SM148E

PIMikroMove

Software Manual

Version: 2.11.0 Date: 02.05.2012

This document describes the following product:

- **PIMikroMove**

Physik Instrumente (PI) GmbH & Co. KG · Auf der Römerstr. 1 76228 Karlsruhe, Germany
Telefon +49 721 4846-0 · Telefax +49 721 4846-1019 · E-Mail info@pi.ws

Original instructions
First printing: 02.05.2012
Document number: SM148E, BRo, Version 2.11.0

Contents

1 Introduction

PIMikroMove is a general purpose graphical user interface for PI motion controllers. You can connect as many controllers or controller networks as you have connected to your PC. Some of the available menus and their items depend on the connected controller type so that not all of the menus, items and windows described in this manual will always be available in your current instance of PIMikroMove.

1.1 About This Document

1.1.1 Goal and Target Audience of this Manual

This manual assumes that the reader has a fundamental understanding of basic servo systems as well as motion control concepts and applicable safety procedures.

For updated releases of this manual, or if you have any questions, contact our customer service department (mailto:info@pi.ws).

1.1.2 Other Applicable Documents

The devices which are mentioned in this documentation are described in their own manuals.

For the latest versions of the user manuals contact our customer service department (mailto:info@pi.ws).

1.1.3 Symbols and Typographic Conventions

The following symbols and typographic conventions are used in this manual:

NOTICE

Dangerous situation

If not avoided, the dangerous situation will result in damage to the equipment.

➢ Actions to take to avoid the situation.

INFORMATION

Information for easier handling, tricks, tips, etc.

Symbol	Meaning
SV0?	Command line or command from PI's General Command Set (GCS) (example: command to get the servo mode)
Start up Controller	Window or dialog name in the software (example: dialog for controller connection and startup of the connected stages)
Start > Settings	Menu path in the software (example: to open the menu, the ***Start*** and ***Settings*** buttons must be clicked in succession)
5	Value that must be entered or selected via the software

1.2 Initial Installation

INFORMATION

Additional components are required when installing PIMikroMove. PIMikroMove supports a broad range of controllers from PI. To handle the necessary communication with a controller, the program relies on the GCS DLL for the specific controller, and, for some controllers, also on the PIStages2.dat database. For every new controller you want to connect to PIMikroMove, you have to ensure its GCS DLL is installed (and database, if required) by running the common setup procedure. If you simply start PIMikroMove without the necessary DLL(s) and database you will get an information message "No GCS DLLs found. PIMikroMove needs these libraries to connect to the PI controllers".

PIMikroMove is provided on the product CD of supported controllers and can be installed with a common setup procedure:

1. Insert the controller product CD in the host PC.
2. If the Setup Wizard does not start automatically, start it from the root directory of the CD by doubleclicking the  icon or the ***setup.exe*** file.
3. Follow the on-screen instructions. You can choose between "typical" and "custom" installation. Typical components are, for example, PIMikroMove, LabView driver and DLLs. "Typical" is recommended.

1.3 Installing Updates

INFORMATION

All customized settings (e.g. settings for user stages or position settings) are kept when PIMikroMove is updated.

INFORMATION

It is strongly recommended to always use the latest versions of PIMikroMove, GCS DLL and PIStages2.dat stage database. The software is maintained continually so that the version on the product CD used for initial installation may be out of date.

➢ Use the PI Update Finder to find and download updates for the software.

➢ Always install the latest version of the software.

Prerequisites

- ✓ If the PI Update Finder is not provided by the product CD: You have downloaded the PI Update Finder (http://www.update.pi-portal.ws) (PIMikroMove provides the ***Help > Download the PI Update Finder*** menu sequence) and unpacked the Zip file into a directory on the host PC.
- ✓ You have the Technical Note "PI Update Finder" (A000T0028) readily available. This document is provided by the product CD or by the Zip file for the PI Update Finder.
- ✓ The PC is connected to the Internet.
  - If the PC is **not** connected to the Internet:
    You have the Technical Note "Updating PI Software" (A000T0032) readily available. This document is provided by the product CD or by the Zip file for the PI Update Finder.

Updating software

1. Download the update (Zip file) using the PI Update Finder:
   - Follow the instructions in the Technical Note for the PI Update Finder (A000T0028).
   - If the PC is **not** connected to the Internet: Follow also the instructions in the Technical Note "Updating PI Software" (A000T0032).
2. Install the update:
   - Follow the instructions in the documentation (Releasenews file or readme file) which accompanies the update Zip file.
   - To update PIMikroMove, simply replace PIMikroMove.exe and PIMikroMove_SM148E.zip in the directory where they where installed with the initial setup procedure. In PIMikroMove, you can use the ***Show version information*** item in the ***Help*** menu or the ***Search for controller software*** item in the ***Connections*** menu to identify the installation path.

1.4 Getting Started

Set up and installation of the motion system is described in the User Manuals of the controller(s) and mechanics.

To start PIMikroMove use the appropriate item on the ***Start*** menu (by default under ***Start > Programs > PI***).

When PIMikroMove starts, the ***Start up Controller*** (p. 171) window is displayed at the first step, ***Connect Controller***. Here is where you connect controllers to PIMikroMove.

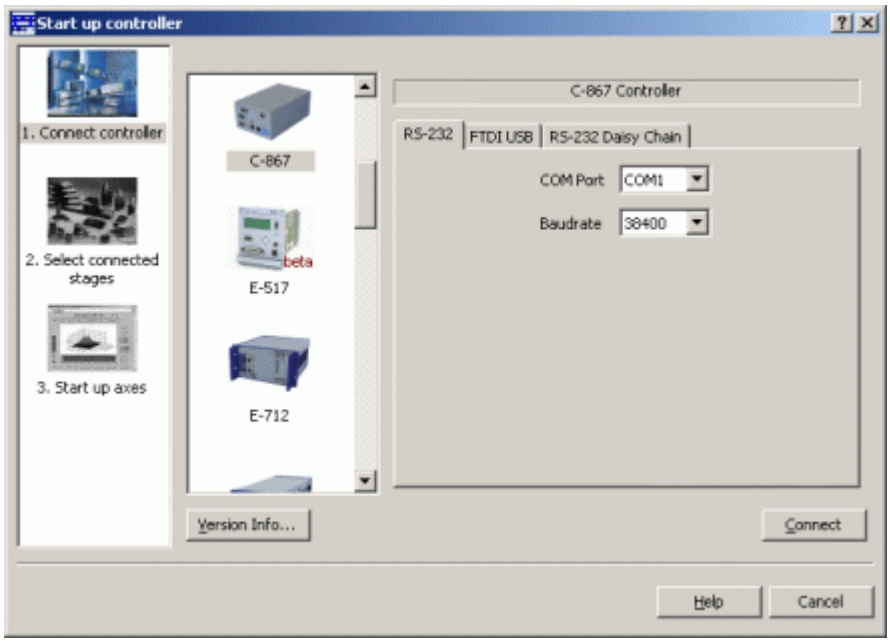


Figure 1: Start up Controller window with the Connect Controller step

Only one controller may be selected at a time, but the dialog can be reopened to connect further controllers to the current session. In the main window of PIMikroMove, use ***New*** connection on the toolbar or the ***Connections > New...*** menu sequence to do this. See "Connect Controller" (p. 171) for more information, especially regarding daisy chain connections.

INFORMATION

If the controller is connected via a USB port, this USB interface might appear as an additional COM port in the port-selection list.

After the connection was established, PIMikroMove guides you through the further steps of the startup procedure. It depends on the controller which of the following steps are necessary:

- Controllers which support loading settings from a stage database and are not preconfigured for a certain stage type:
  Select connected stages (p. 174)
- Hexapod controllers with GCS syntax version 1.0:
  Configure Hexapod (p. 177)
- E-517 interface and display modules:
  Start up E-517 (p. 178)
- System whose axes have to be initialized (e.g. referenced) before normal use:
  Start up axes (p. 179)

The screenshots below show the steps for loading the stage settings from a database and for axis referencing.

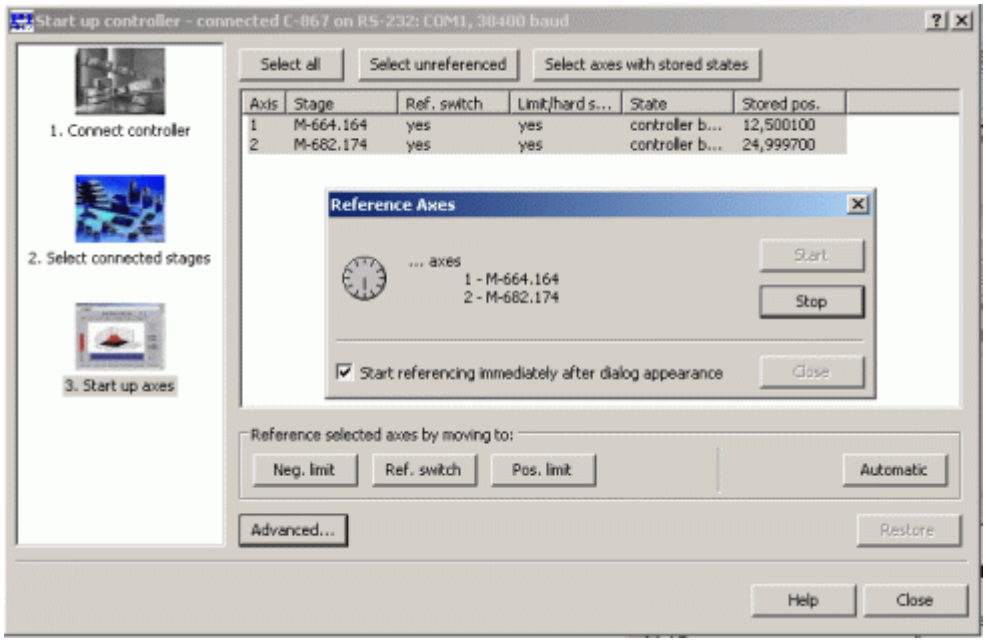


Figure 2: Startup Controller window with the Select connected stages step

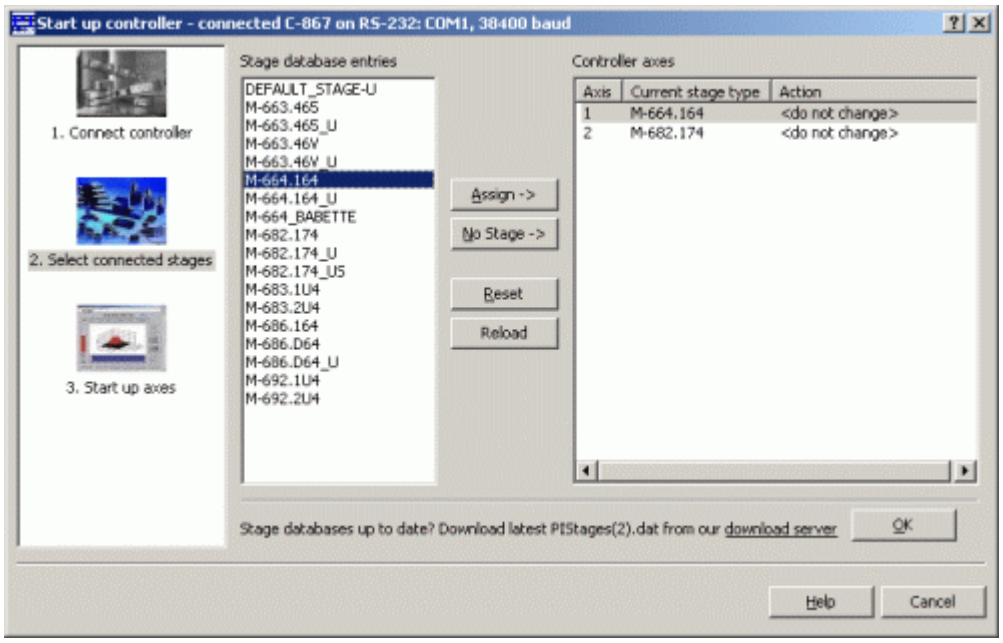


Figure 3: Start up Controller window with the Start up axes step, reference moves are in progress

When the ***Start up controller*** window is closed, the PIMikroMove main window will open. Furthermore, ***Single-axis*** windows may open for the available axes. You may need to resize or move them to see the main window.

2 Main Window

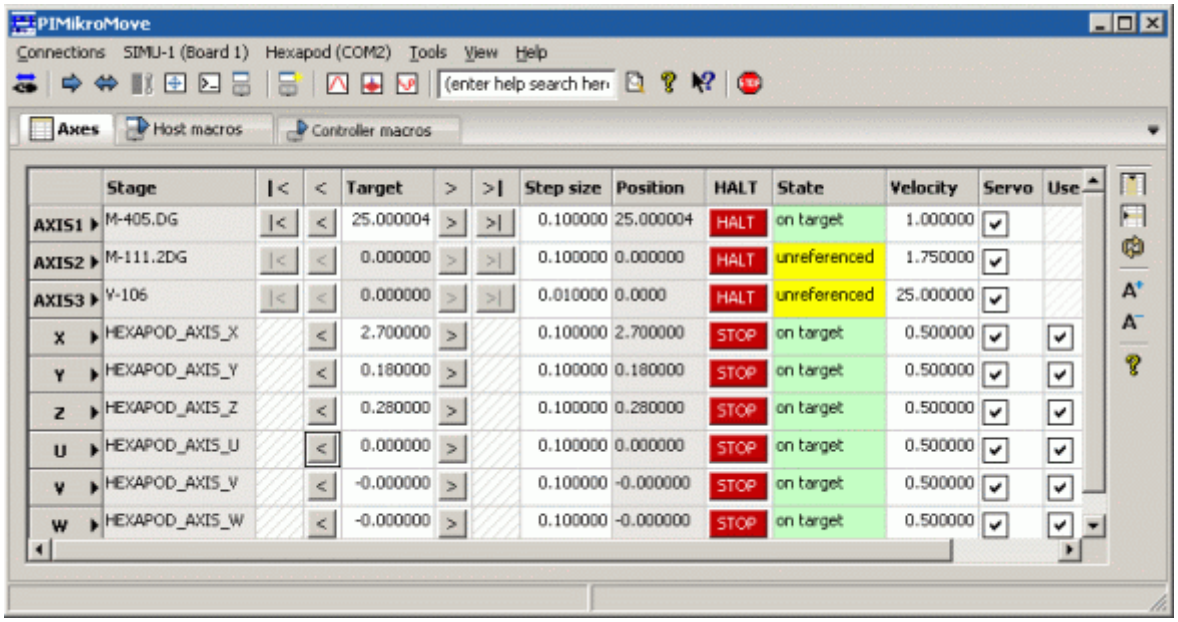


Figure 4: Main window showing the Axes tab card

The main window consists of several parts:

- Menu bar, see "Menu Bar" (p. 12)
- Toolbar, see "Toolbar" (p. 25)
- Pane with tab cards for:
  - Tables for axes, piezo channels, sensor channels and PiezoWalk® channels, see "Table Tab Cards" (p. 28)
  - 3D visualization of the Hexapod for every Hexapod system which is connected to PIMikroMove, see "Hexapod 3D view" (p. 36)
  - Host macros, see "Host Macros" (p. 39)
  - Controller macros (if supported by one or more connected controllers), see "Controller Macros" (p. 64)

The tab cards can be flexibly arranged in the main window by dragging them with the mouse. If a new controller is connected which provides sensor and piezo channels, the corresponding tab cards will automatically be arranged in the main window so that all axes and channels are visible. Exception: If the tab cards were arranged before, PIMikroMove will not change the appearance of the main window.

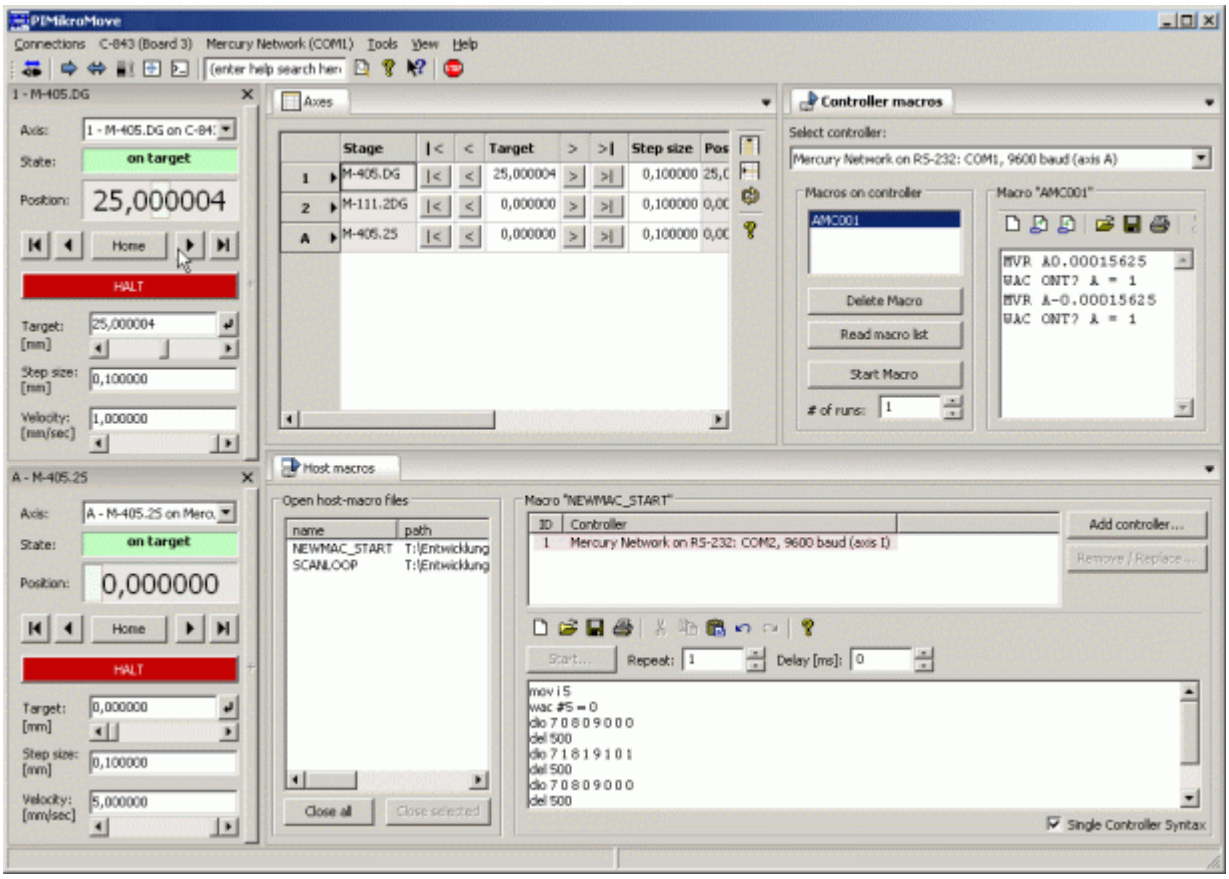


Figure 5: Two Single-axis windows and the Axes, Controller macros and Host macros tab cards, all docked inside the main window

Some other windows can be shown, either as separate windows or docked into the main window. See "Additional Common Windows" (p. 69) and "Controller Specific Windows" (p. 101) for more information.

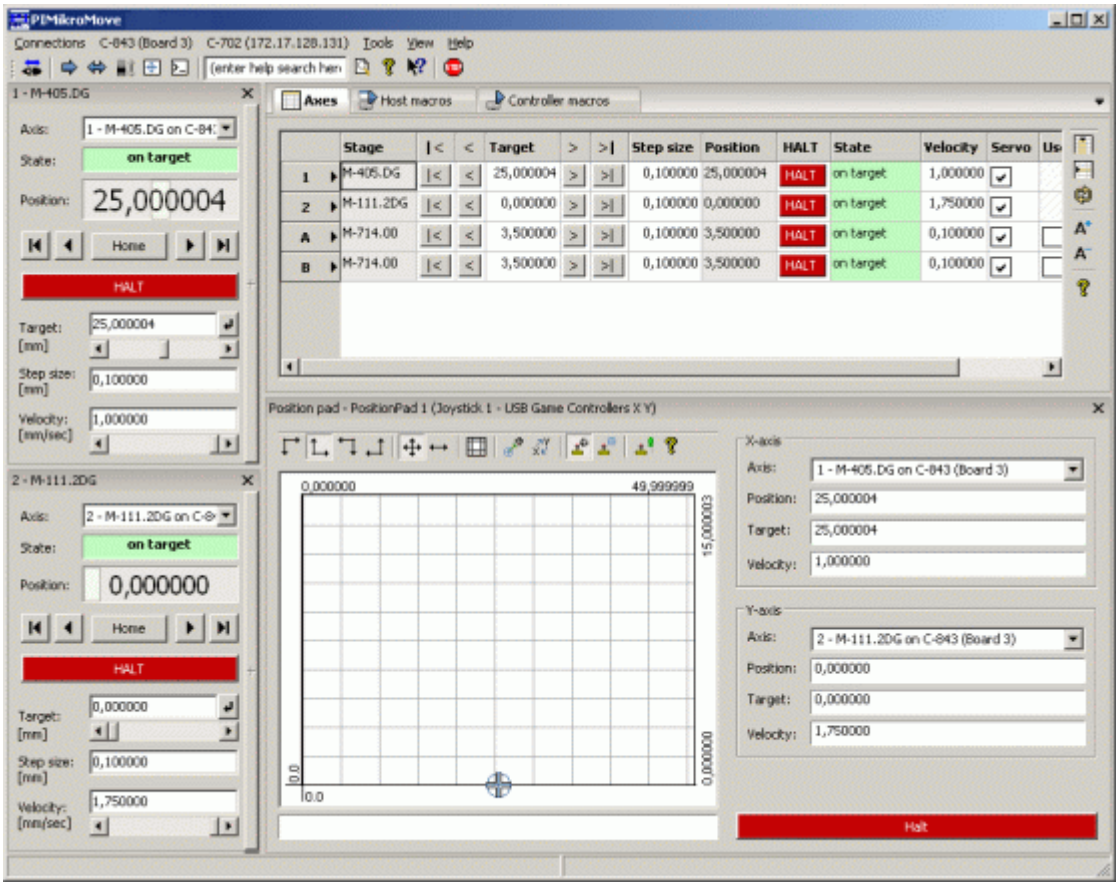


Figure 6: Two Single-axis windows and the Position pad, all docked inside the main window with the Axes tab card displayed

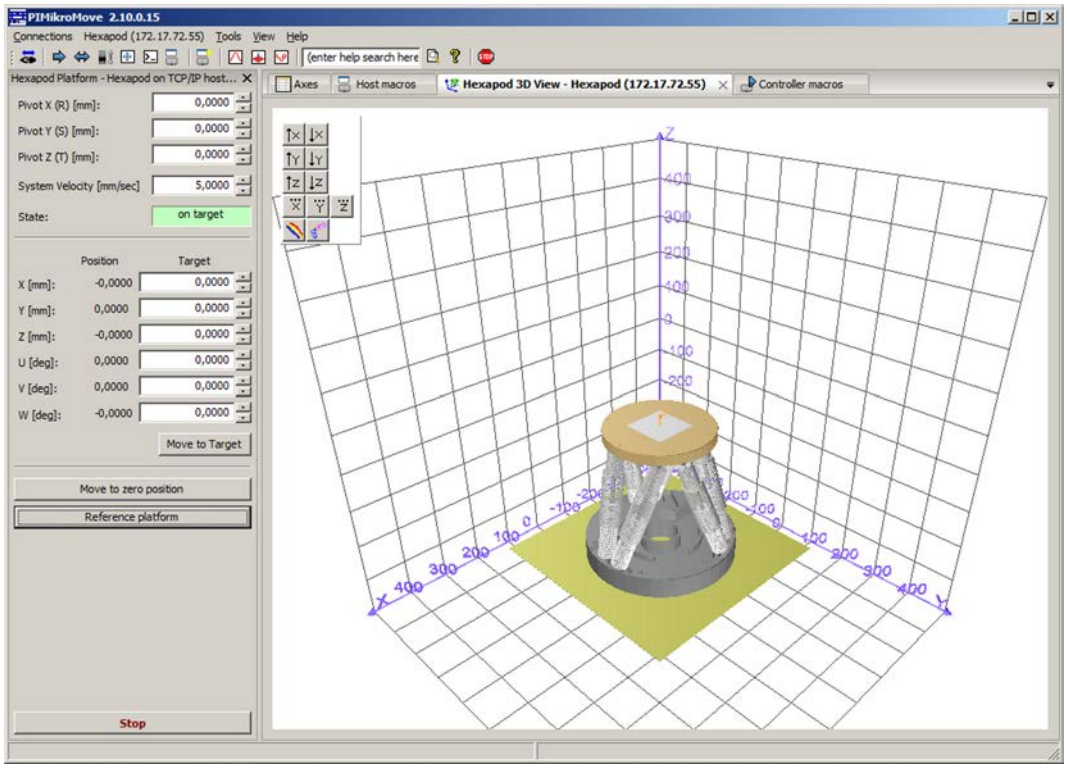


Figure 7: Main window showing the Hexapod 3D View tab card and the Hexapod Platform window (docked)

2.1 Menu Bar

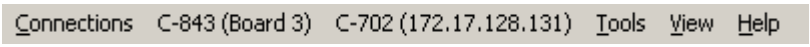


Figure 8: Menu bar

Note that the items of the menu bar are not accessible when the ***Command entry*** window (p. 75) is open.

2.1.1 Connections Menu

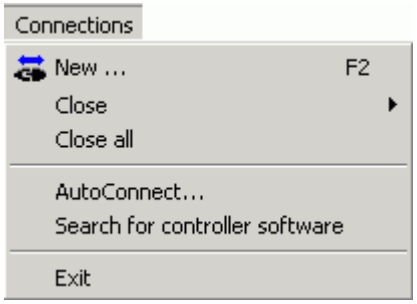


Figure 9: Connections menu

New…

Opens the ***Start up Controller*** window (p. 171) to connect another controller or controller network. Shortcut: F2

Close

Contains a submenu with entries for each connected controller/network. By selecting one of the menu entries, the connection to that controller can be closed. Note that the associated controller menu-bar item will disappear.

Close all

Closes connections to all connected controllers. All controller menu-bar items will disappear.

AutoConnect...

Opens the ***Auto Connect Configuration*** window (p. 183) where you can change the current AutoConnect settings.

Search for controller software

Re-scan the local computer for PI software. This can be done after you have installed new controller software while PIMikroMove is running.

Exit

Close all connections and exit PIMikroMove.

2.1.2 Controller Menu

For each connected controller or network of controllers there will be one controller entry in the menu bar. The text will indicate the class of controller and the interface used for the connection. The items on the controller menus will depend on the controller class—not all items are always available.

INFORMATION

Some networked controllers (e.g. daisy chained Mercury class controllers with native firmware or E-816s on an I2C bus) appear in PIMikroMove as a single multi-axis controller, while others (e.g. daisy chained E-755s) will have individual entries.

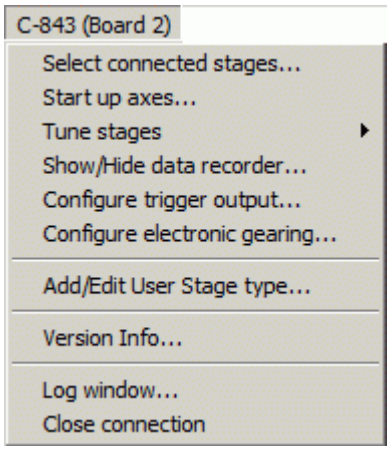


Figure 10: Example: controller menu for a C-843 controller

The items on the various controller menus are explained below.

Select connected stages…

Opens the ***Start up Controller*** window at the ***Select connected stages*** step. Here you can tell the controller and the software what stages are connected. With most controllers, one entry is required for each motion axip. See "Select Connected Stages" (p. 174) for more information.

For a Hexapod controller with GCS syntax version 1.0, ***Select connected stages*** opens the ***Start up Controller*** window at a step renamed ***Configure hexapod***. There, you can indicate whether separate axes (e.g. A and B) are connected, and also which analog input channels are active. All of these options require appropriate previous configuration of the Hexapod controller; see the controller User Manual and "Configure Hexapod" (p. 177) for details.

Start up axes…

Opens the ***Start up Controller*** window at the ***Start up axes*** step. See "Start Up Axes" (p. 179) for more information.

Tune stages

Opens the ***PITuningTool*** for the selected axis. With this tool you can easily determine well-suited parameters for P-I-D servo-controllers. See "PI Tuning Tool" (p. 105) for more information.

Profile Generator

Present if supported by the controller and required DLL found. Opens a series of dialogs for defining and executing certain types of user-defined motion profiles on two axes. Because the profiles are stored and executed on the controller side of the interface, the time resolution is much finer than when setting targets one at a time by command or host macro. See "User Profile Mode" (p. 101) for details.

Show/Hide data recorder...

Opens the ***Data Recorder*** window where you can analyze the recorded data and configure recording and display. See "Data Recorder" (p. 129) for more information.

Show Embedded Scan Window

Present if supported by the controller and if the controller has at least one analog input channel. Opens the ***Embedded Scan & Align Algorithms*** window for the controller or the window of the selected scanning procedure. See "Embedded Scan & Align" (p. 160) for more information.

Show wave table editor…

Opens the ***Wave Table Editor*** window where you can define wave table points for later output, check them in a graphics pane and save them to the controller or to data files on the host PC. See "Wave Table Editor Window" (p. 123) for more information.

Start wave table output…

Opens the ***Start wave table*** window where you can start wave table output and select the start mode (immediately or by external trigger pulses). See "Start Wave Table Output" (p. 126) for more information.

Stop wave table output…

Opens a window where you can stop the wave table output for selected axes. See "Stop Wave Table Output" (p. 127) for more information.

Enable triggered move...

Opens the ***Enable Triggered Move*** window where you can specify if axis motion is to be controlled by external trigger pulses. If triggered motion is enabled, each pulse causes a relative step of a predefined size, and other control sources may not be accepted. See "Enable Triggered Move" (p. 127) for more information.

Show Hexapod Platform Settings

Opens the ***Hexapod Platform*** window where you can set the pivot point coordinates, enter target positions and start or stop motions for the moving platform of the Hexapod. See "Hexapod Platform Settings" (p. 157) for more information.

Hexapod 3D View

Hexapod 3D View > Show displays the ***Hexapod 3D View*** tab card with a visualization of the Hexapod's current position.

Hexapod 3D View > Change visualization… is available when the ***Hexapod 3D View*** tab card is displayed in the main window. The menu sequence opens the ***Hexapod 3D View – Visualization*** window where you can load a DAT file when PIMikroMove cannot find suitable data for the visualization of the connected Hexapod (e.g. when a customized Hexapod is used and you have received a special DAT file from PI).

See "Hexapod 3D View" (p. 36) for more information.

Show service tools…

Opens the ***Hexapod Service Tools*** window where you can perform tests of the Hexapod struts. You should perform Hexapod strut tests only for diagnosis if failure of the Hexapod system occurs. See "Hexapod Service Tools" (p. 163) for more information.

Show/Hide wave generator...

Opens the ***PI Wave Generator Tool*** window where you can create waveforms, configure the data recorder, administer the Dynamic Digital Linearization (DDL) feature and start the wave generator(s). See "PI Wave Generator Tool" (p. 111) for more information.

Configure trigger output...

Opens the ***Configure Trigger Output*** window where you can define trigger conditions and enable them for the digital output lines of the controller. See "Configure Trigger Output" (p. 139) for more information.

Calibrate controller joystick...

Opens the ***Joystick Calibration*** window for the selected joystick device. In this window, you can select a predefined lookup table or create a custom lookup table for the individual joystick axes. See "Calibrate Controller Joystick" (p. 141) for more information.

Configure controller joystick(s)...

Opens the ***Configure Controller Joystick*** window where you can enable joystick control for the individual controller axes. See "Configure Controller Joystick(s)" (p. 144) for more information.

INFORMATION

Do not enable axes with no physical joystick connected, as uncontrolled motion could occur.

Make sure that the joystick is properly calibrated to avoid uncontrolled motion.

INFORMATION

Only joysticks directly connected to the controller(s) are affected here. If you have a joystick connected to the host PC see "Position Pad Window" for how to use it.

Add/Edit User Stage Type...

Here, the current settings of a stage can be stored as a new stage type in the user-stages dat file on the host PC. If the stage type already exists in the user stage database, its settings will be overwritten when the new ones are saved.

For the stage type entry, do not use stage names which already exist in the PIStages2.dat database. If a stage of the same name exists in PIStages2.dat and the user-stages dat file, the parameter settings from PIStages2.dat will be preferred when assigning that stage to an axis (e.g. in the "Select Connected Stages" (p. 174) dialog), and the settings from the user-stages dat file will never be used.

Parameter Configuration

Opens the ***Device Parameter Configuration*** window where you can check and edit the controller parameters. The parameters are grouped under different headings in this window. See "Device Parameter Configuration" (p. 147) for more information.

Save parameters to non-volatile memory

Opens a dialog where you can enter the password for the WPA command. Entering the password and pressing OK will save the current valid values of the controller parameters to the non-volatile memory of the controller where they become the power-on defaults.

Change command level (CCL)

Opens a dialog where you can select a command level and enter the corresponding password. Changing some parameters and running some commands may require the command control level (CCL) 1. The password for command control level 1 is "advanced". Note that command levels higher than 1 are reserved for PI service personnel.

Version info…

Displays version info of the software used for the associated controller.

Configure interface...

Opens the ***Configure Interface*** dialog where you can modify the current and the default communication settings for the controller. See "Configure Interface" (p. 156) for details.

Log window...

Opens a controller-specific log window. In this window, you can monitor the commands which are sent to the controller when you use the controls of ***PIMikroMove***. See "Log Window" (p. 169) for details.

Close connection

Closes the connection to the associated controller or network. The associated controller menu will disappear completely from the menu bar. The connection and hence the controller menu can be restored using the ***Connect > New*** menu sequence.

2.1.3 Tools Menu

The ***Tools*** menu gives access to special windows which in principle are available for all controllers (for exceptions see below). Some of them can also be opened with a key combination or toolbar button.

Note that windows which are in any way controller-specific are accessible from the corresponding controller menu, see "Controller Specific Windows" (p. 101).

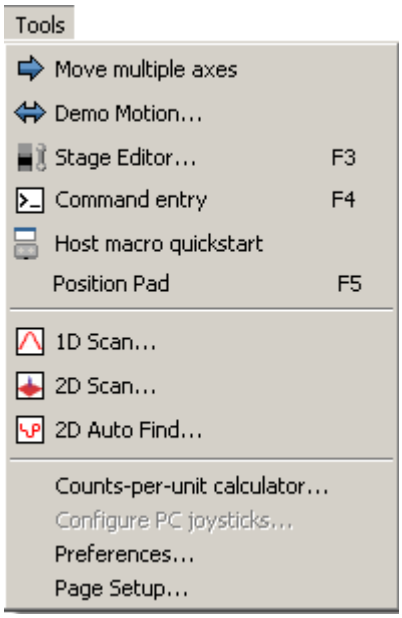


Figure 11: Tools menu

Move multiple axes...

Only available if at least one of the connected axes is referenced and has servo on. Opens dialog for initiating a synchronous move of more than one axis. See "Move Multiple Axes" (p. 69) for details.

Demo Motion…

Only available if at least one of the connected axes is referenced and has servo on. Opens the ***Demo Motion***... window: See "Demo Motion" (p. 71) for details.

Stage Editor…

Only available when the PIStageEditor.exe file is installed on the PC (depending on the controller, the PIStageEditor.exe is automatically installed with the typical setup). Starts the ***PIStageEditor.*** If multiple stage databases are available on the host PC for the connected controllers, you have to select one. Using the ***PIStageEditor*** you can inspect the parameters of all PI stages in the PIStages2.dat stage database. In user-stages dat files you can define and store special parameter sets for new or non-standard stages. See “Stage Editor” (p. 72) and the ***PIStageEditor*** Manual for details. Shortcut: F3

INFORMATION

Depending on the controller, you can also inspect parameters on the tab cards of the main window, in the ***Single-Axis*** window or using the ***Device Parameter Configuration*** window. See the User manual of the appropriate controller for more information.

Command Entry

Designed to give complete control of what is sent to the controller (via the controller GCS DLL). Here commands can be entered and responses seen. Because ***PIMikroMove*** cannot allow itself periodic status or position checks in the background, other windows are not updated while the ***Command entry*** window is open. See Section "Command Entry Window" (p. 75) for details. Shortcut: F4

Host macro quickstart

Displays the ***Host Macro Quick Start*** window where you have quick access to host macros. You can select and start a host macro without the need to switch to the ***Host macros*** tab card. Furthermore, you can assign a customized “short-cut” button to a host macro so that this macro can be started by simply clicking the button. See “Host Macro Quick Start” (p. 60) for details.

INFORMATION

The ***Host Macro Quick Start*** window can be docked to the borders of the main window by dragging it with the left mouse button pressed (see "Main Window" (p. 9)).

Position Pad

Displays the ***Position Pad*** window, in which it is possible to control two axes using the mouse and or using a joystick which is connected to the PC. See "Position Pad Window" (p. 78) for details. Shortcut: F5 (if only one ***Position Pad*** window is available)

If joysticks are connected to the PC before PIMikroMove is started, multiple ***Position Pads*** will be available. The default assignment of the joystick(s) and their axes to the Position Pads can be changed in a separate dialog. See "Configure PC Joysticks Window" (p. 91) for details.

INFORMATION

A ***Position Pad*** window can be docked to the bottom border of the main window by dragging it with the left mouse button pressed (see "Main Window" (p. 9)).

1D Scan...

Opens the ***Scan 1D*** window where you can start a single-axis scan motion during which an input source is measured. See "Scan 1D Window" (p. 83) for details. Do not confuse this PIMikroMove functionality with the embedded fast scanning offered by some controllers.

2D Scan...

Opens the ***Scan 2D*** window with the Scan tab card on top where you can start biaxial scan motion during which an input source is measured. See "Scan 2 D Window" (p. 86) and "Scan Tab Card" (p. 87) for details. Do not confuse this PIMikroMove functionality with the embedded fast scanning offered by some controllers.

2D Auto Find...

Opens the ***Scan 2D*** window with the ***Auto Find*** tab card on top where you can start a scan procedure which tries to find the maximum of an intensity signal by modifying the positions of two axes. The measured intensity signal can be, for example, a selected analog input or the position of a selected axis. See "Scan 2 D Window" (p. 86) and "Auto Find Tab Card" (p. 90) for details. Do not confuse this PIMikroMove functionality with the embedded fast scanning offered by some controllers.

Counts-per-unit calculator...

Opens the ***Counts per unit calculator*** window where you can calculate the values for numerator and denominator of the counts-per-physical-units factor (parameters 0xE and 0xF) for your stage. For stages with incremental sensors, the counts-per-physical-unit factor determines the unit of length to be used for all closed-loop motion commands (the controllers internally use counts for motion command processing). See “Counts-per-Unit Calculator Window” (p. 94) for details.

Configure PC joysticks...

Only available if at least one joystick is connected to the PC at the start of ***PIMikroMove***. Displays the ***Configure PC*** Joysticks window where you can configure the axes and buttons of the connected joysticks (including the joystick system settings) and assign the joystick(s) and their axes to ***Position Pads***. See "Configure PC Joysticks Window" (p. 91) and "Position Pad Window" (p. 78) for details.

INFORMATION

Up to two joysticks can be used. Connect them to the PC ***before*** you start ***PIMikroMove***. If the program is already running when you connect the joysticks, it must be restarted for correct joystick detection.

Preferences…

Controls the behavior of the software user interface. Some settings can not be undone the way they were made, such as turning off future appearances of confirmation dialogs. These settings can always be changed in the ***Preferences***... window.

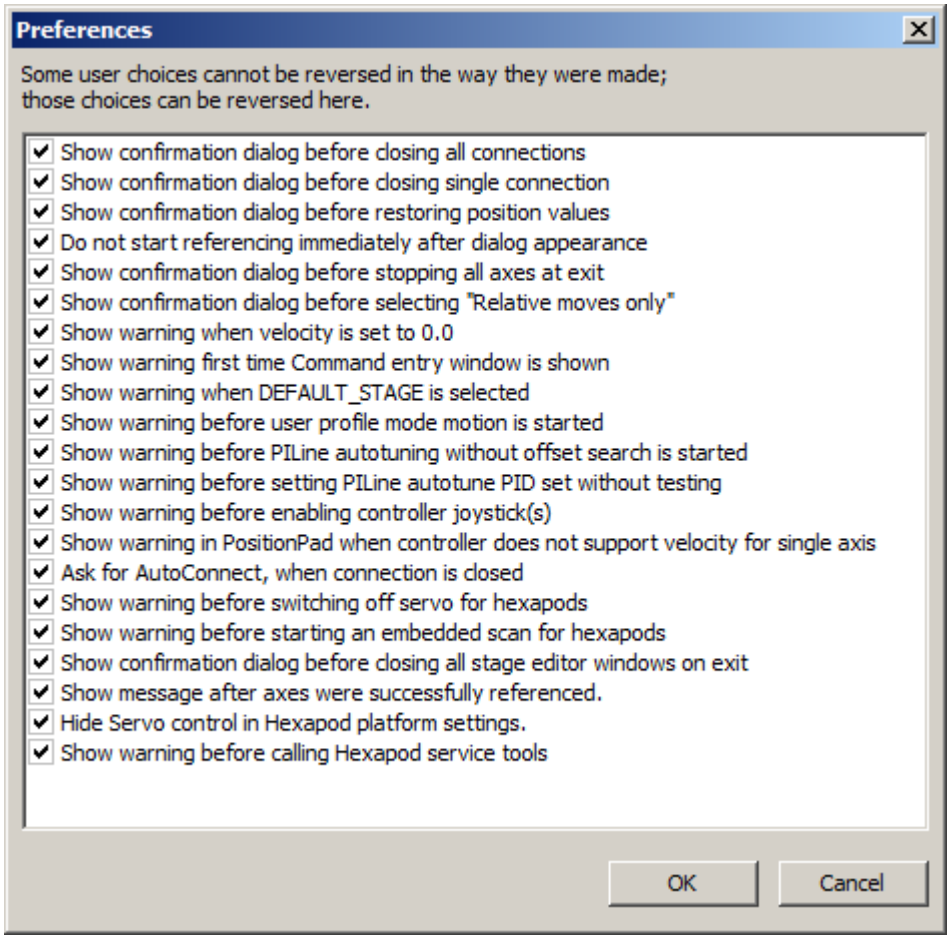


Figure 12: Preferences window

Page Setup…

Allows setting parameters which would be used for printing from the ***Command*** entry window and the ***Controller*** macros tab card.

2.1.4 View Menu

Using the items of the ***View*** menu, you can open separate windows for the individual axes, digital I/O lines and analog input lines of all connected controllers, e.g. for monitoring purposes. Depending on the controller(s) connected to PIMikroMove, not all items of the ***View*** menu are always available.

INFORMATION

Single Axis windows, ***Digital I/O*** windows and ***Analog Input*** windows can be docked to the left or right border of the main window by dragging it with the left mouse button pressed (see "Main Window" (p. 9)).

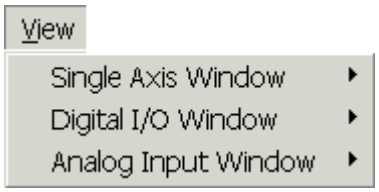


Figure 13: View menu

Single-Axis Window

Opens a window in which data about a single axis is displayed and can be modified. The controller is interrogated at regular intervals (unless ***Command entry*** is open) to update the display. See "Single Axis Window" (p. 95) for details.

Digital I/O Window

Shows a window in which the state of the digital input lines can be monitored and that of the digital output lines changed. The controller is interrogated at regular intervals (unless ***Command entry*** is open) to update the display.

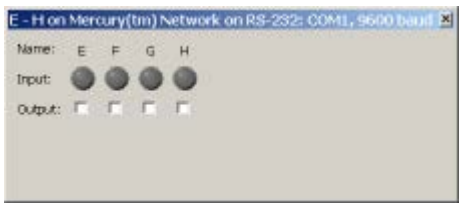


Figure 14: Digital I/O window

Analog Input Window

Shows a window in which the state of the analog input lines can be monitored. The controller is interrogated at regular intervals (unless ***Command entry*** is open) to update the display. Minimum and maximum values are represented by orange bars. If you move the cursor with the mouse into the window, the numerical values are displayed. To reset them, click somewhere inside the window with the right mouse button and select ***Reset minimum and maximum*** value.

If supported by the controller, you can also configure the analog input channel:

- In the ***Gain*** field, select the gain value to be used for optical input.

- In the **# *readings*** field, determine the number of readout values of the analog input that are averaged. The greater the noise of the analog input signal, the higher the number of readout values of the analog signal should be set.

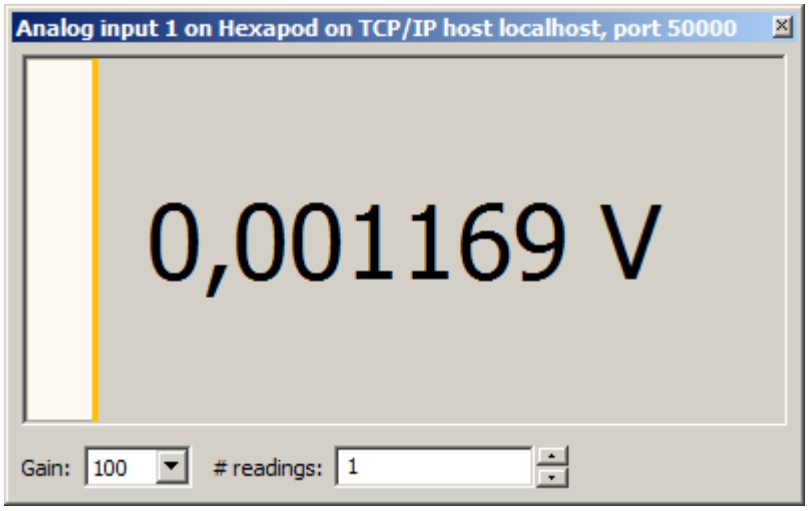


Figure 15: Analog input window

2.1.5 Help Menu

PIMikroMove has a comprehensive help system.

Help

Opens a standard help system with help relating to PIMikroMove. Shortcut: F1

FAQ, Tutorials...

Task-oriented help for PIMikroMove.

Open PI's website

Opens the PI website.

Download the PI Update Finder

Opens the PI website at the download area were the latest version of the PI Update Finder is available. Using the PI Update Finder you can find and download updates for the software (e.g. PIMikroMove, GCS DLL and PIStages2.dat database). For details, see "Installing Updates" (p. 3) and the Technical Notes which are provided with the PI Update Finder.

Show version information…

Opens a window with version information about PIMikroMove and the installed controller software.

About PIMikroMove...

PIMikroMove version information.

2.2 Toolbar

The availability of the individual toolbar icons depends on the connected controller(s) and the axis states.

Note that the toolbar icons are not accessible when the ***Command entry*** window (p. 75) is open.

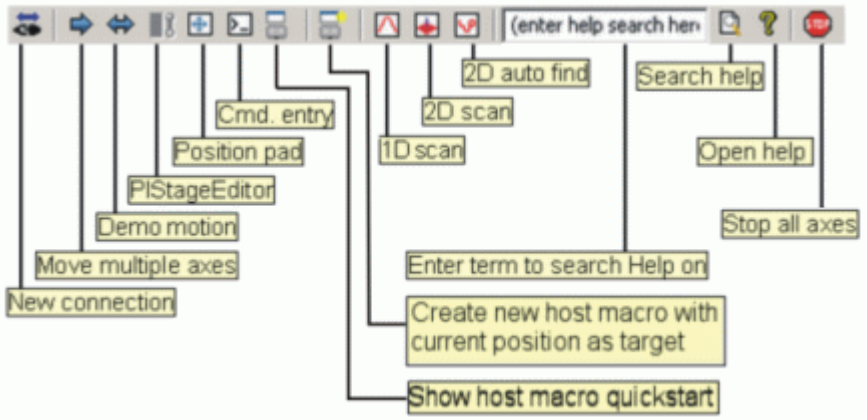


Figure 16: Main window tool bar

New Connection

Show the ***Start up Controller*** window to connect another controller or controller network to PIMikroMove. See "Start up Controller" (p. 171) for details.

Move multiple axes

Only available if at least one of the connected axes is referenced and has servo on. Opens a dialog where targets and velocities for all on-target axes can be entered and the move of all desired axes started at the same time. See "Move Multiple Axes" (p. 69) for details.

Demo Motion

Only available if at least one of the connected axes is referenced and has servo on. Opens a dialog where ranges can be entered and random moves in that range started for demonstration purposes. See "Demo Motion" (p. 71) for details.

PIStageEditor

Only available when the PIStageEditor.exe file is installed on the PC (depending on the controller, the PIStageEditor.exe is automatically installed with the typical setup). Starts the ***PIStageEditor.*** If multiple stage databases are available on the host PC for the connected controllers, you have to select one. Using the ***PIStageEditor*** you can inspect the parameters of all PI stages in the PIStages2.dat stage database. In user-stages dat files you can define and store special parameter sets for new or non-standard stages. See "Stage Editor" (p. 72) and the ***PIStageEditor*** Manual for details.

INFORMATION

Depending on the controller, you can also inspect parameters on the tab cards of the main window, in the ***Single-Axis*** window or using the ***Device Parameter Configuration*** window. See the User manual of the appropriate controller for more information.

Show Position Pad

Only available if for at least one of the connected axes servo is on (axes with incremental sensors must be referenced) and the travel range is known. Displays the ***Position Pad*** window, in which it is possible to control two axes using the mouse pointer to specify the target positions, and/or a joystick (connected to the PC) to vary the velocities. See "Position Pad" (p. 78) for details.

If joysticks are connected to the PC before PIMikroMove is started, multiple ***Position Pads*** will be available. The default assignment of the joystick(s) and their axes to the ***Position Pads*** can be changed in a separate dialog. See "Configure PC Joysticks Window" (p. 91) for details.

Show command entry

Opens the ***Command entry*** window where GCS ASCII commands can be typed in directly. While this window is open, PIMikroMove performs no other communication with the controller, giving the user complete control. As a result, other windows are inactive and not refreshed. See "Command Entry Window" (p. 75) for details.

Show host macro quickstart

Displays the ***Host Macro Quick Start*** window (either docked into the main window or separately) where you have quick access to host macros. You can select and start a host macro without the need to switch to the ***Host macros*** tab card. Furthermore, you can assign a customized “short-cut” button to a host macro so that this macro can be started by simply clicking the button. See “Host Macro Quick Start” (p. 60) for details.

Create new host macro with current position as target

Only available if at least one of the connected axes has servo on (axes with incremental sensors must be referenced). Opens a window where you can create a new host macro containing move command(s) with the current axis position(s) as target(s). See “Create New Host Macro with Current Positions” (p. 62) for details.

Show 1D Scan Dialog

Opens the ***Scan 1D*** window where you can start a single-axis scan motion during which an input source is measured. See "Scan 1D Window" (p. 83) for details. Do not confuse this PIMikroMove functionality with the embedded fast scanning offered by some controllers.

Show 2D Scan Dialog

Opens the ***Scan 2D*** window with the Scan tab card on top where you can start biaxial scan motion during which an input source is measured. See "Scan 2 D Window" (p. 86) and "Scan Tab Card" (p. 87) for details. Do not confuse this PIMikroMove functionality with the embedded fast scanning offered by some controllers.

Show 2D Auto Find Dialog

Opens the ***Scan 2D*** window with the ***Auto Find*** tab card on top where you can start a scan procedure which tries to find the maximum of an intensity signal by modifying the positions of two axes. The measured intensity signal can be, for example, a selected analog input or the position of a selected axis. See "Scan 2 D Window" (p. 86) and "Auto Find Tab Card" (p. 90) for details.Do not confuse this PIMikroMove functionality with the embedded fast scanning offered by some controllers.

Search Help

Searches for terms in PIMikroMove's comprehensive help system, which includes all of this manual.

Open Help

Opens a standard help system with help relating to PIMikroMove.

Stop all

Sends an immediate stop command (ASCII #24) to all axes connected to PIMikroMove.

