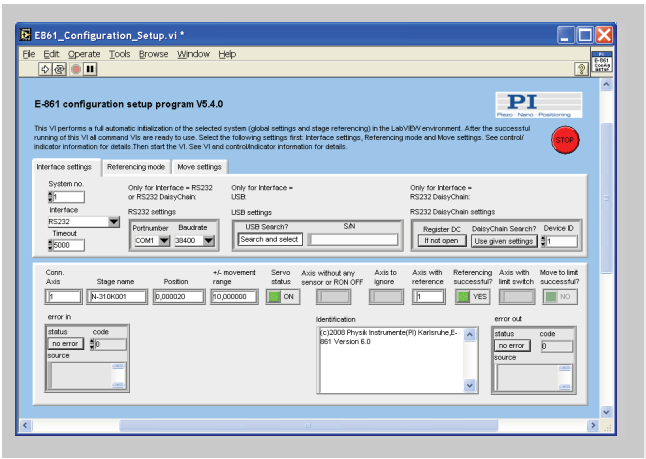
Piezo · Nano · Positioning

PZ 208E **Software Manual**

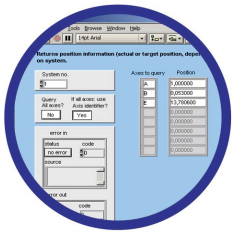
E-861 **LabView Driver Library**

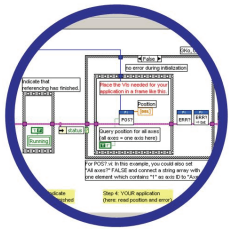
Release: 5.8.1 Date: 2010-02-26

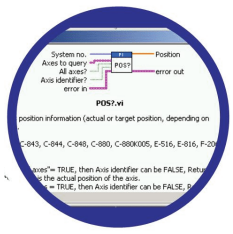


This document describes software for use with the following product(s):

- **E-861** NEXACT® Controller, 1 channel, linear encoder







Auf der Römerstr. 1 · 76228 Karlsruhe, Germany
Tel. +49 721 4846-0 · Fax: +49 721 4846-299
info@pi.ws · www.pi.ws

Table of Contents

Release: 5.8.1
File:E861_GCSLabVIEW_PZ208E_581.doc, 2633728 Bytes

0. Disclaimer

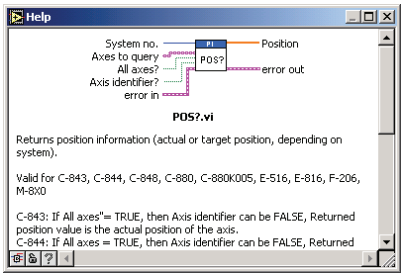
This software is provided “as is”. Physik Instrumente (PI) does not guarantee that this software is free of errors and will not be responsible for any damage arising from the use of this software. The user agrees to use this software on his own responsibility.

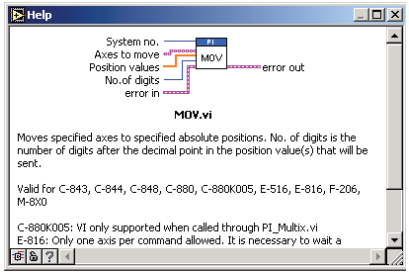
1. Introduction

The LabVIEW software consists of a collection of virtual instrument (VI) drivers. All functionality involves invoking one or more VIs with the appropriate parameter and global variable settings.

These VIs are provided to ease the task of programming your application. They, and the accompanying documentation, assume a prior knowledge of proper LabView programming techniques. The provided "Simple Test” and “Configuration Setup” VIs help to solve the essential initialization steps, but are not intended to provide an out-of-the-box, universal solution to a particular application.

To minimize the need for consulting the manual during programming, each VI comes with a detailed VI description that appears in the *Context Help* window when you move the cursor over the VI icon. Use the *Help→Show Context Help* menu sequence in the LabVIEW environment to display the *Context Help* window.





LabVIEW 7.1 or higher and NI-VISA 3.6 or higher must be installed prior to using this driver set.

To control an analog system, DAQmx 8.3 or higher and a DAQmx-compatible National Instruments DAC card must also be installed.

With Linux operatings systems, the installation is done via the INSTALL script which is to be found in the /linux directory on the software CD (if available, see the controller User manual for more information).

With Windows Vista and PCI cards (C-843, C-843.PM, E-761), the VIs must always be started with the "Run as administrator" option. To do this, click on the VI with the right mouse button and select the "Run as administrator" entry from the context menu.

1.1. PI General Command Set (GCS)

This VI driver set supports the *PI General Command Set*, which is based on ASCII communication with well-defined commands and replies. This makes it possible to control different PI systems, such as the *E-516 Display Module* or the *C-880 Multi-Axis Controller*, with only one driver set simply by “wiring” the correct command parameters to the associated VIs. To achive this, a unique “System no.” must be selected in each “XXXX_Configuration_Setup.vi” (with XXX being the PI product

no. of your system). This System no. is then used in all sub-VIs to tell LabVIEW which connected system to talk to.

Translation Libraries

To control PI systems with a native command set that is not compatible with the *PI General Command Set*, e.g. the *E-710 Digital Piezo Controller* or the *C-843 Motion Control Board*, controller-specific libraries are used. Each such library translates *PI General Command Set* commands to the controller's native language. There is also a universal library which adds this functionality: GCSTranslator; it must be installed on the computer in the GCS_LabVIEW\Low Level folder, no matter whether the system being controlled is *PI General Command Set* compatible or not.

For these and certain other systems (such as PC add-on cards), the required system-specific libraries and data files (e.g. PIStages.dat) must be properly installed. If you install this driver set from within the setup program of the PI software CD ROM, this installation is done automatically. If you want to install this driver set manually, please run "GCSLibrarySetup.exe" from the CD-ROM that came with your system. This setup tool makes sure that all necessary libraries and their data files are correctly registered in the Windows™ environment and can be found by the GCS drivers (if LabVIEW still cannot find PIStages.dat, it may be because it is marked read-only. To see, open Microsoft Explorer, right-click the file PIStages.dat and select *Properties*. Make sure that the *read-only* attribute is not checked.)

Once the libraries and data files for the system to control are installed, this LabVIEW driver set can be used to control a non-GCS-compatible system just like any GCS-compatible system, and PCI/ISA-based controller boards by selecting "DLL" as communication interface (see Section "First Steps for GCS-Compatible PI Controllers" on p. 7 and the "XXXX_Configuration_Setup.vi" (with XXXX being the PI product number of your system) in section 3).

Units and GCS

The GCS system uses physical units of measure. Most controllers and GCS software have default conversion factors chosen to convert hardware-dependent units (e.g. encoder counts) into mm or degrees, as appropriate. These defaults are generally taken from a database of stages that can be connected. The direction of motion associated with positive and negative relative moves can also be controlled by parameter settings. In some cases an additional scale factor can be applied, making a second physical unit available without overwriting the conversion factor for the first. It is also sometimes possible to enter a conversion factor as numerator and denominator of a fraction, reducing the number of digits and outside calculations needed for high-precion entry of gearhead system values. See the DFF.vi and SPA.vi command descriptions (if supported by your PI controller), taking special note of the sections referring specifically to your controller.

1.2. Scope of This Manual

This manual covers only VIs which can be used with the product with which it came, and VIs which must be present for all products supported by this driver set. A VI can be used with a certain product if the product name is mentioned in the "Valid for" line of the VI description..

For VIs which are based on GCS commands, see the User manual of the controller or, if present, the special GCS commands manual for further details.

1.3. VI Structure

The folder structure of the LabVIEW drivers consists of the main folder “GCS_LabVIEW” with the sub-folder “Low Level”.

The main folder “GCS_LabVIEW" contains a terminal VI (for command based systems), a configuration VI (XXXX_Configuration_Setup.vi with XXXX being the PI product number of your system), a simple test VI, and, if available, several sample programs.

The sub-folder “Low Level” contains VIs for the following functions:

- Establishing communication with different PI systems which support the PI General Command Set via RS-232, GPIB or TCP/IP interfaces, or with analog systems, defining the parameter IDs of the connected axes, sending and receiving ASCII characters to/from the specified system or setting and reading voltages for an analog system. These VIs are mainly sub-VIs for the XXXX_Configuration_Setup.vis which overtake the communication parameter setup and initialization of all necessary settings automatically.
- Support functions which are helpful for several common tasks in LabVIEW and are used by the command VIs
- Sending system-specific commands (system-specific commands are separated into function-specific LLBs) which are the “construction set” to build your application.

Additionally, the sub-folder “Low Level” contains GCSTranslator.dll.

Following the data flow concept of LabVIEW, all VIs have their wiring inputs on the left side and their wiring outputs on the right side of each connector pane. For quick integration, this **connector pane** in most cases has the following pattern:

<table>
<tr><td>1</td><td rowspan="3">7</td><td rowspan="3">9</td><td rowspan="3">11</td><td rowspan="3">13</td><td>15</td></tr>
<tr><td>2</td><td>16</td></tr>
<tr><td>3</td><td>17</td></tr>
<tr><td>4</td><td rowspan="3">8</td><td rowspan="3">10</td><td rowspan="3">12</td><td rowspan="3">14</td><td>18</td></tr>
<tr><td>5</td><td>19</td></tr>
<tr><td>6</td><td>20</td></tr>
</table>

The terminals are assigned as follows (if the mentioned, control/indicator is present in one of the supplied libraries):

1 System number
2 Optical board, Interface, or other main input control
3 Axes to query, Affected axes, Number of systems, or other main input control
4 All axes?, Invert order?, or other main input control
5 Axis identifier?, No. of digits, or other main input control
6 Error in
7 Parameter number, Without axis ID?, or other input control
8 Step size, or other input control
9 AA step size, or other input control
10 Input control
11 Input control or output indicator

12 Input control or output indicator
13 Input control or output indicator
14 Input control or output indicator
15 Hidden error, Connected axes, String read, or other main output indicator
16 Axes to query out, Bytes read, or other main output indicator
17 No. of rows, or other main output indicator
18 Output indicator
19 Output indicator
20 Error out

Also note that this driver set does not use the standard LabVIEW error numbers recommended by National Instruments, but rather those used by PI controllers. As a result, the error texts displayed by LabVIEW will not describe the error accurately. Use “GCSTranslateError.vi” to get the description of a PI GCS error number. Some VIs use an additional indicator Controller error to indicate that the selected system has been queried for a controller error with „ERR?“ and reported an error number ≠ zero.

See also chapter 5 on p. 111 for a summary of error numbers produced by this driver set.

In LabVIEW, uncheck *Enable automatic error handling dialogs* in *Tools→Options→New and Changed in 7.x* to prevent that LabVIEW suspends execution and displays an error dialog box for any error that occurs during the execution of the VIs.

Important:

Before running any VIs to control a connected system, “**XXXX_Configuration_Setup.vi**” (located in the main folder, with XXXX being the PI product number of your system) must be run. This initialization VI performs all necessary steps automatically:

1. It opens the communications port,
2. It defines the IDs for the connected axes,
3. It references the connected stages (if appropriate), depending on if the controller requires a referencing before axes can be moved and on your custom settings,
4. It defines the controller name.

After these steps all parameters are saved into global variables, so that other VIs invoked during the same LabView session can access this data at runtime.

As the initialization is a complex procedure which uses a large number of sub-VIs, **XXXX_Configuration_Setup.vi** is password-protected, meaning that you cannot see or modify the diagram. In this way, the full initialization is packed into one single and fully tested procedure which you simply insert into your own application program. For security reasons as well as your convenience, we recommend that you not modify this VI.

For testing a PI system using a command-based interface, the easiest method is to call “PI Terminal.vi”, which is located in the ”GCS_LabVIEW” main folder. This is a ”stand-alone” routine that calls ”PI Ask for Communication Parameters.vi” first and then opens the specified communications ports. It does not, however, define the connected axes of the (motion) systems.

A more system-specific sample VI is “XXXX_Simple_Test.vi” (with XXXX being the PI product number of your system), also located in the “GCS_LabVIEW” main folder. It is available both for command-based and analog systems.

1.4. Working with two PI products which understand PI's General Command Set (GCS) in LabVIEW

When installing the LabVIEW programming support for two different PI products, there are two “Low Level” folders installed, one in each product-specific LabVIEW driver set. This is because every product comes with only the VIs which are used with the product. Another product may have different libraries or different library contents due to the product supporting more or fewer functions. When working with two product-specific LabVIEW driver set installations on one computer, it is important to make sure that LabVIEW always uses the right libraries.

a) When working separately with two products, the “Low Level” folder of each product must be located in the same folder as the product-specific main VI which calls sub-VIs from the product-specific driver set. Otherwise LabVIEW will start searching for sub VIs whereever it finds them, which may result in version conflicts and “broken Run” arrows. Please make sure that no VIs are saved under LabVIEWs own “user.lib” sub-folder. If they are LabVIEW will always find them there first, which will cause errors in many cases.

b) When working with two products in parallel, the libraries should be combined. Please use “MergePIDriver.vi”, located in “MergeDrivers.llb”, to combine two or more PI driver sets. Make sure to work thereafter with the combined libraries instead of the product-specific libraries. If you encounter any broken arrows or error messages after merging please contact your local sales representative with the following information:

    i. Product names of PI LabVIEW drivers to merge
    ii. Version file “version.txt” of all driver sets to merge (located in the Low Level folder of each source driver set after merging)
    iii. Name(s) of VI(s) with broken arrows
    iv. Error code (if any) and name of VI in which the error occurred

Before combining driver sets, please do always check if there is an update available for one of the driver sets to merge, or for the Merge Tool itself.

Select a unique “System no.” in each XXXX_Configuration_Setup.vi (with XXXX being the PI product number of your system) and use this System no. in all command VIs to tell LabVIEW which system to send commands to.

1.5. First Steps for GCS-Compatible PI Controllers

1.5.1. E-861

Step 1: Before you start, please check that the current configuration matches your stage connections. Use PIMikroMove to configure stages connected to the E-861.

Step 2 (advanced users can skip this step): To check communication between the E-861 controller and the host PC, run "E861_Simple_Test.vi". This VI will return the ID string of the E-861 controller. See chapter 3 for a description of this VI and use the *Help→Show Context Help* menu sequence in the LabVIEW environment to display the *Context Help* window with the VI and control/indicator descriptions.

Step 3:

WARNINGS

E861_Configuration_Setup.vi May Cause Move

When you start "E861_Configuration_Setup.vi" with Connected? = TRUE, the VI will automatically determine if the axis has a reference switch or limit switches and, if the referencing mode is ON, will move the stage to the appropriate sensor switch. It is therefore important to make sure that items connected to or mounted on the connected stage cannot be damaged by such a move. If referencing is not possible (because the controller is an open-loop device, the connected stage has no reference or limit switch) or not desired, referencing mode (the mode which tells the controller to reference the stages or not) can be switched off (see chapter 3). See description of RON for details and warnings.

Use Only Open-Loop Move Commands for Open-Loop Systems

If no sensor is present, do not switch the servo on and do not send commands for closed-loop motion, like MOV or MVR. Otherwise the connected mechanics can run into the hardstop at full speed which may cause damage to your hardware setup.

To control one or more E-861 controllers with this driver set, run "E861_Configuration_Setup.vi". This VI performs all steps necessary for a full configuration of the driver VIs in the LabVIEW environment: the definition of the axis ID, the initialization of the connected stage including referencing (if appropriate) and the definition of the controller name. During your testing phase (when you simply run the VIs without wiring them together into a program), do not close "E861_Configuration_Setup.vi"; otherwise all global settings will be lost and the driver VIs will not work.

Do not forget to run "Close connection if open.vi" with Close DaisyChain = TRUE before re-connecting this or any other controller connected to the same interface (except if you want to connect another device to the same DaisyChain).

When programming your application, you can implement "E861_Configuration_Setup.vi" as an initialization VI in your software. See Chapter 3 for a detailed description of "E861_Configuration_Setup.vi" and use the *Help→Show Context Help* menu sequence in the LabVIEW environment to display the *Context Help* window with the VI and control/indicator descriptions.

Before using a joystick connected to the host PC, install joystick driver and calibrate joystick in the Windows control panel.

Default IDs:

Each E-861 in a daisy chain network must have a unique controller address in the range of 1 to 16. The controller address can be set with the DIP switches on the front panel, see the E-861 User manual for details. Default address setting is 1.

In the E-861 firmware, closed-loop motion is always commanded for logical axes (move commands like MOV and MVR), while some open-loop motion commands are related to PiezoWalk® channels (OAD and OSM). Since the E-861 is a single-axis / single-channel device, the terms "axis" and "PiezoWalk channel" can be used synonymously. Default axis ID / PiezoWalk channel ID is "1".

Analog input channels: 1 to 6, lines 1 to 4 can also be used for digital input (see E-861 User manual for pinout)

Digital input lines: 1 to 4, can also be used for analog input (see E-861 User manual for pinout)

Digital output lines: 1 to 4 (see E-861 User manual for pinout)

Joystick (see E-861 User manual for pinout):
ID of joystick device: 1
ID of joystick axis: 1 with joystick-related commands (JAS?, JAX, JAX?, JDT, JLT, JLT?) and 5 with the DRC command, record option 81
ID of joystick button: 1 with the joystick-related JBS? command and 6 with the DRC command, record option 81

Default position unit for closed-loop operation is given by the Axis Unit parameter (ID 0x07000601) which can be read using the SPA? command. It can be µm, mm or nm for PI stages, or counts for unknown stages (“DEFAULT_STAGE” parameter set e.g. loaded from the PIStages2.dat stage database).” Default position unit for open-loop operation is steps/s.

GCS syntax version: 2.0

2. Low Level VIs

The following low-level VIs can be found in the “Low Level” folder:

2.1. Analog controller VIs (“Analog control.llb”)

2.1.1. Analog FGlobal.vi (Analog control.llb)

Valid for	Analog systems (but must be present for all other systems also)
Input	System no. (1), Read(F)/Write (TRUE), VI ref in
Output	VI ref out
Remarks	This VI works as a functional global variable for VI references

2.1.2. Analog functions.vi (Analog control.llb)

Valid for	Analog systems (but must be present for all other systems also) --- Dummy VI
Input	System number (1), String to send (empty string), type specifier VI Refnum, AI Task, AO Task, Waveform to write, Continuously? (TRUE), Error in (no error)
Output	Command, String output, Boolean output, Error out
Remarks	Dummy VI

2.1.3. Analog Receive String.vi (Analog control.llb)

Valid for	Analog systems (but must be present for all other systems also)
Input	System number (1), Read/Write (T) (FALSE), Ini (False), Error in (no error)
Output	String out, Strings out, Error out
Remarks	Works as an old style global variable for String out.

2.1.4. Available Analog Commands.ctl (Analog control.llb)

Valid for	Analog systems (but must be present for all other systems also)
Input	None
Output	None
Remarks	Type definition for available analog commands.

2.1.5. Global Analog.vi (Analog control.llb)

Valid for	Analog systems (but must be present for all other systems also)
Input	None
Output	None
Remarks	A global variable which contains setup information for analog systems.

2.2. Communication VIs ("Communication.llb"):

2.2.1. Available DLL interfaces.ctl (Communication.llb)

Valid for	C-843, C-843.PM, C-865, C-866, C-867, E-517, E-709, E-710, E-712, E-725, E-755, E-761, E-816, E-861, Mercury (but must be present for all other systems also)
Input	None
Output	None
Remarks	Type definition for hardware interfaces available when communicating with a system through a PI GCS DLL.

2.2.2. Available DLLs.ctl (Communication.llb)

Valid for	C-843, C-843.PM, C-865, C-866, C-867, E-517, E-709, E-710, E-712, E-725, E-755, E-761, E-816, E-861, Mercury (but must be present for all other systems also)
Input	None
Output	None
Remarks	Type definition for available GCS DLLs for communicating with a system.

2.2.3. Available interfaces.ctl (Communication.llb)

Valid for	All systems
Input	None
Output	None
Remarks	Type definition for available interfaces for communicating with a system.

2.2.4. Close connection if open.vi (Communication.llb)

Valid for	All systems
Input	System number (1), Error in (no error)
Output	Was connected? (T/F), Error out
Remarks	This VI checks if the connection to the selected system is already open and, if it is, it closes this connection.

2.2.5. Find baudrate.vi (Communication.llb)

Valid for	C-702, C848, C-867, C-880, C-880K005, E-516, E-517, E-709, E-712, E-725, E-755, E-816, E-861, F-206, M-8X0
Input	System number (1), RS-232 Port number (0: COM1), Timeout (2000), Valid baudrates (array of 5 values), Flow control (All FALSE, x13, x11, x0), Termination character (LF), Interface clear (XXX\n), String to Send (*idn?), Error in (no error) C-702: Input and output HW handshake must be TRUE. All other controls=default. C-848: Input and output HW handshake must be TRUE. All other controls=default. C-867: Input and output HW handshake must be FALSE. All other controls=default. C-880: Input and output HW handshake must be TRUE. All other controls=default. C-880K005: All controls=default.

E-516: Input and output HW handshake must be TRUE. All other controls=default.

E-517: Input and output HW handshake must be TRUE. Not available for Interface = GPIB, TCP/IP or DLL (USB). All other controls=default.

E-709: Input and output HW handshake must be TRUE. Not available for Interface = USB. All other controls=default.

E-712: Input and output HW handshake must be TRUE. Not available for Interface = TCP/IP or DLL (USB). All other controls=default.

E-725: Input and output HW handshake must be TRUE. Not available for Interface = TCP/IP or DLL (USB). All other controls=default.

E-753: Input and output HW handshake must be TRUE. Not available for Interface = TCP/IP. All other controls=default.

E-755: Input and output HW handshake must be TRUE. Not available for Interface = DLL and DLL Interface = RS232DC (DaisyChain). Interface clear = \18 (Use "\"Codes Display" to enter), String to Send = err?. All other controls=default.

E-816: Input and output HW handshake must be TRUE. All other controls=default.

E-861: Input and output HW handshake must be FALSE. All other controls=default.

F-206: All controls=default.

M-8X0: All controls=default.

Output	Baudrate out, String read, Error out
Remarks	Opens COM port of given system with valid baudrates until status of Error out is false.

E-861, C-867: The baudrate is set via the DIP switches on the controller front panel. See the controller User manual for details.

2.2.6. GCSTranslator DLL Functions.vi (Communication.llb)

Valid for	C-843, C-843.PM, C-844, C-865, C-866, C-867, E-517, E-709, E-710, E-712, E-725, E-755, E-761, E-816, E-861, Mercury (but must be present in Communication.llb for all other systems also)
Input	System number (1), Function (C844_IsDLLAvailable), String buffer (empty string), String input (empty string), Error in (no error)
Output	DLL I32 Return value, Numerical output, Boolean output (T/F), String output, Error out
Remarks	This VI calls a given function from GCSTranslator.dll. GCSTranslator.dll must be installed. To call a system-specific function, the system-specific GCS DLL must be installed also.

Warning: For XXX_GcsGetANswer , String buffer must be large enough, otherwise the application may crash. Call XXX_GcsGetANswerSize first to determine necessary string length.

2.2.7. Global DaisyChain.vi (Communication.llb)

Valid for	All systems
Input	None
Output	None
Remarks	A global variable which contains setup information for DaisyChain systems.

2.2.8. Global1.vi (Communication.llb)

Valid for	All systems
Input	None
Output	None
Remarks	A global variable which contains communication setup information.

2.2.9. Initialize Global1.vi (Communication.llb)

Valid for	All systems
Input	System number (1), Error in (no error)
Output	Error out
Remarks	This VI initializes Global1 according to the given system no.

2.2.10. Initialize Global DaisyChain.vi (Communication.llb)

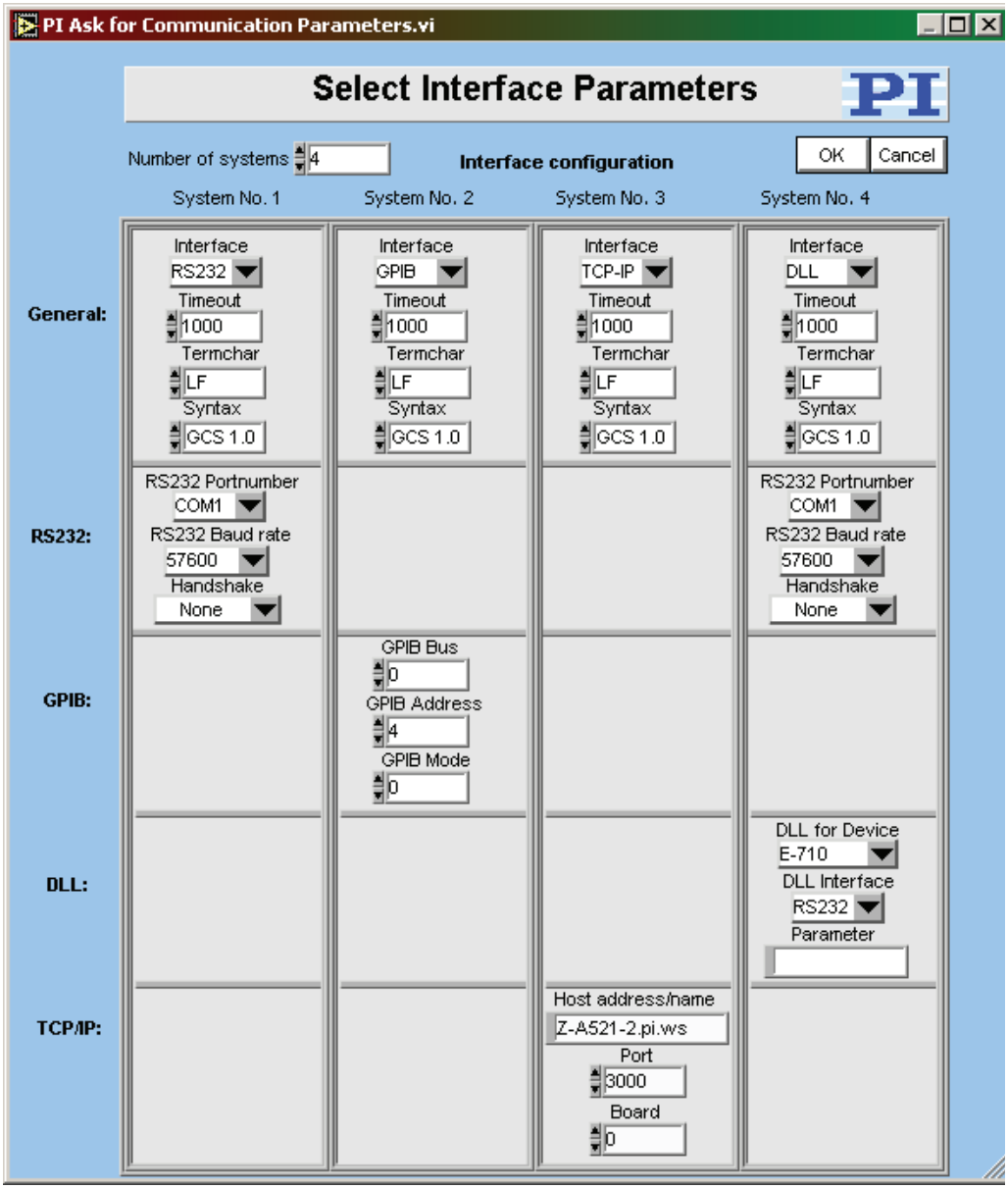
Valid for	C-867, E-755, E-861 (but must be present for all other systems except Analog systems, too)
Input	System number (1), Error in (no error)
Output	Error out
Remarks	This VI initializes Global DaisyChain according to the given system no.

2.2.11. Is DaisyChain open.vi (Communication.llb)

Valid for	C-867, E-755, E-861 (but must be present for all other systems except Analog systems, too)
Input	System number (1), Error in (no error)
Output	Port ID, DC open?, Error out
Remarks	This VI checks if a DaisyChain connection is already open for the communication port defined for the given system no. It does also return the Port ID of the DaisyChain connection if any exists.

2.2.12. PI Ask for Communication Parameters.vi (Communication.llb)

Valid for	All except analog systems
Input	None
Output	Number of systems, Cancel (T/F), Interface configuration, DLL interface configuration, Flow control
Remarks	A user-interface VI for setting up communications parameters (RS-232 or GPIB, number of systems, baudrate, timeout etc.) for up to 4 systems. Press F1 for displaying a help window with the appropriate interface configuration of each PI controller.



2.2.13. PI Open Interface of one system.vi (Communication.llb)

Valid for	All except analog systems
Input	System Number (1), Interface configuration (RS232, 5000, COM1, 57600), DLL Interface configuration (C-843, Board, 1), TCP/IP configuration (localhost, 3000, 0), Flow control (All FALSE, x13, x11, x0), Bitt settings and parity (8, 1bit, no parity), Termination character (LF), Syntax (GCS 1.0), String to send (*idn?), Interface clear (XXX\n), Register DC (FALSE: If not open)
Output	String read, Error out
Remarks	Establishes communication with one connected system. **This VI is called automatically by "XXXX_Configuration_Setup.vi" (with XXXX being the PI product number of your system) and must be completed successfully before any other VI can use the interface.** The interface and error status of the chosen system are cleared by this VI, which sends XXX (no command), *IDN? and ERR?.
	C-702: Interface = RS232 or TCP/IP, RS232: Input and output HW handshake

must be TRUE, Syntax: GCS 1.0; Term char = LF.

C-843: Interface = DLL, DLL for Device = C-843, DLL Interface = Board, Parameter = Board number (1 for first C-843 board), Syntax: GCS 1.0; Term char = LF.

C-843.PM: Interface = DLL, DLL for Device = C-843.PM, DLL Interface = Board, Parameter = Board number (1 for first C-843 board) , Syntax: GCS 1.0; Term char = LF.

C-844: Interface = DLL, DLL for Device = C-844, DLL Interface = RS232 or GPIB, Parameter = empty string, RS232 baud rate = 9600

C-865: Interface = DLL, DLL for Device = C-865, DLL Interface = RS232, Parameter = empty string, RS232 baud rate = set as appropriate, Syntax: GCS 1.0; Term char = LF.

C-866: Interface = DLL, DLL for Device = C-866, DLL Interface = RS232 or USB, RS232: Parameter = empty string, RS232 baud rate = set as appropriate, USB: Parameter = Serial no. of system to connect to, Syntax: GCS 1.0; Term char = LF.

C-867: Single Device: Interface = RS232 or DLL, RS232: Input and output HW handshake must be FALSE. DLL (USB): DLL for Device = C-867, DLL Interface = USB, Parameter = Serial no. of system to connect to. DaisyChain: Interface = DLL, DLL for Device = C-867, DLL Interface = RS232_DC, Parameter = Number of device in chain, Register DC: FALSE. Syntax: GCS 2.0; Term char = LF.

C-880: Interface = RS232 or GPIB, RS232: Input and output HW handshake must be TRUE, Syntax: GCS 1.0; Term char = LF.

C-848: Interface = RS232 or GPIB, RS232: Input and output HW handshake must be TRUE, Syntax: GCS 1.0; Term char = LF.

C-880K005: Interface = RS232, Input and output HW handshake must be FALSE, Syntax: GCS 1.0; Term char = LF.

E-516: Interface = RS232 or GPIB, RS232: Input and output HW handshake must be TRUE, Syntax: GCS 1.0; Term char = LF.

E-517: Interface = RS232, GPIB, TCP/IP or DLL, RS232: Input and output HW handshake must be TRUE, DLL (USB): DLL for Device = E-517, DLL Interface = USB, Parameter = Serial no. of system to connect to. Syntax: GCS 2.0; Term char = LF.

E-709: Interface = RS232 or USB, RS232: Input and output HW handshake must be TRUE, DLL: DLL for Device = E-709, DLL Interface = USB, Parameter = Serial no. of system to connect to. Syntax: GCS 2.0; Term char = LF.

E-710: Interface = DLL, DLL for Device = E-710, DLL Interface = RS232 or GPIB, Parameter = empty string, Syntax: GCS 1.0; Term char = LF.

E-712: Interface = RS232, TCP/IP or DLL, RS232: Input and output HW handshake must be TRUE. DLL: DLL for Device = E-712, DLL Interface = USB, Parameter = Serial no. of system to connect to. Syntax: GCS 2.0; Term char = LF.

E-725: Interface = RS232, TCP/IP or DLL, RS232: Input and output HW handshake must be TRUE. DLL: DLL for Device = E-725, DLL Interface = USB, Parameter = Serial no. of system to connect to. Syntax: GCS 2.0; Term char = LF.

E-753: Interface = RS232 or TCP/IP, RS232: Input and output HW handshake must be TRUE, Syntax: GCS 2.0; Term char = LF.

E-755: Single Device: Interface = RS232, Input and output HW handshake must

be TRUE.
DaisyChain: Interface = DLL, DLL for Device = E-755, DLL Interface = RS232_DC, Parameter = Number of device in chain (first device: 1), Register DC: FALSE.
Syntax: GCS 2.0; Term char = LF.

E-761: Interface = DLL, DLL for Device = E-761, DLL Interface = Board, Parameter = Board number (1 for first E-761 board), Syntax: GCS 1.0; Term char = LF.

E-816: Interface = RS232 or DLL, RS232: Input and output HW handshake must be TRUE. DLL (USB): DLL for Device = E-816, DLL Interface = USB, Parameter = Serial no. of system to connect to. Syntax: GCS 1.0; Term char = LF.

E-861: Single Device: Interface = RS232 or DLL, RS232: Input and output HW handshake must be FALSE. DLL (USB): DLL for Device = E-861, DLL Interface = USB, Parameter = Serial no. of system to connect to.
DaisyChain: Interface = DLL, DLL for Device = E-861, DLL Interface = RS232_DC or USB_DC, Parameter = Number of device in chain, Register DC: FALSE. Syntax: GCS 2.0; Term char = LF.

F-206: Interface = RS232, GPIB or TCP/IP, The error status will not be cleared by this VI. The first ERR? query will report a hidden error with error code 1, which will be cleared during system initialization (INI). RS232: Input and output handshake settings must be FALSE, Syntax: GCS 1.0; Term char = LF.

M-8X0: Interface = RS232, GPIB or TCP/IP. RS232: Input and output handshake settings must be FALSE, Syntax: GCS 1.0; Term char = LF.

Mercury: Interface = DLL, DLL for Device = Mercury, DLL Interface = RS232 (even if using USB), Parameter = empty string, RS232 baud rate = same as controller hardware setting (even if using USB), Syntax: GCS 1.0; Term char = LF.

2.2.14. PI Open Interface.vi (Communication.llb)

Valid for	All except analog systems
Input	Number of systems (1), Interface configuration (RS232, 5000, COM1, 57600), DLL Interface configuration (C-843, Board, 1), TCP/IP configuration (localhost, 3000, 0), Flow control (All FALSE, x13, x11, x0), Bitt settings and parity (8, 1bit, no parity), Termination character (LF), Syntax (GCS 1.0), String to send (*idn?)
Output	Error out
Remarks	Establishes communication with up to four connected systems. The interface and error statuses of all connected systems are cleared by this VI, which sends XXX (no command), *IDN? and ERR?. See “PI Open Interface of one system.vi” for control settings.

2.2.15. PI Receive String.vi (Communication.llb)

Valid for	All systems
Input	System number (1), Strip spaces? (F), Error in (no error)
Output	String read, Bytes read, Error out
Remarks	Read string from selected system.

2.2.16. PI Send String.vi (Communication.llb)

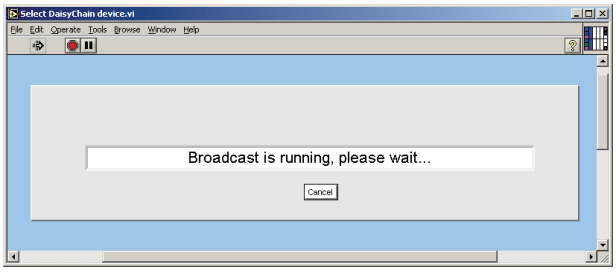
Valid for	All systems
Input	System number (1), String to send (empty string), Attach termination char.? (T), Error in (no error)
Output	Error out
Remarks	Sends command with or without trailing termination character to selected system.

