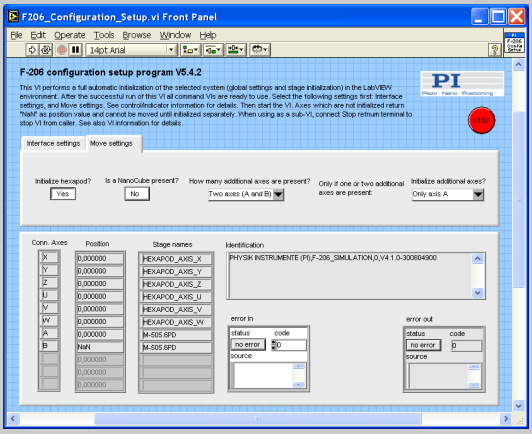
Piezo · Nano · Positioning

MS209E Software Manual

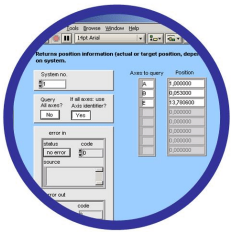
C-887 LabView Driver Library

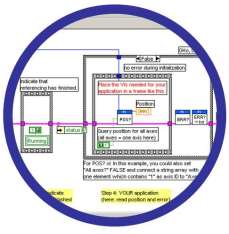
Release: 6.6.0 Date: 2014-07-25

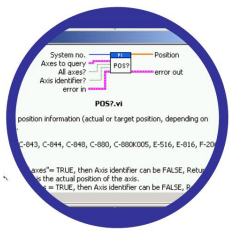


This document describes software for use with the following product(s):

- **C-887** Hexapod Controller, TCP/IP and RS-232 Interfaces
- **F-206** Six-Axis Parallel Kinematics Positioning System
- **M-810/811.xx** Hexapod Positioning System
- **M-824.xx** Hexapod Positioning System
- **M-840.xx** Hexapod Positioning System
- **M-850.xx** Hexapod Positioning System







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: 6.6.0
File:C887_GCS_LabVIEW_MS209E_660.doc, 3653632 Bytes

0. Disclaimer

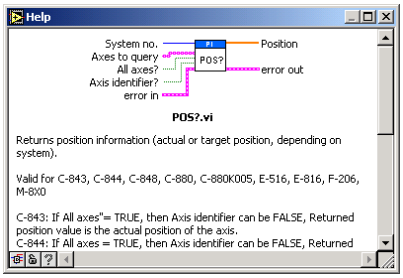
This software is provided “as is”. Physik Instrumente (PI) does not guarantee that this software is free of errors and will not be responsible for any damage arising from the use of this software. The user agrees to use this software on his own responsibility.

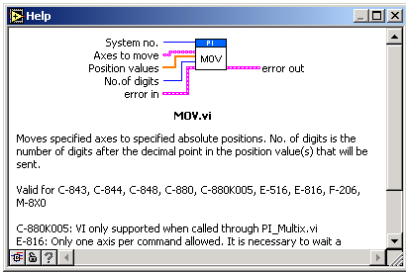
1. Introduction

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 8.6 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 which supports waveform generation 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-517 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. PIStages2.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 operating system environment and can be found by the GCS drivers (if LabVIEW still cannot find PIStages2.dat, it may be because it is marked read-only. To see, open an Explorer, right-click the file PIStages2.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 (see Section "First Steps for GCS-Compatible PI Controllers" on p. 8 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. 160 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 “C:\Users\Public\PI\PI_LabVIEW_Merge_Tool\MergeDrivers.llb” (also available via the Windows start menu), 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 run PIUpdateFinder.exe to 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. Software updates

The installation disk shipped with your product may contain outdated versions of software components or drivers.

To check for the latest versions, we recommend to use the PIUpdateFinder. If this software tool has not already been installed via the CD setup, follow the instructions on http://update.pi-portal.ws/ to download the guided installer of PIUpdateFinder.

The latest versions of software components or drivers are also available on http://www.pi-portal.ws/ in the section of your controller or in the “General Software” section. For log-in instructions, refer to the “X-XXX Releasenews_XXX.pdf” document in the “Manuals” directory of the installation disk.

1.6. First Steps for GCS-Compatible PI Controllers

1.6.1. C-887 (GCS 2.0 set for use with H-206, H-8X0)

To keep compatibility with former versions of this driver set, Hexapod systems are named "F-206" instead of "C-887 + H-206" and "M-8X0" instead of "C-887 + H-8X0" in this driver set.

Step 1: To configure stages connected to the separate axes A or B of the C-887 controller use PIMikroMove and save selection to the controller. For older F-206 or M-8X0 controller versions, please see the controller user manual to find out how to configure stages connected to axes A or B.

Step 2 (advanced users can skip this step): To check communication between the Hexapod controller and the host PC, run "F206_Simple_Test.vi" (for H-206) or "M8X0_Simple_Test.vi" (for H-810, H-811, H-824, H-840, H-850 etc.), depending on the mechanics connected to the C-887 controller. This VI will return the ID and help strings of the Hexapod controller and the axis IDs and stage names of the connected axes (according to your selection of Is a NanoCube present? and How many additional axes are present?). You can drive up to two additional separate, motor-driven axes (PWM-compatible motors with position control) with the C-887 controller (see also the C-887 User Manual). If you have ordered the NCU option, you can drive a 3-axis piezo stage ("NanoCube") with the C-887 controller. Before you proceed with step 3, please check that the current configuration matches your stage connections. 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:

WARNING: F206_Configuration_Setup.vi and M8X0_Configuration_Setup.vi May Cause Move

When you start "F206/M8X0_Configuration_Setup.vi" with Initialize hexapod? = TRUE and/or Initialize additional axes? = TRUE, the VI will automatically move the Hexapod and/or the additional axes to their reference point switches and the NanoCube (if present) to its middle position. It is therefore important to make sure that items connected to or mounted on connected stages cannot be damaged by such a move.

To control one or more Hexapod systems with this driver set, run "F206_Configuration_Setup.vi" (for C-887 + H-206) or "M8X0_Configuration_Setup.vi" (for C-887 + H-810/811/824/840/850 etc.). This VI performs all steps necessary for a full configuration of the driver VIs in the LabVIEW environment: the definition of axis IDs, the initialization of the connected stages 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 "F206/M8X0_Configuration_Setup.vi"; otherwise all global settings will be lost and the driver VIs will not work. When programming your application, you can implement "F206/M8X0_Configuration_Setup.vi" as an initialization VI in your software. See chapter 3 for a detailed description of "F206/M8X0_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.

The axis identifiers of the Hexapod (X, Y, Z, U, V, W), NanoCube (K, L, M, if present) and additional axes (A, B, if any) cannot be changed.

GCS syntax version: 2.0

1.6.2. F-206

For Hexapod systems consisting of C-887 and H-206 (GCS syntax version 2.0 or higher) please refer to “C-887”.

This driver set (PI General LabVIEW Driver Set) and the F-206 LabVIEW driver set from former releases (non-GCS, not supported anymore) are fully compatible and can be used in parallel. The F-206 can be fully controlled with the PI General LabVIEW Driver Set. The axis identifiers of the F-206 (X, Y, Z, U, V, W), NanoCube (K, L, M, if present) and additional axes (A, B, if any) cannot be changed.

Step 1 (advanced users can skip this step): To check communication between the F-206 controller and the host PC, run “F206_Simple_Test.vi”. This VI will return the ID and help strings of the F-206 controller and the axis IDs and stage names of the connected axes (according to your selection of Is a NanoCube present? and How many additional axes are present?). If you have ordered the AC8 option, you can drive up to two additional separate, motor-driven axes (PWM-compatible motors with position control) with the F-206 controller (see also the F-206 User Manual). If you have ordered the NCU option, you can drive a 3-axis piezo stage (“NanoCube”) with the F-206 controller. Before you proceed with step 2, please check that the current configuration matches your stage connections. 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 2:

WARNING: F206_Configuration_Setup.vi May Cause Move

When you start “F206_Configuration_Setup.vi” with Initialize hexapod? = TRUE and/or Initialize additional axes? = TRUE, the VI will automatically move the Hexapod and/or the additional axes to their reference point switches and the NanoCube (if present) to its middle position. It is therefore important to make sure that items connected to or mounted on connected stages cannot be damaged by such a move.

Depending on the firmware version on the controller, it may not be possible to stop motion initiated by INI or fast scanning commands with STOP, #24 or #27.

To control one or more F-206 controllers with this driver set, run “F206_Configuration_Setup.vi”. This VI performs all steps necessary for a full configuration of the driver VIs in the LabVIEW environment: the definition of axis IDs, the initialization of the connected stages 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 “F206_Configuration_Setup.vi”; otherwise all global settings will be lost and the driver VIs will not work. When programming your application, you can implement “F206_Configuration_Setup.vi” as an initialization VI in your software. See chapter 3 for a detailed description of “F206_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.

If the controller is equipped with an F-361 optical power meter board and properly configured, for all VIs displaying analog input values an optional, external F-361 optical power meter (OPM) will be used instead of the optical board mentioned. [A] indicates which Optical Board or which OPM (F-361) analog input to use. Can be A1 or A2. If omitted, A1 is used. If there is an F-361 OPM configuration file

(C:\HEXAPOD\F-361.DAT in the controller file system), the OPM will be addressed and any optical boards present will not be accessible, otherwise the specified optical board will be addressed.

GCS syntax version: 1.0

1.6.3. M-8X0 (M-810 / M-811 / M-824 / M-840 / M-850)

For Hexapod systems consisting of C-887 and H-810/811/824/840/850 etc. (GCS syntax version 2.0 or higher) please refer to “C-887”.

This driver set (PI General LabVIEW Driver Set) and the M-840 / M-850 LabVIEW driver set from former releases (non-GCS, not supported anymore) are fully compatible and can be used in parallel. The M-810 / M-811 / M-824 / M-840 / M-850 can be fully controlled with the PI General LabVIEW Driver Set and is called “M-8X0” from here on. The axis identifiers of the M-8X0 and additional axes (if any) cannot be changed.

Step 1 (advanced users can skip this step): To check communication between the M-8X0 controller and the host PC, run “M8X0_Simple_Test.vi”. This VI will return the ID and help strings of the M-8X0 controller and the axis IDs and stage names of the connected axes (according to your selection of How many additional axes? are connected to the M-8X0 controller). If you have ordered the AC8 option, you can drive up to two additional separate, motor-driven axes (PWM-compatible motors with position control) with the M-8X0 controller (see also the M-8X0 User Manual). Before you proceed with step 2, please check that the current configuration matches your stage connections. 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 2:

WARNING: M8X0_Configuration_Setup.vi May Cause Move

When you start “M_8X0_Configuration_Setup.vi” with Initialize hexapod? = TRUE and/or Initialize additional axes? = TRUE, the VI will automatically move the Hexapod and/or the additional axes to their reference point switches. It is therefore important to make sure that items connected to or mounted on connected stages cannot be damaged by such a move.

Depending on the firmware version on the controller, it may not be possible to stop motion initiated by INI or FSN with STOP, #24 or #27.

To control one or more M-8X0 controllers with this driver set, run “M8X0_Configuration_Setup.vi”. This VI performs all steps necessary for a full configuration of the driver VIs in the LabVIEW environment: the definition of axis IDs, the initialization of the connected stages 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 “M8X0_Configuration_Setup.vi”; otherwise all global settings will be lost and the driver VIs will not work. When programming your application, you can implement “M8X0_Configuration_Setup.vi” as an initialization VI in your software. See chapter 3 for a detailed description of “M8X0_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.

Firmware versions HEX98-22 (10 Dec 1999) and older are not supported by this driver set.

If the controller is equipped with an F-361 optical power meter board and properly configured, for all VIs displaying analog input values an optional, external F-361 optical power meter (OPM) will be used instead of the optical board mentioned. [A] indicates which Optical Board or which OPM (F-361) analog input to use. Can be A1 or A2. If omitted, A1 is used. If there is an F-361 OPM configuration file (C:\HEXAPOD\F-361.DAT in the controller file system), the OPM will be addressed

and any optical boards present will not be accessible, otherwise the specified optical board will be addressed.

GCS syntax version: 1.0

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 as a Dummy VI for all other systems also)
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	Calls Analog Functions (dyn).vi functions dynamically during runtime, depending on String to send.

2.1.3. 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

2.1.4. 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.

2.1.5. 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.

2.1.6. 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-413, C-843, C-843.PM, C-865, C-866, C-867, C-884, E-517, E-709, E-710, E-712, E-725, E-755, E-761, E-816, E-861, E-870, E-871, Hydra, Pollux, Mercury, Mercury_GCS (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-413, C-843, C-843.PM, C-865, C-866, C-867, C-884, E-517, E-709, E-710, E-712, E-725, E-755, E-761, E-816, E-861, E-870, E-871, Hydra, Pollux, Mercury, Mercury_GCS (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. ECO?.vi (Communication.llb)

Valid for	C-887, F-206, M-8X0
Input	System number (1), Send (empty string), Error in (no error)
Output	Reply, Error out
Remarks	Returns echo string. Reply should be equal to Send string. C-887, F-206, M-8X0: Check HLP?/HELP answer to find out if ECO? is supported.

2.2.6. Find baudrate.vi (Communication.llb)

Valid for	C-413, C-702, C848, C-867, C-880, C-880K005, C-884, C-887, E-516, E-517, E-709, E-712, E-725, E-755, E-816, E-861, E-871, F-206, M-8X0, Mercury_GCS
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-413: Input and output HW handshake must be TRUE. All other controls=default. 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. C-884: Input and output HW handshake must be FALSE. All other 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. E-871: Input and output HW handshake must be FALSE. All other controls=default. C-887, F-206, M-8X0:: All controls=default. Mercury_GCS: Input and output HW handshake must be FALSE. All other 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, E-871, C-867, Mercury_GCS: The baudrate is set via the DIP switches on the controller front panel. See the controller User manual for details.

C-884: The baudrate is set via IFC/IFS.vi. See controller User manual for details.

2.2.7. Find host address.vi (Communication.llb)

Valid for	C-702, C-884, C-887, E-517, E-712, E-725, E-753, F-206, M-8X0
Input	Port (50000), Controller names (empty string array), Mode (Find controller by UDP), Error in (no error) C-702: Controller names = C-702, Mode = Find controller by UDP, Port = 50000 C-884: Controller names = C-884, Mode = Find controller by UDP, Port = 50000 E-517: Controller names = E-517, Mode = Find controller by UDP, Port = 50000 E-712: Controller names = E-712, Mode = Find controller by UDP, Port = 50000 E-725: Controller names = E-725, Mode = Find XPort by UDP, Port = 30718 E-753: Controller names = E-753, Mode = Find controller by UDP, Port = 50000 F-206: Controller names = F-206, F-HEX, Mode = Find XPort by UDP, Port = 30718 C-887, M-8X0: Controller names = M-8X0, HEXAPOD, Mode = Find XPort by UDP, Port = 30718
Output	All addresses, All IDs, Address, ID, Error out
Remarks	Performs an UDP broadcast and returns IP addresses of all controllers matching Controller name.

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

Valid for	C-413, C-843, C-843.PM, C-844, C-865, C-866, C-867, C-884, E-517, E-709, E-710, E-712, E-725, E-755, E-761, E-816, E-861, E-870, E-871, Hydra, Pollux, Mercury, Mercury_GCS (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.

2.2.9. Get subnet.vi (Communication.llb)

Valid for	C-702, C-884, C-887, E-517, E-712, E-725, E-753, F-206, M-8X0 (but must be present for all other systems except Analog systems, too)
Input	None
Output	Subnet
Remarks	Calls IPCONFIG and returns subnet broadcast addresses of all installed network cards.

2.2.10. Global DaisyChain.vi (Communication.llb)

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

2.2.11. Global1.vi (Communication.llb)

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

2.2.12. IFC?.vi (Communication.llb)

Valid for	C-702, C-884, C-887, E-517, E-709, E-712, E-725, E-753, F-206, M-8X0, Hydra
Input	System number (1), Interface parameter (Empty string array), All parameters? (F), Error in (no error)
Output	Parameter value, Error out
Remarks	Returns the current interface configuration. E-517, E-753: Note that when the controller is part of a network with DHCP, the static IP address of the controller is returned, not the currently used IP address, which was obtained from the DHCP server. C-887, F-206, M-8X0: Check HLP?/HELP answer to find out if IFC? is supported. Only for GCS syntax version = GCS 2.0 or higher (Check with CSV?.vi. If CSV?.vi is not supported, syntax version is GCS 1.0).

2.2.13. IFS.vi (Communication.llb)

Valid for	C-702, C-884, C-887, E-517, E-709, E-712, E-725, E-753, F-206, M-8X0
Input	System number (1), Password (100), Interface parameter (Empty string array), Parameter value (Empty string array), Error in (no error) E-709: Interface parameter can only be RSBAUD.
Output	Error out
Remarks	If Password is correct, the default parameter(s) for the interface are changed, but the current active parameters are not changed. Settings made with IFS are saved to EPROM and become active with the next startup/reboot. To change settings immediately (but temporarily) use IFC instead (if supported by your controller). C-887, F-206, M-8X0: Check HLP?/HELP answer to find out if IFS is supported. Only for GCS syntax version = GCS 2.0 or higher (Check with CSV?.vi. If CSV?.vi is not supported, syntax version is GCS 1.0).

2.2.14. IFS?.vi (Communication.llb)

Valid for	C-702, C-884, C-887, E-517, E-709, E-712, E-725, E-753, F-206, Hydra, M-8X0
Input	System number (1), Interface parameter (Empty string array), All

	parameters? (F), Error in (no error)
Output	Parameter value, Error out
Remarks	Returns the default Interface configuration which is stored in EPROM. C-887, F-206, M-8X0: Check HLP?/HELP answer to find out if IFS? is supported. Only for GCS syntax version = GCS 2.0 or higher (Check with CSV?.vi. If CSV?.vi is not supported, syntax version is GCS 1.0).

2.2.15. 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.

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

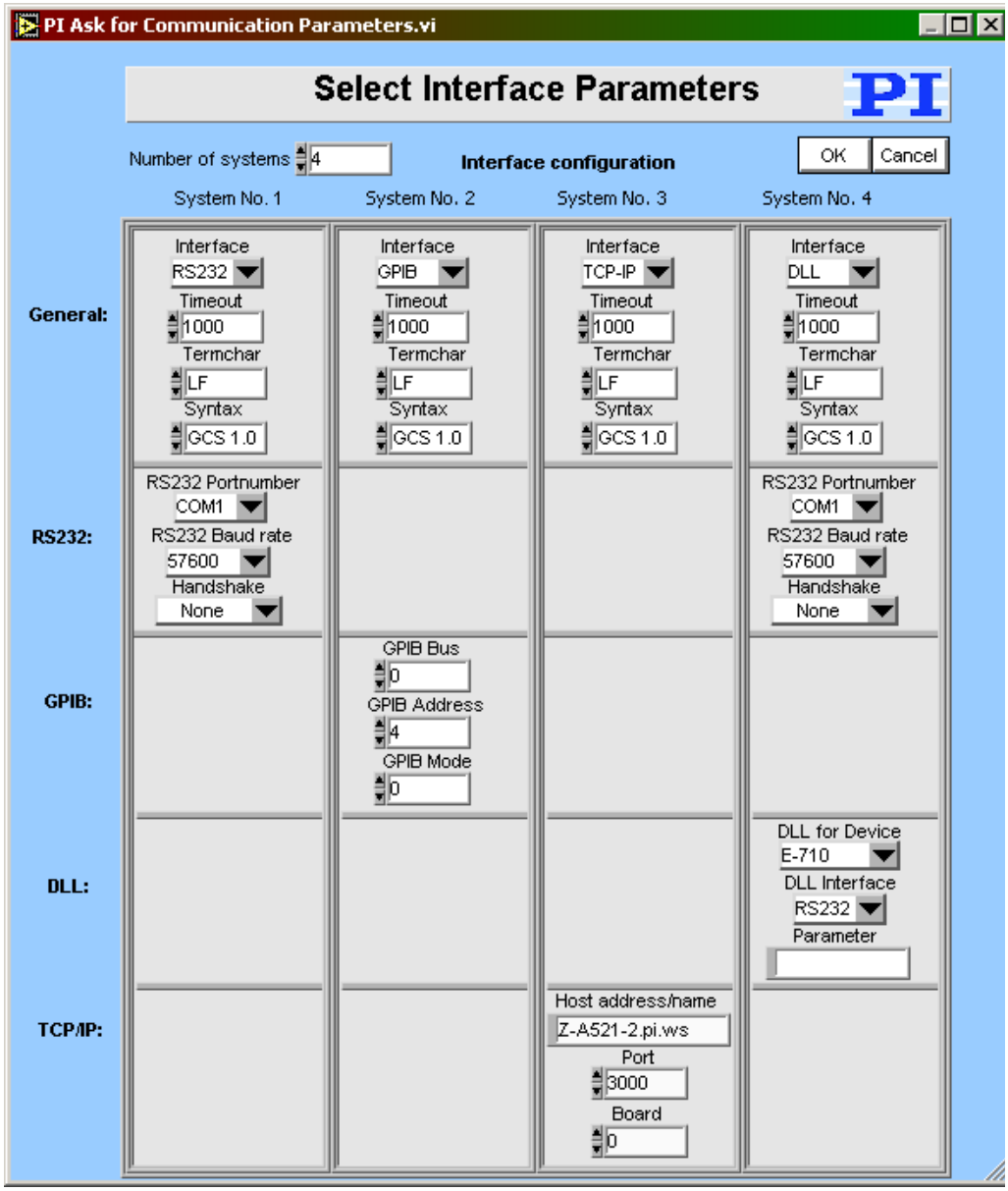
Valid for	C-867, E-709, E-755, E-861, E-871, Mercury_GCS (but must be present for all other systems except Analog systems, too)
Input	System number (1), Error in (no error)
Output	Error out
Remarks	This VI initializes Global DaisyChain according to the given system no. E-709: Only supported if E-709 is used inside C-867K012/K013.

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

Valid for	C-867, E-709, E-755, E-861, E-871, Mercury_GCS (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. E-709: Only supported if E-709 is used inside C-867K012/K013.

2.2.18. 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.



2.2.19. 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-413: Interface = RS232 or DLL, RS232: Input and output HW handshake must be TRUE. DLL: DLL for Device = PI_GCS2_DLL, DLL Interface = USB, Parameter = Serial no. of system to connect to.
Syntax: GCS 2.0; Term char = LF.

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.

C-884: Interface = RS232, TCP/IP or DLL, RS232: Input and output HW handshake must be FALSE. DLL (USB): DLL for Device = PI_GCS2_DLL, DLL Interface = USB, Parameter = Serial no. of system to connect to.
Syntax: GCS 2.0; 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.

E-870: Interface = DLL, DLL (USB): DLL for Device = PI_GCS2_DLL, DLL Interface = USB, Parameter = Serial no. of system to connect to. Syntax: GCS 2.0; Term char = LF.

E-871: Single Device: Interface = RS232 or DLL, RS232: Input and output HW handshake must be FALSE. DLL (USB): DLL for Device = PI_GCS2_DLL, DLL Interface = USB, Parameter = Serial no. of system to connect to.
DaisyChain: Interface = DLL, DLL for Device = PI_GCS2_DLL, 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: F-206 (GCS 1.0): 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.
C-887 + H-206 (GCS 2.0): Interface = RS232 or TCP/IP, RS232: Input and output handshake settings must be FALSE, Syntax: GCS 2.0; Term char = LF.

C-887 + H-206 (GCS 2.0): Interface = RS232 or TCP/IP, RS232: Input and output handshake settings must be FALSE, Syntax: GCS 2.0; Term char = LF.

Hydra: Interface = DLL, DLL (TCP/IP and RS-232): DLL for Device = PI_HydraPollux_GCS2_DLL, DLL Interface = RS232 or TCP/IP, Syntax: GCS 2.0; Term char = LF.

M-8X0: M-810/11/24/40/50 (GCS 1.0): Interface = RS232, GPIB or TCP/IP, RS232: Input and output handshake settings must be FALSE, Syntax: GCS 1.0; Term char = LF.
C-887 + H-810/11/24/40/50 (GCS 2.0): Interface = RS232 or TCP/IP,

RS232: Input and output handshake settings must be FALSE, Syntax: GCS 2.0; Term char = LF.

C-887 + H-810/11/24/40/50 (GCS 2.0): Interface = RS232 or TCP/IP, RS232: Input and output handshake settings must be FALSE, Syntax: GCS 2.0; 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.

Mercury_GCS: Single Device: Interface = RS232 or DLL, RS232: Input and output HW handshake must be FALSE. DLL (USB): DLL for Device = PI_GCS2_DLL, DLL Interface = USB, Parameter = Serial no. of system to connect to.
DaisyChain: Interface = DLL, DLL for Device = PI_GCS2_DLL, DLL Interface = RS232_DC or USB_DC, Parameter = Number of device in chain, Register DC: FALSE. Syntax: GCS 2.0; Term char = LF.

Pollux: Interface = DLL, DLL (RS-232): DLL for Device = PI_HydraPollux_GCS2_DLL, DLL Interface = RS232, Baudrate must be 19200. Syntax: GCS 2.0; Term char = LF.

2.2.20. 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.

2.2.21. 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.