2.3 Table Tab Cards

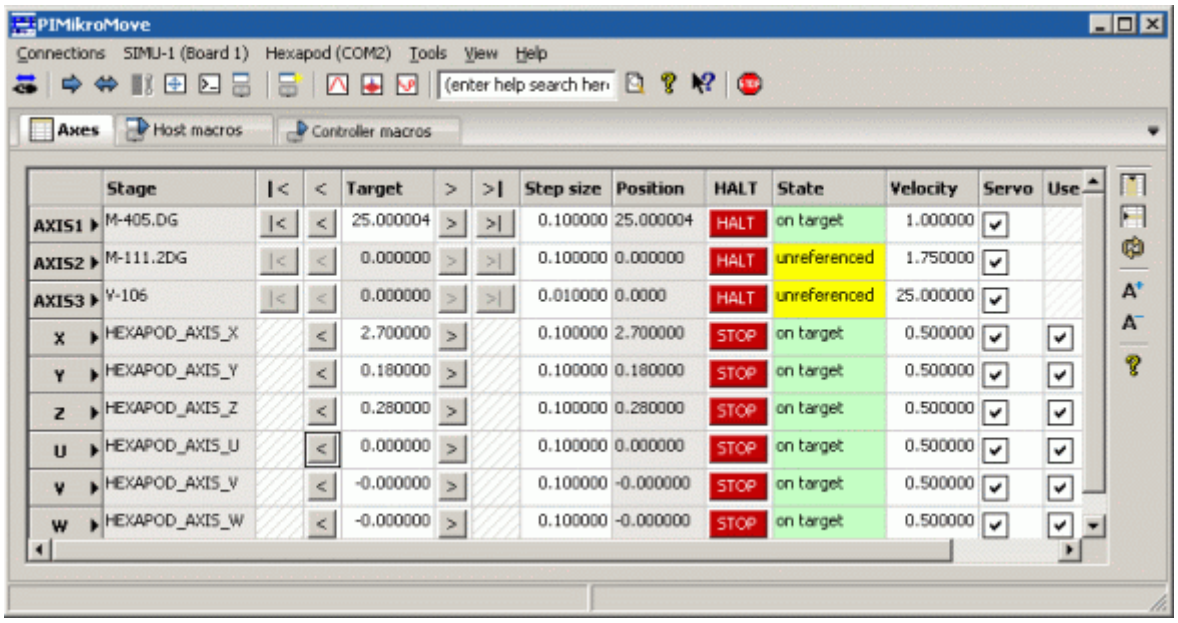


Figure 17: Main window showing the Axes tab card

If supported by the connected controller(s), separate tab cards with tables for axes, piezo channels, sensor channels and PiezoWalk® channels are provided in the main window.

You can select the individual table cards by clicking on their tabs or using the arrow icon(s) on the tab bar(s). It is also possible to display multiple tab cards in the main window by dragging them with the left mouse button pressed.

If a new controller is connected which provides sensor and piezo channels, the corresponding tab cards will automatically be arranged in the main window so that all axes and channels are visible. Exception: If the tab cards were arranged before, PIMikroMove will not change the appearance of the main window.

On a table card, you can specify the columns to be displayed and change the order of the rows (axes or channels). You can also type new values into the white fields.

INFORMATION

The parameter values in the fields are hardware-specific. Wrong values may lead to improper operation or damage of your hardware.

If the target or the open-loop value for an axis is changed, the system will move immediately. PiezoWalk® channels will move when you enter a certain number of steps in the ***OL Number of Steps*** field and click the arrow buttons beside this field.

Controllers will keep the changed settings until the next power on-off cycle. With some controllers, it is possible to store the changed settings with the ***Add/Edit User Stage Type*** item or the ***Save parameters to non-volatile memory*** item of their controller menu (p. 13).

On the right side of the table cards there is a small toolbar with functions to show and hide columns and re-sort the rows. With the ***Refresh*** icon, you can refresh all displayed fields to show their current active values. Using the corresponding buttons, you can increase or decrease the font size used for the table cards.

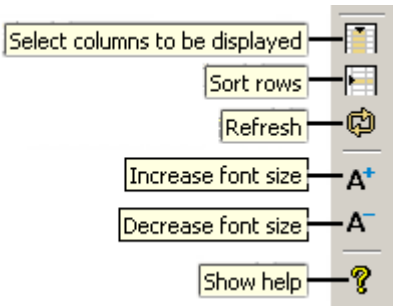


Figure 18: Vertical tool bar of table tab cards

2.3.1 Select Colums to be Displayed

To select the columns to be displayed on the table cards, open the dialog shown below. This can be done from the table cards either by right-clicking into the header of the table or by clicking the corresponding button on the right side of the table.

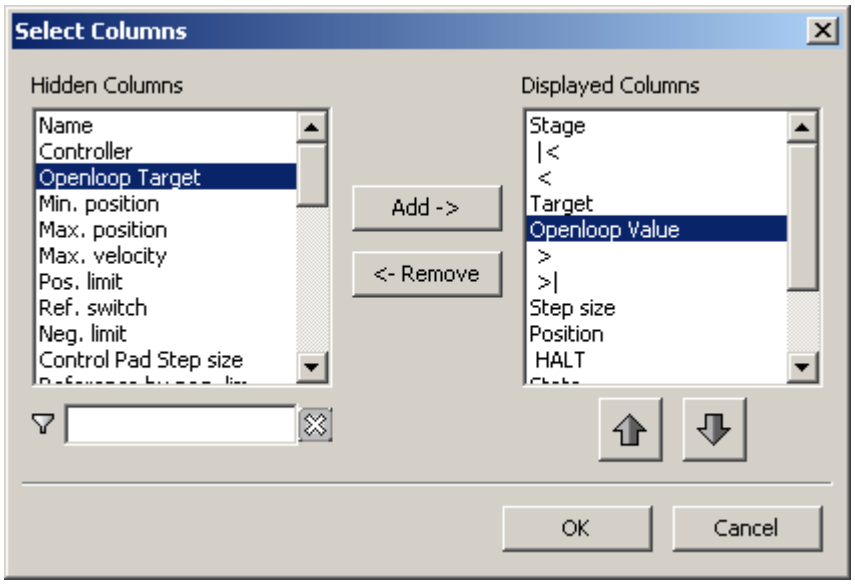


Figure 19: Select columns

Here you can select which axis- or channel-related fields are to be displayed. You can also change the order of the displayed columns with the arrow buttons (top is left, bottom is right). Fields depending on motion are updated when the window is active.

The available fields in parts depend on the connected controller. For a description of "common" fields provided by PIMikroMove see "Fields Displayed" (p. 30). For details regarding controller-specific fields (parameters) see the controller's User manual or GCS DLL manual.

To filter the list of available parameters for certain entries, you can type a part of a parameter name into the input field below the ***Hidden Columns*** pane. The pane will then list only the parameters whose names contain the input you typed. To delete the input, press the ***Clear filter*** text button next to the input field.

2.3.2 Fields Displayed

Axes Table

The following "common" fields are provided by PIMikroMove. They can be displayed if the connected hardware supports the corresponding feature.

Header	Comments
Name	Axis designator
Stage	Stage type
Controller	Controller to which the axis is connected
Target	Current target position
|<	Move to minimum position
<	Perform relative step in negative direction
Step size	Size used for relative steps
>	Perform relative step in positive direction
>|	Move to maximum position
Position	Current position
HALT	Click to stop axis (controller supports HLT command)
STOP	Click to stop all axes connected to the same controller (controller does not support HLT command)
State	State of axis
Min. position	Minimum position
Max. position	Maximum position

Header	Comments
Velocity	Velocity to use during moves
Max. velocity	Limit for velocity
Servo	Servo state
Openloop target	Open-loop target position for axes driven by PiezoWalk® drives (OMA command)
Openloop value	Dimensionless open-loop value for axes driven by piezo actuators, depending on the controller settings it may correspond to a position or to a voltage (approximately)
Pos. limit	Indicates whether positive limit signal is active
Ref. switch	Indicates whether current position is above or below reference point
Neg. limit	Indicates whether negative limit signal is active
Reference by neg. lim.	Referencing is to be done using the negative limit switch: axis moves in negative direction until negative limit switch is tripped, then moves as small amount in positive direction until the limit condition is no longer fulfilled. Afterwards, the absolute position of the axis is set.
Reference by reference	Referencing is to be done using the reference switch: axis moves toward the reference switch until reference signal changes state; if state has changed from high to low, the process is repeated. Afterwards, the absolute position of the axis is set.
Reference by pos. lim.	Referencing is to be done using the positive limit switch: axis moves in positive direction until positive limit switch is tripped, then moves as small amount in negative direction until the limit condition is no longer fulfilled. Afterwards, the absolute position of the axis is set.
Find neg. lim.	Axis moves in negative direction until negative limit switch is tripped; state of axis is not changed.
Find reference	Axis moves toward the reference switch until reference signal changes state; if state has changed from high to low, the process is repeated; state of axis is not changed.
Find pos. lim.	Axis moves in positive direction until positive limit switch is tripped; state of axis is not changed.
Drift compensation	Drift compensation enabled/disabled

Header	Comments
Velocity control	Velocity control enabled/disabled
Online	Online state
Overflow	Overflow state

Piezo Table

The following "common" fields are provided by PIMikroMove. They can be displayed if the connected hardware supports the corresponding feature.

Header	Comments
Name	Channel designator
Controller	Controller to which the piezo channel is connected
Voltage	Current output voltage
Online	Online state

Sensor Table

The following "common" fields are provided by PIMikroMove. They can be displayed if the connected hardware supports the corresponding feature.

Header	Comments
Name	Channel designator
Controller	Controller to which the sensor channel is connected
A/D Value	Current input value from A/D converter
Position	Current position
Normalized Value	Normalized Value

PiezoWalk® Table

The following "common" fields are provided by PIMikroMove. They can be displayed if the connected hardware supports the corresponding feature.

Header	Comments
Name	Channel designator
Controller	Controller to which the PiezoWalk® channel is connected

Header	Comments
Step Amplitude	Step Amplitude
Open-Loop Analog Driving	Voltage for open-loop analog driving (done by the shear piezos)
Open-Loop Velocity	Velocity to use during open-loop nanostepping motion, in step cycles per second
<	Perform open-loop steps in negative direction
Open-Loop Number of Steps	Number of open-loop steps to go, floating point number
>	Perform open-loop steps in positive direction
Open-Loop Remaining Steps	Open-loop remaining steps
HALT	Click to stop PiezoWalk® channel (controller supports HLT command)
STOP	Click to stop all PiezoWalk® channels connected to the same controller (controller does not support HLT command)
Relax	Relax piezos

2.3.3 Sort Rows

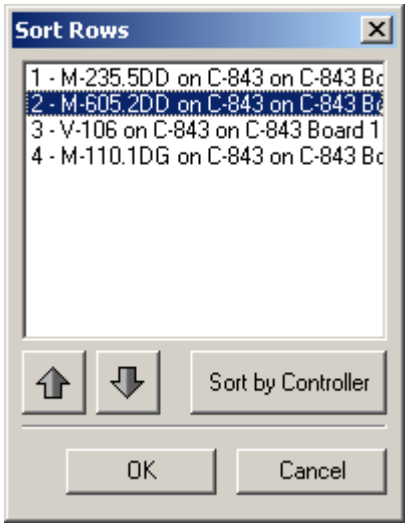


Figure 20: Sort rows

To change the order of the rows displayed on a table card, open the Sort rows dialog. This can be done from the table card either by right-clicking into the header of the table or by clicking the corresponding button on the right side of the table.

"Sort by controller" will sort the axes or channels first by controller and then alphabetically by name. The arrow buttons allow moving the selected axis or channel up or down in the list.

2.3.4 Axis Menu

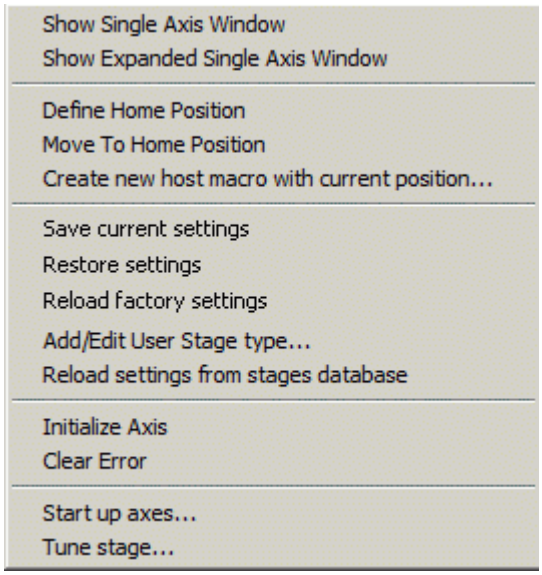


Figure 21: Axis menu, with all possible entries (not all may be available with your controller)

The ***Axis*** menu can be displayed from the ***Axes*** table card by clicking the triangle in the first column of the axes grid, by right-clicking anywhere in the corresponding row or by right-clicking on any free area in the ***Single-Axis Window*** (p. 95). The items of the ***Axis*** menu depend in part on the connected controller.

- ***Show/Hide Single-axis window***
  This will toggle display of the corresponding Single-axis window
- ***Show Expanded Single-axis window***
  This will display the expanded view of the corresponding Single-axis window
- ***Define home position***
  Makes the current physical position the home position of the axis. The current position is set to 0
- ***Move to home position***
  Moves the corresponding axis to its home position by setting the target to 0

- ***Create new host macro with current position***
  Only available if the axis has servo on (axes with incremental sensors must be referenced). Opens a window where you can create a new host macro containing a move command with the current axis position as target. See "Create New Host Macro with Current Positions" (p. 62) for details
- ***Save current settings***
  Saves the current stage configuration and the corresponding motion parameter values of the axis to non-volatile memory of the controller and thus make them the new power-on defaults
- ***Restore settings***
  The stage configuration and the motion parameters which were last saved with ***Save current settings*** are loaded and become the currently active settings
- ***Reload factory settings***
  The factory-default motion parameters of the stage which is connected to the axis are loaded and become the currently active settings
- ***Add/Edit User Stage type...***
  The current stage parameter values can be stored as a new stage type in the user-stages dat file on the host PC. If the stage type already exists in the user stage database, its settings will be overwritten when the new ones are saved. For the stage type entry, do not use stage names which already exist in the PIStages2.dat database. If a stage of the same name exists in PIStages2.dat and the user-stages dat file, the parameter settings from PIStages2.dat will be preferred when assigning that stage to an axis (e.g. in the ***Select connected stages*** dialog (p. 174)), and the settings from the user-stages dat file will never be used.
  See "Stage Editor" (p. 72) for details on stage databases.
- ***Reload settings from stages database***
  The parameter values for the current stage type are reset to the values from the stage database (PIStages2.dat or user-stages dat file). Note that this resets the referencing state of the axis and may switch the servo off.
- ***Initialize axis***
  Initialize the corresponding axis and can sometimes move it off a limit switch
- ***Clear error***
  Shows current error code with explanation and, if possible, returns the axis or axes to a non-error state
- ***Start up axes...***
  Opens the ***Start up Controller*** window at the ***Start up axes*** step. See "Start Up Axes" (p. 179) for more information

- ***Tune stage...***
  Opens the ***PITuningTool***. With this tool you can easily determine well-suited servo parameters for the axis. See "PI Tuning Tool" (p. 105) for more information.

2.4 Hexapod 3D View

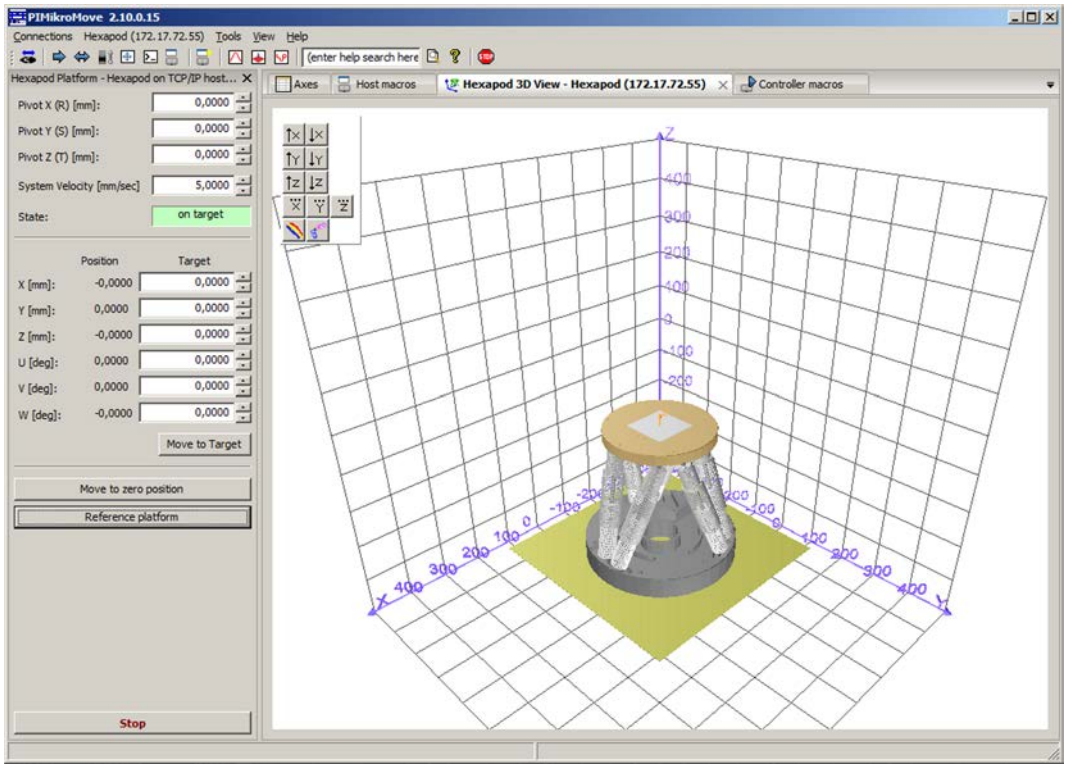


Figure 22: Main window showing the Hexapod 3D View tab card and the Hexapod Platform window (docked)

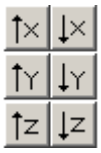
For every Hexapod system (controller or simulation) connected to PIMikroMove, a ***Hexapod 3D View*** tab card is provided in the main window.

The ***Hexapod 3D View*** tab card visualizes the following:

- Current position of the Hexapod
- Current location of the pivot point
- Any objects which were defined for the environment of the Hexapod (see below)

If the ***Hexapod 3D View*** tab card is present in the main window, it can be brought to the front by clicking on its tab or using the arrow icon on the tab bar. To remove the ***Hexapod 3D View*** tab card from the main window, click on the cross in the corner of the tab. To reopen the tab card, use the ***Hexapod 3D View > Show*** menu sequence on the controller menu.

You can change the visualization on the ***Hexapod 3D View*** tab card as follows:

- Change the orientation of the Hexapod with the  buttons on the tab card so that the specified axis points in the specified general direction on the screen.
- Change the position of the grid with the  buttons on the tab card. Click several times to return to starting point.
- Change the background color using the  button on the tab card.
- Rotate the view by dragging the mouse pointer across the tab card with the left mouse button pressed.
- Zoom in or out on the view by dragging the mouse pointer across the tab card with the right mouse button pressed.
- Load a DAT file with geometric data for the Hexapod and select the visualization mode (detailed CAD view or schematic view) using ***Hexapod 3D View > Change visualization…*** This item is available on the controller menu when the ***Hexapod 3D View*** tab card is present in the main window. Loading a DAT file only affects the visualization on the ***Hexapod 3D View*** tab card and not the actual motion of the Hexapod.

  Note that loading a DAT file is only necessary when PIMikroMove cannot find suitable data for the visualization of the connected Hexapod (e.g. when a customized Hexapod is used and you have received a special DAT file from PI).

To visualize motions of the Hexapod, the display on the ***Hexapod 3D View*** tab card is permanently updated. To start motions, using the ***Hexapod Platform*** window is recommended where you can also change the pivot point coordinates. By default, for every Hexapod system (controller or simulation) connected to PIMikroMove, a ***Hexapod Platform*** window is docked in the main window. See "Hexapod Platform Settings" for details.

The display is also updated permanently during moves started, for example, from the ***Axes*** tab card (p. 28) in the main window, from ***Single-Axis*** windows (p. 95), from the ***Demo Motion*** window (p. 71) or from the ***Move multiple axes*** window (p. 69).

INFORMATION

The display on the ***Hexapod 3D View*** tab card is not updated as long as the ***Command entry*** window (p. 75) is open.

You can adapt the visualization on the ***Hexapod 3D View*** tab card to your application by defining items which are present on the Hexapod or in the environment of the Hexapod. This way you can estimate if collisions will occur during the Hexapod motion.

INFORMATION

Defining items on the ***Hexapod 3D View*** tab card does **not** affect the allowed travel ranges for the actual motion of the Hexapod and hence does **not** avoid collisions. You can define configurations for avoiding collisions using the PIVeriMove software which must be ordered separately (order No. C-887.VM1).

Use the  button on the tab card to open the ***Collision tree setup*** pane where you can define items as follows:

- Add a new item to any item of the tree, delete or copy an item: Select the item in the tree, click with the right mouse button and select ***Add*** or ***Delete*** or ***Copy*** from the context menu.
- Change the location, the dimensions, the shape, the color and the type of visualization for the item: Click on the item in the tree. At the bottom of the ***Collision tree setup*** pane, fields and controls are shown with which you can change the properties of the item.

2.5 Host Macros

Host macros allow you to store a sequence of commands in a text file for later execution by a connected controller. Basics of host macro usage are given in "Controls on Host Macro Tab Card" (p. 41) and "Controls in ***Run host macro*** Window" (p. 45).

Host macros can be useful, for example, in the following cases:

- Repeat specific actions:
  Using host macros, you can, for example, store a specific action done with PIMikroMove. To obtain the corresponding command sequence, open a log window for the controller of interest via the ***Log window…*** item on its controller menu (see "Log Window" (p. 169)). In the log window, monitor the commands which are sent to the controller when you use the controls of PIMikroMove. In most cases it will be sufficient to copy the complete command sequence from the log window and save it as host macro. Example: Waveforms defined in the PI Wave Generator Tool (p. 111) for the controller's wave tables are lost when the controller is powered down or rebooted. Using a host macro you can store them permanently.
- Move to certain positions:
  Should it be necessary to recall certain positions, you can use the ***Create new host macro with current position…*** functionality of PIMikroMove. Via the appropriate toolbar button or axis menu item, you can create a new host macro which simply contains a move command with the current position as target. Furthermore, you can assign a separate button to this macro for easy access from the PIMikroMove main window. See "Create New Host Macro with Current Positions" (p. 62) and "Host Macro Quick Start" (p. 60) for details
- Develop controller macros:
  If the controller is able to store sequences of commands ("controller macros"), it is recommended to develop and test them as host macros firstly. Once the host macro is running successfully, you can load it to the controller as controller macro. This way you can use, for example, the ***Run host macro…*** window to monitor macro execution and variable values while developing the macro. Furthermore, the macro will be written to the controller's non-volatile memory only once (when finished) and not every time you change the macro. See "Controls in Run host macro Window" (p. 45) and "Controller Macros" (p. 64) for details.

INFORMATION

Host macros are interpreted by the host PC and PIMikroMove. This can be done for any controller even if the controller itself cannot store macros. If the controller supports macros, it may be faster to use them. (See also "What's the difference between "controller macros" and "host macros"?"). The timing of host macro operation is influenced by the time resolution of the operating system on the host PC (no real time operation).

There are special commands for usage in host macros (WAC (p. 47), MEX (p. 47), JRC (p. 48), BREAK (p. 48), MAC NSTART (p. 49), CMAC (p. 49), LOOP (p. 49), PRINT/MESSAGE/MESSAGE_CANCEL (p. 49), VAR (p. 53), VAR? (p. 53), ADD (p. 53), CPY (p. 54)).

Every other command is sent as-is to the controller when a host macro is executed. After each line sent to the controller, PIMikroMove checks the state of the controller by polling #7 (query the busy state) and #5 (query if axes are moving). If the controller indicates that it is "busy", PIMikroMove will wait before sending the next line of the host macro. If the axes of the stage are "moving", PIMikroMove will not wait but continue host macro execution. Example: Depending on the controller, the FRF command (starts a referencing procedure) will set the controller in the "busy" state. In this case, after sending a line containing FRF PIMikroMove will wait for FRF to finish.

Variables are provided for more flexibility in host macro programming. See "Variables" (p. 54) for details.

Macro command sequences are given for the following examples:

- Wait and Trigger (p. 55)
- Trigger at Specified Position (p. 56)
- X-Y-Scan with Different Macros (p. 57)
- X-Y-Scan with Loops (p. 58)
- Pushbutton Control (p. 59)

2.5.1 Controls on Host Macro Tab Card

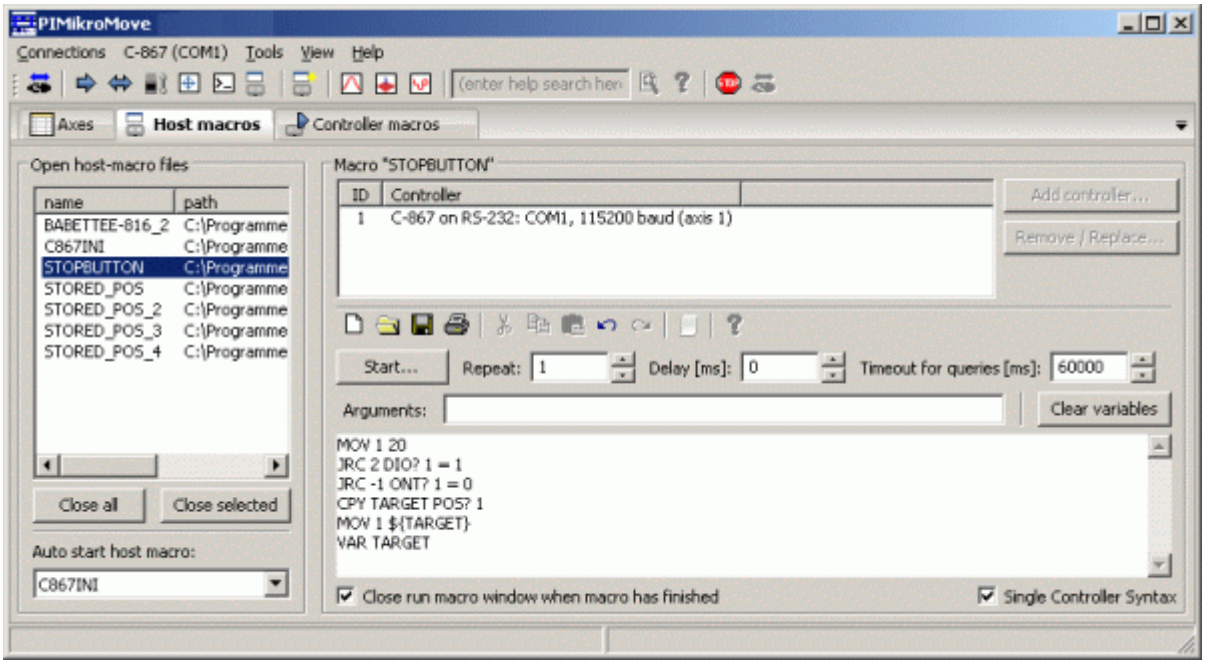


Figure 23: Host macro tab card

The ***Host macros*** tab card gives access to all connected controllers—multiple controllers can be combined in one host macro. You can select the ***Host macros*** tab card by clicking on its tab or using the arrow button in the tab bar.

List of available host macros

The left side of the ***Host macros*** card has a list with the available host macro files. To add a macro existing on the host PC to the list, use the ***Open and show another macro from text file*** button in the icon bar of the card. The host macro selected in the list can be edited, executed or saved, and all controllers affected by this macro are shown in the top right pane of the card. The macro list also shows the directory in which the host macro file is located on the host PC. If the entry is <not saved>, the host macro was not saved to the host PC yet.

Host macro execution

With the ***Start…*** button in the center of the ***Host macros*** card, you can start the execution of the currently selected host macro.

You can also specify that the macro command sequence will be executed repeatedly a certain number of times when the macro is run (***Repeat***).

A fixed execution delay can also be specified between the command lines in the ***Delay [ms]*** field: After #7 has indicated that the controller is not busy anymore, PIMikroMove will wait that amount of time between sending each line. (Note: On Windows systems, the time resolution is about 10 ms, so you will notice no difference between a delay of 5 and 10 ms!)

In the ***Timeout for queries [ms]*** field you can specify how long PIMikroMove has to wait for long-lasting macro commands to be processed.

INFORMATION

Controllers with GCS syntax version 1.0 may send a resonse after a long-lasting command has finished (e.g. after scanning procedures started with FSA for F-206 Hexapod systems or referencing procedures started with MNL, MPL and REF for C-863.10, C-663.10 and C-843 motor controllers). This response has to be "catched" by the host macro to avoid failure in the macro execution. To catch the response, use such long-lasting commands in host macros in combination with a command which checks conditions for further macro execution (e.g. MEX, JRC or WAC). You can specify the timeout value using the ***Timeout for queries [ms]*** field.

Example 1 simply "ignores" the response: JRC 1 MNL A = 1 will cause the macro execution to continue with the next line (jump one line ahead) if MNL returned 1. But JRC will also continue with the next line in any other case.

Example 2: You can "use" the response in combination with WAC, MEX and JRC. E.g. with MEX REF A = 0 the macro execution will be stopped if referencing of axis A failed.

In the ***Arguments*** field you can enter the values for the local variables used in the macro (see "Variables" (p. 54) for details). The values have to be separated by space characters. Macro execution may fail if the macro uses local variables but no entries are made in the Arguments field.

Using the ***Clear variables*** button, you can delete all variables in the host macro interpreter of PIMikroMove.

Macro execution can be monitored, paused and restarted in the ***Run host macro…*** window (see "Controls in Run host macro Window" (p. 45) for details).

INFORMATION

The host macro selected in the macro list can only be executed if all controllers concerned by that host macro are connected to PIMikroMove via the interface(s) used during host macro creation. Controllers which are affected by the currently selected host macro but not connected or connected via an interface other than the one used during macro creation are highlighted in red in the controller list.

If the controller affected by the selected host macro is highlighted in red in the controller list, you can replace it by a currently connected controller. To do this, click on the highlighted controller and use the ***Remove / Replace…*** button and select the new controller from the list that opens (you can also click on the controller entry with the right mouse button to obtain the list).

Auto start host macro

Below the macro list, you can select one of the macros to be executed automatically at the next AutoConnect start of PIMikroMove. Selection is only possible if the AutoConnect feature is activated (see "AutoConnect" (p. 183) for details). Note that the selected macro will only be executed if AutoConnect was successfully finished for all controllers affected by the macro.

Macro editor

The bottom right pane of the ***Host macros*** card provides a macro editor, consisting of a text field and an icon bar with buttons for creating, opening and saving macro command sequences. Note that PIMikroMove provides a saving dialog if there are unsaved host macros when you quit the program.

The following keyboard shortcuts are available:

- Ctrl+S saves the current macro
- Ctrl+R starts the current macro (without saving)

In the text field, you can either edit the currently selected macro or write a new host macro. The individual lines of the macro consist of a controller identifier (integer number from 0 to 255) and the command. If a line starts with ";", "'" or "//" it is ignored and so it can be used to comment macros.

With the controller identifier it is possible to specify the controller which should receive the command. Two special identifiers exist: 0 is the host PC and can only be used for commands that PIMikroMove executes (like DEL and MAC START/NSTART), and 255 stands for a broadcast to all connected controllers, i.e. the command is sent to each controller specified for this macro.

If only one controller is defined for a macro the ***Single Controller Syntax*** can be used, and macros can be written without explicitly specifying the controller ID for each line (see the checkbox on the right bottom side of the card).

Special commands

During macro execution, the commands are sent as-is with no syntax checking before being sent on, except for the following special commands that are interpreted by PIMikroMove directly:

- WAC <command?> <operator> <value> waits until the specified axis condition occurs (see "WAC <command?> <operator> <value>" (p. 47))
- MEX <command?> <operator> <value> stops macro execution when a given condition is fulfilled (see "MEX <command?> <operator> <value>" (p. 47))
- JRC <jump> <command?> <operator> <value> performs a relative jump of the macro execution pointer when a given condition is fulfilled (see "JRC <jump> <command?> <operator> <value>" (p. 48)).
- BREAK pauses macro execution. Macro execution can be continued with  in the Run host macro window.
- DEL <delay in milliseconds> waits for the specified time
- MAC START <name> calls another host macro. MAC NSTART <name> <rp_val> calls another host macro and runs it rp_val times (see "MAC START/NSTART" (p. 49))
- CMAC <options> calls the "MAC" command on the corresponding controller, i.e. gives access to the complete macro functionality of the controller (see "CMAC <options>" (p. 49))
- LOOP <nr of loops> marks the start of a loop in the macro (see "LOOP <nr> / ENDLOOP" (p. 49))
- ENDLOOP marks the end of a loop (see "LOOP <nr> / ENDLOOP" (p. 49))
- PRINT <”text”> <GCS query command> outputs text and command response to the PRINT Statements window during macro execution (see "PRINT / MESSAGE / MESSAGE_CANCEL" (p. 49)).
- MESSAGE <”text”> <GCS query command> outputs text and command response to a message box during macro execution and pauses macro execution (see "PRINT / MESSAGE / MESSAGE_CANCEL" (p. 49)).
- MESSAGE_CANCEL <”text”> <GCS query command> outputs text and command response to a message box during macro execution and pauses or aborts macro execution (see "PRINT / MESSAGE / MESSAGE_CANCEL" (p. 49)).
- VAR <variable> <string> sets a variable to a certain value (see "VAR <variable> <string>" (p. 53)).

- VAR? <variable> gets the variable value. VAR? must be combined with CPY, JRC, MEX or WAC (see "VAR? <variable>" (p. 53)).
- ADD <variable> <FLOAT1> <FLOAT2> adds two values and saves the result to a variable. It is also possible to substract using the ADD command (see "ADD <variable> <FLOAT1> <FLOAT2>" (p. 53)).
- CPY <variable> <command?> copies a command response into a variable (see "CPY <variable> <command?>" (p. 54)).

When a command starts a longer process on the controller (e.g. a move to the reference switch) the execution of the macro is paused as long as this takes place. You do not need to use WAC to wait for such a process to finish.

Version compatibility

In previous versions of ***PIMikroMove***, all commands in a given host macro must address the same controller or controller network. If a macro stored with these versions is loaded and more than one controller is connected the user is prompted to select a controller for this macro. For such old macros ***Single Controller Syntax*** is activated. Once loaded, additional controllers can be added.

Older versions of ***PIMikroMove*** might not be able to load macros saved with the current version.

2.5.2 Controls in Run host macro Window

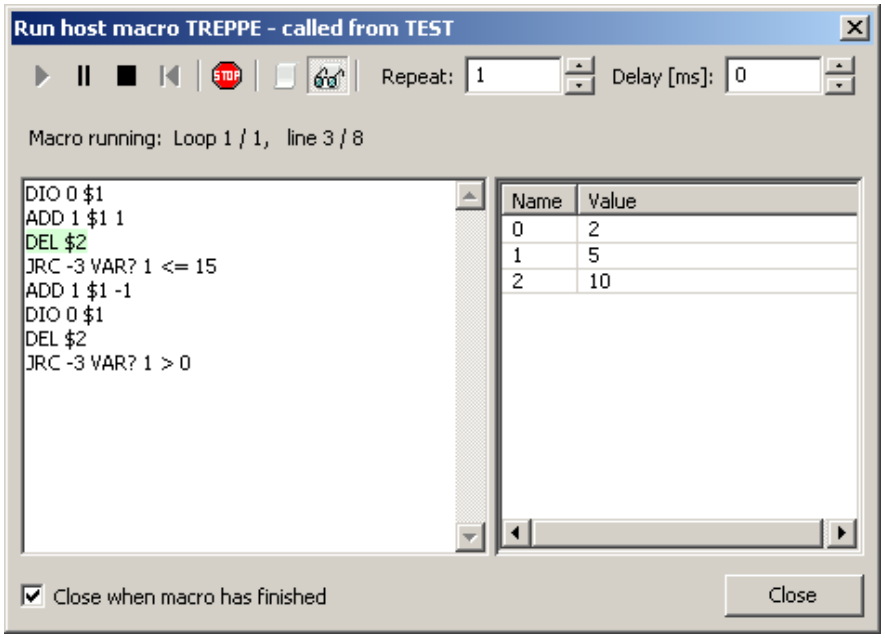


Figure 24: Run host macro window; macro Treppe is executed

When a host macro is executed (***Start…*** button in the center of the ***Host macros*** tab card was pressed), the ***Run host macro*** window opens. There you can monitor macro execution and stop or restart it. This can be helpful, for example, during debugging.

Below the toolbar, the currently running macro is listed. It is either the macro started by the ***Start…*** button on the ***Host macros*** tab card, or another macro called by that macro. See the title bar of the ***Run host macro*** window for the name of the running macro. In the list, the macro execution pointer is visualized by a small green bar.

The toolbar items on the top of the window have the following functions:

-  starts macro execution
-  pauses macros execution, same function as BREAK command
-  stops macro execution in PIMikroMove, i.e. the macro command sequence is no longer sent to the controller
-  resets macro execution pointer and values of local variables when macro execution was paused with  or BREAK. The macro will be restarted with the first line when  is used afterwards.
-  stops all, i.e. not only macro execution in PIMikroMove but also all other tasks on the controller (e.g. motion, scan algorithms, …)
-  opens separate window with PRINT statements. During macro execution, PIMikroMove can output arbitrary text and the response to GCS query commands using the PRINT <”text”> <GCS query command. See "PRINT / MESSAGE / MESSAGE_CANCEL" (p. 49) for details.
-  shows/hides a variable list where you can monitor the variable values. The list comprises all variables which are available for the currently running macro: the macro’s local variables and all global variables present in the host macro interpreter of PIMikroMove . You can delete the variables via the ***Clear variables*** button on the Host macros tab card.
- ***Repeat*** field: You can specify that the macro command sequence will be executed repeatedly a certain number of times when the macro is run.
- ***Delay [ms]*** field: A fixed execution delay can be specified between the command lines: PIMikroMove will then wait that amount of time between sending each line. (Note: On Windows systems, the time resolution is about 10 ms, so you will notice no difference between a delay of 5 and 10 ms!)

2.5.3 WAC <command?> <operator> <value>

WAit until a given **C**ondition of the following type is fulfilled: The condition involves comparing a specified value with the result of a command that returns one value on one line.

<command>	is the command in its full syntax. The command must be one that provides a response consisting of one single value on one line.
<operator>	is the comparison operator; supported are "=" "<=" "<" ">" ">=" "!=" Important: There must be a blank space before and after the operator!
<value>	is the value to be used on the right side of the comparison.

For example: with WAC ONT? 1 = 1 you can wait for axis 1 to be on target. WAC DIO? A = 1 waits for the digital input A to be high. WAC POS? B > 4 waits for axis B to reach a position greater than 4.

2.5.4 MEX <command> <operator> <value>

Stop macro execution when a condition of the following type is fulfilled. The condition involves comparing a specified value with the result of a command that returns one value on one line.

When the macro interpreter accesses this command for execution the condition is checked. If it is true the macro is stopped, otherwise the macro execution continues with the next line. Later fulfillment of the condition will not in itself trigger any action.

<command>	is the command in its full syntax. The command must be one that provides a response consisting of one single value on one line.
<operator>	is the comparison operator; supported are "=" "<=" "<" ">" ">=" "!=" Important: There must be a blank space before and after the operator!
<value>	is the value to be used on the right side of the comparison

For example: MEX DIO? A = 0 will exit the macro if the digital input A is 0.

All arguments of the MEX command can be omitted. In this case, macro execution is stopped by MEX without evaluating a condition.

2.5.5 JRC <jump> <command?> <operator> <value>

Jump relatively depending on a given condition of the following type: one given value is compared with a queried value according to a given rule.

<jump>	is the size of the relative jump. -1 means the macro execution pointer goes to the previous line, 0 means the command is executed again, which is the same behaviour as WAC. 1 goes to the next line, making the command unnecessary, and 2 jumps over the next command. Only jumps within the current macro are allowed.
<command?>	is one query command in its usual syntax. The response has to be a single value and not more.
<operator>	is the comparison operator; supported are "=" "<=" "<" ">" ">=" "!=" Important: There must be a blank space before and after the operator!
<value>	is the value to be used on the right side of the comparison

Example: Using the following macro, you can stop motion of axis "1" using a stop-button connected to a digital input. The result of the POS? 1 query is being copied to the variable TARGET. Then this variable is used as second argument for the MOV command. Thus the stage stays where it just was. To clean up, TARGET is then defined empty with the VAR command which deletes the variable.

Write macro "stop":

```
MOV 1 20
JRC 2 DIO? 1 = 1
JRC -1 ONT? 1 = 0
CPY TARGET POS? 1
MOV 1 ${TARGET}
VAR TARGET
```

2.5.6 BREAK

Pauses macro execution, same as if  is pressed in the ***Run host*** macro window. Macro execution can be continued with  .

BREAK is helpful when debugging macros.

2.5.7 MAC [N]START <name>

With MAC START you can call another macro. <name> specifies the filename of the text file with the macro content. With MAC NSTART you can specify how many times the called macro should be run. Thus you can implement loops by placing the commands of the loop inside a separate macro file and calling it with MAC NSTART (see sample below).

2.5.8 CMAC <optionsY>

If a controller provides macro functionality, CMAC <options> gives complete access to it by calling the "MAC" command. CMAC is to be used with the syntax and options of the MAC command as described in the controller User manual. For example, "CMAC START DEMO" starts the macro DEMO stored on the controller by sending the "MAC START DEMO" command to the controller. With "CMAC BEG <macroname>" and "CMAC END" controller macros can be created at run time.

2.5.9 LOOP <nr> / ENDLOOP

With LOOP and ENDLOOP you can mark a section in the macro as loop code. The number of repetitions must be specified with the argument to LOOP. Loops can be nested without restrictions but the LOOP and ENDLOOP statements must be in the same macro.

2.5.10 PRINT / MESSAGE / MESSAGE CANCEL

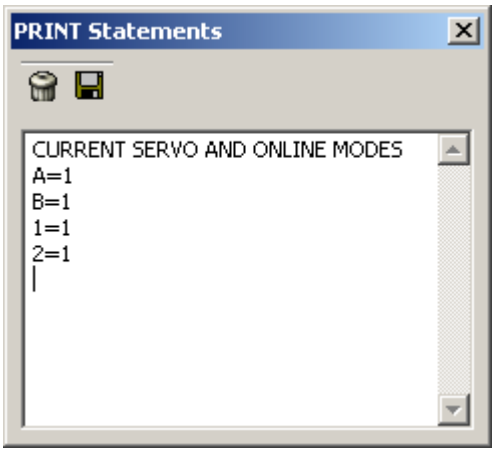
During macro execution, PIMikroMove can output arbitrary text and the response to GCS query commands. Availability of the output and further macro processing depend on the output command used which can be PRINT, MESSAGE or MESSAGE_CANCEL. The output can be set by the optional <"text"> and <GCS query command> arguments of the output commands. All characters between the quotation marks in the <"text"> argument will be output as is. The <GCS query command> argument gives one GCS query command in its usual syntax. This query command will be sent to the controller, and every response from the controller will be output.

Output command details:

- PRINT <”text”> <GCS query command>

  All output is written to the ***PRINT Statements window*** (see figure below).

  This window can be opened using the  ***Show window with PRINT statements*** icon in the toolbar of the ***Host macros*** tab card. Macro execution is not affected by the PRINT command.



In the example shown in the figure above, the output is caused by the following PRINT commands (which are part of an executed host macro):

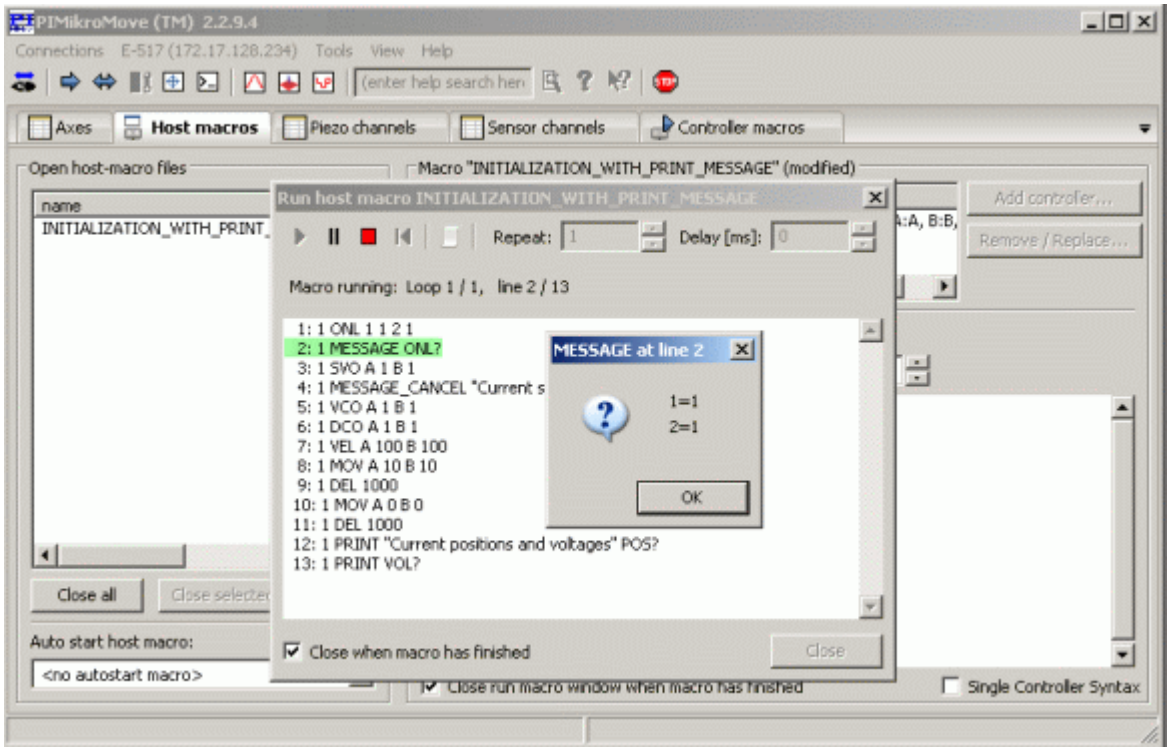
PRINT "Current servo and online modes" SVO?

PRINT ONL?

You can save the content of the ***PRINT Statements*** window to a text file on the host PC.

- MESSAGE <"text"> <GCS query command>

  All output is written to a message box which pops up when the macro interpreter accesses the MESSAGE command for execution (see figure below). Macro execution is then paused until you press ***OK*** in the message box.

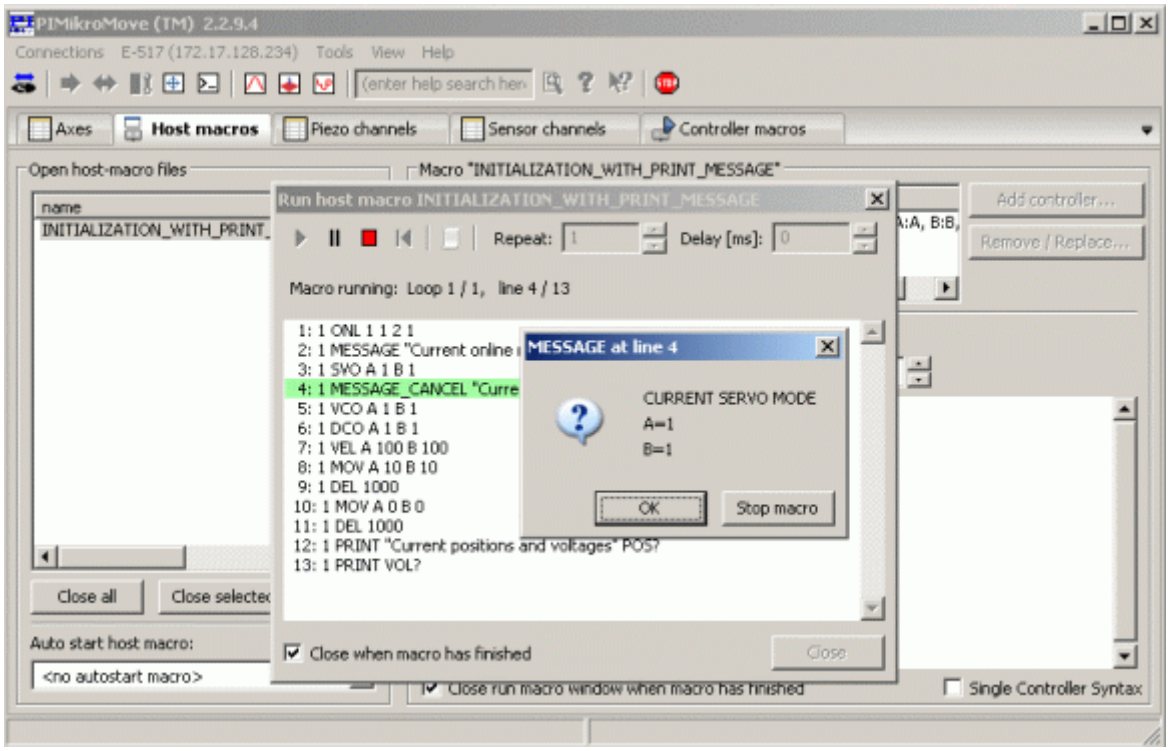


In the example shown in the figure above, the output is caused by the following MESSAGE command (which is part of the executed host macro):

MESSAGE ONL?

- MESSAGE_CANCEL <"text"> <GCS query command>

  All output is written to a message box which pops up when the macro interpreter accesses the MESSAGE_CANCEL command for execution (see figure below). Macro execution is then paused until you press ***OK*** in the message box or can be aborted by the ***Stop macro*** button.



  In the example shown in the figure above, the output is caused by the following MESSAGE_CANCEL command (which is part of the executed host macro):

  MESSAGE_CANCEL "Current Servo Mode" SVO?

2.5.11 VAR <variable> <string>

Set a variable to a certain value. See "Variables" (p. 54) for conventions regarding variable names and values.

<variable>	is the name of the variable whose value is to be set
<string>	is the value to which the variable is to be set. If omitted, the variable is deleted. The value can be given directly or via the value of a variable.

Example: It is possible to set the value of one variable (e.g. SOURCE) to those of another variable (e.g. TARGET):

```
VAR TARGET ${SOURCE}
```

2.5.12 VAR? <variable>

Get variable values. If VAR? is combined with CPY, JRC, MEX or WAC, the response to VAR? has to be a single value and not more.

<variable>	is the name of the variable to be queried. See "Variables" (p. 54) for name conventions.

2.5.13 ADD <variable> <FLOAT1> <FLOAT2>

Add two values and save the result to a variable. For the summands, floating point numbers are expected. They can be given directly or via the value of a variable. See "Variables" (p. 53) for conventions regarding variable names and values.

<variable>	is the name of the variable to which the result is to be saved
<FLOAT1>	is the first summand
<FLOAT2>	is the second summand

It is also possible to substract using the ADD command. Example:

```
ADD c $b -$a
```

will substract the value of variable a from those of variable b. The result will be written to variable c. Note that the value of variable a must not have a negative sign.

2.5.14 CPY <variable> <command?>

Copy a command response into a variable.

<variable>	is the name of the variable to which the command response is to be copied. See "Variables" (p. 54) for conventions regarding variable names.
<command?>	is one query command in its usual syntax. The response has to be a single value and not more.

Example: It is possible to copy the value of one variable (e.g. SOURCE) to another variable (e.g. TARGET):

```
CPY TARGET VAR? SOURCE
```

2.5.15 Variables

With host macros, variables are provided for more flexibility in programming. While global variables can be used for multiple macros, local variables are only valid for a given macro. There is no limit for the number of local and global variables.

The values of local variables must be given in the ***Arguments*** field of the ***Host macros*** tab card before starting the macro. The values have to be separated by space characters. Macro execution may fail if the macro uses local variables but no entries are made in the ***Arguments*** field.

Using the ***Clear*** variables button on the ***Host macros*** tab card, you can delete all variables in the host macro interpreter of PIMikroMove.

During macro execution, variable values can be monitored in the ***Run host macro window*** (p. 45).

Inside of host macros, you can use the ADD, CPY, VAR and VAR? commands to deal with variables.

INFORMATION

The local variable 0 is read-only. Its value gives the number of arguments set for the host macro.

Conventions for variable names:

- Variable names must not contain special characters, especially no “$”.
- The maximum number of characters is 8.
- Names of global variables can consist of characters A to Z and 0 to 9. They must start with an alphabetic character.

- Names of local variables must not contain alphabetic characters. Possible characters are 0 to 9.
- The variable name can also be given via the value of another variable.

If the value of a variable is to be used, the notation must be as follows:

- The variable name must be written with preceding “$”.
- Variable names consisting of multiple characters must be put in curly brackets.
- If the variable name consists of a single character, the curly brackets can be omitted.

Note that if the curly brackets are omitted with variable names consisting of multiple characters, the first character after the “$” is interpreted as the variable name.

2.5.16 Example: Wait and Trigger

Single controller example of how to send a trigger signal after the motion of an axis has finished. Put the logic of waiting and triggering in a macro named WAITTRIG and call it after each MOV or MVR command.