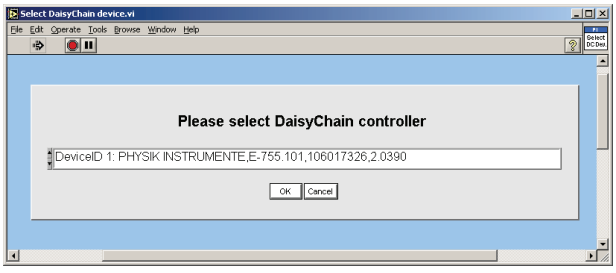
2.2.17. PI VISA Receive Characters.vi (Communication.llb)

Valid for	C-702, C-848, C-867, C-880, C-880K005, E-516, E-517, E-712, E-725, E-753, E-709, E-816, E-861, F-206, M-8X0 (but must be present in Communication.llb for all other systems also)
Input	System number (1), Bytes to read (1), Error in (no error)
Output	String read, Bytes read, Error out
Remarks	This vi reads n bytes (characters) via the chosen VISA interface. Sub-vi for "PI Receive String.vi".

2.2.18. Select DaisyChain device.vi (Communication.llb)

Valid for	C-867, E-755, E-861
Input	System no. (1), Controller name (empty string), DLL Interface (RS232_DC), Timeout (5000), Register DaisyChain (FALSE), RS232 Portnumber (COM 1), RS232 Baudrate (57600), Error in (no error) C-867: Controller name = “C-867”, DLL Interface = RS232_DC E-755: Controller name = “E-755”, DLL Interface = RS232_DC E-861: Controller name = “E-861”, DLL Interface = RS232_DC or USB_DC
Output	Selected DaisyChain controller, DeviceID out, Error out
Remarks	Performs a broadcast, returns names of all controllers matching "Controller name" and lets the user select the appropriate controller from a ring control. VI will also stop if Cancel is TRUE.





2.2.19. Select USB device.vi (Communication.llb)

Valid for	C-866, C-867, E-517, E-709, E-712, E-725, E-816, E-861
Input	System no. (1), Controller name (empty string), DLL Interface (USB), Timeout (5000), Error in (no error) C-866: Controller name = “C-866”, DLL Interface = USB C-867: Controller name = “C-867”, DLL Interface = USB E-517: Controller name = “E-517”, DLL Interface = USB

E-709: Controller name = “E-709”, DLL Interface = USB

E-712: Controller name = “E-712”, DLL Interface = USB

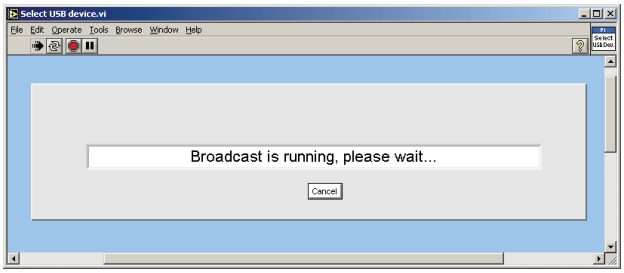
E-725: Controller name = “E-725”, DLL Interface = USB

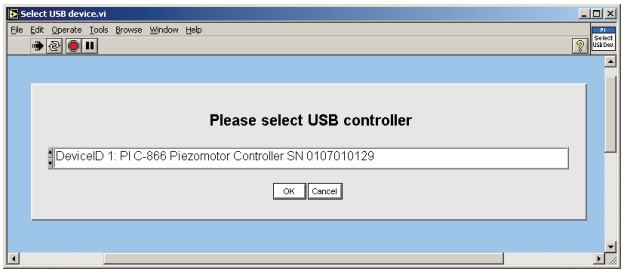
E-816: Controller name = “E-816”, DLL Interface = USB

E-861: Controller name = “E-861”, DLL Interface = USB

Output Selected USB controller, DeviceID out, Error out

Remarks Performs a broadcast, returns names of all controllers matching "Controller name" and lets the user select the appropriate controller from a ring control. VI will also stop if Cancel is TRUE.





2.2.20. Set logging mode.vi (Communication.llb)

Valid for All systems

Input System number (1), Logging mode (OFF), Path in (empty path), File dialog (T)

Output Error out

Remarks Sets logging mode for all communication interfaces. When Logging mode is ON, each string sent to or received from the controller is written to a .txt file for debugging. When File dialog is TRUE, a dialog box will pop up where the file to write can be selected, otherwise Path in must contain a valid path to a .txt file. Depending on the call chain of "Set logging mode.vi", the VI will either stop (correct behavior when called from another VI) or it will remain idle (correct behavior when command VIs from this driver set are to be run manually, i.e. non-programmatically). In the latter case do not forget to press the STOP button when you have finished working with the command VIs.

2.2.21. Syntax.ctl (Communication.llb)

Valid for All systems

Input None

Output None

Remarks Type definition for GCS version.

2.2.22. Termination character.ctl (Communication.llb)

Valid for All systems

Input None

Output None

Remarks Type definition for termination character.

2.3. File handling VIs ("File handling.llb")

2.3.1. Array File.vi (File handling.llb)

Valid for	Analog systems, C-867, C-880, E-517, E-712, E-725, E-761, E-861, F-206, M-8X0. To support analog interfacing, VI must be present for E-816 also.
Input	Path (empty path), Read (F)/Delete (F), ArrayName (empty string), Error in (no error)
Output	Array names, Error out
Remarks	This vi checks the names of all arrays in a data file or deletes a given array from a data file.

2.3.2. File handler.vi (File handling.llb)

Valid for	All systems
Input	Path in (empty path), Read (F) or write (T)? (F), With dialog? (F), Write new file? (F), Default file name (empty string), Extension (txt)
Output	Path out, Cancelled? (T/F), Data added? (T/F)
Remarks	This vi handles file name selections with or without a user interface. Files can be read or written. Path in is the path to the file to read or write. Extension is the file extension for the file to write (e.g. txt, jpg). If Read (F) or write (T) is TRUE, Extension must be given and entry must not have a dot. If With dialog? is TRUE, in every case a dialog box will pop up where the file to read or write can be selected. Default file name is used for naming suggestions if a dialog pops up. If Read (F) or write (T)? is TRUE and Write new file? is TRUE, a dialog box will pop up if the selected file name already exists. If Write new file? is FALSE and the selected file name already exists, a dialog box will pop up to ask if data should be added. Data added? indicates if data was added to an existing file. Cancelled? indicates if the user has cancelled the operation. Path out is NotAPath if operation was cancelled or not successful and contains the selected path for the file which was read or written if the operation was successful.

2.3.3. GetDataFormat.vi (File handling.llb)

Valid for	Analog systems, C-702, C843, C-866, C-867, C-880, E-517, E-709, E-710, E-712, E-725, E-753, E-755, E-761, E-861, F-206, M-8X0. To support analog interfacing, VI must be present for E-816 also.
Input	IOSource (Read (F)/Write (F), Path (empty path), ArrayName (empty string), Datastream (empty string)), Error in (no error)
Output	Header out (Separator, NDim, Remarks), DataOK, Found Header, Data Type, NData, Names out, Sample time, Error out
Remarks	This vi checks the format of a data file. See separate manual "GCSData_User_SM146E.pdf" and control descriptions in the diagram for more information.

2.3.4. MatrixIO.vi (File handling.llb)

Valid for	Analog systems, C-843, C-866, C-867, C-880, E-517, E-709, E-712, E-725, E-753, E-755, E-761, E-861, F-206, M-8X0. To support analog interfacing, VI must be present for E-816 also.
Input	IOSource (Read (F)/Write (F), Path (empty path), ArrayName (empty

	string), Datastream (empty string)), Header in (Separator (\t), NDim (0), Remarks (empty string)), Data names (XName (empty string), YName (empty string), ZName (empty string)), XArray in (empty num. array), YArray in (empty num. array), ZMatrix in (empty 2D num. array), Sample time in (0), (Error in (no error)
Output	Datastream out, Header out (Separator, NDim, Remarks), Data names out (XName, YName, ZName), XArray out, YArray out, ZMatrix out, Sample time out, Error out
Remarks	This vi reads or writes data files in matrix format. See separate manual "GCSData_User_SM146E.pdf" and control descriptions in the diagram for more information.

2.3.5. TableIO.vi (File handling.llb)

Valid for	Analog systems, C-702, C-843, C-866, C-867, C-880, E-517, E-709, E-710, E-712, E-725, E-753, E-755, E-761, E-861, F-206, M-8X0, Mercury. To support analog interfacing, VI must be present for E-816 also.
Input	IOSource (Read (F)/Write (F), Path (empty path), ArrayName (empty string), Datastream (empty string)), Header in (Separator (\t), NDim (0), Remarks (empty string)), Names in (empty string array), Table in (empty 2D num. array), Sample time in (0), (Error in (no error)
Output	Datastream out, Header out (Separator, NDim, Remarks), Names out, Table out, Sample time out, Error out
Remarks	This vi reads or writes data files in table format. See separate manual "GCSData_User_SM146E.pdf" and control descriptions in the diagram for more information. Sub-VI for "DRR?.vi".

2.4. General Command VIs ("General command.llb"):

2.4.1. *IDN?.vi (General command.llb)

Valid for	All systems
Input	System number (1), Error in (no error)
Output	Identification, Error out
Remarks	Returns system identification string. E-816: This command cannot be issued to a slave.

2.4.2. Controller names.ctl (General command.llb)

Valid for	All systems
Input	None
Output	None
Remarks	Type definition for control Controller names.

2.4.3. CSV?.vi (General command.llb)

Valid for	C-702, C-867, E-517, E-709, E-712, E-725, E-753, E-755, E-761, E-861
Input	System number (1), Error in (no error)
Output	Syntax, Error out
Remarks	Returns current GCS syntax version.

2.4.4. Define connected axes.vi (General command.llb)

Valid for	All systems
Input	System number (1), Read from controller?(F), Invert order?(F), Conn. axes (empty string array), Error in (no error) Analog: Only supported when called by Analog_Configuration_Setup.vi C-702: Read from controller = TRUE, Invert order = TRUE C-848: Read from controller = TRUE, Invert order = TRUE C-880: Read from controller = TRUE, Invert order = TRUE F-206: Read from controller = FALSE, Invert order = FALSE, Connected axes = X,Y,Z,U,V,W, (A,B,K,L,M optional) M-8X0: Read from controller = FALSE, Invert order = FALSE, Connected axes = X,Y,Z,U,V,W, (A,B optional) All other systems: Read from controller = TRUE, Invert order = FALSE
Output	Connected axes out, Error out
Remarks	Writes connected axes into Global2 (Array).vi. **This VI is called automatically by "XXXX_Configuration_Setup.vi" (with XXXX being the PI product number of your system) and must be completed successfully before any other axis-specific command VI is called.** Requires "SAI?.vi" to be present.

2.4.5. Define connected systems (Array).vi (General command.llb)

Valid for	All systems
Input	Controller names (array of Enum controls, none), Change only one system? (F), System number (1), Error in (no error) Analog system: Only supported when called by Analog_Configuration_Setup.vi
Output	Controller names out, Error out
Remarks	Defines connected systems and writes controller names into Global2 (Array).vi. **This VI is called automatically by "XXXX_Configuration_Setup.vi" (with XXXX being the PI product number of your system) and must be completed successfully before"General wait for movement to stop.vi" is called.** If Change only one system? is FALSE, all entries from Controller names are written into "Global2 (Array).vi". If Change only one system? is TRUE, only the entry for the given system number is overwritten in "Global2 (Array).vi".

2.4.6. ERR?.vi (General command.llb)

Valid for	All systems.
Input	System number (1), Error in (no error)
Output	Controller error (T/F), Error out

Analog: VI does not report any errors.

Remarks Returns error information. Controller error is TRUE if selected system reports error code ≠ 0. See appendix A of this manual for a list of PI error codes and use “GCSTranslateError.vi” to translate error codes into error descriptions programmatically.

E-816: This command cannot be issued to a slave.

2.4.7. Global2 (Array).vi (General command.llb)

Valid for All systems

Input System (array of Conn. axes (empty string array), Controller name (Enum control, none))

Output None

Remarks A global variable which contains identifiers for all connected axes of all connected systems and the names of all connected systems.

2.4.8. HLP?.vi (General command.llb)

Valid for Analog systems, C-702, C-843, C-843.PM, C-848, C-865, C-866, C-867, C-880, E-516, E-517, E-709, E-712, E-725, E-710, E-753, E-755, E-761, E-861, F-206, Mercury, M-8X0 (but must be present for all other systems also) .

Input System number (1), Error in (no error)

Output Help string, Error out

Remarks Returns help string.

F-206, M-8X0: Check HELP answer to determine if HLP? is supported. HLP? and HELP are equivalent.

2.4.9. HLT.vi (General command.llb)

Valid for C-702, C-843, C-843.PM, C-844, C-848, C-865, C-866, C-867, C-880, E-517, E-710, E-755, E-761, E-861, Mercury

Input System number (1), Affected axes (empty string array), All axes? (F), Axis identifier? (T), Error in (no error)

C-702: If All axes? = TRUE, then Axis identifier? must be TRUE

C-843: If All axes? = TRUE, then Axis identifier? must be TRUE

C-843.PM: If All axes? = TRUE, then Axis identifier? must be TRUE

C-844: If All axes? = TRUE, then Axis identifier? must be TRUE

C-848: If All axes? = TRUE, then Axis identifier? must be TRUE

C-865: If All axes? = TRUE, then Axis identifier? can be FALSE

C-866: If All axes? = TRUE, then Axis identifier? can be FALSE

C-867: If All axes? = TRUE, then Axis identifier? can be FALSE

C-880: If All axes? = TRUE, then Axis identifier? must be TRUE

E-517: If All axes? = TRUE, then Axis identifier? can be FALSE

E-710: If All axes? = TRUE, then Axis identifier? must be TRUE

E-755: If All axes? = TRUE, then Axis identifier? must be TRUE

E-761: If All axes? = TRUE, then Axis identifier? can be FALSE

E-861: If All axes? = TRUE, then Axis identifier? can be FALSE

Mercury: If All axes? = TRUE, then Axis identifier? can be FALSE

Output Error out

Remarks Stops motion of specified axes. HLT sets error code 10, call “ERR?.vi” to reset error after HLT has been called.

2.4.10. HPA?.vi (General command.llb)

Valid for C-843, C-867, E-517, E-709, E-710, E-712, E-725, E-753, E-755, E-761, E-861, Mercury

Input System number (1), Error in (no error)

Output Parameter help string, Error out

Remarks Returns a help string containing information about valid parameter IDs.

2.4.11. Initialize Global2.vi (General command.llb)

Valid for All systems

Input System number (1), Error in (no error)

Output Error out

Remarks This VI initializes Global2 (Array) according to the given system no.

2.4.12. MOV.vi (General command.llb)

Valid for Analog systems, C-702, C-843, C-843.PM, C-844, C-848, C-865, C-866, C-867, C-880, C-880K005, E-516, E-517, E-709, E-710, E-712, E-725, E-753, E-755, E-761, E-816, E-861, F-206, M-8X0, Mercury

Input System number (1), Axes to move (empty string array), Position values (empty num. array, 0), No. of digits (4), Error in (no error)

C-867: This command works only in closed-loop operation.
Motion commands like MOV are not allowed when a joystick is active on the axis.

C-880K005: VI only supported when called through PI_Multix.vi

E-517: Motion commands like MOV are not allowed when the E-517 is in OFFLINE mode or when the wave generator output is active. When a macro is running on the E-517, MOV will be executed not until the macro is finished or stopped. See "Control Value Generation" and "Control Modes" in the E-517 User manual for details.

E-709: Motion commands are not allowed when the wave generator is active or the analog input is used for target generation.

E-712: Motion commands are not allowed when a wave generator is active or the analog input is used for target generation.

E-725: Motion commands are not allowed when a wave generator is active or the analog input is used for target generation.

E-753: Motion commands are not allowed when the wave generator is active or the analog input is used for target generation.

E-755: Command not available for E-755.101.

E-816: Only one axis per command allowed. It is necessary to wait a certain time before sending the next command to prevent it from being lost.

E-861: This command works only in closed-loop operation. With open-loop systems, use OAD, OSM, OMA or OMR instead to command motion.

Motion commands like MOV are not allowed when a joystick is active on the axis.

F-206: No mix between F-206 axes X,Y,Z,U,V,W and separate axes A,B allowed

Output: Error out

Remarks: Moves specified axes to specified absolute positions. No. of digits is the number of digits after the decimal point in the position value(s) that will be sent.

E-710: See also “NMOV.vi” in “Old commands.llb”.

2.4.13. MOV?.vi (General command.llb)

Valid for: C-702, C-843, C-843.PM, C-844, C-848, C-865, C-866, C-867, C-880, E-516, E-517, E-709, E-710, E-712, E-725, E-753, E-755, E-761, E-816, E-861, Mercury

Input: System number (1), Axes to query (empty string array), All axes? (F), Axis identifier? (T), Error in (no error)

C-702: If All axes? = TRUE, then Axis identifier? can be FALSE

C-843: If All axes? = TRUE, then Axis identifier? must be TRUE

C-843.PM: If All axes? = TRUE, then Axis identifier? must be TRUE

C-844: If All axes? = TRUE, then Axis identifier? must be TRUE

C-848: If All axes? = TRUE, then Axis identifier? can be FALSE

C-865: If All axes? = TRUE, then Axis identifier? can be FALSE

C-866: If All axes? = TRUE, then Axis identifier? can be FALSE

C-867: If All axes? = TRUE, then Axis identifier? can be FALSE

C-880: If All axes? = TRUE, then Axis identifier? can be FALSE

E-516: If All axes? = TRUE, then Axis identifier? must be TRUE

E-517: If All axes? = TRUE, then Axis identifier? can be FALSE

E-709: If All axes? = TRUE, then Axis identifier? can be FALSE.

E-710: If All axes? = TRUE, then Axis identifier? must be TRUE

E-712: If All axes? = TRUE, then Axis identifier? can be FALSE.

E-725: If All axes? = TRUE, then Axis identifier? can be FALSE.

E-753: If All axes? = TRUE, then Axis identifier? can be FALSE.

E-755: If All axes? = TRUE, then Axis identifier? can be FALSE. Command not available for E-755.101.

E-761: If All axes? = TRUE, then Axis identifier? can be FALSE

E-816: All axes? = FALSE, only one axis per command allowed.

E-861: If All axes? = TRUE, then Axis identifier? can be FALSE

F-206: Command has different implementation, please use MOV?_old.vi

M-8X0: Command has different implementation, please use MOV?_old.vi

Mercury: If All axes? = TRUE, then Axis identifier? can be FALSE

Output: Target position, Error out

Remarks: Returns commanded target position.

2.4.14. MVR.vi (General command.llb)

Valid for: Analog systems, C-702, C-843, C-843.PM, C-844, C-848, C-865, C-866, C-867, C-880, C-880K005, E-516, E-517, E-709, E-710, E-712, E-725, E-753, E-755, E-761, E-816, E-861, Mercury

Input: System number (1), Axes to move (empty string array), Position values (empty num. array, 0), No. of digits (4), Error in (no error)

C-867: This command works only in closed-loop operation.
Motion commands like MVR are not allowed when a joystick is active on the axis.

C-880K005: VI only supported when called through PI_Multix.vi

E-755: Command not available for E-755.101.

E-816: Only one axis per command allowed. It is necessary to wait a certain time before sending the next command to prevent it from being lost.

E-861: This command works only in closed-loop operation. With open-loop systems, use OAD, OSM, OMA or OMR instead to command motion.
Motion commands like MVR are not allowed when a joystick is active on the axis.

Output: Error out

Remarks: Moves specified axes **relative** to current position. No. of digits is the number of digits after the decimal point in the position value(s) that will be sent.

E-517: Motion commands like MVR are not allowed when the E-517 is in OFFLINE mode or when the wave generator output is active. When a macro is running on the E-517, MVR will be executed not until the macro is finished or stopped. See "Control Value Generation" and "Control Modes" in the E-517 User manual for details.

E-709: Motion commands are not allowed when the wave generator is active or the analog input is used for target generation.

E-710: See also “NMVR.vi” in “Old commands.llb”.

E-712: Motion commands are not allowed when a wave generator is active or the analog input is used for target generation.

E-725: Motion commands are not allowed when a wave generator is active or the analog input is used for target generation.

E-753: Motion commands are not allowed when the wave generator is active or the analog input is used for target generation.

2.4.15. MWG.vi (General command.llb)

Valid for: F-206, M-8X0 (but must be present for C-866, C-867, C-880, E-761 and E-861 also)

Input: System number (1), Axes to move (empty string array), Position values (empty num. array, 0), No. of digits (4), Error in (no error)

F-206: No mix between F-206 axes X,Y,Z,U,V,W and separate axes A,B allowed

M-8X0: Only for controllers based on C-842.80. Check HELP answer to find out if MWG is supported. If not supported, must be present anyway.

Output: Error out

Remarks: Moves specified axes to absolute position without updating graphics on the controller screen (fast move). No. of digits is the number of digits after the decimal point in the position value(s) that will be sent. “Required by 1D

Scan.vi" and "2D Scan.vi".

2.4.16. ONT?.vi (General command.llb)

Valid for	C-702, C-843, C-843.PM, C-848, C-865, C-866, C-867, C-880, E-516, E-517, E-709, E-710, E-712, E-725, E-753, E-755, E-761, E-816, E-861, Mercury (but must be present for all other systems also)
Input	System number (1), Axes to query (empty string array), All axes? (F), Axis identifier? (T), Error in (no error) C-702: If All axes? = TRUE, then Axis identifier? must be TRUE C-843: If All axes? = TRUE, then Axis identifier? must be TRUE C-843.PM: If All axes? = TRUE, then Axis identifier? must be TRUE C-848: If All axes? = TRUE, then Axis identifier? must be TRUE C-865: If All axes? = TRUE, then Axis identifier? can be FALSE C-866: If All axes? = TRUE, then Axis identifier? can be FALSE C-867: If All axes? = TRUE, then Axis identifier? can be FALSE. C-880: If All axes? = TRUE, then Axis identifier? must be TRUE E-516: If All axes? = TRUE, then Axis identifier? can be FALSE. E-517: If All axes? = TRUE, then Axis identifier? can be FALSE E-709: If All axes? = TRUE, then Axis identifier? can be FALSE. E-710: If All axes? = TRUE, then Axis identifier? can be FALSE. E-712: If All axes? = TRUE, then Axis identifier? can be FALSE. E-725: If All axes? = TRUE, then Axis identifier? can be FALSE. E-753: If All axes? = TRUE, then Axis identifier? can be FALSE. E-755: If All axes? = TRUE, then Axis identifier? can be FALSE. Command not available for E-755.101 E-761: If All axes? = TRUE, then Axis identifier? can be FALSE. E-816: All axes? = FALSE, only one axis per command allowed. E-861: If All axes? = TRUE, then Axis identifier? can be FALSE Mercury: If All axes? = TRUE, then Axis identifier? can be FALSE.
Output	Axis on target? (T/F), Error out
Remarks	Indicates whether or not queried axis is at target position.

2.4.17. POS?.vi (General command.llb)

Valid for	Analog systems, C-702, C-843, C-843.PM, C-844, C-848, C-865, C-866, C-867, C-880, C-880K005, E-516, E-517, E-709, E-710, E-712, E-725, E-753, E-755, E-761, E-816, E-861, F-206, M-8X0, Mercury
Input	System number (1), Axes to query (empty string array), All axes? (F), Axis identifier? (T), Error in (no error) C-880K005: VI only supported when called through PI_Multix.vi E-516: If All axes? = TRUE, then Axis identifier? must be TRUE E-710: If All axes? = TRUE, then Axis identifier? must be TRUE E-755: If All axes? = TRUE, then Axis identifier? can be FALSE. Command not available for E-755.101.

E-816: All axes? = FALSE, only one axis per command allowed.

F-206: If All axes? = TRUE, then Axis identifier? must be TRUE

M-8X0: If All axes? = TRUE, then Axis identifier? must be TRUE

All other systems: If All axes? = TRUE, then Axis identifier? can be FALSE.

Output Position, Error out

Remarks Returns position information (actual or target position, depending on system).

F-206: Returned position value is the commanded target position for the axis.

M-8X0: Returned position value is the commanded target position for the axis.

2.4.18. SAI?.vi (General command.llb)

Valid for Analog systems, C-702, C-843, C-843.PM, C-844, C-848, C-865, C-866, C-867, C-880, C-880K005, E-516, E-517, E-709, E-710, E-712, E-725, E-753, E-755, E-761, E-816, E-861, Mercury (but must be present in “General command.llb” for all other systems also)

Input System number (1), Invert order? (F), SAI? ALL (F), Write to Global2? (F), Error in (no error)

Analog: Invert order should be FALSE, SAI? ALL must be FALSE

C-702: Invert order should be TRUE, SAI? ALL must be FALSE

C-843: Invert order should be FALSE, SAI? ALL is supported

C-843.PM: Invert order should be FALSE, SAI? ALL must be FALSE

C-844: Invert order should be FALSE, SAI? ALL must be FALSE

C-848: Invert order should be TRUE, SAI? ALL must be FALSE

C-865: Invert order should be FALSE, SAI? ALL is supported

C-866: Invert order should be FALSE, SAI? ALL is supported

C-867: Invert order should be FALSE, SAI? ALL is supported

C-880: Invert order should be TRUE, SAI? ALL must be FALSE to read all configured axis IDs and must be TRUE to get all physically defined axis IDs

C-880K005: VI only supported when called through PI_Multix.vi, SAI? ALL must be FALSE

E-516: Invert order should be FALSE, SAI? ALL must be FALSE

E-517: Invert order should be FALSE, SAI? ALL is supported

E-709: Invert order should be FALSE, SAI? ALL is supported

E-710: Invert order should be FALSE, SAI? ALL is supported

E-712: Invert order should be FALSE, SAI? ALL is supported

E-725: Invert order should be FALSE, SAI? ALL is supported

E-753: Invert order should be FALSE, SAI? ALL is supported

E-755: Invert order should be FALSE, SAI? ALL is supported

E-761: Invert order should be FALSE, SAI? ALL is supported

E-816: Invert order should be FALSE, SAI? ALL must be FALSE

E-861: Invert order should be FALSE, SAI? ALL is supported

Mercury: Invert order should be FALSE, SAI? ALL is supported

Output Connected axes, Error out

Remarks

Returns axis identifiers of all configured axes and writes them into Global2 (Array).vi. If SAI? ALL is TRUE, all physically available axes are returned, no matter if configured or not. Required by “Define connected axes.vi”. If SAI? ALL is TRUE, returned identifiers normally may not be written to “Global2 (Array).vi”. To write them to “Global2 (Array).vi” nevertheless, set Write to Global2? to TRUE.

E-816: This command cannot be issued to a slave.

2.4.19. SPA.vi (General command.llb)

Valid for

C-702, C-843, C-843.PM, C-844, C-848, C-865, C-866, C-867, C-880, C-880K005, E-516, E-517, E-709, E-710, E-712, E-725, E-753, E-755, E-761, E-816, E-861, Mercury

Input

System number (1), Axis to set (empty string array), Parameter number (empty num. array, 0), Parameter number (hex) (empty hex. array, 0), Parameter value (empty num. array, 0), No. of digits (4), Parameter string (empty string array), Parameter no. format (Decimal: FALSE) (F), Parameter format (Num.: FALSE) (F), Error in (no error)

C-702: Parameter no. format is FALSE (decimal).

WARNING

This command is for setting hardware-specific parameters. Wrong values may lead to improper operation or damage of your hardware! Change settings only after consultation with PI.

C-843: Parameter no. format is FALSE (decimal).

WARNING

This command is primarily for setting hardware-specific parameters of non-PI stages connected to the controller. Please refer to the stage manual for valid parameter settings. If you have a PI stage connected, please do not change any parameters except P (1), I (2), D (3), I-limit (4) and VFF (5).

C-843.PM: Parameter no. format is FALSE (decimal). See C-843 for warnings.

C-848: Parameter no. format is FALSE (decimal). See C-880 for warnings.

C-865: Parameter no. format is FALSE (decimal). See C-843 for warnings.

C-866: Parameter no. format is FALSE (decimal). See C-843 for warnings.

C-867: Parameter no. format is TRUE (hex.). Use “HPA?.vi” to get valid parameter numbers or see the C-867 User Manual. Only one parameter value for only one axis per command allowed. See E-710 for warnings.

C-880: Parameter no. format is FALSE (decimal).

WARNING

This command is for setting hardware-specific parameters of non-PI stages connected to the controller. Please refer to the stage manual for valid parameter settings. If you have a PI stage connected, please do not change any parameters except P (1), I (2), D (3), I-limit (4) and VFF (5).The most important parameter numbers are:

- 1: P-term (0 to 32767)
- 2: I-term (0 to 32767)
- 3: D-term (0 to 32767)
- 4: I-Limit (integration limit) (0 to 32767)

- 5: VFF (velocity feed forward) (0 to 32767)
- 7: motor bias (-32767 to 32767)
- 8: maximum position error (0 to 32767)
- 9: maximum value for the motor output (0 to 32767)
- 10: maximum velocity (allowed range depends on stage)
- 11: maximum allowed acceleration (allowed range depends on stage)
- 13: maximum allowed Jerk (allowed range depends on stage)
- 14, 15: reserved

C-880K005: VI only supported when called through PI_Multix.vi. See C-880 for warnings and description of parameter numbers.

E-516: Parameter no. format is FALSE (decimal).

WARNING

This command is for setting hardware-specific calibration parameters, except parameter number 268500993. Incorrect values may lead to improper operation.

The following parameter numbers are valid:

- 7: Ksen (Coefficient of Sensor K_s). When sensor output change is 1V, the position change of stage is K_s (µm). (- 3.402823466e+38F to 3.402823466e+38F)
- 8: Osen (Offset of Sensor Os). When sensor output is 0V, the actual position of stage is Os (µm). (- 3.402823466e+38F to 3.402823466e+38F)
- 9: Kpzt (Coefficient of PZT voltage amplifier Kpzt). When DAC output change is 1V, the PZT Voltage change is Kpzt (V) (- 3.402823466e+38F to 3.402823466e+38F)
- 10: Opzt (Offset of PZT voltage amplifier Opzt) When DAC output is 0V, the PZT Voltage is Opzt (V) (- 3.402823466e+38F to 3.402823466e+38F)
- 117442816: Tolerance for ONT software emulation (µm) (0 < value < 1000)

E-517: Parameter no. format is TRUE (hex.). Use “HPA?.vi” to get valid parameter numbers or see the E-517 User Manual. See E-710 for warnings.

E-709: Parameter no. format is TRUE (hex.). Only one parameter value for only one axis per command allowed. Use “HPA?.vi” to get valid parameter numbers or see the E-709 User Manual. See E-710 for warnings.

E-710: Parameter no. format is TRUE (hex.) Use “HPA?.vi” to get valid parameter numbers or see the E7XX_GCS_DLL Manual.

WARNING

This command is for setting hardware-specific parameters. Wrong values may lead to improper operation or damage of your hardware!

E-712: Parameter no. format is TRUE (hex.). Use “HPA?.vi” to get valid parameter numbers or see the E-712 User Manual. See E-710 for warnings. Do not set more than 10 parameters at once.

E-725: Parameter no. format is TRUE (hex.). Use “HPA?.vi” to get valid parameter numbers or see the E-725 User Manual. See E-710 for warnings. Do not set more than 10 parameters at once.

E-753: Parameter no. format is TRUE (hex.). Use “HPA?.vi” to get valid parameter

numbers or see the E-753 User Manual. See E-710 for warnings.

E-755: Parameter no. format is TRUE (hex.). Use “HPA?.vi” to get valid parameter numbers or see the E-755 User Manual. See E-710 for warnings.

E-761: Parameter no. format is TRUE (hex.). Use “HPA?.vi” to get valid parameter numbers or see the User Manual. See E-710 for warnings. Do not set more than 10 parameters at once.

E-816: Parameter no. format is FALSE (decimal). See E-516 for warnings and a description of parameter numbers. Each command limited to setting one parameter for only one axis.

E-861: Parameter no. format is TRUE (hex.). Use “HPA?.vi” to get valid parameter numbers or see the E-861 User Manual. Only one parameter value for only one axis per command allowed. See E-710 for warnings.

Mercury: Parameter no. format is FALSE (decimal). See C-843 for warning.

Output Controller error (T/F), Error out

Remarks Sets parameters, waits 100 ms and queries ERR?. For axis-related parameters, Axis to set is the axis name; for piezo- or sensor-related parameters, the channel number; otherwise a parameter-related code. If parameter number is in decimal format, use Parameter number input, for hexadecimal parameter numbers use Parameter number (hex.) input and switch Parameter no. format to TRUE. For numeric parameter values use Parameter value input, for parameter strings use Parameter string input and switch Parameter format to TRUE. Do not mix decimal and hex. parameter numbers or numeric and string parameter values in one call. Parameter numbers which can be set depend on current CCL level. See GCS DLL manual for available parameter numbers and values. No. of digits is the number of digits after the decimal point in the numeric parameter value(s) that will be sent. Controller error is TRUE if selected system reports error code ≠ 0.

C-867, E-516, E-517, E-710, E-712, E-725, E-753, E-755, E-861: The SPA command saves the parameters in RAM only. To save the currently valid parameters to flash ROM, where they become the power-on defaults, you must run WPA.vi. Parameter changes not saved with WPA will be lost when the controller is powered off.

C-843: For precision and convenience with gearbox systems, the counts per physical unit factor can be entered as numerator and denominator of a fraction (parameters 14 and 15).

E-816: This command cannot be issued to a slave.

E-761: The SPA command saves the parameters in RAM only. To save the currently valid parameters to flash ROM, where they become the power-on defaults, you must run WPA.vi. Parameter changes not saved with WPA will be lost when the PC is powered off or the E-761 is rebooted.

Mercury: The SPA command saves the parameters in RAM only. Use PIStageEditor.exe to change parameters or add new stages to the data base permanently.

2.4.20. SPA?.vi (General command.llb)

Valid for C-702, C-843, C-843.PM, C-844, C-848, C-865, C-866, C-867, C-880, C-880K005, E-516, E-517, E-709, E-710, E-712, E-725, E-753, E-755, E-761, E-816, E-861, Mercury, M-8X0

Input

System number (1), Axes to query (empty string array), Parameter no. format (Decimal: FALSE) (F), Without axes? (F), Parameter no. (empty num. array, 0), Parameter no. (hex) (empty hex. array, 0), Error in (no error)

C-702: Parameter no. format is FALSE (decimal).

C-843: Parameter no. format is FALSE (decimal).

C-843.PM: Parameter no. format is FALSE (decimal).

C-848: Parameter no. format is FALSE (decimal).

C-865: Parameter no. format is FALSE (decimal). Parameter number 25 is read-only.

C-866: Parameter no. format is FALSE (decimal). Parameter number 25 is read-only.

C-867: Parameter no. format is TRUE (hex.). Use “HPA?.vi” to get valid parameter numbers. Only one parameter value for only one axis per command allowed. Use Without axes? = TRUE for a query of all parameters.

C-880: Parameter no. format is FALSE (decimal). Additional read-only parameter numbers are:

- 14: Numerator of the counts per physical unit factor (1 to 2147483647) (factor = num./denom.))
- 15: Denominator of the counts per physical unit factor (1 to 2147483647) (factor = num./denom.)
- 16: Drive mode: 0=Analog 1=PWM
- 19: Axis type: 0=Linear 1=Rotary
- 20: Reference switch: 0=no present, 1=present
- 28: Reference status: 0=axis not referenced; 1=axis is referenced

C-880K005: VI only supported when called through PI_Multix.vi

E-516: Parameter no. format is FALSE (decimal).

E-517: Parameter no. format is TRUE (hex.). Use “HPA?.vi” to get valid parameter numbers.