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

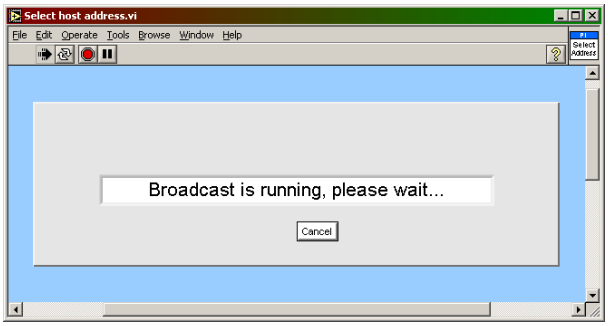
Valid for	All systems
Input	System number (1), String to send (empty string), Attach termination char.? (T), Error in (no error)
Output	Error out
Remarks	Sends command with or without trailing termination character to selected system.

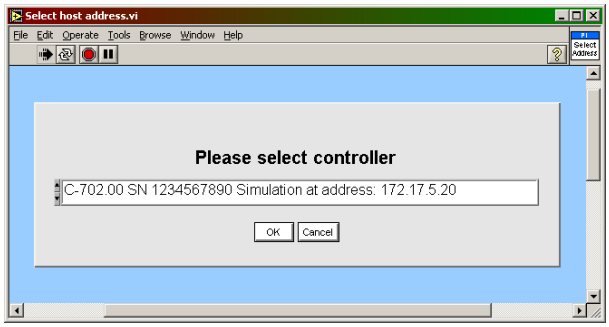
2.2.23. PI VISA Receive Characters.vi (Communication.llb)

Valid for	C-413, C-702, C-848, C-867, C-880, C-880K005, C-884, C-887, E-516, E-517, E-712, E-725, E-753, E-709, E-816, E-861, E-871, F-206, M-8X0 , Mercury_GCS (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".

2.2.24. Select host address.vi (Communication.llb)

Valid for	C-702, C-884, C-887, E-517, E-712, E-725, E-753, F-206, M-8X0
Input	Port (50000), Controller names (empty string array), Mode (Find controller by UDP), Local Stop (FALSE), Error in (no error) C-702: Controller names = C-702, Mode = Find controller by UDP, Port = 50000 C-884: Controller names = C-884, Mode = Find controller by UDP, Port = 50000 E-517: Controller names = E-517, Mode = Find controller by UDP, Port = 50000 E-712: Controller names = E-712, Mode = Find controller by UDP, Port = 50000 E-725: Controller names = E-725, Mode = Find XPort by UDP, Port = 30718 E-753: Controller names = E-753, Mode = Find controller by UDP, Port = 50000 F-206: Controller names = F-206, F-HEX, Mode = Find XPort by UDP, Port = 30718 C-887, M-8X0: Controller names = M-8X0, HEXAPOD, Mode = Find XPort by UDP, Port = 30718
Output	Selected Host address/name, Error out
Remarks	Performs an UDP broadcast, returns IP addresses and 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.





2.2.25. 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.

2.2.26. Syntax.ctl (Communication.llb)

Valid for	All systems
Input	None
Output	None
Remarks	Type definition for GCS version.

2.2.27. Termination character.ctl (Communication.llb)

Valid for	All systems
Input	None
Output	None
Remarks	Type definition for termination character.

2.3. Controller algorithms ("Controller Algorithms.llb")

WARNING

F-206, M-8X0: Depending on the firmware version on the controller, motion initiated by fast scanning commands may not be able to be stopped.

2.3.1. AAP.vi (Controller Algorithms.llb)

Valid for	C-887, F-206, M-8X0
Input	System number (1), Board (1), Axis 1 to scan (empty string), Axis 2 to scan (empty string), Range axis 1 (0.1), Range axis 2 (0.1), Step size (0.05), Repeat pos. (3), Use #7 polling (F), Timeout (s) (60), Refnum stop (F), Local stop (F), Error in (no error) C-887, F-206, M-8X0: Use #7 polling = FALSE, Range axis 2 must be identical with Range axis 1 (no rectangular scan implemented)
Output	Scan successful? (T/F), Error out
Remarks	Performs a fast automated alignment in specified axes, waits until scan is finished (using #7 polling if checked) and indicates whether scan was successful or not. Timeout is only valid if Use #7 polling is FALSE. If Stop refnum or Local stop is TRUE, VI sends #24 and stops. Repeat pos. = 0 turns continuous tracking on, in this case Timeout must be larger than the expected tracking duration and tracking must be stopped manually by using Local stop or Refnum stop. When using as a sub-VI, use Refnum stop to stop VI from caller.

C-887, F-206, M-8X0: For GCS syntax version = GCS 2.0 or higher (Check with CSV?.vi. If CSV?.vi is not supported, syntax version is GCS 1.0), VI calls FSS? automatically to determine if scan was successful or not.

2.3.2. FAA.vi (Controller Algorithms.llb)

Valid for C-887, F-206

Input System number (1), Board (1), Axis to scan (empty string), Range (0.1), Treshold level (0.1), Decrease VEL for scan? (T), Velocity (0.2), Use #7 polling (F), Timeout (s) (300), Stop refnum (F), Local stop (F), Error in (no error)

C-887, F-206: Use #7 polling = FALSE.

Output Scan successful? (T/F), Error out

Remarks Performs a fast angular line scan to maximum in specified axis, waits until scan is ready (using #7 polling if checked) and indicates whether scan was successful or not. Scan starts at (Current position - ½ Range) and stops at (Current position + ½ Range). Velocity is only valid if Decrease VEL for scan? Is TRUE. If Decrease VEL for scan? is TRUE, velocity is decreased before and reset after the scan. Timeout is only valid if Use #7 polling is FALSE. If Stop refnum or Local stop is TRUE, VI resets velocity if it was decreased before, sends #24 and stops. When using as a sub-VI, use Refnum stop to stop VI from caller.

C-887, F-206: Check HLP?/HELP answer to find out if FAA is supported. Only for GCS syntax version = GCS 1.0 (Check with CSV?.vi. If CSV?.vi is not supported, syntax version is GCS 1.0). For compatibility reasons, VI calls FLM automatically for GCS 2.0.

2.3.3. FAM.vi (Controller Algorithms.llb)

Valid for C-887, F-206

Input System number (1), Board (1), Axis 1 to scan (empty string), Axis 2 to scan (empty string), Area size (0.2), Treshold level (0.1), Distance between scan lines (0.02), Use #7 polling (F), Timeout (s) (300), Refnum stop (F), Local stop (F), Error in (no error)

C-887, F-206: Use #7 polling = FALSE

Output Scan successful? (T/F), Error out

Remarks Performs a fast angular scan to maximum in specified axes, waits until scan is finished using #7 polling (if checked) and indicates whether scan was successful or not. Timeout is only valid if Use #7 polling is FALSE. If Refnum stop or Local stop is TRUE, VI sends #24 and stops. When using as a sub-VI, use Refnum stop to stop VI from caller.

C-887, F-206: Check HLP?/HELP answer to find out if FAM is supported. Only for GCS syntax version = GCS 1.0 (Check with CSV?.vi. If CSV?.vi is not supported, syntax version is GCS 1.0). For compatibility reasons, VI calls FSM automatically for GCS 2.0.

2.3.4. FAS.vi (Controller Algorithms.llb)

Valid for C-887, F-206

Input System number (1), Board (1), Axis 1 to scan (empty string), Axis 2 to scan (empty string), Area size (0.2), Treshold level (0.1), Distance between scan lines (0.02), Use #7 polling (F), Timeout (s) (300), Refnum stop (F), Local

stop (F), Error in (no error)

C-887, F-206: Use #7 polling = FALSE

Output: Scan successful? (T/F), Error out

Remarks: Performs a fast angular scan in specified axes, waits until scan is finished using #7 polling (if checked) and indicates whether scan was successful or not. Timeout is only valid if Use #7 polling is FALSE. If Refnum stop or Local stop is TRUE, VI sends #24 and stops. When using as a sub-VI, use Refnum stop to stop VI from caller.

C-887, F-206: Check HLP?/HELP answer to find out if FAS is supported. Only for GCS syntax version = GCS 1.0 (Check with CSV?.vi. If CSV?.vi is not supported, syntax version is GCS 1.0). For compatibility reasons, VI calls FSC automatically for GCS 2.0.

2.3.5. FIO.vi (Controller Algorithms.llb)

Valid for: C-887, F-206, M-8X0

Input: System number (1), Board (1), Axis 1 to scan (empty string), Axis 2 to scan (empty string), Max. area size (0.2), Max. area size 2 (0.2), Treshold level (0.1), Angular area, deg (0.2), Step size linear spiral (0.01), Use #7 polling (F), Timeout (s) (300), Refnum stop (F), Local stop (F), Error in (no error)

C-887, F-206, M-8X0: Use #7 polling = FALSE. Max. area size 2 must be identical to Max. area size.

Output: Scan successful? (T/F), Error out

Remarks: Performs a fast input/output automated alignment procedure in specified axes, waits until scan is ready using #7 polling (if checked) and indicates whether scan was successful or not. Timeout is only valid if Use #7 polling is FALSE. If Refnum stop or Local stop is TRUE, VI sends #24 and stops. When using as a sub-VI, use Refnum stop to stop VI from caller.

C-887, F-206, M-8X0: Check HLP?/HELP answer to find out if FIO is supported. For GCS syntax version = GCS 2.0 or higher (Check with CSV?.vi. If CSV?.vi is not supported, syntax version is GCS 1.0), VI calls FSS? automatically to determine if scan was successful or not.

2.3.6. FLM.vi (Controller Algorithms.llb)

Valid for: C-880, C-887, F-206, M-8X0

Input: System number (1), Board (1), Axis to scan (empty string), Range (0.1), Treshold level (0.1), Step size (0.001), Decrease VEL for scan? (F), Velocity (0.2), Use #7 polling? (T), Timeout, s (300), Scan direction (Scan only right, +), Refnum stop (F), Local stop (F), Error in (no error)

C-880: Use #7 polling? = TRUE. Scan direction is not valid.

C-887, F-206, M-8X0: Use #7 polling? = FALSE. Check HLP?/HELP answer to find out if FLM is supported. Only for GCS syntax version = GCS 2.0 or higher (Check with CSV?.vi. If CSV?.vi is not supported, syntax version is GCS 1.0). Step size is not valid.

Output: Scan successful? (T/F), Controller Error, Error out

Remarks: Performs a line scan to maximum in specified axis, waits until scan is ready (using #7 polling if checked) and indicates whether scan was successful or not. Velocity is only valid if Decrease VEL for scan? is TRUE. If Decrease VEL for scan? is TRUE, velocity is decreased before and reset after the scan. Timeout is only valid if Use #7 polling is FALSE. If Refnum stop or

Local stop is TRUE, VI sends #24 and stops. When using as a sub-VI, use Refnum stop to stop VI from caller.

2.3.7. FLS.vi (Controller Algorithms.llb)

Valid for	C-880, C-887, F-206, M-8X0
Input	System number (1), Board (1), Axis to scan (empty string), Range (0.1), Treshold level (0.1), Step size (0.001), Decrease VEL for scan? (F), Velocity (0.2), Use #7 polling? (T), Timeout, s (300), Scan direction (Scan only right, +), Refnum stop (F), Local stop (F), Error in (no error) C-880: Use #7 polling? = TRUE. Scan direction is not valid. C-887, F-206, M-8X0: Use #7 polling? = FALSE. Check HLP?/HELP answer to find out if FLS is supported. Only for GCS syntax version = GCS 2.0 or higher (Check with CSV?.vi. If CSV?.vi is not supported, syntax version is GCS 1.0). Step size is not valid.
Output	Scan successful? (T/F), Controller Error, Error out
Remarks	Performs a line scan in specified axis, waits until scan is ready (using #7 polling if checked) and indicates whether scan was successful or not. Velocity is only valid if Decrease VEL for scan? is TRUE. If Decrease VEL for scan? is TRUE, velocity is decreased before and reset after the scan. Timeout is only valid if Use #7 polling is FALSE. If Refnum stop or Local stop is TRUE, VI sends #24 and stops. When using as a sub-VI, use Refnum stop to stop VI from caller.

2.3.8. FSA.vi (Controller Algorithms.llb)

Valid for	C-880, C-887, F-206, M-8X0
Input	System number (1), Board (1), Axis 1 to scan (empty string), Axis 2 to scan (empty string), Range axis 1 (0.1), Treshold level (0.1), Step size (0.05), AA step size (0.001), Range axis 2 (0.1), Use #7 polling (F), Timeout (s) (300), Refnum stop (F), Local stop (F), Error in (no error) C-880: Use #7 polling = TRUE C-887, F-206, M-8X0: Use #7 polling = FALSE, Range axis 2 must be identical with Range axis 1 (no rectangular scan implemented)
Output	Scan successful? (T/F), Error out
Remarks	Performs a 2D scan and align in specified axes, waits until scan is ready (using #7 polling if checked) and indicates whether scan was successful or not. For a square scan, Range axis 2 must be identical to Range axis 1, otherwise a rectangular scan is performed. Timeout is only valid if Use #7 polling is FALSE. If Refnum stop or Local stop is TRUE, VI sends #24 and stops. When using as a sub-VI, use Refnum stop to stop VI from caller. C-887, F-206, M-8X0: Check HLP?/HELP answer to find out if FSA is supported. For GCS syntax version = GCS 2.0 or higher (Check with CSV?.vi. If CSV?.vi is not supported, syntax version is GCS 1.0), VI calls FSS? automatically to determine if scan was successful or not.

2.3.9. FSC.vi (Controller Algorithms.llb)

Valid for	C-880, C-887, F-206, M-8X0
Input	System number (1), Board (1), Axis 1 to scan (empty string), Axis 2 to scan (empty string), Range axis 1 (0.1), Treshold level (0.1), Step size (0.05), Range axis 2 (0.1), Use #7 polling (F), Timeout (s) (300), Refnum stop (F),

Local stop (F), Error in (no error)

C-880: Use #7 polling = TRUE

C-887, F-206, M-8X0: Use #7 polling = FALSE, Range axis 2 must be identical with Range axis 1 (no rectangular scan implemented)

Output: Scan successful? (T/F), Error out

Remarks: Performs a 2D scan in specified axes, waits until scan is ready using #7 polling (if checked) and indicates whether scan was successful or not. For a square scan, Range axis 2 must be identical to Range axis 1, otherwise a rectangular scan is performed. Timeout is only valid if Use #7 polling is FALSE. If Refnum stop or Local stop is TRUE, VI sends #24 and stops. When using as a sub-VI, use Refnum stop to stop VI from caller.

C-887, F-206, M-8X0: Check HLP?/HELP answer to find out if FSC is supported. For GCS syntax version = GCS 2.0 or higher (Check with CSV?.vi. If CSV?.vi is not supported, syntax version is GCS 1.0), VI calls FSS? automatically to determine if scan was successful or not.

2.3.10. FSM.vi (Controller Algorithms.llb)

Valid for: C-880, C-887, F-206, M-8X0

Input: System number (1), Board (1), Axis 1 to scan (empty string), Axis 2 to scan (empty string), Range axis 1 (0.1), Treshold level (0.1), Step size (0.05), Range axis 2 (0.1), Use #7 polling (F), Timeout (s) (300), Refnum stop (F), Local stop (F), Error in (no error)

C-880: Use #7 polling = TRUE

C-887, F-206, M-8X0: Use #7 polling = FALSE, Range axis 2 must be identical with Range axis 1 (no rectangular scan implemented)

Output: Scan successful? (T/F), Error out

Remarks: Performs a 2D scan to maximum in specified axes, waits until scan is ready using #7 polling (if checked) and indicates whether scan was successful or not. For a square scan, Range axis 2 must be identical to Range axis 1, otherwise a rectangular scan is performed. Timeout is only valid if Use #7 polling is FALSE. If Refnum stop or Local stop is TRUE, VI sends #24 and stops. When using as a sub-VI, use Refnum stop to stop VI from caller.

C-887, F-206, M-8X0: Check HLP?/HELP answer to find out if FSM is supported. For GCS syntax version = GCS 2.0 or higher (Check with CSV?.vi. If CSV?.vi is not supported, syntax version is GCS 1.0), VI calls FSS? automatically to determine if scan was successful or not.

2.3.11. FSN.vi (Controller Algorithms.llb)

Valid for: C-887, F-206, M-8X0

Input: System number (1), Board (1), Axes to scan (empty string array), Scan range (empty num. array), No. of digits (4), Treshold level (1.0), D (1), R (0), C(0), Use #7 polling (F), Timeout (s) (300), Refnum stop (F), Local stop (F), Error in (no error)

F-206: Use #7 polling = FALSE, Axes to scan can be a subset of (X,Y,Z,U,V,W). Scan cannot be stopped.

C-887, M-8X0: Use #7 polling = FALSE, Axes to scan can be a subset of (X,Y,Z,U,V,W). Scan cannot be stopped.

Output: Scan successful? (T/F), Error out

Remarks Performs a 1D scan following a trajectory described by the given parameters, waits until scan is ready (using #7 polling if checked) and indicates whether scan was successful or not. Timeout is only valid if Use #7 polling is FALSE. See User Manual for further details. If Refnum stop or Local stop is TRUE, VI sends #24 and stops. When using as a sub-VI, use Refnum stop to stop VI from caller.

C-887, F-206, M-8X0: Check HLP?/HELP answer to find out if FSN is supported. Only for GCS syntax version = GCS 1.0 (Check with CSV?.vi. If CSV?.vi is not supported, syntax version is GCS 1.0).

2.3.12. FSN?.vi (Controller Algorithms.llb)

Valid for C-887, F-206, M-8X0

Input System number (1), Queried axes (X,Y,Z,U,V,W), Error in (no error)

F-206: Queried axes = X,Y,Z,U,V,W

C-887, M-8X0: Queried axes = X,Y,Z,U,V,W

Output Maximum level, Position values, Error out

Remarks Returns the maximum level found during the last FSN scan and the coordinates where it was found.

C-887, F-206, M-8X0: Check HLP?/HELP answer to find out if FSN? is supported. Only for GCS syntax version = GCS 1.0 (Check with CSV?.vi. If CSV?.vi is not supported, syntax version is GCS 1.0).

2.3.13. FSS?.vi (Controller Algorithms.llb)

Valid for C-887, F-206, M-8X0

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

Output Scan successful?, Error out

Remarks Returns the result of the last scan algorithm.

C-887, F-206, M-8X0: Check HLP?/HELP answer to find out if FSS? is supported. Only for GCS syntax version = GCS 2.0 or higher (Check with CSV?.vi. If CSV?.vi is not supported, syntax version is GCS 1.0).

2.4. Coordinate systems VIs (“Coordinate Systems.llb”)

2.4.1. KCP.vi (Coordinate Systems.llb)

Valid for C-887, F-206, M-8X0

Input System number (1), Source CS (empty string), Destination CS (empty string), Error in (no error)

C-887, F-206, M-8X0: Check HLP?/HELP answer to find out if KCP is supported. Only for GCS syntax version = GCS 2.0 or higher (Check with CSV?.vi. If CSV?.vi is not supported, syntax version is GCS 1.0).

Output Controller error, Error out

Remarks Copies a coordinate system (e. g. in order to create a backup copy), and queries ERR?. Controller error is TRUE if selected system reports error code not equal to 0.

2.4.2. KEN.vi (Coordinate Systems.llb)

Valid for	C-887, F-206, M-8X0
Input	System number (1), Name (empty string), Error in (no error) C-887, F-206, M-8X0: Check HLP?/HELP answer to find out if KEN is supported. Only for GCS syntax version = GCS 2.0 or higher (Check with CSV?.vi. If CSV?.vi is not supported, syntax version is GCS 1.0).
Output	Controller error, Error out
Remarks	Enables an already defined coordinate system; i.e. assigns “enabled” state and queries ERR?. KEN sets the pivot point coordinates (“SPI.vi”) to zero when a KSD, KSW or KST coordinate system is enabled. Activating KLF, KLD, KSB or KSF does not change the pivot point settings. If the Hexapod is moving, the command cannot be applied. Enabling coordinate systems of type KLD, KLF and KSB requires command level 1 (“CCL.vi”). KEN settings are volatile but can be saved as power-on default with “WPA.vi” using the password “SKS”. Controller error is TRUE if selected system reports error code ≠ 0.

2.4.3. KEN?.vi (Coordinate Systems.llb)

Valid for	C-887, F-206, M-8X0
Input	System number (1), Names to query (empty string array), All names? (F), Error in (no error) C-887, F-206, M-8X0: Check HLP?/HELP answer to find out if KEN? is supported. Only for GCS syntax version = GCS 2.0 or higher (Check with CSV?.vi. If CSV?.vi is not supported, syntax version is GCS 1.0).
Output	Types, Error out
Remarks	Returns enabled coordinate systems. The returned information depends on the arguments used. If no argument is set, all enabled coordinate systems are returned. The KEN? command sets an error code if a coordinate system with the specified name is not defined.

2.4.4. KET?.vi (Coordinate Systems.llb)

Valid for	C-887, F-206, M-8X0
Input	System number (1), Types to query (empty string array), All types? (F), Error in (no error) C-887, F-206, M-8X0: Check HLP?/HELP answer to find out if KET? is supported. Only for GCS syntax version = GCS 2.0 or higher (Check with CSV?.vi. If CSV?.vi is not supported, syntax version is GCS 1.0).
Output	Names, Error out
Remarks	Returns enabled coordinate system types. The returned information depends on the arguments used. If no argument is set, all names of enabled coordinate systems are returned sorted by type.

2.4.5. KLC?.vi (Coordinate Systems.llb)

Valid for	C-887, F-206, M-8X0
Input	System number (1), Name (empty string), Name 2 (empty string), Item (empty string), Item 2 (empty string), Error in (no error) C-887, F-206, M-8X0: Check HLP?/HELP answer to find out if KLC? is supported. Only for GCS syntax version = GCS 2.0 or higher (Check with CSV?.vi. If

CSV?.vi is not supported, syntax version is GCS 1.0).

Output Response, Error out

Remarks Returns parameters of available Work/Tool combinations. The returned information depends on the used arguments. Applicable parameters are: name of Work coordinate system, name of Tool coordinate system, NLM, PLM, SSL, and SST. Name 2 is only valid in combination with Name; Item is only valid in combination with Name and Name 2; Item 2 is only valid in combination with Name, Name 2 and Item.

2.4.6. KLD.vi (Coordinate Systems.llb)

Valid for C-887, F-206, M-8X0

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

F-206, M-8X0: Check HLP?/HELP answer to find out if KLD is supported. Only for GCS syntax version = GCS 2.0 or higher (Check with CSV?.vi. If CSV?.vi is not supported, syntax version is GCS 1.0). Axes to set can be X, Y, Z, U, V and W.

Output Error out

Remarks Defines a levelling coordinate system (KLD-type) and queries ERR?. A coordinate system defined with this command is intended to eliminate Hexapod misalignment. Use KLD in case misalignment is known via an external measurable deviation.

If there is already a coordinate system with the same name, KLD will replace it if it is not used. Activating a KLD type coordinate system ("KEN.vi") does not cause any motion, and "POS?.vi" will report new position values. If the Hexapod is moving, the command cannot be applied. All pivot point coordinates ("SPI.vi") will remain unchanged. "SPI.vi" is allowed when a KLD type coordinate system is enabled. For new or replaced KLD type coordinate systems, the parent coordinate system will be set to "PI_LEVELLING". When replacing an existing coordinate system of this type, the chain relations will be changed and the parent will be set to "PI_LEVELLING". For axes which are not given the position values will be set to 0. Coordinate system settings are volatile but can be saved as power-on default with "WPA.vi" using the password "SKS". This command requires command level 1 ("CCL.vi").

2.4.7. KLF.vi (Coordinate Systems.llb)

Valid for C-887, F-206, M-8X0

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

C-887, F-206, M-8X0: Check HLP?/HELP answer to find out if KLF is supported. Only for GCS syntax version = GCS 2.0 or higher (Check with CSV?.vi. If CSV?.vi is not supported, syntax version is GCS 1.0).

Output Controller error, Error out

Remarks Defines a levelling coordinate system (KLF-type) and queries ERR?. A coordinate system defined with this command is intended to eliminate Hexapod misalignment. Use KLF in case the Hexapod is already in the aligned position. Controller error is TRUE if selected system reports error code not equal to 0.

If there is already a coordinate system with the same name, KLF will replace it if it is not used. Activating a KLF type coordinate system

("KEN.vi") does not cause any motion, and "POS?.vi" will report new position values. If the Hexapod is moving, the command cannot be applied. All pivot point coordinates ("SPI.vi") will remain. "SPI.vi" is allowed when a KLF type coordinate system is enabled. For new or replaced KLF type coordinate systems, the parent coordinate system will be set to "PI_LEVELLING". Coordinate system settings are volatile but can be saved as power-on default with "WPA.vi" using the password "SKS". This command requires command level 1 ("CCL.vi").

2.4.8. KLN.vi (Coordinate Systems.llb)

Valid for	C-887, F-206, M-8X0
Input	System number (1), Child CS (empty string), Parent CS (empty string), Error in (no error) C-887, F-206, M-8X0: Check HLP?/HELP answer to find out if KLN is supported. Only for GCS syntax version = GCS 2.0 or higher (Check with CSV?.vi. If CSV?.vi is not supported, syntax version is GCS 1.0).
Output	Controller error, Error out
Remarks	Links two coordinate systems together by defining a parent-child relation; thus forming a chain, and queries ERR?. KLN does not allow linking a coordinate system to itself. KLN does not check for cyclic coordinate system relations. Check is done with KEN. Linking a KSB-type coordinate system as child requires command level 1 ("CCL.vi"). Coordinate system settings are volatile but can be saved as power-on default with "WPA.vi" using the password "SKS". Controller error is TRUE if selected system reports error code not equal to 0.

