                                      | Programm. relay 2. tripping time constant   |
| V160      | DWORD | Yes  | [ ]         | 1          | V160                                                                  | Special applications selection              |
| V165      | DINT  | No   | mm/min      | 1          | V165*163.840                                                          | AP03 Mst: Max compens. torque equalizer     |
| V166      | REAL  | No   | mm/min/Arms | 0.01       | V166*(Ipeak*100)*5.656941/1000                                        | AP03 Mst: Proportional torq. equaliz. gain  |
| V168      | DINT  | Yes  | [ ]         | 1          | V168                                                                  | AP03 Mst: N.of slaves on the local link     |
| V169      | DWORD | Yes  | [ ]         | 1          | V169                                                                  | Link Tandem configuration                   |
| V170      | DWORD | Yes  | [ ]         | 1          | V170                                                                  | Tandem application configuration            |
| V171      | REAL  | No   | Arms        | 0.01       | -2*V171*1853609.716/(Ipeak*100)                                       | AP03 Master: Preload Tandem current         |
| V172      | REAL  | No   | Rpm/Arms    | 0.01       | V172*(Ipeak*100)/21.5792                                              | AP03 Mst: Proportional torq. equaliz. gain  |
| V173      | REAL  | No   | ms          | 0.01       | 32767*V265/(1000*V173)                                                | AP03 Mst: Integral torque equalizer gain    |
| V175      | DINT  | No   | Rpm         | 1          | V175*1342.18                                                          | AP03 Mst: Max compens. torque equalizer     |
| V176      | DINT  | No   | ms          | 1          | (V176*1000)/V265                                                      | AP03 Mst: T.out A52.41 (saturated trq eq.)  |
| V177      | REAL  | No   | Arms/s      | 0.01       | 2*V177*1.853609716*65536*V265/(Ipeak*100)                             | AP03 Mst: Preload ramp speed                |
| V180      | DINT  | Yes  | %           | 1          | (V180*V108*13107)/(Ipeak*100)                                         | Current percentage (V108) for EPS           |
| V182      | DINT  | Yes  | %           | 1          | V182*32767/100                                                        | Phase U/W relative weight for EPS           |
| V183      | DINT  | Yes  | %           | 1          | V183*32767/100                                                        | Phase V/W relative weight for EPS           |
| V190      | DINT  | No   | [ ]         | 1          | V190                                                                  | AP06: Max displacement meas. S1-S2 Ph.U.    |
| V191      | DINT  | No   | ms          | 1          | V191                                                                  | AP06: Max duration measure displac. S1-S2   |
| V205      | REAL  | No   | Hz          | 0.01       | (0.102943708*V265*V205)/(1.0+0.00000314159*V265*V205)                 | AP13: RMS extractor filter                  |
| V206      | REAL  | No   | Rpm         | 0.01       | V206*1342.18                                                          | AP13: RMS filter output saturation          |
| V207      | REAL  | No   | %           | 0.01       | V206*1342.18*V207/100.0                                               | AP13: Offset of modulation factor           |
| V208      | REAL  | No   | mm/min      | 0.01       | V208*163.840                                                          | AP13: RMS filter output saturation          |
| V209      | REAL  | No   | %           | 0.01       | V208*163.840*V209/100.0                                               | AP13: Offset of modulation factor           |
| V210      | REAL  | No   | [ ]         | 0.01       | -V210*1024                                                            | AP08: Active damping 1 factor               |
| V217      | REAL  | No   | [ ]         | 0.01       | 32.0*V217                                                             | AP12: Active damping 2 K1                   |
| V218      | REAL  | No   | Hz          | 0.01       | (0.102943708*V265*V218)/(V217+0.00000314159*V265*V218)                | AP12: Active damping 2 filter               |
| V219      | DINT  | No   | [ ]         | 1          | 32767.0*(V217-0.00000314159*V265*V218)/(V217+0.00000314159*V265*V218) | AP12: coeff.A Active damping 2 filter       |
| V220      | DWORD | Yes  | [ ]         | 1          | V220                                                                  | Reference configuration                     |
| V221      | DWORD | Yes  | [ ]         | 1          | V221                                                                  | Enable configurations                       |
| V222      | DINT  | No   | [ ]         | 1          | V222                                                                  | Shift number of analog stimulus             |
| V223      | DINT  | Yes  | Hz          | 1          | (0.102943708*V265*V223)/(1+0.00000314159*V265*V223)                   | Lowpass Filter analog reference             |
| V224      | DINT  | Yes  | Hz          | 1          | 32768 - 2*(0.102943708*V265*V223)/(1+0.00000314159*V265*V223)         | Lowpass Filter analog reference [hidden]    |
| V225      | DWORD | Yes  | [ ]         | 1          | V225                                                                  | Digital input configuration [GP_INA/B]      |
| V238      | DWORD | Yes  | [ ]         | 1          | V238                                                                  | UGA filter on S1 sensor (RCMDS20)           |
| V239      | DWORD | Yes  | [ ]         | 1          | V239                                                                  | UGA filter on S2 sensor (RCMDS21)           |
| V240*     | REAL  | Yes  | s           | 0.01       | V240*100                                                              | First time constant for I^2t                |
| V241*     | DINT  | Yes  | s           | 1          | V241*1                                                                | Second time constant for I^2t               |
| V245      | REAL  | No   | Hz          | 0.01       | V245*13.1072                                                          | Freq. threshold. Change of time const. I^2t |
| V246      | REAL  | No   | Hz          | 0.01       | V246*13.1072                                                          | Frequency threshold hysteresis V245         |
| V250      | DWORD | Yes  | [ ]         | 1          | V250                                                                  | ST Revision                                 |

| Param. N° | Type   | Halt | [PU]          | Resolution | Conversion rule from Vxx in physical unit to Vxx in internal unit                 | Description                                 |
|-----------|--------|------|---------------|------------|-----------------------------------------------------------------------------------|---------------------------------------------|
| V251      | DINT   | Yes  | [ ]           | 1          | V251                                                                              | Parameter Compatibility                     |
| V252      | STRING | Yes  | [ ]           |            | V252                                                                              | Drive type                                  |
| V253      | DWORD  | Yes  | [ ]           | 1          | V253                                                                              | Drive size                                  |
| V254      | DINT   | Yes  | [ ]           | 1          | V254                                                                              | Motor number                                |
| V255      | STRING | Yes  | [ ]           |            | V255                                                                              | Motor type                                  |
| V256*     | STRING | Yes  | [ ]           |            | V256                                                                              | Motor power Connection type                 |
| V257*     | DINT   | Yes  | [ ]           | 1          | V257                                                                              | Motor poles pairs number                    |
| V258      | STRING | Yes  | [ ]           |            | V258                                                                              | Motor sensor type                           |
| V259      | REAL   | Yes  | mm            | 0.01       | V259                                                                              | Linear Motor Pole Pitch [180 degrees el.]   |
| V260      | DINT   | No   | [ ]           | 1          | V260                                                                              | Speed regulator Parameters exchange         |
| V261      | DINT   | No   | [ ]           | 1          | V261                                                                              | Position regulator Parameters exchange      |
| V263      | DINT   | Yes  | us            | 50         | V263                                                                              | System sampling time                        |
| V264      | DINT   | Yes  | us            | 100        | V264/V263                                                                         | Speed loop sampling time                    |
| V265      | DINT   | Yes  | us            | 200        | V265/V263                                                                         | Position loop sampling time                 |
| V266      | REAL   | Yes  | ms            | 0.2        | V266*1000/V263                                                                    | CNC Task Sampling time                      |
| V267      | DINT   | Yes  | A             | 1          | V267*100                                                                          | ST Peak current                             |
| V268      | DINT   | Yes  | mDeg          | 1          | V268*65.536/360.0                                                                 | Electrical phase offset Hall effect sensor  |
| V269      | DINT   | Yes  | mDeg          | 1          | V269*65.536/360.0                                                                 | Position offset for electrical phasing      |
| V270      | DINT   | Yes  | Ohm           | 1          | V270                                                                              | Motor PTC-NTC value                         |
| V271      | DINT   | Yes  | us            | 50         | V271/V263                                                                         | Current loop sampling time                  |
| V272      | DWORD  | Yes  | [ ]           | 1          | V272                                                                              | Motor thermal sensor configuration          |
| V273      | DWORD  | Yes  | [ ]           | 1          | V273                                                                              | Electrical phase offset management          |
| V301      | DINT   | No   | mm/min/<br>mm | 1          | V301                                                                              | Proportional position gain [0]              |
| V305      | DINT   | Yes  | [ ]           | 1          | V305                                                                              | Measure module for rot. axis/spindle abs[0] |
| V306      | DINT   | Yes  | [ ]           | 1          | V306                                                                              | S1 sensor MULTI coefficient [0]             |
| V307      | DINT   | Yes  | [ ]           | 1          | V307                                                                              | S1 sensor DIVI coefficient [0]              |
| V310      | DINT   | Yes  | [ ]           | 1          | V310                                                                              | S1 sensor offset in CNC Phys. Unit [0]      |
| V311      | DINT   | Yes  | [ ]           | 1          | V311                                                                              | S2 sensor MULTI coefficient [0]             |
| V312      | DINT   | Yes  | [ ]           | 1          | V312                                                                              | S2 sensor DIVI coefficient [0]              |
| V313      | DINT   | Yes  | [ ]           | 1          | V313                                                                              | S2 sensor offset in CNC Phys. Unit [0]      |
| V314      | DWORD  | Yes  | [ ]           | 1          | V314                                                                              | Sinusoidal signals range of S1 Sensor       |
| V315      | DWORD  | Yes  | [ ]           | 1          | V315                                                                              | Sinusoidal signals range of S2 Sensor       |
| V317      | REAL   | Yes  | V             | 0.01       | V317*1638.4                                                                       | S1 sensor power supply voltage              |
| V318      | REAL   | Yes  | V             | 0.01       | V318*1638.4                                                                       | S2 sensor power supply voltage              |
| V340      | DINT   | No   | [ ]           | 1          | V340                                                                              | Step of speed calculation S1                |
| V341      | DINT   | No   | [ ]           | 1          | V341                                                                              | Step of speed calculation S2                |
| V342      | REAL   | No   | Hz            | 0.01       | $(0.102943708 \cdot V264 \cdot V342) / (1 + 0.00000314159 \cdot V264 \cdot V342)$ | Speed filter by S1 sensor                   |
| V343      | REAL   | No   | Hz            | 0.01       | $(0.102943708 \cdot V264 \cdot V343) / (1 + 0.00000314159 \cdot V264 \cdot V343)$ | Speed filter by S2 sensor                   |
| V344      | DINT   | Yes  | [ ]           | 1          | V344                                                                              | S1 Sensor code ID                           |
| V345      | DINT   | Yes  | [ ]           | 1          | V345                                                                              | S2 Sensor code ID                           |
| V346      | DINT   | Yes  | [ ]           | 1          | V346                                                                              | S1 Sensor number                            |
| V347      | DINT   | Yes  | [ ]           | 1          | V347                                                                              | S2 Sensor number                            |
| V348*     | DWORD  | Yes  | [ ]           | 1          | V348                                                                              | S1 Sensor configuration                     |
| V349*     | DWORD  | Yes  | [ ]           | 1          | V349                                                                              | S2 Sensor configuration                     |
| V350      | DWORD  | Yes  | [ ]           | 1          | V350                                                                              | Position loop sensor config.                |
| V351      | DWORD  | Yes  | [ ]           | 1          | V351                                                                              | Speed loop sensor config.                   |
| V352      | DWORD  | Yes  | [ ]           | 1          | V352                                                                              | Current loop sensor config.                 |

1

2

3

4

5

6

A

| Param. N° | Type | Halt | [PU] | Resolution | Conversion rule from Vxx in physical unit to Vxx in internal unit | Description                                          |
|-----------|------|------|------|------------|-------------------------------------------------------------------|------------------------------------------------------|
| V353      | REAL | Yes  | [ ]  | 0.01       | V353*100                                                          | Period of position sensor/motor turn [0]             |
| V355      | DINT | Yes  | [ ]  | 1          | V355                                                              | AP02: Mechanical ratio: load turns                   |
| V356      | DINT | Yes  | [ ]  | 1          | V356                                                              | AP02: Mechanical ratio: sensor turns                 |
| V359      | DINT | Yes  | [ ]  | 1          | $(65536*3*4*1024/V353)*100/V264$                                  | k_velrot120                                          |
| V360      | DINT | Yes  | [ ]  | 1          | V360                                                              | Period by turn sensor S1                             |
| V361      | DINT | Yes  | [ ]  | 1          | V361                                                              | Absolute resolution by turn sensor S1                |
| V362      | DINT | No   | [ ]  | 1          | $V301*1.31072*32768*4/V360$                                       | kpposrot10                                           |
| V363      | DINT | Yes  | [ ]  | 1          | $(65536*3*1024*4/V360)*100/V264$                                  | k_velrot10                                           |
| V364      | DINT | Yes  | [ ]  | 1          | $V361/V360$                                                       | sinuarot10                                           |
| V365      | DINT | Yes  | [ ]  | 1          | V365                                                              | Number of absolute sensor turns S1                   |
| V367      | DINT | No   | [ ]  | 1          | V367                                                              | Not used                                             |
| V368      | DINT | Yes  | [ ]  | 1          | V368                                                              | Intervals between coded references S1                |
| V370      | DINT | Yes  | [ ]  | 1          | V370                                                              | Period by turn sensor S2                             |
| V371      | DINT | Yes  | [ ]  | 1          | V371                                                              | Absolute resolution by turn sensor S2                |
| V372      | DINT | No   | [ ]  | 1          | $V301*1.31072*32768*4/V353$                                       | kpposrot120                                          |
| V373      | DINT | Yes  | [ ]  | 1          | $(65536*3*1024*4/V370)*100/V264$                                  | k_velrot20                                           |
| V374      | DINT | Yes  | [ ]  | 1          | $V371/V370$                                                       | sinuarot20                                           |
| V375      | DINT | Yes  | [ ]  | 1          | V375                                                              | Number of absolute sensor turns S2                   |
| V376      | DINT | No   | [ ]  | 1          | $V301*1.31072*32768*4/V370$                                       | kpposrot20                                           |
| V377      | DINT | No   | [ ]  | 1          | V377                                                              | Not used                                             |
| V378      | DINT | Yes  | [ ]  | 1          | V378                                                              | Intervals between coded references S2                |
| V380      | DINT | Yes  | um   | 1          | V380                                                              | Period of linear sensor S1                           |
| V381      | DINT | Yes  | nm   | 1          | V381                                                              | Absolute resolution of linear sensor S1              |
| V382      | DINT | Yes  | [ ]  | 1          | $(V259*2000/V380)+0.1$                                            | npslin10                                             |
| V383      | DINT | No   | [ ]  | 1          | $V301*5.24288*V380*4$                                             | kpposlin10                                           |
| V384      | DINT | Yes  | [ ]  | 1          | $(24576*V380*4)*100/V264$                                         | k_vellin10                                           |
| V385      | DINT | Yes  | [ ]  | 1          | $V380*1000/V381$                                                  | sinualin10                                           |
| V386      | DINT | Yes  | nm   | 1          | $(V259*2000000/V381)+0.1$                                         | ralinppm10                                           |
| V390      | DINT | Yes  | um   | 1          | V390                                                              | Period of linear sensor S2                           |
| V391      | DINT | Yes  | nm   | 1          | V391                                                              | Absolute resolution of linear sensor S2              |
| V392      | DINT | Yes  | [ ]  | 1          | $(V259*2000/V390)+0.1$                                            | npslin20                                             |
| V393      | DINT | No   | [ ]  | 1          | $V301*5.24288*V390*4$                                             | kpposlin20                                           |
| V394      | DINT | Yes  | [ ]  | 1          | $(24576*V390*4)*100/V264$                                         | k_vellin20                                           |
| V395      | DINT | Yes  | [ ]  | 1          | $V390*1000/V391$                                                  | sinualin20                                           |
| V396      | DINT | Yes  | nm   | 1          | $(V259*2000000/V391)+0.1$                                         | ralinppm20                                           |
| V400      | DINT | No   | mOhm | 1          | $V400*3.2768*V406*(V271/50)/V403$                                 | Rs at 120 degrees C                                  |
| V401      | DINT | No   | mOhm | 1          | $V401*3.2768*V406*(V271/50)/V403$                                 | Rr at 150 degrees C                                  |
| V402      | DINT | No   | uH   | 1          | $(V402+V408*1000)*65.536/V403$                                    | sigmaLs                                              |
| V403      | REAL | No   | mH   | 0.01       | V403                                                              | Lr                                                   |
| V404      | DINT | No   | [ ]  | 1          | $133774.75*3276.8*(V271/50)/((I_{peak}*100)*V403)$                | $[I.U.=(K_i/K_v) /L_r * (T_s*2^{16})=a04] \ K_i/K_v$ |
| V405      | DINT | No   | [ ]  | 1          | $133774.75*3276.8*V407*(V271/50)/((I_{peak}*100)*V403)$           | $[U.I.=a04b*ratio\_gamma=a04a]a04a$                  |
| V406      | DINT | No   | [ ]  | 1          | V406                                                              | twotosh0                                             |
| V407      | DINT | No   | [ ]  | 1          | V407                                                              | ratio_gamma                                          |
| V408      | REAL | No   | mH   | 0.01       | V408                                                              | Additional extern inductance                         |
| V409      | DINT | No   | V    | 1          | V409*8                                                            | ST flux-weakening voltage                            |
| V410      | REAL | No   | [ ]  | 0.01       | V410*256                                                          | Proportional gain flux loop I.r.                     |
| V411      | REAL | No   | ms   | 0.01       | $32.767*V264/V411$                                                | Integral gain flux loop I.r.                         |

| Param. N° | Type  | Halt | [PU]     | Resolution | Conversion rule from Vxx in physical unit to Vxx in internal unit                                               | Description                                 |
|-----------|-------|------|----------|------------|-----------------------------------------------------------------------------------------------------------------|---------------------------------------------|
| V412      | DINT  | No   | %        | 1          | $0.64 \cdot V412$                                                                                               | Lr saturation level l.r.                    |
| V413      | DINT  | No   | [ ]      | 1          | $32768 / (0.64 \cdot V412)$                                                                                     | ksatinv l.r.                                |
| V414      | DINT  | No   | Rpm      | 1          | $V414 \cdot 0.3431457 \cdot V257 \cdot (V271 / 100)$                                                            | Observer engagement speed l.r.              |
| V415      | DINT  | No   | Rpm      | 1          | $V415 \cdot 0.3431457 \cdot V257 \cdot (V271 / 100)$                                                            | Observer engagement hysteresis l.r.         |
| V417      | DINT  | No   | Rpm      | 1          | $V417 \cdot V257 \cdot 0.21845333$                                                                              | Defluxing starting speed l.r.               |
| V425      | DINT  | No   | %        | 1          | $V425 \cdot V427 \cdot V412 \cdot 131.072 \cdot 1.4142135 / (I_{peak} \cdot 100)$                               | min_flux as a %age of V427 l.r.             |
| V426      | REAL  | No   | Arms/V*s | 0.01       | $V426 \cdot 7592.3856 \cdot V264 / (I_{peak} \cdot 100)$                                                        | K deflux l.r.                               |
| V427      | REAL  | No   | Arms     | 0.01       | $V427 \cdot V412 \cdot 13107.2 \cdot 1.4142135 / (I_{peak} \cdot 100)$                                          | Magnetization current l.r.                  |
| V428      | REAL  | No   | Arms     | 0.01       | $V428 \cdot 1853609.716 / (I_{peak} \cdot 100)$                                                                 | Magnetization current limit l.r.            |
| V448      | DINT  | No   | uH       | 1          | $(V402 + V407 \cdot V408 \cdot 1000) \cdot 65.536 / V403$                                                       | sigmaLs h.r.                                |
| V450      | REAL  | No   | [ ]      | 0.01       | $V450 \cdot 256$                                                                                                | Proportional flux loop gain h.r.            |
| V451      | REAL  | No   | ms       | 0.01       | $32.767 \cdot V264 / V451$                                                                                      | Integral flux loop gain h.r.                |
| V452      | DINT  | No   | %        | 1          | $0.64 \cdot V452$                                                                                               | Lr saturation level h.r.                    |
| V453      | DINT  | No   | [ ]      | 1          | $32768 / (0.64 \cdot V452)$                                                                                     | ksatinv h.r.                                |
| V454      | DINT  | No   | Rpm      | 1          | $V454 \cdot 0.3431457 \cdot V257 \cdot (V271 / 100)$                                                            | Observer engagement speed h.r.              |
| V455      | DINT  | No   | Rpm      | 1          | $V455 \cdot 0.3431457 \cdot V257 \cdot (V271 / 100)$                                                            | Observer engagement hysteresis h.r.         |
| V457      | DINT  | No   | Rpm      | 1          | $V457 \cdot V257 \cdot 0.21845333$                                                                              | Defluxing starting speed h.r.               |
| V465      | DINT  | No   | %        | 1          | $V465 \cdot V467 \cdot V452 \cdot 131.072 \cdot 1.4142135 / (I_{peak} \cdot 100)$                               | min_flux as a %age of V467 h.r.             |
| V466      | REAL  | No   | Arms/V*s | 0.01       | $V466 \cdot 7592.3856 \cdot V264 / (I_{peak} \cdot 100)$                                                        | K deflux h.r.                               |
| V467      | REAL  | No   | Arms     | 0.01       | $V467 \cdot V452 \cdot 13107.2 \cdot 1.4142135 / (I_{peak} \cdot 100)$                                          | Magnetization current h.r.                  |
| V468      | REAL  | No   | Arms     | 0.01       | $V468 \cdot 1853609.716 / (I_{peak} \cdot 100)$                                                                 | Magnetization current limit h.r.            |
| V470      | REAL  | No   | mH       | 0.01       | $((V402 / 1000.0) + V408) \cdot (I_{peak} \cdot 100) \cdot 0.01503011016$                                       | SigmaLs flux-weakening l.r.                 |
| V471      | REAL  | No   | mH       | 0.01       | $((V402 / (V407 \cdot 1000.0)) + V408) \cdot (I_{peak} \cdot 100) \cdot 0.01503011016$                          | SigmaLs flux-weakening h.r.                 |
| V472      | DINT  | No   | [ ]      | 1          | $(V400 + V401) \cdot (I_{peak} \cdot 100) \cdot 65.536 / 133772.76$                                             | Total resistance power lost Ri2 l.r.        |
| V473      | DINT  | No   | [ ]      | 1          | $(V400 + V401) \cdot (I_{peak} \cdot 100) \cdot 65.536 / (V407 \cdot 133772.76)$                                | Total resistance power lost Ri2 h.r.        |
| V480      | REAL  | No   | %        | 0.01       | $V480 \cdot 655.35$                                                                                             | Correction factor for MP006 Delivered Power |
| V481*     | DWORD | Yes  | [ ]      | 1          | V481                                                                                                            | Control mode selection                      |
| V488      | DINT  | No   | mOhm     | 1          | $V488 \cdot (I_{peak} \cdot 100) \cdot 65.536 \cdot (234.5 + 120.0) / ((234.5 + 20.0) \cdot 2 \cdot 133772.76)$ | Resistance tt at 20 degrees C               |
| V489      | REAL  | No   | mH       | 0.01       | $(V408 + V489) \cdot (I_{peak} \cdot 100) \cdot 0.01503011016$                                                  | Motor Lq inductance                         |
| V491      | REAL  | No   | Arms     | 0.01       | $V491 \cdot 1853609.716 / (I_{peak} \cdot 100)$                                                                 | Id_Max (defluxing parameter)                |
| V492      | REAL  | No   | Arms     | 0.01       | $V492 \cdot 1853609.716 / (I_{peak} \cdot 100)$                                                                 | Id current without defluxing                |
| V496      | DINT  | No   | Rpm      | 1          | $V496 \cdot V257 \cdot 0.21845333$                                                                              | Defluxing starting speed                    |
| V497      | REAL  | No   | m/min    | 0.01       | $V497 \cdot 109.22667 \cdot V257 / V259$                                                                        | Defluxing starting speed                    |
| V498      | REAL  | No   | Arms/V*s | 0.01       | $V498 \cdot 7592.3856 \cdot V264 / (I_{peak} \cdot 100)$                                                        | K Iq reduction                              |
| V499      | REAL  | No   | Arms/V*s | 0.01       | $V499 \cdot 7592.3856 \cdot V264 / (I_{peak} \cdot 100)$                                                        | K flux-weakening Id                         |
| V500      | DWORD | No   | [ ]      | 1          | V500                                                                                                            | Filter configuration                        |
| V501      | DINT  | No   | [ ]      | 1          | V501                                                                                                            | ph11 (FILTER 1 for torque current ref.)     |
| V502      | DINT  | No   | [ ]      | 1          | V502                                                                                                            | ph12 (FILTER 1 for torque current ref.)     |
| V503      | DINT  | No   | [ ]      | 1          | V503                                                                                                            | ph21 (FILTER 1 for torque current ref.)     |
| V504      | DINT  | No   | [ ]      | 1          | V504                                                                                                            | ph22 (FILTER 1 for torque current ref.)     |
| V505      | DINT  | No   | [ ]      | 1          | V505                                                                                                            | ga1 (FILTER 1 for torque current ref.)      |