E-709: Parameter no. format is TRUE (hex.). Only one parameter value for only one axis per command allowed. Use Without axes? = TRUE for a query of all parameters. Use “HPA?.vi” to get valid parameter numbers.

E-710: Parameter no. format is TRUE (hex.). Use “HPA?.vi” to get valid parameter numbers.

E-712: Parameter no. format is TRUE (hex.). Use “HPA?.vi” to get valid parameter numbers. Do not query more than 10 parameter no. at once (except with Without axes? = TRUE).

E-725: Parameter no. format is TRUE (hex.). Use “HPA?.vi” to get valid parameter numbers. Do not query more than 10 parameter no. at once (except with Without axes? = TRUE).

E-753: Parameter no. format is TRUE (hex.). Use “HPA?.vi” to get valid parameter numbers.

E-755: Parameter no. format is TRUE (hex.). Use “HPA?.vi” to get valid parameter numbers.

E-761: Parameter no. format is TRUE (hex.). Use “HPA?.vi” to get valid parameter numbers. Do not query more than 10 parameter no. at once (except with Without axes? = TRUE).

E-816: Parameter no. format is FALSE (decimal).

E-861: Parameter no. format is TRUE (hex.). Use “HPA?.vi” to get valid parameter numbers. Only one parameter value for only one axis per command allowed. Use Without axes? = TRUE for a query of all parameters.

Mercury: Parameter no. format is FALSE (decimal).

M-8X0: Parameter no. format is FALSE (decimal). Axes to query can be 1 to 6 (corresponds to strut no.). Parameter no. can be 512 (reports if strut is extended or retracted) or 513 (reports commanded strut length). Only one parameter value for only one axis per command allowed.

Output Parameter value, Parameter string, Error out

Remarks Returns parameter values for queried items and parameter numbers. For axis-related parameters, Axis to query is the axis name; for piezo- or sensor-related parameters, the channel number; otherwise a parameter-related code. If parameter number is in decimal format, use “Parameter no.” input, for hexadecimal parameter numbers use “Parameter no. (hex)” input and switch “Parameter no. format” to TRUE. If Without axes? is TRUE, all available parameter for all axes/designators are returned. For parameter numbers which output a string use Parameter string output. See GCS DLL Manual for available parameter numbers.

E-816: This command cannot be issued to a slave

C-843: The following parameter number outputs a string:
60: stage name (maximum 14 characters)

C-843.PM: The following parameter number outputs a string:
60: stage name (maximum 14 characters)

C-866: The following parameter number outputs a string:
60: stage name (maximum 14 characters)

C-865: The following parameter number outputs a string:
60: stage name (maximum 14 characters)

E-861: The following parameter number outputs a string:
60 (0x3C): stage name (maximum 16 characters)

Mercury: The following parameter number outputs a string:
60: stage name (maximum 14 characters)

2.4.21. STP.vi (General command.llb)

Valid for Analog systems, C-702, C-843, C-843.PM, C-844, C-848, C-865, C-866, C-867, C-880, E-516, E-517, Ê-709, E-712, E-725, E-753, E755, E-761, E-861, Mercury (but must be present for E-710 also). To support analog interfacing, VI must be present for E-816 also.

Input System number (1), Affected axes? (empty string array), All axes? (F), Axis identifier? (T), Error in (no error)

Analog: All axes? = TRUE, Axis identifier = FALSE. STP does not set any error code.

C-702: All axes? = TRUE, Axis identifier? = FALSE

C-843: All axes? = TRUE, Axis identifier? = FALSE

C-843.PM: All axes? = TRUE, Axis identifier? = FALSE

C-844: All axes? = TRUE, Axis identifier? = FALSE

C-848: All axes? = TRUE, Axis identifier? = FALSE

C-865: All axes? = TRUE, Axis identifier? = FALSE

C-866: All axes? = TRUE, Axis identifier? = FALSE

C-867: All axes? = TRUE, Axis identifier? = FALSE

C-880: All axes? = TRUE, Axis identifier? = FALSE

E-516: If All axes? = TRUE, then Axis identifier? must be TRUE

E-517: All axes? = TRUE, Axis identifier? = FALSE

E-709: All axes? = TRUE, Axis identifier? = FALSE

E-712: If All axes? = TRUE, then Axis identifier? can be FALSE

E-725: If All axes? = TRUE, then Axis identifier? can be FALSE

E-753: If All axes? = TRUE, then Axis identifier? can be FALSE

E-755: If All axes? = TRUE, then Axis identifier? can be FALSE

E-761: All axes? = TRUE, Axis identifier? = FALSE

E-861: All axes? = TRUE, Axis identifier? = FALSE

Mercury: All axes? = TRUE, Axis identifier? = FALSE

Output Error out

Remarks Stops motion of specified axes. To stop a referencing routine (REF, MNL, MPL) or fast scan routine (FSC, FSA etc.), or AutoZero procedure (ATZ), or wave generator run (WGO), use “#24.vi”. STP sets error code 10, call “ERR?.vi” to reset error after STP has been called.

E-517: STP.vi stops motion of all axes caused by move commands (MOV, MVR, GOH, SVA, SVR). Furthermore, it stops macros (MAC) and wave generator output (WGO).

E-709, E-712, E-725, E-753: STP.vi stops motion of all axes caused by move commands (MOV, MVR, SVA, SVR), by the wave generator (WGO), by analog control input and autozero motion (ATZ).

2.4.22. SVO.vi (General command.llb)

Valid for C-702, C-843, C-843.PM, C-844, C-848, C-865, C-866, C-867, C-880, E-516, E-517, E-709, E-710, E-712, E-725, E-753, E-755, E-761, E-816, E-861, F-206, M-8X0, Mercury

Input System number (1), Without axis ID?(F), Axes to command (empty string array), Servo mode (empty bool. array, F), Error in (no error)

E-755: Without axis ID = FALSE. When the servo mode is switched off, RNP is automatically performed for the corresponding Nexline channel, which could take a few seconds. Command not available for E-755.101.

E-816: Without axis ID = FALSE. Only one axis per command allowed.

F-206: Without axis ID = TRUE, only first field of Servo mode array is valid

M-8X0: Without axis ID = TRUE, only first field of Servo mode array is valid

All other systems: Without axis ID = FALSE

Output Error out

Remarks Sets servo-control mode for given axes. If Without axis ID is TRUE, then Axes to command is ignored and first field of Servo mode array is used.

M-8X0: Check HELP answer to find out if SVO is supported.

E-516, E-517: Make sure that all servo switches on the piezo control electronics are set to "Off" to give the interface/display module complete control over the servo state.

E-861: If you have enabled servo controller is busy for 100 ms, only after this period a next command can be performed.

2.4.23. SVO?.vi (General command.llb)

Valid for C-702, C-843, C-843.PM, C-844, C-848, C-865, C-866, C-867, C-880, E-516, E-517, E-709, E-710, E-712, E-725, E-753, E-755, E-761, E-816, E-861, F-206, M-8X0, Mercury

Input System number (1), Axes to query (empty string array), All axes? (F), Axis identifier? (T), Error in (no error)

C-843, C-843.PM, C-844: If All axes? = TRUE, then Axis identifier? must be TRUE

C-865, C-866: If All axes? = TRUE, then Axis identifier? must be TRUE

E-516, E-710: If All axes? = TRUE, then Axis identifier? must be TRUE

E-816: All axes? = FALSE, only one axis per command allowed.

F-206: All axes? = TRUE, Axis identifier? = FALSE

M-8X0: All axes? = TRUE, Axis identifier? = FALSE

All other systems: If All axes? = TRUE, then Axis identifier? can be FALSE

Output Servo status (T/F), Error out

F-206: Only first field of servo status array is valid

M-8X0: Only first field of servo status array is valid

Remarks Returns servo status of queried axes.

M-8X0: Check HELP answer to find out if SVO? is supported.

2.4.24. VEL.vi (General command.llb)

Valid for Analog systems, C-702, C-843, C-843.PM, C-844, C-848, C-865, C-866, C-867, C-880, C-880K005, E-516, E-517, E-709, E-710, E-712, E-725, E-753, E-755, E-761, E-861, F-206, M-8X0, Mercury. To support analog interfacing, VI must be present for E-816 also.

Input System number (1), Without axis ID? (F), No. of digits (4), Axes to set (empty string array), Velocity values (empty num. array, 0), Error in (no error)

Analog: Without axis ID? = FALSE; Velocity unit is µm/sec

C-867: Without axis ID? = FALSE. Velocity unit is mm/s.

C-880: Without axis ID? = FALSE, for NanoCube axes command is not valid

C-880K005: VI only supported when called through PI_Multix.vi

E-516, E-709, E-712, E-725, E-753, E-861: Without axis ID? = FALSE. Velocity unit is µm/s.

E-517: Without axis ID? = FALSE. Velocity unit is µm/s in closed-loop operation and V/s in open-loop operation.

E-710, E-761: Without axis ID? = FALSE. Velocity unit is µm/ms.

E-755: Without axis ID? = FALSE. Velocity unit is µm/s. Command not available for E-755.101.

F-206: F-206 platform velocity: Without axis ID? = TRUE; velocity of axes A and/or B: Without axis ID? = False; axes K,L,M: command not valid

M-8X0: M-8X0 platform velocity: Without axis ID? = TRUE; velocity of axes A and/or B: Without axis ID? = False

All other systems: Without axis ID? = FALSE

Output Error out, Controller error

Remarks

Sets velocity and checks for error. If Without axis ID? is TRUE, then Axes to set is ignored and first field of Velocity values array is used for velocity command. The velocity should not be set to 0. Number of digits is the number of digits after the decimal point in the velocity value(s) that will be sent. Controller error is TRUE if selected system reports error code ≠ 0.

C-867: The VEL command saves the parameters in RAM only. To save the currently valid parameters to flash ROM, where they become the power-on defaults, you must run WPA.vi. Parameter changes not saved with WPA will be lost when the C-867 is powered off.

E-516: The VEL command saves the parameters in RAM only. To save the currently valid parameters to flash ROM, where they become the power-on defaults, you must run WPA.vi. Parameter changes not saved with WPA will be lost when the E-516 is powered off.

E-517, E-709, E-712, E-725, E-753, E-755: Velocity settings made with VEL are present in RAM only and will be reset to default ("Servo Loop Slew Rate" value) when the controller is powered down or rebooted.

E-761: The VEL command saves the "Servo Loop Slew Rate" parameter in RAM only. To save the currently valid parameter to flash ROM, where it becomes the power-on default, you must run WPA.vi. Parameter changes not saved with WPA will be lost when the PC is powered off or the E-761 is rebooted.

E-861: The VEL setting only takes effect when the given axis is in closed-loop operation (servo on). For open-loop operation, use OVL instead. The maximum value which can be set with the VEL command is given by the Closed-loop velocity parameter, ID 0xA (can be changed with SPA and SEP). On power-on, the current closed-loop velocity is half the maximum.

2.4.25. VEL?.vi (General command.llb)

Valid for

Analog systems, C-702, C-843, C-843.PM, C-844, C-848, C-865, C-866, C-867, C-880, C-880K005, E-516, E-517, E-709, E-710, E-712, E-725, E-753, E-755, E-761, E-861, F-206, M-8X0, Mercury. To support analog interfacing, VI must be present for E-816 also.

Input

System number (1), Axes to query (empty string array), All axes? (F), Axis identifier? (T), Error in (no error)

Analog: If All axes? = TRUE, then Axis identifier? can be FALSE; Velocity unit is µm/s

C-702, C-848, C-880, Mercury: If All axes? = TRUE, then Axis identifier? can be FALSE

C-865, C-866, C-867: If All axes? = TRUE, then Axis identifier? can be FALSE. Velocity unit is mm/s.

C-843, C-843.PM, C-844, E-516: If All axes? = TRUE, then Axis identifier? must be TRUE

C-880K005: VI only supported when called through PI_Multix.vi

E-710: If All axes? = TRUE, then Axis identifier? must be TRUE. Velocity unit is µm/ms.

E-517: If All axes? = TRUE, then Axis identifier? can be FALSE. Velocity unit is µm/s in closed-loop operation and V/s in open-loop operation.

E-709, E-712, E-725, E-753, E-861: If All axes? = TRUE, then Axis identifier? can be FALSE. Velocity unit is µm/s.

E-755: If All axes? = TRUE, then Axis identifier? can be FALSE. Velocity unit is µm/s. Command not available for E-755.101.

E-761: If All axes? = TRUE, then Axis identifier? can be FALSE. Velocity unit is µm/ms.

F-206: Velocity of F-206: All axes? = TRUE AND Axis identifier? = FALSE; velocity of axes A,B: All axes? must be FALSE; axes K,L,M: command not valid

M-8X0: Velocity of M-8X0: All axes? = TRUE AND Axis identifier? = FALSE; velocity of axes A,B: All axes? must be FALSE

Output: Velocity, Error out

C-880: NanoCube axes will report velocity = 0

F-206: F-206 velocity: only first field of velocity array is valid

M-8X0: M-8X0 velocity: only first field of velocity array is valid

Remarks: Returns velocity setting for specified axes.

2.4.26. VMO.vi (General command.llb)

Valid for: C-702, C-848, C-880, F-206, M-8X0 (but must be present for C-866, C-867, E-761 and E-861 also)

Input: System number (1), Axes to command (empty string array), Position values (empty num. array, 0), No. of digits (4), Error in (no error)

Output: Move possible (T/F), Error out

Remarks: Virtual movement. Indicates whether a move to the specified position is possible or not. Stage will **not** be moved. No of digits is the number of digits after the decimal point in the position value(s) that will be sent.

2.5. Joystick-specific VIs ("Joystick.llb")

2.5.1. Calculate joystick scaling.vi (Joystick.llb)

Valid for: C-702, C-843, C-843.PM, C-844, C-848, C-865, C-866, C-867, C-880, E-517, E-709, E-710, E-712, E-725, E-753, E-755, E-761, E-861

Input: XPos min (0), YPos min (0), XPos max (65535), YPos max (65535), Resolution factor X (150), Resolution factor Y (150)

Output: XPos min scaled, YPos min scaled, XPos max scaled, YPos max scaled, XPos Center scaled, YPos Center scaled, Scaling factor X, Scaling factor Y

Remarks: Sub-VI for operation with a joystick connected to the game port of the host computer. Calculates joystick position scaling. If Resolution factor * = *Pos max, maximum resolution is achieved. *Pos min and *Pos max depend on the Windows joystick calibration.

2.5.2. JAS?.vi (Joystick.llb)

Valid for: C-867, E-861, Mercury

Input: System number (1), Joystick ID to query (empty string array), Without JoystickIDs? (F), Joystick axis (empty num. array, 0), Error in (no error)

Output: Parameter value, Parameter string, Error out

Remarks: Returns joystick axes status values (current amplitude) of joysticks connected to the controller.

C-867, E-861, Mercury: The current amplitude gives a factor which is applied to the

velocity set with VEL (closed-loop operation) or OVL (E-861 open-loop operation), the range is -1.0 to 1.0. Only one Joystick ID to query for one Joystick axis per command allowed. Use Without JoystickIDs? = TRUE to query all Joystick IDs.

2.5.3. JAX.vi (Joystick.llb)

Valid for	C-843, C-867, E-861
Input	System number (1), JoystickID (0), Joystick axis (0), Axes to set (empty string array), Error in (no error)
Output	Error out
Remarks	Enables control of specified axes with an axis of a joystick connected to the controller. If Axes to set is an empty array, the current settings are cleared and no axes are controlled. Settings will not influence "Joystick_Operation_Sample_Program.vi" which works with a joystick connected to the host computer.

2.5.4. JAX?.vi (Joystick.llb)

Valid for	C-843, C-867, E-861, Mercury
Input	System number (1), Joystick ID to query (empty string array), Without JoystickIDs? (F), Joystick axis (empty num. array, 0), Error in (no error)
Output	Controlled axes, Error out
Remarks	Returns axes which are controlled by joystick axes of joysticks connected to the controller.

2.5.5. JBS?.vi (Joystick.llb)

Valid for	C-867, E-861, Mercury
Input	System number (1), Joystick ID to query (empty string array), Without JoystickIDs? (F), Joystick button (empty num. array, 0), Error in (no error)
Output	State, Error out
Remarks	Returns state (TRUE/FALSE) of queried buttons of joysticks connected to the controller.

2.5.6. JDT.vi (Joystick.llb)

Valid for	C-867, E-861, Mercury
Input	System number (1), JoystickID (empty num. array, 0), Joystick axis (empty num. array, 0), Lookup table type (empty num. array), Error in (no error)
Output	Controller error, Error out
Remarks	Sets default lookup table for joysticks connected to the controller, waits 100 ms and queries ERR?. Lookup table type can be: 1: linear 2: parabolic 3: cubic 4: exponential Controller error is TRUE if selected system reports error code ≠ 0. Settings will not influence "Joystick_Operation_Sample_Program.vi" which works with a joystick connected to the host computer. C-867, E-861: The following default lookup tables are provided:

1 = linear
2 = parabolic

Mercury: The following default lookup tables are provided:
1 = linear
3 = cubic

2.5.7. JLT.vi (Joystick.llb)

Valid for	C-867, E-861, Mercury
Input	System number (1), JoystickID (0), Joystick axis (0), Start point (1), Factor (empty num. array, 1), Error in (no error)
Output	Controller error, Error out
Remarks	Sets joystick lookup table for a joystick connected to the controller. The amplitudes of the joystick axes are mapped to velocities of the controller axes. StartPoint is the start point in the lookup table, starting with 1. Valid values for Factor are -1.0 to 1.0 and define the factor which is applied to the currently set velocity (VEL in closed-loop operation, OVL in open-loop operation). No. of digits is the number of digits after the decimal point in the Factor value(s) that will be sent. Controller error is TRUE if selected system reports error code ≠ 0. Settings will not influence "Joystick_Operation_Sample_Program.vi" which works with a joystick connected to the host computer. C-867, E-861, Mercury: 256 values must be written. The first point corresponds to the maximum joystick axis displacement in negative direction, the 256th point to the maximum displacement in positive direction.

2.5.8. JLT?.vi (Joystick.llb)

Valid for	C-867, E-861, Mercury
Input	System number (1), JoystickIDs (Empty num. array, 0), Joystick axes (Empty num. array, 0), X0 (1), N (100), Without parameter? (F), Error in (no error)
Output	Names, Data, Error out
Remarks	Returns joystick lookup table for joysticks connected to the controller. For large N values, communication timeout must be set long enough, otherwise a comm.error may occur. C-867, E-861: JoystickIDs and Joystick axes must be omitted in the JLT? command, while X0 and N are always required.

2.5.9. JON.vi (Joystick.llb)

Valid for	C-843, C-867, E-861, Mercury
Input	System number (1), JoystickID to command (empty num. array, 0), Joystick mode (empty num. array, 0), Error in (no error) C-843, C-867, E-861, Mercury: Joystick mode can be 0 (OFF) or 1 (ON).
Output	Controller error, Error out
Remarks	Enables or disables joysticks connected to the controller. Controller error is TRUE if selected system reports error code ≠ 0. Settings will not influence "Joystick_Operation_Sample_Program.vi" which works with a joystick connected to the host computer. E-861: Motion commands like MOV or OSM are not allowed when a joystick is

active on the axis.

C-843, C-867, Mercury: Motion commands like MOV are not allowed when a joystick is active on the axis.

2.5.10. JON?.vi (Joystick.llb)

Valid for	C-843, C-867, E-861, Mercury
Input	System number (1), JoystickID to query (empty string array), All joysticks? (F), Error in (no error)
Output	Joystick status, Error out
Remarks	Returns activation status of queried joysticks connected to the controller.

2.5.11. Read joystick.vi (Joystick.llb)

Valid for	C-702, C-843, C-843.PM, C-844, C-848, C-865, C-866, C-867, C-880, E-517, E-709, E-710, E-712, E-725, E-753, E-755, E-761, E-861
Input	Joystick ID (0), Error in (no error)
Output	XPos, YPos, Button 1 pressed (T/F), Button 2 pressed (T/F), Error out
Remarks	Sub-VI for operation with a joystick connected to the game port of the host computer. Reads joystick position and button status for a standard 2-button 2-axis joystick. Install joystick driver and calibrate joystick in the Windows control panel before running this VI.

2.5.12. Scale joystick data.vi (Joystick.llb)

Valid for	C-702, C-843, C-843.PM, C-844, C-848, C-865, C-866, C-867, C-880, E-517, E-709, E-710, E-712, E-725, E-753, E-755, E-761, E-861
Input	XPos (0), YPos (0), XPos Center (0), YPos Center (0), Dead band X (0), Dead band Y (0), Factor X (0), Factor Y (0)
Output	XPos scaled, YPos scaled
Remarks	Sub-VI for operation with a joystick connected to the game port of the host computer. Scales joystick position. Use output value from "Calculate joystick scaling.vi" for Factor *. Dead band * is the maximum scaled position value that does not result in any motion.

2.6. Limit- and reference-specific commands (“Limits.llb”)

2.6.1. FED.vi (Limits.llb)

Valid for	C-843, C-867, E-861
Input	System number (1), Axis to set (empty string array), EdgeID (empty num. array, 0), Parameter (empty num. array, 0), Error in (no error) C-843: EdgeID can be 1 to 3. Parameter value can be -1, 0 or 1, depending on EdgeID. C-867, E-861: EdgeID can be 1 to 3. Parameter value must be 0.
Output	Error out

Remarks Moves given axis to a given signal edge. If multiple axes are to command, they are moved synchronously. This command does not change the reference state of the axis and does not set a certain position value. Valid Parameter values depend on EdgeID:

- EdgeID 1: negative limit switch, Parameter value is 0 when the default setting should be used (e.g. from pistages.dat), 1 when active high, -1 when active low
- EdgeID 2: positive limit switch, Parameter value is 0 when the default setting should be used (e.g. from pistages.dat), 1 when active high, -1 when active low
- EdgeID 3: reference switch, Parameter value is 0 when the default setting should be used (e.g. from pistages.dat), 1 when active high, -1 when active low
- EdgeID 4: autofind AxisIn (is one input line of the motion chip carrying an external sensor signal which changes its state at a certain position), Parameter value gives the signal state to the left of the edge (high or low)
- EdgeID 5: find arbitrary edge (i.e. change of the state of the AxisIn signal), Parameter value gives the direction of motion: 1=positive, -1= negative

C-843, C-867, E-861: The firmware detects the presence or absence of reference switch and limit switches using controller parameters (ID 0x14 for reference switch; ID 0x32 for limit switches). According to the values of those parameters, the controller enables or disables FED motions to the appropriate signal edges. Adapt the parameter values to your hardware using SPA (or SEP, if supported).

2.6.2. FNL.vi (Limits.llb)

Valid for C-843, C-843.PM, C-867, E-755, E-861

Input System number (1), Affected axes (empty string array), All axes? (F), Axis identifier? (T), Error in (no error)

C-843, C-843.PM, C-867, E-861: If All axes? = TRUE, then Axis identifier? must be TRUE

E-755: If All axes? = TRUE, then Axis identifier? can be FALSE. Command not available for E-755.101

Output Error out

Remarks This VI starts a fast move to negative limit of the specified axes. Use #7 polling to determine end of this referencing procedure.

C-867: Servo must be enabled with SVO for the commanded axis prior to using this command (closed-loop operation).
The reference mode must be set to "1" (factory default) with the RON command if referencing is to be done by performing a reference move.
The negative limit switch of the mechanics is used to determine the negative physical limit of the travel range. The difference of VALUE_AT_REF_POS (parameter ID 0x16) and DISTANCE_REF_TO_N_LIM (parameter ID 0x17) is set as the current position when the axis is at the negative limit switch (value can be negative). If the soft limits (MAX_TRAVEL_RANGE_POS, parameter ID 0x15, and MAX_TRAVEL_RANGE_NEG, parameter ID 0x30) are used to reduce the travel range, the limit switches can not be used for reference moves. FNL and FPL commands will provoke an error message, and only the reference switch can be used for a reference move (FRF command).

E-861: The reference mode must be set to "1" (factory default) with the RON command if referencing is to be done by performing a reference move.

The negative limit switch of the mechanics is used to determine the negative physical limit of the travel range. The difference of VALUE_AT_REF_POS (parameter ID 0x16) and DISTANCE_REF_TO_N_LIM (parameter ID 0x17) is set as the current position when the axis is at the negative limit switch (value can be negative). If the soft limits (MAX_TRAVEL_RANGE_POS, parameter ID 0x15, and MAX_TRAVEL_RANGE_NEG, parameter ID 0x30) are used to reduce the travel range, the limit switches can not be used for reference moves. FNL and FPL commands will provoke an error message, and only the reference switch can be used for a reference move (FRF command).

2.6.3. FPL.vi (Limits.llb)

Valid for C-843, C-843.PM, C-867, E-755, E-861

Input System number (1), Affected axes (empty string array), All axes? (F), Axis identifier? (T), Error in (no error)

C-843, C-843.PM, C-867, E-861: If All axes? = TRUE, then Axis identifier? must be TRUE

E-755: If All axes? = TRUE, then Axis identifier? can be FALSE. Command not available for E-755.101

Output Error out

Remarks This VI starts a fast move to positive limit of the specified axes. Use #7 polling to determine end of this referencing procedure.

C-867: Servo must be enabled with SVO for the commanded axis prior to using this command (closed-loop operation).
The reference mode must be set to "1" (factory default) with the RON command if referencing is to be done by performing a reference move.
The positive limit switch of the mechanics is used to determine the positive physical limit of the travel range. The sum of VALUE_AT_REF_POS (parameter ID 0x16) and DISTANCE_REF_TO_P_LIM (parameter ID 0x2F) is set as the current position when the axis is at the positive limit switch. If the soft limits (MAX_TRAVEL_RANGE_POS, parameter ID 0x15, and MAX_TRAVEL_RANGE_NEG, parameter ID 0x30) are used to reduce the travel range, the limit switches can not be used for reference moves. FNL and FPL commands will provoke an error message, and only the reference switch can be used for a reference move (FRF command).

E-861: The reference mode must be set to "1" (factory default) with the RON command if referencing is to be done by performing a reference move.
The positive limit switch of the mechanics is used to determine the positive physical limit of the travel range. The sum of VALUE_AT_REF_POS (parameter ID 0x16) and DISTANCE_REF_TO_P_LIM (parameter ID 0x2F) is set as the current position when the axis is at the positive limit switch. If the soft limits (MAX_TRAVEL_RANGE_POS, parameter ID 0x15, and MAX_TRAVEL_RANGE_NEG, parameter ID 0x30) are used to reduce the travel range, the limit switches can not be used for reference moves. FNL and FPL commands will provoke an error message, and only the reference switch can be used for a reference move (FRF command).

2.6.4. FRF.vi (Limits.llb)

Valid for C-843, C-843.PM, C-867, C-880 (only K006/K007 version), C-880K005, E-712, E-861

Input System number (1), Affected axes (empty string array), All axes? (F), Axis identifier? (T), Error in (no error)

C-843, C-843.PM, C-867, C-880, E-861: If All axes? = TRUE, then Axis identifier? must be TRUE

C-880K005: VI only supported when called through PI_Multix.vi

E-712: If All axes? = TRUE, then Axis identifier? must be TRUE. Check HLP? answer to find out if FRF is supported.

Output Error out

Remarks This VI starts a fast referencing of the specified axes. Use #7 polling to determine end of this referencing procedure.

C-867: Servo must be enabled with SVO for the commanded axis prior to using this command (closed-loop operation).
The reference mode must be set to "1" (factory default) with the RON command if referencing is to be done by performing a reference move.
The value of the VALUE_AT_REF_POS parameter (ID 0x16) is set as the current position when the axis is at the reference switch. Use FNL or FPL (if supported) instead of FRF to perform a reference move for an axis which has no reference sensor but limit switches.

E-861: The reference mode must be set to "1" (factory default) with the RON command if referencing is to be done by performing a reference move.
The value of the VALUE_AT_REF_POS parameter (ID 0x16) is set as the current position when the axis is at the reference switch. Use FNL or FPL (if supported) instead of FRF to perform a reference move for an axis which has no reference sensor but limit switches.

2.6.5. FRF?.vi (Limits.llb)

Valid for C-843, C-843.PM, C-866, C-867, C-880 (only C-880K006/C-880K007 version), C-880K005, E-712, E-755, E-861

Input System number (1), Axes to query (empty string array), All axes? (F), Axis identifier? (T), Error in (no error)

C-843, C-843.PM, C-866, C-867, C-880, E-861: If All axes? = TRUE, then Axis identifier? can be FALSE

C-880K005: VI only supported when called through PI_Multix.vi

E-712: If All axes? = TRUE, then Axis identifier? can be FALSE. Check HLP? answer to find out if FRF? is supported.

E-755: If All axes? = TRUE, then Axis identifier? can be FALSE. Command not available for E-755.101

Output Referenced? (T/F), Error out

Remarks Indicates whether queried axes have been referenced (using REF, FNL, FPL, FRF, MPL, MNL, or - if reference mode is OFF - using POS) successfully or not.

2.6.6. GOH.vi (Limits.llb)

Valid for C-702, C-843, C-843.PM, C-844, C-848, C-865, C-866, C-867, C-880, C-880K005, E-517, E-710, E-755, E-761, E-861, Mercury

Input System number (1), GOH axes (empty string array), All axes? (F), Axis identifier? (T), Error in (no error)

C-702, C-843, C-843.PM, C-844, C-848, C-865, C-866, C-867, C-880, E-710, E-861: If All axes? = TRUE, then Axis identifier? must be TRUE

C-880K005: VI only supported when called through PI_Multix.vi

E-755: If All axes? = TRUE, then Axis identifier? can be FALSE. Command not

available for E-755.101.

E-517, E-761, Mercury: If All axes? = TRUE, then Axis identifier? can be FALSE

Output: Error out

Remarks: Moves specified axes to their home positions.

2.6.7. LIM?.vi (Limits.llb)

Valid for: C-702, C-843, C-843.PM, C-848, C-865, C-866, C-867, C-880, E-755, E-861, Mercury

Input: System number (1), Axes to query (empty string array), All axes? (F), Axis identifier? (T), Error in (no error)

All systems: If All axes? = TRUE, then Axis identifier? can be FALSE

Output: Axis with limit switch? (T/F), Error out

Remarks: Indicates whether queried axes have limit switches or not.

2.6.8. RON.vi (Limits.llb)

Valid for: C-702, C-843, C-843.PM, C-848, C-865, C-866, C-867, C-880, C-880K005, E-712, E-861, Mercury

Input: System number (1), Without axis ID?(F), Axes to command (empty string array), Reference mode (empty bool. array, F), Error in (no error)

C-880K005: VI only supported when called through PI_Multix.vi

All other systems: Without axis ID = FALSE

Output: Error out

Remarks: Sets reference mode for given axes. If Without axis ID is TRUE, then Axes to command is ignored and first field of the Reference mode array is used for the reference mode.

If the reference mode of an axis is ON, the axis must be driven to the reference switch (using “REF.vi” or "FRF.vi") or, if no reference switch is available, to a limit switch (positive limit switch: using “MPL.vi” or "FPL.vi"; negative limit switch: using “MNL.vi” or "FNL.vi") before any other motion can be commanded in closed-loop operation.

If reference mode is OFF, no referencing is required for the axis. In closed-loop operation, only relative moves can be commanded (using “MVR.vi”), unless the actual position is set with POS.vi. Afterwards, relative and absolute moves can be commanded.

For stages with neither reference nor limit switch, reference mode is automatically OFF.

WARNINGS:

If reference mode is switched off, and relative moves are commanded, stages can be driven into the mechanical hard stop if moving to a position which is outside the travel range!

If reference mode is switched off, and the actual position is incorrectly set with “POS.vi”, stages can be driven into the mechanical hard stop when moving to a position which is thought to be within the travel range of the stage, but actually is not.

E-712: Check HLP? answer to find out if RON is supported.

2.6.9. RON?.vi (Limits.llb)

Valid for	C-702, C-843, C-843.PM, C-848, C-865, C-866, C-867, C-880, C-880K005, E-712, E-861, Mercury
Input	System number (1), Axes to query (empty string array), All axes? (F), Axis identifier? (T), Error in (no error) C-880K005: VI only supported when called through PI_Multix.vi All other systems: If All axes? = TRUE, then Axis identifier? can be FALSE
Output	Reference on? (T/F), Error out
Remarks	Indicates whether queried axes have reference mode ON or OFF. See "RON.vi" above for description of reference mode. E-712: Check HLP? answer to find out if RON is supported.

2.6.10. TMN?.vi (Limits.llb)

Valid for	Analog systems, C-702, C-843, C-843.PM, C-844, C-848, C-865, C-866, C-867, C-880, E-517, E-709, E-710, E-712, E-725, E-753, E-755, E-761, E-861, Mercury. To support analog interfacing, VI must be present for E-816 also.
Input	System number (1), Axes to query (empty string array), All axes? (F), Axis identifier? (T), Error in (no error) Analog, C-702, C-848, C-865, C-866, C-867, C-880, E-517, E-709, E-712, E-725, E-753, E-761, E-861, Mercury: If All axes? = TRUE, then Axis identifier? can be FALSE C-843, C-843.PM, C-844, E-710: If All axes? = TRUE, then Axis identifier? must be TRUE E-755: If All axes? = TRUE, then Axis identifier? can be FALSE. Command not available for E-755.101.
Output	Minimum travel limit, Error out
Remarks	Returns minimum (low-end) travel limit (if present, position of negative limit switch, or value of negative soft limit, if set, whichever is higher). C-867: The minimum commandable position is defined by the MAX_TRAVEL_RANGE_NEG parameter ID 0x30 (SPA). E-517, E-761: Get the minimum accessible position value, i.e. the value of the "Range min limit" parameter (ID 0x07000000). Note: The minimum position which can be commanded depends either on the "Range min limit" parameter or - if it is greater than the "Range min limit" parameter value - on the value of the negative soft limit set with NLM. E-861: The minimum commandable position is defined by the MAX_TRAVEL_RANGE_NEG parameter, ID 0x30 (SPA).

2.6.11. TMX?.vi (Limits.llb)

Valid for	Analog systems, C-702, C-843, C-843.PM, C-844, C-848, C-865, C-866, C-867, C-880, E-517, E-709, E-710, E-712, E-725, E-753, E-755, E-761, E-861, Mercury. To support analog interfacing, VI must be present for E-816 also.
Input	System number (1), Axes to query (empty string array), All axes? (F), Axis identifier? (T), Error in (no error) Analog, C-702, C-848, C-865, C-866, C-867, C-880, E-517, E-709, E-712, E-725, E-753, E-761, E-861, Mercury: If All axes? = TRUE, then Axis identifier?

can be FALSE

C-843, C-843.PM, C-844, E-710: If All axes? = TRUE, then Axis identifier? must be TRUE

E-755: If All axes? = TRUE, then Axis identifier? can be FALSE. Command not available for E-755.101.

Output Maximum travel limit, Error out

Remarks Returns maximum (high-end) travel limit (if present, position of positive limit switch or value of positive soft limit, if set, whichever is lower).

C-867: The maximum commandable position is defined by the MAX_TRAVEL_RANGE_POS parameter ID 0x15 (SPA).

E-517, E-761: Get the maximum accessible position value, i.e. the value of the "Range max limit" parameter (ID 0x07000001). Note: The maximum position which can be commanded depends either on the "Range max limit" parameter or—if it is smaller than the "Range max limit" parameter value—on the value of the positive soft limit set with PLM.

E-861: The maximum commandable position is defined by the MAX_TRAVEL_RANGE_POS parameter, ID 0x15 (SPA).

2.6.12. TRS?.vi (Limits.llb)

Valid for C-867, E-712, E-755, E-861

Input System number (1), Axes to query (empty string array), All axes? (F), Axis identifier? (T), Error in (no error)

C-867, E-755, E-861: If All axes? = TRUE, then Axis identifier? can be FALSE

E-712: If All axes? = TRUE, then Axis identifier? can be FALSE. Check HLP? answer to find out if TRS? Is supported

Output Reference sensor? (T/F), Error out

Remarks Indicates whether or not given axes have a reference sensor with direction sensing.