2.4.9. KLN?.vi (Coordinate Systems.llb)

Valid for	C-887, F-206, M-8X0
Input	System number (1), Names to query (empty string array), All names? (F), Error in (no error) C-887, F-206, M-8X0: Check HLP?/HELP answer to find out if KLN? is supported. Only for GCS syntax version = GCS 2.0 or higher (Check with CSV?.vi. If CSV?.vi is not supported, syntax version is GCS 1.0).
Output	Response, Error out
Remarks	Returns coordinate system chains. A chain consists of minimum two linked coordinate systems. In case of a cyclic definition the reported chain ends with the cyclic coordinate system.

2.4.10. KLS?.vi (Coordinate Systems.llb)

Valid for	C-887, F-206, M-8X0
Input	System number (1), Name (empty string), Item (empty string), Item 2 (empty string), Error in (no error) C-887, F-206, M-8X0: Check HLP?/HELP answer to find out if KLS? is supported. Only for GCS syntax version = GCS 2.0 or higher (Check with CSV?.vi. If CSV?.vi is not supported, syntax version is GCS 1.0).
Output	Response, Error out
Remarks	Returns parameters of all available coordinate systems. The returned information depends on the used arguments. To query parameters of Work/Tool combinations use the "KLC?" command. The x, y, z, u, v, w values

of all Levelling coordinate systems are displayed according to current coordinate system settings. Item is only valid in combination with Name; Item 2 is only valid in combination with Name and Item.

2.4.11. KLT?.vi (Coordinate Systems.llb)

Valid for C-887, F-206, M-8X0

Input System number (1), Start Coordinate System (empty string), End Coordinate System (empty string), Error in (no error)

C-887, F-206, M-8X0: Check HLP?/HELP answer to find out if KLT? is supported. Only for GCS syntax version = GCS 2.0 or higher (Check with CSV?.vi. If CSV?.vi is not supported, syntax version is GCS 1.0).

Output Response, Error out

Remarks Returns the resulting coordinate system of a chain. A chain consists of minimum two linked coordinate systems. The resulting coordinate system is calculated beginning with the first specified coordinate system. The calculation ends with the second coordinate system if specified. If no second coordinate system is specified, "ZERO" is used for calculation. The position of the resulting coordinate system is canonical (SPI swivel (x, y, z) = (0, 0, 0)). If no parameters are used, the positions of all coordinate systems are returned with second coordinate system "ZERO". End Coordinate System is only valid in combination with Start Coordinate System.

2.4.1. KRM.vi (Coordinate Systems.llb)

Valid for C-887, F-206, M-8X0

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

C-887, F-206, M-8X0: Check HLP?/HELP answer to find out if KRM is supported. Only for GCS syntax version = GCS 2.0 or higher (Check with CSV?.vi. If CSV?.vi is not supported, syntax version is GCS 1.0).

Output Controller error, Error out

Remarks Removes a coordinate system and queries ERR?.

Coordinate system settings are volatile but can be saved as power-on default with "WPA.vi" using the password "SKS". If the coordinate system to remove is in use or does not exist, an error is set. If the coordinate system is used (e.g. part of an enabled chain), it cannot be removed. If a coordinate system which is part of a chain is removed, its parent and child are joined, so that the chain is still valid but KLT? results may differ. Coordinate systems of type KLD, KLF and KSB require command level 1 to remove them ("CCL.vi"). Controller error is TRUE if selected system reports error code not equal to 0.

2.4.2. KSB.vi (Coordinate Systems.llb)

Valid for C-887, F-206, M-8X0

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

C-887, F-206, M-8X0: Check HLP?/HELP answer to find out if KSB is supported. Only for GCS syntax version = GCS 2.0 or higher (Check with CSV?.vi. If CSV?.vi is not supported, syntax version is GCS 1.0). Axes to set can be X, Y, Z, U, V and W.

Output Error out

Remarks

Defines a Base coordinate system (KSB-Type) and queries ERR?. Use KSB in order to change the orientation of the Base coordinate system.

KSB only can rotate the coordinate system by 0, 90, 180, 270, -90, -180, -270 degrees. If there is already a coordinate system with the same name, KSB will replace it if it is not used. Activating a KSB type coordinate system ("KEN.vi") does not cause any motion, and "POS?.vi" will report new position values. If the Hexapod is moving, the command can be applied. All pivot point coordinates ("SPI.vi") will remain. "SPI.vi" is allowed when a KSB type coordinate system is enabled. For new or replaced KSB type coordinate systems, the parent coordinate system will be set to the currently enabled KSB coordinate system. When replacing a coordinate system of this type, the chain relations will not be changed if the coordinate system to replace is of the same type. For axes which are not given, the position values will be set to 0. Coordinate system settings are volatile but can be saved as power-on default with "WPA.vi" using the password "SKS". This command requires command level 1 ("CCL.vi").

2.4.3. KSD.vi (Coordinate Systems.llb)

Valid for

C-887, F-206, M-8X0

Input

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

C-887, F-206, M-8X0: Check HLP?/HELP answer to find out if KSD is supported. Only for GCS syntax version = GCS 2.0 or higher (Check with CSV?.vi. If CSV?.vi is not supported, syntax version is GCS 1.0). Axes to set can be X, Y, Z, U, V and W.

Output

Error out

Remarks

Defines a new KSD-type coordinate system in order to set a "directed" swivel with the parameters X, Y, Z (relative to the Hexapod platform) and queries ERR?. The coordinate system is rotated with the parameters U, V, W.

If there is already a coordinate system with the same name, KSD will replace it if it is not used. Activating a KSD type coordinate system ("KEN.vi") does not cause any motion, and "POS?.vi" will report new position values. If the Hexapod is moving, the command can be applied. All pivot point coordinates ("SPI.vi") will be reset to the previous values after disabling. "SPI.vi" is not allowed when a KSD type coordinate system is enabled. For new or replaced KSD type coordinate systems, the parent coordinate system will be set to "ZERO". When replacing a coordinate system of this type, the chain relations will not be changed if the coordinate system to replace is of the same type. For axes which are not given, the position values will be set to 0. Coordinate systems settings are volatile but can be saved as power-on default with "WPA.vi" using the password "SKS". This command requires command level 0 or 1.

2.4.4. KSF.vi (Coordinate Systems.llb)

Valid for

C-887, F-206, M-8X0

Input

System number (1), Name (empty string), Error in (no error)

C-887, F-206, M-8X0: Check HLP?/HELP answer to find out if KSF is supported. Only for GCS syntax version = GCS 2.0 or higher (Check with CSV?.vi. If CSV?.vi is not supported, syntax version is GCS 1.0).

Output	Controller error, Error out
Remarks	Defines a coordinate system based on the current position of the Hexapod platform and queries ERR?.
	If there is already a coordinate system with the same name, KSF will replace it if it is not used. Activating a KSF type coordinate system ("KEN.vi") does not cause any motion, and "POS?.vi" will report position (0, 0, 0 ,0 ,0 ,0). If the Hexapod is moving, the command cannot be applied. All pivot point coordinates ("SPI.vi") will remain. "SPI.vi" is allowed when a KSF type coordinate system is enabled. For new or replaced KSF type coordinate systems, the parent coordinate system will be set to "ZERO". When replacing a coordinate system of this type, the chain relations will be changed and the parent will be set to "ZERO". Coordinate systems that are currently in use cannot be changed. Coordinate system settings are volatile but can be saved as power-on default with "WPA.vi" using the password "SKS". This command requires command level 0 or 1.Controller error is TRUE if selected system reports error code not equal to 0.

2.4.5. KST.vi (Coordinate Systems.llb)

Valid for	C-887, F-206, M-8X0
Input	System number (1), Name (empty string), Axes to set (empty string array), Position values (empty num. array), No. of digits (4), Error in (no error)
	C-887, F-206, M-8X0: Check HLP?/HELP answer to find out if KST is supported. Only for GCS syntax version = GCS 2.0 or higher (Check with CSV?.vi. If CSV?.vi is not supported, syntax version is GCS 1.0). Axes to set can be X, Y, Z, U, V and W.
Output	Error out
Remarks	Defines a new Tool coordinate system (KST-type) and queries ERR?.
	If there is already a coordinate system with the same name, KST will replace it if it is not used. Activating a KST type coordinate system ("KEN.vi") does not cause any motion, and "POS?.vi" will report new position values. If the Hexapod is moving, the command can be performed. All pivot point coordinates ("SPI.vi") will be reset to the previous values after disabling. "SPI.vi" is not allowed when a KST type coordinate system is enabled. For new or replaced KST type coordinate systems, the parent coordinate system will be set to "ZERO". When replacing a coordinate system of this type, the chain relations will not be changed if the coordinate system to replace is of the same type. Coordinate systems that are currently in use cannot be changed. For axes which are not given, the position values will be set to 0.
	Coordinate system settings are volatile but can be saved as power-on default with "WPA.vi" using the password "SKS". This command requires command level 0 or 1.

2.4.6. KSW.vi (Coordinate Systems.llb)

Valid for	C-887, F-206, M-8X0
Input	System number (1), Name (empty string), Axes to set (empty string array), Position values (empty num. array), No. of digits (4), Error in (no error)
	C-887, F-206, M-8X0: Check HLP?/HELP answer to find out if KSW is supported. Only for GCS syntax version = GCS 2.0 or higher (Check with CSV?.vi. If CSV?.vi is not supported, syntax version is GCS 1.0). Axes to set can be X,

Y, Z, U, V and W.

Output Error out

Remarks Defines a new Work coordinate system (KSW-type) and queries ERR?.

If there is already a coordinate system with the same name, KSW will replace it if it is not used. Activating a KSW type coordinate system ("KEN.vi") does not cause any motion, and "POS?.vi" will report new position values. If the Hexapod is moving, the command can be performed. All pivot point coordinates ("SPI.vi") will be reset to the previous values after disabling. "SPI.vi" is not allowed when a KSW type coordinate system is enabled. For new or replaced KSW type coordinate systems, the parent coordinate system will be set to "ZERO". When replacing a coordinate system of this type, the chain relations will not be changed if the coordinate system to replace is of the same type. Coordinate systems that are currently in use cannot be changed. For axes which are not given, the position values will be set to 0.

Coordinate system settings are volatile but can be saved as power-on default with "WPA.vi" using the password "SKS". This command requires command level 0 or 1.

2.4.7. MRT.vi (Coordinate Systems.llb)

Valid for C-887, F-206, M-8X0

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-887, F-206, M-8X0: Check HLP?/HELP answer to find out if MRT is supported. Only for GCS syntax version = GCS 2.0 or higher (Check with CSV?.vi. If CSV?.vi is not supported, syntax version is GCS 1.0).

Output Error out

Remarks Executes a relative move in the Tool coordinate system. No. of digits is the number of digits after the decimal point in the position value(s) that will be sent.

Target position results from calculating the translation first and then the rotation. Only Hexapod axes can be moved. If no tool coordinate system is explicitly defined, the default tool coordinate system (0, 0, 0, 0, 0, 0) is used.

2.4.8. MRW.vi (Coordinate Systems.llb)

Valid for C-887, F-206, M-8X0

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-887, F-206, M-8X0: Check HLP?/HELP answer to find out if MRW is supported. Only for GCS syntax version = GCS 2.0 or higher (Check with CSV?.vi. If CSV?.vi is not supported, syntax version is GCS 1.0).

Output Error out

Remarks Executes a relative move in the Work coordinate system. No. of digits is the number of digits after the decimal point in the position value(s) that will be sent. The target position results from calculating the translation first and then the rotation. Only Hexapod axes can be moved. If no Work coordinate system is explicitly defined, the default Work coordinate system (0, 0, 0, 0, 0, 0) is used.

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

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

Valid for	Analog systems, C-413, C-867, C-880, C-884, C-887, E-517, E-712, E-725, E-761, E-861, E-871, F-206, M-8X0, Mercury_GCS. To support analog interfacing, VI must be present for E-709 and 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.

2.5.2. 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.

2.5.3. GetDataFormat.vi (File handling.llb)

Valid for	Analog systems, C-413, C-702, C843, C-866, C-867, C-880, C-884, C-887, E-517, E-709, E-710, E-712, E-725, E-753, E-755, E-761, E-861, E-871, F-206, M-8X0, Mercury_GCS. 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. Sub-VI for "Show_Save_Load_XY_Data.vi". See separate manual "GCSData_User_SM146E.pdf" and control descriptions in the diagram for more information.

2.5.4. MatrixIO.vi (File handling.llb)

Valid for	Analog systems, C-413, C-702, C-843, C-866, C-867, C-880, C-884, C-887, E-517, E-709, E-710, E-712, E-725, E-753, E-755, E-761, E-861, E-

	871, F-206, M-8X0, Mercury_GCS. 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. Sub-VI for “1D_Scan.vi”, “2D_Scan.vi”, “Show_Save_Load_XY_Data.vi” and “Show_Save_Load_XYZ_Data.vi”. See separate manual “GCSData_User_SM146E.pdf” and control descriptions in the diagram for more information.

2.5.5. READ.vi (File handling.llb)

Valid for	C-887, F-206, M-8X0
Input	System number (1), File name (empty string), Error in (no error)
Output	File contents, Error out
Remarks	Reads the contents of the given file.

C-887, F-206, M-8X0: Check HLP?/HELP answer to find out if READ is supported. Only for GCS syntax version = GCS 1.0 (Check with CSV?.vi. If CSV?.vi is not supported, syntax version is GCS 1.0)..File name can be any file stored on the controller with file extension .dat, .txt or .mac. For files outside the HEXAPOD directory the full path must be given.

2.5.6. TableIO.vi (File handling.llb)

Valid for	Analog systems, C-413, C-702, C-843, C-866, C-867, C-880, C-887, E-517, E-709, E-710, E-712, E-725, E-753, E-755, E-761, E-861, E-870, E-871, F-206, M-8X0, Mercury, Mercury_GCS. 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. Sub-VI for “DDL.vi””DRR?.vi”, “GWD?.vi”, “JLT?.vi”, “TWS.vi” etc. See separate manual “GCSData_User_SM146E.pdf” and control descriptions in the diagram for more information. Sub-VI for “DRR?.vi”.

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

2.6.1. *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.

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

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

2.6.3. CSV.vi (General command.llb)

Valid for	E-517, C-887, F-206, M-8X0
Input	System number (1), GCS syntax version (1.00), Error in (no error) E-517: GCS syntax version must be GCS 2.0. GCS 1.0 switches to E-516 compatibility mode (use E-516 GCS LabVIEW driver set instead when working in E-516 compatibility mode). Requires command level 1, see CCL.vi. C-887, F-206, M-8X0: Check HLP?/HELP answer to find out if CSV is supported. Only for combinations of mechanics and C-887 controller.
Output	Error out, Controller error
Remarks	Sets GCS syntax version. Controller error is TRUE if selected system reports error code ≠ 0.

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

Valid for	C-413, C-702, C-867, C-884, C-887, E-517, E-709, E-712, E-725, E-753, E-755, E-761, E-861, E-870, E-871, F-206, Hydra, Pollux, M-8X0, Mercury_GCS
Input	System number (1), Error in (no error)
Output	Syntax, Error out
Remarks	Returns current GCS syntax version. C-887, F-206, M-8X0: Check HLP?/HELP answer to find out if CSV? is supported. Only for GCS syntax version = GCS 2.0 or higher (If CSV?.vi is not supported, syntax version is GCS 1.0).

2.6.5. 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

C-887, F-206, M-8X0: Read from controller = FALSE, Invert order = FALSE, Connected axes = X,Y,Z,U,V,W, (A,B,K,L,M 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.

2.6.6. 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".

2.6.7. 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.

2.6.8. 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.

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

Valid for	Analog systems, C-413, C-702, C-843, C-843.PM, C-848, C-865, C-866, C-867, C-880, C-884, C-887, E-516, E-517, E-709, E-712, E-725, E-710, E-753, E-755, E-761, E-861, E-870, E-871, F-206, Hydra, Pollux, M-8X0, Mercury, Mercury_GCS (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. C-887, F-206, M-8X0: For GCS syntax version = GCS 1.0 (Check with CSV?.vi. If CSV?.vi is not supported, syntax version is GCS 1.0), check HELP answer to determine if HLP? is supported. HLP? and HELP are equivalent.

2.6.10. HLT.vi (General command.llb)

Valid for	C-413, C-702, C-843, C-843.PM, C-844, C-848, C-865, C-866, C-867, C-880, C-884, C-887, E-517, E-710, E-755, E-761, E-861, E-871, F-206, Hydra, Pollux, M-8X0, Mercury, Mercury_GCS
Input	System number (1), Affected axes (empty string array), All axes? (F), Axis identifier? (T), Error in (no error) C-413: If All axes? = TRUE, then Axis identifier? can be FALSE 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 C-884: If All axes? = TRUE, then Axis identifier? can be FALSE 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 E-871: If All axes? = TRUE, then Axis identifier? can be FALSE C-887, F-206, M-8X0: If All axes? = TRUE, then Axis identifier? can be FALSE Hydra, Pollux: If All axes? = TRUE, then Axis identifier? can be FALSE Mercury: If All axes? = TRUE, then Axis identifier? can be FALSE Mercury_GCS: 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. C-887, F-206, M-8X0: Check HLP?/HELP answer to find out if HLT is supported. Only for GCS syntax version = GCS 2.0 or higher (Check with CSV?.vi. If

CSV?.vi is not supported, syntax version is GCS 1.0).

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

Valid for: C-413, C-843, C-867, C-884, C-887, E-517, E-709, E-710, E-712, E-725, E-753, E-755, E-761, E-861, E-870, E-871, F-206, Hydra, Pollux, Mercury, Mercury_GCS, M-8X0

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.

C-887, F-206, M-8X0: Check HLP?/HELP answer to find out if HPA? is supported. Only for GCS syntax version = GCS 2.0 or higher (Check with CSV?.vi. If CSV?.vi is not supported, syntax version is GCS 1.0).

2.6.12. 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.

2.6.13. MOV.vi (General command.llb)

Valid for: Analog systems, C-413, C-702, C-843, C-843.PM, C-844, C-848, C-865, C-866, C-867, C-880, C-880K005, C-884, C-887, E-516, E-517, E-709, E-710, E-712, E-725, E-753, E-755, E-761, E-816, E-861, E-871, F-206, Hydra, Pollux, M-8X0, Mercury, Mercury_GCS

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-413: Motion commands are not allowed when the wave generator is active or the analog input is used for target generation.

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

C-884: This command works only in closed-loop operation. Motion commands like MOV are not allowed when control via a Human Interface Device (HID) is active on the axis.

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.

E-871: This command works only in closed-loop operation. Motion commands like MOV are not allowed when a joystick is active on the axis.

C-887, F-206, M-8X0: For GCS syntax version = GCS 1.0 (Check with CSV?.vi. If CSV?.vi is not supported, syntax version is GCS 1.0), no mix between F-206 axes X,Y,Z,U,V,W, separate axes A, B and NanoCube axes K, L, M in one call allowed. Subsequent MOV commands are handled in a buffer on controller, so please use General Wait for motion to stop.vi or #5.vi to check for motion to stop before sending the next MOV.
For GCS syntax version = GCS 2.0 or higher, subsequent MOV commands override the target position of the previous MOV command. Behavior can be changed by setting parameter 0x19001900 (SPA.vi), see controller User Manual for details.

Hydra, Pollux: This command works only in closed-loop operation.

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

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”.

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

Valid for: C-413, C-702, C-843, C-843.PM, C-844, C-848, C-865, C-866, C-867, C-880, C-884, C-887, E-516, E-517, E-709, E-710, E-712, E-725, E-753, E-755, E-761, E-816, E-861, E-871, F-206, Hydra, Pollux, M-8X0, Mercury, Mercury_GCS

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

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: For GCS syntax version = GCS 1.0 (Check with CSV?.vi. If CSV?.vi is not supported, syntax version is GCS 1.0), command has different implementation, please use MOV?_old.vi

C-887, M-8X0: For GCS syntax version = GCS 1.0 (Check with CSV?.vi. If CSV?.vi is not supported, syntax version is GCS 1.0), command has different implementation, please use MOV?_old.vi

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

Output Target position, Error out

Remarks Returns commanded target position.

2.6.15. MVR.vi (General command.llb)

Valid for Analog systems, C-413, C-702, C-843, C-843.PM, C-844, C-848, C-865, C-866, C-867, C-880, C-880K005, C-884, C-887, E-516, E-517, E-709, E-710, E-712, E-725, E-753, E-755, E-761, E-816, E-861, E-871, F-206, Hydra, Pollux, M-8X0, Mercury, Mercury_GCS

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

C-884: This command works only in closed-loop operation. Motion commands like MOV are not allowed when control via a Human Interface Device (HID) is active on the axis.

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.

E-871: This command works only in closed-loop operation. Motion commands like MOV are not allowed when a joystick is active on the axis.

C-887, F-206, M-8X0: Check HLP?/HELP answer to find out if MVR is supported. Only for GCS syntax version = GCS 2.0 or higher (Check with CSV?.vi. If CSV?.vi is not supported, syntax version is GCS 1.0).

Hydra, Pollux: This command works only in closed-loop operation.

Mercury_GCS: This command works only in closed-loop operation.
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.

C-413: Motion commands are not allowed when the wave generator is active or the analog input is used for target generation.

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.

2.6.16. MWG.vi (General command.llb)

Valid for: C-887, F-206, M-8X0 (but must be present for C-866, C-867, C-880, C-884, E-761, E-861, E-871 and Mercury_GCS 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: Only for GCS syntax version = GCS 1.0 (Check with CSV?.vi. If CSV?.vi is not supported, syntax version is GCS 1.0). No mix between F-206 axes X,Y,Z,U,V,W ,separate axes A,B and NanoCube axes K,L,M allowed. For compatibility reasons, VI calls MOV.vi if GCS syntax version is > GCS 1.0.

C-887, M-8X0: Only for GCS syntax version = GCS 1.0 (Check with CSV?.vi. If CSV?.vi is not supported, syntax version is GCS 1.0), and only for controllers based on C-842.80. Check HLP?/HELP answer to find out if MWG is supported. If not supported, this VI must be present anyway. For compatibility reasons, VI calls MOV.vi if GCS syntax version is > GCS 1.0.

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”.

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

Valid for: C-413, C-702, C-843, C-843.PM, C-848, C-865, C-866, C-867, C-880, C-884, C-887, E-516, E-517, E-709, E-710, E-712, E-725, E-753, E-755, E-761, E-816, E-861, E-871, F-206, Hydra, Pollux, M-8X0, Mercury, Mercury_GCS (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)

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.

All other systems: 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.

C-887, F-206, M-8X0: Check HLP?/HELP answer to find out if ONT? is supported. Only for GCS syntax version = GCS 2.0 or higher (Check with CSV?.vi. If CSV?.vi is not supported, syntax version is GCS 1.0).

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

Valid for: Analog systems, C-413, C-702, C-843, C-843.PM, C-844, C-848, C-865, C-866, C-867, C-880, C-880K005, C-884, C-887, E-516, E-517, E-709, E-710, E-712, E-725, E-753, E-755, E-761, E-816, E-861, E-871, F-206, Hydra, Pollux, M-8X0, Mercury, Mercury_GCS

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.

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).

C-887, F-206, M-8X0: For GCS syntax version = GCS 1.0 (Check with CSV?.vi. If CSV?.vi is not supported, syntax version is GCS 1.0), returned position value is the commanded target position for the axis.

2.6.19. PUN?.vi (General command.llb)

Valid for: C-413, C-887, E-712, E-725, E-753, F-206, M-8X0

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

C-413, C-887, E-712, E-725, E-753, F-206, M-8X0: If All axes? = TRUE, then Axis identifier? can be FALSE

Output: Position unit, Error out

Remarks: Returns the name of the current position unit for queried axes.

E-712, E-725, E-753: Check HLP? answer to find out if PUN? is supported.

C-887, F-206, M-8X0: Check HLP?/HELP answer to find out if PUN? is supported. Only for GCS syntax version = GCS 2.0 or higher (Check with CSV?.vi. If CSV?.vi is not supported, syntax version is GCS 1.0).

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

Valid for: Analog systems, C-413, C-702, C-843, C-843.PM, C-844, C-848, C-865, C-866, C-867, C-880, C-880K005, C-884, C-887, E-516, E-517, E-709, E-710, E-712, E-725, E-753, E-755, E-761, E-816, E-861, E-871, F-206, Hydra, Pollux, M-8X0, Mercury, Mercury_GCS (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-848: Invert order should be TRUE, SAI? ALL must be FALSE

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

All other systems: 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.

C-887, F-206, M-8X0: Check HLP?/HELP answer to find out if SAI? is supported. Only for GCS syntax version = GCS 2.0 or higher (Check with CSV?.vi. If CSV?.vi is not supported, syntax version is GCS 1.0).

2.6.21. SPA.vi (General command.llb)

Valid for: C-413, C-702, C-843, C-843.PM, C-844, C-848, C-865, C-866, C-867, C-880, C-880K005, C-884, C-887, E-516, E-517, E-709, E-710, E-712, E-725, E-753, E-755, E-761, E-816, E-861, E-871, F-206, Hydra, Pollux, M-8X0, Mercury, Mercury_GCS

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-413: Parameter no. format is TRUE (hex.). Use "HPA?.vi" to get valid parameter numbers or see the C-413 User Manual. See E-710 for warnings. Do not set more than 4 parameters at once.

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.

C-884: Parameter no. format is TRUE (hex.). Use “HPA?.vi” to get valid parameter numbers or see the C-884 User Manual. Do not set more than 4 parameters at once. See E-710 for warnings.

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 is limited to set 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.

E-870: Parameter no. format is TRUE (hex.). Use “HPA?.vi” to get valid parameter numbers or see the E-870 User Manual. See E-710 for warnings. Each command is limited to set one parameter for only one channel.