Macro waittrig.txt:

```
WAC ONT? A = 1

DIO 1 1

DEL 100

DIO 1 0
```

You can use this in other macros:

```
MOV A 10

MAC START WAITTRIG

MOV A 0

MAC START WAITTRIG
```

2.5.17 Example: Trigger at Specified Position

Single controller example of how to send a trigger signal at a specified position during the motion of an axis.

```
DIO 1 0
MOV 1 10
WAC POS? 1 > 5
DIO 1 1
WAC ONT? 1 = 1
DIO 1 0
MOV 1 0
WAC ONT? 1 = 1
```

This macro will do the following:

Line 1: set digital output 1 to 0

Line 2: start motion of axis 1, target 10

Line 3: wait until the position of 1 is greater than 5

Line 4: set digital output 1 to 1

Line 5: wait until axis 1 is on target

Line 6: set digital output 1 to 0

Line 7: move back to 0

Line 8: wait until axis is on target

2.5.18 Example: X-Y-Scan with Different Macros

Example of loops implemented with MAC NSTART: All three macros together will perform a 10x10 mm scan. At each target point during the scan one of the digital outputs is set high for 100 ms to trigger an external device.

Macro scanxy.txt:

This macro will move the two axes A of controller 1 and C of controller 2 to their home positions and wait for both axes to stop. Then the macro scanrow.txt is called 10 times.

```
1 GOH A
2 GOH C
255 WAC #5 = 0
0 MAC NSTART SCANROW 10
```

Macro scanrow.txt:

This macro will move axis A of controller 1 by 1 mm and wait for it to stop. Then the macro scanstep.txt is called 10 times. At the end, axis C of controller 2 is moved back to its home position.

```
1 MVR A 1
1 WAC ONT? A = 1
0 MAC NSTART SCANSTEP 10
2 GOH C
2 WAC ONT? C = 1
```

Macro scanstep.txt:

This macro will move axis C of controller 2 by 1 mm, wait for it to stop and set the digital output 1 of controller 1 to 1 for 100 milliseconds.

```
2 MVR C 1
2 WAC ONT? C = 1
1 DIO 1 1
```

```
0 DEL 100
1 DIO 1 0
```

2.5.19 Example: X-Y-Scan with Loops

This macro will move the two axes A of controller 1 and C of controller 2 to their home positions and start a 10x10 mm scan with both axes.

Macro scanxyloop.txt:

```
1 GOH A
2 GOH C
255 WAC #5 = 0
0 LOOP 10
1   MVR A 1
1   WAC ONT? A = 1
0   LOOP 10
2      MVR C 1
2      WAC ONT? C = 1
1      DIO 1 1
0      DEL 100
1      DIO 1 0
0   ENDLOOP
2   GOH C
2   WAC ONT? C = 1
0 ENDLOOP
```

2.5.20 Example: Pushbutton Control

If you have connected four push buttons to 4 digital inputs you can implement pushbutton control of five different macros using the "MEX" command.

The following single controller example illustrates a small program: you can move axis A 1 mm forward or backward with button 1 and 2 and you can switch between two velocities with buttons 3 and 4.

Macro KEYPBLOOP.txt

This macro will call the macros for the individual buttons and then call itself.

```
MAC START KEYPB1
MAC START KEYPB2
MAC START KEYPB3
MAC START KEYPB4
MAC START KEYPBLOOP
```

Macro KEYPB1.txt

This macro will exit if digital input 1 (keyPB 1) is not active, otherwise a relative move of 1 mm is made for axis A. The short delay prevents this macro being called too rapidly when the button is pressed.

```
MEX DIO? 1 = 0
MVR A 1
DEL 100
```

Macro KEYPB2.txt

This macro will exit if digital input 2 (keyPB 2) is not active, otherwise a relative move of -1 mm is made for axis A. The short delay prevents this macro being called to rapidly when the button is pressed.

```
MEX DIO? 2 = 0
MVR A -1
DEL 100
```

Macro KEYPB3.txt

This macro will exit if digital input 3 (keyPB 3) is not active, otherwise the velocity is set to 0.5. The short delay prevents this macro being called too rapidly when the button is pressed.

```
MEX DIO? 3 = 0
VEL A 0.5
DEL 100
```

Macro KEYPB4.txt

This macro will exit if digital input 4 (keyPB 4) is not active, otherwise the velocity is set to 1.5. The short delay prevents this macro being called too rapidly when the button is pressed.

```
MEX DIO? 4 = 0
VEL A 1.5
DEL 100
```

When you combine commands for multiple controllers in those macros, it is also possible to connect a pushbutton control device to one controller but use it to command another controller. Example: Connect a pushbutton box to a Mercury class controller and use it to control a C-843 motor controller card in the PC.

2.5.21 Host Macro Quick Start

The ***Host Macro Quick Start*** window is accessible from the PIMikroMove main window via a toolbar button and via the ***Tools*** menu.

The ***Host Macro Quick Start*** functionality offers quick access to host macros:

- You can select and start a host macro without the need to switch to the ***Host macros*** tab card
- You can assign a customized “short-cut” button to a host macro so that this macro can be started by simply clicking the button

For successful macro execution, the affected axes (controllers) must be connected via the same interface(s) as used during macro creation (see "Controls on Host Macro Tab Card" (p. 41) for more information).

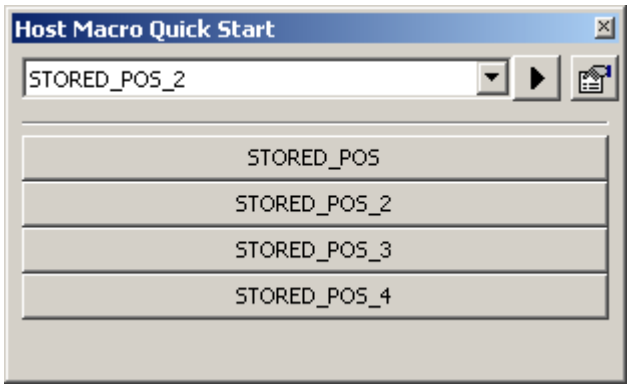


Figure 25: Host Macro Quick Start window with macro selection field and "short-cut" buttons

INFORMATION

The ***Host Macro Quick Start*** window can be docked to the borders of the main window by dragging it with the left mouse button pressed (see "Main Window" (p. 9)).

The macro selection field lists the same macros as the ***Host macros*** tab card. If you want a host macro to be included in the list, load it on the ***Host macros*** tab card using the ***Open and show another macro from text file*** button in the icon bar. Macros created as described in "Create New Host Macro with Current Positions" (p. 62) are automatically added to the list. Execution of the selected macro can be started by clicking the  icon.

The "short-cut" buttons can be configured in a separate window which can be opened via the  icon.

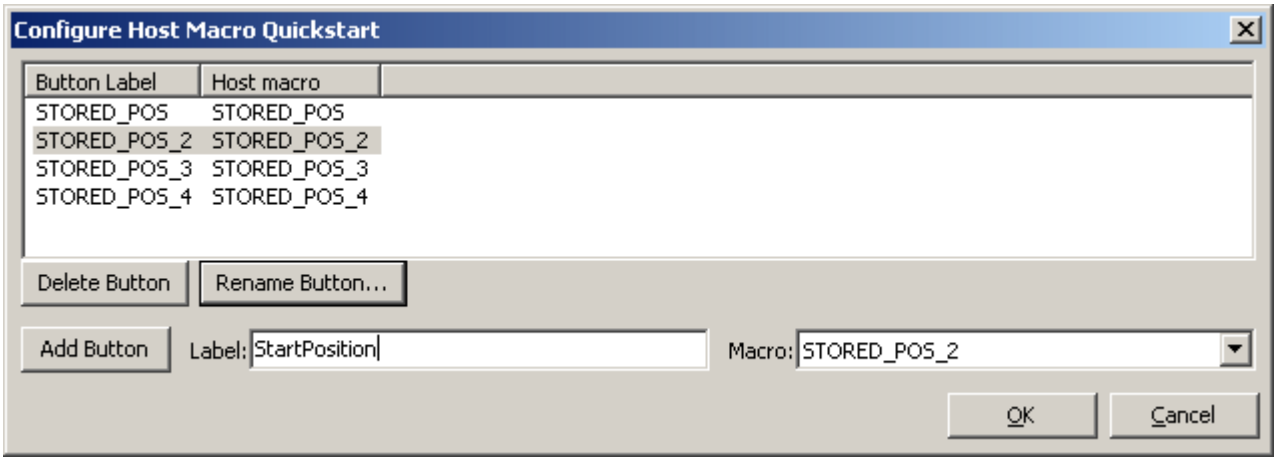


Figure 26: Window for configuration of the "short-cut" buttons used for host macro quickstart

The ***Configure Host Macro Quickstart*** window lists the existing "short-cut" buttons with their label and assigned macro. You can delete the selected button (the assigned macro is not deleted) or rename it. The button label can differ from the name of the assigned macro.

To create a new "short-cut" button, enter its label in the ***Label*** field, select the macro to be assigned in the ***Macro*** field, and click ***Add Button***. New buttons are also added when creating new host macros as described in "Create New Host Macro with Current Positions" (p. 62).

INFORMATION

It is not possible to change the macro assignment of an existing button. If you want to assign another macro to a button, delete this button and create it again with the new macro selected.

2.5.22 Create New Host Macro with Current Positions

The ***Create new host macro with current position*** window is accessible via a main-window toolbar button and the individual ***Axis*** menus. The functionality offered by this window is useful whenever it is necessary to recall certain positions for selected axes.

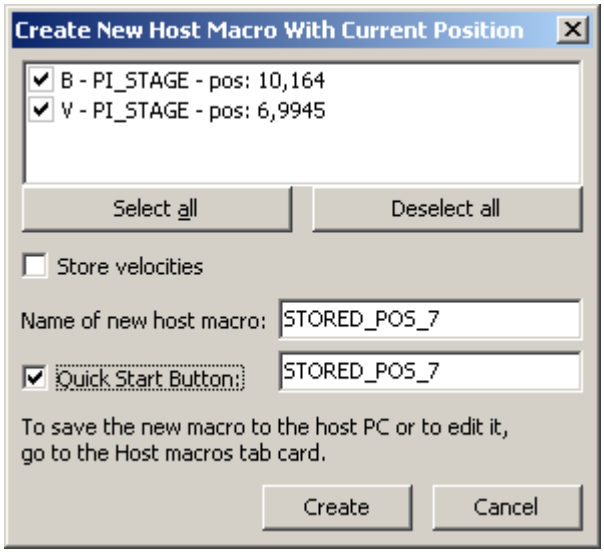


Figure 27: The new host macro will move the axes to the positions shown in the topmost pane

INFORMATION

This window is only available when at least one of the connected axes has servo on. Axes equipped with incremental sensors must be in the referenced state to appear in this window.

The topmost pane of the window shows the available axes with their current positions. These are the position values that will be used in the host macro to be created (position changes after window opening are ignored). Using the checkboxes, you can select the axes to be included in the host macro.

If supported by the controller(s), the current velocity settings of the selected axes can be included in the host macro by checking the ***Store velocities*** box.

Enter the name for the host macro to be created in the corresponding field.

For easy access, a separate "short-cut" button will be assigned to the host macro if the ***Quick Start Button*** box is checked. In the field to the right of the checkbox, you can enter the button label which may differ from the host macro name. You can start the host macro at a later date by simply clicking the button as long as the axes (controllers) affected by the host macro are connected via the interface(s) used for the current connection. See "Host Macro Quick Start" (p. 60) for more information.

Clicking the ***Create*** button creates the new host macro with the necessary motion commands and—if selected—velocity commands. The macro is not saved to the host PC yet. To save it to the host PC or to edit it like any other host macro, go to the ***Host macros*** tab card. The new host macro is listed there with <not saved> as path information. See "Controls on Host Macro Tab Card" (p. 41) for details.

2.6 Controller Macros

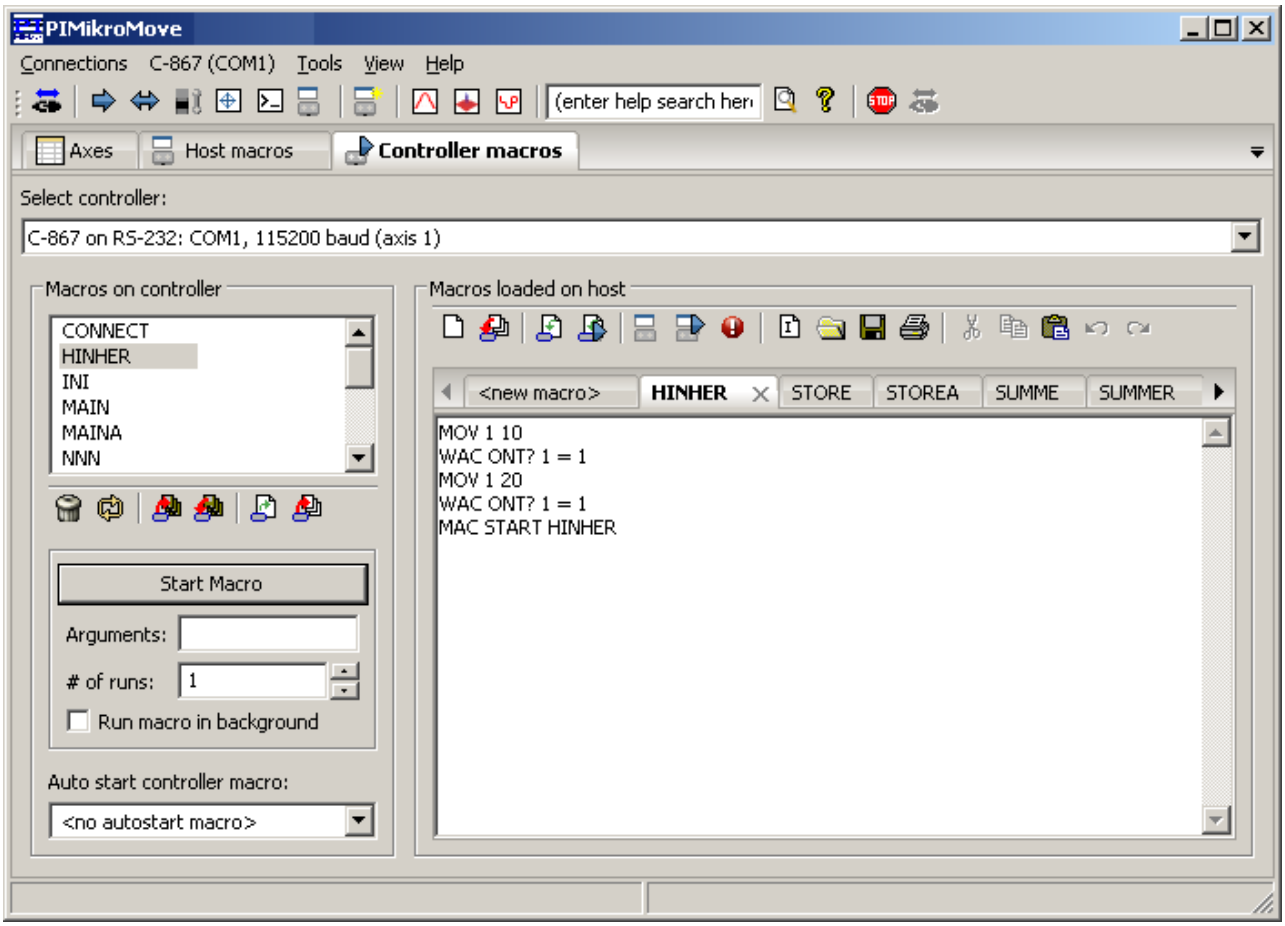


Figure 28: Controller macros tab card

Controller macros are sequences of commands which are stored directly on the controller. They can be executed independently without a host PC. The ***Controller*** macros tab card gives access to the complete macro functionality of all connected controllers which provide the macro feature. You can select the ***Controller macros*** card by clicking on its tab or using the arrow button in the tab bar.

In the ***Select controller*** field, choose the controller whose macros are to be displayed.

The left side of the ***Controller macros*** card has a list with the macros stored on the controller. The icons below the macro list have the following functions:

- displays the names of all macros stored on the controller
- deletes the selected macro(s) on the controller

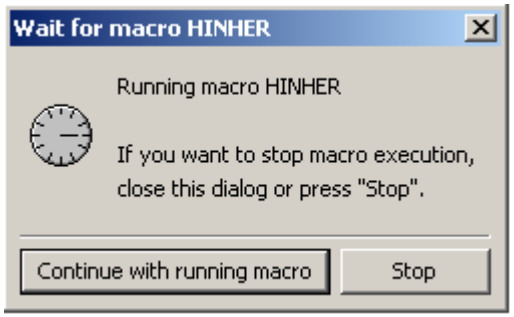
-  saves the selected macro(s) as text file(s) to the host PC
-  sends macro(s) stored on the host PC to the controller
-  loads the selected macro from the controller
-  loads all selected macros from the controller

You can start a macro (***Start Macro*** button) and specify that the macro command sequence will be executed repeatedly a certain number of times when the macro is run (***# of runs*** field). Provided that the controller supports variables, enter the values of the local variables used by the macro to be started in the ***Arguments*** field. The values have to be separated by space characters. Macro execution may fail if the macro uses local variables but no entries are made in the ***Arguments*** field. See the User Manual of the controller for a detailed description of variables.

INFORMATION

Depending on the controller, the  icon is present in the toolbar of the right tab card pane. Using this icon you can report the first error which occurred during macro execution. The error report contains the name of the macro, the line in the macro where the error occurred, the error code and the erroneous command which was sent to the parser.

If the ***Run macro in background*** box is available (depends on controller) and checked, you can continue working with PIMikroMove and perform other tasks (e.g. start a host macro) while the macro is running on the controller. If this box is not checked, you can choose background processing of the macro via the ***Continue with running macro*** button in a separate dialog (see figure below).



In the bottom field on the left side, you can select one of the macros stored on the controller to be the start-up macro. This macro will be executed automatically every time the controller is powered on or rebooted.

The right pane of the ***Controller*** macros card contains a text field where the commands of a macro can be composed and changed. Toolbar buttons are available for

- Creating macro command sequences
- Saving macros to / loading them from the controller
- You can save / load either a single macro or multiple macros.
- Note that with controllers which require a separate saving procedure, the Send macro to controller and Send all open macros to controller buttons will save macros to volatile memory only, and the Save all changes to EEPROM button must be used afterwards to save them permanently.
- Load and run controller macros as host macros. That way, controller macros become also available on the Host macros tab card (p. 39) where they can be executed by controllers which do not provide the macro feature.
- Opening macros from / saving them to text files on the host PC.

The following keyboard shortcuts are available:

- Ctrl+S saves the current macro
- Ctrl+D sends the current macro to controller
- Ctrl+R starts the current macro; if the macro content has changed, the macro is sent to controller before it is started

Most of the controllers that can be connected to PIMikroMove automatically save all changes regarding macros to non-volatile memory. But some controllers require a separate, password-protected procedure to save changes permanently (e.g. if a controller macro was created, has been changed, deleted or (re-)defined as start-up macro). If this is the case for one of the connected controllers, the ***Save all changes to EEPROM*** button is present in the bottom left corner of the ***Controller macros*** tab card (not shown in the figure above).

INFORMATION

For the password required when using ***Save all changes to EEPROM***, see the WPA command description in the controller's User Manual.

Depending on the controller, ***Save all changes to EEPROM*** may save further settings to the controller's non-volatile memory. See the controller's User Manual for details.

To save macro content you typed in the text field, first use the ***Send macro to controller*** button and then the ***Save all changes to EEPROM*** button.

For further details regarding the macro functionality see the controller's User Manual. If the controller does not support the macro feature, you can use the host macro functionality of PIMikroMove instead. See "Host Macros" (p. 39) and "Whats the difference between "controller macros" and "host macros"?" (p. 192) for details.

3 Additional Common Windows

The windows described below are available for almost all controllers (exceptions noted below). They are accessible via the ***Tools*** menu and the ***View*** menu in the main window. Some can also be opened with a key combination or toolbar button.

Windows which are in any way controller-specific are accessible from the corresponding controller menu, see "Controller Specific Windows" (p. 101).

3.1 Move Multiple Axes

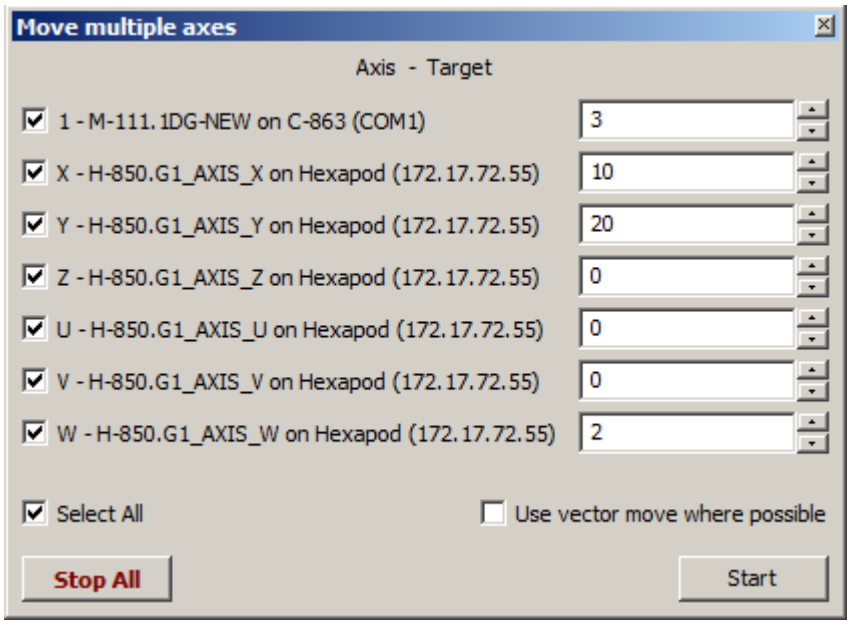


Figure 29: Move multiple axes

INFORMATION

This window is only available when at least one of the connected axes has servo on. Axes equipped with incremental sensors must be in the referenced state to appear in this window.

The ***Move multiple axes*** window allows starting and stopping a move of several axes by clicking a single button. You can configure which axes will be moved and set their targets. The velocities to be used for the axes can be set, for example, on the ***Axes tab*** card in the main window.

The start of the moves will not be synchronized if they are connected to different controllers. Axes connected to the same controller will be synchronized to the extent the controller supports synchronous moves.

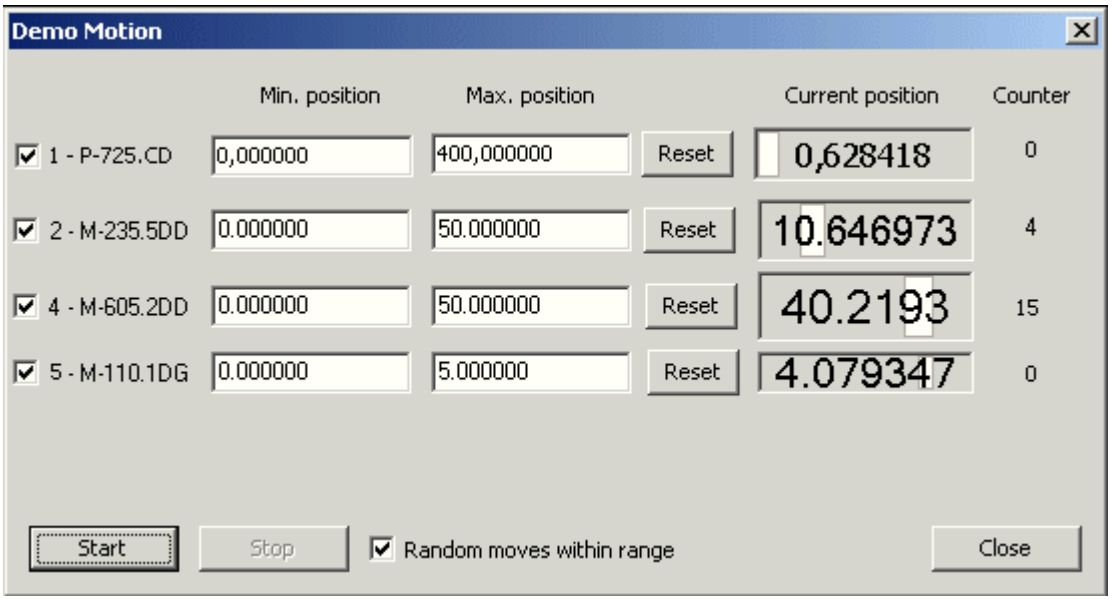
With controllers supporting simultaneous "vectorial" motion of coupled axes (MVE command), you can check ***Use vector move where possible***. The axes to be moved must then be connected to the same controller. If they are mounted in a way that they move perpendicular to each other, the combined motion of them will describe a linear path. This is achieved by appropriate calculation of accelerations, velocities and decelerations. The current settings for velocity, acceleration and deceleration define the maximum possible values, and the slowest axis determines the resulting velocities.

INFORMATION

The ***Move multiple axes*** window by default opens docked to the left border of the main window. You can undock and dock it again by dragging it with the left mouse button pressed (see "Main Window" (p. 9)).

3.2 Demo Motion…

Opens the ***Demo Motion*** window:



INFORMATION

This window is only available when at least one of the connected axes has servo on. Axes equipped with incremental sensors must be in the referenced state to appear in this window.

The motion range can be set in the Min. position and Max. position fields. Reset will reset the position limits to their default values.

INFORMATION

With Hexapod mechanics, the default position limits may be too large for your model type. Adapt the limits to the travel range values given for that model in the Hexapod User manual.

The white bar in the ***Current position*** field for each axis serves as a graphic representation of the axis' position. The ***Counter*** values give the number of target changes during the motion.

You can switch between two different motion modes:

- Random moves (within the given range)

- Full-range moves in alternating directions, i.e. back and forth between the specified minimum and maximum positions.

While the ***Demo Motion*** window is active, the ***Stop*** buttons in other windows are ineffective.

3.3 Stage Editor

INFORMATION

The ***PIStageEditor*** is only available when the PIStageEditor.exe file is installed on the PC.

The ***PIStageEditor*** is a tool that permits examining stage database files which are located on the host PC. The content of these stage database files is used by PIMikroMove e.g. for reading stage parameters like motion and travel range settings. Depending on the controller, the PIStageEditor.exe file is installed automatically with the setup procedure described in "Initial Installation" (p. 3) and described in a separate manual (SM144E). For stage database details see the subsections below.

3.3.1 Opening the PIStageEditor

You can open the ***PIStageEditor*** from the main window as follows:

- Via the ***Tools > Stage Editor*** menu sequence
- Via the ***Show PI Stage Editor*** icon in the toolbar

If multiple stage databases are available on the host PC for the connected controllers, you have to select one when opening the ***PIStageEditor***.

INFORMATION

Depending on the controller, you can also inspect parameters on the tab cards of the main window (p. 9), in the ***Single-Axis*** window (p. 95) or using the ***Device Parameter Configuration*** window (p. 147). See the User manual of the appropriate controller for more information.

3.3.2 Stage Database Files on the Host PC

If supported for the controller, the following stage databases can be located on the host PC:

- ***PIStages2.dat*** contains parameter sets for all standard stages from PI and is automatically installed on the host PC with the setup. It cannot be edited. You should check at regular intervals if a new version is provided by PI and install it on your host PC (see "Installing Updates" (p. 3)).
- If supported by the controller, ***Prefix_UserStages2.dat*** allows you to create and save your own stages (see "Editing User DAT Files" (p. 74)). This database is created the first time you connect stages in the host software (i.e. the first time the ***Select connected stages*** startup step is performed in PIMikroMove (p. 174)). *Prefix* depends on the GCS library used, e.g. if your controller uses the PI GCS 2 library, *Prefix* will be PI. There can be one file of this type for each different GCS library.
- ***M-xxx.dat*** and ***N-xxx.dat*** files contain parameter sets for custom stages delivered by PI. Those files have to be copied to the host PC according to the accompanying instructions. M-xxx.dat and N-xxx.dat files cannot be edited; should changes become necessary, you must obtain a new version from PI.

The PIStages2.dat, ***Prefix***UserStages2.dat and M-xxx.dat/N-xxx.dat databases are located in the ...\PI\GcsTranslator directory on the host PC. The location of the PI directory is that specified upon installation, usually in C:\Documents and Settings\All Users\Application Data (Windows XP) or C:\ProgramData (Windows Vista and Windows 7). You can use the ***Version Info*** item in the controller menu or the ***Search for controller software*** item in the ***Connections*** menu to identify the GcsTranslator path.

INFORMATION

For users which have already installed older versions of PIMikroMove and PIStageEditor:

The format of the stage parameter (DAT) files has changed (more parameters provided), realized by a file version change from 1 to 2. Note that PIStages and ***Prefix***Userstages DAT files with version 2 contain a "2" in their file name, e.g. PIStages2.dat (instead of PIStages.dat for version 1).

Existing ***Prefix***Userstages DAT files of version 1 are automatically converted to version 2 files the first time you connect stages in the host software (i.e. the first time the ***Select connected stages*** startup step is performed in PIMikroMove). The ***Edit user stages data…*** item in the controller menu opens the ***PIStageEditor*** with the version 2 file (***Prefix***UserStages2.dat). Parameters which were not present in version 1 are set to default values during conversion.

Version 4 and newer of the ***PIStageEditor*** supports stage parameter files of version 2 (with ***Help > About PiStageEditor*** you can check the version of the ***PIStageEditor***). If it is necessary to update the ***PIStageEditor***, run either the setup from the latest revision of the CD for your controller, or download the latest revision

of the ***PIStageEditor*** from the PI website. It can be found there in the same directory like the PIStages2.dat stage database. See "Installing Updates" for instructions and make sure to copy the PiStageEditor.exe to the directory where PIMikroMove was installed with the initial setup procedure. In PIMikroMove, you can use the ***Show version information*** item in the ***Help*** menu or the ***Search for controller software*** item in the ***Connections*** menu to identify the installation path.

3.3.3 Editing User DAT Files

To simply store the current, active parameters under a new stage name in a user-stages dat file on the host PC, use the ***Add/Edit User Stage Type...*** item on the controller menu (p. 13) or on the ***Axis*** menu (p. 34).

You can also use the ***PIStageEditor*** to view and edit entries in user-stages dat files.

When saving entries to user-stages dat files, do not use stage names which already exist in the PIStages2.dat database. If a stage of the same name exists in PIStages2.dat and the user-stages dat file, the parameter settings from PIStages2.dat will be preferred when assigning that stage to an axis (e.g. in the "Select Connected Stages" (p. 174) dialog), and the settings from the user-stages dat file will never be used.

3.4 Command Entry Window

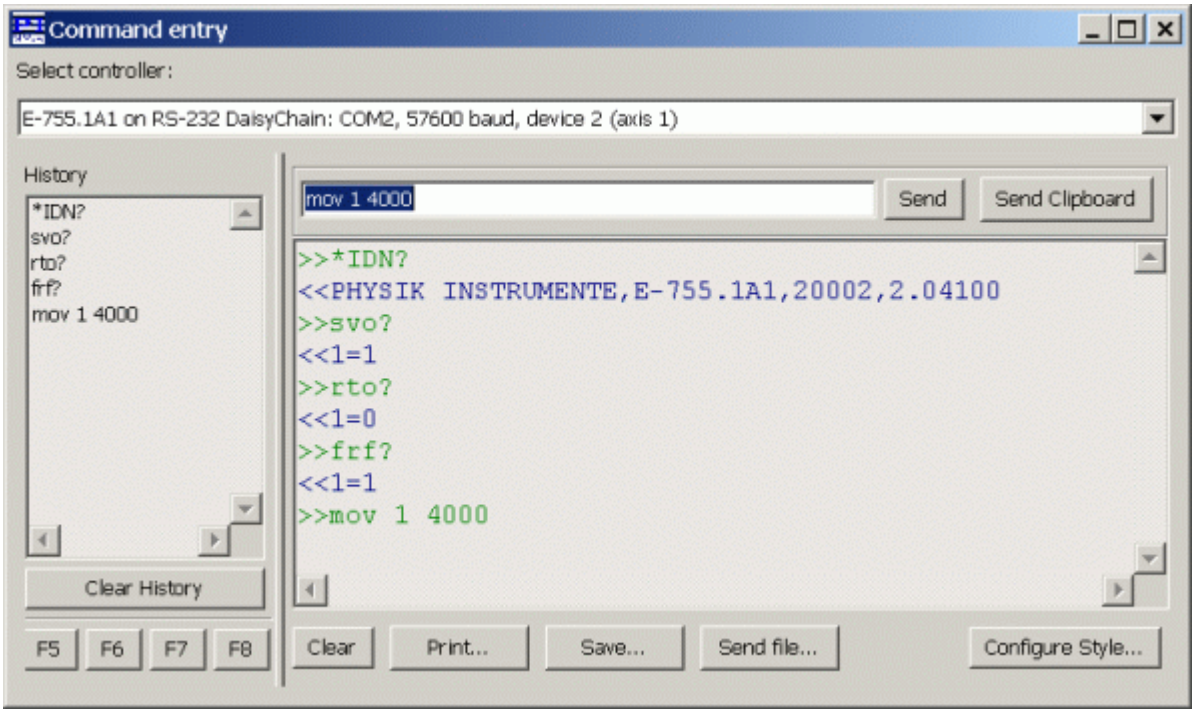


Figure 30: Command entry window

The ***Command entry*** screen gives you complete control over anything that is sent to the controller and shows all the controller's responses. For this reason, all other controls are disabled as long as the ***Command entry*** window is open. (See "Why is everything disabled when Command entry is active?" (p. 188))

Here you can address GCS commands to a controller or controller network without any background processing done by PIMikroMove (the commands are sent via the GCS DLL for the specific controller). For detailed descriptions of the commands, see the controllers User Manual, GCS DLL Manual or the GCS Commands Software Manual. In addition, the response to the HLP? command consists of a listing of all commands and their arguments. It is also possible to send single bytes. To do so, enter a "#" followed by the decimal value of the ASCII character. For example "#24" will send the byte with value 24 (^X), which is interpreted by most controllers to stop all axes and macros. If you need to send "#" as a character itself, type "##" or "#35".

INFORMATION

With controllers connected in a daisy chain, always use the "normal" syntax and not the special syntax with TargetID and SenderID which is described in the controllers User Manual. The GCS DLL used by PIMikroMove internally handles the TargetIDs and SenderIDs.

In the ***Select controller*** field, choose the controller or controller network which is to be commanded.

INFORMATION

Some networked controllers (e.g. daisy chained Mercury class controllers with native firmware or E-816s on an I2C bus) appear in PIMikroMove as a single multi-axis controller, while others (e.g. daisy chained E-755s) will have individual entries.

The left pane of the ***Command entry*** window shows the history of the commands typed in the ***Send*** field. When the cursor is placed in the ***Send*** text field, the contents of that field will be replaced by commands in the history list if you press the ⇑ or ⇓ arrow keys; pressing any other key returns you to the ***Send*** field where you can edit the command before sending. Four frequently used commands or command sequences are accessible with the F5, F6, F7 and F8 keys or buttons. To edit a sequence, hold down the control key while pressing the key or clicking the button.

The contents of the clipboard can be sent as-is with ***Send Clipboard***.

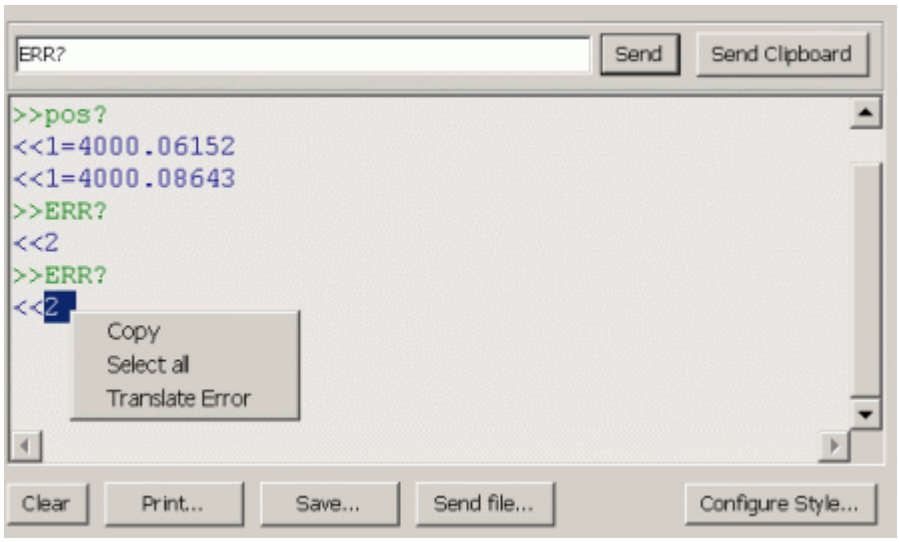


Figure 31: Display pane with error code context menu

The display pane below the ***Send*** field by default lists all sent and received messages. If you right-click in the display, a menu opens which allows to select the whole display content and copy it to the clipboard. Furthermore, the ***Translate Error*** menu item is available when you have marked a numeric character in the display. ***Translate Error*** is useful to obtain the detailed description which exists for every numeric response to an ERR? command.

You can modify the display appearance using a dialog which is opened with the ***Configure style*** button. For example, it is possible to switch off the display of the strings sent, so only received messages are shown. You can also change the font and color of sent and received messages and activate / deactivate prefixes, show the received messages in hexadecimal format or show control characters in received messages.

The display content can be deleted with the ***Clear*** button, sent to a printer with the ***Print*** button and saved to a file on the host PC using ***Save…***

With ***Send file…*** it is possible to load a command sequence from a text file on the host PC and send it to the controller.

INFORMATION

Before you use ***Send file …***, make sure that the command sequence will not cause damage to your system. The commands are sent as-is and immediately executed.

The commands sent are not shown in the window. Send ERR? to find out if anything went wrong.

3.5 Position Pad Window

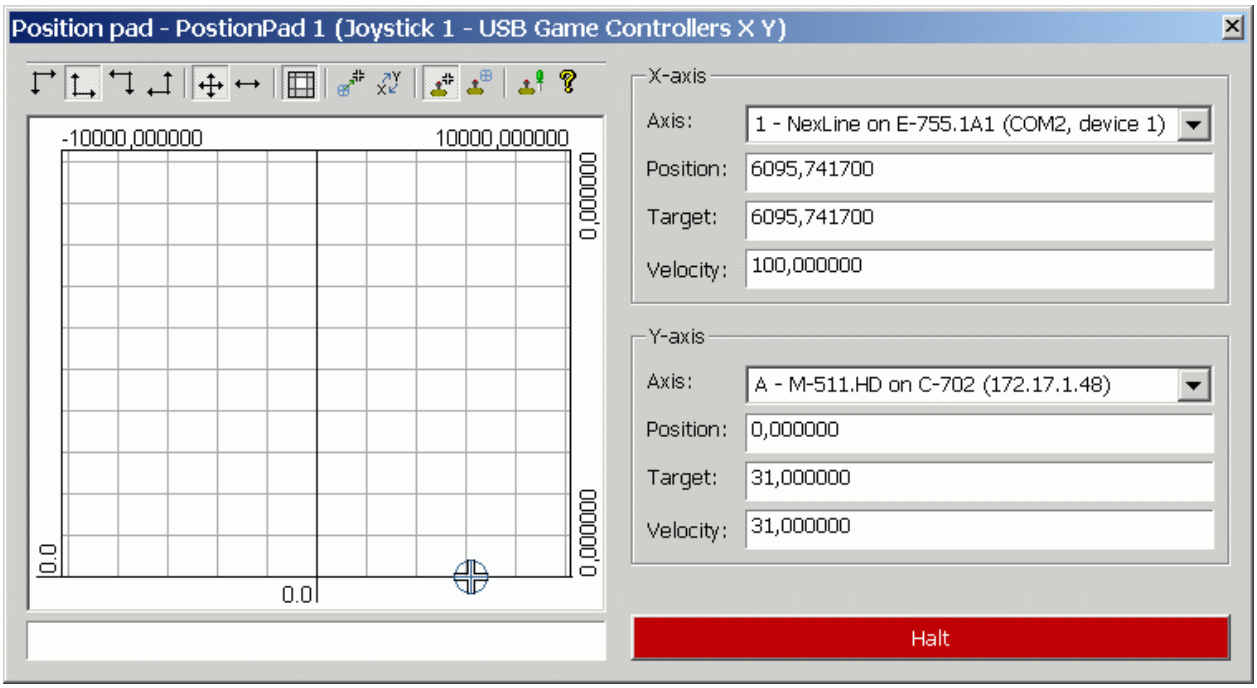


Figure 32: Position Pad

INFORMATION

This window is only available when the following conditions are fulfilled for at least one of the connected axes:

- Absolute positions can be commanded (servo must be on, and axes equipped with incremental sensors must be in the referenced state).
- The travel range is known.

With the position pad you can control and visualize the position of two axes. The black cross marks the target position and the blue cross the current position.

You can click or drag the target position with the mouse. If a joystick is connected to the PC, you can use it to move the axes by setting either the target or the velocity.

INFORMATION

Up to two joysticks can be used. Connect them to the PC ***before*** you start PIMikroMove. If the program is already running when you connect the joysticks, it must be restarted for correct joystick detection.

If joysticks are connected to the PC before PIMikroMove is started, multiple preset ***Position Pads*** will be available. Presetting means that the first two axes and buttons of every joystick are assigned to a separate ***Position Pad***. The default assignment of the joysticks and their axes and buttons to the ***Position Pads*** can be changed in a separate window which opens with ***Tools > Configure PC joysticks…*** In this window you can also create additional ***Position Pads*** which will give access to more joystick axes and buttons. See "Configure PC Joysticks Window" (p. 91) for details.

All available ***Position Pads*** can be open at the same time so that all connected joysticks can be used simultaneously.

INFORMATION

Make sure that the joystick is properly calibrated to avoid uncontrolled motion. See "Configure PC Joysticks Window" (p. 91) for details.

INFORMATION

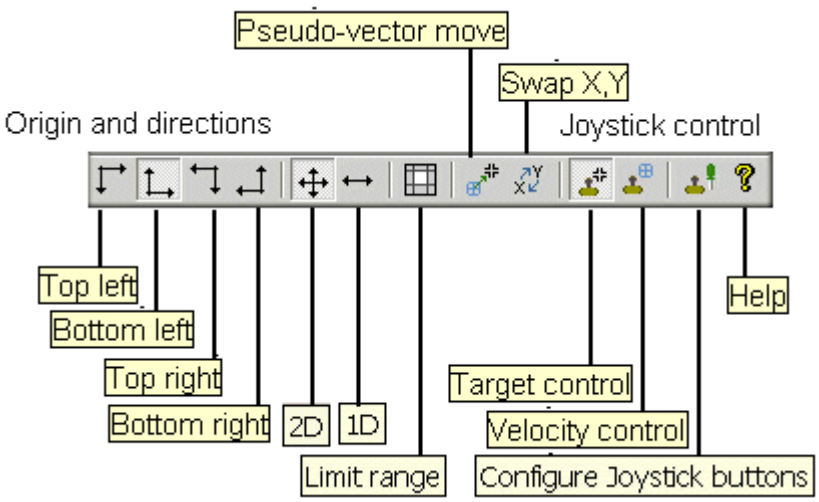
A joystick which is directly connected to the controller(s) cannot be used in the ***Position Pad*** window but with the ***Configure controller joystick(s)*** item from the corresponding controller menu (see "Configure Controller Joystick(s)" (p. 144)).

The left side of the ***Position Pad*** window shows the toolbar and the position grid. On the right side you can choose which two axes to control and see their position, target and velocity settings. To change the maximum velocity and the step size, use the corresponding single-axis windows or the Axes tab card (p. 28).

INFORMATION

A ***Position Pad*** window can be docked to the bottom border of the main window by dragging it with the left mouse button pressed (see "Main Window" (p. 9)).

3.5.1 Position Pad Toolbar



- The first four buttons are mutually exclusive and specify the orientation of the coordinate system.
- ***2D*** and ***1D*** allow switching between one or two active axes. With ***1D*** it is possible to control a single axis with the joystick x-axis without any effect on any other axis.
- ***Limit Range*** can limit the displayed range of the ***Position Pad***. The targets for velocity control are also set to these limited values.
- When ***Pseudo-vector moves*** is activated, the velocities are set so that both axes arrive at the target at approximately the same.
- With ***Swap X and Y axes*** you can interchange the display position of the axes.
- The ***Target control*** icon is only available when the operating system has recognized a joystick. If this icon is not available, you can move the target in the ***Position Pad*** by dragging it with the mouse (before or after the axes have reached it). When ***Target control*** is available and activated, you can set a target either by clicking with the mouse or by using a joystick which is connected to the PC.

  Details for target generation with the joystick: Every 100 ms the ***Position Pad*** refreshes the target as follows:

  new target = old target + step size * joystick displacement

  where

step size can be given in the corresponding single-axis windows or on the ***Axes*** tab card.

joystick displacement is given as a value between -1 and 1

- ***Velocity control*** is only available when the operating system has recognized a joystick. When ***Velocity control*** is activated, you use a joystick which is connected to the PC to set the current speed and direction. To change the maximum allowed velocities, use the corresponding single-axis windows or the ***Axes*** tab card. It is not possible to set targets with the mouse in ***Velocity control*** mode.
- ***Configure Joystick Buttons*** is only available when the operating system has recognized a joystick. This icon opens a dialog where you can configure the actions that will be performed when one of the joystick buttons is pressed.

3.5.2 Configure Joystick Buttons

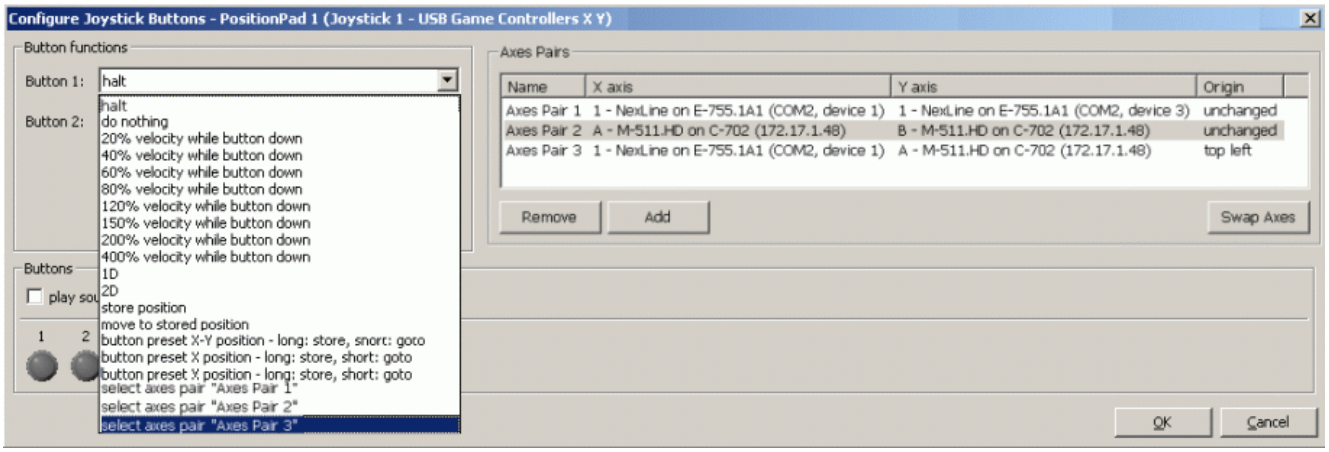


Figure 33: Configure Joystick Buttons dialog

You can select the action that is to be executed when one of the joystick buttons is pressed by selecting it from a predefined list in the ***Button functions*** pane:

- ***halt*** will stop both axes and disable any motion as long as the button is pressed
- ***do nothing*** ignores pressing the button
- ***nnn% velocity while button down*** sets the velocity to the specified percent of the current set velocity, but restores the old value when the button is released– only in velocity control mode!
- ***1D*** disables the Y-axis, only the X-axis will be active
- ***2D*** enable both, X- and Y-axis
- ***store position*** stores current position

- ***move to stored position*** moves to the last stored position
- ***button preset X-Y position—long: store, short: goto*** a long press stores the current positions for X and Y (if ***2D*** is enabled). The positions are confirmed by a short beep and the cursor will be drawn with a thicker outline for a short time. A short press will move to these stored positions.
- ***button preset X position—long: store, short: goto*** same as above, only X position is stored
- ***button preset Y position—long: store, short: goto*** same as above, only Y position is stored
- ***select axis pair …*** selects one of the user defined axes pairs which are listed in the ***Axis Pairs*** pane of the ***Configure Joystick Buttons*** dialog. With the buttons below the list you can administer the axis pairs.

By pressing the ***Add*** button in the ***Axis Pairs*** pane, you can define axis pairs which can be activated by the joystick buttons:

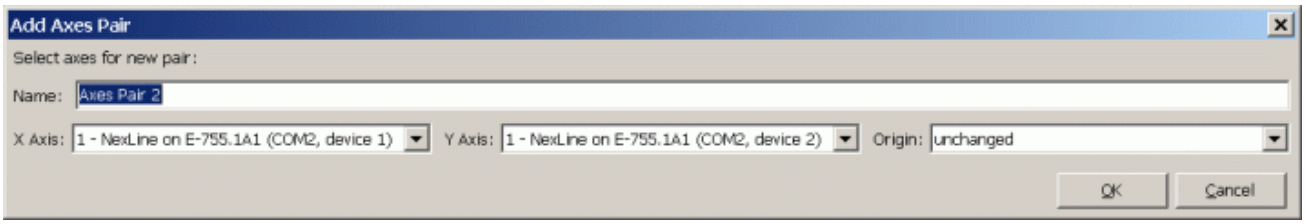


Figure 34: Add Axes Pair dialog

Each pair has information about the axes to be selected and the orientation of the ***Position Pad*** coordinate system. For example, you can define two pairs which have the same X- and Y-axis but different origins so that only the ***Position Pad*** orientation will change. If you set ***Origin*** to ***unchanged*** only the axes are activated and the orientation remains unchanged.

With the ***Swap Axes*** button, you can swap the assignment of the mechanics axes to the joystick X- and Y-axis. With ***Remove*** you can delete axis pairs from the list so that they can no longer be selected by the joystick buttons.

When you activate the ***play sound*** checkbox, a sound will be generated every time an action caused by a joystick button is performed.

3.6 Scan 1D Window

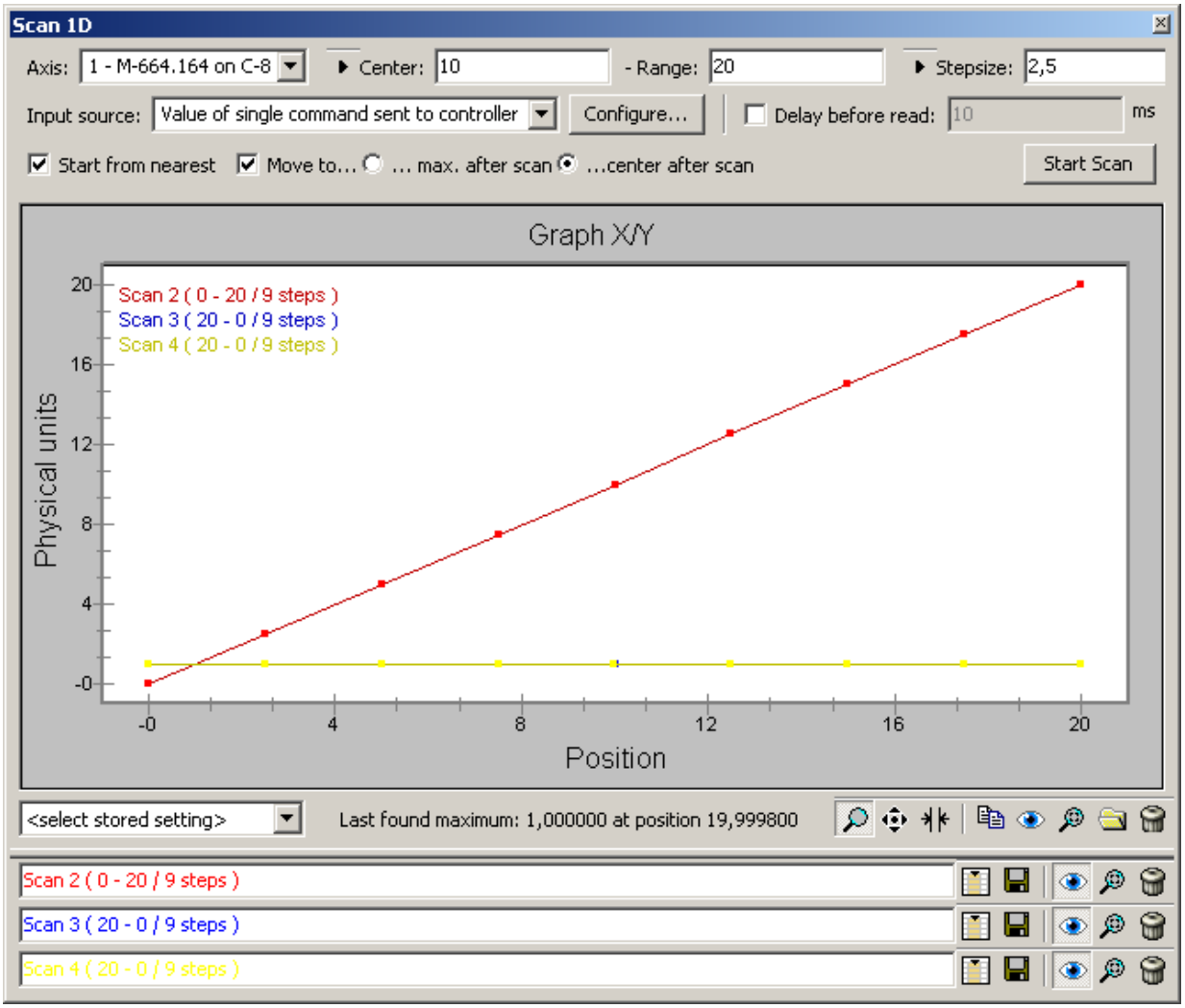


Figure 35: Scan 1D window

With the ***Scan 1D*** window you can measure an input source while moving an axis. Moved axis and input source need not be controlled by the same controller. So it will be necessary for the axis to stop after each step, since synchronization can not be assured otherwise.