2.7. Macro Functions (“Macros.llb”)

2.7.1. #8.vi (Macros.llb)

Valid for C-702, C-848, C-867, C-880, E-516, E-517, E-816, E-861

Input System number (1), Error in (no error)

Output Macro running? (T/F), String read, Error out

Remarks Sends ASCII #8 without Linefeed and returns Macro running? indicating whether a macro is running or not.

E-816: This command cannot be issued to a slave. Check controller manual to find out if #8 is supported.

2.7.2. Define macro contents.vi (Macros.llb)

Valid for C-702, C-844, C-848, C-867, C-880, E-517, E-816, E-861, F-206, Mercury

Input System number (1), Macro contents (empty string), Error in (no error)

Output Error out

Remarks Defines macro contents. Each command to be stored in the macro must be written on one line, terminated with the enter key. MAC BEG.vi must be called before running this VI and MAC END.vi must be called afterwards. Macros are stored as entered and may be affected by any change of scale factor before execution.

E-816: Macro will not be saved to FLASH until WPA .vi was run. Changes not saved with WPA are only present in RAM and will be lost when the controller is powered off. This command cannot be issued to a slave. Check controller manual to find out if MAC is supported.

2.7.3. MAC BEG.vi (Macros.llb)

Valid for C-702, C-844, C-848, C-867, C-880, E-516, E-517, E-816, E-861, F-206, Mercury

Input System number (1), Macro name (empty string), Error in (no error)

C-702, C-848, C-867, C-880, E-516, E-517, E-816, F-206: Macro name must be between 1 and 8 characters

Mercury: For valid Macro names see GCS DLL Manual.

Output Error out

Remarks Begin macro recording. Because controller will not answer VI queries during macro recording phase, command VIs cannot be run after this VI to define the macro. Run “Define macro contents.vi” and finish with “MAC END.vi” to define a macro.

E-516: Macro will not be saved to FLASH until WPA .vi was run. Changes not saved with WPA are only present in RAM and will be lost when the controller is powered off.

E-816: Macro will not be saved to FLASH until WPA .vi was run. Changes not saved with WPA are only present in RAM and will be lost when the controller is powered off. This command cannot be issued to a slave. Check controller manual to find out if MAC BEG is supported.

2.7.4. MAC DEF.vi (Macros.llb)

Valid for C-867, E-516, E-517, E-816, E-861

Input System number (1), Macro name (empty string), Error in (no error)

Output Error out

Remarks Define autostart macro. To disable the autostart macro, Macro name must be an empty string.

E-516: Autostart macro definition will not be saved to FLASH until WPA.vi was run. Changes not saved with WPA are only present in RAM and will be lost when the controller is powered off.

E-816: Autostart macro definition will not be saved to FLASH until WPA.vi was run. Changes not saved with WPA are only present in RAM and will be lost when the controller is powered off. This command cannot be issued to a slave. Check controller manual to find out if MAC DEF is supported.

2.7.5. MAC DEF?.vi (Macros.llb)

Valid for C-867, E-516, E-517, E-816, E-861

Input System number (1), Error in (no error)

Output Autostart macro, Error out

Remarks Get name of autostart macro.

E-816: This command cannot be issued to a slave. Check controller manual to find out if MAC DEF? is supported.

2.7.6. MAC DEL.vi (Macros.llb)

Valid for C-702, C-844, C-848, C-867, C-880, E-516, E-517, E-816, E-861, F-206, Mercury

Input System number (1), Macro name (empty string), With dialog? (T), Error in (no error)

Output Hidden error (T/F), Error out

Remarks Delete macro. If "With dialog" is TRUE, a dialog box pops up to confirm the deletion. Hidden error is TRUE if selected system reports error code ≠ 0.

E-516: Changes not saved with WPA.vi are only present in RAM and will be lost when the controller is powered off.

E-816: Changes not saved with WPA.vi are only present in RAM and will be lost when the controller is powered off. This command cannot be issued to a slave. Check controller manual to find out if MAC DEL is supported.

2.7.7. MAC END.vi (Macros.llb)

Valid for C-702, C-844, C-848, C-867, C-880, E-516, E-517, E-816, E-861, F-206, Mercury

Input System number (1), Error in (no error)

Output Error out

Remarks Stops current macro recording.

E-516: Macro will not be saved to FLASH until WPA .vi was run. Changes not saved with WPA are only present in RAM and will be lost when the controller is powered off.

E-816: Macro will not be saved to FLASH until WPA .vi was run. Changes not saved with WPA are only present in RAM and will be lost when the controller is powered off. This command cannot be issued to a slave. Check controller manual to find out if MAC END is supported.

2.7.8. MAC NSTART.vi (Macros.llb)

Valid for C-702, C-867, E-516, E-517, E-816, E-861, Mercury

Input System number (1), Macro name (empty string), N (1), Error in (no error)

Output Error out

Remarks Start macro N times.

E-816: This command cannot be issued to a slave. Check controller manual to find out if MAC NSTART is supported.

2.7.9. MAC START.vi (Macros.llb)

Valid for C-702, C-844, C-848, C-867, C-880, E-516, E-517, E-816, E-861, F-206, Mercury

Input System number (1), Macro name (empty string), Error in (no error)

Output Error out

Remarks Start macro.

C-702, C-848, C-867, C-880, E-516, E-517, E-861: Use #8.vi to determine when macro execution has finished.

E-816: Use #8.vi to determine when macro execution has finished. This command cannot be issued to a slave. Check controller manual to find out if MAC START is supported.

F-206, Mercury: #8.vi is not supported.

2.7.10. MAC?.vi (Macros.llb)

Valid for C-702, C-844, C-848, C-867, C-880, E-516, E-517, E-816, E-861, F-206, Mercury

Input System number (1), Get contents? (F), Macro name (empty string), Error in (no error)

Output Macro names or contents, Error out

Remarks If Get contents is FALSE, returns names of all available macros, if TRUE, returns contents of one specified macro.

E-816: This command cannot be issued to a slave. Check controller manual to find out if MAC? is supported.

2.7.11. RMC?.vi (Macros.llb)

Valid for C-867, E-861

Input System number (1), Error in (no error)

Output Running macros, Error out

Remarks List macros currently running.

2.8. Old commands and commands with alternate implementations ("Old commands.llb")

2.8.1. #5_old.vi (Old commands.llb)

Valid for F-206, M-8X0 (but must be present for all other systems also)

Input System number (1), Error in (no error)

Output Overall system moving? (T/F), Sep. Axis 1 moving? (T/F), Sep. Axis 2 moving? (T/F), Error out

Remarks Polls the motion status of the F-206/M-8X0 and/or up to 2 additional connected axes by sending the single ASCII character 5. Required by "General wait for movement to stop.vi".

2.8.2. Wait for hexapod system axes to stop.vi (Old commands.llb)

Valid for F-206, M-8X0 (but must be present for all other systems also)

Input System number (1), All axes? (T), Axes to wait for (empty string array), Stop refnum (F), Local stop (F), Error in (no error)

To wait for the hexapod to stop, only one hexapod axis (X, Y, Z, U, V or W) needs to be commanded, because the VI cannot distinguish between the different hexapod axes.

F-206: Axes to wait for can be any of X, Y, Z, U, V, W, A, B, K, L, M

M-8X0: Axes to wait for can be any of X, Y, Z, U, V, W, A, B

Output: Error out

Remarks: This vi waits for the specified axes of a PI hexapod system (hexapod axes X, Y, Z, U, V, W and separate axes A, B) to stop using #5 polling. If a NanoCube axis (K, L or M) is commanded, the VI will return immediately. If one of the hexapod axes (X, Y, Z, U, V or W) is commanded, it will wait for all six hexapod axes to stop. It returns immediately if a communications error occured, or if Local stop or Stop refnum is TRUE. When using as a sub-VI, use Refnum stop to stop VI from caller. Required by “General wait for movement to stop.vi”.

2.9. Commands for Optical or Analog Signals (“Optical or Analog Input.llb”)

2.9.1. MOV and TAV?.vi (Optical or Analog Input.llb)

Valid for: C-865, C-866, C-867, C-880, E-761, E-861, F-206, M-8X0, Mercury

Input: System number (1), Board (1), No. of digits (4), Wait before TAV?, ms (1), Polling cycle time, ms (1), Axes to move (empty string array), Position values (empty num. array), Error in (no error)

E-761: Board = 4.

Output: Analog value, Error out

Remarks: Moves stage to absolute position (MOV.vi), waits for the specified axes to stop (General wait for movement to stop.vi) and queries TAV? (TAV?.vi). A wait time before the TAV? query can be defined. “Define connected systems.vi” must be run before running this VI. Requires “Wait for axes to stop.vi”, “#5.vi”, “STA?.vi”, “#5_old.vi”, “ONT?.vi” and “Wait for hexapod system axes to stop.vi” to be present.

E-761: The output is the current voltage at the analog input line, with gain and offset.

2.9.2. MWG and TAV?.vi (Optical or Analog Input.llb)

Valid for: F-206 (but must be present for C-866, C-867, C-880, E-761, E-861 and M-8X0 also)

Input: System number (1), Board (1), No. of digits (4), Wait before TAV?, ms (1), Polling cycle time, ms (1), Axes to move (empty string array), Position values (empty num. array), Error in (no error)

Output: Analog value, Error out

Remarks: Moves stage to absolute position (MWG.vi), waits for the specified axes to stop (General wait for movement to stop.vi) and queries TAV? (TAV?.vi). A wait time before the TAV? query can be defined. “Define connected systems.vi” must be run before running this VI. Requires “Wait for axes to stop.vi”, “#5.vi”, “STA?.vi”, “#5_old.vi”, “ONT?.vi” and “Wait for hexapod

system axes to stop.vi" to be present. Required by "1D Scan.vi" and "2D Scan.vi".

2.9.3. TAC?.vi (Optical or Analog Input.llb)

Valid for	C-702, C-866, C-867, C-880, E-709, E-861, Mercury
Input	System number (1), Error in (no error)
Output	Analog channels, Error out
Remarks	Returns the number of installed analog channels.

2.9.4. TAV?.vi (Optical or Analog Input.llb)

Valid for	C-865, C-866, C-867, C-880, E-761, E-861, F-206, M-8X0, Mercury
Input	System number (1), Board (1), Query (Value), Error in (no error)
	C-865, C-866: Query = Value, Board = 1.
	C-867: Query = Value, Board = 1 to 4.
	C-880: Query = Value.
	E-761: Query = Value, Board = 4.
	E-861: Query = Value, Board = 1 to 4.
	F-206, M-8X0: Query can be Value, Range or Power unit.
	Mercury: Query = Value. Board = analog input channel ID, can be 1-4, 5-7, 8-11 etc., see GCS DLL Manual for details
Output	Analog value, Range, Power unit, Error out
	C-865, C-866, C-867, C-880, E-761, E-861, Mercury: Range and Power unit are not valid
Remarks	Returns the current analog value in volts, the range of the optical head or the power unit of the Analog value, depending on Query. Query time for Analog value will depend on "NAV" settings.
	E-761: The output is the current voltage at the analog input line, with gain and offset.

2.9.5. TSC?.vi (Optical or Analog Input.llb)

Valid for	E-517, E-709, E-710, E-712, E-725, E-753, E-755, E-761 (but must be present for Analog systems, C-867, E-516, E-816 and E-861 also)
Input	System number (1), Error in (no error)
Output	Number of sensor channels, Error out
Remarks	Returns the number of available sensor channels.
	E-517: Using the Sensor Enable parameter, ID 0x02000000, you can change the E-517 configuration in case of hardware changes, e.g. if you install additional sensor and/or amplifier channels in the system. If this parameter is changed, the Number Of Sensor Channels parameter is adapted automatically. E.g. if parameter 0x02000000 is set to "disabled" for a sensor channel, this sensor channel is no longer included in the TSC? response. See "Configure Axes and Channels" in the E-517 User manual for details.
	E-712, E-725: The response comprises all ADC channels of the device: "genuine" sensors (capacitive sensors integrated in the mechanics) and "general purpose" analog input channels.
	E-753: The response comprises all ADC channels of the device: the "genuine"

sensor (capacitive sensor integrated in the mechanics) and the "general purpose" analog input.

E-709: The response comprises all ADC channels of the device: the "genuine" sensor (sensor integrated in the mechanics) and the "general purpose" analog input.

2.10. PZT specific VIs ("PZT voltage.llb")

2.10.1. OAD.vi (PZT voltage.llb)

Valid for	E-755, E-861
Input	System number (1), No. of digits (4), Channel to set (empty string array), Driving voltage (empty num. array), Error in (no error)
Output	Error out, Hidden error
Remarks	Sets driving voltage amplitude of open-loop analog driving for given PiezoWalk channels. No. of digits is the number of digits after the decimal point in the voltage value(s) that will be sent. Hidden error is TRUE if selected system reports error code ≠ 0.

E-861: Servo must be disabled for the commanded axis prior to using this command (open-loop operation). In open-loop operation, an RNP command must be sent each time the motion mode is to be changed from stepping motion (OSM command) to analog motion (OAD command) and vice versa. RNP brings the drive to a full-holding-force, zero-drive-voltage Relaxed state.
The first OAD sent for a NEXACT® linear drive which is in the Relaxed state prepares the drive for analog motion (brings it to the Analog state) before the actual motion is done. Once the drive is in the Analog state, each subsequent OAD motion will be executed immediately.
The first OAD after a change of motion mode and the RNP procedure can take up to four times the slewrate value (parameter ID 0x7000002; can be changed with SPA and SEP).
After open-loop analog motion was done with OAD, an RNP command must be sent before the servo can be switched on with SVO for closed-loop operation.
Motion commands like OAD are not allowed when a joystick is active on the axis.

2.10.2. OAD?.vi (PZT voltage.llb)

Valid for	E-755, E-861
Input	System number (1), Channel to query (empty string array), All channels? (F), Channel identifier? (T), Error in (no error)
	E-755, E-861: If All channels? = TRUE, then Channel identifier? must be FALSE
Output	Driving voltage, Error out
Remarks	Returns last commanded open-loop Analog Driving voltage of given PiezoWalk channels.

2.10.3. OSM.vi (PZT voltage.llb)

Valid for	E-755, E-861

Input System number (1), Affected channels (empty string array), No. of steps (empty num. array, 0), Error in (no error)

E-755: No. of digits = 0

Output Error out

Remarks Open-loop step moving of given PiezoWalk channels. Use SSA.vi to set step size.

E-861: The velocity for open-loop step motion depends on the step size and on the step frequency. The step size is given by the amplitude of the transport voltage which can be set with SSA and queried with SSA?. The step frequency can be set with OVL and queried with OVL?.
In open-loop operation, a RNP command must be sent each time the motion mode is to be changed from stepping motion (OSM command) to analog motion (OAD command) and vice versa. RNP brings the drive to a full-holding-force, zero-drive-voltage Relaxed state.
The first OSM sent for a NEXACT® linear drive which is in the Relaxed state prepares the drive for stepping motion (brings it to the In Motion state) before the actual motion is done. Once the drive is in the In Motion state, each subsequent OSM motion will be executed immediately.
The first OSM after a change of motion mode and the RNP procedure can take up to four times the slewrate value (parameter ID 0x7000002; can be changed with SPA and SEP).
To command parts of a step cycle, make sure to increase No. of digits.
Motion commands like OSM are not allowed when a joystick is active on the axis.

2.10.4. RNP.vi (PZT voltage.llb)

Valid for E-755, E-861

Input System number (1), NexlineChannel (empty string array), Adj. voltage (empty num. array, 0), No. of digits (4), Error in (no error)

Output Error out

Remarks Relax piezos of given PiezoWalk® channels without motion and wait until procedure has stopped. The aim of this procedure is to reduce all applied voltages when the target is reached and thus to increase the lifetime of the piezos. To compensate a preload which would lead to a small motion of the slider when the piezo voltage is zero an adjustment voltage can be applied. VI will also stop if Stop refnum or Local stop is TRUE.

E-755: The RNP procedure is automatically performed when the servo is switched off and when RTO is used.

E-861: Adj. voltage must be zero to set the voltages to 0.
Servo must be disabled for the commanded axis prior to using this command (open-loop operation).
In open-loop operation, an RNP command must be sent each time the motion mode is to be changed from stepping motion (OSM, OMA, OMR) to analog motion (OAD) or vice versa.
After open-loop analog motion was done with OAD, an RNP command must be sent before the servo can be switched on with SVO for closed-loop operation.
The RNP procedure can take up to four times the slewrate value (parameter ID 0x7000002; can be changed with SPA and SEP).
You can query the current state of the system (E-861 and NEXACT® linear drive) using #4 and SRG?.

2.11. Support VIs for scanning algorithms (“Scan support.llb”)

2.11.1. Axis names.vi (Scan support.llb)

Valid for	Analog systems, C702, C-866, C-867, C-880, E-517, E-709, E-710, E-712, E-725, E-753, E-755, E-761, E-861, F-206, M-8X0. To support analog interfacing, VI must be present for E-816 also.
Input	Names (empty string array)
Output	X axis name, Y axis name, Z axis name
Remarks	Checks if "Names" contains three strings for axis names. If this is not the case, it assigns "X Values", "Y Values" and/or "Z Values" as the missing axis name.

2.11.2. Calculate 1D scan positions.vi (Scan support.llb)

Valid for	C-866, C-867, C-880, E-761, E-861, F-206, M-8X0
Input	Scan direction (0: Left & right (- & +)), Start position (0), Range (0), Step size (0)
Output	Minimum position, Maximum position, No. of steps, 1D position array, 1D intensity array
Remarks	Calculates 1D scan positions according to Scan direction, Start position, Range and Step size and returns Minimum position, Maximum position, No. of steps and initialized 1D position array and 1D intensity array. See “1D Scan.vi” for possible Scan directions.

2.11.3. Maximum Intensity?.vi (Scan support.llb)

Valid for	C-866, C-867, C-880, E-761, E-861, F-206, M-8X0
Input	I(in) (0), Gain for I(in) (10), Position for I(in) (empty num. array), I(max, in) (0), Gain for I(max, in) (10), Position for I(max, in) (empty num. array), Clear Maximum (F), Error in (no error)
Output	I(max, out), Gain for I(max, out), Position for I(max, out), Error out
Remarks	Checks if the current intensity value (I(in)), multiplied by the current gain value (Gain for I(in)), is larger than the last maximum intensity value (I(max, in)), multiplied by the last gain value (Gain for I(max, in)), and returns the new maximum intensity values (intensity, gain and position). If Clear Maximum is TRUE, I(max, out) and Position for I(max, out) is set to zero.

2.12. Special commands (“Special command.llb”)

2.12.1. #24.vi (Special command.llb)

Valid for	Analog systems, C-702, C-843, C-843.PM, C-844, C-848, C-865, C-866, C-867, C-880, C-880K005, E-516, E-517, E-709, E-712, E-725, E-753, E-755, E-761, E-816, E-861, F-206, M-8X0, Mercury (but must be present for E-710 also). To support analog interfacing, VI must be present for E-816 also.
Input	System number (1), Error in (no error)
	Analog systems: #24 does not set any error code. When used with any digital controller, does not influence connection between selected analog input

channel and axis.

C-880K005: VI only supported when called through PI_Multix.vi

Output Error out

Remarks Stops all motion (by sending the single ASCII character 24). #24 sets error code 10, call “ERR?.vi” to reset error after #24 has been called.

E-761: #24 does not take effect to analog input which is used for "direct" axis control (see the E-761 User manual). To disable "direct" control for an axis, the value of the corresponding "Aux-Input to target factor" parameter (ID 0x06000902) must be set to 0 with SPA.

E-816: This command cannot be issued to a slave. Check controller manual to find out if #24 is supported.

F-206, M-8X0: Depending on the firmware version on the controller, this command may not take immediate effect for motion initiated by INI or fast scanning commands.

2.12.2. #4.vi (Special command.llb)

Valid for C-867, E-861, M-8X0

Input System number (1), Error in (no error)

Output Status information, Error out

Remarks Request status information by sending the single ASCII character 4. Answer is controller specific. See User Manual for decoding.

M-8X0: Reports error status. Check HELP answer to find out if #4 is supported. Only for controllers based on C-842.80.

2.12.3. #5.vi (Special command.llb)

Valid for Analog systems, C-702, C-843, C-843.PM, C-844, C-848, C-865, C-866, C-867, C-880, E-517, E-709, E-712, E-725, E-753, E-755, E-761, E-861, Mercury (but must be present for all other systems also)

Input System number (1), Error in (no error)

Output Axis moving? (T/F), Error out

Remarks Polls the motion status of the connected axes by sending the single ASCII character 5. Connected axes are read from Global2.vi and displayed on the front panel for assignment. Required by “General wait for movement to stop.vi” and “Wait for axes to stop.vi”.

Analog: Motion status can only be determined for all connected axes, not for single axes.

F-206: Different coding in answer, please use #5_old.vi

M-8X0: Different coding in answer, please use #5_old.vi

2.12.4. #6.vi (Special command.llb)

Valid for C-702, C-848, C-880, E-517, E-761, F-206, M-8X0 (but must be present for C-866, C-867 and E-861 also)

Input System number (1), Error in (no error)

Output Pos change? (T/F), Error out

F-206: Only first field of Pos change? array is valid, refers to whole system.

M-8X0: Only first field of Pos change? array is valid, refers to whole system.

Remarks Polls for change in position of the connected axes by sending the single ASCII character 6. After a position change #6 answer is reset to FALSE with next POS? query. Connected axes are read from Global2.vi and displayed on the front panel for assignment.

E-517: #6 can be used in open-loop and closed-loop operation.
The query considers only motion caused by control sources (e.g. move commands), but ignores position changes caused by amplifier noise.

2.12.5. #7.vi (Special command.llb)

Valid for C-702, C-843, C-843.PM, C-844, C-848, C-865, C-866, C-867, C-880, C-880K005, E-517, E-709, E-710, E-712, E-725, E-755, E-761, E-861, Mercury (but must be present for E-516, E-753, E-816, F-206, M-8X0 also)

Input System number (1), Error in (no error)

C-880K005: VI only supported when called through PI_Multix.vi

Output Ready? (T/F), String read, Error out

Remarks Sends the single ASCII character 7 and returns the ready status of the controller. Sub-VI for “Wait for answer of longlasting command.vi”.

E-712: This VI is to be used during reference moves only. Check HLP? answer to find out if #7 is supported.

2.12.6. ACC.vi (Special command.llb)

Valid for C-843, C-867, E-861

Input System number (1), Without axis ID? (F), No. of digits (4), Axes to set (empty string array), Acceleration values (empty num. array, 0), Error in (no error)

C-843, C-867, E-861: Without axis ID? = FALSE. Acceleration unit is mm/s².

Output Error out, Controller error

Remarks Sets closed-loop acceleration and checks for error. If Without axis ID? is TRUE, then Axes to set is ignored and first field of Acceleration values array is used for acceleration command. Number of digits is the number of digits after the decimal point in the acceleration value(s) that will be sent. Controller error is TRUE if selected system reports error code ≠ 0.

2.12.7. ACC?.vi (Special command.llb)

Valid for C-843, C-867, E-861

Input System number (1), Axes to query (empty string array), All axes? (F), Axis identifier? (T), Error in (no error)

C-843, C-867, E-861: If All axes? = TRUE, then Axis identifier? can be FALSE.
Acceleration unit is mm/s².

Output Acceleration, Error out

Remarks Returns closed-loop acceleration setting for specified axes.

2.12.8. CST?.vi (Special command.llb)

Valid for C-702, C-843, C-843.PM, C-844, C-848, C-865, C-866, C-867, C-880, E-709, E-710, E-712, E-725, E-753, E-755, E-761, E-861, F-206, M-8X0, Mercury

Input System number (1), Axes to query (empty string array), All axes? (F), Axis

identifier? (T), Error in (no error)

C-702, C-843, C-843.PM, C-844, C-848, C-880, F-206, M-8X0: If All axes? = TRUE, then Axis identifier? must be TRUE

C-865, C-866, C-867, E-709, E-710, E-712, E-725, E-753, E-755, E-761, E-861, Mercury: If All axes? = TRUE, then Axis identifier? can be FALSE

Output Stage names, Error out

Remarks Returns the name of the connected stage for queried axes.

F-206, M-8X0: Check HELP answer to find out if CST? is supported.

2.12.9. DEC.vi (Special command.llb)

Valid for C-843, C-867, E-861

Input System number (1), Without axis ID? (F), No. of digits (4), Axes to set (empty string array), Deceleration values (empty num. array, 0), Error in (no error)

C-843, C-867, E-861: Without axis ID? = FALSE. Deceleration unit is mm/s².

Output Error out, Controller error

Remarks Sets closed-loop deceleration and checks for error. If Without axis ID? is TRUE, then Axes to set is ignored and first field of Deceleration values array is used for deceleration command. Number of digits is the number of digits after the decimal point in the deceleration value(s) that will be sent. Controller error is TRUE if selected system reports error code ≠ 0.

2.12.10. DEC?.vi (Special command.llb)

Valid for C-843, C-867, E-861

Input System number (1), Axes to query (empty string array), All axes? (F), Axis identifier? (T), Error in (no error)

C-843, C-867, E-861: If All axes? = TRUE, then Axis identifier? can be FALSE. Deceleration unit is mm/s².

Output Deceleration, Error out

Remarks Returns closed-loop deceleration setting for specified axes.

2.12.11. DIO.vi (Special command.llb)

Valid for C-702, C-843, C-843.PM, C-848, C-865, C-866, C-867, C-880, E-861, Mercury

Input System number (1), DO's to command (empty string array), DO mode (empty bool. array, F), DO mode format (boolean), DO pattern (0), Error in (no error)

C-702, C-843, C-843.PM, C-848: DO's to command can be A-H. DO mode format is Boolean, DO pattern is not valid.

C-865, C-866: DO's to command can be A-B. . DO mode format is Boolean, DO pattern is not valid.

C-867, E-861: For DO mode format = Boolean, DO's to command can be 1-4 and only one DO per command allowed. For DO mode format = hexadecimal, DO's to command and DO mode are not valid and DO pattern must be set correctly (all DO's are set with one single command).

C-880: DO's to command can be A-H (one C-842 inside), A-P (two C-842 inside) etc. DO mode format is Boolean, DO pattern is not valid.

Mercury: DO's to command can be A-D, E-H, I-L etc., see GCS DLL Manual for details. DO mode format is Boolean, DO pattern is not valid.

Output: Error out

Remarks: Switches digital outputs on or off. For DO mode format = Boolean, DO mode must be selected for each DO to command. For DO mode format = hexadecimal (decimal), DO mode and DO's to command are not valid and DO pattern must be selected correctly in hexadecimal (decimal) format.

2.12.12. DIO?.vi (Special command.llb)

Valid for: C-702, C-843, C-843.PM, C-848, C-865, C-866, C-867, C-880, E-517, E-761, E-861, Mercury

Input: System number (1), DI's to query (empty string array), All DI's? (F), DI identifier? (T), Invert order for TVI? (T), Query pattern? (F), Error in (no error)

C-702, C-848, C-880: If All DI's = TRUE, then DI identifier can be FALSE and Invert order for TVI? must be TRUE. Query pattern? is not valid.

C-843, C-843.PM, C-865, C-866: If All DI's = TRUE, then DI identifier must be TRUE and Invert order for TVI? must be FALSE. Query pattern? is not valid.

C-867, E-861: If All DI's = TRUE, then DI identifier must be FALSE. Invert order for TVI? is not valid. DI's to query are 1-4. Query pattern? can be TRUE.

E-517: If All DI's = TRUE, then DI identifier can be FALSE. Invert order for TVI? is not valid. DI's to query are 1-3. Query pattern? is not valid.

E-761: All DI's = FALSE. DI's to query are "1". Query pattern? is not valid.

Mercury: All DI's must be FALSE. DI's to query can be A-D, E-H, I-L etc., see GCS DLL Manual for details. Query pattern? is not valid.

Output: DI value (T/F), Error out

Remarks: Returns digital input values for queried digital inputs. Uses "TIO?.vi" (GCS I and II) and "TVI?.vi" (GCS I) to determine available DI identifiers if All DI's = TRUE and DI identifier = TRUE. If Query pattern? = TRUE, returns binary pattern for the digital input status of all channels.

E-761: Note that the E-761 has no genuine digital input lines, but the analog input is internally interpreted as digital input for triggering tasks (see E-761 User Manual), and its signal state can be queried by this command. If the voltage on the analog input is < 0.8 V, the signal is interpreted as LOW, if the voltage is ≥ 2.4 V, the signal is interpreted as HIGH.

2.12.13. DRC.vi (Special command.llb)

Valid for: C-702, C-843, C-866, C-867, E-517, E-709, E-710, E-712, E-725, E-753, E-755, E-861

Input: System number (1), Rec. table (0), Source ID (empty string), Rec. option (0), Trigger option (0), Error in (no error)

C-843, C-866, E-517, E-709, E-712, E-725, E-753, E-861: Trigger option must be 0.

E-710: Rec. table and Source ID must be identical.

Output: Controller error (T/F), Error out

Remarks: This VI configures the data recording, waits 100 ms and queries ERR?. See GCS DLL manual or User manual for available recording and trigger

options. GCS 2.0: Trigger option must be 0. Controller error is TRUE if selected system reports error code ≠ 0.

C-843: See User Manual for available record options. The C-843 has four data recorder tables. The available points per table depend on the host computer's memory only. Some hardware revisions do not allow the parallel use of DIO and the data recorder. To switch between both, the C-843 needs to be reconnected.

C-866: See C-866_GCS_Commands_SM150E.pdf for available record options.

C-867: See User Manual for available record options.
The C-867 has four data recorder tables with 8192 points per table.

E-517: See User Manual for available record options. The number of data recorder tables is 3 with 8192 points per table.
The current data recorder configuration is saved with WPA, in addition to the current parameter values and other settings.

E-709: See User Manual for available record options. By default, the number of data recorder tables is 4. It can be reduced by setting the appropriate parameter value, see User Manual for details.

E-712, E-725, E-753: See User Manual for available record options.
By default, the number of data recorder tables is 8. It can be reduced by setting the appropriate parameter value, see User Manual for details.

E-861: See User Manual for available record options.
The E-861 has two data recorder tables with 1024 points per table.

2.12.14. DRC?.vi (Special command.llb)

Valid for: C-702, C-843, C-866, C-867, E-517, E-709, E-710, E-712, E-725, E-753, E-755, E-861

Input: System number (1), Rec. table to query (empty num. array, 0), With Rec. table IDs? (F), Error in (no error)

Output: Source ID (empty string array), Rec. option (empty num. array, 0), Trigger option (empty num. array, 0), Error out

C-843, C-866. C-867, E-517, E-709, E-712, E-725, E-753, E-861: Trigger option is not valid.

Remarks: This VI returns the data recording configuration (Source ID, Rec. option and Trigger option) for the queried record table. GCS 2.0: Trigger option is not valid.

2.12.15. DRR?.vi (Special command.llb)

Valid for: Analog systems, C-702, C-843, C-866, C-867, E-517, E-709, E-710, E-712, E-725, E-753, E-755, E-761, E-861, M-8X0. To support analog interfacing, VI must be present for E-816 also.

Input: System number (1), Rec. table IDs (Empty num. array, 0), xo (0), N (100), Nmax (1024), Without parameter? (FALSE), Error in (no error)

Analog: Rec. table IDs, xo, N and Nmax are not valid. Without parameter? must be TRUE.

C-702: Xo >= 0. Nmax = 262144.

C-843: Xo >= 1. Check C-843 User Manual for valid Nmax values. Some hardware revisions don't allow the parallel use of DIO and the data recorder. To switch between both modes the C-843 needs to be reconnected. If N = -1 all points of the last record are returned.

C-866: Xo >= 1. Nmax = 32,256. If N = -1 all points of the last record are returned.

C-867: Xo >= 1. Nmax = 8192.

E-517: Xo >= 1. Nmax = 8192.

E-709: Xo >= 1. Nmax = 4096.

E-710: Xo >= 1. Nmax = 32256.

E-712: Xo >= 1. Nmax = 262,144.

E-725: Xo >= 1. Nmax = 262,144.

E-753: Xo >= 1. Nmax = 65,536.

E-755: Xo >= 1. Nmax = 4096.

E-761: Xo >= 0. Nmax = 8192.

E-861: Xo >= 1. Nmax = 1024.

M-8X0: Rec. table IDs, xo, N and Nmax are not valid. Without parameter? must be TRUE.

Output Data, Names, Sample time, Error out

Remarks Returns N recorded data points. N must be less than or equal to Nmax. For large N values, communication timeout must be set long enough, otherwise a communication error may occur.

C-843: The number of tables is 4. The available points are in equal shares assigned to the available data recorder tables. By default, the number of tables is 4. It can be reduced with DRC, see C-843 User Manual for details.

C-867: The number of tables is 4.

E-517: The number of tables is 3.

E-709: The 4096 points are in equal shares assigned to the available data recorder tables. By default, the number of tables is 4. It can be reduced by setting the appropriate parameter value, see E-709 User Manual for details.

E-761: Recording takes place for all recorder tables as long as the wave generator is running for an arbitrary axis, when an impulse is started with IMP or when a step is started with STE. The assignment of axis and data sources to the recorder tables is as follows:

table 1: axis 1 actual position

table 2: axis 2 actual position

table 3: axis 3 actual position

table 4: analog input voltage (same value as read with TAV?, i.e. contains gain and offset for the analog input, see E-761 User Manual).
The maximum number of data points is 8192 per recorder table.

E-712: The 262,144 points are in equal shares assigned to the available data recorder tables. By default, the number of tables is 8. It can be reduced by setting the appropriate parameter value, see E-712 User Manual for details.

E-725: The 262,144 points are in equal shares assigned to the available data recorder tables. By default, the number of tables is 8. It can be reduced by setting the appropriate parameter value, see E-725 User Manual for details.

E-753: The 65,536 points are in equal shares assigned to the available data recorder tables. By default, the number of tables is 8. It can be reduced by setting the appropriate parameter value, see E-753 User Manual for details.

E-861: Two data recorder tables with 1024 points per table are provided.

M-8X0: Only supported if controller is based on C-842.80 board. Check HELP

answer to find out if DRR? is supported. Returns 360 motor current values recorded during execution of DRV.

2.12.16. DRR? and display data.vi (Special command.llb)

Valid for: Analog systems, C-702, C-843, C-866, C-867, E-517, E-709, E-710, E-712, E-725, E-753, E-755, E-761, E-861, M-8X0. To support analog interfacing, VI must be present for E-816 also.

Input: System number (1), Rec. table IDs (Empty num. array, 0), xo (0), N (100), Nmax (1024), Without parameter? (FALSE), Error in (no error)

Analog: Rec. table IDs, xo, N and Nmax are not valid. Without parameter? must be TRUE.

C-702: Xo >= 0. Nmax = 262144.

C-843: Xo >= 1. Nmax depends on the host computer's memory only. Some hardware revisions don't allow the parallel use of DIO and the data recorder. To switch between both modes the C-843 needs to be reconnected.

C-866: Xo >= 1. Nmax = 32,256. If N = -1 all points of the last record are returned.

C-867: Xo >= 1. Nmax = 8192.

E-517: Xo >= 1. Nmax = 8192.

E-709: Xo >= 1. Nmax = 4096.

E-710: Xo >= 1. Nmax = 32256.

E-712: Xo >= 1. Nmax = 262,144.

E-725: Xo >= 1. Nmax = 262,144.

E-753: Xo >= 1. Nmax = 65,536.

E-755: Xo >= 1. Nmax = 4096.

E-761: Xo >= 0. Nmax = 8192.

E-861: Xo >= 1. Nmax = 1024.

M-8X0: Rec. table IDs, xo, N and Nmax are not valid. Without parameter? must be TRUE.

Output: Data, Names, Sample time, Error out

Remarks: Returns N recorded data points and displays them in a 2D graph by calling "Show_Save_Load_XY_Data.vi. N must be less than or equal to Nmax. For large N values, communication timeout must be set long enough, otherwise a communication error may occur.

