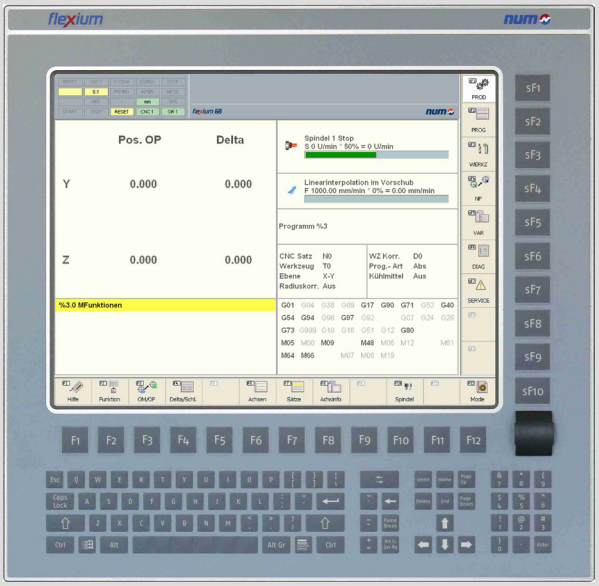
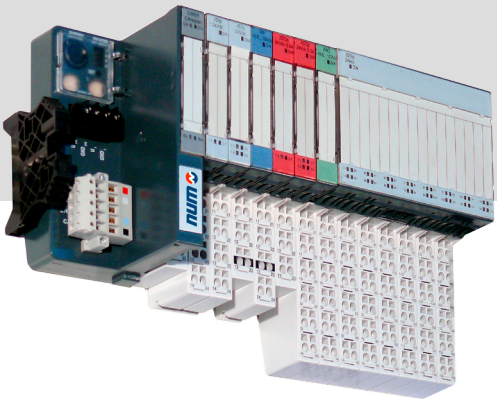
Flexium

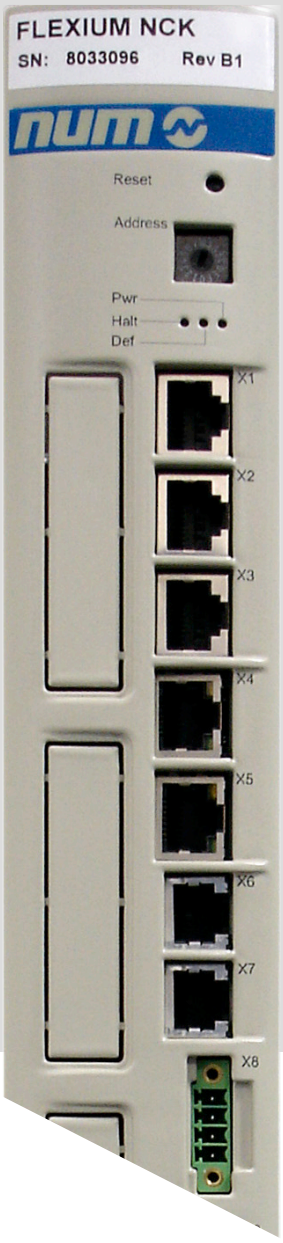
Commissioning manual

M00010EN-03

Edition 04.2011







www.num.com

num®
CNC HighEnd Applications

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.

Summary

Page deliberately empty.

Page deliberately empty.

Summary

Page deliberately empty.

1 Foreword

1.1 Use of the manual

The Commissioning Manual gives information on how to customize the Flexium system for a particular machine. This manual is for use by qualified personnel only.

To reduce the risk due to an improper use of the product, do not perform any servicing other than that contained in this manual unless you are qualified to do so.

To ensure proper use of these products, please always read this User's manual carefully and retain for future reference. Should the unit require maintenance, please contact a NUM authorized service center only.

1.2 Use of symbols 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.

1.3 Document revisions

Date	Index	Reason for revision
12 - 2008	00	Document creation
03 - 2009	01	Chapter 2.6 DPLC settings and connections. Chapter 2.5.2 new note.
06 - 2010	02	Revised to conform to software version 3.1.0.10.
04 - 2011	03	Revised to conform to software version 3.3.0.0. New chapters: - 2.9 MULTI-NCK configuration - 3.1.5 External parameters read by PLC - 3.2.7 Spindle load - 3.2.8 External parameters: Written by PLC - Read by Part program - 3.9 Special Settings - 3.9.1 Drive test points selection - 3.9.2 Spindle load settings - 3.10.1 Drive Test Point settings - 3.10.2 Spindle Load settings - 4 Flexium Functions “New chapter”. - Annexe A.1 - Configuration of Language.in - Annexe A.2 - Example of programming the MP04 using the NUM_MP04 function block. Modified chapters: - 2.6 System start up - 2.7 Flexium Tools - 2.7.1 Installation of he software - 2.7.2 Create a test project - 2.7.3 Going on line - 2.7.4.2 Login to application - 3.1 Information from NCK - 3.1.2.2 M Functions with acknowledge - 3.1.2.3 Decoded M functions - 3.1.2.4 List of M functions - 3.1.3.1 Axis status - 3.1.10 Embedded NCK inputs - 3.1.11 Exchange area from HMI to PLC - 3.1.11 Exchange signals to all NCK - 3.2.2 General exchange signals - 3.2.3 Outputs to the axes channels - 3.2.5 Servo drive control - 3.2.6 Spindle control - 3.2.13 Embedded NCK outputs - 3.3 Informations for the HMI - 3.4.1 Modification of tool wear offset by the PLC - 3.5 PLC Message - 3.9 Extended Help - 5.4.2 List of Servo-Control Settings Parameters

1.4 How to read the manual

The machine Commissioning Manual gives information on how to set the parameters to customise the Flexium for a particular machine.

Chapter	Description
1	**Foreword** This chapter introduces: - The use of the manual.
2	**General Commissioning** This chapter introduces: - General description of the NCK and system commissioning operations. - General configuration of the system.
3	**Flexium HMI Exchange Zone** This chapter introduces: - Description and use of the M functions, external parameters Exxxxx and variables. - General informations about Tools, Exchange area from HMI to PLC, Special functions .....
4	**Flexium functions** This chapter introduces: - Description and use of the Automatic homing. - Description and use of Auxiliary axes .... - Description and use of the Independent Channels, Duplicated and synchronised axes.... - Description and use of the Tandem function, Spindle synchronisation.... - Description and use of the Axis – Spindle commutation, Delayed Power down in case of error...
5	**Machine Parameters Integration** This chapter introduces: - Description of the Axis Declaration Channels assignment and Parameters. - Description of the Measurement Settings Channels assignment and Parameters. - Description of the Servo-Control Settings Channels assignment and Parameters. - Description of the Setting of Travels Channels assignment and Parameters. - Description of the Spindle Setting Channels assignment and Parameters. - Description of the Miscellaneous Parameters..
6	**Axis Calibration** This chapter introduces: - Detailed description of functions related to travel points, Corrections Records. - Detailed description of functions related Corrections Table..
7	**Inter-axis Calibration** This chapter introduces: - Detailed description of Inter-axis Calibration and Compensation Calibration. - Description of differents type Corrections.
A	Annexe - Configuration of Language.ini. - Example of programming the MP04 using the NUM_MP04 function block.

1.5 List of Parameters

The list of Parameters is available in the index section at the end of the manual.

1.6 NUM Agencies

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

1.7 Index

The index at the end of the manual facilitates access to infomation by keywords.

Page deliberately empty.

Page deliberately empty.

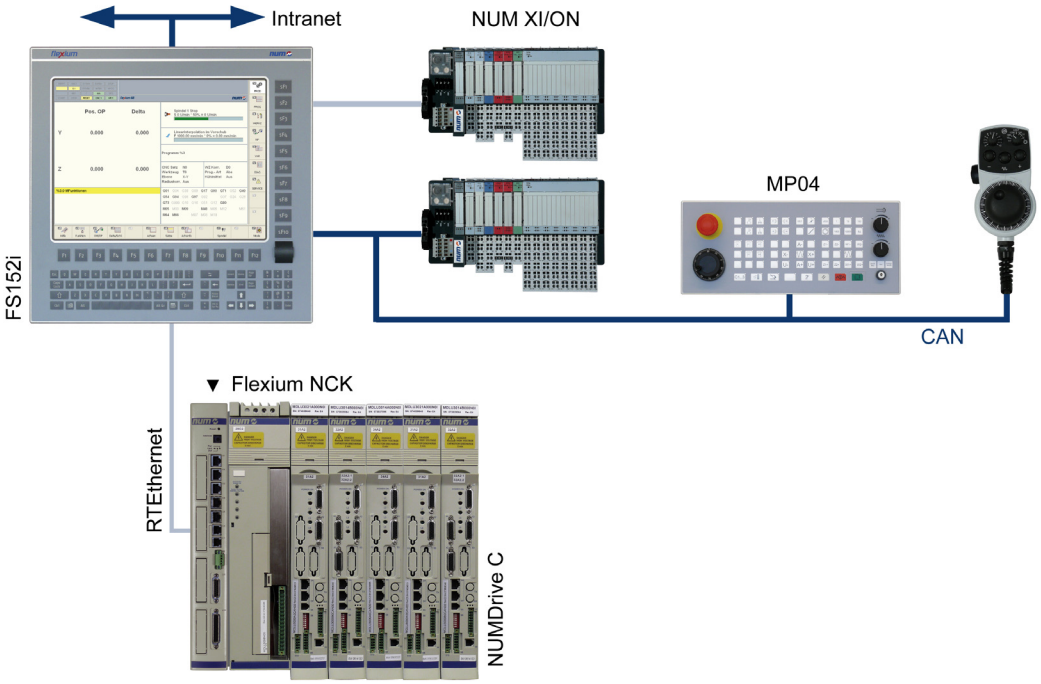
Summary

Page deliberately empty.

2 General Commissioning

2.1 Configuration

A standard Flexium application structure is as follows:



It includes the following main elements:

- **Flexium_NCK**: CNC Kernel dedicated for axes control.
  Some particular applications may include several NCK modules.
- **FS152i**: PLC and HMI functions. In some application FS152i can be subsituted by a DPLC module (PLC function alone) linked to a PC
  (HMI function + power demanding application like CAD, supervision ...)
- **XION**: Distributed and modular I/Os standard or on CANopen Bus
  (specific to Flexium 68).
- **MP04**: Operator panel on CANopen Bus
- **NUMDrive**: Drives and motors controlled via a digital interface (DISC NT)
- Various peripherals on **CANopen bus** (e.g. Handwheel).

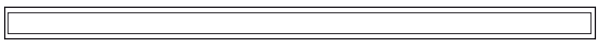
2.2 NCK Initial Operations

Three status LEDs are located on the front panel of the NCK. When powering up is complete, the green LED should be on steady. If other LEDs are lit, then there is an error in the system.

A LED can be OFF, ON or Blinking. There are several different possible LED blinking variants. They are described in the table below.

For detailled information about mechanical and electrical characteristics as well as wiring, please refer to the Flexium Installation manual M00009.

2.2.1 LED status signalling

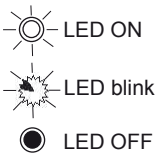
LED status	Meaning	Status timing
	LED ON	
	Blink 1	
	Blink 2	
	Blink 3	
	Blink 4	
	Blink 5 : two LEDs alternately blinking	
	LED OFF	

Legend : LED ON
LED blink
LED OFF

2.2.2 LED error signalling

The different errors and their associated LED states are described in the table below.

Led Status			
Red	**Orange**	**Green**	**Cause**
			Power OFF
			System OK
			Waiting for firmware upload. FS152i or DPLC not fully started
			Firmware upload in progress.
			Firmware upload aborted.
			Machine error (e.g. excess following error). CNC_Ready flag will be OFF
			Communication error between CNC and PLC. CNC_Ready flag and Watch Dog will be OFF.
			Too many DISC NT errors. Wiring – Perturbations or Drive fault.
			DISC NT overload. Check the configuration.
			Processor exception. Firmware error – Reboot system.
			Processor in Halt. Hardware error.

Legend :


In case of system malfunction, restart the full system (NCK and PLC).

2.2.3 Assigning an address to the NCK

Each NCK module must receive a unique address. This address is assigned with the aid of the 16-position rotary switch located on the NCK front panel.
As the application is directly linked to this address make sure the rotary switch is set properly.

The machine parameter P99 N1 defines the master NCK.
In standard configuration (one NCK module) P99 N1 must correspond to the physical address of the NCK module.

In Multi-NCK configurations each module should have its own physical address but its P99 N1must contain the address of the master NCK module.

For more detailed information on P99 please refer to the Flexium Parameters manual M00021.

2.3 FS152i

The PC Panel FS152i is delivered ready to connect to the NCK unit.
No software loading or adjustments are necessary.

For detailed information about wiring and configuring the FS152i, please refer to the FS152i Installation and Commissioning manual M00014.

2.4 MP04

2.4.1 Diagnostic LEDs

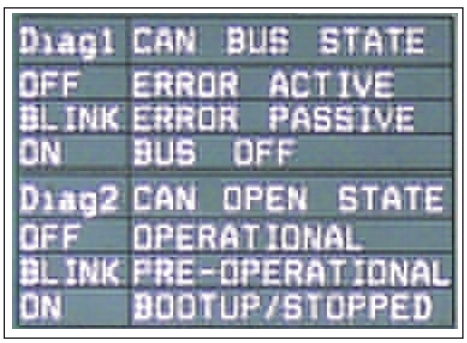
For diagnostic purposes there are 6 LEDs on the electronic board of the MP04:

For the explanation of the LEDs see the table below:

LED	Colour	Status	Possible cause and remedy
Run	Green	OFF	The module is in bootload state. No firmware loaded.
		Blinking	The firmware is working properly.
Diag1	Red	OFF	CAN-Bus-State=Error active (error free).
		Blinking	CAN-Bus-State=Error passive (disturbed, but working).
		ON	CAN-Bus-State=Bus OFF (node disconnected).
Diag2	Red	OFF	CAN-Bus-State=operational.
		Blinking	CAN-Bus-State=Pre operational.
		ON	CAN-Bus-State=Stopped or during Boot up. MP04 is in boot load mode.
Power	Green	OFF	The MP04 is switched off. The internal +5V power supply is malfunctioning.
		ON	The MP04 is switched on and the internal +5V power supply is functioning properly.
RX	Green		Receive CAN data (1)
TX	Red		Transmit CAN data (1)

(1) The light intensity is proportional to the bus traffic.

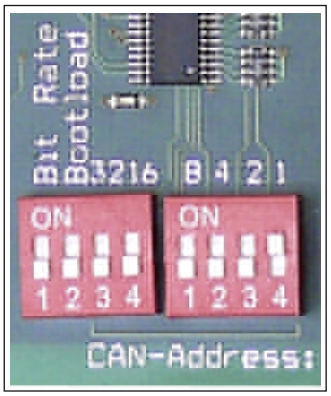
The two diagnostic LEDs are also described on the electronic board:



2.4.2 Address and Bit rate settings

Switches from left to right:

Block	Description	Position down (OFF)	Position up (ON)
Left	Bit rate	125 k Baud	500 k Baud
Left	Bootload	OFF	ON
Left	Address 32	OFF	ON
Left	Address 16	OFF	ON
Right	Address 8	OFF	ON
Right	Address 4	OFF	ON
Right	Address 2	OFF	ON
Right	Address 1	OFF	ON



Two bit rates are possible with the MP04 **(125k and 500k Baud)**.
The “Bootload” switch is used to load new firmware. It must be in the down position (OFF) during normal operation.

The CAN address can be set between 1 and 64 **(Pos 0 = CAN address 64)**

Example
Bus addresse 36:

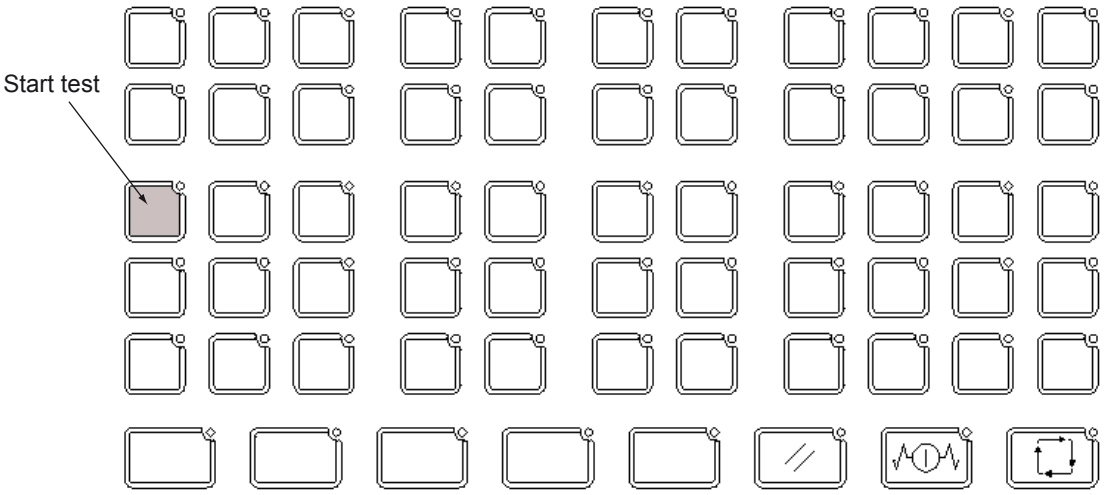
Bit rate	Bootload	32	16	8	4	2	1
X	OFF	ON	OFF	OFF	ON	OFF	OFF

2.4.3 Test program

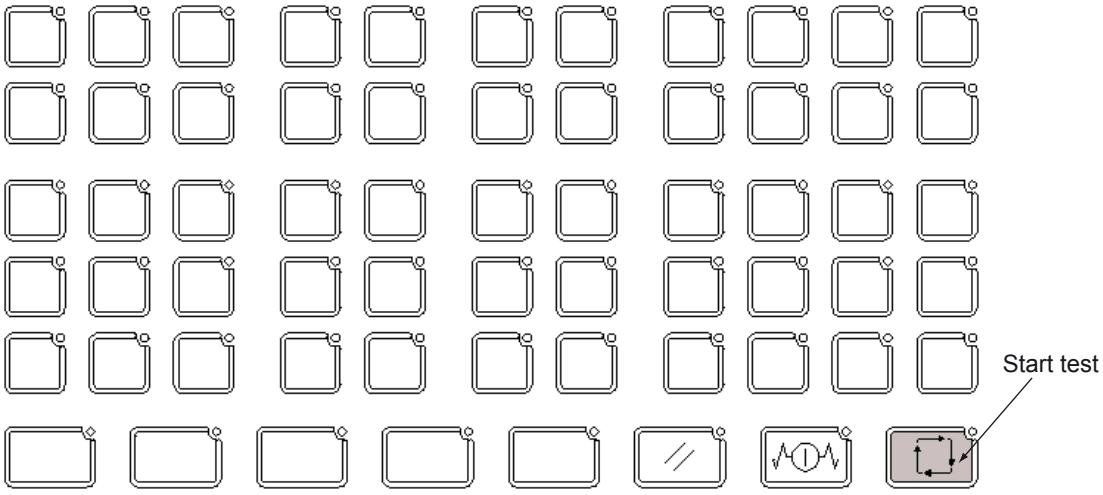
The MP04 has an integrated test program to test the LEDs, the keys, the potentiometers, the handwheel, the IOs, the DIP switches of the address and the firmware.

1. Switch off the power of the MP04.

2. Press the key *"Start test"*, hold it depressed and switch power on.
   The *"Start test"* key may now be released.



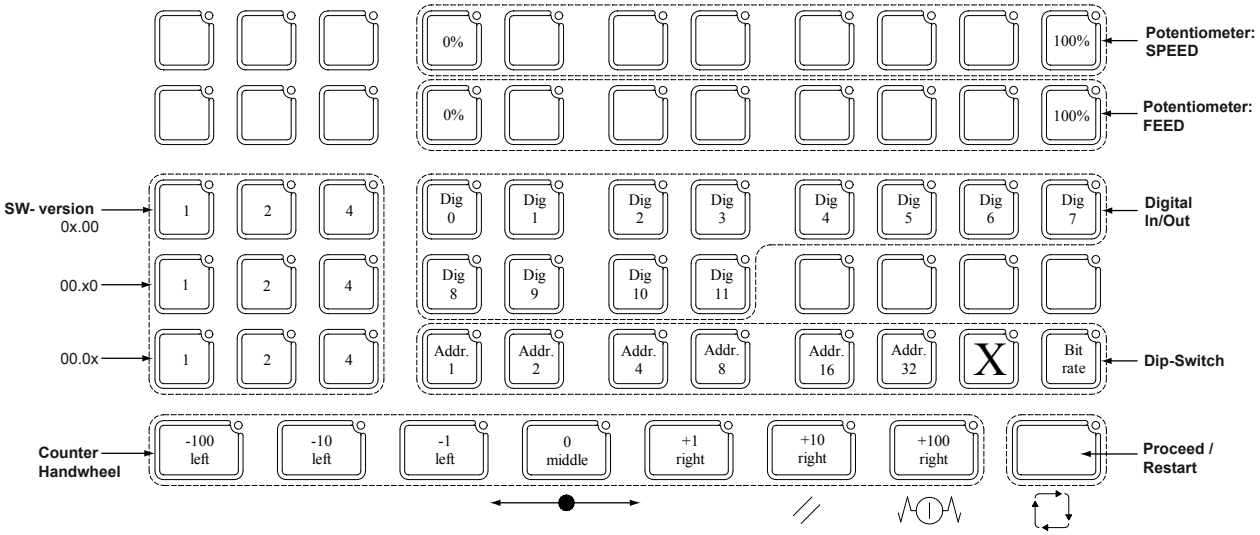
3. All LEDs will light up for 250ms one after another. So the function of the LEDs can be checked.
   After completion of this test, the LED of the *"Start"* key will flash.



4. Now the keys can be tested by pressing them. If the key is working properly the LED of the key lights up.

5. To select the next test, press the green *"Start"* key.

6. The areas shown below display the states and values of the hardware.



The values displayed are:

- *"SW-version"* : (software version) Software version of the panel. The version is displayed in binary.
- *"Counter handwheel"*: By turning the handwheel clockwise the LEDs will light up one after the other from left to right. (anticlockwise = right to left).
- *"Potentiometer SPEED"*: By turning the potentiometer the LEDs will light up from 0% to 100%.
- *"Potentiometer FEED"*: By turning the potentiometer the LEDs will light up from 0% to 100%.
- *"Digital In/Output"*: By pressing a key the relevant digital output will be forced. The LEDs shows the state of the digital inputs.
- *"Dip-Switch"*: Shows the positions of the DIP switches on the rear side of the MP04.

7. To restart the test, press the *"Start"* key with the flashing LED again.
8. To exit the test mode, switch off the power of the MP04.

2.4.4 Programming

The MP04 can be used like any other I/O. However to simplify the programming a function block has been created that access directly buttons and LEDs by row and column.

An example of usage of this function block is given in annex.

2.5 NUM XI/ON

For information about the NUM XI/ON Gateways and modules please refer to following manuals:

- XI/ON Gateway code M00001
- XI/ON digital IOs code M00002
- XI/ON analogue IOs code M00003
- XI/ON RS232 code M00004
- XI/ON RS485/422 code M00005.

2.6 System Start-up

The system elements are already programmed. For an initial startup please follow these steps:

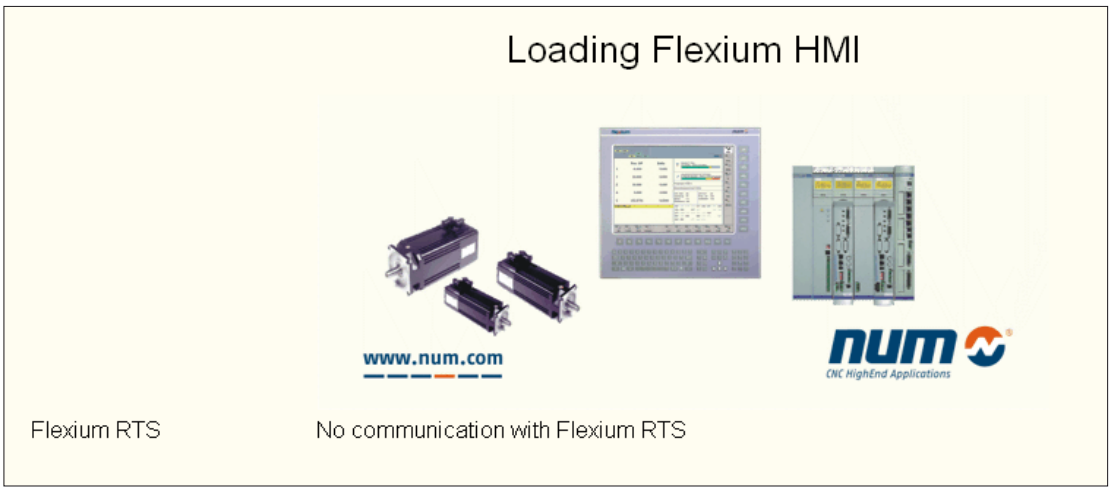
- Connect the ETH2 port of the FS152i (or DPLC) to the X5 port of the NCK.
  (standard Ethernet cable)
- Power ON FS152i and NCK 24V DC (for wiring refer to the Flexium Installation manual M00009)
  - Green LED of the NCK should be ON and the yellow LED blinking.
- Once the FS152i is initialised, Flexium HMI will open and you should eventually get the message:
  FLEXIUM CNC APPLICATION MISSING.

If the HMI did not start, please check that the Flexium HMI is well defined in the Start-up menu of Windows:

"C:\Program Files\NUM\Flexium HMI\FXBrowser\Bin\FXBrowser.exe"

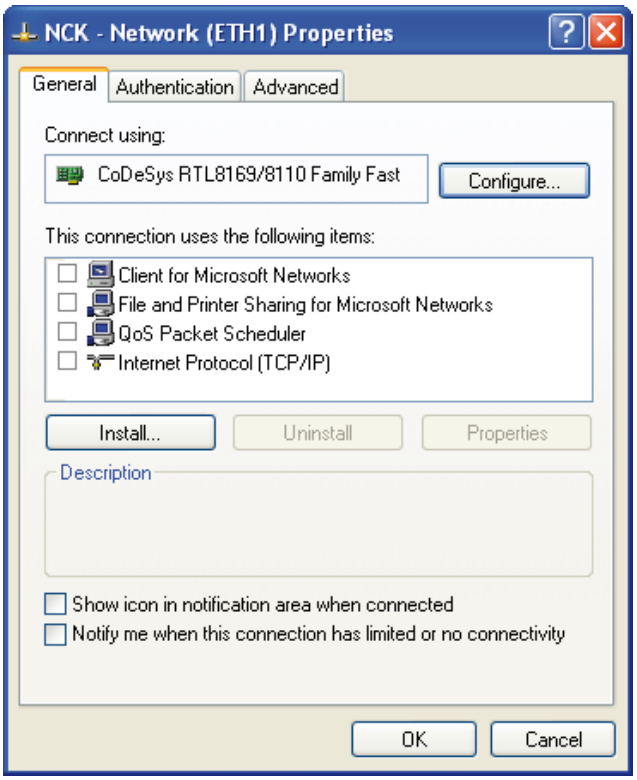
It is also possible to launch this file manually.

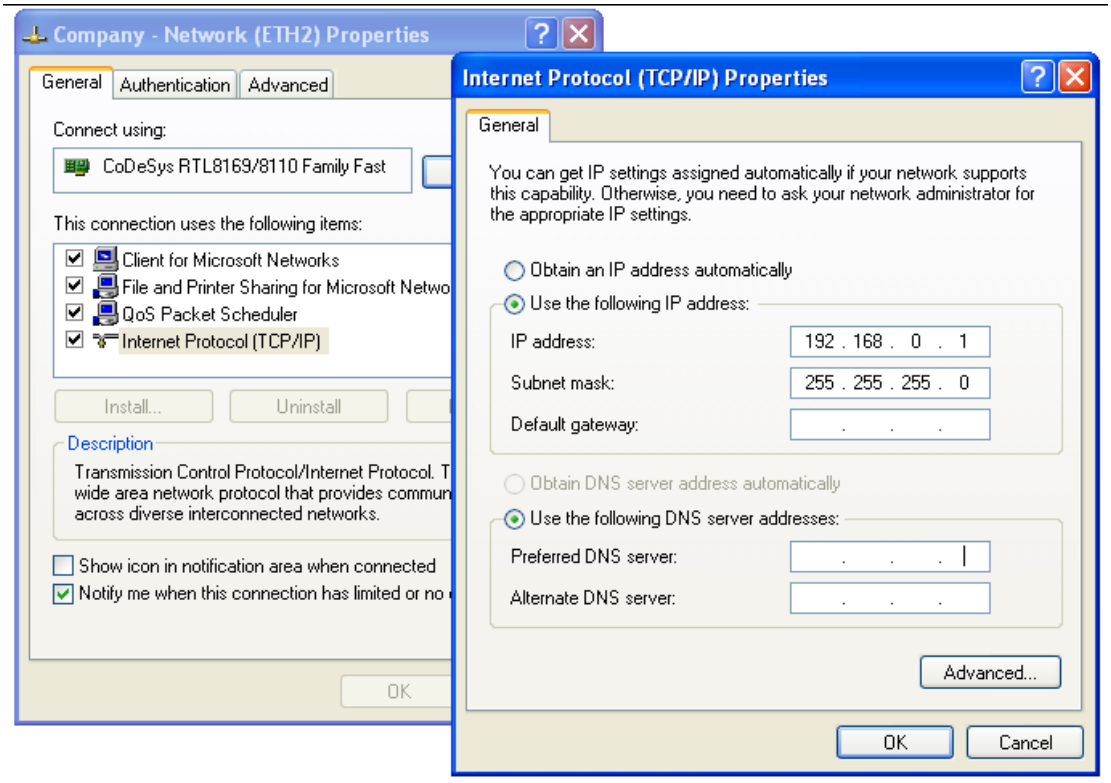
If you get the message: "NO COMMUNICATION WITH FLEXIUM RTS", please check the following:



- In *Windows Control Panel Network* connections:

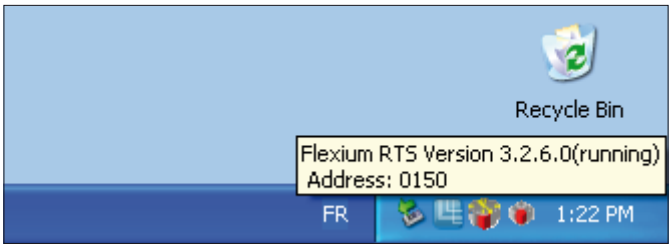
  - *ETH 1 Properties* : All protocols should be disabled
  - *ETH 2 Properties* : Internet Protocol should be set at a fixed address. By default this address is 192.168.0.1
  - *ETH3* : For the time being this port should be disabled.





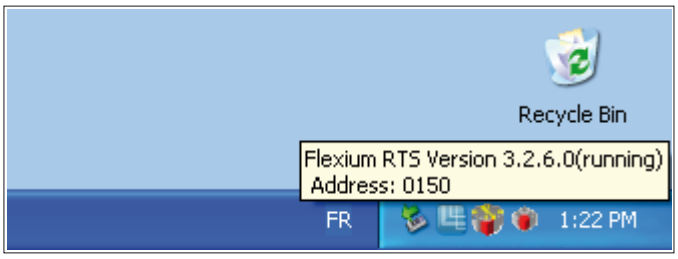
- Make sure the *RTS icon* in the system tray is red. Moving the mouse on it you should get the message *'Running'*. Please note the address.

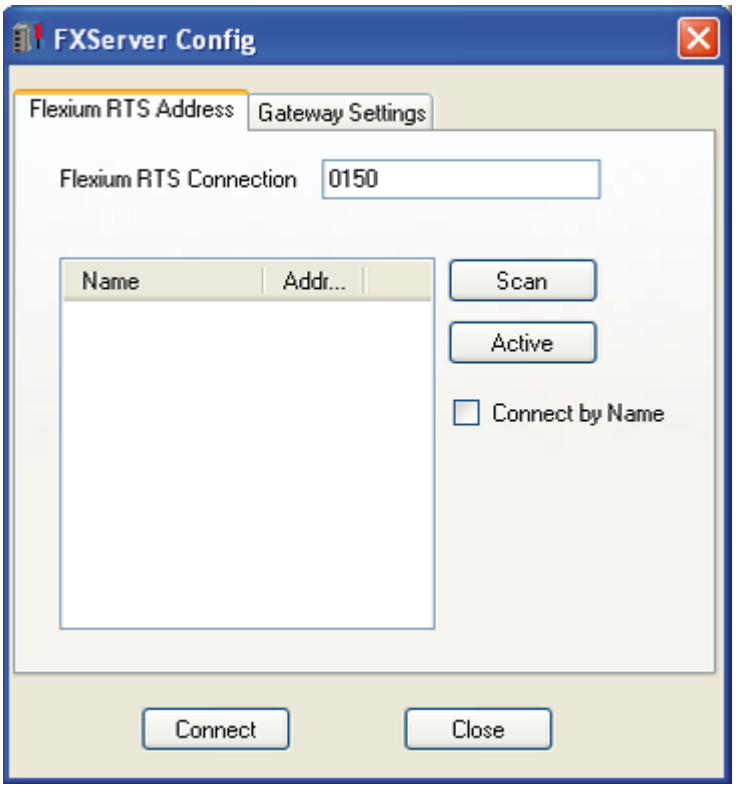
  The last two digits should be (in hexadecimal) the same as the last group of address of ETH2.
  e.g. ETH2 : *10.209.106.80* => *80* converted in Hex is 50 the RTS address should end with *50 (0050, 0150, …).*



If the RTS is not running, you can start it by a right click on the icon and selecting the corresponding menu.

- Check the *FX Server Icon* in the system tray, by moving the mouse on it.
  It should show the same address as the one for the RTS. If this is not the case please right click and select:





- **Settings**
- Press ***Scan*** on the window that just opened (*FX Server Config*)
- Select the right address when the scan is complete
- Press ***Active*** , then ***Connect*** and ***Close***.

It is also possible to get access to *FX Server Config* from the first page of Flexium HMI. To do so, be sure to click on the HMI window (to get the focus) before pressing Ctrl + S.

At this stage you are now ready to create a first application.

2.7 Flexium Tools

2.7.1 Installation of the software Flexium Tool and Flexium HMI

Flexium Tools is already installed on the FS152i.
You can also install it, as well as the HMI, on any commissioning PC.

Flexium Suite can be downloaded from:

- ***http://flexium.num.com***
- user name = ***'customer'***
- password = ***'flexium4u'***.

Unless instructed to do so, do not download any other version than Public version.
After downloading, extract the file obtained in any temporary directory and launch 'Flexium Suite'.

To install the software open the program file *"FlexiumSuite.exe"*.
After start-up you'll see the following window:

To install the desired software (*Flexium Tools or Flexium HMI*) click on the corresponding item.

According to options and versions additional menus can be shown.

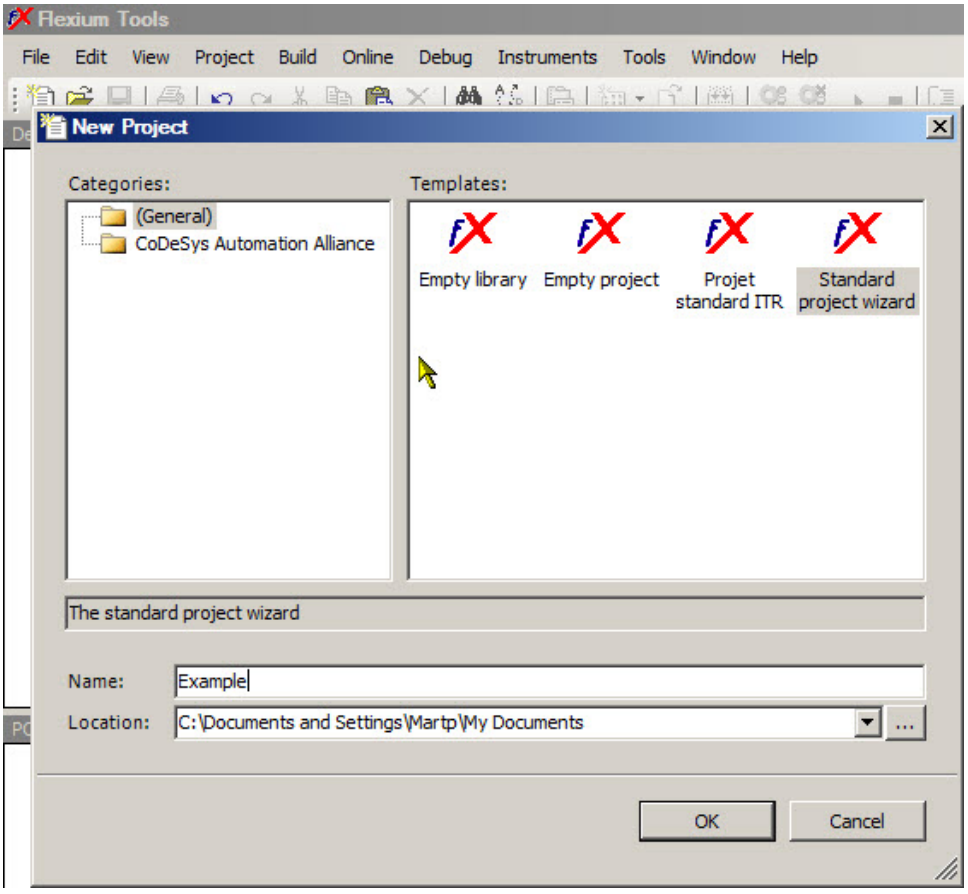
2.7.2 Create a test project

Flexium Tools can be executed on the FS152i itself or on a separate comissioning PC.
To be able to go *"On Line"* with the Flexium System, it is neccessary to connect one ETH port of the commissioning PC with ETH2 of the FS152i or DPLC Unit.

The *IP address* of the commissioning PC shoud be in the same range as the address of ETH2 *(default 192.168.0.x*).

