POSITIONING AND CONTOURING CONTROL SYSTEM

APCI-8001 AND APCI-8008

TOOLSET SOFTWARE
MCFG FOR WINDOWS

Rev. 5/112018

www.addi-data.com

Figures

Tables

1 The *mcfg.exe* utility program

What is this manual for?	This document contains all the necessary information on using the **MCFG program**, a component of the TOOLSET software for the **xPCI-800x** positioning and motion control units. The individual menus and the screens for the MCFG application are explained below.
Which devices belong to the xPCI-800x family?	The xPCI-800x family includes positioning and motion control units of the third generation, i.e. currently the positioning and motion control units APCI-8001, APCI-8008 and CPCI-8401. Other devices are being developed.
Which operating systems are supported?	The MCFG program can be used under the following operating systems: Windows NT 4.0, Windows 2000 and Windows XP.

The *mcfg.exe* utility program (referred to below as **MCFG**) is run as a Windows application in the MDI standard and provides a high-performance commissioning, diagnostic and configuration interface for the xPCI-800x family of control units. The MCFG application also includes a convenient programming environment for stand-alone application (SAP) program development.

1.1 Installing and configuring a working environment

A few conditions apply to using the MCFG program. Instructions for installing and configuring a working environment can be found below.

1.1.1 Installing a hardware driver for the motion control board

For Windows versions since XP, the hardware driver must be installed before each use of a board. The driver is to be found on the Toolset CD in the \inf directory for the respective operating system versions.

1.1.2 Installing a miniport driver for earlier Windows versions

This is only required for Windows 95/98/ME NT and 2000. Otherwise, this driver must not be used anymore!

If the miniport driver has not yet been installed, this should be done first. The miniport driver “RNWMC (sys, vxd, dll)” contains a high-performance device driver which enables hardware access to the xPCI-800x devices. The driver can currently be installed under the following Windows operating systems: Windows 95 / 98 / Me, Windows NT 4.0, Windows 2000 and Windows 2003 Server.

The miniport driver can be found on the “xPCI-800x TOOLSET CD-ROM” in the *Miniport* sub-directory. From there, the *ksetup.exe* program must be called to install it.

If the miniport driver is used, the file McuG3.inf must be installed as the hardware driver (→ kwdm.sys).

1.1.3 Configuring a working environment

Later, the MCFG program needs at least two files. Creating a local working directory on the hard drive is also recommended. This local working directory is referred to below as **LA**. The *system.dat* and *rwmos.elf* files should be stored in the LA.
The *rwmos.elf* file is the operating system of the control module and must be transferred to the control unit at least once every time the PC is started (control unit boot) so that this can be used.
The *system.dat* file is a binary file and contains the settings from mcfg. This system file is transferred to the control unit when it is booted. This communicates the user's settings to the control unit.

1.1.3.1 Selecting the SYSTEM.DAT file

The system file can be created using, or based on, several methods. For new users, these are the methods of copying or creating. For those migrating or in the event of a version change, this is the conversion method.

- **Copying** the “xPCI-800x TOOLSET CD-ROM”. In the *Firmware & System.dat Files* sub-directory, further sub-directories are created for specific control unit types. The relevant *system.dat* files can be found here.
  Note: After copying, the write protection on the file must be removed!

- **Creating** a *system.dat* using the *sysgen.exe* utility program. The *sysgen.exe* utility program is located in the *Toolset* sub-directory of the “xPCI-800x TOOLSET CD-ROM”. This should be copied to the LA and called in a Windows console using one of the parameters [APCI-8001 or APCI-8008]. The following motion control boards are compatible here: APCI-8001 and APCI-8008. Alternatively, a current version of the sysgen.exe program can be downloaded from the Internet under Download SYSGEN.EXE.
  After creating the system.dat file, the sysgen.exe application can be deleted from the LA again.

- **Converting** an existing system file with an older version or another control unit type. The *sysconv.exe* utility program can be found in the *Toolset* sub-directory of the “xPCI-800x TOOLSET CD-ROM”. SYSCONV should be copied to the LA and called in a Windows console using one of the parameters [/APCI-8001 or /APCI-8008].
  The following motion control boards are compatible here: APCI-8001 and APCI-8008. Alternatively, a current version of the sysconv.exeprogram can be downloaded from the Internet under Download SYSCONV.EXE.
  After creating the system.dat file, the sysconv.exe application can be deleted from the LA again.

- **Project installation** of a complete program package such as McuWIN. A project directory is created when installing the McuWIN programming interface. After the installation, this directory contains the system.dat and RWMOS.ELF files (see below). During the installation procedure, the mcfg environment is set up on these files in the same way, provided mcfg has **already** been installed.

1.1.3.2 Selecting the RWMOS.ELF file

In the *Firmware & System.dat Files* sub-directory of the “xPCI-800x TOOLSET CD-ROM”, further sub-directories are created for specific control unit types. The relevant *rwmos.elf* files can be found here. The corresponding file should be copied to the LA.

The working environment is set up according to these steps. Only the mcfg.exe application can then be installed.

1.2 Installing / updating MCFG

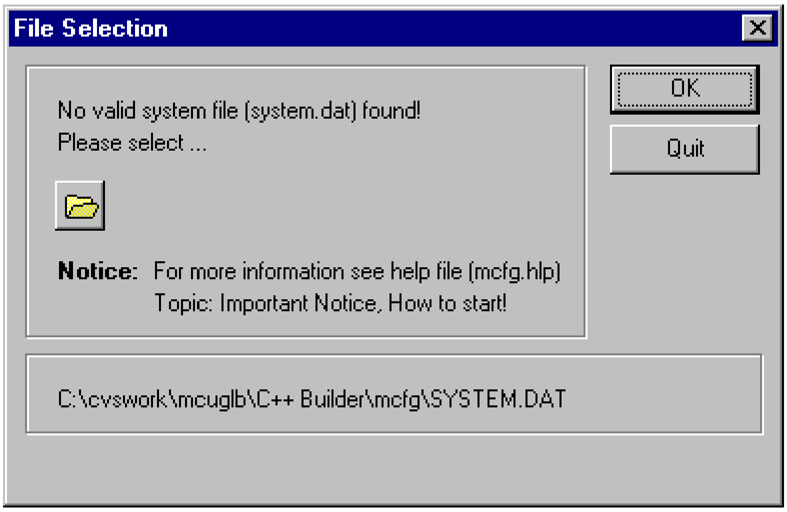
The *mcfg.exe* toolset program can be found on the “xPCI-800x TOOLSET CD-ROM” in the *Mcfg* sub-directory. From there, the *setup.exe* program can be called to install it.
Alternatively, a current version (mcfg.exe) can be downloaded from the Internet under Download mcfg Installationsdatei [Download mcfg installation file].

Note: When updating the MCFG application to a new version, the old MCFG application must first be deinstalled. Where necessary, any sensitive data such as INI files, the system.dat file or other important files created by the user must be backed up in advance.

1.3 Starting MCFG

When starting the MCFG application, the following screen appears when it is called after the first installation:

Figure 1-1: Start screen after first installation



File Selection

No valid system file (system.dat) found!
Please select ...

Notice: For more information see help file (mcfg.hlp)
Topic: Important Notice, How to start!

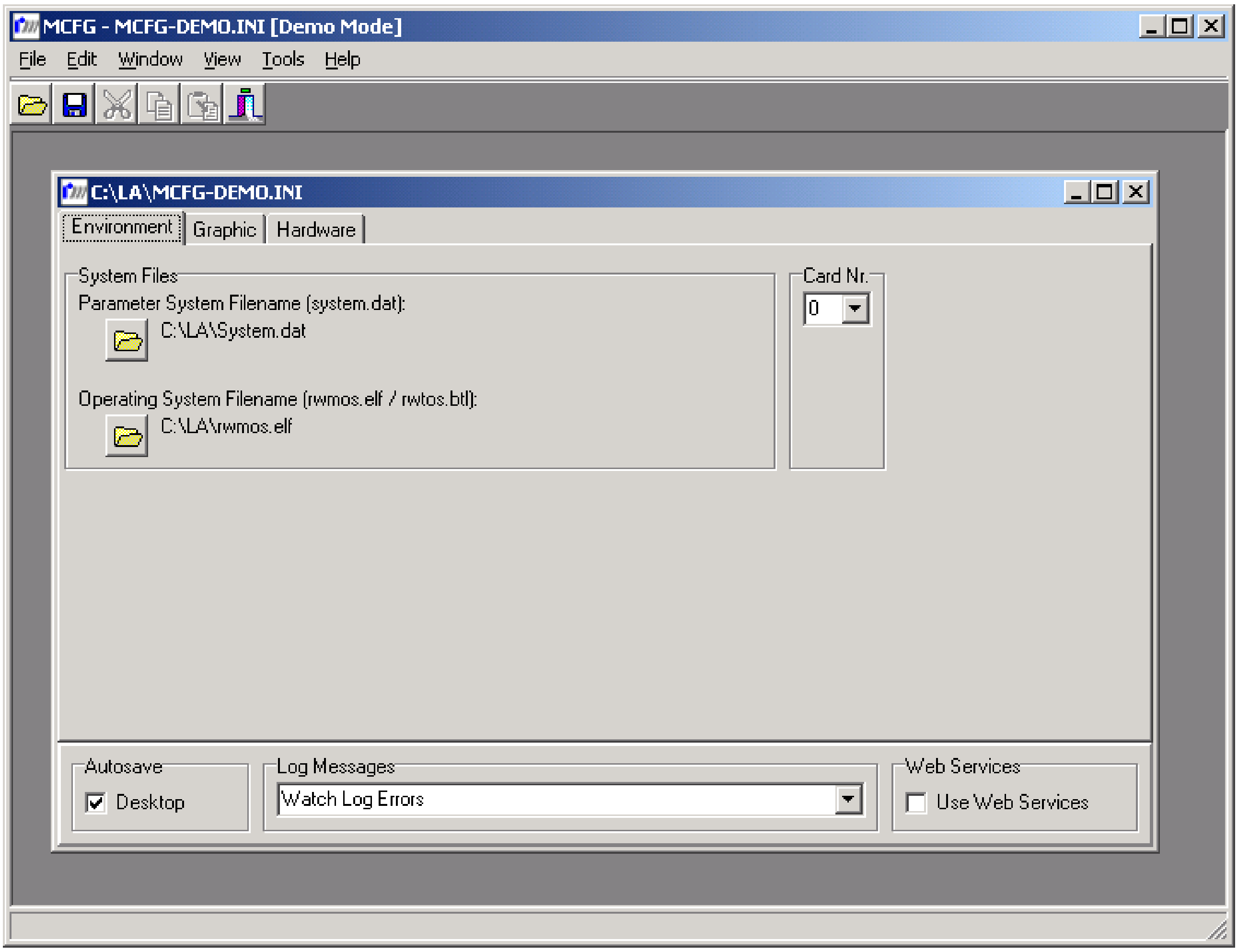
C:\cvswork\mcuglb\C++ Builder\mcfg\SYSTEM.DAT

A system file (system.dat) should first be selected from the LA and confirmed with OK so that you can continue to use the MCFG application.

1.4 Setting up project parameters

The MCFG application supports project-based system data management. For this, various states and specifications are saved in a project file (INI file).

Figure 1-2: Defining project parameters


MCFG - MCFG-DEMO.INI [Demo Mode]
File Edit Window View Tools Help
C:\LA\MCFG-DEMO.INI
Environment
Graphic
Hardware
System Files

Parameter System Filename (system.dat): C:\LA\System.dat

Operating System Filename (rwmos.elf / rwtos.btl): C:\LA\rwmos.elf

Card Nr.: 0

Autosave: ☑ Desktop

Log Messages: Watch Log Errors

Web Services: ☐ Use Web Services

Various project-specific settings can be performed via the menu items [File][Project Parameter]. This includes the system file (system.dat) created in the LA directory and control unit-specific operating system software (rwmos.elf). These two files should always be selected.

1.4.1 Additional project options

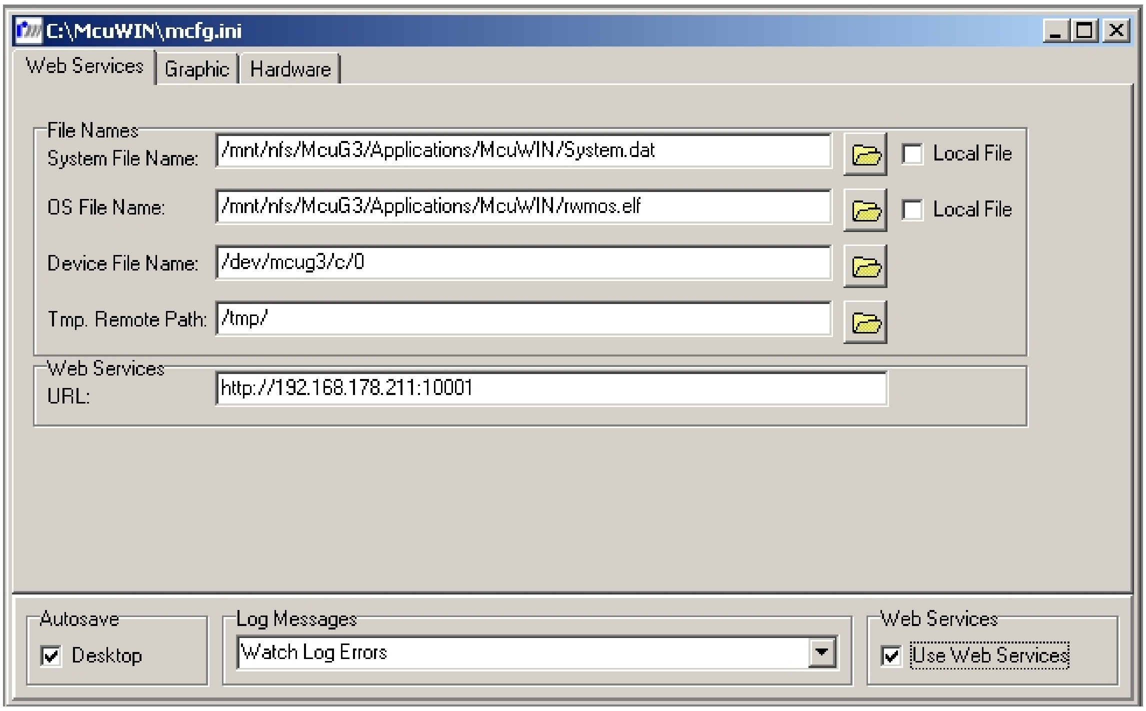
The “Autosave Desktop” option saves the current desktop of the MCFG application and restores it the next time the MCFG is started.
Various levels of error handling can be selected under “Log Messages”.
The “Web Services” option can be activated if a control module is to be accessed remotely via a network connection which is installed in a Linux system, for example.

The “Graphic” tab enables the display options for the graphic display screen to be set.

Hardware-dependent add-on options can be selected and activated on the “Hardware” tab.