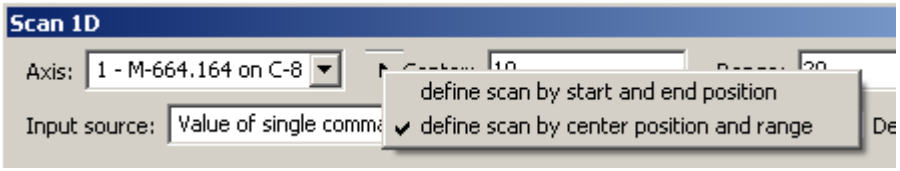
The measured data is visualized and can be saved to a file on the host PC.

In the ***Axis*** field you can select the axis to be moved, and in the ***Input source*** field the input source to be measured.Possible input sources are:

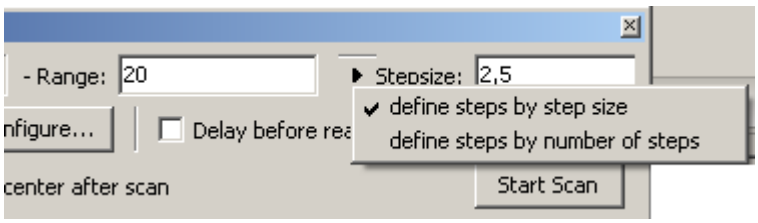
- Analog inputs

- Current positions of the connected axes
- Target positions of the connected axes
- Responses to query commands sent by PIMikroMove. To select a command response for the input source, use the ***Value of single command sent to controller…*** entry in the list of possible input sources. Then specify the command to be sent via the ***Configure…*** button next to the ***Input source*** field.

To define the scanning range, you can give its start and end positions or its center position and a range value.



The steps to be performed between the single measurement points can be defined by the step size or by the number of steps.



Furthermore, you have the following configuration options for motion and measurement:

- To reduce the time required to move to the first position, you check ***Start from nearest***. Note that if you repeat the process, the axis may then move back and forth.
- After scanning, you can move the axis back to the position where the input source took its maximum value if you check ***Move to max. after scan***. If the scanning range is defined via its center position and a range value, you can also move the axis back to the center position if you select ***Move to … center after scan***.
- If ***Delay before read*** is checked, the software will wait the specified time (in milliseconds) after the axis has completed one step and before a new value of the input source is measured.

You can save the current configuration settings or reload and delete stored settings via the selection field below the graphics pane.

To start scan motion and measurement, click the ***Start Scan*** button.

The central pane of the window shows the measured scan data in a 2D graph. Below this graphics pane, the individual scans are listed and can be administered separately. The list contains all scans which were performed successfully since you had opened the ***Scan 1D*** window. Every new scan is also added to the graphics pane and can be hidden or shown again via its list entry.

Below the graphics pane, there is a common graph toolbar which applies to all scan data:

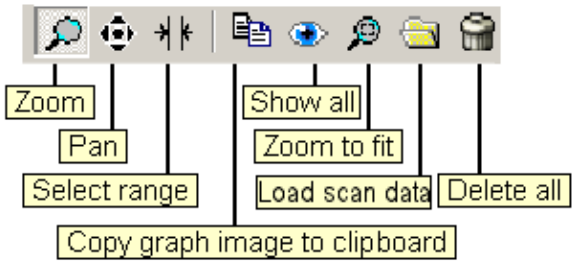


Figure 36: Scan 1D graph toolbar

- ***Zoom***: allows zooming of the displayed range. You can restore the range shown by clicking into the graphics display with the right mouse button or by clicking the ***Zoom to fit*** button.
- ***Pan***: move the displayed range with the left mouse button pressed
- ***Select Range***: select a new scanning range with the mouse
- ***Copy graph image to clipboard***: copy a bitmap of the display in the graphics pane to the clipboard
- ***Show All***: shows all scan data
- ***Zoom to fit***: zooms so that all data are completely shown
- ***Load scan data***: load scan data which was saved before in GCS ARRAY format (see the GCS Array Manual, SM146E, for details)
- ***Delete All***: delete all scan data

In addition to the common graph toolbar, every line of the scan list has its own toolbar for administering this scan data separately:

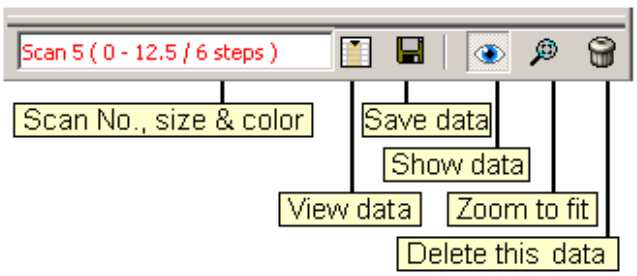


Figure 37: Scan list toolbar

- ***View data***: show a window displaying the array data for this scan, i.e. a table with positions and measured values
- ***Save data***: export data to disk as CSV (e.g. for Excel import) or GCS ARRAY (see the GCS Array Manual, SM146E, for details)
- ***Visible*** (Show data): show or hide the scan data from the 2D graphics display
- ***Zoom to fit***: zooms so that this data is completely shown
- ***Delete***: delete only this data

3.7 Scan 2D Window

With the ***Scan 2D*** window, you can measure an input source while moving two axes. Moved axes and input source need not be controlled by the same controller. So it will be necessary for the axes to stop after each step, since synchronization cannot be assured otherwise.

Possible input sources are:

- Analog inputs
- Current positions of the connected axes
- Target positions of the connected axes
- Responses to query commands sent by PIMikroMove.

Two tab cards in the ***Scan 2D*** window offer you the following scan functionality:

- ***Scan*** tab card: data is measured during the motion, visualized and can be saved to a file on the host PC. See "Scan Tab Card" (p. 87) for details.
- ***Auto Find*** tab card: data is measured during the motion, and the Auto Find procedure tries to find the maximum of the measured input source signal. See "Auto Find Tab Card" (p. 90) for details.

3.7.1 Scan Configuration

How to configure the scan motion to be started:

1. In the ***X axis*** and ***Y axis*** fields select the axes to be moved.
2. The range to be scanned is given by the position values in the ***Start*** and ***End*** fields. The size of the steps made between the single measurement points is given by the ***Stepsize*** field. Type suitable values in these fields.
3. Select the input source to be measured in the ***Input source*** field. To select a command response for the input source, use the ***Value of single command sent to controller…*** entry. Then specify the command to be sent via the ***Configure…*** button next to the ***Input source*** field.

4. Optionally: Check ***Delay before read***, if you want the software to wait the specified time (in milliseconds) after the two axes have completed their steps and before a new value of the input source is measured

When the scan motion is configured, select the ***Scan*** tab card or the ***Auto Find*** tab card and start the scan motion as described below in the corresponding section.

3.7.2 Scan Tab Card

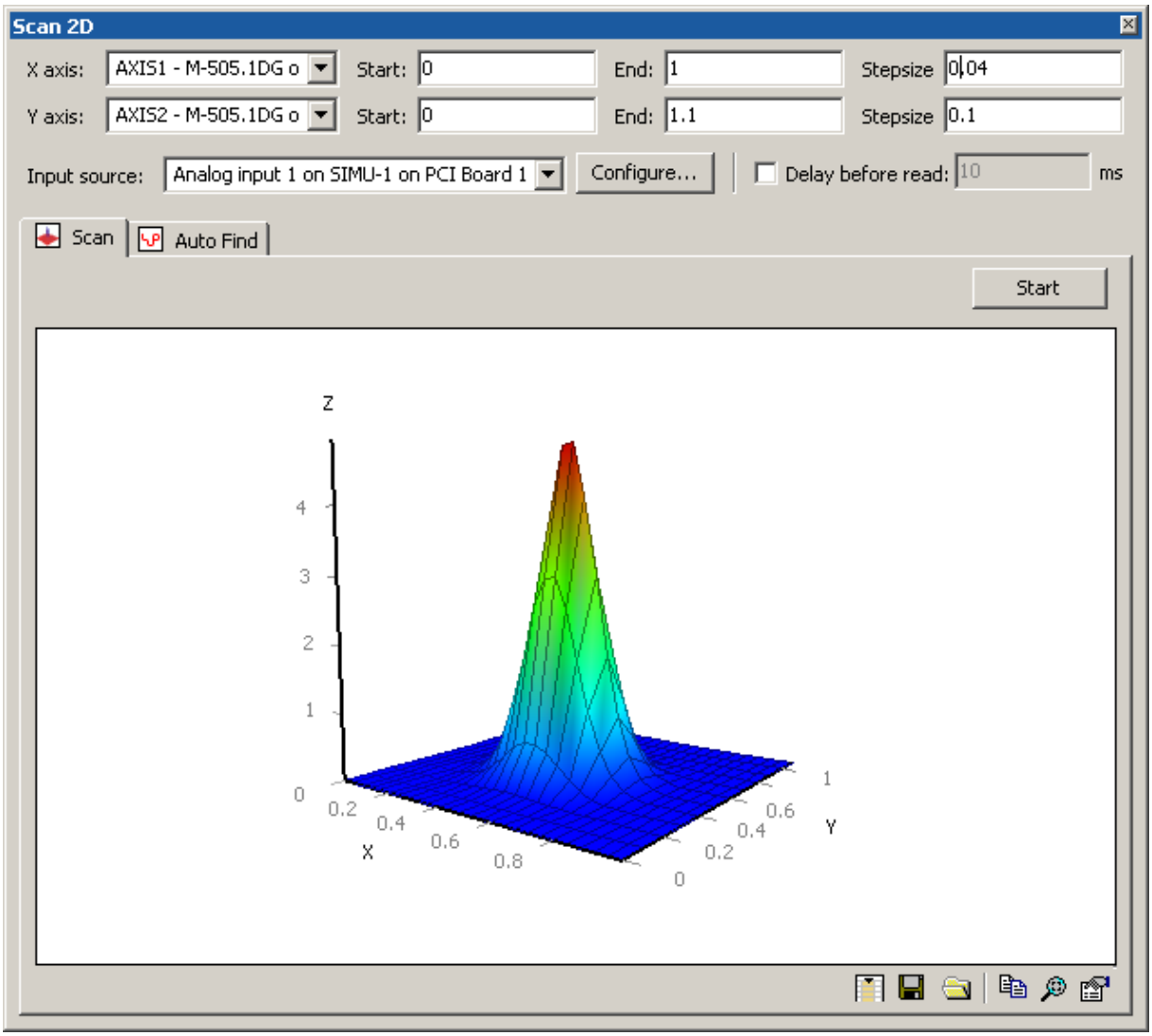


Figure 38: Scan tab card with 3D graph

If you start the scan from the ***Scan*** tab card, data is measured during the motion, visualized and can be saved to a file on the host PC.

Make sure that the scan motion configuration is correct (axes, input source, range and step values (p. 86)). To start scan motion and measurement, click the ***Start*** button.

The central pane of the ***Scan*** tab card shows the measured scan data in a 3D graph. Every new scan overwrites the content of the graphics pane.

You have the following options for mouse control in the graphics pane:

- Drag with left mouse button pressed: rotate view
- Drag with left mouse button + SHIFT pressed: shift view
- Drag with right mouse button pressed: zoom

Below the graphics pane, there is a toolbar with which you can administer the scan data:

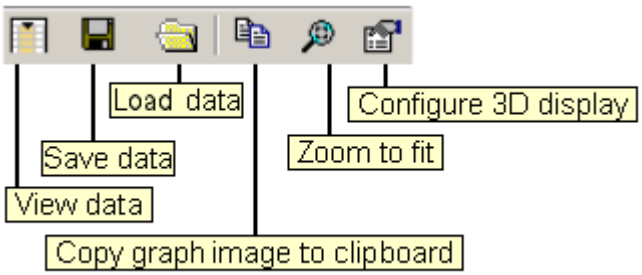


Figure 39: Scan data toolbar

- ***View data***: show a window displaying the array data for this scan, i.e. a table with positions and measured values
- ***Save data***: export data to disk as CSV (e.g. for Excel import) or GCS ARRAY (see the GCS Array Manual, SM146E, for details)
- ***Load data***: load scan data which was saved before in GCS ARRAY format
- ***Copy graph image to clipboard***: copy a bitmap of the display in the graphics pane to the clipboard
- ***Zoom to fit***: zooms so that the data is completely shown
- ***Configure display properties***: opens the ***Configure 3D Graph*** dialog, where several settings for the display can be changed

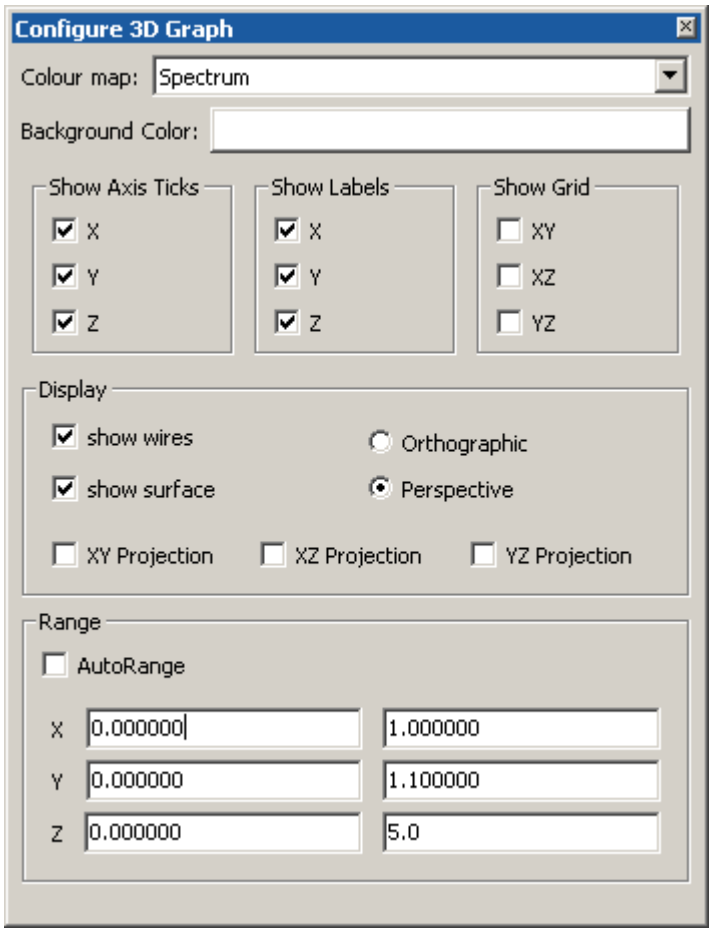


Figure 40: Configure 3D Graph window

3.7.3 Auto Find Tab Card

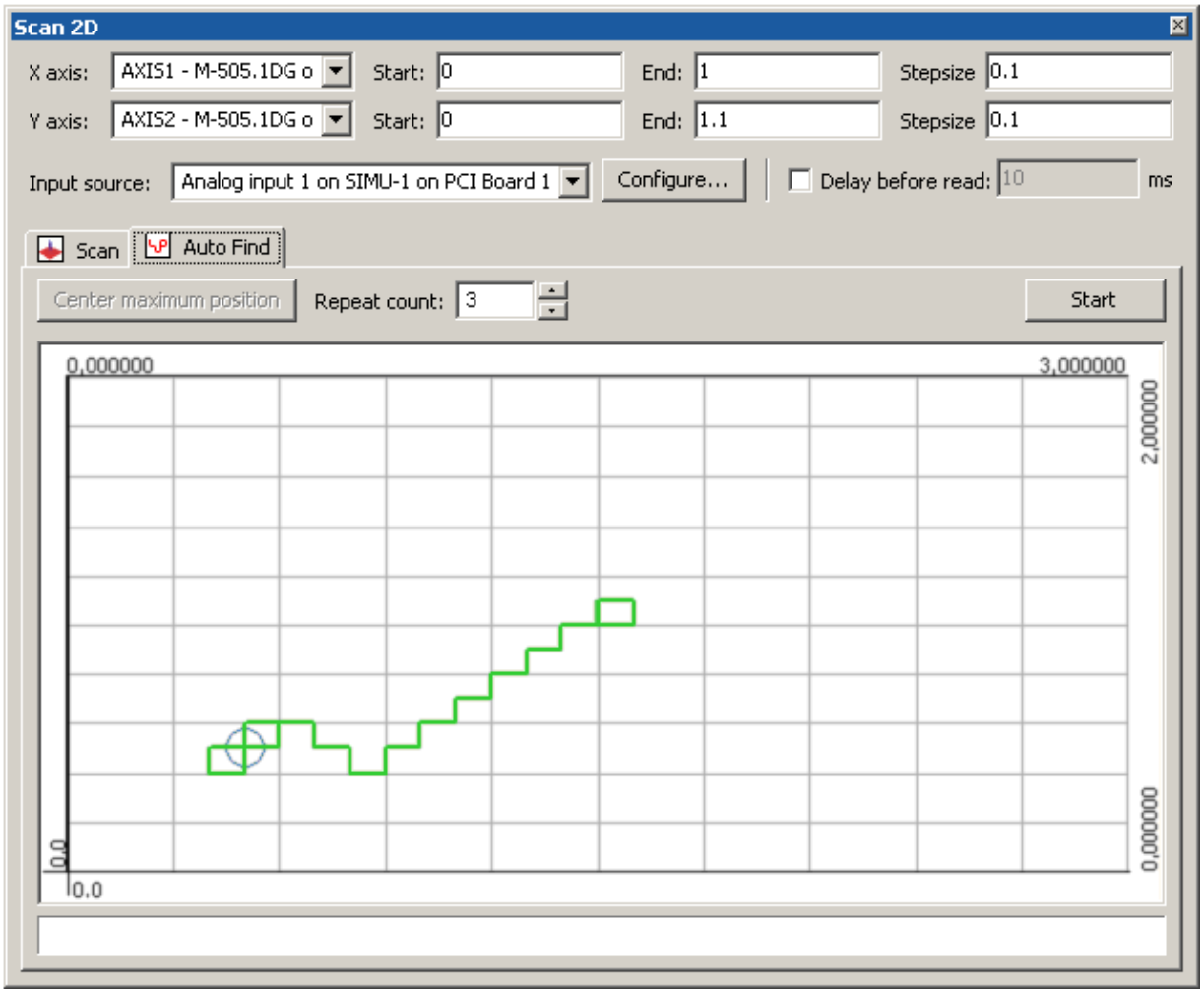


Figure 41: Auto Find tab card

If you start the scan from the ***Auto Find*** tab card, the Auto Find procedure tries to find the maximum of an intensity signal by modifying the position of two axes. The mechanics is moved one step and the measured intensity is compared with the last value. If the intensity decreases the next move will be in the opposite direction. If the mechanics reaches the same position a specified number of times, the Auto Find procedure will stop.

Make sure that the scan motion configuration is correct (axes, input source, range and step values (p. 86)). The Auto Find procedure will start in the middle of the defined range. In the ***Repeat count*** field, set the number of times the mechanics has to reach the same position before Auto Find stops. To start the Auto Find procedure, click the ***Start*** button.

The central pane of the ***Auto Find*** tab card shows the scan motion path in a 2D graph. Every new Auto Find procedure overwrites the content of this graphics pane.

3.8 Configure PC Joysticks Window

INFORMATION

This window is only available if at least one joystick is already connected to the PC at the start of PIMikroMove. Up to two joysticks can be used.

In the ***Configure PC Joysticks*** window, you can configure, test and calibrate joysticks connected to the PC and assign them to ***Position Pads***. Furthermore you can create, configure and remove ***Position Pads*** (see also "Position Pad Window" (p. 78)).

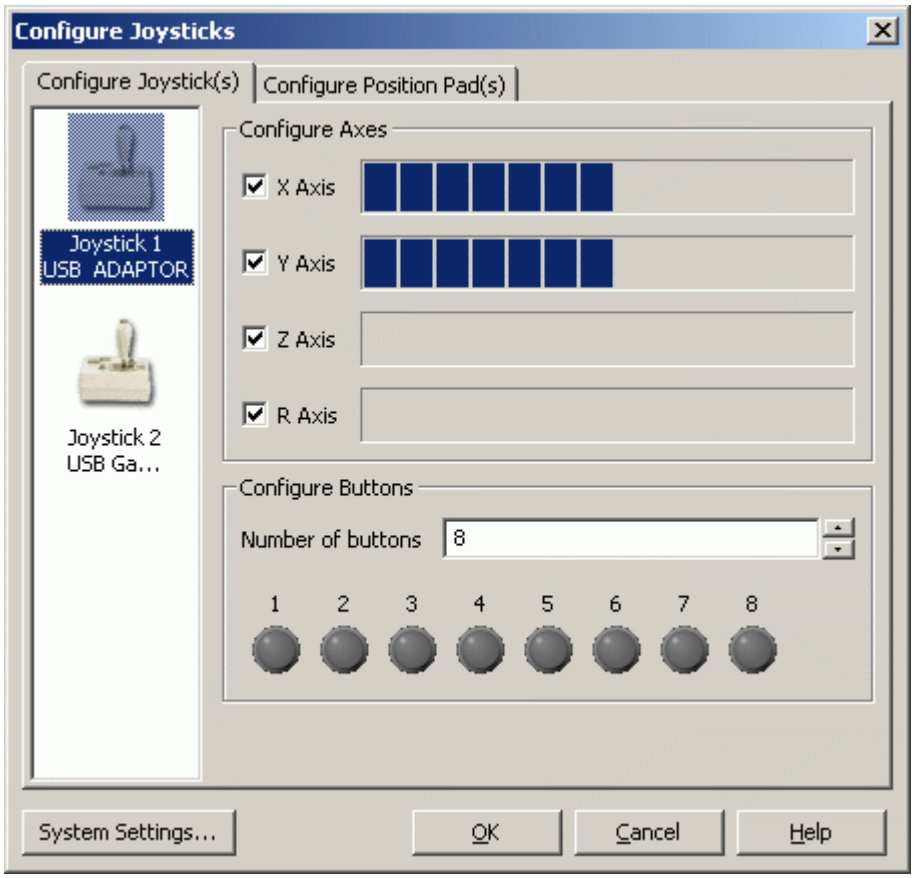


Figure 42: Configure Joystick(s) tab card

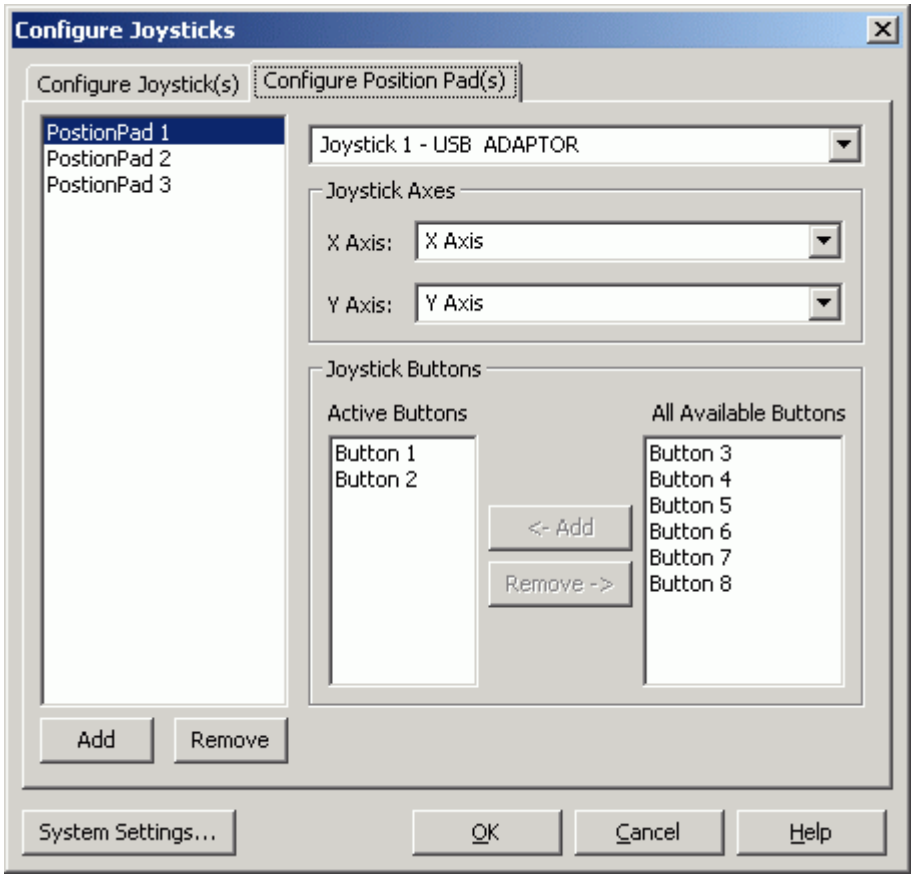


Figure 43: Configure Position Pad(s) tab card

Proceed as follows:

1. Connect up to two joysticks to the PC.
2. Start PIMikroMove. If the program was already running when you connected the joysticks, it must be restarted for correct joystick detection.
3. Open the ***Configure PC Joysticks*** window with ***Tools > Configure PC joysticks….***
4. On the ***Configure Joystick(s)*** tab card, all connected joysticks are listed in the leftmost column. Select the joystick you want to configure and indicate which of its axes and buttons are to be used.
5. Test and calibrate the joystick:
   Move the joystick(s) and press the buttons while watching the corresponding displays (bars and LEDs) on the ***Configure Joysticks*** tab card. If it should be necessary to calibrate the joystick, click the ***System Settings*** button to open the operating system's joystick control panel. Follow the calibration instructions there.

6. On the ***Configure Position Pad(s)*** tab card, all available ***Position Pads*** are listed in the leftmost column. Select the ***Position Pad*** you want to configure. Assign one of the available joysticks to this ***Position Pad***, assign axes of that joystick to the two axes of the ***Position Pad***, and select the joystick buttons to be used in the ***Position Pad***.

   A joystick can be used by multiple ***Position Pads***, but each of its axes and buttons can be assigned only once. Joystick axes and buttons already assigned to a ***Position Pad*** are no longer available for selection.

   The action to be started by a joystick button can be configured in the ***Position Pad*** window to which that button was assigned. See "Configure Joystick Buttons" (p. 81) for details.

   To create a new ***Position Pad***, click the ***Add*** button, enter a name for the new ***Position Pad*** and click ***OK***. It will then be added to the list and can be configured.

7. If you have finished the configuration of joysticks and ***Position Pads***, click ***OK*** to close the ***Configure PC Joysticks*** window. Now you can open the individual ***Position Pad*** windows via ***Tools > Position Pad*** and work there with the joysticks as described in "Position Pad Window" (p. 78).

3.9 Counts-per-Unit Calculator Window

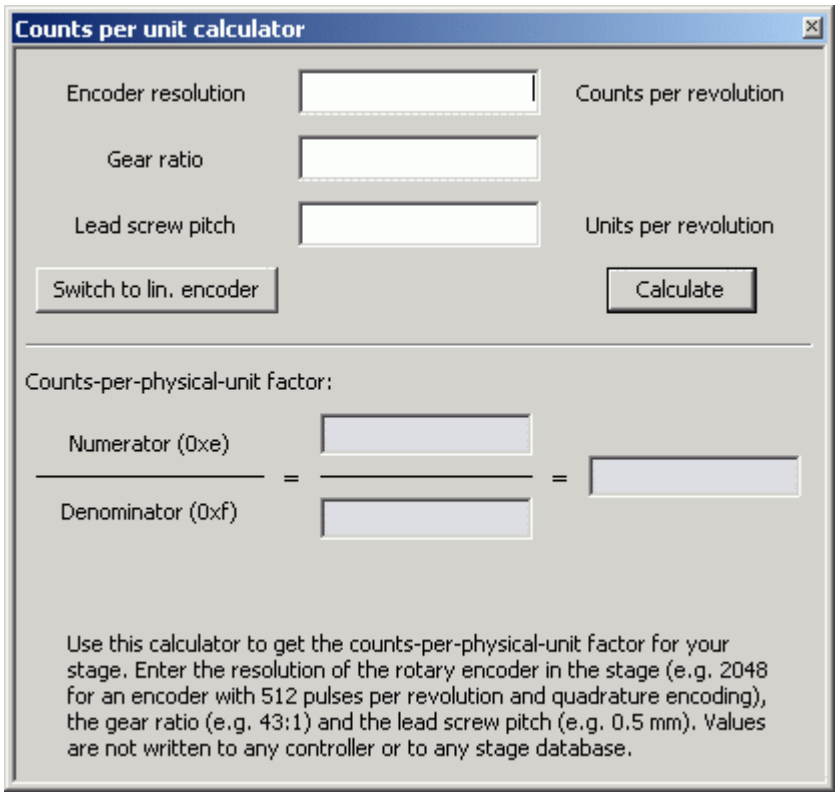


Figure 44: Calculation window for the counts-per-physical-units factor

This calculator window may be helpful when you start working with a stage that is equipped with incremental encoder and not part of any stage database yet. Using this window, you can calculate the values for the numerator and the denominator of the counts-per-physical-units factor (parameters 0xE and 0xF) for your stage. For stages with incremental sensors, the counts-per-physical-unit factor determines the unit of length to be used for all closed-loop motion commands (the controllers internally use counts for motion command processing).

INFORMATION

The values from the calculator window are not written to the controller or to any stage database. If you want to use this values, make a note of them. Then enter them on the ***Axes*** tab card (may be you have to display the appropriate columns first; see "Table Tab Cards" (p. 28)). To create an new stage database entry for your stage, you can use the the ***Add/Edit User Stage type…*** entry in the corresponding controller menu (p. 13).

When you change the numerator and denominator of the counts-per-physical-unit factor, all other parameters whose unit is based on the unit of length must be adapted too, e.g. closed-loop velocity and parameters regarding the travel range. Some controllers (e.g. C-867) adapt the appropriate parameters automatically.

3.10 Single-Axis Window

A ***Single-Axis*** window provides controls for a single axis.

As long as it is not expanded, a ***Single-Axis*** window can be docked to the left or right border of the main window (see "Main Window" (p. 9)) by dragging it with the left mouse button pressed. Docked ***Single-Axis*** windows cannot be expanded.

To open a ***Single-Axis*** window, use the ***Single Axis Window*** item in the ***View*** menu or the corresponding item of the ***Axis*** menu from the ***Axes*** tab card (p. 28).

3.10.1 Controls of the Single-Axis Window

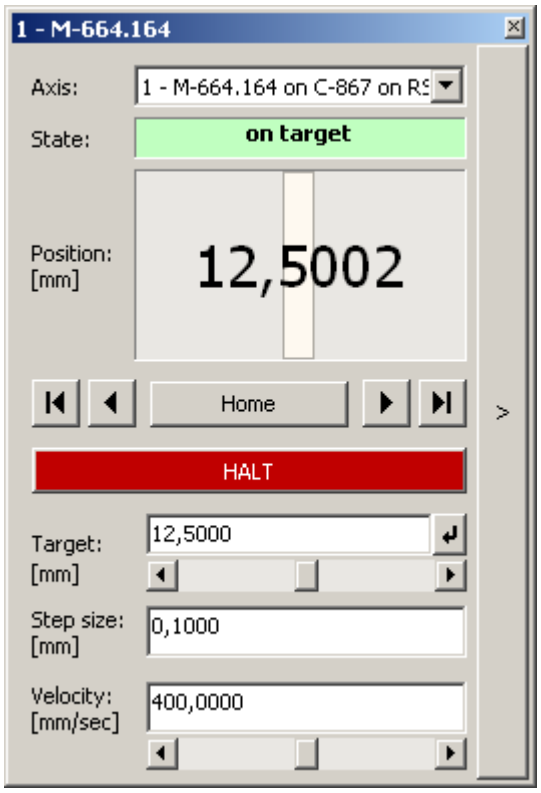


Figure 45: Single-Axis window

In the ***Single-Axis*** window you can enter target position, velocity and step size for closed-loop operation (servo on). If supported by the controller, a target value can also be set for open-loop operation (servo off). The target value in open-loop operation is to be interpreted as follows:

- ***OL Target***: is the open-loop target position for axes driven by PiezoWalk® drives (controller supports the OMA command)
- ***OL Value***: is an dimensionless open-loop value for axes driven by piezo actuators (controller supports SVA command). Depending on the controller it may correspond to a position or to a voltage (approximately).

If you right-click on any free area in the window you can see the ***Axis*** Menu (p. 34) for the axis displayed in the window.

INFORMATION

Single-Axis windows are not available for systems which do not support logical axes, e.g. E-755.101 controllers which can only command PiezoWalk® channels.

If supported for the controller, a ***Single-Axis*** window can be expanded by clicking the rightmost > button. To collapse an expanded window, click the < button between the leftmost and the center pane.

If expanded, the ***Single-Axis*** window shows two additional panes.

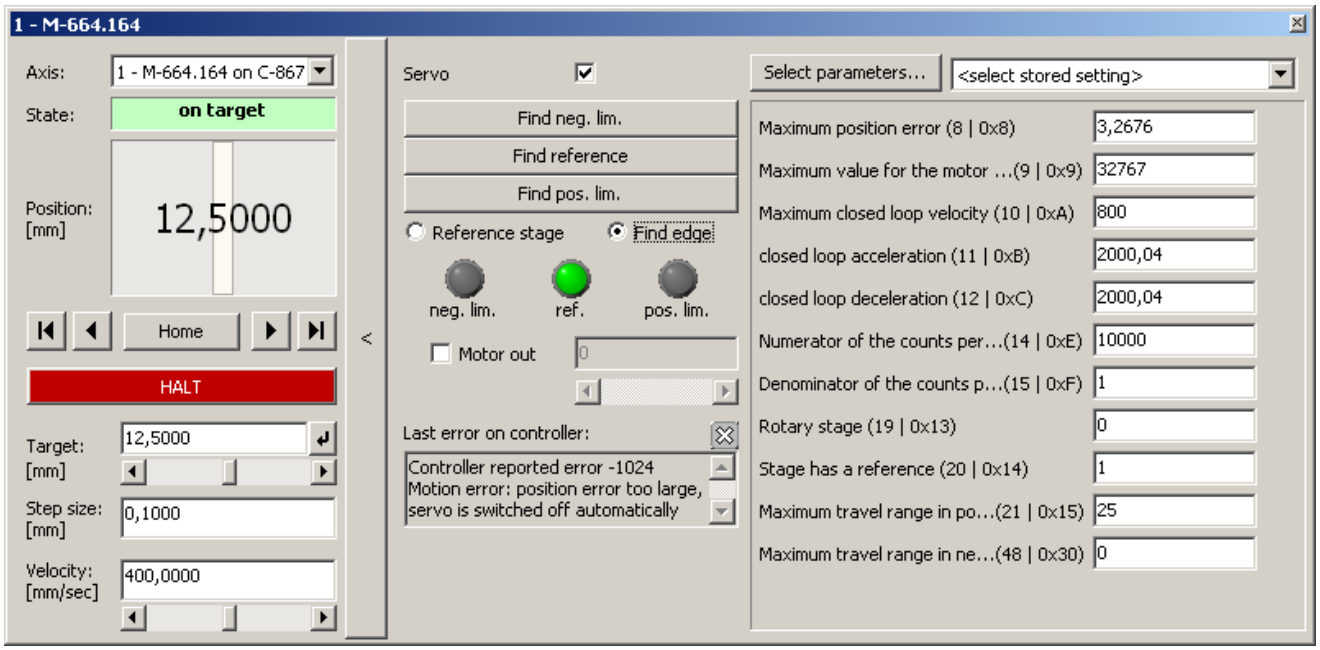


Figure 46: Expanded Single-Axis window – "Find Edge" is activated

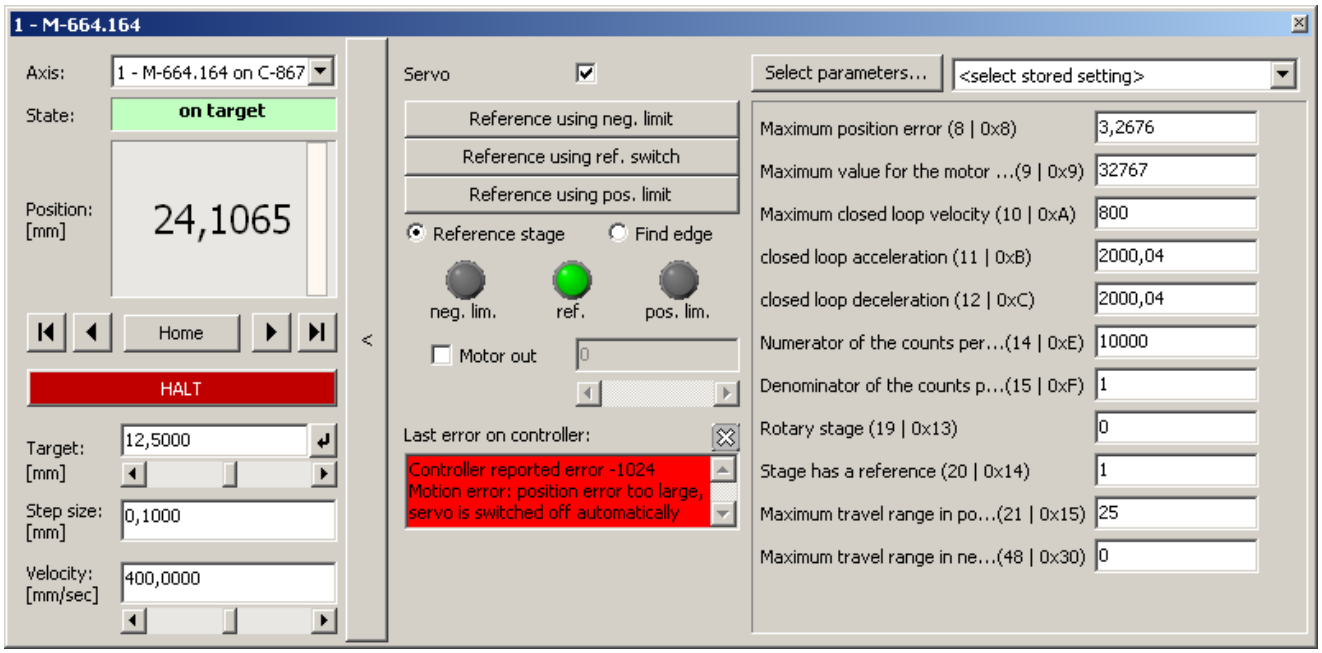


Figure 47: Expanded Single-Axis window – the error display is highlighted in red since an error occurred during the last second

In the center pane, you can

- Switch servo on/off using the appropriate checkbox
- Initiate a referencing procedure:
  If the ***Reference stage*** and ***Find Edge*** radio buttons are present (depends on controller), make sure that ***Reference stage*** is selected. To start the reference move, click one of the ***Reference using…*** buttons (see also "Referencing" (p. 179)). If the axis does not support referencing or a given type of referencing (e.g. has no reference switch), the respective button(s) will be dimmed.
- Initiate a “find edge” motion (availability depends on controller) which can help to detect or check the travel range of the mechanics:
  Select the ***Find Edge*** radio button. To start the “find edge” motion, click one of the ***Find …*** buttons. The axis then moves to the corresponding switch (limit or reference). The reference state of the axis remains unchanged.
- Check the state of limit and reference switches using the appropriate LEDs. If the axis is not equipped with such switches, the respective LEDs will be dimmed.

- Set the output voltage to the motor directly:
  Activate the ***Motor Out*** checkbox (servo is then switched off automatically).
  Enter a value in the field next to the checkbox or use the slider below the field.
  32767 is equivalent to 100% output voltage in positive direction and should cause a rapid move to the positive end of the travel range (appropriate value for negative direction is -32767).
  If the axis does not support setting the motor output directly, the checkbox and the appropriate controls will be dimmed (at present, only PILine® piezomotor stages and rotary stages support this feature).

NOTICE

Damage due to crashes!

Limit switches may not be effective when the motor ouput is set directly, so that the stage may run into the hard stop. Rapid motion may occur which might cause damage to the loads connected to the stage.

➢ Set the motor output directly only for testing purposes.

- Check the last occurred error in the corresponding display. After an error occurred, the display is highlighted in red for one second. Then the display color changes to gray. This recurs with every new error (the former error code is overwritten). You can clear the display using the button in its top right corner.
  Note that errors which are provoked by incorrect use of the controls in PIMikroMove will be displayed in separate message windows.

There may be some parameters of the stage or controller which have often to be checked or changed. For that purpose, the rightmost pane of the expanded ***Single Axis*** window provides a parameter list. See below for details.

3.10.2 Change Parameters in the Single-Axis Window

To check or change parameters of stage or controller, you can use the parameter list in the rightmost pane of the expanded ***Single-Axis*** window.

Configure the parameter list:

You can configure the content of the parameter list using the ***Select parameters…*** button which opens the same dialog as used for the column selection in the PIMikroMove main window (see "Select Columns to be displayed" (p. 29) for details). It is possible to save or load certain list configurations under user-defined names via the list box in the top right corner.

Edit parameter values:

Type new values in the parameter fields. As long as a value is shown in blue, it is only present in PIMikroMove but not yet sent to the controller. Press Enter on your keyboard to send the value to the controller's volatile memory.

Depending on the controller, parameters may be write protected by their command level setting. If a parameter field is grayed out, changing the parameter may require the command control level (CCL) 1. Use the ***Change command level (CCL)*** entry in the controller menu to open a dialog where you can select a command level and enter the corresponding password. The password for command control level 1 is "advanced". Note that command levels higher than 1 are reserved for PI service personnel.

Save parameter changes:

If stage databases on the host PC are supported for the controller, the ***Add/Edit User Stage type…*** functionality is accessible from the ***Single-Axis*** window. Using ***Add/Edit User Stage type...***, the current stage parameter values will be stored as a new stage type in the user-stages dat file. If the stage type already exists in the user stage database, its settings will be overwritten when the new ones are saved.

Proceed as follows:

1. Open the ***Axis*** menu (p. 34) by right-clicking in the center pane of the expanded ***Single-Axis*** window.
2. Select ***Add/Edit User Stage type...***
3. Enter the new stage type.
   For the stage type entry, do not use stage names which already exist in the PIStages2.dat database. If a stage of the same name exists in PIStages2.dat and the user-stages dat file, the parameter settings from PIStages2.dat will be preferred when assigning that stage to an axis (e.g. in the ***Select connected stages*** window (p. 174)), and the settings from the user-stages dat file will never be used.
   See "Stage Editor" (p. 72) for details on stage databases.

You can also save parameter changes using the following items of the controller menu (p. 13) in the PIMikroMove main window (availability depends on controller):

- ***Add/Edit User Stage type…***
- ***Save parameters to non-volatile memory***
- ***Parameter Configuration…***

3.11 Preferences…

Controls the behavior of the software user interface. In particular, the requirement for confirmation of various actions can be turned on or off. Note that the confirmation dialog boxes themselves have a checkbox to turn off future appearances of the dialog, but the ***Preferences...*** window is the only place to turn them back on.

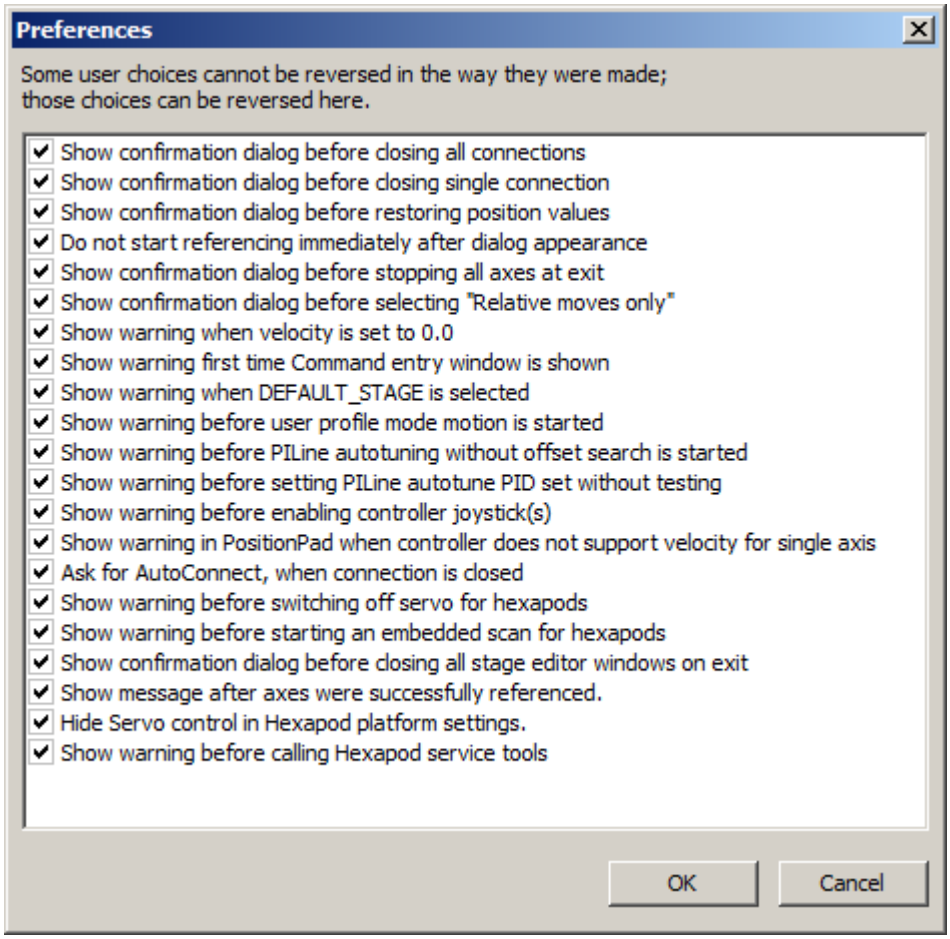


Figure 48: Preferences window

4 Controller Specific Windows

Windows which are in any way controller-specific are accessible from the corresponding controller menu.

Note that windows which are available for all controllers can be opened from the main window using the items of the ***Tools*** and ***View*** menus. Some can also be opened with a key combination or toolbar button. See "Additional Common Windows" (p. 69).

4.1 Profile Generator

The ***Profile Generator*** menu item is on the controller menu of supporting controllers (e.g. C-843). It gives access to some of the complex motion profile functions of supporting controllers (if the required controller-specific DLL is not present, this menu item will not appear).

With motion profiles it is possible to specify in more detail what happens while one or more axes move from their current position to a new one. When a special mode of the motion controller is used, a much finer time resolution is possible than by sending discrete move commands over an interface. With multi-axis systems, this also provides an easy way to specify motion along a specific trajectory.

A simple type of motion profile for a single axis can be defined by a set of time and position value pairs. In that definition, the axis takes the amount of time specified in the first pair to move from the first position to the position in the next pair, etc. To make the motion smoother, velocity specifications can be added, expanding the pairs to sets of 3 values. Profiles with up to 5 specified values in each set are supported: time, position, velocity, acceleration and jerk (the rate of change of acceleration). Interpolation between the specified values is performed by the motion control chip. With five-value data sets, a cubic spline interpolation is possible.

4.1.1 Axis Assignment

When the ***Profile generator*** item on the controller menu is selected, the ***Profile axes*** dialog box appears.

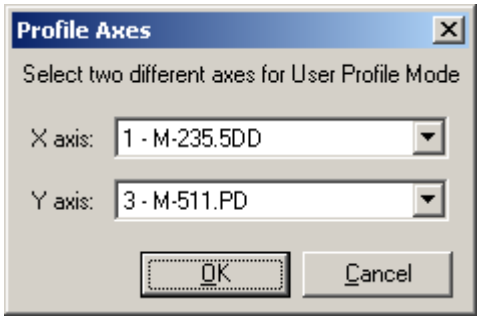


Figure 49: Axis assigment for User Profile Mode

In it, you specify two axes to be used in ***User Profile Mode*** and how to display the resulting profiles in the X-Y plane on the next screen. Only referenced axes with servo on will be available in the lists.

4.1.2 Profile Definitions for 2 Axes

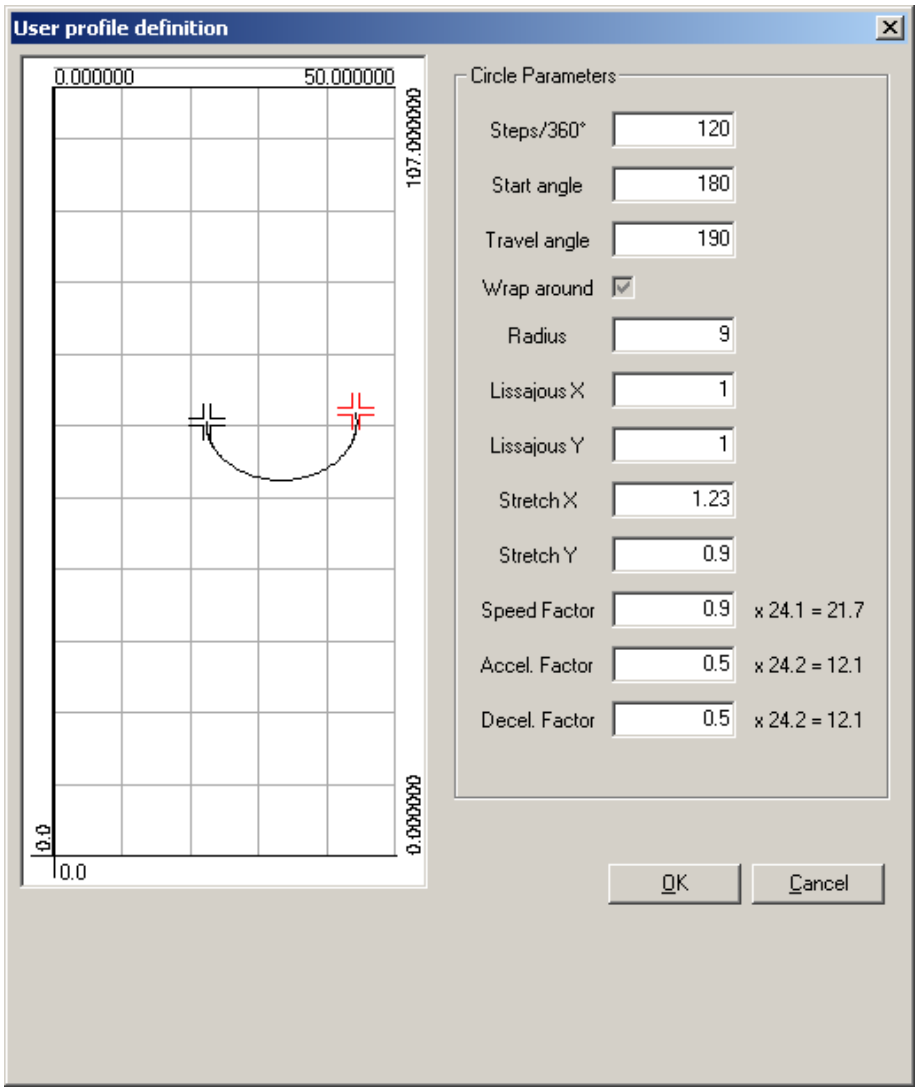


Figure 50: User profile definition window for circular-type motion

In the ***User profile definition*** window, sinewave motion profiles for two axes are defined. When run synchronously and visualized on an X-Y plane, the resulting motion forms a circle, ellipse or Lissajoux figure, depending on the user-specified parameters.