- **Open Flexium Tools**

  The initial password is '*fxtools*'.
  You can change this password by the menu *Tools - User account*.



- *Select*
- *Files*:
- *New Project:*
- *Standard Project Wizard*
- Enter a *project name* and validate by OK.
- Press *Next* and Validate all choice until *'Source Code Download'*, then press *'Finish'*

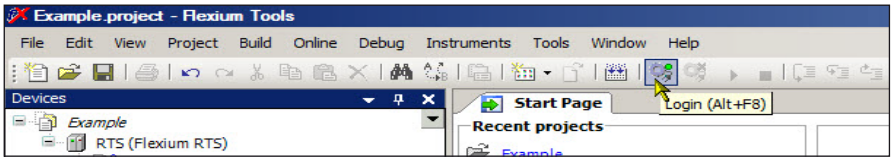
In the menu *'View'*, select *'Devices'*. If this window remains closed you can reveal it by clicking on the Tab *'Devices'* on the left side of the screen.

You have created an empty project for a single NCK at address 0.
Make sure the NCK rotary switch is set at the same address.

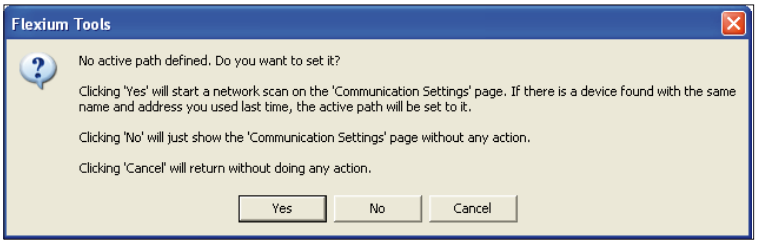
2.7.3 Going On Line

Once this project is created, it is necessary to compile and store it in the PLC.
This is done automatically by going *On Line*.

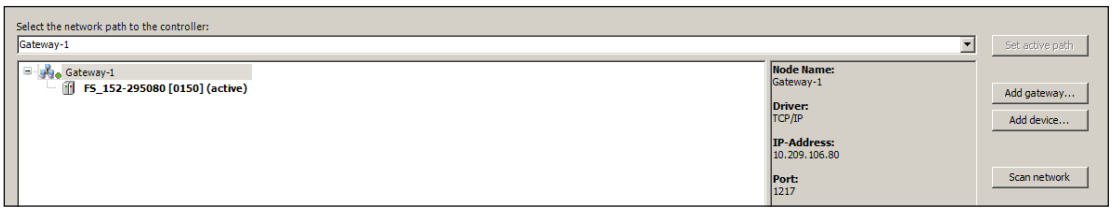
- Press Alt + F8 or the *On Line* button in the menu bar.



- You should get the following message:



- Press *'YES'*. This will search for a communication port and open the associated page.
  If there is no connection marked *'Active'*,
  - press *'SCAN NETWORK'* select the device with the same address as the RTS and activate it (*SET ACTIVE PATH*):



- Once this is done, the communication settings are recorded and you can go *On-Line* easily.
- At this stage once you get the request to transfer the project source, you can answer *NO*.

However, when your project will be finished it is ABSOLUTELY NECESSARY to transfer the source files. Without source files on the target further interventions on the machine can be very difficult.

In order to protect a specific know-how, it is possible to create in a particular library some programs which will not be readable by a third party.

- After the transfer, the PLC will be in stop mode, press F5 to Run it.
- Once the PLC is in run, launch *FLEXIUM HMI*.
  You will be requested to load the licence file.

2.7.3.1 Load the licence file

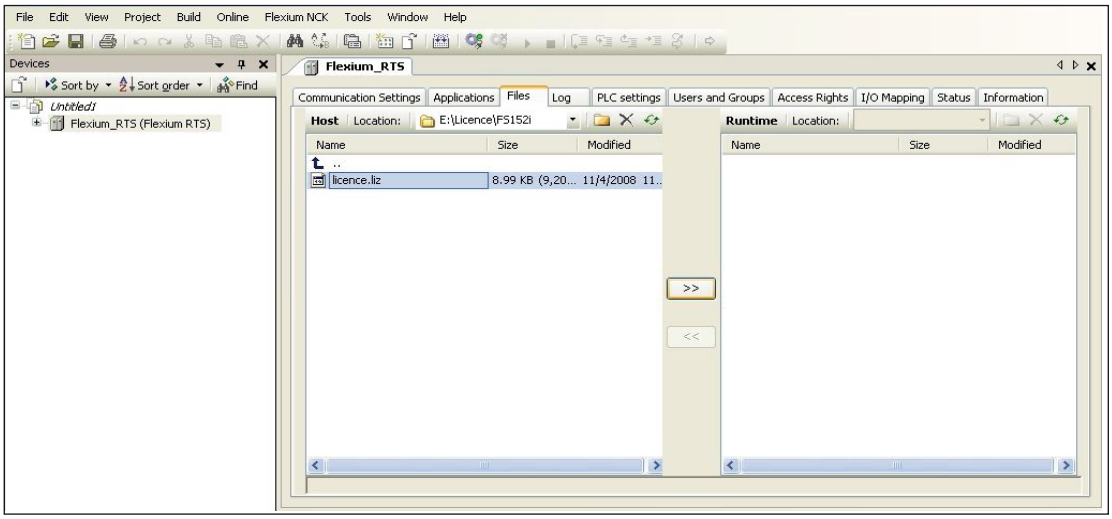
The licence file is delivered on an *USB stick* with the NCK. The file included should be stored on the FS152i (or DPLC) to activate the system.

This file can be stored via Flexium Tools or via an utilty on the USB Stick itself.
For DPLC using the USB stick is not possible.

Using USB Stick

- Insert the stick on the USB port of the panel
- Launch *'Flexium Licence Loader'* from this stick
- Press *'SCAN NETWORK'*
- Select the *right system* (in most cases there is only one)
- Select *'LOAD LICENCE'*
- Choose the *right licence file*
- Press *'LOAD'*
- Restart the system.

Using Flexium Tools



- Click on folder *'Files'* in the menu *'Flexium_RTS'*.

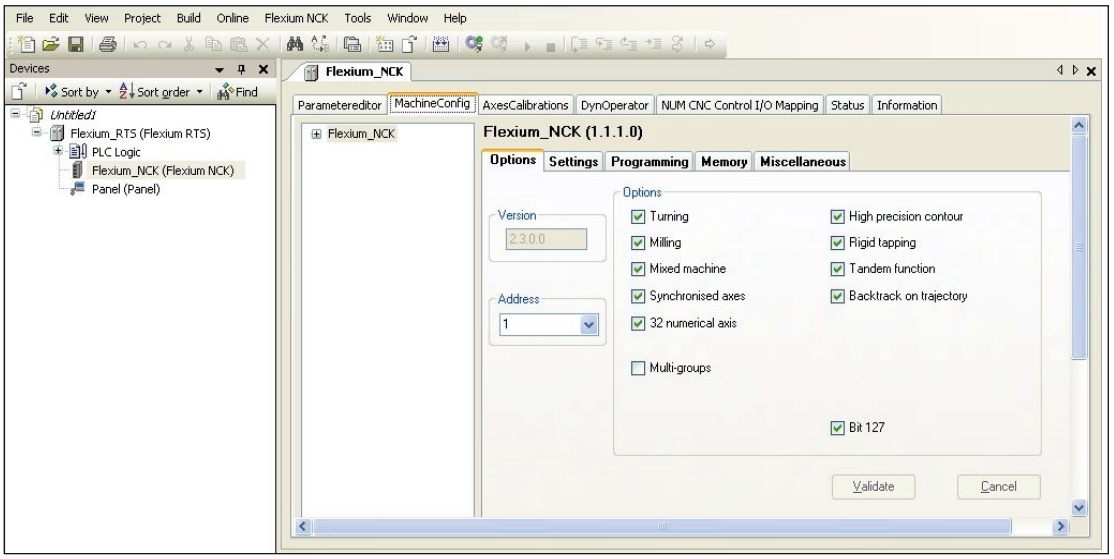
- Select the file *'licence.liz'* on your host drive and transfer it with the *'>>'* button to the *Runtime window*. (The path will be set automatically)

If you use a licence downloaded from the licence server on the internet or the licence file from the USB stick delivered with your NCK, the name of the file must be changed to *"licence.liz"*.

- After restarting the system, it will run with a minimum application.
  Use this moment to get familiar with the different contexts of Flexium HMI.
  The password for the *Service contex*t is ***'oem'***.

2.7.3.2 NCK settings

Once familiar with HMI and the first use of Flexium Tools, you are now ready for a more advanced application programming including I/Os and motors.

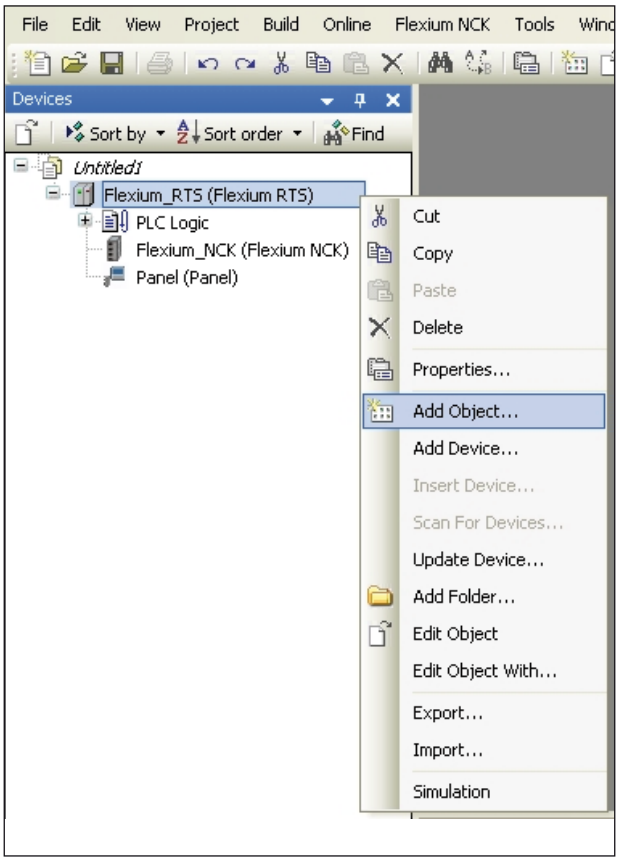


- Double click the *'Flexium_NCK (Flexium NCK)'* icon
- Choose *'MachineConfig'*
- Enter the Flexium NCK's address in the field 'Address'. The Flexium NCK address is given by the position of the rotary switch in front of the NCK unit.
- Press *'Validate'*.

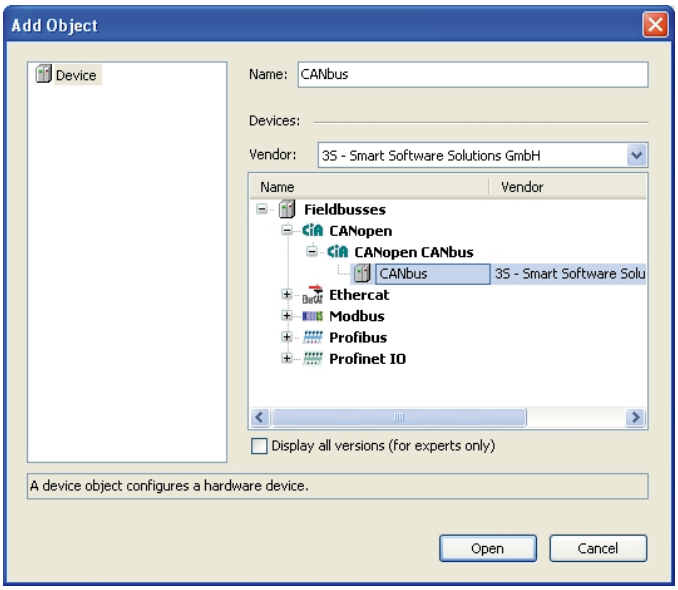
To set the NCK parameters please consult the Flexium Parameters manual M00021.

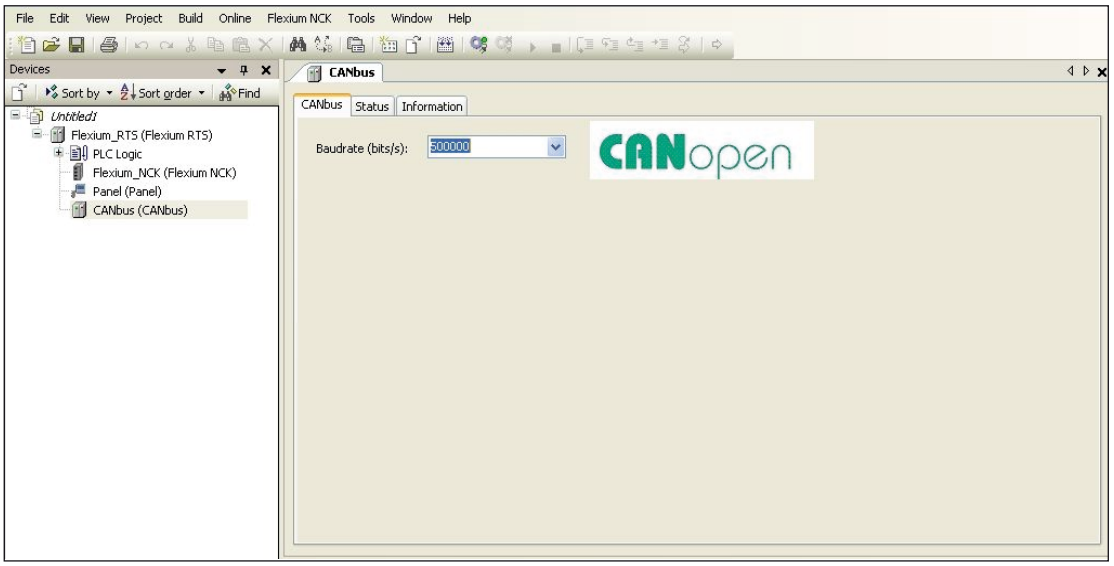
2.7.3.3 Add CANbus

- Right click on *'Flexium_RTS(Flexium RTS)'* and then *'Add Object'*



- Select *'CANbus'* and then press *"open"*



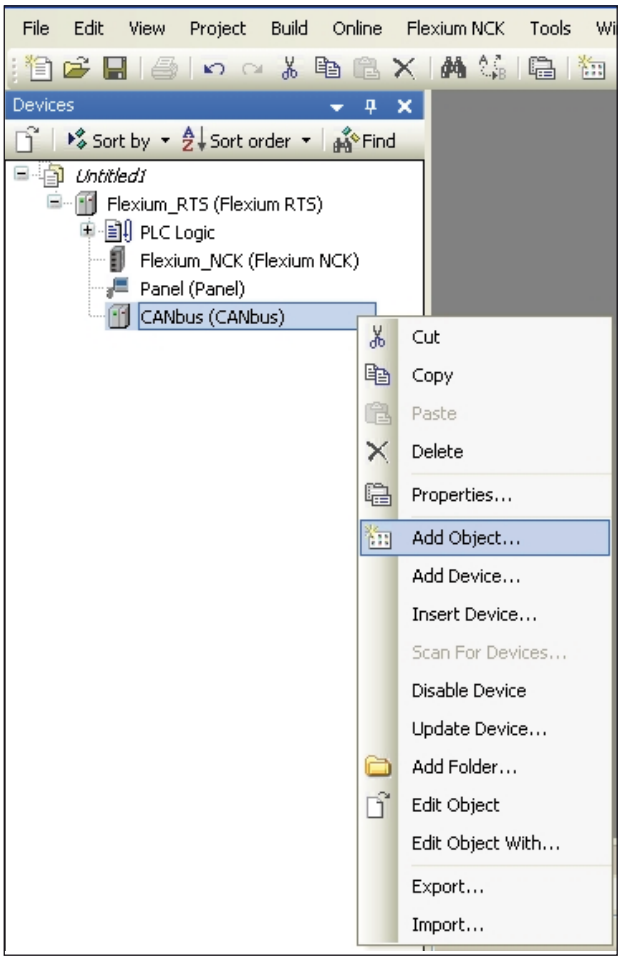


- Adjust the Baudrate for the CANbus relevant to the CAN assignment.

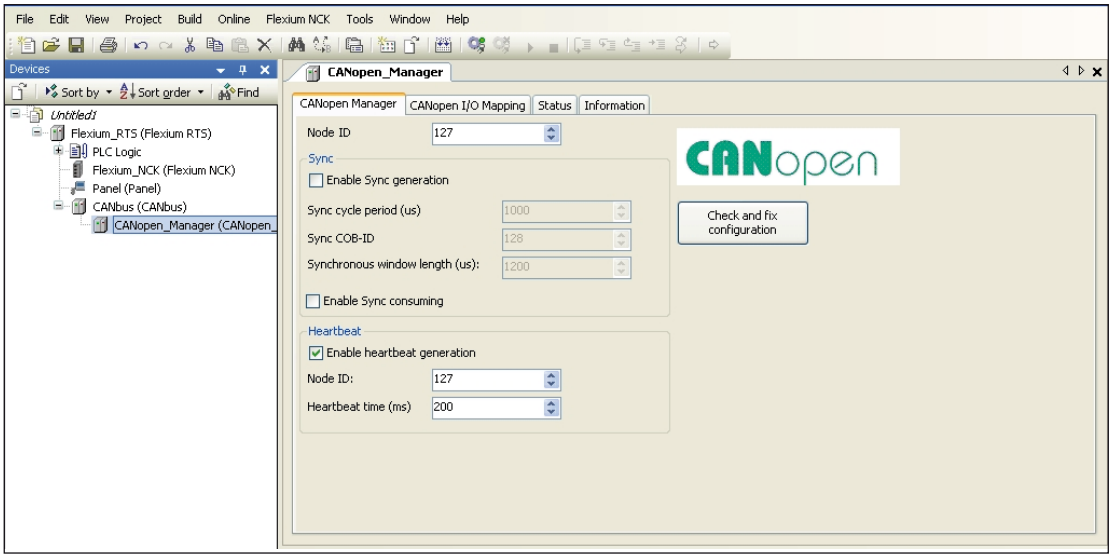
If a MP04 is used, only 125000 or 500000 bits/s are possible.
For more detailed information please refer to CANopen manual M00012.

2.7.3.4 Add CAN Master

- Right click on *'CANbus(CANbus)'* and then *'Add Object'*

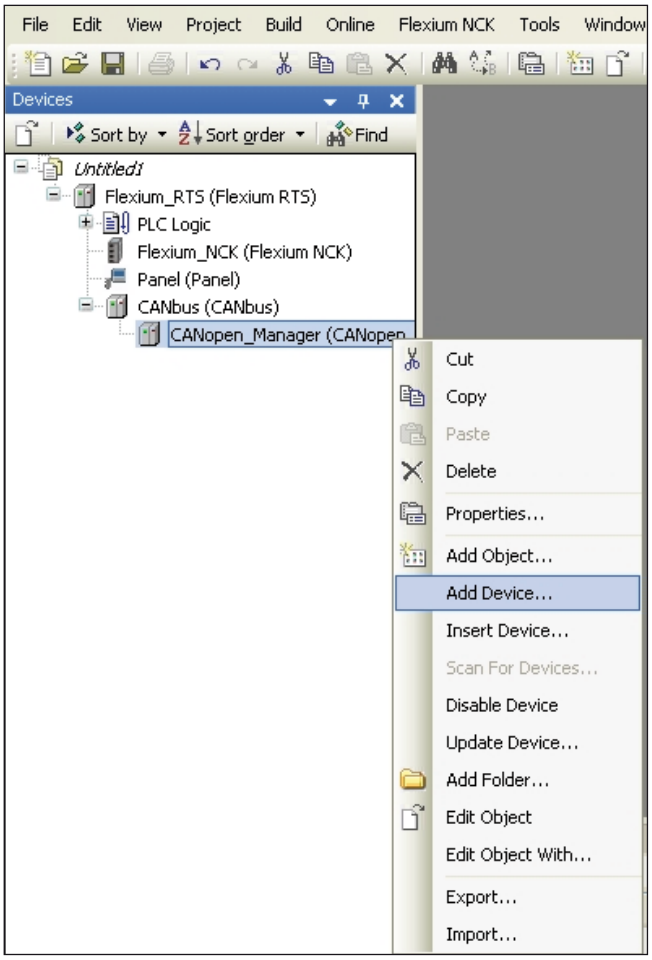


- Ensure *'enable heartbeat generation'* is active (Note Node ID and time)

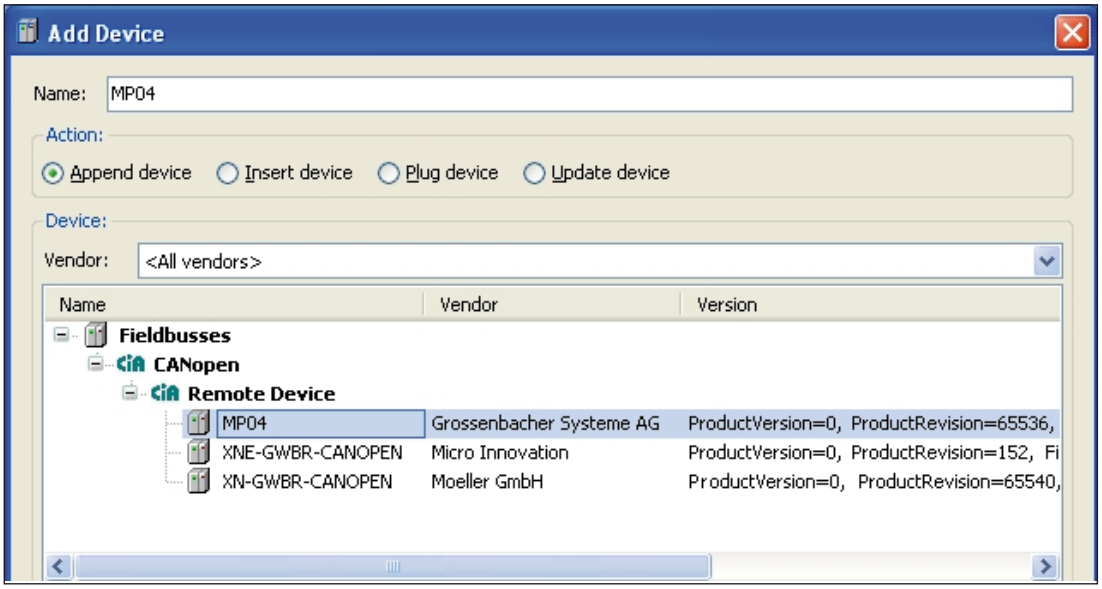


2.7.3.5 Add MP04 to CAN Master

- Right click on *'CANopen Manager(CANopen Manager)'* and then *'Add Device'*

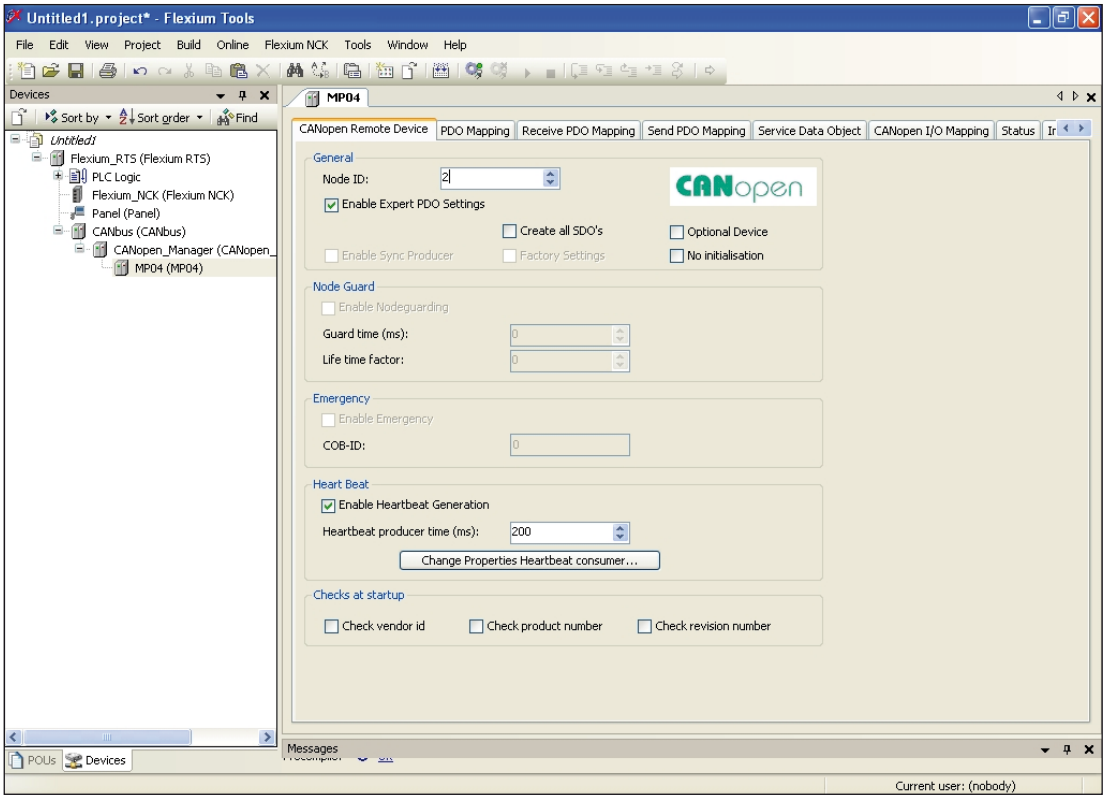


- Now select the MP04 to be connected to the CANbus and press *'Add device'*

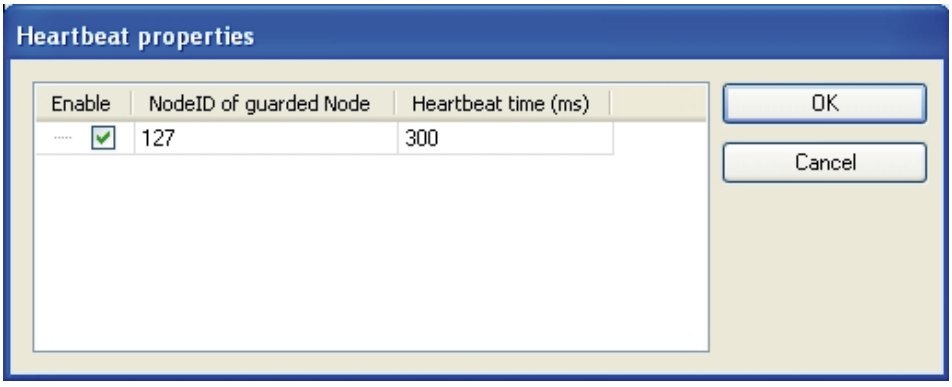


2.7.3.6 MP04 settings

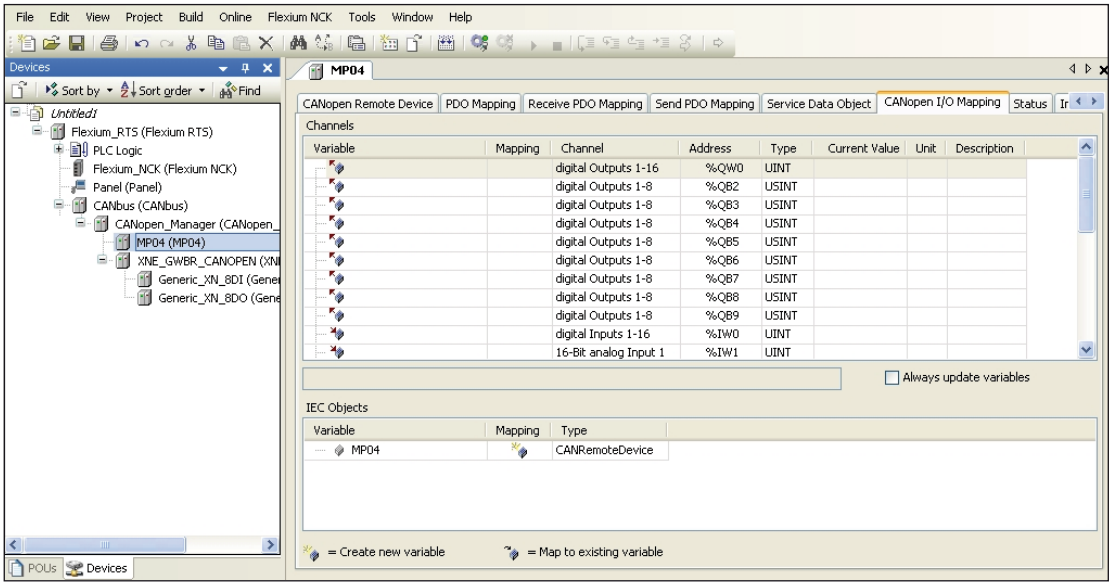
- Double click the icon *'MP04 (MP04)'*
- Enter *'Node ID'* of your MP04 (CAN address of the MP04)
- Delete the three checkboxes in the section *'Checks at startup'* (Check vendor ID, Check product number and Check revision number)
- Activate *'Enable Heartbeat Generation'*
  - Enter *'Heartbeat producer time'* (eg. 200ms)
- Press button *'Change Properties Heartbeat consumer…'*



- Compare the *'NodeID of guarded Node' with the 'Node ID'* of the CANopen Manager. (they must be the same)



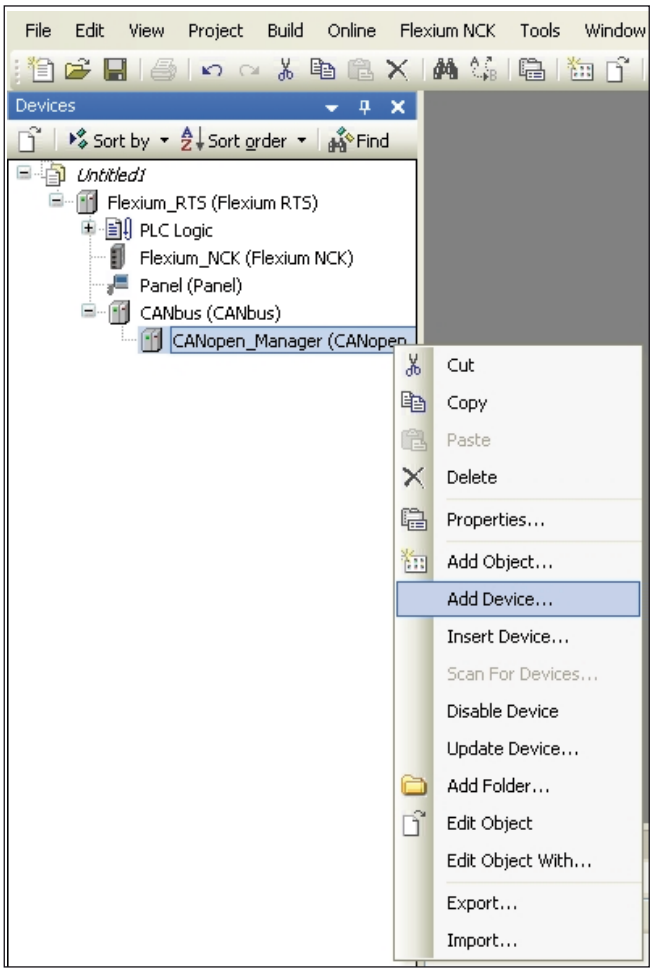
- The several input/output addresses of the MP04 are displayed in the folder *'CANopen I/O Mapping'*
- if needed you can enter variable names for the IOs by double clicking the *'Variable'* field or you can change the address of the IOs by double clicking *'Address'* field.



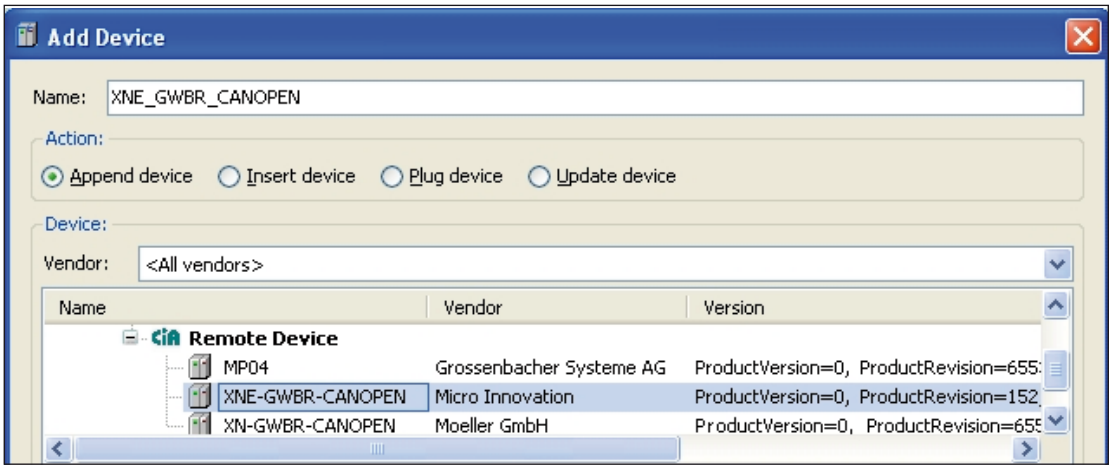
For more detailed information please refer to CANopen manual M00012.

2.7.3.7 Add NUM XI/ON Gateway to CAN Master

- Right click the icon *'CANopen Manager(CANopen Manager)'* and then choose *'Add Device'* .

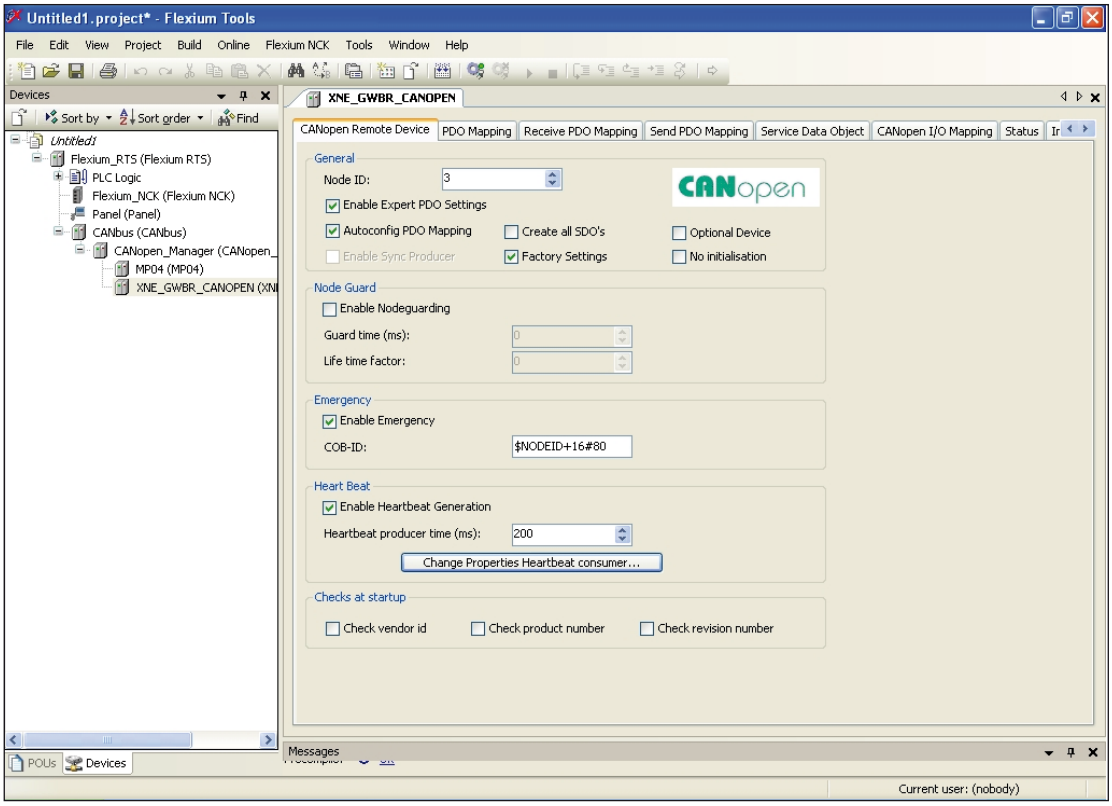


- Now select the Gateway you are using to connected to the CANbus and click on *'Add device'* .