E-871: Parameter no. format is TRUE (hex.). Use “HPA?.vi” to get valid parameter numbers or see the controller User Manual. Only one parameter value for only one axis per command allowed. See E-710 for warnings.

C-887, F-206, M-8X0: Parameter no. format is TRUE (hex.). Use “HPA?.vi” to get valid parameter numbers or see the controller User Manual.

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

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

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

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-, sensor, PIShift- or demux-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-413, C-867, C-884, E-516, E-517, E-710, E-712, E-725, E-753, E-755, E-861, E-870, E-871, Hydra, Pollux, Mercury_GCS: 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.

C-887, F-206, M-8X0: Check HLP?/HELP answer to find out if SPA is supported. Only for GCS syntax version = GCS 2.0 or higher (Check with CSV?.vi. If CSV?.vi is not supported, syntax version is GCS 1.0).

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.

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

Valid for: C-413, C-702, C-843, C-843.PM, C-844, C-848, C-865, C-866, C-867, C-880, C-880K005, C-884, C-887, E-516, E-517, E-709, E-710, E-712, E-725, E-753, E-755, E-761, E-816, E-861, E-870, E-871, F-206, Hydra, Pollux, M-8X0, Mercury, Mercury_GCS

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-413: Parameter no. format is TRUE (hex.). Use “HPA?.vi” to get valid parameter numbers. Do not query more than 6 parameter no. at once (except with Without axes? = TRUE).

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

C-884: Parameter no. format is TRUE (hex.). Use “HPA?.vi” to get valid parameter numbers. Do not query more than 4 parameter no. at once (except with Without axes? = TRUE) Use Without axes? = TRUE for a query of all parameters.

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.

E-870: Parameter no. format is TRUE (hex.). Use “HPA?.vi” to get valid parameter numbers. Only one parameter value for only one channel per command allowed. Use Without axes? = TRUE for a query of all parameters.

E-871: 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).

F-206: Check HLP?/HELP answer to find out if SPA? is supported. Only for GCS syntax version = GCS 2.0 or higher (Check with CSV?.vi. If CSV?.vi is not supported, syntax version is GCS 1.0). Parameter no. format is TRUE (hex.). Use “HPA?.vi” to get valid parameter numbers. Length of command is limited by firmware (appr. 10 parameters per call), see controller User Manual for details. Use Without axes? = TRUE for a query of all parameters.

Hydra, Pollux: 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) Use Without axes? = TRUE for a query of all parameters.

C-887, M-8X0: Check HLP?/HELP answer to find out if SPA? is supported.
For GCS syntax version = GCS 2.0 or higher (Check with CSV?.vi. If CSV?.vi is not supported, syntax version is GCS 1.0), Parameter no. format is TRUE (hex.). Use “HPA?.vi” to get valid parameter numbers. Length of command is limited by firmware (appr. 10 parameters per call), see controller User Manual for details. Use Without axes? = TRUE for a query of all parameters.
For GCS syntax version = GCS 1.0, Parameter no. format is FALSE (decimal), Axes to query can be 1 to 6 (corresponds to strut no.) and 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.

Mercury_GCS: 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.

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-, sensor, PIShift or demux-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)

2.6.23. STP.vi (General command.llb)

Valid for: Analog systems, C-413, C-702, C-843, C-843.PM, C-844, C-848, C-865, C-866, C-867, C-880, C-884, C-887, E-516, E-517, Ê-709, E-712, E-725, E-753, E755, E-761, E-861, E-870, E-871, F-206, Hydra, Pollux, M-8X0, Mercury, Mercury_GCS (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.

All other systems: 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).

C-413, 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, if supported), by analog control input and autozero motion (ATZ).

C-887, F-206, M-8X0: Check HLP?/HELP answer to find out if STP is supported. Only for GCS syntax version = GCS 2.0 or higher (Check with CSV?.vi. If CSV?.vi is not supported, syntax version is GCS 1.0). For GCS 1.0, use STOP.vi instead.

2.6.24. SVO.vi (General command.llb)

Valid for: C-413, C-702, C-843, C-843.PM, C-844, C-848, C-865, C-866, C-867, C-880, C-884, C-887, E-516, E-517, E-709, E-710, E-712, E-725, E-753, E-755, E-761, E-816, E-861, E-871, F-206, Hydra, Pollux, M-8X0, Mercury, Mercury_GCS

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.

C-887, F-206, M-8X0: For GCS syntax version = GCS 2.0 or higher (Check with CSV?.vi. If CSV?.vi is not supported, syntax version is GCS 1.0), Without axis ID = FALSE. For GCS 1.0, Without axis ID = TRUE and 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.

C-887, M-8X0: Check HLP?/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.

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

Valid for: C-413, C-702, C-843, C-843.PM, C-844, C-848, C-865, C-866, C-867, C-880, C-884, C-887, E-516, E-517, E-709, E-710, E-712, E-725, E-753, E-755, E-761, E-816, E-861, E-871, F-206, Hydra, Pollux, M-8X0, Mercury, Mercury_GCS

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.

C-887, F-206, 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

C-887, F-206, M-8X0: For GCS syntax version = GCS 1.0 (Check with CSV?.vi. If CSV?.vi is not supported, syntax version is GCS 1.0), only first field of servo status array is valid.

Remarks: Returns servo status of queried axes.

C-887, F-206, M-8X0: Check HLP?/HELP answer to find out if SVO? is supported.

2.6.26. VEL.vi (General command.llb)

Valid for: Analog systems, C-413, C-702, C-843, C-843.PM, C-844, C-848, C-865, C-866, C-867, C-880, C-880K005, C-884, C-887, E-516, E-517, E-709, E-710, E-712, E-725, E-753, E-755, E-761, E-861, F-206, Hydra, Pollux, M-8X0, Mercury, Mercury_GCS. 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-413, C-867, C-884, Mercury_GCS: 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.

C-887, F-206, M-8X0: For GCS syntax version = GCS 2.0 or higher (Check with CSV?.vi. If CSV?.vi is not supported, syntax version is GCS 1.0), VEL

command is only valid for velocity of axes A and/or B, Without axis ID? = False. For platform velocity, VI sends command VLS instead, in this case Without axis ID? = TRUE. For axes K, L, M command is not valid.
For GCS syntax version = GCS 1.0, to set platform velocity: Without axis ID? = TRUE; to set velocity of axes A and/or B: Without axis ID? = False; axes K, L, M: command not valid.

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-413: Velocity settings made with VEL are present in RAM only and will be reset to default (parameter 0x06010400) when the controller is powered down or rebooted.

C-867, C-884, Hydra, Pollux, Mercury_GCS: 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 controller 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.

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

Valid for Analog systems, C-413, C-702, C-843, C-843.PM, C-844, C-848, C-865, C-866, C-867, C-880, C-880K005, C-884, C-887, E-516, E-517, E-709, E-710, E-712, E-725, E-753, E-755, E-761, E-861, F-206, Hydra, Pollux, M-8X0, Mercury, Mercury_GCS. 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-413, C-865, C-866, C-867, C-884, Hydra, Pollux, Mercury_GCS: 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.

C-887, F-206, M-8X0: For GCS syntax version = GCS 2.0 or higher (Check with CSV?.vi. If CSV?.vi is not supported, syntax version is GCS 1.0), and velocity of axes A and/or B, All axes? must be FALSE. For platform velocity, All axes? must be TRUE AND Axis identifier? = FALSE; VI sends VLS? instead. For axes K, L, M command is not valid.
For GCS syntax version = GCS 1.0, Velocity of platform: All axes? = TRUE AND Axis identifier? = FALSE; velocity of axes A, B: All axes? must be FALSE; axes K, L, M: command not valid

Output Velocity, Error out

C-880: NanoCube axes will report velocity = 0

C-887, F-206, M-8X0: For platform velocity: only first field of velocity array is valid

Remarks Returns velocity setting for specified axes.

2.6.28. VER?.vi (General command.llb)

Valid for C-702, C-848, C-866, C-880, C-884, C-887, E-516, E-517, E-755, E-761, E-871, F-206, Hydra, Pollux, M-8X0, Mercury, Mercury_GCS

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

Output Firmware version, Error out

Remarks Returns firmware string.

2.6.29. VMO.vi (General command.llb)

Valid for C-702, C-848, C-880, C-887, F-206, M-8X0 (but must be present for C-866, C-867, C-884, E-761, E-861, E-871 and Mercury_GCS 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.

C-887, F-206, M-8X0: For GCS syntax version = GCS 1.0 (Check with CSV?.vi. If CSV?.vi is not supported, syntax version is GCS 1.0), VI sends VMO command. For GCS 2.0 and higher, VI sends VMO? command instead.

2.7. Joystick-specific VIs (“Joystick.llb”)

2.7.1. HIB?.vi (Joystick.llb)

Valid for	C-867, C-884, C-887, E-871, F-206, M-8X0
Input	System number (1), DeviceID to query (empty num. array, 0), All DeviceIDs? (F), DeviceButton (empty num. array, 0), Error in (no error)
Output	State, Error out
Remarks	Returns the state of the buttons of a Human Interface Device (HID). See controller User Manual for details. C-867: Check HLP? answer to find out if HIB? is supported. C-887, F-206, M-8X0: Check HLP?/HELP answer to find out if HIB? is supported. Only for GCS syntax version = GCS 2.0 or higher (Check with CSV?.vi. If CSV?.vi is not supported, syntax version is GCS 1.0).

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

2.8.1. FRF.vi (Limits.llb)

Valid for	C-413, C-843, C-843.PM, C-867, C-880 (only K006/K007 version), C-880K005, C-884, C-887, E-712, E-861, E-871, F-206, Hydra, Pollux, M-8X0, Mercury_GCS (but must be present for C-702, C-848, E-710, C-865, C-866 and Mercury also))
Input	System number (1), Affected axes (empty string array), All axes? (F), Axis identifier? (T), Error in (no error) C-413, C-843, C-843.PM, C-867, C-880, C-884, C-887, E-861, E-871, F-206, Hydra, Pollux, M-8X0, Mercury_GCS: 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-413, C-867, C-884, E-871, Hydra, Pollux, Mercury_GCS: 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. See User Manual for details and specifics. 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.

C-887, F-206, M-8X0: Check HLP?/HELP answer to find out if FRF is supported. Only for GCS syntax version = GCS 2.0 or higher (Check with CSV?.vi. If CSV?.vi is not supported, syntax version is GCS 1.0). For GCS syntax version = GCS 1.0 use INI.vi for initialization.

2.8.2. FRF?.vi (Limits.llb)

Valid for	C-413, C-843, C-843.PM, C-866, C-867, C-880 (only C-880K006/C-880K007 version), C-880K005, C-884, C-887, E-712, E-755, E-861, E-871, F-206, Hydra, Pollux, M-8X0, Mercury_GCS
Input	System number (1), Axes to query (empty string array), All axes? (F), Axis identifier? (T), Error in (no error)
	C-413, C-843, C-843.PM, C-866, C-867, C-880, C-884, E-861, E-871, Hydra, Pollux, Mercury_GCS: 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
	C-887, F-206, M-8X0: If All axes? = TRUE, then Axis identifier? can be FALSE. Check HLP?/HELP answer to find out if FRF? is supported. Only for GCS syntax version = GCS 2.0 or higher (Check with CSV?.vi. If CSV?.vi is not supported, syntax version is GCS 1.0).
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.

2.8.3. LIM?.vi (Limits.llb)

Valid for	C-413, C-702, C-843, C-843.PM, C-848, C-865, C-866, C-867, C-880, C-884, C-887, E-712, E-755, E-861, E-871, F-206, Hydra, Pollux, M-8X0, Mercury, Mercury_GCS
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.
	C-887, F-206, M-8X0: Check HLP?/HELP answer to find out if LIM? is supported. Only for GCS syntax version = GCS 2.0 or higher (Check with CSV?.vi. If CSV?.vi is not supported, syntax version is GCS 1.0).
	E-712: Check HLP? answer to find out if LIM? is supported.

2.8.4. NLM.vi (Limits.llb)

Valid for	C-702, C-844, C-848, C-880, C-887, E-516, E-517, E-755, E-761, F-206, M-8X0
Input	System number (1), Affected axes (empty string array), Negative soft limit (empty num. array, 0), No. of digits (4), Error in (no error)

Output Error out

Remarks Sets negative soft limit for the specified axes. No. of digits is the number of digits after the decimal point in the limit values that will be sent.

E-516, E-517: The NLM 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 device is powered off.

E-755: Command not available for E-755.101

E-761: The NLM 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 with "Affected axes" as an empty array. Parameter changes not saved with WPA will be lost when the E-761 board is powered off or rebooted.

C-887, F-206, M-8X0: Check HLP?/HELP answer to find out if command is supported.

2.8.5. NLM?.vi (Limits.llb)

Valid for C-702, C-844, C-848, C-880, C-887, E-516, E-517, E-755, E-761, F-206, M-8X0

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

C-702, C-844, C-848, C-880, E-517, E-761: If All axes? = TRUE, then Axis identifier? can be FALSE

E-516: 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

C-887, F-206, M-8X0: If All axes? = TRUE, then Axis identifier? can be FALSE. Check HLP?/HELP answer to find out if command is supported.

Output Negative soft limit, Error out

Remarks Returns negative soft limit of queried axes.

2.8.6. PLM.vi (Limits.llb)

Valid for C-702, C-844, C-848, C-880, C-887, E-516, E-517, E-755, E-761, F-206, M-8X0

Input System number (1), Affected axes (empty string array), Positive soft limit (empty num. array, 0), No. of digits (4), Error in (no error)

Output Error out

Remarks Sets positive soft limit for the specified axes. No. of digits is the number of digits after the decimal point in the limit values that will be sent.

E-516, E-517: The PLM 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 device is powered off.

E-755: Command not available for E-755.101

E-761: The PLM 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 with "Affected axes" as an empty array. Parameter changes not saved with WPA will be lost when the E-761 board

is powered off or rebooted.

C-887, F-206, M-8X0: Check HLP?/HELP answer to find out if command is supported.

2.8.7. PLM?.vi (Limits.llb)

Valid for	C-702, C-844, C-848, C-880, C-887, E-516, E-517, E-755, E-761, F-206, M-8X0
Input	System number (1), Axes to query (empty string array), All axes? (F), Axis identifier? (T), Error in (no error) C-702, C-844, C-848, C-880, E-517, E-761: If All axes? = TRUE, then Axis identifier? can be FALSE E-516: 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 C-887, F-206, M-8X0: If All axes? = TRUE, then Axis identifier? can be FALSE. Check HLP?/HELP answer to find out if command is supported.
Output	Positive soft limit, Error out
Remarks	Returns positive soft limit of queried axes.

2.8.8. RON?.vi (Limits.llb)

Valid for	C-413, C-702, C-843, C-843.PM, C-848, C-865, C-866, C-867, C-880, C-880K005, C-884, C-887, E-712, E-861, E-871, F-206, Hydra, Pollux, M-8X0, Mercury, Mercury_GCS
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. C-887, F-206, M-8X0: Check HLP?/HELP answer to find out if RON? is supported. Only for GCS syntax version = GCS 2.0 or higher (Check with CSV?.vi. If CSV?.vi is not supported, syntax version is GCS 1.0).

2.8.9. SSL.vi (Limits.llb)

Valid for	C-702, C-844, C-848, C-880, C-887, E-755, F-206, M-8X0
Input	System number (1), Without axis ID? (F), Affected axes (empty string array), Soft limit mode (Empty bool. array, F), Error in (no error) C-702: Without axis ID = FALSE C-844: Without axis ID = FALSE C-848: Without axis ID = FALSE C-880: Without axis ID = FALSE E-755: Without axis ID = FALSE. Command not available for E-755.101. C-887, F-206, M-8X0: Without axis ID = FALSE. Check HLP?/HELP answer to find

out if command is supported.

Output Error out

Remarks Sets soft limits on or off (set by NLM.vi and PLM.vi) for specified axes. If Without axis ID is TRUE, then Affected axes is ignored and first field of Soft limit mode array is used.

2.8.10. SSL?.vi (Limits.llb)

Valid for C-702, C-844, C-848, C-880, C-887, E-755, F-206, M-8X0

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-844: 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

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

C-887, F-206, M-8X0: If All axes? = TRUE, then Axis identifier? can be FALSE. Check HLP?/HELP answer to find out if command is supported.

Output Soft limit status?, Error out

Remarks Returns soft-limit status for queried axes.

2.8.11. TMN?.vi (Limits.llb)

Valid for Analog systems, C-413, C-702, C-843, C-843.PM, C-844, C-848, C-865, C-866, C-867, C-880, C-884, C-887, E-517, E-709, E-710, E-712, E-725, E-753, E-755, E-761, E-861, E-871, F-206, Hydra, Pollux, M-8X0, Mercury, Mercury_GCS. 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-413, C-702, C-848, C-865, C-866, C-867, C-880, C-884, C-887, E-517, E-709, E-712, E-725, E-753, E-761, E-861, E-871, F-206, Hydra, Pollux, M-8X0, Mercury, Mercury_GCS: 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-413, C-867, C-884, E-871, Hydra, Pollux, Mercury_GCS: 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).

C-887, F-206, M-8X0: Check HLP?/HELP answer to find out if TMN? is supported. Only for GCS syntax version = GCS 2.0 or higher (Check with CSV?.vi. If CSV?.vi is not supported, syntax version is GCS 1.0). TMN? returns the minimum travel range of the axis with all other axes positions being zero. If this is not the case, the available travel range may be less.

2.8.12. TMX?.vi (Limits.llb)

Valid for: Analog systems, C-413, C-702, C-843, C-843.PM, C-844, C-848, C-865, C-866, C-867, C-880, C-884, C-887, E-517, E-709, E-710, E-712, E-725, E-753, E-755, E-761, E-861, E-871, F-206, Hydra, Pollux, M-8X0, Mercury, Mercury_GCS. 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-413, C-702, C-848, C-865, C-866, C-867, C-880, C-884, C-887, E-517, E-709, E-712, E-725, E-753, E-761, E-861, E-871, F-206, Hydra, Pollux, M-8X0, Mercury, Mercury_GCS: 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-413, C-867, C-884, E-871, Hydra, Pollux, Mercury_GCS: 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).

C-887, F-206, M-8X0: Check HLP?/HELP answer to find out if TMX? is supported. Only for GCS syntax version = GCS 2.0 or higher (Check with CSV?.vi. If CSV?.vi is not supported, syntax version is GCS 1.0). TMX? returns the maximum travel range of the axis with all other axes positions being zero. If this is not the case, the available travel range may be less.

2.8.13. TRA?.vi (Limits.llb)

Valid for: C-887, F-206, M-8X0

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

C-887, F-206, M-8X0: Check HLP?/HELP answer to find out if TRA? is supported. Only for GCS syntax version = GCS 2.0 or higher (Check with CSV?.vi. If CSV?.vi is not supported, syntax version is GCS 1.0).

Output	Max. position, Error out
Remarks	Returns the maximum absolute position which can be reached from the current position in the given direction. The direction is defined by a vector (X, Y, Z, U, V, W) = (x, y, z, u, v, w). Softlimits are taken into account. In case the Hexapod is in motion, an error will be set.

2.8.14. TRS?.vi (Limits.llb)

Valid for	C-413, C-867, C-884, C-887, E-712, E-755, E-861, E-871, F-206, Hydra, Pollux, M-8X0, Mercury_GCS
Input	System number (1), Axes to query (empty string array), All axes? (F), Axis identifier? (T), Error in (no error) C-413, C-867, C-884, C-887, E-755, E-861, E-871, F-206, Hydra, Pollux, M-8X0, Mercury_GCS: 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. C-887, F-206, M-8X0: Check HLP?/HELP answer to find out if TRS? is supported. Only for GCS syntax version = GCS 2.0 or higher (Check with CSV?.vi. If CSV?.vi is not supported, syntax version is GCS 1.0).

2.9. Macro Functions (“Macros.llb”)

2.9.1. #8.vi (Macros.llb)

Valid for	C-702, C-848, C-867, C-880, C-884, C-887, E-516, E-517, E-816, E-861, E-871, F-206, M-8X0, Mercury_GCS
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. C-887, F-206, M-8X0: Check HLP?/HELP answer to find out if #8 is supported. Only for GCS syntax version = GCS 2.0 or higher (Check with CSV?.vi. If CSV?.vi is not supported, syntax version is GCS 1.0).

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

Valid for	C-702, C-844, C-848, C-867, C-880, C-884, C-887, E-517, E-816, E-861, E-871, F-206, M-8X0, Mercury, Mercury_GCS
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.

C-887, M-8X0: Check HLP?/HELP answer to find out if MAC is supported. Only for GCS syntax version = GCS 2.0 or higher (Check with CSV?.vi. If CSV?.vi is not supported, syntax version is GCS 1.0).

2.9.3. MAC BEG.vi (Macros.llb)

Valid for: C-702, C-844, C-848, C-867, C-880, C-884, C-887, E-516, E-517, E-816, E-861, E-871, F-206, M-8X0, Mercury, Mercury_GCS

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

C-702, C-848, C-867, C-880, C-884, C-887, E-516, E-517, E-816, E-871, F-206, M-8X0, Mercury_GCS: 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.

C-887, M-8X0: Check HLP?/HELP answer to find out if MAC BEG is supported. Only for GCS syntax version = GCS 2.0 or higher (Check with CSV?.vi. If CSV?.vi is not supported, syntax version is GCS 1.0).

2.9.4. MAC DEF.vi (Macros.llb)

Valid for: C-867, C-884, C-887, E-516, E-517, E-816, E-861, E-871, F-206, M-8X0, Mercury_GCS

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.

C-887, F-206, M-8X0: Check HLP?/HELP answer to find out if MAC DEF is supported. Only for GCS syntax version = GCS 2.0 or higher (Check with

CSV?.vi. If CSV?.vi is not supported, syntax version is GCS 1.0).

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

Valid for	C-867, C-884, C-887, E-516, E-517, E-816, E-861, E-871, F-206, M-8X0, Mercury_GCS
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.

C-887, F-206, M-8X0: Check HLP?/HELP answer to find out if MAC DEF? is supported. Only for GCS syntax version = GCS 2.0 or higher (Check with CSV?.vi. If CSV?.vi is not supported, syntax version is GCS 1.0).

2.9.6. MAC DEL.vi (Macros.llb)

Valid for	C-702, C-844, C-848, C-867, C-880, C-884, C-887, E-516, E-517, E-816, E-861, E-871, F-206, M-8X0, Mercury, Mercury_GCS
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.

C-887, M-8X0: Check HLP?/HELP answer to find out if MAC DEL is supported. Only for GCS syntax version = GCS 2.0 or higher (Check with CSV?.vi. If CSV?.vi is not supported, syntax version is GCS 1.0).

2.9.7. MAC END.vi (Macros.llb)

Valid for	C-702, C-844, C-848, C-867, C-880, C-884, C-887, E-516, E-517, E-816, E-861, E-871, F-206, M-8X0, Mercury, Mercury_GCS
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.

C-887, M-8X0: Check HLP?/HELP answer to find out if MAC END is supported. Only for GCS syntax version = GCS 2.0 or higher (Check with CSV?.vi. If CSV?.vi is not supported, syntax version is GCS 1.0).

2.9.8. MAC ERR?.vi (Macros.llb)

Valid for	C-884, C-887, E-871, F-206, Mercury_GCS, M-8X0
Input	System number (1), Error in (no error)
Output	Error information, Error out
Remarks	Get error information from macro execution.
	C-887, F-206, M-8X0: Check HLP?/HELP answer to find out if MAC ERR? is supported.

2.9.9. MAC FREE?.vi (Macros.llb)

Valid for	C-867, C-887, E-516, E-517, E-816, F-206, M-8X0
Input	System number (1), Error in (no error)
Output	No. of characters, Error out
Remarks	Returns amount of free memory for macro recording (in number of characters).
	C-867: Check HLP? answer to find out if MAC FREE? is supported.
	E-816: This command cannot be issued to a slave. Check controller manual to find out if MAC FREE? is supported.
	C-887, F-206, M-8X0: Check HLP?/HELP answer to find out if MAC FREE? is supported. Only for GCS syntax version = GCS 2.0 or higher (Check with CSV?.vi. If CSV?.vi is not supported, syntax version is GCS 1.0).

2.9.10. MAC NSTART.vi (Macros.llb)

Valid for	C-702, C-867, C-884, C-887, E-516, E-517, E-816, E-861, E-871, F-206, M-8X0, Mercury, Mercury_GCS
Input	System number (1), Macro name (empty string), N (1), Arguments (empty string), 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. Arguments is not supported.
	C-887, F-206, M-8X0: Check HLP?/HELP answer to find out if MAC NSTART is supported. Only for GCS syntax version = GCS 2.0 or higher (Check with CSV?.vi. If CSV?.vi is not supported, syntax version is GCS 1.0). Use #8.vi to determine when macro execution has finished.

2.9.11. MAC START.vi (Macros.llb)

Valid for	C-702, C-844, C-848, C-867, C-880, C-884, C-887, E-516, E-517, E-816, E-861, E-871, F-206, M-8X0, Mercury, Mercury_GCS
Input	System number (1), Macro name (empty string), Arguments (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 E-871,: Use #8.vi to determine when macro execution has finished. For C-702, C-848, C-880, E-516 and E-517 Arguments is not supported.

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. Arguments is not supported.

F-206: Check HLP?/HELP answer to find out if MAC START and #8 are supported. If yes, use #8.vi to determine when macro execution has finished.

C-887, M-8X0: Only for GCS syntax version = GCS 2.0 or higher (Check with CSV?.vi. If CSV?.vi is not supported, syntax version is GCS 1.0). Use #8.vi to determine when macro execution has finished.