| Param. N° | Type | Halt | [PU] | Resolution | Conversion rule from Vxx in physical unit to Vxx in internal unit | Description                             |
|-----------|------|------|------|------------|-------------------------------------------------------------------|-----------------------------------------|
| V506      | DINT | No   | []   | 1          | V506                                                              | ga2 (FILTER 1 for torque current ref.)  |
| V507      | DINT | No   | []   | 1          | V507                                                              | r*r (FILTER 1 for torque current ref.)  |
| V508      | DINT | No   | []   | 1          | V508                                                              | C1 (FILTER 1 for torque current ref.)   |
| V509      | DINT | No   | []   | 1          | V509                                                              | C2 (FILTER 1 for torque current ref.)   |
| V511      | DINT | No   | []   | 1          | V511                                                              | ph11 (FILTER 2 for torque current ref.) |
| V512      | DINT | No   | []   | 1          | V512                                                              | ph12 (FILTER 2 for torque current ref.) |
| V513      | DINT | No   | []   | 1          | V513                                                              | ph21 (FILTER 2 for torque current ref.) |
| V514      | DINT | No   | []   | 1          | V514                                                              | ph22 (FILTER 2 for torque current ref.) |
| V515      | DINT | No   | []   | 1          | V515                                                              | ga1 (FILTER 2 for torque current ref.)  |
| V516      | DINT | No   | []   | 1          | V516                                                              | ga2 (FILTER 2 for torque current ref.)  |
| V517      | DINT | No   | []   | 1          | V517                                                              | r*r (FILTER 2 for torque current ref.)  |
| V518      | DINT | No   | []   | 1          | V518                                                              | C1 (FILTER 2 for torque current ref.)   |
| V519      | DINT | No   | []   | 1          | V519                                                              | C2 (FILTER 2 for torque current ref.)   |
| V521      | DINT | No   | []   | 1          | V521                                                              | ph11 (FILTER 3 for torque current ref.) |
| V522      | DINT | No   | []   | 1          | V522                                                              | ph12 (FILTER 3 for torque current ref.) |
| V523      | DINT | No   | []   | 1          | V523                                                              | ph21 (FILTER 3 for torque current ref.) |
| V524      | DINT | No   | []   | 1          | V524                                                              | ph22 (FILTER 3 for torque current ref.) |
| V525      | DINT | No   | []   | 1          | V525                                                              | ga1 (FILTER 3 for torque current ref.)  |
| V526      | DINT | No   | []   | 1          | V526                                                              | ga2 (FILTER 3 for torque current ref.)  |
| V527      | DINT | No   | []   | 1          | V527                                                              | r*r (FILTER 3 for torque current ref.)  |
| V528      | DINT | No   | []   | 1          | V528                                                              | C1 (FILTER 3 for torque current ref.)   |
| V529      | DINT | No   | []   | 1          | V529                                                              | C2 (FILTER 3 for torque current ref.)   |
| V531      | DINT | No   | []   | 1          | V531                                                              | ph11 (FILTER 4 for torque current ref.) |
| V532      | DINT | No   | []   | 1          | V532                                                              | ph12 (FILTER 4 for torque current ref.) |
| V533      | DINT | No   | []   | 1          | V533                                                              | ph21 (FILTER 4 for torque current ref.) |
| V534      | DINT | No   | []   | 1          | V534                                                              | ph22 (FILTER 4 for torque current ref.) |
| V535      | DINT | No   | []   | 1          | V535                                                              | ga1 (FILTER 4 for torque current ref.)  |
| V536      | DINT | No   | []   | 1          | V536                                                              | ga2 (FILTER 4 for torque current ref.)  |
| V537      | DINT | No   | []   | 1          | V537                                                              | r*r (FILTER 4 for torque current ref.)  |
| V538      | DINT | No   | []   | 1          | V538                                                              | C1 (FILTER 4 for torque current ref.)   |
| V539      | DINT | No   | []   | 1          | V539                                                              | C2 (FILTER 4 for torque current ref.)   |
| V541      | DINT | No   | []   | 1          | V541                                                              | ph11 (FILTER 1 for speed ref.)          |
| V542      | DINT | No   | []   | 1          | V542                                                              | ph12 (FILTER 1 for speed ref.)          |
| V543      | DINT | No   | []   | 1          | V543                                                              | ph21 (FILTER 1 for speed ref.)          |
| V544      | DINT | No   | []   | 1          | V544                                                              | ph22 (FILTER 1 for speed ref.)          |
| V545      | DINT | No   | []   | 1          | V545                                                              | ga1 (FILTER 1 for speed ref.)           |
| V546      | DINT | No   | []   | 1          | V546                                                              | ga2 (FILTER 1 for speed ref.)           |
| V547      | DINT | No   | []   | 1          | V547                                                              | r*r (FILTER 1 for speed ref.)           |
| V548      | DINT | No   | []   | 1          | V548                                                              | C1 (FILTER 1 for speed ref.)            |
| V549      | DINT | No   | []   | 1          | V549                                                              | C2 (FILTER 1 for speed ref.)            |
| V551      | DINT | No   | []   | 1          | V551                                                              | ph11 (FILTER notch for AP13)            |
| V552      | DINT | No   | []   | 1          | V552                                                              | ph12 (FILTER notch for AP13)            |
| V553      | DINT | No   | []   | 1          | V553                                                              | ph21 (FILTER notch for AP13)            |
| V554      | DINT | No   | []   | 1          | V554                                                              | ph22 (FILTER notch for AP13)            |
| V555      | DINT | No   | []   | 1          | V555                                                              | ga1 (FILTER notch for AP13)             |
| V556      | DINT | No   | []   | 1          | V556                                                              | ga2 (FILTER notch for AP13)             |
| V557      | DINT | No   | []   | 1          | V557                                                              | r*r (FILTER notch for AP13)             |
| V558      | DINT | No   | []   | 1          | V558                                                              | C1 (FILTER notch for AP13)              |
| V559      | DINT | No   | []   | 1          | V559                                                              | C2 (FILTER notch for AP13)              |
| V600      | DINT | No   | %    | 1          | V600*V108*18536.09716/(Ipeak*100)                                 | V/f: Maximal motor current (% of V108)  |
| V601      | DINT | No   | Hz   | 1          | V601*13.1072                                                      | V/f: Rated frequency                    |
| V602      | REAL | No   | V    | 0.01       | V602*8                                                            | V/f: Rated voltage                      |

| Param. N° | Type | Halt | [PU]       | Resolution | Conversion rule from Vxx in physical unit to Vxx in internal unit | Description                                 |
|-----------|------|------|------------|------------|-------------------------------------------------------------------|---------------------------------------------|
| V603      | REAL | No   | V          | 0.01       | V603*8                                                            | V/f: V                                      |
| V604      | DINT | No   | mOhm       | 1          | $V604 * (I_{peak} * 100) * 65.536 * 1.4 / 133772.76$              | V/f: Stator resistance @ 20 degrees C       |
| V605      | DINT | No   | %          | 1          | $V605 * V108 * 18536.09716 / (I_{peak} * 100)$                    | V/f: Intermittent overload (% of V108)      |
| V606      | DINT | No   | Hz         | 1          | V606*13.1072                                                      | V/f: Freq. start of overload reduction      |
| V1045     | REAL | No   | mm/min     | 0.01       | V1045*163.840                                                     | Speed dynamic check [1]                     |
| V1046     | REAL | No   | Rpm        | 0.01       | V1046*1342.18                                                     | Speed dynamic check [1]                     |
| V1050     | REAL | No   | Arms       | 0.01       | $V1050 * 1853609.716 / (I_{peak} * 100)$                          | Torque current limit [1]                    |
| V1051     | REAL | No   | mA/Rpm     | 0.01       | $V1051 * 5656.854 / (I_{peak} * 100)$                             | Proportional speed loop gain [1]            |
| V1052     | REAL | No   | ms         | 0.01       | $32767 * V264 / (V1052 * 1000)$                                   | Integral speed loop gain [1]                |
| V1053     | REAL | No   | ms         | 0.01       | $V264 * 32767 / (V1053 * 1000)$                                   | AP10: 2^ Integral speed loop gain [1]       |
| V1057     | REAL | No   | Arms       | 0.01       | $V1057 * 1853609.716 / (I_{peak} * 100)$                          | Torque current limit h.r. [1]               |
| V1058     | REAL | No   | mA/Rpm     | 0.01       | $V1058 * 5656.854 / (I_{peak} * 100)$                             | Proportional speed loop gain h.r. [1]       |
| V1059     | REAL | No   | ms         | 0.01       | $32767 * V264 / (V1059 * 1000)$                                   | Integral speed loop gain h.r. [1]           |
| V1060     | REAL | No   | Arms/m/min | 0.01       | $V1060 * 46340.95 / (I_{peak} * 100)$                             | Proportional speed loop gain [1]            |
| V1075     | REAL | No   | ms         | 0.01       | $V264 * 32767 / (V1075 * 1000)$                                   | AP10: 2^ Integral speed loop gain h.r. [1]  |
| V1091     | DINT | No   | Rpm/s      | 1          | $V1091 * V265 * 1.34218 / 1000$                                   | Acceleration/deceleration ramp [1]          |
| V1092     | DINT | No   | Rpm/s      | 1          | $V1092 * V265 * 1.34218 / 1000$                                   | Acceleration/deceleration ramp h.r.[1]      |
| V1093     | DINT | No   | mm/s^2     | 1          | $V1093 * 60 * V265 * 0.163840 / 1000$                             | Acceleration/deceleration ramp [1]          |
| V1094     | DINT | No   | %          | 1          | $V1094 * 163840 * V1095 / 100$                                    | Speed reference Limit [% maximum speed] [1] |
| V1095     | DINT | No   | m/min      | 1          | V1095*163840                                                      | Maximum motor speed [1]                     |
| V1096     | DINT | No   | m/min      | 1          | V1096*163840                                                      | Threshold for overspeed alarm [1]           |
| V1097     | DINT | No   | %          | 1          | $V1097 * 1342.18 * V1098 / 100$                                   | Speed reference Limit [% maximum speed] [1] |
| V1098     | DINT | No   | Rpm        | 1          | V1098*1342.18                                                     | Maximum motor speed [1]                     |
| V1099     | DINT | No   | Rpm        | 1          | V1099*1342.18                                                     | Overspeed alarm threshold [1]               |
| V1301     | DINT | No   | mm/min/mm  | 1          | V1301                                                             | Proportional position gain [1]              |
| V1305     | DINT | Yes  | [ ]        | 1          | V1305                                                             | Measure module for rot. axis/spindle abs[1] |
| V1306     | DINT | Yes  | [ ]        | 1          | V1306                                                             | S1 sensor MULTI coefficient [1]             |
| V1307     | DINT | Yes  | [ ]        | 1          | V1307                                                             | S1 sensor DIVI coefficient [1]              |
| V1310     | DINT | Yes  | [ ]        | 1          | V1310                                                             | S1 sensor offset in CNC Phys. Unit [1]      |
| V1311     | DINT | Yes  | [ ]        | 1          | V1311                                                             | S2 sensor MULTI coefficient [1]             |
| V1312     | DINT | Yes  | [ ]        | 1          | V1312                                                             | S2 sensor DIVI coefficient [1]              |
| V1313     | DINT | Yes  | [ ]        | 1          | V1313                                                             | S2 sensor offset in CNC Phys. Unit [1]      |
| V1353     | REAL | Yes  | [ ]        | 0.01       | V1353*100                                                         | Period of position sensor/motor turn [1]    |
| V1359     | DINT | Yes  | [ ]        | 1          | $(65536 * 3 * 4 * 1024 / V1353) * 100 / V264$                     | k_velrot121                                 |
| V1362     | DINT | No   | [ ]        | 1          | $V1301 * 1.31072 * 32768 * 4 / V360$                              | kpposrot11                                  |
| V1372     | DINT | No   | [ ]        | 1          | $V1301 * 1.31072 * 32768 * 4 / V1353$                             | kpposrot121                                 |
| V1376     | DINT | No   | [ ]        | 1          | $V1301 * 1.31072 * 32768 * 4 / V370$                              | kpposrot21                                  |
| V1383     | DINT | No   | [ ]        | 1          | $V1301 * 5.24288 * V380 * 4$                                      | kpposlin11                                  |
| V1393     | DINT | No   | [ ]        | 1          | $V1301 * 5.24288 * V390 * 4$                                      | kpposlin21                                  |

(\*): Applicable to the Mono-Axis drives only.

**Whenever is needed to change the parameters, due to Multi-Spindle application (one drive - different motor) with asynchronous motors controlled in sensorless mode, such parameters may be changed with the drive in Run Status and with Torque off.**

## A.2.2 Dependencies table

| Param. N° | Parameters impacted by this change                                                                                                  |
|-----------|-------------------------------------------------------------------------------------------------------------------------------------|
| V52       | V53                                                                                                                                 |
| V59       | V75                                                                                                                                 |
| V95       | V94                                                                                                                                 |
| V98       | V97                                                                                                                                 |
| V100      | V125, V180, V491                                                                                                                    |
| V105      | V114                                                                                                                                |
| V106      | V118                                                                                                                                |
| V108      | V109, V125, V128, V171, V180, V600, V605                                                                                            |
| V115      | V129                                                                                                                                |
| V116      | V117                                                                                                                                |
| V143      | V141, V142                                                                                                                          |
| V153      | V151, V152                                                                                                                          |
| V206      | V207                                                                                                                                |
| V208      | V209                                                                                                                                |
| V217      | V218, V219                                                                                                                          |
| V218      | V219                                                                                                                                |
| V223      | V224                                                                                                                                |
| V245      | V246                                                                                                                                |
| V257      | V98, V99, V104, V120, V414, V415, V417, V454, V455, V457, V496, V497, V1098, V1099                                                  |
| V259      | V120, V382, V386, V392, V396, V497                                                                                                  |
| V263      | V98, V99, V119, V144, V154, V264, V265, V266, V271, V1098, V1099                                                                    |
| V264      | V52, V53, V59, V75, V342, V343, V359, V363, V373, V384, V394, V411, V426, V451, V466, V498, V499, V1052, V1053, V1059, V1075, V1359 |
| V265      | V91, V92, V93, V173, V176, V177, V205, V218, V219, V223, V224, V1091, V1092, V1093                                                  |
| V271      | V98, V99, V103, V107, V112, V119, V400, V401, V403, V404, V405, V414, V415, V454, V455, V1098, V1099                                |
| V301      | V362, V372, V376, V383, V393                                                                                                        |
| V353      | V359, V372                                                                                                                          |
| V360      | V362, V363, V364, V1362                                                                                                             |
| V361      | V364                                                                                                                                |
| V370      | V373, V374, V376, V1376                                                                                                             |
| V371      | V374                                                                                                                                |
| V380      | V382, V383, V384, V385, V1383                                                                                                       |
| V381      | V385, V386                                                                                                                          |
| V390      | V392, V393, V394, V395, V1393                                                                                                       |
| V391      | V395, V396                                                                                                                          |
| V400      | V472, V473                                                                                                                          |
| V401      | V472, V473                                                                                                                          |
| V402      | V448, V470, V471                                                                                                                    |
| V403      | V400, V401, V402, V404, V405, V448                                                                                                  |
| V406      | V400, V401                                                                                                                          |
| V407      | V114, V117, V118, V402, V405, V448, V471, V473                                                                                      |
| V408      | V105, V106, V114, V118, V402, V448, V470, V471, V489                                                                                |
| V412      | V413, V425, V427                                                                                                                    |
| V427      | V425                                                                                                                                |
| V452      | V453, V465, V467                                                                                                                    |
| V467      | V465                                                                                                                                |

---

| Param.<br>N° | Parameters impacted by this change |
|--------------|------------------------------------|
| V491         | V492                               |
| V600         | V605                               |
| V601         | V606                               |
| V1052        | V1053                              |
| V1059        | V1075                              |
| V1095        | V1094                              |
| V1098        | V1097                              |
| V1301        | V1362, V1372, V1376, V1383, V1393  |
| V1353        | V1359, V1372                       |

---



The “Parameters impacted” are as parameters which have to be controlled after the main parameter is modified.  
For instance whenever V52 is modified you must modify V53 as indicated in the NUMDrive conversion table.

### A.2.3 NUMDrive C parameters format and limits

| Param. N° | Type  | Resolution | Min value [PU]           | Max value [PU]                                                                      |
|-----------|-------|------------|--------------------------|-------------------------------------------------------------------------------------|
| V20       | DWORD | 1          | 0                        | 3h                                                                                  |
| V21       | DWORD | 1          | 0                        | 3h                                                                                  |
| V22       | DWORD | 1          | 0                        | 3h                                                                                  |
| V23       | DINT  | 1          | 0                        | 31030                                                                               |
| V30       | DWORD | 1          | 0                        | 7FFFFFFh                                                                            |
| V32       | DWORD | 1          | 1610612736               | 7FFFFFFh                                                                            |
| V33       | DWORD | 1          | 0                        | 7FFFFFFh                                                                            |
| V34       | DWORD | 1          | 0                        | 7FFFFh                                                                              |
| V35       | DWORD | 1          | 0                        | 1FFFh                                                                               |
| V36       | DINT  | 1          | 1                        | 16                                                                                  |
| V37       | REAL  | 0.01       | 0.2                      | 2.0                                                                                 |
| V40       | REAL  | 0.01       | 0.0                      | 50.0                                                                                |
| V41       | REAL  | 0.01       | 1.0                      | 3000.0                                                                              |
| V42       | REAL  | 0.01       | 0.1                      | 5000.0                                                                              |
| V43       | REAL  | 0.01       | 1.0                      | 120.0                                                                               |
| V44       | DINT  | 1          | 0                        | 100                                                                                 |
| V45       | REAL  | 0.01       | 0.0                      | 20000.0                                                                             |
| V46       | REAL  | 0.01       | 0.0                      | 2000.0                                                                              |
| V50       | REAL  | 0.01       | 0.0                      | $(I_{peak} \cdot 100) / 141.42$                                                     |
| V51       | REAL  | 0.01       | 0.0                      | $32767 \cdot (I_{peak} \cdot 100) / 5656.854$                                       |
| V52       | REAL  | 0.01       | $2 \cdot V_{264} / 1000$ | $32767 \cdot V_{264} / (0.99 \cdot 1000)$                                           |
| V53       | REAL  | 0.01       | V52                      | $V_{264} \cdot 32767 / (0.99 \cdot 1000)$                                           |
| V57       | REAL  | 0.01       | 0.0                      | $(I_{peak} \cdot 100) / 141.42$                                                     |
| V58       | REAL  | 0.01       | 0.0                      | $32767 \cdot (I_{peak} \cdot 100) / 5656.854$                                       |
| V59       | REAL  | 0.01       | $2 \cdot V_{264} / 1000$ | $32767 \cdot V_{264} / (0.99 \cdot 1000)$                                           |
| V60       | REAL  | 0.01       | 0.0                      | $32767 \cdot (I_{peak} \cdot 100) / 46340.95$                                       |
| V75       | REAL  | 0.01       | V59                      | $V_{264} \cdot 32767 / (0.99 \cdot 1000)$                                           |
| V80       | REAL  | 0.01       | 0.0                      | 1.0                                                                                 |
| V81       | REAL  | 0.01       | 1.0                      | 2.0                                                                                 |
| V91       | DINT  | 1          | 4                        | 1000000                                                                             |
| V92       | DINT  | 1          | 4                        | 1000000                                                                             |
| V93       | DINT  | 1          | 5                        | 100000                                                                              |
| V94       | DINT  | 1          | 0                        | 200                                                                                 |
| V95       | DINT  | 1          | 0                        | 1000                                                                                |
| V96       | DINT  | 1          | 0                        | 1000                                                                                |
| V97       | DINT  | 1          | 0                        | 200                                                                                 |
| V98       | DINT  | 1          | 0                        | $140000 \cdot (V_{263} + 0.25 \cdot (V_{271} - V_{263})) / (V_{257} \cdot V_{271})$ |
| V99       | DINT  | 1          | 0                        | $140000 \cdot (V_{263} + 0.25 \cdot (V_{271} - V_{263})) / (V_{257} \cdot V_{271})$ |
| V100      | REAL  | 0.01       | 0.0                      | $(I_{peak} \cdot 100) / 141.42$                                                     |
| V101      | DINT  | 1          | 0                        | 1000                                                                                |
| V102      | REAL  | 0.01       | 0.0                      | $926790 \cdot 4 / (I_{peak} \cdot 100)$                                             |
| V103      | REAL  | 0.01       | $2 \cdot V_{271} / 1000$ | $32767 \cdot V_{271} / (1000 \cdot 0.99)$                                           |
| V104      | REAL  | 0.01       | 0.0                      | $32767 \cdot V_{257} / (16.237976 \cdot 2)$                                         |
| V105      | REAL  | 0.01       | 0.0                      | $(32767 / ((I_{peak} \cdot 100) \cdot 0.01503011016)) - V_{408}$                    |
| V106      | REAL  | 0.01       | 0.0                      | $(32767 / ((I_{peak} \cdot 100) \cdot 0.01503011016)) - V_{408}$                    |