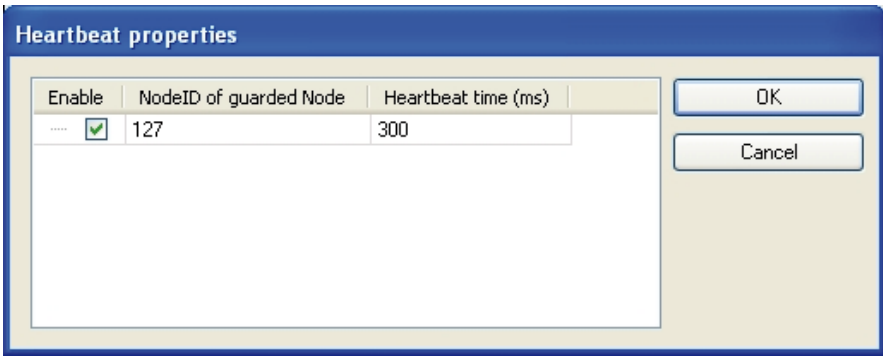


2.7.3.8 NUM XI/ON Gateway settings

- Double click the icon *'XNE_GWBR_CANOPEN(XNE_GWBR_CANOPEN)'*
- Enter the *'Node ID'* of your XI/ON Gateway (CAN address of the XI/ON Gateway)
- Delete the three checkboxes in the section *'Checks at startup'*
  (Check vendor ID, Check product number and Check revision number)
- Checkbox *'Enable Heartbeat Generation'*
  - Enter *'Heartbeat producer time'* (eg. 200ms)
- Press button *'Change Properties Heartbeat consumer…'*

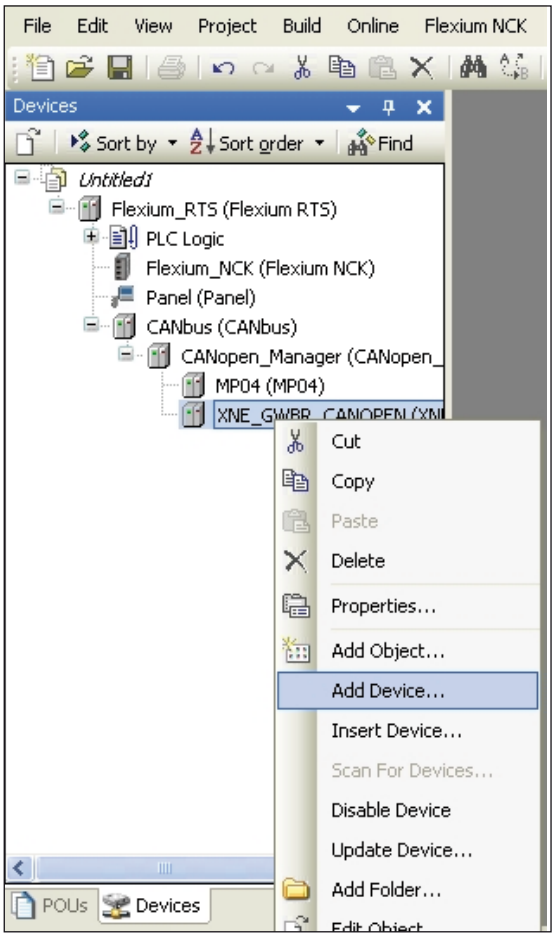


- Compare the *'NodeID of guarded Node'* with the *'Node ID'* of the CANopen Manager.
  (They must be the same)

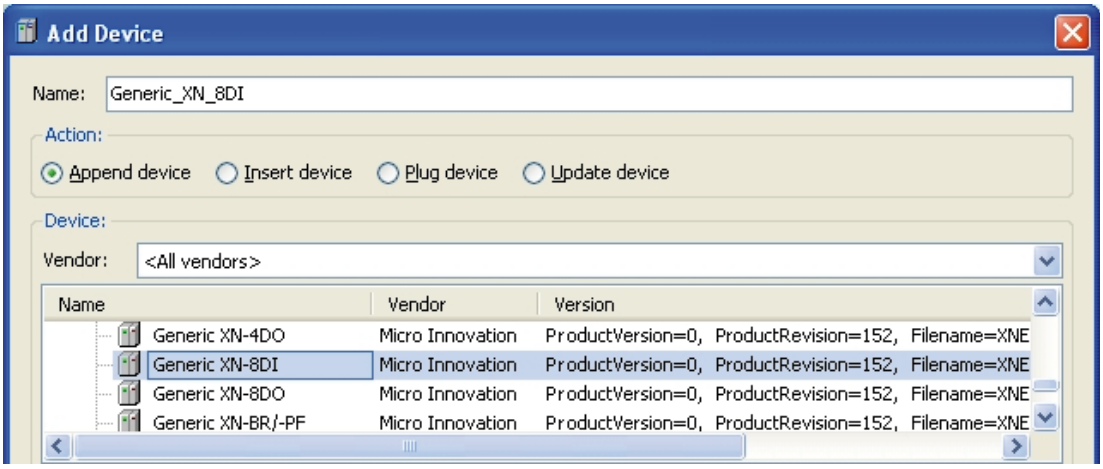


2.7.3.9 Add NUM XI/ON IO Module to Gateway

- Right click on *'Gateway'* and then *'Add Device'*

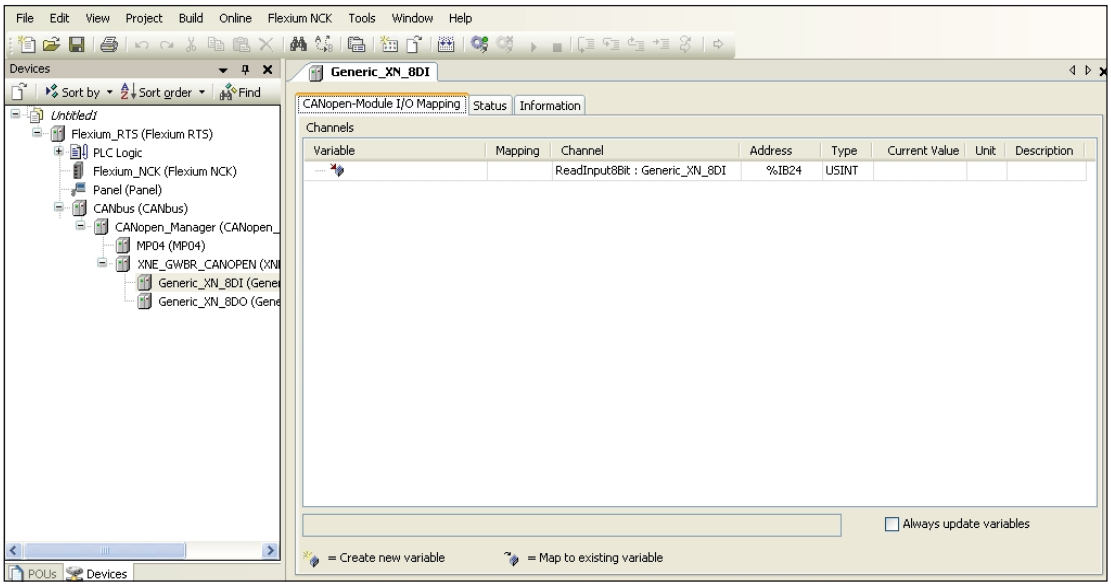


- Now select the IO-module you are using with this gateway and press *'Add device'*
- Add all IO-module in the physical order in which they are installed.



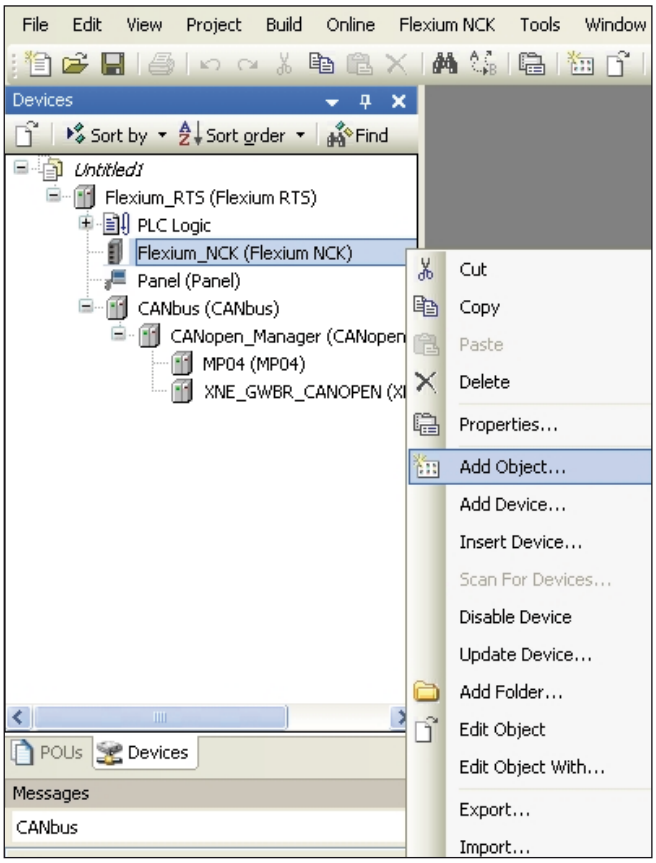
2.7.3.10 IO settings

- Double click the relevant IO slave icon.
- in the *'CANopen-Module I/O Mapping'* tabbed window the data of the IO-module will be displayed.
- if needed you can enter a variable name for the IOs of this block by double clicking the *'Variable'* field or change the address of the module by double clicking the *'Address'* field.

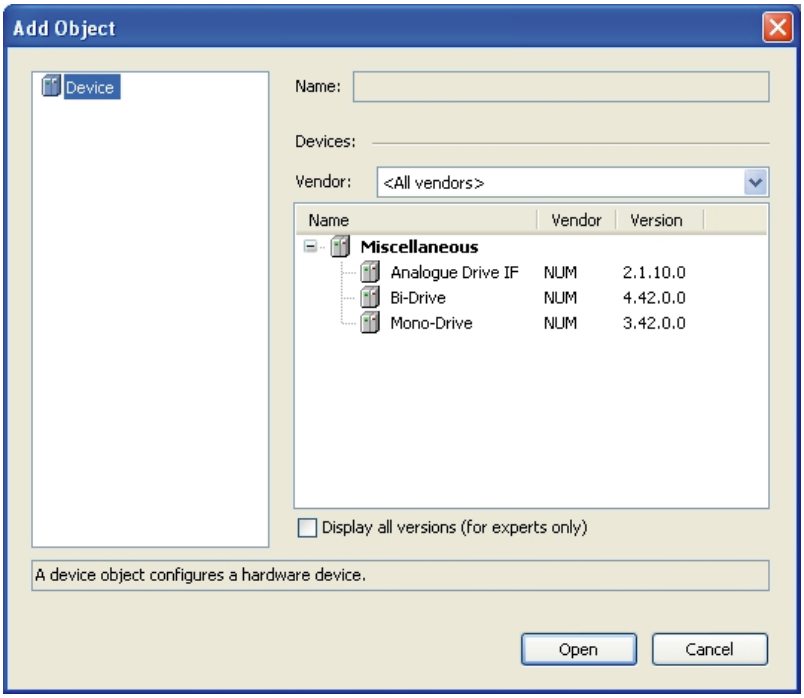


2.7.3.11 Add Drive to Flexium NCK

- Right click on *'Flexium_NCK(Flexium NCK)'* and then *'Add Object'*



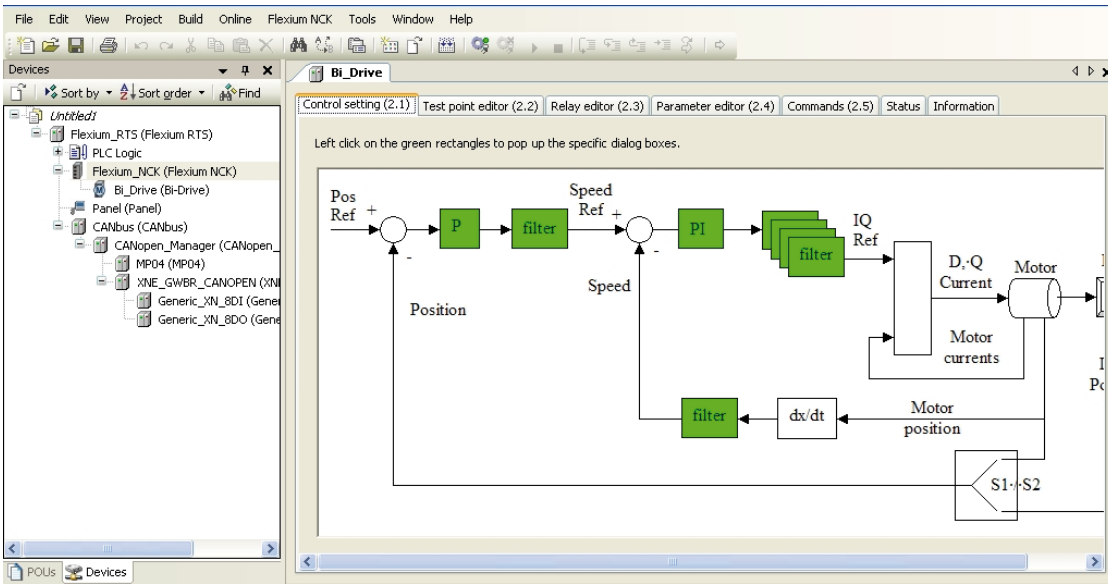
- Select the relevant drive and press *"open"*
- For all installed drives proceed the same way.



The version number can be different in your application.

2.7.3.12 Drive settings

- Double click the relevant *drive icon*.



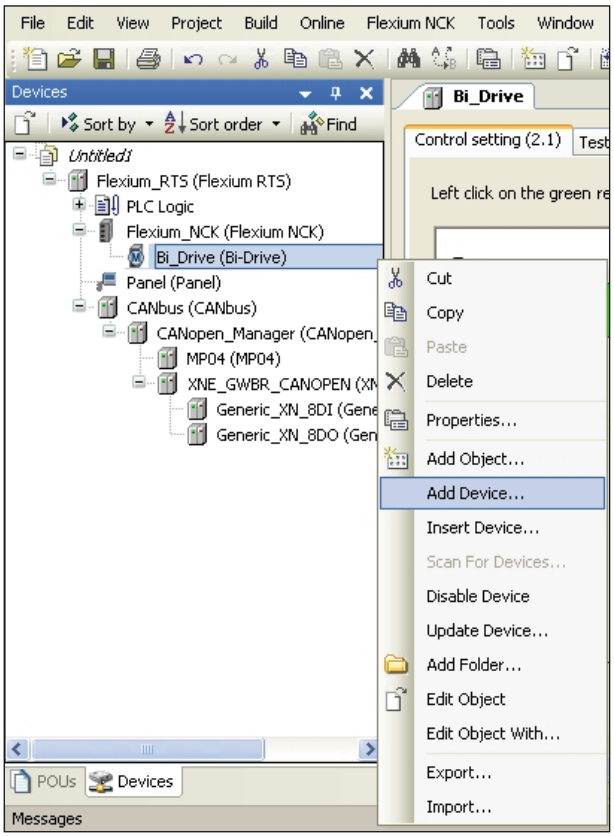
To adjust drive parameters please consult the corresponding Parameters manual:

- Analogue Drives: *'MDLA2 modular drive Installation and maintenance manual code 738x008'*.

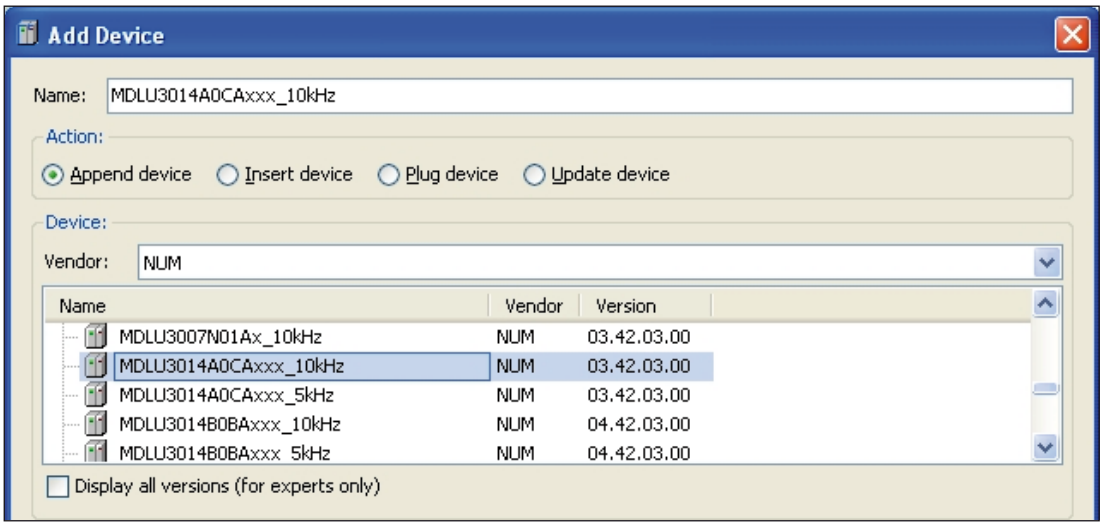
- Digital Drives: *'NUMDrive C Mono-Axis and Bi- Axes and NUM HP Drives: Parameters Manual code AMOMAN007'*.

2.7.3.13 Add Power Unit and Sensors to Drive

- Right click the *'Drive'* icon and then *'Add Device'*

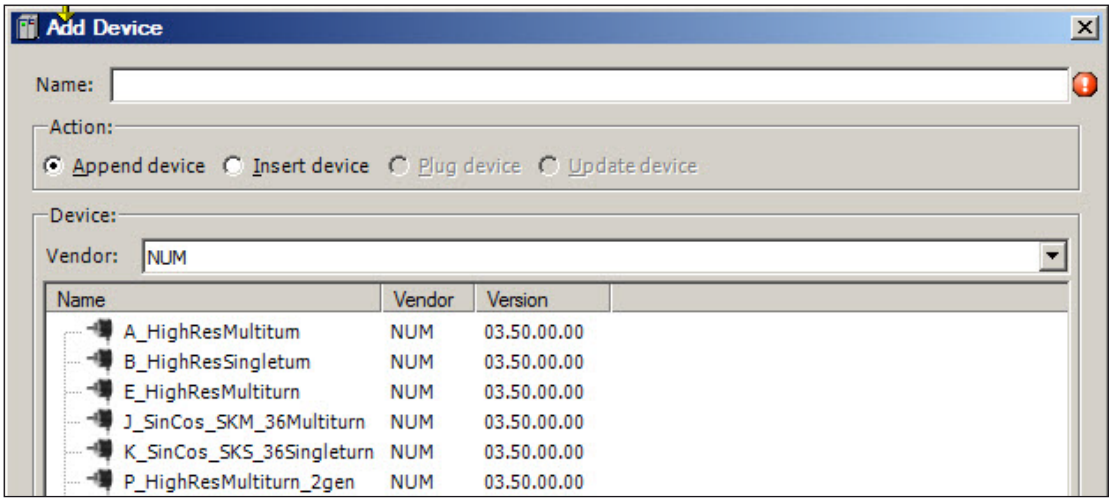


- Select the relevant power unit and then press *'Add Device'*.



All possible power units and sensors are displayed for selection.
After choosing the power unit and pressing *'Add Device'* only sensors will be displayed because only one power unit is allowed.

- Choose the relevant sensor and then press *'Add Device'*.
- For more than one sensor proceed the same way.

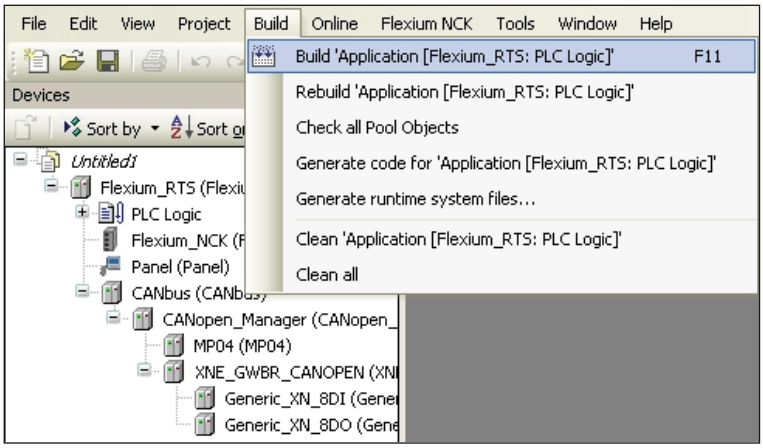


Maximum two sensors are possible for one axis.

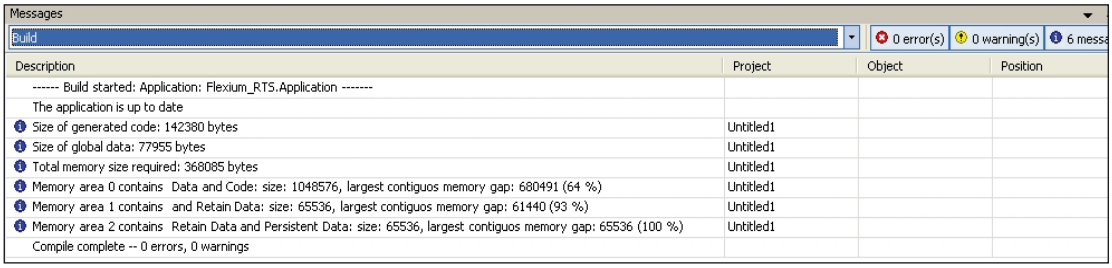
2.7.4 Download the application

2.7.4.1 Check your application

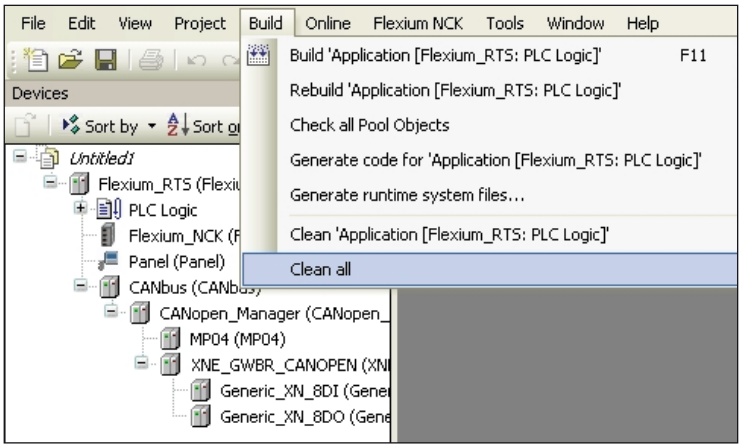
- Choose *'Build Application'* in the *'Build'* tabbed window.



- In the *'Messages'* window you can check if your application generated any faults, warnings or messages.



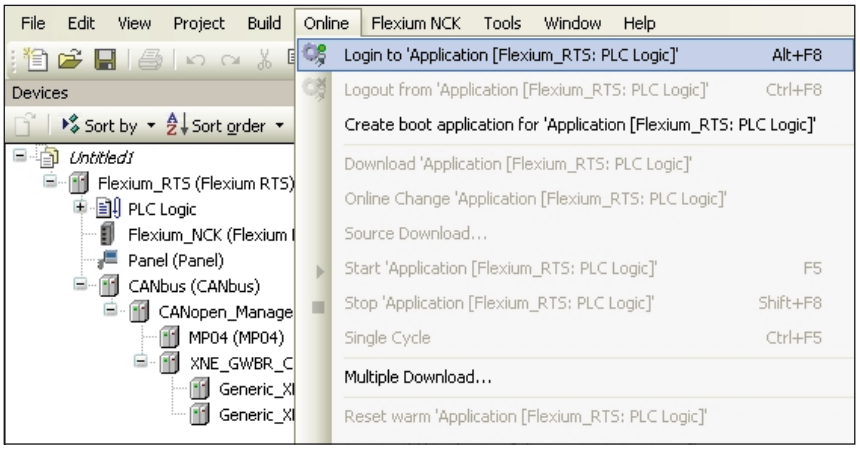
- To clean the compiled application choose *'Clean all'* in the folder *'Build'*



The machine keeps working but online changes are no longer possible.

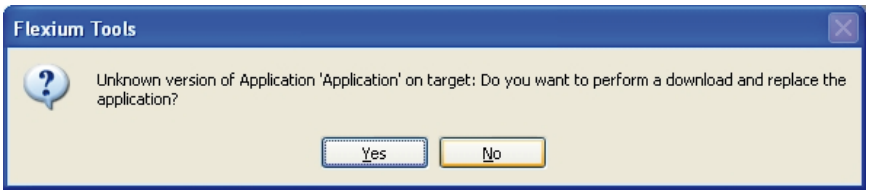
2.7.4.2 Login to application

- Choose the *'online'* tabbed window
- then choose *'login to Application'*.



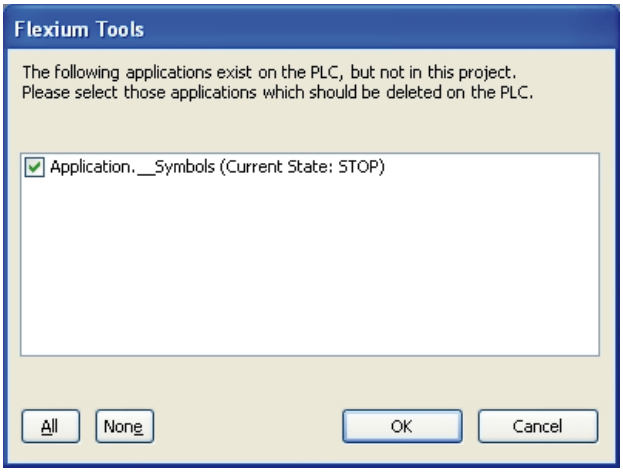
If your application does not match to the already downloaded application you will get the following message:

- By choosing *'Yes'* your application will be downloaded
- By choosing *'No'* your application will not be downloaded.

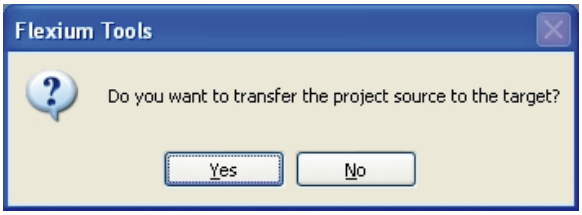


By pressing *'Yes'* the NCK will be stopped and will reboot after download.
Note: your machine will also stop.

- By choosing *'OK'* your symbols will be downloaded.



- Press YES if you want to transfer the project source code.

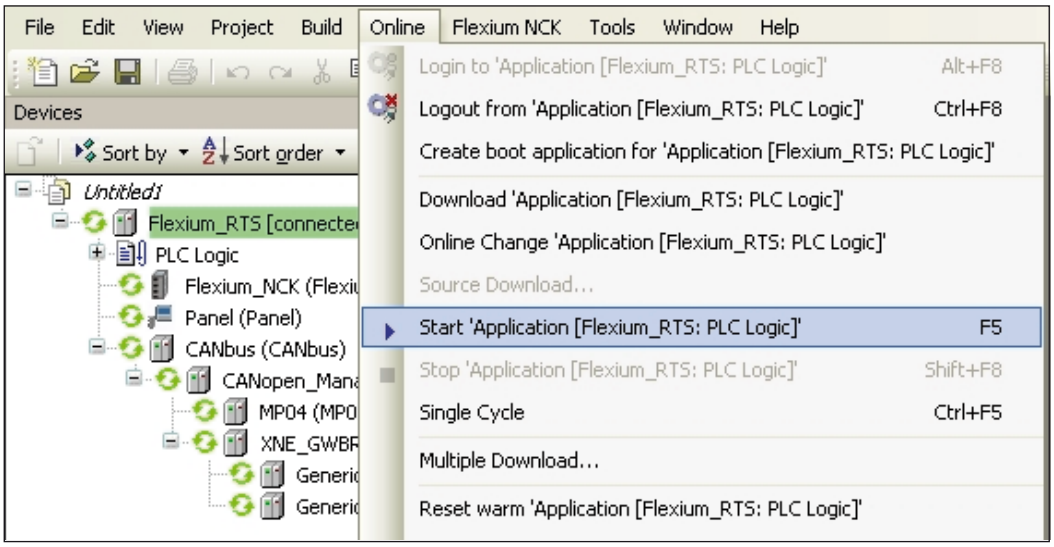


As transferring the source code can take quite a long time, it is not necessary to transfer it systematically.
(The source files remains on your commissioning PC).

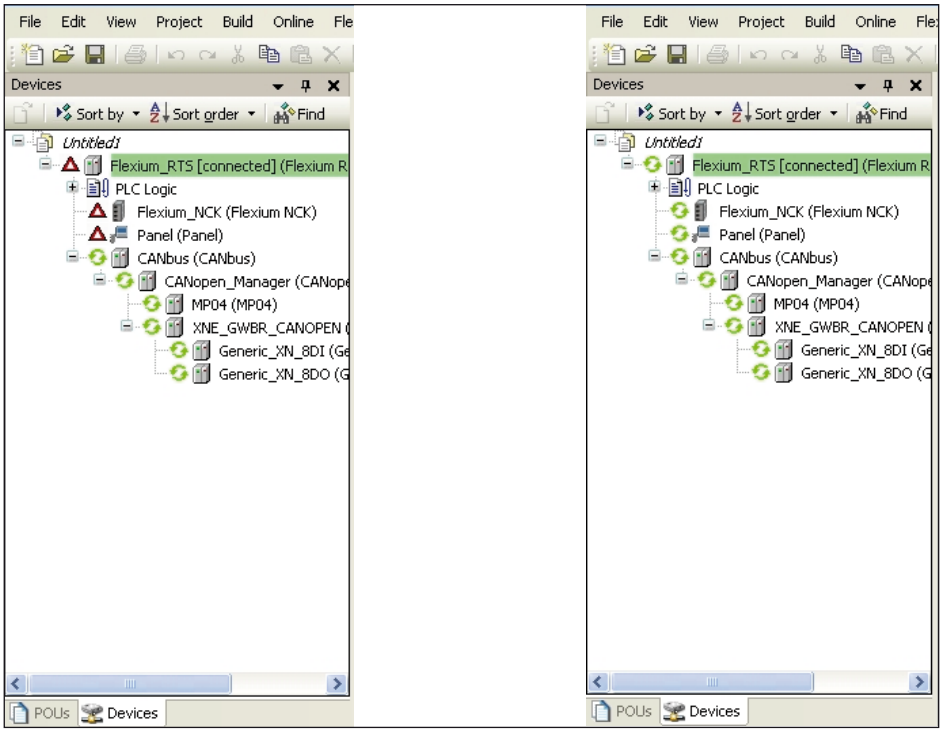
However it is absolutely mandatory to transfer the source code once the machine integration is complete and after each change. Without source file it might be impossible to intervene on the machine.

2.7.4.3 Starting an Application

- To start the PLC choose *'Start Application'* in the *'Online'* tabbed window.



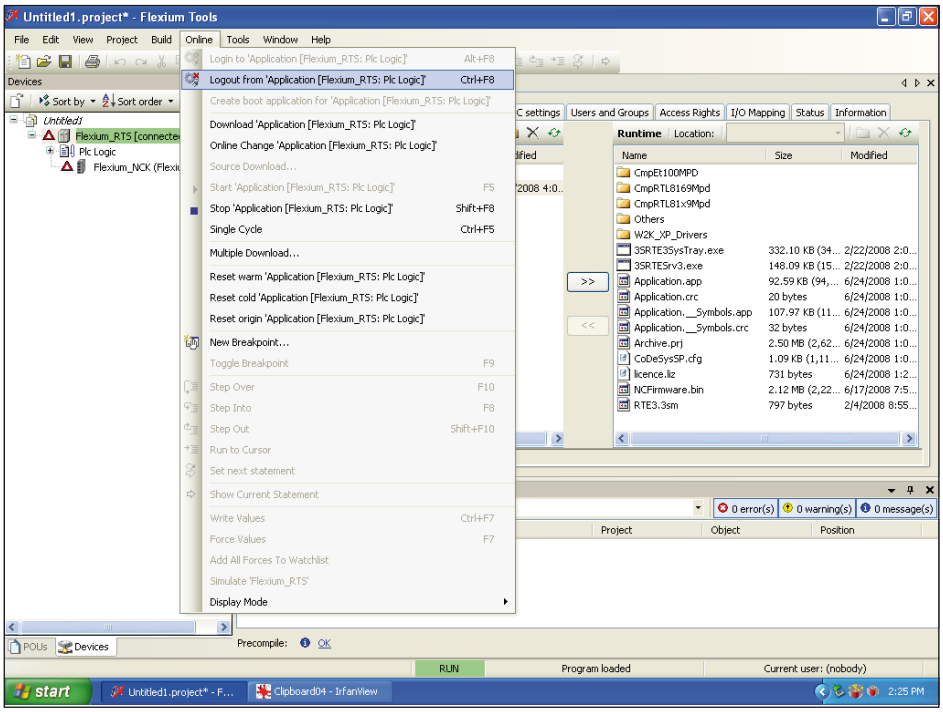
- At first the left hand picture like the one on the left will be displayed (some green circles and some red triangles)
- After a few seconds (depending on your application and CNC program) you should see the right hand picture.



A properly working application should show only green circles.

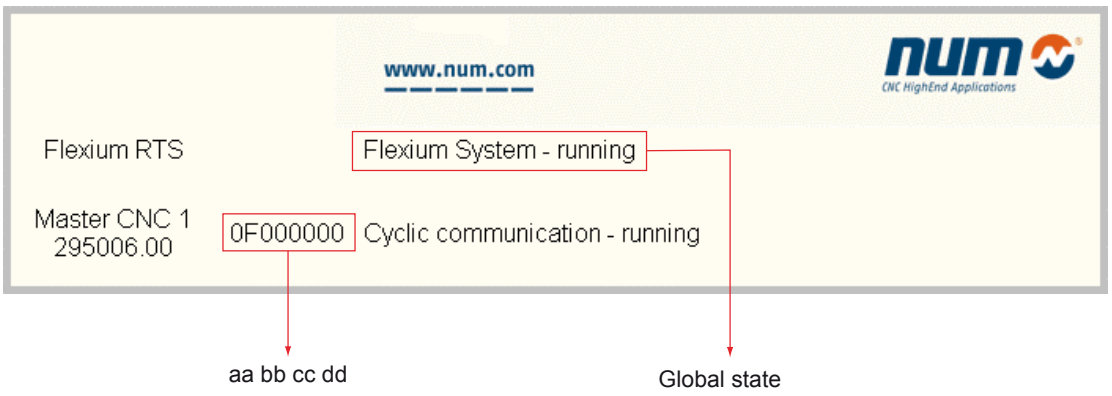
2.7.5 Close Flexium Tools

- Choose the *'Online'* tabbed window and then *'Logout from Application'*
- Close Flexium Tools.



2.8 System Start up messages

During the system start up, coded and plain text messages show the various initialization steps:



2.8.1 Description of the codes

- **aa** = Start up state
- **bb** = State message
- **cc** = Index (e.g. 00 for Memory zone 0 or 01 for drive 01 etc.)
- **dd** = Specific NCK message.

2.8.2 Start up state (aa)

Message No.	Message text
01	*Establishing connection with NCK* Communication between FS152i and NCK over Ethernet.
02	*Loading: Licence* Loading and checking the licence file.
03	*Loading: Parameters* Loading the parameters of the NCK and the drives.
04	*Loading: Memory area cc* cc corresponds to memory zone 0 (00), memory zone 1 (01) etc.
05	Loading: *Interaxis calibration*
06	Loading: *Axis calibration*
08	Loading: *Dynamic Operators in C*
09	*Initializing*
0A	*Loading: NCK software* Loading the NCK software.
0B	*Starting: NCK software* The NCK software is starting.
0C	*Reading: Checking file integrity* The NCK checks if the system was shut down correctly the last time and compares the data in the zone “User”.
0F	*Cyclic communication* Communication between NCK and PLC
1F	*End cyclic communication* End of communication between NCK and PLC
20	*Saving the memory areas* Saving the memory areas of the NCK.
21	*Reset NCK* The NCK is resetting.

2.8.3 State messages (bb)

Additional information for Start up state (aa)

Message No.	Message text
00	Running
01	Completed
02	File cannot be opened
03	Transmission cannot be started
04	Segment error
05	Transmission cannot be stopped
06	Communication error
07	Error
09	Cancelled by user
0F	Reading RTS licences
20	Boot loader and NCK software not compatible
21	NCK Device and NCK software not compatible

2.8.4 State messages (bb)

Additional information for Start up state number if aa=09

Message No.	Message text
00	Scan DISC NT
01	Parameter transfer
02	Drive parameter
14	Searching DISC NT drives
15	Drives are only present in the project
16	Comparing drive versions
17	Drive cc Power Unit not compatible cc: drive number.
18	Drive cc Kernel Board not compatible cc: drive number
19	Drive cc boot software release not compatible cc: drive number
1A	Drive cc drive software not compatible cc: drive number.
1B	Drive cc parameters are not updated cc: drive number.
28	Setting drive cc in Halt cc: drive number.
29	Writing drive cc parameters cc: drive number.
2A	Writing drive cc EEPROM cc: drive number.
2B	Setting drive cc in Run cc: drive number.
3C	Comparing NCK and drive parameters
3D	Error parameters drive cc (difference between V266 and P50) cc: drive number.