The graph at the left of the window is made from the time and position values in the profiles generated for the two axes. In the display the positions are shown connected by straight lines, but the motion controller does a cubic spline interpolation between them.

The user parameters to be entered have the following meanings:

- Steps/360 — for a true circle, the number of points in time per each 360 degrees of motion, for which the profiles contain explicit values for axis position, velocity, acceleration and jerk*. Interpolation is used for the motion between the specified positions.
- Start angle — for a true circle, the angular coordinate of the starting point*.
- Travel angle — for a true circle, the difference between the angular coordinate of the last position specified and the first position specified*.
- Wrap around — if activated, motion continues from the last to the first position specified.
- Radius — a linear scale factor applied on top of Stretch X & Y.
- Lissajous X, Y — if both equal, then an ellipse (or circle) is generated, otherwise a Lissajous figure with the corresponding frequency ratio
- Stretch X, Y — relative scaling in the X and Y directions (must be equal for a circle)
- Speed factor — is multiplied times the maximum velocity of the slower axis to give the speed at which to trace the figure, after start-up and before slow-down. Cannot be > 1
- Accel. factor — factor by which the maximum acceleration is further reduced. Cannot be > 1
- Decel. factor — factor by which the maximum deceleration is further reduced. Cannot be > 1
- OK — close the window and, after optional confirmation, start the motion (PIMikroMove will send the appropriate GCS commands to the controller).
- Cancel — the window is closed without any motion being performed.

*In the general case, the angular position is the phase angle of the corresponding point on one of the single-axis sinewave components, divided by that axis' Lissajoux factor.

4.2 PI Tuning Tool

The ***PI Tuning Tool***, available on the controller menu of supporting controllers, is designed to determine reasonable servo-control parameters for the different axes on the controller. While the axis moves stepwise, data like real-time positions are recorded and displayed. After a step the axis is automatically moved back to the starting position.

The parameters to be tuned and the recorded data available for display depend on the controller. For detailed descriptions of recommended tuning methods, the servo-algorithm used and the available record options see the User manual and the GCS Commands manual for the controller.

INFORMATION

When a step is performed, the ***PI Tuning Tool*** automatically switches the servo on for the axis. When you close the ***PI Tuning Tool***, the servo is set back to the state which was active before the start of the ***PI Tuning Tool***.

It is not necessary to reference an axis before the ***PI Tuning Tool*** is opened for that axis. With unreferenced axes, be sure to not drive the axis into its limits.

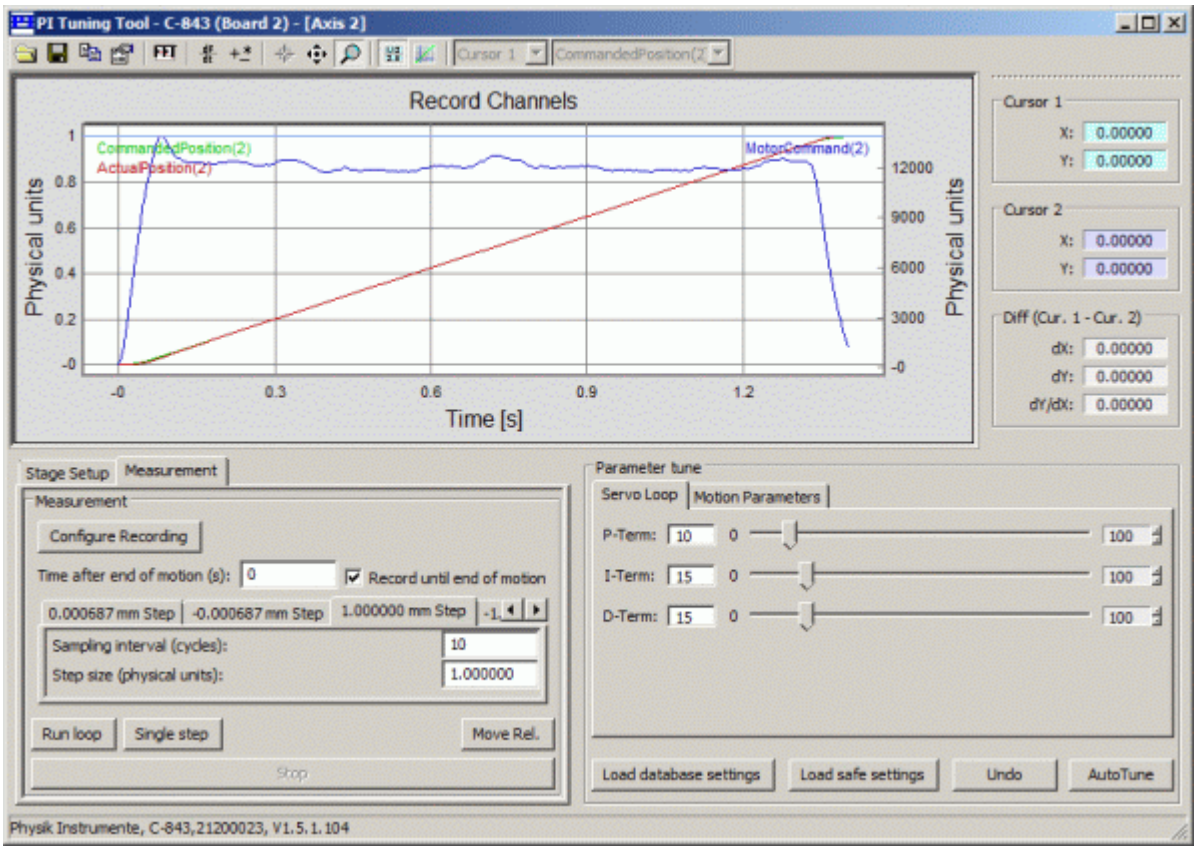


Figure 51: PITuningTool window for C-843 controllers

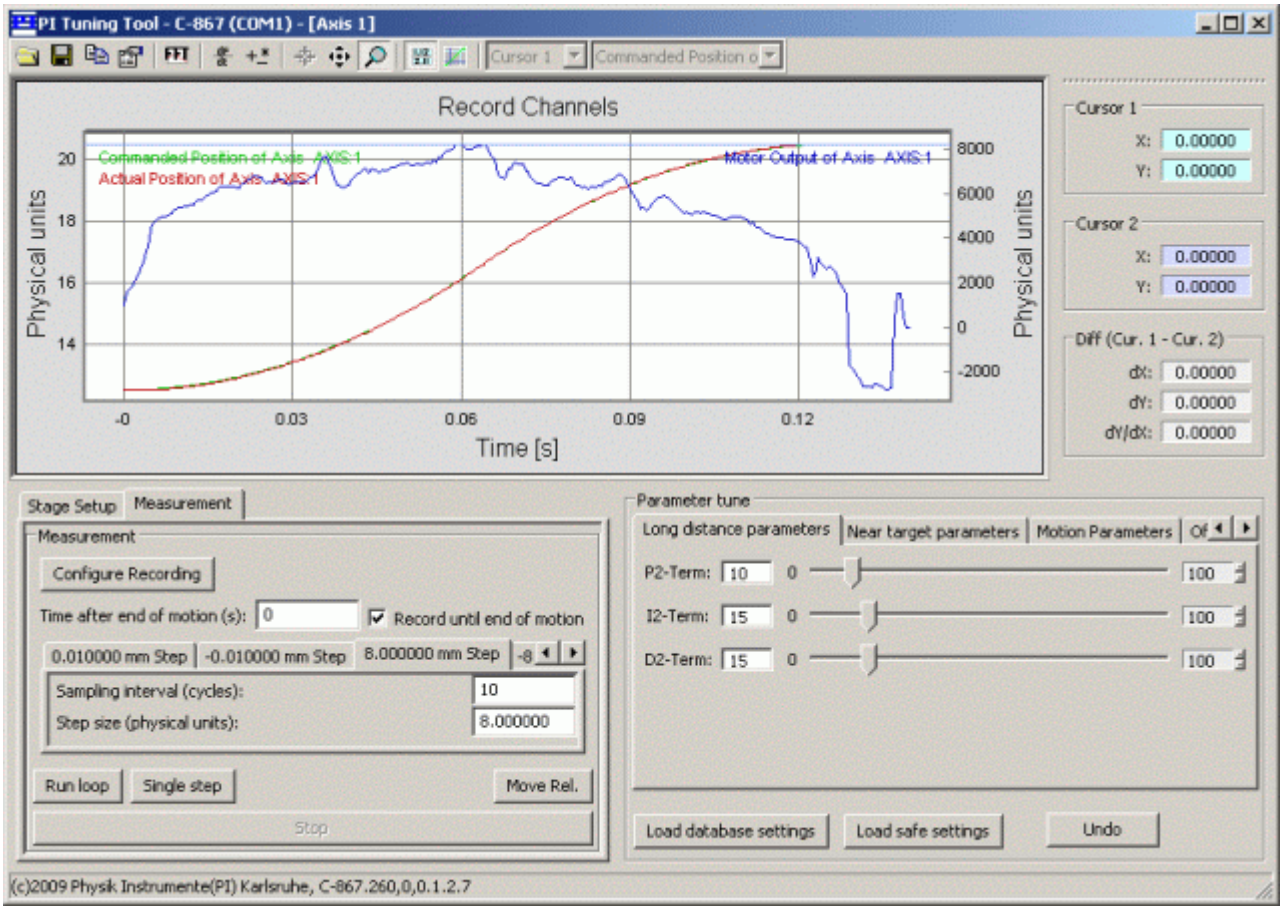


Figure 52: PITuningTool window for C-867 controllers

NOTICE

Damage due to oscillations!

The tuning procedure for the servo parameters may cause your stage to oscillate. Oscillations can cause damage to the stage and the load to be moved. The best results, however, are obtained when the stage has the same load during parameter tuning as in later operation.

- Make sure that the stage is properly secured, and remove any loads that might be harmed by that oscillation.
- If possible, use a dummy load.

4.2.1 Stage Setup Tab Card

On this tab card, you can select a stage type from the list (and thus load the appropriate parameter values). The current valid stage type for the axis is preselected. Use a suitable DEFAULT_STAGE entry for stages not listed or if you do not want to be subject to any restraints like limits, reference positions, etc. For more information regarding DEFAULT_STAGE entries see "How to use a stage that is not known to PIMikroMove" (p. 189).

Add/Save adds a new entry to the list of stage types or refreshes an existing entry. That way you can save the tuning results and make them also available in the list of stage database entries in the ***Select connected stages*** step (p. 174). The entries are saved automatically in the controller-specific user-stages dat file which is located in the \GCSTranslator directory.

4.2.2 Measurement Tab Card

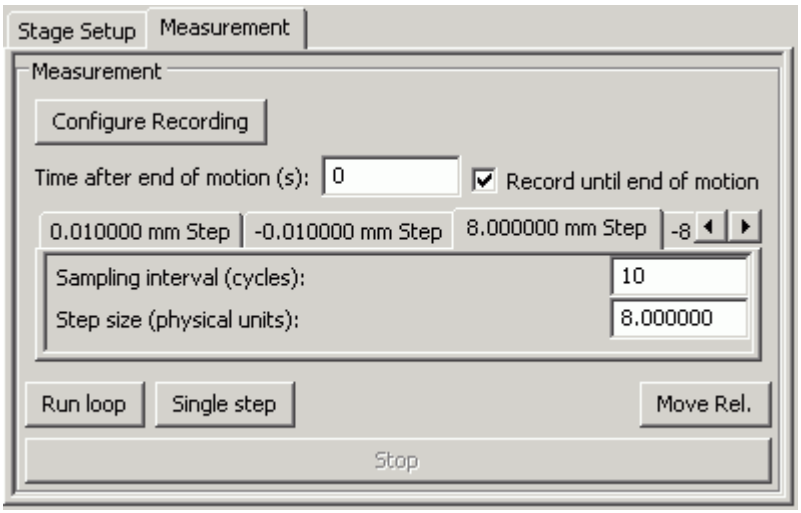


Figure 53: Measurement tab card for C-867 controllers

On this tab card, you can configure and start the measurement.

When you open the ***PI Tuning Tool***, a suitable data recorder configuration is preset. To change it, press the ***Configure Recording*** button. A dialog opens where you can make the same settings as in the configuration dialog described in "Configure Data Recorder" (p. 130), except of the trigger source setting.

INFORMATION

The display always shows the **last-recorded content** of the data recorder tables. Data source and record option for the data you want to see in the display must be configured before the recording is started.

Configuration changes subsequent to recording are possible for actions performed by PIMikroMove (e.g. calculation of derivatives, arithmetic operations, application of bitmasks or scaling, see “Configure Data Recorder” and ”Configure Graphics Pane” for details).

Configure the length of the recording using the corresponding input field and checkbox. Check the box to end recording automatically the specified amount of time after the motion has finished. When unchecked you can set the total time of recording.

The individual ***Step*** tab cards offer predefined settings for sampling interval and step size which are adapted to the current valid stage type. When changing the preset values keep in mind the following:

NOTICE

Damage due to crashes!

With an unreferenced axis, the ***PI Tuning Tool*** does not check if step size entries will exceed the travel range of the axis.

➢ Make sure that the given step size values will not drive the axis into its limits.

- ***Sampling interval*** defines the time in cycles between two data points stored in the controllers data recorder and displayed in the graphics pane afterwards. The larger the value, the longer the time period to be covered by the recording will be.
- If you are working with a DEFAULT_STAGE, the step size is in counts, otherwise it is in physical units (e.g. millimeter). A minus sign in the ***Step size*** line reverses the direction of motion. Adjust the value according to your specific application.

The ***Run loop*** and ***Single step*** buttons start motion and measurement. The motion range corresponds to the defined ***Step size***, and after each "forward" step the axis is moved back to the starting position. Only the measurement results of the "forward" motion are displayed, but not that of the motion back to the starting position. If ***Single step*** was used, the axis stops automatically after one "back and forth" motion cycle, while the continuous "back and forth" motion caused by the ***Run loop*** button must be stopped using the ***Stop*** button.

With the ***Move Rel.*** button you can move the axis by the ***Step size*** value without recording any data. After such a relative motion the axis does not move back to the starting position.

4.2.3 Parameter-Tune Pane

The individual tab cards of the ***Parameter tune*** pane give access to the parameters to be tuned. The available parameters depend on the servo-algorithm used by the controller. For detailed descriptions of the servo-algorithm and recommended tuning methods see the controller User Manual.

Parameters can be changed before a move is started, and also during the axis motion. This allows for direct reading of the effects of parameter adjustment. To change a parameter, move either the corresponding slider, or enter the value directly into the appropriate number field:

- If using the sliders: Once a slider is activated (click on it and you will see a dashed border around it) use the right/left keys for fine adjustment. Use the number fields to the right to define the slider ranges.
- If entering values directly: A value is not adopted until you press ***Enter***. If the value is outside of the slider range, the range limit will be set to this new value.

Load database settings retrieves the stage-specific values from the stage database (PIStages2.dat, *Prefix*_UserStages2.dat or M-xxx.dat/N-xxx.dat file). ***Load safe settings*** calls a safe parameter set stored in the ***PI Tuning Tool***. Undo restores the parameter values which were valid for the last performed step.

4.2.4 Graphics Pane and its Configuration

The display of recorded position and target data in the graphics pane is an indispensable feature for parameter tuning. You can configure the graphics pane via its toolbar. A short description will be displayed if the mouse cursor hovers over a toolbar button. For detailed descriptions of the toolbar buttons see "Configure Graphics Pane" (p. 134).

4.2.5 Autotune Option

Depending on the controller, the ***AutoTune*** button may be present in the bottom right corner of the PI Tuning Tool window. If so, automatic tuning is available as a password-protected option which can be purchased before or after initial delivery. Autotuning will calculate servo parameters automatically (P, I and D values).

NOTICE

Damage due to oscillations!

The tuning procedure for the servo parameters may cause your stage to oscillate. Oscillations can cause damage to the stage and the load to be moved. The best results, however, are obtained when the stage has the same load during parameter tuning as in later operation.

- ➢ Make sure that the stage is properly secured, and remove any loads that might be harmed by that oscillation.
- ➢ If possible, use a dummy load.

To use autotuning click the ***AutoTune*** button in the lower right of the ***PI Tuning Tool*** window and enter your password.

The parameters involved and the method used depend on the controller.

For C-843 controllers, the following dialog initiates the autotuning procedure:

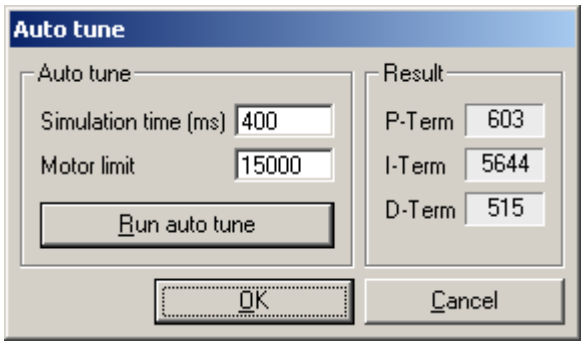


Figure 54: Autotune dialog for one axis on C-843 controller, including sample results

4.3 PI Wave Generator Tool

The ***Show/Hide wave generator*** menu item is on the controller menu of supporting controllers (e.g. E-753). It gives access to a separate ***PI Wave Generator*** Tool window where you can

- Create waveforms for later output by the wave generator(s) and save them to data files, see "Wave Table Editor Tab Card" (p. 113)
- Configure the data recorder, see "Data Recorder Tab Card" (p. 115)
- Use the Dynamic Digital Linearization (DDL) feature, a standard or optionally purchased password-protected feature which can be activated before or after initial delivery, see "DDL Settings Tab Card" (p. 117)

- Start the wave generator(s) with the selected waveform(s) and options, see "Wave Generator Tab Card" (p. 121)

A graphics pane is always available, either displaying the data related to the currently active tab card or data loaded from a file. With the toolbar on top of the ***PI Wave Generator*** Tool window, you can configure the graphics pane and import/export data to/from the display. A short description will be displayed if the mouse cursor hovers over a toolbar button. For detailed descriptions of the toolbar buttons see "Configure Graphics Pane" (p. 134).

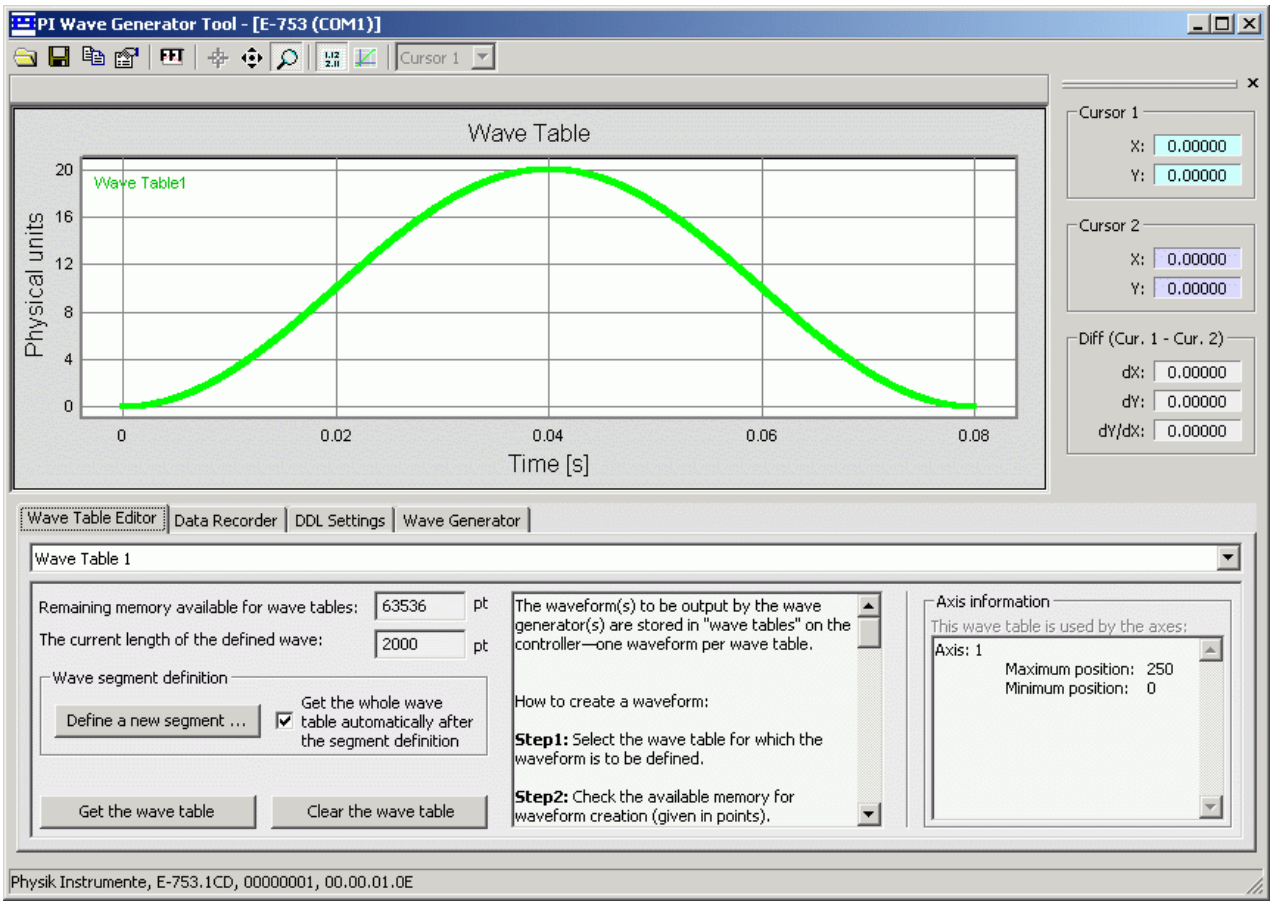


Figure 55: PI Wave Generator Tool window

See the controller User Manual for the controller-specific properties, e.g. for the number of wave tables, data recorder tables and DDL tables, the number of points per table, the assignment of wave tables to axes and wave generators and the assignment of DDL tables to axes.

4.3.1 Wave Table Editor Tab Card

The waveform(s) to be output by the wave generator(s) are stored in wave tables on the controller—one waveform per wave table. It is also possible to save waveforms to data files on the host PC and to load them back to the wave tables at a later time.

The available tables and wave types depend on your controller. See the controller User Manual for more information.

How to create a waveform:

1. Select the wave table in which the waveform is to be stored.
2. Check the maximum possible length of the waveform (given in points).
3. Create the waveform segment by segment. For every new segment, click ***Define a new segment…*** to open the separate definition dialog. See below for details.
   The order of the segments in a waveform can not be changed later. To change individual segments or modify their order, the complete waveform must be recreated segment by segment.
4. Check the waveform in the graphics display. To refresh the display, click ***Get the wave table***.
   It is not possible to select individual segments for display. Only the complete waveform can be displayed.
5. Optionally: Save the content of the graphics display (i.e. the waveform) to a data file on the host PC using the ***Export data*** icon in the toolbar.

To delete the complete wave table content, click ***Clear the wave table***. Segments cannot be deleted separately.

To assign wave tables to wave generator(s) and to start the wave generator output, go to the ***Wave Generator*** tab card (p. 121). The ***Axis information*** pane on the ***Wave Table Editor*** tab card shows the assignment of the current active wave table to the axes according to the settings on the ***Wave Generator*** tab card.

INFORMATION

The wave table content is lost when the controller is powered down or rebooted.

Waveforms which are supposed to run at the same time (each with a different wave generator) should have the same length. Otherwise all waveform output will be adapted to have the same cycle length—in most cases cut to the length of the shortest waveform currently running (depends on controller).

Define Wave Table Segment

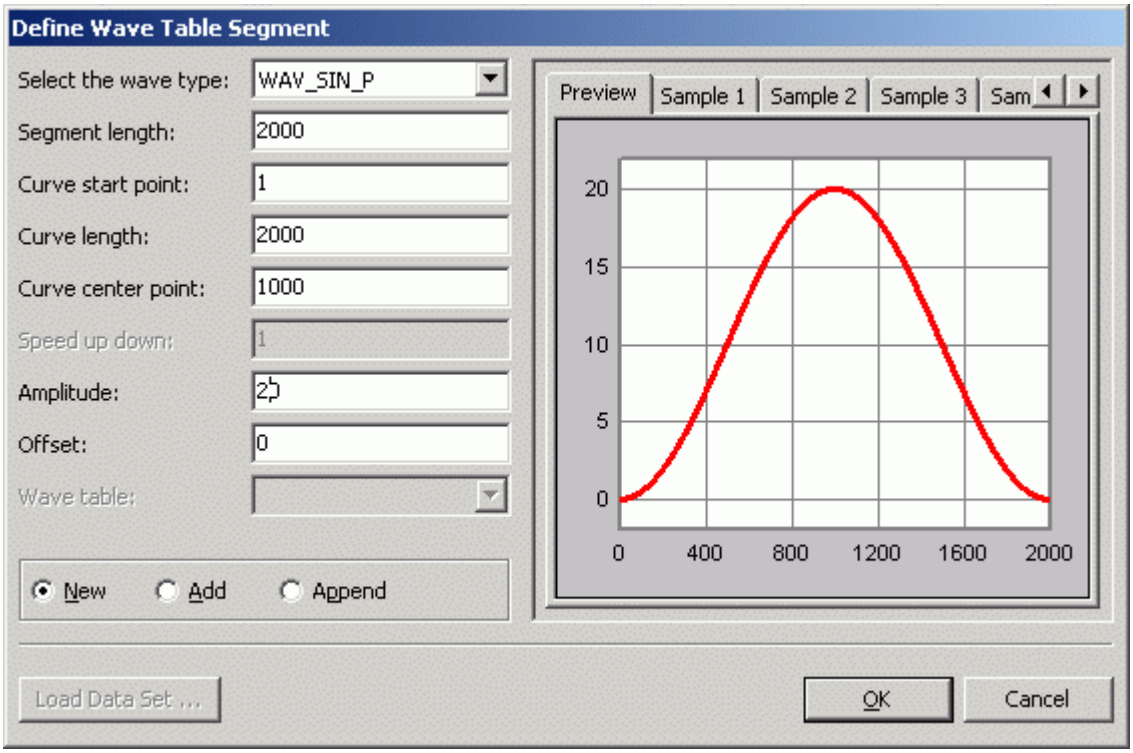


Figure 56: Dialog for segment definition

Define segment handling:

- When the ***New*** radio button is active, the content of the wave table will be completely overwritten with the new segment.
- The ***Add*** option will add the point values of the newly defined segment to those already in the wave table point by point (i.e. modifies the waveform amplitude).
- The ***Append*** option will append the currently defined segment to the previously defined wave table content (i.e. concatenates newly defined segment to existing waveform).

Define the curve shape for the segment:

1. Select the wave type—the fields necessary for the segment definition are then adapted depending on the wave type selected. The WAV_PNT wave type permits loading user-defined curves from data files on the host PC.

2. Enter suitable values in the available definition fields. See the examples in the right pane of the definition dialog for more information.
   For the WAV_PNT wave type, click ***Load Data Set…*** to load points from a file on the host PC. If the loaded file contains multiple waveforms, select one in the ***Wave table*** field. ***Curve start point*** gives the starting point in the wave table and ***Curve length*** the length of the wave (in points).
3. Click ***OK*** to send the defined segment to the wave table on the controller.

4.3.2 Data Recorder Tab Card

During wave generator output, data is recorded in record tables on the controller. You can configure the recording and select tables to be read afterwards.

The available tables and settings depend on the controller. Settings not supported by the controller are dimmed.

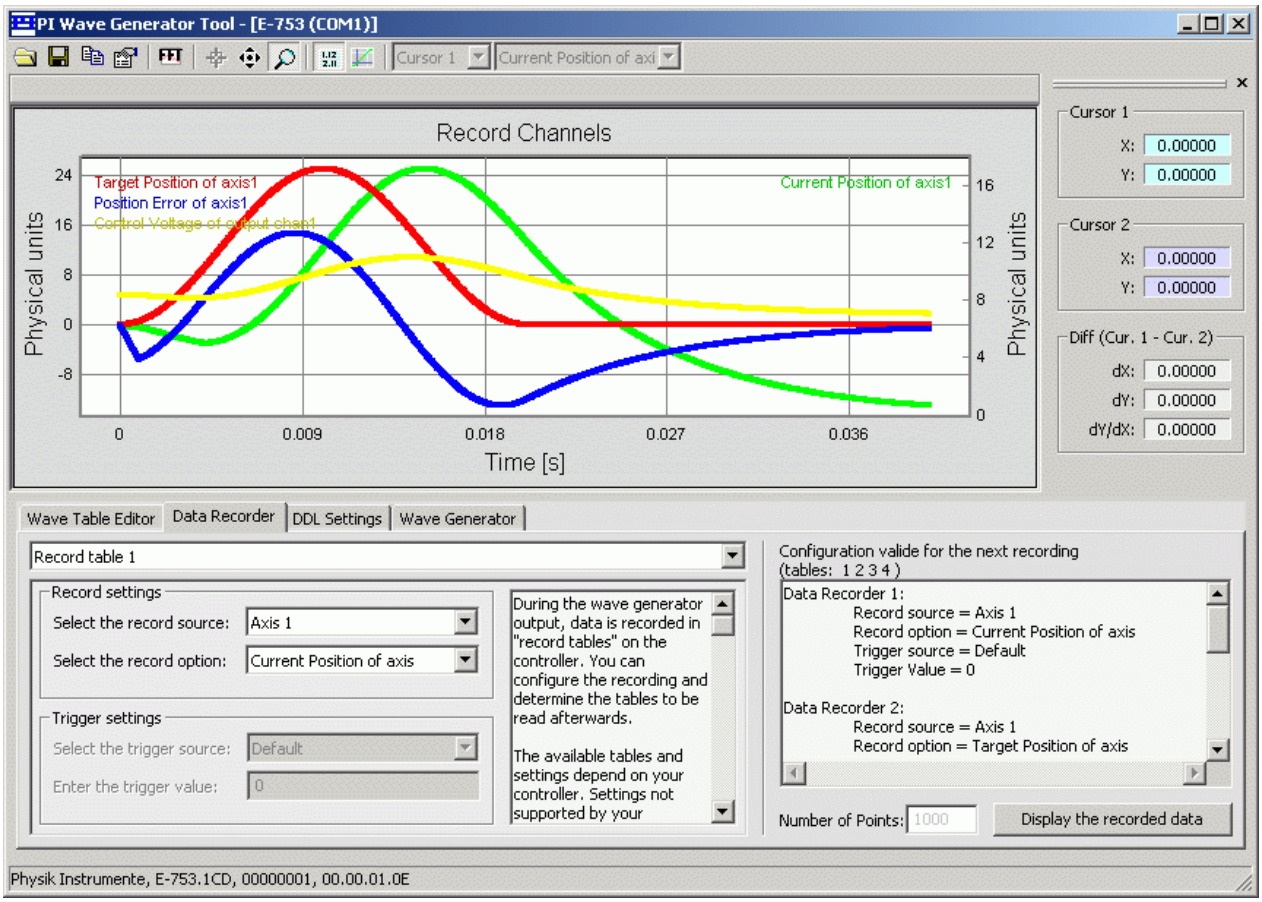


Figure 57: Data Recorder tab card

INFORMATION

Record settings and the graphics display may differ:

The graphics display always shows the record tables listed in the ***Configuration*** pane. But while the display shows the last recorded content of the tables, the settings listed in the ***Configuration*** pane will be applied to the next recording. For this reason:

- Use the graph labeling in the display to identify the displayed content.
- All data you want to see in the display must be configured before the recording is started.

How to configure the next recording:

1. Select the record table to configure.
2. Select the record source.
3. Select a suitable record option.
   If you select ***disable***, the record table is no longer listed in the ***Configuration*** pane, and it will be excluded from the display the next time you click ***Display the recorded data***.
4. Select the trigger source and enter a suitable trigger value.
5. Optionally: Set the number of points to display.
   By default, this value will be set to the length of the waveform which is output by the wave generator.
   The ***Number of Points*** value is valid for every record table listed in the ***Configuration*** pane. It is not possible to vary the displayed number of points from table to table.

To load the last recorded data to the display, click ***Display the recorded data***.

Recording is started automatically when a wave generator is started on the ***Wave Generator*** tab card (p. 121). As long as a wave generator is running, recording can be restarted by clicking ***Display the recorded data***.

INFORMATION

Depending on the controller, data may be recorded in a table even when the record option is set to ***disabled*** for that table. If this is the case, you can load that data to the display by setting any other record option for the table and clicking ***Display the recorded data***. Use the graph labeling in the display to identify the displayed content.

4.3.3 DDL Settings Tab Card

INFORMATION

On most controllers, the Dynamic Digital Linearization (DDL) feature must be expressly ordered. You can activate it after purchase and without opening the device. See the controller User Manual for more information.

DDL is used in addition to the standard servo-control algorithm and makes it possible to achieve significantly better position accuracy for dynamic applications with periodic motion. A DDL table is used to compensate for the tracking error of an axis. The tables and settings available depend on the type of controller. See the controller User Manual for more information.

An initialization phase is required to fill the DDL table with data. DDL initialization must be repeated in the following cases:

- New stage connected
- Servo parameters were changed, e.g. due to load changes
- Waveform was changed

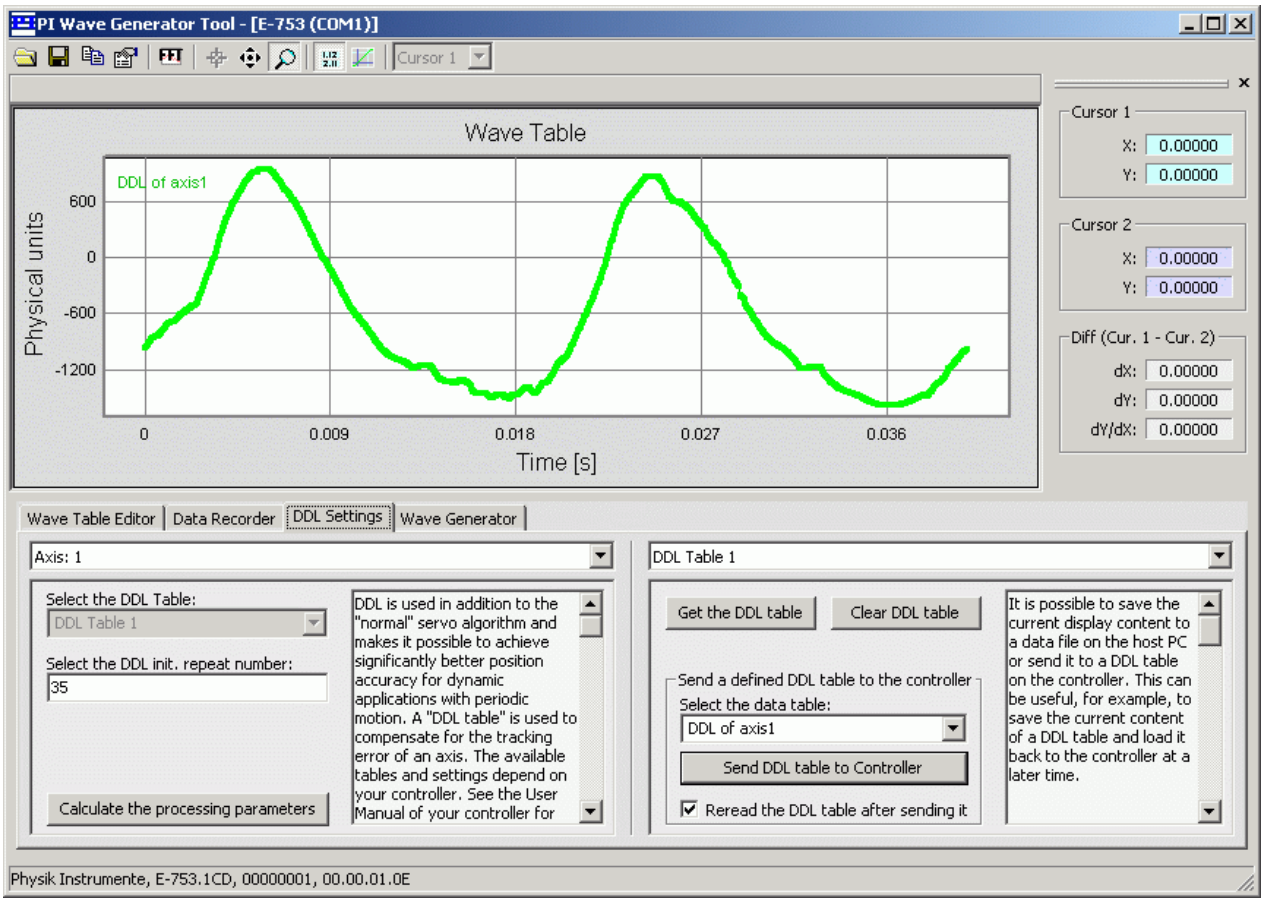


Figure 58: DDL Settings tab card

INFORMATION

Dynamic Digital Linearization (DDL) will amplify residual system resonances and cause oscillations to build up—the more wave generator cycles used for DDL initialization, the stronger the effect.

Before you work with DDL, use the NanoCapture software to eliminate oscillation by adjusting the notch filter frequency, servo-loop P-term (loop gain), I-term (time constant), and slew rate. See "Servo-Controller Dynamic Calibration" in the controller's User manual and the NanoCapture manual for more information.

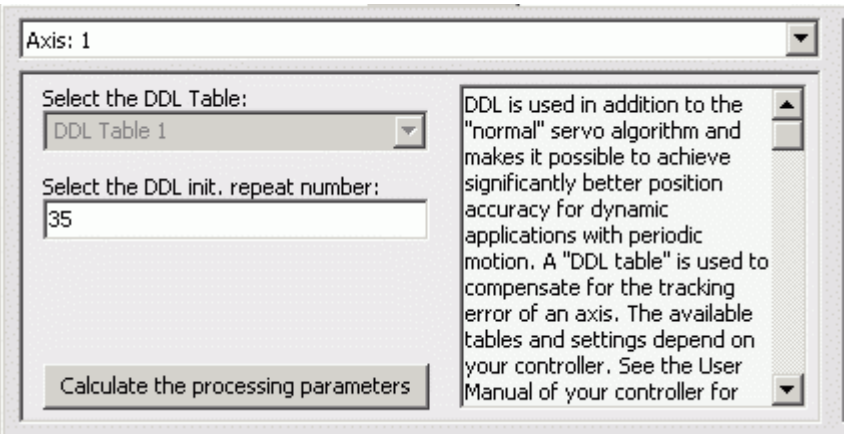


Figure 59: Left pane of the DDL Settings tab card

How to initialize and use DDL for an axis:

1. Make sure that servo is switched on.
2. On the ***Wave Table Editor*** tab card (p. 113), define the waveform.
3. In the left pane of the ***DDL Settings*** tab card, configure DDL for the axis:
   a) Select the axis.
   b) Select the DDL table to use for the axis (presently, the factory default can not be changed).
   c) Set the number of wave generator cycles to use for DDL initialization (***DDL init. repeat***, factory default = 35).
   d) If the number of DDL initialization cycles and/or the servo parameters were changed since the last time the internal DDL processing parameters were calculated, click ***Calculate the processing parameters***.
4. On the ***Wave Generator*** tab card (p. 121):
   a) Select the wave generator and hence the axis (the same axis as selected on the DDL Settings tab card).
   b) Assign the waveform (i.e. the wave table) to the axis.
   c) Activate the ***Use and reinitialize*** DDL flag.
   d) Start the wave generator and thus also the DDL initialization/usage.
5. Optionally: Examine the contents of the DDL table in the graphics display. On the ***DDL Settings*** tab card, click ***Get the DDL table*** to load the data.

As long as your application does not change, you can use the current DDL table content without new initialization. In this case, deactivate the ***Use and reinitialize DDL*** flag on the ***Wave Generator*** tab card and start the wave generator with the ***Use DDL*** flag activated.

INFORMATION

Starting DDL initialization for all axes at the same time is recommended. Each new initialization will stop all running initialization processes.

The DDL table content and the calculated processing parameters will be lost when the controller is powered down or rebooted.

Depending on the controller, it may be possible to save the DDL processing parameters to non-volatile memory.

You can examine and manage the DDL table content using the graphics display and its toolbar buttons (see "PI Wave Generator Toolbar") and the controls in the right pane of the DDL Settings tab card.

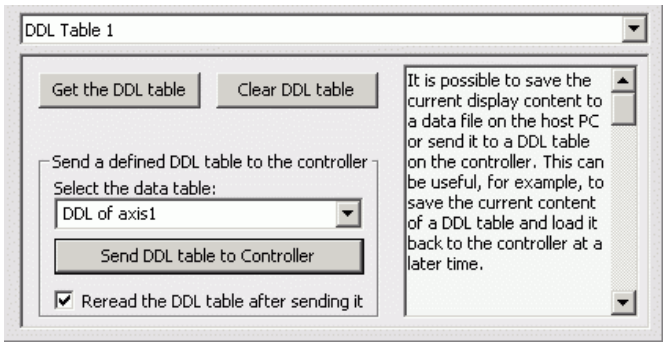


Figure 60: Right pane of the DDL Settings tab card

It is possible to save the current display content to a data file on the host PC or to send it to a DDL table on the controller. This can be useful if you want, for example, to save the current content of a DDL table and load it back to the controller at a later time.

To save the display content to a file:

1. Select a DDL table.
2. Click ***Get the DDL table*** to load the content of the selected DDL table from the controller to the display.
3. Click the ***Export data*** button in the toolbar of the ***PI Wave Generator Tool*** window to save the display content to a file on the host PC.

To send the current display content to a DDL table on the controller:

1. Select a DDL table (to this table the data will be written on the controller).
2. Optionally: Load DDL data to the display from a file on the PC using the ***Import data*** button in the toolbar of the ***PI Wave Generator Tool*** window.
3. If the display contains multiple DDL tables, select one in the ***Select the data table*** field.

4. To send the (selected) data from the display to the controller, click ***Send DDL table to controller***.
5. Optionally: If the ***Reread the DDL table after sending it*** checkbox was not activated, you can load the new DDL table content to the display using ***Get the DDL table*** to check if data was written successfully.

To delete the content of the currently selected DDL table on the controller, click ***Clear DDL table***.

4.3.4 Wave Generator Tab Card

A wave generator can output a waveform stored in a wave table on the controller to an axis. The output can be started with several options.

The number of wave generators and the settings available depend on the type of controller. See the controller User Manual for more information.

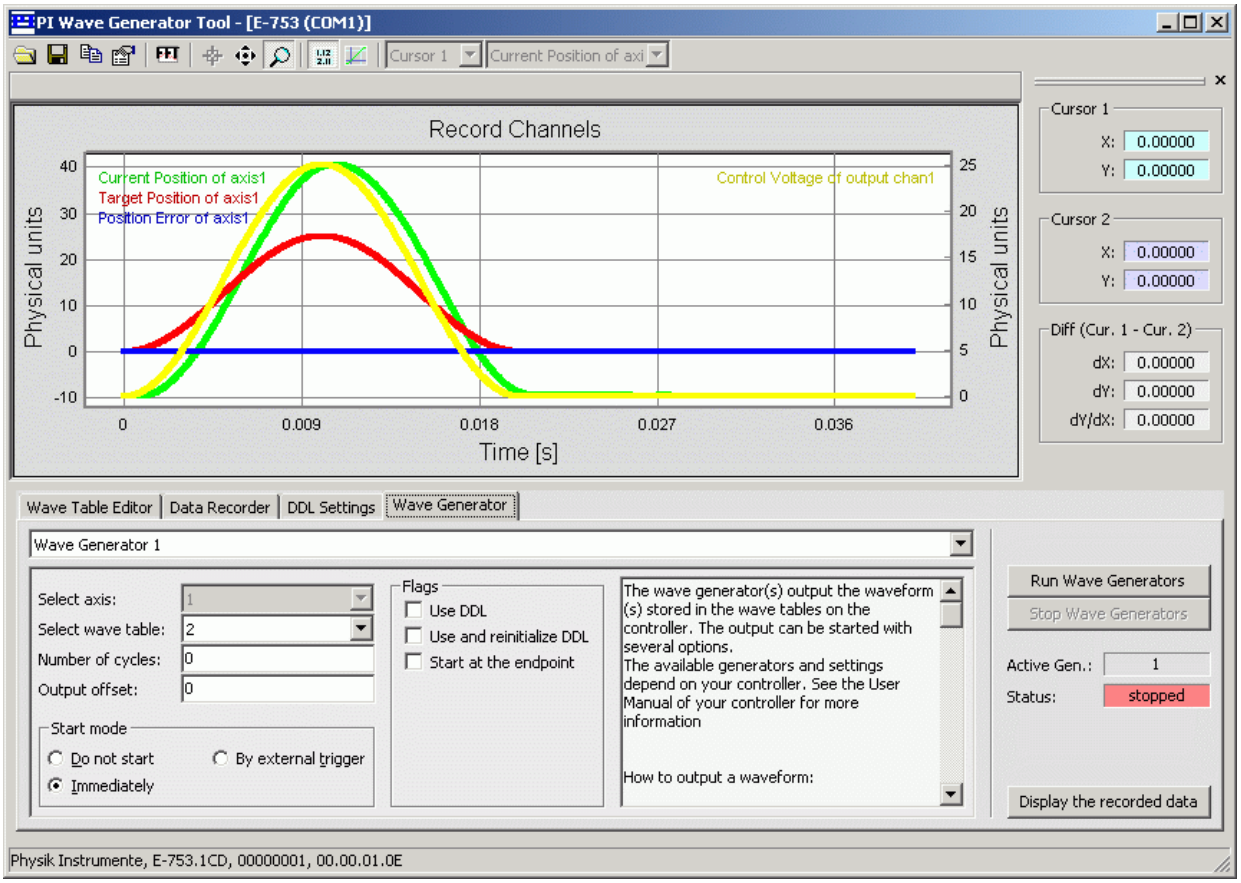


Figure 61: Wave Generator tab card; appearance depends on controller

How to output a waveform:

1. Select a wave generator; all the settings made in Steps 2 through 7 will refer to this wave generator.
2. Select the axis to be controlled by the wave generator (presently, the factory default can not be changed).
3. If selection is supported by the controller, select the wave table and hence the waveform to be output by the wave generator.

   The waveform must have been previously defined on the ***Wave Table Editor*** tab card (p. 113).

   The same wave table can be selected for multiple wave generators.
4. Set the duration of the wave generator output by setting the number of output cycles. If ***Number of cycles*** is set to 0, the wave generator will run non-stop until you click ***Stop Wave Generators***.
5. Optionally: Set a position offset which will be added to the waveform.
6. Set the start mode for the wave generator. This does not start the wave generator output yet!
7. Optionally: Set options for the wave generator output by activating the corresponding flag(s).
8. Start wave generator output by clicking ***Run Wave Generators***. This starts all wave generators whose ***Start mode*** is different from ***Do not start***.

You can stop all wave generator output by clicking ***Stop Wave Generators***.

When the wave generator output is started, data recording starts automatically. To load the last-recorded data to the display, click ***Display the recorded data***. As long as a wave generator is running, recording is then restarted. Recording can be configured on the ***Data Recorder*** tab card (p. 115).

INFORMATION

If multiple wave generators are running simultaneously, with some controllers, their output cycles will be adapted to have the same length—on most controllers, cut to the length of the shortest waveform currently running.

If the ***Start at the endpoint*** flag is activated, the ***Output Offset*** value will be incremented internally with each output cycle. When the wave generator is stopped, the current valid offset value will be automatically loaded back to the ***Output Offset*** field.

Servo must be on if the wave generator is started with the ***Use DDL*** or ***Use and reinitialize DDL*** flags activated. Otherwise the DDL feature will not work.

Wave generator output will continue even if the ***PI Wave Generator*** Tool is closed or if the high voltage output is deactivated.

The assignment of wave tables to wave generators is displayed on the ***Wave Table Editor*** tab card, in the ***Axis information*** pane.

Depending on the controller, it may be possible to modify the wave generator output by setting the wave table rate. See the controller User Manual for more information.

4.4 Wave Table Operation

Some controllers, e.g. E-816, can be controlled by an internal "wave table" which outputs user-specified target points. The interpretation of these target points (e.g. as positions or voltages) depends on the controller. The wave table feature is especially important in dynamic applications which require periodic motion of the axes.

The following menu items on the controller menu of supporting controllers (e.g. E-816) refer to wave table operation:

- ***Show wave table editor…***: Define wave table points for later output and save them to the controller or to data files on the host PC. A graphics pane is available. See "Wave Table Editor Window" (p. 123) for details
- ***Start wave table output…***: Wave table output can be started immediately or by external trigger pulses. See "Start Wave Table Output" (p. 126) for details
- ***Stop wave table output…***: See "Stop Wave Table Output" (p. 127) for details

4.4.1 Wave Table Editor Window

Use the Show wave table editor… item in the controller menu to open the ***Wave Table Editor*** window. In this window, you can define wave table points for later output, check them in a graphics pane and save them to the controller or to data files on the host PC.

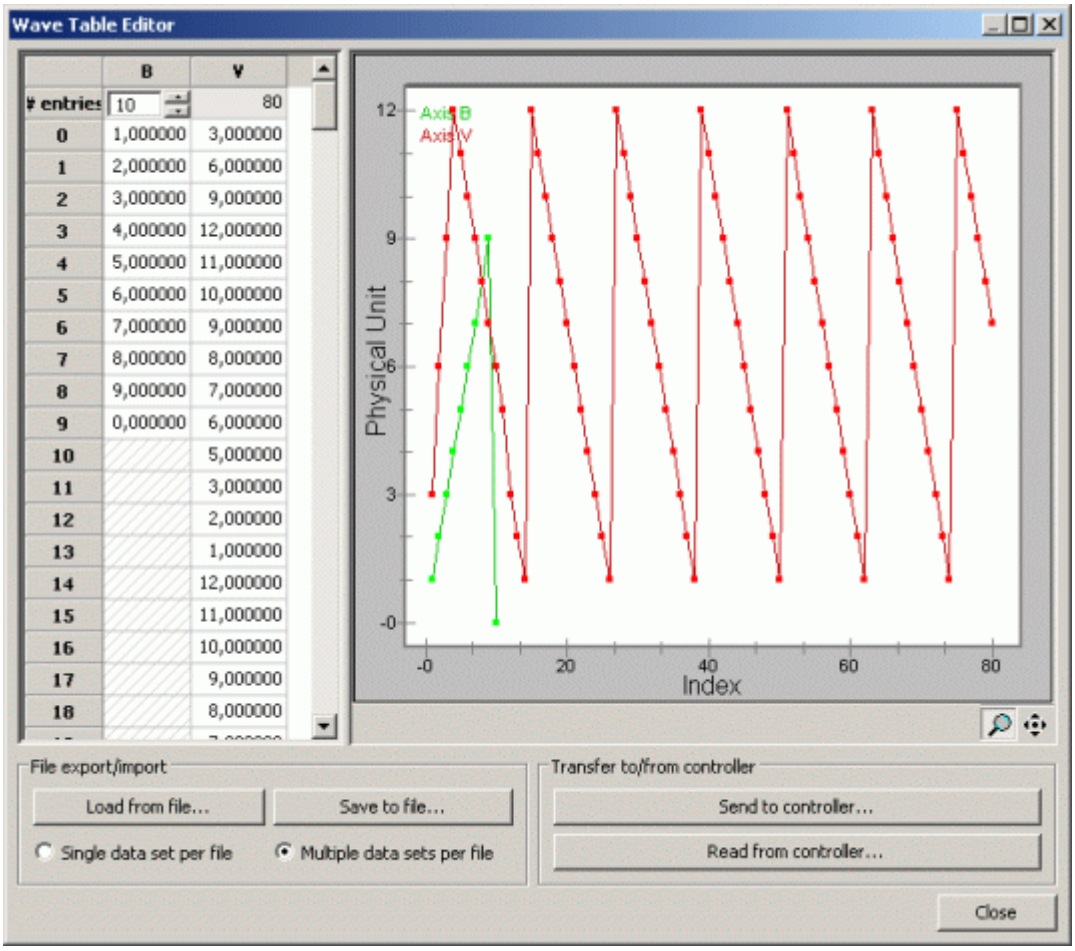


Figure 62: Wave Table Editor window

The table in the leftmost pane of the ***Wave Table Editor*** window lists the wave table content, i.e. the target points for all currently connected axes which support wave table operation. The table content is diagramed by the graphics pane.

INFORMATION

When the ***Wave Table Editor*** window is opened, the table does not show the actual values from the controller(s) yet. To load the data for the individual axes from the controller(s), click the ***Read from controller...*** button below the graphics pane. Reading can take some time, depending on the number of points to read.