1.4.1.1 Use Web Services

If the “Use Web Services” option is set in the “Project Parameter” window, the following dialog window opens:



C:\McuWIN\mcfg.ini

System File Name: /mnt/nfs/McuG3/Applications/McuWIN/System.dat ☐ Local File

OS File Name: /mnt/nfs/McuG3/Applications/McuWIN/rwmos.elf ☐ Local File

Device File Name: /dev/mcug3/c/0

Tmp. Remote Path: /tmp/

URL: http://192.168.178.211:10001

☑ Desktop

Watch Log Errors

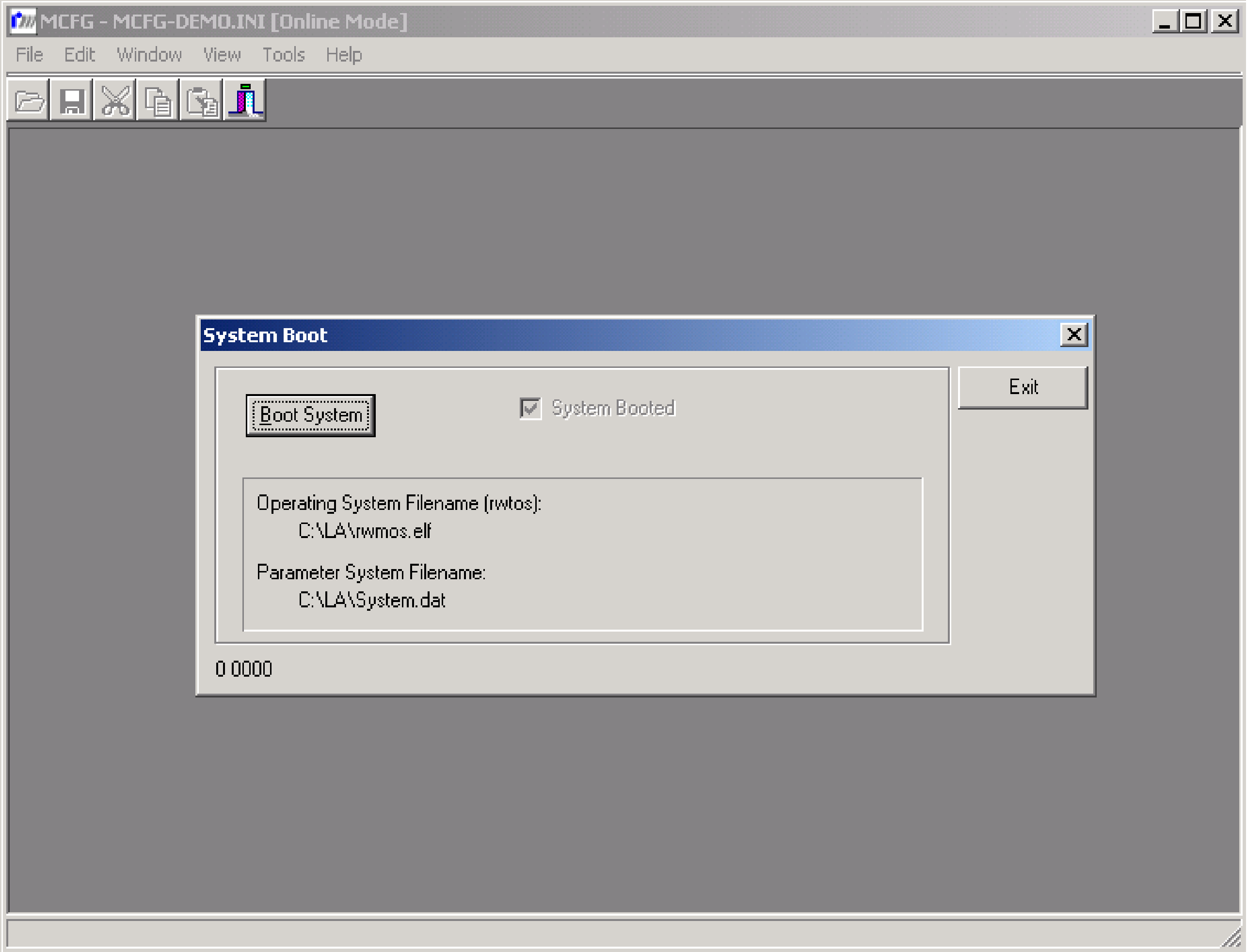
☑ Use Web Services

The access data which enables mcfg to access the APCI-8001 can be entered here. A description of this procedure can be found in the manual “PM Linux”.

1.5 Booting the control unit

At this point, the most important setups have been made and it should now be possible to boot the control unit. The boot process can be triggered via the boot dialog in [Tools][System Boot ...].

Figure 1-3: Booting the control unit



MCFG - MCFG-DEMO.INI [Online Mode]
File Edit Window View Tools Help
System Boot
Boot System
☑ System Booted
Exit

Operating System Filename (rwtos):
C:\LA\rwmos.elf

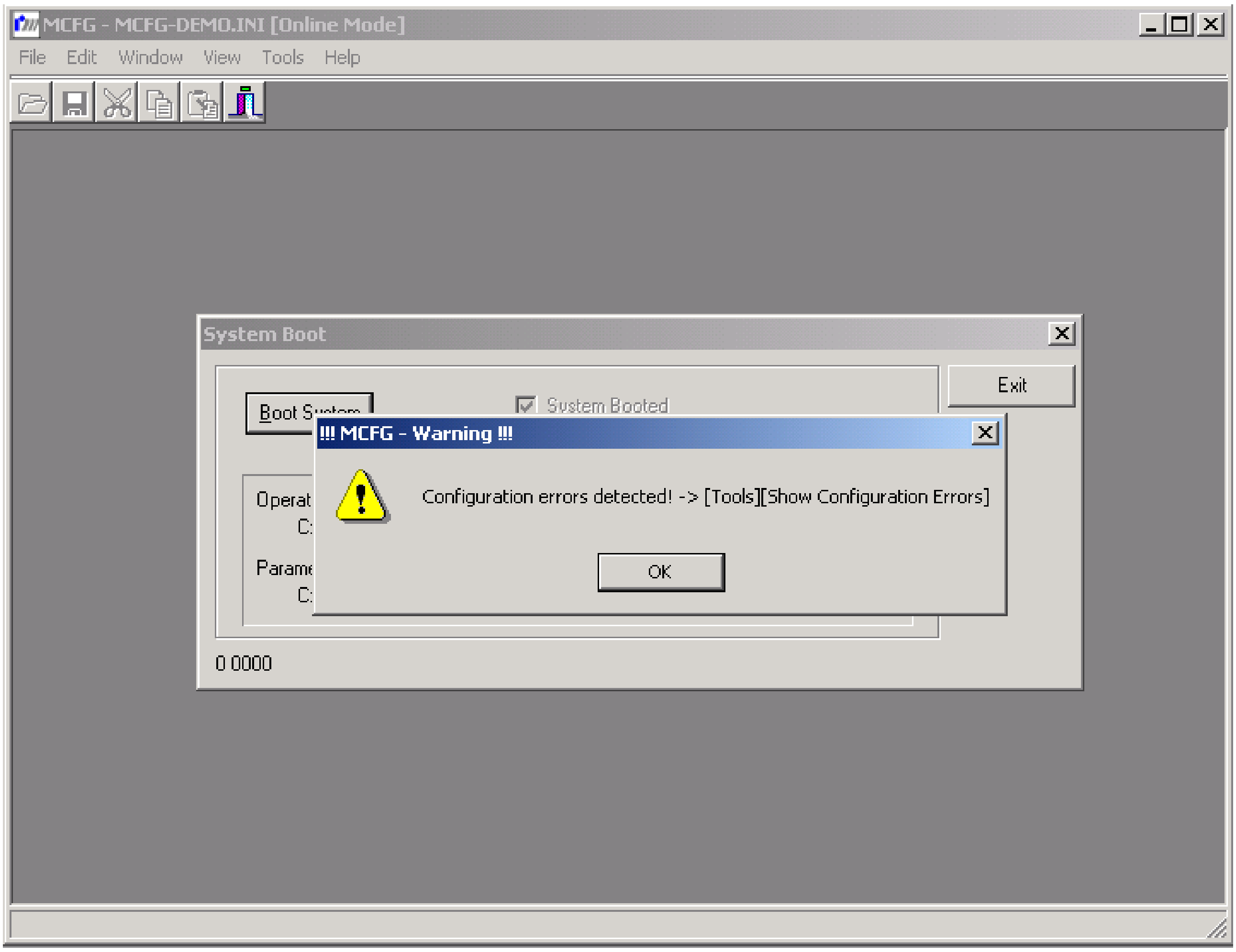
Parameter System Filename:
C:\LA\System.dat

0 0000

If the control unit has been booted successfully, this can be seen in the header line of the MCFG application. Here, the [Online Mode] entry should be displayed next to the project name.
If configuration errors are shown when booting the control unit, reference should be made to the following section.

1.6 Configuration errors

Figure 1-4: Screen message in the case of configuration errors



Configuration errors detected! -> [Tools][Show Configuration Errors]

The warning “Configuration errors detected” is shown when the control unit is booted or *mcfg.exe* is started, provided the *{cef}* error flag in the axis status register *{axst}* is set in at least one axis channel. This configuration error results from a data inconsistency between the system file (*system.dat*) and the data stored residually on the xPCI-800x module.

The configuration error can be cleared by saving the system data in the menu [File][System Data] with the command [File][Save], or by clicking the blue disk icon in the mouse palette. The “System Data” window must be open and active while saving. Before the saving operation takes place, the system must always display the warning “System will be reset”, otherwise the current system file will not match the one selected in the project [File][Project Parameter], or the control unit is still in Offline Mode.

1.7 Using *mcfg.exe*

Ideally, *mcfg.exe* should be used with mouse support. The screen has been designed with main and subforms according to MDI guidelines. This means that all windows, except dialog windows, can be positioned anywhere in the screen area of the MCFG main form. The screen session last saved can be reactivated at program start on request (see Section 1.4).

1.7.1 Help screens

Various menus are explained in more detail in help screens. The help text is even field-related in part. If help text for a desired menu or field exists, this appears after clicking the function key [F1].
Further help on using MCFG can be found via the menu [Help][Help Topics].

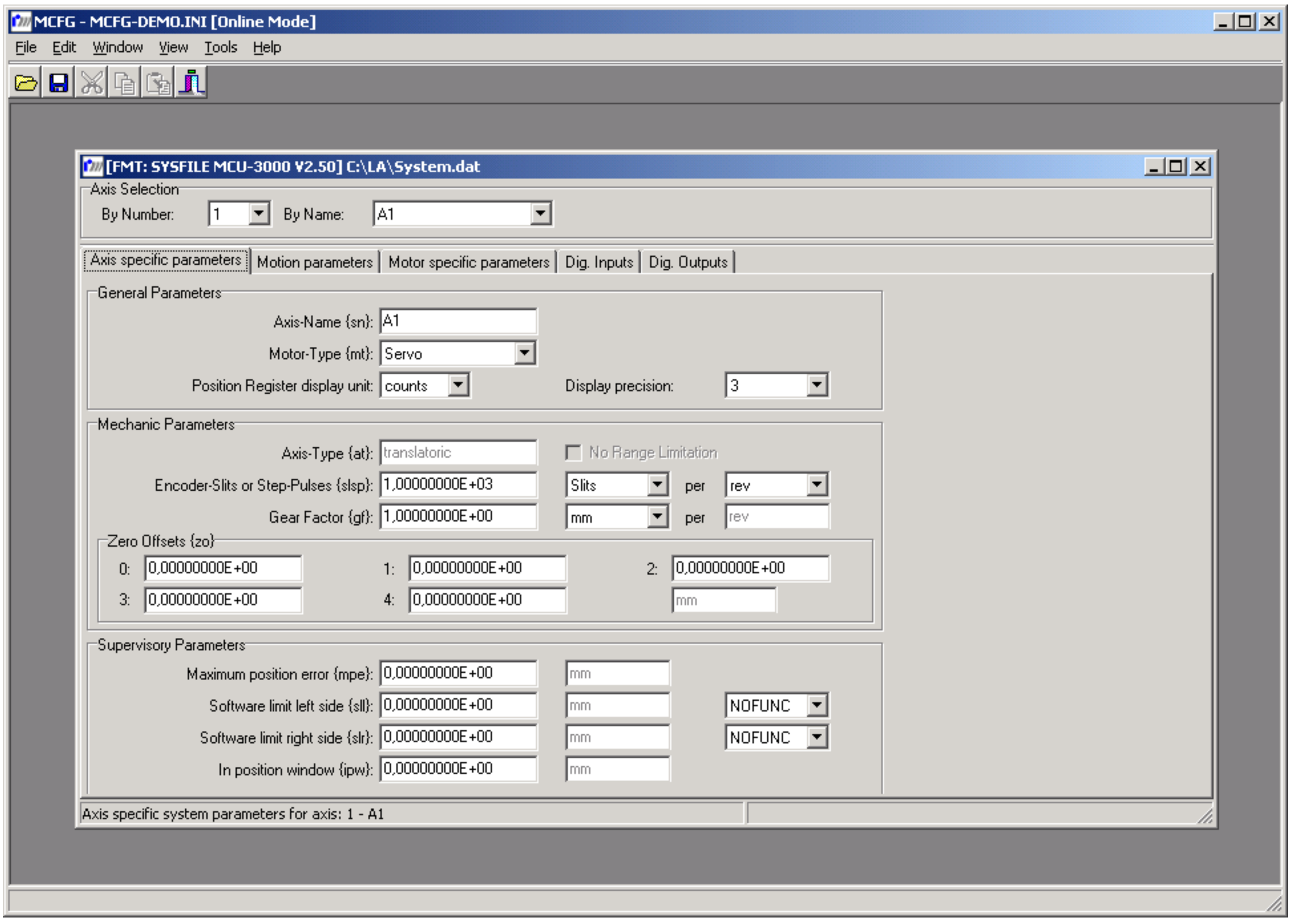
1.7.2 Editing system data [File] [System Data]

This window is used to edit the axis-specific motor and system parameters. The parameters involved are system-specific default values for the individual axis channels. Most parameters can also be requested and set during the runtime using special read and write commands. The information that is recorded here is stored in the SYSTEM.DAT file.
The correct axis number must be selected for all entries. The individual data is edited in groups under several menus.

Note: The system data must be transferred to the APCI-8001 at least once after a control system boot procedure (e.g. in mcfg.exe) so that the operating program (*rwmos.elf*) can be used on the APCI-8001. The data is loaded either from the PCAP user program or from the *mcfg.exe* utility program. In *mcfg.exe* and in the various sample programs, the system file (*system.dat*) is only transferred once.