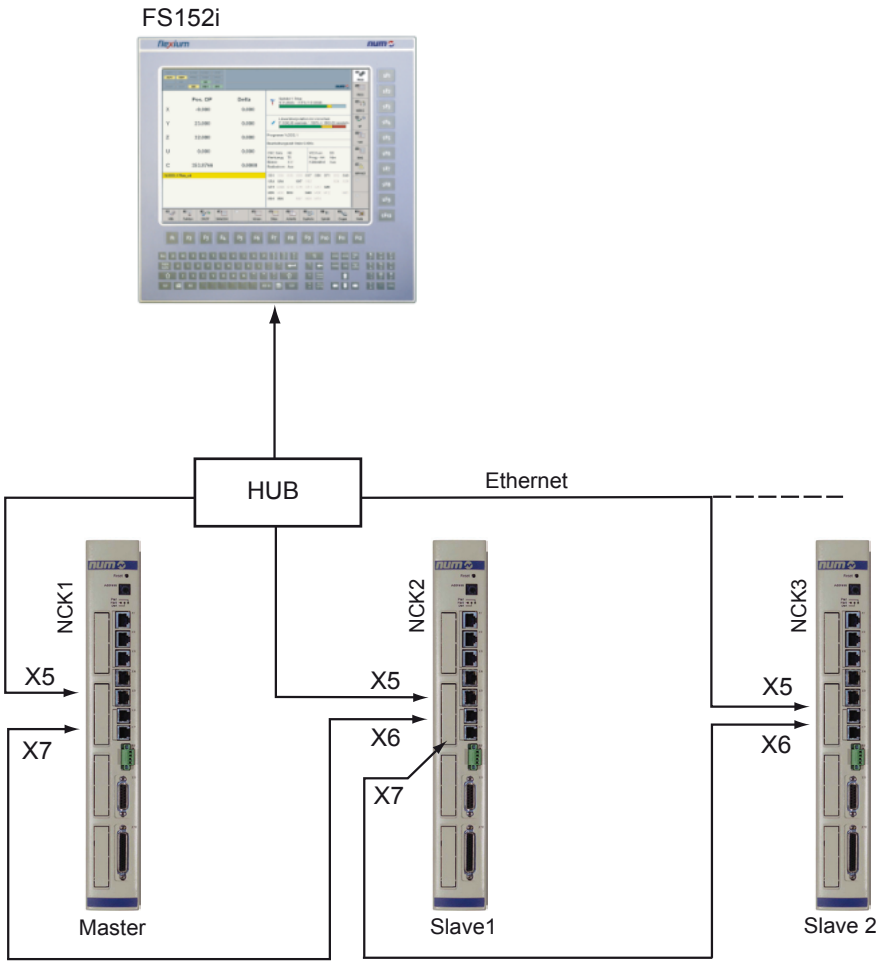
2.8.5 Error message - Troubleshooting

Error No.	Message text
01000000	**Establishing connection with NCK - running**
	If this message is displayed for a longer time: - check the address of the NCK - compare address switch on the NCK with the address in your project - check the connection of the Ethernet cable between FS152i and NCK (Eth1 <-> X5) - check the NCK power supply (Power LED should be on) - if everything above is ok press the Reset switch on the NCK.
02000000	**Loading: Licence - running**
	If this message is displayed for a longer time: - Ethernet not initialized correctly - Stop RTS - Switch off power of the NCK - Switch on power of the NCK - Start RTS.
02020000	**Loading: Licence - File cannot be opened**
	Check licence file (*c:\Program Files\NUM\Flexium RTS\licence.liz*)
0A210000	**Loading: NCK software - NCK Device and NCK software not compatible**
	The Firmware version of the NCK in your project must match with the version of the NCK Software (*c:\Program Files\NUM\Flexium RTS\NCFirmware.bin*). The version number consists of 4 digits. The HMI will start up when the first two digits are compatible. For an official version, all 4 digits must match.
0A020000	**Loading: NCK software - File cannot be opened**
	Check the file NCFirmware.bin (*c:\Program Files\NUM\Flexium RTS*).
0B000000	**Starting: NCK software - running**
	If this message is displayed for a longer time: - NCK did not start up. - Stop RTS - Switch off power of the NCK - Switch on power of the NCK - Start RTS.
0F060000	**Cyclic communication - communication error**
	For the master NCK: - check machine parameter P99 word N1 in the FlexiumTools (the value must be the same as the address switch on the master NCK). For a slave NCK: - check connection of the clock cable between the NCKs. - master connector X7 to slave connector X6, termination resistor on X7of the last slave. Please refer to Flexium Installation manual M00009.
093D00yy	**Initializing - Error parameters drive yy (difference between V266 and P50)**
	Check parameters of the drive yy and the NCK parameters in the project.

2.9 MULTI-NCK configuration

The Flexium system allows for controlling several CNCs with one single PLC (MULTI- NCK configuration) this can be used in particular in particular to expand the number of axes or channels.

A MULTI-NCK structure is implemented as follows:



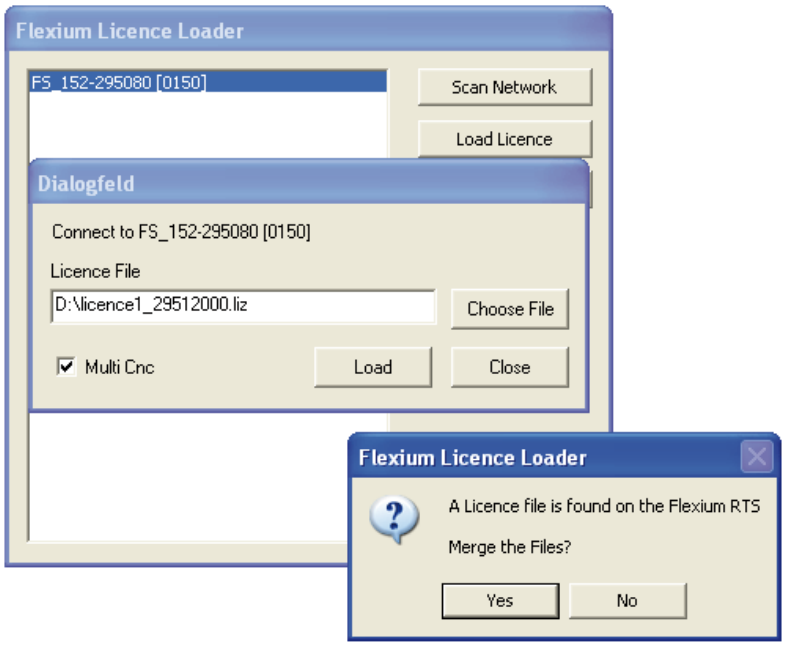
- Each NCK should have a different address and be declared in the project
- Parameter P99 defines the master CNC address
- The panel is connected to the X5 connector of all CNCs via a HUB
- The CNC clocks are interlinked:
  - X7 of the master CNC to X6 of first slave and so on (cable ref FXHC 081 530 for 20 cm)
  - X7 of the last slave should receive a termination connector (FXHC 081 540).

For X. connectors please refer to Flexium Installation manual M00009.

2.9.1 Licence management

Each NCK has its own license file. However as only one license file is allowed the different files need to be concatenated. This is easily done via the License Loader.

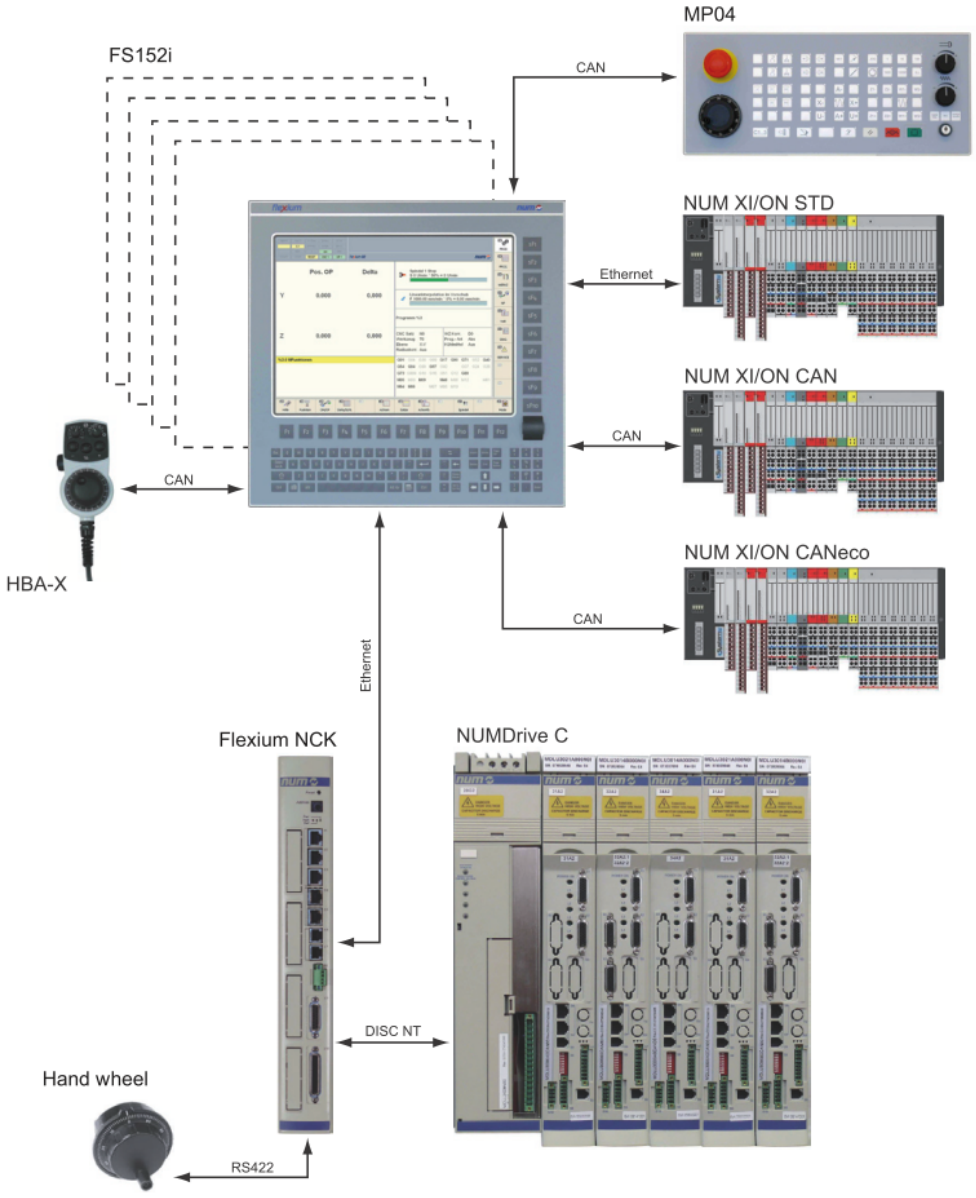
- Select Multi CNC configuration.
- Load the first license file (any one)
- When loading the following licenses files, the system will ask you if you want to merge the files just answer YES.



2.9.2 Multi Panel configuration

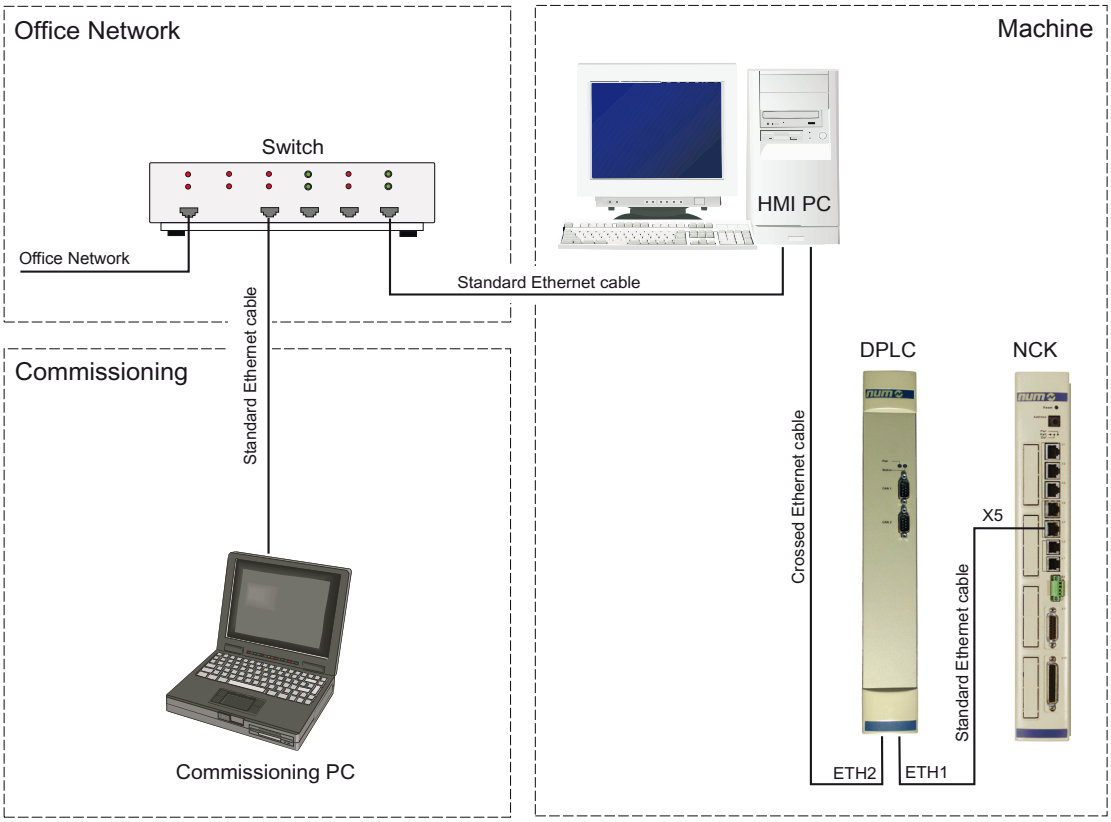
The system also proposes Multi Panel configuration.
In such a configuration each panel (FS152i) should be declared in the project and get a specific name, thus allowing different properties. One panel features the RTS and provides PLC functions.

Other panels are connected to the X5 port of the NCK via a HUB.
There is no other particular parameterization to implement.



2.10 Connecting a DPLC inside the Flexium system

The cabling of the DPLC, NCK, PC, Switch (optional) and Commissioning PC (optional) has to be as follow:



Please ensure that connections are made using the recommended Ethernet cables (standard/crossed).

For some applications, the computing power of a FS152i is not high enough, e.g. if a CAD/CAM system like NUMROTO should run on it. In this case an office PC can be used.
But for the real-time part (PLC) a dedicated hardware has to be utilized.

This hardware is called DPLC (Dedicated Programmable Logic Controller).

A DPLC contains interfaces for connecting a VGA monitor, keyboard and mouse (by PS2 or USB). In standard application, they are not used. However they can be utilized for special solutions.

Controlling and set-up of the DPLC will be done by the HMI PC.

2.10.1 Settings of the HMI PC

Before connecting the PC to the DPLC, the Ethernet settings have to be adapted.
The IP address of the network card on which the Ethernet cable to the DPLC is connected has to be set on a fixed value. The IP addresses of PC and DPLC must be in the same range.

For example, *the default IP address of the DPLC is 192.168.0.1 (ETH2)*. The IP address of the PC has to be set on *192.168.0.x* as x is a vacant number in the network (e.g. x=2).

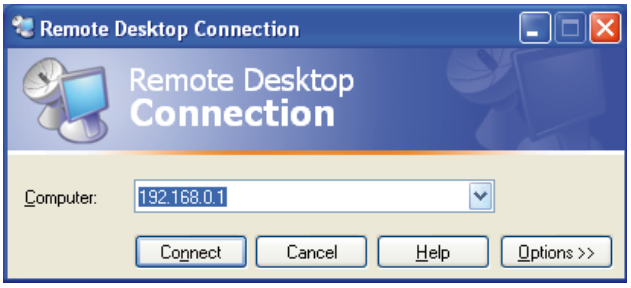
The IP address of the DPLC can be changed. In this case, the IP address of the PC must be set accordingly (same address range).

Now the PC is ready for connecting to the DPLC.
The settings of the DPLC will be done by remote desktop connection.

2.10.2 Remote Desktop Connection with Windows

The program *"Remote Desktop Connection"* is located under the following path:
- *START/Programs/Accessories/Communications/Remote Desktop Connection*

- After starting *"Remote Desktop Connection"*, the IP address of the DPLC has to be entered.



- Press *"Connect"*

- Login to the DPLC

The default settings of the DPLC are:

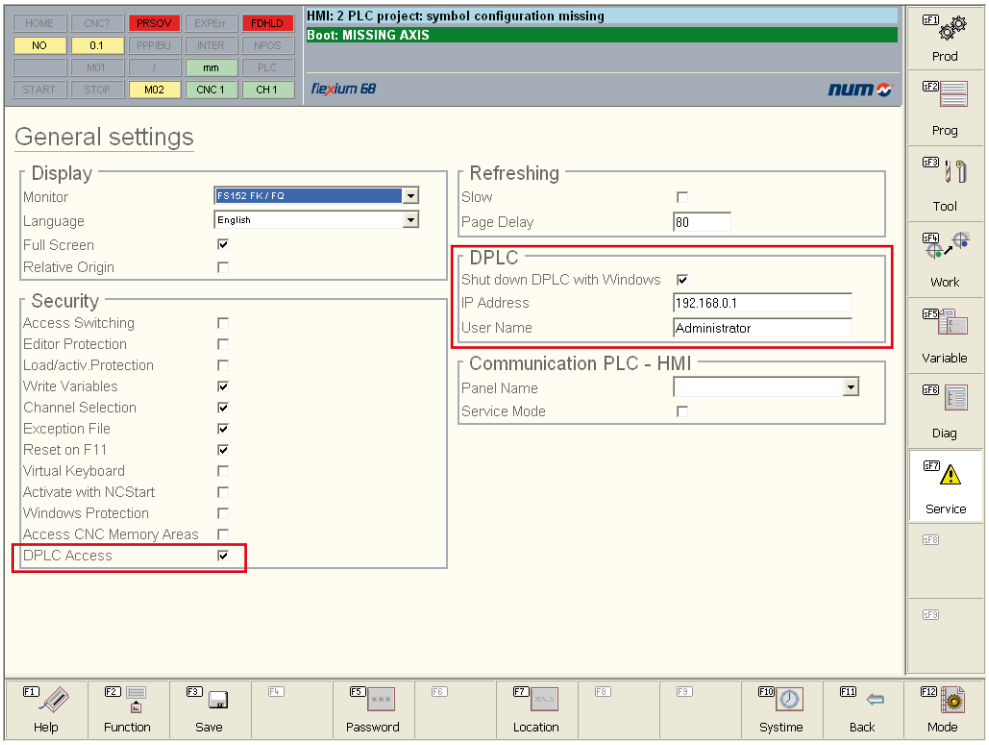
- User name: *Administrator*
- Password: *oem*

The settings of the DPLC are the same as for the FS152i.

2.10.2.1 Remote Desktop Connection with Flexium HMI

For an easier remote desktop access, it is recommended to activate the checkbox *"DPLC Access"* in the context *"Service, General settings"* of the Flexium HMI after the first startup.

The default password for the service context is: *oem*

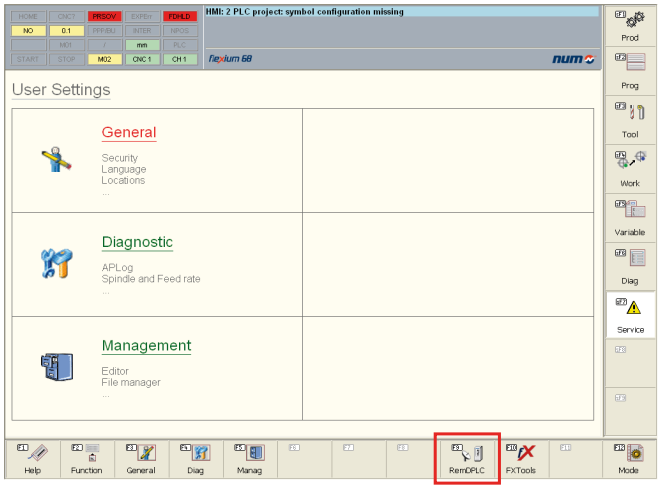


- Activate the checkbox *"Shut down DPLC with Windows"* if you want to shut down the DPLC when Windows of the HMI PC is shut down.

It is mandatory to shut down the DPLC properly to allow saving NCK data and remanent PLC variables as well as to prevent damage to Windows. This can be done by the function above, an UPS or by PLC IO's.

- Enter the IP address of the DPLC
- Enter the user name requested by the DPLC Windows.

Now it is possible to connect to the DPLC over the *"F9 RemDPLC"* button in the service context or the icon in the task list.



or

2.10.3 Settings of the Commissioning PC

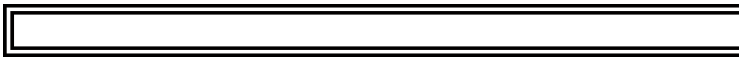
The commissioning PC has no access to DPLC by *"Remote Desktop Connection"*. It is only possible to get access to the DPLC by FlexiumTools. Gateway has to be installed on the HMI PC to establish a connection. Gateway is part of the Flexium Suite and will be installed automatically.

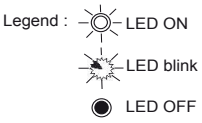
To get access to the DPLC proceed as described in chapter 2.10.

2.10.4 DPLC signalling

- Two status LEDs (power and status) are located on the front panel of the DPLC.
- A LED can be OFF, ON or Blinking.

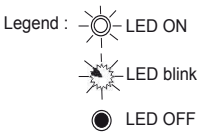
The table below shows the various possible states.

LED status	Meaning	Status timing
	LED OFF	
	LED ON	
	Blink 1	
	Blink 2	



2.10.4.1 Meaning of the LED states

Power (green)	Status (orange)	Signalling
		Power is switched off or the DPLC is shut down. After switching power off, the next power on should not occur before 20 sec.
		The DPLC is starting up.
		The **R**eal **T**ime **S**ystem (RTS) is running
		Bus cycle task of the application is running



As soon as the power LED is lit, the operating system has started.
It is mandatory to shut down the DPLC properly to allow saving NCK data and remanent PLC variables as well as to prevent damage to Windows.

Page deliberately empty.

Page deliberately empty.

Summary

Page deliberately empty.

3 Flexium PLC Exchange Zone

3.1 Informations from the NCK

3.1.1 General informations

Syntax: **Flexium_NCK.RCNC.General.** *'variable name'*

Variable name	Description
AxisRecall	General axis recall. Set or reset by a PLC request. In STOP mode and at the end of SEARCH, indicates that axis recall is enabled. Set by the CNC after *"W_ALLCNC.AxisRecall"* = 1. Reset by the CNC after *"W_ALLCNC.AxisRecall"* = 0.
BackForwardOnPath	Backward/forward movement on path. Set to indicate backward or forward CNC movement. Reset to cancel this state.
BlockSkip	Block skip status. Image of the "/" field of the CNC status window. If the bit is set, program blocks starting with "/" are ignored. The state of the bit can be changed by pressing the "/" key on the panel or by *"WCNC.Channel[x].BlockSkip"* = 1.
CncError	CNC error. Image of field "CN??" in the CNC status window. Indicates a machine error or a part program error. The machine error number is contained in *"RCNC.General.MachineError"*. Set by occurrence of a machine error (E30-E33, E36, E40-E71) or a part program error. Reset by the RESET key on the panel or *"W_ALLCNC.Reset"* = 1.
CncReady	CNC ready. Indicates that power can be applied to the machine. Set at power on and by the RESET key on the operator panel or *"W_ALLCNC.Reset"* =1. Reset by detection of an excessive following error on an axis, poor signal or encoder complementary channel fault detected on an axis.
DripFeed	Drip feed mode status ready. Indicates that the CNC is ready to operate in drip feed mode.
EmergencyRetract	General emergency retract. Set to indicate execution of an emergency retract program. Set by the CNC after *"W_ALLCNC.EmergencyRetract"* = 1 if the emergency retraction program is enabled. Reset by detection of M00 or M02.
Intervention	Cycle hold state. After a cycle stop, the switch to intervention state occurs the first time the AXIS RECALL key on the machine panel is pressed. Set by *"WCNC.Channel[x].AxisRecall"* = 1. Reset by *"WCNC.Channel[x].AxisRecall"* = 0.

MachineError	Machine Error Number. This variable contains the machine error number detected by the system (errors 18, 30-33, 35, 36, 39-71, 210-241, 245, 300-331) as a decimal code. Part program error numbers are not available. REMARK. Refer to the Flexium HMI Operator manual M00017 for the list of machine errors.
M01	Optional program stop status. Image of field M01 of the CNC status window. If the bit is set, optional program stops in a part program will be executed. The state of this bit can be changed by pressing the M01 key on the panel or by *"W_ALLCNC.M01"* = 1.
Mode	Current Mode. The value of the variable is the image of the current CNC mode: The current mode is defined by Enum.mode Possible values — Numerical values - AUTO — 0 - SINGLE — 1 - MDI — 2 - RAPID — 3 - SEARCH — 4 - TEST — 6 - MANUAL — 7 - HOME — 8 - NO MODE — 11
NMAuto	N/M AUTO functionality. Set to indicate that the N/M functionality (2/3, 3/5, etc.) is enabled.
ProgramActive	Program active. Indicates that a part program is being executed in AUTO, SINGLE, MDI or DRYRUN mode. Set by the first action on the START key on the machine operator panel, *"W_ALLCNC.Start"* = 1. Reset by detection of M00 (program stop), M01 (optional stop), M02 (end of program); the RESET key on the operator panel, *"W_ALLCNC.Reset"* = 1 and at power on.
ProgramNumber	Active Program Number. This variable contains the number of the active program. The value 0XFFFF (-1) indicates there is no active program. The value 0x0000 (0) indicates the selection of the drip feed mode.
ProgramStop	Program stop. Set to indicate a program stop caused by M00 or enabled M01. Reset by the START key on the machine operator panel. (*W_ALLCNC.Start*).
Reset	CNC reset in progress. Pulse bit with a duration of 100ms indicating a system reset. While this pulse is high, demands from the PLC function are ignored. Set by the RESET key on the panel, a reset request from the PLC, *"W_ALLCNC.Reset"* =1, at the end of execution of a part program (M02) or when the CNC is turned on. This variable is automatically reset after 100ms.

Start	Cycle in progress. Set by the START key on the machine panel, *"W_ALLCNC.Start"* = 1. Reset by the RESET key on the panel, *"W_ALLCNC.Reset"* = 1 or at the end of execution of the program (M02).
Stop	Cycle stop. Indicates the STOP state of the system (program stopped during execution and enabling of axis jogs). Set by the STOP key on the machine operator panel *"W_ALLCNC.Stop"* = 1. Reset by the CYCLE key on the machine operator panel *"W_ALLCNC.Start"* = 1.
JogIncrement	Current jog increment. The value of this variable is the image of the current jog increment: The value is of type Enum.JogInc Possible values / Mnemonic / Numerical value 0,1µ - 0,00001" / Inc_0_1 / 0 1µ - 0,0001" / Inc_1 / 1 10µ - 0,001" / Inc_10 / 2 100µ - 0,01" / Inc_100 / 3 1 mm - 0,1" / Inc_1000 / 4 10 mm- 1" / Inc_10000 / 5 Unlimited / Inc_Continuous / 6 Handwheel / Inc_Handwheel / 8 0,01µ - 0,000001" / Inc_0_01 / 9 0,001µ - 0,0000001" / Inc_0_001 / 10 The increment is in mm or inches depending on the value of variable *"W_ALLCNC.InchUnit"*. The smallest increments are only available if the measurement resolution is in accordance.

3.1.2 Information from the axes channels

All these variables apply to independent CNC axes channels.
Only the variables Reset, Start, EmergencyRetract, AxisNotInPosition and CncError apply to PLC axis channels.

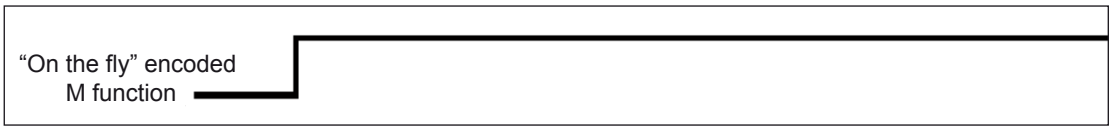
Syntax: **Flexium_NCK.RCNC.Channel[x]** *'variable name'* (x being the channel number 1 to 8)

Variable name	Description
AxisNotInPosition	Axis waiting to reach position. When accurate positioning is required by the CNC program (functions G09, M00, M02, M10) or in MDI or JOG mode: whenever movement demands have finished, the variable is set as long as the axis has not reached position.
AxisRecall	Axis recall. Indicates that axis recall is enabled.
BlockSkip	Block skip enabled.
CncError	Error in channel. Indicates a programming error or the absence of a part program in the channel.
CycleNumber	Number of the current machining cycle. Used to read the machining cycle subroutine number from %10000 to %10255 (0 for %10000 to 0xFF for %10255).
EmergencyRetract	Emergency retract in progress. Indicates execution of an emergency retraction sequence. Set after reading of *"W_ALLCNC.EmergencyRetract"* = 1 by the CNC if the emergency retraction sequence has been defined by G75 N... Reset by detection of M00 or M02.
Intervention	Intervention status.
Mode	Current mode. The value of this variable is the image of the current CNC mode on independent axes channels.
M01	Optional program stop enabled.
ProgramActive	Program active on independent CNC axes channel. Indicates that a part program is active in one of the following modes: AUTO, SINGLE, DRYRUN, SEARCH, TEST or MDI.
ProgramStop	Indicates a program stop caused by M00 or enabled M01.
Rapid	Rapid positioning (G00) active. Indicates execution of G00 in the current block of the part program. Set by execution of the function G00 and reset at its cancellation.

Reset	Reset in progress. Pulse bit with a duration of 100ms that indicates a reset on the channel. While this pulse is set, requests from the PLC are ignored. Set by the Reset key on the operator panel, by a reset request from the PLC *"W_ALLCNC.Reset"* = 1, at the end of execution of a part program (M02) or at CNC power on.
Start	Cycle in progress. Indicates that the channel is executing a part program block.
Stop	Cycle stop active. Indicates system intervention status (program stopped during execution and axis jogs enabled).
ThreadCutting	Thread cutting in progress. Indicates execution of a thread cutting cycle: - G31 (thread chasing), - G33 (thread cutting) or G38 (sequenced thread cutting), - G84K (rigid tapping). Set by execution of function G31, G33, G38 or G84. Reset by cancellation of the function.
ToolNumber	Tool Number Requested by the channel. Contains the tool number (decimal values from 0 to 65535). T functions are considered as pre-move and modal by the system which does not wait for an acknowledge.

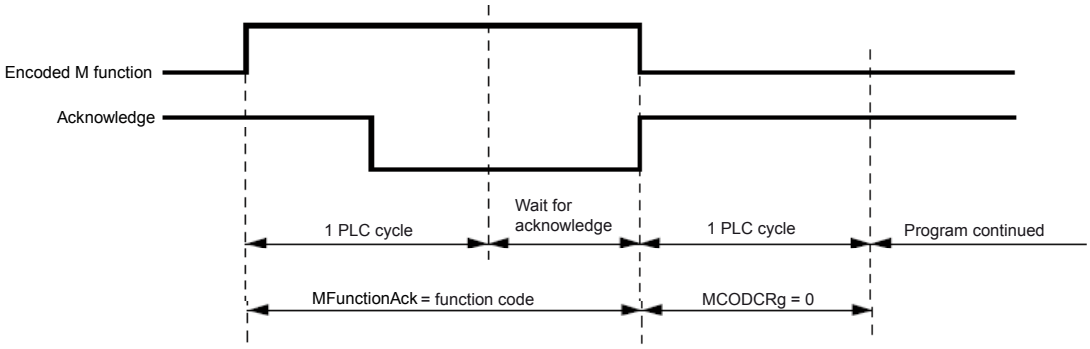
3.1.2.1 M Functions without acknowledge

Variable name	Description
MFunctionNoAck	Encoded M Function without acknowledge. This variable is used to read "on the fly" encoded M functions without acknowledge from M200 to M899. These functions are considered pre-move and modal by the system. The part program is continued without waiting for an acknowledge. Used in part programs, they are accessible by the PLC and must be decoded in the application. Only one modal encoded M function can be included in a part program block. One modal and one nonmodal encoded M function can be included in the same part program block. M functions must always be decoded in a fast sequential task.



3.1.2.2 M Functions with acknowledge

Variable name	Description
MFunctionAck	Encoded M Function with acknowledge. This variable is used to read the encoded M functions with acknowledge up to M199. These functions are considered post move and non-modal by the system. Used in part programs, they are accessible by the PLC and must be decoded in the application. Only one non-modal encoded M function can be included in a part program block. One modal and one nonmodal encoded M function can be included in the same part program block.



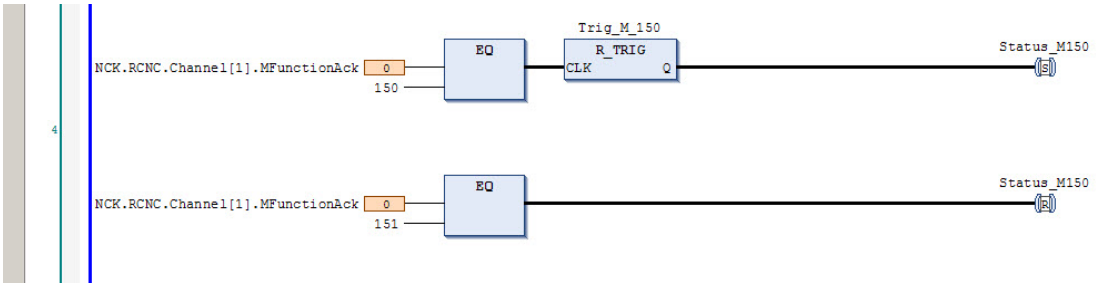
To avoid multiple executions of the function, it is recommended to use a rising trigger in the detection routine.

Modal functions

A modal function remains stored and enabled during the execution of several part program blocks, until it is cancelled.

Example

M function decoding on Channel 1: M150 modal cancelled by M151 non modal



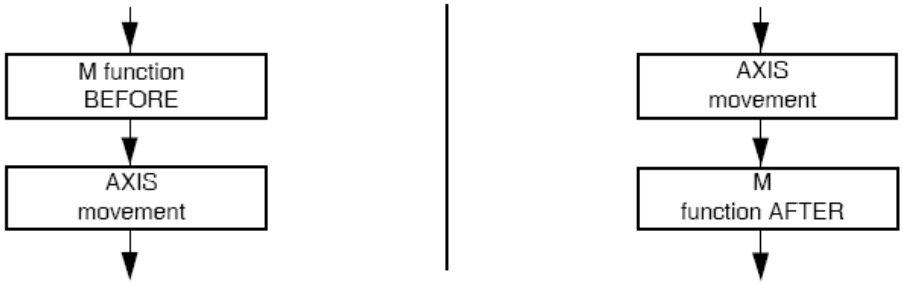
Notice the S and R coils
- M150 is triggered for example to start a flow chart.
- M151 has no effect except cancelling M150.

3.1.2.3 Decoded M functions

These functions used in part programs are accessible for reading by the PLC.
These functions are pre-defined by the system (e.g. axis clamping, spindle speed range, etc.).

The PLC reads the function on a bit assigned to a decoded M function.

A distinction is made between:



Modal functions

A modal function remains stored and enabled during the execution of several part program blocks, until it is cancelled.

Example

```
N100 M3 M40 S1000
N110 X100
N120 M5
```

M3 and M40 detected and enabled.
Movement on X. M3 and M40 remain enabled.
M5 sent to PLC cancelling M3 M40.

Nonmodal functions

A nonmodal function is enabled only during execution of the part program block containing it.

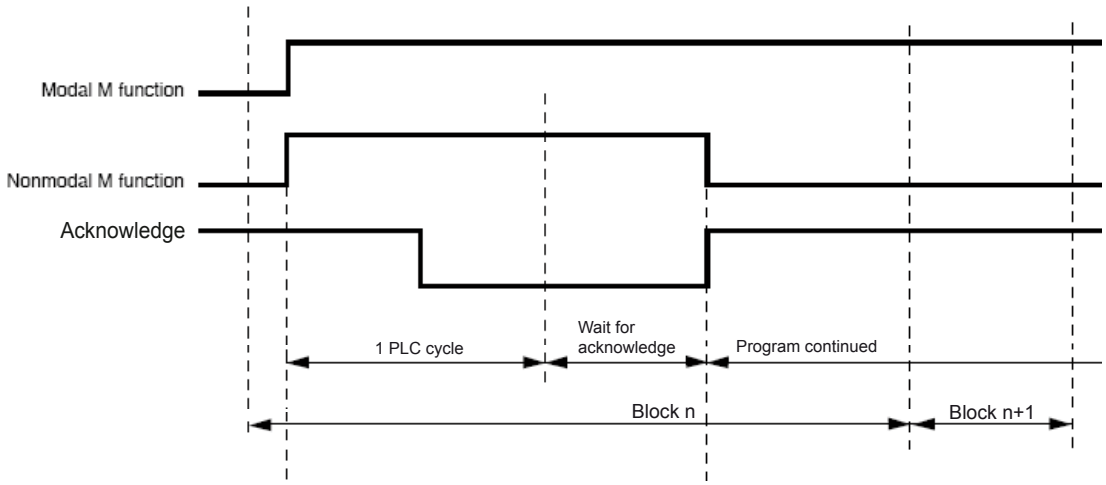
Example

```
N100 X100 Z200 M6
N110 X50
```

M6 detected and enabled.
M6 cancelled by the acknowledge of the previous block.

All decoded M functions are acknowledged by the signal *"Channel[x]. AckMFunction"*. The status of *"Channel[1].AckMFunction"* to *"Channel[8]. AckMFunction"* determines whether the part program continues or waits for the acknowledge.

The PLC application must manage *"Channel[1].AckMFunction" to "Channel[8]. AckMFunction"* for all programmed functions as well as their possible cancellation by another M function or by an unexpected Reset or INIT.



If *Channel[1].AckMFunction* to *Channel[8].AckMFunction* remain high, the part program is continued after a single PLC cycle.

If *Channel[1].AckMFunction* to *Channel[8].AckMFunction* remain low, the system waits until the signal is set again. At the power up these flags should be set to complete the initialisation.