The leftmost column of the table shows the index of the target points. The other columns belong to individual axes, with the table header showing the axis identifier. In the first row of the axis columns (***# entries***), you can determine the number of points to be written for each axis. The maximum number of points depends on the controller (e.g. 256 with E-816 firmware rev. 3.20 and newer). All subsequent rows of the axis columns contain the values of the target points. The interpretation of the target points depends on the controller used and on the current servo mode (e.g. with E-816: voltages in open-loop operation (servo off) and positions in closed-loop operation (servo on)).

To change a value, click into the corresponding field and enter the new value, or, with the ***# entries*** fields, use the arrow buttons which appear. The graphics pane is updated immediately.

INFORMATION

Values entered in the table are not written to the controller or to a file on the host PC until you click the corresponding button.

To write the current table content to the controller and hence overwrite the values which may already be present there, click the ***Send to controller…*** button below the graphics pane. Depending on the controller, the written points may be automatically saved to non-volatile memory (see controller User Manual for details).

You can save the data from the table to a file on the host PC using the ***Save to file…*** button (Wave Table file with the extension .tbl). The current table content is considered as a single data set. By activating the corresponding radio button, you can save every data set to a new file, or save multiple data sets to the same file. To load the data back from a file on the host PC, use the ***Load from file …*** button. You can assign the data sets to the individual axes when loading.

INFORMATION

Wave table content which has not been sent to the controller or saved to a file on the host PC is lost when the ***Wave Table Editor*** window is closed.

To start the wave table output, close the ***Wave Table Editor*** window and select the ***Start wave table output…*** item from the controller menu. See "Start Wave Table Output" (p. 126) for details.

4.4.2 Start Wave Table Output

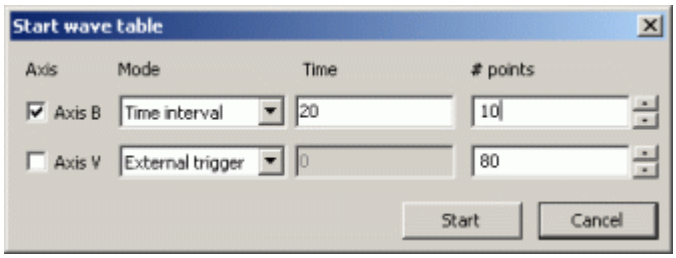


Figure 63: Window for mode selection and start of wave table output

Use the ***Start wave table output…*** item in the controller menu to open the ***Start wave table*** window. Each line in this window is dedicated to one axis.

In the ***Axis*** column, you can determine if the wave table output is to be started for the axis when clicking the ***Start*** button. Removing the check in the ***Axis*** column means that the state of the wave table output for that axis will remain unchanged.

In the rightmost column, ***# points***, you specify the number of wave table points to use for output. If 0, wave table output is disabled. See the User Manual of your controller for the maximum number of points (E-816: 256 points with firmware rev. 3.20 and newer).

In the ***Mode*** column, you can select the following start options (default configuration, for other wave table output configurations see the controller User Manual):

- ***Time interval*** means that the wave table output is started immediately when clicking the Start button. Each point will be output for the amount of time specified in the Time column (in milliseconds). Output will roll over for the points specified by # points and continue until stopped, e.g. using the Stop Wave Table Output menu item in the controller menu

  Note that wave table output is disabled when the Time interval value is 0.
- ***External trigger*** means that the wave table output is to be started by external trigger pulses after the ***Start*** button was clicked. By default, one wave table point is output each time an external trigger pulse is received. Make sure that the controller is configured to accept trigger input, and that a suitable trigger signal is available. See the User Manual of your controller for details

4.4.3 Stop Wave Table Output

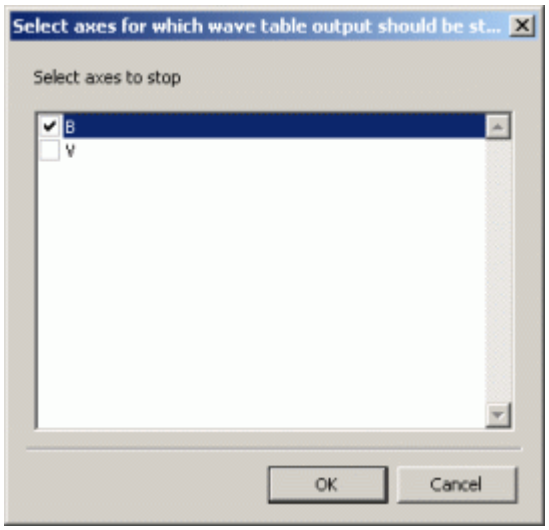


Figure 64: Window for stop of wave table output

Use the ***Stop wave table output…*** item in the controller menu to open the corresponding axis selection window. To stop the wave table output for an axis, check the corresponding checkbox and click ***OK***.

You can stop the wave table output for an axis also in the ***Start wave table window*** by entering 0 in the corresponding ***# points*** field (see "Start Wave Table Output" (p. 126)).

4.5 Enable Triggered Move

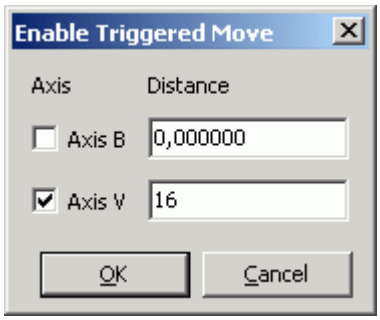


Figure 65: Window where triggered motion can be enabled

With some controllers, e.g. E-816, axis motion can be controlled by external trigger pulses. Each pulse causes a relative step of a predefined size.

The ***Enable triggered move…*** item is on the controller menu of supporting controllers (e.g. E-816). It opens the corresponding window where each line belongs to one axis.

Before enabling triggered motion, make sure that the controller is configured to accept trigger input, and that a suitable trigger signal is available (see the User Manual of your controller for details).

To enable or disable triggered motion for the individual axes, use the ***Axis*** checkboxes. Triggered motion is enabled for an axis when its box is checked. To set the step size to be used, type the corresponding value(s) in the ***Distance*** field(s). The interpretation of the step size value (e.g. as position in µm or as voltage in volts) depends on the controller.

INFORMATION

Depending on the controller, triggered motion and other control sources (e.g. target values given on the ***Axes*** tab card or wave table output) may be mutually exclusive. If this is the case, other control sources will cause an error for the axis as long as triggered motion is enabled for this axis.

4.6 Data Recorder

The ***Show/Hide data*** recorder menu item is on the controller menu of supporting controllers. It gives access to the ***Data Recorder*** window.

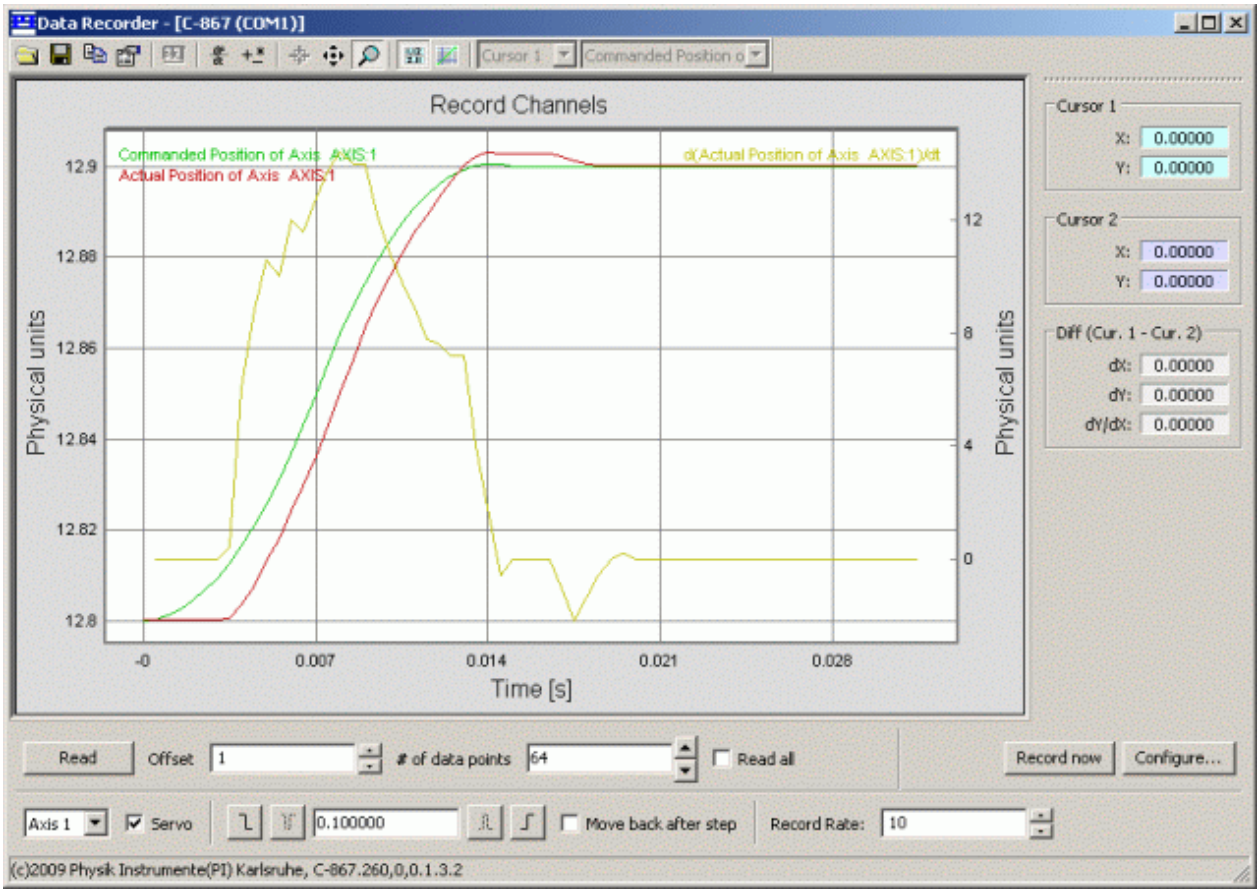


Figure 66: Data Recorder window with displayed data

In the ***Data Recorder*** window, you can:

- Configure the data recorder (p. 130) and define actions to be applied by PIMikroMove to the recorded data.
- Start data recording via step and impulse buttons (p. 132) and via the ***Record now*** button (p. 133). Note that depending on the controller, data recording may also be triggered by actions done outside of the ***Data Recorder*** window.
- Read recorded data (p. 133) from the controller to display it in the graphics pane. For data reading it is irrelevant if recording was started inside or outside of the ***Data Recorder*** window.
- Configure the graphics pane (p. 134) and export or import data to it.

See the controller User Manual for the controller-specific data recorder properties, e.g. for available record options and trigger settings, for the number of data recorder tables and the maximum number of points per table.

INFORMATION

Depending on the controller, there may be additional tools which use the data recording functionality (see, for example, "PI Tuning Tool" or "PI Wave Generator Tool"). The ***Data Recorder*** window can not be opened as long as any of these tools is in use.

INFORMATION

The display always shows the **last-recorded content** of the data recorder tables. Data source and record option for the data you want to see in the display must be configured before the recording is started.

Configuration changes subsequent to recording are possible for actions performed by PIMikroMove (e.g. calculation of derivatives, arithmetic operations, application of bitmasks or scaling, see "Configure Data Recorder" (p. 130) and "Configure Graphics Pane" (p. 134) for details).

4.6.1 Configure Data Recorder

With the ***Configure...*** button you can obtain the ***Configure Data Recorder*** dialog. In this dialog, you can change the assignment of data sources (axes or channels) and record options to the data recorder tables, and set the trigger option for recording. Furthermore, you can configure actions to be applied by PIMikroMove to the recorded data.

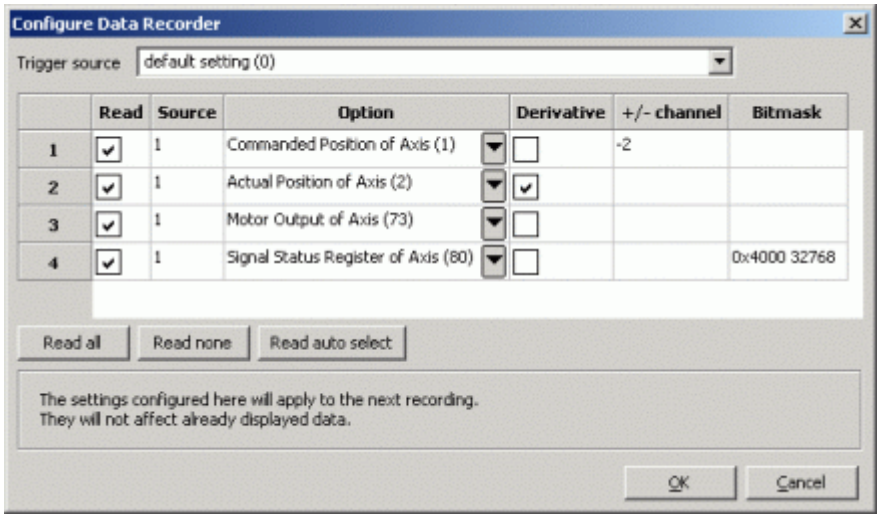


Figure 67: Configure Data Recorder dialog

In the leftmost column of the dialog, the available data recorder tables are listed.

In the ***Read*** column, you can define whether the content of the corresponding record table should be read from the controller and displayed. To select all data recorder tables, click the ***Read all*** button below the table. With the ***Read none*** button, you can deselect all tables. The ***Read auto select*** button selects all data recorder tables for which a record option is set in the ***Option*** column.

In the ***Source*** column, you can type in the identifier of the data source (e.g. an axis ID) which is to be assigned to a data recorder table. To select the record option, i.e. the type of data to be recorded, double-click in the fields of the ***Option*** column. Changes of settings for ***Source*** and ***Option*** will always be applied to the next recording, but not to the data which was already recorded.

In the ***Derivative, +/-channel*** and ***Bitmask*** columns you can define actions to be performed by PIMikroMove to the recorded data immediately after reading them from the controller. The results of these actions are only present on the host PC but not in the data recorder tables of the controller. Each selected action adds a graph to the display:

- In the ***Derivative*** column, you can select if the derivative of the recorded data is to be calculated.
  Note that you can also calculate derivatives later using the ***Add derivatives…*** icon in the data recorder toolbar (p. 134).

- In the ***+/- channel*** column, you can define arithmetic operations to be applied to the recorded data. Supported are addition and subtraction. Example: you want to have a separate graph showing the difference of the data in the first and the second data recorder table. In this case, enter "-2" in the ***+/-channel*** field of the first data recorder table (see figure above).
  Note that you can also perform arithmetic operations later using the ***Math operation…*** icon in the data recorder toolbar (p. 134).
- In the ***Bitmask*** column, you can define a bitmask filter to be applied to the recorded data. Application of this filter makes sense only if the recorded data is bit-mapped, e.g. if you have recorded the content of the signal-state register. For the status bit you want to see as separate graph, enter the bit value either in decimal or in hexadecimal format in the ***Bitmask*** field of the data recorder table. To define multiple bits for the same data recorder table, enter the bit values separated by a space. Example: you will record the content of the signal-state register for a C-867 controller in the fourth data recorder table, and you want to have separate graphs showing the "IsMoving" and "OnTarget" bits of the register. In this case, enter the values for bit 14 ("IsMoving") and bit 15 ("OnTarget") in the ***Bitmask*** field of the fourth data recorder table. In the figure above, the value of bit 14 is entered in hexadecimal format (0x4000) while the value of bit 15 is given in decimal format (32768).

If supported by the controller, the ***Trigger source*** field is present: there you can specify how recording is to be triggered.

INFORMATION

Do not select next command (2) for ***Trigger source***. Because of the background processing done by ***PIMikroMove*** recording would start immediately.

See the controller User Manual for the possible sources and options and for the default settings.

4.6.2 Perform Step or Impulse Measurements

In the bottom line of the ***Data Recorder*** window, you can start a step or impulse response measurement. The recorded data will be read and displayed after the recording has finished.

Control description, from left to right:

- Axis selection field: Select the axis for which the measurement is to be performed.
- ***Servo*** checkbox: Select the servo state. Depending on the controller, the measurement can be done in closed-loop or open-loop operation (i.e. with servo on or off).
- Start buttons for step and impulse with negative offset: Start motion in negative direction and recording. Buttons are grayed out if the corresponding action is not supported by the controller.
- Offset input field: Enter the amplitude to be used for the step or impulse. The unit of the amplitude in open-loop and closed-loop operation depends on the controller. See the User manual of the controller for more information.
- Start buttons for step and impulse with positive offset: Start motion in positive direction and recording. Buttons are grayed out if the corresponding action is not supported by the controller.
- ***Move back after step*** checkbox: If selected, after a step the axis moves back to the starting position.
- ***Record Rate*** field: Specify the number of servo-loop cycles before each next data point is recorded, must be an integer value ≥ 1. The larger the value, the longer the time period covered by recording. The duration of a recording can be calculated as follows:
  Rec. Duration = Servo Cycle Time * Record Rate * Number of Points
  where *Servo Cycle Time* and *Number of Points* (length of the data recorder table) depend on the controller, see controller User Manual
  Note that the ***Record Rate*** is a global data recorder setting which will not only affect step or impulse measurements.

4.6.3 Start Recording Without Motion

If supported by the controller, you can start recording without any axis motion using the ***Record now*** button. PIMikroMove will then send a predefined suitable command to the controller. The recorded data will be read and displayed after the recording has finished.

4.6.4 Read Data from Controller and (Re)Apply Actions to the Data

Below the graphics pane of the ***Data Recorder*** window, you can set the starting point (***Offset***) and the number of data points (***# data***) to be read from the controller. To read the data and display it in the graphics pane, click the ***Read*** button. Reading can take some time depending on the number of points to be read. For data reading it is irrelevant if recording was started inside or outside of the ***Data Recorder*** window.

If you have configured actions to be applied to the recorded data, e.g. calculation of derivatives, sums or differences, or application of bitmasks, they are (re)applied when you click ***Read***. See "Configure Data Recorder" (p. 130) for more information regarding such actions performed by PIMikroMove.

4.6.5 Configure Graphics Pane

You can configure the graphics pane via its toolbar. A short description will be displayed if the mouse cursor hovers over a toolbar button.

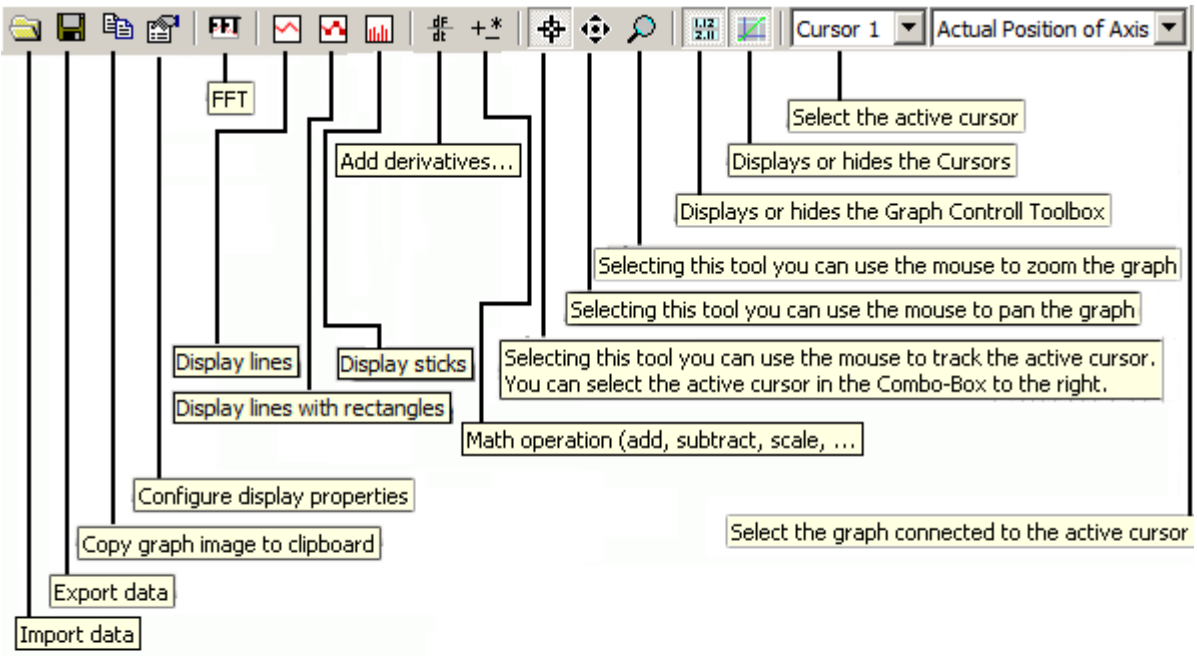


Figure 68: Toolbar for windows with graphic pane

Import data

You can load data to the display by importing them from .dat files on the host PC (GCS Array format, see GCS Array manual, SM146E).

Export data

You can save the display content to the host PC by exporting the data as "comma separated values" (.csv format, e.g. for MS Excel) or in GCS Array format (see GCS Array manual, SM146E).

Copy graph image to clipboard

You can save the display content to the host PC by copying the displayed graph image as bitmap to a file.

Configure display properties

You can open a dialog to configure the display properties of the graphics pane:

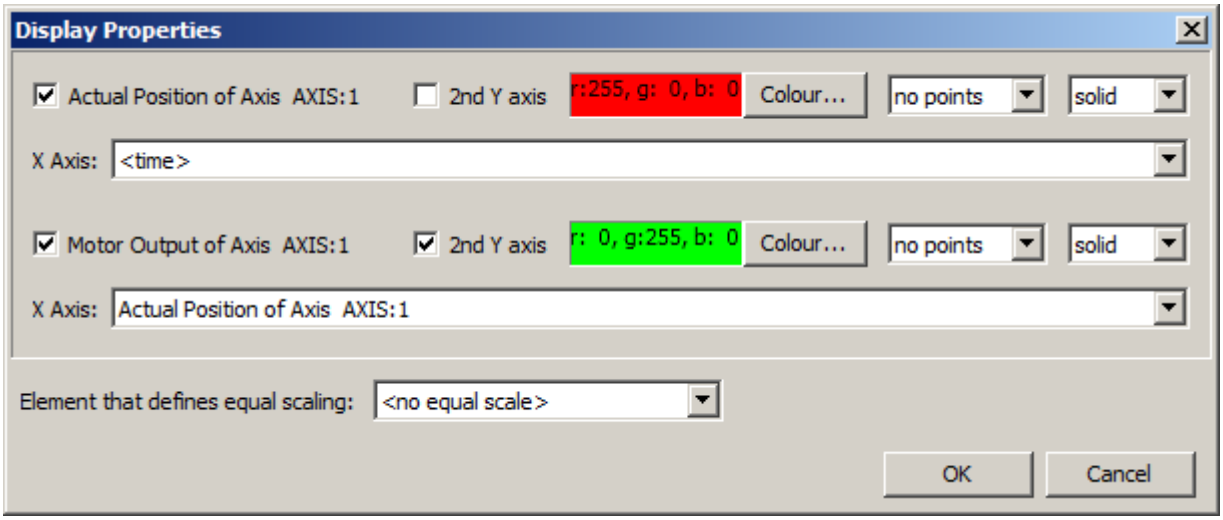


Figure 69: Display Properties dialog

In the ***Display Properties*** dialog, you can select the data to be displayed.

It is recommended to assign any data whose range is greatly different from that of other data to a second (rightmost) Y-axis in the graphics pane.

By default, the time is used for the x-coordinate of the data display. For all data, you can replace the time as x-coordinate by any other data present in the ***Display Properties*** dialog. To do this, select the desired data in the ***X Axis*** field below the line of the data which is to be used for the y-coordinate. This can be helpful, for example, if you want to display the actual position of an axis over the motor output for that axis.

If the data for the y-coordinate and the x-coordinate of a curve have the same physical unit, the scaling of the coordinates may differ in the display. You can select a base for equal scaling of the x- and y-coordinate in the ***Element that defines equal scaling*** field. Example: If you have a multi-axis controller and want to display the actual position of axis 1 over the actual position of axis 2, first select the actual position of axis 2 in the ***X Axis*** field below the line of the actual position of axis 1. Then select the actual position of axis 1 as the base for equal scaling.

Furthermore you can set the colour and the appearance of data points and lines in the display for every single curve. To set the appearance of data points and lines identical for all data, use the corresponding toolbar buttons of the graphics pane instead.

FFT

With the ***FFT*** (fast Fourier transformation) button, you can switch between two display modes in the graphics pane: the "normal" display and a frequency diagram of the data read. The current display properties for the FFT diagram can differ from those for the "normal" display—use the ***Display Properties*** dialog to select the preferred settings.

Display lines

Sets the appearance of all data in the display to "lines". To change the appearance of single curves, configure them in the ***Display Properties*** dialog.

Display lines with rectangles

Sets the appearance of all data in the display to "lines with rectangles". To change the appearance of single curves, configure them in the ***Display Properties*** dialog.

Display sticks

Sets the appearance of all data in the display to "sticks". To change the appearance of single curves, configure them in the ***Display Properties*** dialog.

Add derivatives…

Opens a dialog where you can select data for which derivatives are to be calculated by PIMikroMove. These derivatives are only present on the host PC but not in the data recorder tables of the controller. Derivatives can be calculated for all recorded data and for all data resulting from former calculations (derivatives or results of the arithmetic operations described below). Their appearance in the display can be configured via the ***Display Properties*** dialog (e.g. show or hide data in the display; see above).

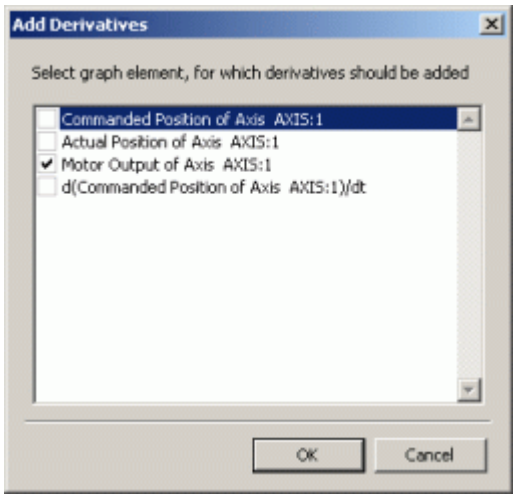


Figure 70: Add Derivatives dialog

Math operation (add, substract, scale,…)

Opens a dialog where you can define an arithmetic operation or a scaling factor which will be applied by PIMikroMove to the selected data. The results are only present on the host PC but not in the data recorder tables of the controller. Arithmetic operations or the scaling factor can be applied to all recorded data and to all data resulting from former calculations (also to derivatives, see above). The appearance of the calculated data in the display can be configured via the ***Display Properties*** dialog (e.g. show or hide data in the display; see above).

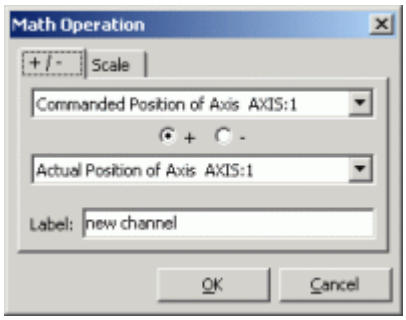


Figure 71: Tab card for arithmetic operations

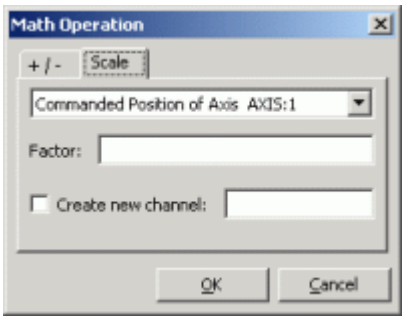


Figure 72: Tab card for scaling

... use the mouse to track the active cursor

This icon is only available when the cursors are displayed (see below). If you activate this icon, you can move one of two cursors along a graph by dragging it with the left mouse button pressed. Choose the active cursor and the graph connected to the active cursor in the toolbar (see below). The two cursors can be connected to the same graph. While you move a cursor, you can read out the individual coordinates in the cursor information panes of the ***Graph Control Toolbox*** (see below).

... use the mouse to pan the active graph

If you activate this icon, you can scroll the content of the graphics pane by placing the mouse pointer in the graphics pane and dragging it with the left mouse button pressed. To get back the initial state of the display and undo any scrolling actions, click on the zoom icon (see below) and double-right-click in the display.

... use the mouse to zoom the graph

If you activate this icon, you can magnify a selected area of the graphics pane. To do this, drag the mouse diagonally across the desired area with the left mouse button pressed.

To get back the initial state of the display and undo any zooming or panning (see above) actions, double-right-click the display.

Displays or hides the Graph Control Toolbox

The ***Graph Control Toolbox*** contains several cursor information panes and is located to the right of the graphics pane (if displayed). Note that in the ***Diff*** (Cur. 1 - Cur. 2) pane, the absolute values of the difference are displayed. You cannot move the cursors to negative time values.

Displays or hides the Cursors

The cursors in the graphics pane are shown or hidden. If the cursors are hidden, the cursor-related icons in the toolbar are deactivated (i.e. you can not track the cursors with the mouse and it is not possible to select the active cursor and the graph connected to the active cursor).

Select the active cursor

This combo box is only available when cursor tracking is activated in the toolbar (see above). You can select the cursor which is to be moved by the mouse.

Select the graph connected to the active cursor

This combo box is only available when cursor tracking is activated in the toolbar (see above). You can specify along which graph the active cursor is to move.

4.7 Configure Trigger Output

The ***Configure trigger output...*** menu item is on the controller menu of supporting controllers. It gives access to a separate ***Configure Trigger Output*** window where you can specify trigger conditions for the digital output lines present on the controller.

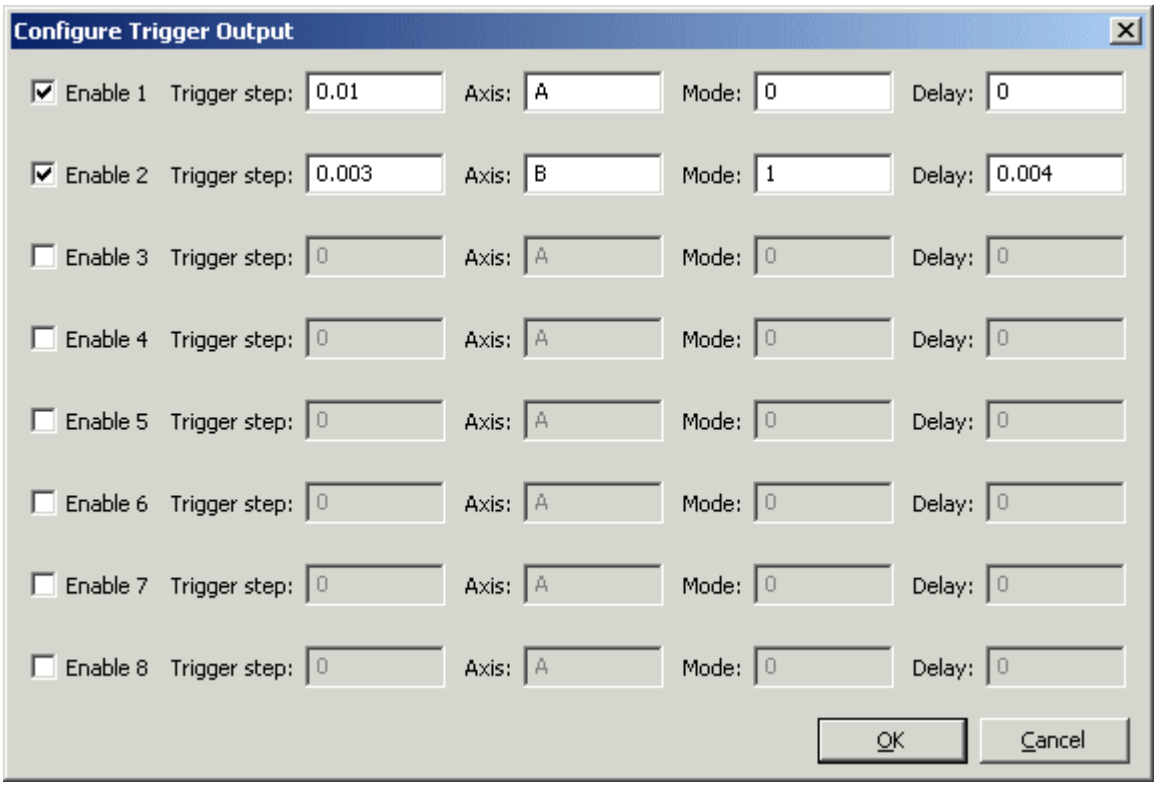


Figure 73: Example: Configure Trigger Output window for C-702

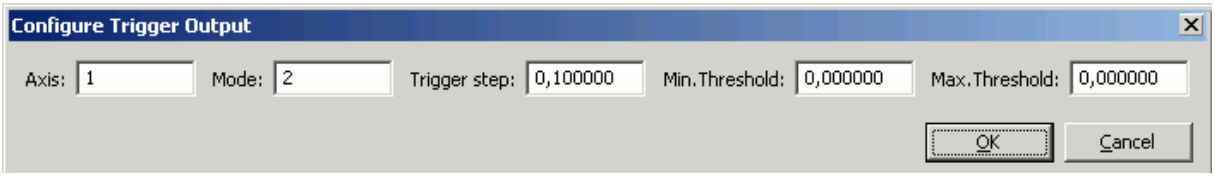


Figure 74: Example: Configure Trigger Output window for E-753 (only one digital output line is provided for trigger output)

INFORMATION

In C-702, C-867 and Mercury (C-863 and C-663) controllers, programmable digital outputs and trigger outputs share the same circuitry. Make sure that the same lines are not activated for both functions. Digital I/O is configured from the ***View*** menu. Should lines be called using alphabetic characters in the ***Digital I/O*** Window, then A is the same as line 1 in the ***Configure Trigger Output*** window, B is 2, etc.

Each line in the ***Configure Trigger Output*** window is dedicated to one digital output line of the controller. The fields shown depend on the controller (see the description of the CTO command in the controller User Manual for more information):

- If ***Enable*** check boxes are present, the trigger settings only become active for a digital output line when the box is checked.
- The ***Mode*** setting specifies the trigger mode to be used. The available trigger modes depend on the controller. The mode specified determines which of the other fields are required / ignored:

  0 = Position distance; axis-related. With this trigger mode, a trigger pulse is output whenever the axis has covered the ***Trigger step*** distance.

  1 = Position distance + wait time; axis-related. With this trigger mode, a trigger pulse is output a specified time (***Delay time***) after the axis has covered the ***Trigger step*** distance.

  2 = OnTarget; axis-related. With this trigger mode, the on-target status of the selected axis determines the state of the digital output line (active-high logic; this status can also be read with the ONT? command in the ***Command entry*** window).

  3 = MinMaxThreshold; axis related. With this trigger mode, values for ***Min.Threshold*** and ***Max.Threshold*** must be defined. When the axis position of the selected axis is inside the band specified by the ***Min.Threshold*** and ***Max.Threshold*** values, the digital output line is set high, otherwise it is set low.

  4 = Generator Trigger; related to the wave generator output. With this trigger mode, the trigger actions on the digital output line must be defined for certain waveform points using the TWS command in the ***Command entry*** window.

  5 = MotionError; the ***Axis*** setting is irrelevant with this trigger mode. The selected trigger line becomes active when a motion error occurs. The line will stay active until the error code is reset to 0 (by a query).

  6 = InMotion; axis related. The selected trigger line is active as long as the selected axis is in motion (the in-motion state can also be read with the #4, #5 or SRG? commands in the ***Command entry*** window).

7 = Position+Offset; axis related. With this trigger mode, the first trigger pulse is written when the axis has reached the position given by ***Position***. The next trigger pulses each are written when the axis position equals the sum of the last valid trigger position and the increment value given by ***Trigger step***. Trigger output ends when the axis position exceeds the value given by ***StopThreshold***. The sign of the ***Trigger step*** value determines for which direction of motion trigger pulses are to be output. Trigger processing is done by the DSP of the controller.

8 = SingleTrigger; axis related. With this trigger mode, a trigger pulse is written when the axis has reached the trigger position given by ***Position***.

9 = HardwareTrigger; axis related. This trigger mode is the same as the Position+Offset mode, but done by the FPGA of the controller and therefore much faster. This trigger mode can only be selected for output lines 1 and 2. Output line 1 must be connected to the first axis and output line 2 to the second axis via the ***Axis*** fields.

- For the axis-related trigger modes, the digital output line is associated with the controller axis whose identifier is entered in the ***Axis*** field.

See the controller User Manual for more information (e.g. command descriptions, I/O connector pinout and examples).

4.8 Calibrate Controller Joysticks

The ***Calibrate controller joystick...*** menu item is on the controller menu of supporting controllers. It gives access to a separate ***Joystick Calibration*** window for the selected joystick device. In this window, you can "calibrate" the individual axes of the joystick device by changing their lookup tables in the controller as follows:

- Select the linear or the parabolic standard lookup table. The parabolic lookup table offers more sensitivity during slow motions.
- Measure joystick parameters and write a custom lookup table.

INFORMATION

Only joysticks directly connected to the controller are affected here. If you have a joystick connected to the host PC see "Configure PC Joysticks Window" (p. 91) for how to test and calibrate it.

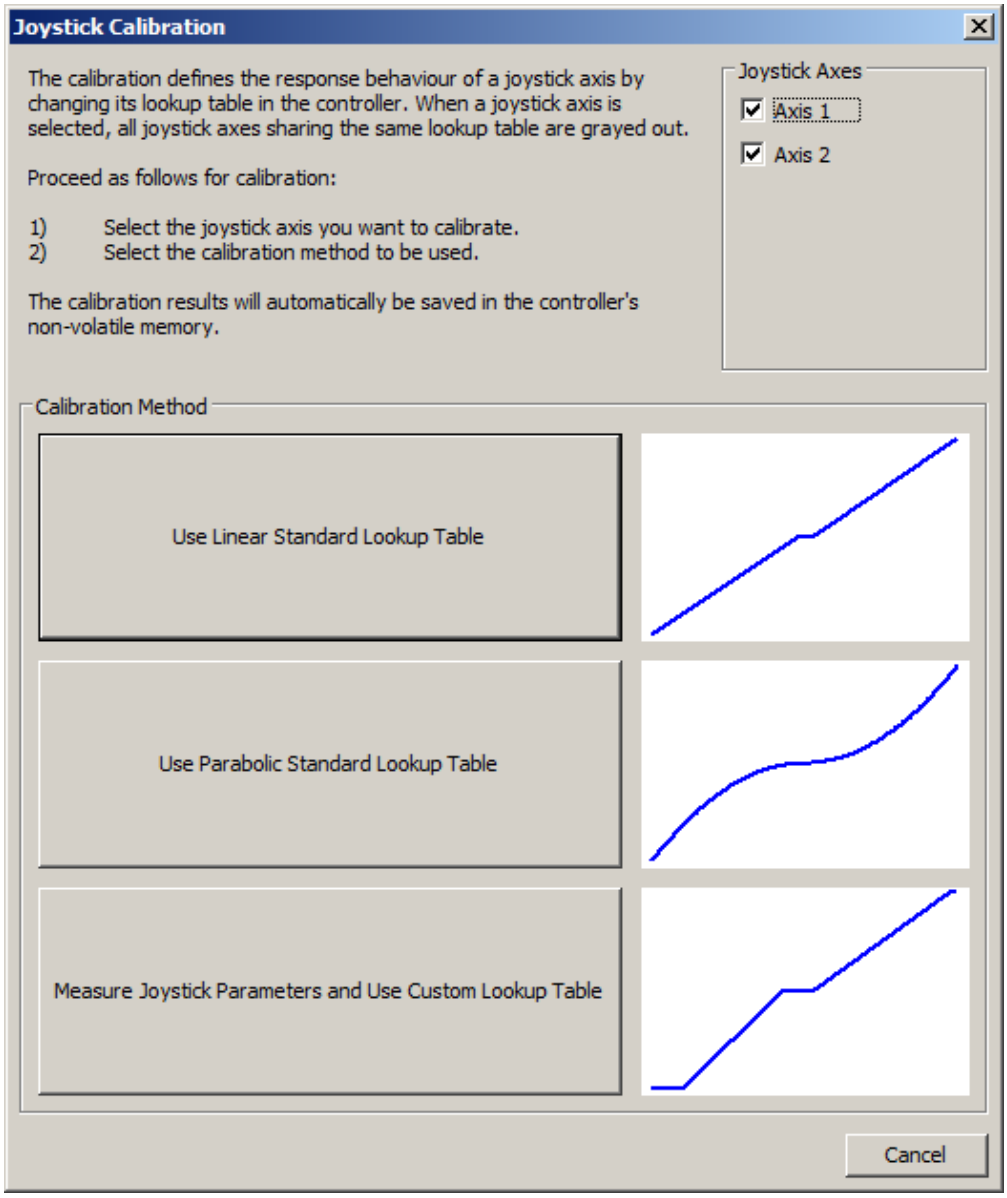


Figure 75: Joystick Calibration window for a C-867.260 two-axis controller which supports two axes per joystick device

The calibration of the joystick axes is necessary in the following cases:

- The joystick response behaviour is not suitable for your application.
- You are using the Z-axis of a C-819.30 joystick device: This joystick axis cannot be used with the linear or parabolic standard lookup tables and must be calibrated via the measurement of the joystick parameters which are then written to a custom lookup table. Calibration must be repeated whenever the Z-axis of the joystick is connected to another controller.

INFORMATION

If the joystick device provides operating elements for mechanical adjustment of the joystick axes, use them before you calibrate the joystick axes with PIMikroMove. For more information, see the User Manual of your controller.

Proceed as follows for calibration:

1. Select the joystick axes to be calibrated by checking the corresponding boxes in the ***Joystick Axes*** pane of the ***Joystick Calibration*** window.

   Depending on the controller, multiple joystick axes may share the same lookup table. When one of the joystick axes is selected in the ***Joystick Axes*** pane, all other joystick axes sharing the same lookup table become grayed out.

2. Select the preferred calibration method using the appropriate button in the ***Calibration Method*** pane of the ***Joystick Calibration*** window:

   – If you want to use the linear lookup table type for the joystick axes, click on ***Use Linear Standard Lookup Table***. This loads the corresponding lookup table type in the controller, and the calibration is finished.

   – If you want to use the parabolic lookup table type for the joystick axes, click on ***Use Parabolic Standard Lookup Table***. This loads the corresponding lookup table type in the controller, and the calibration is finished.

   – If you have connected the Z-axis of an C-819.30 joystick device or if you just want to map the joystick behaviour to a custom lookup table on the controller, click on ***Measure Joystick Parameters and Use Custom Lookup Table***. This opens the ***Controller Joystick Calibration*** window.

3. In the ***Controller Joystick Calibration*** window (see figure below), proceed as follows:

   a) Move the joystick to all extreme positions. This way, the custom lookup table values are identified.
   b) Release the joystick.
   c) If you want to change the dead band of the joystick axis (i.e. the range around the mid position of the joystick axis where no motion is caused), move the corresponding ***Dead band*** slider.
   d) If you want to have the custom lookup table content a parabolic shape, select the corresponding ***Parabolic curve*** box.

e) To write the custom lookup table values to the nonvolatile memory of the controller, click ***OK***. For every lookup table which is written, a separate window opens where you can monitor the write process. Calibration is finished after all windows were closed automatically.

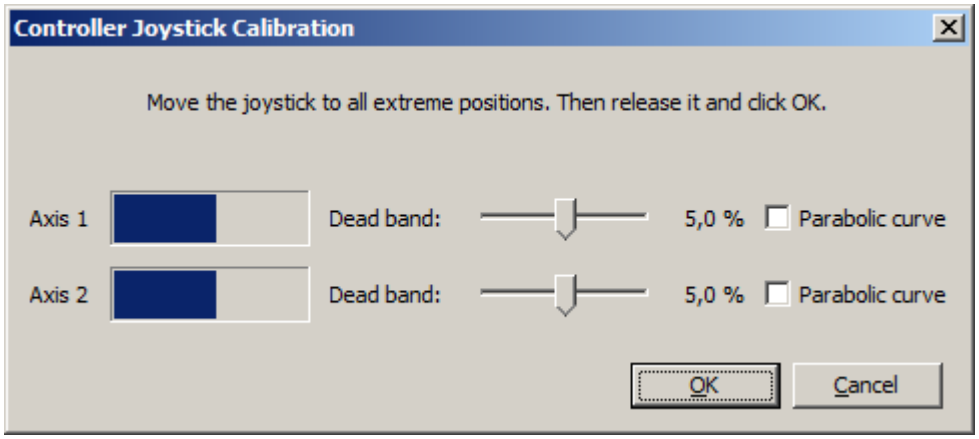


Figure 76: Controller Joystick Calibration window for the measurement of the joystick behaviour

4.9 Configure Controller Joystick(s)

The ***Configure controller joystick(s)...*** menu item is on the controller menu of supporting controllers. It gives access to a separate ***Configure Controller Joystick*** window where you can enable/disable direct joystick control for the individual controller axes.

INFORMATION

Only joysticks directly connected to the controller(s) are affected here. If you have a joystick connected to the host PC see "Position Pad Window" (p. 78) for how to use it.

With C-843 PCI cards, a joystick connected to the PC is available via both the ***Configure controller joysticks*** menu item and the ***Position Pad*** window.

If a controller axis is under direct joystick control, its velocity is determined by the joystick position. The maximum velocity for the axes can be set in the fields of the ***Velocity*** column on the ***Axes*** tab card (see "Table Tab Cards" (p. 28) for details).

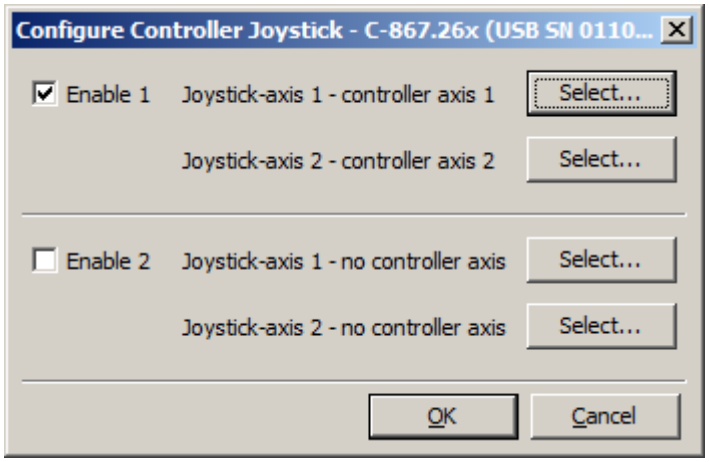


Figure 77: Configure Controller Joysticks window for a C-867.260 two-axis controller supporting two joystick devices each with two axes

NOTICE

Uncontrolled motion!

If no physical joystick device is connected or if the joystick axes are not properly calibrated, uncontrolled motion of the controller axes assigned to that joystick device can occur. Uncontrolled motion can damage the stage or the application.

- Do **not** enable joystick devices with no physical joystick connected.
- Make sure that the joystick is properly calibrated. See "Calibrate Controller Joystick" (p. 141) for details.

In the ***Configure Controller Joystick*** window, each ***Enable*** checkbox is dedicated to one joystick device connected to the controller.

INFORMATION

If the axes of a joystick device are connected to the above-mentioned single-axis controllers (e.g. via Y-cable), they are handled as separate joystick devices with one joystick axis each. In this case, each joystick axis has its own Enable checkbox, and the number next to the checkbox is the device number of the controller to which the joystick axis is connected; set with the DIP switches 1 to 4 on the controller front panel, can be 1 to 16.

Selecting/deselecting the ***Enable*** checkbox for a joystick device enables/disables joystick control for all controller axes assigned to the axes of that joystick device. The assignment of controller axes to joystick axes is displayed next to the ***Enable*** checkbox(es) and can be changed using the ***Select…*** buttons.

All settings not supported by the controller are grayed out. Changed settings are applied not before you press ***OK***. Joystick control remains active until you uncheck the associated ***Enable*** checkbox or exit PIMikroMove.

See the User Manual of your controller for more information regarding joystick control.

4.10 Configure Electronic Gearing

The ***Configure electronic gearing...*** menu item is on the controller menu of supporting multi-axis controllers (e.g. C-843). It gives access to a separate window where you can configure and enable/disable electronic gearing. Using the electronic gearing functionality, it is possible to link a "master" and a "geared" (slave) axis, so that motion of the master automatically entails proportional motion of the slave. See the User manual of your controller for details.

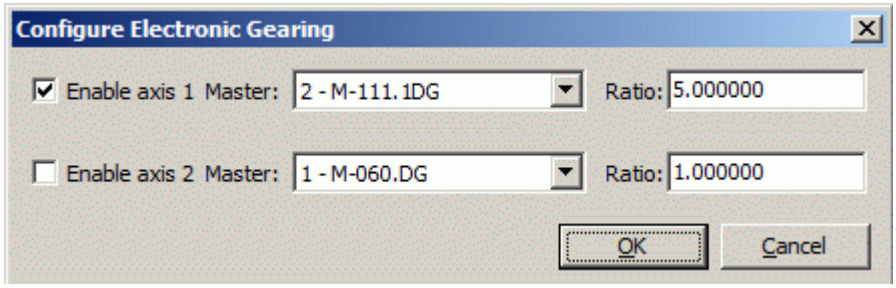


Figure 78: Configure Electronic Gearing window for a two-axis controller

In the ***Configure Electronic Gearing*** window, each line belongs to one controller axis. Using the ***Enable axis n*** checkbox, you can enable electronic gearing for axis n. This means that this axis will be linked as slave to the axis selected in the corresponding ***Master*** list box. In the ***Ratio*** input field you can type the gear ratio (ratio = Master/Slave) to be applied.

INFORMATION

Via electronic gearing you can only link axes which are located on the same controller.

It is not possible to link axes so that they form a circle.

The ***Ratio*** setting is checked automatically upon the activation of electronic gearing. If the slave axis is not able to follow the master axis, you have to adapt the ***Ratio*** value.

In PIMikroMove, slave axes are indicated as shown in the figure below.

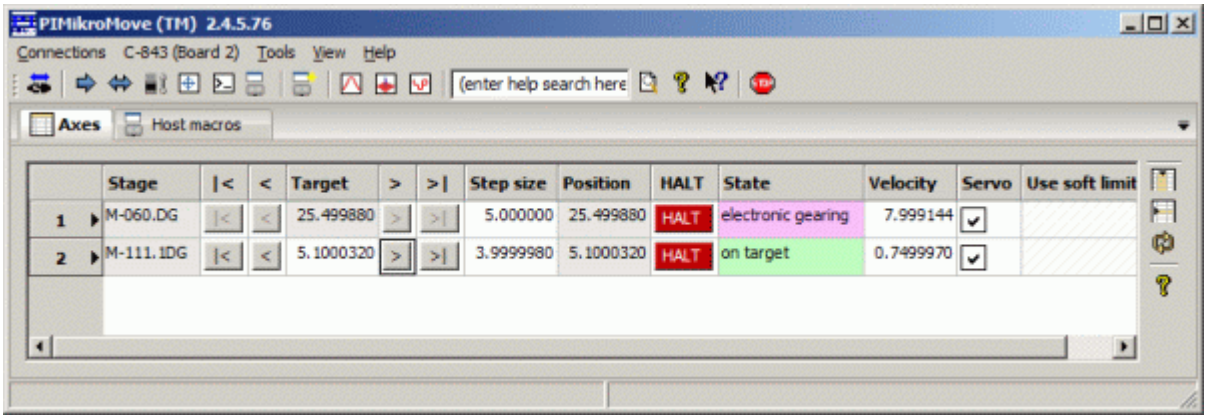


Figure 79: PIMikroMove main window; electronic gearing is activated for axis 1, with axis 2 being the master

The interrelations of electronic gearing and other operations are as follows:

- **Referencing**: Electronic gearing can only be activated for axes which are referenced. Reference moves are not allowed for axes which are involved in electronic gearing (as master or slave), and their referencing mode can not be changed.
- **Move commands**: A slave axis can not be commanded directly by move commands. It is only moved when its master axis moves. When motion is commanded for the master axis, the available travel ranges for master and slave are checked.
- **Joystick control**: Joystick operation is possible for master axes. Slave axes connected to a joystick-controlled master will move correspondingly. Slave axes can not be assigned to joystick axes, i.e. they cannot be controlled directly by a joystick. If joystick control is enabled for an axis, electronic gearing cannot be activated for that axis.

4.11 Device Parameter Configuration

The ***Parameter Configuration...*** menu item is on the controller menu of supporting controllers (e.g. E-517). It gives access to a separate ***Device Parameter Configuration*** window where you can check, modify and save controller parameters. Working with the ***Device Parameter Configuration*** window is helpful e.g. when you want to adapt the controller to your application.