| Param. N° | Type  | Resolution | Min value [PU]     | Max value [PU]                    |
|-----------|-------|------------|--------------------|-----------------------------------|
| V107      | DINT  | 1          | 1                  | 3000                              |
| V108      | REAL  | 0.01       | (Ipeak*100)/4715.0 | H1a5a/100                         |
| V109      | REAL  | 0.01       | -(V108)            | V108                              |
| V110      | REAL  | 0.01       | 0.0                | (Ipeak*100)/141.42                |
| V111      | REAL  | 0.01       | 0.0                | 926790*4/(Ipeak*100)              |
| V112      | REAL  | 0.01       | 2*V271/1000        | 32767*V271/(1000*0.99)            |
| V114      | REAL  | 0.01       | 0.0                | 32767.0                           |
| V115      | REAL  | 0.01       | (Ipeak*100)/4715.0 | H1a5a/100                         |
| V116      | REAL  | 0.01       | 0.0                | 32767/((Ipeak*100)*0.01503011016) |
| V117      | REAL  | 0.01       | 0.0                | 32767.0                           |
| V118      | REAL  | 0.01       | 0.0                | 32767.0                           |
| V119      | REAL  | 0.01       | 0.0                | 3.0                               |
| V120      | REAL  | 0.01       | 0.0                | 32767*50*V257/(V259*54.12659)     |
| V121      | REAL  | 0.01       | 0.0                | 1.0                               |
| V122      | REAL  | 0.01       | 1.0                | 2.0                               |
| V123      | DINT  | 1          | 1                  | 3                                 |
| V125      | DINT  | 1          | 0                  | (100.0*V100)/V108                 |
| V128      | DINT  | 1          | 13                 | 100                               |
| V129      | DINT  | 1          | 13                 | 100                               |
| V130      | DINT  | 1          | 1                  | 500                               |
| V131      | DINT  | 1          | 1                  | 500                               |
| V132      | DINT  | 1          | 1                  | 500                               |
| V133      | DINT  | 1          | 1                  | 500                               |
| V134      | DINT  | 1          | -2147483648        | 2147483647                        |
| V135      | DINT  | 1          | -2147483648        | 2147483647                        |
| V136      | DINT  | 1          | -2147483648        | 2147483647                        |
| V137      | DINT  | 1          | -2147483648        | 2147483647                        |
| V138      | DWORD | 1          | 0                  | Fh                                |
| V139      | DWORD | 1          | 0                  | 1FFh                              |
| V140      | DINT  | 1          | 1                  | 500                               |
| V141      | REAL  | 0.01       | -2147483648.0      | 2147483647.0                      |
| V142      | REAL  | 0.01       | -2147483648.0      | 2147483647.0                      |
| V143      | REAL  | 0.01       | -2147483648.0      | 2147483647.0                      |
| V144      | DINT  | 1          | 5                  | 1500                              |
| V150      | DINT  | 1          | 1                  | 500                               |
| V151      | REAL  | 0.01       | -2147483648.0      | 2147483647.0                      |
| V152      | REAL  | 0.01       | -2147483648.0      | 2147483647.0                      |
| V153      | REAL  | 0.01       | -2147483648.0      | 2147483647.0                      |
| V154      | DINT  | 1          | 5                  | 1500                              |
| V160      | DWORD | 1          | 0                  | 7FFh                              |
| V165      | DINT  | 1          | 0                  | 4000                              |
| V166      | REAL  | 0.01       | 0.0                | 32767*1000/(5.656941*(Ipeak*100)) |
| V168      | DINT  | 1          | 1                  | 1                                 |
| V169      | DWORD | 1          | 0                  | 3h                                |
| V170      | DWORD | 1          | 0                  | 3Fh                               |
| V171      | REAL  | 0.01       | -(V108)            | V108                              |
| V172      | REAL  | 0.01       | 0.0                | 32767*21.5792/(Ipeak*100)         |
| V173      | REAL  | 0.01       | 2*V265/1000        | 32767*V265/(1000*0.99)            |
| V175      | DINT  | 1          | 0                  | 400                               |

| Param. N° | Type   | Resolution | Min value [PU] | Max value [PU]              |
|-----------|--------|------------|----------------|-----------------------------|
| V176      | DINT   | 1          | 0              | (32767*V265)/1000           |
| V177      | REAL   | 0.01       | 0.1            | 10000.0                     |
| V180      | DINT   | 1          | 0              | V100*127/V108               |
| V182      | DINT   | 1          | 0              | 200                         |
| V183      | DINT   | 1          | 0              | 200                         |
| V190      | DINT   | 1          | 1              | 10000                       |
| V191      | DINT   | 1          | 0              | 1000                        |
| V205      | REAL   | 0.01       | 0.0            | 100.0                       |
| V206      | REAL   | 0.01       | 0.1            | (128.0*32768.0-1.0)/1342.18 |
| V207      | REAL   | 0.01       | -99.99         | 0.0                         |
| V208      | REAL   | 0.01       | 1.0            | (128.0*32768.0-1.0)/163.840 |
| V209      | REAL   | 0.01       | -99.99         | 0.0                         |
| V210      | REAL   | 0.01       | 0.0            | 8.0                         |
| V217      | REAL   | 0.01       | 0.1            | 6.0                         |
| V218      | REAL   | 0.01       | 2.0            | 150.0                       |
| V219      | DINT   | 1          | 0              | 32767                       |
| V220      | DWORD  | 1          | 0              | 3FFFh                       |
| V221      | DWORD  | 1          | 0              | 1Fh                         |
| V222      | DINT   | 1          | -15            | 15                          |
| V223      | DINT   | 1          | 1              | 318309.8/V265               |
| V224      | DINT   | 1          | 0              | 318309.8/V265               |
| V225      | DWORD  | 1          | 0              | 7Fh                         |
| V238      | DWORD  | 1          | 0              | 3FFFh                       |
| V239      | DWORD  | 1          | 0              | 3FFFh                       |
| V240      | REAL   | 0.01       | 0.01           | 2.0                         |
| V241      | DINT   | 1          | 1              | 400                         |
| V245      | REAL   | 0.01       | 1.25           | 32767/13.1072               |
| V246      | REAL   | 0.01       | 0.25           | V245-1.00                   |
| V250      | DWORD  | 1          | 0              | Fh                          |
| V251      | DINT   | 1          | 0              | 7                           |
| V252      | STRING |            | 7              | Z                           |
| V253      | DWORD  | 1          | 0              | Fh                          |
| V254      | DINT   | 1          | 0              | 999                         |
| V255      | STRING |            | 0              | 2                           |
| V256      | STRING |            | 0              | V                           |
| V257      | DINT   | 1          | 1              | 32767                       |
| V258      | STRING |            | 0              | z                           |
| V259      | REAL   | 0.01       | 0.0            | 32767.0                     |
| V260      | DINT   | 1          | 0              | 1                           |
| V261      | DINT   | 1          | 0              | 1                           |
| V263      | DINT   | 50         | 50             | 50                          |
| V264      | DINT   | 100        | 100            | 100                         |
| V265      | DINT   | 200        | 200            | 200                         |
| V266      | REAL   | 0.2        | 2.0            | 20.0                        |
| V267      | DINT   | 1          | 7              | 400                         |
| V268      | DINT   | 1          | 0              | 359999                      |
| V269      | DINT   | 1          | 0              | 359999                      |
| V270      | DINT   | 1          | 0              | 7000                        |
| V271      | DINT   | 50         | 50             | 100                         |

| Param. N° | Type  | Resolution | Min value [PU] | Max value [PU]     |
|-----------|-------|------------|----------------|--------------------|
| V272      | DWORD | 1          | 0              | 7h                 |
| V273      | DWORD | 1          | 0              | 7Fh                |
| V301      | DINT  | 1          | 0              | 50000              |
| V305      | DINT  | 1          | 0              | 2147483647         |
| V306      | DINT  | 1          | -32767         | 32767              |
| V307      | DINT  | 1          | 1              | 32767              |
| V310      | DINT  | 1          | -2147483648    | 2147483647         |
| V311      | DINT  | 1          | -32767         | 32767              |
| V312      | DINT  | 1          | 1              | 32767              |
| V313      | DINT  | 1          | -2147483648    | 2147483647         |
| V314      | DWORD | 1          | 0              | 3Fh                |
| V315      | DWORD | 1          | 0              | 3Fh                |
| V317      | REAL  | 0.01       | 1.0            | 9.2                |
| V318      | REAL  | 0.01       | 2.0            | 9.2                |
| V340      | DINT  | 1          | 0              | 4                  |
| V341      | DINT  | 1          | 0              | 4                  |
| V342      | REAL  | 0.01       | 50.0           | 3000.00*100.0/V264 |
| V343      | REAL  | 0.01       | 50.0           | 3000.00*100.0/V264 |
| V344      | DINT  | 1          | -2147483648    | 2147483647         |
| V345      | DINT  | 1          | -2147483648    | 2147483647         |
| V346      | DINT  | 1          | 0              | 9999               |
| V347      | DINT  | 1          | 0              | 9999               |
| V348      | DWORD | 1          | 0              | 1FFFFFFh           |
| V349      | DWORD | 1          | 0              | 1FFFFFFh           |
| V350      | DWORD | 1          | 0              | 1h                 |
| V351      | DWORD | 1          | 0              | 1h                 |
| V352      | DWORD | 1          | 0              | 1h                 |
| V353      | REAL  | 0.01       | 1.0            | 2147483000/100     |
| V355      | DINT  | 1          | 0              | 32767              |
| V356      | DINT  | 1          | 1              | 32767              |
| V359      | DINT  | 1          | 0              | 32767              |
| V360      | DINT  | 1          | 1              | 131072             |
| V361      | DINT  | 1          | 1              | 2147483647         |
| V362      | DINT  | 1          | 0              | 32767              |
| V363      | DINT  | 1          | 0              | 32767              |
| V364      | DINT  | 1          | 0              | 32767              |
| V365      | DINT  | 1          | 1              | 32767              |
| V367      | DINT  | 1          | 0              | 2147483647/4       |
| V368      | DINT  | 1          | 1              | 32767              |
| V370      | DINT  | 1          | 1              | 131072             |
| V371      | DINT  | 1          | 1              | 2147483647         |
| V372      | DINT  | 1          | 0              | 32767              |
| V373      | DINT  | 1          | 0              | 32767              |
| V374      | DINT  | 1          | 0              | 32767              |
| V375      | DINT  | 1          | 1              | 32767              |
| V376      | DINT  | 1          | 0              | 32767              |
| V377      | DINT  | 1          | 0              | 2147483647/4       |
| V378      | DINT  | 1          | 1              | 32767              |
| V380      | DINT  | 1          | 1              | 8000               |

| Param. N° | Type | Resolution | Min value [PU]                                                                   | Max value [PU]                                                            |
|-----------|------|------------|----------------------------------------------------------------------------------|---------------------------------------------------------------------------|
| V381      | DINT | 1          | 1                                                                                | 32767                                                                     |
| V382      | DINT | 1          | 0                                                                                | 2147483647                                                                |
| V383      | DINT | 1          | 0                                                                                | 32767                                                                     |
| V384      | DINT | 1          | 0                                                                                | 32767                                                                     |
| V385      | DINT | 1          | 0                                                                                | 32767                                                                     |
| V386      | DINT | 1          | 0                                                                                | 32767                                                                     |
| V390      | DINT | 1          | 1                                                                                | 8000                                                                      |
| V391      | DINT | 1          | 1                                                                                | 32767                                                                     |
| V392      | DINT | 1          | 0                                                                                | 2147483647                                                                |
| V393      | DINT | 1          | 0                                                                                | 32767                                                                     |
| V394      | DINT | 1          | 0                                                                                | 2147483647                                                                |
| V395      | DINT | 1          | 0                                                                                | 32767                                                                     |
| V396      | DINT | 1          | 0                                                                                | 32767                                                                     |
| V400      | DINT | 1          | 0                                                                                | $32767 \cdot 50 \cdot V403 / (3.2768 \cdot V406 \cdot V271)$              |
| V401      | DINT | 1          | 0                                                                                | $32767 \cdot 50 \cdot V403 / (3.2768 \cdot V406 \cdot V271)$              |
| V402      | DINT | 1          | 0                                                                                | $(32767 \cdot V403 / 65.536) - V408 \cdot 1000 \cdot V407$                |
| V403      | REAL | 0.01       | $133774.75 \cdot 3276.8 \cdot (V271/50) / ((I_{peak} \cdot 100) \cdot 32767.00)$ | 32767.0                                                                   |
| V404      | DINT | 1          | 0                                                                                | 32767                                                                     |
| V405      | DINT | 1          | 0                                                                                | 32767                                                                     |
| V406      | DINT | 1          | 1                                                                                | 16                                                                        |
| V407      | DINT | 1          | 3                                                                                | 4                                                                         |
| V408      | REAL | 0.01       | 0.0                                                                              | 20.0                                                                      |
| V409      | DINT | 1          | 0                                                                                | 700/1.4142135                                                             |
| V410      | REAL | 0.01       | 0.0                                                                              | 127.9                                                                     |
| V411      | REAL | 0.01       | $2 \cdot V264 / 1000$                                                            | $V264 \cdot 32.767 / 0.99$                                                |
| V412      | DINT | 1          | 2                                                                                | 100                                                                       |
| V413      | DINT | 1          | 0                                                                                | 32767                                                                     |
| V414      | DINT | 1          | 0                                                                                | $32767 \cdot 100 / (0.3431457 \cdot V257 \cdot V271)$                     |
| V415      | DINT | 1          | 0                                                                                | $32767 \cdot 100 / (0.3431457 \cdot V257 \cdot V271)$                     |
| V417      | DINT | 1          | 0                                                                                | $32767 / (V257 \cdot 0.21845333)$                                         |
| V425      | DINT | 1          | 1                                                                                | 20                                                                        |
| V426      | REAL | 0.01       | 0.0                                                                              | $32767 \cdot (I_{peak} \cdot 100) / (7592.3856 \cdot V264)$               |
| V427      | REAL | 0.01       | 0.0                                                                              | $13107 \cdot (I_{peak} \cdot 100) / (V412 \cdot 13107.2 \cdot 1.4142135)$ |
| V428      | REAL | 0.01       | 0.0                                                                              | $(I_{peak} \cdot 100) / 141.42$                                           |
| V448      | DINT | 1          | 0                                                                                | 32767                                                                     |
| V450      | REAL | 0.01       | 0.0                                                                              | 127.9                                                                     |
| V451      | REAL | 0.01       | $2 \cdot V264 / 1000$                                                            | $V264 \cdot 32.767 / 0.99$                                                |
| V452      | DINT | 1          | 0                                                                                | 100                                                                       |
| V453      | DINT | 1          | 0                                                                                | 32767                                                                     |
| V454      | DINT | 1          | 0                                                                                | $32767 \cdot 100 / (0.3431457 \cdot V257 \cdot V271)$                     |
| V455      | DINT | 1          | 0                                                                                | $32767 \cdot 100 / (0.3431457 \cdot V257 \cdot V271)$                     |
| V457      | DINT | 1          | 0                                                                                | $32767 / (V257 \cdot 0.21845333)$                                         |
| V465      | DINT | 1          | 1                                                                                | 20                                                                        |
| V466      | REAL | 0.01       | 0.0                                                                              | $32767 \cdot (I_{peak} \cdot 100) / (7592.3856 \cdot V264)$               |
| V467      | REAL | 0.01       | 0.0                                                                              | $13107 \cdot (I_{peak} \cdot 100) / (V452 \cdot 13107.2 \cdot 1.4142135)$ |
| V468      | REAL | 0.01       | 0.0                                                                              | $(I_{peak} \cdot 100) / 141.42$                                           |
| V470      | REAL | 0.01       | 0.0                                                                              | 32767.0                                                                   |
| V471      | REAL | 0.01       | 0.0                                                                              | 32767.0                                                                   |

| Param. N° | Type  | Resolution | Min value [PU] | Max value [PU]                                                                             |
|-----------|-------|------------|----------------|--------------------------------------------------------------------------------------------|
| V472      | DINT  | 1          | 0              | 32767                                                                                      |
| V473      | DINT  | 1          | 0              | 32767                                                                                      |
| V480      | REAL  | 0.01       | 0.0            | 100.0                                                                                      |
| V481      | DWORD | 1          | 0              | 7h                                                                                         |
| V488      | DINT  | 1          | 0              | $32767 * ((234.5 + 20.0) * 2 * 133772.76) / ((I_{peak} * 100) * 65.536 * (234.5 + 120.0))$ |
| V489      | REAL  | 0.01       | 0.0            | $(32767 / ((I_{peak} * 100) * 0.01503011016)) - V408$                                      |
| V491      | REAL  | 0.01       | -V100          | 0.0                                                                                        |
| V492      | REAL  | 0.01       | V491           | 0.0                                                                                        |
| V496      | DINT  | 1          | 0              | $32767 / (0.21845333 * V257)$                                                              |
| V497      | REAL  | 0.01       | 0.0            | $32767 / (109.22667 * V257 / V259)$                                                        |
| V498      | REAL  | 0.01       | 0.0            | $32767 * (I_{peak} * 100) / (7592.3856 * V264)$                                            |
| V499      | REAL  | 0.01       | 0.0            | $32767 * (I_{peak} * 100) / (7592.3856 * V264)$                                            |
| V500      | DWORD | 1          | 0              | 1FFh                                                                                       |
| V501      | DINT  | 1          | -32767         | 32767                                                                                      |
| V502      | DINT  | 1          | 0              | 65535                                                                                      |
| V503      | DINT  | 1          | 0              | 65535                                                                                      |
| V504      | DINT  | 1          | -32767         | 32767                                                                                      |
| V505      | DINT  | 1          | -32767         | 32767                                                                                      |
| V506      | DINT  | 1          | -32767         | 32767                                                                                      |
| V507      | DINT  | 1          | 0              | 32767                                                                                      |
| V508      | DINT  | 1          | -32767         | 32767                                                                                      |
| V509      | DINT  | 1          | -32767         | 32767                                                                                      |
| V511      | DINT  | 1          | -32767         | 32767                                                                                      |
| V512      | DINT  | 1          | 0              | 65535                                                                                      |
| V513      | DINT  | 1          | 0              | 65535                                                                                      |
| V514      | DINT  | 1          | -32767         | 32767                                                                                      |
| V515      | DINT  | 1          | -32767         | 32767                                                                                      |
| V516      | DINT  | 1          | -32767         | 32767                                                                                      |
| V517      | DINT  | 1          | 0              | 32767                                                                                      |
| V518      | DINT  | 1          | -32767         | 32767                                                                                      |
| V519      | DINT  | 1          | -32767         | 32767                                                                                      |
| V521      | DINT  | 1          | -32767         | 32767                                                                                      |
| V522      | DINT  | 1          | 0              | 65535                                                                                      |
| V523      | DINT  | 1          | 0              | 65535                                                                                      |
| V524      | DINT  | 1          | -32767         | 32767                                                                                      |
| V525      | DINT  | 1          | -32767         | 32767                                                                                      |
| V526      | DINT  | 1          | -32767         | 32767                                                                                      |
| V527      | DINT  | 1          | 0              | 32767                                                                                      |
| V528      | DINT  | 1          | -32767         | 32767                                                                                      |
| V529      | DINT  | 1          | -32767         | 32767                                                                                      |
| V531      | DINT  | 1          | -32767         | 32767                                                                                      |
| V532      | DINT  | 1          | 0              | 65535                                                                                      |
| V533      | DINT  | 1          | 0              | 65535                                                                                      |
| V534      | DINT  | 1          | -32767         | 32767                                                                                      |
| V535      | DINT  | 1          | -32767         | 32767                                                                                      |
| V536      | DINT  | 1          | -32767         | 32767                                                                                      |
| V537      | DINT  | 1          | 0              | 32767                                                                                      |
| V538      | DINT  | 1          | -32767         | 32767                                                                                      |

| Param. N° | Type | Resolution | Min value [PU]           | Max value [PU]                                                                      |
|-----------|------|------------|--------------------------|-------------------------------------------------------------------------------------|
| V539      | DINT | 1          | -32767                   | 32767                                                                               |
| V541      | DINT | 1          | -32767                   | 32767                                                                               |
| V542      | DINT | 1          | 0                        | 65535                                                                               |
| V543      | DINT | 1          | 0                        | 65535                                                                               |
| V544      | DINT | 1          | -32767                   | 32767                                                                               |
| V545      | DINT | 1          | -32767                   | 32767                                                                               |
| V546      | DINT | 1          | -32767                   | 32767                                                                               |
| V547      | DINT | 1          | 0                        | 32767                                                                               |
| V548      | DINT | 1          | -32767                   | 32767                                                                               |
| V549      | DINT | 1          | -32767                   | 32767                                                                               |
| V551      | DINT | 1          | -32767                   | 32767                                                                               |
| V552      | DINT | 1          | 0                        | 65535                                                                               |
| V553      | DINT | 1          | 0                        | 65535                                                                               |
| V554      | DINT | 1          | -32767                   | 32767                                                                               |
| V555      | DINT | 1          | -32767                   | 32767                                                                               |
| V556      | DINT | 1          | -32767                   | 32767                                                                               |
| V557      | DINT | 1          | 0                        | 32767                                                                               |
| V558      | DINT | 1          | -32767                   | 32767                                                                               |
| V559      | DINT | 1          | -32767                   | 32767                                                                               |
| V600      | DINT | 1          | 0                        | $(I_{peak} \cdot 100) / (V_{108} \cdot 1.4142)$                                     |
| V601      | DINT | 1          | 10                       | 1500                                                                                |
| V602      | REAL | 0.01       | 100.0                    | 500.0                                                                               |
| V603      | REAL | 0.01       | 0.0                      | 50.0                                                                                |
| V604      | DINT | 1          | 0                        | $32767 \cdot 133772.76 / ((I_{peak} \cdot 100) \cdot 65.536 \cdot 1.4)$             |
| V605      | DINT | 1          | 0                        | $V_{600} \cdot 91 / 100$                                                            |
| V606      | DINT | 1          | $V_{601} / 2$            | 1500                                                                                |
| V1045     | REAL | 0.01       | 0.0                      | 20000.0                                                                             |
| V1046     | REAL | 0.01       | 0.0                      | 2000.0                                                                              |
| V1050     | REAL | 0.01       | 0.0                      | $(I_{peak} \cdot 100) / 141.42$                                                     |
| V1051     | REAL | 0.01       | 0.0                      | $32767 \cdot (I_{peak} \cdot 100) / 5656.854$                                       |
| V1052     | REAL | 0.01       | $2 \cdot V_{264} / 1000$ | $32767 \cdot V_{264} / (0.99 \cdot 1000)$                                           |
| V1053     | REAL | 0.01       | $V_{1052}$               | $V_{264} \cdot 32767 / (0.99 \cdot 1000)$                                           |
| V1057     | REAL | 0.01       | 0.0                      | $(I_{peak} \cdot 100) / 141.42$                                                     |
| V1058     | REAL | 0.01       | 0.0                      | $32767 \cdot (I_{peak} \cdot 100) / 5656.854$                                       |
| V1059     | REAL | 0.01       | $2 \cdot V_{264} / 1000$ | $32767 \cdot V_{264} / (0.99 \cdot 1000)$                                           |
| V1060     | REAL | 0.01       | 0.0                      | $32767 \cdot (I_{peak} \cdot 100) / 46340.95$                                       |
| V1075     | REAL | 0.01       | $V_{1059}$               | $V_{264} \cdot 32767 / (0.99 \cdot 1000)$                                           |
| V1091     | DINT | 1          | 4                        | 1000000                                                                             |
| V1092     | DINT | 1          | 4                        | 1000000                                                                             |
| V1093     | DINT | 1          | 5                        | 100000                                                                              |
| V1094     | DINT | 1          | 0                        | 200                                                                                 |
| V1095     | DINT | 1          | 0                        | 1000                                                                                |
| V1096     | DINT | 1          | 0                        | 1000                                                                                |
| V1097     | DINT | 1          | 0                        | 200                                                                                 |
| V1098     | DINT | 1          | 0                        | $140000 \cdot (V_{263} + 0.25 \cdot (V_{271} - V_{263})) / (V_{257} \cdot V_{271})$ |
| V1099     | DINT | 1          | 0                        | $140000 \cdot (V_{263} + 0.25 \cdot (V_{271} - V_{263})) / (V_{257} \cdot V_{271})$ |
| V1301     | DINT | 1          | 0                        | 50000                                                                               |
| V1305     | DINT | 1          | 0                        | 2147483647                                                                          |
| V1306     | DINT | 1          | -32767                   | 32767                                                                               |

---

| Param.<br>N° | Type | Resolution | Min value [PU] | Max value [PU] |
|--------------|------|------------|----------------|----------------|
| V1307        | DINT | 1          | 1              | 32767          |
| V1310        | DINT | 1          | -2147483648    | 2147483647     |
| V1311        | DINT | 1          | -32767         | 32767          |
| V1312        | DINT | 1          | 1              | 32767          |
| V1313        | DINT | 1          | -2147483648    | 2147483647     |
| V1353        | REAL | 0.01       | 1.0            | 2147483000/100 |
| V1359        | DINT | 1          | 0              | 32767          |
| V1362        | DINT | 1          | 0              | 32767          |
| V1372        | DINT | 1          | 0              | 32767          |
| V1376        | DINT | 1          | 0              | 32767          |
| V1383        | DINT | 1          | 0              | 32767          |
| V1393        | DINT | 1          | 0              | 32767          |

### A.3 NUMDrive X parameters Conversion rules, Dependencies, Limits

- Modifying a drive parameter requires to define its value in both Physical and Internal units. The following table describes the conversion rule between them.
- The Halt column indicates if the drive must be in Halt or not to accept the new value. Please note that some of those parameters are not visible with Flexium Tools or WPM; they are listed here for dependencies reasons.
- The second table defines the dependencies between parameters. That is a parameter value has an effect on the values of other parameters.
- The third table defines the limit values of all parameters. When, changing the value of a parameter, the programmer must pay attention not to exceed the limit of all depending parameters.