Mercury: #8.vi is not supported.

2.9.12. MAC?.vi (Macros.llb)

Valid for	C-702, C-844, C-848, C-867, C-880, C-884, C-887, E-516, E-517, E-816, E-861, E-871, F-206, M-8X0, Mercury, Mercury_GCS
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.

C-887, M-8X0: Check HLP?/HELP answer to find out if MAC? is supported. Only for GCS syntax version = GCS 2.0 or higher (Check with CSV?.vi. If CSV?.vi is not supported, syntax version is GCS 1.0).

2.9.13. RMC?.vi (Macros.llb)

Valid for	C-867, C-884, C-887, E-861, E-871, F-206, M-8X0, Mercury_GCS
Input	System number (1), Error in (no error)
Output	Running macros, Error out
Remarks	List macros currently running.

2.9.14. VAR.vi (Macros command.llb)

Valid for	C-884, C-887, E-871, F-206, M-8X0, Mercury_GCS
Input	System number (1), Variable name (empty string), Value (empty string), Error in (no error)
Output	Error out
Remarks	Sets a variable to a certain value. A variable is present in RAM only. See controller User Manual for details regarding local and global variables.

C-887, F-206, M-8X0: Check HLP?/HELP answer to find out if VAR is supported. For GCS syntax version = GCS 2.0 or higher (Check with CSV?.vi. If CSV?.vi is not supported, syntax version is GCS 1.0).

Mercury_GCS: Check HLP? answer to find out if VAR? is supported.

2.9.15. VAR?.vi (Macros command.llb)

Valid for	C-884, C-887, E-871, F-206, M-8X0, Mercury_GCS
Input	System number (1), Variable name (empty string array), Error in (no error)
Output	Value, Error out

Remarks	Returns the value of given variable name(s). See controller User Manual for details regarding local and global variables. C-887, F-206, M-8X0: Check HLP?/HELP answer to find out if VAR? is supported. For GCS syntax version = GCS 2.0 or higher (Check with CSV?.vi. If CSV?.vi is not supported, syntax version is GCS 1.0). Mercury_GCS: Check HLP? answer to find out if VAR? is supported.

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

2.10.1. #5_old.vi (Old commands.llb)

Valid for	C-887, 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 C-887/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".

2.10.2. Define connected systems.vi (Old commands.llb)

Valid for	C-843, C-843.PM, C-844, C-848, C-865, C-880, C-880K005, E-516, E-710, E-761, E-816, F-206, M-8X0, Mercury
Input	Controller names (cluster of 4 Enum controls, none), Change only one system? (F), System number (1), Error in (no error)
Output	Controller names out, Error out
Remarks	Defines connected systems and writes controller names into Global2.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 four entries from Controller names are written into "Global2.vi". If Change only one system? is TRUE, only the entry for the given system number is overwritten in "Global2.vi". Old VI – only for compatibility reasons available. Limited to 4 systems. Use Define connected systems (Array).vi instead.

2.10.3. HELP.vi (Old commands.llb)

Valid for	C-887, F-206, M-8X0
Input	System number (1), Error in (no error)
Output	Help string, Error out
Remarks	Returns help string. C-887, F-206, M-8X0: Only for GCS syntax version = GCS 1.0 (Check with CSV?.vi. If CSV?.vi is not supported, syntax version is GCS 1.0). For GCS 2.0 use HLP?.vi instead.

2.10.4. INI hexaxes and wait until finished.vi (Old commands.llb)

Valid for	C-887, F-206, M-8X0
Input	System number (1), Timeout (sec, 90 sec), INI hexapod? (FALSE), INI sep. axes (empty string array), Stop refnum (F), Local stop (F), Error in (no error)
Output	Hexapod initialized?, Sep. axes initialized?, Error out
Remarks	Initializes hexapod and/or additional axes and waits until procedure has finished, Stop refnum or Local stop is TRUE or a timeout condition has occured. When using as a sub-VI, use Stop refnum to stop VI from caller. C-887, F-206, M-8X0: Check HLP?/HELP answer to find out if INI is supported. Only for GCS syntax version = GCS 1.0 (Check with CSV?.vi. If CSV?.vi is not supported, syntax version is GCS 1.0). GCS 1.0: Since version 4.0.0 of the F-206 firmware and version 4.5.0 of the M-8X0 firmware it is possible to stop the ini procedure by sending #24. GCS 2.0 or higher: VI calls FRF instead for compatibility.

2.10.5. MOV?_old.vi (Old commands.llb)

Valid for	C-887, F-206, M-8X0
Input	System number (1), Stop refnum (F), Local stop (F), Error in (no error)
Output	Moving?, Error out
Remarks	Sends MOV? and waits for response of 0 (not moving). In case of no answer, VI stops after 60 s. VI also stops if Local stop or Stop refnum is TRUE. When using as a sub-VI, use Stop refnum to stop VI from caller. C-887, F-206, M-8X0: Only for GCS syntax version = GCS 1.0 (Check with CSV?.vi. If CSV?.vi is not supported, syntax version is GCS 1.0).

2.10.6. STOP.vi (Old commands.llb)

Valid for	C-887, F-206, M-8X0
Input	System number (1), Error in (no error)
Output	Error out
Remarks	Stops single motion (to stop INI or fast scan routines use #24). C-887, F-206, M-8X0: Check HLP?/HELP answer to find out if STOP is supported. Only for GCS syntax version = GCS 1.0 (Check with CSV?.vi. If CSV?.vi is not supported, syntax version is GCS 1.0). For GCS 2.0 or higher, use STP.vi instead.

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

Valid for	C-887, 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. C-887, 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".

2.11. Commands for Optical or Analog Signals ("Optical or Analog Input.llb")

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

Valid for	C-865, C-866, C-867, C-880, C-884, C-887, E-761, E-861, E-871, F-206, M-8X0, Mercury, Mercury_GCS
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.

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

Valid for	C-887, F-206 (but must be present for C-866, C-867, C-880, C-884, E-761, E-861, M-8X0 and Mercury_GCS 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".

2.11.3. NAV.vi (Optical or Analog Input.llb)

Valid for	C-880, C-887, F-206, M-8X0
Input	System number (1), Board (1), Number of readings (1), Error in (no error)
Output	Hidden error (T/F), Error out
Remarks	Sets averaging count for analog readings. Hidden error is TRUE if selected system reports error code ≠ 0. For large count values, the communication timeout value may need to be increased to prevent a timeout error during TAV? Queries.

2.11.4. NAV?.vi (Optical or Analog Input.llb)

Valid for	C-880, C-887, F-206, M-8X0
Input	System number (1), Board (1), Error in (no error)
Output	Number of readings, Error out
Remarks	Returns averaging-count setting (used for analog readings).

2.11.5. SGA.vi (Optical or Analog Input.llb)

Valid for	C-880, C-887, F-206, M-8X0, but must be present for C-884, too.
Input	System number (1), Board (1), Gain (10), Error in (no error)
Output	Hidden error (T/F), Error out
Remarks	Sets the gain value for the optical input. See User Manual for allowable gain values. Controller error is TRUE if selected system reports error code ≠ 0.

2.11.6. SGA?.vi (Optical or Analog Input.llb)

Valid for	C-880, C-887, F-206, M-8X0, but must be present for C-884, too.
Input	System number (1), Board (1), Error in (no error)
Output	Gain, Error out
Remarks	Returns gain-value setting for specified optical input.

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

Valid for	C-702, C-866, C-867, C-880, C-884, C-887, E-861, E-871, F-206, M-8X0, Mercury, Mercury_GCS
Input	System number (1), Error in (no error)
Output	Analog channels, Error out
Remarks	Returns the number of installed analog channels. C-887, F-206, M-8X0: Check HLP?/HELP answer to find out if TAC? is supported. Only for GCS syntax version = GCS 2.0 or higher (Check with CSV?.vi. If CSV?.vi is not supported, syntax version is GCS 1.0).

2.11.8. TAD?.vi (Optical or Analog Input.llb)

Valid for	C-413, C-887, E-517, E-709, E-710, E-712, E-725, E-753, E-755, E-761, E-870, F-206, M-8X0
Input	System number (1), Sensors to query (empty string array), All sensors? (F), Sensor identifier? (T), Error in (no error)

E-710: If All sensors? = TRUE, then Sensor identifier? must be TRUE.

C-413, C-887, E-517, E-709, E-712, E-725, E-753, E-870, F-206, M-8X0: If All sensors? = TRUE, then Sensor identifier? must be FALSE.

E-755: If All sensors? = TRUE, then Sensor identifier? must be FALSE. Command not available for E-755.101.

E-761: Sensors to query can be 1 to 3. If All sensors? = TRUE, then Sensor identifier? can be FALSE.

Output AD value, Error out

Remarks Returns AD value for the specified sensor number.

C-413: Sensors to query comprises all ADC channels of the device: can be a "genuine" sensor (capacitive sensor integrated in the mechanics) or a "general purpose" analog input.

E-517: The input signal channels to be queried with TAD? are the sensor channels of the piezo control electronics, IDs = 1 to 3 (actually available IDs depend on the response to TSC?).

E-709: Sensors to query comprises all ADC channels of the device: can be the "genuine" sensor (sensor integrated in the mechanics, identifier = 1) and the "general purpose" analog input (identifier = 2).

E-710: Command is available for 4-channel version only and in command level 1 only (see “CCL.vi”, “CCL?.vi”)

E-712: Sensors to query comprises all ADC channels of the device: can be a "genuine" sensor (capacitive sensor integrated in the mechanics) or a "general purpose" analog input.

E-725: Sensors to query comprises all ADC channels of the device: can be a "genuine" sensor (capacitive sensor integrated in the mechanics) or a "general purpose" analog input.

E-753: Sensors to query comprises all ADC channels of the device: can be the "genuine" sensor (capacitive sensor integrated in the mechanics, identifier = 1) and the "general purpose" analog input (identifier = 2).

E-870: Sensors to query comprises all ADC channels of the device: can be the analog command input (identifier = 1) or other internal on board sources for diagnosis information (identifier = 2…5).

C-887, F-206, M-8X0: Check HLP?/HELP answer to find out if TAD? is supported. Only for GCS syntax version = GCS 2.0 or higher (Check with CSV?.vi. If CSV?.vi is not supported, syntax version is GCS 1.0).

2.11.9. TAV.vi (Optical or Analog Input.llb)

Valid for C-880, C-887, F-206, M-8X0

Input System number (1), Board (1), Set (0:Value), Analog value (0), Range (0), Power unit (µW), (Error in (no error)

C-880: Set = Value

C-887, F-206, M-8X0: Set can be Range or Power unit.

Output Hidden error, Error out

Remarks Sets analog value, range for optical head or power unit for optical head, depending on Set.

C-887, F-206, M-8X0: Check HLP?/HELP answer to find out if TAV is supported. Only for GCS syntax version = GCS 1.0 (Check with CSV?.vi. If CSV?.vi is not supported, syntax version is GCS 1.0).

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

Valid for: C-865, C-866, C-867, C-880, C-884, C-887, E-761, E-861, E-871, F-206, M-8X0, Mercury, Mercury_GCS

Input: System number (1), Board (1), Query (Value), Error in (no error)

C-865, C-866: Query = Value, Board = 1.

C-867, C-884, E-861, E-871, Mercury_GCS: Query = Value, Board = 1 to 4.

C-880: Query = Value.

E-761: Query = Value, Board = 4.

C-887, F-206, M-8X0: For GCS syntax version = GCS 1.0 (Check with CSV?.vi. If CSV?.vi is not supported, syntax version is GCS 1.0), Query can be Value, Range or Power unit. For GCS syntax version = GCS 2.0 or higher, Query = Value.

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, C-884, E-761, E-861, E-871, Mercury, Mercury_GCS: 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.

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

Valid for: C-413, C-887, 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, E-861, E-870, E-871, F-206, M-8X0 and Mercury_GCS 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.

C-413, 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.

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

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

Valid for	Analog systems, C-413, C-702, C-866, C-867, C-880, C-884, C-887, E-517, E-709, E-710, E-712, E-725, E-753, E-755, E-761, E-861, E-871, F-206, M-8X0, Mercury_GCS. 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. Sub-VI for “Show_Save_Load_XY_Data.vi”

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

Valid for	C-866, C-867, C-880, C-884, C-887, E-761, E-861, E-871, F-206, M-8X0, Mercury_GCS
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.

2.12.3. Calculate 2D linear spiral.vi (Scan support.llb)

Valid for	C-880, C-887, F-206, M-8X0
Input	Start position ax1 (0), Start position ax2 (0), Range (0), Step size (0)
Output	Min. position ax1, Max. position ax1, No. of steps, 1D pos. array ax1, Min. position ax2, Max. position ax2, 1D pos. array ax2, 2D intensity array
Remarks	Calculates position values for a 2D linear spiral scan and initializes intensity array according to Start position of axes 1 and 2, Step size and Range. The VI returns Minimum position, Maximum position, No. of steps and initialized 1D position array for axes 1 and 2 and a 2D intensity array.

2.12.4. Calculate 2D meander.vi (Scan support.llb)

Valid for	C-880, C-887, F-206, M-8X0
Input	Start position ax1 (0), Start position ax2 (0), Range ax1 (0), Range ax2 (0), Step size (0), Direction ax1 (TRUE: +), Start direction ax2 (TRUE: +)
Output	Min. position ax1, Max. position ax1, No. of steps, 1D pos. array ax1, Min. position ax2, Max. position ax2, 1D pos. array ax2, 2D intensity array
Remarks	Calculates position values for a 2D linear meander scan and initializes intensity array according to start position of axes 1 and 2, Step size, Range of axes 1 and 2 and scan directions of axes 1 and 2. The VI returns Minimum position, Maximum position, No. of steps and initialized 1D position array for axes 1 and 2 and a 2D intensity array.

2.12.5. Calculate 2D scan positions.vi (Scan support.llb)

Valid for	C-880, C-884, C-887, E-761, F-206, M-8X0
Input	Scan direction (0: Scan from middle), Start position ax1 (0), Range ax1(0), Step size ax1(0), Start position ax2 (0), Range ax2(0), Step size ax2(0)
Output	Min. position ax1, Max. position ax1, No. of steps ax1, 1D pos. array ax1, Min. position ax2, Max. position ax2, No. of steps ax2, 1D pos. array ax2, 2D intensity array
Remarks	Calculates 2D scan positions according to Scan direction, Start position, Range and Step size of axes 1 and 2 and returns Minimum position, Maximum position, No. of steps and initialized 1D position array for axes 1 and 2 and a 2D intensity array. See “2D Scan.vi” for possible Scan directions.

2.12.6. Decrease Gain?.vi (Scan support.llb)

Valid for	C-880, C-887, F-206, M-8X0
Input	System no. (1), Board (1), I(in) (0), Gain for I(in) (10), Level (6), Error in (no error)
Output	I(out), Gain for I(out), Controller error, Error out
Remarks	Decreases gain if intensity is higher than Level.

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

Valid for	C-866, C-867, C-880, C-884, C-887, E-761, E-861, E-871, F-206, M-8X0, Mercury_GCS
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.

2.12.8. PIGraph3D_DLL_Functions.vi (Scan support.llb)

Valid for	C-880, C-884, C-887, F-206, M-8X0, E-761
Input	ID in (1), Window title (empty string), Function (Get ID), ColourMap (Spectrum), File path (empty string), Projections (Plot only), Autoscale? (T), Show grid (T), Orthogonal? (T), No. of digits (Empty num. array, 3), Datastream (empty string), Error in (no error)
Output	ID out, autorange, Error out
Remarks	This vi calls a given function from GCSTranslator.dll. GCSTranslator.dll must be installed, and wxPIGraph3D.dll must be installed in PI\GCSTranslator.

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

2.13.1. #11.vi (Special command.llb)

Valid for	C-887, F-206, M-8X0
Input	System number (1), Error in (no error)
Output	Free trajectory points, Error out
Remarks	Returns the number of free memory points for trajectory definition (e.g. for a dynamic move) by sending the single ASCII character 11.

C-887, F-206, M-8X0: Check HLP?/HELP answer to find out if #11 is supported. Only for GCS syntax version = GCS 2.0 or higher (Check with CSV?.vi. If CSV?.vi is not supported, syntax version is GCS 1.0).

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

Valid for	Analog systems, C-413, C-702, C-843, C-843.PM, C-844, C-848, C-865, C-866, C-867, C-880, C-880K005, C-884, C-887, E-516, E-517, E-709, E-712, E-725, E-753, E-755, E-761, E-816, E-861, E-870, E-871, F-206, Hydra, Pollux, M-8X0, Mercury, Mercury_GCS (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.

C-887, 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.

2.13.3. #27.vi (Special command.llb)

Valid for	C-702, C-848, C-880, C-887, F-206, M-8X0
Input	System number (1), Error in (no error)
Output	Error out (T, -1, System halted)
Remarks	Sends the single ASCII character 27 (ESC, system abort) and sets error flag.

C-887, 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.

2.13.4. #3.vi (Special command.llb)

Valid for	C-887, F-206, M-8X0
Input	System number (1), Error in (no error)
Output	Queried axes out, Position, Error out
Remarks	Returns position information by sending the single ASCII character 3. Command is equivalent to POS?.

F-206: Check HLP?/HELP answer to find out if #3 is supported. Only for GCS syntax version = GCS 2.0 or higher (Check with CSV?.vi. If CSV?.vi is not supported, syntax version is GCS 1.0).

C-887, M-8X0: Check HLP?/HELP anwer to find out if #3 is supported. For GCS syntax version = GCS 1.0 (Check with CSV?.vi. If CSV?.vi is not supported, syntax version is GCS 1.0), reports target position of hexapod axes X, Y, Z, U, V, W. Only for controllers based on C-842.80.
For GCS syntax version = GCS 2.0 or higher, reports current position.

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

Valid for	C-867, C-884, C-887, E-861, E-871, F-206, Hydra, Pollux, M-8X0, Mercury_GCS
Input	System number (1), Error in (no error)
Output	Status information, Status information (txt), Error out
Remarks	Request status information by sending the single ASCII character 4. Answer is controller specific. See controller User Manual for decoding.

F-206: Check HLP?/HELP answer to find out if #4 is supported. Only for GCS syntax version = GCS 2.0 or higher (Check with CSV?.vi. If CSV?.vi is not supported, syntax version is GCS 1.0).

C-887, M-8X0: Check HLP?/HELP answer to find out if #4 is supported. Only for controllers based on C-842.80 or GCS syntax version = GCS 2.0 or higher (Check with CSV?.vi. If CSV?.vi is not supported, syntax version is GCS 1.0).

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

Valid for	Analog systems, C-413, C-702, C-843, C-843.PM, C-844, C-848, C-865, C-866, C-867, C-880, C-884, C-887, E-517, E-709, E-712, E-725, E-753, E-755, E-761, E-861, E-870, E-871, Hydra, Pollux, Mercury, Mercury_GCS (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 conn. axes, not for single axes.

F-206: For GCS syntax version = GCS 1.0 (Check with CSV?.vi. If CSV?.vi is not supported, syntax version is GCS 1.0), coding in answer is different, please use #5_old.vi.

M-8X0: For GCS syntax version = GCS 1.0 (Check with CSV?.vi. If CSV?.vi is not supported, syntax version is GCS 1.0), coding in answer is different, please use #5_old.vi.

2.13.7. #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, C-884, E-861, E-871 and Mercury_GCS also)

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

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

F-206: For GCS syntax version = GCS 1.0 (Check with CSV?.vi. If CSV?.vi is not supported, syntax version is GCS 1.0), only first field of Pos change? array is valid, refers to whole system.

C-887, M-8X0: For GCS syntax version = GCS 1.0 (Check with CSV?.vi. If CSV?.vi is not supported, syntax version is GCS 1.0), 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.

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

Valid for C-413, C-702, C-843, C-843.PM, C-844, C-848, C-865, C-866, C-867, C-880, C-880K005, C-884, C-887, E-517, E-709, E-710, E-712, E-725, E-755, E-761, E-861, E-870, E-871, Hydra, Pollux, Mercury, Mercury_GCS (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

F-206: Check HLP?/HELP answer to find out if #7 is supported. Only for GCS syntax version = GCS 2.0 or higher (Check with CSV?.vi. If CSV?.vi is not supported, syntax version is GCS 1.0).

C-887, M-8X0: Check HLP?/HELP answer to find out if #7 is supported. Only for GCS syntax version = GCS 2.0 or higher (Check with CSV?.vi. If CSV?.vi is not supported, syntax version is GCS 1.0).

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.

2.13.9. CCL.vi (Special command.llb)

Valid for C-413, C-867, C-887, E-517, E-709, E-710, E-712, E-725, E-753, E-755, E-870, E-871, F-206, M-8X0

Input System number (1), Password (100), Command level (0), Error in (no error)

C-413, C-867, E-517, E-709, E-710, E-712, E-725, E-753, E-755, E-870, E-871, F-206, M-8X0: Command level can be 0 (only commands needed for normal operation are available) or 1 (all commands from command level 0 plus special commands for advanced users are available). Password for CCL 1 is “ADVANCED”.

Output Error out, Hidden error

Remarks If password is correct, this vi sets the command level of the controller and queries ERR?. Hidden error is TRUE if selected system reports error code ≠ 0. User “HLP?.vi” to determine which commands are available in the current command level.

C-867: Check HLP? answer to find out if CCL is supported.

C-887, F-206, M-8X0: Check HLP?/HELP answer to find out if CCL is supported. Only for GCS syntax version = GCS 2.0 or higher (Check with CSV?.vi. If CSV?.vi is not supported, syntax version is GCS 1.0).

2.13.10. CCL?.vi (Special command.llb)

Valid for C-413, C-702, C-867, C-887, E-517, E-709, E-710, E-712, E-725, E-753, E-755, E-816, E-870, E-871, F-206, M-8X0

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

Output Command level, Error out

Remarks Returns the current command level.

C-867: Check HLP? answer to find out if CCL? is supported.

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

C-887, F-206, M-8X0: Check HLP?/HELP answer to find out if CCL? is supported. Only for GCS syntax version = GCS 2.0 or higher (Check with CSV?.vi. If CSV?.vi is not supported, syntax version is GCS 1.0).

2.13.11. CST.vi (Special command.llb)

Valid for C-702, C-843, C-843.PM, C-844, C-848, C-865, C-866, C-880, C-884, C-887, E-710, E-761, E-871, F-206, Hydra, Pollux, M-8X0, Mercury

Input System number (1), Axis ID’s (empty string array), Stage names (empty string array), Error in (no error)

C-887, F-206, M-8X0: Only valid for separate axes A and/or B.

Output Error out

Remarks Assigns axes to stages and queries “ERR?”. With this command the stage assignment of the connected axes can be changed. Valid stage names can be listed with VST?.vi.

C-884: Command is available via USB interface only.

E-761: The settings are automatically written to non-volatile memory.
Valid stage names are "ID-STAGE for configured axes (a stage should be connected) and "NOSTAGE" for non-configured axes (no stage should be connected). The axis configuration as "ID-Stage" is required before you can address any move command to this axis (i.e. to the connected stage).

E-871: Command is available via USB, USB DaisyChain or RS232 DaisyChain interface only.

C-887, F-206, M-8X0: Check HLP?/HELP answer to find out if CST is supported. Only for GCS syntax version = GCS 2.0 or higher (Check with CSV?.vi. If CSV?.vi is not supported, syntax version is GCS 1.0).

Pollux: See HydraPollux_GCS_DLL Manual for specialities of the Pollux controller family regarding stage name settings.

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

Valid for C-413, C-702, C-843, C-843.PM, C-844, C-848, C-865, C-866, C-867, C-880, C-884, C-887, E-709, E-710, E-712, E-725, E-753, E-755, E-761, E-861, E-871, F-206, Hydra, Pollux, M-8X0, Mercury, Mercury_GCS

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, C-887, F-206, M-8X0: If All axes? = TRUE, then Axis identifier? must be TRUE

C-413, C-865, C-866, C-867, C-884, E-709, E-710, E-712, E-725, E-753, E-755, E-761, E-861, E-871, Hydra, Pollux, Mercury, Mercury_GCS: 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.

C-887, F-206, M-8X0: Check HLP?/HELP answer to find out if CST? is supported.

Pollux: See HydraPollux_GCS_DLL Manual for specialities of the Pollux controller family regarding stage name settings.

2.13.13. DEL.vi (Special command.llb)

Valid for C-702, C-848, C-880, C-884, C-887, E-516, E-517, E-816, E-871, F-206, M-8X0, Mercury

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

All systems: Delay time unit is ms.

Output Error out

Remarks Delays the command interpreter of specified system for given Delay time. DEL is used within macros primarily. Do not mistake MAC DEL which deletes macros for DEL which delays.

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

C-887, F-206, M-8X0: Check HLP?/HELP answer to find out if DEL is supported. Only for GCS syntax version = GCS 2.0 or higher (Check with CSV?.vi. If CSV?.vi is not supported, syntax version is GCS 1.0).

2.13.14. DIO.vi (Special command.llb)

Valid for C-413, C-702, C-843, C-843.PM, C-848, C-865, C-866, C-867, C-880, C-884, C-887, E-861, E-871, F-206, Hydra, Mercury, Mercury_GCS, M-8X0

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-413: DO mode format is Boolean, DO's to command can be 1 to 5. DO pattern is not valid.

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, E-871, Mercury_GCS: For DO mode format = Boolean, DO's to command can be 1 to 4. Only one DO per command call is 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-884: For DO mode format = Boolean, DO's to command can be 1 to 4. Up to four DO's per command call are 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.