The individual parameters apply to different items, e.g. to the whole system, to individual axes or to individual input/sensor channels and output/piezo channels (for parameter details see the controller User Manual). The parameters available depend on the controller firmware. In the ***Device Parameter Configuration*** window, they are grouped under different headings because listing all parameters at once would be too confusing. In the toolbar, the ***Parameter Group*** selection box permits choosing the group of parameters that will be displayed in the window.

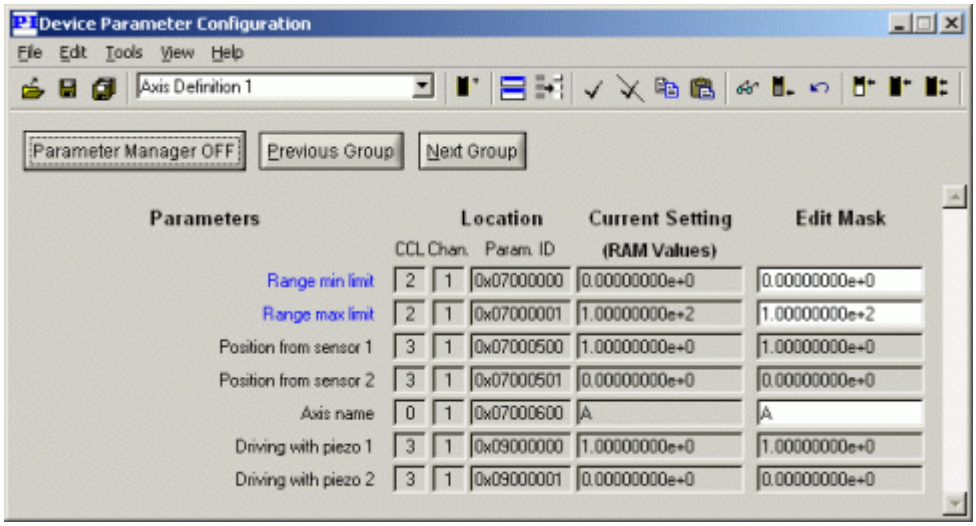


Figure 80: Device Parameter Configuration window for E-517, lists the parameters which define the properties of the first axis

The ***Chan.*** column of the ***Device Parameter Configuration*** window always shows the index of the axis, input/sensor channel or output/piezo channel related to the parameter. These indices are also present in the group headings which are used for parameter group selection. Example: The first axis is named A, but the axis definition parameters are grouped under the ***Axis Definition*** 1 heading, and the ***Chan.*** column contains the value 1 instead of A.

The following buttons are available for parameter list control and parameter management:

- ***Previous Group*** and ***Next Group*** buttons, which permit paging through parameter groups faster than using the ***Parameter Group*** selection box.
- ***Parameter Manager***: indicator/button, when OFF indicates that the ***Parameter Manager*** functions are hidden and the parameter list is displayed as in the figure above. Clicking it reverses the situation, displaying the ***Parameter Manager*** controls and hiding the parameter list. See "Parameter Manager" (p. 150) for details.

The parameter values displayed are those from the controller's volatile memory (RAM values), i.e. the currently active settings.

4.11.1 Load / Edit / Save Parameter Values

NOTICE

Wrong parameter values may lead to improper operation or damage of your hardware!

➢ It is strongly recommended to save the parameter values of the controller to a file on the host PC before you make any changes in non-volatile memory. This way the original settings can be restored if the new parameter settings will not prove satisfactory. See the User manual of the controller for detailed instructions on how to create backup files.

Settings can only be edited in the ***Edit Mask*** column of the ***Device Parameter Configuration*** window. You have several options to handle the parameter values via the content of the ***Edit Mask*** column. Use the ***Parameter Manager*** controls (p. 150) or the items/icons of menu bar and toolbar (p. 152) for the following actions:

- You can load/save parameter values from/to a file on the host PC. Loaded parameter values will be written to the ***Edit Mask***, and only the values in the ***Edit Mask*** column can be saved to a parameter file. To save the current settings or the default settings you have to copy them first to the ***Edit Mask*** and then save to a file.
- To write a new value to the controller, write it in the corresponding ***Edit Mask*** field, select it and then use the appropriate write function. Multiple selections are possible. Only the selected ***Edit Mask*** values will be written to the controller.

  For test purposes, write the value as current setting (i.e. to RAM which is volatile memory). After that, the corresponding ***Current Setting*** value will be displayed in red.

  If you write the value as default setting, it will be saved in the controller's non-volatile memory (EEPROM) where it becomes the power-on default.

- You can compare the parameter value from volatile memory (RAM) with the one from non-volatile memory (EEPROM) using the ***Compare Current Setting with Default Setting*** menu item. Parameters that are different in RAM and EEPROM when you perform this comparison are then displayed in red in the ***Current Setting*** column. This comparison is automatically performed in the ***Device Parameter Configuration*** window whenever the parameters are uploaded from the controller.

Many parameters have a password protection. The protection status of the individual parameters can be seen in the ***CCL*** column of the ***Device Parameter Configuration*** window. You can change only parameters with ***CCL*** values 0 or 1. Parameters with ***CCL*** values larger than 1 can be modified by PI service personnel only. While changing ***CCL*** 0 parameters requires no password, for changing ***CCL*** 1 parameters you have to enter “advanced” in the ***Password*** dialog which pops up automatically.

4.11.2 Parameter Manager

Using the ***Parameter Manager*** controls, you can easily copy the values of selected parameter types, and it is also possible to copy the default settings (EEPROM values) to the ***Edit Mask***, which is not possible with the menu items and toolbar icons described in "Menu Bar and Toolbar" (p. 152).

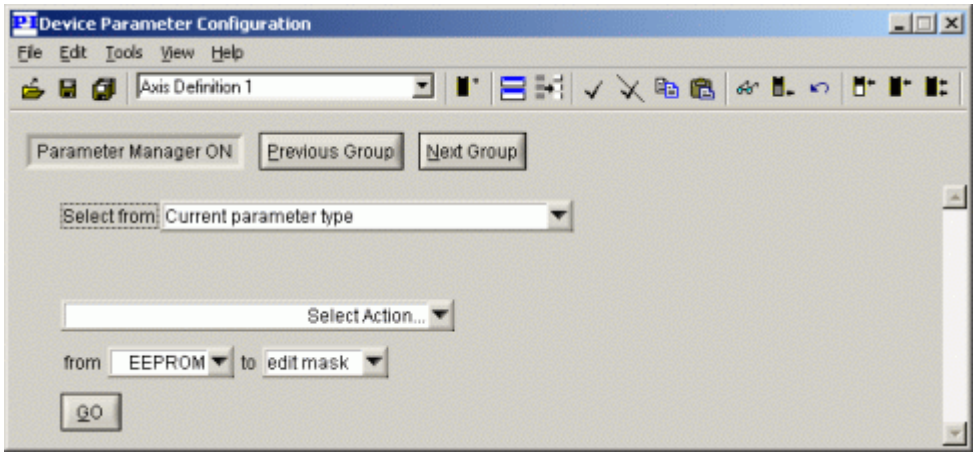


Figure 81: Device Parameter Configuration window for E-517, with the Parameter Manager controls enabled

Parameter Manager controls:

- ***Parameter Manager*** indicator/button when ON, indicates that the ***Parameter Manager*** controls are displayed and the parameter list is hidden; it also permits toggling back to the parameter list.
- In the ***Select from*** selection box, you can select the parameter type to work with, e.g. axis-related parameters or piezo-channel-related parameters.
- Further selection boxes permit defining the action to perform, that is: copying the values of the concerned parameters from EEPROM, RAM or ***Edit Mask*** to RAM or ***Edit Mask***.

The window layout changes depending on what is in the Select from selection box. The ***Current*** parameter type entry refers to that selected in the Parameter Group selection box (in the toolbar).

If the selection is not ***Current parameter type***, the window contains:

- Check boxes for choosing all concerned parameter types to take into account for the action to perform.
- An ***All*** check box that permits taking all the concerned parameter types into account for the action to perform.

If the selection is neither ***Current parameter type*** nor ***All parameter types***, the window also contains a selection box permitting you to choose all or a single axis / piezo channel / sensor channel / wave generator, depending on the action to perform.

The selected action will be performed on the corresponding parameters when pressing the ***Go*** button.

An example of the ***Parameter Manager*** window configuration for copying all sensor-channel-related parameters from EEPROM to the ***Edit Masks*** is shown in the figure below.

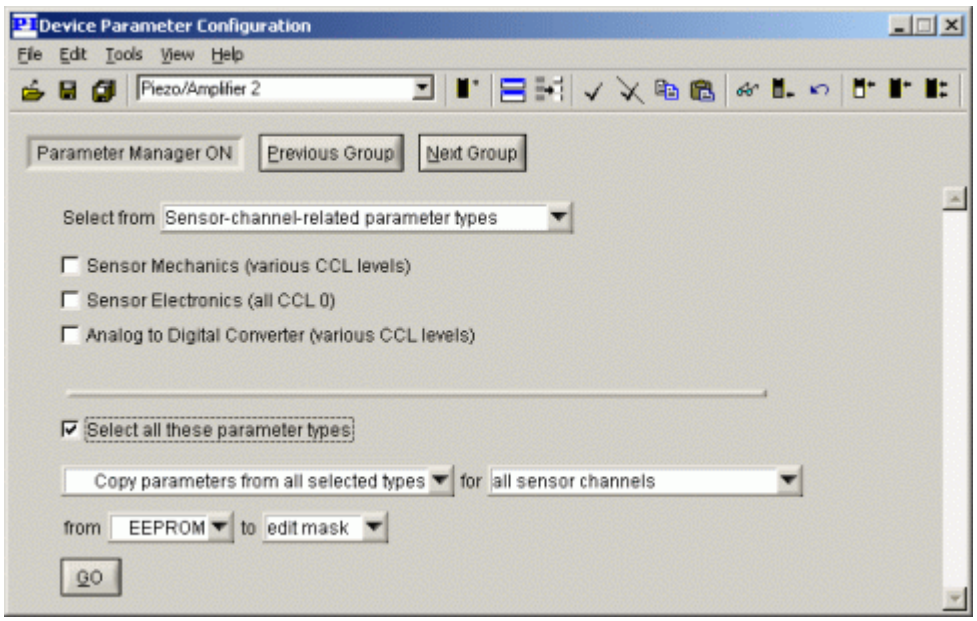


Figure 82: Device Parameter Configuration window for E-517, all sensor-channel-related parameters are to be copied from non-volatile memory to Edit Mask

4.11.3 Menu Bar and Toolbar

Most of the menu items are available on the toolbar, and all icons are available on the menus.

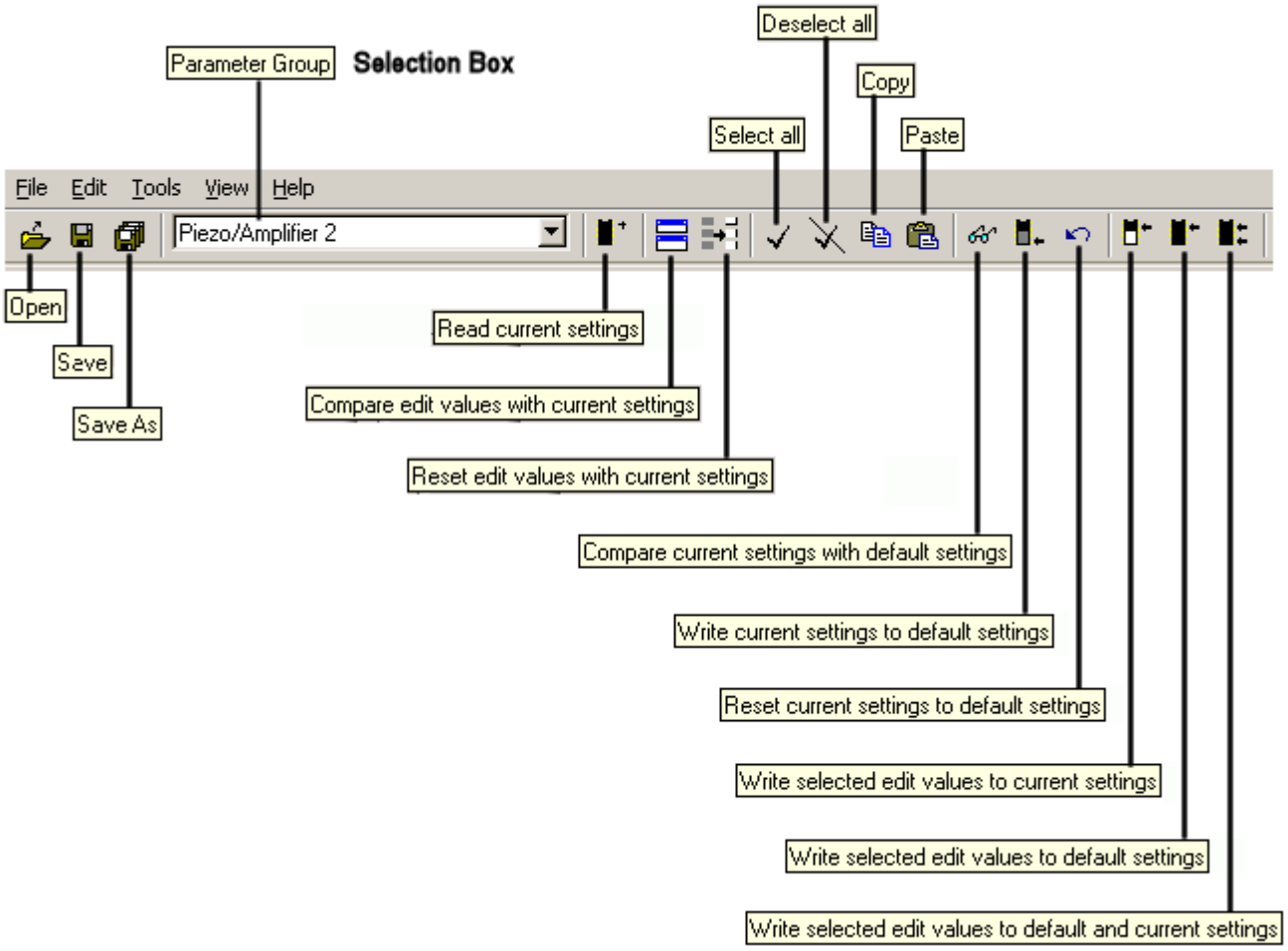


Figure 83: Menu bar and toolbar of the Device Parameter Configuration window

The items of the *File* menu are:

- ***Load parameter file*** permits loading all parameters from a parameter file (with the extension .pam) and writing them in the corresponding ***Edit Masks*** field.
- ***Load and select*** permits loading all parameters from a parameter file (with the extension .pam) and writing them in the corresponding ***Edit Mask*** fields. The corresponding ***Edit Mask*** values will be selected automatically.

- ***Save Edit Values As*** and ***Save Edit Values*** permit saving all parameters from the ***Edit Masks*** into a parameter file.
- ***Close*** permits closing the ***Device Parameter Configuration*** window.

The items of the *Edit* menu are:

- ***Copy*** permits copying the selected ***Edit Mask*** parameter values from the current parameter group to the clipboard.
- ***Paste*** permits writing the parameter saved in the clipboard to the same position in the ***Edit Mask*** of the current parameter group. The parameter group where the parameters have been copied from and the one where the parameter will be pasted must be compatible. If the parameter groups are not compatible a corresponding message will pop up.
- ***Select all*** and ***Deselect all*** permit selecting or deselecting all ***Edit Mask*** values. In both cases, this applies either to the current parameter group or to all parameters, depending on what you select in the associated submenu.

  Note: Individual parameter edit values can be selected by clicking it with the left mouse button or deselected by clicking it with the right mouse button. Selected parameters are displayed in white in a grey background. Only selected parameters will be used in write operations.

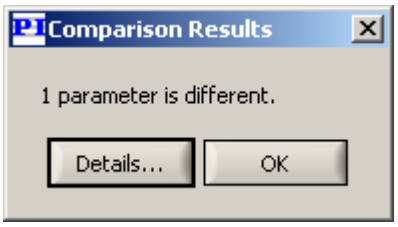
- ***Search…*** permits searching for a parameter via a part of its name, value or ID.
- ***Search Invalid Values…*** permits detecting invalid values from the ***Edit Mask*** and ***Current Setting*** fields (e.g. empty fields or—if present—"invalid" items from value selection boxes).

The results of ***Search…*** or ***Search Invalid Values…*** can be saved in a *.txt or a *.csv file.

The items of the *Tools* menu are:

- ***Read Current Setting*** permits reading the RAM values from the controller for either the current parameter group or for all parameters, depending on what you select in the associated submenu:
- ***Write selected Edit Values*** permits writing values in the Edit Mask to the controller RAM or EEPROM or to both, depending on what you select in the associated submenu.
- ***Reset Edit Values with Current Setting*** permits resetting the values in the ***Edit Masks*** to the values from the ***Current Setting*** column. This applies either to the parameter that is currently activated with the cursor or to all parameters from the current parameter group, or to all parameters, depending on what you select in the associated submenu:

- ***Compare Edit Values with Current Setting…*** permits comparing the values in the ***Edit Mask*** with the RAM values for all parameters. At the end of this action, a dialog gives you access to precise information about the parameters that differ.



  Pressing the ***Details…*** button opens a dialog box with more information about the first parameter that differs and a ***Next*** button for moving on to the next parameter (if there is any).

  As you go through the different parameters, the parameter group of the respective parameter is selected in the ***Parameter Group*** selection box in the ***Device Parameter Configuration*** window, and the parameter in question is highlighted (selected), so that it can easily be found.

  The results can be saved in a *.txt or a *.csv file.

- ***Compare Current Setting with Default Setting*** permits comparing RAM and EEPROM values for either the current parameter group or for all parameters, depending on what you selected in the associated submenu. The concerned current setting values are displayed in red if they differ from the corresponding default setting value.
- ***Reset Current Setting to Default Setting*** permits resetting the RAM values to the EEPROM values for all parameters.
- ***Write Current Setting to Default Setting*** permits writing the RAM values to the EEPROM for all parameters.

The ***View*** menu contains only one item, ***Axis Matrices***. It opens the ***Axis Matrices*** window where you can check (and edit, if supported by the controller) the assignment of sensor channels and piezo channels to the motion axes of the controller.

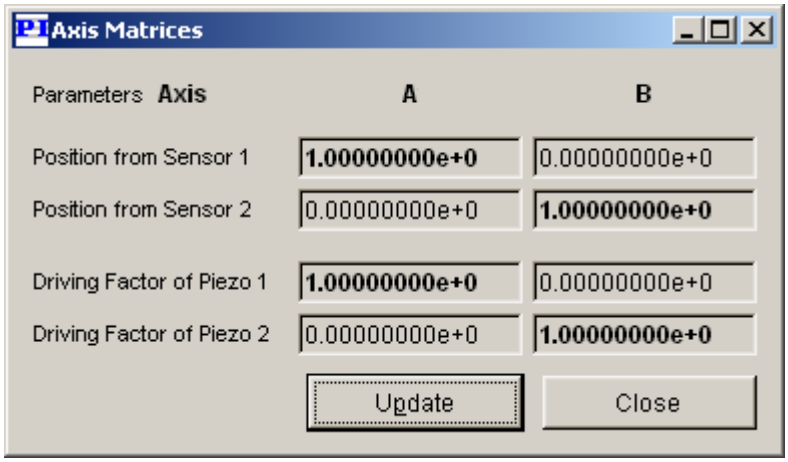


Figure 84: Axis Matrices window for E-517 (the third axis / sensor channel / piezo channel are deactivated and hence not shown)

The ***Help*** menu also contains only one item, ***Version Info***. It pops up a window displaying the identification string obtained from the controller, including firmware version number and—if supported for the controller— the versions of the underlying drivers and libraries.

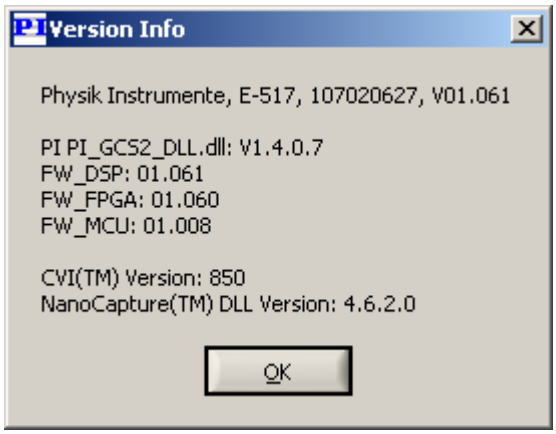


Figure 85: Version Info window for E-517

4.12 Configure Interface

The ***Configure interface...*** menu item is on the controller menu of supporting controllers (e.g. C-702). It gives access to a separate ***Configure Interface*** window where you can specify the settings for the controller side of the communications interface.

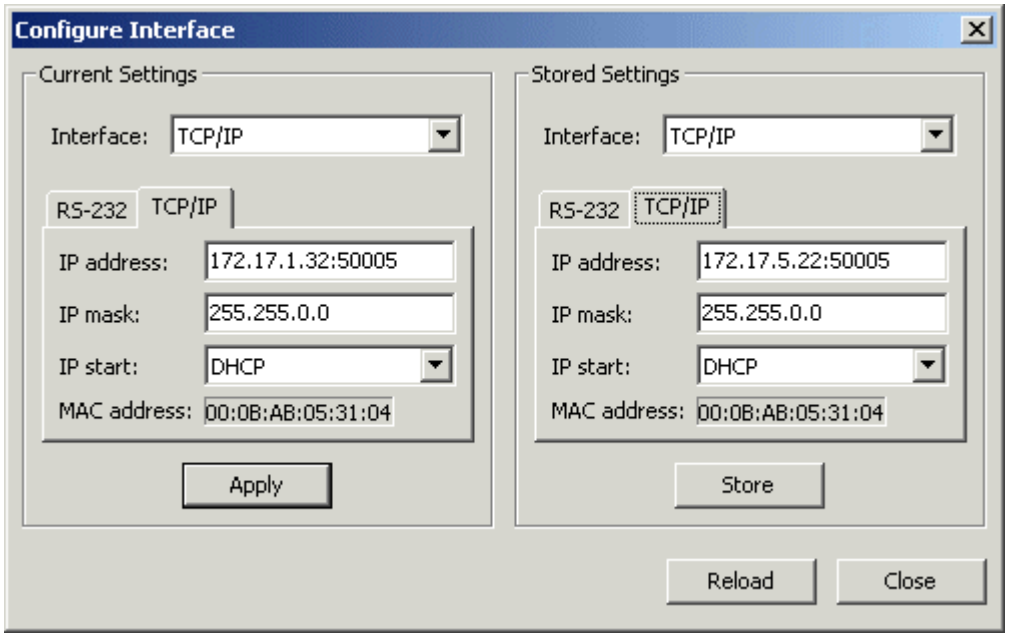


Figure 86: Configure Interface window for C-702

Depending on the controller, you can specify the current settings and/or the power-on default settings.

A set of default communication parameters is stored on the controller. These values are loaded into the controllers RAM on power-up or reset and become the active settings. The default settings can be read and changed in the ***Stored Settings*** pane. To keep changes click the ***Store*** button. They become active only with the next power-up or reset.

The current active settings can be changed in the ***Current Settings*** pane—the new settings become active immediately when you click the ***Apply*** button. The host PC may need to change its interface configuration to maintain communications—close the connection and reconnect with the new settings. When the controller is powered down, settings made in the ***Current Settings*** pane are lost.

The settings used for the current active communications and the last-stored default settings can be read back to both panes by clicking the ***Reload*** button.

With ***Close*** the window is closed.

See the controller User Manual for communications details and default settings.

See "Connections Menu" (p. 13) and "Connect Controller" (p. 171) for how to handle communications connections in ***PIMikroMove***.

4.13 Hexapod Platform Settings

For every Hexapod system (controller or simulation) connected to PIMikroMove, a separate ***Hexapod Platform*** window by default is docked in the main window. You can reopen the ***Hexapod Platform*** window using the ***Show Hexapod Platform Setting*** item of the controller menu.

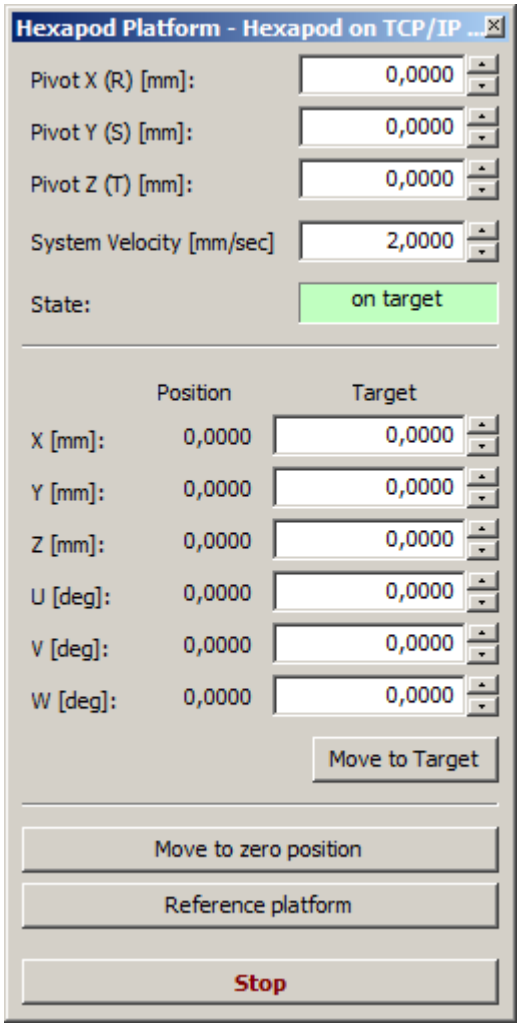


Figure 87: Hexapod Platform window

INFORMATION

During a motion of the Hexapod platform, all controls of the ***Hexapod Platform*** window except for the ***Stop*** button are disabled.

The controls and displays of the ***Hexapod Platform*** window are not accessible and not updated as long as the Command entry (p. 75) window is open.

In the ***Hexapod Platform*** window, you can configure, start and stop motions of the Hexapod platform as follows:

- Change the pivot point coordinates if necessary:

  Enter a new value in a ***Pivot …*** field and press ENTER or move the cursor to another field. This sends the value to the controller, and the text color changes from blue to black again.

  The pivot point can only be changed if the Hexapod platform is not tilted (U, V and W must be 0). See the Hexapod User Manual for more information.

- Set the system velocity if necessary:

  Enter a new value in the ***System Velocity*** field and press ENTER or move the cursor to another field. This sends the value to the controller, and the text color changes from blue to black again. If a new value is not accepted by the controller, the old value reappears in the field.

- Start motions to variable target positions:

  Enter target positions for the axes of the Hexapod platform in the corresponding ***Target*** fields and start the motion using the ***Move to Target*** button.

  If a node of the calculated trajectory or the target position cannot be reached, the motion is not started, and an error message is displayed.

  The ***Target*** fields and the ***Move to Target*** button are deactivated as long as the Hexapod platform has not performed a reference move successfully.

- Start motion to the zero position (X = Y = Z = U = V = W = 0):

  Start the motion using the ***Move to zero position*** button.

  The ***Move to zero position button*** is deactivated as long as the Hexapod platform has not performed a reference move successfully.

- Start a reference move:

Start the reference move of the Hexapod platform using the ***Reference platform*** button. The reference move can also be started via the ***Start up axes…*** item of the controller menu or the ***Axis*** menu. See also "Start Up Axes" (p. 179).

- Stop motions:

  Using the ***Stop*** button, you can stop motions started by the ***Move to Target*** or ***Move to zero position*** buttons.

You can monitor the Hexapod platform as follows:

- The current state of the Hexapod platform is shown in the ***State*** field of the ***Hexapod Platform*** window.
- The current position of the Hexapod platform is shown in the ***Position*** column of the ***Hexapod Platform*** window.
- The pivot point and motions of the Hexapod are visualized on the ***Hexapod 3D View*** tab card (p. 36).

An ***Axis*** menu (p. 34) for all axes of the Hexapod platform can be displayed from the ***Hexapod Platform*** window by right-clicking on any free area in the window. The ***Clear Error*** item of this menu can be helpful when the ***State*** field indicates an error.

INFORMATION

By default, the ***Servo*** checkbox is hidden in the ***Hexapod Platform*** window since switching the servo mode off can cause damage to the Hexapod. If you want to show the ***Servo*** checkbox anyway in the ***Hexapod Platform*** window:

1. Open the ***Preferences*** window (p. 100) using the ***Tools > Preferences…*** menu sequence in the main window.
2. Uncheck the corresponding checkbox (***Hide Servo control in Hexapod platform settings.***).
3. Close the ***Preferences*** window by clicking ***OK***.
4. Close the ***Hexapod Platform*** window. This is necessary because the changed setting becomes active not until the ***Hexapod Platform*** window is reopened.
5. Reopen the ***Hexapod Platform*** window using the ***Show Hexapod Platform Setting*** item of the controller menu.

By default, the ***Hexapod Platform*** window is docked to the left border of the main window. You can undock and dock it again by dragging it with the left mouse button pressed (see also "Main Window" (p. 9)).

4.14 Embedded Scan and Align Algorithms

The ***Show Embedded Scan Window*** menu item is on the controller menu of supporting controllers (e.g. Hexapod controllers or Hexapod system simulations). It gives access to the special scan and align algorithms which are implemented in the controller firmware. The supported algorithms and the access to them depend on the GCS syntax version of the controller:

- GCS syntax version 1.0:
  Show Embedded Scan Window opens the ***Embedded Scan & Align Algorithms*** window. In this window, you can select the algorithm to be used. The content of the window depends on the selected algorithm.
- GCS syntax version 2.0:
  With the ***Show Embedded Scan Window*** menu item you have to select the algorithm to be used. For each of the selected algorithms, a separate window opens. Multiple windows can be open at the same time.

See the User Manual of your Hexapod system or of your Hexapod controller for the supported algorithms and a more detailed description.

INFORMATION

There are algorithms for line scans and for area scans. Although with line scans only a distance to be scanned along an axis must be given, the corresponding field is labled ***Area*** for compatibility reasons. With area scans, the ***Area*** field in fact gives the side length of the square area to be scanned.

Examples for GCS syntax version 1.0:

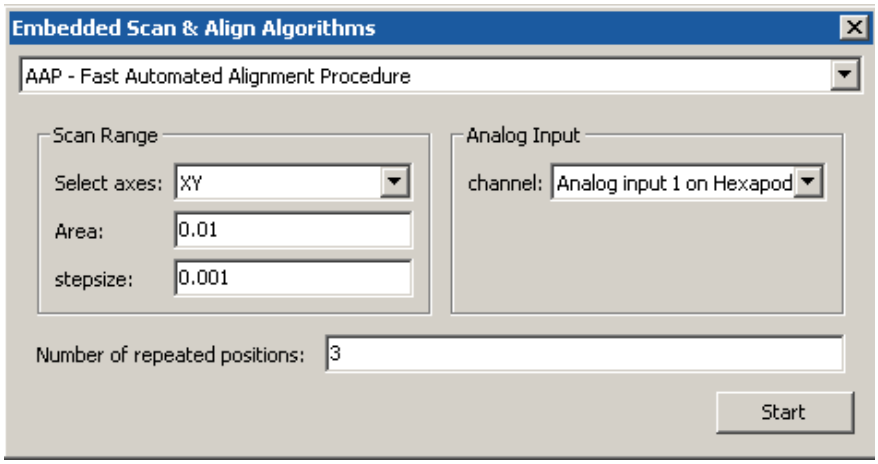


Figure 88: Embedded scan & align window with panes for the AAP algorithm

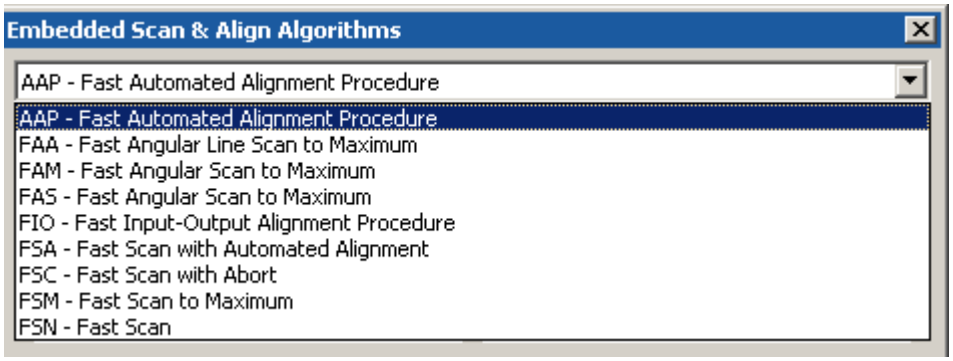


Figure 89: List of supported algorithms for F-206 Hexapod

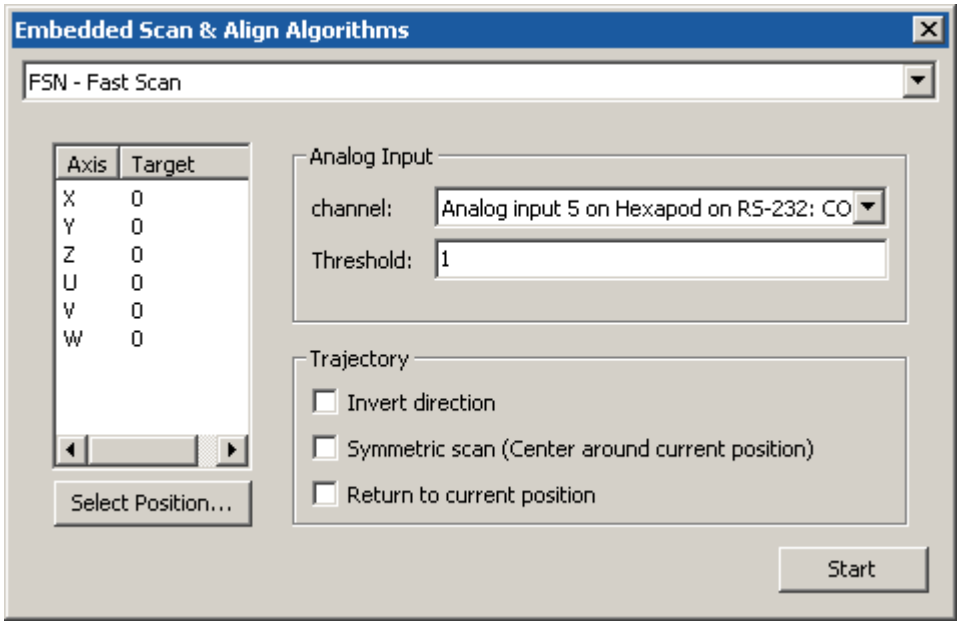


Figure 90: Embedded scan & align window with panes for the FSN algorithm

Examples for GCS syntax version 2.0:

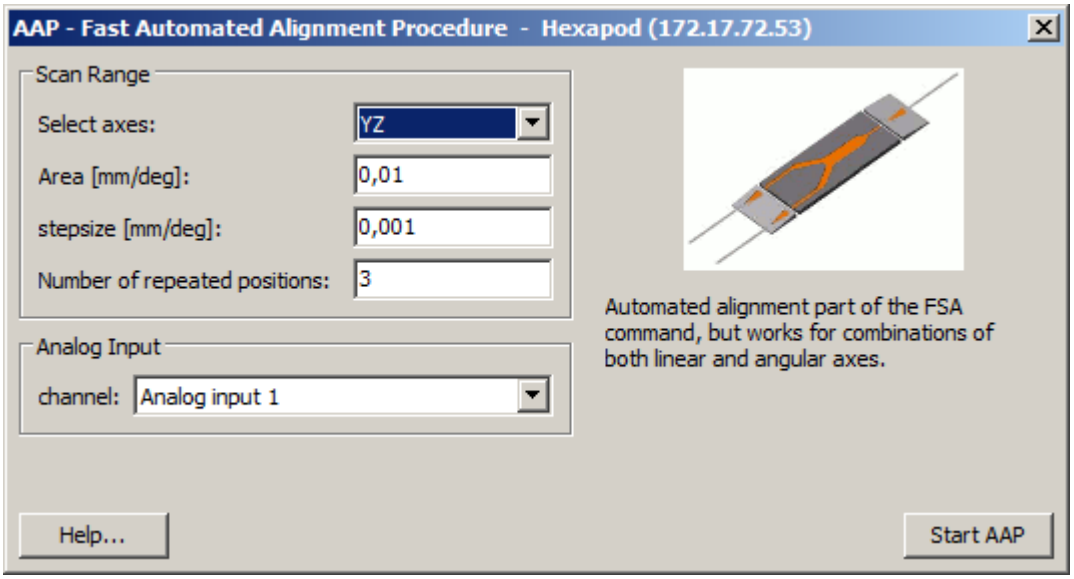


Figure 91: GCS syntax version 2.0: window for AAP alignment algorithm

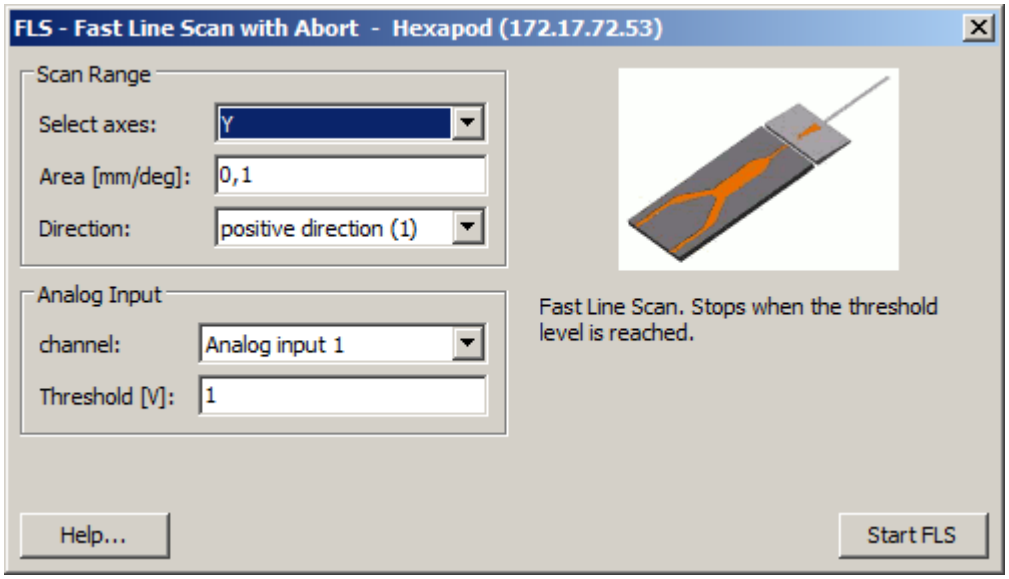


Figure 92: GCS syntax version 2.0: window for FLS line scan algorithm

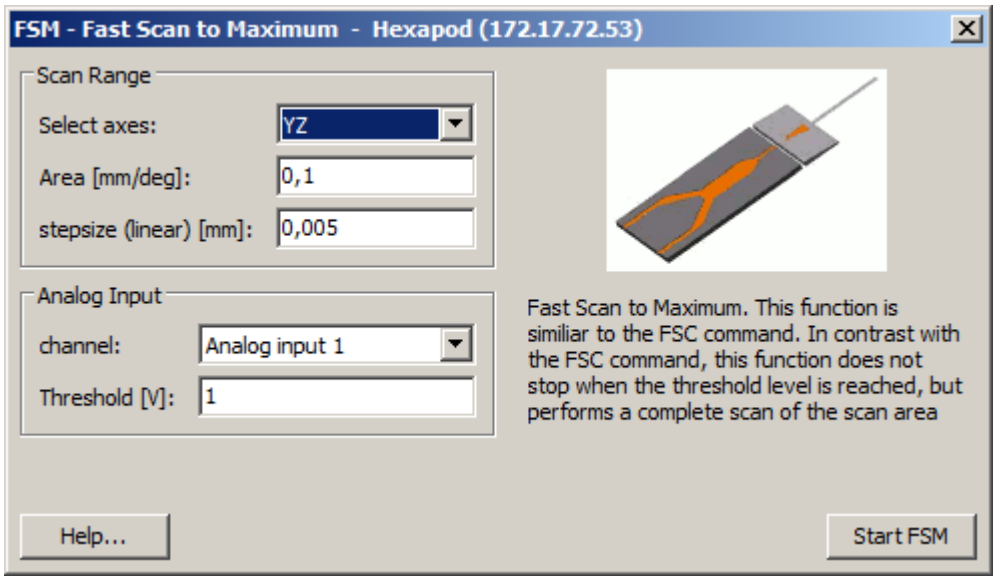


Figure 93: GCS syntax version 2.0: window for FSM area scan algorithm

4.15 Hexapod Service Tools

The ***Show service tools…*** menu item is on the controller menu of supporting controllers (e.g. Hexapod controllers).

Use the ***Show service tools…*** menu item only if failure of the Hexapod system occurs. It gives access to the ***Hexapod Service Tools*** window where you can perform the following tests for the single Hexapod struts:

- **Impulse response measurement** of a Hexapod strut. Failures may be indicated by large overshot of the actual position, by oscillations of the actual position or by large differences between actual position and commanded position.
- **Measurement of the motor output during one spindle revolution**, the measurement result is given in percent of the maximum motor output and makes it possible to evaluate the control reserve. Failures may be indicated if the motor output value continuously exceeds 80 % or if it shows an abnormal high value at a certain point of the travel range.

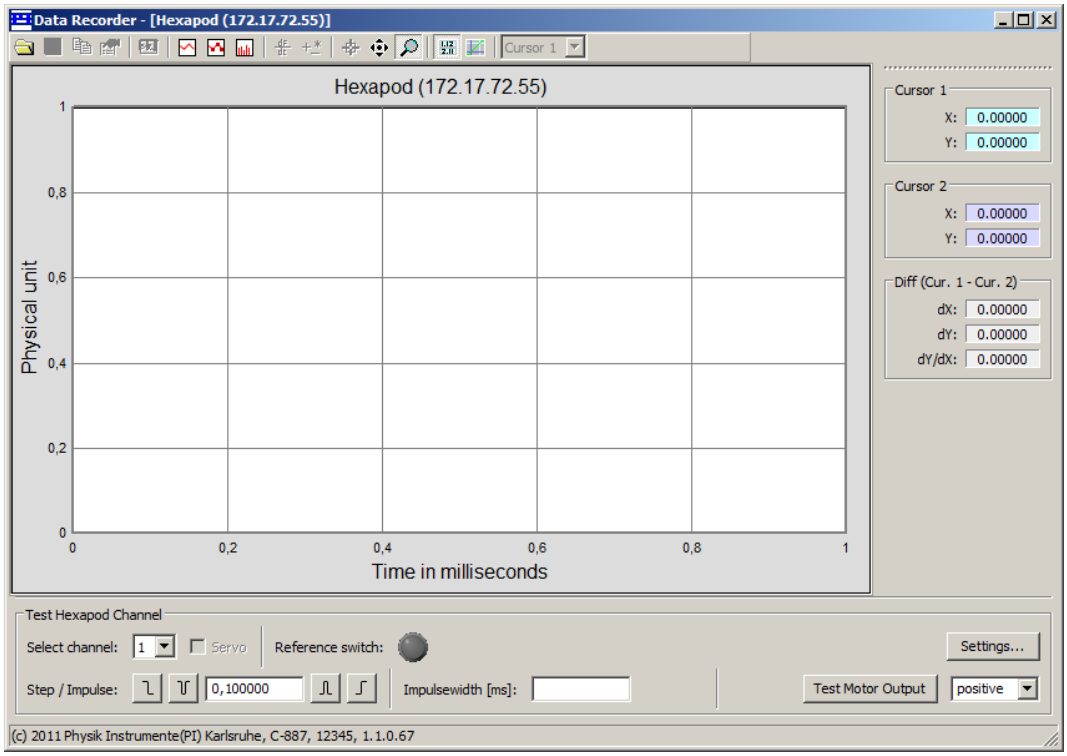


Figure 94: Hexapod Service Tools window; in older versions of PIMikroMove the window may be named "Data Recorder"

4.15.1 Performing Strut Tests

NOTICE

Damage due to collisions during the strut test!

During the strut test the Hexapod moves in an unpredicatble way. There is no collision check even if a configuration for avoiding collisions was stored on the controller using the PIVeriMove software. Soft limits set with the NLM and PLM commands for the moving platform of the Hexapod are ignored during a strut test.

Thus colllisions between the Hexapod, the load to be moved and the environment are possible. Collisions can damage the Hexapod, the load to be moved and the environment.

- ➢ Make sure that no collisions between the Hexapod, the load to be moved and the environment are possible during a strut test.
- ➢ Do not place any objects in areas where they can get caught by moving parts during a strut test.
- ➢ Observe the Hexapod during a strut test to be able to take actions if a fault occurs.

NOTICE

Damage due to unwanted position changes during a strut test!

During a strut test the Hexapod strut can reach a limit switch. Thus the servo mode for the axes of the moving platform of the Hexapod is automatically switched off.

If the actual load on the Hexapod exceeds the maximum holding force which is based on the self-locking of the actuators, switching off the servo mode for the axes of the moving platform of the Hexapod can cause unwanted position changes of the Hexapod.

Thus colllisions between the Hexapod, the load to be moved and the environment are possible. Collisions can damage the Hexapod, the load to be moved and the environment.

- ➢ Before you start a strut test, make sure that the actual load on the moving platform of the Hexapod does not exceed the maximum holding force which is based on the self-locking of the actuators.

INFORMATION

Recommendations for performing strut tests:

- If possible, command the Hexapod to the reference position before a strut test to have the largest possible travel range in the positive and negative direction for each Hexapod strut.
- Set the velocity to a suitable value with the VLS command before a strut test.

INFORMATION

Instead of the impulse response you can also measure and evaluate a step response of the Hexapod strut. But after performing the step the Hexapod strut does not move back to the starting position. If multiple steps are performed, the Hexapod strut can thus reach a limit switch. When the limit switch is reached the servo mode is switched off for the moving platform of the Hexapod.

Proceed as follows for strut tests:

1. Select the ***Show service tools… item*** of the controller menu.

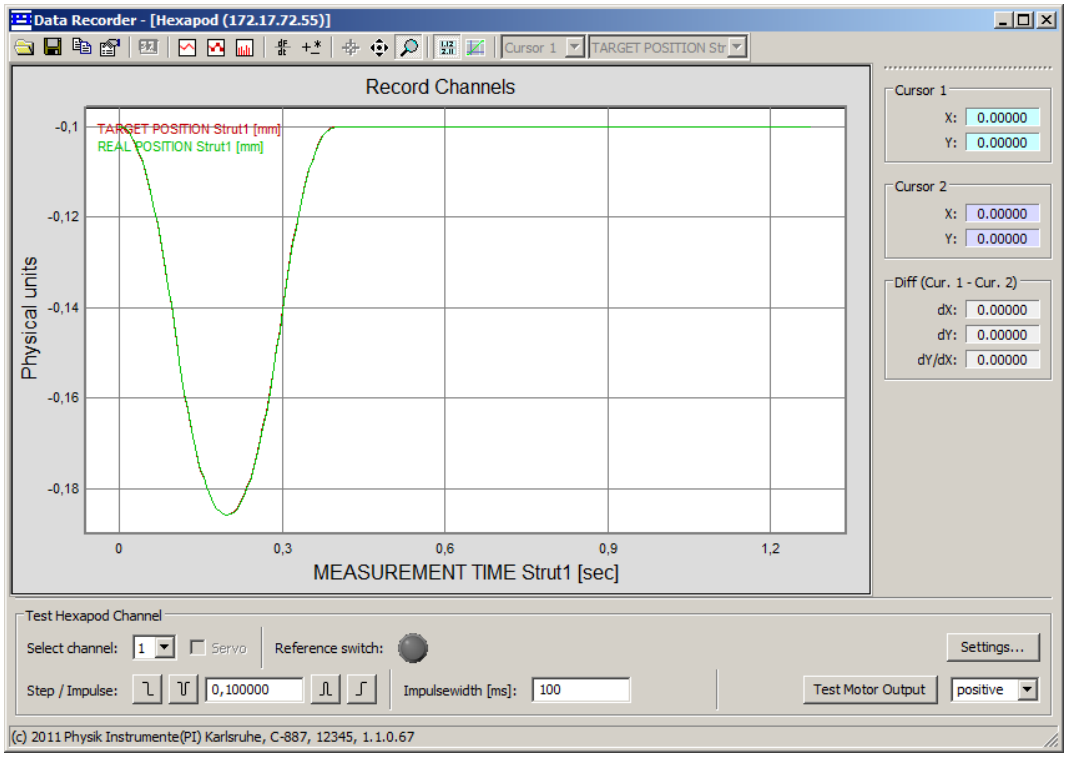
   A window opens with a message saying that damage is possible during the usage of the ***Hexapod Service Tools*** window.

2. Click ***Show service tools*** in the message window to open the ***Hexapod Service Tools*** window.

3. In the ***Hexapod Service Tools*** window, measure the impulse response for a Hexapod strut:

   a) In the ***Select channel*** field, select the strut for which the impulse response is to be measured.
   b) In the ***Impulsewidth [ms]*** field, enter the pulse width of the impulse in milliseconds. Recommended value: 100 ms.
   c) Click on the  or  button to start the impulse in positive or negative direction.
   Starting the impulse also starts the recording of the current position and the target position of the Hexapod strut. The Hexapod strut moves according to the given pulse width of the impulse and moves back to the starting position afterwards.
   d) Evaluate the impulse response using the curves in the graphics pane of the ***Hexapod Service Tools*** window (see figure below).

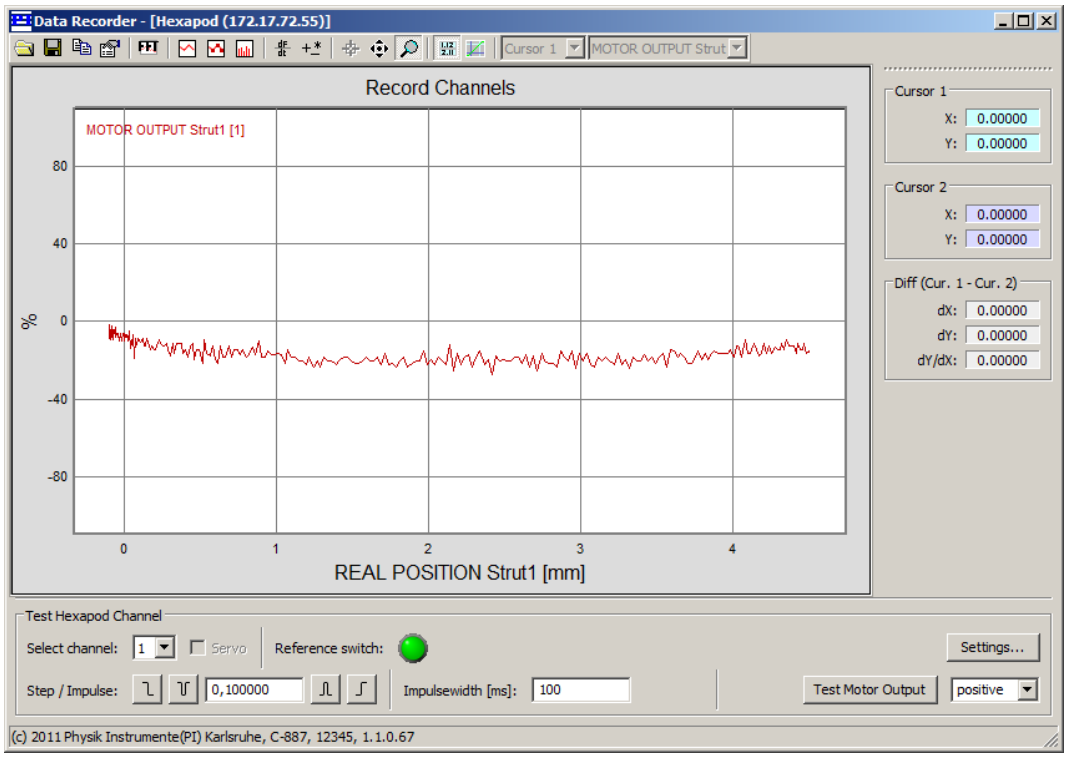
A step response measurement can be started using the  or  button. The step size is given by the entry in the field between the buttons for the impulse start.



4. In the ***Hexapod Service Tools*** window, measure the motor output during one spindle revolution of a Hexapod strut:

   a) In the ***Select channel*** field, select the strut for which the motor output is to be measured.
   b) In the field to the right of the ***Test Motor Output*** button, select the direction (positive or negative) for the spindle revolution.
   c) Click on the ***Test Motor Output*** button to start one spindle revolution of the Hexapod strut in the selected direction.
   Starting the spindle revolution also starts the recording of the current motor output and the current position of the Hexapod strut. The Hexapod strut does not move back to the starting position after the spindle revolution.
   d) Evaluate the recorded motor output using the curve in the graphics pane of the ***Hexapod Service Tools*** window (see figure below).