#### A.3.1 Conversion rules table

| Param. N° | Type  | Halt | [PU]       | Resolution | Conversion rule from Vxx in physical unit to Vxx in internal unit | Description                                 |
|-----------|-------|------|------------|------------|-------------------------------------------------------------------|---------------------------------------------|
| V25       | DWORD | Yes  | [ ]        |            | V25                                                               | Port Ix1 configuration                      |
| V26       | DWORD | Yes  | [ ]        |            | V26                                                               | Port Ix2 configuration                      |
| V27       | DWORD | Yes  | [ ]        |            | V27                                                               | Port Ix3 configuration                      |
| V28       | DWORD | Yes  | [ ]        |            | V28                                                               | Port I/Ox4 configuration                    |
| V29       | DWORD | Yes  | [ ]        |            | V29                                                               | Port Ox5 configuration                      |
| V30       | DWORD | Yes  | [ ]        | 1          | V30                                                               | General drive configuration                 |
| V32       | DWORD | Yes  | [ ]        | 1          | V32                                                               | Config. braking following alarms [1=Brake]  |
| V33       | DWORD | Yes  | [ ]        | 1          | V33                                                               | Alarm reset configuration [1=Reset]         |
| V34       | DWORD | Yes  | [ ]        | 1          | V34                                                               | Configuration DF1/DF2 relay [1=Open]        |
| V35       | DWORD | Yes  | [ ]        | 1          | V35                                                               | SW functions [0=Not available][1=Available] |
| V36       | UDINT | Yes  | [ ]        | 1          | V36                                                               | FSoE address SAMX                           |
| V37       | REAL  | Yes  | s          | 0.01       | $V37 \cdot 1000$                                                  | Delay time of power detection               |
| V39       | REAL  | No   | %          | 0.01       | $V39 \cdot 1024 / 100$                                            | Bump suppressor speed limit                 |
| V40       | REAL  | No   | %          | 0.01       | $V40 \cdot 65536 / 100$                                           | Speed reached dynamic threshold             |
| V41       | REAL  | No   | mm/min     | 0.01       | $V41 \cdot 163.840 \cdot (V263 / 50)$                             | Motor zero speed threshold, enable, brake   |
| V42       | REAL  | No   | rpm        | 0.01       | $V42 \cdot 1342.18 \cdot (V263 / 50)$                             | Motor zero speed threshold, enable, brake   |
| V43       | REAL  | Yes  | s          | 0.01       | $V43 \cdot 1000$                                                  | Braking Timeout on config. V30, V32         |
| V45       | REAL  | No   | mm/min     | 0.01       | $V45 \cdot 163.840 \cdot (V263 / 50)$                             | Speed dynamic check [0]                     |
| V46       | REAL  | No   | rpm        | 0.01       | $V46 \cdot 1342.18 \cdot (V263 / 50)$                             | Speed dynamic check [0]                     |
| V50       | REAL  | No   | Arms       | 0.01       | $V50 \cdot 1853609.716 / (I_{peak} \cdot 100)$                    | Torque current limit [0]                    |
| V51       | REAL  | No   | mArms/rpm  | 0.01       | $V51 \cdot 90509.668 \cdot (50 / V263) / (I_{peak} \cdot 100)$    | Proportional speed loop gain [0]            |
| V52       | REAL  | No   | ms         | 0.01       | $32767 \cdot (2 \cdot V263) / (V52 \cdot 1000)$                   | Integral speed loop gain [0]                |
| V57       | REAL  | No   | Arms       | 0.01       | $V57 \cdot 1853609.716 / (I_{peak} \cdot 100)$                    | Torque current limit h.r. [0]               |
| V58       | REAL  | No   | mArms/rpm  | 0.01       | $V58 \cdot 90509.668 \cdot (50 / V263) / (I_{peak} \cdot 100)$    | Proportional speed loop gain h.r. [0]       |
| V59       | REAL  | No   | ms         | 0.01       | $32767 \cdot (2 \cdot V263) / (V59 \cdot 1000)$                   | Integral speed loop gain h.r. [0]           |
| V60       | REAL  | No   | Arms/m/min | 0.01       | $V60 \cdot 741455.2 \cdot (50 / V263) / (I_{peak} \cdot 100)$     | Proportional speed loop gain [0]            |
| V80       | REAL  | No   | [ ]        | 0.01       | $32768.0 \cdot V80$                                               | Set Point Weight speed loop gain            |

| Param. N° | Type  | Halt | [PU]      | Resolution | Conversion rule from Vxx in physical unit to Vxx in internal unit | Description                            |
|-----------|-------|------|-----------|------------|-------------------------------------------------------------------|----------------------------------------|
| V81       | REAL  | No   | [ ]       | 0.01       | 16384.0*V81                                                       | Anti-windup speed loop gain            |
| V91       | UDINT | No   | rpm/s     | 1          | $V91*(4*V263)*1.34218*(V263/50)/1000$                             | Acceleration/deceleration ramp [0]     |
| V92       | UDINT | No   | rpm/s     | 1          | $V92*(4*V263)*1.34218*(V263/50)/1000$                             | Acceleration/deceleration ramp h.r.[0] |
| V93       | UDINT | No   | mm/s^2    | 1          | $V93*60*(4*V263)*0.16384*(V263/50)/1000$                          | Acceleration/deceleration ramp [0]     |
| V94       | UDINT | No   | %         | 1          | $V94*163840*V95/100$                                              | Speed reference Limit (% V95) [0]      |
| V95       | UDINT | No   | m/min     | 1          | $V95*163840*(V263/50)$                                            | Maximum motor speed [0]                |
| V96       | UDINT | No   | m/min     | 1          | $V96*163840*(V263/50)$                                            | Overspeed alarm threshold [0]          |
| V97       | UDINT | No   | %         | 1          | $V97*1342.18*V98/100$                                             | Speed reference Limit (% V98) [0]      |
| V98       | UDINT | No   | rpm       | 1          | $V98*1342.18*(V263/50)$                                           | Maximum motor speed [0]                |
| V99       | UDINT | No   | rpm       | 1          | $V99*1342.18*(V263/50)$                                           | Overspeed alarm threshold [0]          |
| V100      | REAL  | No   | Arms      | 0.01       | $V100*1853609.716/(I_{peak}*100)$                                 | Deliverable current limit              |
| V101      | UDINT | No   | Vrms      | 1          | $V101*8$                                                          | Voltage limits Vd and Vq               |
| V102      | REAL  | No   | Vrms/Arms | 0.01       | $(V102*(I_{peak}*100))/28.28427/4$                                | Proportional Iq current loop gain      |
| V103      | REAL  | No   | ms        | 0.01       | $32767*V271/(1000*V103)$                                          | Integral Iq current loop gain          |
| V104      | REAL  | No   | mVrms/rpm | 0.01       | $V104*16.237976*2*(50/V263)/V257$                                 | Motor EMF phase compensation           |
| V105      | REAL  | No   | mH        | 0.01       | $(V105+V408)*(I_{peak}*100)*0.01503011016*(50/V263)$              | Compensation of motor Lq inductance    |
| V106      | REAL  | No   | mH        | 0.01       | $(V106+V408)*(I_{peak}*100)*0.01503011016*(50/V263)$              | Compensation of motor Ld inductance    |
| V107      | UDINT | Yes  | Hz        | 1          | $(0.102943708*V271*V107)/(1+0.00000314159*V271*V107)$             | Lowpass Filter electrical speed        |
| V108      | REAL  | No   | Arms      | 0.01       | $V108*1853609.716/(I_{peak}*100)$                                 | Motor rated current [I^2t]             |
| V109      | REAL  | No   | Arms      | 0.01       | $V109*1853609.716/(I_{peak}*100)$                                 | Vertical load compensation current     |
| V110      | REAL  | No   | Arms      | 0.01       | $V110*1853609.716/(I_{peak}*100)$                                 | Deliverable current limit h.r.         |
| V111      | REAL  | No   | Vrms/Arms | 0.01       | $(V111*(I_{peak}*100))/28.28427/4$                                | Proportional Iq current gain h.r.      |
| V112      | REAL  | No   | ms        | 0.01       | $32767*V271/(1000*V112)$                                          | Integral Iq current gain h.r.          |
| V114      | REAL  | No   | mH        | 0.01       | $((V105*1.00/V407)+V408)*(I_{peak}*100)*0.01503011016*(50/V263)$  | Motor Lq inductance compensation h.r.  |
| V115      | REAL  | No   | Arms      | 0.01       | $V115*1853609.716/(I_{peak}*100)$                                 | Motor rated current [I^2t] h.r.        |
| V116      | REAL  | No   | mH        | 0.01       | $V116*(I_{peak}*100)*0.01503011016*(50/V263)$                     | Lm (for compensation)                  |
| V117      | REAL  | No   | mH        | 0.01       | $V116*(I_{peak}*100)*0.01503011016*(50/V263)/V407$                | Lm (for compensation) h.r.             |
| V118      | REAL  | No   | mH        | 0.01       | $((V106*1.00/V407)+V408)*(I_{peak}*100)*0.01503011016*(50/V263)$  | Motor Ld inductance compensation h.r.  |
| V119      | REAL  | No   | cycles    | 0.01       | $V119*(V271/50)*1024$                                             | Phase lead                             |
| V120      | REAL  | No   | Vrms/m/s  | 0.01       | $V120*V259*54.12659/(V257*V263)$                                  | Linear Motor EMF phase compensation    |
| V121      | REAL  | No   | [ ]       | 0.01       | $(1.0-V121)*((I_{peak}*100)*(I_{peak}*100))/(100.0*V123*V123)$    | Iq gain reduction at V123 current      |
| V122      | REAL  | No   | [ ]       | 0.01       | $(1.0-V122)*((I_{peak}*100)*(I_{peak}*100))/(100.0*V124*V124)$    | Iq gain reduction at V124 current h.r. |
| V123      | REAL  | No   | Arms      | 0.01       | $V123*1853609.716/(I_{peak}*100)$                                 | Maximal motor current                  |
| V124      | REAL  | No   | Arms      | 0.01       | $V124*1853609.716/(I_{peak}*100)$                                 | Maximal motor current h.r.             |
| V125      | UDINT | No   | %         | 1          | $V125*V108*18536.09716/(I_{peak}*100)$                            | Current percentage V108 for EPSM       |
| V126      | UDINT | No   | %         | 1          | $(V126*V108*13107)/(I_{peak}*100)$                                | Current percentage V108 for EPS        |
| V128      | UDINT | No   | %         | 1          | $V128*V108*18536.09716/(I_{peak}*100)$                            | Locked rotor current I^2t (% V108)     |

| Param. N° | Type  | Halt | [PU]            | Resolution | Conversion rule from Vxx in physical unit to Vxx in internal unit                        | Description                                        |
|-----------|-------|------|-----------------|------------|------------------------------------------------------------------------------------------|----------------------------------------------------|
| V129      | UDINT | No   | %               | 1          | $V129 \cdot V115 \cdot 18536.09716 / (I_{peak} \cdot 100)$                               | Locked rotor current $I^{A2t}$ (% V115) h.r.       |
| V130      | UDINT | No   | []              | 1          | V130                                                                                     | Associate MP with TP1                              |
| V131      | UDINT | No   | []              | 1          | V131                                                                                     | Associate MP with TP2                              |
| V132      | UDINT | No   | []              | 1          | V132                                                                                     | Associate MP with TP3                              |
| V133      | UDINT | No   | []              | 1          | V133                                                                                     | Associate MP with TP4                              |
| V135      | DWORD | Yes  | []              | 1          | V135                                                                                     | Configuration of braking relay                     |
| V136      | UDINT | No   | ms              | 1          | V136                                                                                     | Brake ON delay                                     |
| V137      | UDINT | No   | ms              | 1          | V137                                                                                     | Brake OFF delay                                    |
| V140      | UDINT | No   | []              | 1          | V140                                                                                     | Associate MP with programmable relay 1             |
| V141      | DINT  | No   | []              | 1          | V141                                                                                     | Programmable relay 1, threshold 1                  |
| V142      | DINT  | No   | []              | 1          | V142                                                                                     | Programmable relay 1, threshold 2                  |
| V144      | UDINT | No   | ms              | 1          | $V144 / (V263/1000)$                                                                     | Programm. relay 1, tripping time constant          |
| V145      | DWORD | No   | []              | 1          | V145                                                                                     | Configuration of programmable relay 1              |
| V150      | UDINT | No   | []              | 1          | V150                                                                                     | Associate MP with programmable relay 2             |
| V151      | DINT  | No   | []              | 1          | V151                                                                                     | Programmable relay 2, threshold 1                  |
| V152      | DINT  | No   | []              | 1          | V152                                                                                     | Programmable relay 2, threshold 2                  |
| V154      | UDINT | No   | ms              | 1          | $V154 / (V263/1000)$                                                                     | Programm. relay 2, tripping time constant          |
| V155      | DWORD | No   | []              | 1          | V155                                                                                     | Configuration of programmable relay 2              |
| V160      | DWORD | Yes  | []              | 1          | V160                                                                                     | Special applications selection                     |
| V162      | UDINT | Yes  | []              | 1          | V162                                                                                     | AP02: Mechanical ratio: load turns                 |
| V163      | UDINT | Yes  | []              | 1          | V163                                                                                     | AP02: Mechanical ratio: sensor turns               |
| V164      | DWORD | Yes  | []              | 1          | V164                                                                                     | AP02: configuration                                |
| V165      | UDINT | No   | mm/min          | 1          | $V165 \cdot 163.840 \cdot (V263/50)$                                                     | AP03 Mst: Max compens. torque equalizer            |
| V166      | REAL  | No   | mm/min/<br>Arms | 0.01       | $V166 \cdot (I_{peak} \cdot 100) \cdot 5.656941 \cdot (V263/50) / 1000$                  | AP03 Mst: Proportional torq. equaliz. gain         |
| V169      | DWORD | Yes  | []              | 1          | V169                                                                                     | AP03: Anti-backlash role                           |
| V170      | DWORD | Yes  | []              | 1          | V170                                                                                     | AP03: Anti-backlash configuration                  |
| V171      | REAL  | No   | Arms            | 0.01       | $-2 \cdot V171 \cdot 1853609.716 / (I_{peak} \cdot 100)$                                 | AP03 Master: Preload Tandem current                |
| V172      | REAL  | No   | rpm/<br>Arms    | 0.01       | $V172 \cdot (I_{peak} \cdot 100) \cdot (V263/50) / 21.5792$                              | AP03 Mst: Proportional torq. equaliz. gain         |
| V173      | REAL  | No   | ms              | 0.01       | $32767 \cdot (4 \cdot V263) / (1000 \cdot V173)$                                         | AP03 Mst: Integral torque equalizer gain           |
| V175      | UDINT | No   | rpm             | 1          | $V175 \cdot 1342.18 \cdot (V263/50)$                                                     | AP03 Mst: Max compens. torque equalizer            |
| V176      | UDINT | No   | ms              | 1          | $(V176 \cdot 1000) / (4 \cdot V263)$                                                     | AP03 Mst: T.out A52.30 (saturated trq eq.)         |
| V177      | REAL  | No   | Arms/s          | 0.01       | $2 \cdot V177 \cdot 1.853609716 \cdot 65536 \cdot (4 \cdot V263) / (I_{peak} \cdot 100)$ | AP03: Master: Preload ramp speed                   |
| V180      | UDINT | Yes  | []              | 1          | V180                                                                                     | AP04: Torque duplication total drives nbr          |
| V181      | UDINT | Yes  | []              | 1          | V181                                                                                     | AP04: Torque duplication address                   |
| V182      | DWORD | Yes  | []              | 1          | V182                                                                                     | AP04: Torque duplication configuration             |
| V185      | UDINT | Yes  | []              | 1          | V185                                                                                     | AP05: Winding duplication total drives nbr         |
| V186      | UDINT | Yes  | []              | 1          | V186                                                                                     | AP05: Winding duplication address                  |
| V187      | DWORD | Yes  | []              | 1          | V187                                                                                     | AP05: Winding duplication configuration            |
| V190      | UDINT | No   | []              | 1          | V190                                                                                     | AP06: Max displacement meas. S1-S2 Ph.U.           |
| V191      | UDINT | No   | ms              | 1          | V191                                                                                     | AP06: Max duration measure displac. S1-S2          |
| V200      | DWORD | Yes  | []              | 1          | V200                                                                                     | AP08: Inter-drive DEMX data exchange role          |
| V201      | DWORD | Yes  | []              | 1          | V201                                                                                     | AP08: Inter-drive DEMX data exchange configuration |
| V202      | UDINT | Yes  | []              | 1          | V202                                                                                     | AP08: Inter-drive DEMX data exchange MP            |
| V217      | REAL  | No   | []              | 0.01       | $32.0 \cdot V217$                                                                        | AP12: Active damping 2 K1                          |