1.7.2.1 System data, “Axis specific parameters” tab

Figure 1-5: System data, “Axis specific parameters” tab



[FMT: SYSFILE MCU-3000 V2.50] C:\LA\System.dat

Axis Selection

By Number: 1

By Name: A1

General Parameters

Axis-Name {sn}: A1

Motor-Type {mt}: Servo

Position Register display unit: counts

Display precision: 3

Mechanic Parameters

Axis-Type {at}: translatoric

☐ No Range Limitation

Encoder-Slits or Step-Pulses {slsp}: 1,00000000E+03 Slits per rev

Gear Factor {gf}: 1,00000000E+00 mm per rev

Zero Offsets {zo}

0: 0,00000000E+00

1: 0,00000000E+00

2: 0,00000000E+00

3: 0,00000000E+00

4: 0,00000000E+00

mm

Supervisory Parameters

Maximum position error {mpe}: 0,00000000E+00 mm

Software limit left side {sll}: 0,00000000E+00 mm NOFUNC

Software limit right side {slr}: 0,00000000E+00 mm NOFUNC

In position window {ipw}: 0,00000000E+00 mm

Axis specific system parameters for axis: 1 - A1

1.7.2.1.1 Symbolic axis name

Each axis channel can be assigned a symbolic *Axis-Name* {sn} with up to 10 characters. This axis name is defined automatically in the *rw_SymPas* programming language. If the axes are accessed under rw_SymPas, indicators of system variables must not be used. If using G-code programs, the axis names must not end with numeric characters.

1.7.2.1.2 Motor type

The *Motor-Type* {mt} parameter can be used to choose between stepper and servo drives. The Stepper setting should be selected if servo motors with pulse/direction setpoint value interfaces are to be used. Motor type-specific parameters are specified on the [Motor specific parameters] tab. The hardware jumper for the setpoint value channel output signals (see CM manual) must be configured according to this setting.

1.7.2.1.3 Axis type

The *Axis-Type* parameter {at} cannot be selected. It is obtained automatically by selecting the counter unit of the gear factor {gf}. If the unit concerned is a distance unit (mm, m, inch...), the axis type is specified as translatory. If angular units (rad, deg...) are selected, the axis type is defined as rotational. With rotational axes, the “No Range Limitation” flag can be used to deactivate the otherwise normal restriction of the traversing range to one revolution.

Note: If an interpolated procedure for more than one axis is required, the axes involved must be of the same type. An exception can be made here if the surface area processing methods are used (see PM).

1.7.2.1.4 Unit for displaying the position registers

In the *Position Register display unit* field, you can select a display unit used internally for *mcfg.exe*. If position setpoint or position actual values are displayed in the various status windows [Sections 1.7.6.1 and 1.7.6.3], the display will use this unit.

1.7.2.1.5 Display precision of the position register

The *Display precision* field can be used to specify the accuracy of the above-mentioned position registers. The value entered specifies the number of decimal places.

1.7.2.1.6 Encoder slits or number of steps for stepper motor axes

The number of encoder slits {slsp} is entered along with a unit of measurement *Slits*. Both rotational (angular encoders or rotary encoders) and translatory (linear scales) pulse measuring systems can be parameterised. This value is quadrupled internally as the xPCI-800x's evaluation electronics also performs a quadrupling operation. On stepper motor systems, the *Pulses* measuring unit should be selected. The *{slsp}* step pulse factor is not quadrupled. The actual value corresponds to the number of output step pulses. The reference value of the measuring system must also be selected. On stepper motors and rotary encoders, this is generally one revolution (rev). With linear encoders, this is a translatory unit, e.g. mm.

1.7.2.1.7 Gear factor

The *Gear Factor* {gf} parameter specifies a step-up or step-down ratio between the actual value pulse acquisition and feed travel or angle of rotation. The gear factor is completed by a denominator and counter unit.
The denominator unit is automatically the reference value of the measuring system. The counter unit selected is used to define two system variables. The first of these is the axis type which is defined as either translatory (linear axes) for distance units or as rotational (round, rotary axes) for angular units. The selected unit is also the basic unit for all axis-specific motion commands (*jog* commands) and their profile parameters (velocity and acceleration, described below). The gear factor must never be 0.

1.7.2.1.8 Zero offsets

Each axis channel can be assigned five different zero offsets *Zero Offsets* {*zo}*. Using the SAP and PCAP commands azo(), the desired offset parameter can be activated for the selected axis channels. Zero offsets are used to specify a new system of coordinates, without having to affect the actual machine zero.

1.7.2.1.9 Maximum position error

The *Maximum position error {mpe}* parameter is used to specify the maximum permitted deviation between the setpoint and actual position of the motor axis. If this value is exceeded, this error will not affect profile generation and position control, but it will be displayed in the *axst* axis status register. Reaction to this status register can be either event-controlled or by query.

Note: The position error monitoring function is only executed if the position control loop is closed and a value greater than zero is specified for {mpe}.

1.7.2.1.10 Software limits

For each motor axis, you can define a left-hand (*Software limit left side* {*sll*}) and right-hand *(Software limit right side* {*slr*}) software limit. If this limit is exceeded, a parameter can be used to specify how this error state is to be handled by the control unit. The following options are available:

Table 1-1: Modes of operation for software limits

Function type	Description
NOFUNC	(No Function) The software limit will be ignored.
TOM	(Turn Off Motor) No value will be output on the setpoint value channel in the case of servo drives since the axis would move deeper into the limit switch area. In the case of speed controllers, this means a speed desired value of 0 with a corresponding holding torque. But in the case of current amplifiers, this means a current desired value of 0 and no holding torque. If the position setpoint value falls below the actual position, the axis will be moved in uncontrolled mode. If the position setpoint value falls below the limit position, the limit switch status will be cancelled again.
SMA	(Stop Motor Abruptly) The axis will be abruptly stopped at the specified limit position in position control. Movement beyond the software limit is prevented. If the position setpoint value falls below the limit position, the limit switch status will be cancelled again.
SMD	(Stop Motor Deceleration) When responding to this software limit, the axis will be automatically decelerated with the axis-specific delay {*sdec}* to velocity 0 and then held in position control. Movement beyond the limit is prevented. If the position setpoint value falls below the limit position, the limit switch status will be cancelled again. This function type is particularly recommended for cascaded speed control loops and stepper motor drives.

Note: Monitoring of the software limits is only activated on referenced axes.

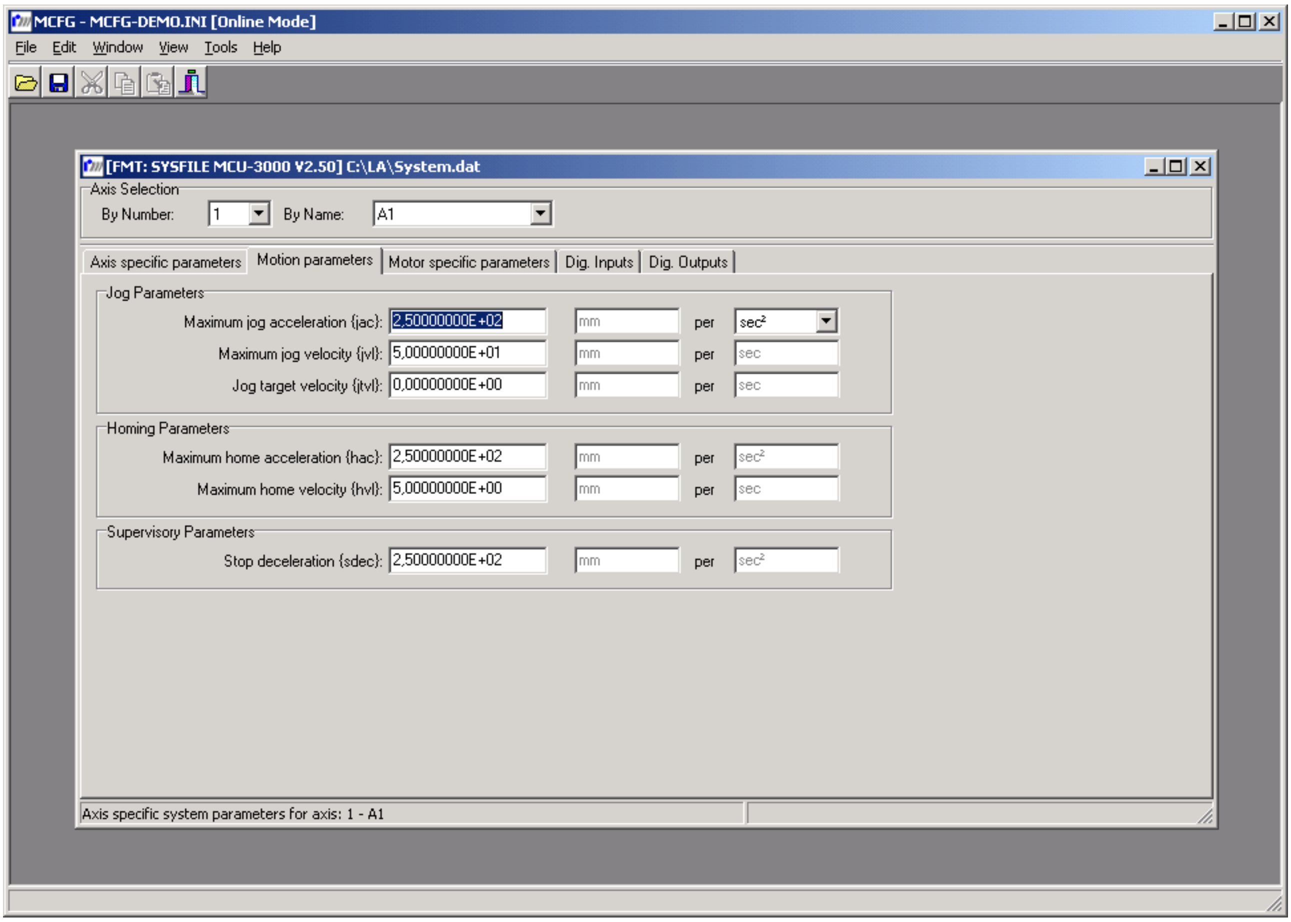
1.7.2.1.11 In-position window

The *In position window* {*ipw*} parameter is used to specify if the *ip* flag is set in the bit-coded *axst* register. This flag is set after the end of the profile has been reached (*pe* flag in *axst*) and if the position differential between the setpoint and actual position of the motor axis as specified in {ipw} is not met. All xPCI-800x digital outputs planned with the IP function are set or reset in the same manner as the *ip* flag.

Note: In-position window monitoring only takes place if a value greater than zero has been specified for {ipw}.

1.7.2.2 System data, “Motion parameters” tab

Figure 1-6: System data, “Motion parameters” tab



By Number: 1
By Name: A1

Maximum jog acceleration {jac}: 2,50000000E+02 mm per sec²
Maximum jog velocity {jvl}: 5,00000000E+01 mm per sec
Jog target velocity {jtvl}: 0,00000000E+00 mm per sec
Maximum home acceleration {hac}: 2,50000000E+02 mm per sec²
Maximum home velocity {hvl}: 5,00000000E+00 mm per sec
Stop deceleration {sdec}: 2,50000000E+02 mm per sec²

Axis specific system parameters for axis: 1 - A1

1.7.2.2.1 Jog parameters

The Jog Parameters specify the axis-specific limit data for jog positioning mode. These are acceleration *Maximum jog acceleration* {*jac*}, velocity *Maximum jog velocity* {*jvl*} and target velocity *Jog target velocity* {*jtvl*}. Normally, the target velocity is set to 0 here and may be specified by the application programming.

Note: These values are initial values which are active after transferring the system file to the control unit. As soon as these values have been overwritten by the application, the relevant programmed values become valid.

1.7.2.2.2 Homing parameters

Like the Jog Parameters, the Home Parameters *Maximum home acceleration* {*hac*} and *Maximum home velocity* {*hvl*} specify the limit data for the reference point search run. They are used automatically with all *home* commands as parameters for profile generation. Usually, the home velocity *Maximum home velocity* is only a fraction of the jog velocity, particularly when the reference switch is located near a limit switch.

Note: These values are initial values which are active after transferring the system file to the control unit. As soon as these values have been overwritten by the application, the relevant programmed values become valid.

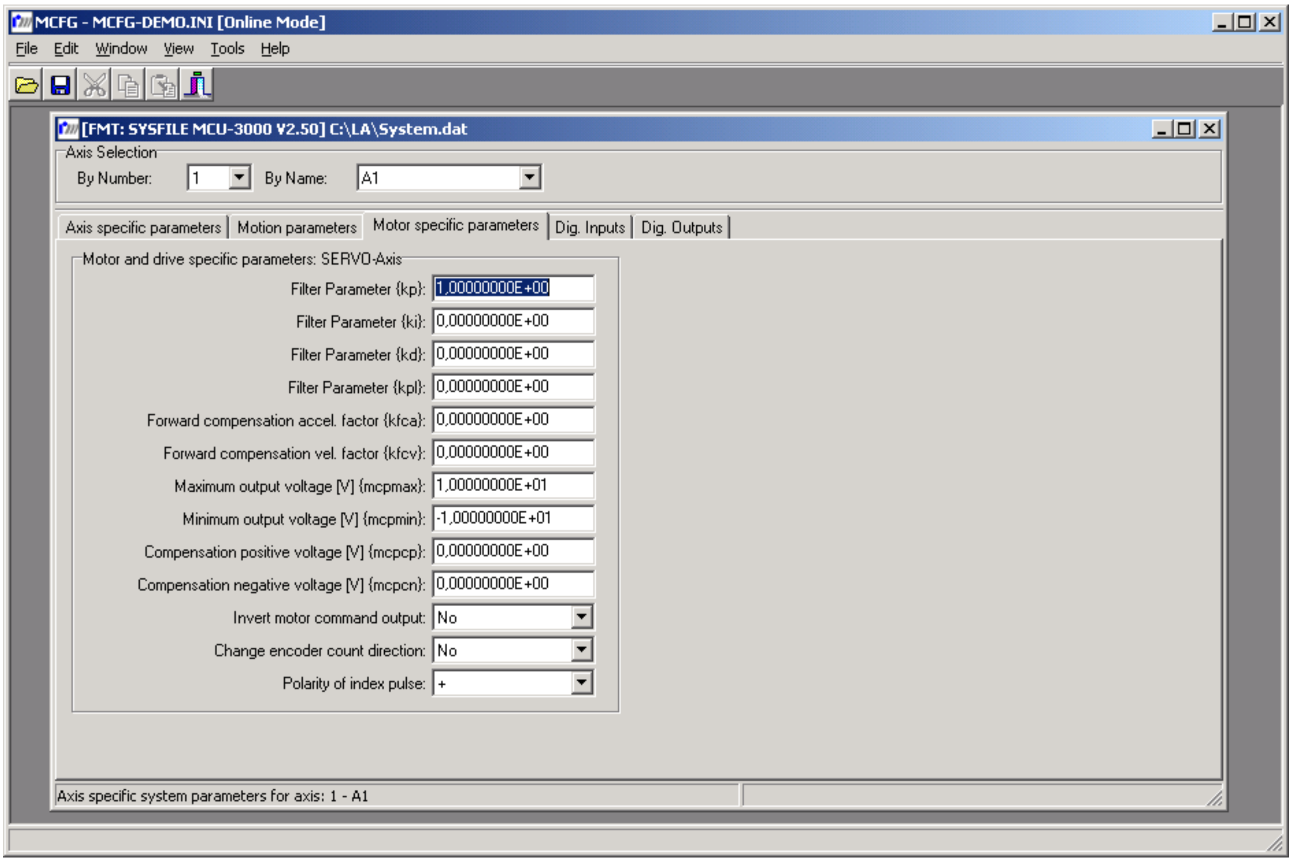
1.7.2.2.3 Supervisory parameters

The *Stop deceleration parameter {sdec}* specifies the delay with which the motor is to be decelerated when activating a hardware or software limit planned with the SMD function or when calling the jog stop command (js).