C-843: The number of tables is 4. The available points are in equal shares assigned to the available data recorder tables. By default, the number of tables is 4. It can be reduced with DRC, see C-843 User Manual for details.

C-867: The number of tables is 4.

E-517: The number of tables is 3.

E-709: The 4096 points are in equal shares assigned to the available data recorder tables. By default, the number of tables is 4. It can be reduced by setting the appropriate parameter value, see E-709 User Manual for details.

E-761: Recording takes place for all recorder tables as long as the wave generator is running for an arbitrary axis, when an impulse is started with IMP or when a step is started with STE. The assignment of axis and data sources to the recorder tables is as follows:

table 1: axis 1 actual position

table 2: axis 2 actual position

table 3: axis 3 actual position

table 4: analog input voltage (same value as read with TAV?, i.e. contains gain and offset for the analog input, see E-761 User Manual).
The maximum number of data points is 8192 per recorder table.

E-712: The 262,144 points are in equal shares assigned to the available data recorder tables. By default, the number of tables is 8. It can be reduced by setting the appropriate parameter value, see E-712 User Manual for details.

E-725: The 262,144 points are in equal shares assigned to the available data recorder tables. By default, the number of tables is 8. It can be reduced by setting the appropriate parameter value, see E-725 User Manual for details.

E-753: The 65,536 points are in equal shares assigned to the available data recorder tables. By default, the number of tables is 8. It can be reduced by setting the appropriate parameter value, see E-753 User Manual for details.

E-861: Two data recorder tables with 1024 points per table are provided.

M-8X0: Only supported if controller is based on C-842.80 board. Check HELP answer to find out if DRR? is supported. Returns 360 motor current values recorded during execution of DRV.

2.12.17. DRT.vi (Special command.llb)

Valid for	C-702, C-843, C-866, C-867, E-709, E-755, E-861
Input	System number (1), DataRecorderTable (empty num. array, 0), Trigger source (empty num. array, 0), Value (empty string array), Error in (no error)
Output	Controller error (T/F), Error out
Remarks	This vi defines a trigger source for data recorder tables, waits 100 ms and queries ERR?. See GCS DLL manual or User manual for available trigger sources and values. Controller error is TRUE if selected system reports error code ≠ 0.

C-866: See C-866_GCS_Commands_SM150E.pdf for available trigger sources and values. DataRecorderTable = 0 (the specified trigger source is set for all data recorder tables).

C-843, C-867, E-709, E-861: See User Manual for available trigger sources and values. DataRecorderTable = 0 (the specified trigger source is set for all data recorder tables).

2.12.18. DRT?.vi (Special command.llb)

Valid for	C-702, C-843, C-866, C-867, E-709, E-755, E-861
Input	System number (1), Rec. table to query (empty num. array, 0), Error in (no error)
Output	Trigger source (empty num array, 0), Value (empty string array), Trigger option (empty num. array, 0), Error out
Remarks	This VI returns the Data Recorder Trigger source and value for the queried data recorder tables.

2.12.19. HDR?.vi (Special command.llb)

Valid for	C-866, C-867, E-517, E-709, E-712, E-725, E-753, E-861

Input	System number (1), Section header (empty string), Error in (no error)
Output	Data recorder help string, Lines, Enum values, Error out
Remarks	Returns help on data recording: possible parameter values for record sources and record trigger options (DRC, DRT), parameters to set and other information. If Section header contains a valid section name, Lines and Enum values return the correspondig section content.

2.12.20. OAC.vi (Special command.llb)

Valid for	E-861
Input	System number (1), Without axis ID? (F), No. of digits (4), Axes to set (empty string array), Acceleration values (empty num. array, 0), Error in (no error) E-861: Without axis ID? = FALSE. Acceleration unit is steps/s².
Output	Error out, Controller error
Remarks	Sets open-loop acceleration and checks for error. If Without axis ID? is TRUE, then Axes to set is ignored and first field of Acceleration values array is used for acceleration command. Number of digits is the number of digits after the decimal point in the acceleration value(s) that will be sent. Controller error is TRUE if selected system reports error code ≠ 0.

2.12.21. OAC?.vi (Special command.llb)

Valid for	E-861
Input	System number (1), Axes to query (empty string array), All axes? (F), Axis identifier? (T), Error in (no error) E-861: If All axes? = TRUE, then Axis identifier? can be FALSE. Acceleration unit is steps/s².
Output	Acceleration, Error out
Remarks	Returns open-loop acceleration setting for specified axes.

2.12.22. ODC.vi (Special command.llb)

Valid for	E-861
Input	System number (1), Without axis ID? (F), No. of digits (4), Axes to set (empty string array), Deceleration values (empty num. array, 0), Error in (no error) E-861: Without axis ID? = FALSE. Deceleration unit is steps/s².
Output	Error out, Controller error
Remarks	Sets open-loop deceleration and checks for error. If Without axis ID? is TRUE, then Axes to set is ignored and first field of Deceleration values array is used for deceleration command. Number of digits is the number of digits after the decimal point in the deceleration value(s) that will be sent. Controller error is TRUE if selected system reports error code ≠ 0.

2.12.23. ODC?.vi (Special command.llb)

Valid for	E-861
Input	System number (1), Axes to query (empty string array), All axes? (F), Axis identifier? (T), Error in (no error)

E-861: If All axes? = TRUE, then Axis identifier? can be FALSE. Deceleration unit is steps/s².

Output Deceleration, Error out

Remarks Returns open-loop deceleration setting for specified axes.

2.12.24. OMA.vi (Special command.llb)

Valid for E-861

Input System number (1), Axes to move (empty string array), Position values (empty num. array, 0), No. of digits (4), Error in (no error)

E-861: This command works only in open-loop operation. With closed-loop systems, use MOV instead to command motion. Do not call OMA or OMR for axes which have no sensor, otherwise stage will run into hard stop. Motion commands are not allowed when a joystick is active on the axis.

Output Error out

Remarks Open-loop Move Absolute. Sets new absolute open-loop target position for given axes. Servo must be disabled for all commanded axes prior to using this command. No. of digits is the number of digits after the decimal point in the position value(s) that will be sent.

E-861: OMA can not be processed as long as motion commanded by a former OMA or OMR command is still performed.

2.12.25. OMA?.vi (Special command.llb)

Valid for E-861

Input System number (1), Axes to query (empty string array), All axes? (F), Axis identifier? (T), Error in (no error)

E-861: If All axes? = TRUE, then Axis identifier? can be FALSE.

Output Target position, Error out

Remarks Returns commanded open-loop target position.

2.12.26. OMR.vi (Special command.llb)

Valid for E-861

Input System number (1), Axes to move (empty string array), Position values (empty num. array, 0), No. of digits (4), Error in (no error)

E-861: This command works only in open-loop operation. With closed-loop systems, use MVR instead to command motion. Do not call OMA or OMR for axes which have no sensor, otherwise stage will run into hard stop. Motion commands are not allowed when a joystick is active on the axis.

Output Error out

Remarks Open-loop Move Relative. Sets new relative open-loop target position for given axes. Servo must be disabled for all commanded axes prior to using this command. No. of digits is the number of digits after the decimal point in the position value(s) that will be sent.

E-861: OMR can not be processed as long as motion commanded by a former OMR or OMA command is still performed.

2.12.27. OVL.vi (Special command.llb)

Valid for E-861

Input	System number (1), No. of digits (4), Channels to set (empty string array), Velocity values (empty num. array, 0), Error in (no error) All systems: Without axis ID? = FALSE
Output	Error out, Controller error
Remarks	Sets open-loop velocity in steps/s and checks for error. Number of digits is the number of digits after the decimal point in the velocity value(s) that will be sent. Controller error is TRUE if selected system reports error code ≠ 0. E-861: The velocity for open-loop stepping motion is also influenced by the step amplitude set with SSA. The maximum value which can be set with OVL is given by the Open-loop velocity parameter, ID 0x7000201 (can be changed with SPA and SEP).

2.12.28. OVL?.vi (Special command.llb)

Valid for	E-861
Input	System number (1), Channels to query (empty string array), All axes? (F), Error in (no error)
Output	Velocity, Error out
Remarks	Returns open-loop velocity in steps/s for specified channels.

2.12.29. POS.vi (Special command.llb)

Valid for	C-702, C-843, C-843.PM, C-848, C-865, C-866, C-867, C-880, C-880K005, E-712, E-755, E-861, Mercury
Input	System number (1), Axes to set (empty string array), Position values (empty num. array, 0), No. of digits (4), Error in (no error) C-880K005: VI only supported when called through PI_Multix.vi E-712: Check HLP? answer to find out if POS is supported. E-755: Command not available for E-755.101 E-861: With open-loop systems, this command is not useful because there is no position sensor.
Output	Error out
Remarks	Assigns new position value to current position without moving the stage. Command can only be used when the reference mode is switched off (see "RON.vi"). No. of digits is the number of digits after the decimal point in the position value(s) that will be sent. Warning: If the current position is incorrectly set on an axis with reference mode OFF, the stage can be driven into the mechanical hard stop when moving to a position which is thought to be within the travel range of the stage, but actually is not.

2.12.30. RBT.vi (Special command.llb)

Valid for	C-702, C-867, E-517, E-709, E-712, E-725, E-753, E-755, E-761, E-861, F-206, M-8X0
Input	System number (1), Error in (no error)
Output	Error out
Remarks	Reboots the controller. The controller behaves like after a cold start.

2.12.31. RPA.vi (Special command.llb)

Valid for	C-867, E-517, E-709, E-710, E-712, E-725, E-753, E-755, E-761, E-861
Input	System number (1), Affected axes (empty string array), Parameter to restore (empty num. array), Parameter to restore (hex.) (empty hex. array), Parameter no. format (Decimal: FALSE) (F), Error in (no error) E-710, E-761: If Affected axes = empty array, all parameters for all axes are restored. Parameter no. format is TRUE (hex.). Use "HPA?.vi" to get valid parameter numbers or see the E7XX_GCS_DLL Manual. E-517, E-712, E-725, E-753, E-755: If Affected axes = empty array, all parameters for all axes are restored. Parameter no. format is TRUE (hex.). Use "HPA?.vi" to get valid parameter numbers or see User Manual. C-867, E-709, E-861: You can reset either all parameters or one single parameter with RPA. If Affected axes = empty array, all parameters for all axes are restored. Parameter no. format is TRUE (hex.). Use "HPA?.vi" to get valid parameter numbers or see User Manual.
Output	Error out, Hidden error
Remarks	Replaces the current values of the given parameters to restore for Affected axes in the controller RAM with the values from non-volatile memory , waits 5000 ms and queries ERR?. For axis-related parameters, Affected axes is the axis name; for piezo- or sensor-related parameters, the channel number; otherwise a parameter-related code. If parameter number is in decimal format, use Parameter to restore input, for hexadecimal parameter numbers use Parameter to restore (hex.) input and switch Parameter no. format to TRUE. Do not mix decimal and hex. parameter numbers in one call. See GCS DLL manual for available parameter numbers and values. If Affected axes is an empty array, RPA is sent without axis (item) and parameter specification and controller restores all values for all axes (items). Hidden error is TRUE if selected system reports error code ≠ 0.

2.12.32. RTR.vi (Special command.llb)

Valid for	C-843, C-867, E-517, E-709, E-712, E-725, E-753, E-761, E-861
Input	System number (1), Table rate (1), Error in (no error)
Output	Error out, Hidden error
Remarks	This vi sets the table rate and queries ERR?. The table rate is the number of servo-loop cycles to be used in data recording operations. Settings larger than 1 make it possible to cover longer time periods with a limited number of points. Hidden error is TRUE if selected system reports error code ≠ 0.

2.12.33. RTR?.vi (Special command.llb)

Valid for	C-843, C-867, E-517, E-709, E-712, E-725, E-753, E-761, E-861
Input	System number (1), Error in (no error)
Output	Record table rate, Error out
Remarks	Returns the current table rate.

2.12.34. SEP.vi (Special command.llb)

Valid for	C-867, E-517, E-709, E-710, E-712, E-725, E-753, E-755, E-761, E-861
Input	System number (1), Password (100), Axis to set (empty string array), Parameter number (empty num. array, 0), Parameter number (hex) (empty

hex. array, 0), Parameter value (empty num. array, 0), No. of digits (4), Parameter string (empty string array), Parameter no. format (Decimal: FALSE) (F), Parameter format (Num.: FALSE) (F), Error in (no error)

C-867: Parameter no. format is TRUE (hex.). You can write only one single parameter per SEP command. Interpolation unit parameters (if any) can not be changed with SEP. See C-867 User manual for more information.

E-709: Parameter no. format is TRUE (hex.). You can write only one single parameter per SEP command.

E-710: Parameter no. format is TRUE (hex.). Command is available in command level 1 only (see "CCL.vi", "CCL?.vi"). Command writes parameters to Eprom and RAM

E-517, E-712, E-725, E-753, E-755, E-761: Parameter no. format is TRUE (hex.)

E-712, E-725, E-753: Do not set more than 10 parameters at once.

E-861: Parameter no. format is TRUE (hex.)
You can write only one single parameter per SEP command.
The GEMAC parameters (ID 0x7000010 to ID 0x700001F) can not be changed with SEP. Use SPA and WPA instead to save their values to non-volatile memory. See "GEMAC Parameter Adjustment" in the E-861 User manual for more information.

Output: Hidden error (T/F), Error out

Remarks: If Password is correct, this VI sets parameters for commanded axes to EPROM, waits 100 ms and queries ERR?. For axis-related parameters, Axis to set is the axis name; for piezo- or sensor-related parameters, the channel number; otherwise a parameter-related code. If parameter number is in decimal format, use Parameter number input, for hexadecimal parameter numbers use Parameter number (hex.) input and switch Parameter no. format to TRUE. For numeric parameter values use Parameter value input, for parameter strings use Parameter string input and switch Parameter format to TRUE. Do not mix decimal and hex. parameter numbers or numeric and string parameter values in one call. Parameter numbers which can be set depend on current CCL level. Use "HPA?.vi" to get valid parameter numbers or see GCS DLL manual for available parameter numbers and values. No. of digits is the number of digits after the decimal point in the numeric parameter value(s) that will be sent. Hidden error is TRUE if selected system reports error code ≠ 0.

Note: This command sets the same parameters as SPA, but SPA writes them only to volatile memory, while SEP only writes to non-volatile memory. After parameters were set with SEP, use RPA to activate them (write them to volatile memory), or they become active after next power up.

E-712, E-725, E-753: Requires command level 1 ("CCL.vi")

E-709: Required command level depends on parameter ("CCL.vi")

E-517: Requires command level 1 ("CCL.vi") for parameter 0X02000000 (used to enable/disable axes which is only required if the hardware configuration is changed), 0x04000E00 and 0x04000E01 (used to configure unit and format of the LCD display on the E-517 front panel).

2.12.35. SEP?.vi (Special command.llb)

Valid for: C-867, E-517, E-709, E-710, E-712, E-725, E-753, E-755, E-761, E-861

Input: System number (1), Axes to query (empty string array), Parameter no. format (Decimal: FALSE) (F), Without axes? (F), Parameter no. (empty

num. array, 0), Parameter no. (hex) (empty hex. array, 0), Error in (no error)

All systems: Use “HPA?.vi” to get valid parameter numbers. Parameter no. format is TRUE (hex).

C-867, E-709: Only one parameter value for only one axis per command allowed. Use Without axes? = TRUE for a query of all parameters.

E-712, E-725, E-753: Do not query more than 10 parameter no. at once (except with Without axes? = TRUE).

E-861: Only one parameter value for only one axis per command allowed. Use Without axes? = TRUE for a query of all parameters.

Output: Parameter value, Parameter string, Error out

Remarks: Returns parameter values from non-volatile memory for queried axes and parameter numbers. For axis-related parameters, Axes to query is the axis name; for piezo- or sensor-related parameters, the channel number; otherwise a parameter-related code. If parameter number is in decimal format, use “Parameter no.” input, for hexadecimal parameter numbers use “Parameter no. (hex)” input and switch “Parameter no. format” to TRUE. Do not mix decimal and hex. parameter numbers in one call. If Without axes? is TRUE, all available parameter for all axes/designators are returned. For parameter numbers which output a string use Parameter string output. See GCS DLL Manual for available parameter numbers.

2.12.36. SMO.vi (Special command.llb)

Valid for: C-702, C-843, C-843.PM, C-844, C-848, C-865, C-866, C-867, C-880

Input: System number (1), No. of digits (4), Axes to set (empty string array), Motor output (empty num. array, 0), Error in (no error)

All systems: No. of digits is 0.

Output: Error out, Hidden error

Remarks: Sets the motor output directly and checks for error. Command will only be executed if channel is in servo-off mode (SVO.vi).

Caution: In servo-off mode limit switches are not enabled!

No. of digits is the number of digits after the decimal point in the motor output value(s) that will be sent. Hidden error is TRUE if selected system reports error code ≠ 0.

2.12.37. SMO?.vi (Special command.llb)

Valid for: C-702, C-843, C-843.PM, C-844, C-848, C-865, C-866, C-867, C-880

Input: System number (1), Axes to query (empty string array), All axes? (F), Axis identifier? (T), Error in (no error)

C-702, C-848, C-865, C-866, C-867, C-880: If All axes? = TRUE, then Axis identifier? can be FALSE

C-843, C-843.PM, C-844: If All axes? = TRUE, then Axis identifier? must be TRUE

Output: Current motor output, Error out

Remarks: Returns the current motor output. In servo-on mode, the actual value as set by the regulator is reported. In servo-off mode, the value set by the last SMO command is reported.

2.12.38. SRG?.vi (Special command.llb)

Valid for C-843, C-843.PM, C-866, C-867, E-861, Mercury

Input System number (1), Axes to query (empty string array), Register no. (empty num. array, 0), Error in (no error)

C-843, C-843.PM: Register numbers are:

- 1: Event status register
- 2: Activity status register
- 3: Signal status register

C-866: Register numbers are:

- 1: Event status register
- 2: Activity status register
- 3: Signal status register
- 4: Signal sense register

C-867: Register number is:

- 1: Status register

E-861: Register number is:

- 1: Status register

Mercury: Register number is:

- 3: Signal status register

Output Parameter value, Error out

Remarks Returns register values for queried axes and register numbers. See Motion Processor or Controller User Manual for a description of bit-coded answer.

2.12.39. SSA.vi (Special command.llb)

Valid for E-755, E-861

Input System number (1), No. of digits (4), Channels to set (empty string array), Step voltage (empty num. array), Error in (no error)

Output Error out, Hidden error

Remarks Sets driving voltage amplitude for all kind of step motion of specified PiezoWalk channels. No. of digits is the number of digits after the decimal point in the Step voltage value(s) that will be sent. Hidden error is TRUE if selected system reports error code ≠ 0.

E-755: After changing the step voltage, “APG.vi” must be run again.

E-861: SSA changes the value of the Bending Voltage parameter (ID 0x07000003) in volatile memory (can be saved as power-on default with WPA, can also be changed with SPA and SEP).
The SSA setting takes effect for stepping motion in open-loop operation (servo off). Decreasing the step size will decrease the velocity for open-loop stepping motion. The velocity for open-loop stepping motion is also influenced by the open-loop velocity set with OVL.

2.12.40. SSA?.vi (Special command.llb)

Valid for E-755, E-861

Input System number (1), Channels to query (empty string array), All channels? (F), Channel identifier? (F), Error in (no error)

E-755, E-861: If All channels? = TRUE, then Channel identifier? must be FALSE.

Output Step voltage, Error out

Remarks Returns driving voltage amplitude for step mode (step size) of given PiezoWalk channel.

2.12.41. STA?.vi (Special command.llb)

Valid for C-702, C-848, C-880, C-880K005, M-8X0 (but must be present in Special command.llb for all other systems also)

Input System number (1), Axes to query (empty string array), All axes? (F), Axis identifier? (T), Error in (no error)

C-702: If All axes? = TRUE, then Axis identifier? can be FALSE

C-848: If All axes? = TRUE, then Axis identifier? can be FALSE

C-880: If All axes? = TRUE, then Axis identifier? can be FALSE

C-880K005: VI only supported when called through PI_Multix.vi

M-8X0: All axes? = TRUE, Axis identifier? = FALSE

Output Axis status, Error out

C-702: See GCS DLL manual or User manual for supported status bits.

C-848, C-880:

The status word for each axis is a 16-bit register containing the following information (bit encoding is 0 = LSB, 15 = MSB):

Bit #	Description
0	Motion complete flag. This bit is set (1) when the axis trajectory has completed. This flag is only valid for the S-curve, trapezoidal, and velocity contouring profile modes.
1	Wrap-around condition flag. This bit is set (1) when the axis has reached one end of its travel range and has wrapped to the other end of the travel range. Specifically, when traveling in a positive direction past the position +1,073,741,823, the axis will wrap to position -1,073,741,824, and vice-versa. The bit can be reset with the CLR command.
2	Breakpoint reached flag. This bit is set (1) when one of the breakpoint conditions has occurred.
3	Index pulse received flag. This bit is set (1) when an index pulse has been received.
4	Motion error flag. This bit is set (1) when the maximum position error is exceeded. This bit can only be reset when the axis is no longer in a motion error condition
5	Positive limit switch flag. This bit is set (1) when the positive limit switch goes active.
6	Negative limit switch flag. This bit is set (1) when the negative limit switch goes active.
7	Command error flag. This bit is set (1) when an erroneous command has been received by the motion control chip.
8*	Servo-control on/off status (1 indicates on, 0 indicates off).
9*	Axis on/off status (1 indicates on, 0 indicates off). The C-848 always has the axis ON.

10* In-motion flag. This bit is continuously updated and indicates whether or not the axis is in motion: 1 indicates axis is in motion, 0 not in motion.

11* Reserved (may contain 0 or 1)

12*,13* Current axis # (13 bit = high bit, 12 bit = low bit). Axis encoding is as follows:

Bit 13	Bit12	MC Axis	C-848 Axis
0	0	1	A
0	1	2	B
1	0	3	C
1	1	4	D

14,15 Reserved (may contain 0 or 1)

C-880K005:

The status word for each axis is a 16-bit register containing the following information (bit encoding is 0 = LSB, 15 = MSB):

Bit # Description

0 Motion complete flag. Set to 1 when motion is completed. SetMotionCompleteMode determines if this bit is based on the trajectory generator position or the encoder position.

1 Wrap-around condition flag. This bit is set (1) when the actual (encoder) position wraps from maximum allowed position to minimum or vice versa.

2 Breakpoint 1 reached flag. This bit is set (1) when breakpoint 1 is triggered.

3 Capture received flag. This bit is set (1) when a position caputre occures.

4 Motion error flag. This bit is set (1) when a motion error occurs

5 Positive limit switch flag. This bit is set (1) when the positive limit switch goes active.

6 Negative limit switch flag. This bit is set (1) when the negative limit switch goes active.

7 Instruction error flag. This bit is set (1) when an instruction error occurs.

8-10 Reserved, may be 0 or 1.

11 Commutation error flag. This bit is set (1) when a commutation error occurs.

12-13 Reserved, may be 0 or 1.

14 Breakpoint 2 reached flag. This bit is set (1) when breakpoint 2 is triggered.

15 Reserved, may be 0 or 1.

Remarks Returns axis status (integer). Required by “General wait for movement to stop.vi” and “Wait for axes to stop.vi”.

M-8X0: Check HELP answer to find out if STA? is supported. Command is equivalent to #4.

2.12.42. STE.vi (Special command.llb)

Valid for Analog systems, C-843, C-843.PM, C-848, C-865, C-866, C-867, C-880, E-517, E-709, E-710, E-712, E-725, E-753, E-755, E-761, E-861. To support analog interfacing, VI must be present for E-816 also.

Input System number (1), Axis to command (empty string), Step size (0), Delay

(0), No. of digits (4), Error in (no error)

All systems: Delay = 0.

Output Error out

Remarks Performs a step-move from, and back to, the current position with specified step size (amplitude). If supported, Delay sets the number of servo loops between position recording (GCS 2.0: Delay must be 0).. No. of digits is the number of digits after the decimal point in the step size (amplitude) values that will be sent. Controller saves a definite number of position values which can be read out with STE?.vi (GCS 1.0) or DRR?.vi (GCS 2.0). Use “General wait for movement to stop.vi” before calling “STE?.vi” or “DRR?.vi” to make sure that motion has finished before reading back the saved values. For an impulse-move, see “IMP.vi”.

Analog: Use DRR?.vi or DRR? and display data.vi to read position values back.

C-843: Controller saves up to 32,640 position values for all 4 channels in sum. Use STE?.vi to read position values back.

C-843.PM: Controller saves up to 32,640 position values for all 4 channels in sum. Use STE?.vi to read position values back.

C-848: Controller saves 1024 position values. Use STE?.vi to read position values back.

C-865: Controller saves up to 32,640 position values. Use STE?.vi to read position values back.

C-866: Controller saves up to 32,256 position values. STE will overwrite DRC settings of Rec. table 1 to record actual position values. Use DRC to define additional record options for Rec. table no. 2 to 4. Record table rate is reset to 1 by STE. Use STE?.vi to read position values back or DRR? to read all Rec. tables back. You can also use MVR in combination with DRC to record values of a step motion. Use DRR? to read values back then.

C-867: Controller saves up to 8192 position values.
Motion commands like STE are not allowed when the joystick is active for the axis. Use DRC to define record options. Use DRR?.vi or DRR? and display data.vi to read recorded values back. You can also use MVR in combination with DRC to record values of a step motion.

C-880: Controller saves 1024 position values. Use STE?.vi to read position values back.

E-517: Controller saves up to 8,192 position values. Use DRR?.vi or DRR? and display data.vi to read recorded values back. The number of servo cycles used for data recording depends on the setting made with RTR.
Motion commands like STE are not allowed when the E-517 is in OFFLINE mode or when the wave generator output is active. When a macro is running on the E-517, STE will be executed not until the macro is finished or stopped. See "Control Value Generation" and "Control Modes" in the E-517 User manual for details.

E-709: Controller saves up to 4096 position values. Use DRR?.vi or DRR? and display data.vi to read recorded values back. The number of servo cycles used for data recording depends on the setting made with RTR.
Motion commands are not allowed when the wave generator is active or the analog input is used for target generation.

E-710: Controller saves 8192 position values. “Table Rate” parameter, set with SPA, is used as sampling interval instead of Delay. Caution: Table Rate parameter influences Wave Generator, not only STE. Use STE?.vi to read position values back.

E-712: Controller saves up to 262,144 position values. Use DRR?.vi or DRR? and

display data.vi to read recorded values back. The number of servo cycles used for data recording depends on the setting made with RTR.
Motion commands are not allowed when a wave generator is active or the analog input is used for target generation.

E-725: Controller saves up to 262,144 position values. Use DRR?.vi or DRR? and display data.vi to read recorded values back. The number of servo cycles used for data recording depends on the setting made with RTR.
Motion commands are not allowed when a wave generator is active or the analog input is used for target generation.

E-753: Controller saves up to 65,536 position values. Use DRR?.vi or DRR? and display data.vi to read recorded values back. The number of servo cycles used for data recording depends on the setting made with RTR.
Motion commands are not allowed when the wave generator is active or the analog input is used for target generation.

E-755: Controller saves 4,096 position values. Use DRR?.vi or DRR? and display data.vi to read recorded values back.

E-761: Controller saves 8192 position values. The number of servo cycles used for data recording depends on the setting made with RTR. Use DRR?.vi or STE?.vi to read position values back.

E-861: Step response measurements provide meaningful results only in closed-loop operation. Controller saves up to 1,024 position values.
Motion commands like STE are not allowed when the joystick is active for the axis. Use DRC to define record options. Use DRR?.vi or DRR? and display data.vi to read recorded values back. You can also use MVR in combination with DRC to record values of a step motion.

2.12.43. TIO?.vi (Special command.llb)

Valid for	C-702, C-843, C-843.PM, C-848, C-866, C-867, C-880, E-517, E-761, E-861, Mercury (but must be present for E-816 also)
Input	System number (1), Error in (no error)
Output	No. of dig. inputs, No. of dig. outputs, Error out
Remarks	Returns the number of digital inputs and outputs available in the controller. E-761: The E-761 has no genuine digital input and output lines, but the analog input is internally interpreted as digital input for triggering tasks (see E-761 User Manual), and its signal state can be queried by the DIO? command.

2.12.44. TNR?.vi (Special command.llb)

Valid for	C-702, C-866, C-867, E-517, E-709, E-710, E-712, E-725, E-753, E-755, E-761, E-861
Input	System number (1), Error in (no error)
Output	Number of Rec. tables, Error out
Remarks	Returns the number of recording tables.

2.12.45. TPC?.vi (Special command.llb)

Valid for	E-517, E-709, E-710, E-712, E-725, E-753, E-755, E-761 (but must be present for Analog systems, C-867, E-516, E-816 and E-861 also)
Input	System number (1), Error in (no error)
Output	Number of piezo channels, Error out

Remarks Returns the number of available piezo channels.

E-517: Using the Sensor Enable parameter, ID 0x02000000, you can change the E-517 configuration in case of hardware changes, e.g. if you install additional sensor and/or amplifier channels in the system. If this parameter is changed, the Number Of Piezo Channels parameter is adapted automatically. E.g. if parameter 0x02000000 is set to "disabled" for a sensor channel, the corresponding piezo channel is disabled too and no longer included in the TPC? response. See "Configure Axes and Channels" in the E-517 User manual for details.

E-709, E-712, E-725: Returns all Output Signal Channels (piezo channels + analog output channels).

2.12.46. TVI?.vi (Special command.llb)

Valid for C-702, C-843, C-843.PM, C-844, C-848, C-865, C-866, C-867, C-880, E-517, E-710, E-761, Mercury (but must be present for E-816 and E-861 also)

Input System number (1), Invert order, Error in (no error)

C-702, C-848, C-880: Invert order should be TRUE. Returns valid axis identifiers.

C-843, C-843.PM, C-844, C-865, C-866, E-710, E-761, Mercury: Invert order must be FALSE. Returns valid axis identifiers.

C-867, E-517: Invert order must be FALSE. Returns valid characters for axis IDs.

Output Valid axis IDs, Error out

Remarks GCS 1.0: Get valid axis identifiers. Should be called before axes are renamed with SAI.vi.

GCS 2.0: Get valid characters for axis IDs.

2.12.47. WPA.vi (Special command.llb)

Valid for C-867, E-516, E-517, E-709, E-710, E-712, E-725, E-753, E-755, E-761, E-816, E-861

Input System number (1), Password (100), Affected axes (empty string array), Parameter no. format (Decimal: FALSE) (F), Parameter to save (empty num. array), Parameter to save (hex.) (empty hex. array), Parameter, Error in (no error)

C-867: Affected axes = empty array, the currently valid values of all parameters affected by the specified password are saved (see below). Parameter no. format is TRUE (hex).

E-517, E-709, E-710, E-712, E-725, E-753, E-755, E-761: If Affected axes = empty array, all parameters for all axes are saved. Parameter no. format is TRUE (hex).

E-516, E-816: Affected axes and Parameter to save = empty array

E-816: This command cannot be issued to a slave.

E-861: If Affected axes = empty array, the currently valid values of all parameters affected by the specified password are saved (see below). Parameter no. format is TRUE (hex).

Output Error out, Hidden error

Remarks If password is correct, this vi writes current settings of the given parameter numbers for Affected axes to non-volatile memory of the controller, waits 3000 ms (E-725: waits for controller ready by polling with #7) and queries ERR?. For axis-related parameters, Affected axes is the axis name; for piezo- or sensor-related parameters, the channel number; otherwise a

parameter-related code. If parameter number is in decimal format, use Parameter to save input, for hexadecimal parameter numbers use Parameter to save (hex) input and switch Parameter no. format to TRUE. Do not mix decimal and hex. parameter numbers in one call. See GCS DLL Manual for available parameter numbers. If "Affected axes" is an empty array, WPA is sent without axis and parameter specification. Hidden error is TRUE if selected system reports error code ≠ 0.

WARNING:

If current parameter values are incorrect, the system may malfunction. Be sure that you have the correct parameter settings before using the WPA command.

C-867: The WPA command saves the currently valid parameter values to non-volatile memory, where they become the power-on defaults. Settings not saved with WPA will be lost when the C-867 is powered off or rebooted. Parameters can be changed in volatile memory with SPA, ACC, DEC and VEL. The password for writing to non-volatile memory depends on the parameter and can be "100" or "4711". See the parameter list in "Controller Parameters" in the C-867 User manual for the password assignment. WPA must be used without specifying any arguments except of the password, the currently valid values of all parameters affected by the specified password are saved.

E-516: The WPA command saves the currently valid parameters listed below to flash ROM, where they become the power-on defaults. Parameter changes not saved with WPA will be lost when the E-516 is powered off. Communication interface, enabled channels and display format, averaging (AVG), drift compensation mode (DCO), velocity control mode (VCO) and velocity (VEL), offset and gain for position and output voltage display, mode and tolerance for on-target reading (SPA), position limits (NLM, PLM), voltage limits (VMA, VMI), macros and default macro setting.

E-517: The WPA command saves the currently valid parameter values to non-volatile memory, where they become the power-on defaults. Settings not saved with WPA will be lost when the E-517 is powered off or rebooted. Parameters can be changed in volatile memory with SPA, IFC, SAI, DFH, VMI, VMA, WAV, WGC, WOS, WTR, RTR, and VEL. Furthermore, WPA saves the current settings of NLM, PLM (position soft limits), VCO (velocity control mode), DRC (data recorder configuration), CTO (trigger output configuration) and CSV (GCS syntax version, i.e. E-517 or E-516 mode; requires command level 1 (“CCL.vi”).

E-709: The WPA command saves the currently valid parameter values to non-volatile memory, where they become the power-on defaults. Settings not saved with WPA will be lost when the E-709 is powered off or rebooted. Parameters can be changed in volatile memory with SPA, AOS, ATZ, IFC, RTR, VEL, WOS and WTR. Depending on the parameter to be saved, it may be necessary to switch to command level 1 (“CCL.vi”).

E-710: Command is available in command level 1 only (see “CCL.vi”, “CCL?.vi”).

E-712: The WPA command saves the currently valid parameter values to non-volatile memory, where they become the power-on defaults. Settings not saved with WPA will be lost when the E-712 is powered off or rebooted. Parameters can be changed in volatile memory with SPA, AOS, ATZ, DPO, IFC, RTR, VEL, WOS and WTR. Requires command level 1 (“CCL.vi”).

E-725: The WPA command saves the currently valid parameter values to non-volatile memory, where they become the power-on defaults. Settings not saved with WPA will be lost when the E-725 is powered off or rebooted.

Parameters can be changed in volatile memory with SPA, AOS, ATZ, DPO, IFC, RTR, VEL, WOS and WTR. Requires command level 1 (“CCL.vi”). As the WPA command takes up to 90 seconds to finish execution, “WPA.vi” polls for the controller ready signal (#7) before returning.

E-753: The WPA command saves the currently valid parameter values to non-volatile memory, where they become the power-on defaults. Settings not saved with WPA will be lost when the E-753 is powered off or rebooted. Parameters can be changed in volatile memory with SPA, AOS, ATZ, DPO, IFC, RTR, VEL, WOS and WTR. Requires command level 1 (“CCL.vi”).

E-755: The WPA command saves the currently valid parameter values to non-volatile memory, where they become the power-on defaults. Settings not saved with WPA will be lost when the E-755 is powered off or rebooted. Parameters can be changed in volatile memory with SPA, APG, BDR and SSA.

E-761: The WPA command saves the currently valid parameter values and the additional settings listed below to non-volatile memory, where they become the power-on defaults. Settings not saved with WPA will be lost when the PC is powered off or the E-761 is rebooted. Additional settings saved with WPA: Velocity control mode (VCO), position limits (NLM, PLM).