| Param. N° | Type   | Halt | [PU]      | Resolution | Conversion rule from Vxx in physical unit to Vxx in internal unit                                                               | Description                                                   |
|-----------|--------|------|-----------|------------|---------------------------------------------------------------------------------------------------------------------------------|---------------------------------------------------------------|
| V218      | REAL   | No   | Hz        | 0.01       | $(0.102943708 \cdot (4 \cdot V263) \cdot V218) / (V217 + 0.00000314159 \cdot (4 \cdot V263) \cdot V218)$                        | AP12: Active damping 2 filter                                 |
| V219      | UDINT  | No   | [ ]       | 1          | $32767.0 \cdot (V217 - 0.00000314159 \cdot (4 \cdot V263) \cdot V218) / (V217 + 0.00000314159 \cdot (4 \cdot V263) \cdot V218)$ | AP12: coeff.A Active damping 2 filter                         |
| V220      | DWORD  | Yes  | [ ]       | 1          | V220                                                                                                                            | Reference configuration                                       |
| V221      | DWORD  | Yes  | [ ]       | 1          | V221                                                                                                                            | Enable configurations                                         |
| V238      | DWORD  | Yes  | [ ]       |            | V238                                                                                                                            | Filter on squared counter input S1                            |
| V239      | DWORD  | Yes  | [ ]       |            | V239                                                                                                                            | Filter on squared counter input S2                            |
| V240      | REAL   | No   | s         | 0.01       | $(0.5 + (262144 \cdot 0.2048 / V240))$                                                                                          | I <sup>2</sup> t motor: first time constant                   |
| V241      | UDINT  | No   | s         | 1          | $(0.5 + (1048576 \cdot 0.2048 / V241))$                                                                                         | I <sup>2</sup> t motor: second time constant                  |
| V242      | REAL   | No   | Arms      | 0.01       | $4096 \cdot V108 / V123$                                                                                                        | Ratio nominal motor current / maximum motor current           |
| V243      | REAL   | No   | Arms      | 0.01       | $4096 \cdot V115 / V124$                                                                                                        | Ratio nominal motor current / maximum motor current h.r.      |
| V244      | REAL   | No   | Arms      | 0.01       | $4096 \cdot V108 \cdot V128 / (100 \cdot V123)$                                                                                 | Ratio locked rotor current / maximum motor current            |
| V245      | REAL   | No   | Hz        | 0.01       | $V245 \cdot 13.1072 \cdot (V263 / 50)$                                                                                          | I <sup>2</sup> t motor: threshold change of time const.       |
| V246      | REAL   | No   | Hz        | 0.01       | $V246 \cdot 13.1072 \cdot (V263 / 50)$                                                                                          | I <sup>2</sup> t motor: frequency threshold hysteresis (V245) |
| V247      | REAL   | No   | Arms      | 0.01       | $4096 \cdot V115 \cdot V129 / (100 \cdot V124)$                                                                                 | Ratio locked rotor current / maximum motor current h.r.       |
| V248      | REAL   | Yes  | s         | 0.01       | $(0.5 + (262144 \cdot 0.2048 / V248))$                                                                                          | I <sup>2</sup> t drive: first time constant                   |
| V249      | UDINT  | Yes  | s         | 1          | $(0.5 + (1048576 \cdot 0.2048 / V249))$                                                                                         | I <sup>2</sup> t drive: second time constant                  |
| V250      | DWORD  | Yes  | [ ]       | 1          | V250                                                                                                                            | Motor XML information                                         |
| V251      | UDINT  | Yes  | [ ]       | 1          | V251                                                                                                                            | Parameter compatibility                                       |
| V252      | STRING | Yes  | [ ]       |            | V252                                                                                                                            | Drive type                                                    |
| V253      | DWORD  | Yes  | [ ]       | 1          | V253                                                                                                                            | Drive size                                                    |
| V255      | STRING | Yes  | [ ]       |            | V255                                                                                                                            | Motor type                                                    |
| V256      | STRING | Yes  | [ ]       |            | V256                                                                                                                            | Motor power connection type                                   |
| V257      | UDINT  | No   | [ ]       | 1          | V257                                                                                                                            | Motor poles pairs number                                      |
| V259      | REAL   | Yes  | mm        | 0.01       | V259                                                                                                                            | Linear motor pole pitch [180 degrees el.]                     |
| V263      | UDINT  | Yes  | us        | 25         | V263                                                                                                                            | System sampling time                                          |
| V268      | UDINT  | Yes  | mDeg      | 1          | $V268 \cdot 65.536 / 360.0$                                                                                                     | Electrical phase offset Hall effect sensor                    |
| V269      | UDINT  | Yes  | mDeg      | 1          | $V269 \cdot 65.536 / 360.0$                                                                                                     | Position offset for electrical phasing                        |
| V270      | UDINT  | Yes  | Ohm       | 1          | V270                                                                                                                            | Motor thermal sensor PTC-NTC value                            |
| V271      | UDINT  | Yes  | us        | 25         | V271/V263                                                                                                                       | Current loop sampling time                                    |
| V272      | DWORD  | Yes  | [ ]       | 1          | V272                                                                                                                            | Motor thermal sensor configuration                            |
| V273      | DWORD  | Yes  | [ ]       | 1          | V273                                                                                                                            | Electrical phase offset management                            |
| V301      | UDINT  | No   | mm/min/mm | 1          | V301                                                                                                                            | Proportional position gain [0]                                |
| V305      | UDINT  | Yes  | [ ]       | 1          | V305                                                                                                                            | Measure modulo for rot. axis/spindle abs[0]                   |
| V306      | DINT   | Yes  | [ ]       | 1          | V306                                                                                                                            | S1 sensor MULTI coefficient [0]                               |
| V307      | UDINT  | Yes  | [ ]       | 1          | V307                                                                                                                            | S1 sensor DIVI coefficient [0]                                |
| V310      | DINT   | Yes  | [ ]       | 1          | V310                                                                                                                            | S1 sensor offset in CNC increment [0]                         |
| V311      | DINT   | Yes  | [ ]       | 1          | V311                                                                                                                            | S2 sensor MULTI coefficient [0]                               |
| V312      | UDINT  | Yes  | [ ]       | 1          | V312                                                                                                                            | S2 sensor DIVI coefficient [0]                                |
| V313      | DINT   | Yes  | [ ]       | 1          | V313                                                                                                                            | S2 sensor offset in CNC increment [0]                         |
| V314      | DWORD  | Yes  | [ ]       | 1          | V314                                                                                                                            | S1 sensor sinusoidal / TTL signals range                      |
| V315      | DWORD  | Yes  | [ ]       | 1          | V315                                                                                                                            | S2 sensor sinusoidal / TTL signals range                      |
| V317      | REAL   | Yes  | V         | 0.01       | $V317 \cdot 100.0$                                                                                                              | S1 sensor power supply voltage                                |

| Param. N° | Type  | Halt | [PU] | Resolution | Conversion rule from Vxx in physical unit to Vxx in internal unit             | Description                                |
|-----------|-------|------|------|------------|-------------------------------------------------------------------------------|--------------------------------------------|
| V318      | REAL  | Yes  | V    | 0.01       | V318*100.0                                                                    | S2 sensor power supply voltage             |
| V320      | DWORD | Yes  | []   |            | V320                                                                          | S1 sensor incremental traces configuration |
| V321      | DWORD | Yes  | []   |            | V321                                                                          | S2 sensor incremental traces configuration |
| V322      | DWORD | Yes  | []   |            | V322                                                                          | S1 sensor communication protocol           |
| V323      | DWORD | Yes  | []   |            | V323                                                                          | S2 sensor communication protocol           |
| V324      | DWORD | Yes  | []   | 1          | V324                                                                          | S1 sensor initialization configuration     |
| V325      | DWORD | Yes  | []   | 1          | V325                                                                          | S2 sensor initialization configuration     |
| V326      | DWORD | No   | []   | 1          | V326                                                                          | S1 sensor run-time options                 |
| V327      | DWORD | No   | []   | 1          | V327                                                                          | S2 sensor run-time options                 |
| V328      | DWORD | No   | []   | 1          | V328                                                                          | S1 Sensor alarms handling                  |
| V329      | DWORD | No   | []   | 1          | V329                                                                          | S2 Sensor alarms handling                  |
| V330      | DWORD | Yes  | []   |            | V330                                                                          | S1 sensor homing configuration             |
| V331      | DWORD | Yes  | []   |            | V331                                                                          | S2 sensor homing configuration             |
| V340      | UDINT | No   | []   | 1          | V340                                                                          | Step of speed calculation S1 sensor        |
| V341      | UDINT | No   | []   | 1          | V341                                                                          | Step of speed calculation S2 sensor        |
| V342      | REAL  | No   | Hz   | 0.01       | $(0.102943708 * (2 * V263) * V342) / (1 + 0.00000314159 * (2 * V263) * V342)$ | Speed filter by S1 sensor                  |
| V343      | REAL  | No   | Hz   | 0.01       | $(0.102943708 * (2 * V263) * V343) / (1 + 0.00000314159 * (2 * V263) * V343)$ | Speed filter by S2 sensor                  |
| V344      | UDINT | Yes  | []   | 1          | V344                                                                          | S1 sensor code ID                          |
| V345      | UDINT | Yes  | []   | 1          | V345                                                                          | S2 sensor code ID                          |
| V350      | DWORD | Yes  | []   | 1          | V350                                                                          | Position loop sensor config.               |
| V353      | REAL  | Yes  | []   | 0.01       | V353*100                                                                      | Period of position sensor/motor turn [0]   |
| V359      | UDINT | Yes  | []   | 1          | $(65536 * 3 * 4 * 1024 / V353)$                                               | k_velrot120                                |
| V360      | UDINT | Yes  | []   | 1          | V360                                                                          | Period by turn S1 sensor                   |
| V361      | UDINT | Yes  | []   | 1          | V361                                                                          | Absolute resolution by turn S1 sensor      |
| V362      | UDINT | No   | []   | 1          | $V301 * 1.31072 * 327.68 * 4 * (2 * V263) / V360$                             | kpposrot10                                 |
| V363      | UDINT | Yes  | []   | 1          | $(65536 * 3 * 1024 * 4 / V360)$                                               | k_velrot10                                 |
| V364      | UDINT | Yes  | []   | 1          | V361/V360                                                                     | sinuarot10                                 |
| V365      | UDINT | Yes  | []   | 1          | V365                                                                          | Number of absolute S1 sensor turns         |
| V368      | UDINT | Yes  | []   | 1          | V368                                                                          | Intervals between S1 coded references      |
| V369      | UDINT | Yes  | []   | 1          | V369                                                                          | Multiplier of S1 coded references          |
| V370      | UDINT | Yes  | []   | 1          | V370                                                                          | Period by turn S2 sensor                   |
| V371      | UDINT | Yes  | []   | 1          | V371                                                                          | Absolute resolution by turn S2 sensor      |
| V372      | UDINT | No   | []   | 1          | $V301 * 1.31072 * 327.68 * 4 * (2 * V263) / V353$                             | kpposrot120                                |
| V373      | UDINT | Yes  | []   | 1          | $(65536 * 3 * 1024 * 4 / V370)$                                               | k_velrot20                                 |
| V374      | UDINT | Yes  | []   | 1          | V371/V370                                                                     | sinuarot20                                 |
| V375      | UDINT | Yes  | []   | 1          | V375                                                                          | Number of absolute S2 sensor turns         |
| V376      | UDINT | No   | []   | 1          | $V301 * 1.31072 * 327.68 * 4 * (2 * V263) / V370$                             | kpposrot20                                 |
| V378      | UDINT | Yes  | []   | 1          | V378                                                                          | Intervals between S2 coded references      |
| V379      | UDINT | Yes  | []   | 1          | V379                                                                          | Multiplier of S2 coded references          |
| V380      | UDINT | Yes  | um   | 1          | V380                                                                          | Period of linear S1 sensor                 |
| V381      | UDINT | Yes  | nm   | 1          | V381                                                                          | Absolute resolution of linear S1 sensor    |
| V382      | UDINT | Yes  | []   | 1          | $(V259 * 2000 / V380) + 0.1$                                                  | npslin10                                   |
| V383      | UDINT | No   | []   | 1          | $V301 * 5.24288 * V380 * 4 * (2 * V263 / 100)$                                | kpposlin10                                 |
| V384      | UDINT | Yes  | []   | 1          | $(24576 * V380 * 4)$                                                          | k_vellin10                                 |

| Param. N° | Type  | Halt | [PU]            | Resolution | Conversion rule from Vxx in physical unit to Vxx in internal unit                         | Description                                                                 |
|-----------|-------|------|-----------------|------------|-------------------------------------------------------------------------------------------|-----------------------------------------------------------------------------|
| V385      | UDINT | Yes  | [ ]             | 1          | $V380 \cdot 1000 / V381$                                                                  | sinualin10                                                                  |
| V386      | UDINT | Yes  | [ ]             | 1          | $(V259 \cdot 2000000 / V381) + 0.1$                                                       | ralinppm10                                                                  |
| V389      | UDINT | Yes  | [ ]             | 1          | V389                                                                                      | S1 sensor serial position datum bit size                                    |
| V390      | UDINT | Yes  | um              | 1          | V390                                                                                      | Period of linear S2 sensor                                                  |
| V391      | UDINT | Yes  | nm              | 1          | V391                                                                                      | Absolute resolution of linear S2 sensor                                     |
| V392      | UDINT | Yes  | [ ]             | 1          | $(V259 \cdot 2000 / V390) + 0.1$                                                          | npslin20                                                                    |
| V393      | UDINT | No   | [ ]             | 1          | $V301 \cdot 5.24288 \cdot V390 \cdot 4 \cdot (2 \cdot V263 / 100)$                        | kpposlin20                                                                  |
| V394      | UDINT | Yes  | [ ]             | 1          | $(24576 \cdot V390 \cdot 4)$                                                              | k_vellin20                                                                  |
| V395      | UDINT | Yes  | [ ]             | 1          | $V390 \cdot 1000 / V391$                                                                  | sinualin20                                                                  |
| V396      | UDINT | Yes  | nm              | 1          | $(V259 \cdot 2000000 / V391) + 0.1$                                                       | ralinppm20                                                                  |
| V399      | UDINT | Yes  | [ ]             | 1          | V399                                                                                      | S2 sensor serial position datum bit size                                    |
| V400      | UDINT | No   | mOhm            | 1          | $V400 \cdot 3.2768 \cdot V406 \cdot (V271 / 50) / V403$                                   | Rs at 120 °C                                                                |
| V401      | UDINT | No   | mOhm            | 1          | $V401 \cdot 3.2768 \cdot V406 \cdot (V271 / 50) / V403$                                   | Rr at 150 °C                                                                |
| V402      | UDINT | No   | uH              | 1          | $(V402 + V408 \cdot 1000) \cdot 65.536 / V403$                                            | sigmaLs                                                                     |
| V403      | REAL  | No   | mH              | 0.01       | V403                                                                                      | Lr                                                                          |
| V404      | UDINT | No   | [ ]             | 1          | $133774.75 \cdot 3276.8 \cdot (V271 / 50) / ((I_{peak} \cdot 100) \cdot V403)$            | $[U.l. = (K_i / K_v) / L_r \cdot (T_s \cdot 2^{16}) = a04] \quad K_i / K_v$ |
| V405      | UDINT | No   | [ ]             | 1          | $133774.75 \cdot 3276.8 \cdot V407 \cdot (V271 / 50) / ((I_{peak} \cdot 100) \cdot V403)$ | $[U.l. = a04b \cdot ratio\_gamma = a04a] \quad a04a$                        |
| V406      | UDINT | No   | [ ]             | 1          | V406                                                                                      | twotosh0                                                                    |
| V407      | UDINT | No   | [ ]             | 1          | V407                                                                                      | ratio_gamma                                                                 |
| V408      | REAL  | No   | mH              | 0.01       | V408                                                                                      | Additional extern inductance                                                |
| V409      | UDINT | No   | Vrms            | 1          | V409*8                                                                                    | ST flux-weakening voltage                                                   |
| V410      | REAL  | No   | [ ]             | 0.01       | V410*256                                                                                  | Proportional gain flux loop                                                 |
| V411      | REAL  | No   | ms              | 0.01       | $32.767 \cdot (2 \cdot V263) / V411$                                                      | Integral gain flux loop                                                     |
| V412      | UDINT | No   | %               | 1          | $0.64 \cdot V412$                                                                         | Lr saturation level                                                         |
| V413      | UDINT | No   | [ ]             | 1          | $32768 / (0.64 \cdot V412)$                                                               | ksatinv                                                                     |
| V414      | UDINT | No   | rpm             | 1          | $V414 \cdot 0.3431457 \cdot V257 \cdot (V271 / 100)$                                      | Observer engagement speed                                                   |
| V415      | UDINT | No   | rpm             | 1          | $V415 \cdot 0.3431457 \cdot V257 \cdot (V271 / 100)$                                      | Observer engagement hysteresis                                              |
| V417      | UDINT | No   | rpm             | 1          | $V417 \cdot V257 \cdot 0.21845333 \cdot (V263 / 50)$                                      | Defluxing starting speed                                                    |
| V425      | UDINT | No   | %               | 1          | $V425 \cdot V427 \cdot V412 \cdot 131.072 \cdot 1.4142135 / (I_{peak} \cdot 100)$         | Min_flux as a %age of V427                                                  |
| V426      | REAL  | No   | Arms/<br>Vrms*s | 0.01       | $V426 \cdot 7592.3856 \cdot 100 \cdot (V263 / 50) / (I_{peak} \cdot 100)$                 | K deflux                                                                    |
| V427      | REAL  | No   | Arms            | 0.01       | $V427 \cdot V412 \cdot 13107.2 \cdot 1.4142135 / (I_{peak} \cdot 100)$                    | Magnetization current                                                       |
| V428      | REAL  | No   | Arms            | 0.01       | $V428 \cdot 1853609.716 / (I_{peak} \cdot 100)$                                           | Magnetization current limit                                                 |
| V434      | REAL  | No   | Vrms/<br>Arms   | 0.01       | $(V434 \cdot (I_{peak} \cdot 100)) / 28.28427 / 4$                                        | Proportional Id current loop gain                                           |
| V435      | REAL  | No   | ms              | 0.01       | $32767 \cdot V271 / (1000 \cdot V435)$                                                    | Integral Id current loop gain                                               |
| V448      | UDINT | No   | uH              | 1          | $(V402 + V407 \cdot V408 \cdot 1000) \cdot 65.536 / V403$                                 | sigmaLs h.r.                                                                |
| V450      | REAL  | No   | [ ]             | 0.01       | V450*256                                                                                  | Proportional flux loop gain h.r.                                            |
| V451      | REAL  | No   | ms              | 0.01       | $32.767 \cdot (2 \cdot V263) / V451$                                                      | Integral flux loop gain h.r.                                                |
| V452      | UDINT | No   | %               | 1          | $0.64 \cdot V452$                                                                         | Lr saturation level h.r.                                                    |
| V453      | UDINT | No   | [ ]             | 1          | $32768 / (0.64 \cdot V452)$                                                               | ksatinv h.r.                                                                |
| V454      | UDINT | No   | rpm             | 1          | $V454 \cdot 0.3431457 \cdot V257 \cdot (V271 / 100)$                                      | Observer engagement speed h.r.                                              |
| V455      | UDINT | No   | rpm             | 1          | $V455 \cdot 0.3431457 \cdot V257 \cdot (V271 / 100)$                                      | Observer engagement hysteresis h.r.                                         |
| V457      | UDINT | No   | rpm             | 1          | $V457 \cdot V257 \cdot 0.21845333 \cdot (V263 / 50)$                                      | Defluxing starting speed h.r.                                               |
| V465      | UDINT | No   | %               | 1          | $V465 \cdot V467 \cdot V452 \cdot 131.072 \cdot 1.4142135 / (I_{peak} \cdot 100)$         | Min_flux as a %age of V467 h.r.                                             |

| Param. N° | Type  | Halt | [PU]            | Resolution | Conversion rule from Vxx in physical unit to Vxx in internal unit                       | Description                                             |
|-----------|-------|------|-----------------|------------|-----------------------------------------------------------------------------------------|---------------------------------------------------------|
| V466      | REAL  | No   | Arms/<br>Vrms*s | 0.01       | $V466 * 7592.3856 * 100 * (V263/50) / (I_{peak} * 100)$                                 | K deflux h.r.                                           |
| V467      | REAL  | No   | Arms            | 0.01       | $V467 * V452 * 13107.2 * 1.4142135 / (I_{peak} * 100)$                                  | Magnetization current h.r.                              |
| V468      | REAL  | No   | Arms            | 0.01       | $V468 * 1853609.716 / (I_{peak} * 100)$                                                 | Magnetization current limit h.r.                        |
| V470      | REAL  | No   | mH              | 0.01       | $((V402/1000.0) + V408) * (I_{peak} * 100) * 0.01503011016 * (50/V263)$                 | SigmaLs flux-weakening                                  |
| V471      | REAL  | No   | mH              | 0.01       | $((V402/(V407 * 1000.0)) + V408) * (I_{peak} * 100) * 0.01503011016 * (50/V263)$        | SigmaLs flux-weakening h.r.                             |
| V472      | UDINT | No   | [ ]             | 1          | $(V400 + V401) * (I_{peak} * 100) * 65.536 / 133772.76$                                 | Total resistance power lost Ri2                         |
| V473      | UDINT | No   | [ ]             | 1          | $(V400 + V401) * (I_{peak} * 100) * 65.536 / (V407 * 133772.76)$                        | Total resistance power lost Ri2 h.r.                    |
| V474      | REAL  | No   | Vrms/<br>Arms   | 0.01       | $(V474 * (I_{peak} * 100) / 28.28427) / 4$                                              | Proportional Id current gain h.r.                       |
| V475      | REAL  | No   | ms              | 0.01       | $32767 * V271 / (1000 * V475)$                                                          | Integral Id current gain h.r.                           |
| V479      | REAL  | No   | Hz              | 0.01       | $(0.102943708 * (4 * V263) * V479) / (1.0 + 0.00000314159 * (4 * V263) * V479)$         | Low-pass filtering of Delivered Power MP6 and LOAD MP11 |
| V480      | REAL  | No   | %               | 0.01       | $V480 * 655.35$                                                                         | Correction factor for MP6                               |
| V481      | DWORD | Yes  | [ ]             | 1          | V481                                                                                    | Control mode selection                                  |
| V488      | UDINT | No   | mOhm            | 1          | $V488 * (I_{peak} * 100) * 65.536 * (234.5 + 120.0) / ((234.5 + 20.0) * 2 * 133772.76)$ | Resistance tt at 20 °C                                  |
| V489      | REAL  | No   | mH              | 0.01       | $(V408 + V489) * (I_{peak} * 100) * 0.01503011016 * (50/V263)$                          | Motor Lq inductance                                     |
| V491      | REAL  | No   | Arms            | 0.01       | $V491 * 1853609.716 / (I_{peak} * 100)$                                                 | Id_Max (defluxing parameter)                            |
| V492      | REAL  | No   | Arms            | 0.01       | $V492 * 1853609.716 / (I_{peak} * 100)$                                                 | Id current without defluxing                            |
| V496      | UDINT | No   | rpm             | 1          | $V496 * V257 * 0.21845333 * (V263/50)$                                                  | Defluxing starting speed                                |
| V497      | REAL  | No   | m/min           | 0.01       | $V497 * 109.22667 * V257 * (V263/50) / V259$                                            | Defluxing starting speed                                |
| V498      | REAL  | No   | Arms/<br>Vrms*s | 0.01       | $V498 * 7592.3856 * 100 * (V263/50) / (I_{peak} * 100)$                                 | K Iq reduction                                          |
| V499      | REAL  | No   | Arms/<br>Vrms*s | 0.01       | $V499 * 7592.3856 * 100 * (V263/50) / (I_{peak} * 100)$                                 | K flux-weakening Id                                     |
| V500      | DWORD | No   | [ ]             | 1          | V500                                                                                    | Filter configuration                                    |
| V501      | DINT  | No   | [ ]             | 1          | V501                                                                                    | ph11 (FILTER 1 for torque current ref.)                 |
| V502      | UDINT | No   | [ ]             | 1          | V502                                                                                    | ph12 (FILTER 1 for torque current ref.)                 |
| V503      | UDINT | No   | [ ]             | 1          | V503                                                                                    | ph21 (FILTER 1 for torque current ref.)                 |
| V504      | DINT  | No   | [ ]             | 1          | V504                                                                                    | ph22 (FILTER 1 for torque current ref.)                 |
| V505      | DINT  | No   | [ ]             | 1          | V505                                                                                    | ga1 (FILTER 1 for torque current ref.)                  |
| V506      | DINT  | No   | [ ]             | 1          | V506                                                                                    | ga2 (FILTER 1 for torque current ref.)                  |
| V507      | UDINT | No   | [ ]             | 1          | V507                                                                                    | r*r (FILTER 1 for torque current ref.)                  |
| V508      | DINT  | No   | [ ]             | 1          | V508                                                                                    | C1 (FILTER 1 for torque current ref.)                   |
| V509      | DINT  | No   | [ ]             | 1          | V509                                                                                    | C2 (FILTER 1 for torque current ref.)                   |
| V511      | DINT  | No   | [ ]             | 1          | V511                                                                                    | ph11 (FILTER 2 for torque current ref.)                 |
| V512      | UDINT | No   | [ ]             | 1          | V512                                                                                    | ph12 (FILTER 2 for torque current ref.)                 |
| V513      | UDINT | No   | [ ]             | 1          | V513                                                                                    | ph21 (FILTER 2 for torque current ref.)                 |
| V514      | DINT  | No   | [ ]             | 1          | V514                                                                                    | ph22 (FILTER 2 for torque current ref.)                 |
| V515      | DINT  | No   | [ ]             | 1          | V515                                                                                    | ga1 (FILTER 2 for torque current ref.)                  |
| V516      | DINT  | No   | [ ]             | 1          | V516                                                                                    | ga2 (FILTER 2 for torque current ref.)                  |
| V517      | UDINT | No   | [ ]             | 1          | V517                                                                                    | r*r (FILTER 2 for torque current ref.)                  |
| V518      | DINT  | No   | [ ]             | 1          | V518                                                                                    | C1 (FILTER 2 for torque current ref.)                   |