Note: This value is an initial value which is active after transferring the system file to the control unit. As soon as this value has been overwritten by the application, the relevant programmed value becomes valid.

1.7.2.3 System data, “Motor specific parameters” tab for servo axes

Figure 1-7: System data, “Motor specific parameters” tab (here servo motor axes)


MCFG - MCFG-DEMO.INI [Online Mode]
File Edit Window View Tools Help
[FMT: SYSFILE MCU-3000 V2.50] C:\LA\System.dat
Axis Selection
Axis specific parameters
Motion parameters
Motor specific parameters
Dig. Inputs
Dig. Outputs

By Number: 1
By Name: A1

Motor and drive specific parameters: SERVO-Axis

Filter Parameter {kp}: 1,00000000E+00
Filter Parameter {ki}: 0,00000000E+00
Filter Parameter {kd}: 0,00000000E+00
Filter Parameter {kpl}: 0,00000000E+00
Forward compensation accel. factor {kfca}: 0,00000000E+00
Forward compensation vel. factor {kfcv}: 0,00000000E+00
Maximum output voltage [V] {mcpmax}: 1,00000000E+01
Minimum output voltage [V] {mcpmin}: -1,00000000E+01
Compensation positive voltage [V] {mcpcp}: 0,00000000E+00
Compensation negative voltage [V] {mcpcn}: 0,00000000E+00
Invert motor command output: No
Change encoder count direction: No
Polarity of index pulse: +

Axis specific system parameters for axis: 1 - A1

1.7.2.3.1 Filter parameters

A distinction is made between stepper and servo drives for these parameters. For servo axes, a PIDF filter parameter set can be set with coefficients {kp} for proportional amplification, {ki} for integral-action coefficient, {kd} for derivative-action coefficient and {kpl} for an additional phase lead. The factors {kfca} and {kfcv} can be used to specify compensation parameters for current amplifiers or speed controllers. These enable axis positioning to be made almost free of lag error, even at high acceleration values. The filter is set according to the information given in the CM / section 6.2. For stepper motor systems, internal coefficients are used and therefore do not need to be set by the user.

1.7.2.3.2 Maximum output voltage

In the case of servo axes, the {maxmcp} and {minmcp} variables can be used to limit the manipulated variable output to the maximum and minimum values required. It is usual to set the maximum value to 10 V and the minimum value to -10V. If the maximum value is reduced, this means the manipulated variable for the power amplifier is also reduced. In the case of speed controllers, this means a reduction in the speed manipulating range, and in the case of current amplifiers a reduction in the torque range.

1.7.2.3.3 Minimum output voltage

In the case of servo axes with hydraulic motors, the {mcpcp} and [mcpcm} variables can be used to set a compensation voltage for the manipulated variable output. In regular operation, these are the minimum output voltages on the analog setpoint value channel. The system-related switch-on hysteresis of valves customarily used for controlling the hydraulic motors can thus be suppressed.
No compensation voltage is normally required with other types of servo motors.

1.7.2.3.4 Invert motor command output

The *Invert motor command output* parameter can be used to invert the sign of the manipulated variable (motor command output) in the case of servo axes. This is particularly helpful when there is phase angle rotation in the control system or when the axial direction is to be changed. The phase angle between the manipulated variable and the angle actual value is determined by the polarity of the motor lines and encoder signals or mechanical components, such as the gear unit.

1.7.2.3.5 Change encoder count direction

The *Change encoder count direction* parameter is used to invert the count direction for the pulse acquisition channels. This is particularly helpful when there is phase angle rotation in the control system or when the axial direction is to be changed.

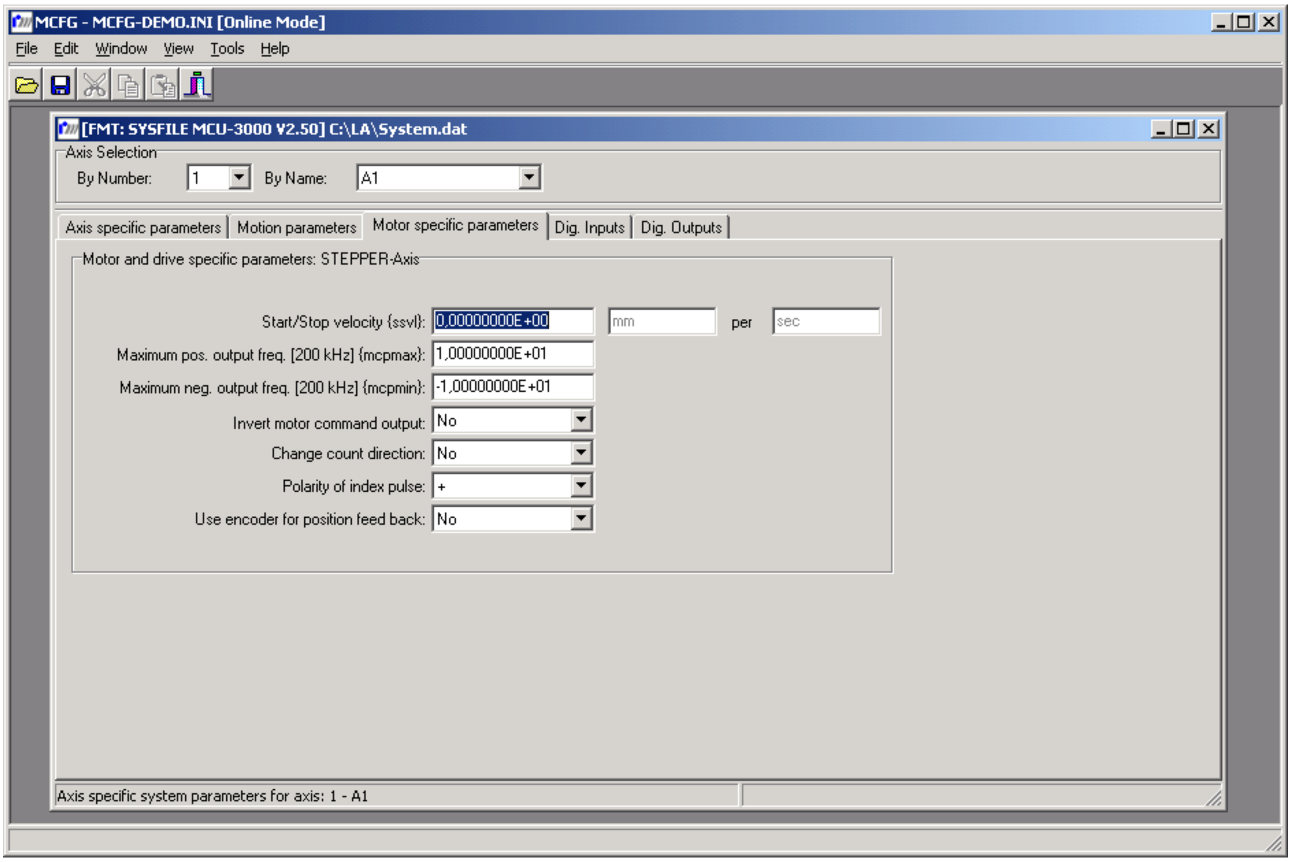
1.7.2.3.6 Polarity of index pulse

The *Polarity of index pulse* parameter is used to specify the polarity of the zero-track signal (index pulse) of a pulse encoder.

Note: The current status of the zero-track signal is also displayed in the [File][Dialog Functions][Show Digital Inputs / Status] menu in the NDX field.

1.7.2.4 System data, “Motor specific parameters” tab for stepper motor axes

Figure 1-8: System data, “Motor specific parameters” tab (here stepper motor axes)


MCFG - MCFG-DEMO.INI [Online Mode]
File
Edit
Window
View
Tools
Help
[FMT: SYSFILE MCU-3000 V2.50] C:\LA\System.dat
Axis specific parameters
Motion parameters
Motor specific parameters
Dig. Inputs
Dig. Outputs

Axis Selection

By Number: 1

By Name: A1

Motor and drive specific parameters: STEPPER-Axis

Start/Stop velocity {ssvl}: 0,00000000E+00 mm per sec

Maximum pos. output freq. [200 kHz] {mcpmax}: 1,00000000E+01

Maximum neg. output freq. [200 kHz] {mcpmin}: -1,00000000E+01

Invert motor command output: No

Change count direction: No

Polarity of index pulse: +

Use encoder for position feed back: No

Axis specific system parameters for axis: 1 - A1

1.7.2.4.1 Start/Stop velocity

This parameter is reserved for future upgrades.

1.7.2.4.2 Maximum pos. output freq./Maximum neg. output freq.

In the case of stepper motor axes, the {maxmcp} and {minmcp} variables can be used to limit the manipulated variable output to the maximum and minimum values required in units of 200 kHz. Normally, the maximum value is 10 and the minimum value is -10. Thus, the maximum frequencies are specified at 2 MHz. As a result, the input frequency on the power amplifier can be limited to valid values.

1.7.2.4.3 Invert motor command output

The *Invert motor command output* parameter can be used to invert the sign of the manipulated variable in the case of stepper motor axes. As phase angle rotation is not possible in the control system on stepper motors, this parameter only makes sense if the axial direction is to be changed. The count direction must also be changed at the same time.

1.7.2.4.4 Change count direction

The *Change count direction* parameter is used to invert the count direction for the pulse acquisition channels. As phase angle rotation is not possible in the control system on stepper motors, this parameter only makes sense if the axial direction is to be changed. The manipulated variable must also be changed at the same time.

1.7.2.4.5 Polarity of index pulse

The *Polarity of index pulse* parameter is not supported here.

1.7.2.4.6 Use encoder for position feed back

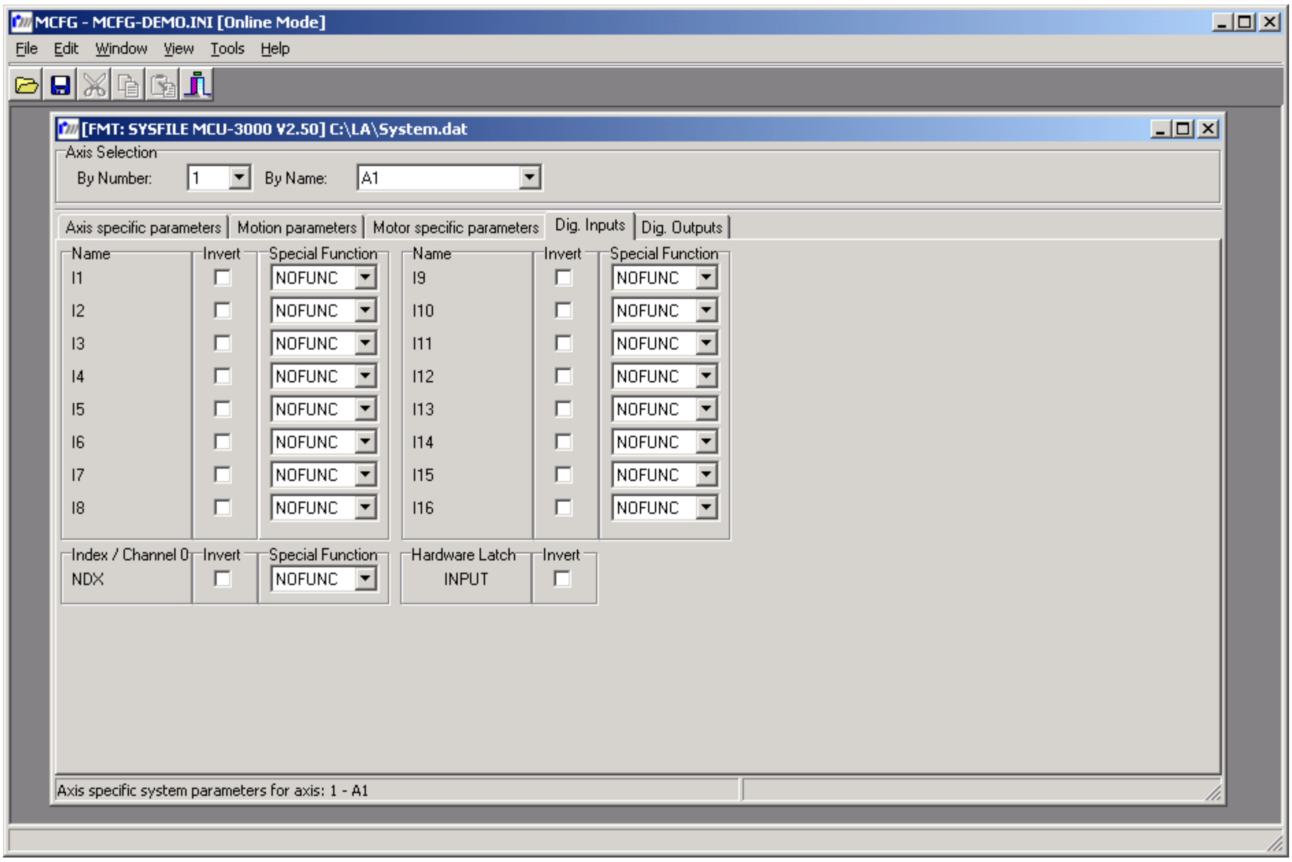
The "*Use encoder for position feed back"* option is only of importance with stepper motor systems or pulse and direction outputs, and if the *"optionEV"* option is included in RWMOS*.*
If the "*Use encoder for position feed back"* option is activated, with a software latch (LP input or lps command), instead of the current position the encoder signal (aux channel) is latched by stepper motor systems.
Furthermore, in this case and in the graphic system analysis for stepper motors, the encoder position is shown in the graphic instead of the real position rp. This can be helpful particularly in the case of servo systems with a pulse/direction interface.
With gear tracking (GEAR), in actual value tracking, the aux channel for the position calculation is also used instead of the actual position rp. If the resolution of the stepper motor system differs from that of the encoder channel, the axis-specific variable gfaux must be set when the system is started. This is not an absolute gear factor, but the ratio of encoder channel resolution to stepper motor resolution. For all axes, 1.0 is the default value of gfaux.
With the option "*Use Encoder for position feedback"*, the rdrv command or axis qualifier delivers the actual velocity of the aux channel instead of the actual value channel (rp).

Note: The axis qualifier GFAUX can be used (default value: 1.0) to scale the conversion between aux and rp for the graphic system analysis.

1.7.2.5 System data, “Digital Inputs” tab

Axis-specific planning of hardware properties for digital inputs of xPCI-800x takes place in this menu.