E-861: The WPA command saves the currently valid parameter values to non-volatile memory, where they become the power-on defaults. Settings not saved with WPA will be lost when the E-861 is powered off or rebooted. The password for writing to non-volatile memory depends on the parameter and can be "100" or "4711". See the parameter list in "Controller Parameters" in the E-861 User manual for the password assignment. When WPA is used without specifying any arguments except of the password, the currently valid values of all parameters affected by the specified password are saved. Otherwise only one single parameter can be saved per WPA command. Parameters can be changed in volatile memory with SPA, SSA, ACC, DEC, VEL, OVL, OAC and ODC .

2.13. Support VIs (“Support.llb”)

Support VIs are sub-VIs for command VIs which make certain programming tasks more convenient. They can also be used for building main programs.

Caution: Please do not change these VIs, as that might cause the command VIs that use them to fail.

2.13.1. Analyse input string for terminal.vi (Support.llb)

Valid for	All except analog systems
Input	String new (empty string), Last string sent (empty string)
Output	String out, Out not equal to in? (T/F), Attach term. char.? (T/F)
Remarks	This VI is a sub-VI for “PI Terminal.vi”. It analyses String new and returns it in String out if it is not empty and does not contain a “#” at the beginning. In case of an empty new string, Last string sent is returned. If String new contains a “#” character, the corresponding ASCII character is returned.

2.13.2. Assign booleans from string to axes.vi (Support.llb)

Valid for	All Systems
Input	System number (1), Queried axes (empty string array), All axes queried? (F), Input string (empty string), Error in (no error)
Output	Booleans(T/F), Error out
Remarks	This VI assigns numerical values from input string to boolean values for queried axes. If All axes? is TRUE, connected axes are read from Global2.vi and displayed on the front panel for assignment. Example: An input string like “A=0SpaceLinefeedB=1Linefeed” or “0SpaceLinefeed1Linefeed” will be converted to an output array consisting of two values “FALSE; TRUE”.

2.13.3. Assign DRC values.vi (Support.llb)

Valid for	C-702, C-843, C-866, C-867, E-517, E-709, E-710, E-712, E-725, E-753, E-755, E-861
Input	Input string (empty string), Queried Rec. table (empty num. array, 0), Error in (no error)
Output	Source ID (empty string array), Rec. option (empty num. array, 0), Trigger option (empty num. array), Queried Rec. table out (empty num. array), Rec. option string (empty string array), Error out
Remarks	This VI assigns values (Source ID, Rec. option etc.) from Input string to Queried Rec. tables. Sub-VI for DRC?.vi. GCS 2.0: Trigger option is not valid.

2.13.4. Assign DRT values from string to axes.vi (Support.llb)

Valid for	C-702, C-866, C-867, E-755, E-861
Input	Input string (empty string), Parameter no. format (Decimal: FALSE, F), Syntax (GCS 1.0), Queried axes (empty string array), Parameter no. (empty num. array, 0), Parameter no. (hex) (empty hex. array, 0), Error in (no error)
Output	Parameter values, Parameter strings, Queried axes out, Parameter no. out, Parameter no. (hex) out, Error out
Remarks	This VI assigns numerical values / strings from input string to queried axes and parameter numbers. Sub-VI for “DRT?.vi”.

2.13.5. Assign NaN for chosen axes.vi (Support.llb)

Valid for	Analog systems, C-702, C-843, C-843.PM, C-848, C-865, C-866, C-867, C-880, E-517, E-709, E-712, E-725, E-753, E-755, E-861, F-206, M-8X0, Mercury. To support analog interfacing, VI must be present for E-816 also.
Input	Queried axes (empty string array), Values (empty num. array), Axes subset (empty string array), Value to set (NaN)
Output	New values
Remarks	This VI returns “NaN” or any given Value to set for the given axes subset.

2.13.6. Assign SPA values from string to axes.vi (Support.llb)

Valid for	C-702, C-843, C-843.PM, C-844, C-848, C-865, C-866, C-867, C-880, C-880K005, E-516, E-517, E-709, E-710, E-712, E-725, E-753, E-755, E-761,

	E-816, E-861, Mercury
Input	Input string (empty string), Parameter no. format (Decimal: FALSE, F), Syntax (GCS 1.0), Queried axes (empty string array), Parameter no. (empty num. array, 0), Parameter no. (hex) (empty hex. array, 0), Error in (no error)
Output	Parameter values, Parameter strings, Queried axes out, Parameter no. out, Parameter no. (hex) out, Error out
Remarks	This VI assigns numerical values / strings from input string to queried axes and parameter numbers. Sub-VI for “SPA?.vi” and “SEP?.vi”.

2.13.7. Assign values from string to axes.vi (Support.llb)

Valid for	All systems
Input	System number (1), Queried axes (empty string array), All axes queried? (F), Axes related? (T), Input string (empty string), Error in (no error)
Output	Values, Strings, Error out
Remarks	This VI assigns numerical values and/or single lines from input string to queried axes. If All axes? is TRUE, connected axes are read from Global2.vi and displayed on the front panel for assignment. If All axes? is TRUE and Axes related? is FALSE, item names from Input string are displayed instead of connected axes.

2.13.8. Boolean array calculations.vi (Support.llb)

Valid for	All systems
Input	Array1 (empty bool. array), Array2 (empty bool. array), Array3 (empty bool. array), Operator (AND)
Output	Array out
Remarks	This vi performs a boolean operation of up to three boolean input arrays. The difference to LabVIEWs own boolean operators is that the input arrays can have different sizes. The missing elements are considered to be FALSE elements and the resulting array contains the maximum number of elements.

2.13.9. Build channel query command substring.vi (Support.llb)

Valid for	Analog systems, C-867, E-516, E-517, E-709, E-710, E-712, E-725, E-753, E755, E-761, E-816, E-861
Input	System number (1), Channels to query in (empty string array), Query all channels? (F), With space? (F), Channel identifer? (T), Channel type (piezo), Error in (no error)
Output	Command substring, Channels to query out, Number of rows, Error out
Remarks	This VI builds a query command substring for channel query commands. If All channels? is TRUE, channels to command are determined in a controller specific way and returned in Channels to query out, otherwise Channels to query out is identical with Channels to query in. Number of rows is size of the Channels to query out array. If Channel identifier? is FALSE, command substring is an empty string (e.c. for systems which accept commands like VMA? without channel IDs). If With space? is TRUE, a space character is added between the channel identifiers.

2.13.10. Build command substring.vi (Support.llb)

Valid for	All systems
Input	Affected axes (empty string array), No. of digits (4), Parameters (empty num. array, 0), Parameters (hex.) (empty hex. array), Parameter no. format (Decimal: FALSE) (F), With space? (F)
Output	Command substring
Remarks	This VI builds a command substring by combining axis identifier and parameter. If parameter number is in decimal format, use Parameters input, for hexadecimal parameter numbers use Parameters (hex.) input and switch Parameter no. format to TRUE. Do not mix decimal and hex. parameter numbers in one call. No. of digits is the number of digits after the decimal point in the parameter value(s) that will be sent. Example: For Affected axes = A; B, Parameters = 1.2342; 2.3 and No. of digits = 3 the resulting string is “SpaceA1.234SpaceB2.300”.

2.13.11. Build DIO? query command substring.vi (Support.llb)

Valid for	C-702, C-843, C-843.PM, C-848, C-865, C-866, C-867, C-880, E-517, E-761, E-816, E-861, Mercury
Input	System number (1), DI’s to query in (empty string array), Query all DI’s? (F), DI identifer? (T), Invert order for TVI? (T), Error in (no error)
Output	Command substring, DI’s to query out, Number of rows, Error out
Remarks	This VI builds a DIO? query command substring. If Query all DI’s is TRUE, available analog inputs are read using TIO? and DI identifiers are assigned using TVI? (valid identifiers are assigned to available DI’s in ascending order) (GCS 1.0) or 1 to x with x being the number of available analog inputs (GCS 2.0). Number of rows is the size of the ”DI’s to query out” array. If DI identifier is FALSE, command substring is an empty string.

2.13.12. Build num command substring.vi (Support.llb)

Valid for	All systems
Input	No. of digits (4), Num 1 (empty num. array, 0), Num 2 (empty num. array, 0)
Output	Command substring
Remarks	This VI builds a command substring by combining Num1, Space and Num2. No. of digits is the number of digits after the decimal point in the Num 1/2 value(s) that will be sent. Example: For Num 1 = 1.24; 3.25456, Num 2 = 5.0; 7.4321 and No. of digits = 3 the resulting string is “Space1.240Space5.000Space3.255Space7.432"

2.13.13. Build query command substring.vi (Support.llb)

Valid for	All systems
Input	System number (1), Axes to query in (empty string array), Query all axes? (F), With space? (F), Axis identifer? (T),
Output	Command substring, Axes to query out, Number of rows
Remarks	This VI builds a query command substring. If All axes? is TRUE, connected axes are read from “Global2.vi” and returned in Axes to query out, otherwise Axes to query out is identical with Axes to query in. Number of

rows is size of the Axes to query out array. If Axis identifier? is FALSE, command substring is an empty string (e.c. for systems which accept commands like POS? without axis IDs). If With space? is TRUE or system supports GCS 2.0, a space character is added between the axes identifiers.

Example: If axes A;B;C;D are connected to the system to command, Axes to query in is A;B;D, Query all axes? is TRUE and Use Axis identifier? is TRUE, resulting Command substring is "ABCD", Number of rows is 4 and Axes to query out is A;B;C;D. If With space? is TRUE, the resulting Command substring is "A B C D".

2.13.14. Build SPA command substring.vi (Support.llb)

Valid for	C-702, C-843, C-843.PM, C-844, C-848, C-865, C-866, C-867, C-880, C-880K005, E-516, E-517, E-709, E-712, E-725, E-753, E-816, E-861, Mercury (but must be present for E-710, E-755 and E-761 also)
Input	Axes to set (empty string array), No. of digits (4), Parameter no. format (Decimal: FALSE, F), Parameter format (Num.: FALSE, F), Parameter number (empty num. array, 0), Paramter number (hex) (empty hex. array, 0), Parameter values (empty num. array, 0), Parameter strings (empty string array), With space? (F)
Output	SPA command substring
Remarks	This VI builds a command substring for the SPA command. No. of digits is the number of digits after the decimal point in the parameter value(s) that will be sent. Sub-VI for "SPA.vi", "CTO.vi", "WTR.vi".

2.13.15. Build SPA query command substring.vi (Support.llb)

Valid for	C-702, C-843, C-843.PM, C-844, C-848, C-865, C-866, C-867, C-880, C-880K005, E-516, E-517, E-709, E-710, E-712, E-725, E-753, E-755, E-761, E-816, E-861, Mercury
Input	Axes to query (empty string array), Parameter no. format (Decimal: FALSE, F), Syntax (GCS 1.0), Parameter number (empty num. array, 0), Paramter number (hex) (empty hex. array, 0)
Output	Command substring, Number of rows
Remarks	This VI builds an SPA? Command substring. Axes and parameters are combined into a substring, depending on Parameter no. format.. Number of rows is size of Axes to query array. Sub-VI for "SPA?.vi" and "SEP?.vi" .

2.13.16. Build stringplusnum substring.vi (Support.llb)

Valid for	All systems
Input	Sequence (String1String2String3Value1Value2), String1 (empty string array), String2 (empty string array), String3 (empty string array), Value1 (empty num. array, 0), Value2 (empty num. array, 0), No. of digits Value1 (6), No. of digits Value2 (6), Input selection (T,T,T,T,F), Error in (no error)
Output	Substring, Error out
Remarks	This vi builds a command substring by combining up to three strings and two values in the given order.

2.13.17. Commanded axes connected?.vi (Support.llb)

Valid for	All systems

Input	System number (1), Commanded axes (empty string array), Error in (no error)
Output	Controller error (T/F), Error out
Remarks	This VI checks if Commanded axes are a subset of all connected axes (read from “Global2 (Array).vi”) and returns Controller error TRUE if this is not the case. Connected axes are defined by “Define connected axes.vi”, which is called by “XXX_Configuration_Setup.vi” automatically. White space strings in Commanded axes are ignored.

2.13.18. Convert error to warning.vi (Support.llb)

Valid for	All systems
Input	Error in (no error), Codes (empty num. array)
Output	Error out
Remarks	If code is one of the code numbers given in Codes, resets error status to "no error" and adds "Warning: " to "source".

2.13.19. Convert num array to string.vi (Support.llb)

Valid for	All systems
Input	Number of digits (4), Num. values (empty num. array)
Output	Output string
Remarks	This vi converts an array of numerical values to a space separated output string. The difference to LabVIEW's native Array to Spreadsheet String function is that no carriage return or newline is added.

2.13.20. Convert num value to syntax selection.vi (Support.llb)

Valid for	All systems
Input	GCS syntax version (1,00)
Output	Syntax
Remarks	This VI converts a numerical value to the corresponding GCS syntax version.

2.13.21. Count occurrences in string.vi (Support.llb)

Valid for	All systems
Input	Input string (empty string), Expression (empty string)
Output	Occurrences
Remarks	This VI counts, how often an expression occurs in a string.

2.13.22. Cut out additional spaces.vi (Support.llb)

Valid for	All systems
Input	Mode (All Spaces), String (empty string)
Output	String out
Remarks	Searches for spaces in String and cuts them out, depending on Mode.

2.13.23. Define axes to command from boolean array.vi (Support.llb)

Valid for	All systems
Input	Axes to query (empty string array), Command axis? (empty bool. array, F)
Output	Axes to command, Remaining axes
Remarks	This VI returns only those axis IDs from the Axes to query array in the Axes to command array which have a boolean value TRUE in the Command axis? array, and all remaining axes in the Remaining axes array.

2.13.24. GCSTranslateError.vi (Support.llb)

Valid for	All systems
Input	Error in (no error)
Output	Error out, GCS Error?, Error description
Remarks	Returns if error in contains a GCS error code and if this is the case, it displays the corresponding error message and appends it to source in error out.

2.13.25. General wait for movement to stop.vi (Support.llb)

Valid for	All systems
Input	System no. (1), Axes to wait for (empty string array), All axes? (T), Polling cycle time, ms (1), Additional wait time, ms (0), Error in (no error) E-816: All axes? = FALSE, only one axis per command allowed F-206: VI will not wait for INI procedure to complete. M-8X0: VI will not wait for INI procedure to complete.
Output	Error out
Remarks	This VI waits for the specified axes to stop. An additional wait time can be specified. The wait method depends on the system to command. “XXX_Configuration_Setup.vi” (with XXX being the product name of your system) must be run before running this vi. Requires “Wait for axes to stop.vi”, “#5.vi”,“STA?.vi”, “#5_old.vi”, “ONT?.vi” and “Wait for hexapod system axes to stop.vi” to be present.

2.13.26. Get all axes.vi (Support.llb)

Valid for	All systems
Input	System no. (1)
Output	Conn. Axes
Remarks	This VI reads all connected axes for given system from "Global2 (Array).vi". Connected axes are defined by "Define connected axes.vi", which is called by "XXX_Configuration_Setup.vi" automatically.

2.13.27. Get arrays without blanks.vi (Support.llb)

Valid for	All systems
Input	String array in (empty string array), Values in (empty num. array), Booleans in (empty bool. array, F), Array size in (0)
Output	String array out, Values out, Booleans out, Array size out

Remarks	Returns the string array and related values and boolean arrays without white space string fields.

2.13.28. Get lines and values from string.vi (Support.llb)

Valid for	All systems
Input	Array size (0), Input string (empty string)
Output	Numerical values, Strings
Remarks	This VI returns numerical values and single lines from input string without any axis assignment. If number of lines/values (Array size) is known, algorithm is faster, otherwise Array size = 0 should be used. Sub-VI for "VST?.vi" and "STE?.vi".

2.13.29. Get lines from string.vi (Support.llb)

Valid for	All systems
Input	Array size (0), Input string (empty string)
Output	Strings
Remarks	This VI returns single lines from input string. If number of lines (Array size) is known, algorithm is faster, otherwise Array size = 0 should be used. Sub-VI for "VST?.vi".

2.13.30. Get string array size without blanks.vi (Support.llb)

Valid for	All systems
Input	String array (empty string array)
Output	Corrected array size
Remarks	This VI returns the size of a string array without counting white space strings.

2.13.31. How often does string contain regular expression.vi (Support.llb)

Valid for	All systems
Input	Regular expression (empty string), String (empty string)
Output	Number
Remarks	This VI returns a count of the occurances of a regular expression in a string.

2.13.32. Increase array size.vi (Support.llb)

Valid for	All systems
Input	Size (0), Array in (empty num. array, NaN), Only if Array is not empty?
Output	Array out
Remarks	If size of Array in is smaller than Size, this VI increases the size of Array in to Size. If Array in is an empty array and Only if Array is not empty? is FALSE, VI builds an array of zeros with the size of Size.

2.13.33. Return single characters from string.vi (Support.llb)

Valid for	All systems

Input	Input string (empty string), Invert order (F), Error in (no error)
Output	Character array (empty string array), Error out
Remarks	Get single characters from input string.

2.13.34. Return space.vi (Support.llb)

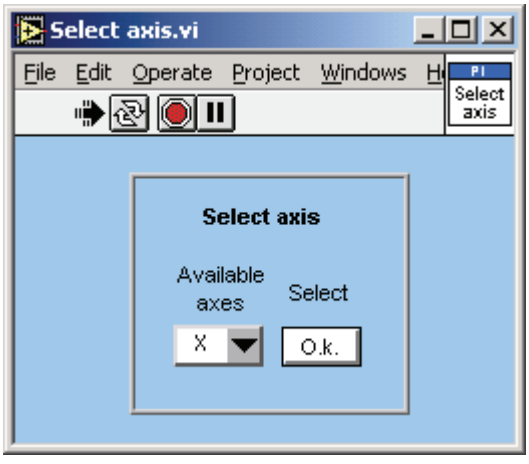
Valid for	All systems
Input	System no. (1), With space? (F)
Output	String out, Space returned?
Remarks	This VI returns a space character in String out if With space? is TRUE or GCS syntax version is higher than 1.0.

2.13.35. Round with options.vi (Support.llb)

Valid for	All systems
Input	No. of digits to round to (2), Round mode selection (Round to nearest), Numeric in (0), Num array in (empty num. array)
Output	Numeric out, Num array out
Remarks	Rounds Numeric in and Num array in according to No. of digits to round to and Round mode selection.

2.13.36. Select axis.vi (Support.llb)

Valid for	All systems
Input	System number (1)
Output	Selected axis, Index of axis in Global2
Remarks	This VI reads all connected axes from Global2 and writes them into a menu ring control for selection. The selected axis and it's index in Global2 are returned.



2.13.37. Select values for chosen axes.vi (Support.llb)

Valid for	All systems
Input	Queried axes (empty string array), Values (empty num. array), Axes subset (empty string array)
Output	Values subset
Remarks	This VI returns only values for the given axes subset.

2.13.38. Select with boolean array input.vi (Support.llb)

Valid for	All systems
Input	Size (0), T string (empty string), F string (empty string), T/F (empty boolean array)
Output	String array out
Remarks	This vi returns a string array of a given size with T string and F string, depending on the boolean value at the corresponding index of T/F.

2.13.39. Selection to string array.vi (Support.llb)

Valid for	All systems
Input	Selection array (empty Menu Ring array, 0), String input (empty string array)
Output	String array
Remarks	This vi returns a string array which contains strings according to the selected value of String input. Example: For Selection array = (2,0,1) and String input = (A,B,C) the resulting String array is (C,A,B).

2.13.40. Set RON and return RON status.vi (Support.llb)

Valid for	C-702, C-843, C-843.PM, C-848, C-865, C-866, C-867, C-880, E-712, E-861, Mercury
Input	System number (1), All axes? (F), Affected axes (empty string array), Reference mode (empty bool. Array, F), Error in (no error)
Output	Axes with RON OFF, Axes with RON ON, Error out
Remarks	Sets RON mode ON or OFF and returns which of the connected axes have RON mode ON and which have RON mode OFF. If reference mode is OFF, no referencing is required for the axis. Only relative moves can be commanded (using "MVR.vi"), unless the actual position is set with POS.vi. Afterwards, relative and absolute moves can be commanded. For stages with neither reference nor limit switch, reference mode is automatically OFF. WARNING! If reference mode is switched off, and relative moves are commanded, stages can be driven into the mechanical hard stop if moving to a position which is outside the travel range! If reference mode is switched off, and the actual position is incorrectly set with "POS.vi", stages can be driven into the mechanical hard stop when moving to a position which is thought to be within the travel range of the stage, but actually is not.

2.13.41. String with ASCII code conversion.vi (Support.llb)

Valid for	All systems
Input	Input string (empty string)
Output	Output string
Remarks	Converts each ASCII control code from Input string to "\x" with x being the

ASCII code of the corresponding character for better readability of log files.

2.13.42. Substract axes array subset from axes array.vi (Support.llb)

Valid for	All systems
Input	Axes to query (empty string array), Axes subset (empty string array)
Output	Axes to command, All present?
Remarks	This VI returns only these axes IDs from the Axes to query array which are **not** present in the Axes subset array. If no axes IDs are returned, All present? is TRUE. Needed by "Define axes to command from boolean array.vi".

2.13.43. Unbundle/bundle interface clusters for PI Terminal.vi (Support.llb)

Valid for	All except analog systems
Input	System number (1), Interface configuration (RS232, 1000, COM1, 57600), DLL interface configuration (C-843, Board, 1), Flow control (All FALSE, x13, x11, x0), TCP/IP Configuration (localhost, 3000, 0), Termination character (LF)
Output	Interface, RS232 configuration system, GPIB configuration system, DLL for device, DLL interface, TCP/IP config. system, Term. char
Remarks	This VI is a sub-VI for "PI Terminal.vi". It unbundles Interface configuration and DLL interface configuration and returns the cluster contents in a different composition which is used by "PI Terminal.vi".

2.13.44. Wait for axes to stop.vi (Support.llb)

Valid for	C-702, C-843, C-843.PM, C-844, C-848, C-865, C-867, C-880, E-517, E-709, E-712, E-725, E-753, E-755, E-861, Mercury (but must be present in Support.llb for all other systems also)
Input	System number (1), Axes to wait for (empty string array), With status bit polling? (F), Polling cycle time, ms (400), Stop refnum (F), Local stop (F), Error in (no error) C-880: With status bit polling? = TRUE All other systems: With status bit polling? = FALSE
Output	Error out
Remarks	This VI waits for the specified axes to stop using #5 polling. It also stops if a communication error occured, Stop refnum or Local stop is TRUE. Requires "STA?.vi" to be present. Required by "General wait for movement to stop.vi". When using as a sub-VI, use Stop refnum to stop VI from caller.

2.13.45. Wait for controller ready.vi (Support.llb)

Valid for	C-702, C-843, C-843.PM, C-848, C-865, C-866, C-867, C-880, E-517, E-709, E-710, E-712, E-755, E-761, E-861, Mercury (but must be present for E-816 also)
Input	System number (1), Polling time, ms (50), Stop refnum (F), Local stop (F), Error in (no error)
Output	Stopped (T/F), Error out
Remarks	This vi waits for controller ready signal using #7 polling and stops also if Stop refnum or Local stop is TRUE, or if a communications error occured.

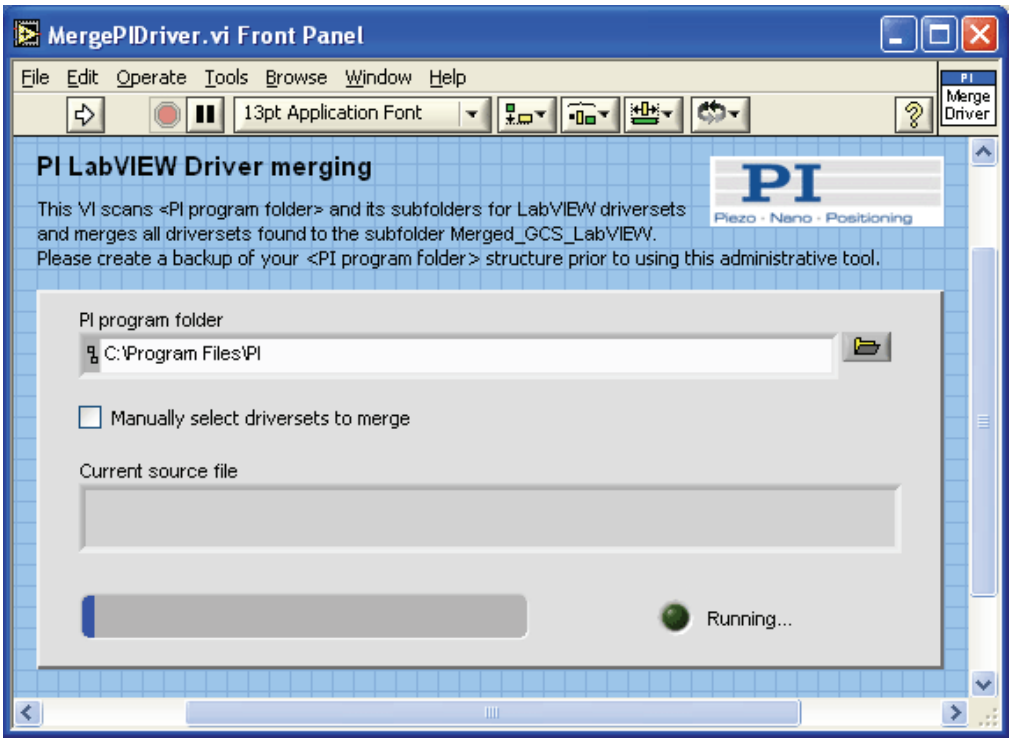
Requires "#7.vi" to be present. When using as a sub-VI, use Stop refnum to stop VI from caller.

The following VIs can be found in the “GCS_LabVIEW_MergeTool” folder:

2.14. Merge Driverset VIs (“MergeDrivers.llb”)

2.14.1. MergePIDriver.vi (MergeDrivers.llb)

Valid for	All systems
Input	PI program folder (C:\Program Files\PI)
Output	none
Remarks	Please create a backup of your <PI program folder> structure prior to using this administrative tool.
	This VI scans the <PI program folder> and its subfolders for any LabVIEW driver sets and merges all driver sets found to the subfolder “Merged_GCS_LabVIEW”. An optional supplemental driver set in the VI's subfolder “MergeSupport” will also be merged.
	Filetypes considered are *.vi, *.llb and *.dll. Files that are already present in the destination path will be overwritten only if the source version is higher than the destination version. The files in the source folders are renamed to prevent LabVIEW from accessing them. Any read-only file attribute will be reset.



Constraints and remarks:

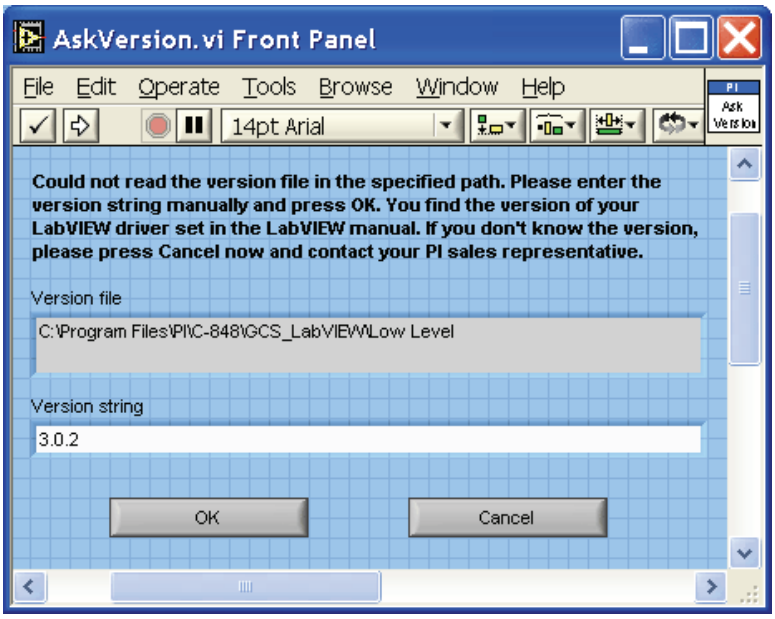
- The optional supplemental merge support folder must be named

"MergeSupport" and be located in the “MergeDrivers.llb” folder

- “MergeDrivers.llb” does not need to be located inside the <PI program folder> structure
- The destination folder is named "Merged_GCS_LabVIEW" and created inside the <PI program folder>
- All Low Level folders must be named "Low Level"
- The "communication.llb" file must be present in a valid PI GCS driver set's Low Level folder
- File names are case insensitive
- Customer VIs must not have the same name as any PI VI
- The "Merged_GCS_LabVIEW\Low Level\merged.txt" file must not be modified outside this merge tool
- There must not be any recursive links inside the <PI program folder> structure
- You need write access rights for the <PI program folder> and its subfolders

2.14.2. AskVersion.vi (MergeDrivers.llb)

Valid for	All systems
Input	Version.txt (empty path constant), Error in (no error)
Output	Version string, Error out
Remarks	Opens a modal dialog to ask for a version string. A “Version.txt” file is then generated in the corresponding Low Level directory.



2.14.3. BuildPath.vi (MergeDrivers.llb)

Valid for	All systems
Input	Version info (Cluster of VI (empty string), Version (0), LLB (empty string) and Path (empty path)), Dest. Path (empty path)
Output	Source path, Dest. path

Remarks	Builds a VI's full path from Version info record.

2.14.4. DeleteObsolete.vi (MergeDrivers.llb)

Valid for	All systems
Input	Dest. Version (empty array of cluster of Path (empty path), LLB (empty string), VI (empty string) and Version (0)), Error in (no error)
Output	Dest. version out, Error out (no error)
Remarks	Deletes old VIs specified in the batch file <obsolete.txt> from merged libraries if the latest version of all merged driver sets is newer than the version specified in the file.

2.14.5. FileOperation.vi (MergeDrivers.llb)

Valid for	All systems
Input	Source path (empty path), Destination path (empty path), Error in (no error)
Output	Error out
Remarks	Resets read only attributes. Deletes a single file if destination path is not empty. Moves a single file if destination path and source path are not empty.

2.14.6. FindVIs.vi (MergeDrivers.llb)

Valid for	All systems
Input	Highlevel paths (array of path (empty path)), error in (no error)
Output	Source versions, Array of cluster of (Path, LLB, VI, Version)
Remarks	Scans the Highlevel paths folders and their "Low Level" folders for VIs, VIs in LLBs and DLLs and adds them to the Source Versions array. Duplicate entries of same VIs in the array are reduced to the latest one only.

2.14.7. GetHighlevelFolders.vi (MergeDrivers.llb)

Valid for	All systems
Input	PI folder (empty path), error in (no error)
Output	Highlevel paths, Error out
Remarks	Scans PI folder and all subfolders for the essential "Communication.llb" PI driver library. If found, the appropriate Highlevel path is added to the Highlevel paths array. Creates subfolder "Merged_GCS_LabVIEW"

2.14.8. GetVersion.vi (MergeDrivers.llb)

Valid for	All systems
Input	Highlevel path (empty path), Error in (no error)
Output	Version, Error out
Remarks	Opens the “Highlevel path\Low Level\Version.txt” file and searches for the line starting with (case insensitive) "VERSION". The following string is converted to a double constant Version, see “VersionString.vi”. If file or “Version” is not found, a dialog window pops up. See “AskVersion.vi”.

2.14.9. MergeFiles.vi (MergeDrivers.llb)

Valid for	All systems
Input	Dest. path (empty path), Source versions (empty array of cluster of Path (empty path), LLB (empty string), VI (empty string) and Version (0)), Dest. versions (empty array of cluster of Path (empty path), LLB (empty string), VI (empty string) and Version (0)), Source VI refnum (empty refnumber), Percent done (empty refnumber), Error in (no error)
Output	# of merged files, Dest. versions out, Error out
Remarks	Copies all files in Source versions to the appropriate destination folder. Updates or adds an entry to the Dest versions array.

2.14.10. RemoveOlderVIs.vi (MergeDrivers.llb)

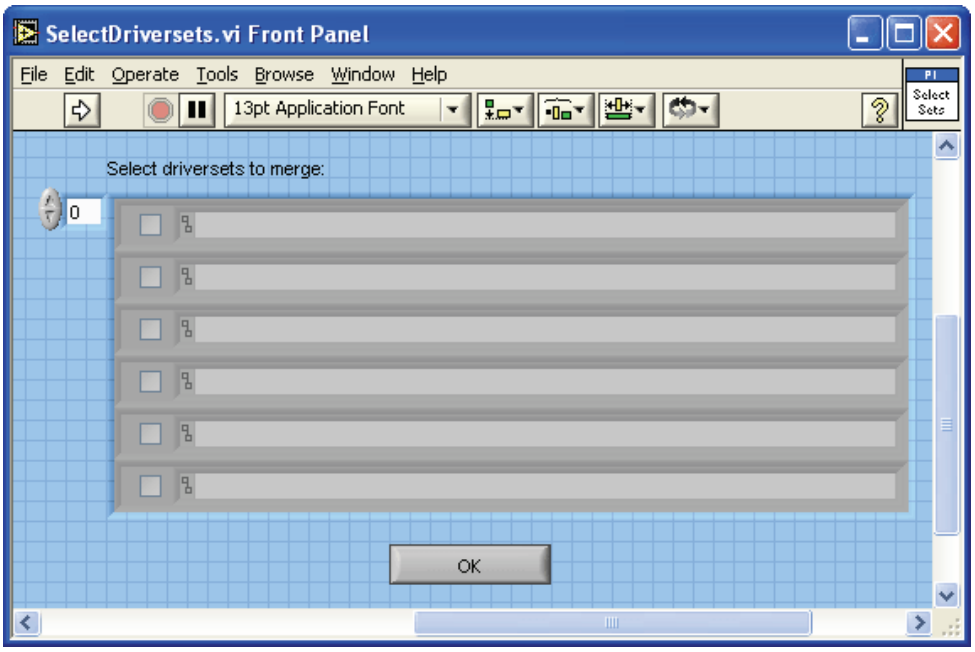
Valid for	All systems
Input	Source versions (empty array of cluster of Path (empty path), LLB (empty string), VI (empty string) and Version (0)), Dest. versions (empty array of cluster of Path (empty path), LLB (empty string), VI (empty string) and Version (0)), Error in (no error)
Output	Source versions out, Error out (no error)
Remarks	Reduces the Source versions array. Entry is removed if 1. Source is not newer than destination, or 2. Analog "dummy" VI would overwrite a functional VI, or 3. Source is merge support supplement and destination nonexistent

2.14.11. Rename.vi (MergeDrivers.llb)

Valid for	All systems
Input	Highlevel paths (empty path array), Error in (no error)
Output	Error out (no error)
Remarks	Renames all files *.vi, *.llb, *.dll in the Highlevel paths and their "Low Level" subfolders to *.v_, *.ll_ or *.dl_ to prevent LabVIEW from accessing them.

2.14.12. SelectDriversets.vi (MergeDrivers.llb)

Valid for	All systems
Input	Highlevel paths (empty path array), Error in (no error)
Output	Highlevel paths out, Error out (no error)
Remarks	Lets the user select driver sets to merge interactively.



2.14.13. SetAttributes.vi (MergeDrivers.llb)

Valid for	All systems
Input	Path (empty path), Error in (no error)
Output	Error out (no error)
Remarks	Resets read only file attributes of all files in Path recursively.

2.14.14. VersionInfo_Load.vi (MergeDrivers.llb)

Valid for	All systems
Input	Dest. Path (empty path), Error in (no error)
Output	Dest. versions, Error out (no error)
Remarks	Loads the version information of VIs in Dest Path from the "Low Level\merged.txt" file.

2.14.15. VersionInfo_Save.vi (MergeDrivers.llb)

Valid for	All systems
Input	Dest. Path (empty path), Dest. versions (empty array of cluster of Path (empty path), LLB (empty string), VI (empty string) and Version (0)), Error in (no error)
Output	Error out (no error)
Remarks	Saves the version information of all merged VIs in Dest. path to the "Low Level\merged.txt" file, also if there is an incoming error.