| Param. N° | Type  | Halt | [PU] | Resolution | Conversion rule from Vxx in physical unit to Vxx in internal unit     | Description                              |
|-----------|-------|------|------|------------|-----------------------------------------------------------------------|------------------------------------------|
| V519      | DINT  | No   | [ ]  | 1          | V519                                                                  | C2 (FILTER 2 for torque current ref.)    |
| V521      | DINT  | No   | [ ]  | 1          | V521                                                                  | ph11 (FILTER 3 for torque current ref.)  |
| V522      | UDINT | No   | [ ]  | 1          | V522                                                                  | ph12 (FILTER 3 for torque current ref.)  |
| V523      | UDINT | No   | [ ]  | 1          | V523                                                                  | ph21 (FILTER 3 for torque current ref.)  |
| V524      | DINT  | No   | [ ]  | 1          | V524                                                                  | ph22 (FILTER 3 for torque current ref.)  |
| V525      | DINT  | No   | [ ]  | 1          | V525                                                                  | ga1 (FILTER 3 for torque current ref.)   |
| V526      | DINT  | No   | [ ]  | 1          | V526                                                                  | ga2 (FILTER 3 for torque current ref.)   |
| V527      | UDINT | No   | [ ]  | 1          | V527                                                                  | r*r (FILTER 3 for torque current ref.)   |
| V528      | DINT  | No   | [ ]  | 1          | V528                                                                  | C1 (FILTER 3 for torque current ref.)    |
| V529      | DINT  | No   | [ ]  | 1          | V529                                                                  | C2 (FILTER 3 for torque current ref.)    |
| V531      | DINT  | No   | [ ]  | 1          | V531                                                                  | ph11 (FILTER 4 for torque current ref.)  |
| V532      | UDINT | No   | [ ]  | 1          | V532                                                                  | ph12 (FILTER 4 for torque current ref.)  |
| V533      | UDINT | No   | [ ]  | 1          | V533                                                                  | ph21 (FILTER 4 for torque current ref.)  |
| V534      | DINT  | No   | [ ]  | 1          | V534                                                                  | ph22 (FILTER 4 for torque current ref.)  |
| V535      | DINT  | No   | [ ]  | 1          | V535                                                                  | ga1 (FILTER 4 for torque current ref.)   |
| V536      | DINT  | No   | [ ]  | 1          | V536                                                                  | ga2 (FILTER 4 for torque current ref.)   |
| V537      | UDINT | No   | [ ]  | 1          | V537                                                                  | r*r (FILTER 4 for torque current ref.)   |
| V538      | DINT  | No   | [ ]  | 1          | V538                                                                  | C1 (FILTER 4 for torque current ref.)    |
| V539      | DINT  | No   | [ ]  | 1          | V539                                                                  | C2 (FILTER 4 for torque current ref.)    |
| V541      | DINT  | No   | [ ]  | 1          | V541                                                                  | ph11 (FILTER 1 for speed ref.)           |
| V542      | UDINT | No   | [ ]  | 1          | V542                                                                  | ph12 (FILTER 1 for speed ref.)           |
| V543      | UDINT | No   | [ ]  | 1          | V543                                                                  | ph21 (FILTER 1 for speed ref.)           |
| V544      | DINT  | No   | [ ]  | 1          | V544                                                                  | ph22 (FILTER 1 for speed ref.)           |
| V545      | DINT  | No   | [ ]  | 1          | V545                                                                  | ga1 (FILTER 1 for speed ref.)            |
| V546      | DINT  | No   | [ ]  | 1          | V546                                                                  | ga2 (FILTER 1 for speed ref.)            |
| V547      | UDINT | No   | [ ]  | 1          | V547                                                                  | r*r (FILTER 1 for speed ref.)            |
| V548      | DINT  | No   | [ ]  | 1          | V548                                                                  | C1(FILTER 1 for speed ref.)              |
| V549      | DINT  | No   | [ ]  | 1          | V549                                                                  | C2(FILTER 1 for speed ref.)              |
| V600      | UDINT | No   | %    | 1          | $V600 \cdot V108 \cdot 18536.09716 / (I_{peak} \cdot 100)$            | V/f: Maximal motor current (% of V108)   |
| V601      | UDINT | No   | Hz   | 1          | $V601 \cdot 13.1072 \cdot (V263/50)$                                  | V/f: Rated frequency                     |
| V602      | REAL  | No   | Vrms | 0.01       | $V602 \cdot 8$                                                        | V/f: Rated voltage                       |
| V603      | REAL  | No   | V    | 0.01       | $V603 \cdot 8$                                                        | V/f: V0                                  |
| V604      | UDINT | No   | mOhm | 1          | $V604 \cdot (I_{peak} \cdot 100) \cdot 65.536 \cdot 1.4 / 133772.76$  | V/f: Stator resistance @ 20 degrees C    |
| V605      | UDINT | No   | %    | 1          | $V605 \cdot V108 \cdot 18536.09716 / (I_{peak} \cdot 100)$            | V/f: Intermittent overload (% of V108)   |
| V606      | UDINT | No   | Hz   | 1          | $V606 \cdot 13.1072 \cdot (V263/50)$                                  | V/f: Freq. start of overload reduction   |
| V607      | REAL  | No   | mH   | 0.01       | $V607 \cdot (I_{peak} \cdot 100) \cdot 0.01503011016 \cdot (50/V263)$ | V/f: Q-Q stabilizing inductance          |
| V608      | REAL  | No   | mH   | 0.01       | $V608 \cdot (I_{peak} \cdot 100) \cdot 0.01503011016 \cdot (50/V263)$ | V/f: D-Q stabilizing inductance          |
| V620      | UDINT | No   | %    | 1          | $V620 \cdot V108 \cdot 18536.09716 / (I_{peak} \cdot 100)$            | Target Brake-Test current (% V108) l.r.  |
| V621      | UDINT | No   | %    | 1          | $V621 \cdot V115 \cdot 18536.09716 / (I_{peak} \cdot 100)$            | Target Brake-Test current (% V115) h.r.  |
| V622      | UDINT | No   | ms   | 1          | V622                                                                  | Brake-Test core phase duration           |
| V623      | UDINT | No   | [ ]  | 1          | V623                                                                  | Brake-Test position window               |
| V680      | DWORD | No   | [ ]  | 1          | V680                                                                  | Oscilloscope control                     |
| V681      | DWORD | Yes  | [ ]  |            | V681                                                                  | Oscilloscope configuration               |
| V682      | UDINT | Yes  | [ ]  | 1          | V682                                                                  | Associate MP with oscilloscope channel 1 |

| Param. N° | Type  | Halt | [PU] | Resolution | Conversion rule from Vxx in physical unit to Vxx in internal unit | Description                                |
|-----------|-------|------|------|------------|-------------------------------------------------------------------|--------------------------------------------|
| V683      | UDINT | Yes  | []   | 1          | V683                                                              | Associate MP with oscilloscope channel 2   |
| V684      | UDINT | Yes  | []   | 1          | V684                                                              | Associate MP with oscilloscope trigger     |
| V685      | UDINT | Yes  | us   | 50         | $V685/(2 \times V263)$                                            | Oscilloscope sample time                   |
| V686      | UDINT | Yes  | []   | 1          | V686                                                              | Samples number to be recorded              |
| V687      | UDINT | Yes  | []   | 1          | V687                                                              | Samples number to be recorded post trigger |
| V688      | DINT  | Yes  | []   | 1          | V688                                                              | Threshold level (I.U.) for MP value        |
| V700      | DWORD | Yes  | []   |            | V700                                                              | Motor part number 1° characters group      |
| V701      | DWORD | Yes  | []   |            | V701                                                              | Motor part number 2° characters group      |
| V702      | DWORD | Yes  | []   |            | V702                                                              | Motor part number 3° characters group      |
| V703      | DWORD | Yes  | []   |            | V703                                                              | Motor part number 4° characters group      |
| V704      | DWORD | Yes  | []   |            | V704                                                              | Motor part number 5° characters group      |
| V705      | DWORD | Yes  | []   |            | V705                                                              | Motor part number 6° characters group      |
| V706      | DWORD | Yes  | []   |            | V706                                                              | Motor part number 7° characters group      |
| V707      | DWORD | Yes  | []   |            | V707                                                              | Motor part number 8° characters group      |
| V708      | DWORD | Yes  | []   |            | V708                                                              | Motor part number 9° characters group      |
| V709      | DWORD | Yes  | []   |            | V709                                                              | Motor part number 10° characters group     |
| V712      | DWORD | Yes  | []   |            | V712                                                              | S1 sensor part number 1° characters group  |
| V713      | DWORD | Yes  | []   |            | V713                                                              | S1 sensor part number 2° characters group  |
| V714      | DWORD | Yes  | []   |            | V714                                                              | S1 sensor part number 3° characters group  |
| V715      | DWORD | Yes  | []   |            | V715                                                              | S1 sensor part number 4° characters group  |
| V716      | DWORD | Yes  | []   |            | V716                                                              | S1 sensor part number 5° characters group  |
| V717      | DWORD | Yes  | []   |            | V717                                                              | S1 sensor part number 6° characters group  |
| V718      | DWORD | Yes  | []   |            | V718                                                              | S1 sensor part number 7° characters group  |
| V719      | DWORD | Yes  | []   |            | V719                                                              | S1 sensor part number 8° characters group  |
| V720      | DWORD | Yes  | []   |            | V720                                                              | S1 sensor part number 9° characters group  |
| V721      | DWORD | Yes  | []   |            | V721                                                              | S1 sensor part number 10° characters group |
| V724      | DWORD | Yes  | []   |            | V724                                                              | S2 sensor part number 1° characters group  |
| V725      | DWORD | Yes  | []   |            | V725                                                              | S2 sensor part number 2° characters group  |
| V726      | DWORD | Yes  | []   |            | V726                                                              | S2 sensor part number 3° characters group  |
| V727      | DWORD | Yes  | []   |            | V727                                                              | S2 sensor part number 4° characters group  |
| V728      | DWORD | Yes  | []   |            | V728                                                              | S2 sensor part number 5° characters group  |
| V729      | DWORD | Yes  | []   |            | V729                                                              | S2 sensor part number 6° characters group  |
| V730      | DWORD | Yes  | []   |            | V730                                                              | S2 sensor part number 7° characters group  |
| V731      | DWORD | Yes  | []   |            | V731                                                              | S2 sensor part number 8° characters group  |
| V732      | DWORD | Yes  | []   |            | V732                                                              | S2 sensor part number 9° characters group  |
| V733      | DWORD | Yes  | []   |            | V733                                                              | S2 sensor part number 10° characters group |
| V900      | DINT  | Yes  | []   | 1          | V900                                                              | AP07: DEMX instruction #0                  |
| V901      | DINT  | Yes  | []   | 1          | V901                                                              | AP07: DEMX instruction #1                  |
| V902      | DINT  | Yes  | []   | 1          | V902                                                              | AP07: DEMX instruction #2                  |
| V903      | DINT  | Yes  | []   | 1          | V903                                                              | AP07: DEMX instruction #3                  |
| V904      | DINT  | Yes  | []   | 1          | V904                                                              | AP07: DEMX instruction #4                  |
| V905      | DINT  | Yes  | []   | 1          | V905                                                              | AP07: DEMX instruction #5                  |
| V906      | DINT  | Yes  | []   | 1          | V906                                                              | AP07: DEMX instruction #6                  |
| V907      | DINT  | Yes  | []   | 1          | V907                                                              | AP07: DEMX instruction #7                  |
| V908      | DINT  | Yes  | []   | 1          | V908                                                              | AP07: DEMX instruction #8                  |
| V909      | DINT  | Yes  | []   | 1          | V909                                                              | AP07: DEMX instruction #9                  |

| Param. N° | Type  | Halt | [PU]       | Resolution | Conversion rule from Vxx in physical unit to Vxx in internal unit | Description                                 |
|-----------|-------|------|------------|------------|-------------------------------------------------------------------|---------------------------------------------|
| V910      | DINT  | Yes  | [ ]        | 1          | V910                                                              | AP07: DEMX instruction #10                  |
| V911      | DINT  | Yes  | [ ]        | 1          | V911                                                              | AP07: DEMX instruction #11                  |
| V912      | DINT  | Yes  | [ ]        | 1          | V912                                                              | AP07: DEMX instruction #12                  |
| V913      | DINT  | Yes  | [ ]        | 1          | V913                                                              | AP07: DEMX instruction #13                  |
| V914      | DINT  | Yes  | [ ]        | 1          | V914                                                              | AP07: DEMX instruction #14                  |
| V915      | DINT  | Yes  | [ ]        | 1          | V915                                                              | AP07: DEMX instruction #15                  |
| V920      | DINT  | No   | [ ]        | 1          | V920                                                              | AP07: DEMX parameter #0                     |
| V921      | DINT  | No   | [ ]        | 1          | V921                                                              | AP07: DEMX parameter #1                     |
| V922      | DINT  | No   | [ ]        | 1          | V922                                                              | AP07: DEMX parameter #2                     |
| V923      | DINT  | No   | [ ]        | 1          | V923                                                              | AP07: DEMX parameter #3                     |
| V924      | DINT  | No   | [ ]        | 1          | V924                                                              | AP07: DEMX parameter #4                     |
| V925      | DINT  | No   | [ ]        | 1          | V925                                                              | AP07: DEMX parameter #5                     |
| V926      | DINT  | No   | [ ]        | 1          | V926                                                              | AP07: DEMX parameter #6                     |
| V927      | DINT  | No   | [ ]        | 1          | V927                                                              | AP07: DEMX parameter #7                     |
| V928      | DINT  | No   | [ ]        | 1          | V928                                                              | AP07: DEMX parameter #8                     |
| V929      | DINT  | No   | [ ]        | 1          | V929                                                              | AP07: DEMX parameter #9                     |
| V930      | DINT  | No   | [ ]        | 1          | V930                                                              | AP07: DEMX parameter #10                    |
| V931      | DINT  | No   | [ ]        | 1          | V931                                                              | AP07: DEMX parameter #11                    |
| V932      | DINT  | No   | [ ]        | 1          | V932                                                              | AP07: DEMX parameter #12                    |
| V933      | DINT  | No   | [ ]        | 1          | V933                                                              | AP07: DEMX parameter #13                    |
| V934      | DINT  | No   | [ ]        | 1          | V934                                                              | AP07: DEMX parameter #14                    |
| V935      | DINT  | No   | [ ]        | 1          | V935                                                              | AP07: DEMX parameter #15                    |
| V1045     | REAL  | No   | mm/min     | 0.01       | $V1045 * 163.840 * (V263/50)$                                     | Speed dynamic check [1]                     |
| V1046     | REAL  | No   | rpm        | 0.01       | $V1046 * 1342.18 * (V263/50)$                                     | Speed dynamic check [1]                     |
| V1050     | REAL  | No   | Arms       | 0.01       | $V1050 * 1853609.716 / (I_{peak} * 100)$                          | Torque current limit [1]                    |
| V1051     | REAL  | No   | mArms/rpm  | 0.01       | $V1051 * 90509.668 * (50/V263) / (I_{peak} * 100)$                | Proportional speed loop gain [1]            |
| V1052     | REAL  | No   | ms         | 0.01       | $32767 * (2 * V263) / (V1052 * 1000)$                             | Integral speed loop gain [1]                |
| V1057     | REAL  | No   | Arms       | 0.01       | $V1057 * 1853609.716 / (I_{peak} * 100)$                          | Torque current limit h.r. [1]               |
| V1058     | REAL  | No   | mArms/rpm  | 0.01       | $V1058 * 90509.668 * (50/V263) / (I_{peak} * 100)$                | Proportional speed loop gain h.r. [1]       |
| V1059     | REAL  | No   | ms         | 0.01       | $32767 * (2 * V263) / (V1059 * 1000)$                             | Integral speed loop gain h.r. [1]           |
| V1060     | REAL  | No   | Arms/m/min | 0.01       | $V1060 * 741455.2 * (50/V263) / (I_{peak} * 100)$                 | Proportional speed loop gain [1]            |
| V1091     | UDINT | No   | rpm/s      | 1          | $V1091 * (4 * V263) * 1.34218 * (V263/50) / 1000$                 | Acceleration/deceleration ramp [1]          |
| V1092     | UDINT | No   | rpm/s      | 1          | $V1092 * (4 * V263) * 1.34218 * (V263/50) / 1000$                 | Acceleration/deceleration ramp h.r.[1]      |
| V1093     | UDINT | No   | mm/s^2     | 1          | $V1093 * 60 * (4 * V263) * 0.163840 * (V263/50) / 1000$           | Acceleration/deceleration ramp [1]          |
| V1094     | UDINT | No   | %          | 1          | $V1094 * 163840 * V1095 / 100$                                    | Speed reference Limit (% V1095) [1]         |
| V1095     | UDINT | No   | m/min      | 1          | $V1095 * 163840 * (V263/50)$                                      | Maximum motor speed [1]                     |
| V1096     | UDINT | No   | m/min      | 1          | $V1096 * 163840 * (V263/50)$                                      | Overspeed alarm threshold [1]               |
| V1097     | UDINT | No   | %          | 1          | $V1097 * 1342.18 * V1098 / 100$                                   | Speed reference Limit (% V1098) [1]         |
| V1098     | UDINT | No   | rpm        | 1          | $V1098 * 1342.18 * (V263/50)$                                     | Maximum motor speed [1]                     |
| V1099     | UDINT | No   | rpm        | 1          | $V1099 * 1342.18 * (V263/50)$                                     | Overspeed alarm threshold [1]               |
| V1301     | UDINT | No   | mm/min/mm  | 1          | V1301                                                             | Proportional position gain [1]              |
| V1305     | UDINT | Yes  | [ ]        | 1          | V1305                                                             | Measure modulo for rot. axis/spindle abs[1] |

| Param. N° | Type  | Halt | [PU] | Resolution | Conversion rule from Vxx in physical unit to Vxx in internal unit       | Description                              |
|-----------|-------|------|------|------------|-------------------------------------------------------------------------|------------------------------------------|
| V1306     | DINT  | Yes  | [ ]  | 1          | V1306                                                                   | S1 sensor MULTI coefficient [1]          |
| V1307     | UDINT | Yes  | [ ]  | 1          | V1307                                                                   | S1 sensor DIVI coefficient [1]           |
| V1310     | DINT  | Yes  | [ ]  | 1          | V1310                                                                   | S1 sensor offset in CNC increment [1]    |
| V1311     | DINT  | Yes  | [ ]  | 1          | V1311                                                                   | S2 sensor MULTI coefficient [1]          |
| V1312     | UDINT | Yes  | [ ]  | 1          | V1312                                                                   | S2 sensor DIVI coefficient [1]           |
| V1313     | DINT  | Yes  | [ ]  | 1          | V1313                                                                   | S2 sensor offset in CNC increment [1]    |
| V1353     | REAL  | Yes  | [ ]  | 0.01       | V1353*100                                                               | Period of position sensor/motor turn [1] |
| V1359     | UDINT | No   | [ ]  | 1          | $(65536 \cdot 3 \cdot 4 \cdot 1024 / V1353)$                            | k_velrot121                              |
| V1362     | UDINT | No   | [ ]  | 1          | $V1301 \cdot 1.31072 \cdot 327.68 \cdot 4 \cdot (2 \cdot V263) / V360$  | kpposrot11                               |
| V1372     | UDINT | No   | [ ]  | 1          | $V1301 \cdot 1.31072 \cdot 327.68 \cdot 4 \cdot (2 \cdot V263) / V1353$ | kpposrot121                              |
| V1376     | UDINT | No   | [ ]  | 1          | $V1301 \cdot 1.31072 \cdot 327.68 \cdot 4 \cdot (2 \cdot V263) / V370$  | kpposrot21                               |
| V1383     | UDINT | No   | [ ]  | 1          | $V1301 \cdot 5.24288 \cdot V380 \cdot 4 \cdot (2 \cdot V263 / 100)$     | kpposlin11                               |
| V1393     | UDINT | No   | [ ]  | 1          | $V1301 \cdot 5.24288 \cdot V390 \cdot 4 \cdot (2 \cdot V263 / 100)$     | kpposlin21                               |

### A.3.2 Dependencies table

This table indicates the parameters for which a change on another has an effect.

| Param. N° | Parameters impacted by this change                                                                                                                                                                                                                                                                                                                                                                                                                                                                                  |
|-----------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| V95       | V94                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                 |
| V98       | V97                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                 |
| V100      | V125, V126                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                          |
| V105      | V114                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                |
| V106      | V118                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                |
| V108      | V109, V123, V125, V126, V128, V171, V242, V244, V600, V605, V620                                                                                                                                                                                                                                                                                                                                                                                                                                                    |
| V115      | V124, V129, V243, V247, V621                                                                                                                                                                                                                                                                                                                                                                                                                                                                                        |
| V116      | V117                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                |
| V123      | V121, V242, V244                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                    |
| V124      | V122, V243, V247                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                    |
| V128      | V244                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                |
| V129      | V247                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                |
| V217      | V218, V219                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                          |
| V218      | V219                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                |
| V245      | V246                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                |
| V257      | V98, V99, V104, V120, V414, V415, V417, V454, V455, V457, V496, V497, V1098, V1099                                                                                                                                                                                                                                                                                                                                                                                                                                  |
| V259      | V120, V382, V386, V392, V396, V497                                                                                                                                                                                                                                                                                                                                                                                                                                                                                  |
| V263      | V41, V42, V45, V46, V51, V52, V58, V59, V60, V91, V92, V93, V95, V96, V98, V99, V104, V105, V106, V114, V116, V117, V118, V120, V144, V154, V165, V166, V172, V173, V175, V176, V177, V218, V219, V245, V246, V271, V342, V343, V362, V372, V376, V383, V393, V411, V417, V426, V451, V457, V466, V470, V471, V479, V489, V496, V497, V498, V499, V601, V606, V607, V608, V685, V1045, V1046, V1051, V1052, V1058, V1059, V1060, V1091, V1092, V1093, V1095, V1096, V1098, V1099, V1362, V1372, V1376, V1383, V1393 |
| V271      | V98, V99, V103, V107, V112, V119, V400, V401, V403, V404, V405, V414, V415, V435, V454, V455, V475, V1098, V1099                                                                                                                                                                                                                                                                                                                                                                                                    |
| V301      | V362, V372, V376, V383, V393                                                                                                                                                                                                                                                                                                                                                                                                                                                                                        |
| V353      | V359, V372                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                          |
| V360      | V362, V363, V364, V1362                                                                                                                                                                                                                                                                                                                                                                                                                                                                                             |
| V361      | V364                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                |
| V370      | V373, V374, V376, V1376                                                                                                                                                                                                                                                                                                                                                                                                                                                                                             |
| V371      | V374                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                |
| V380      | V382, V383, V384, V385, V1383                                                                                                                                                                                                                                                                                                                                                                                                                                                                                       |
| V381      | V385, V386                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                          |
| V390      | V392, V393, V394, V395, V1393                                                                                                                                                                                                                                                                                                                                                                                                                                                                                       |
| V391      | V395, V396                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                          |
| V400      | V472, V473                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                          |
| V401      | V472, V473                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                          |
| V402      | V448, V470, V471                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                    |
| V403      | V400, V401, V402, V404, V405, V448                                                                                                                                                                                                                                                                                                                                                                                                                                                                                  |
| V406      | V400, V401                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                          |
| V407      | V114, V117, V118, V402, V405, V448, V471, V473                                                                                                                                                                                                                                                                                                                                                                                                                                                                      |
| V408      | V105, V106, V114, V118, V402, V448, V470, V471, V489                                                                                                                                                                                                                                                                                                                                                                                                                                                                |
| V412      | V413, V425, V427                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                    |
| V427      | V425                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                |
| V452      | V453, V465, V467                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                    |
| V467      | V465                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                |

| Param.<br>N° | Parameters impacted by this change |
|--------------|------------------------------------|
| V491         | V492                               |
| V600         | V605                               |
| V601         | V606                               |
| V686         | V687                               |
| V1095        | V1094                              |
| V1098        | V1097                              |
| V1301        | V1362, V1372, V1376, V1383, V1393  |
| V1353        | V1359, V1372                       |