Figure 1-9: System data, “Digital Inputs” tab



MCFG - MCFG-DEMO.INI [Online Mode]
File
Edit
Window
View
Tools
Help
[FMT: SYSFILE MCU-3000 V2.50] C:\LA\System.dat
Axis specific parameters
Motion parameters
Motor specific parameters
Dig. Inputs
Dig. Outputs

Axis Selection

By Number: 1

By Name: A1

Name	Invert	Special Function	Name	Invert	Special Function
I1	☐	NOFUNC	I9	☐	NOFUNC
I2	☐	NOFUNC	I10	☐	NOFUNC
I3	☐	NOFUNC	I11	☐	NOFUNC
I4	☐	NOFUNC	I12	☐	NOFUNC
I5	☐	NOFUNC	I13	☐	NOFUNC
I6	☐	NOFUNC	I14	☐	NOFUNC
I7	☐	NOFUNC	I15	☐	NOFUNC
I8	☐	NOFUNC	I16	☐	NOFUNC

Index / Channel 0	Invert	Special Function	Hardware Latch	Invert
NDX	☐	NOFUNC	INPUT	☐

Axis specific system parameters for axis: 1 - A1

All digital inputs can be assigned to a special function in the *Special Function* field. The special function becomes operative whenever the correspondingly planned digital input is activated. The significance and operation of all possible special functions is explained below:

Table 1-2: Modes of operation for digital inputs

Function type	Description
NOFUNC	(No Function) The input has no special function. It serves merely as a freely programmable digital input.
REF	(Reference Switch) The inputs planned with this function act as reference or stop switches. If a digital input is activated by a reference switch (cam), for example, during the execution of special reference travel commands (e.g. *jhl()* command), the selected axis channel will be decelerated to a velocity of 0 with the axis-specific reference travel deceleration.
LSL_TOM	(Limit Switch Left Turn Off Motor) The inputs planned with this function act in the same way as left-hand hardware limit switches. When this input responds, no value will be output on the setpoint value channel in the case of servo drives since the axis would move deeper into the limit switch area. In the case of speed controllers, this means a speed desired value of 0 with a corresponding holding torque. But in the case of current amplifiers this means a current desired value of 0 and no holding torque. The limit switch is normally tripped by movement in a negative direction and exceeding the corresponding limit. If the position setpoint value falls below the actual position, the axis will be moved in uncontrolled mode. If the position falls below the setpoint value at which the limit switch was identified, the limit switch status will be cancelled again.
LSL_SMA	(Limit Switch Left Stop Motor Abruptly). This input also functions in the same manner as the left-hand hardware limit switch but, when activated, it causes the axis to be held at its current position in the position control operating mode. Movement beyond the limit is prevented. If the position setpoint value falls below the limit position, the limit switch status will be cancelled again.
LSL_SMD	(Limit Switch Left Stop Motor (with) Deceleration) The inputs planned with this function act in the same way as left-hand hardware limit switches. When this input responds, the axis is automatically decelerated with the axis-specific delay {*sdec}* to velocity 0 and then held in position control. Movement beyond the limit is prevented. The limit switch is normally tripped by movement in a negative direction and exceeding the corresponding limit. If the position setpoint value falls below the position, the limit switch status will be cancelled again. This function type is particularly recommended for cascaded speed control loops and stepper motor drives. **Note:** The *sdec* delay must be set to ensure that the drive is stopped safely without the motor axis running into a mechanical limit and thus causing damage. To protect the drive, additional hardware limit switches which only enable the power amplifiers in the permitted direction of travel should be set.
LSR_TOM	(Limit Switch Right Turn Off Motor) This mode of operation here is identical to that for LSL_TOM, except that this limit switch is planned for the right-hand limit.
LSR_SMA	(Limit Switch Right Stop Motor Abruptly). The mode of operation here is identical to that for LSL_SMA, except that this limit switch is planned for the right-hand limit.
LSR_SMD	(Limit Switch Right Stop Motor (with) Deceleration) The mode of operation here is identical to that for LSL_SMD, except that this limit switch is planned for the right-hand limit.
EO	Emergency Out) This input signals that an emergency power-off button in the drive system has been pressed. **Note:** This input has no effect on the drive system. How the user reacts to this event is at his/her discretion. This signal can be evaluated by interrogation (PM / section 4.4.44) or by means of an EVENT handler (PM / section 6.4.2)

Function type	**Description**
DR	(Drive Ready) This input signals whether the power amplifier connected to this channel is showing operational readiness. This signal can be evaluated by interrogation (PM / section 4.4.44) or by means of an EVENT handler (PM / section 6.4.3).
UI	(User Input) This input has no effect on the drive system. Since an EVENT handler is also available for this input type, the alternative cyclical interrogation (polling) of inputs can be dispensed with. This signal can be evaluated by interrogation (PM / section 4.4.44) or by means of an EVENT handler (PM / section 6.4.7).
LP	(Latch Position) When activating this input, the actual position {rp} of the corresponding motor axis is stored temporarily. If a latch procedure has been triggered, the *lpsf* flag of the *axst* register is set. The temporarily stored position can then be read in using the PCAP command *rdlp()* [PM / section 4.4.82] or the SAP axis qualifier *lp*. The *lpsf* flag is automatically deleted by the read-in procedure. The maximum delay of the latch procedure is 2 scanning intervals (2.56 ms).

1.7.2.5.1 Invert the xPCI-800x digital inputs

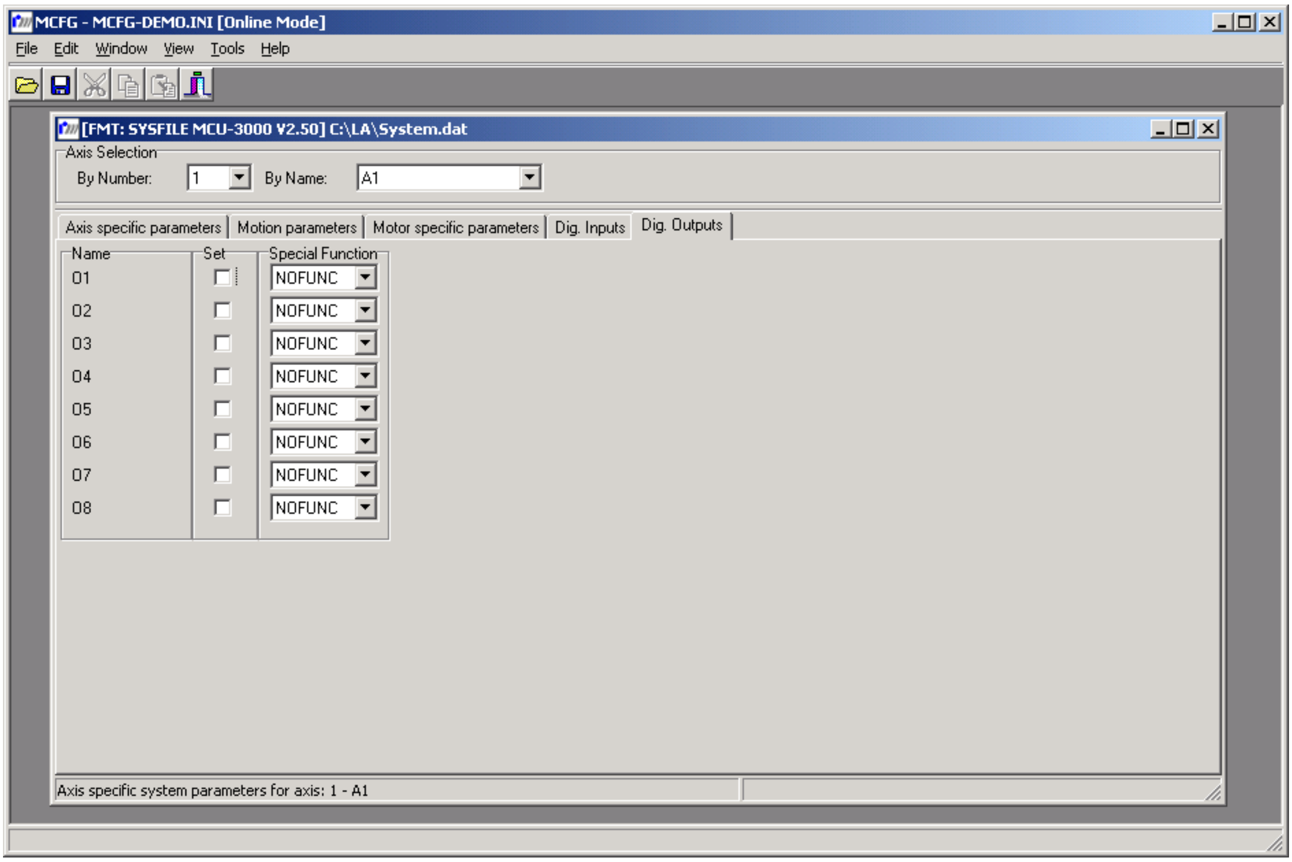
All xPCI-800x digital inputs can be inverted for the specific axis using the software. The desired selection is made in the *Invert* field using the spacebar or by clicking with the mouse. NC or NO contacts can thus be operated at the respective inputs without additional hardware. The information on whether or not an input is to be inverted is also saved in the non-volatile working memory of the xPCI-800x control unit. It is also possible to invert the active edge of the zero-track signal and the hardware latch strobe input for hardware latch processes.

Note: In the factory, all inputs of the **APCI-8001** system electronics have been planned without inversion. The inputs are activated when a voltage of +24 V is applied.

1.7.2.6 System data, “Digital Outputs” tab

Axis-specific planning of hardware properties for digital outputs of xPCI-800x takes place in this menu.

Figure 1-10: System data, “Digital Outputs” tab


MCFG - MCFG-DEMO.INI [Online Mode]
File
Edit
Window
View
Tools
Help
[FMT: SYSFILE MCU-3000 V2.50] C:\LA\System.dat
Axis Selection
Axis specific parameters
Motion parameters
Motor specific parameters
Dig. Inputs
Dig. Outputs

By Number: 1

By Name: A1

Name	Set	Special Function
01	☐	NOFUNC
02	☐	NOFUNC
03	☐	NOFUNC
04	☐	NOFUNC
05	☐	NOFUNC
06	☐	NOFUNC
07	☐	NOFUNC
08	☐	NOFUNC

Axis specific system parameters for axis: 1 - A1

All digital outputs can be assigned to a special function in the *Special Function* field. The significance and operation of these special functions is explained below:

Table 1-3: Modes of operation for digital outputs

Function type	Description
NOFUNC	(No Function) The output has no special function. It serves merely as a freely programmable digital output.
PAE	(Power Amplifier Enable) This output is set whenever the corresponding axis channel is switched into position control. It is used to enable the external power amplifier module. This is enabled by the *cl()* command. The output is reset as soon as the position control is switched off. This is the case, for example, with the *ra(), rs()* or *ol()* commands. The power amplifiers must be enabled in the following situations, for example: In interference situations or due to the system-related offset drift in the case of speed control devices. **Note:** All outputs are reset in terms of the hardware whenever the hardware is reset following a power-on, power-fail or reset procedure. The commercially available power amplifiers are enabled with a potential-free relay contact. These could be controlled by a PAE planned output. A relay point (NO) is provided at the 10-pin FB connector X5 for each axis channel. In the factory, this relay point is planned with PAE function.
IP	(In position) This output is set whenever the axis channel involved has reached the end of its profile and, in addition, the actual position and the setpoint position are within the position window specified in {ipw} [section 1.7.2.1.11].
MPE	(Maximum position error) This output is set whenever the position error specified in {mpe} is exceeded [section 1.7.2.1.9].
SIGN SPEC	In the case of stepper motor axes with encoder evaluation, this function is used to assign the sign signal to a digital output. This option can also be used for application-specific special functions. This is documented separately if required.

1.7.2.6.1 Initial state of the xPCI-800x digital outputs

All digital outputs can be assigned a default value in the *Set* field. This value is output after system initialisation, especially after booting or a hardware reset. During the reset operation, 0 V is output at all digital outputs.
The initial state of the digital outputs can be set for a specific axis. However, the value “1” or “set” is dominant, i.e. if “set” is activated for any axis, the significance of other axis plans is no longer of any importance.

1.7.2.7 Save system data [Save Changes]

All hardware and software parameters in the *system.dat* system file are saved under the menu item [File][Save]. The same action is triggered by activating the blue disk symbol in the mouse palette.
The save operation also enables various system information to be stored as resident on the xPCI-800x module. System information saved includes, for example, the motor types and the inversion of inputs including parameters.
If an error occurs while saving, a screen then appears with the error message: “Configuration errors detected” [section 1.6].
This save operation must also take place at least once after replacing a module in an existing application so that the system can be run.

After saving, the newly programmed system data is available on the xPCI-800x.

Note: The system data must be transferred to the xPCI-800x at least once after a boot procedure so that the operating program (*rwmos.elf*) can be used on the xPCI-800x. The data is loaded either from the PCAP user program or from the *mcfg.exe*. utility program.
In *mcfg.exe* and in the various PCAP example programs, the system file *system.dat* is transferred just once.

1.7.3 Motion Tools

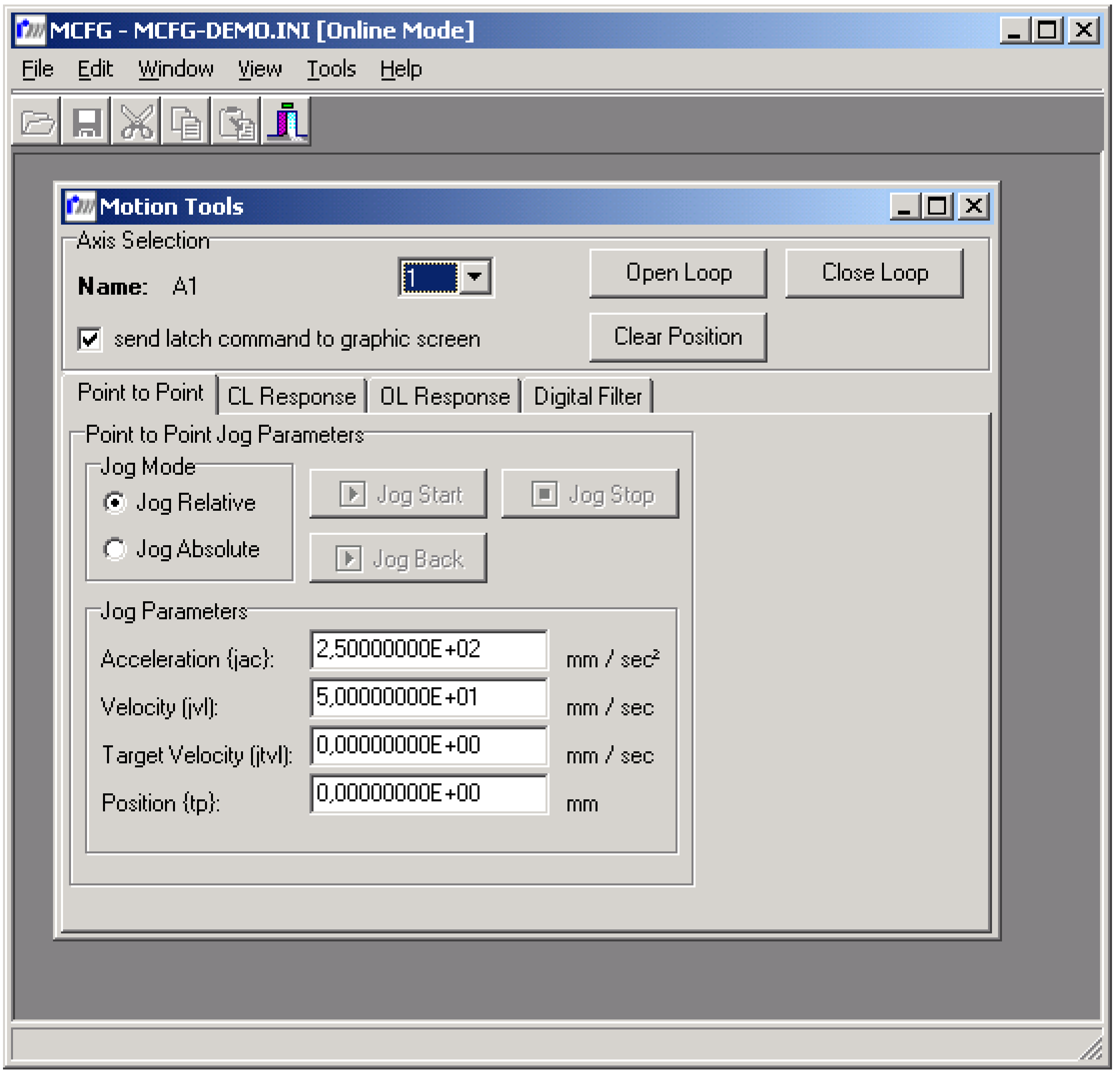
1.7.3.1 Point to Point

This menu enables rapid control of the selected drive. Here, particular attention should be paid to the "Graphic system analysis" in section 1.7.4 using which motion processes can be recorded as a graph. The selected axis channel can be moved absolutely to the desired target position (angle) or relatively to the specified traverse path (angle). In this context, the information on the APCI-8001 position controller (PM / section 2.1) and on APCI-8001 profile generation (PM / section 2.2) should be taken into consideration.