2.14.16. VersionString.vi (MergeDrivers.llb)

Valid for	All systems
Input	Version string (empty string)
Output	Version
Remarks	Converts a PI version string to a double constant. Only the 5 most significant digits in the string are considered. Digit separator is the decimal

point character. Decimal point is optional, i.e. every single character is treated as a significant digit (1.29 is the same as 1.2.9). Only alphanumeric characters '0-9' and 'A-I' are evaluated, all others are omitted. Case does not matter. "Beta" decreases double by 0.5. First digit can be 0-99.

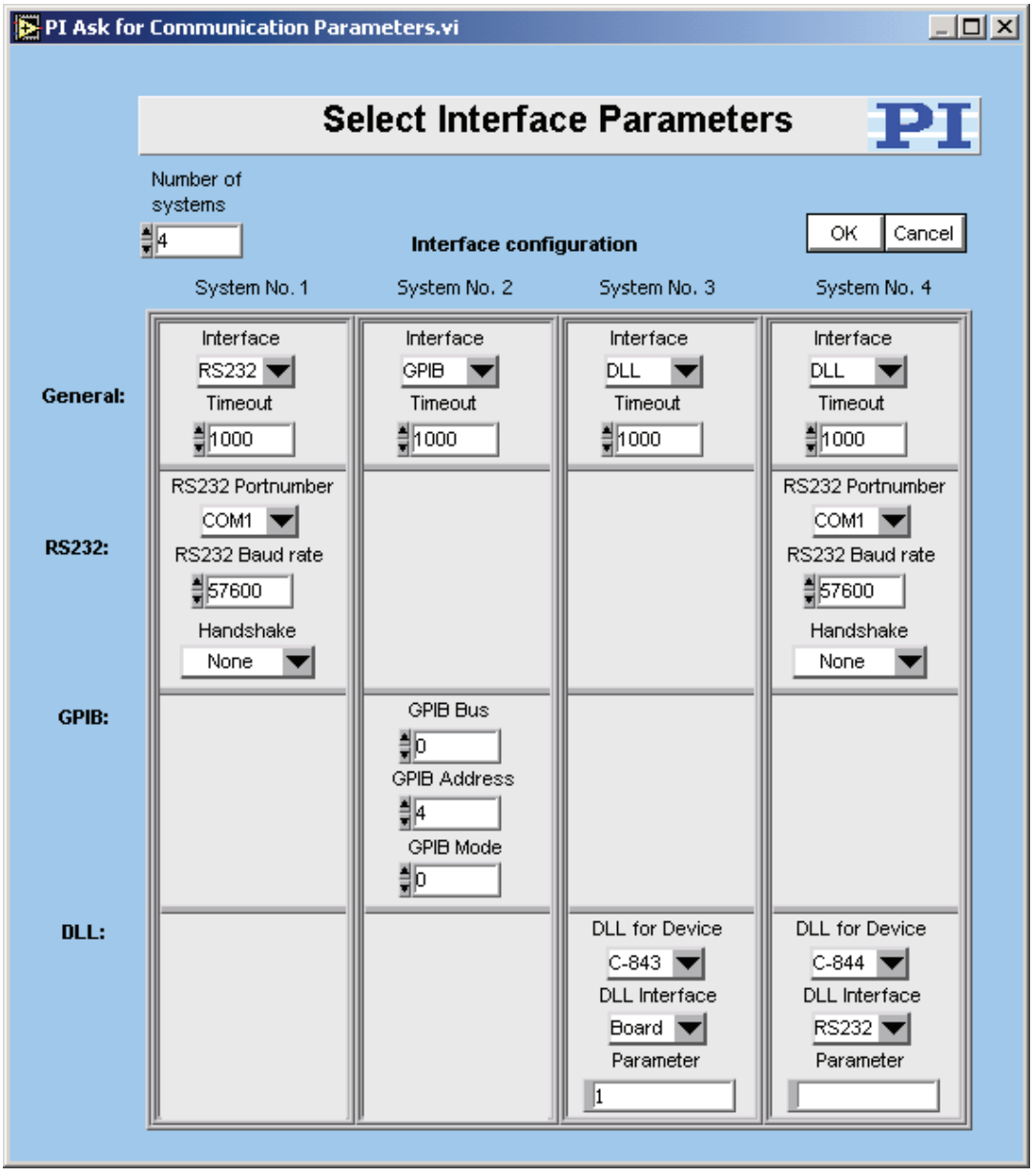
Sample version string: 3.1.4a, 3.1.4(a), 2.8beta, 1.29
Illegal version string: 1.2.3.456 consists of 6 digits

3. High Level VIs

3.1. PI Terminal.vi

The terminal VI is a stand-alone application. It first asks the user to specify the full configuration (number of controlled systems, RS-232, GPIB, TCP/IP or DLL communication, communications parameters), then it establishes a connection with a selected system. This will work for all PI devices which support the PI General Command Set, or at least follow the same syntax rules and support the *IDN? and ERR? commands.

After starting the VI, the interface parameters of the systems with which to communicate must be selected. For this reason, “PITerminal.vi” calls “PI Ask for Communication Parameters.vi”. Select here the number of connected PI systems that you want to communicate with. For each system, select the appropriate interface parameters.



C-702:	Interface = RS232 or TCP/IP, RS232: Input and output HW handshake must be TRUE. Syntax: GCS 1.0; Term char = LF.
C-843:	Interface = DLL, DLL for Device = C-843, DLL Interface = Board, Parameter = Board number (1 for first C-843 board). Syntax: GCS 1.0; Term char = LF.
C-843.PM:	Interface = DLL, DLL for Device = C-843.PM, DLL Interface = Board, Parameter = Board number (1 for first C-843 board). Syntax: GCS 1.0; Term char = LF.
C-844:	Interface = DLL, DLL for Device = C-844, DLL Interface = RS232 or GPIB, Parameter = empty string, RS232 baud rate = 9600. Syntax: GCS 1.0; Term char = LF.
C-865:	Interface = DLL, DLL for Device = C-865, DLL Interface = RS232, Parameter = empty string, RS232 baud rate = set as appropriate. Syntax: GCS 1.0; Term char = LF.
C-866:	Interface = DLL, DLL for Device = C-866, DLL Interface = RS232 or USB, RS232: Parameter = empty string, RS232 baud rate = set as appropriate, USB: Parameter = Serial no. of system to connect to, Syntax: GCS 1.0; Term char = LF.
C-867:	Single Device: Interface = RS232 or DLL, RS232: Input and output HW handshake must be FALSE. DLL (USB): DLL for Device = C-867, DLL Interface = USB, Parameter = Serial no. of system to connect to. DaisyChain: Interface = DLL, DLL for Device = C-867, DLL Interface = RS232_DC, Parameter = Number of device in chain, Register DC: FALSE. Syntax: GCS 2.0; Term char = LF.
C-880, C-848:	Interface = RS232 or GPIB, RS232: Input and output HW handshake must be TRUE. Syntax: GCS 1.0; Term char = LF.
C-880K005:	Interface = RS232, Input and output HW handshake must be FALSE. Syntax: GCS 1.0; Term char = LF.
E-516:	Interface = RS232 or GPIB, RS232: Input and output HW handshake must be TRUE. Syntax: GCS 1.0; Term char = LF.
E-517:	Interface = RS232, GPIB, TCP/IP or DLL, RS232: Input and output HW handshake must be TRUE, DLL (USB): DLL for Device = E-517, DLL Interface = USB, Parameter = Serial no. of system to connect to. Syntax: GCS 2.0; Term char = LF.
E-709:	Interface = RS232 or DLL, RS232: Input and output HW handshake must be TRUE. DLL (USB): DLL for Device = E-709, DLL Interface = USB, Parameter = Serial no. of system to connect to. Syntax: GCS 2.0; Term char = LF.
E-710:	Interface = DLL, DLL for Device = E-710, DLL Interface = RS232 or GPIB, Parameter = empty string. Syntax: GCS 1.0; Term char = LF.
E-712:	Interface = RS232, TCP/IP or DLL, RS232: Input and output HW handshake must be TRUE. DLL: DLL for Device = E-712, DLL Interface = USB, Parameter = Serial no. of system to connect to. Syntax: GCS 2.0; Term char = LF.
E-725:	Interface = RS232, TCP/IP or DLL, RS232: Input and output HW handshake must be TRUE. DLL: DLL for Device = E-725, DLL Interface = USB, Parameter = Serial no. of system to connect to. Syntax: GCS 2.0; Term char = LF.
E-753:	Interface = RS232 or TCP/IP, RS232: Input and output HW handshake must be TRUE. Syntax: GCS 2.0; Term char = LF.
E-755:	Single Device: Interface = RS232, Input and output HW handshake must be TRUE. DaisyChain: Interface = DLL, DLL for Device = E-755, DLL

	Interface = RS232_DC, Parameter = Number of device in chain (first device: 1). Syntax: GCS 2.0; Term char = LF.
E-761:	Interface = DLL, DLL for Device = E-761, DLL Interface = Board, Parameter = Board number (1 for first E-761 board). Syntax: GCS 1.0; Term char = LF.
E-816:	Interface = RS232 or DLL, RS232: Input and output HW handshake must be TRUE. DLL (USB): DLL for Device = E-816, DLL Interface = USB, Parameter = Serial no. of system to connect to. Syntax: GCS 1.0; Term char = LF.
E-861:	Single Device: Interface = RS232 or DLL, RS232: Input and output HW handshake must be FALSE. DLL (USB): DLL for Device = E-861, DLL Interface = USB, Parameter = Serial no. of system to connect to. DaisyChain: Interface = DLL, DLL for Device = E-861, DLL Interface = RS232_DC, Parameter = Number of device in chain (first device: 1), Register DC: FALSE. Syntax: GCS 2.0; Term char = LF.
F-206:	Interface = RS232, GPIB or TCP/IP, The error status will not be cleared by this VI. The first ERR? query will report a hidden error with error code 1, which will be cleared during system initialization (INI). RS232: Input and outputhandshake settings must be FALSE. Syntax: GCS 1.0; Term char = LF.
M-8X0:	Interface = RS232, GPIB or TCP/IP. RS232: Input and output handshake settings must be FALSE. Syntax: GCS 1.0; Term char = LF.
Mercury:	Interface = DLL, DLL for Device = Mercury, DLL Interface = RS232 (even if using USB), Parameter = empty string, RS232 baud rate = same as set on controller hardware, Syntax: GCS 1.0; Term char = LF.

If the chosen timeout value is greater than 300 ms, it will automatically be set to 300 ms for a fluid program operation.

In the upper window ("Send") the user can enter commands which will be transmitted to the chosen device one line at a time when the ENTER key is pressed.

All controller responses are displayed in the Receive response window, which can be cleared by pressing the Clear Receive Window button or F2.

The view style of the Receive window can be changed to Show all characters or Hex View using the menu ring above the Receive window.

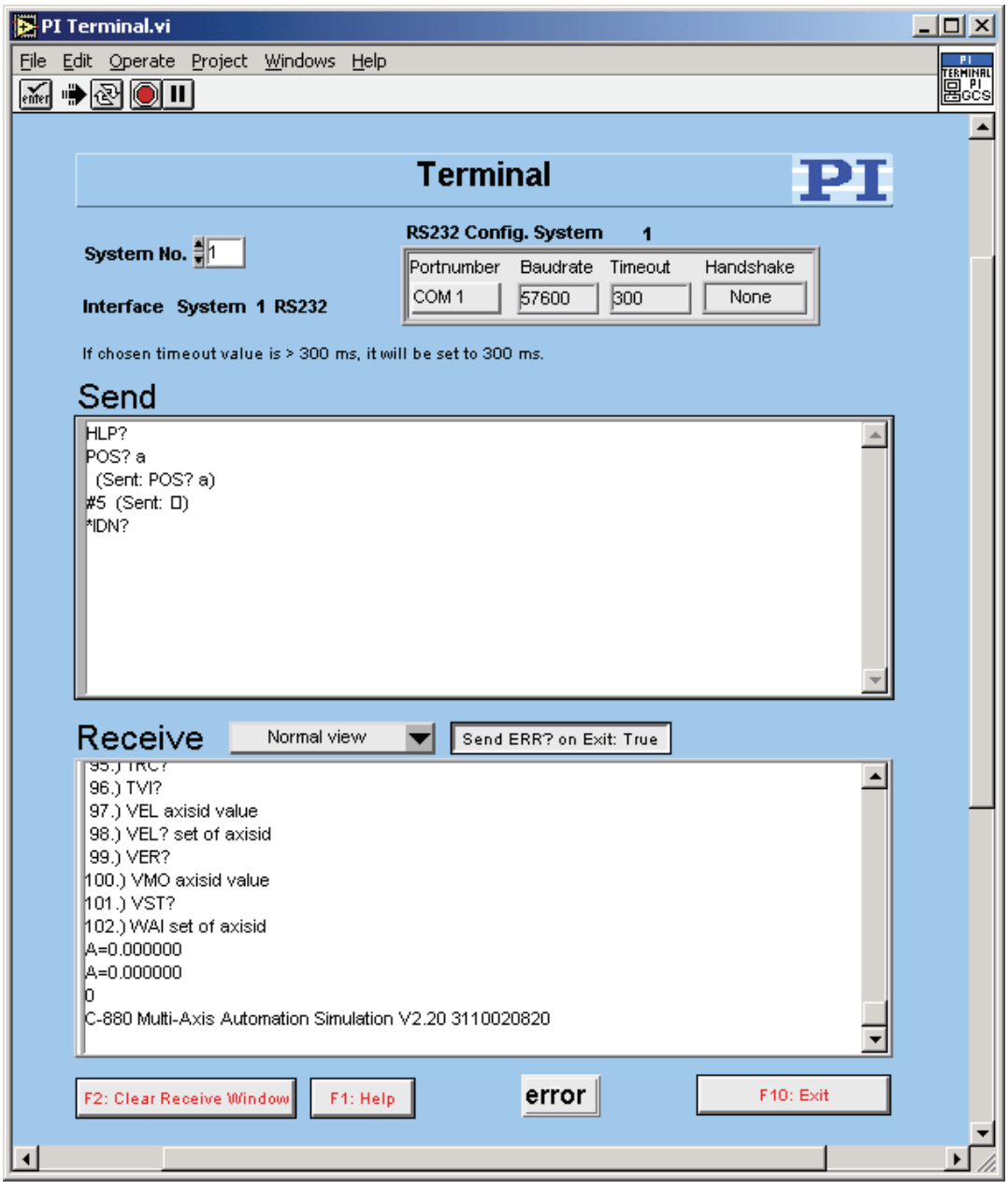
Exit or F10 will terminate the terminal application.

To send the last command again, just press the ENTER key again. The next line will then show the following entry: "(Send: *cmd*)" with *cmd* being the command from the line before, which was resent.

When the terminal application has just been started, pressing ENTER without entering a command will send "*IDN?" to the chosen system.

New commands can only be inserted into the last line of the Send window. The user can scroll through the history of the Send window using the scroll bar or the cursor up/down keys, but cannot change the history or resend commands by pressing ENTER unless in the last line. Pressing ENTER will always resend the last command, no matter where the cursor is positioned. Selecting text and using copy and paste (Ctrl+C, Ctrl+V) works for single lines, if only the contents of one single line (the command text) is selected and copied, not the full line (including the LineFeed) or multiple lines.

Many of PI's General Command Set compatible devices support single-byte commands. For example, the user can stop a fast scan of a C-880 or F-206 by sending an ASCII 24 (decimal). To enter this command into the Send window simply type a "#" followed by the decimal value of the byte to be sent, e.g. enter "#24" and presses ENTER to stop a fast scan. An entry "(Send: *)" will be added to the original command with * being the corresponding ASCII character of the single byte sent.



Pressing F1 or the Help button will pop up a help window. To return to the terminal application, press Esc. If Send ERR? on Exit? is TRUE, an “ERR?” query is sent to the device when Exit is pressed to prevent the controller from keeping an error condition produced during the use of the terminal application.

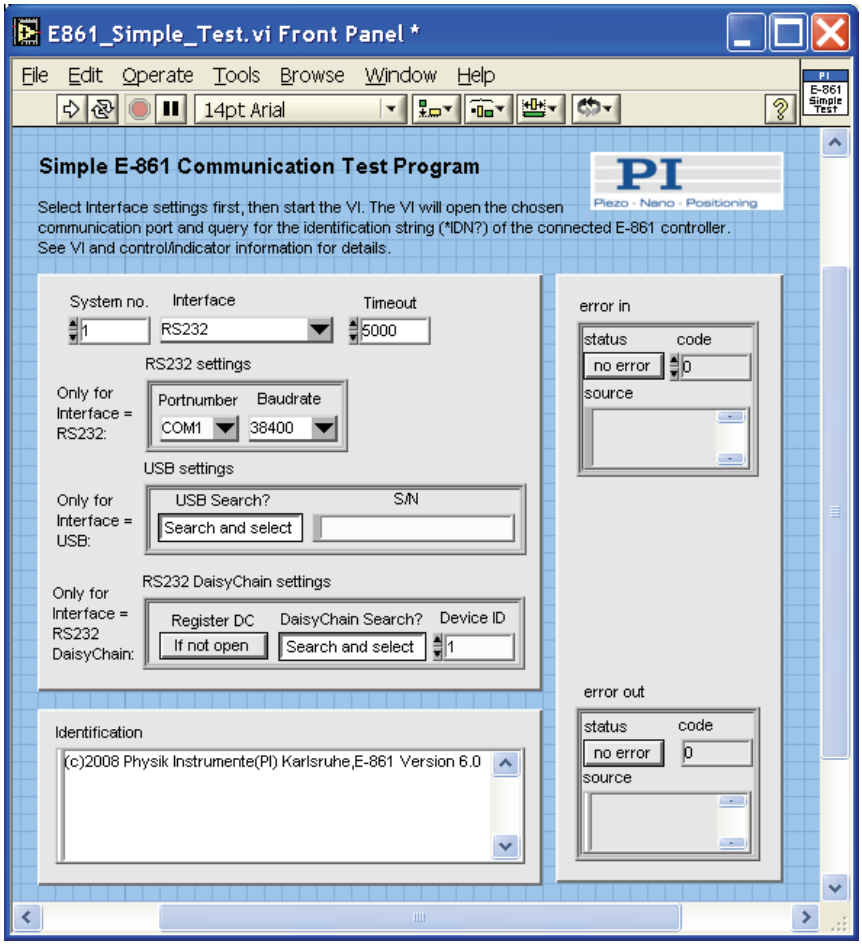
3.2. E861_Simple_Test.vi

This simple test VI is a stand-alone sample application. Use the *Help→Show Context Help* menu sequence in the LabVIEW environment to display the *Context Help* window with the VI and control/indicator descriptions.

Specify

- System number (= 1 in a one-system configuration),
- Interface (RS232, RS232 DaisyChain or USB),
- For Interface = RS232 or RS232 DaisyChain: appropriate port number and RS232 baud rate, or
- For Interface = USB or USB DaisyChain: appropriate Serial Number of the controller to connect to or if you want to search for the controller, or
- For Interface = RS232 DaisyChain or USB DaisyChain: select if you want to register the DaisyChain if it is not already open only (FALSE) or in any case (TRUE), if you want to call a dialog to select the connected E-861 controller from a list (TRUE) or if you know the Device ID of the controller to connect (FALSE, Device ID must match the device ID of the controller in the DaisyChain network). If one device of a DaisyChain is already connected with LabVIEW, do not switch Register DC to TRUE; and
- Timeout value (in milliseconds).

Then start the VI. The VI will open the chosen communication port and query for the identification string of the connected E-861 controller. The diagram shows how to combine the driver and support VIs for these tasks.



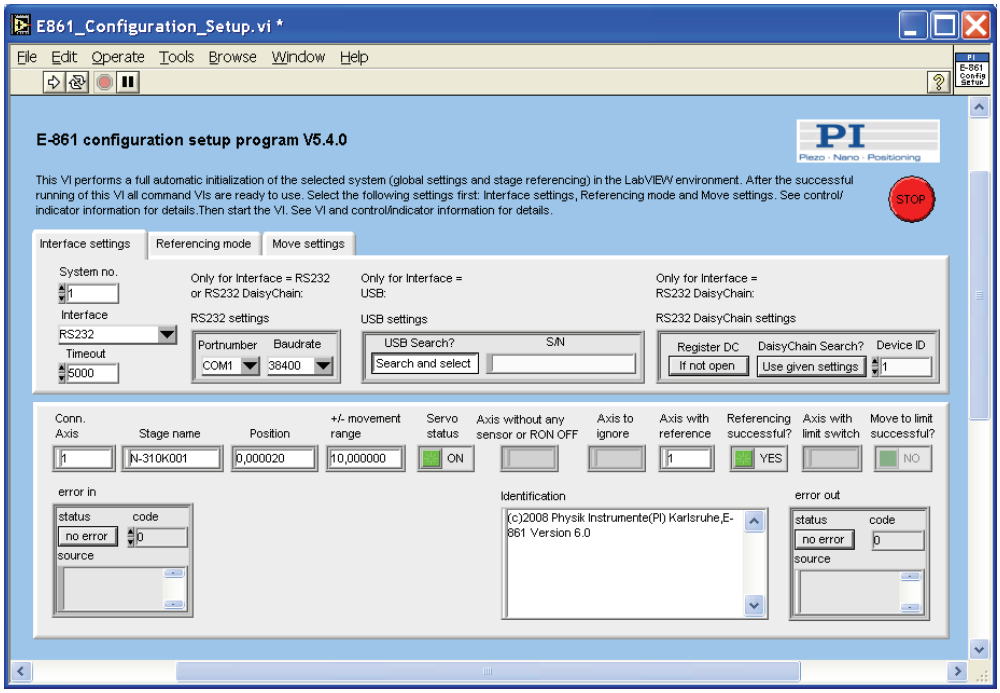
3.3. E861_Configuration_Setup.vi

This VI performs a fully automatic initialization of the selected system (global settings and stage referencing) in the LabVIEW environment. Use the *Help→Show Context Help* menu sequence in the LabVIEW environment to display the *Context Help* window with the VI and control/indicator descriptions.

After the successful running of this VI, all command VIs are ready to use. Specify the correct parameters first:

- System No.: 1 in a one-system-only configuration.
- Interface (RS232, RS232 DaisyChain, USB or USB DaisyChain)
- For Interface = RS232 or RS232 DaisyCain: RS-232 settings: Choose the port number the E-861 is connected to and an appropriate baud rate.
- For Interface = USB or USB DaisyChain: appropriate Serial Number of the controller to connect to or if you want to search for the controller.
- For Interface = RS232 DaisyChain or USB DaisyChain: select if you want to register the DaisyChain if it is not already open only (FALSE) or in any case (TRUE), if you want to call a dialog to select the connected E-861 controller from a list (TRUE) or if you know the Device ID of the controller to connect (FALSE, Device ID must match the device ID of the controller in the DaisyChain network). If one device of a DaisyChain is already connected with LabVIEW, do not switch Register DC to TRUE
- Timeout: choose an appropriate timeout value here (default is 5000 ms).
- Reference mode (only if needed and for closed-loop systems only, i.e. if a position feedback device is provided for the mechanics), see description of RON.vi for details and warnings)
- Whether or not axis is connected, closed-loop and can be moved (depends on mechanical setup).
- If axis is connected, closed-loop and can be moved, whether or not servo is to be switched on.
- If axis is connected, closed-loop, can be moved and servo is ON, which limit switch to use for referencing an axis that is to be referenced but has no reference switch.
- If axis is connected, closed-loop, can be moved, servo is ON and referencing was successful, whether or not axis is to be moved to the middle position of its travel range.

Then start the VI.



"E861_Configuration_Setup.vi" performs the following initialization tasks:

1. Runs "PI Open Interface of one system.vi" to open a connection to the E-861 controller.
2. Runs "*IDN?.vi" to query for the controller identification string.
3. Defines the selected system to be "E-861".
4. Runs "Define connected axes.vi" with Read from controller = TRUE and Invert Order = FALSE.
5. Runs "CST?.vi" to query for the connected stage (use PIMikroMove to configure stages connected to E-861). If Connected? is FALSE (because no stage is connected to the controller, axis is open-loop (i.e. the connected mechanics does not contain a position feedback device) or cannot be moved), proceeds with step 18, otherwise step 6.
6. Runs "SVO.vi" to switch servo on (if Switch servo on? is TRUE)
7. Runs "SVO?.vi" to display servo status of axis
8. Sets the reference mode for the axis, if specified (Reference mode tab).
9. Reads whether axis has reference mode OFF.
10. If not, determines if axis has a reference switch (TRS?).
11. If yes, moves axis to the reference position (FRF) and queries if referencing was successful (FRF?).
12. If axis has no reference switch but has reference mode ON, reads if axis has a limit switch (LIM?).
13. If yes, moves axis to the positive or negative limit position (FPL/FNL) as specified and queries if referencing was successful (FRF?).
14. Reads the position range (TMN?, TMX?).
15. If Move to middle? is TRUE, moves axis to the middle position of its range (MOV) if it has been referenced before by a reference move (FRF/FPL/FNL).
16. Waits for motion to stop.

17. Runs “POS?.vi” to query for the position.
18. Runs “ERR?.vi” to query the controller for its error status.
19. Runs “GCSTranslateError.vi” to append the error message which corresponds with a GCS error number returned by “ERR?.vi” to Source from Error out.

If axis has referencing mode OFF, referencing was not successful, or axis was designated not to be moved, it will report NaN as position value. An axis with RON OFF can only be moved relative until its position is set with "POS.vi". See description of "RON.vi" for details. If Axis was designated not to be moved during runtime of E861_Configuration_Setup it cannot be moved until referenced separately. If no position feedback device (sensor) is present at all, the axis can only be moved with open-loop motion commands, see "OAD.vi" and "OSM.vi" for details (see also E861_Sample_Application_OpenLoop.vi for how to handle open-loop systems).

Warning: If no sensor is present, do not switch the servo on and do not send commands for closed-loop motion, like MOV or MVR. Furthermore, do not use the OMA and OMR open-loop motion commands. Otherwise the connected mechanics can run into the hardstop at full speed which may cause damage to your hardware setup.

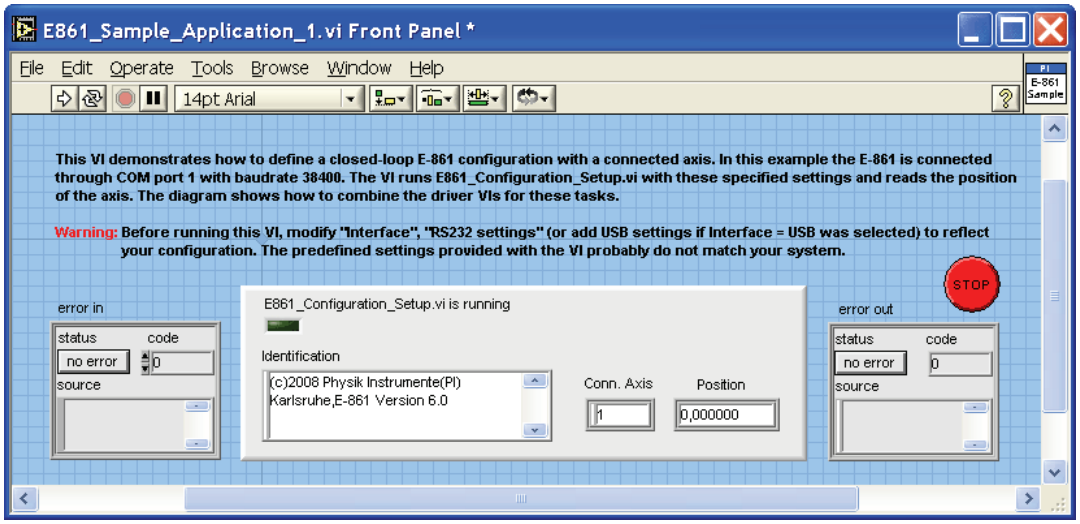
Use this VI as the initialization VI for the E-861 in your application.

As the initialization is a complex procedure which uses a large number of sub-VIs, E861_Configuration_Setup.vi is password-protected, meaning that you cannot see or modify the diagram. In this way, the full initialization is packed into one single and fully tested procedure which you simply insert into your own application program. For security reasons, as well as your convenience, we recommend that you not modify this VI.

3.4. E861_Sample_Application_1.vi

This VI demonstrates how to define a closed-loop E-861 configuration with a connected axis. In this example the E-861 is connected through COM port 1 with baudrate 38400. The VI runs E861_Configuration_Setup.vi with these specified settings and reads the position of the axis. The diagram shows how to combine the driver VIs for these tasks.

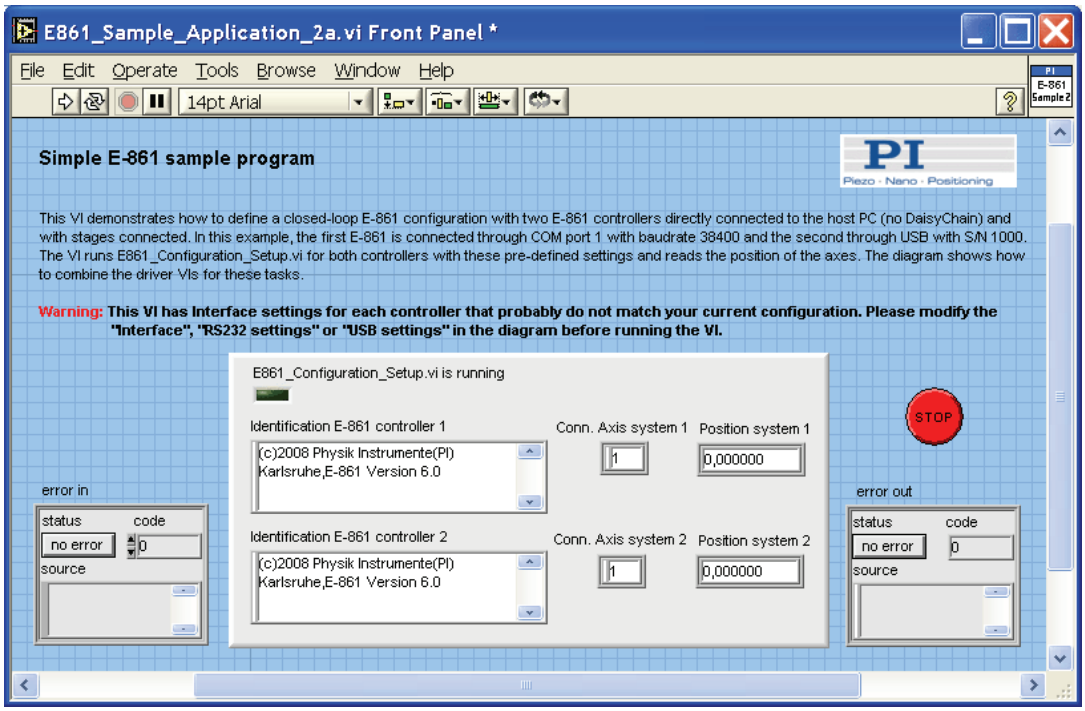
Warning: Before running this VI, modify Interface and RS232 settings (or add USB settings if Interface = USB was selected) to reflect your configuration. The predefined settings provided with the VI probably do not match your system.



3.5. E861_Sample_Application_2a.vi

This VI demonstrates how to define a closed-loop E-861 configuration with two E-861 controllers directly connected to the host PC (no DaisyChain) and with stages connected. In this example, the first E-861 is connected through COM port 1 with baudrate 38400 and the second through USB with S/N 1000. The VI runs E861_Configuration_Setup.vi for both controllers with these pre-defined settings and reads the position of the axes. The diagram shows how to combine the driver VIs for these tasks.

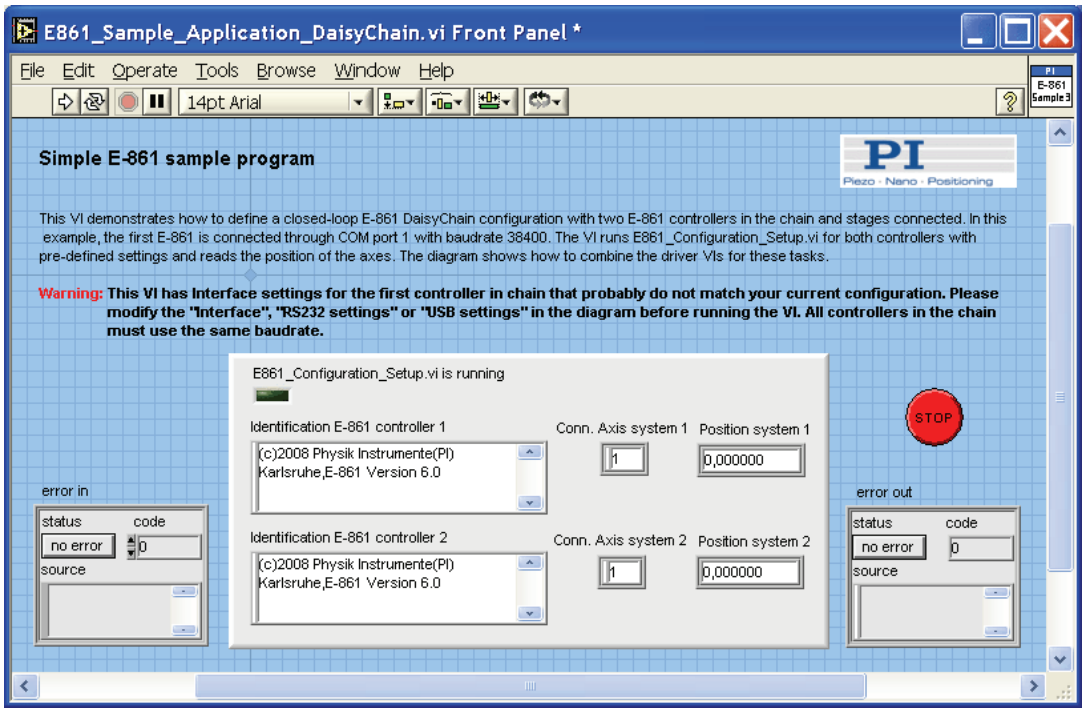
Warning: This VI has Interface settings for each controller that probably do not match your current configuration. Please modify the "Interface", "RS232 settings" or "USB settings" in the diagram before running the VI.



3.6. E861_Sample_Application_DaisyChain.vi

This VI demonstrates how to define a closed-loop E-861 DaisyChain configuration with two E-861 controllers in the chain and stages connected. In this example, the first E-861 is connected through COM port 1 with baudrate 38400. The VI runs E861_Configuration_Setup.vi for both controllers with pre-defined settings and reads the position of the axes. The diagram shows how to combine the driver VIs for these tasks.

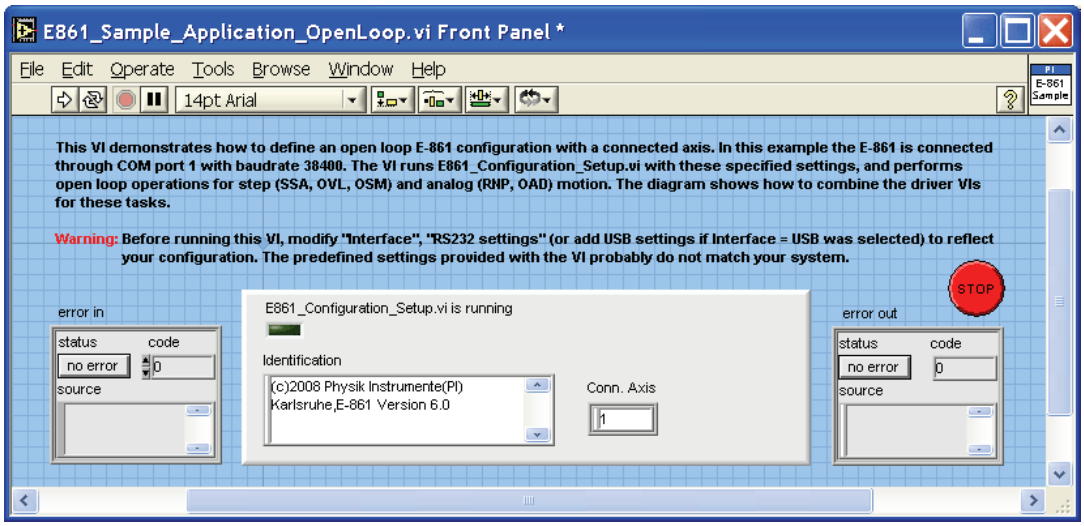
Warning: This VI has Interface settings for the first controller in chain that probably do not match your current configuration. Please modify the "Interface", "RS232 settings" or "USB settings" in the diagram before running the VI. All controllers in the chain must use the same baudrate.