C-887, F-206, M-8X0: For DO mode format = Boolean, DO's to command can be 1 to 8. 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). Check HLP?/HELP answer to find out if DIO is supported. Only for GCS syntax version = GCS 2.0 or higher (Check with CSV?.vi. If CSV?.vi is not supported, syntax version is GCS 1.0).

Hydra: For DO mode format = Boolean, DO's to command can be 1-3 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).

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.

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

Valid for C-413, C-702, C-843, C-843.PM, C-848, C-865, C-866, C-867, C-880, C-884, C-887, E-517, E-709, E-761, E-861, E-871, F-206, Hydra, Mercury, Mercury_GCS, M-8X0

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-413: If All DI's = TRUE, then DI identifier must be FALSE. Invert order for TVI? is not valid. DI's to query are 1 to 4. Query pattern? is not valid.

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, C-884, E-861, E-871, Mercury_GCS: If All DI's = TRUE, then DI identifier must be FALSE. Invert order for TVI? is not valid. DI's to query are 1 to 4. Query pattern? can be TRUE.

C-887, F-206, M-8X0: If All DI's = TRUE, then DI identifier must be FALSE. Invert order for TVI? is not valid. DI's to query are 1 to 8. Query pattern? can be TRUE. Check HLP?/HELP answer to find out if DIO? is supported. Only for GCS syntax version = GCS 2.0 or higher (Check with CSV?.vi. If CSV?.vi is not supported, syntax version is GCS 1.0).

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-709: If All DI's = TRUE, then DI identifier can be FALSE. Invert order for

TVI? is not valid. Query pattern? is not valid.

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

Hydra: If All DI's = TRUE, then DI identifier must be FALSE. Invert order for TVI? is not valid. DI's to query are 1-2. Query pattern? can be TRUE.

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-709: Check HLP? answer to find out if DIO? is supported.

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.

2.13.16. DPA.vi (Special command.llb)

Valid for C-887, F-206, M-8X0

Input System number (1), Password (100), Affected items (empty string array), Parameter no. format (Decimal: FALSE) (F), Parameter to reset (empty num. array), Parameter to reset (hex.) (empty hex. array), Parameter, Error in (no error)

C-887, F-206, M-8X0: If Affected items = empty array, the currently valid values of all parameters affected by the specified password are reset. Parameter no. format is FALSE (num.).

Output Error out, Controller error

Remarks If password is correct, resets parameters or settings to default values, waits 3000 ms and queries ERR?. It does not overwrite settings in the non-volatile memory. For details, see the specific documentation. If parameter no. is in decimal format, use "Parameter to reset" input, for hexadecimal parameter numbers use "Parameter to reset (hex)" input and switch "Parameter no. format" to TRUE. Do not mix decimal and hex. parameter numbers in one call. If "Affected items" is an empty array, DPA is sent without item and parameter specification. Controller error is TRUE if selected system reports error code ≠ 0.

C-887, F-206, M-8X0: Check HLP?/HELP answer to find out if SPA is supported. Only for GCS syntax version = GCS 2.0 or higher (Check with CSV?.vi. If CSV?.vi is not supported, syntax version is GCS 1.0).

2.13.17. DRC.vi (Special command.llb)

Valid for C-413, C-702, C-843, C-866, C-867, C-884, C-887, E-517, E-709, E-710, E-712, E-725, E-753, E-755, E-861, E-871, F-206, M-8X0, Mercury_GCS

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

C-413, C-843, C-866, C-884, C-887, E-517, E-709, E-712, E-725, E-753, E-861, E-871, F 206, M-8X0 , Mercury_GCS: 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.

C-884: See User Manual for available record options. The C-884 has eight 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.

C-413, 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.

E-871: See User Manual for available record options.
The E-871 has two data recorder tables with 1024 points per table.

C-887, F-206, M-8X0: Check HLP?/HELP answer to find out if DRC is supported. Only for GCS syntax version = GCS 2.0 or higher (Check with CSV?.vi. If CSV?.vi is not supported, syntax version is GCS 1.0). See User Manual for available record options, trigger options, number of data recorder tables and points per table. The number of points can be changed by setting the appropriate parameter value, see User Manual of the controller for details.

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

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

Valid for C-413, C-702, C-843, C-866, C-867, C-884, C-887, E-517, E-709, E-710, E-712, E-725, E-753, E-755, E-861, E-871, F-206, M-8X0, Mercury_GCS

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-413, C-843, C-866. C-867, C-884, C-887, E-517, E-709, E-712, E-725, E-753, E-861, E-871, F-206, M-8X0, Mercury_GCS: 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.

C-887, F-206, M-8X0: Check HLP?/HELP answer to find out if DRC? is supported. Only for GCS syntax version = GCS 2.0 or higher (Check with CSV?.vi. If CSV?.vi is not supported, syntax version is GCS 1.0).

2.13.19. DRL?.vi (Special command.llb)

Valid for	C-413, C-867, C-884, C-887, E-709, E-712, E-725, E-753, E-871, F-206, M-8X0
Input	System number (1), Rec. table to query (empty num. array, 0), With Rec. table IDs? (F), Error in (no error)
Output	No. of rec. values (empty num. array, 0), Error out
Remarks	This VI returns the number of recorded data values for the given record tables.

C-867: Check HLP? answer to find out if DRL? is supported.

E-712, E-725, E-753: Check HLP? answer to find out if DRL? is supported.

C-887, F-206, M-8X0: Check HLP?/HELP answer to find out if DRL? is supported. Only for GCS syntax version = GCS 2.0 or higher (Check with CSV?.vi. If CSV?.vi is not supported, syntax version is GCS 1.0).

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

Valid for	Analog systems, C-413, C-702, C-843, C-866, C-867, C-884, C-887, E-517, E-709, E-710, E-712, E-725, E-753, E-755, E-761, E-861, E-871, F-206, M-8X0, Mercury_GCS. 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-413: Xo >= 1. Nmax = 4096.

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.

C-884: 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.

E-871: Xo >= 1. Nmax = 1024.

F-206: Xo >= 1. See the User Manual of the controller for Nmax default value (can be changed with SPA.vi).

C-887, M-8X0: For GCS syntax version = 1.0 (Check with CSV?.vi. If CSV?.vi is not supported, syntax version is GCS 1.0), Rec. table IDs, xo, N and Nmax are not valid and Without parameter? must be TRUE.
For GCS syntax version = 2.0, Xo >= 1. See the User Manual of the controller for Nmax default value (can be changed with SPA.vi).

Mercury_GCS: Xo >= 1. Nmax = 1024.

Output Data, Names, Sample time, Error out

Remarks Returns N recorded data points. If N is greater than Nmax, multiple queries are sent. For large Nmax values, communication timeout must be set long enough, otherwise a communication error may occur.

C-413: The 4096 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 C-413 User Manual for details.

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.

C-884: The number of tables is 8.

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.

E-871: Two data recorder tables with 1024 points per table are provided.

F-206: Check HLP?/HELP answer to find out if DRR? is supported. Only for GCS syntax version = GCS 2.0 or higher (check with CSV?.vi. If CSV?.vi is not supported, syntax version is GCS 1.0). See controller User Manual for

available data recorder tables and points.

C-887, M-8X0: Only supported if controller is based on C-842.80 board or GCS syntax version = GCS 2.0 or higher (Check with CSV?.vi. If CSV?.vi is not supported, syntax version is GCS 1.0). Check HLP?/HELP answer to find out if DRR? is supported. For GCS 1.0, returns 360 motor current values recorded during execution of DRV.
For GCS 2.0 and higher, see controller User Manual for available data recorder tables and points.

Mercury_GCS: Two data recorder tables with 1024 points per table are provided.

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

Valid for: Analog systems, C-413, C-702, C-843, C-866, C-867, C-884, C-887, E-517, E-709, E-710, E-712, E-725, E-753, E-755, E-761, E-861, E-871, F-206, M-8X0, Mercury_GCS. 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-413: Xo >= 1. Nmax = 4096.

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.

C-884: 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.

E-871: Xo >= 1. Nmax = 1024.

F-206: Xo >= 1. See C-887 User Manual for Nmax default value(can be changed with SPA.vi).

C-887, M-8X0: For GCS syntax version = 1.0 (Check with CSV?.vi. If CSV?.vi is not supported, syntax version is GCS 1.0), Rec. table IDs, xo, N and Nmax are not valid and Without parameter? must be TRUE.
For GCS syntax version = 2.0, Xo >= 1. See C-887 User Manual for Nmax default value (can be changed with SPA.vi).

Mercury_GCS: Xo >= 1. Nmax = 1024.

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. If Sample time is zero, it is set to 1.0 for displaying data in the 2D graph only.

C-413: The 4096 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 C-413 User Manual for details.

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.

C-884: The number of tables is 8.

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.

E-871: Two data recorder tables with 1024 points per table are provided.

F-206: Check HLP?/HELP answer to find out if DRR? is supported. Only for GCS syntax version = GCS 2.0 or higher (check with CSV?.vi. If CSV?.vi is not supported, syntax version is GCS 1.0). See controller User Manual for available data recorder tables and points.

C-887, M-8X0: Only supported if controller is based on C-842.80 board or GCS syntax version = GCS 2.0 or higher (Check with CSV?.vi. If CSV?.vi is not supported, syntax version is GCS 1.0). Check HLP?/HELP answer to find out if DRR? is supported. For GCS 1.0, returns 360 motor current values recorded during execution of DRV.
For GCS 2.0 and higher, see controller User Manual for available data recorder tables and points.

Mercury_GCS: Two data recorder tables with 1024 points per table are provided.

2.13.22. DRT.vi (Special command.llb)

Valid for	C-413, C-702, C-843, C-866, C-867, C-884, C-887, E-709, E-712, E-725, E-753, E-755, E-861, E-871, F-206, M-8X0, Mercury_GCS
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-413, C-843, C-867, C-884, E-709, E-861, E-871, Mercury_GCS: See User Manual for available trigger sources and values. DataRecorderTable = 0 (the specified trigger source is set for all data recorder tables).

E-712, E-725, E-753: See User Manual for available trigger sources and values. The specified trigger source is set for all data recorder tables. Check HLP? answer to find out if DRT is supported.

C-887, F-206, M-8X0: Check HLP?/HELP answer to find out if DRT is supported. Only for GCS syntax version = GCS 2.0 or higher (Check with CSV?.vi. If CSV?.vi is not supported, syntax version is GCS 1.0). See User Manual of the controller for available trigger sources and values. DataRecorderTable = 0 (the specified trigger source is set for all data recorder tables).

2.13.23. DRT?.vi (Special command.llb)

Valid for	C-413, C-702, C-843, C-866, C-867, C-884, C-887, E-709, E-712, E-725, E-753, E-755, E-861, E-871, F-206, M-8X0, Mercury_GCS
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.

E-712, E-725, E-753: Check HLP? answer to find out if DRT? is supported.

C-887, F-206, M-8X0: Check HLP?/HELP answer to find out if DRT? is supported. Only for GCS syntax version = GCS 2.0 or higher (Check with CSV?.vi. If CSV?.vi is not supported, syntax version is GCS 1.0).

2.13.24. DRV.vi (Special command.llb)

Valid for	C-887, F-206, M-8X0
Input	System number (1), Strut to command (empty string), Mode (empty num. array, 1), Error in (no error)

C-887, F-206, M-8X0: Strut to command can be 1 to 6. Mode can be 1 (forward spindle rotation) or -1 (backward rotation)

Output	Error out
Remarks	A spindle revolution of one single strut is performed while monitoring the motor current. For system maintenance purposes. User "DRR?.vi" or

"DRR? and display data.vi" to read values back.

C-887, F-206: Check HLP?/HELP answer to find out if DRV? is supported. Only for GCS syntax version = GCS 2.0 or higher (Check with CSV?.vi. If CSV?.vi is not supported, syntax version is GCS 1.0). Requires command level 1 (CCL).

M-8X0: Only supported if controller is based on C-842.80 or GCS syntax version = GCS 2.0 or higher (Check with CSV?.vi. If CSV?.vi is not supported, syntax version is GCS 1.0). Check HLP?/HELP answer to find out if DRV is supported. For GCS 2.0, requires command level 1 (CCL).

2.13.25. HDR?.vi (Special command.llb)

Valid for	C-413, C-866, C-867, C-884, C-887, E-517, E-709, E-712, E-725, E-753, E-861, E-871, F-206, M-8X0, Mercury_GCS
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.

C-887, F-206, M-8X0: Check HLP?/HELP answer to find out if HDR? is supported. Only for GCS syntax version = GCS 2.0 or higher (Check with CSV?.vi. If CSV?.vi is not supported, syntax version is GCS 1.0).

2.13.26. IMP.vi (Special command.llb)

Valid for	C-413, C-887, E-517, E-709, E-710, E-712, E-725, E-753, E-755, E-761, F-206, M-8X0
Input	System number (1), Axis to command (empty string), Impulse size (0), Delay (0), No. of digits (4), Error in (no error)

C-413, C-887, E-517, E-709, E-710, E-712, E-725, E-753, E-755, F-206, M-8X0: Delay = 0.

E-761: Delay is impuls width in servo loops. Default value is 0 (for one servo loop)

Output	Error out
Remarks	Performs a single impulse-move (two equal moves in opposite directions in quick succession) from the current position with specified Impulse size (amplitude), and records fixed number of actual positions at specified intervals thereafter. If supported, Delay sets the number of servo loops between each position recording. No. of digits is the number of digits after the decimal point in the impulse amplitude values that will be sent. Controller saves a fixed number of position values, which can be read out with IMP?.vi (GCS 1.0) or DRR?.vi (GCS 2.0).

E-517: Controller saves up to 8,192 position values. Typically, IMP is used in open-loop mode. For a single step-move, see “STE.vi”. Use DRR?.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 IMP 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, IMP 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.

C-413, E-709: Controller saves up to 4096 position values. Typically, IMP is used in open-loop mode. For a single step-move, see “STE.vi”. Use DRR?.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 32,256 position values. Width of impulse and sampling interval taken from “Table Rate” parameter, set with “SPA_Hex.vi”.
Caution: “Table Rate” parameter influences Wave Generator also, not only IMP. Typically, IMP is used in open loop mode.For a single step-move, see “STE.vi”. Use IMP?.vi to read position values back.

E-712: Controller saves up to 262,144 position values. Typically, IMP is used in open-loop mode. For a single step-move, see “STE.vi”. Use DRR?.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. Typically, IMP is used in open-loop mode. For a single step-move, see “STE.vi”. Use DRR?.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. Typically, IMP is used in open-loop mode. For a single step-move, see “STE.vi”. Use DRR?.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. Typically, IMP is used in open-loop mode. For a single step-move, see “STE.vi”. Use DRR?.vi to read recorded values back.

E-761: Controller saves 8,192 position values. Typically, IMP is used in open loop mode. For a single step-move, see “STE.vi”. The number of servo cycles used for data recording depends on the setting made with RTR. Use DRR?.vi or IMP?.vi to read position values back.

C-887, F-206, M-8X0: Check HLP?/HELP answer to find out if IMP is supported. Only for GCS syntax version = GCS 2.0 or higher (Check with CSV?.vi. If CSV?.vi is not supported, syntax version is GCS 1.0). See controller User Manual for number of points (can be changed by setting the appropriate parameter value). For a single step move, see “STE.vi”. Use DRR?.vi to read recorded values back.

2.13.27. INI.vi (Special command.llb)

Valid for: C-702, C-843, C-843.PM, C-844, C-848, C-865, C-866, C-880, C-887, E-710, F-206, Hydra, Pollux, M-8X0, Mercury

Input: System number (1), INI 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? must be TRUE

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

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

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

C-887, F-206, M-8X0: Initialize hexapod: All axes? = TRUE, Axis identifier? = FALSE (separate axes A, B (if present) will not be initialized); initialize axes A, B: All axes? = FALSE, axes K ,L, M: command not valid (NanoCube is initialized with hexapod); VI will not wait for INI procedure to finish. Use “INI hexaxes and wait until finished.vi” to initialize hexapod and/or separate axes and wait for the procedure to finish. Depending on the firmware version on the controller, it may not be possible to stop motion initiated by INI with STOP, #24 or #27.

Hydra, Pollux: 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: Initializes axes. System-specific: see individual GCS-DLL Manual for details.

C-844: It is necessary to wait a certain time – appr. 4 s – before sending the next command to prevent it from being lost.

C-887, F-206, M-8X0: Check HLP?/HELP answer to find out if INI is supported. Only for GCS syntax version = GCS 1.0 (Check with CSV?.vi. If CSV?.vi is not supported, syntax version is GCS 1.0). For GCS 2.0 or higher, VI calls FRF instead for compatibility.

2.13.28. MAR!.vi (Special command.llb)

Valid for: M-8X0

Input: System number (1), Move mode (Move Z), Translation (0,0,0), Rotation (0,0,0), Error in (no error)

Output: Motion started (T/F), Motion stopped (T/F), Error out

Remarks: M-8X0-specific command. Moves stage. Waits for M-8X0 controller to reply 1 when all calculations are done and the movement starts, and another 1 when motion is complete.

M-8X0: Check HLP?/HELP answer to find out if MAR! is supported. Only for GCS syntax version = GCS 1.0 (Check with CSV?.vi. If CSV?.vi is not supported, syntax version is GCS 1.0). For compatibility, calls MOV.vi for GCS 2.0 or higher (in this case Motion started returns TRUE immediately, Motion stopped is determined by #5 polling.

2.13.29. MOV!.vi (Special command.llb)

Valid for: C-887, F-206, M-8X0

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

Output: Error out

Remarks: Moves specified axes to specified absolute positions and allows setting a new target position during the move without the move being interrupted. Use SCT.vi to set the cycle time for a periodic send. See User Manual for details. No. of digits is the number of digits after the decimal point in the

position value(s) that will be sent.

C-887, F-206, M-8X0: Check HLP?/HELP answer to find out if MOV! is supported. Only for GCS syntax version = GCS 1.0 (Check with CSV?.vi. If CSV?.vi is not supported, syntax version is GCS 1.0). Axes to move can be X, Y, Z, U, V, W. For compatibility, calls MOV.vi for GCS 2.0 or higher. In this case, parameter 0x19001900 must be set to 1 (SPA.vi), see controller User Manual for details.

2.13.30. RBT.vi (Special command.llb)

Valid for	C-413, C-702, C-867, C-884, C-887, E-517, E-709, E-712, E-725, E-753, E-755, E-761, E-861, E-870, E-871, F-206, Hydra, Pollux, M-8X0, Mercury_GCS
Input	System number (1), Error in (no error)
Output	Error out
Remarks	Reboots the controller. The controller behaves like after a cold start.

C-887, F-206, M-8X0: For GCS syntax version = GCS 2.0 or higher (Check with CSV?.vi. If CSV?.vi is not supported, syntax version is GCS 1.0).

2.13.31. RTR.vi (Special command.llb)

Valid for	C-413, C-843, C-867, C-884, C-887, E-517, E-709, E-712, E-725, E-753, E-761, E-861, E-871, F-206, M-8X0, Mercury_GCS
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.

C-887, F-206, M-8X0: Check HLP?/HELP answer to find out if RTR is supported. Only for GCS syntax version = GCS 2.0 or higher (Check with CSV?.vi. If CSV?.vi is not supported, syntax version is GCS 1.0).

2.13.32. RTR?.vi (Special command.llb)

Valid for	C-413, C-843, C-867, C-884, C-887, E-517, E-709, E-712, E-725, E-753, E-761, E-861, E-871, F-206, M-8X0, Mercury_GCS
Input	System number (1), Error in (no error)
Output	Record table rate, Error out
Remarks	Returns the current table rate.

C-887, F-206, M-8X0: Check HLP?/HELP answer to find out if RTR? is supported. Only for GCS syntax version = GCS 2.0 or higher (Check with CSV?.vi. If CSV?.vi is not supported, syntax version is GCS 1.0).

2.13.33. SCT.vi (Special command.llb)

Valid for	C-887, F-206, M-8X0
Input	System number (1), Cycle time (s) (1), Error in (no error)
Output	Error out, Hidden error
Remarks	Sets cycle time for MOV!/MOV command. See User Manual for details.

Controller error is TRUE if selected system reports error code ≠ 0.

C-887, F-206, M-8X0: Check HLP?/HELP answer to find out if SCT is supported. For GCS syntax version = GCS 1.0 (Check with CSV?.vi. If CSV?.vi is not supported, syntax version is GCS 1.0), sets cycle time for MOV! command. For GCS 2.0 or higher, sets cycle time for MOV command. Parameter 0x19001900 must be set to 1 in this case (SPA.vi), see controller User Manual for details.

2.13.34. SCT?.vi (Special command.llb)

Valid for	C-887, F-206, M-8X0
Input	System number (1), Error in (no error)
Output	Cycle time, s, Error out
Remarks	Returns current cycle time for MOV/MOV! command.

C-887, F-206, M-8X0: Check HLP?/HELP answer to find out if SCT? is supported. For GCS syntax version = GCS 1.0 (Check with CSV?.vi. If CSV?.vi is not supported, syntax version is GCS 1.0), returns cycle time for MOV! command.
For GCS 2.0 or higher, returns cycle time for MOV command. Parameter 0x19001900 must be set to 1 in this case (SPA.vi), see controller User Manual for details.

2.13.35. SPI.vi (Special command.llb)

Valid for	C-887, F-206, M-8X0
Input	System number (1), Pivot point coordinates (0,0,0), Error in (no error)
Output	Error out, Hidden error
Remarks	F-206- and M-8X0-specific. Sets the Pivot Point. Command will only be executed if all angular coordinates of current position are zero. Controller error is TRUE if selected system reports error code ≠ 0.

C-887, F-206, M-8X0: Pivot point coordinates are called (R, S, T) or (X, Y, Z).

2.13.36. SPI?.vi (Special command.llb)

Valid for	C-887, F-206, M-8X0
Input	System number (1), Error in (no error)
Output	Pivot point coordinates, Error out
Remarks	F-206- and M-8X0-specific. Returns the Pivot Point coordinates.

C-887, F-206, M-8X0: Pivot point coordinates are called (R, S, T) or (X, Y, Z).

2.13.37. SSN?.vi (Special command.llb)

Valid for	C-702, C-848, C-866, C-880, C-887, E-517, E-709, E-755, E-761, E-816, F-206, M-8X0
Input	System number (1), With channel ID (F), Channel name (empty string), Error in (no error)

E-816: With channel ID = TRUE

All other systems:: With channel ID = FALSE

Output	Serial number, Error out
Remarks	Returns controller serial number.

E-709: Check HLP? answer to find out if SSN? is supported.

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

C-887, F-206, M-8X0: Check HLP?/HELP answer to find out if SSN? is supported. Only for GCS syntax version = GCS 2.0 or higher (Check with CSV?.vi. If CSV?.vi is not supported, syntax version is GCS 1.0).

2.13.38. SST.vi (Special command.llb)

Valid for	C-702, C-848, C-867, C-880, C-887, E-871, F-206, M-8X0
Input	System number (1), Axes to set (empty string array), Step size (empty num. array, 0), No. of digits (4), Error in (no error)
	C-887, F-206, M-8X0: Only axes X, Y, Z, U, V, W allowed
Output	Error out
Remarks	Sets step size for manual control. No. of digits is the number of digits after the decimal point in the step size values that will be sent.
	C-702: Sets step size for cursor control (see also “SCA.vi”)
	C-848: Sets step size for cursor control (see also “SCA.vi”) and position control with a joystick connected to the C-848 (see also “JEN.vi”). Only older C-848s have a joystick connection.
	C-867: Step size is applied to a motion caused by an Human Interface (HI) device. HI control must be configured appropriately with HIA (Function = 2 (relative position), DeviceAxis = 3 or 4). Check HLP? answer to find out if SST is supported.
	C-880: Sets step size for Manual Control Pad, cursor control (see also “SCA.vi”) and position control with a joystick connected to the C-880 (see also “JEN.vi”).
	E-871: Step size is applied to a motion caused by an Human Interface (HI) device. HI control must be configured appropriately with HIA (Function = 2 (relative position), DeviceAxis = 3 or 4)
	C-887, F-206, M-8X0: Sets step size for Manual Control Pad

2.13.39. SST?.vi (Special command.llb)

Valid for	C-702, C-867, C-880, C-887, E-871, F-206, M-8X0
Input	System number (1), Axes to query (empty string array), All axes? (F), Axis identifier? (T), Error in (no error)
	C-702, C-867, C-880, E-871: If All axes? = TRUE, then Axis identifier? can be FALSE
	C-887, F-206, M-8X0: All axes? = TRUE, Axis identifier? = FALSE
Output	Step size, Error out
	C-887, F-206, M-8X0: Only values for axes X, Y, Z, U, V, W are valid
Remarks	Returns step-size setting (e.g. for manual control pad, see “SST.vi” for details).
	C-867: Check HLP? answer to find out if SST? is supported.

2.13.40. STA?.vi (Special command.llb)

Valid for	C-702, C-848, C-880, C-880K005, C-887, F-206, 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

C-887, F-206, 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”.

C-887, F-206, M-8X0: Check HLP?/HELP answer to find out if STA? is supported. Command is equivalent to #4. For details see User Manual of the controller.

2.13.41. STE.vi (Special command.llb)

Valid for Analog systems, C-413, C-843, C-843.PM, C-848, C-865, C-866, C-867, C-880, C-884, C-887, E-517, E-709, E-710, E-712, E-725, E-753, E-755, E-761, E-861, E-871, F-206, M-8X0, Mercury_GCS. 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 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.

C-884: Controller saves 8192 position values. Motion commands like STE are not allowed when control via a Human Interface Device (HID) is active for the axis of the controller. 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.

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.

C-413, 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.