Parameters for point-to-point motion [MOVING PARAMETERS]

Name	Function
Axis #	Shows the axis number of the axis channel currently selected with which the point-to-point motion is executed.
Jog Mode	Selects jog mode. Jog Absolute means that the path or angle specified in position is based on the machine zero and that the position is moved directly. Jog Relative means that starting from the current position, the movement made is relative (also incremental) to the path or angle specified in position.
Acceleration {jac}	Acceleration value for the point-to-point movement to be executed, taking into account the units selected.
Velocity {jvl}	Maximum velocity value for the point-to-point movement to be executed, taking into account the units selected.
Target velocity {jtvl}	Target velocity for the point-to-point movement to be executed, taking into account the units selected. **Note:** With a target velocity of <> 0 the system will continue infinitely.
Position {tp}	Target position (angle) or relative traverse path (angle). The selected traverse path is run in the axis-specific unit.
Open Loop	Opens a position control loop. Aborts any profile running at that time. The value 0 is also output on the setpoint value channel. All digital outputs planned with the PAE function are set to inactive.
Close Loop	Closes a position control loop. The actual position is transferred as the target position. All digital outputs planned with the PAE function are set to active.
Jog Start	Mouse button to start the movement profile taking into account the parameters specified above. However, so that the movement can be executed, the position control loop must be closed (Close Loop). If the "send latch command to graphic screen" flag is set and if a "Graphic" window is opened for the selected axis, clicking on this button will simultaneously trigger a recording process for the graphics screen. This procedure can be used to verify the behaviour of the axes when setting filter parameters.
Jog Stop	Mouse button for stopping the current movement profile. The axis decelerates to a standstill using the programmed StopDeceleration {sdec} (see section 1.7.2.2.3).
Jog Back	Mouse button for going back to the last movement profile which was executed with Jog Start.

Figure 1-11: Motion Tools, “Point to Point” tab



Axis Selection

Name: A1 1

☑ send latch command to graphic screen

Jog Mode

● Jog Relative

○ Jog Absolute

Jog Parameters

Acceleration {jac}:	2,50000000E+02	mm / sec²
Velocity {jvl}:	5,00000000E+01	mm / sec
Target Velocity {jtvl}:	0,00000000E+00	mm / sec
Position {tp}:	0,00000000E+00	mm

This mode of operation is particularly well-suited to assessing the control and positioning behaviour of the axis channel selected. Using the specified system parameters, the motor is moved absolutely or relatively using a trapezoidal speed profile. Normally, the control behaviour is optimised first by adapting the filter parameters with small acceleration and velocity parameters. Ideally, the profile of the setpoint and actual values in the graphic display should be as similar as possible. If the filter parameters are adjusted to the system, the limit values for acceleration and velocity can be determined with this mode of operation. Here again, the aim is for small difference between the setpoint and actual value in order to ensure the best possible positioning behaviour.

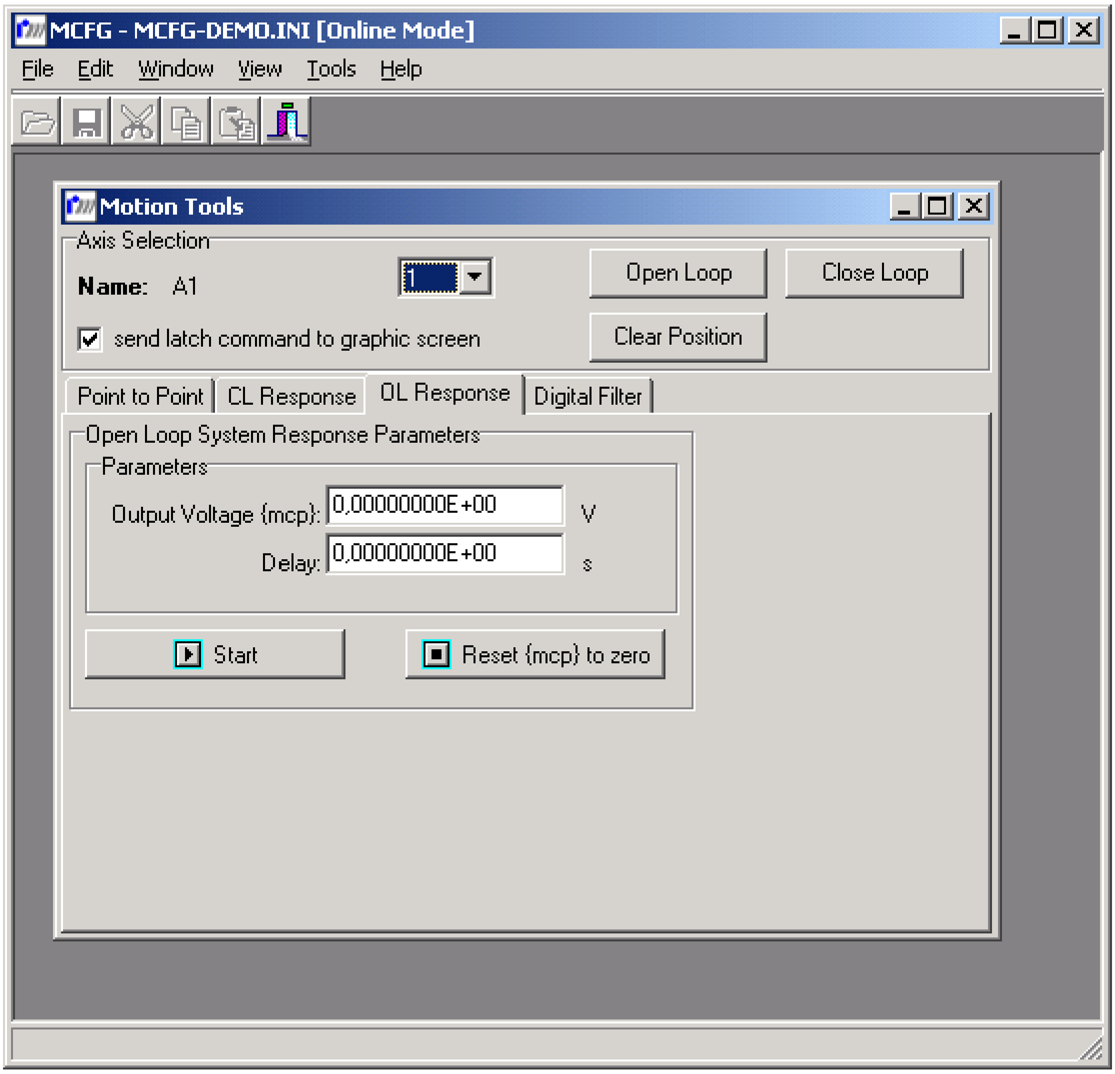
This mode of operation is also suitable for the experimental determination of the compensation parameters {kfca} and {kfcv}. In this case, the other filter parameters are set to 0. The compensation parameters {kfca} and {kfcv} are now set so that the profile of the setpoint and actual values match as well as possible. This setting should be carried out with medium acceleration.

1.7.3.2 “CL Response” tab

The command for a setpoint jump on an axis can be given on this tab.

1.7.3.3 “OL Response” tab

Figure 1-12: Motion Tools, “OL Response” tab



Name: A1

Axis: 1

☑ send latch command to graphic screen

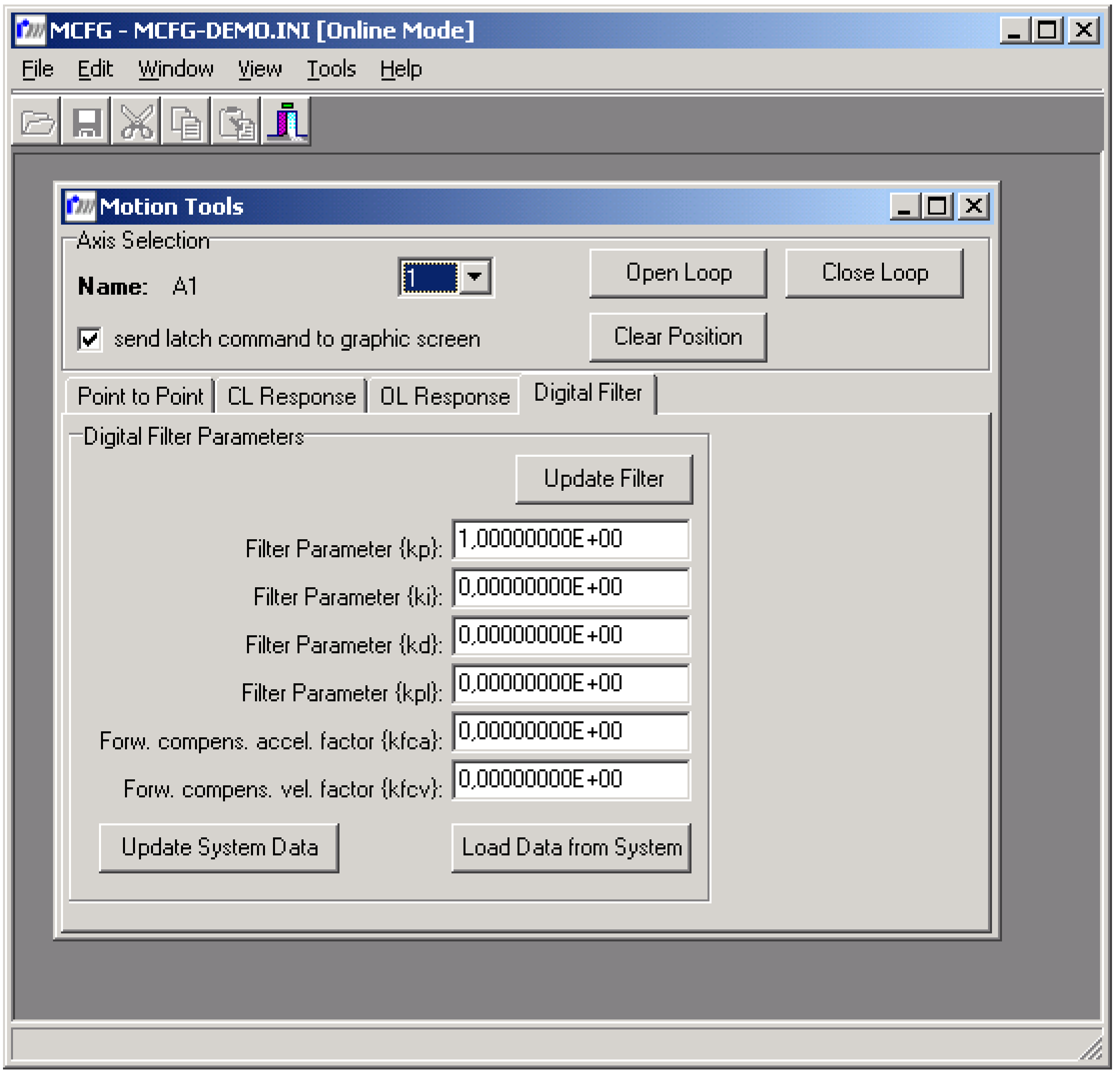
Output Voltage (mcp): 0,00000000E+00 V

Delay: 0,00000000E+00 s

In this mode of operation, a pulse with the selected output voltage and delay is output on the selected axis channel. The control loop is open in this mode of operation. As in section 1.7.3.1, the movement process (jump response) triggered can also be recorded here and displayed in a graphic screen. This jump response can be used to dimension the controller parameters. Upon initial operation, particular attention should be paid to whether a positive output value, and/or a positioning movement in a positive count direction, has an impact on recording actual values. If this is not the case, the phase assignment must be justified by suitable measures since otherwise the drive system would immediately pull in one direction when the control loop is closed. Suitable measures include inverting manipulated variable output, inverting the actual value count direction or changes to wiring.
Furthermore, a jump response recorded in this way can be used directly by current-controlled systems for semi-automatic filter design.

1.7.3.4 “Digital Filter” tab

Figure 1-13: Motion Tools, “Digital Filter” tab



Name: A1 — 1

☑ send latch command to graphic screen

Filter Parameter {kp}: 1,00000000E+00

Filter Parameter {ki}: 0,00000000E+00

Filter Parameter {kd}: 0,00000000E+00

Filter Parameter {kpl}: 0,00000000E+00

Forw. compens. accel. factor {kfca}: 0,00000000E+00

Forw. compens. vel. factor {kfcv}: 0,00000000E+00

Filter parameters can be entered in this screen and temporarily activated using the “Update Filter” button.
To transfer the filter parameters shown as resident in the stored system data (System.dat), first click on the “Update System Data” button. This transfers the values to the system data mask as per 1.7.2.3.1. This data can then be stored (see section 1.7.2.7).
To read the values displayed from the system data mask, click on the “Load Data from System” button.