1

2

3

4

5

6

A

### A.3.3 NUMDrive X parameters format and limits

| Param. N° | Type  | Resolution | Min value [PU]                     | Max value [PU]                                                                             |
|-----------|-------|------------|------------------------------------|--------------------------------------------------------------------------------------------|
| V25       | DWORD |            | 1                                  | 80000000h                                                                                  |
| V26       | DWORD |            | 1                                  | 80000000h                                                                                  |
| V27       | DWORD |            | 1                                  | 80000000h                                                                                  |
| V28       | DWORD |            | 1                                  | 80000000h                                                                                  |
| V29       | DWORD |            | 1                                  | 80000000h                                                                                  |
| V30       | DWORD | 1          | 0                                  | FFFFFFFFh                                                                                  |
| V32       | DWORD | 1          | 0                                  | 7FFFFFFh                                                                                   |
| V33       | DWORD | 1          | 0                                  | 3FFFFFFh                                                                                   |
| V34       | DWORD | 1          | 0                                  | 1FFFFFFh                                                                                   |
| V35       | DWORD | 1          | 0                                  | 7FFFh                                                                                      |
| V36       | UDINT | 1          | 1                                  | 1023                                                                                       |
| V37       | REAL  | 0.01       | 0.2                                | 2.0                                                                                        |
| V39       | REAL  | 0.01       | 0.1                                | 200.0                                                                                      |
| V40       | REAL  | 0.01       | 0.0                                | 50.0                                                                                       |
| V41       | REAL  | 0.01       | 1.0                                | 3000.0                                                                                     |
| V42       | REAL  | 0.01       | 0.1                                | 5000.0                                                                                     |
| V43       | REAL  | 0.01       | 0.1                                | 120.0                                                                                      |
| V45       | REAL  | 0.01       | 0.0                                | 20000.0                                                                                    |
| V46       | REAL  | 0.01       | 0.0                                | 2000.0                                                                                     |
| V50       | REAL  | 0.01       | 0.0                                | $(I_{peak} \cdot 100) / 141.42$                                                            |
| V51       | REAL  | 0.01       | 0.0                                | $16777200 \cdot (I_{peak} \cdot 100) / (90509.668 \cdot (50/V_{263}))$                     |
| V52       | REAL  | 0.01       | $2 \cdot (2 \cdot V_{263}) / 1000$ | $32767 \cdot (2 \cdot V_{263}) / (0.99 \cdot 1000)$                                        |
| V57       | REAL  | 0.01       | 0.0                                | $(I_{peak} \cdot 100) / 141.42$                                                            |
| V58       | REAL  | 0.01       | 0.0                                | $16777200 \cdot (I_{peak} \cdot 100) / (90509.668 \cdot (50/V_{263}))$                     |
| V59       | REAL  | 0.01       | $2 \cdot (2 \cdot V_{263}) / 1000$ | $32767 \cdot (2 \cdot V_{263}) / (0.99 \cdot 1000)$                                        |
| V60       | REAL  | 0.01       | 0.0                                | $16777200 \cdot (I_{peak} \cdot 100) / (741455.2 \cdot (50/V_{263}))$                      |
| V80       | REAL  | 0.01       | 0.0                                | 1.0                                                                                        |
| V81       | REAL  | 0.01       | 1.0                                | 2.0                                                                                        |
| V91       | UDINT | 1          | 4                                  | 1000000                                                                                    |
| V92       | UDINT | 1          | 4                                  | 1000000                                                                                    |
| V93       | UDINT | 1          | 5                                  | 100000                                                                                     |
| V94       | UDINT | 1          | 0                                  | 200                                                                                        |
| V95       | UDINT | 1          | 0                                  | 1000                                                                                       |
| V96       | UDINT | 1          | 0                                  | 1000                                                                                       |
| V97       | UDINT | 1          | 0                                  | 200                                                                                        |
| V98       | UDINT | 1          | 0                                  | $140000 \cdot 50 \cdot (1 + 0.25 \cdot ((V_{271}/V_{263}) - 1)) / (V_{257} \cdot V_{271})$ |
| V99       | UDINT | 1          | 0                                  | $140000 \cdot 50 \cdot (1 + 0.25 \cdot ((V_{271}/V_{263}) - 1)) / (V_{257} \cdot V_{271})$ |
| V100      | REAL  | 0.01       | 0.0                                | $(I_{peak} \cdot 100) / 141.42$                                                            |
| V101      | UDINT | 1          | 0                                  | 1000                                                                                       |
| V102      | REAL  | 0.01       | 0.0                                | $926790 \cdot 4 / (I_{peak} \cdot 100)$                                                    |
| V103      | REAL  | 0.01       | $2 \cdot V_{271} / 1000$           | $32767 \cdot V_{271} / (1000 \cdot 0.99)$                                                  |
| V104      | REAL  | 0.01       | 0.0                                | $32760 \cdot V_{257} / (16.237976 \cdot 2 \cdot (50/V_{263}))$                             |
| V105      | REAL  | 0.01       | 0.0                                | $(32760 / ((I_{peak} \cdot 100) \cdot 0.01503011016 \cdot (50/V_{263}))) - V_{408}$        |
| V106      | REAL  | 0.01       | 0.0                                | $(32760 / ((I_{peak} \cdot 100) \cdot 0.01503011016 \cdot (50/V_{263}))) - V_{408}$        |
| V107      | UDINT | 1          | 1                                  | 3000                                                                                       |
| V108      | REAL  | 0.01       | $(I_{peak} \cdot 100) / 4715.0$    | $(I_{peak} \cdot 100) \cdot 0.9 / 141.42$                                                  |
| V109      | REAL  | 0.01       | $-(V_{108})$                       | V108                                                                                       |

| Param. N° | Type  | Resolution | Min value [PU]                  | Max value [PU]                                                               |
|-----------|-------|------------|---------------------------------|------------------------------------------------------------------------------|
| V110      | REAL  | 0.01       | 0.0                             | $(I_{peak} \cdot 100) / 141.42$                                              |
| V111      | REAL  | 0.01       | 0.0                             | $926790 \cdot 4 / (I_{peak} \cdot 100)$                                      |
| V112      | REAL  | 0.01       | $2 \cdot V271 / 1000$           | $32767 \cdot V271 / (1000 \cdot 0.99)$                                       |
| V114      | REAL  | 0.01       | 0.0                             | 32767.0                                                                      |
| V115      | REAL  | 0.01       | $(I_{peak} \cdot 100) / 4715.0$ | $(I_{peak} \cdot 100) \cdot 0.9 / 141.42$                                    |
| V116      | REAL  | 0.01       | 0.0                             | $32760 / ((I_{peak} \cdot 100) \cdot 0.01503011016 \cdot (50 / V263))$       |
| V117      | REAL  | 0.01       | 0.0                             | 32767.0                                                                      |
| V118      | REAL  | 0.01       | 0.0                             | 32767.0                                                                      |
| V119      | REAL  | 0.01       | 0.0                             | 3.00                                                                         |
| V120      | REAL  | 0.01       | 0.0                             | $32760 \cdot V263 \cdot V257 / (V259 \cdot 54.12659)$                        |
| V121      | REAL  | 0.01       | 0.4                             | 1.0                                                                          |
| V122      | REAL  | 0.01       | 0.4                             | 1.0                                                                          |
| V123      | REAL  | 0.01       | $1.12 \cdot V108$               | $5 \cdot V108$                                                               |
| V124      | REAL  | 0.01       | $1.12 \cdot V115$               | $5 \cdot V115$                                                               |
| V125      | UDINT | 1          | 0                               | $(100.0 \cdot V100) / V108$                                                  |
| V126      | UDINT | 1          | 0                               | $V100 \cdot 127 / V108$                                                      |
| V128      | UDINT | 1          | 50                              | 100                                                                          |
| V129      | UDINT | 1          | 50                              | 100                                                                          |
| V130      | UDINT | 1          | 1                               | 500                                                                          |
| V131      | UDINT | 1          | 1                               | 500                                                                          |
| V132      | UDINT | 1          | 1                               | 500                                                                          |
| V133      | UDINT | 1          | 1                               | 500                                                                          |
| V135      | DWORD | 1          | 0                               | 7h                                                                           |
| V136      | UDINT | 1          | 25                              | 1500                                                                         |
| V137      | UDINT | 1          | 25                              | 1500                                                                         |
| V140      | UDINT | 1          | 1                               | 500                                                                          |
| V141      | DINT  | 1          | -2147483648                     | 2147483647                                                                   |
| V142      | DINT  | 1          | -2147483648                     | 2147483647                                                                   |
| V144      | UDINT | 1          | 5                               | 1500                                                                         |
| V145      | DWORD | 1          | 0                               | FFFFFFFFh                                                                    |
| V150      | UDINT | 1          | 1                               | 500                                                                          |
| V151      | DINT  | 1          | -2147483648                     | 2147483647                                                                   |
| V152      | DINT  | 1          | -2147483648                     | 2147483647                                                                   |
| V154      | UDINT | 1          | 5                               | 1500                                                                         |
| V155      | DWORD | 1          | 0                               | FFFFFFFFh                                                                    |
| V160      | DWORD | 1          | 0                               | FFFh                                                                         |
| V162      | UDINT | 1          | 0                               | 32767                                                                        |
| V163      | UDINT | 1          | 1                               | 32767                                                                        |
| V164      | DWORD | 1          | 0                               | 3h                                                                           |
| V165      | UDINT | 1          | 0                               | 4000                                                                         |
| V166      | REAL  | 0.01       | 0.0                             | $32760 \cdot 1000 / (5.656941 \cdot (I_{peak} \cdot 100) \cdot (V263 / 50))$ |
| V169      | DWORD | 1          | 0                               | 1h                                                                           |
| V170      | DWORD | 1          | 0                               | 7h                                                                           |
| V171      | REAL  | 0.01       | $-(V108)$                       | V108                                                                         |
| V172      | REAL  | 0.01       | 0.0                             | $32760 \cdot 21.5792 / ((I_{peak} \cdot 100) \cdot (V263 / 50))$             |
| V173      | REAL  | 0.01       | $2 \cdot (4 \cdot V263) / 1000$ | $32767 \cdot (4 \cdot V263) / (1000 \cdot 0.99)$                             |
| V175      | UDINT | 1          | 0                               | 400                                                                          |
| V176      | UDINT | 1          | 0                               | $32760 \cdot (4 \cdot V263) / 1000$                                          |
| V177      | REAL  | 0.01       | 0.1                             | 10000.0                                                                      |

| Param. N° | Type   | Resolution | Min value [PU] | Max value [PU]            |
|-----------|--------|------------|----------------|---------------------------|
| V180      | UDINT  | 1          | 2              | 4                         |
| V181      | UDINT  | 1          | 0              | 3                         |
| V182      | DWORD  | 1          | 0              | 7h                        |
| V185      | UDINT  | 1          | 2              | 4                         |
| V186      | UDINT  | 1          | 0              | 3                         |
| V187      | DWORD  | 1          | 0              | 7h                        |
| V190      | UDINT  | 1          | 1              | 100000                    |
| V191      | UDINT  | 1          | 0              | 1000                      |
| V200      | DWORD  | 1          | 0              | 1h                        |
| V201      | DWORD  | 1          | 0              | 3h                        |
| V202      | UDINT  | 1          | 1              | 500                       |
| V217      | REAL   | 0.01       | 0.1            | 6.0                       |
| V218      | REAL   | 0.01       | 2.0            | 150.0                     |
| V219      | UDINT  | 1          | 0              | 32767                     |
| V220      | DWORD  | 1          | 0              | 1Fh                       |
| V221      | DWORD  | 1          | 0              | 3Fh                       |
| V238      | DWORD  |            | 1              | 80000000h                 |
| V239      | DWORD  |            | 1              | 80000000h                 |
| V240      | REAL   | 0.01       | 1.0            | 5.0                       |
| V241      | UDINT  | 1          | 5              | 1000                      |
| V242      | REAL   | 0.01       | 0.0            | 32767.0                   |
| V243      | REAL   | 0.01       | 0.0            | 32767.0                   |
| V244      | REAL   | 0.01       | 0.0            | 32767.0                   |
| V245      | REAL   | 0.01       | 1.25           | 32760/(13.1072*(V263/50)) |
| V246      | REAL   | 0.01       | 0.25           | V245-1.00                 |
| V247      | REAL   | 0.01       | 0.0            | 32767.0                   |
| V248      | REAL   | 0.01       | 1.0            | 5.0                       |
| V249      | UDINT  | 1          | 5              | 1000                      |
| V250      | DWORD  | 1          | 0              | FFFFh                     |
| V251      | UDINT  | 1          | 1              | 1                         |
| V252      | STRING |            | I              | L                         |
| V253      | DWORD  | 1          | 0              | Fh                        |
| V255      | STRING |            | 0              | 2                         |
| V256      | STRING |            | 0              | P                         |
| V257      | UDINT  | 1          | 1              | 32767                     |
| V259      | REAL   | 0.01       | 1.0            | 32760.0                   |
| V263      | UDINT  | 25         | 25             | 50                        |
| V268      | UDINT  | 1          | 0              | 359999                    |
| V269      | UDINT  | 1          | 0              | 359999                    |
| V270      | UDINT  | 1          | 0              | 80000                     |
| V271      | UDINT  | 25         | V263           | 2*V263                    |
| V272      | DWORD  | 1          | 0              | 7h                        |
| V273      | DWORD  | 1          | 0              | 1FFh                      |
| V301      | UDINT  | 1          | 0              | 50000                     |
| V305      | UDINT  | 1          | 0              | 2147483647                |
| V306      | DINT   | 1          | -67108863      | 67108863                  |
| V307      | UDINT  | 1          | 1              | 65535                     |
| V310      | DINT   | 1          | -2147483648    | 2147483647                |
| V311      | DINT   | 1          | -67108863      | 67108863                  |

| Param. N° | Type  | Resolution | Min value [PU] | Max value [PU]    |
|-----------|-------|------------|----------------|-------------------|
| V312      | UDINT | 1          | 1              | 65535             |
| V313      | DINT  | 1          | -2147483648    | 2147483647        |
| V314      | DWORD | 1          | 0              | Fh                |
| V315      | DWORD | 1          | 0              | Fh                |
| V317      | REAL  | 0.01       | 2.5            | 11.5              |
| V318      | REAL  | 0.01       | 2.5            | 11.5              |
| V320      | DWORD |            | 1              | 80000000h         |
| V321      | DWORD |            | 1              | 80000000h         |
| V322      | DWORD |            | 1              | 80000000h         |
| V323      | DWORD |            | 1              | 80000000h         |
| V324      | DWORD | 1          | 0              | 3FFh              |
| V325      | DWORD | 1          | 0              | 1FFh              |
| V326      | DWORD | 1          | 0              | 7h                |
| V327      | DWORD | 1          | 0              | 7h                |
| V328      | DWORD | 1          | 0              | FFh               |
| V329      | DWORD | 1          | 0              | FFh               |
| V330      | DWORD |            | 1              | 80000000h         |
| V331      | DWORD |            | 1              | 80000000h         |
| V340      | UDINT | 1          | 0              | 6                 |
| V341      | UDINT | 1          | 0              | 6                 |
| V342      | REAL  | 0.01       | 50.0           | 3000.00*(50/V263) |
| V343      | REAL  | 0.01       | 50.0           | 3000.00*(50/V263) |
| V344      | UDINT | 1          | 0              | 4294967295        |
| V345      | UDINT | 1          | 0              | 4294967295        |
| V350      | DWORD | 1          | 0              | 1h                |
| V353      | REAL  | 0.01       | 1.0            | 100000.0          |
| V359      | UDINT | 1          | 0              | 32767             |
| V360      | UDINT | 1          | 1              | 360000            |
| V361      | UDINT | 1          | 1              | 2147483647        |
| V362      | UDINT | 1          | 0              | 32767             |
| V363      | UDINT | 1          | 0              | 32767             |
| V364      | UDINT | 1          | 0              | 32767             |
| V365      | UDINT | 1          | 1              | 65536             |
| V368      | UDINT | 1          | 1              | 32767             |
| V369      | UDINT | 1          | 1              | 32                |
| V370      | UDINT | 1          | 1              | 360000            |
| V371      | UDINT | 1          | 1              | 2147483647        |
| V372      | UDINT | 1          | 0              | 32767             |
| V373      | UDINT | 1          | 0              | 32767             |
| V374      | UDINT | 1          | 0              | 32767             |
| V375      | UDINT | 1          | 1              | 65536             |
| V376      | UDINT | 1          | 0              | 32767             |
| V378      | UDINT | 1          | 1              | 32767             |
| V379      | UDINT | 1          | 1              | 32                |
| V380      | UDINT | 1          | 1              | 8000              |
| V381      | UDINT | 1          | 1              | 32767             |
| V382      | UDINT | 1          | 0              | 4294967295        |
| V383      | UDINT | 1          | 0              | 32767             |
| V384      | UDINT | 1          | 0              | 32767             |

| Param. N° | Type  | Resolution | Min value [PU]                                                                   | Max value [PU]                                                             |
|-----------|-------|------------|----------------------------------------------------------------------------------|----------------------------------------------------------------------------|
| V385      | UDINT | 1          | 0                                                                                | 32767                                                                      |
| V386      | UDINT | 1          | 0                                                                                | 32767                                                                      |
| V389      | UDINT | 1          | 0                                                                                | 48                                                                         |
| V390      | UDINT | 1          | 1                                                                                | 8000                                                                       |
| V391      | UDINT | 1          | 1                                                                                | 32767                                                                      |
| V392      | UDINT | 1          | 0                                                                                | 4294967295                                                                 |
| V393      | UDINT | 1          | 0                                                                                | 32767                                                                      |
| V394      | UDINT | 1          | 0                                                                                | 4294967295                                                                 |
| V395      | UDINT | 1          | 0                                                                                | 32767                                                                      |
| V396      | UDINT | 1          | 0                                                                                | 32767                                                                      |
| V399      | UDINT | 1          | 0                                                                                | 48                                                                         |
| V400      | UDINT | 1          | 0                                                                                | $32760 \cdot 50 \cdot V403 / (3.2768 \cdot V406 \cdot V271)$               |
| V401      | UDINT | 1          | 0                                                                                | $32760 \cdot 50 \cdot V403 / (3.2768 \cdot V406 \cdot V271)$               |
| V402      | UDINT | 1          | 0                                                                                | $(32760 \cdot V403 / 65.536) - V408 \cdot 1000 \cdot V407$                 |
| V403      | REAL  | 0.01       | $133774.75 \cdot 3276.8 \cdot (V271/50) / ((I_{peak} \cdot 100) \cdot 32760.00)$ | 32760.0                                                                    |
| V404      | UDINT | 1          | 0                                                                                | 32767                                                                      |
| V405      | UDINT | 1          | 0                                                                                | 32767                                                                      |
| V406      | UDINT | 1          | 1                                                                                | 16                                                                         |
| V407      | UDINT | 1          | 3                                                                                | 4                                                                          |
| V408      | REAL  | 0.01       | 0.0                                                                              | 20.0                                                                       |
| V409      | UDINT | 1          | 0                                                                                | 700/1.4142135                                                              |
| V410      | REAL  | 0.01       | 0.0                                                                              | 127.9                                                                      |
| V411      | REAL  | 0.01       | $2 \cdot (2 \cdot V263) / 1000$                                                  | $(2 \cdot V263) \cdot 32.767 / 0.99$                                       |
| V412      | UDINT | 1          | 2                                                                                | 100                                                                        |
| V413      | UDINT | 1          | 0                                                                                | 32767                                                                      |
| V414      | UDINT | 1          | 0                                                                                | $32760 \cdot 100 / (0.3431457 \cdot V257 \cdot V271)$                      |
| V415      | UDINT | 1          | 0                                                                                | $32760 \cdot 100 / (0.3431457 \cdot V257 \cdot V271)$                      |
| V417      | UDINT | 1          | 0                                                                                | $32760 / (V257 \cdot 0.21845333 \cdot (V263/50))$                          |
| V425      | UDINT | 1          | 1                                                                                | 20                                                                         |
| V426      | REAL  | 0.01       | 0.0                                                                              | $32760 \cdot (I_{peak} \cdot 100) / (7592.3856 \cdot 100 \cdot (V263/50))$ |
| V427      | REAL  | 0.01       | 0.0                                                                              | $13107 \cdot (I_{peak} \cdot 100) / (V412 \cdot 13107.2 \cdot 1.4142135)$  |
| V428      | REAL  | 0.01       | 0.0                                                                              | $(I_{peak} \cdot 100) / 141.42$                                            |
| V434      | REAL  | 0.01       | 0.0                                                                              | $926790 \cdot 4 / (I_{peak} \cdot 100)$                                    |
| V435      | REAL  | 0.01       | $2 \cdot V271 / 1000$                                                            | $32767 \cdot V271 / (1000 \cdot 0.99)$                                     |
| V448      | UDINT | 1          | 0                                                                                | 32767                                                                      |
| V450      | REAL  | 0.01       | 0.0                                                                              | 127.9                                                                      |
| V451      | REAL  | 0.01       | $2 \cdot (2 \cdot V263) / 1000$                                                  | $(2 \cdot V263) \cdot 32.767 / 0.99$                                       |
| V452      | UDINT | 1          | 2                                                                                | 100                                                                        |
| V453      | UDINT | 1          | 0                                                                                | 32767                                                                      |
| V454      | UDINT | 1          | 0                                                                                | $32760 \cdot 100 / (0.3431457 \cdot V257 \cdot V271)$                      |
| V455      | UDINT | 1          | 0                                                                                | $32760 \cdot 100 / (0.3431457 \cdot V257 \cdot V271)$                      |
| V457      | UDINT | 1          | 0                                                                                | $32760 / (V257 \cdot 0.21845333 \cdot (V263/50))$                          |
| V465      | UDINT | 1          | 1                                                                                | 20                                                                         |
| V466      | REAL  | 0.01       | 0.0                                                                              | $32760 \cdot (I_{peak} \cdot 100) / (7592.3856 \cdot 100 \cdot (V263/50))$ |
| V467      | REAL  | 0.01       | 0.0                                                                              | $13107 \cdot (I_{peak} \cdot 100) / (V452 \cdot 13107.2 \cdot 1.4142135)$  |
| V468      | REAL  | 0.01       | 0.0                                                                              | $(I_{peak} \cdot 100) / 141.42$                                            |
| V470      | REAL  | 0.01       | 0.0                                                                              | 32767.0                                                                    |
| V471      | REAL  | 0.01       | 0.0                                                                              | 32767.0                                                                    |