3.1.2.4 List of M functions

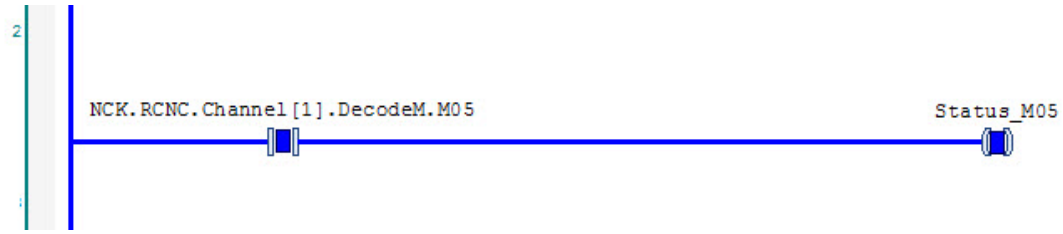
M funct.	Definition	Cancelled by	Type of function			
			Pre	Post	Modal	Non modal
M00	Program stop	Start		●		●
M01	Optional program stop	Start		●		●
M02	End of part program	Reset		●		●
M03	Spindle CW rotation	M0, M2, M4, M5, M19	●		●	
M04	Spindle CCW rotation	M0, M2, M3, M5, M19	●		●	
M05 *	Spindle stop	M3, M4		●	●	
M06	Tool change	AckMFunction		●		●
M07	Coolant 2	M2, M9	●			
M08	Coolant 1	M2, M9	●			
M09 *	Coolant stop	M7, M8		●	●	
M10	Axis clamp (Initialised after a reset if blocable axes are declared)	M11		●	●	
M11	Axis unclamp	M10	●		●	
M12	Programmed feed stop	Start		●		●
M19	Indexed spindle stop	M0, M2, M3, M4, M5		●	●	
M40	Spindle range 1	M2, M41, M42, M43, M44, M45	●		●	
M41	Spindle range 2	M2, M40, M42, M43, M44, M45	●		●	
M42	Spindle range 3	M2, M40, M41, M43, M44, M45	●		●	
M43	Spindle range 4	M2, M40, M41, M42, M44, M45	●		●	
M44	Spindle range 5	M02, M40, M41, M42, M43, M45	●		●	
M45	Spindle range 6	M02, M40, M41, M42, M43, M44	●		●	
M48 *	Feed rate and spindle override enabled	M49		●	●	
M49	Feed rate and spindle override forced at 100%	M48, M2	●		●	
M61	Disable current spindle in a channel	M62, M63, M64, M65		●	●	
M62	Spindle reference applied to spindle 3	M61, M63, M64, M65	●		●	
M63	Spindle reference applied to spindle 4	M61, M62, M64, M65	●		●	
M64	Spindle reference applied to spindle 1	M61, M62, M63, M65	●		●	
M65	Spindle reference applied to spindle 2	M61, M62, M63, M64	●		●	
M66	Spindle 1 measurement enabled	M67, M68, M69, M2	●		●	

M67	Spindle 2 measurement enabled	M66, M68, M69, M2	●		●	
M68	Spindle 3 measurement enabled	M66, M67, M69, M2	●		●	
M69	Spindle 4 measurement enabled	M66, M67, M68, M2	●		●	
M997	Forced block continuation	M998, M999, M2	●		●	
M998 *	Programmed enable of MDI and EDIT modes and subroutine calls by the machine processor.	M999, M997	●		●	
M999	Programmed inhibit of MDI and EDIT modes and subroutine calls by the PLC.	M997, M998, M2	●		●	

(*) These functions are initialised at Power On, by a Reset and by the function M02.

Example in Ladder language

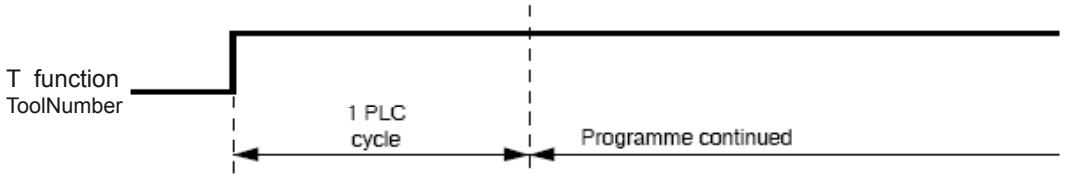
Detect M05



3.1.2.5 Tools

Variable name	Description
ToolNumber	Tool number requested by the channel. Contains the tool number (decimal value from 0 to 99999999). T functions are considered as pre-move and modal by the system which does not wait for an acknowledge.

Processing of T functions

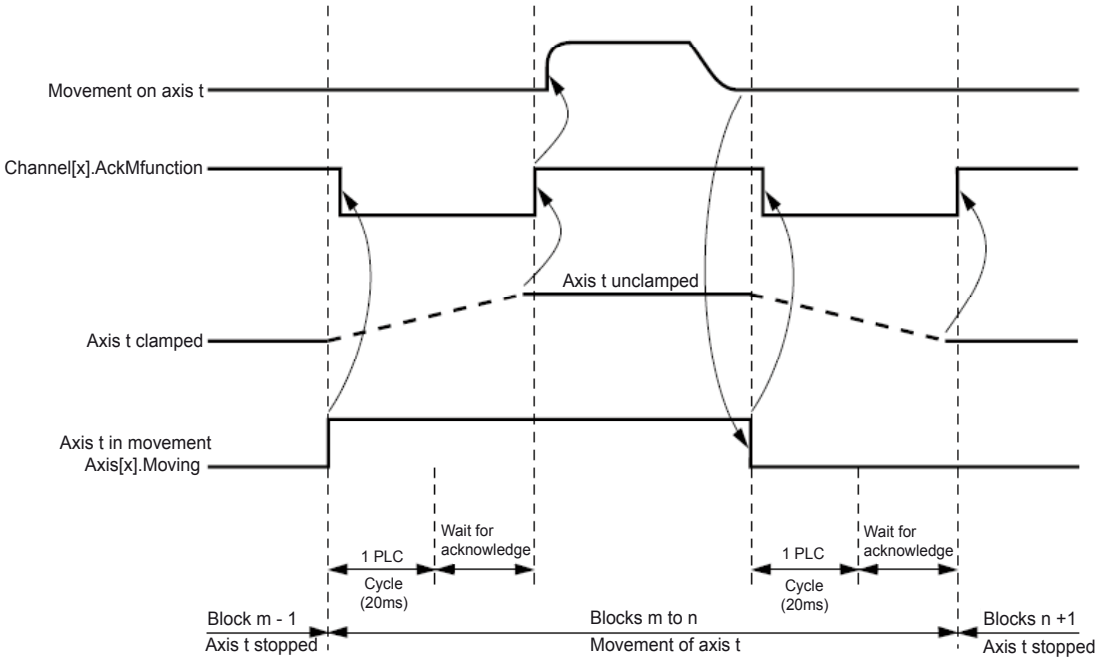


3.1.3 Axes information

Syntax: **Flexium_NCK.RCNC.Axis[x].** *'variable name'* (x being the axis number 0 to 31)

Variable name	Description
Axis[x].Clamp	Indicates that the axis n° X can be clamped. It is defined by parameter P8 and can be modified by E parameter E913xx.
Axis[x].Moving	This variable has two different behaviours according to the clamping status. For clampable axes there are two possibilities selected by M10/M11 (modals): - M10: Axes are working with clamp. - M11: Axes are always unclamped. In M10 each time an axis will change in the transition between two blocks (start or stop) the system will position the flags **.Moving** according to the next status and wait for acknowledge from the PLC until the clamp/unclamp operation is finished. In M11 axes are supposed to be unclamped all the time and the flags **.Moving** will be positionned simultaneously to the motion. (for a continuous move preventing marks on the part)
Clamp	Indicates that the axis can be clamped. Reset if the axis cannot be clamped. A reset places the axes in the configuration specified in machine parameter P8.
Moving	Indicates the axis is in motion during execution of a block in a part program or in MDI mode. Set at the start of execution of the block. Reset at the end of execution of the block if it includes M00 or M01, at the end of execution of the block in MDI mode, before carrying out axis clamping, by the Reset key on the operator panel or variable *"W_ALLCNC.Reset"* = 1.
NotHomed	Indicates that homing on the axis has not been completed. Reset when homing is completed. Set by a system restart.

Principle of axis clamping/unclamping



If the axis is still moving at block n + 1, the variable *Axis[x].Moving* remains high and the blocks are sequenced.

3.1.3.1 Axis status

Syntax: **Flexium_NCK.RCNC.Axis[x].Status.** *'variable name'* (x being the axis number 0 to 31)

Variable name	Description
ConsWPMActive	1: WPM connected or in console mode
DCBusPresent	1: Bus voltage present
DigInput_A	1: S7 connector pin 4 = 24V
DigInput_B	1: S7 connector pin 5 = 24V
DriveInRunStatus	1: Drive in run state
DriveOk	1: No error
EPS_EPSMActive	1: Self-calibration in progress
HighElectrical gear	0: Low speed range 1: High speed range
MotorMoving	1: Motor speed is higher than +/- V41 or V42
Prog01	1: S7 connector pin 2 = 24V
Prog02	1: S7 connector pin 3 = 24V
PWMCutOff	1: Emergency stop or SAM intervention
ReferenceEnabled	1: Reference (speed) enabled
SBCEOK	1: SBCE status OK
SpeedReached	1: Speed reached
TorqueEnabled	1: Drive enabled
Warning	1: Drive is issuing a warning

3.1.4 Spindle status

Syntax: **Flexium_NCK.RCNC.Spindle[x].** *'variable name'* (x being the spindle number from 1 to 4)

Variable name	Description
PositionReached	Spindle in position. On a spindle indexing or synchronisation request, indicates that the spindle is in position or synchronised. Set when the required position is reached. Reset after leaving the required position, by oscillations and by cancellation of function M19.
Speed	Spindle speed reference. Contains the hexadecimal code of the spindle servo-drive reference in the programmed speed range. The absence of functions M3 and M4 in the part program forces the variable to zero.
SpeedReached	Set to indicate that the rotation speed of the spindle is correct, i.e. that its speed is within the tolerance interval specified in parameter E9035x (see Flexium Programming manual M00018)
Stopped	Set when the spindle is stopped, i.e. its rotation speed is less than the setting of parameter E9034x (spindle speed threshold, see Flexium Programming manual M00018).

3.1.5 External parameters read by PLC

Syntax: **Flexium_NCK.RCNC.** *'variable name'*

E parameter	Description
E10000[x]	External parameters E10000[x] (x being a number from 0 to 31). These parameters are managed by the part program where they can be written and read. They are used to make part program status available to the PLC application in read-only mode. Format: Boolean (0/1)
E30000[x]	External parameters E30000[x] (x being a number from 0 to 255). These parameters are managed by the part program where they can be written and read. They are used to make part program status available to the PLC application in read-only mode. Format: DINT = Double integer (32 Bit)
E33000[x]	External parameters E33000[x] (x being a number from 0 to 999). These parameters can be written or read by the part program. The first 10 parameters can be written by dynamic operators. Format: USINT (8 Bit)
E36000[x]	External parameters E36000[x] (x being a number from 0 to 999). Access to the PLC memory. These parameters can be written or read by the part program. The first 2 parameters can be written by dynamic operators. Format: USINT (16 Bit)
E37000[x]	External parameters E37000[x] (x being a number from 0 to 127). These parameters can be written by dynamic operators in the NCK. Format: USINT (Byte)

For more detailed informations on the E parameters please refer to the Flexium Programming manual M00018.

3.1.6 Embedded NCK Inputs

Syntax: **Flexium_NCK.RCNC.IO** *'variable name'*

Variable name	Description
AnalogInput[x]	Value of the analogue input, 12 bits ADC 0-10V, located on the NCK unit (X9 connector. Please refer to Flexium Installation manual M00009). x being a number from 0 to 3. The data is also available in the parameters E44100 to E44103.
DigitalInput[x]	State of the digital input byte located on the NCK unit (X10 connector. Please refer to Flexium Installation manual M00009). x being a number from 0 to 1. The data is also available in the parameters E44000 to E44001.

3.1.7 Exchange area from HMI to PLC

Syntax: **Panel.FlexiumHMI.** *'variable name'* (*"Panel"* being the name of the HMI display given in the application)

Variable name	Description
CommunicationReady	Set if the communication with the HMI is running
RHMI.Channel.Channel	Contains the number of the selected axes channel (1 to 8)
RHMI.Channel.NCK	Contains the address of the active NCK
RHMI.TeachIn.Insert	Teach-in: acknowledge for block insert (100ms)
RHMI.TeachIn.Overwrite	Teach-in: acknowledge for block overwrite (100ms)
RHMI.TeachIn.TeachInOpen	Teach-in editor selected

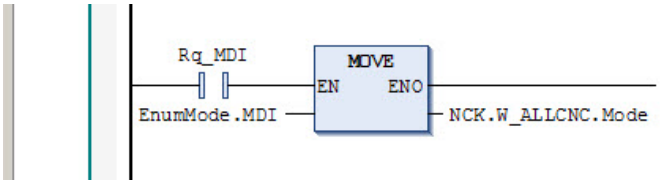
Do not forget to set the variable **Panel.FlexiumHMI.WHMI.Channel.NCK** at the same value as the NCK address. If this is not the case, the system will not know which NCK to address for the above RHMI variables and the functionality will not work.

3.2 Informations to the NCK

3.2.1 Exchange signals to all NCK

Syntax: **Flexium_NCK.W_ALLCNC.** *'variable name'*

Variable name	Description
AxisRecall	Axis recall selection. This request is taken into account when *"RCNC.Channel[x].Stop"* = 1 and all the axis jogs are released. It is a bistable type command. The first pulse sets *"RCNC.General. Intervention"* and enables the axis jogs in both directions. If at least one axis has been moved in "INTERV" mode, a second pulse sets *"RCNC.General.AxisRecall"* and enables a single direction of movement of the axis jogs to return the axes to their initial position.
BlockSkip	Block skip toggle. A pulse forces a change of state to enable or inhibit block skip depending on the previous state.
CancelLagError	It is possible to cancel the lag error even if the CNC is not in alarm. Resuming execution will be done after an axis recall procedure. The PLC command «*Flexium_NCK.W_ALLCNC.CancelLagError* » reset the lag value even if there is no following error (error 40+x). To avoid risks related to moving axes, clearing the lag value when there is no following error is taken into account when the 2 conditions below are met: - CNC is in STOP mode - PLC variable CancelLagError is set To be able to restart the program, it's mandatory to move in Jog mode the axis to the INPOS position, before to START the program.
DifferentialAxisOffset	Set to enable a differential axis offset with a handwheel. Used with the external parameter E9530x and the dynamic operator 24.
EmergecyRetract	Emergency retract request. This request is taken into account in the AUTO, SINGLE and DRYRUN modes. The current block is interrupted and the system branches to the last emergency retract sequence declared in the part program by the function G75. If no emergency retract sequence was defined, this command is processed in the same way as *"W_AllCNC.Stop"*.
FeedEnable	Feed authorised on all axes channels. Enables axes movement in all modes. The state of this signal is displayed as the *"STOP"* indicator of the CNC status window (*"STOP"* indicates that this variable is not set).
InchUnit	Display units (metric or inch system). Set to enable dimension entry and display in inches. Reset to enable dimension entry and display in m.
IndependentGroup	Common channels/independent channels switchover. This command is latched. A change of state of *"IndependentGroup"* is detected only on a general reset requested by the PLC. IndependentGroup = 0: Common channels. IndependentGroup = 1: Independent channels.

LockDynToolOffset	Wear offset load enable. Set to enable load of the wear offsets by the PLC and disable load from the HMI.
LockMode	Mode selection by the PLC. Set to disable mode selection by the NCK and enable mode selection by the PLC. Mode selection by the PLC: the mode is encoded in *" W_ALLCNC.Mode.EnumMode"*. The mode codes are the same as those of *"WCNC.General.Mode"* for common modes.
LockJog	JOG selection by the PLC. Set to disable selection of the jog keys by the NCK and enable selection by the PLC.
M01	Optional stop (M01) toggle. A pulse forces a change of state to enable or inhibit optional stop depending on the previous state.
ManualSpeed.(Enum-ManSpeed)	Enable selection of rapid feed rates in the JOG and HOME modes or setting the handwheel increment multiplier. Possible values — Numerical values - NormalJog — 0 - SlowJog — 2 - FastJog- SlowJog — 1 - HandWheel_1 — 4 - HandWheel_10 — 5 - HandWheel_100 — 6 The feed rates are modulated by the feed override potentiometer. Example *Flexium_NCK.W_ALLCNC.ManualSpeed:= EnumManSpeed.SlowJog;* This value is only taken into account in common channels mode. For PLC Channels or independant channels it is the *Flexium_NCK.WCNC.Channel[i].ManualSpeed* value which is considered.
Mode. (EnumMode)	Requested Mode. The following CNC modes can be requested: The value is from type EnumMode. Possible values — Numerical values - AUTO — 0 - SINGLE — 1 - MDI — 2 - RAPID — 3 - SEARCH — 4 - TEST — 6 - MANUAL — 7 - HOME — 8 - NO MODE — 11 Example in Ladder 

NMAuto	N/M AUTO functionality. Set to enable the AUTO N/M (2/3, 3/5, et.) functionality. This command is operative when signal *"Start"* goes low.
NoStopBlockEnd	No stop at end of block. Enables execution of a Cycle Start in the AUTO, SINGLE, MDI, RAPID modes and enables block sequencing in the AUTO and RAPID modes. Reset of this variable causes the cycle to stop at the end of execution of the current block.
ReducedSpeed	Causes a switch to low speed. Set to force the speed reductions defined in words N3 and N4 of parameter P31 (see Flexium Parameter manual M00021). This value is only taken into account in common channels mode. For PLC Channels or independant channels it is the *Flexium_NCK.WCNC.Channel[i].ReducedSpeed* value which is considered.
ReloadDynOp	If *"ReloadDynOp"* is set, the dynamic operators in C are reloaded by a general CNC Reset. This flag is currently not used in a Flexium application.
Reset	Reset request. Also resets the PLC axes in case of a machine error. Only taken account if there is no movement on the axes.
Start	Cycle start pulse. Allows execution of the AUTO, SINGLE, MDI, DRYRUN, SEARCH, and TEST modes. A pulse command must be used for *"Start"* to prevent the restart of machining after detection of M02 or a reset in the AUTO and DRYRUN modes.
Stop	Machining stop request. This request is taken into account in the AUTO, SINGLE, MDI, DRYRUN and incremental JOG modes. The first pulse stops machining. Machining is restarted by activating *"Start"*.
SynchAxisPowerOn	Motor power on. Setting this variable to 0 indicates a motor power failure of a synchronised axis to the NCK (after a synchronisation error). Set to 1 to indicate power return and axis synchronisation enable to the CNC.

3.2.2 General exchange signals

Syntax: **Flexium_NCK.WCNC.General.** *'variable name'*

Variable name	Description
BacktrackAuto	Automatic axes recall after intervention. Set to enable recall. Reset to cancel recall.
BacktrackBackward	Backward movement requested on the programmed path after an intervention. Set to enable the request. Reset to inhibit the request.
BacktrackForward	Forward movement requested on the programmed path after an intervention and a prior backtrack backward. Set to enable the request. Reset to inhibit the request.
LockMode. (EnumMode)	Inhibited CNC modes.
LockIncJog. (EnumJogInc)	Inhibited jog increments. Every selectable jog increment can be locked.
ProgramNumber	Requested Program Number. Used to set the part program with the requested number as the active program or request machining in drip feed mode. The program number or drip feed machining request is read by the system on the rising edge of the reset flag *"W_ALLCNC.Reset".* 0: No program number requested by the PLC. 1 to 0x270F (9999): Program number specified by the PLC. -2 (0xFFFE): Machining in drip feed mode requested by the PLC.
JogIncrement. (EnumJogInc)	JOG increment selection. The variable value corresponds to the jog increment requested: Possible values — Numerical values - Inc_0_001 — 10 (Currently not used) - Inc_0_01 — 9 - Inc_0_1 — 0 - Inc_1 — 1 - Inc_10 — 2 - Inc_100 — 3 - Inc_1000 — 4 - Inc_10000 — 5 - Inc_Continous — 6 - Inc_Handwheel — 8 The jog increment is in mm or inches depending on the value of variable InchUnit. Example *Flexium_NCK.WCNC.General.JogIncrement:=EnumJogInc.Inc_Handwheel*

3.2.3 Outputs to the axes channels

Syntax: **Flexium_NCK.WCNC.Channel[x].** *'variable name'* (x being the channel number 1 to 8)

Variables *"PlcAxisMode"* apply only to PLC axes channels.
Variables *"Stop"*, *"AxisRecall"*, *"BlockSkip"* and *"M01"* apply only to CNC axes channels.

Variable name	Description
AckMFunction	M function acknowledge. Reset: program continuation is suspended. Set: enables processing to continue.
AxisRecall	Select axis recall on independent CNC axes channel. This request is accepted in AUTO, SINGLE and RAPID (DRYRUN) modes.
BlockSkip	Enable block skip on independent CNC axes channel. A pulse enables or inhibits block skip by toggling from the previous state.
ChannelEnable	Channel enable. Set to enable use of the axes channel. Enabling or inhibiting is effective only after a reset or M02.
EndExtMovement	End of external movement control on the channel. Reset to inhibit cancellation of CYCLE in SINGLE and MDI modes or sequencing to the next block in AUTO and DRYRUN modes. Set to allow normal execution of the mode. This variable is tested at the end of execution of each block.
EmergencyRetract	Emergency retraction request. This request is accepted in AUTO and SINGLE modes. The current block is interrupted and the system branches to the last emergency retraction sequence declared in the part program by function G75. If no emergency retraction sequence was specified, this signal is processed in the same way as *"W_ALLCNC.Stop"*.
FeedEnable	Feed authorisation for the channel. This variable is active if general feed authorisation *"W_ALLCNC. FeedEnable"* = 1. Reset to stop movement on the axes channel in all modes with movements. Movement is resumed when "FeedEnable" = 1.
FeedOverride	Value of the Feed Rate potentiometer.
M01	Enable optional program stop (M01) on independent CNC axes channel. A pulse enables or inhibits the optional program stop by toggling from the previous state.

ManualSpeed.(Enum-ManSpeed)	Enable selection of rapid feed rates in the JOG and HOME modes or setting the handwheel increment multiplier. Possible values — Numerical values - NormalJog — 0 - SlowJog — 2 - FastJog- SlowJog — 1 - HandWheel_1 — 4 - HandWheel_10 — 5 - HandWheel_100 — 6 The feed rates are modulated by the feed override potentiometer. Example *Flexium_NCK.WCNC.Channel[2].ManualSpeed:=EnumManSpeed.SlowJog* This value is only considered for PLC Channels or Independant channels. In Common channels mode it is the value *Flexium_NCK.W_AllCNC.ManualSpeed* which is considered.
Mode.(EnumMode)	Mode requested on independent axes channel. The following modes can be requested: Possible values — Numerical values - AUTO — 0 - SINGLE — 1 - MDI — 2 - RAPID — 3 - SEARCH — 4 - TEST — 6 - MANUAL — 7 - HOME — 8 - NO MODE — 11 Example *Flexium_NCK.W_ALLCNC.Mode : = EnumMode.Manual*
PlcAxisMode	Activates AUTO or SINGLE mode on PLC axes channel. Reset: AUTO mode enabled on the next block. Set: SINGLE mode enabled for the current block. This flag is only meaningful if the channel is enabled.
PlcCallSubroutine	Subroutine call by the PLC. During execution of a part program, setting *“PlcCallSubroutine”* causes a branch to subroutine %9999.x (where x is the channel number). Latching the bit or setting it again is ignored during execution of the subroutine. No confirmation is sent by the system. The subroutine itself should send an acknowledge to the PLC to reset *“PlcCallSubroutine”* (M function, external parameter, etc.). If only one CNC axes channel is declared, program %9999 is called (i.e. %9999.0).
PlcInterrupt	Block interrupt. Set to stop movement on the axes channel, then go to the next block or jump to another block. Function G10 with its associated arguments must be present in the part program.

ProgramNumber	Requested program number, for independent axes channels only. Used to set the part program with the requested number as the active program. The program number is read by the system on the rising edge of the reset flag *"W_ALLCNC.Reset"*. 0: No program number requested by the PLC. 1 to 9999: Program number specified by the PLC.
Rapid	Latched high speed command during program execution. This command is used during a cycle (*WCNC.Channel[x].Start = 1*). Set to enable movement at the highest possible speed. Reset to enable movement at normal feed rate.
ReducedSpeed	Defined in words N3 and N4 of parameter P31. (see Flexium Parameter manual M00021)
Reset	Reset request on PLC or independent axes channel. Taken into account if there is no movement on the axes. It is during a reset on a channel that the flag "ChannelEnable" is taken into account and the presence of the part program assigned to the PLC channel is detected.
Start	Cycle start request on PLC or independent axes channel. Allows execution of the AUTO and SINGLE modes for the PLC axes channels. A pulse command must be used for "Start" to prevent resumption of machining after detection of M02 or a reset in the AUTO mode. This signal is ignored unless the channel is valid.
Stop	Cycle stop request on independent CNC axes channel. This request is accepted in AUTO, SINGLE, RAPID (DRYRUN), SEARCH, TEST and MDI modes.

3.2.4 Outputs to the axes

Syntax: **Flexium_NCK.WCNC.Axis[x].** *'variable name'* (x being the axis number 0 to 31)

Variable name	Description
Stop	Stop axis. In machining mode or JOG mode, setting a bit that addresses one of the axes in movement stops the axes of the channel to which it belongs. In machining mode, if this axis does not move in the block being executed but its movement is programmed in the next block, an end-of-block stop request is generated and execution of the movement at the beginning of the next block remains suspended whilst the stays set. In NM auto mode, action on the jogs or handwheel is ignored as long as the bit corresponding to the axis directly controlled is set.
JogPos	Positive jog on axis
JogNeg	Negative jog on axis

3.2.5 Servo drive control

Syntax: **Flexium_NCK.WCNC.Axis[x]. ServoCtr.** *'variable name'* (x being the axis number 0 to 31)

Variable name	Description
CurrentReduction	The current reduction function is used to reduce the current on the digital axes and spindles. Let $I_{maximum}$ be the maximum current taking into account static current limiting and α be the value of the byte: - If α is negative or zero ($00, $80-$FF), there is no current reduction. - If α is positive ($01-$7F), the maximum permissible current is: $I_{maximum} = I_{max\_stat} \times [(127 - \alpha)/127]$. The dynamic current reduction applied to a master digital servo-drive is transmitted to associated slave digital servo-drives. When operating in antibacklash configuration, the dynamic reduction applied to a master digital servo-drive has no effect on the master and slave preload currents.
RampDisabled	Used to modify the bit 5 (speed ramp) in the drive command register (0x1A00). 0: the speed ramp is controlled by V30 bit 14 1: the speed ramp is always disabled. (Please also refer to NUMDrive Parameters manual AMOMAN007)
ReferenceEnable	0: reference not enabled 1: reference enabled
SAMDisableRequest	Used to modify the bit 11 (SAM disable) in the drive command register (0x1A00). A transition from 0 to 1 changes the SAM status from 6 to 4. (Please also refer to NUMDrive Parameters manual AMOMAN007)
SpindleElectricalConfiguration	0: low range 1: high range
TorqueEnable	0: torque not enabled 1: torque enabled

3.2.6 Spindle control

Syntax: **Flexium_NCK.WCNC.Spindle[x].** *'variable name'* (x being the spindle number 1 to 4)

Variable name	Description
IndexReq	Spindle indexing request by the PLC.
Lock	Spindle power. - 0: Indicates to the NCK that spindle power is on. - 1: Indicates to the NCK that spindle is manually inhibited or locked.
Override	Spindle speed override value (0 to 255). The extreme speed override values are defined in machine parameter P67.
PlcSpeedControl	Spindle control. Set to enable spindle control by the PLC. The speed value is transmitted to the spindle drive by *"Spindle[x].Speed"*.
ReducedSpeed	For safety reasons, the PLC can enable a limit spindle speed via machine parameter P43. Set to enable the speed limit.
Stop	Spindle stop requested by the PLC function. Latched command. The spindle remains stopped as long as this bit remains set. When the bit is reset, the spindle can begin rotating again.
Speed	Spindle speed value. Used to send the spindle servo-drive reference in binary code on 14 bits. The spindle direction is managed by adding a sign bit (bit 15).

3.2.7 Spindle load

Syntax: **Flexium_NCK.WCNC.Spindle[x].Load.** *'variable name'* (x being the spindle number 1 to 4)

Variable name	Description
ByPlc	Spindle load display controlled by the PLC. 0: Indicates that spindle load is displayed according to the drive testpoint value. 1: Indicates that spindle load is displayed according to the PLC load value.
Value	Load value for PLC controlled display (0 to 150)

3.2.8 External parameters Written by PLC - Read by Part Program

Syntax: **Flexium_NCK.WCNC.** *'variable name'*

E parameter	Description
E20000[x]	External parameters E20000[x] (x being a number from 0 to 31). These parameters are managed by the PLC application where they can be written and read. They can be read in the part program. They are used to make machine status data available to the part program. Format: Bool (0/1)
E40000[x]	External parameters E40000[x] (x being a number from 0 to 255). These parameters are managed by the PLC application where they can be written and read. They are used to transfer signed numerical values like dimensions, offset, etc. for use in the part program. Format: DINT (32 Bits)
E43000[x]	External parameters E43000[x] (x being a number from 0 to 999). Addressing of the PLC data grouped by Bytes. These parameters can be read by the part program. The first 10 parameters can be read by Dynamic operators. Format: USINT (8 Bits)
E46000[x]	External parameters E46000[x] (x being a number from 0 to 999). Addressing of the PLC data grouped by 16 bits. These parameters can be read by the part program. The first 2 parameters can be read by Dynamic operators. Format: UINT (16 bits)
E47000[x]	External parameters E47000[x] (x being a number from 0 to 127). These parameters can be written by the PLC application. They are used to make PLC information available to the CNCK in read-only mode. They can be read by Dynamic operators. Format: Byte (8 bits)

For more detailed informations on the E parameters please refer to the Flexium Programming manual M00018.

3.2.9 Embedded NCK outputs

Syntax: **Flexium_NCK.WCNC.IO.** *'variable name'*

Variable name	Description
AnalogOutput[x]	Value of the analogue output (x being a number from 0 to 1). 16 bits DAC +/-10V. Parameters used for this purpose: E34100 to E34101.
DigitalOutput[x]	State of the digital output (x being a number from 0 to 1). Parameters used for this purpose: E34000 to E34001.
LockAnalogOutput	Enable writing of analogue outputs by the PLC. - 0: writing by PLC disabled - 1: writing by PLC enabled.
LockDigitalOutput	Enable writing of digital outputs by the PLC. - 0: writing by PLC disabled - 1: writing by PLC enabled.

3.2.10 Miscellaneous

Syntax: **Flexium_NCK.** *'function name'*

Function name	Description
CycleCounterPLC	This counter is incremented at every data exchange PLC to NCK.
CycleCounterNCK	This counter is incremented at every data exchange NCK to PLC.
CommunicationReady	Indicates that the communication between NCK and PLC is OK.
DeviceAdr	Setting of the address switch on the NCK.
HTRCycle	This counter is incremented at every HTR cycle (P50).

3.3 Informations for the HMI

Syntax: **Panel.FlexiumHMI.WHMI.** *'variable name'* (*"Panel"* being replaced by the name of the Operator panel (e.g. FS152i) given in the device tree of the project)

Variable name	Description
Channel.Channel	Axes Channel selection by the PLC. Used to display all data relating to the selected axes channel (part program, program variable, etc.). The data entered from the CNC keyboard in MDI mode is assigned to the selected axes channel. - 1 Selection of axes channel 1 - 2 Selection of axes channel 2 - 3 Selection of axes channel 3 - 4 Selection of axes channel 4 - 5 Selection of axes channel 5 - 6 Selection of axes channel 6 - 7 Selection of axes channel 7 - 8 Selection of axes channel 8.
Channel.NCK	NCK address selection by the PLC.
Channel.Lock	Set to enable channel and NCK selection by the PLC.
Security.LockSfPage[x]	Set to lock access to context pages (x being the context number 2 to 9).
Security.LockEditor	Set to lock access to the Flexium HMI editor.
Security.LockEditorLoadAndActivate	Set to lock the function "Load and activate" in the Flexium HMI editor.
Security.LockWriteVar	Set to lock the function "Write variables" in the context "Variables".
Security.LockChannelSelection	Set to lock axes channel selection by the HMI.
Security.LockExceptionPage	Set to lock access to the Exception page in the context "Diagnostic".
Security.LockHistory	Set to lock access to the History page in the context "Diagnostic".
Security.LockMemoryAccess	Set to lock write access to the protected memory areas.
TeachIn.Insert	Teach-in: insert a block.
TeachIn.Lock	Selection of the teach-in mode. - 1: teach-in under PLC control - 0: teach-in from HMI.
TeachIn.Overwrite	Teach-in: overwrite a block.

For each panel of the machine, an exchange area is required. For this reason they need to be all declared in the device tree of the project, each one with a different name so they can have different properties and authorisations.

3.4 Special Functions

3.4.1 Modification of tool wear offset by the PLC

The wear offsets modifications are accumulated by the NCK.
To enable the function: set *W_ALLCNC. LockDynToolOffset*

Syntax: **Flexium_NCK.** *'function'*

Function	
Flexium_NCK.ToolDyn(Correction: INT, Command: EnumToolDynCommand, ToolNumber: BYTE)	
Possible command values - IncrementCorrection_X_L - IncrementCorrection_Z_R - ResetCorrection_X_L - ResetCorrection_Z_R - ResetCorrection_all	Numerical values 16#1 16#2 16#81 16#82 16#83
Return values - 2: Error - 1: Busy0: OK 1: In progress	

Example
answer:= Flexium_NCK.ToolDyn(123,EnumToolDynCommand.IncrementCorrection_X_L,5);

The tool wear offset of D5 is incremented in X or L by the value of 123 internal units.

3.4.2 Touch probe and interrupts

Three functions are available to manage the touch probe signals.

3.4.2.1 Associate interrupt inputs lines with axes channel

Syntax: **Flexium_NCK.** *'function'*

Function
Flexium_NCK.TouchProbe.ChannelsCfg(LineNumber: BYTE, Channels: ARRAY [1..8] OF BOOL) Return values - 2: Error - 1: Busy0: 0: OK 1: In progress

Example
Channels[1]:=TRUE;
answer:= Flexium_nck.TouchProbe.ChannelsCfg(1,Channels);

Associate interrupt line 1 to channel 1.

3.4.2.2 Configuring an interrupt input

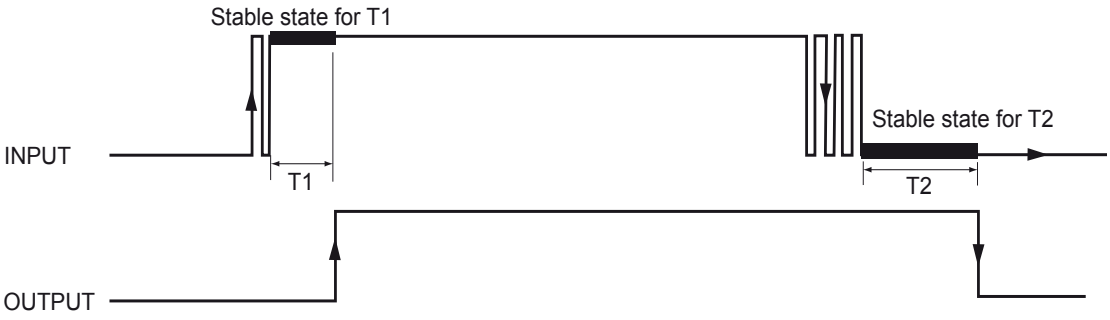
Syntax: **Flexium_NCK.** *'function'*

Function
Flexium_NCK.TouchProbe.HardwareCfg(LineNumber : BYTE,) Valid : BOOL, Edge : EnumTPEdge, FilterFallingEdge : BOOL, FilterRisingEdge : BOOL, FilterTimeFallingEdge : EnumTPTime, FilterTimeRisingEdge : EnumTPTime,) Possible EnumTPEdge values Numerical values - Rising 0 - Falling 1 Possible EnumTPTime values (ms) Numerical values - Time_05 0 - Time_1 1 - Time_4 2 - Time_8 3 Return values - 2: Error - 1: Busy 0: OK 1: In progress

Example
Flexium_nck.TouchProbe.HardwareCfg(1,TRUE,enumtpedge.Rising,TRUE,TRUE,enumtptime.Time_4,enumtptime.Time_1);

Enable interrupt line 1 with rising edge, filter on falling and rising edge, filter of 4ms on falling edge and 1ms on rising edge.

Example for filtering on rising edge (T1) and falling edge (T2)



INPUT: signal sent by the probe
OUTPUT: signal sent for processing after filtering.

3.4.2.3 Reading the state of an interrupt input

Syntax: **Flexium_NCK.** *'function'*

Function
Flexium_NCK.TouchProbe.InputReading(LineNumber: BYTE, Interrupt: POINTER TO BOOL, State: POINTER TO BOOL) Return values -2: Error -1: Busy 0: OK 1: In progress

Example
inter:BOOL;
interstate:BOOL;

Flexium_nck.TouchProbe.InputReading(1,ADR(inter),ADR(interstate));

The call of this function resets the interrupt flag (inter) and indicates the state of the interrupt line (interstate).

3.5 PLC Messages

The PLC can issue and reset machine specific messages used as operator guidance, warnings or error messages.

These messages can initiate various reactions, defined as reaction classes.
8 different reaction class are defined.

Each reaction class can:
- Have a specific action on the machine
- Be displayed in a specific color
- Be Logged or Not when appearing and disappearing.

To acknowledge the reactions and messages, different acknowledge classes are also available.

4 different acknowldgeclass are defined.
Each acknowledge class can have a specific method for clear the message (on an action or after a predefined time) **if the condition is not active anymore**.

PLC messages are stored in an ASCII file with the extension *.txt (or *.xpi to retain compatibility with already existing message files).

The *FXTool Library NUMPlcMsgInterface *.*.*.* (NUM)* must be present in the project.

Usual location of a message file

*C:\Program Files\NUM\Flexium HMI\FXBrowser\Language*** (**: language version)

Example
Location of an English message file

C:\Program Files\NUM\Flexium HMI\FXBrowser\Language\en

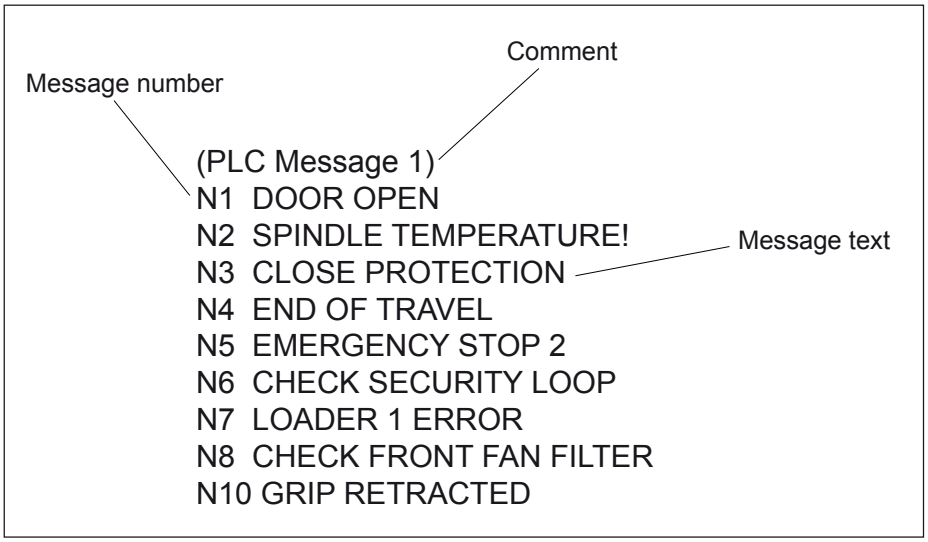
Configuration of the PLC message files in Language.ini

All message files must be defined in the configuration file Language.ini:
Please refer to Annex chapter A.