E-871: Controller saves up to 1024 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-887, F-206, M-8X0: Check HLP?/HELP answer to find out if STE is supported. Only for GCS syntax version = GCS 2.0 or higher (Check with CSV?.vi. If CSV?.vi is not supported, syntax version is GCS 1.0). See controller User Manual for number of points (can be changed by setting the appropriate parameter value). For a single impulse-move, see “IMP.vi”. Use DRR?.vi to read recorded values back.

Mercury_GCS: 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.

2.13.42. TIO?.vi (Special command.llb)

Valid for	C-413, C-702, C-843, C-843.PM, C-848, C-866, C-867, C-880, C-884, E-517, E-709, E-712, E-761, E-861, E-871, Hydra, Mercury, Mercury_GCS (but must be present for C-887, E-816, F-206, M-8X0 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.

E-709, E-712: Check HLP? answer to find out if TIO? is supported.

2.13.43. TNR?.vi (Special command.llb)

Valid for	C-413, C-702, C-866, C-867, C-884, C-887, E-517, E-709, E-710, E-712, E-725, E-753, E-755, E-761, E-861, E-871, F-206, M-8X0, Mercury_GCS

Input	System number (1), Error in (no error)
Output	Number of Rec. tables, Error out
Remarks	Returns the number of recording tables. C-887, F-206, M-8X0: Check HLP?/HELP answer to find out if TNR? is supported. Only for GCS syntax version = GCS 2.0 or higher (Check with CSV?.vi. If CSV?.vi is not supported, syntax version is GCS 1.0).

2.13.44. TPC?.vi (Special command.llb)

Valid for	C-413, C-887, 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, C-884, E-516, E-816, E-861, E-870, E-871, F-206 and M-8X0 and Mercury_GCS 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).

2.13.45. 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, C-884, E-517, E-710, E-761, E-871, Hydra, Pollux, Mercury, Mercury_GCS (but must be present for C-413, C-887, E-709, E-816, E-861, F-206 and M-8X0 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, C-884, E-517, E-871, Hydra, Pollux, Mercury_GCS: 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.

2.13.46. VST?.vi (Special command.llb)

Valid for	C-702, C-843, C-843.PM, C-844, C-848, C-865, C-866, C-880, C-884, C-887, E-710, E-761, E-871, F-206, Hydra, Pollux, M-8X0, Mercury
Input	System number (1), Error in (no error)
Output	Available stages, Error out
Remarks	Returns the names of all stages which can be connected to the controller.

C-884: Command is available via USB interface only.

E-871: Command is available via USB, USB DaisyChain or RS232 DaisyChain interface only.

C-887, F-206, M-8X0: Check HLP?/HELP answer to find out if VST? is supported. Only for GCS syntax version = GCS 2.0 or higher (Check with CSV?.vi. If CSV?.vi is not supported, syntax version is GCS 1.0).

2.13.47. VLS.vi (Special command.llb)

Valid for	C-887, F-206, M-8X0 (but must be present for C-880, too).
Input	System number (1), No. of digits (4), Velocity (0), Error in (no error)
	C-887, F-206, M-8X0: Only valid for GCS syntax version = GCS 2.0 or higher (Check with CSV?.vi. If CSV?.vi is not supported, syntax version is GCS 1.0).
Output	Error out, Controller error
Remarks	Sets platform velocity 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.

2.13.48. VLS?.vi (Special command.llb)

Valid for	C-887, F-206, M-8X0 (but must be present for C-880, too).
Input	System number (1), Error in (no error)
	C-887, F-206, M-8X0: Only valid for GCS syntax version = GCS 2.0 or higher (Check with CSV?.vi. If CSV?.vi is not supported, syntax version is GCS 1.0).
Output	Velocity, Error out
Remarks	Returns platform velocity setting.

2.13.49. WPA.vi (Special command.llb)

Valid for	C-413, C-867, C-884, C-887, E-516, E-517, E-709, E-710, E-712, E-725, E-753, E-755, E-761, E-816, E-861, E-870, E-871, F-206, Hydra, Pollux, Mercury_GCS, M-8X0
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-413, C-867, C-884, Hydra, Pollux, Mercury_GCS: 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).
	C-887, F-206, M-8X0: Check HLP?/HELP answer to find out if WPA is supported. Only for GCS syntax version = GCS 2.0 or higher (Check with CSV?.vi. If CSV?.vi is not supported, syntax version is GCS 1.0). Affected axes and Parameter to save is only valid for Password = 100 or 101 (If Affected axes = empty array, all parameters for all axes are saved). Parameter no. format is TRUE (hex).
	E-517, E-709, E-710, E-712, E-725, E-753, E-755, E-761, E861, E-870, E-871: 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.

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-413: 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-413 is powered off or rebooted. Parameters can be changed in volatile memory with SPA, AOS, ATZ, CMO, RTR, VEL, WOS and WTR. Depending on the parameter to be saved, it may be necessary to switch to command level 1 (“CCL.vi”).

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 controller is powered off or rebooted. Parameters can be changed in volatile memory with SPA, ACC, DEC and VEL. 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.

C-884: Depending on the password, the WPA command saves the currently valid parameter values and/or the current HI device configuration to non-volatile memory, where they become the power-on defaults. Settings not saved with WPA will be lost when the controller is powered off or rebooted.
Valid passwords and affected settings:
100: all parameters, settings of HDT, HIA, HIT
101: all parameters
HID: settings of HDT, HIA, HIT
Parameters can be changed in volatile memory with SPA, ACC, DEC and VEL.
WPA must be used without specifying any arguments except of the password.

C-887, F-206, M-8X0: Depending on the password, the WPA command saves the currently valid parameter values and/or the current coordinate systems settings or stage assignments to non-volatile memory, where they become the power-on defaults. Settings not saved with WPA will be lost when the controller is powered off or rebooted.
Valid passwords and affected settings:

SKS: coordinate system settings
A12: stage assignments for separate axes A and B

101: all parameters
100: all parameters and settings of SKS and A12
Parameters can be changed in volatile memory with SPA.

Hydra, Pollux: 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 controller is powered off or rebooted. Parameters can be changed in volatile memory with SPA, ACC, and VEL. 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 .

E-870: The WPA command saves the currently valid parameter values to non-volatile memory, including the joystick configuration, where they become the power-on defaults. Settings not saved with WPA will be lost when the E-870 is powered off or rebooted. Parameters can be changed in volatile memory with SPA.

E-871: Depending on the password, the WPA command saves the currently valid parameter values and/or the current HI device configuration to non-volatile memory, where they become the power-on defaults. Settings not saved with WPA will be lost when the controller is powered off or rebooted.
Valid passwords and affected settings:
100: all parameters, settings of HDT, HIA, HIT
101: all parameters
HID: settings of HDT, HIA, HIT
Parameters can be changed in volatile memory with SPA.
WPA must be used without specifying any arguments except of the password.

Mercury_GCS: 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 Mercury is powered off or rebooted. Parameters can be changed in volatile memory with SPA, ACC, DEC and VEL.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.

2.14. 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.

2.14.1. 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.

2.14.2. 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".

2.14.3. Assign DRC values.vi (Support.llb)

Valid for	C-413, C-702, C-843, C-866, C-867, C-884, C-887, E-517, E-709, E-710, E-712, E-725, E-753, E-755, E-861, E-871, F-206, M-8X0, Mercury_GCS
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.

2.14.4. Assign DRT values from string to axes.vi (Support.llb)

Valid for	C-413, C-702, C-866, C-867, C-884, C-887, E-755, E-861, E-871, F-206, M-8X0, Mercury_GCS
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”.

2.14.5. Assign NaN for chosen axes.vi (Support.llb)

Valid for	Analog systems, C-413, C-702, C-843, C-843.PM, C-848, C-865, C-866, C-867, C-880, C-884, C-887, E-517, E-709, E-712, E-725, E-753, E-755, E-861, E-871, F-206, Hydra, Pollux, M-8X0, Mercury, Mercury_GCS. 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.

2.14.6. Assign SPA values from string to axes.vi (Support.llb)

Valid for	C-413, C-702, C-843, C-843.PM, C-844, C-848, C-865, C-866, C-867, C-880, C-880K005, C-884, C-887, E-516, E-517, E-709, E-710, E-712, E-725, E-753, E-755, E-761, E-816, E-861, E-870, E-871, Hydra, Pollux, Mercury, Mercury_GCS
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”.

2.14.7. 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.

2.14.8. 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.

2.14.9. Build channel query command substring.vi (Support.llb)

Valid for	Analog systems, C-413, C-867, C-884, C-887, E-516, E-517, E-709, E-710, E-712, E-725, E-753, E755, E-761, E-816, E-861, E-870, E-871, F-206, Mercury_GCS, M-8X0
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.

2.14.10. 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”.

2.14.11. Build DIO? query command substring.vi (Support.llb)

Valid for	C-413, C-702, C-843, C-843.PM, C-848, C-865, C-866, C-867, C-880, C-884, C-887, E-517, E-709, E-761, E-816, E-861, E-871, F-206, Hydra, Pollux, Mercury, Mercury_GCS, M-8X0
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.

2.14.12. 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"

2.14.13. 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".

2.14.14. Build SPA command substring.vi (Support.llb)

Valid for	C-413, C-702, C-843, C-843.PM, C-844, C-848, C-865, C-866, C-867, C-880, C-880K005, C-884, C-887, E-516, E-517, E-709, E-712, E-725, E-753, E-816, E-861, E-870, E-871, F-206, Hydra, Pollux, M-8X0, Mercury, Mercury_GCS (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".

2.14.15. Build SPA query command substring.vi (Support.llb)

Valid for	C-413, C-702, C-843, C-843.PM, C-844, C-848, C-865, C-866, C-867, C-880, C-880K005, C-884, C-887, E-516, E-517, E-709, E-710, E-712, E-725, E-753, E-755, E-761, E-816, E-861, E-870, E-871, Hydra, Pollux, Mercury,

	Mercury_GCS
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” .

2.14.16. 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.

2.14.17. Combine axes arrays.vi (Support.llb)

Valid for	All systems
Input	Axes subset 1 (empty string array), Axes subset 2 (empty string array),
Output	Combined axes, Double axes
Remarks	This vi combines axes from Axes subset 1 and Axes subset 2 and returns the combined and sorted axes array plus axes which had double entries.

2.14.18. 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.

2.14.19. Commanded stage name available?.vi (Support.llb)

Valid for	C-702, C-843, C-843.PM, C-844, C-848, C-865, C-866, C-880, C-884, C-887, E-710, E-761, E-871, F-206, Hydra, Pollux, Mercury, M-8X0
Input	System number (1), Commanded stages (empty string array), Error in (no error)
Output	Hidden error (T/F), Error out
Remarks	This VI checks if Commanded stages is a subset of all available stages and returns Hidden error TRUE if this is not the case. Available stages are defined by “VST?.vi”.

2.14.20. 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".

2.14.21. 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.

2.14.22. 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.

2.14.23. 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.

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.

2.14.24. 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.

2.14.25. Determine angular axis.vi (Support.llb)

Valid for	C-887, F-206, M-8X0
Input	System no. (1), Axis 1 (empty string), Axis 2 (empty string), Error in (no error)
Output	Ang. axis out, NC axis 1, NC axis 2, Error out
Remarks	Determines angular axis spanned by Axis 1 and Axis 2, and NanoCube axis identifiers corresponding with Axis 1 and Axis 2.

2.14.26. 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.

2.14.27. 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), Add. wait only? (F), Stop refnum (F), Error in (no error) E-816: All axes? = FALSE, only one axis per command allowed C-887, F-206, M-8X0: For GCS syntax version = GCS 1.0 (Check with CSV?.vi. If CSV?.vi is not supported, syntax version is GCS 1.0), 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. If Add. wait only? is TRUE, VI waits the given Additional wait time only. 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. VI does not time out, so when using as a sub-VI, use Stop refnum to stop VI from caller.

2.14.28. 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.

2.14.29. 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.

2.14.30. 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”.

2.14.31. 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”.

2.14.32. 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.

2.14.33. Get total number of commanded axes.vi (Support.llb)

Valid for	All systems
Input	System no. (1), String (empty string)
Output	Equal?, Axes, Commanded axes, Index of axis
Remarks	This vi returns how many axes to command String contains and if that number is equal to the total number of connected axes. Additionally it returns all commanded axes and their index in the Connected Axes array.

2.14.34. 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.

2.14.35. 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.

2.14.36. Longlasting one-axis command.vi (Support.llb)

Valid for	C-413, C-702, C-843, C-843.PM, C-844, C-848, C-865, C-866, C-880, C-880K005, C-884, C-887, E-710, E-761, E-871, F-206, Hydra, Pollux, Mercury (but must be present for E-709, E-712, E-725, and E-753 also)
Input	System number (1), Axis to command (empty string), Command (empty string), Axis and value? (F), Value (NaN), Error in (no error)
Output	Answer (T/F), Error out
Remarks	This VI sends a command (like REF, MNL or MPL), polls with #7 for controller-ready signal and returns original (boolean) command response.

2.14.37. Manual VMO.vi (Support.llb)

Valid for	C-844, C-848, C-848.PM, C-865, C-880, C-887, E-516, E-710, E-761, E-816, F-206, M-8X0
Input	System number (1), Axes to command (empty string array), Minimum pos. (empty num. array), Maximum pos. (empty num. array), Position values (empty num. array, 0), 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 by checking if the commanded position value is within the given position range. Axes will NOT be moved.

2.14.38. Parse KLS? type answer by type.vi (Support.llb)

Valid for	C-887, F-206, M-8X0
Input	Type (empty string), KLS? response (empty string), Error in (no error)
Output	Names, Error out
Remarks	Filters the response of KLS? type answers by the selected type and returns all names related to that type.

2.14.39. 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.

2.14.40. Return space.vi (Support.llb)

Valid for	All systems

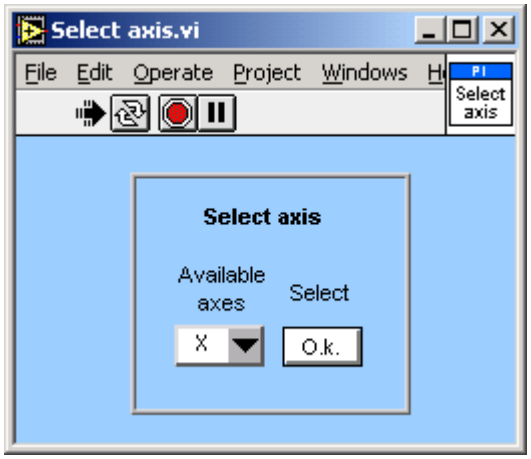
Input	System no. (1), With space? (F)
Output	String out, Space returned?
Remarks	This VI returns a space character in String out if With space? is TRUE or GCS syntax version is higher than 1.0.

2.14.41. 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.

2.14.42. 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.



2.14.43. 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.

2.14.44. 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.

2.14.45. 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).

2.14.46. Send string and wait for answer or timeout.vi (Support.llb)

Valid for	C-887, F-206, M-8X0
Input	System number (1), Timeout, s (60), String to send (empty string), Attach linefeed? (TRUE), Error in (no error)
Output	String read, Error out
Remarks	Sends a string and waits until an answer has come or a timeout condition has occured. Timeout is independent of the global communication timeout value. Needed by “INI hexaxes and wait until finished.vi”. C-887, F-206, M-8X0: INI procedure of GCS 1.0 firmware (Check with CSV?.vi. If CSV?.vi is not supported, syntax version is GCS 1.0) can be stopped with #24 since firmware version 4.0.0 (F-206) and 4.5.0 (M-8X0).

2.14.47. Sep axes ini result in order.vi (Support.llb)

Valid for	C-887, F-206, M-8X0
Input	Sep. axes to initialize (empty string array), Sep. axes initialized? (empty bool. Array)
Output	A/B initialized?
Remarks	Returns result of "Sep. axes initialized?" according to order of "Sep. axes to initialize".

2.14.48. 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.

2.14.49. 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".

2.14.50. 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".

2.14.51. Wait for answer 0 or 1 without polling.vi (Support.llb)

Valid for	C-702, C-843, C-843.PM, C-844, C-848, C-865, C-866, C-880, C-880K005, C-887, F-206, M-8X0, Mercury
Input	System number (1), Start time (0), Timeout (s) (60), Stop refnum (F), Local stop (F), Error in (no error)
Output	Scan successful? (T/F), Error out
Remarks	This VI waits until answer 0 or 1 comes or timeout is reached without single-character polling. VI also stops if Stop refnum or Local stop is TRUE. When using as a sub-VI, use Stop refnum to stop VI from caller.

2.14.52. Wait for answer of longlasting command.vi (Support.llb)

Valid for	C-413, C-702, C-843, C-843.PM, C-844, C-848, C-865, C-866, C-880, C-880K005, C-884, C-887, E-710, E-761, E-871, Hydra, Pollux, Mercury (but must be present for E-516, E-709, E-712, E-725, E-753, F-206 and M-8X0, too)
Input	System number (1), Stop refnum (F), Local stop (F), Error in (no error)
Output	Answer (T/F), Error out
Remarks	This VI waits for the answer of commands like REF, MPL, MNL or scanning routines using #7 polling and stops if answer has come, Stop refnum or Local stop is TRUE, or if a communications error occured. Sub-VI for Long-lasting, one-axis commands and controller-algorithm commands. Requires "#7.vi" to be present. When using as a sub-VI, use Stop refnum to stop VI from caller.

2.14.53. Wait for axes to stop.vi (Support.llb)

Valid for	C-413, C-702, C-843, C-843.PM, C-844, C-848, C-865, C-867, C-880, C-884, C-887, E-517, E-709, E-712, E-725, E-753, E-755, E-861, E-870, E-871, Hydra, Pollux, Mercury , Mercury_GCS (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.

2.14.54. Wait for controller ready.vi (Support.llb)

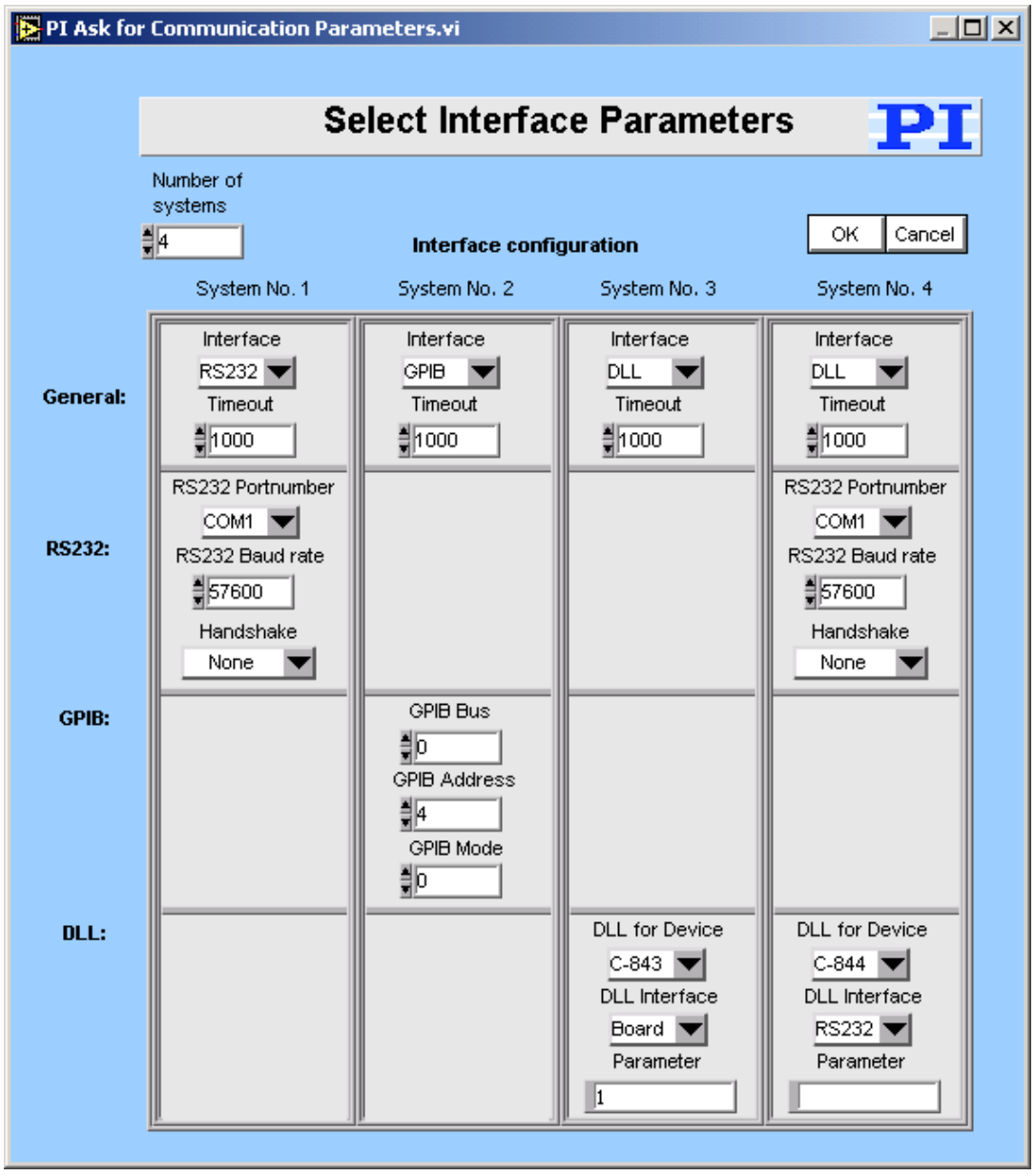
Valid for	C-413, C-702, C-843, C-843.PM, C-848, C-865, C-866, C-867, C-880, C-884, C-887, E-517, E-709, E-710, E-712, E-755, E-761, E-861, E-870, E-871, F-206, Hydra, Pollux, Mercury , Mercury_GCS, M-8X0 (but must be present for E-816 also)
Input	System number (1), Polling time, ms (50), Stop refnum (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.

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-413: Interface = RS232 or DLL, RS232: Input and output HW handshake must be TRUE. DLL: DLL for Device = PI_GCS2_DLL, DLL Interface = USB, Parameter = Serial no. of system to connect to. Syntax: GCS 2.0; Term char = LF.

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.

C-884: Interface = RS232, TCP/IP or DLL, RS232: Input and output HW handshake must be FALSE. DLL (USB): DLL for Device = PI_GCS2_DLL, DLL Interface = USB, Parameter = Serial no. of system to connect to. Syntax: GCS 2.0; Term char = LF.

C-887, F-206, M-8X0: For GCS syntax version = GCS 1.0 (older firmware or compatibility mode, can be set with CSV.vi), Interface = RS232, GPIB (not available for C-887 controller) 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. For GCS syntax version = GCS 2.0 (C-887 controller), Interface = RS232 or TCP/IP. RS232: Input and output handshake settings must be FALSE. Syntax: GCS 2.0; 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.
E-870:	Interface = DLL, DLL (USB): DLL for Device = PI_GCS2_DLL, DLL Interface = USB, Parameter = Serial no. of system to connect to. Syntax: GCS 2.0; Term char = LF.
E-871:	Single Device: Interface = RS232 or DLL, RS232: Input and output HW handshake must be FALSE. DLL (USB): DLL for Device = PI_GCS2_DLL, DLL Interface = USB, Parameter = Serial no. of system to connect to. DaisyChain: Interface = DLL, DLL for Device = PI_GCS2_DLL, DLL Interface = RS232_DC or USB_DC, Parameter = Number of device in chain, Register DC: FALSE. Syntax: GCS 2.0; Term char = LF
Hydra:	Interface = DLL, DLL (TCP/IP and RS-232): DLL for Device = PI_HydraPollux_GCS2_DLL, DLL Interface = RS232 or TCP/IP, Syntax: GCS 2.0; 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.
GCS_Mercury:	Single Device: Interface = RS232 or DLL, RS232: Input and output HW handshake must be FALSE. DLL (USB): DLL for Device = PI_GCS2_DLL, DLL Interface = USB, Parameter = Serial no. of system to connect to. DaisyChain: Interface = DLL, DLL for Device = PI_GCS2_DLL, DLL Interface = RS232_DC or USB_DaisyChain, Parameter = Number of device in chain, Register DC: FALSE. Syntax: GCS 2.0; Term char = LF.

Pollux: Interface = DLL, DLL (RS-232): DLL for Device = PI_HydraPollux_GCS2_DLL, DLL Interface = RS232, Baudrate must be 19200, Syntax: GCS 2.0; 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.