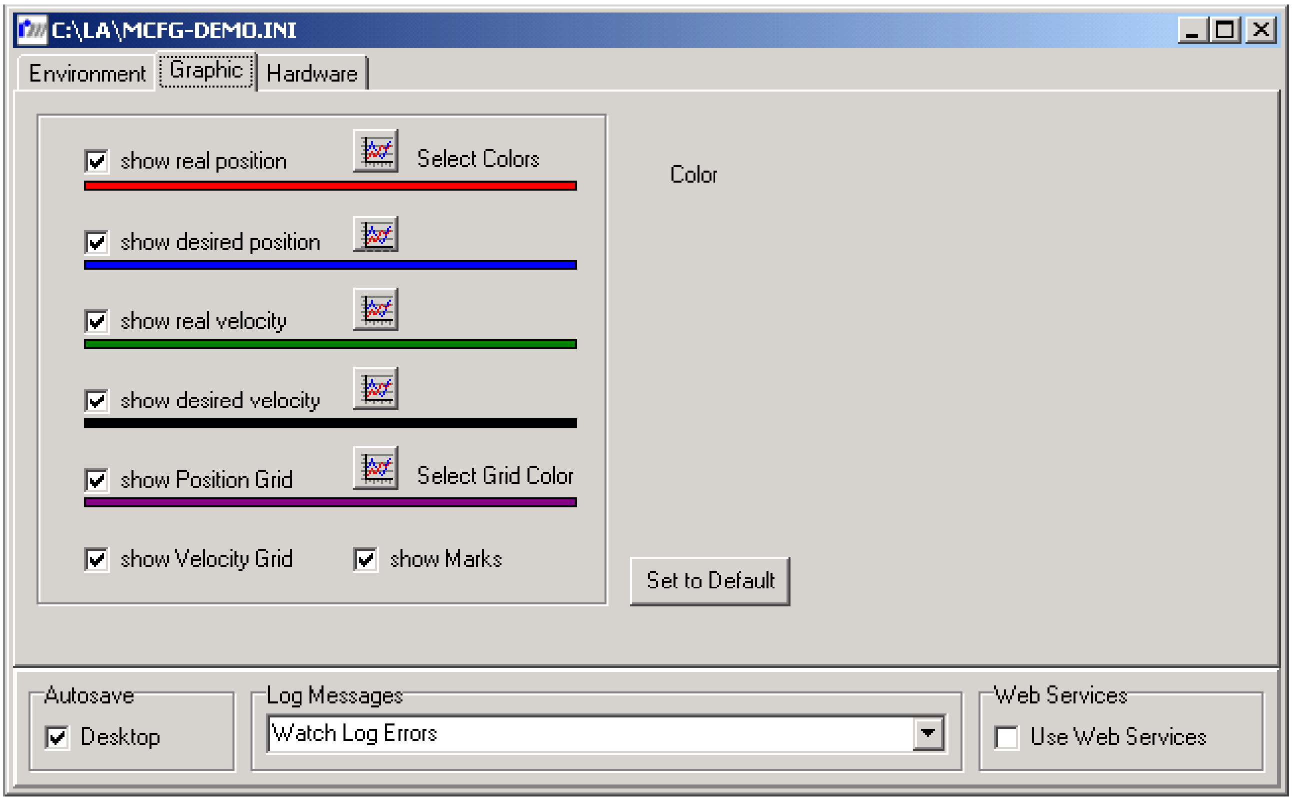
1.7.4 Graphic system analysis

One further important property of the *mcfg.exe* utility program is the option to display various axis-specific control and process variables on the screen as graphics.

1.7.4.1 Setup for graphic system analysis

First of all, the settings required for the graphic system analysis need to be made. These are made on the “Project Parameter” [File][Project Parameter] form and the Graphic tab (see also section 1.4).

Figure 1-14: Project Parameter, “Graphic” tab



☑ show real position

☑ show desired position

☑ show real velocity

☑ show desired velocity

☑ show Position Grid

☑ show Velocity Grid

☑ show Marks

Autosave: ☑ Desktop

Log Messages: Watch Log Errors

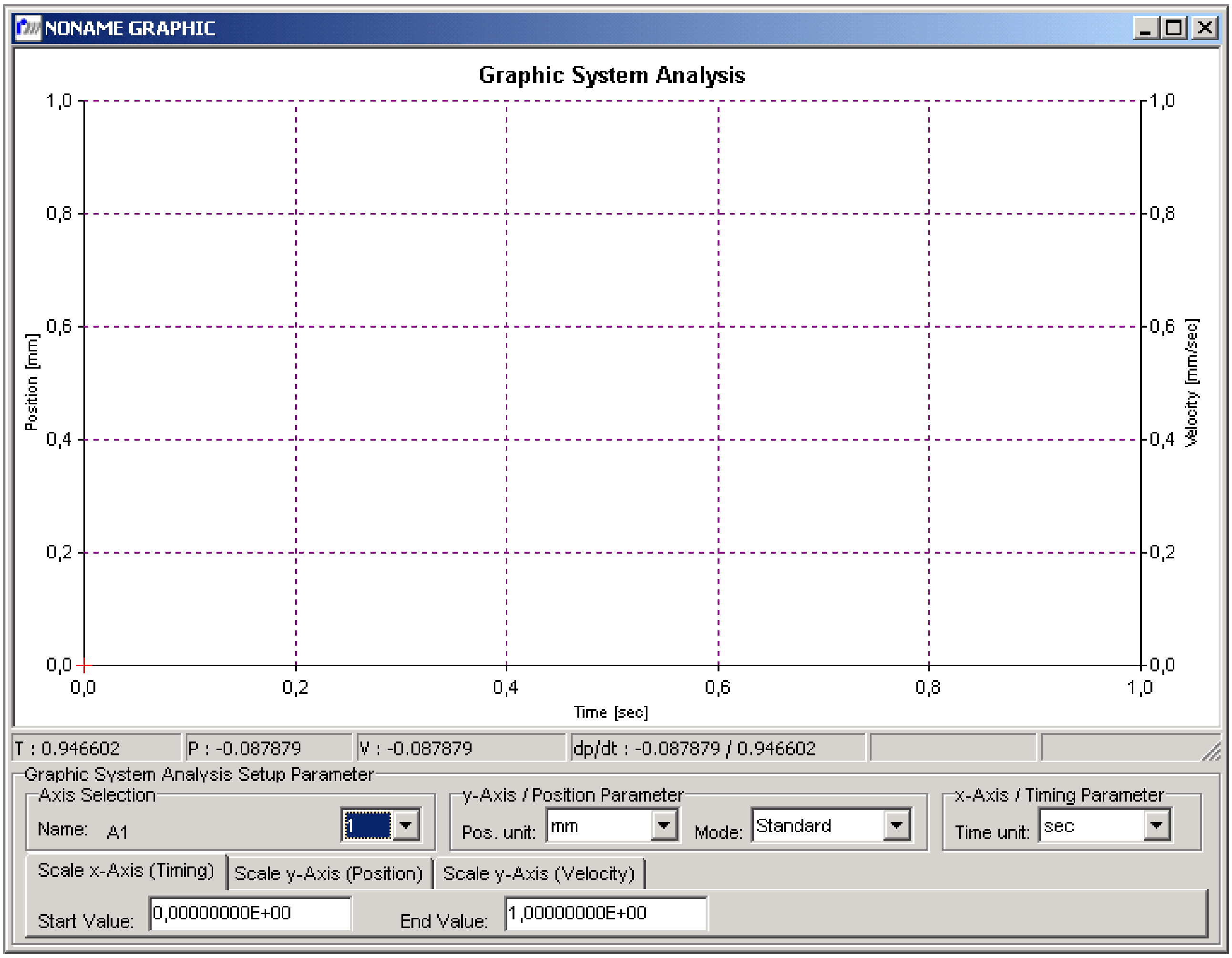
Web Services: ☐ Use Web Services

Various details on selecting the graphs and assigning the colours can be set in this menu. Also, it is possible to select or deselect setpoint and actual values of positions and velocities.

1.7.4.2 Graphic analysis window

The following screen is shown under the menu item [File][Graphic Analysis]. This is a system of coordinates with details on time, position and velocity. The coordinates are selected using the methods described in section 1.7.4.1.

Figure 1-15: Graphic system analysis



T : 0.946602 | P : -0.087879 | V : -0.087879 | dp/dt : -0.087879 / 0.946602

Name: A1 1

Pos. unit: mm

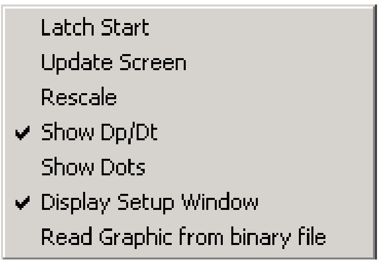
Mode: Standard

Time unit: sec

Start Value: 0,00000000E+00

End Value: 1,00000000E+00

It can be used via the menu bar or by right-clicking in the Graphic window. The following menu selection opens:



1.7.4.3 Scale the Graphic screen [Graph Scale Parameters]

In the lower part of the screen you can specify the scale for the graphic output. The scaling can be carried out for each of the position actual and setpoint values and for the velocity actual and setpoint values. 1,000 measuring values are available for recoding the actual and setpoint values. These measuring values are stored in hard real time in the internal working memory of the xPCI-800x devices. With this number, the recording duration of 1.28s (1,000 * scanning time) can be reached with a set cycle time of 1.28ms. If the recording duration is greater than this value, waiting times are added by xPCI-800x between the individual measuring times. Thus, recordings of virtually any duration are possible. The timely latching and backup up of data is carried out on the xPCI-800x here. A latch process can therefore be started and the result displayed on the screen one hour later, for example. Here, it should be noted that only 1,000 values are ever recorded for each recording process. When recording over a longer period, the display cannot be closed in any way. If this is desired, “scanner interface” methods can be used, for example, which are documented separately.

Note: So that a change in the path unit or angle unit takes effect during the scaling, latching must be carried out again before displaying the graphic screen. By contrast, path limit values, time units and time limit values can be changed as desired without latching again.

1.7.4.4 Record axis movements [Latch Start]

This menu item can be used to start a recording procedure. Normally, however, recording is triggered by an open loop jump [1.7.3.3] or by starting a movement profile [1.7.3.1] in the Motion Tools window [1.7.3].

1.7.4.5 Show graphs [Update Screen]

The selected graphs are shown on the screen with the corresponding scaling parameters specified. The vertical axis is always scaled in the colours of the actual values.
The following points must also be considered in order to graphically analyse the axis-specific positioning and control behaviour: unlike the actual values, the setpoint values are always specified with decimal places. For this reason, even at high resolution, the graphs may differ greatly, particularly in the velocity graphs, since these are calculated through differentiation from the path information. A system-related offset between setpoint and actual values must also be taken into account since the actual values are only ever available at the start of a new scanning interval. Despite these limitations, the graphic output offers a simple and effective way of optimising the positioning and control behaviour of the drive axes.

1.7.4.6 Save graphs [SAVE]

Graphs currently shown can be saved on the hard drive as a file with the extension .grf and can be restored later on a different system.
The add-on program GetGraph.exe offers another more extensive option to store entire records. With this program, all data recorded on an axis is read out directly by the control unit and saved on the hard drive in a file with the extension .grb. These files can be displayed with the menu item [Read Graphic from binary file].

1.7.4.7 Rescale

With the shift key held down, the left mouse button can be used to select an area of the screen which is to be shown zoomed in or out when the mouse button is released. By left-clicking in the Graphic window while holding down the shift key, the original scaling can be restored.

1.7.4.8 Set zeros

By left-clicking on a point in the graphic while holding down the Ctrl key, the position and time displays under dp/dt in the status lines directly beneath the graphic display are set to zero. This means that a difference measurement to another point can be made by positioning the mouse cursor.

1.7.4.9 Calculate acceleration

With the Alt key held down, the left mouse button can be used to draw a straight line over an increase in velocity. After releasing the mouse button, the window for filter design opens. This shows the acceleration value of the slope of the straight line.

1.7.4.10 Calculate controller parameters for current-controlled systems

For current-controlled drive systems a filter parameter set can be calculated semi-automatically. Here, the jump response of the open control loop should first be recorded. This should produce a velocity profile with a linear increase in velocity. With the ALT key held down, the left mouse button can now be used to draw a straight line in this velocity slope. After releasing the mouse button, a parameter window is opened in which the limit frequency of the closed control loop can be entered (default 30 Hz). The parameter calculation can then be started by clicking on the OK button. The parameters are entered in the “Digital Filter” tab of the Motion Tools window and can be immediately taken over here for a test.

1.7.4.11 Show the position error profile

The “Position Error” option can be selected in the Mode selection menu. After a screen update, the setpoint profile and the position error profile can be displayed above the positioning movement recorded in the Graphic window.

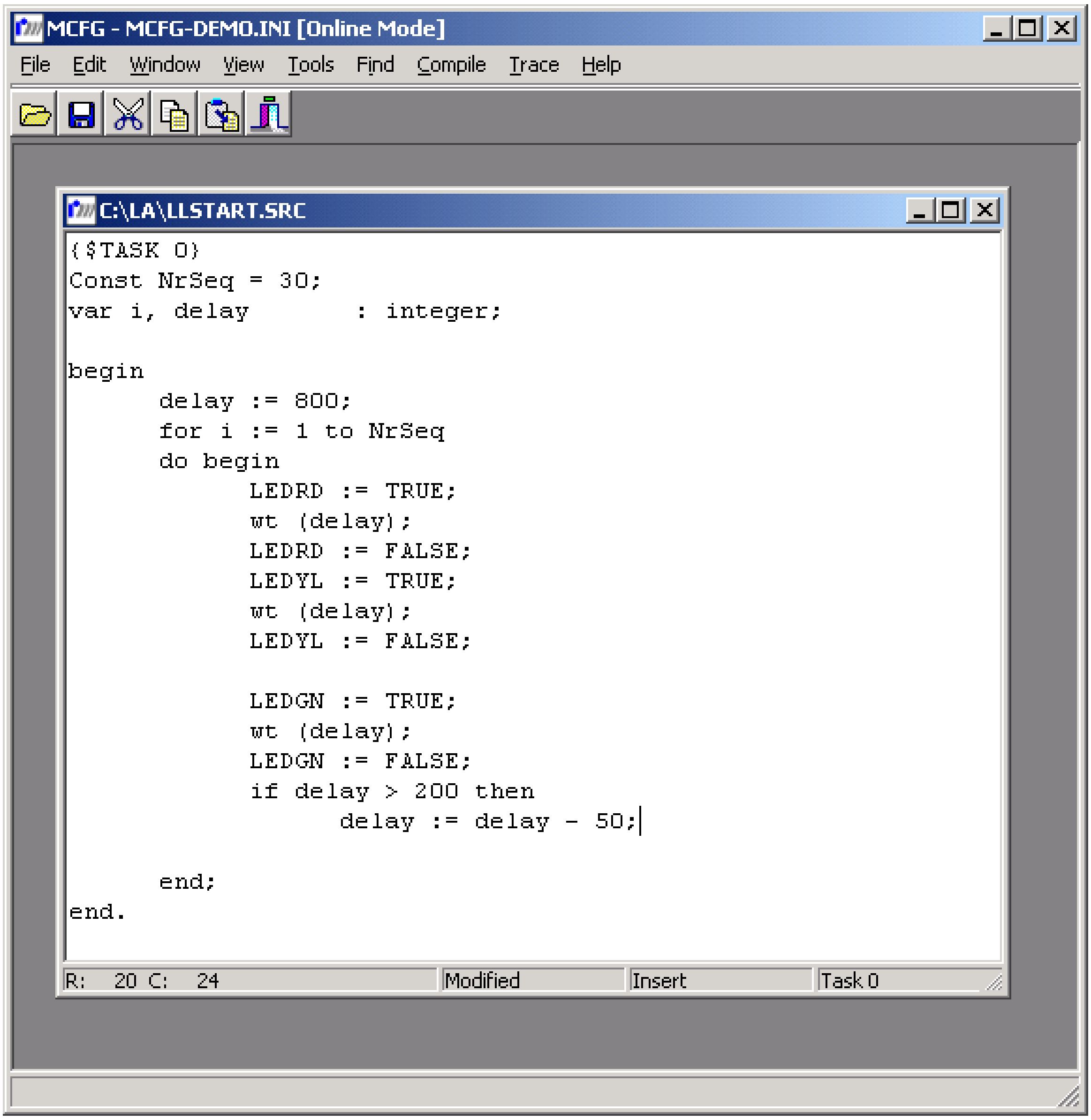
1.7.4.12 Show the profile of the manipulated variable output

The “Motor Command Outlet” option can be selected in the Mode selection menu. After a screen update, the setpoint profile and the profile of the manipulated variable output in V can be displayed above the positioning movement recorded in the Graphic window.