Example
Configuration of a PLC message file in the English section of language.ini

PLCErrorFile=C:\Program Files\NUM\Flexium HMI\FXBrowser\Language\en\filename.txt

Format of a PLC message file



Each PLC message can be made of several lines. The first line will be displayed in the upper window of FLEXIUM HMI (max 3 lines with scrolling)

The other lines will be displayed in the Diagnostic context.
Please refer to Flexium HMI Operator manual M00017.

3.5.1 Activating a message

The function “Insert” is used to initiate a message. This function also allows defining the reaction and acknowledge classes.

Flexium_RTS.PlcMsg.Insert(IN,Number,ReactionClass,ReceiptClass)

- *IN (Bool)*: if this bit is TRUE the message is activated
- *Number (UINT)*: number of the message to be activated (1 to 65535)
- *ReactionClass (BYTE)*: defines the reaction class of the message [1-8]
- *ReceiptClass (BYTE)*: defines the acknowledge class of the message [1-4].

3.5.1.1 The reaction class

Depending on the active reaction class, the PLC application can take the appropriate action.
Up to 8 different reaction classes can be defined.

The active reaction classes can be read with the function:
Flexium_RTS.PlcMsg.Q_Reaction[1-8]

Example

Action if reaction class is active.

```
IF Flexium_RTS.PlcMsg.Q_Reaction[4] THEN
    Flexium_NCK.W_ALLCNC.NoStopBlockEnd:=FALSE;
END_IF
```

3.5.1.2 The acknowledge class

The acknowledge class defines how a message of a certain class can be reset.
Up to 4 different acknowledge classes can be defined.

The active acknowledge classes can be read with the function:
Flexium_RTS.PlcMsg.Q_Receipt[1-4]

3.5.2 Reset messages

The following function is used to reset messages:

Flexium_RTS.PlcMsg.DeleteReceiptClass(IN,ReceiptClass)

- *IN (BOOL)*: if this bit is TRUE, all messages of the selected class are reset
- *ReceiptClass (INT)*: defines the acknowledge class of the messages to be reset [1-4].

Example

Action to reset messages of the acknowledge class 3 with an acknowledge button.

```
IF Flexium_RTS.PlcMsg.Q_Receipt[3] THEN
    IF AckButton3 THEN
            Flexium_RTS.PlcMsg.DeleteReceiptClass(TRUE,3);
    END_IF
END_IF
```

3.5.2.1 Self-resetting messages

The following function allows defining self-resetting messages.

Flexium_RTS.PlcMsg.SettingReceiptClass(ReceiptClass, SelfReceiptTime)

- *ReceiptClass(INT)*: defines the acknowledge class of the self-resetting messages [1-4]
- *SelfReceiptTime (TIME)*: >0: message is reset after this time in milliseconds
  0: reset by the acknowledge class.

3.5.2.2 Message state

To check if a certain message is active, the following function can be used.

return:=Flexium_RTS.PlcMsg.IsActive(Number)

- *return (BYTE)*: reaction class if the message is active, otherwise 0
- *Number (UINT)*: number of the message to be checked.

3.6 Extended Help

The Extended Help or Message Help allows assigning a supplementary help document to each basic message.

Extended Help is available as an HTML message or a JPG picture.
When an Extended Help message is available, a symbol (see figure below) will be shown on the left of the initial message and, in addition, the function key F3 (MsgHelp) will be available in Diag context.

To associate an extended help message to a PLC message it is just necessary to create an .HTM or .JPG file whose name will *'PLC_'* followed by the inital message number.

E.g. PLC_102.JPG : Extended help (graphic) associated with PLC message n° 102.
PLC_56.HTM : Extended help (Html) associated with PLC message n° 56.

3.6.1 Configuration of in Language.ini

All message files must be defined in the configuration file *Language.ini*:

Example
Configuration of a PLC message help file in the English section of language.ini:

UserMsgHelpLanguage=en

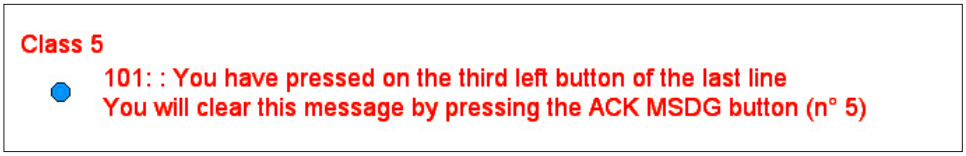
For more detailed informations please refer to Appendix A1.

Extended Help message location

*C:\Program Files\NUM\Flexium HMI\FXBrowser\MsgHelp\User** (**: language version)

Example
Location of an English message help file:

C:\Program Files\NUM\Flexium HMI\FXBrowser\MsgHelp\User\en.**



3.7 Special Settings

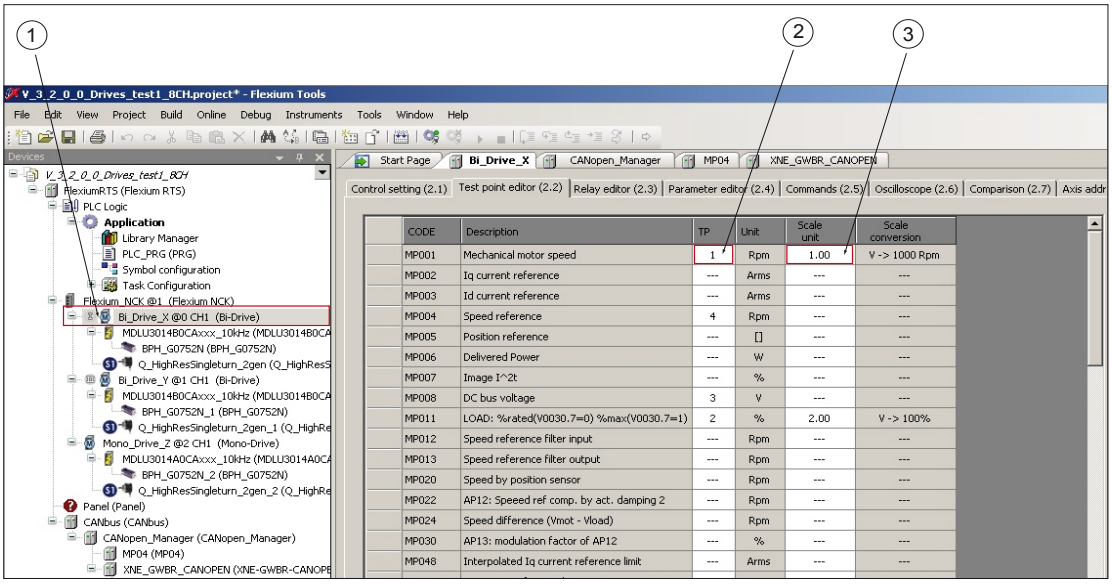
3.7.1 Drive Test Point settings

More than 100 measure points (MP...) are available on every DISC NT drive.

With the following FxTools function, it is possible to assign four of these measure points on each drive to test points that can be displayed in the HMI.

The test point display page is located in the context *"Diagnostic"* of the HMI.
Please refer to Flexium HMI Operator manual M00017.

If any analogue or digital test outputs of the drives are already used in the machine system, a modification of the test points can cause a malfunction of the machine.



Procedure to define a test point:

1. Double-click on the desired drive icon to open its properties page.
   Select the tab *"Test point editor 2.2"*.

2. Click on the measure point in the column *"TP"* to open the field.
   Enter a test point number (1 to 4) and press Enter
   This changes automatically the order of the former numbering.

3. If the scale unit has to be modified, click on the value in the column *"Scale unit"* to open the field. Enter a new scale unit value and press Enter

3.7.2 Spindle Load settings

The spindle load can be displayed on the main page of the HMI context *"Production"* and on the page *"Spindle"* of the same context. Please refer to Flexium HMI Operator manual M00017.

In order to display the spindle load, the load value itself has to be made available to the HMI. This can be done in two different ways:

By the PLC:

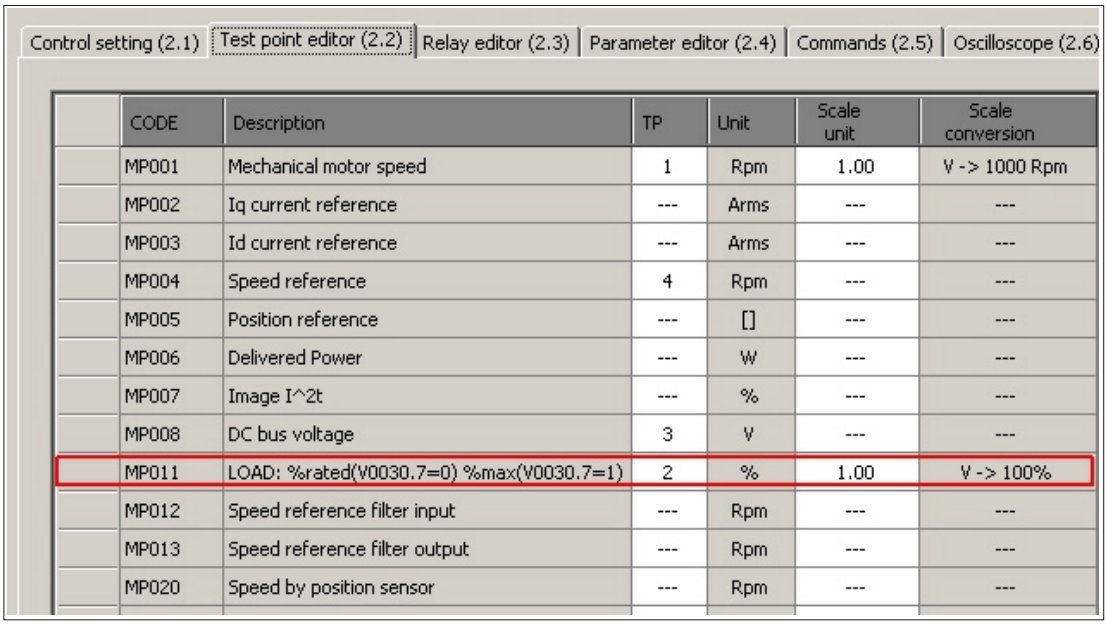
- Enable spindle load display by the PLC setting the variable *Flexium_NCK.WCNC.Spindle[n].Load.ByPlc = 1*

- Writing the value (0 to 150) to be displayed in the variable *Flexium_NCK.WCNC.Spindle[n].Load.Value*

Using a test point of the spindle DISC NT drive:

- Disable spindle load display by the PLC setting the variable *Flexium_NCK.WCNC.Spindle[n].Load.ByPlc = 0*

- Assigning the measure point MP011 of the spindle drive to one of the 4 available testpoint (1 to 4).



CODE	Description	TP	Unit	Scale unit	Scale conversion
MP001	Mechanical motor speed	1	Rpm	1.00	V -> 1000 Rpm
MP002	Iq current reference	---	Arms	---	---
MP003	Id current reference	---	Arms	---	---
MP004	Speed reference	4	Rpm	---	---
MP005	Position reference	---	[]	---	---
MP006	Delivered Power	---	W	---	---
MP007	Image I^2t	---	%	---	---
MP008	DC bus voltage	3	V	---	---
MP011	LOAD: %rated(V0030.7=0) %max(V0030.7=1)	2	%	1.00	V -> 100%
MP012	Speed reference filter input	---	Rpm	---	---
MP013	Speed reference filter output	---	Rpm	---	---
MP020	Speed by position sensor	---	Rpm	---	---

For the description of the test point assignment, please refer to the previous pages *"Drive Test Point settings"*.

Page deliberately empty.

Page deliberately empty.

Summary

Page deliberately empty.

4 Flexium functions

4.1 Automatic homing

For axes not fitted with absolute measurement systems, a specific homing macro %9990.9 is included in the CNC firmware. This macro recognises the type of axis (linear, modulo, …) and include the processing for synchronised axes (one Slave only).

This macro is not editable nor visible and is called via G77 H9990.9 or G199.

Other files

- %9990 : Homing routine in Home mode (order of execution)
- %10199 : Homing routine for call in MDI, Auto or by G199

Two parameters must be passed before calling %9990.9:

- L0: Axis number (0 .. 8 respectively X, Y, Z, U, V, W, A, B, C)

- L1: Maximum retraction value. In case the axis is already on the home switch the retraction will not exceed the L1 value. This is to prevent an over travel in case the limit switch is defective. L1 should be positive in all cases.
  Explicit programming of L1 is not mandatory, if its value is 0, then retraction is unlimited.

4.1.1 Homing mode

On machines with a single axis channel, homing is performed on each axis separately. %9990 will specify the order of execution and speed.

On machine with multiple axes channels, each channel should have a different program %9990.i, I being the channel number.

Order and speed are specified in each %9990.i, moreover synchronisation points if required must be programmed.

4.1.1.1 Example of an homing program for multi channel

```
%9990.2                    Example for Channel 2
IF [.RG80] = 999 THEN
   G77 H9990.9             Call by G199
ELSE                       Call by Start in Home mode
   L0=2 L1=20              Z axis retraction value limited to 20 mm
   F2000 G77 H9990.9       Execution

   G78 Q1 P1.1             Wait until channel 1 has executed the first operation

   L0=0 L1=25              X axis, retraction limit 25 mm
   F3000 G77 H9990.9       Execution
   L0=1                    Y axis
   G77 H9990.9             Execution
ENDI
M2
```

The program for a single channel machine is quite similar – synchronisations points are not required. Obviously the synchronisation as described is an example and should be adapted to the each machine.

4.1.1.2 Automatic Homing Procedure in Homing Modes

A multichannel CNC must be in common group state. Homing is not effective in independent channels mode.

The user selects homing mode and presses the Cycle start button.
Programs %9990.i (or program %9990) are (is) started.

Homing is performed on each axes requested assuming they are declared in the channel and measured.

The operator is informed of the progress via messages:

X : MOVING OFF AXIS REFERENCE SWITCH
X AXIS REFERENCE CYCLE IN PROGRESS
X AXIS REFERENCING COMPLETE

If homing was not possible, the cycle is stopped. The operator acknowledges the message below by pressing <Return>

X AXIS REFERENCING FAILED. QUIT :

The following message is then displayed:

STOP AUTOMATIC REFERENCING CYCLE

In case of error the following data should be controlled :
P2, P9, P17, P16, P15, P1.

For more detailed informations on Pxx please refer to Flexium Parameters manual M00021.

4.1.2 Call by function G199: AUTO or MDI Mode

The following block can be executed in AUTO or MDI mode as well as in SINGLE and RAPID:

[F…] [L1 = …] G199 <axes list >

where <axes list> = {X, Y, Z, U, V, W, A, B and/or C}

- Homing is performed on all axes programmed in the list in the order XYZ UVW ABC.
- The feed rate as well as the retraction distance can be programmed.
- Homing is performed on the measured axes declared in the group.
- Macro %10199 is activated and calls programs %9990 (or 9990.i).

- Except in the case of a machine error, a return is always made to the calling program, which is responsible for checking whether homing was effective.

Example: Homing of the C axis at 2000 deg/mm in MDI mode

F2000 G199 C <Enter> and Cycle start

CAUTION

Regardless of the order of the axes programmed after G199, the movements required for homing are carried out by the macro %10199 in the order XYZ, then UVW and finally ABC.delivered.

4.1.3 Case of axes synchronised by machine parameters

Regardless whether it is called by function G199 or in homing mode, macro 9990.9 detects the presence of axes synchronised by machine parameters.

If only one axis is slaved to the programmed axis, homing is conducted jointly on the master and slave axes.

- Macro 9990.9 assumes that the master and slave axes have the same characteristics: type, homing direction, travel, switch presence, area covered by the switch, etc.

- Parameters P16 (reference switch position) may be slightly different for the two axes and therefore the encoder marker pulses may be reached “almost together” in the same movement.

- After homing the master axis, the macro continues movement over a length equal to the difference between P16 for the master and slave axes.

At the end of the cycle, the macro checks that the two axes are in the same state.
If homing is complete on both axes, the cycle ends. If not, the following message is displayed:

MASTER/SLAVE AXIS REFERENCE INCOMPLETE

4.1.3.1 Case of several slave axes

If several axes are slaved to the programmed axis, the macro displays the following message at the beginning of the cycle:

MACRO 9990.9; ONLY CONTROLS ONE AXIS

In such exceptional configuration, program 9990.9 will require to be adapted.

4.2 Auxiliary axes

The function Auxiliary axes is intended to control DISC NT axes independently of the machining. These axes are connected on the NCK but are not under direct control of a part program.

This functionality is additional to the CAN PLC controlled axes and provides an alternative solution for tool changer, palletisation system or else.

Being connected to the NCK, these axes share some features with the CNC interpolable axes:
- Axes are grouped per channel, they interpolate within a channel
- There is a maximum of 9 axes par channel
- The number of auxiliary channels is at most 8 minus the number of CNC channels considering that there is at least one CNC channel per NCK
- The maximum number of axes (auxiliary and CNC) connected to one NCK is 32
- The auxiliary axes are under control of an ISO program.

4.2.1 Operating auxiliary axes

- Auxiliary axes can operate in continuous mode independently of the main NCK mode.
- All auxiliary channels are fully independent.

4.2.1.1 Reset

- Auxiliary channel can be reset via the channel specific reset request.
- An CNC reset does not reset the auxiliary channels.

4.2.1.2 Homing – Jog

It is possible to control auxiliary axes in Jog or Home mode but this can only be done if the main CNC mode is Jog or Home and the auxiliary channel declared as non enabled.
(see PLC exchange area)

4.2.1.3 Programming

Auxiliary axes are controlled via an ISO program.
However the following functions are not accepted:
- M01: Optional stop
- M12: Programmed intervention
- G75: Emergency disengagement.

4.2.1.4 ISO Program

Auxiliary axes are under control of one single program “%9998.i”, “i“ being the auxiliary channel number. This program must be stored in the OEM or USER zone of part programs.
The program is automatically selected when a Channel reset command is issued.

4.2.2 PLC exchange zone

The specific functions for auxiliary channels are:

...WCNC.Channel[x].ChannelEnable	Information required, should be set except to control the auxiliary axes in Jog or Home mode
...WCNC.Channel[x].PlcAxisMode	Boolean information : False = Automatic True = Block by block
...WCNC.Channel[x].Start	The channel must be enabled for the Start to be active
...WCNC.Channel[x].Reset	Should be sent to activate the auxiliary channel.

The signification of the other flags is identical as for the regular channels.

4.3 Independent Channels

The function *'Independent Channels'* gives the possibility to control each channel in a different mode, each channel with its own commands and its own part program.
The PLC Flags are at the channel level. They are described in the previous chapters of this document. For memory they are the following:

Read functions	Write functions
Name M01 BlockSkip Intervention ProgramActive ProgramStop CncError AxisNotInPosition EmergencyRetract AxisRecall Start Stop Reset CycleNumber TreadCutting Rapid MFunctionNoAck Mode MFunctionAck DecodeM ToolNumber	Name PlcAxisMode Rapid AckMFunction PlcCallSubroutine PlcInterrupt ChannelEnable EndExtMovement FeedEnable M01 BlockSkip EmergencyRetract AxisRecall Start Stop Reset FeedOveride Mode ProgramNumber ReducedSpeed ManualSpeed

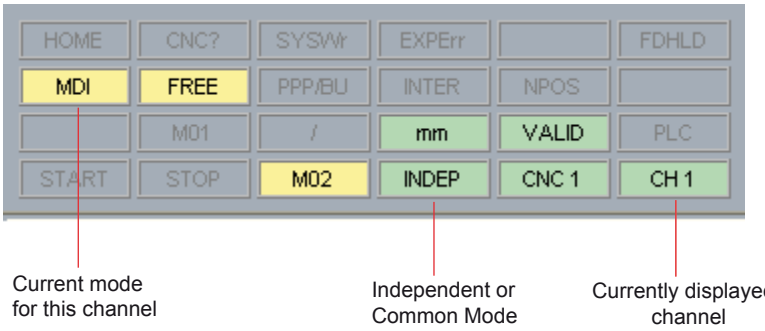
- Home mode as well as Backtracking is not possible in *'Independent Channels'* mode.
- Homing the axes requires to be in common mode.

The mode Independent channel is selected via the flag *...W_ALLCNC.IndependentGroup* which to be taken into account require a PLC request reset *...W_ALLCNC.Reset*.

The status *'Independent Channels'* is confirmed via the value 16 in *...RCNC.General.Mode*.
It is also shown in the Flexium HMI status window by the *word INDEP* as opposed to *COMM* for common channel.

Current Program and Mode can be selected by the PLC at the Channel level
e.g. … *WCNC.Channel[x].Mode* := EnumMode.MDI

They can also be selected via the HMI, in such case the selection will be for the channel currently displayed.
Current program is selected in MDI mode by typing *%<prog number> followed by <Enter>*.



4.4 Duplicated and synchronised axes

4.4.1 Principle

The duplicated and synchronised axes function links a “Slave” axis with a “Master” axis.
The system calculates the master axis reference variation and adds it to the Slave axis reference.

Several Slave axes can be driven by the same Master axis,
The Slave axes make the same movements as the Master axis. A Slave axis may however move symmetrically with respect to the Master axis.

Duplication is done without feedback between Slave and Master (axes get the same command).

Synchronisation adds a feedback to compensate for difference in reaction as well as a control of the error between Slave and Master. When synchronisation is active the symmetrical move is not authorised anymore.

4.4.2 Implementation

Implementation consists of:

- Defining the couplings (i.e. assigning a Master axis to a Slave axis),
- Enabling the couplings (i.e. choosing the type of duplication).

The Slave axis must have the same characteristics as the Master axis, i.e:

- Speed
- Acceleration
- Position loop time constant.

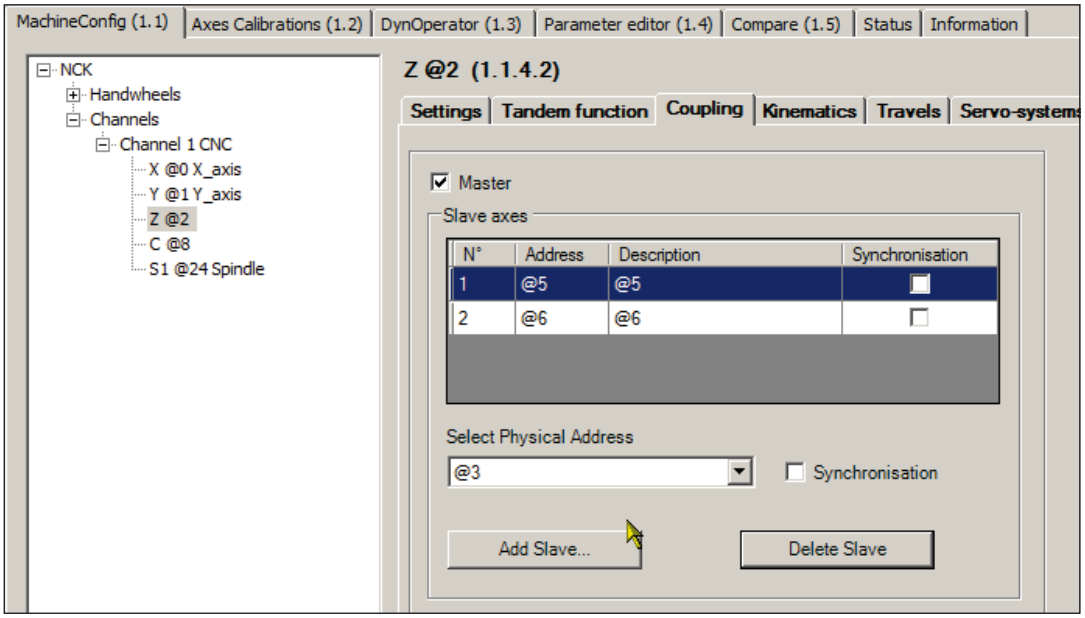
4.4.2.1 Defining Axes Couplings

- Coupling are defined by giving for each axis its *'Master'* address or -1 if there is no Master.
- They are defined by machine parameter P27 or external parameters E94100 to E94131.
- Couplings defined by machine parameters can not be changed by E Parameters

Example

```
E94105 = 2
E94106 = 2
```

Axes 5 and 6 are both Slaves of Master axis 2
This can also be done via the tab *'Coupling'* of each master axis within Flexium Tools



Coupling with Symmetry

For such Duplicated axes, the Slave axis moves symmetrically with respect to its Master axis.

Coupling with Synchronisation

For such Duplicated axes, the Slave axis is synchronised with respect to its Master axis.
Synchronisation consists of measuring the difference in movement of the Slave axis with respect to the Master axis and adding a fraction of it into the Slave reference.
Coupling cannot be both synchronised and symmetric.

Synchronisation Error is the difference between the Master axis measurement and the Slave axis measurement.

Two thresholds are defined for checking this error:

- Threshold 1 (default 160microns): the maximum permissible error after synchronisation. Above this value, the system generates a message but allows resynchronisation of the Slave axis with its Master axis.
- Threshold 2 (default 600microns): the maximum error above which the system generates a message and inhibits synchronisation.

The error generated by overrun of thresholds 1 and 2 clears the CNCReady signal and displays ERROR 39. These thresholds are defined by machine parameter P24 and cannot be modified by programming.

4.4.2.2 Enabling axes duplication

Enabling in Automatic Mode E960xx

This is defined by E parameters E96000 to E96031 respectively.
Coupling is immediately enabled by a nonzero value in the slave relative parameter.

Enabling Synchronisation E962xx, P28

This is defined by machine parameter P28 (immediately at power on) or by E parameters E96200 to E96231.
A nonzero value in the slave relative parameter validates the synchronisation.

Enabling in HOME, JOG modes or Cycle hold E961xx

This is defined by E parameters E96100 to E96131.
Coupling is immediately enabled by a nonzero value for the slave relative parameter.

Enabling Symmetry E963xx

This is defined by E parameters E96300 to E96331.
A nonzero value for the slave relative parameter indicates that symmetry is active on this coupling.

For more detailed informations on Exxxxx please refer to Flexium Parameters manual M00018.

4.4.2.3 Synchronisation

Synchronisation differs according to whether it is enabled by machine parameter P28 or external parameter E962xx.

4.4.2.4 Enabling by Machine Parameters

Synchronisation is enabled immediately at power on and cannot be cancelled by programming.

In Manual
Action on the *JOG control* of the Master axis moves the Slave axis synchronously with the Master axis. Action on the *JOG control* of the Slave axis is inoperative.

In Homing
A homing simulation from the keyboard or a homing request for a Slave axis are inoperative.
A programmed (G199) or manually controlled homing request for the Master axis is also applied to all its Slave axes, the synchronisation error detection threshold 1 is inhibited.

When all the axes have reached their home position and if the synchronisation error is below threshold 2 the system will realign the coupled axes by gradually setting the Slave axis reference equal to the Master axis reference, then it initialises the control of the synchronisation error.

This requires the PLC flag *...W_ALLCNC.SynchAxisPowerOn* to be set.
This flags should reflect the fact that the drives are energised and able to control the motors.

In MDI and AUTO Modes
The coupling cannot be modified, programming the Master axis moves the Slave axis synchronously. Programming a Slave axis is prohibited.

Synchronisation Error
An overrun of synchronisation threshold will clear the *CNCReady* signal and generate ERROR 39.

If the error is below threshold 2, an CNC reset will force the system to resynchronises the coupled axes by gradually setting the Slave axis reference equal to the Master axis reference, **assuming that the signal NCK.W_ALLCNC.SynchAxisPowerOn is ON**.

If the error is above threshold 2, the error cannot be cancelled. It means the machine is excessively skewed and the axes must be repositioned manually then the system must be reset.

4.4.2.5 Enabling by External Parameters

The Slave axis and Master axis references can be different. The system performs synchronisation by holding: Slave axis reference - Master axis reference = constant difference

At Power On
The axes are independent. When an external parameter E962xx = will be set to 1 is programmed, synchronisation of the corresponding coupling will become active.

In Manual
If synchronisation is enabled, the movements of the Slave axis are synchronous with the Master axis. Otherwise, they are independent.

In Homing
Before enabling synchronisation, homing is carried out independently on the axes. Homing simulation from the keyboard is authorised.
Once the synchronisation is enabled, if the axes were not homed, only homing of the Master axis is required (homing from the keyboard is refused). Since a difference is allowed between the references, there is no resynchronisation.

In such a case, the current position and the software limits of the Slave axis are ignored.

In MDI and AUTO Modes
It is possible to modify the coupling by programming, e.g.: disable, re-enable, change of Master axis.

Synchronisation Error
An overrun of synchronisation threshold will clear the CNCReady signal and generate ERROR 39.

Cancelling the Synchronisation Error if the error is below threshold 2 after an CNC reset by the operator, as soon as signal *...W_ALLCNC.SynchAxisPowerOn* will be is set, the system resynchronises the coupled axes by gradually setting the Slave axis reference equal to the Master axis reference.

If the error is above threshold 2, it cannot be cancelled. The axes must be repositioned manually then the system must be reset.

4.4.3 Tuning

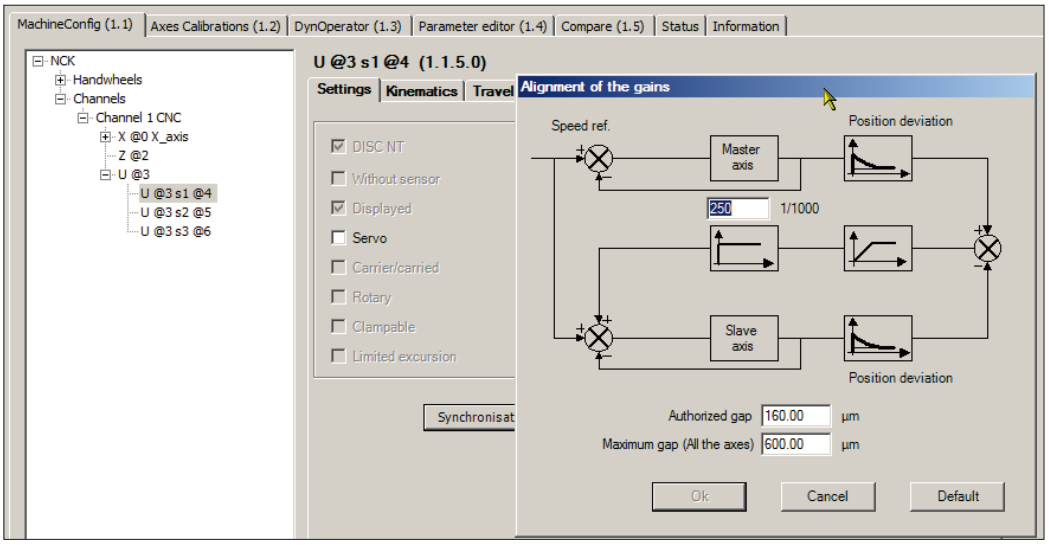
- **Parameter P24** is used to tune the function.
  Please also refer to Flexium Parameters manual M00021.

- **Words 00 to 31**
  Threshold 1 for each axis (default 160).
  A fault is generated if this threshold is exceeded but resynchronisation is possible.

- **Words 32 to 63**
  Fraction of the sync error to be reinjected in the slave reference (default 250/1000)

- **Word 64**
  Maximum synchronisation error (default 600µm).
  No possibility of resynchronisation if exceeded.

In Flexium Tools, for slave axis settings, the button *Synchronisation* gives access to these values:



4.4.4 Restrictions

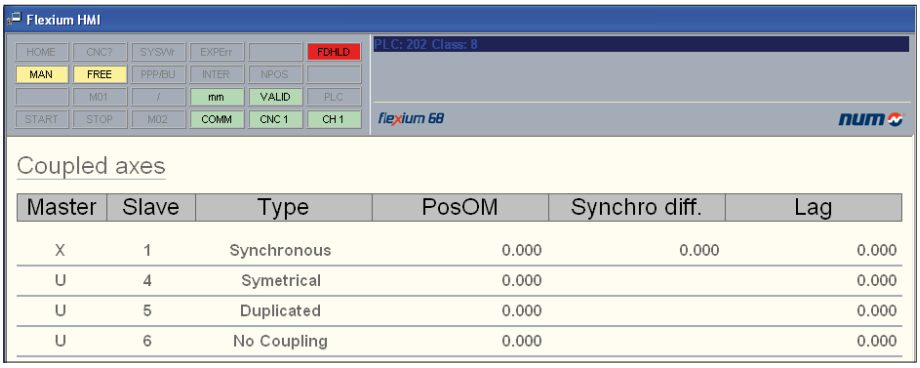
The following conditions and restrictions apply to parameter setting and programming with axis couplings:

- **An CNC reset does not cancel the couplings programmed**
- A coupling can only be redefined if it is disabled (no on-the-fly change of Master axis).
- An axis cannot be its own Master.
- A coupling cannot be enabled for duplication, synchronisation or symmetry unless it is already defined.
- A coupling defined by machine parameter P27 cannot be modified but can be enabled or disabled by programming.
- A coupling enabled by machine parameter P28 cannot be modified and must have been defined by parameter P27.
- The characteristics of a coupling cannot be modified unless the coupling is disabled.
- Synchronisation can only be enabled if the Slave axis does not belong to any channel: either the axis has been freed by programming E7x005 = -1 (x = programming the address of the axis), or the corresponding word of parameter P9 contains FF.
- It is recommended to program function G09 (accurate stopping at end of block before going to next block) on the movements preceding and following a programmed synchronisation.
- A programming error generates ERROR 92.

For more detailed informations on Pxx please refer to Flexium Parameters manual M00021.

4.4.5 Display

When coupled axes are defined, the function key F9 , *"Dupl Axes"* will be active.
Pressing this key will display the current status of the duplication:



The status in manual mode is the following (it can be different in auto mode):

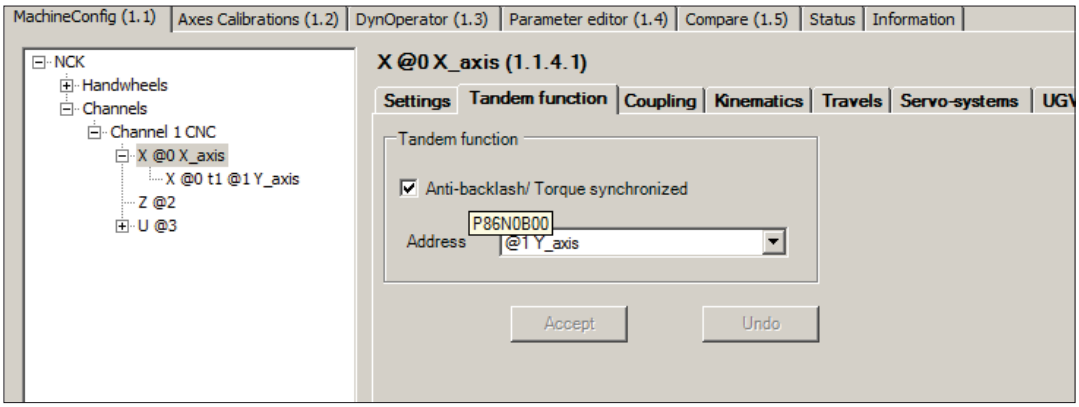
- Axis 1: is synchronised to axis X : the synchronisation error is Zero
- Axis 4: is duplicated with symmetry to axis U
- Axis 5: is duplicated with axis U also but no symmetry
- Axis 6: is currently independent.

4.5 Tandem function

This is a drive feature.

For more detailed informations please refer to the NUMDrive parameters manual AMOMAN007 chapter relative to AP03.

This function is validated by the tab Tandem in the axes configuration of Flexium Tools.



4.6 Spindle synchronisation

The system offers the possibility to synchronize one or several Slave spindles on a Master spindle. The functionality is achieved for a speed controlled Master spindle.
Synchronisation requires the signals:

…W_ALLCNC.SynchAxisPowerOn to be set and the related spindles not to be locked

...WCNC.Spindle[x].Lock, PLC controlled

...WCNC.Spindle[x].PlcSpeedControl or limited in speed *…WCNC.Spindle[x].ReduceSpeed*.

Synchronisation is based on E parameters, some of them could be preset in the machine parameters.

4.6.1 Parameters

Acceleration and speed		
E90330 - 90333 **E90390 - 90393**	Spindle acceleration Max speed for the selected gear (Not required for synchronization)	P42 P46
Position		
E79001 **E90200 - 90203** **E90300 - 90303**	Channel specific spindle reference Spindle 1 to 4 references Positioning speed	 P43
Measures		
E79000 **E90100 - 90103** **E90110 - 90113**	Channel specific spindle actual position Spindle 1 to 4 actual position Spindle modulo (counts/turn)	 P40 extended to the next power of 2
Positioning		
E90310 - 90313 **E90320 - 90323** **E93524 - 93527**	In positioning or synchronization window Gain factor for indexing or synchronization Spindles in synch. (Yes = 1, No = 0)	P44 P45
Offset, Reference		
E95024 - 95027 **E91124 - 91127**	Reference offset Spindle referenced (when = 0)	P16
Master / Slave assignement		
E94124 - 94127	Master to slave assignment (Eg spindle 1 = Master, 3 = Slave E94126 = 24	
Synchronization parameters for the Master spindle		
E90360 - 90363 **E90370 - 90373** **E90380 - 90383** **E95224 - 90227**	Acceleration is function of speed, No = 0, Yes = 1 Max speed at max torque Max speed at max power rpm Synchronisation error	
Remark: meaning of E90360 to 90363 - Value < E90370 : Then acceleration = E90330 - E90370 < Value < E90380 Acceleration = E90330 * E90370 / Value		
Synchronization parameters for the Slave spindle		
E90360 - 90363 **E90370 - 90373** **E90380 - 90383** **E98024 - 98027** **E95224 - 90227**	Acceleration vs speed adjustment factor in 1/1000 Max speed at max torque Max speed at max power Integration time (ms) Synchronisation error	

4.6.2 Activating the synchronisation

Synchronisation is activated by loading the address of the Master spindle into parameter E94124 (or 5, 6, 7) of the Slave spindle.