If multiple spindle revolutions are performed in the same direction, the Hexapod strut can reach the limit switch. When the limit switch is reached the servo mode is switched off for the moving platform of the Hexapod.



5. If the test results indicate failure of the Hexapod:
   a) Do not operate the Hexapod system anymore.
   b) Send the test results to the PI customer service department for diagnosis (mailto:info@pi.ws).

4.15.2 Configuration of Data Recording and Graphics Pane

When you open the ***Hexapod Service Tools*** window, a suitable data recorder configuration is preset which cannot be changed. You can only change the following settings in a separate window which can be opened with the ***Settings…*** button:

- ***Record Rate*** field: Specify the number of servo-loop cycles before each next data point is recorded, must be an integer value ≥ 1. The larger the value, the longer the time period covered by recording. The duration of a recording can be calculated as follows:
  Rec. Duration = Servo Cycle Time * Record Rate * Number of Points
  where *Servo Cycle Time* and *Number of Points* (length of the data recorder table) depend on the controller, see controller User Manual
  Note that the ***Record Rate*** is a global data recorder setting which will not only affect measurements in the ***Hexapod Service Tools*** window.
- ***# of data points*** field: You can set the number of data points to be read from the controller for the display.

You can configure the graphics pane of the ***Hexapod Service Tools*** window via its toolbar. A short description will be displayed if the mouse cursor hovers over a toolbar button. For detailed descriptions of the toolbar buttons see "Configure Graphics Pane" (p. 134).

4.16 Log Window

The ***Log window...*** item is on every controller menu and opens a controller-specific ***Log...*** window. In this window, you can monitor the commands which are sent to the controller when you use the controls of PIMikroMove. This is a good way to see what commands are required for certain actions and to learn the command syntax.

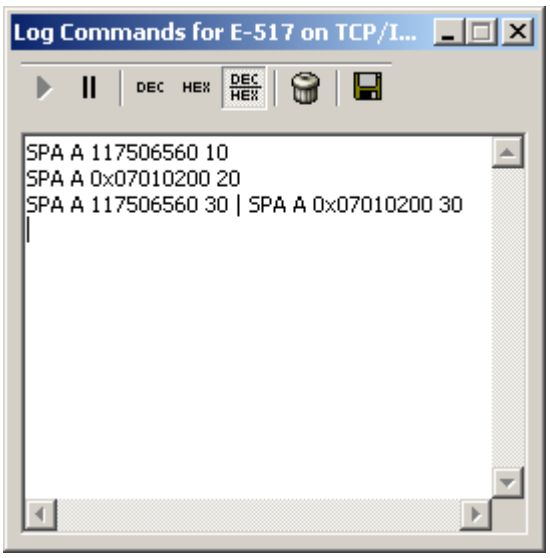


Figure 95: Log window for E-517

With the ***Start*** and ***Pause*** icons, you can activate and deactivate logging. Furthermore, you can clear the window content or save it to a text file on the host PC.

The ***DEC***, ***HEX*** and ***DEC|HEX*** buttons modify the number format for parameter IDs used with SPA.

INFORMATION

The following are not shown in the Log window:

- Status requests sent by PIMikroMove
- Response from the controller
- Commands sent from the ***Command entry*** window of PIMikroMove

5 Start up Controller—Details

This section explains how to connect a controller or set of networked controllers to PIMikroMove and how to start up the connected stages. The ***Start up Controller*** dialog is shown at start of PIMikroMove, and can later be opened by a number of different menu sequences.

PIMikroMove guides you through the startup procedure which typically comprises multiple steps. It depends on the controller which of the following steps have to be performed:

- All controllers:
  Connect controller (p. 171)
- Controller which support loading settings from a stage database:
  Select connected stages (p. 174)
- Hexapod controllers with GCS syntax version 1.0:
  Configure Hexapod (p. 177)
- E-517 interface and display modules:
  Start up E-517 (p. 178)
- System whose axes have to be initialized (e.g. referenced) before normal use:
  Start up axes (p. 179)

If the hardware configuration remains unchanged (controller, communication interface, connected stages), you can automate the startup procedure using the AutoConnect functionality (p. 183). AutoConnect means that you define the startup steps once—thereafter you will no longer have to deal with them because the program performs them automatically.

If you start PIMikroMove from a command line or via batch processing, you can choose some options for the program start, e.g. use a customized configuration file instead of the default file. See "Command Line Options" (p. 186) for more information.

5.1 Connect Controller

You can show the ***Connect Controller*** dialog at any time with ***New connection*** on the toolbar or with the ***Connections > New…*** menu sequence. This way, you can connect additional controllers to PIMikroMove. It is not necessary to close an existing connection before additional controllers can be connected since PIMikroMove is designed to handle multiple connections. For example, the axes of all connected controllers will be listed in the main window.

5.1.1 General

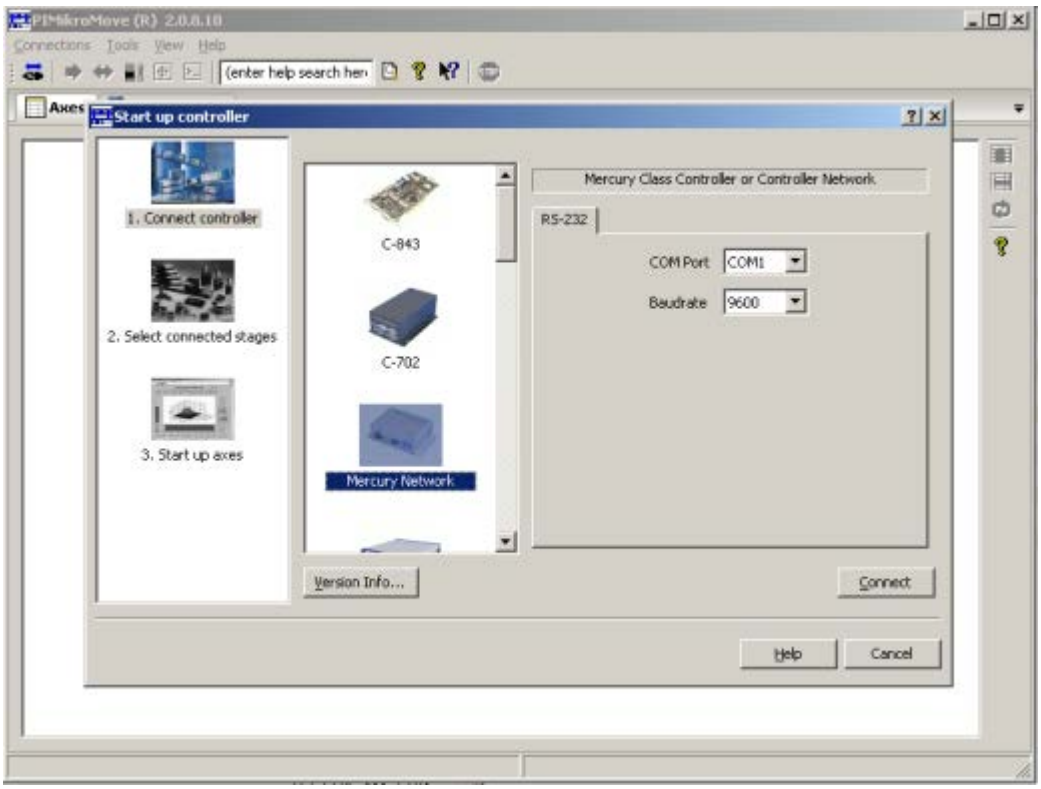


Figure 96: Start up Controller window with Connect Controller selected

The list box in the middle of the window shows symbols for all controllers known to the host system. At first, select the controller you want to use from that list. Then, select the tab card for the interface you want to use in the right pane of the window. On that tab card, choose the correct interface settings (baud rate, COM port, IP address, …). When all settings are done, click ***Connect*** to establish the connection. If the connection is established successfully, the next step of the ***Start up controller*** window will be shown automatically.

Please note that not all of the interfaces shown may be physically present on your system.

INFORMATION

If the controller is connected via a USB port, this USB interface might appear as an additional COM port in the port-selection list.

With ***Version info*** you can show the versions of the software components found by PIMikroMove. This information can be helpful in diagnosing problems.

(PIMikroMove searches once at startup for controller software. If you want to search again or show the version info at a later time, use the ***Connections > Search for controller software menu sequence.***)

5.1.2 Daisy Chain Connection

Some controllers can be connected to PIMikroMove via a daisy chain of multiple devices. See the controller User manual for hardware connection details.

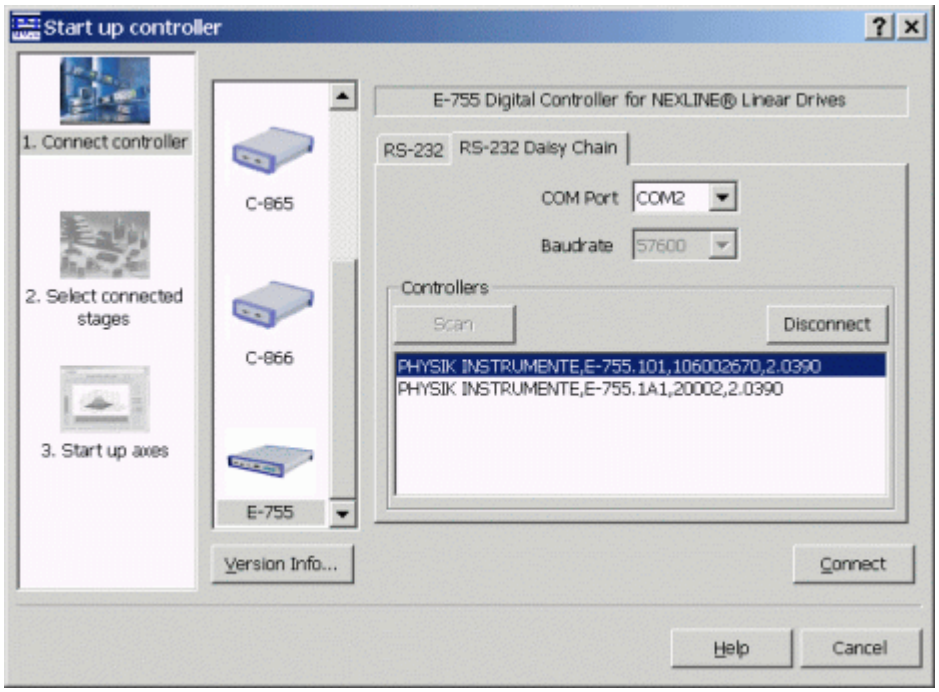


Figure 97: Start up controller screen after clicking Scan on RS-232 Daisy Chain tab card with 2 controllers on the network

If you want to connect one or more devices which are part of a daisy chain, use one of the ***… Daisy Chain*** tab cards (availability depends on controller):

- ***RS-232 Daisy Chain***:
  Set the COM port to use on the host PC and the baud rate. Then click the ***Scan*** button to find all controllers in the daisy chain. Select one controller from the list and click the ***Connect*** button.
- ***USB Daisy Chain***:
  Click the ***Scan*** button to find all controllers in the daisy chain. Select one controller from the list and click the ***Connect*** button.

To connect other controllers on the daisy chain, use the main window ***Connections > New*** menu item to reopen the Start up controller window and, on the corresponding ***Daisy Chain*** tab card, select the desired controller and click ***Connect***.

To disconnect one of the controllers listed on a ***… Daisy Chain*** tab card, select it in the list and click ***Disconnect***. If the ***Disconnect*** button is dimmed, use the items of the ***Connections*** menu in the PIMikroMove main window to close the connection.

If you want to connect a controller with daisy chain feature as stand-alone device, do not use the ***… Daisy Chain*** tab card(s).

5.2 Select Connected Stages

You can manually open the ***Select connected stages*** window at any time with the ***Select stages*** item on the controller menu of the corresponding controller or controller network.

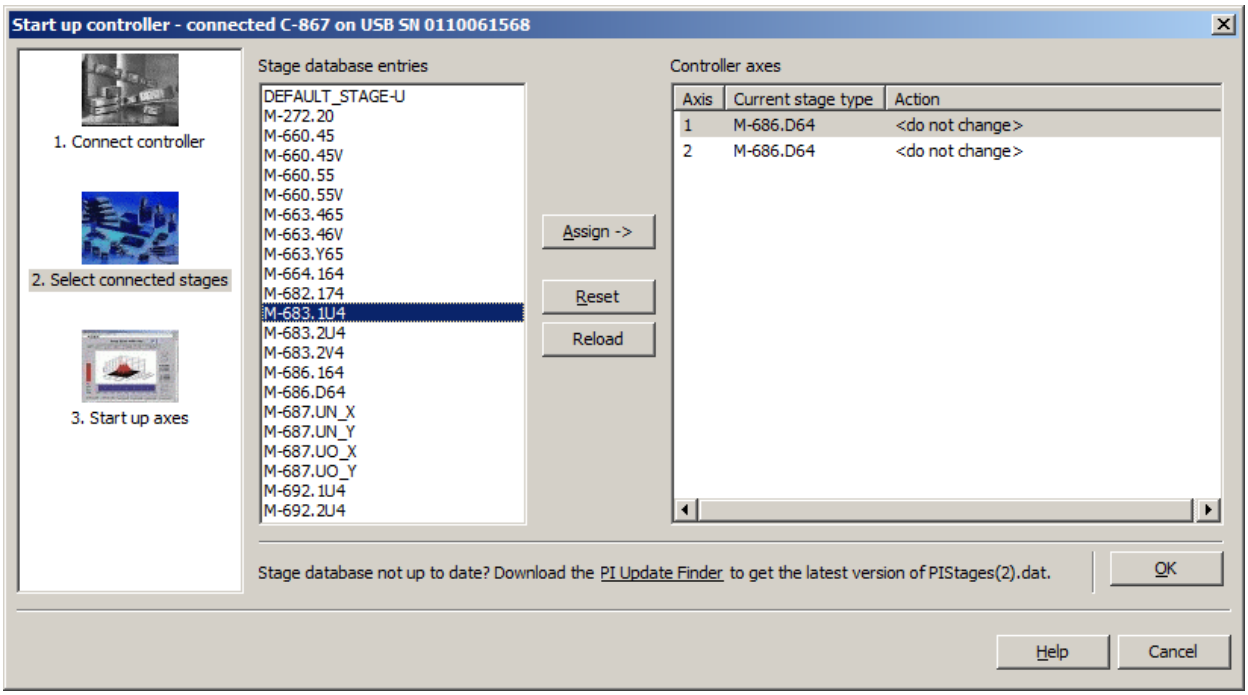


Figure 98: Startup Controller window with the Select connected stages step

In this window you can select the types of the connected stages for the various axes and hence load the proper settings from a stage database, provided that the controller supports this feature. Two lists are provided:

- The list of all stage database entries suitable for the controller (left); is the sum of entries in the PIStages2.dat, the controller-specific user-stages dat file and any M-xxx.dat or N-xxx.dat files (single-stage databases for custom mechanics provided by PI). Depending on the controller, these files are located on the host PC (e.g. with C-863 or C-843) or on the controller (e.g. with C-702 or the C-887 Hexapod controller).
- The list of all axes of the controller (right); shows axis identifiers, current valid stage types and the actions which will be performed when pressing ***OK***. With C-887 Hexapod controllers, only the axes A and B are listed here.

If you cannot find a PI stage in the list of stage database entries, download the latest version of the PIStages2.dat stage database from the PI Website: see "Installing Updates" (p. 3), or contact your Physik Instrumente Sales Engineer. The ***Select connected stages*** window offers a link for the download of the PI Update Finder.

If you want to connect a non-PI stage, see "How to use a stage that is not known to PIMikroMove" (p. 189).

The following actions can be chosen for the controller axes highlighted in the list (actions will be performed not until you press ***OK***):

- Assign the entry selected in the stage database list to the selected axes using the ***Assign*** button. Alternatively, you can also double-click the stage database entry. When pressing ***OK***, the stage-specific settings will be loaded from the stage database to the controller.
  Action column of the ***Controller axes*** list shows "Set to: Stagename"
- Deactivate axes using the ***No Stage*** button (if supported by the controller): You can deactivate an axis if required by your application, e.g. if no stage is connected to the corresponding socket, or if motion of the axis is strictly forbidden. Deactivation of an axis means that this axis is not available for axis-related commands any more (e.g. motion commands, position queries). Hence after pressing ***OK*** the axis will not be visible any more on the ***Axes*** tab card or in ***Single-Axis*** windows of PIMikroMove. You can undo axis deactivation at any time in the Select connected stages window by selecting a stage database entry and using the ***Assign*** button (see above).
  Action column of the ***Controller axes*** list shows "Set to: NOSTAGE"
- Load the parameters of the current valid stage type again from the stage database using the Reload button. This is necessary if the settings of the current valid stage type have changed in the stage database. If you do not reload them, they will not come into effect. When pressing ***OK***, the modified parameter values will then be loaded to the controller.
  Action column of the Controller axes list shows "Reload: Stagename"

- Discard all changes using the ***Reset*** button. This will undo all action selections made before using the ***Assign***, ***No Stage*** or ***Reload*** buttons.
  Action column of the ***Controller axes*** list shows "<do not change>"

OK performs the actions shown in the ***Action*** column. If <do not change> is shown in the ***Action*** column of the ***Controller axes*** list, pressing ***OK*** will not change any settings for the corresponding axis on the controller. ***OK*** also closes the window, and the ***Start up axes*** window is displayed afterwards.

INFORMATION

Controllers which are able to store parameter values in non-volatile memory may come with preset parameter values, especially when delivered with custom stages. The ***Current stage type*** column of the ***Controller axes*** list then already shows the name of the custom stage. In this case, do not choose any stage database entry from the list. Before you click ***OK*** in the ***Select connected stages*** window, make sure that the ***Action*** column of the ***Controller axes*** list shows "<do not change>".

5.3 Configure Hexapod

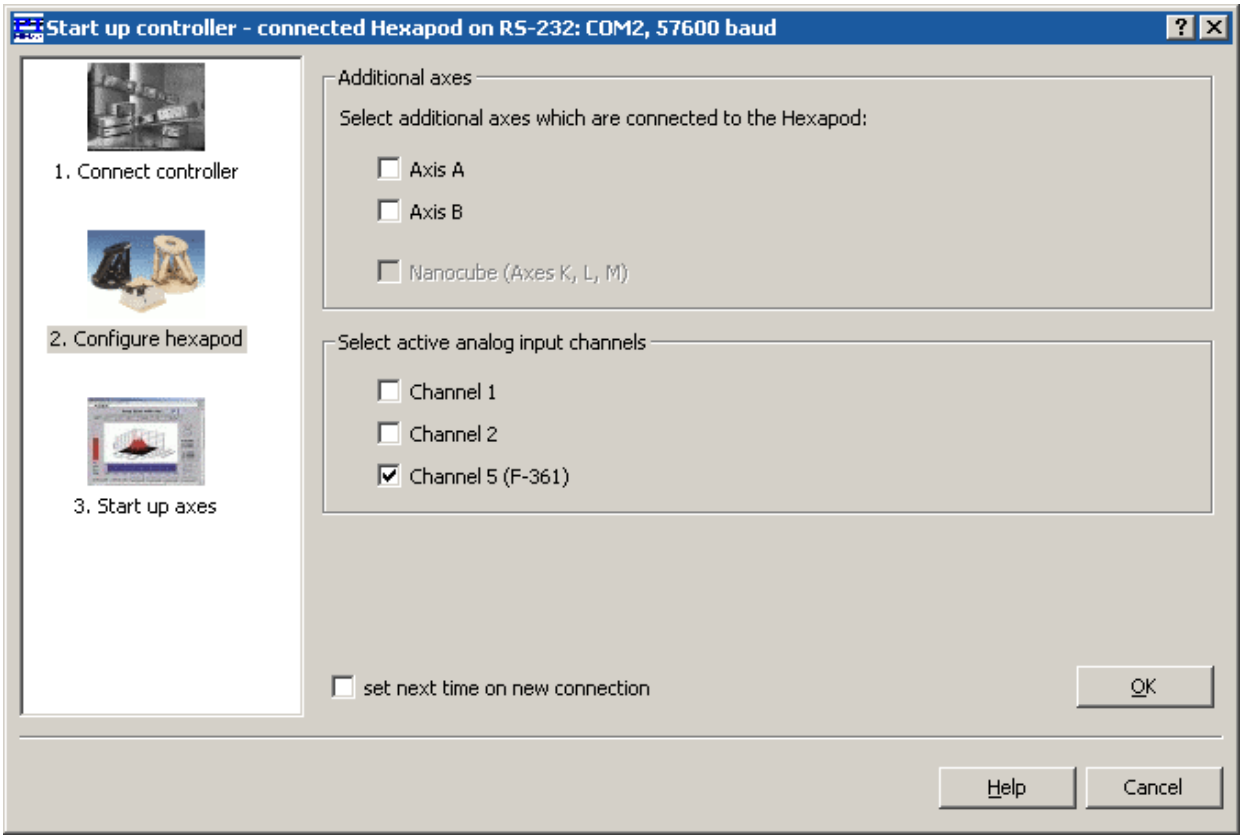


Figure 99: Start up controller window with the Configure Hexapod step

This dialog is only available if the connected controller is a Hexapod controller with GCS syntax version 1.0. Here you have to indicate whether separate motorized axes A and B or NanoCube® axes K, L, and M (F-206 only) are connected, and what analog inputs are available. To use these axes and inputs, the controller must have been preconfigured for them. See the system User Manual for details.

5.4 Start Up E-517

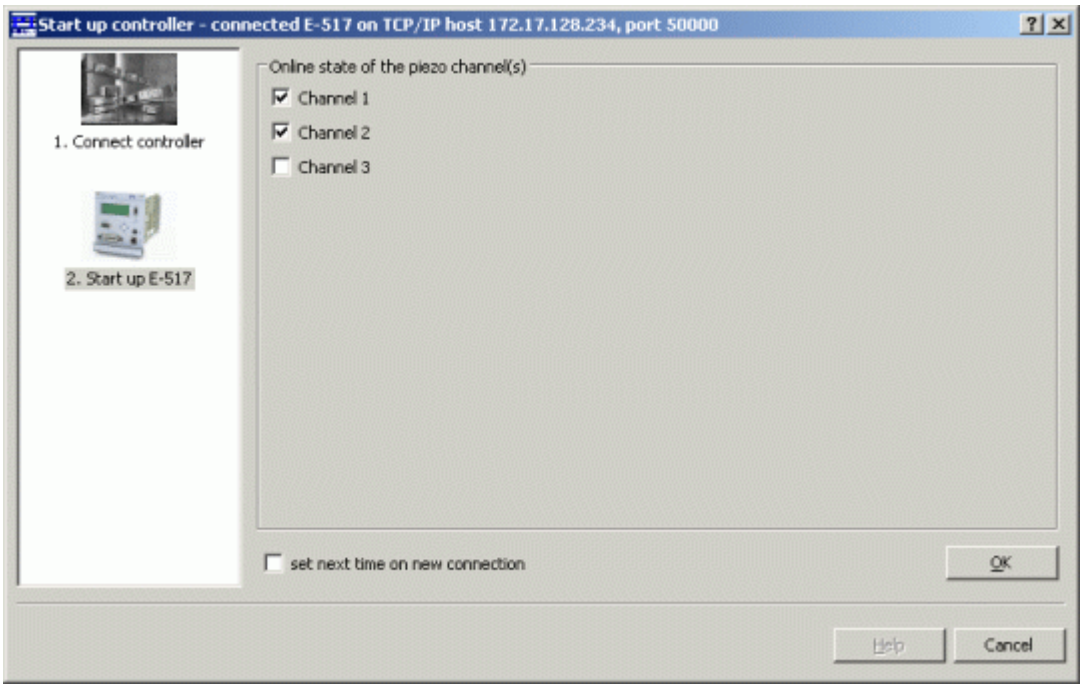


Figure 100: Start up controller window with the Start Up E-517 step

This dialog is only available if the connected controller is an E-517 interface and display module. The online state of a (piezo) channel determines the applicable control sources for the output voltage and hence for the axis motion:

- Checkbox selected = ONLINE mode:
  PIMikroMove controls the piezo channel and hence the motion of the corresponding axis. Any analog control input voltage and DC offset settings on the analog piezo control electronics will be ignored.
- Checkbox not selected = OFFLINE mode:
  The channel and hence the motion of the corresponding axis can not be controlled by PIMikroMove but only by analog control input voltage and DC offset settings applied to the analog piezo control electronics.

INFORMATION

In ONLINE mode the SERVO switches of all channels must be set to OFF on the analog piezo control electronics.

The ***Start up E-517*** dialog will not appear again, and the current settings will be adopted if you select ***set next time on new connection***. You can reopen the dialog from the E-517 controller menu with the ***Start up axes…*** item.

5.5 Start Up Axes

This dialog is available if a system requires that its axes be initialized before normal use. You can return to this window by selecting ***Start up axes...*** in the controller menu of the corresponding controller or in the ***Axis*** menu opened from the ***Axes*** tab card.

5.5.1 Referencing

For systems with incremental position sensors, the initialization includes determining the absolute position of each axis (referencing). Referencing is usually done by driving the motor slowly until a reference or limit switch with a known position is reached, or, if the absolute current position is known not to have changed since having been saved on the host, it can be restored, or, if it is otherwise known, it can be entered manually.

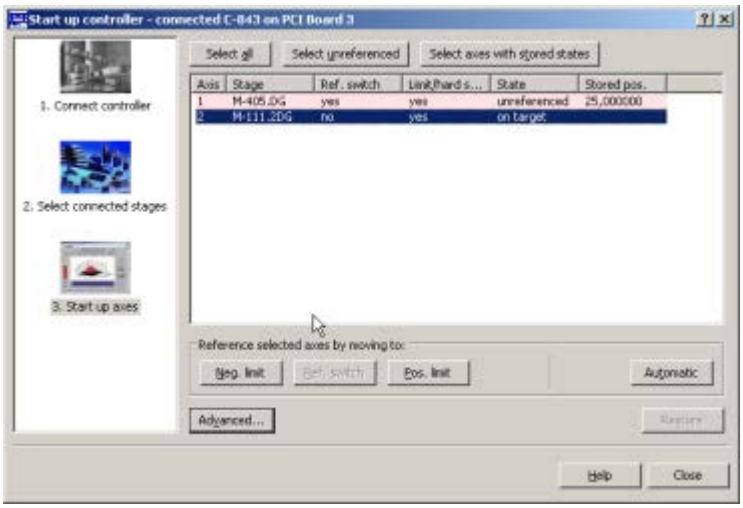


Figure 101: Start up Controller window with the Start up axes step for a system with incremental position sensors

Neg. limit, Pos. limit, and ***Ref.*** switch buttons are for a reference move of the selected (highlighted) axis or axes. If any of the selected axes does not support a given type of referencing (e.g. has no reference switch), the respective button will be dimmed.

Restore is available if the state of the marked axis is ***Unreferenced*** and positions were saved on the host PC the last time the connection to the controller was closed.

NOTICE

Damage due to crashes!

With ***Restore***, the saved position will be set as the current absolute position and will be the basis of all future calculations and range checks. If the stage has moved and the value is no longer correct, you could easily crash the stage.

➢ Be sure that the affected axes have not moved in any way since the positions were saved.

For more information see "Why do I need to reference an axis?" (p. 187).

If you need more control or you cannot move the stage to one of the known positions, call the ***Advanced startup*** (p. 182) dialog by clicking ***Advanced…***.

5.5.2 AutoZero

For systems with linear piezo actuators, both the range of sensor position values and the range of the output drive voltages are limited. If mechanical drift of the piezo actuator causes too great a shift in the relation between these ranges, then the usable closed-loop travel range will be reduced. Such an offset can be compensated by the AutoZero function.

INFORMATION

During the AutoZero procedure, the axis will move, and the motion can cover the whole travel range.

When the piezo stage is first installed in the application, AutoZero must be run. Afterwards, AutoZero should only be executed in the following cases:

- The system is subjected to temperature changes.
- The load applied to the axis has changed (note that the effect of load changes depends on the stiffness of the stage axis).
- The mounting or environment of the stage has changed (e.g. its orientation).

Especially if absolute moves are needed, AutoZero should **not** be executed during normal operation because AutoZero changes the mechanical zero position of the axis.

INFORMATION

AutoZero is not effective on non-linear axes.

See the User Manual of the piezo controller for more information.

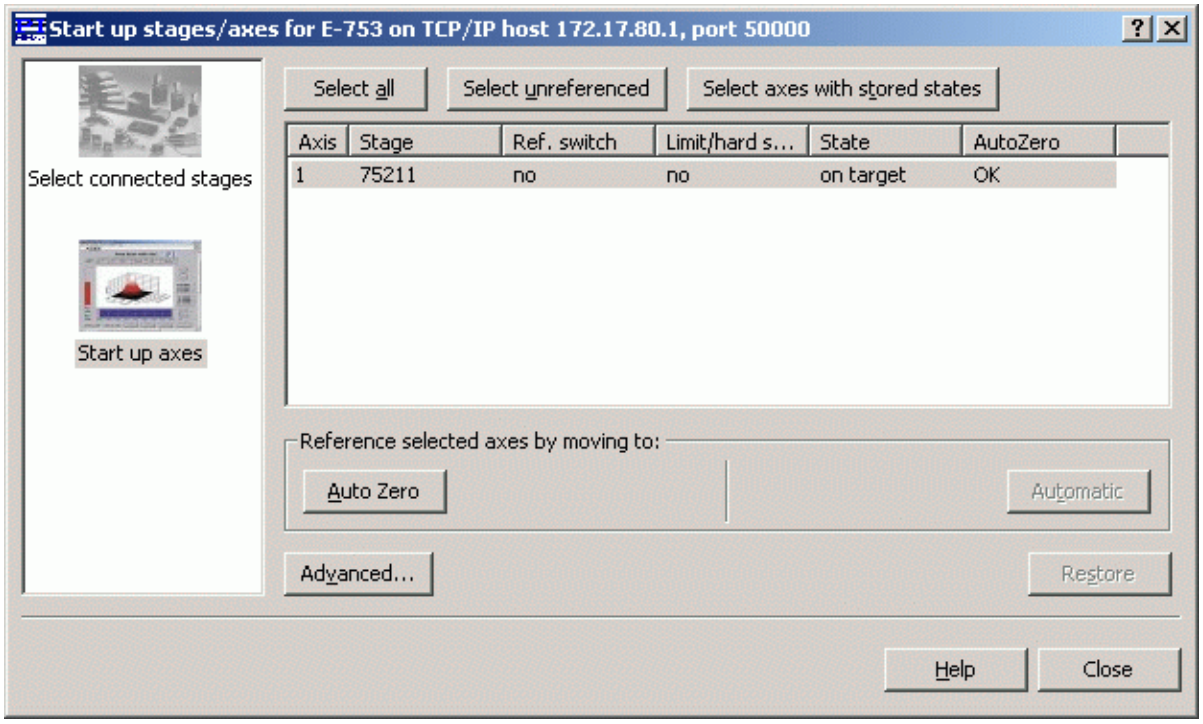


Figure 102: Startup Controller window with the Start up axes step for a system with linear piezo actuators

Click ***Auto Zero*** to start the AutoZero procedure.

5.5.3 Advanced Startup

For systems which must be referenced (with incremental position sensors):

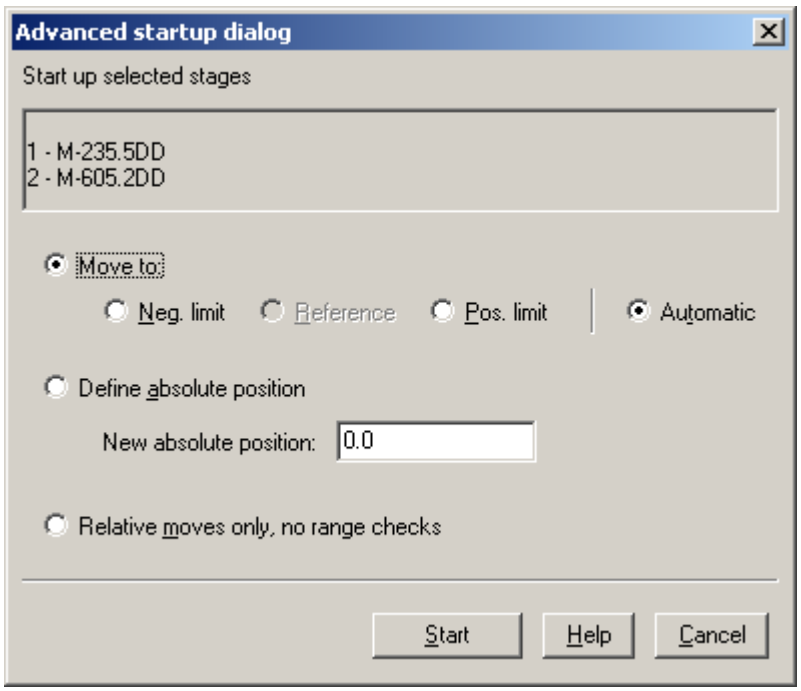


Figure 103: Advanced startup of selected stages

This dialog gives you more control over startup of the stage axes. In this example ***Move to Reference*** is dimmed because some of the selected axes do not have a reference switch. You have the following options:

- ***Define absolute position***: Inform PIMikroMove (and thus the motion controller) manually where the axes currently are. This makes absolute positioning possible and the status will be set to on target. Note that the numeric value entered is interpreted as (scaled) physical units.
- ***Relative moves only***: abandon absolute positioning. When this is done, all range checks are switched off

NOTICE

Damage due to crashes!

The ranges for axis motion (e.g. minimum and maximum commandable position of an axis or soft limits) are not adjusted when you define the absolute position manually. This may result in target positions which are allowed by the controller but cannot be approached by the stage. In this case, the stage can hit the hard stop or objects in the environment and be damaged. Target positions are also possible which can be approached by the stage but are denied by the controller.

- ➢ Be sure to choose the correct value when defining the absolute position manually.

NOTICE

Damage due to crashes!

If you choose to enable relative moves only, no more range checks are performed. You could easily crash the stage.

- ➢ Carefully command the distances for relative moves.

For more information see "Why do I need to reference an axis?" (p. 187) and "My stage has limit switches, so why do I have to worry about crashing it?" (p. 188).

5.6 AutoConnect

If the hardware configuration (controller, communication interface, connected stages) remains unchanged, you can use the AutoConnect feature. AutoConnect makes it possible to let all the startup steps described in "Connect Controller" (p. 171), "Select Connected Stages" (p. 174) and "Start Up Axes" (p. 179) run automatically in a predefined way at the start of PIMikroMove.

The ***AutoConnect Control Wizard*** assists you when defining the AutoConnect configuration. The wizard starts when you close the connection to a controller and click Yes in the AutoConnect selection dialog which opens automatically. If you close all connections, separate dialogs and wizards will appear for the individual controllers. Depending on the controller, AutoConnect configuration can be done for each axis separately or for the controller as a whole.

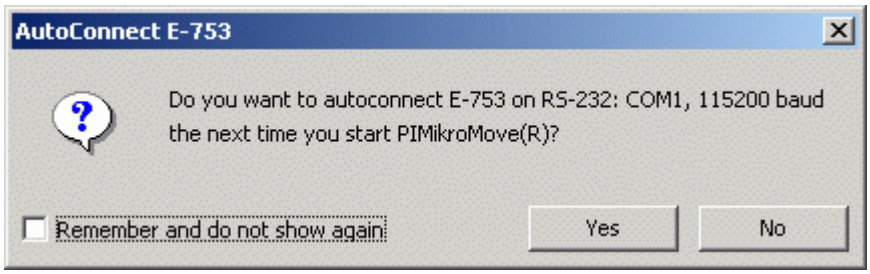


Figure 104: AutoConnect selection dialog

The ***AutoConnect Control Wizard*** guides you through setting up the AutoConnect functionality:

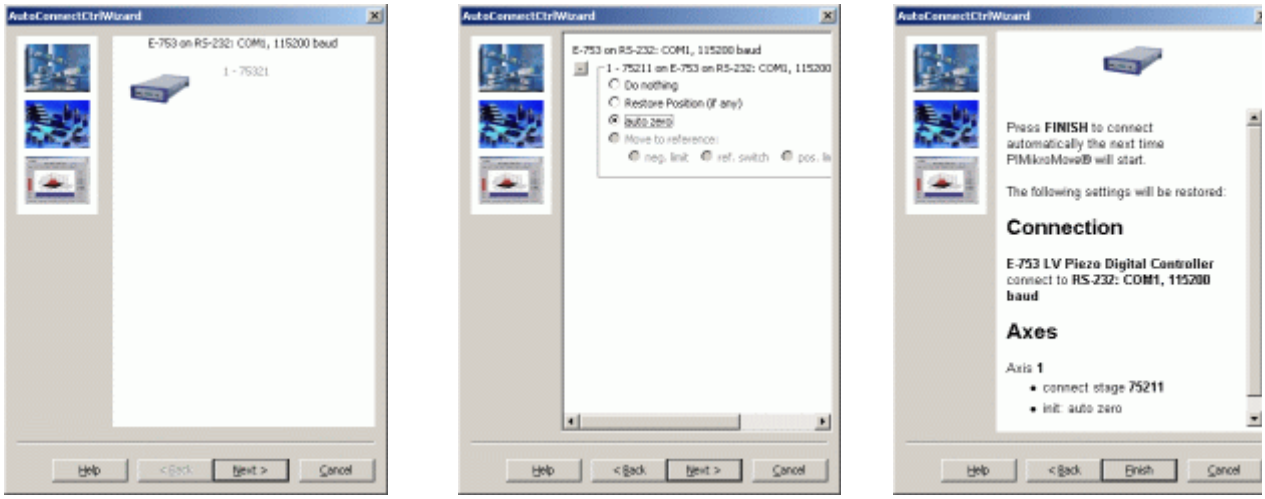


Figure 105: Auto Connect Configuration steps

In the first step (leftmost figure above), you select the axes to be auto-connected (if axis selection is supported by the controller). In the second step, you define how to perform the axis initialization, e.g. by an AutoZero procedure (p. 180) or by a reference move (p. 179) (center figure above). In the last step (rightmost figure above), you save the AutoConnect settings by clicking ***Finish***. Information saved includes the current interface configuration and the current stage-to-axis assignment.

The current AutoConnect settings can be changed in the ***Auto Connect Configuration*** window, which is accessible via the ***Connections > AutoConnect...*** menu sequence. Click on one of the icons in the leftmost pane of the window to select the settings you want to change. In addition to the settings which can also be made in the ***AutoConnect Control Wizard***, you can define a host macro to be the auto start macro. This macro will be executed when AutoConnect has finished successfully for all controllers affected by the macro (see "Host Macros" (p. 39) for details).

For controllers which are currently not connected, you can only deactivate settings in the ***Auto Connect Configuration*** window. To save changes, click ***OK***.

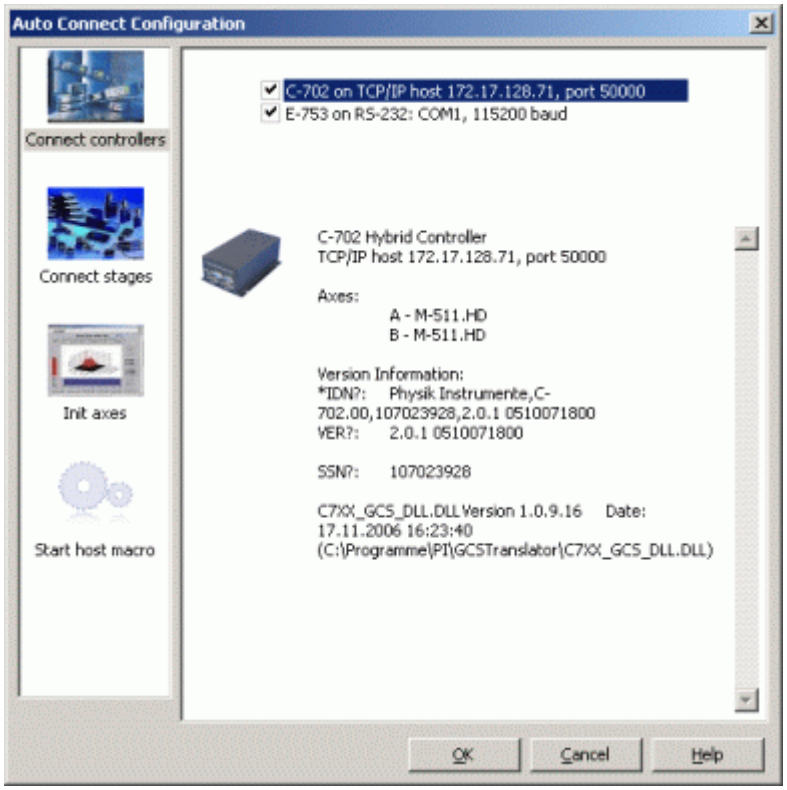


Figure 106: Auto Connect Configuration window with 'Connect controllers' selected

If you select ***Connect controllers*** in the ***Auto Connect Configuration*** window, you can not only deactivate/activate AutoConnect for the individual controllers, but also get information about the hardware and software versions.

If you want to do all the startup steps manually even though AutoConnect is activated, you can skip the AutoConnect procedure as follows:

- If you start ***PIMikroMove*** from a command line, you can use the "-na" or "—noautoconnect" switch (see also "Command Line Options" (p. 186))
- The ***Wait Dialog*** at program start provides a ***Stop*** button with which you can abort AutoConnect

5.7 Command Line Options

If you start PIMikroMove from a command line or via batch processing, you can choose the following options:

- "-v" or "--version" switch opens a window with version information about PIMikroMove and the installed controller software.
- "-h" or "--help" opens a window with a summary of the available command line options.
- "-na" or "--noautoconnect" starts PIMikroMove without the AutoConnect functionality (see "AutoConnect" (p. 183) for more information).
- "--nowaitautoconnect" suppresses the count-down window if the AutoConnect functionality is activated. AutoConnect is then executed immediately upon the start of PIMikroMove.
- "-c" or "--config" starts PIMikroMove without using the standard configuration file which holds, for example, the settings for the communication interfaces or for AutoConnect. You must provide a filename when using this option.
- "-m" or "--macro" with filename as argument will start a host macro after PIMikroMove has been started. This works only if AutoConnect has been performed successfully, and if all controllers affected by this host macro are connected.
- "-d" or "--delay" with an integer as argument gives the delay in milliseconds between two controller status updates (minimum is 10).

6 Tutorials - Frequently Asked Questions

This chapter contains some general information useful in understanding the background of PIMikroMove and answers some frequently asked questions. It also has a collection of short recipes for accomplishing common tasks.

6.1 How can I connect another controller to PIMikroMove?

You can show the ***Connect Controller*** dialog at any time with ***New connection*** on the toolbar or with the ***Connections > New...*** menu sequence. This way, you can connect additional controllers to PIMikroMove. It is not necessary to close an existing connection before additional controllers can be connected since PIMikroMove is designed to handle multiple connections. For example, the axes of all connected controllers will be listed in the main window.

6.2 Why do I need to reference an axis?

To move a stage to an absolute position the motion controller needs to know the current position. Many stages, however, have only relative position sensors (encoders). The controller must somehow determine where the stage is. This can be done by moving to a known fixed position. PIMikroMove uses a limit or reference switch to define such a known point (the reference position is the point where the reference signal changes state). For Hexapod controllers, reference motion (initialization) is the only feasible option.

If a stage has no limit switch or reference signal, or the stage cannot move to any of these positions without damaging the application, you can either enter the current absolute position directly, or—if the current application does not need it—you can operate using relative moves only, i.e. without absolute positions.

To reference an axis by moving to a known point, to enter the current absolute position or to start the axis in relative move mode use the Start up axes (p. 179) dialog.

6.3 Why is everything disabled when Command entry is active?

When the ***Command entry*** window is open, all other controls containing data regarding the selected controller are disabled (***Single-axis*** windows, ***Position pad***, …).

This is done to avoid confusion between the commands entered directly and those sent in the background by PIMikroMove to update its displays (PIMikroMove must continuously check the state of the axes and the controller).

For example, if you want to find out how a specific command works and the command you entered contains a typo or invalid data, then the controller does nothing. To find out what went wrong, you can check the error state with ERR?. If background activity had been taking place, however, the error state may have been cleared in the meantime.

The ***Command entry*** window is designed to give you complete control over what is sent to and received from the controller.

6.4 My stage has limit switches, so why do I have to worry about crashing it?

To avoid sacrificing valuable travel range, limit switches are sometimes installed very close to the hard stop. If run into the limit switch at maximum speed, especially with a load on the platform, the momentum can cause the stage to crash into the hard stop even if the motor shuts down immediately.

6.5 How do I connect another controller to PIMikroMove?

To connect another controller or controller network to PIMikroMove you do not need to close any existing connections. You can connect as many controllers in software as you have physically connected to your PC. Simply call the Start up controller (p. 171) window with the ***Connections > New...*** menu sequence or the ***New connection*** button on the toolbar.

6.6 How can I deactivate a connected stage?

To deactivate an axis, choose ***Select connected stages*** (p. 174) from the corresponding controller menu (p. 13). In the window that opens, select the axes to be deactivated or disconnected and click the ***NOSTAGE*** button. The axis will disappear from the axes table and, if open, the corresponding axis window will be closed. The effect of deactivating an axis in motion or an axis executing a controller macro may vary.

6.7 How do I add a controller to a controller network?

Certain types of controllers can be networked together in a daisy chain and all controlled off one interface on the host PC (e.g. Mercury class controllers like C-863, C-663). PIMikroMove sometimes treats such controller networks as a single, multi-axis controller, sometimes as separate controllers. See the controllers' User Manuals for instructions on configuring such a network.

6.8 How to use a stage that is not known to PIMikroMove

Almost all PI stages with their parameters are stored in the PIStages2 database. If your PI stage is not in the list show in the ***Select connected stages*** startup step, download the latest version of PIStages2.dat from the PI Website as described in "Installing Updates" (p. 3). Note that the version number of PIStages2.dat refers to the database format, not to the information in the list.

With some controllers, it is possible to connect third-party stages. If you want to connect a stage that is not from PI, select the suitable DEFAULT_STAGE entry from the list of stage database entries:

- DEFAULT_STAGE: stage with analog-driven DC-motor
- DEFAULT_STAGE_P: stage with PWM-driven DC-motor
- DEFAULT_STAGE_S: stage with stepper motor
- DEFAULT_STAGE_U: stage with PILine® piezomotor
- DEFAULT_STAGE_N: stage with NEXACT® linear drive

All these entries feature the following stage parameter presettings:

- No limit switches
- No reference switch
- Axis position unit of 1 encoder count

Note that you can start the axis only by entering the absolute position or by choosing to allow relative moves only (usually in the Start up axes window (p. 179)). The units for all moves, positions and accelerations will be encoder counts.

If you have changed some settings and can work with your stage, you can save the settings as a new stage type to the UserStages2 database. Use the ***Add/Edit User Stage Type…*** item of the controller menu.

See also "How can I create a new stage type in the PI stages database?" (p. 190).

INFORMATION

Depending on the controller, you can also inspect parameters on the tab cards of the main window, in the ***Single-Axis*** window or using the ***Device Parameter Configuration*** window. See the User manual of the appropriate controller for more information.

6.9 How can I create a new stage type in the PI stages database?

The easiest way to add a new stage type to the UserStages2 database is to modify the parameters of an existing stage type and save them under a new name. Thereafter you can select this newly defined stage in PIMikroMove or in other PI software as well.

Proceed as follows:

1. Assign the stage type that comes closest to your stage to the appropriate axis. See "Select Connected Stages" (p. 174) for how to do this. Afterwards the ***Start Up Axes*** dialog may open—you can close this dialog because at this point it is not necessary to reference the axis.
2. Open the ***Single-Axis*** window for the axis (“Single-Axis Window" (p. 95)). To do this, either use the ***Single Axis*** Window item in the ***View*** menu or call the corresponding Axis menu from the Axes tab card.
3. Expand the ***Single-Axis*** window via the rightmost > button.
4. In the rightmost pane of the expanded ***Single-Axis*** window, display the columns for the parameters you want to modify. To do this, click the ***Select parameters…*** button. See "Select Columns to be Displayed" (p. 29) for further details.
5. Type new values in the parameter fields. As long as a value is shown in blue, it is only present in PIMikroMove but not yet sent to the controller. Press Enter on your keyboard to send the value to the controller’s volatile memory.
6. Right-click in the center pane of the expanded ***Single-Axis*** window and select ***Add/Edit User Stage type...*** from the menu that appears.

7. To save the modified settings as a new stage, enter a new name for your stage in the as stage type field and click ***OK***.

   For the stage type entry, do not use stage names which already exist in the PIStages2.dat database. If a stage of the same name exists in PIStages2.dat and the user-stages dat file, the parameter settings from PIStages2.dat will be preferred when assigning that stage to an axis (e.g. in the Select connected stages dialog (p. 174)), and the settings from the user-stages dat file will never be used.

8. The new stage is now displayed on the Axes tab card, and you can work with it (e.g. reference the stage: right-click on the axis row and select the ***Start up axes...*** item, see "Start Up Axes" (p. 179)). If you want to further modify the stage parameters, use the ***Add/Edit User Stage type*** menu item again to save the changes.

6.10 How can I modify default settings of a stage type in the PI stages database?

You can modify the default parameters of a stage type loaded from the PIStages2.dat database. Afterwards, add the modified parameters as a new stage type to the UserStages2 database. Proceed as follows:

1. Assign the stage type you want to modify to the appropriate axis. See "Select Connected Stages" (p. 174) for how to do this. Afterwards the ***Start Up Axes*** dialog may open—you can close this dialog because at this point it is not necessary to reference the axis.
2. Open the ***Single-Axis*** window for the axis (“Single-Axis Window" (p. 95)). To do this, either use the ***Single Axis*** Window item in the View menu or call the corresponding ***Axis*** menu from the ***Axes*** tab card.
3. Expand the ***Single-Axis*** window via the rightmost > button.
4. In the rightmost pane of the expanded ***Single-Axis*** window, display the columns for the parameters you want to modify. To do this, click the ***Select parameters…*** button. See "Select Columns to be Displayed" (p. 29) for further details.
5. Type new values in the parameter fields. As long as a value is shown in blue, it is only present in PIMikroMove but not yet sent to the controller. Press Enter on your keyboard to send the value to the controller’s volatile memory.
6. Right-click in the center pane of the expanded ***Single-Axis*** window and select ***Add/Edit User Stage type...*** from the menu that appears.

7. To save the modified settings as a new stage, enter a new name for your stage in the ***as stage type*** field and click ***OK***.

   For the stage type entry, do not use stage names which already exist in the PIStages2.dat database. If a stage of the same name exists in PIStages2.dat and the user-stages dat file, the parameter settings from PIStages2.dat will be preferred when assigning that stage to an axis (e.g. in the Select connected stages dialog (p. 174)), and the settings from the user-stages dat file will never be used.

8. The new stage is now displayed on the ***Axes*** tab card, and you can work with it (e.g. reference the stage: right-click on the axis row and select the ***Start up axes...*** item, see "Start Up Axes" (p. 179)). If you want to further modify the stage parameters, use the ***Add/Edit User Stage type*** menu item again to save the changes.

6.11 I cannot find parameter xyz in the GUI

If you want to view or modify a certain parameter of the axes, or you are missing a function it maybe only hidden from display. If you want to change the set of displayed parameters and functions see "Select Columns to be Displayed" (p. 29).

6.12 Whats the difference between" controller macros" and "host macros"?

“Controller macros” are stored on the controller. They can be started by the host PC but the controller itself parses and executes the commands in the macro.

“Host macros” are stored on the host PC. PIMikroMove will parse each line and execute some commands (start macro, wait for condition, …) and send the other commands to the controller.

So controller macros are faster than host macros, since the timing behaviour of PIMikroMove depends on the operating system and each command must be transferred to the controller.

Since host macros are simple text files they may be used for different controllers—as long as the axis identifiers are still valid.

Host macros can even add macro functionality to controllers which would otherwise have no possibility to do repeated tasks.

6.13 Are there any shortcuts with PIMikroMove?

There are the following general shortcuts:

F1 Opens help window

F2 Opens ***Connect Controller*** dialog

F3 Opens ***PIStageEditor***

F4 Opens ***Command entry*** window

F5 Opens ***Position Pad*** window (provided if only one ***Position Pad*** is available)

On the ***Host macros*** tab card:

- Ctrl+S saves the current macro
- Ctrl+R starts the current macro (without saving)

On the ***Controller macros*** tab card:

- Ctrl+S saves the current macro
- Ctrl+D sends the current macro to controller
- Ctrl+R starts the current macro; if the macro content has changed, the macro is sent to controller before it is started