| Param. N° | Type  | Resolution | Min value [PU]                   | Max value [PU]                                                                                                     |
|-----------|-------|------------|----------------------------------|--------------------------------------------------------------------------------------------------------------------|
| V472      | UDINT | 1          | 0                                | 32767                                                                                                              |
| V473      | UDINT | 1          | 0                                | 32767                                                                                                              |
| V474      | REAL  | 0.01       | 0.0                              | $926790 \cdot 4 / (I_{peak} \cdot 100)$                                                                            |
| V475      | REAL  | 0.01       | $2 \cdot V271 / 1000$            | $32767 \cdot V271 / (1000 \cdot 0.99)$                                                                             |
| V479      | REAL  | 0.01       | 1.0                              | 2000.0                                                                                                             |
| V480      | REAL  | 0.01       | 0.0                              | 100.0                                                                                                              |
| V481      | DWORD | 1          | 0                                | 7h                                                                                                                 |
| V488      | UDINT | 1          | 0                                | $32760 \cdot ((234.5 + 20.0) \cdot 2 \cdot 133772.76) / ((I_{peak} \cdot 100) \cdot 65.536 \cdot (234.5 + 120.0))$ |
| V489      | REAL  | 0.01       | 0.0                              | $(32760 / ((I_{peak} \cdot 100) \cdot 0.01503011016)) - V408$                                                      |
| V491      | REAL  | 0.01       | $-(I_{peak} \cdot 100) / 141.42$ | 0.0                                                                                                                |
| V492      | REAL  | 0.01       | V491                             | 0.0                                                                                                                |
| V496      | UDINT | 1          | 0                                | $32760 / (0.21845333 \cdot V257 \cdot (V263 / 50))$                                                                |
| V497      | REAL  | 0.01       | 0.0                              | $32760 / (109.22667 \cdot V257 \cdot (V263 / 50) / V259)$                                                          |
| V498      | REAL  | 0.01       | 0.0                              | $32760 \cdot (I_{peak} \cdot 100) / (7592.3856 \cdot 100 \cdot (V263 / 50))$                                       |
| V499      | REAL  | 0.01       | 0.0                              | $32760 \cdot (I_{peak} \cdot 100) / (7592.3856 \cdot 100 \cdot (V263 / 50))$                                       |
| V500      | DWORD | 1          | 0                                | FFh                                                                                                                |
| V501      | DINT  | 1          | -32767                           | 32767                                                                                                              |
| V502      | UDINT | 1          | 0                                | 65535                                                                                                              |
| V503      | UDINT | 1          | 0                                | 65535                                                                                                              |
| V504      | DINT  | 1          | -32767                           | 32767                                                                                                              |
| V505      | DINT  | 1          | -32767                           | 32767                                                                                                              |
| V506      | DINT  | 1          | -32767                           | 32767                                                                                                              |
| V507      | UDINT | 1          | 0                                | 32767                                                                                                              |
| V508      | DINT  | 1          | -32767                           | 32767                                                                                                              |
| V509      | DINT  | 1          | -32767                           | 32767                                                                                                              |
| V511      | DINT  | 1          | -32767                           | 32767                                                                                                              |
| V512      | UDINT | 1          | 0                                | 65535                                                                                                              |
| V513      | UDINT | 1          | 0                                | 65535                                                                                                              |
| V514      | DINT  | 1          | -32767                           | 32767                                                                                                              |
| V515      | DINT  | 1          | -32767                           | 32767                                                                                                              |
| V516      | DINT  | 1          | -32767                           | 32767                                                                                                              |
| V517      | UDINT | 1          | 0                                | 32767                                                                                                              |
| V518      | DINT  | 1          | -32767                           | 32767                                                                                                              |
| V519      | DINT  | 1          | -32767                           | 32767                                                                                                              |
| V521      | DINT  | 1          | -32767                           | 32767                                                                                                              |
| V522      | UDINT | 1          | 0                                | 65535                                                                                                              |
| V523      | UDINT | 1          | 0                                | 65535                                                                                                              |
| V524      | DINT  | 1          | -32767                           | 32767                                                                                                              |
| V525      | DINT  | 1          | -32767                           | 32767                                                                                                              |
| V526      | DINT  | 1          | -32767                           | 32767                                                                                                              |
| V527      | UDINT | 1          | 0                                | 32767                                                                                                              |
| V528      | DINT  | 1          | -32767                           | 32767                                                                                                              |
| V529      | DINT  | 1          | -32767                           | 32767                                                                                                              |
| V531      | DINT  | 1          | -32767                           | 32767                                                                                                              |
| V532      | UDINT | 1          | 0                                | 65535                                                                                                              |
| V533      | UDINT | 1          | 0                                | 65535                                                                                                              |
| V534      | DINT  | 1          | -32767                           | 32767                                                                                                              |
| V535      | DINT  | 1          | -32767                           | 32767                                                                                                              |

| Param. N° | Type  | Resolution | Min value [PU]    | Max value [PU]                                                            |
|-----------|-------|------------|-------------------|---------------------------------------------------------------------------|
| V536      | DINT  | 1          | -32767            | 32767                                                                     |
| V537      | UDINT | 1          | 0                 | 32767                                                                     |
| V538      | DINT  | 1          | -32767            | 32767                                                                     |
| V539      | DINT  | 1          | -32767            | 32767                                                                     |
| V541      | DINT  | 1          | -32767            | 32767                                                                     |
| V542      | UDINT | 1          | 0                 | 65535                                                                     |
| V543      | UDINT | 1          | 0                 | 65535                                                                     |
| V544      | DINT  | 1          | -32767            | 32767                                                                     |
| V545      | DINT  | 1          | -32767            | 32767                                                                     |
| V546      | DINT  | 1          | -32767            | 32767                                                                     |
| V547      | UDINT | 1          | 0                 | 32767                                                                     |
| V548      | DINT  | 1          | -32767            | 32767                                                                     |
| V549      | DINT  | 1          | -32767            | 32767                                                                     |
| V600      | UDINT | 1          | 0                 | $(I_{peak} \cdot 100) / (V_{108} \cdot 1.4142)$                           |
| V601      | UDINT | 1          | 10                | 1500                                                                      |
| V602      | REAL  | 0.01       | 100.0             | 500.0                                                                     |
| V603      | REAL  | 0.01       | 0.0               | 50.0                                                                      |
| V604      | UDINT | 1          | 0                 | $32760 \cdot 133772.76 / ((I_{peak} \cdot 100) \cdot 65.536 \cdot 1.4)$   |
| V605      | UDINT | 1          | 0                 | $V_{600} \cdot 91 / 100$                                                  |
| V606      | UDINT | 1          | $V_{601} / 2$     | 1500                                                                      |
| V607      | REAL  | 0.01       | 0.0               | $32760 / ((I_{peak} \cdot 100) \cdot 0.0150311016 \cdot (50 / V_{263}))$  |
| V608      | REAL  | 0.01       | 0.0               | $32760 / ((I_{peak} \cdot 100) \cdot 0.01503011016 \cdot (50 / V_{263}))$ |
| V620      | UDINT | 1          | 10                | 120                                                                       |
| V621      | UDINT | 1          | 10                | 120                                                                       |
| V622      | UDINT | 1          | 50                | 500                                                                       |
| V623      | UDINT | 1          | 1                 | 2000                                                                      |
| V680      | DWORD | 1          | 0                 | 3h                                                                        |
| V681      | DWORD |            | 1                 | 80000000h                                                                 |
| V682      | UDINT | 1          | 1                 | 500                                                                       |
| V683      | UDINT | 1          | 1                 | 500                                                                       |
| V684      | UDINT | 1          | 1                 | 500                                                                       |
| V685      | UDINT | 50         | $2 \cdot V_{263}$ | 6553500                                                                   |
| V686      | UDINT | 1          | 2                 | 2048                                                                      |
| V687      | UDINT | 1          | 1                 | $V_{686} - 1$                                                             |
| V688      | DINT  | 1          | -2147483648       | 2147483647                                                                |
| V700      | DWORD |            | 0                 | 7E7E7E7Eh                                                                 |
| V701      | DWORD |            | 0                 | 7E7E7E7Eh                                                                 |
| V702      | DWORD |            | 0                 | 7E7E7E7Eh                                                                 |
| V703      | DWORD |            | 0                 | 7E7E7E7Eh                                                                 |
| V704      | DWORD |            | 0                 | 7E7E7E7Eh                                                                 |
| V705      | DWORD |            | 0                 | 7E7E7E7Eh                                                                 |
| V706      | DWORD |            | 0                 | 7E7E7E7Eh                                                                 |
| V707      | DWORD |            | 0                 | 7E7E7E7Eh                                                                 |
| V708      | DWORD |            | 0                 | 7E7E7E7Eh                                                                 |
| V709      | DWORD |            | 0                 | 7E7E7E7Eh                                                                 |
| V712      | DWORD |            | 0                 | 7E7E7E7Eh                                                                 |
| V713      | DWORD |            | 0                 | 7E7E7E7Eh                                                                 |
| V714      | DWORD |            | 0                 | 7E7E7E7Eh                                                                 |
| V715      | DWORD |            | 0                 | 7E7E7E7Eh                                                                 |

| Param. N° | Type  | Resolution | Min value [PU] | Max value [PU] |
|-----------|-------|------------|----------------|----------------|
| V716      | DWORD |            | 0              | 7E7E7E7Eh      |
| V717      | DWORD |            | 0              | 7E7E7E7Eh      |
| V718      | DWORD |            | 0              | 7E7E7E7Eh      |
| V719      | DWORD |            | 0              | 7E7E7E7Eh      |
| V720      | DWORD |            | 0              | 7E7E7E7Eh      |
| V721      | DWORD |            | 0              | 7E7E7E7Eh      |
| V724      | DWORD |            | 0              | 7E7E7E7Eh      |
| V725      | DWORD |            | 0              | 7E7E7E7Eh      |
| V726      | DWORD |            | 0              | 7E7E7E7Eh      |
| V727      | DWORD |            | 0              | 7E7E7E7Eh      |
| V728      | DWORD |            | 0              | 7E7E7E7Eh      |
| V729      | DWORD |            | 0              | 7E7E7E7Eh      |
| V730      | DWORD |            | 0              | 7E7E7E7Eh      |
| V731      | DWORD |            | 0              | 7E7E7E7Eh      |
| V732      | DWORD |            | 0              | 7E7E7E7Eh      |
| V733      | DWORD |            | 0              | 7E7E7E7Eh      |
| V900      | DINT  | 1          | -2147483648    | 2147483647     |
| V901      | DINT  | 1          | -2147483648    | 2147483647     |
| V902      | DINT  | 1          | -2147483648    | 2147483647     |
| V903      | DINT  | 1          | -2147483648    | 2147483647     |
| V904      | DINT  | 1          | -2147483648    | 2147483647     |
| V905      | DINT  | 1          | -2147483648    | 2147483647     |
| V906      | DINT  | 1          | -2147483648    | 2147483647     |
| V907      | DINT  | 1          | -2147483648    | 2147483647     |
| V908      | DINT  | 1          | -2147483648    | 2147483647     |
| V909      | DINT  | 1          | -2147483648    | 2147483647     |
| V910      | DINT  | 1          | -2147483648    | 2147483647     |
| V911      | DINT  | 1          | -2147483648    | 2147483647     |
| V912      | DINT  | 1          | -2147483648    | 2147483647     |
| V913      | DINT  | 1          | -2147483648    | 2147483647     |
| V914      | DINT  | 1          | -2147483648    | 2147483647     |
| V915      | DINT  | 1          | -2147483648    | 2147483647     |
| V920      | DINT  | 1          | -2147483648    | 2147483647     |
| V921      | DINT  | 1          | -2147483648    | 2147483647     |
| V922      | DINT  | 1          | -2147483648    | 2147483647     |
| V923      | DINT  | 1          | -2147483648    | 2147483647     |
| V924      | DINT  | 1          | -2147483648    | 2147483647     |
| V925      | DINT  | 1          | -2147483648    | 2147483647     |
| V926      | DINT  | 1          | -2147483648    | 2147483647     |
| V927      | DINT  | 1          | -2147483648    | 2147483647     |
| V928      | DINT  | 1          | -2147483648    | 2147483647     |
| V929      | DINT  | 1          | -2147483648    | 2147483647     |
| V930      | DINT  | 1          | -2147483648    | 2147483647     |
| V931      | DINT  | 1          | -2147483648    | 2147483647     |
| V932      | DINT  | 1          | -2147483648    | 2147483647     |
| V933      | DINT  | 1          | -2147483648    | 2147483647     |
| V934      | DINT  | 1          | -2147483648    | 2147483647     |
| V935      | DINT  | 1          | -2147483648    | 2147483647     |
| V1045     | REAL  | 0.01       | 0.0            | 20000.0        |

| Param. N° | Type  | Resolution | Min value [PU]                     | Max value [PU]                                                                               |
|-----------|-------|------------|------------------------------------|----------------------------------------------------------------------------------------------|
| V1046     | REAL  | 0.01       | 0.0                                | 2000.0                                                                                       |
| V1050     | REAL  | 0.01       | 0.0                                | $(I_{peak} \cdot 100) / 141.42$                                                              |
| V1051     | REAL  | 0.01       | 0.0                                | $16777200 \cdot (I_{peak} \cdot 100) / (90509.668 \cdot (50 / V_{263}))$                     |
| V1052     | REAL  | 0.01       | $2 \cdot (2 \cdot V_{263}) / 1000$ | $32767 \cdot (2 \cdot V_{263}) / (0.99 \cdot 1000)$                                          |
| V1057     | REAL  | 0.01       | 0.0                                | $(I_{peak} \cdot 100) / 141.42$                                                              |
| V1058     | REAL  | 0.01       | 0.0                                | $16777200 \cdot (I_{peak} \cdot 100) / (90509.668 \cdot (50 / V_{263}))$                     |
| V1059     | REAL  | 0.01       | $2 \cdot (2 \cdot V_{263}) / 1000$ | $32767 \cdot (2 \cdot V_{263}) / (0.99 \cdot 1000)$                                          |
| V1060     | REAL  | 0.01       | 0.0                                | $16777200 \cdot (I_{peak} \cdot 100) / (741455.2 \cdot (50 / V_{263}))$                      |
| V1091     | UDINT | 1          | 4                                  | 1000000                                                                                      |
| V1092     | UDINT | 1          | 4                                  | 1000000                                                                                      |
| V1093     | UDINT | 1          | 5                                  | 100000                                                                                       |
| V1094     | UDINT | 1          | 0                                  | 200                                                                                          |
| V1095     | UDINT | 1          | 0                                  | 1000                                                                                         |
| V1096     | UDINT | 1          | 0                                  | 1000                                                                                         |
| V1097     | UDINT | 1          | 0                                  | 200                                                                                          |
| V1098     | UDINT | 1          | 0                                  | $140000 \cdot 50 \cdot (1 + 0.25 \cdot ((V_{271} / V_{263}) - 1)) / (V_{257} \cdot V_{271})$ |
| V1099     | UDINT | 1          | 0                                  | $140000 \cdot 50 \cdot (1 + 0.25 \cdot ((V_{271} / V_{263}) - 1)) / (V_{257} \cdot V_{271})$ |
| V1301     | UDINT | 1          | 0                                  | 50000                                                                                        |
| V1305     | UDINT | 1          | 0                                  | 2147483647                                                                                   |
| V1306     | DINT  | 1          | -67108863                          | 67108863                                                                                     |
| V1307     | UDINT | 1          | 1                                  | 65535                                                                                        |
| V1310     | DINT  | 1          | -2147483648                        | 2147483647                                                                                   |
| V1311     | DINT  | 1          | -67108863                          | 67108863                                                                                     |
| V1312     | UDINT | 1          | 1                                  | 65535                                                                                        |
| V1313     | DINT  | 1          | -2147483648                        | 2147483647                                                                                   |
| V1353     | REAL  | 0.01       | 1.0                                | 100000.0                                                                                     |
| V1359     | UDINT | 1          | 0                                  | 32767                                                                                        |
| V1362     | UDINT | 1          | 0                                  | 32767                                                                                        |
| V1372     | UDINT | 1          | 0                                  | 32767                                                                                        |
| V1376     | UDINT | 1          | 0                                  | 32767                                                                                        |
| V1383     | UDINT | 1          | 0                                  | 32767                                                                                        |
| V1393     | UDINT | 1          | 0                                  | 32767                                                                                        |

**A.4 NCK parameters****A.4.1 NCK Parameters for Flexium<sup>+</sup>**

The following parameters can be updated on the fly :

|                    |                                                                                                        |
|--------------------|--------------------------------------------------------------------------------------------------------|
| <b>P11</b>         | Measurement conversion coefficient for analog axes                                                     |
| <b>P13</b>         | Handwheel measurement conversion coefficient                                                           |
| <b>P15</b>         | Homing direction                                                                                       |
| <b>P16</b>         | Machine home position                                                                                  |
| <b>P17</b>         | Axes travel limits                                                                                     |
| <b>P18</b>         | Backlash error compensation                                                                            |
| <b>P19</b>         | High Speed Cutting: Feed-forward acceleration time constant -<br>Dry friction correction time constant |
| <b>P20</b>         | Dry friction correction amplitude                                                                      |
| <b>P23</b>         | Maximum following error                                                                                |
| <b>P24.. P25</b>   | Axes synchronization and duplication                                                                   |
| <b>P30</b>         | Maximum axes traverse rate                                                                             |
| <b>P32</b>         | Maximum permissible acceleration                                                                       |
| <b>P33</b>         | Approach speed                                                                                         |
| <b>P40</b>         | Number of sensor spindle pulses by revolution                                                          |
| <b>P41</b>         | Spindle reference reversal (according to the range selected)                                           |
| <b>P42</b>         | Spindle origin                                                                                         |
| <b>P43</b>         | Spindle indexing speed and spindle speed limits                                                        |
| <b>P44</b>         | Spindle indexing in-position window                                                                    |
| <b>P45</b>         | Spindle indexing gain                                                                                  |
| <b>P46</b>         | Speed range and max speed for spindles @0 ... @7                                                       |
| <b>P47</b>         | Speed range and max speed for spindles @8 ... @15                                                      |
| <b>P48</b>         | Speed range and max speed for spindles @16 ... @23                                                     |
| <b>P49</b>         | Speed range and max speed for spindles @24 ... @31                                                     |
| <b>P55 N0 ÷ N7</b> | Speed anticipation coefficient - High Speed Cutting: number of filter terms                            |
| <b>P56</b>         | Channel specific time constant                                                                         |
| <b>P57</b>         | Dynamic lag control                                                                                    |
| <b>P66</b>         | Time constant for axis                                                                                 |
| <b>P70 N1, N5</b>  | Analog ports configuration                                                                             |
| <b>P73</b>         | Servo-system loop coefficient for analog axes                                                          |
| <b>P85</b>         | Drives coupling assignment                                                                             |
| <b>P116</b>        | Electronic Gear Box function (EGB)                                                                     |
| <b>P120 N2, N5</b> | Machine configuration                                                                                  |

---

## A.4.2 NCK Parameters for Flexium

The following parameters can be updated on the fly :

|            |                                                                     |
|------------|---------------------------------------------------------------------|
| <b>P3</b>  | Servo-controlled and interpolated axes                              |
| <b>P6</b>  | Spindle declaration and assignment to channels                      |
| <b>P10</b> | Axis measurement direction                                          |
| <b>P11</b> | Axis measurement conversion coefficient                             |
| <b>P12</b> | Handwheel measurement direction                                     |
| <b>P13</b> | Handwheel measurement conversion coefficient                        |
| <b>P14</b> | Handwheel declaration                                               |
| <b>P15</b> | Homing Direction                                                    |
| <b>P16</b> | Home position in machine dimensions                                 |
| <b>P17</b> | Axis travel limit                                                   |
| <b>P18</b> | Backlash error compensation                                         |
| <b>P19</b> | High Speed Cutting machining                                        |
| <b>P20</b> | Direction of axis speed reference                                   |
| <b>P21</b> | Servo-System loop gain coefficient for analog axes                  |
| <b>P23</b> | Maximum following error                                             |
| <b>P24</b> | Synchronised axes characteristics                                   |
| <b>P25</b> | Enable monitoring of analog axes signals                            |
| <b>P30</b> | Maximum axes traverse rates                                         |
| <b>P32</b> | Maximum permissible acceleration                                    |
| <b>P33</b> | Approach speed                                                      |
| <b>P40</b> | Spindle measurement conversion                                      |
| <b>P41</b> | Spindle reference reversal                                          |
| <b>P42</b> | Spindle origins                                                     |
| <b>P43</b> | Spindle indexing speed and speed limit by PLC                       |
| <b>P44</b> | Spindel indexing inposition window                                  |
| <b>P45</b> | Spindle indexing gain                                               |
| <b>P46</b> | Spindle 1 speed range                                               |
| <b>P47</b> | Spindle 2 speed range                                               |
| <b>P48</b> | Spindle 3 speed range                                               |
| <b>P49</b> | Spindle 4 speed range                                               |
| <b>P55</b> | Speed anticipation coefficient - Speed reference filter             |
| <b>P56</b> | Channel specific time constant                                      |
| <b>P57</b> | Dynamic lag control                                                 |
| <b>P70</b> | Axis/Spindle switching. Sensorless DISC NT spindles and Analog port |
| <b>P77</b> | Filtering SSI encoder faults                                        |
| <b>P85</b> | Tandem Function: master/slave coupled axes                          |
| <b>P86</b> | Tandem Function: Defines master axes with torque synchronisation    |

Page deliberately empty

|   |
|---|
| 1 |
| 2 |
| 3 |
| 4 |
| 5 |
| 6 |
| A |

---

Page deliberately empty

**Published by NUM Spa**

Via F. Somma 62  
20012 Cuggiono (MI) (Italy)  
Tel. : +39 02 979 691  
website: [www.num.com](http://www.num.com)

Printed September 2021