Example: Spindle 3 = Master, 0 and 1 = Slaves

```
E94124 = 27
E94125 = 27
```

Synchronisation is refused if the spindles are not homed (error 94).

The spindle must be in rotation before requesting synchronisation this will define the synchronisation direction. Forward if they rotate in the same direction, reverse if not.

4.6.3 Setting procedure

- Check that the speed control is accurate for all spindles.
- Reduce as much as possible the spindle acceleration ramp generated by the drives.
- If possible choose for the slave spindle the one with the highest dynamic.
- Adjust parameters E9037x and E9038y according to the motor characteristics.
- Accurately adjust the spindle acceleration (E9033x).
  If the master spindle has a higher dynamic than the slave set them both at the Slave value.
- Adjust proportional gain on the Slave.
  - E9032x = 500 to start with
  - E98024 (or 5, 6, 7) = 0
  - E9036x = 1000

At speeds about 1000rpm, adjust E9032x to optimize the synchronisation error behaviour.

- Then starting with E98024 (5, 6, 7) at about 500 adjust it until a slight overshoot with no oscillation of the synchronisation error.
- Check at high speeds. If there is an oscillation of the synchronisation error increase the value of E9036x of the slave.

For more detailed informations on Exxxxx please refer to Flexium Programming manual M00018.

4.6.4 Example

```
E94124 = - 1          Assign Master / Slave
E94125 = - 1          Assign Master / Slave
```

- **Start spindle synchronization**

```
M65 M67 M3 S1000
M64 M66 M3 S1000 EC0
```

- **Spindle synchronization**
  Define parameters of the Master

```
E91024 = 0            Speed command
E90320 = 1000         Gain setting P45
E90330 = 5000         Acceleration setting
E90360 = 1            Acceleration function of speed
E90370 = 1500         Rpm at max torque
E90380 = 6000         Rpm at max power
```

  Define parameters of the Slave

```
E91025 = 0            Speed command
E90321 = 1000         Gain setting : P45
E90331 = 10000        Acceleration setting : P32
E90361 = 1000         Acceleration vs speed factor
E90371 = 4000         Rpm at max Torque
E90381 = 6000         Rpm at max providers
E98025 = 000          Integration factor must be zero before syncing
E94125 = 24           Assign Master / Slave
```

- **Wait for synchronization**

```
N10 G4 F1.5
G79 E93525 = 0 N10
E98025 = 500          Set integrator factor
```

4.7 Axis / Spindle commutation

4.7.1 General

The Axis / Spindle commutation function is required for commuting on the same motor and same drive two “Axes” with different dynamics and kinematics characteristics.

The Flexium system offers the possibility to use up to four motor/drive pairs to control alternatively an axis or a spindle, (e.g. C axis on a lathe).

Overall, the process is:

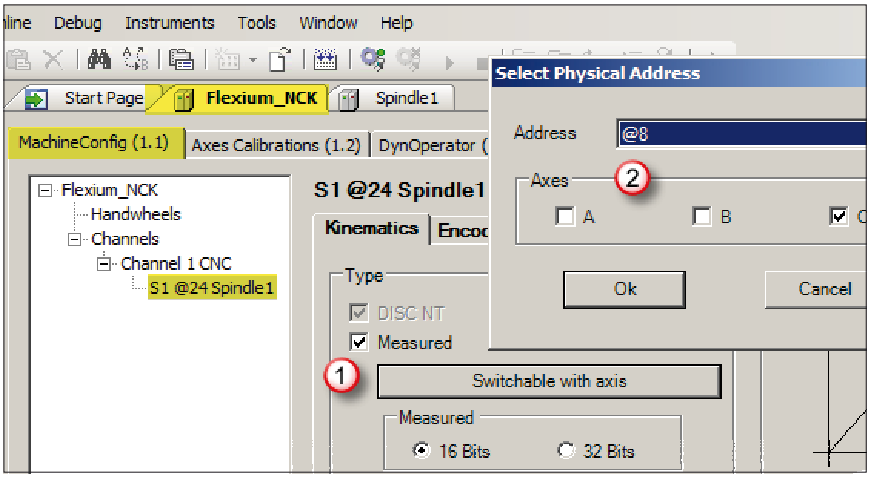
- Declare the spindle (P6)
- Declare the axis as visible, modulo and measured (P0, P1, P2)
- Declare axis and spindle as permutable (P70 N8)
- The drive address should be set at the spindle address (@24 to @27)

- Create a macro to activate the permutation and, according to the kinematics used, select the correct set of speed and position loop parameters.

For more detailed informations on Pxx please refer to Flexium Parameters manual M00021.

4.7.2 Operation with Flexium Tools

The spindle drive should be inserted in the Device tree.



- Select *Flexium NCK editor – Tab MachineConfig – Spindle*

- Declare the spindle as Measured (1)
  - The button ‘*Switchable with axis*’ appears
  - Press this button
  - Select *Name* and *Address* of the associated axis (2)

- The additional axis is created
  - Declare it as visible
  - Select the measure type (16 or 32 bits)

- Parameterization is finished.

4.7.3 16 or 32 Bit measurement

By default the position measurement is on 16 bit, therefore, in most case it will be necessary to home the associated axis at each revalidation.

Homing will be necessary if at the max spindle speed the associated axis position variation exceeds 32000 counts/per sampling period, which is about 2700 rpm with a 2ms scan and 1/1000° resolution.

It is possible to extend the measurement resolution on 32 bits. This gives the possibility to use one single high precision encoder on the spindle assuming the kinematic is accurate enough.

Using 32 Bit resolution is done in two steps:

- Declare in P70 N7 the spindle associated axis as 32 Bit resolution
  (bit cleared = 32 Bit measurement).

- Set P40 in accordance:

  In case of 32 Bit measure P40 does not represent the spindle measure counts.
  Its value is the axis modulo (given through P4 N4) divided by an integer

  - P40 should be large enough in order that the number of spindle increments divided by P40 does not exceed 32000 at max speed.

  - It should be smaller than 32767

  - Several different values are valid.
    A too small value will limit the precision at low speeds while a too high value will prevent keeping up at high speeds. Except particular cases, a value of 10000 for P40 (2500 in the graphic editor) covers most cases (more than 10000 rpm at 1/10000° resolution).

  - Check E parameter E9012b (b = spindle number 0 to 3).
    If it is equal to -1 then the value of P40 is not correct.
    This parameter defines the coefficient conversion.

4.7.4 Programming

Switching drive command

- `E9422x = yz` (yz = axis address)
  - `X = 4, 5, 6, 7` According to the spindle number
  - `YZ =` Associated axis address

e.g. `E94224 = 24` Control the spindle
`E94224 = 08` Control the C axis

Selecting a set of drive parameters

- `E3532x = ab` (x = 4, 5, 6, 7 for spindle 1, 2, 3, 4) – (ab : parameter set)
  - a and b values can be 0 or 1
  - a refers to the drive parameter V261: position loop, value 0 means first set of value, value 1 second set of values.
  - b refers to the drive parameter V260 : speed loop, value 0 means first set of value, value 1 second set of values.

e.g. : `E35325 = 01`
Use first set (a=0) of position loop parameters and second set (b=1) of speed loop parameters for drive at @ 25.

For more detailed informations on V260, V261 drive parameters please refer to NUMDrive Parameters manual AMOMAN007.

4.7.4.1 Examples

Switching from spindle1 (@24) to C axis (@8)

```
E78005 = 8          C axis control
E91008 = 1          C axis measured
E94224 = 8          Spindle drive controlled by C command
E35324 = 11         Select second set of position and speed parameters
$ C axis mode       Operator message
M72                 Indicates C axis mode (should be PLC processed)
```

Switching from C axis to spindle

```
E91008 = 0          C not measured
E78005 = -1         C not controlled anymore
E94224 = 24         Spindle drive controlled by S value
E35224 = 0          Select first set of position and speed parameters
$ Spindle mode active   Operator message
M71                 Indicates Spindle mode
```

4.8 Delayed Power down in case of error

This function is used when a machine shut down due to a servo error may have dangerous consequences (e.g. Grinder, Gear Hobber…).

It is intended to mask the error long enough to permit disengagement and return to a safe position.

The functionality is base on the following E parameters:

E93700 to E93731: Axis experiencing measurement error or excess following error

Boolean parameters related to axes 0 to 31. A parameter is set to indicate a measurement or excess following error on the corresponding axis.

- These parameters are read-only and are cleared only on a reset.
- These parameters can be read by dynamic operator.
  In this case, they consist in a list of 32 bits, where bit 0 addresses axis 0, bit 1 axis 1, etc. In this case, the entire list is addressed by E93700.

E92100 to E92131

Boolean parameters related to physical axes 0 to 31.

- These parameters are read/ write by program and dynamic operators.
- When accessed by dynamic operators, they consist in a list of 32 bits, where bit 0 addresses axis 0, bit 1 axis 1, etc. In this case, the entire list is addressed by E92100.

- When a measurement or excess following error is detected
  - This error is immediately stored in E937xx
  - If the corresponding parameter E921xx is set, then the error 40+x or 210+x is not issued and the *…RCNC.General.CncReady* flag remains high. In case an excess following error occurs, the speed setting calculated during the previous sampling period is maintained.

- After analysis of the possible errors causes, the bits set in list E93700 (axes in error) are reset in list E92100 (error masks).

- The dynamic operators are executed

- At the next scan, if an axis reports an error in the list 93700 and this error is not masked in the list 92100 then a machine error is generated and the *...RCNC.General.CncReady* flag is cleared

Since the dynamic operators operate before the above mentioned step, they can delay the error generation by maintaining the faulty axis in list 92100. They can also monitor the following error.

Various actions are possible:

- Transmission to the PLC of the list of axes in error.
- Forcing the speed reference to zero for certain axes and braking at controlled speed on others using the channel feed rate override (E79002).
- Direct write on a PLC line for fast intervention of an external device, etc.

4.8.1 Example

Consider a CNC channel including two axes with addresses 0 and 1 and a auxiliary channel including two axes U and V with addresses 2 and 3.

If a excess following error occurs on an auxiliary axis, we want to force the speed setting for the auxiliary channel to zero and if the CNC axes are moving, request a stop of these axes by setting the feed rate override to zero, and then wait for the action to be performed before shutting down power.

4.8.1.1 CNC channel program

```
VAR
  [over]=79002
  [V_cn]=79004
  [lx_AP]=80000
  [lx_ERR]=93700
  [lx_CTRL]=92100
  [zero]=81000
ENDV
```

```
E[lx_AP]=12                          Addresses 2 and 3 = 1100 in binary = 12
E92102=1 E92103=1                    First initialisation for monitoring axes 2 and 3
E[zero]=0

O1=9 E[lx_CTRL] / E[lx_AP] / / 5 /   Test for fault on PLC axis – Jump to 5 if none
O2=5 E[potar] / E[zero]              Set CNC override to zero
O3=9 E[V_cn] / E[zero] / / 5 /       Wait If speed not zero
O4=5 E[lx_CTRL] / E[lx_AP]           Reset mask lx_CTRL when speed Zero
                                     mask reset ➔ CNCReady will be cleared
```

4.8.1.2 Auxiliary channel program

```
VAR
  [lx_ERR]=93700
  [zero]=81000
ENDV

E[zero]=0

O5=9 E[zero] / E[lx_ERR] / / 8 /     Test for error on axes – Jump to 8 if none
O6=2 E95002 / E90002 / E73000        Force auxiliary axes speed to 0
O7=2 E95003 / E90003 / E74000        Idem for axis 3
```

4.9 Acceleration with jerk control

This type of acceleration type also named gradual or trapezoidal acceleration is based on E Parameters E11013, E32006 and E32008.

Gradual acceleration allows faster speed build-up and stopping, especially at high speeds (e.g. positioning blocks). Gradual acceleration uses constant jerk (acceleration variation), whereas more classical saw-tooth acceleration uses a jerk function of the speed level (large for high speed variations).

However, for irregular maximum speed profiles (several adjacent curve with radius discontinuities), saw-tooth acceleration allows smoother speed variations.

For more detailed informations on Exxxxx please refer to Flexium Programming manual M00018.

4.9.1 Declaring the type of acceleration

E11013
Gradual acceleration is enabled by programming E11013=-1
(Value=0 means constant acceleration; value=+1 means saw-tooth acceleration).

The setting of this parameter is maintained after a reset. At power on, this parameter is set for saw-tooth acceleration (E11013=1) if enabled by machine parameter (P7, word 1, bit 5).

E32006
Parameter P53 (E32006 by programming) is used to adjust the settings.
This value in ms gives the time to achieve maximum acceleration (Jerk= Gmax/P53). It is also the speed build-up time for very small speed levels, when speed variation <= 1 G * sampling period.
With gradual acceleration, the setting of P53 is limited to 200 sampling periods.

Parameter P53 is common to all axis groups, whereas the setting of E11013 can be different for each group.

E32008
The type of acceleration described above generates a constant jerk. A variation of this function is created to define a trapezoidal jerk.
It is based on:

E32008: Time (in ms) to reach a constant jerk

The total acceleration time is defined by P53 or E32006.

Trapezoidal jerk is declared by:
E11013= -1,
E32006 = xxxx (acceleration rise time in ms)
E32008 non zero

Parameter E32008 is set to zero at power on, it is programmable and its value is maintained after a reset. The setting of E32008 is limited to half the setting of E32006.

E32008=0 (default value): acceleration with constant jerk.
E32008= P53/2: saw-tooth jerk.

4.9.2 Constraints related to gradual acceleration

4.9.2.1 Sequenced thread cutting (G38)

Trapezoidal acceleration cannot be used with sequenced thread cutting.
During analysis of function G38 or G38+, if trapezoidal acceleration is present, it will be disabled (E11013 forced low).

It is not re-enabled automatically and must be reprogrammed by E11013= -1.
If trapezoidal acceleration is present during a G33 thread cutting cycle, it is cancelled during the cycle, and automatically re-enabled at the end.

4.9.2.2 5-axes polynomial interpolation

Speed is managed differently in the 5-axis smooth polynomial interpolator than in the linear and circular interpolators.

In a sequence of blocks including linear axes and rotary axes with large movements on the rotary axes compared with the linear axes, when the classical interpolated blocks are followed by polynomial interpolated blocks, then a speed of zero is forced at the end of the block preceding the first polynomial block.
Similarly, when a polynomial block is followed by a line or a circle, a speed of zero is forced at the end of the last polynomial block.

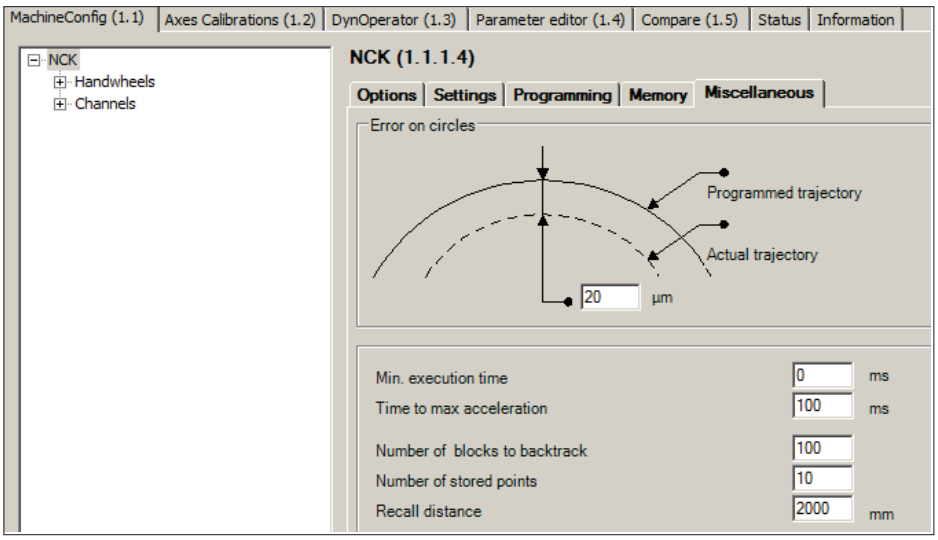
4.10 Backtrack and resume

This function optional and available on the first channel only can be activated after a feedhold.

Backtrack along the programmed trajectory will be executed by a maintained action on *..WCNC.General.BacktrackBackward* .

Resume by a maintained action on *...WCNC.General.BacktrackForward* .
The flag *..RCNC.General.BackForwardOnPath* indicates the status of the function.

In order to be able to backtrack, the system will memorise a certain number of trajectory blocks. This number is defined in parameter P114. It can be set via FlexiumTools.



Backtracking and return will be executed at the programmed speed. Resume can be executed in Auto, Block by block or Rapid and a change of mode is possible during this phase.

Please note that the T, S and M function are not memorised and will not be taken into account during the backtracking. Even though it is feasible backtracking should not include a tool change as this could lead to a collision.

Similarly, do not backtrack above a G10, G24, G26, G39, G39+.

It is possible to offset the programmed trajectory in Intervention mode, an axis recall will be necessary to cancel this offset. When no offset is active an action on the cycle start button is possible and normal execution will resume even before the interruption point.

For more detailed informations on Gxx please refer to Flexium Programming manual M00018.

4.10.1 Auto recall

A similar function is auto recall. After axis disengagement in Intervention mode and in addition to the free return, it is possible to return along the disengagement path.

This return is activated by a permanent action on the flag *...WCNC.General.BacktrackAuto* .

The number of trajectory elements for this phase is named *'Number of stored points'* in Flexium Tools and declared in word 2 of P114.

It is limited to 10. If more than 10 trajectory elements were executed the system will interpolate between the current position and the 10^{th}.

The return speed is the jog speed. However the last part of the resume path could be executed at the programmed feedrate. The length of this last part (Recall distance) is declared in word 2 of P114.

Page deliberately empty.

Page deliberately empty.

Summary

Page deliberately empty.

5 Machine Parameters Setup

5.1 Introduction

Two methods are available to adapt the CNC system to a particular machine-tool.
- A series of specific graphical pages for logical data entry.
- A simple parameter editor.

5.1.1 Specific pages

Each page contains associated information.
For example: *" CNC options ", Axis kinematics", "travels"*…

This method offers several advantages:
- Diagrams facilitate comprehension.
- The values are controlled.
- The units selections are simplified.
- It is possible to cancel multiple values.
- It is possible to reset the default values.
- Overall coherence is checked.

The pages are arranged in order to offer a logical order of integration and adjustment.
Nevertheless the user is free to choose a different order, to return to previous pages etc.

5.1.2 Parameters Editor

The parameters editor presents all parameters in the form of a list.
For each parameter it gives: description, limit values and the default value.

To facilitate integration chapters 4.2 and onwards gather the parameters by families.
For example: *"Axis Declaration"*, *"Measurement Settings"*…

At the beginning of each chapter, you have a table allowing you to describe your machine.
These tables facilitate the information gathering process.

The "Flexium Parameters manual M00021" gives the precise description of each parameter.

5.2 Axis Declaration

5.2.1 Data Tables

5.2.1.1 Channel Assignment

Channel No.	1	2	3	4	5	6	7	8
CNC Channel								
PLC Auxiliary channel								
Milling machine channel								
Lathe channel								
Main front turret								
Main rear turret								
Secondary turret								

5.2.1.2 Axis Assignment to the Channels

	Main linear axes			Secondary linear axes			Rotary axes		
Axis name	X	Y	Z	U	V	W	A	B	C
Channel 1									
Channel 2									
Channel 3									
Channel 4									
Channel 5									
Channel 6									
Channel 7									
Channel 8									

5.2.1.3 Definition of the Axes in the Channel

	Main linear axes			Secondary linear axes			Rotary axes		
Axis name	X	Y	Z	U	V	W	A	B	C
Physical address (@ 0-31)									
Measured axis									
Displayable axis									
Servo-controlled and interpolated axis									
Axis programmed by diameter									
Axis with clamps									
Modulo axis									
Axis with limited excursion									
Axis with quantified movement									
Carried axis									
Carried axis									
Coupled (driven) axis									

5.2.1.4 Handwheel Definition

	Handwheel 1	Handwheel 2	Handwheel 3	Handwheel 4
Handwheel present				
Physical handwheel address (The address (0 to 3 inclusive) can be identical to an axis address)				

5.2.2 List of Axis Declaration Parameters

Parameter	Description
P0-N0	Displayed axes
P1-N0	Modulo rotary axes
P1-N1	Limited excursion rotary axes
P2-N0	Measured axes
P3-N0	Servo-controlled and interpolated axes
P4-N0	X axis programmable by diameter
P4-N1	U axis programmable by diameter
P4-N2	Internal unit and dimension acquisition/display format for linear axis
P4-N3	X and U axes controlled with reference to diameter in incremental JOG
P4-N4	Internal unit and dimension acquisition/display format for rotary axis
P5-N0	Type of machine channel: lathe or mill
P5-N1	Type of turret : Rear or front
P8-N0	Axes with clamps
P9-N0 to N31	Axis assignment to channels
P14-N0	Handwheels declaration
P14-N1	Filtering of the measurement (handwheel)
P14-N2	Poor signal and encoder channel complementary check declaration (handwheel)
P14-N3	Encoder physically connected to an analogue interface (handwheel)
P14-N4-N7	Poor signal and encoder channel complementary setting (handwheel)
P27-N0 to N31	Couplings assignment for the duplicated and synchronised axes
P28-N0	Synchronised axis coupling enable
P64-N0	Carrier and carried axes
P65-N0 to N31	Axes with quantified movements
P70-N0	Axes connected to an analogue port
P70-N6	DISC NT Sensorless spindles
P70-N7	Axis measurement on 32 bits (only for Axis/ spindle switching)
P70-N8	DISC NT Spindles swichable with an axis
P85-N0 to N31	Master/slave coupled axes (tandem function)
P86-N0	Master axes with torque synchronisation (tandem function)
P97-N0	Number of CNC channels
P97-N1	Number of auxiliary channels

P70 words N1 to N5 do not exist.

5.3 Measurement Settings for analogue axes

5.3.1 Data Tables

5.3.1.1 Measurements of the Axes in the Channel

	Main linear axes			Secondary linear axes			Rotary axes		
Axis name	X	Y	Z	U	V	W	A	B	C
Physical axis address (@ 0 to 31)									
Measured axes									
Measurement reversal									
Measurement conversion coefficient	MULTI								
	DIVI								
Encoder error check									

5.3.1.2 Handwheel Definition

		Handwheel 1	Handwheel 2
Physical handwheel address (The addresses (0 to 3 inclusive) can be identical to an axis address)			
Measurement reversal			
Measurement conversion coefficient	MULTI		
	DIVI		
Overspeed coefficient in G12	MULTI		
	DIVI		

5.3.2 List of Measurement Settings Parameters

Parameter	Description
P10-N0	Axis measurement direction
P11-N0 to N31	Axis measurement conversion coefficients
P12-N0	Handwheel measurement direction
P13-N0 to N17	Handwheel measurement conversion coefficients
P25-N0	Poor signal and encoder channel complementarity check declaration
P25-N1	Encoder physically connected to an analogue interface
P26-N0 to N31	Poor signal and encoder channel complementarity check

P10 and P11 are for analogue axes only.

5.4 Servo-Control Settings

5.4.1 Data Tables

5.4.1.1 Servo-Control Characteristics of the Axes in the Channel

	Main linear axes			Secondary linear axes			Rotary axes		
Axis name	X	Y	Z	U	V	W	A	B	C
Position servo-loop constant (ms)									
Maximum feed rate (mm/min, or deg/min)									
Servo-drive reference for VMAX (9V recommended)									
Feed rate buildup time TV (ms) (time measured at 63% VMAX)									
Maximum acceleration DDM=VITMAX/TV (mm/s^2, or deg/s^2)									
Maximum work rate on the path VTmax									
Acceleration for movement at the work rate DDm=VTmax/TV (mm/s^2, or deg/s^2)									
Stopping accuracy at the work rate (µm)									
Servo-control error tolerated on circles ($µm/smp^2$)									

5.4.1.2 Speeds and Accelerations of the Axes in the Channel

	Main linear axes			Secondary linear axes			Rotary axes		
Axis name	X	Y	Z	U	V	W	A	B	C
Physical axis address (@ 0-31)									
Fast JOG speed									
Normal JOG speed									
Slow JOG speed									
Acceleration for movement at work rate									
Speed reference direction									
KVAR servo-control coefficient									
Maximum following error EMAX (in internal units)									

The KVAR coefficient (used to defined the position loop gain) is used for analogue axes only.

5.4.2 List of Servo-Control Settings Parameters

Parameter	Description
P19-N0 to N31	Acceleration anticipation in very high speed machining
P19-N32 to N63	Dry friction correction amplitude in very high speed machining
P19-N64 to N95	Dry friction correction time in very high speed machining
P20-N0	Axis speed reference direction
P21-N0 to N31	Servo-system loop gain coefficient
P22-N0 to N31	In-position windows
P23-N0 to N31	Maximum following errors
P24-N0 to N31	Characteristics of synchronised axis pairs
P30-N0 to N31	Maximum axis traverse rates
P31-N0 to N2	JOG speeds
P31-N3 to N4	Reduced speeds
P32-N0 to N31	Maximum permissible accelerations
P33-N0 to N31	Approach speed
P52-N0	Servo-loop error tolerated on circles
P53-N0	Setting of the speed buildup in the path
P55-N0 to N15	Speed anticipation coefficients and the number of terms for VHSM
P56-N0 to N7	Servo-loop time constant
P57-N0 to N31	Dynamic movement control
P66-N0 to N31	Time constant per axis

P20 and P21 are for analogue axes ports only.

5.5 Setting of Travels

5.5.1 Data Tables

5.5.1.1 Travels and Reversal Error Compensation for the Axes in the Channel

	Main linear axes			Secondary linear axes			Rotary axes		
Axis name	X	Y	Z	U	V	W	A	B	C
Physical axis address (@ 0 to 31)									
Maximum travel on the axis (in IU)									
Negative homing									
Position of the ORPOM (in IU)									
Minimum or negative travel (in IU)									
Maximum or positive travel (in IU)									
Reversal error compensation (in IU)									

The internal measurement unit (IU) is mm/10, mm/100, µm, µm/10 or µm/100 for linear axes or deg/10, deg/100, deg/1000 or deg/10000 for rotary axes.
Please refer to “Axes Programmable by Diameter and Internal Measurement” in the Flexium Parameters manual M00021.

5.5.2 List of Travels Parameters Settings

Parameter	Description
P15-N0	Datum switch approach direction in homing mode
P15-N1	Ignore the datum switch in HOME mode
P16-N0 to N31	Datum switch positions in machine dimensions
P17-N0 to N63	Axis travel limits
P18-N0 to N31	Backlash error compensation

5.6 Spindle Setting

5.6.1 Data Tables

5.6.1.1 Spindle Definition

	Spindle 1	Spindle 2	Spindle 3	Spindle 4
Spindle present				
Spindle physical address (imposed address)	@ 24	@ 25	@ 26	@ 27
Reversed spindle reference				
Spindle origin (expressed in internal measurement units)				
Number of increments per encoder revolution				
Spindle origin (expressed in internal measurement units)				
Maximum spindle speed (expressed in internal measurement units)				
In-position window (expressed in internal measurement units)				
Spindle servo-loop gain (in rpm/rev)				

5.6.1.2 Spindle Speed Ranges

	Spindle 1		Spindle 2		Spindle 3		Spindle 4	
Spindle speed range (rpm/rev)	Min	Max	Min	Max	Min	Max	Min	Max
Range M40								
Range M41								
Range M42								
Range M43								
Range M44								
Range M45								

5.6.1.3 Spindle Definitions

	Spindle 1	Spindle 2	Spindle 3	Spindle 4
Spindle present				
Spindle physical address (imposed address)	@ 24	@ 25	@ 26	@ 27
Lathe spindle				
Milling machine spindle				
Measured spindle				

5.6.2 List of the Spindle Settings Parameters

Parameter	Description
P6-N0 to N8	Declaration
P40-N0 to N3	Measurement conversion
P41-N0 to N3	Reference reversal
P42-N0 to N3	Spindle origins
P43-N0 to N3	Maximum speed for indexing
P43-N4 to N7	Maximum speed (by PLC program)
P44-N0 to N3	Indexing in-position windows
P45-N0 to N3	Indexing servo-loop gain
P46-N0 to N17	Speed range for spindle 1
P47-N0 to N17	Speed range for spindle 2
P48-N0 to N17	Speed range for spindle 3
P49-N0 to N17	Speed range for spindle 4

5.6.3 Rigid Tapping Parameters

Parameter	Description
P62-N0 to N7	Spindle acceleration and minimum spindle reference in the hole bottom
P63-N0 to N23	Cycle axis integration time constant

5.7 Miscellaneous Parameters

5.7.1 List of the Miscellaneous Parameters

Parameter	Description
P7-N0 to N2	Miscellaneous function parameter setting
P35-N0 to N31	Subroutine call by M function
P50-N0	Sampling period
P51-N0	Minimum block execution time
P58-N0	Interaxis calibration table reservation
P58-N1	Program stack
P67-N0 to N23	Potentiometer min-max range
P95-N0 to N2	Memory area allocation
P99-N0	Ratio between PLC and CNC real cycle time
P99-N1	Identifier of the master CNC
P99-N2	Maximum time for the detection of the error 249
P114-N0 to N2	Backup on path, Auto recall after INTERV

Page deliberately empty.

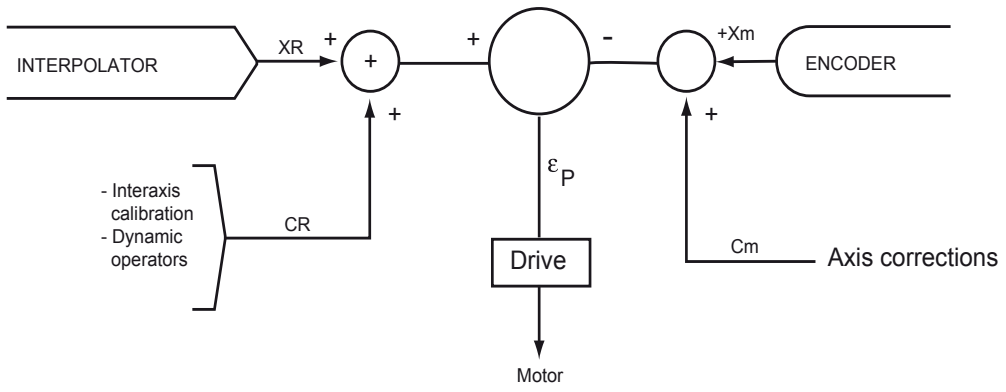
Summary

Page deliberately empty.

6 Axis Calibration

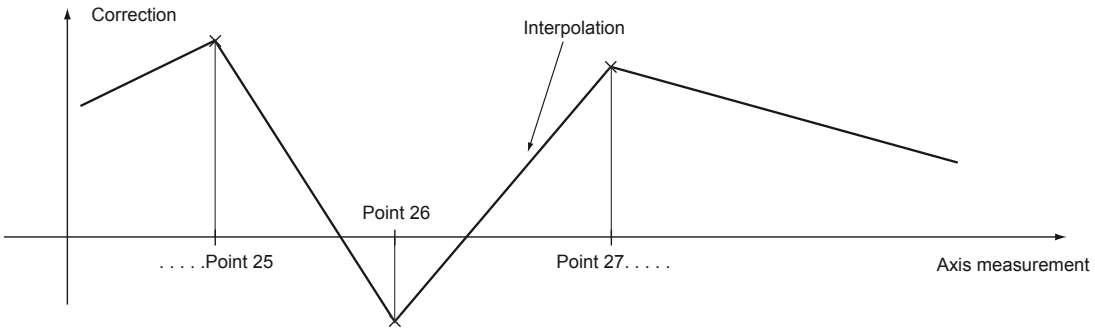
6.1 General

Axis calibration allows the system to add a correction to the measured axis position based on the real axis position.



The axis calibration is carried out on both linear and rotary axes. The corrections are entered for a limited number of points per axis.

The system computes the corrections between two points by linear interpolation.

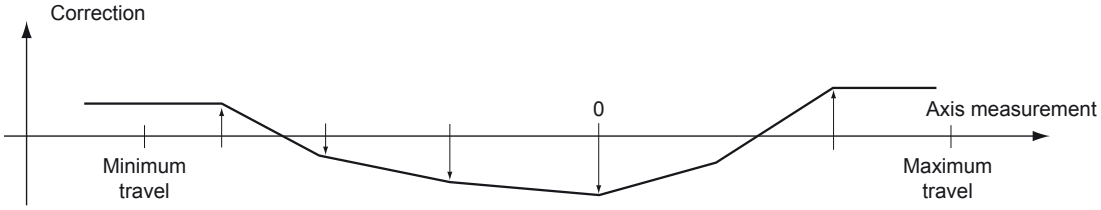


It is recommended to correct the measurements of the minimum and maximum travel points (defined by machine parameter P17). Otherwise, the value of the last correction is applied to these points.

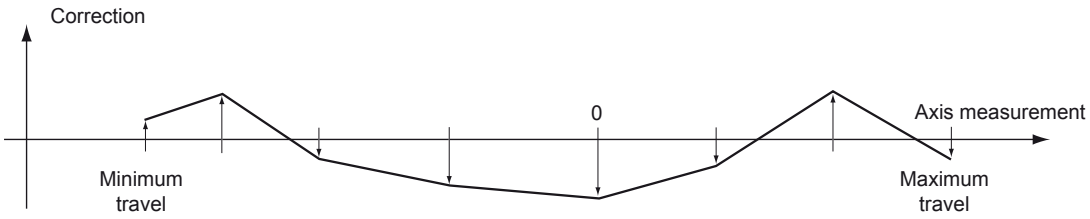
For more detailed informations on P17 please refer to Flexium Parameters manual M00021.

.

6.1.1 Minimum and maximum travel points not corrected



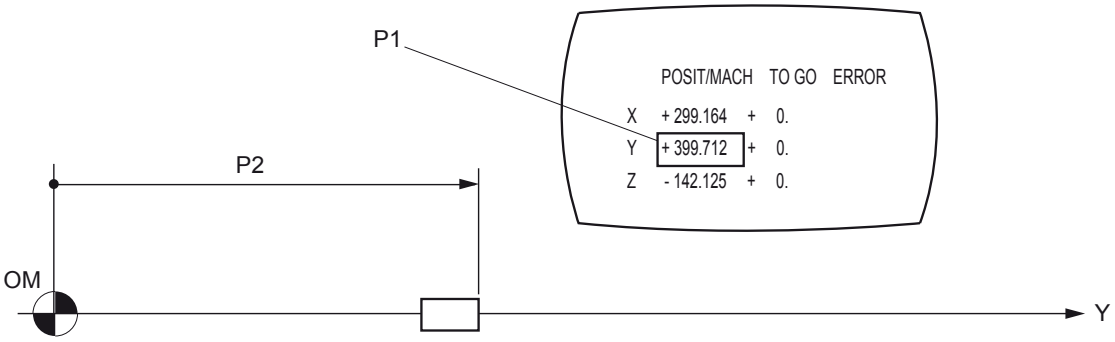
6.1.2 Minimum and maximum travel points corrected



A maximum number of approximately 2600 points can be corrected for all the axes combined. There are no restrictions on how the points are distributed between axes.

6.2 Correction Records

The real axis positions are taken for a series of points to determine the corrections required:



The corrections are in the system internal units (IU).

Axis number:

Measured position (P1)							
Real position (P2)							
Correction (P2 - P1)							

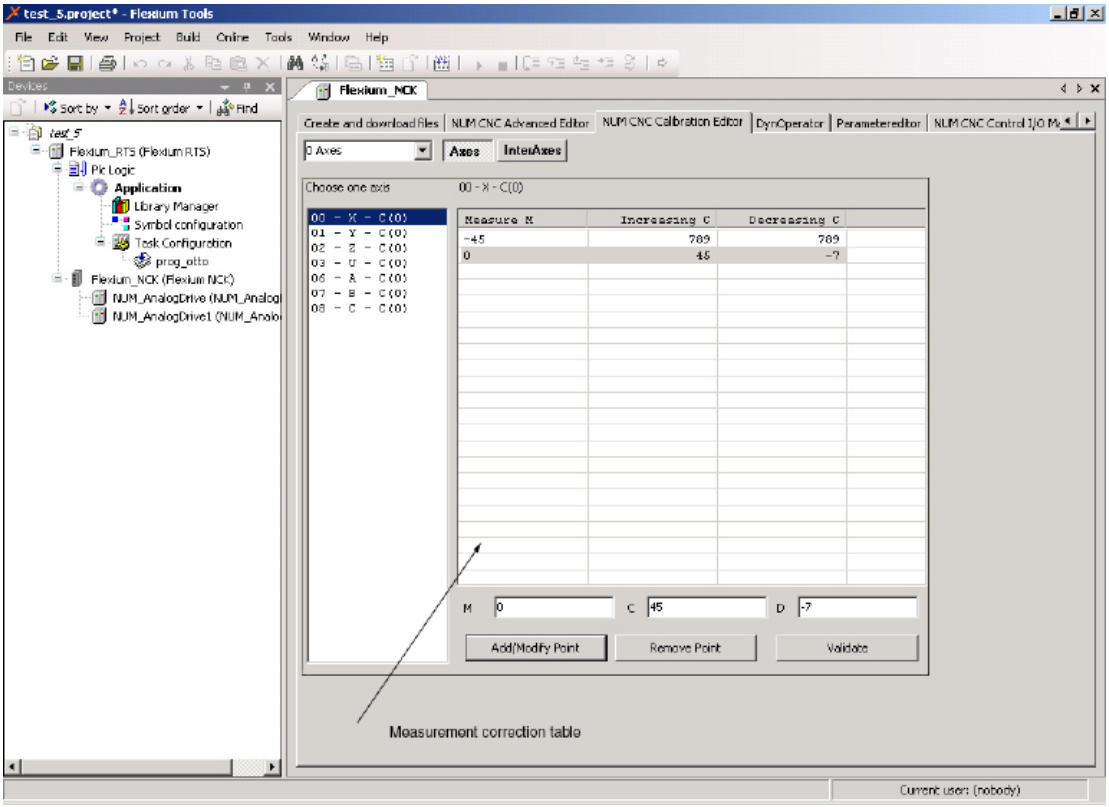
The measured values and corresponding corrections should be transferred to the correction tables. (See chapter 7.3)

A correction table must have at least three points. The correction range is between -32768 and +32767 units.
For a rotary axis, the corrections must be identical for the points 0 and 360 degrees.