3.7. E861_Sample_Application_OpenLoop.vi

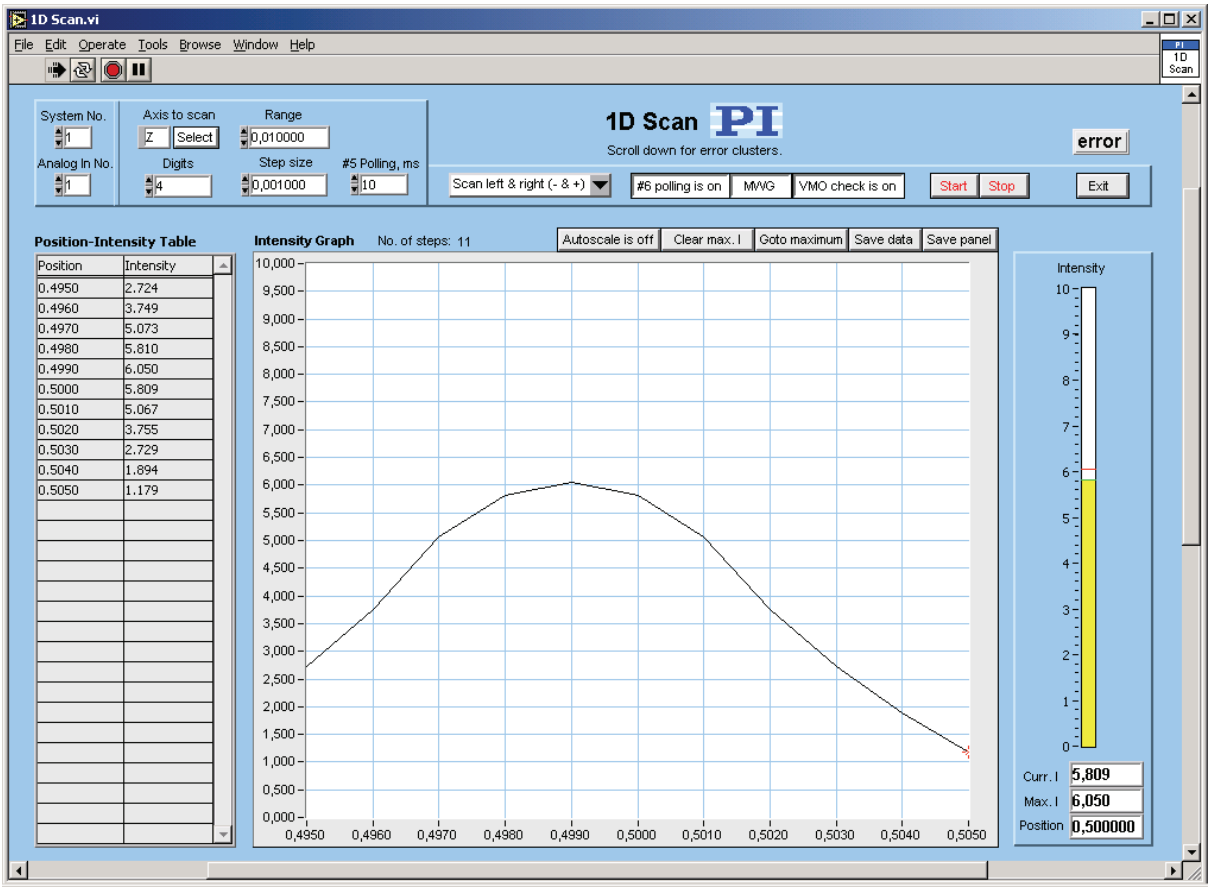
This VI demonstrates how to define an open-loop E-861 configuration with a connected axis. In this example the E-861 is connected through COM port 1 with baudrate 38400. The VI runs E861_Configuration_Setup.vi with these specified settings, and performs open-loop operations for rough (SSA, OVL, OSM) and fine (RNP, OAD) positioning. In detail, it sets the open-loop driving voltage amplitude for step motion to 55 V (SSA), sets the open-loop velocity to 200 steps/s (OVL), performs an open-loop step motion of 200 steps (OSM), relaxes the piezos without motion (RNP) and performs an analog motion of -10 V (OAD; with an N-310.01 NEXACT® linear drive, for example, this corresponds to a motion of about 0.9 µm in negative direction). The diagram shows how to combine the driver VIs for these tasks

Warning: Before running this VI, modify Interface and RS232 settings (or add USB settings if Interface = USB was selected) to reflect your configuration. The predefined settings provided with the VI probably do not match your system.



3.8. 1D Scan.vi

This VI performs a one-dimensional (1D) scan along one axis while monitoring an analog signal. The direction of the scan can be selected. After the scan, the axis to scan is optionally moved to the start position or to the position where the maximum occurred. The range and step size can be specified. After the scan, the axis can be moved to the maximum intensity position. Run XXX_Configuration_Setup.vi before starting this VI.



Start the VI and select the system number (System No. = 1 in a one-system configuration) and Analog In number (Analog In No. = number of the analog channel to observe). Pressing Select will open a user interface which shows a list box with all axes connected to the selected system. Choose the axis to scan here or type the ID of the axis to scan directly into the Axis to scan control. Then, select Range and Step size for the scan. Digits is the number of digits after the decimal point in the position values that will be sent. #5 Polling is the cycle time in ms of the polling used to determine if motion has stopped after commanding a move. If this value is too large, scans may last very long. Set Scan orientation as needed (see possible orientations below). If #6 polling is on, VI will poll for position changes when idle. See control descriptions below to determine whether your system supports #6 polling. If MWG is on, the VI will use “MWG.vi” instead of “MOV.vi”. This will result in a faster scan for systems which support the MWG command. See control descriptions below to determine if your system supports MWG. If VMO check is on, the VI will check whether the minimum and maximum position values of the scan line are within the system workspace before starting the scan. It is assumed that all positions between these points are reachable. See control descriptions below to determine whether your system supports VMO.

Intensity graph shows the scanned intensity distribution. Autoscale can be switched on or off. If Autoscale is off, intensity is scaled from 0-10.

Press Goto max to go to the position where the maximum intensity was found during the scan. With Clear max. the intensity value and the position value for the maximum intensity can be cleared. Press Save data to save scan data (axis, position and intensity information). Data will be saved in GCS Array format. With Save panel a screen copy of this VI can be saved as a JPG file.

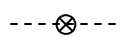
Position-Intensity Table will show the position value of the axis to scan and the corresponding intensity value which was captured during the scan. On the right side of the VI panel, the current and maximum intensity values and the current position value are shown.

Press Stop to stop the scan and Exit to stop execution of this VI.

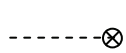
To use this VI as a sub-VI for your application without user interaction, wire the desired scan parameters (Axis to scan, Range, Step size, No. of pos. dig., #5 polling, #6 polling?, Use MWG?, VMO check?) to the corresponding terminals, and wire a TRUE constant to the Start scan, Goto max and Exit terminals. After calling the VI, it will make a scan, go to the maximum of the intensity distribution and finish execution.

When using as a sub-VI, connect Stop refnum terminal to stop VI from caller.

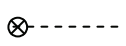
Valid for	C-866, C-867, C-880, E-761, E-861, F-206, M-8X0
Input	System No. (1), Analog In No. (1), Axis to scan (Y), Range (0.02), Step size (0.001), Digits (4), #5 polling, ms (10), Scan direction (0: Scan left&right (-&+)), #6 polling? (TRUE), Use MWG? (FALSE), VMO check? (TRUE), Exit (FALSE), Error in (no error) C-866: #6 polling? = FALSE, Use MWG? =FALSE, VMO check? =FALSE C-867: #6 polling? = FALSE, Use MWG? =FALSE, VMO check? =FALSE C-880: #6 polling? = TRUE, Use MWG? = FALSE, VMO check? = TRUE E-761: #6 polling? = TRUE, Use MWG? = FALSE, VMO check? = FALSE E-861: #6 polling? = FALSE, Use MWG? =FALSE, VMO check? =FALSE F-206: #6 polling? = TRUE, Use MWG? = TRUE, VMO check? = TRUE M-8X0: #6 polling? = TRUE, Use MWG? = FALSE, VMO check? = TRUE
Output	1D Array I, 1D Array X, Intensity, Position, Error out
Remarks	Scan direction can be



0: *Left & right (- & +)* meaning that the scan starts at (Start position – ½ Range) and stops at (Start position + ½ Range),



1: *To the left (-)* meaning that scan starts at (Start position – Range) and stops at Start position, or



2: *To the right (+)* meaning that scan starts at Start position and stops at (Start position + Range).

3.9. Joystick Operation.vi

This VI can be used to control 2 closed-loop axes (which can belong to one or two connected systems) with a standard 2-button, 2-axis joystick connected to the game port of the host computer. The absolute value of the joystick position is converted into velocity values for the two stages connected to the system being commanded. Two velocity levels for each axis can be specified, e.g. one for fast, rough positioning and one for slow, fine positioning. Joystick button 2 switches between these levels. The sign of the joystick position determines whether the move command issued contains the positive or negative travel limit (read automatically if Read travel range from controller? is TRUE) of the corresponding axis. When the joystick is “in the middle position”, the velocity of the corresponding axis is set to zero.

Dead band * is the maximum size of the scaled joystick position value that does not result in any motion.

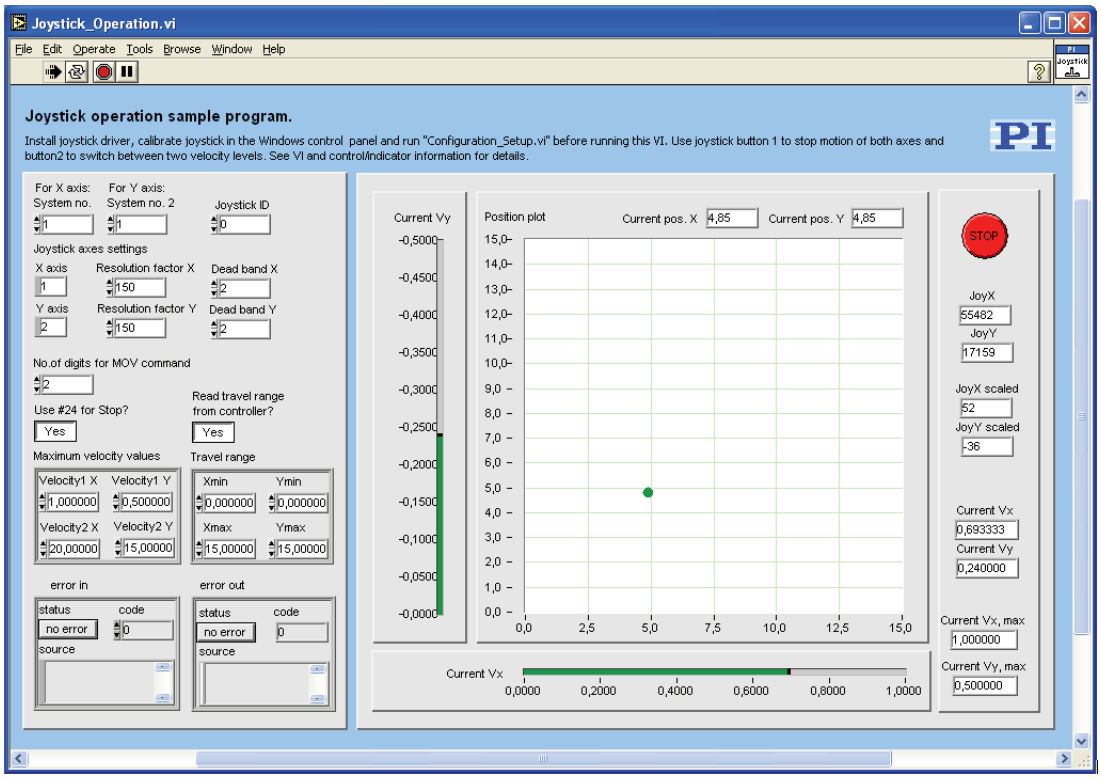
Resolution factor * determines the joystick resolution.

No. of digits is the number of digits after the decimal point in the position values that will be sent by the MOV command.

If joystick button 1 is pressed, a stop command (STP or #24, depending on Use 24 for stop, is sent to the controller. The diagram shows how to combine the driver and support VIs for these tasks.

If only one motion axis is to be controlled, Y Axis must be identical to X Axis, System no. 2 must be identical to System no. 1, and Y axis of joystick is to be used for control of the motion axis.

Important: Install joystick driver and calibrate joystick in the Windows Control Panel before running this VI.



Valid for C-702, C-843, C-843.PM, C-844, C-848, C-865, C-866, C-867, C-880, E-517, E-709, E-710, E-712, E-725, E-753, E-755, E-761,

E-861

Input

System number (1), System no. 2 (1), Joystick ID (0), X axis (empty string), Y axis (empty string), Resolution factor X (150), Resolution factor Y (150), Dead band X (2), Dead band Y (2), No. of digits (2), Use #24 for stop (TRUE), Read from controller (T), Maximum velocity values (2, 20, 0.5, 15), Travel range (0, 10, 0, 5), Error in (no error)

C-702: Use #24 for stop = TRUE, Read from controller = TRUE
Run “C702_Configuration_Setup.vi” prior to running this VI.

C-843: Use #24 for stop = TRUE, Read from controller = TRUE;
Run “C843_Configuration_Setup.vi” prior to running this VI.

C-843.PM: Use #24 for stop = TRUE, Read from controller = TRUE;
Run “C843_PM_Configuration_Setup.vi” prior to running this VI.

C-844: Use #24 for stop = TRUE, Read from controller = TRUE
Run “C844_Configuration_Setup.vi” prior to running this VI.

C-848: Use #24 for stop = TRUE, Read from controller = TRUE
Run “C848_Configuration_Setup.vi” prior to running this VI.

C-865: Use #24 for stop = TRUE, Read from controller = TRUE;
Run “C865_Configuration_Setup.vi” prior to running this VI.

C-866: Use #24 for stop = TRUE, Read from controller = TRUE;
Run “C866_Configuration_Setup.vi” prior to running this VI.

C-867: Use #24 for stop = TRUE, Read from controller = TRUE;
Run “C867_Configuration_Setup.vi” prior to running this VI.

C-880: Use #24 for stop = TRUE, Read from controller = TRUE
Run “C880_Configuration_Setup.vi” prior to running this VI.

E-517: Use #24 for stop = TRUE, Read from controller = TRUE.
Default position unit is µm, default velocity unit is µm/s.
Run “E517_Configuration_Setup.vi” prior to running this VI.

E-709: Use #24 for stop = TRUE, Read from controller = TRUE.
Default position unit is µm, default velocity unit is µm/s.
Run “E709_Configuration_Setup.vi” prior to running this VI.

E-710: Use #24 for stop = FALSE, Read from controller = TRUE.
Default position unit is µm, default velocity unit is µm/ms. Joystick button 1 is not supported (controller does not support #24 or STP).
Run “E710_Configuration_Setup.vi” prior to running this VI.

E-712: Use #24 for stop = TRUE, Read from controller = TRUE.
Default position unit is µm, default velocity unit is µm/s.
Run “E712_Configuration_Setup.vi” prior to running this VI.

E-725: Use #24 for stop = TRUE, Read from controller = TRUE.
Default position unit is µm, default velocity unit is µm/s.
Run “E725_Configuration_Setup.vi” prior to running this VI.

E-753: Use #24 for stop = TRUE, Read from controller = TRUE.
Default position unit is µm, default velocity unit is µm/s.
Run “E753_Configuration_Setup.vi” prior to running this VI.

E-755: Use #24 for stop = TRUE, Read from controller = TRUE.
Default position unit is µm, default velocity unit is µm/s.
Run “E755_Configuration_Setup.vi” prior to running this VI. VI does not work with E-755.101.

E-761: Use #24 for stop = TRUE, Read from controller = TRUE.
Default position unit is µm, default velocity unit is µm/ms.

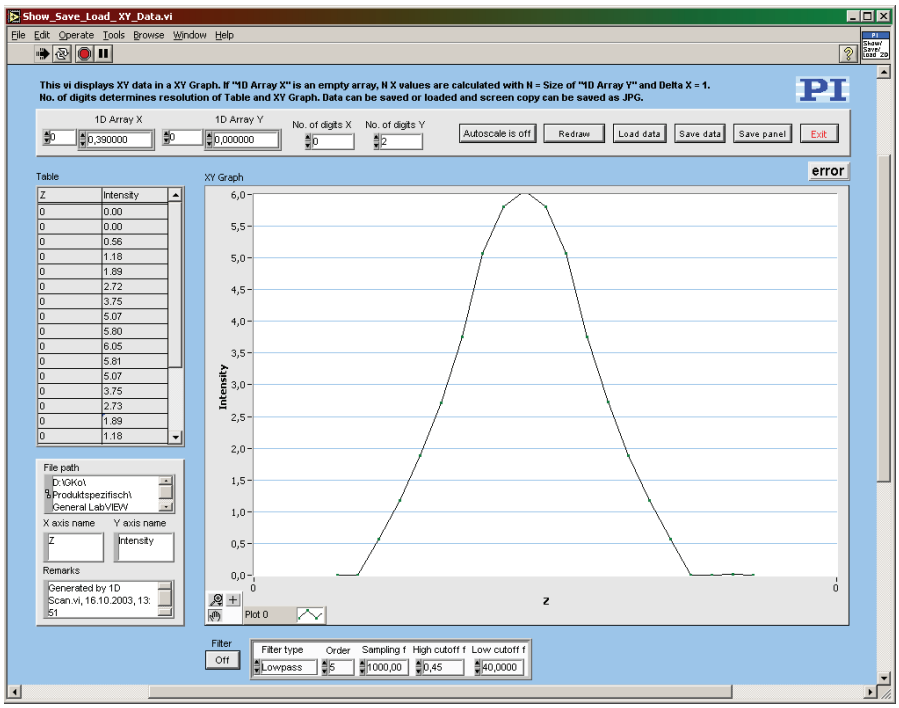
Run “E761_Configuration_Setup.vi” prior to running this VI.

E-861: Use #24 for stop = TRUE, Read from controller = TRUE;
Run “E861_Configuration_Setup.vi” prior to running this VI.

Use the *Help→Show Context Help* menu sequence in the LabVIEW environment to display the *Context Help* window with the VI and control/indicator descriptions.

3.10. Show_Save_Load_XY_Data.vi

This VI displays XY data in an XY Graph. If 1D Array X is an empty array, N X values are calculated with N = Size of 1D Array Y and Delta X = 1. No. of digits determines the resolution of Table and XY Graph. Data can be saved or loaded and a screen copy can be saved as JPG.



If data (1D Array X, 1D Array Y) are sent to the VI via the corresponding connectors, the VI will display the corresponding graphics after being called. To load data at runtime, press the Load data button. A dialog will pop up where a data file to open can be selected. The VI can read data in GCSArray, GCSTable and simple ASCII column format. Autoscale can be switched on or off. If Autoscale is off, the Y axis of the graph is scaled from 0-10.

Filter can be used to apply a filter to the current graph. For Filter = TRUE, a Lowpass, Highpass, Bandpass or Bandstop filter with appropriate settings can be selected.

Press Save data to save data (file header and numerical data). Data will be saved in GCS Array format. The file header will contain information given in X axis name, Y axis name and Remarks. With Save panel a screen copy of this VI can be saved as a JPG file. XY Graph will show the Y values over the corresponding X values. Table contains the numerical values for X and Y. Press Exit to stop execution of this VI.

Valid for	Analog systems, C-843, C-866, C-867, C-880, E-517, E-709, E-712, E-725, E-753, E-755, E-761, E-861, F-206, M-8X0. To support analog interfacing, VI must be present for E-816 also.
Input	1D Array X (empty num. array), 1D Array Y (empty num. array), 2D Array Z (empty 2D num. array), No. of digits X (, No. of digits Y, No. of digits Z, Autoscale, Error in (no error)
Output	Error out
Remarks	

4. PI Systems Currently Supported by This Driver Set

Product	works with LabVIEW driver version (or higher)	if product firmware/ drivers version is equal to or newer than
Analog	5.2.2	-
C-702	4.0.0	1.4.0
C-843	2.01 – 2.02	MC-DLL 1.0.2.2
	2.05 – 2.06	MC-DLL 1.0.2.3
	3.1.2., 3.1.2a	MC-DLL 1.0.2.3
	3.4.3	MC-DLL 1.0.2.8
	3.6.1	MC-DLL 1.0.2.8 GCS_DLL 1.3.1
	5.7.4	GCS_DLL 2.0.0
C-843.PM	3.1.0	MC-DLL 1.0.2.5
	3.4.3a	MC-DLL 1.0.2.5
	3.6.2	MC-DLL 1.0.2.5 GCS_DLL 1.3.0
C-848	3.0.2	1.0
C-865	3.3.0	MC_C865.dll 1.0
C-866	5.2.1	MC_C866.dll 1.0
C-867	5.6.0	1.1.0.0
	5.7.2	C-867: 2.0.0.0 C-867.160: 1.0.0.0
C-880	1.1	2.00
	1.2	2.10
	2.04	2.20
	2.05 – 2.06	2.21
	3.2.0	2.40
C-880K005	2.06	1.0
C-880K006	2.06	1.0
C-880K007	2.06	1.0
E-516	1.0 – 2.02	DSP V3.01, MCU V5
	2.05 – 2.06	DSP V3.11, MCU V5
	3.4.2	DSP V3.30, MCU V5
E-517	5.7.0	1.1.0
E-709	5.8.0	GCSTranslator.dll V1.6.0.1 or higher
E-710 3- & 4-channel versions	3.4.0	5.027
	3.4.4 (a, b)	5.0.33, 6.0.33
E-710 6-	3.4.4 (a, b)	2.13

channel		
E-712	5.3.1	1.0.1.0
E-725	5.5.0	02.00.03.00 For USB: E7XX_GCS2_DLL.dll V2.2.0
E-753	5.2.0	1.0.0
E-755	5.1.0	2.0.4.1 E7XX_GCS2_DLL.dll V1.1.0
E-761	3.5.0 5.4.1	1.0.0 2.0.1
E-861	5.4.0 5.8.1	6.0 7.0, GCSTranslator.dll V 1.6.0.2
E-816	2.01 – 2.06 5.3.0 5.7.1	2.02 2.1.1 3.2.0
F-206	1.1 – 2.06 5.4.2	Fhx0035 and higher Fhx0035 and higher, FHXF6 recommended
M-840	2.03 – 2.06 2.2.0 3.0.1 3.1.1 5.4.2	Hex0037 and higher Hex0037 and higher, Hex0045 recomm. Hex0037 and higher, Hex0050 recomm. Hex0037 and higher, Hex0051 recomm. Hex0037 and higher, HEXF6 recommended
M-850	2.03 – 2.06 3.0.1 3.1.1 5.4.2	Hex0040 and higher Hex0040 and higher, Hex0050 recomm. Hex0040 and higher, Hex0051 recomm. Hex0040 and higher, HEXF6 recommended
Mercury	3.6.0 5.7.3	1.0.6 PI_MERCURY_GCS_DLL.dll V 1.0.0.17 2.4.0 PI_MERCURY_GCS_DLL.dll V 2.0.0

5. Appendix A

Error codes are not unambiguous, but can result from a PI error message or LabVIEW internal error code. In addition to the list below see National Instruments error codes.

100	PI LabVIEW driver reports error. See source control for details.
0	No error
1	Parameter syntax error
2	Unknown command
3	Command length out of limits or command buffer overrun
4	Error while scanning
5	Unallowable move attempted on unreferenced axis, or move attempted with servo off
6	Parameter for SGA not valid
7	Position out of limits
8	Velocity out of limits
9	Attempt to set pivot point while U,V and W not all 0
10	Controller was stopped by command
11	Parameter for SST or for one of the embedded scan algorithms out of range
12	Invalid axis combination for fast scan
13	Parameter for NAV out of range
14	Invalid analog channel
15	Invalid axis identifier
16	Unknown stage name
17	Parameter out of range
18	Invalid macro name
19	Error while recording macro
20	Macro not found
21	Axis has no brake
22	Axis identifier specified more than once
23	Illegal axis
24	Incorrect number of parameters
25	Invalid floating point number
26	Parameter missing
27	Soft limit out of range
28	No manual pad found

29	No more step-response values
30	No step-response values recorded
31	Axis has no reference sensor
32	Axis has no limit switch
33	No relay card installed
34	Command not allowed for selected stage(s)
35	No digital input installed
36	No digital output configured
37	No more MCM responses
38	No MCM values recorded
39	Controller number invalid
40	No joystick configured
41	Invalid axis for electronic gearing, axis can not be slave
42	Position of slave axis is out of range
43	Slave axis cannot be commanded directly when electronic gearing is enabled
44	Calibration of joystick failed
45	Referencing failed
46	OPM (Optical Power Meter) missing
47	OPM (Optical Power Meter) not initialized or cannot be initialized
48	OPM (Optical Power Meter) Communication Error
49	Move to limit switch failed
50	Attempt to reference axis with referencing disabled
51	Selected axis is controlled by joystick
52	Controller detected communication error
53	MOV! motion still in progress
54	Unknown parameter
55	No commands were recorded with REP
56	Password invalid
57	Data Record Table does not exist
58	Source does not exist; number too low or too high
59	Source Record Table number too low or too high
60	Protected Param: current Command Level (CCL) too low
61	Command execution not possible while Autozero is running
62	Autozero requires at least one linear axis
63	Initialization still in progress
64	Parameter is read-only
65	Parameter not found in non-volatile memory
66	Voltage out of limits

67	Not enough memory available for requested wave curve
68	Not enough memory available for DDL table; DDL can not be started
69	Time delay larger than DDL table; DDL can not be started
70	The requested arrays have different lengths; query them separately
71	Attempt to restart the generator while it is running in single step mode
72	Motion commands and wave generator activation are not allowed when analog target is active
73	Motion commands are not allowed when wave generator output is active; use WGO to disable generator output
74	No sensor channel or no piezo channel connected to selected axis (sensor and piezo matrix)
75	Generator started (WGO) without having selected a wave table (WSL).
76	Interface buffer did overrun and command couldn't be received correctly
77	Data Record Table does not hold enough recorded data
78	Data Record Table is not configured for recording
79	Open-loop commands (SVA, SVR) are not allowed when servo is on
80	Hardware error affecting RAM
81	Not macro command
82	Macro counter out of range
83	Joystick is active
84	Motor is off
85	Macro-only command
86	Invalid joystick axis
87	Joystick unknown
88	Move without referenced stage
89	Command not allowed in current motion mode
90	No tracing possible while digital IOs are used on this HW revision. Reconnect to switch operation mode.
91	Move not possible, would cause collision
100	PI LabVIEW driver reports error. See source control for details.
200	No stage connected to axis
201	File with axis parameters not found
202	Invalid axis parameter file
203	Backup file with axis parameters not found
204	PI internal error code 204
205	SMO with servo on
206	uudecode: incomplete header
207	uudecode: nothing to decode
208	uudecode: illegal UUE format

209	CRC32 error
210	Illegal file name (must be 8-0 format)
211	File not found on controller
212	Error writing file on controller
213	VEL command not allowed in DTR Command Mode
214	Position calculations failed
215	The connection between controller and stage may be broken
216	The connected stage has driven into a limit switch, call CLR to resume operation
217	Strut test command failed because of an unexpected strut stop
218	While MOV! is running position can only be estimated!
219	Position was calculated during MOV motion
230	Invalid handle
231	No bios found
232	Save system configuration failed
233	Load system configuration failed
301	Send buffer overflow
302	Voltage out of limits
303	Open-loop motion attempted when servo ON
304	Received command is too long
305	Error while reading/writing EEPROM
306	Error on I2C bus
307	Timeout while receiving command
308	A lengthy operation has not finished in the expected time
309	Insufficient space to store macro
310	Configuration data has old version number
311	Invalid configuration data
333	Internal hardware error
400	Wave generator index error
401	Wave table not defined
402	Wave type not supported
403	Wave length exceeds limit
404	Wave parameter number error
405	Wave parameter out of range
406	WGO command bit not supported
555	BasMac: unknown controller error
601	Not enough memory
602	Hardware voltage error

603	Hardware temperature out of range
1000	Too many nested macros
1001	Macro already defined
1002	Macro recording not activated
1003	Invalid parameter for MAC
1004	Deleting macro failed
1005	Controller is busy with some lengthy operation (e.g. reference move, fast scan algorithm)
2000	Controller already has a serial number
4000	Sector erase failed
4001	Flash program failed
4002	Flash read failed
4003	HW match code missing/invalid
4004	FW match code missing/invalid
4005	HW version missing/invalid
4006	FW version missing/invalid
4007	FW update failed
0	No error occurred during function call
-1	Error during com operation (could not be specified)
-2	Error while sending data
-3	Error while receiving data
-4	Not connected (no port with given ID open)
-5	Buffer overflow
-6	Error while opening port
-7	Timeout error
-8	There are more lines waiting in buffer
-9	There is no interface or DLL handle with the given ID
-10	Event/message for notification could not be opened
-11	Function not supported by this interface type
-12	Error while sending "echoed" data
-13	IEEE488: System error
-14	IEEE488: Function requires GPIB board to be CIC
-15	IEEE488: Write function detected no listeners
-16	IEEE488: Interface board not addressed correctly
-17	IEEE488: Invalid argument to function call
-18	IEEE488: Function requires GPIB board to be SAC
-19	IEEE488: I/O operation aborted
-20	IEEE488: Interface board not found

-21	IEEE488: Error performing DMA
-22	IEEE488: I/O operation started before previous operation completed
-23	IEEE488: No capability for intended operation
-24	IEEE488: File system operation error
-25	IEEE488: Command error during device call
-26	IEEE488: Serial poll-status byte lost
-27	IEEE488: SRQ remains asserted
-28	IEEE488: Return buffer full
-29	IEEE488: Address or board locked
-30	RS-232: 5 data bits with 2 stop bits is an invalid combination, as is 6, 7, or 8 data bits with 1.5 stop bits
-31	RS-232: Error configuring the COM port
-32	Error dealing with internal system resources (events, threads, ...)
-33	A DLL or one of the required functions could not be loaded
-34	FTDIUSB: invalid handle
-35	FTDIUSB: device not found
-36	FTDIUSB: device not opened
-37	FTDIUSB: IO error
-38	FTDIUSB: insufficient resources
-39	FTDIUSB: invalid parameter
-40	FTDIUSB: invalid baud rate
-41	FTDIUSB: device not opened for erase
-42	FTDIUSB: device not opened for write
-43	FTDIUSB: failed to write device
-44	FTDIUSB: EEPROM read failed
-45	FTDIUSB: EEPROM write failed
-46	FTDIUSB: EEPROM erase failed
-47	FTDIUSB: EEPROM not present
-48	FTDIUSB: EEPROM not programmed
-49	FTDIUSB: invalid arguments
-50	FTDIUSB: not supported
-51	FTDIUSB: other error
-52	Error while opening the COM port: was already open
-53	Checksum error in received data from COM port
-54	Socket not ready, you should call the function again
-55	Port is used by another socket
-56	Socket not connected (or not valid)
-57	Connection terminated (by peer)

-58	Can't connect to peer
-59	Operation was interrupted by a nonblocked signal
-60	No Device with this ID is present
-61	Driver could not be opened (on Vista: run as administrator!)
-1001	Unknown axis identifier
-1002	Number for NAV out of range--must be in [1,10000]
-1003	Invalid value for SGA--must be one of 1, 10, 100, 1000
-1004	Controller sent unexpected response
-1005	No manual control pad installed, calls to SMA and related commands are not allowed
-1006	Invalid number for manual control pad knob
-1007	Axis not currently controlled by a manual control pad
-1008	Controller is busy with some lengthy operation (e.g. reference move, fast scan algorithm)
-1009	Internal error--could not start thread
-1010	Controller is (already) in macro mode--command not valid in macro mode
-1011	Controller not in macro mode--command not valid unless macro mode active
-1012	Could not open file to write or read macro
-1013	No macro with given name on controller, or macro is empty
-1014	Internal error in macro editor
-1015	One or more arguments given to function is invalid (empty string, index out of range, ...)
-1016	Axis identifier is already in use by a connected stage
-1017	Invalid axis identifier
-1018	Could not access array data in COM server
-1019	Range of array does not fit the number of parameters
-1020	Invalid parameter ID given to SPA or SPA?
-1021	Number for AVG out of range--must be >0
-1022	Incorrect number of samples given to WAV
-1023	Generation of wave failed
-1024	Motion error while axis in motion, call CLR to resume operation
-1025	Controller is (already) running a macro
-1026	Configuration of PZT stage or amplifier failed
-1027	Current settings are not valid for desired configuration
-1028	Unknown channel identifier
-1029	Error while reading/writing wave generator parameter file
-1030	Could not find description of wave form. Maybe WG.INI is missing?
-1031	The WGWaveEditor DLL function was not found at startup
-1032	The user cancelled a dialog

-1033	Error from C-844 Controller
-1034	DLL necessary to call function not loaded, or function not found in DLL
-1035	The open parameter file is protected and cannot be edited
-1036	There is no parameter file open
-1037	Selected stage does not exist
-1038	There is already a parameter file open. Close it before opening a new file
-1039	Could not open parameter file
-1040	The version of the connected controller is invalid
-1041	Parameter could not be set with SPA--parameter not defined for this controller!
-1042	The maximum number of wave definitions has been exceeded
-1043	The maximum number of wave generators has been exceeded
-1044	No wave defined for specified axis
-1045	Wave output to axis already stopped/started
-1046	Not all axes could be referenced
-1047	Could not find parameter set required by frequency relation
-1048	Command ID given to SPP or SPP? is not valid
-1049	A stage name given to CST is not unique
-1050	A uuencoded file transfered did not start with "begin" followed by the proper filename
-1051	Could not create/read file on host PC
-1052	Checksum error when transfering a file to/from the controller
-1053	The PiStages.dat database could not be found. This file is required to connect a stage with the CST command
-1054	No wave being output to specified axis
-1055	Invalid password
-1056	Error during communication with OPM (Optical Power Meter), maybe no OPM connected
-1057	WaveEditor: Error during wave creation, incorrect number of parameters
-1058	WaveEditor: Frequency out of range
-1059	WaveEditor: Error during wave creation, incorrect index for integer parameter
-1060	WaveEditor: Error during wave creation, incorrect index for floating point parameter
-1061	WaveEditor: Error during wave creation, could not calculate value
-1062	WaveEditor: Graph display component not installed
-1063	User Profile Mode: Command is not allowed, check for required preparatory commands
-1064	User Profile Mode: First target position in User Profile is too far from current position
-1065	Controller is (already) in User Profile Mode
-1066	User Profile Mode: Block or Data Set index out of allowed range

-1067	ProfileGenerator: No profile has been created yet
-1068	ProfileGenerator: Generated profile exceeds limits of one or both axes
-1069	ProfileGenerator: Unknown parameter ID in Set/Get Parameter command
-1070	ProfileGenerator: Parameter out of allowed range
-1071	User Profile Mode: Out of memory
-1072	User Profile Mode: Cluster is not assigned to this axis
-1073	Unknown cluster identifier
-1074	The installed device driver doesn't match the required version. Please see the documentation to determine the required device driver version.
-1075	The library used doesn't match the required version. Please see the documentation to determine the required library version.
-1076	The interface is currently locked by another function. Please try again later.

6. Index

End of document