1.7.5 The integrated text editor

The text editor integrated in the *mcfg.exe* development environment was specially designed for creating and processing rw_SymPas source texts. However, it is also possible to create these source text files using another text editor.

Figure 1-16: Text Editor



```
{$TASK 0}
Const NrSeq = 30;
var i, delay       : integer;

begin
      delay := 800;
      for i := 1 to NrSeq
      do begin
            LEDRD := TRUE;
            wt (delay);
            LEDRD := FALSE;
            LEDYL := TRUE;
            wt (delay);
            LEDYL := FALSE;

            LEDGN := TRUE;
            wt (delay);
            LEDGN := FALSE;
            if delay > 200 then
                  delay := delay - 50;

      end;
end.
```

Various actions can be triggered from the editor in the command bar. This primarily includes loading and saving *rw_SymPas* program files and compiling and executing the corresponding files.

1.7.5.1 The header of the editor window

Since it is possible to edit multiple text files simultaneously, the appropriate file names are displayed in the header of the relevant editor window.

1.7.5.2 The status line

This is the bottom line of the editor [Figure 1-16].

Abbreviation	Description
R	(Row) The number of the line (row) in which the cursor is currently located is shown here. The first line of a text is number 1. The numbering relates to the start of the text and not to the position of the cursor within the window.
C	(Column) Shows the current column position of the cursor and also refers to the source text.
Modified	When a change has been made in the active editor window, this is signalled with this message text.
Insert (Overwrite)	Means that “Insert mode” is currently activated. Newly entered characters are inserted in the existing text at the current cursor position, any text to the right of this is moved accordingly. The [Einf] (Insert) button can be used to switch between Insert and Overwrite. In the latter case, old text after the cursor position is replaced by the new text, i.e. overwritten.

1.7.5.3 Editor commands

The commands for editing text such as the basic cursor movements, highlighting text, copying and deleting individual characters, words or highlighted blocks are largely compatible with currently available Windows editors and are therefore not explained here in more detail.
Common mouse and keyboard operations as well as drag-drop operations are supported.

1.7.5.4 Special functions of the CNC-Edit editor

When the editor environment is active, the menu bar is extended by special functions which are only available in editor mode. These special functions are described below.

1.7.5.4.1 Compile menu

This menu calls the integrated NCC compiler and compiles the content of the currently selected editor. If the compiler finds a source text line with errors, an error message is output with the error type and error line. After the user has acknowledged the error, the editor cursor is placed on the line with the error.

1.7.5.4.1.1 Syntax SAP program check [Syntax Check]

Using the [Syntax Check] option, only a syntax check of the source text program is carried out.

1.7.5.4.1.2 Syntax check and creation of a CNC file [File]

The [File] option is identical to the [Syntax Check] but, in the case of a compilation run with errors, also creates a file with the current source text file name and extension *.cnc*. This file can, for example, be transferred by the PC application program to the CNC task using the load command *txbf()* (PM / section 4.4.129) and run from there in stand-alone mode. If the SAP program does not contain a *$TASK* instruction, the CNC file is created for the task number currently selected [Compile][Select Task].
A CNC file can also be created using the Windows console program *ncc.exe* (OM / section 4.3).

1.7.5.4.2 Run menu

This menu is used to control the program run of one or more CNC tasks.

1.7.5.4.2.1 Start a program run control for a CNC task [Trace current selected CNC-Task]

The [Trace current selected CNC-Task] function is identical to the [Compile][File] function described in the previous section, but also loads the CNC file created using the PCAP command *txbf()* (PM / section 4.4.129) onto the control module and enables the CNC task to automatically run the CNC program currently included in the editor using the PCAP command *startcnct()* (PM / section 4.4.123). If the load process is run successfully, the source text line currently being edited by the CMC task is highlighted in the editor window and thus shows the processing sequence of the CNC program file. The source text line highlighted is shown in the active section of the editor window and, in addition, the corresponding source text line number is displayed in the status line. The trace process can be interrupted at any time by pressing [ESC].
If there is no *$TASK in the* SAP program, the CNC file is created for the task number currently selected [F3] and processed in the corresponding task.

1.7.5.4.2.2 Stop CNC task [Stop current selected CNC-Task]

The [Stop current selected CNC-Task] option has the effect that the CNC task selected [F3] no longer processes the CNC file currently loaded. In this context, note that approved EVENT handling procedures are also no longer processed.

1.7.5.4.2.3 Continue CNC task [Continue Trace in current CNC-Task]

With the [Continue Trace in current CNC-Task] option, the program currently loaded in the CNC task [F3] is continued. However, a check is carried out beforehand to determine whether the editor source text program is consistent with the CNC file loaded. If this is the case, the current source text line is highlighted in turn and displayed in the editor’s active screen.

1.7.5.4.3 Restart all CNC-Tasks

This menu function starts all CNC tasks (0 to 3). All CNC programs loaded are started from the beginning of the program.

1.7.5.4.4 Stop all CNC-Tasks

All CNC tasks are stopped and the CNC programs stored there are stopped as well.

1.7.5.4.5 Continue all CNC programs [Continue all CNC-Tasks]

All stopped CNC programs are continued with this function.

1.7.5.4.6 Spooler menu

With the spooler options, commands currently being spooled can be started [Start ...], stopped [Stop ...] and deleted [Delete ...]. The commands are triggered synchronously on all axes in which spooled commands exist.

1.7.5.4.7 Setup menu

1.7.5.4.7.1 Set trajectory parameter [Set CNC-specific parameter]

The CNC-specific program data such as trajectory acceleration and trajectory velocity can be entered in this menu. However, this parameter can also be programmed using predefined *System-Parameter* in the SAP user program.
This menu is available as part of the configuration parameter menu.

1.7.5.4.7.2 Set Compiler Mode (in preparation)

In this menu, you can choose between the two programming languages *rw_SymPas* or G-code programming (based on DIN 66025 or RS-274). The NCC compiler runs the syntax check for the selected programming language.

1.7.5.4.7.3 Select CNC Task

In this menu, a task number (0 to 3) which is required for task control and CNC file creation can be selected.

1.7.5.4.8 Display menu

CNC task-specific information is displayed on the screen in this menu.

1.7.5.4.9 System menu

The closing [Close ...] or opening [Open ...] of the position control loops of all axes can be enabled in this menu. Also, it is possible to reset all axis channels using the [Reset ...] instruction. Internally, clicking on [Reset ...] stops any CNC program which may be running using the PCAP command *stopcnct()* (PM / section 4.4.125). The control module is then reset using the PCAP *rs()* (PM / section 4.4.112).

1.7.5.5 Program execution in single-step operating mode

In the integrated editor it is possible to run and test SAP programs in single-step operating mode. Single-step operating mode can be accessed by activating the Single-Step-Mode option in the menu Trace – Trace Options. If program execution is now started via the menu or a shortcut, the system will be in single-step mode. Alternatively, the instruction

```
startcnct (TaskNumber);
```

can be added at any point in the program. In this case, the following line is still run and the system remains in single-step mode.
The program can then be executed line by line via the menu or using the F10 shortcut. With the F8 shortcut, the program will continue in normal mode at the relevant position. Here, it should be noted that the program code of included files is treated as an individual line.

1.7.6 Dialog functions [Dialog Functions Menu]

This menu can be used to query various states of the control module, set outputs, reset axis channels and restart movement profiles.

1.7.6.1 Display axis status [Open Axis Status Window]

This menu item opens an axis-specific status window for the currently selected axis channel [F2]. The window can be placed anywhere on the screen and can be minimised or enlarged. Multiple status windows for various axes can also be displayed on the screen at the same time.

Layout of the axis status window [ACTUAL VALUES]

Name	Meaning
Axis #	Number of the selected axis channel.
Name	Symbolic axis name {sn} [section 1.7.2.1.1].
{DP}	Desired Position. The display takes into account the selected unit and precision [section 1.7.2.1.4 and 1.7.2.1.5].
{RP}	Real Position. As for {DP}.
{AXST}	In this field, the bit-coded axst status register is displayed in hexadecimal format (PM / section 4.4.44 – rdaxst()
{LSM}	Current freely available spool area in bytes.

1.7.6.2 Close axis status display [Close Axis Status Window]

This menu item closes the status window for the currently selected axis channel [F2].

1.7.6.3 Axis status report [Display Axis Status Report]

This menu item enables certain status information on all the axis channels in the system to be displayed. The user is given a quick overview of the status of the entire drive system. The display uses 40-column mode so that it can also be read from greater distances.

Layout of the axis status report [AXIS STATUS REPORT]

Name	Meaning
Axis name	Symbolic axis name {sn} [section 1.7.2.1.1].
Actual position	Actual position {rp}. The display takes into account the selected unit and precision [section 1.7.2.1.4 and 1.7.2.1.5].
Flag L	A hardware or software limit switch was detected.
Flag M	The maximum permitted position error was exceeded.
Flag C	Axis is currently in position control.
Flag P	The end of the profile has been reached.

1.7.6.4 Display bit information of the axis status register *axst* [Display Detailed Axis Status]

This function is used to display the current status of the axis-specific [F2] *axst* register. The display results from the bit information stored in the *axst* register. The first column (S column) shows the current status of the relevant bit. Each item of bit information is described in more detail using its bit number in the *axst* word, its symbolic name and its functions. The last column (Error/Status) shows whether an error or a status flag applies for each bit.

1.7.6.5 Show digital inputs and status information [Show Inputs / Status]

This function is used to display the current status of the digital inputs and various status information of the selected axis channel [F2].

Current input and status information of the control module [INPUTS / STATUS]

Name	Meaning
{digi}	Status of digital inputs (PM / section 4.4.52.1)
{digi}	NDX field: If a zero-trace (index) incremental encoder is used, its status is also shown here (PM / section 4.4.52.1)
{digi}	EE field: If there is an error in the measuring value recording system, this is shown here (PM / section 4.4.52.1)
{digi}	NDXL field: The edge transition from NDX is shown here (PM / section 4.4.52.1)
{digi}	STRBL field: The edge transition from NDX is shown here (PM / section 4.4.52.1)
{epc}	Number of EEPROM programming cycles (PM / section 4.4.62 – rdepc())
{edv}	Shows the validity of the EEPROM data (PM / section 4.4.70 – rdifs())
{pfe}	If the operating voltage has fallen below 4.75 V, this is shown here. Every time the PC is switched on again this flag is set to “1” (PM / section 4.4.70 – rdifs())
{wdog}	Shows that the module has been reset following a watchdog error (PM / section 4.4.70 – rdifs())
{iae}	Invalid Access Error. This flag shows an internal non-permitted access error (PM / section 4.4.70 – rdifs())

1.7.6.6 Show CNC task status and common variables [Show CNC-Task Status / Variables]

Status information on the CNC task [CNC-TASK-STATUS / COMMON VARIABLES]

Name	Meaning
Error-Nr	If an error is caused by an SAP program, a system-internal error number is shown in this field. In this case, the SAP program is stopped.
Error-Line	If the Error-Nr [Error No.] described above is not equal to 0, the line number of the SAP program in which the error occurred is also shown in this field.
Current Program	If an SAP program was transferred to the control module, the corresponding file name is shown here.
Running / Not Running	If an SAP program is running, the display switches to “Running”.
Running in task #	This is the CNC task number in which the current SAP program is being processed. ([F3] = task number selection)
Stack	Shows the stack area (bytes) of the CNC task currently freely available.
Current source line #	Shows the source text line number currently being run.
Common Variables CI0..CI99 und CD0 .. CD99	The current status of all freely available common integers and common double variables is shown here. (PM / Sections 4.4.48 – rdcd(), 4.4.49 – rdci() and 6.3.1)

1.7.6.7 Edit digital outputs [Edit Outputs]

The digital outputs can be set and/or reset in this menu (PM / section 4.4.137 – wrdigo()). The outputs are set and/or reset by clicking with the mouse or by pressing the “Blank” key.

Note: The selected axis channel [F2] is of no importance when setting/resetting the outputs, i.e. the fact that all outputs of the module can be addressed via the various axis channels.

1.7.6.8 System Reset

This menu can be used to perform the axis-specific reset [F2] and the complete system reset.
How to use *Reset selected axis* is described in the PM / section 4.4.42 – ra().
The functionality of *Reset whole system* is described in the PM / section 4.4.112 – rs().

1.7.7 Automatic functions [Automatic Functions Menu]

This menu can be used to start, stop and continue CNC programs. CNC programs concern AutoCode files which are automatically generated by compiling SAP source text programs in the CNC editor environment [section 1.7.5] or using the command line compiler *ncc.exe*.
The CNC programs are created for a specific CNC task by compiling the SAP source text file with the aid of a compiler control command ($TASK) or by selecting a CNC task number (function key [F3] in the *mcfg.exe* program). Up to 4 different CNC programs can be processed at the same time. The CNC tasks 0 to 3 are available for this.
The functions described below are also accessible via the special functions of the editor CNC-Edit [section 1.7.5.4].

1.7.7.1 Download CNC Program

This menu can be used first of all to select a CNC file. It is selected using the [+] key or the Enter key. To activate the selected program, the selection must be completed with the *Download-File* command. The file extension *.cnc* is always suggested for the file selection.
The *Download* command transfers the selected CNC file to the control module, which, among other things, also contains the information on the CNC task to which this AutoCode file needs to be transferred. The corresponding task is stopped beforehand. This state remains in effect even after the transfer.

1.7.7.2 Restart current selected CNC Task

This menu function starts the currently selected CNC task. This can be selected with the [F3] function key.
The CNC program loaded there is started from the beginning of the program.

1.7.7.3 Stop CNC task [Stop current selected CNC-Task]

The currently selected CNC task [F3] is stopped and, thus, the CNC program saved there is also stopped.

1.7.7.4 Continue CNC task [Continue current selected CNC-Task]

All stopped CNC programs are continued with this function.

1.7.7.5 Restart all CNC-Tasks

This menu function starts all CNC tasks (0 to 3). All loaded CNC programs are started from the beginning of the program.

1.7.7.6 Stop all CNC-Tasks

All CNC tasks are stopped and the CNC programs stored there are stopped as well.

1.7.7.7 Continue all CNC-Tasks

All stopped CNC programs are continued with this function.

1.7.7.8 System Reset

This menu is identical to the menu described in the section 1.7.6.8.