6.3 Operations on Axis Measurement Correction Tables

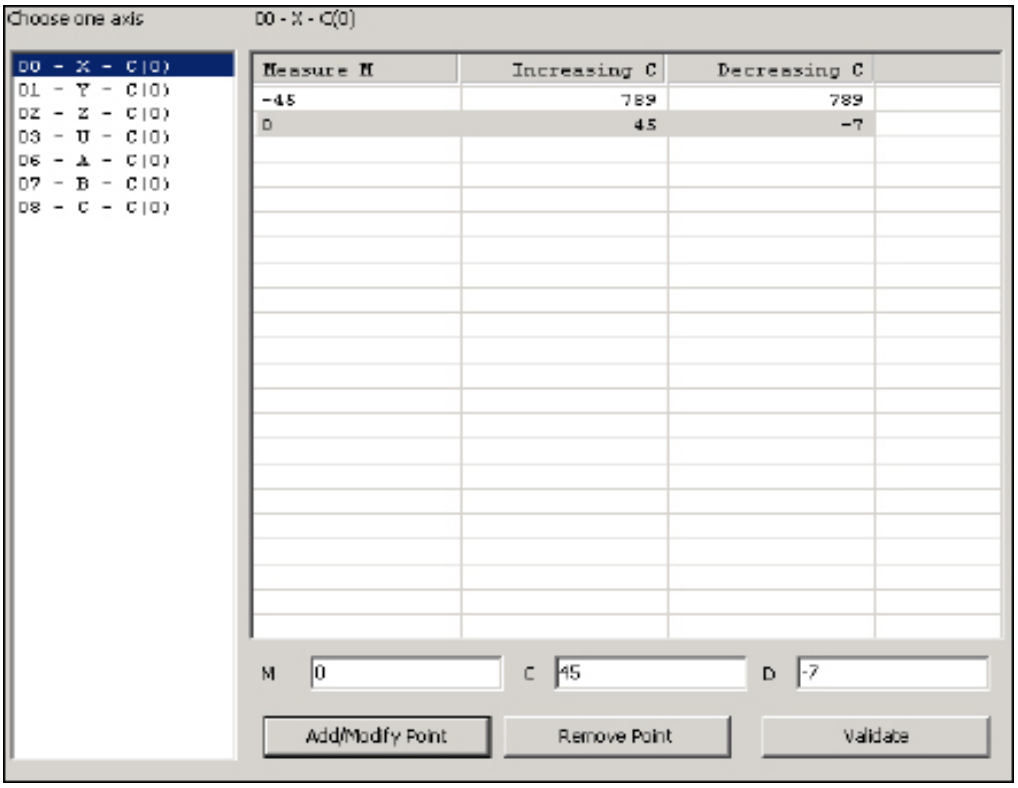
The records of all processes which have been applied in the Flexium System are saved in the Flexium Tools *project* file. Therefore, all these processes can be repeated in the future.

6.3.1 View of the “NUM CNC Compensation Editor” as part of the Flexium Tools



The correction records should already be known (See chapter 6.2). The corrections can be entered in the measurement correction table in any order.

6.3.2 Measurement Correction Table



6.3.2.1 Interpretation of the measurement correction table

- The header gives the number of the selected axis,
- The first column, *"Measure M"* is the point's demanded position (in internal units),
- The following two columns, *"C Increasing"* and *"D Decreasing"* are the corrections to be applied at that point (in internal units).

6.3.2.2 Performing an axis compensation calibration

- Open an existing or create a new Flexium Tools *project* file.
- Select the *'Flexium NCK'* tab.
- Select the *'NUM CNC Compensation Editor'* tab.
- Click the *'Axes'* button.
- Choose the axis to be calibrated.

6.3.2.3 Modifying or Adding a Correction

- In the text boxes below the table, enter the necessary correction values.
- Click *'Add / Modify Point'* button.

6.3.2.4 Deleting a Correction

- Select the correction to be deleted.
- Click the *'Remove Point'* button to delete a correction.
- Chose another axis to be calibrated, if necessary.
- Click the *'Validate'* button to validate your corrections.

Your modifications will be lost if they are not validated.
(by the *'Validate'* button)

Page deliberately empty.

Page deliberately empty.

Summary

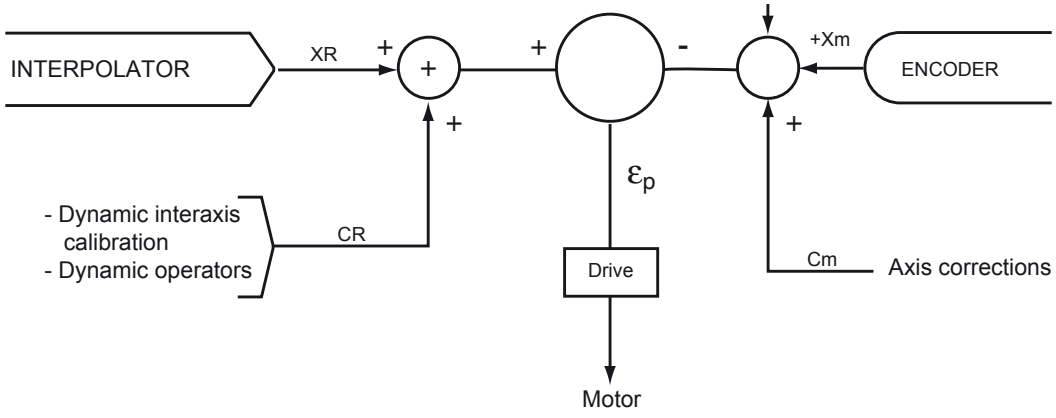
Page deliberately empty.

7 Inter-axis Calibration

7.1 Description of Inter-axis Calibration

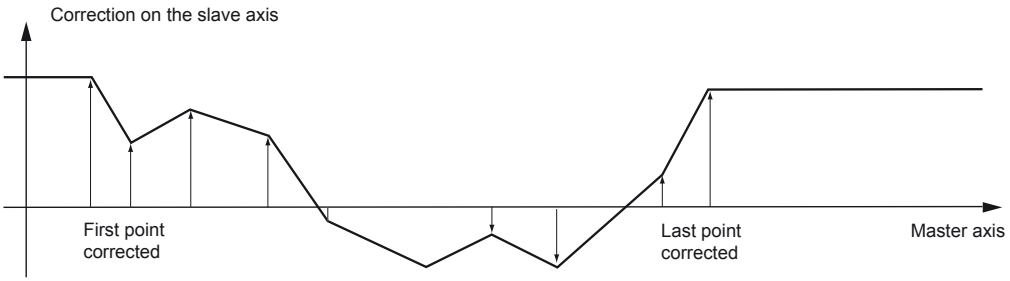
7.1.1 General

Inter-axis calibration allows the system to add a correction to the reference of a slave axis (reference created by the interpolators) from the position reference of a master axis.

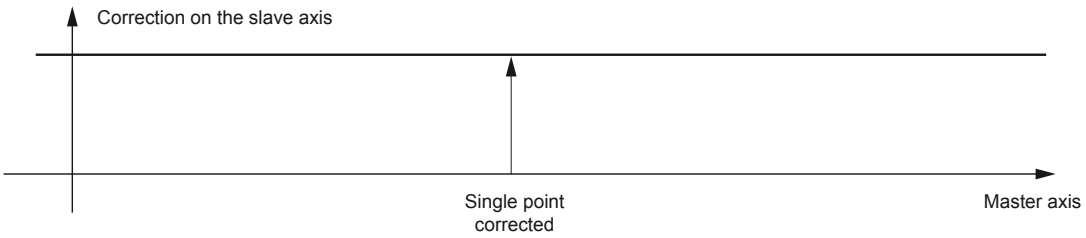


The corrections concern both linear and rotary axes.

They are entered for a limited number of points per axis. The system computes the corrections between two points by linear interpolation. Beyond the boundary points, the corrections are constant.



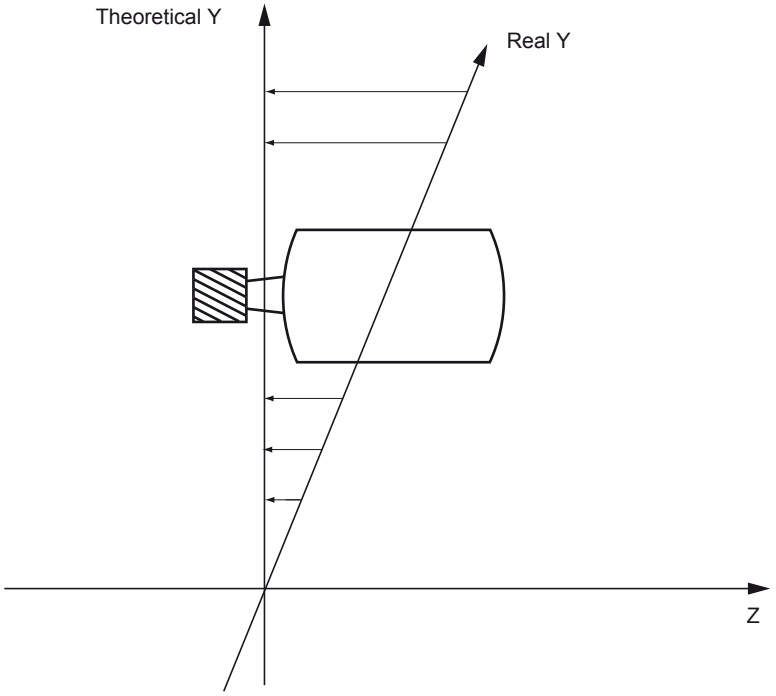
7.1.1.1 Special case of correction in a single point



7.1.2 Examples of Inter-axis Compensation Calibration

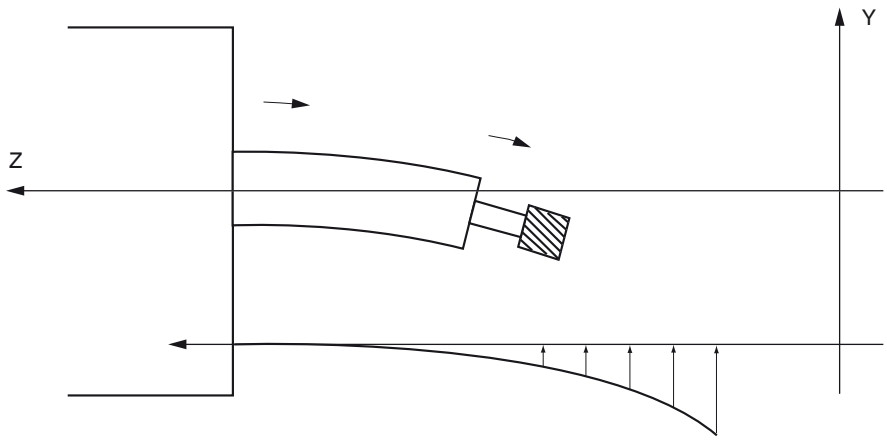
The deformations are exaggerated for clarity in the examples below.

7.1.2.1 Correction of a perpendicular alignment fault



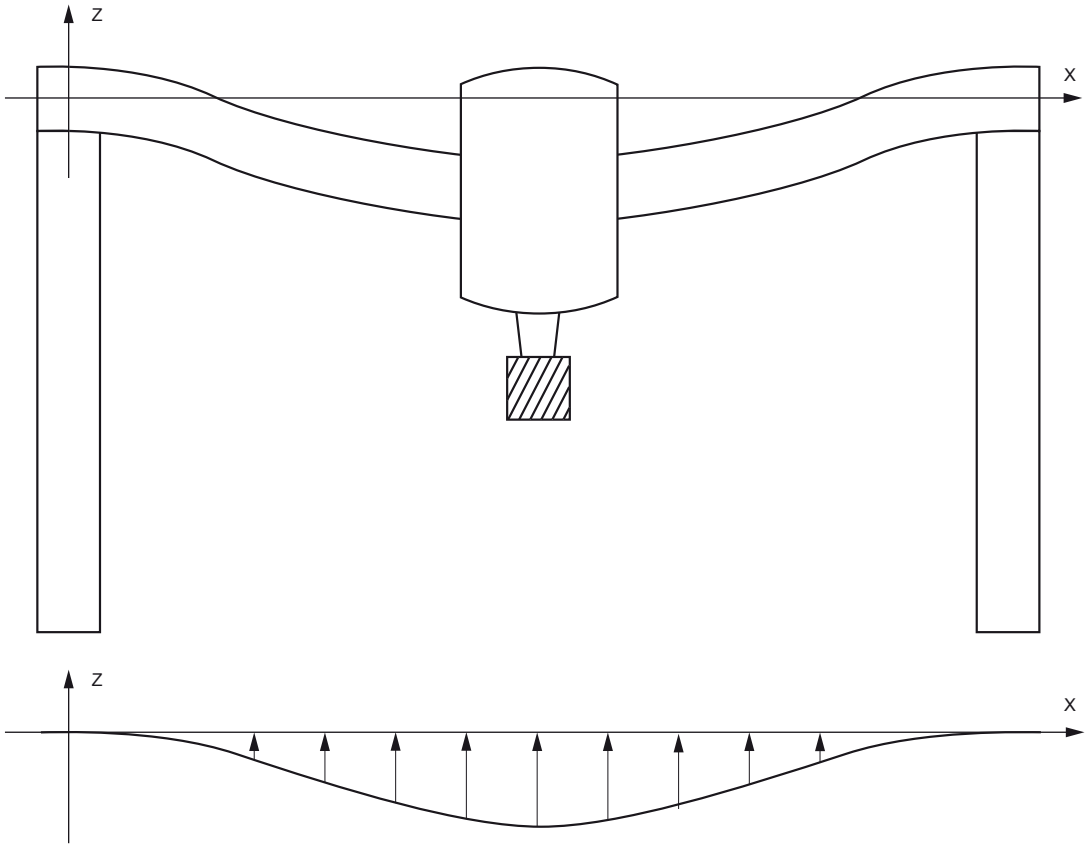
Z is corrected according to the position of the Y axis.

7.1.2.2 Correction of spindle flexion



Y is corrected according to the extension of the spindle.

7.1.2.3 Correction of sag of the cross piece of a gantry machine



Sag in Z is corrected according to the position on the X axis.

7.1.2.4 Correction of an axis expansion

An axis can be self-corrected (exclusively for the dynamic calibration: master axis same as slave axis) to take thermal expansions into account.

The correction tables can be written by the PLC program according to the measured temperatures then used by the system. The application of these corrections may prove to be difficult because of the thermal inertia of machines.

7.1.3 Tools for Inter-axis Compensation Calibration

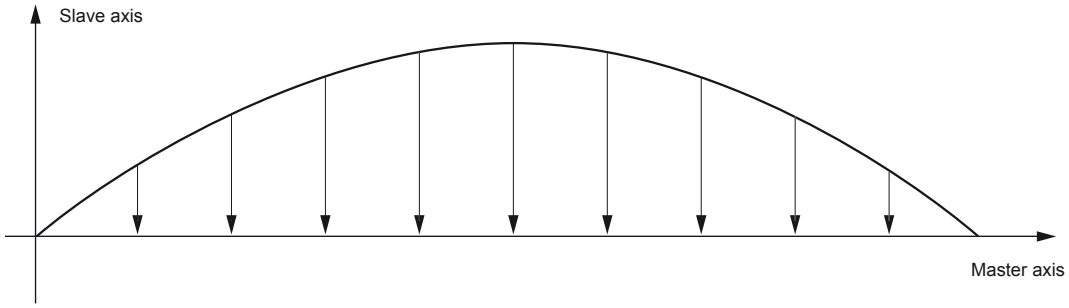
Two tools can be used for inter-axis calibration:

- Calibration by the *'NUM CNC Calibration Editor'* (see chapter 7.2) to make permanent corrections. These corrections take into consideration the deformations of the machine,
- Dynamic calibration using parameters E81xxx and E82xxx (see chapter 7.3). They can be modified at any time by the part program.
  These corrections are suited to variables that change over time such as thermal expansion.

The effects of inter-axis calibration by the *'NUM CNC Calibration Editor'* and of the dynamic calibration are added together, which is why it is not recommended to use both methods simultaneously.

7.1.4 Records of the Corrections on the Axes

The corrections to be made to the slave axis are measured for a series of reference positions on the master axis.



The corrections are made in the system internal units (IU).
Master axis No.: Slave axis No.:

Master axis position:							
Slave axis correction:							

The measured values should be transferred to the correction tables.
(see chapter 7.2 and chapter 7.3)

For inter-axis calibration by the *'NUM CNC Calibration Editor'*, the maximum corrections are ± 9999 units.
For dynamic inter-axis calibration, the maximum difference between two consecutive corrections is ± 65.000 units.

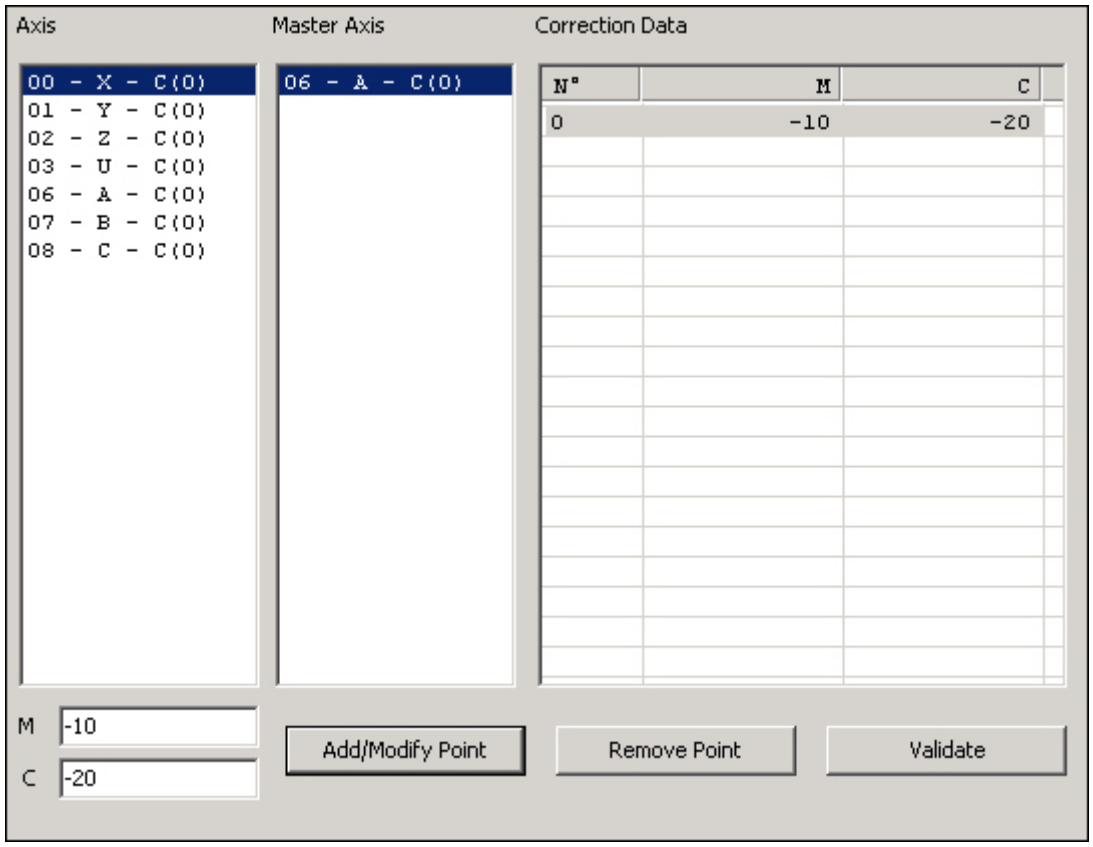
7.2 Inter-axis Calibration by 'NUM CNC Calibration Editor'

As in the case of axis calibration, all the records about applied inter-axis calibration are saved in the Flexium Tools *project* file. Therefore, all these processes can also be repeated in the future.

- A slave axis can have only one master axis.
- A master axis can have several slave axes.
- An axis cannot be its own master axis (contrary to dynamic compensation).

The correction records should already be known (see chapter 7.1). The corrections can be entered in any order.

7.2.1 Measurement Correction Table



7.2.2 Interpretation of the measurement correction table

- The first column, *"N"* is the number of the slave axis,
- The second column, *"M"* is a measured point on the master axis (in internal units).
- The last column, *"C"* is the correction to be applied (in internal units).

7.2.3 Carry out the interaxis compensation calibration

- Open an existing or create a new Flexium Tools *project* file.
- Select the *'Flexium NCK'* tab.
- Select the *'NUM CNC Calibration Editor'* tab.
- Click the *'InterAxis'* button.
- Choose the master axis and the slave axis.

7.2.4 Modifying or Adding a Correction

- In the text boxes below the table, enter the position on the master axis and correction value to be applied to the slave axis.
- Click *'Add / Modify Point'* button.

7.2.5 Deleting a Correction

- Select the correction to be deleted.
- Click the *'Remove Point'* button to delete correction.
- Chose another pair master axis / slave axis to be calibrated, if necessary.
- Click the *'Validate'* button to validate your corrections.

Your modifications will be lost if they are not validated (by the *'Validate'* button).

7.3 Dynamic Inter-axis Calibration

The possibility of writing correction tables from a part program depends on the setting of machine parameter P7 (Please refer to Flexium Parameters manual M00021):

- *P7 word 0 bit 5 = 0*: write by part program enabled,
- *P7 word 0 bit 5 = 1*: write by part program inhibited.

7.3.1 Addresses of the Correction Parameters

The corrections are made using parameters E81xxx, E82xxx and E940xx:

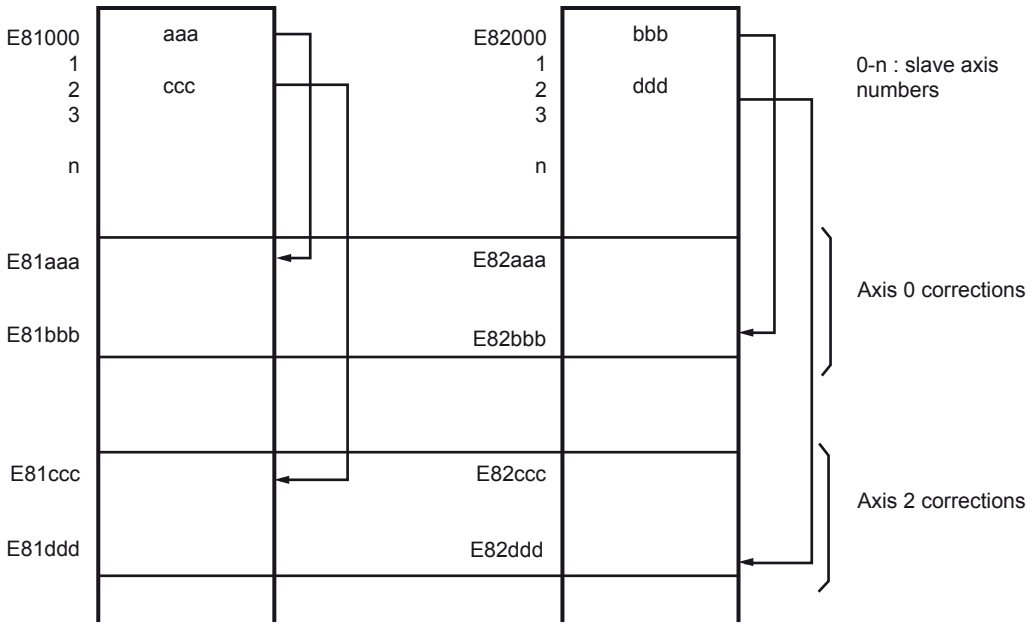
- *parameters E81xxx* address the reference positions of the master axes,
- *parameters E82xxx* address the corresponding corrections on the slave axes,
- *parameters E940xx* assign a master axis to a slave axis.

The current slave axis correction is accessible in read-only parameters E950xx.

7.3.2 Correction Tables

The size of the correction tables (number of E81xxx parameters and E82xxx parameters) is defined by machine parameter P58, word 0 (Please refer to Flexium Parameters manual M00021).

The maximum size of the correction tables is 1000 E81xxx E82xxx parameters. The correction tables can be schematically represented as follows:



The first 32 parameters E81xxx and E82xxx are each assigned to the axis with the equivalent physical address. For instance, E81003 and E82003 are assigned to axis 3.
They define the limits of the correction table assigned to that axis.

The subsequent parameters define:

- The reference position of the master axis (parameters E81xxx),
- The corresponding correction to be applied to the slave axis (parameters E82xxx).

The reference positions and slave axis corrections are in system internal units.
For more detailed informations on Exxxx please refer to Flexium Programming manual M00018.

A master axis is assigned to a slave axis by *E940xx = yy* where:

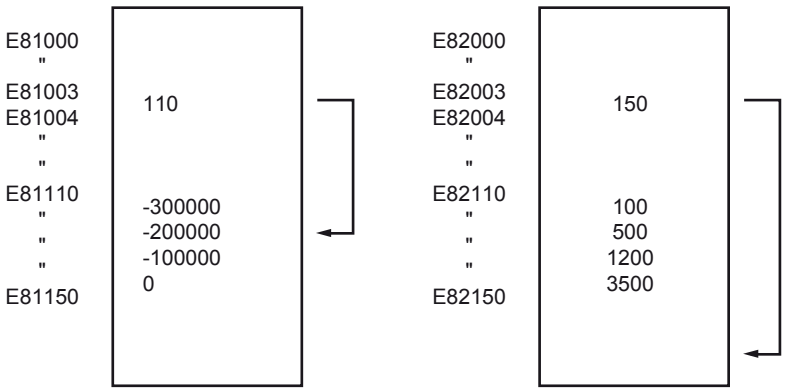
- *xx* is the slave axis number,
- *yy* is the corresponding master axis number.

E940xx = -1 means that no master axis is assigned to slave axis xx.

For a corrected axis, the reference positions on the master axis must be defined in increasing order.
Any unused location in the correction table can be used as extra E80xxx parameters (local data written and read by the CNC).

Example

E94003 = 1



- *E94003 = 1* means that slave axis 3 is assigned to master axis 1.

- *E81003 = 110 and E82003 = 150* mean that the parameters defining the corrections of axis 3 are between E81110 and E81150 for the reference positions on master axis 1 and between E82110 and E82150 for the corresponding corrections to be applied to slave axis 3.

- *E81110 = -300000 and E82110 = 100* mean that the first reference position on master axis 1 is located at -300000 IU (i.e. -300 mm if the internal system unit is micrometre) and that the corresponding correction on slave axis 3 is 100 IU.
  (i.e. 100µm if the internal system unit is micrometre)

7.3.3 Writing and Enabling the Correction Tables

Parameters E81xxx, E82xxx and E940xx can be written by the part program.

7.3.3.1 Conditions for Writing Parameters E81xxx and E82xxx

All E940xx parameters are equal to -1
This means that no correction table is enabled.
All E81xxx and E82xxx parameters can be modified without restriction.

At least one E940xx parameter is different from -1
This means that there is at least one master axis.
All the master axes must be reset (by setting parameters E940xx to -1).

As long as there is a master axis, you must not modify:
- the parameters defining the limits of the correction tables
- the parameters defining the reference positions and the corrections.

7.3.3.2 Conditions for Writing Parameters E940xx

To change the master axis, it is necessary to disable the correction linkage (parameter = -1).
For instance, the following steps are necessary to change from master axis 3 to master axis 1 for slave axis 2:
- E94002 = -1,
- E94002 = 1.

An axis can be slaved to its own reference, e.g. E94002 = 2.

A consistency test is made to the correction table when writing an E940xx parameter (table limits, master axis reference points listed by increasing order). When an inconsistency is detected, the parameter setting is rejected and error 95 is generated.

7.3.3.3 Procedure for Writing and Enabling the Correction Tables

The logical order for writing parameters E81xxx, E82xxx and E940xx is as follows:
- Cancel enabling of all the correction tables: *E940xx = -1*,
- Write the correction tables by modifying parameters *E81xxx and E82xxx*,
- Assign master axes to the slave axes: *E940xx=yy*.

Page deliberately empty.

Summary

Page deliberately empty.

A Appendix

A.1 Configuration of Language.ini

All HMI message files must be defined in the configuration file Language.ini.
This configuration file contains NUM default settings which must not be modified.

Uncontrolled changes to file Language.ini can cause malfunction of the HMI.

File type	Description	Path
*.ini	Language configuration file	C:\Program Files\NUM\Flexium HMI\FXBrowser\Bin

In the list below, “Path” stands for C:\Program Files\NUM\Flexium HMI\FXBrowser\

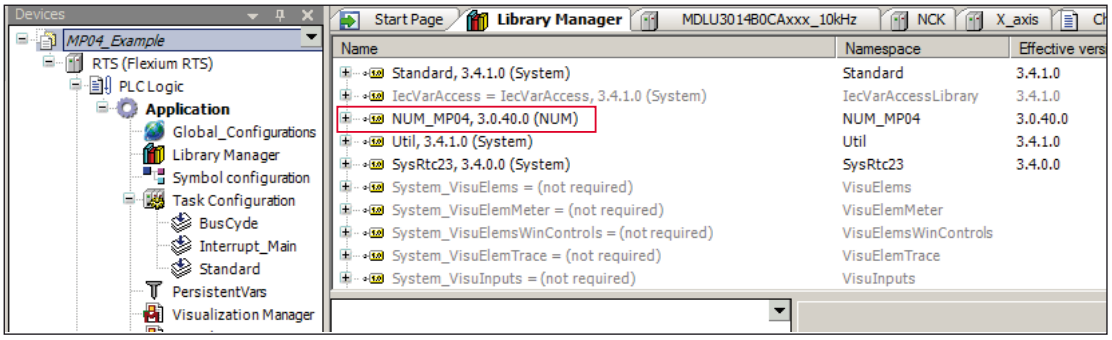
“Language.ini”	
[General]	
CurrentLanguage=1	// English language active
NumberOfLanguage=5	// Number of languages
[1]	// Section 1 (English)
LanguageName=English	// Language name
LanguageShort=en	// Language symbol for file extensions
HelpLanguage=en	// Language of the integrated help
UserMsgHelpLanguage=	// Language of the user message help
SystemMsgHelpLanguage=	// Language of the system message help
TestPointLanguage=en	// Language of the test point description
HMIFile0=Path\Language\en\HMIText_en.txt	// NUM HMI text file
HMIFile1=	// User HMI text file
HMIFile2=Path\Language\en\FXLibTexte_en.txt	// FXLib text file
HMIFile3=Path\Language\en\HMIOptionText_en.txt	// HMI options text file
HMIFile4=Path\Language\en\NUMROTOHMITexte_en.txt	// NUMROTO text file
RunTimeSystemErrors=Path\Language\en\RTSError_en.txt	// RTS message file
BootMonitoring=Path\Language\en\BootMonitoring_en.txt	// Boot Monitoring message file
CNCErrorFile0=Path\Language\en\CNC99899_en.xpi	// NUM CNC message file
CNCErrorFile1=Path\Language\en\CNC20600_en.xpi	// NUM CNC message file
CNCErrorFile2=Path\Language\en\CNC20700_en.xpi	// NUM CNC message file
CNCErrorFile3=Path\Language\en\CNC20800_en.xpi	// NUM CNC message file
CNCErrorFile4=	// reserved for NUM

CNCErrorFile5=	// reserved for NUM
CNCErrorFile6=	// User/OEM CNC global message file
CNCErrorFile7=	// User/OEM CNC global message file
CNCErrorFile8=	// User/OEM CNC global message file
CNCErrorFile9=	// User/OEM CNC global message file
CNC0ErrorFile=	// User/OEM message file for CNC 0
CNC1ErrorFile=	// User/OEM message file for CNC 1
CNC2ErrorFile=	// User/OEM message file for CNC 2
CNC3ErrorFile=	// User/OEM message file for CNC 3
CNC4ErrorFile=	// User/OEM message file for CNC 4
CNC5ErrorFile=	// User/OEM message file for CNC 5
CNC6ErrorFile=	// User/OEM message file for CNC 6
CNC7ErrorFile=	// User/OEM message file for CNC 7
CNC8ErrorFile=	// User/OEM message file for CNC 8
CNC9ErrorFile=	// User/OEM message file for CNC 9
CNC10ErrorFile=	// User/OEM message file for CNC 10
CNC11ErrorFile=	// User/OEM message file for CNC 11
CNC12ErrorFile=	// User/OEM message file for CNC 12
CNC13ErrorFile=	// User/OEM message file for CNC 13
CNC14ErrorFile=	// User/OEM message file for CNC 14
CNC15ErrorFile=	// User/OEM message file for CNC 15
DriveErrorFile=Path\Language\en\descri_a.tab	// NUM Drive error message file
DriveWarningFile=Path\Language\en\DriveWarning_en.txt	// NUM Drive warning message file
DriveErrorExtensions=Path\Language\en\DesalExt_a.txt	// Extended NUM Drive message file
SysErrorFile=Path\Language\en\Sysboot_en.txt	// NUM System message file
UniteErrorFile=Path\Language\en\DncText_en.txt	// NUM UNITE message file
HMIBootFile=Path\Language\en\HmiBoot_en.txt	// NUM boot message file
ConvertSen=Path\Language\en\Convert.sen	// Extended NUM Drive message file
TestPointFile=Path\Language\en\Descri_a.tp	// Test points description
PLCErrorFile=	// User/OEM PLC message file
MachinePanelP=Path\MachinePanel\PanelP\MachinePanelP_en.txt	// Settings for the integrated machine panel P
MachinePanelF=Path\MachinePanel\PanelF\MachinePanelF_en.txt	// Settings for the integrated machine panel F
HMIMessage=Path\Language\en\HmiMessage_en.txt	// HMI message file
UserMessage=	// Customized program messages (SDK)

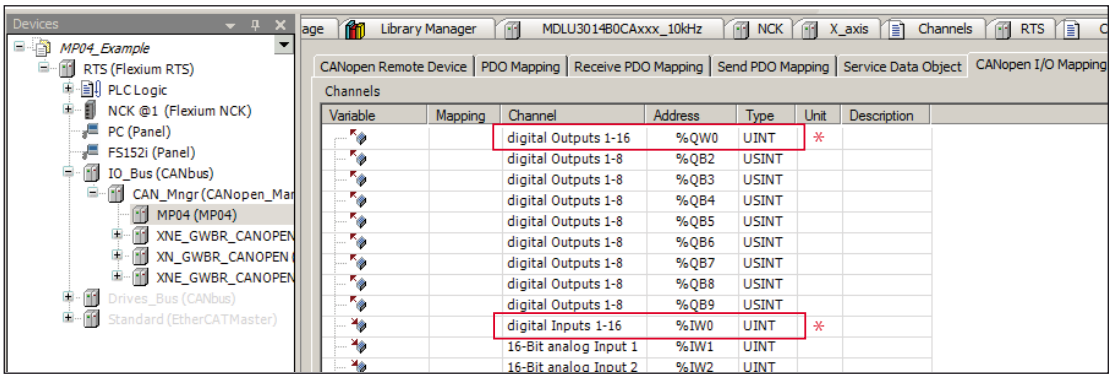
A.2 Example of programming the MP04 using the NUM_MP04 function block

1 - Add the library NUM_MP04 in your project
2 - Add all MP04 machine panel of the machine in the device tree
3 - Create an instance of NUM_MP04 for each physical machine panel
4 - Set the hardware address of each instance
5 - Program using the predefined IOs.

1 Include the Library NUM_MP04 in the Library manager



2 Include an MP04 in the device tree and check the I/O mapping (first %QW and first %IW)



Make a note of the two values (*) above.

3 Create the Following Global variable file : (named GVL in this example)

```
VAR_GLOBAL
            FX_MP04 : NUM_MP04;      // Will bet the MP04 Name
END_VAR
```

FX_MP04 is an example of name of the first machine panel.
If they are several machine panels on the machine, each one should have a different name.

4 Create the following Global variable file in the device window

```
VAR_CONFIG
    GVL.FX_MP04.inpcan              AT %IB0          : MP04inpcan;
    GVL.FX_MP04.outcan              AT %QB0          : MP04outcan;
END_VAR
```

The syntax should be exactly respected with the following remark:

- The prefix GVL. In the second file should be replaced by the name of the first file.
- The value behind %IB and %QB is the number found in the I/O mapping of the MP04.
- When creating a global variable file a line VAR_GLOBAL is automatically inserted, replace GLOBAL by CONFIG in the second file.

5 Examples of programming (Structured text)

```
NCK.W_ALLCNC.Start              := FX_MP04.xBtnStart        // Processing the START button
FX_MP04.xLampStart              := NCK.RCNC.General.Start   // Activating the start lamp
NCK.WCNC.Spindle[1].Override    := FX_MP04.bSpeed           // Spindle 1 override
```

Page deliberately empty.

Page deliberately empty.

Symbol

A

B

C

D

E

F

G

H

I

J

K

L

M

N

O

P

R

S

T

U

V

W

X

Published by NUM Spa

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

Printed April 2011

CNC HighEnd Applications