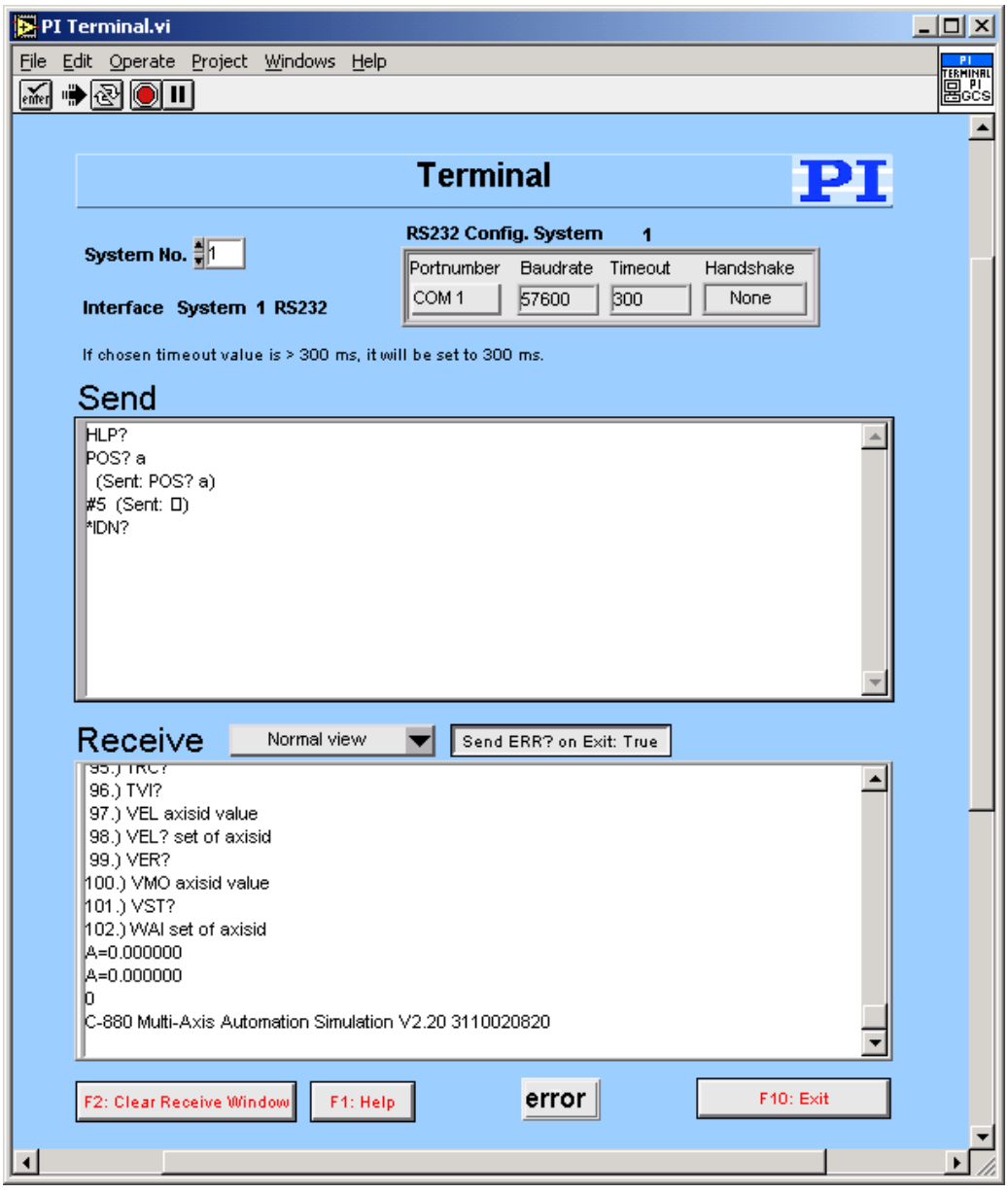
In the upper window ("Send") the user can enter commands which will be transmitted to the chosen device one line at a time when the ENTER key is pressed.

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.

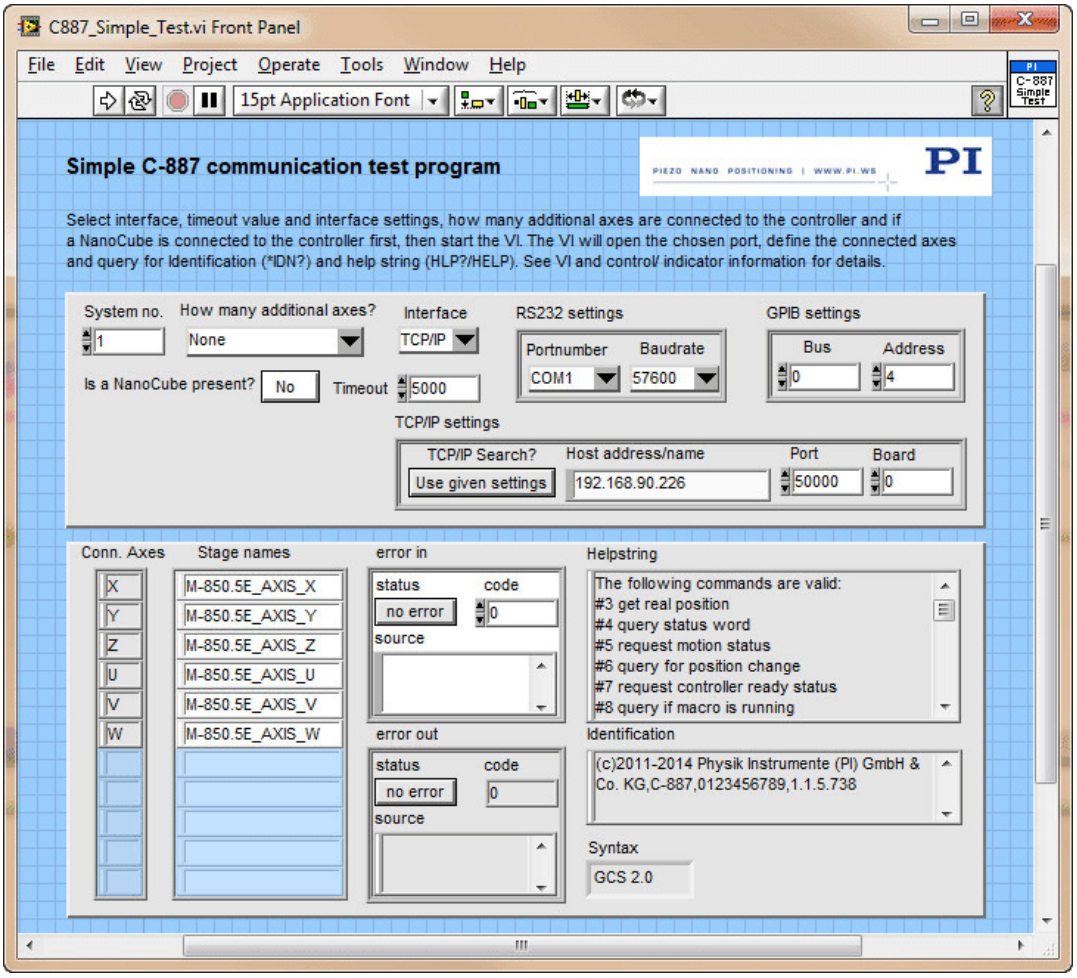
3.2. C887_Simple_Test.vi

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.

Select

- System number (= 1 in a one-system configuration),
- interface,
- timeout value,
- interface settings
- if any additional axes are connected to the controller,
- and if a NanoCube is connected to the controller,

then start the VI. The VI will open the chosen port and query for the identification (*IDN?) and help string (HLP? for GCS II or HELP for GCS I). Depending on the chosen values of How many additional axes? and Is a NanoCube present? the connected axes ID's are returned. The diagram shows how to combine the driver VIs for these tasks.



3.3. C887_Configuration_Setup.vi

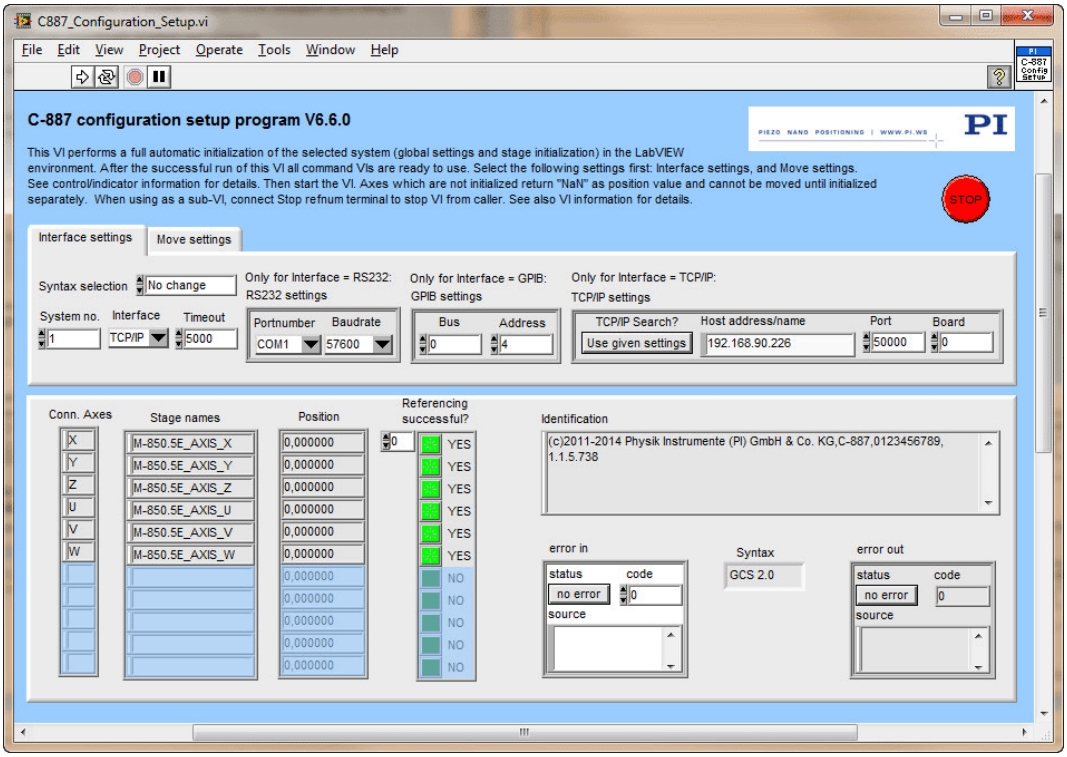
This VI performs a full automatic initialization of the selected system (global settings and stage referencing) in the LabVIEW environment. Use the *Help→Show Context Help* menu sequence in the LabVIEW environment to display the *Context Help* window with the VI and control/indicator descriptions.

It automatically detects if the connected Hexapod system is GCS 1.0 (e.g. F-206, M-8X0) or 2.0 (e.g. C-887 + H-206, C-887 + H-8XX) and works for both versions. After a successful run of this VI, all command VIs are ready to use. If you have separate axes A or B connected to the C-887 controller, use PIMIkroMove.exe to define connected stages and save selection to the controller.

First specify the following:

- System number (= 1 in a one-system-only configuration),
- Interface
- Timeout value
- Interface settings
- Whether the Hexapod is to be initialized
- If a NanoCube is connected to the controller (if you have ordered the NCU option, you can drive a 3-axis piezo stage (“NanoCube”) with the controller)
- How many additional axes are connected to the controller (you can drive up to two additional PI stages with the controller)
- Whether the additional axes are to be initialized.

Then start the VI.



“C887_Configuration_Setup.vi” performs the following initialization tasks:

1. Runs “PI Open Interface of one system.vi” to open the chosen port.
2. Runs “*IDN?.vi” to query for the controller identification string.
3. Defines the selected system to be "F-206" (“C-887” is not available separately).
4. Runs “CST?.vi” to determine which stages are connected to the controller.
5. Runs “Define connected axes.vi” with Read from controller = FALSE and Connected axes depending on the selection of How many additional axes are present? and Is a NanoCube present?.
6. Initializes the Hexapod and separate axes (if any) according to your selections Initialize hexapod? and Initialize additional axes? by running “INI hexaxes and wait until finished.vi” or “FRF.vi”. The NanoCube (if present) will be automatically initialized with Initialize hexapod? = TRUE.
7. Runs “POS?.vi” to query for the position of all axes.
8. Runs “ERR?.vi” to query the controller for its error status.
9. Runs “GCSTranslateError.vi” to append the error message which corresponds with a GCS error number returned by “ERR?.vi” to Source from Error out.

Axes which are designated not to be initialized report NaN as position value and cannot be moved until initialized separately.

Use this VI as the initialization VI for the hexapod system in your application.

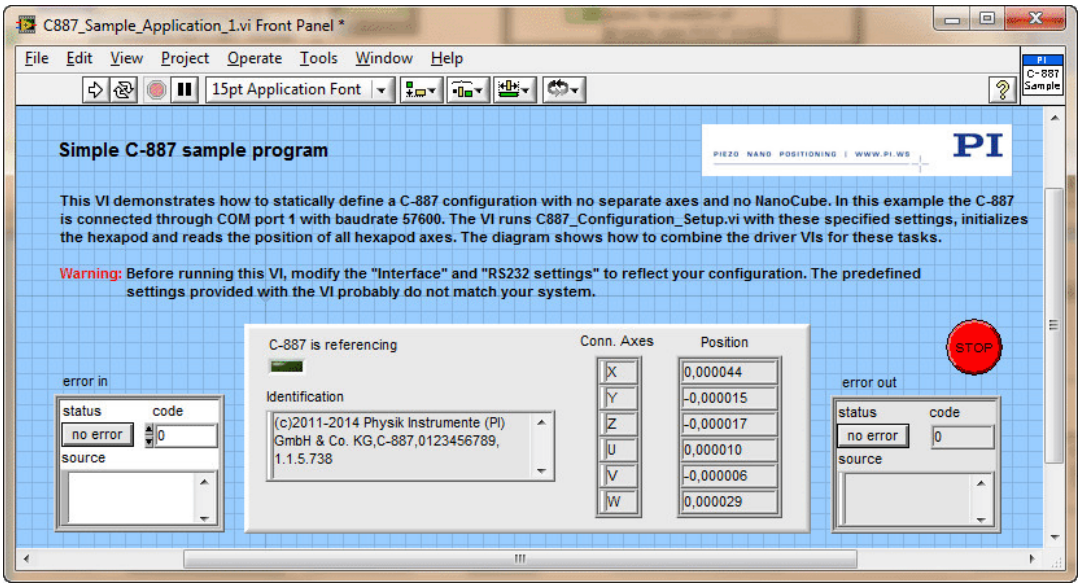
When using as a sub-VI, connect Stop refnum terminal to stop VI from caller.

As the initialization is a complex procedure which uses a large number of sub-VIs, C887_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. C887_Sample_Application_1.vi

This VI demonstrates how to statically define a C-887 system configuration with no separate axes and no NanoCube. In this example the controller is connected through COM port 1 with baudrate 57600. The VI runs C887_Configuration_Setup.vi with these specified settings, initializes the Hexapod and reads the position of all Hexapod axes. The diagram shows how to combine the driver VIs for these tasks.

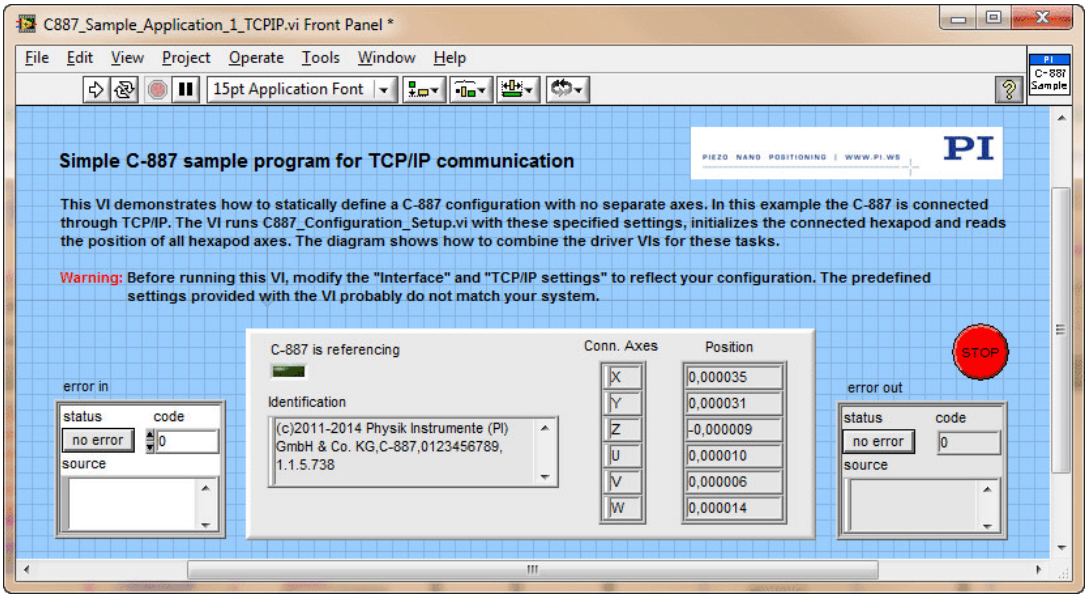
Warning: Before running this VI, modify the "Interface" and "RS232 settings" to reflect your configuration. The predefined settings provided with the VI probably do not match your system.



3.5. C887_Sample_Application_1_TCPIP.vi

This VI demonstrates how to statically define a C-887 system configuration with no separate axes and no NanoCube. In this example the controller is connected through TCP/IP. The VI runs C887_Configuration_Setup.vi with these specified settings, initializes the connected Hexapod and reads the position of all Hexapod axes. The diagram shows how to combine the driver VIs for these tasks.

Warning: Before running this VI, modify the "Interface" and "TCP/IP settings" to reflect your configuration. The predefined settings provided with the VI probably do not match your system.



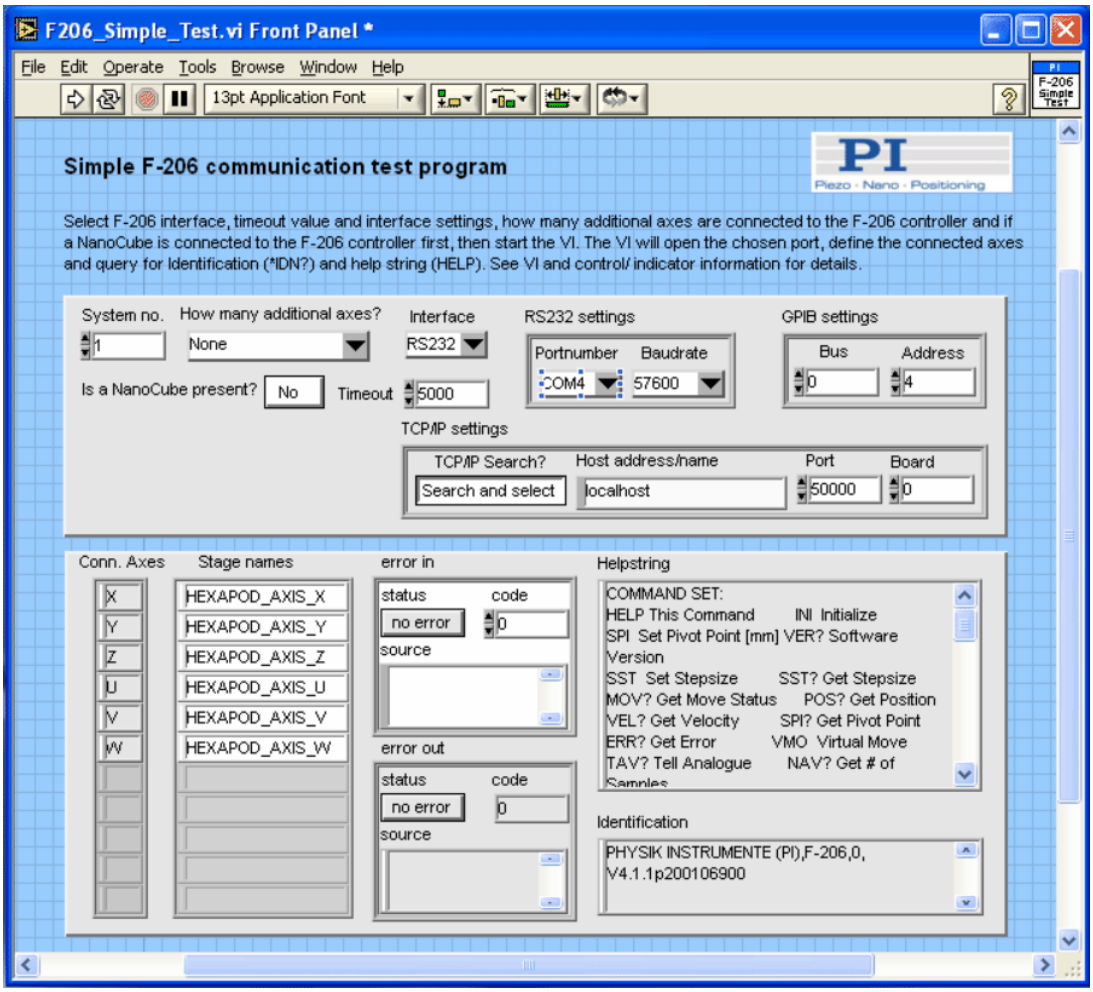
3.6. F206_Simple_Test.vi

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.

Select

- System number (= 1 in a one-system configuration),
- interface,
- timeout value,
- interface settings
- if any additional axes are connected to the controller,
- and if a NanoCube is connected to the controller,

then start the VI. The VI will open the chosen port and query for the identification (*IDN?) and help string (HELP). Depending on the chosen values of How many additional axes? and Is a NanoCube present? the connected axes ID's are returned. The diagram shows how to combine the driver VIs for these tasks.



3.7. F206_Configuration_Setup.vi

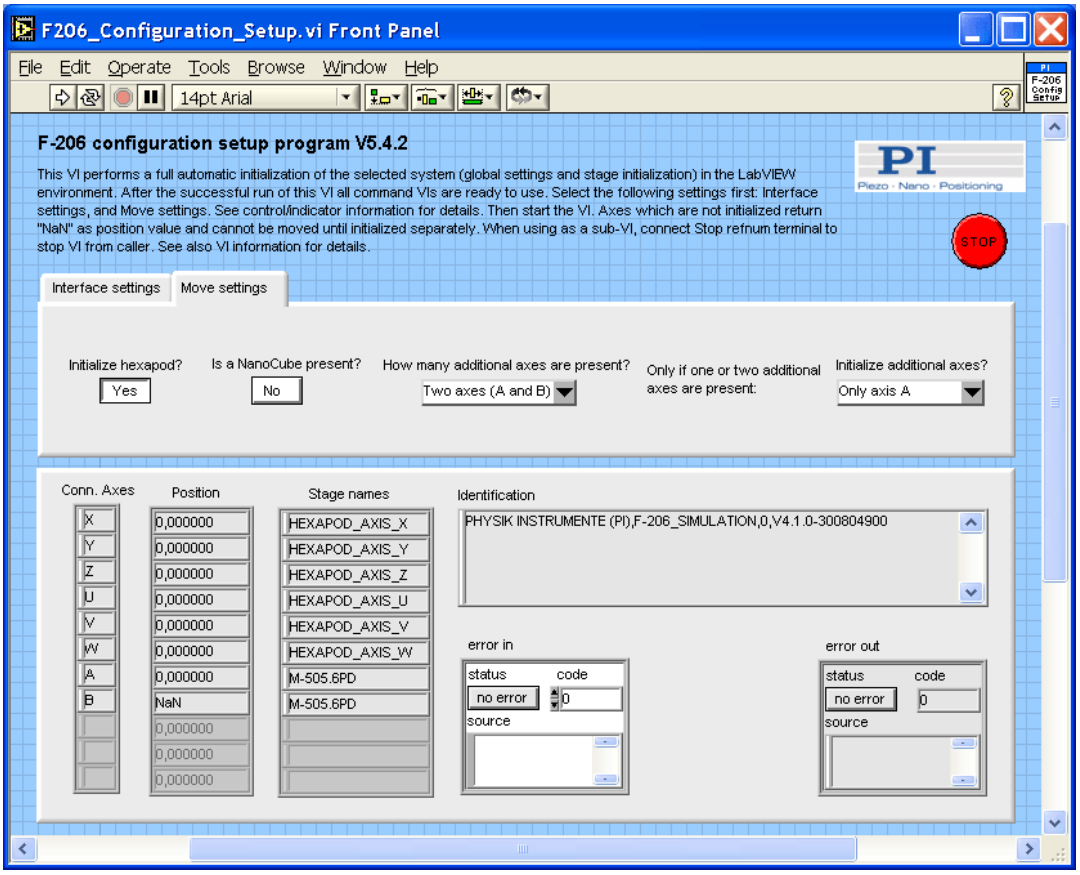
This VI is available for compatibility reasons only. For new applications, please use C887_Configuration_Setup.vi instead.

This VI performs a full 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.

It automatically detects if the connected Hexapod system is GCS 1.0 (F-206) or 2.0 (C-887 + H-206) and works for both versions. After a successful run of this VI, all command VIs are ready to use. First specify the following:

- System number (= 1 in a one-system-only configuration),
- Interface
- Timeout value
- Interface settings
- Whether the Hexapod is to be initialized
- If a NanoCube is connected to the controller (if you have ordered the NCU option, you can drive a 3-axis piezo stage (“NanoCube”) with the controller)
- How many additional axes are connected to the controller (you can drive up to two additional PI stages with the controller)
- Whether the additional axes are to be initialized.

Then start the VI.



"F206_Configuration_Setup.vi" performs the following initialization tasks:

10. Runs "PI Open Interface of one system.vi" to open the chosen port.
11. Runs "*IDN?.vi" to query for the controller identification string.
12. Defines the selected system to be "F-206".
13. Runs "CST?.vi" to determine which stages are connected to the controller.
14. Runs "Define connected axes.vi" with Read from controller = FALSE and Connected axes depending on the selection of How many additional axes are present? and Is a NanoCube present?.
15. Initializes the Hexapod and separate axes (if any) according to your selections Initialize hexapod? and Initialize additional axes? by running "INI hexaxes and wait until finished.vi" or "FRF.vi". The NanoCube (if present) will be automatically initialized with Initialize hexapod? = TRUE.
16. Runs "POS?.vi" to query for the position of all axes.
17. Runs "ERR?.vi" to query the controller for its error status.
18. Runs "GCSTranslateError.vi" to append the error message which corresponds with a GCS error number returned by "ERR?.vi" to Source from Error out.

Axes which are designated not to be initialized report NaN as position value and cannot be moved until initialized separately.

Use this VI as the initialization VI for the F-206 or C-887 + H-206 in your application.

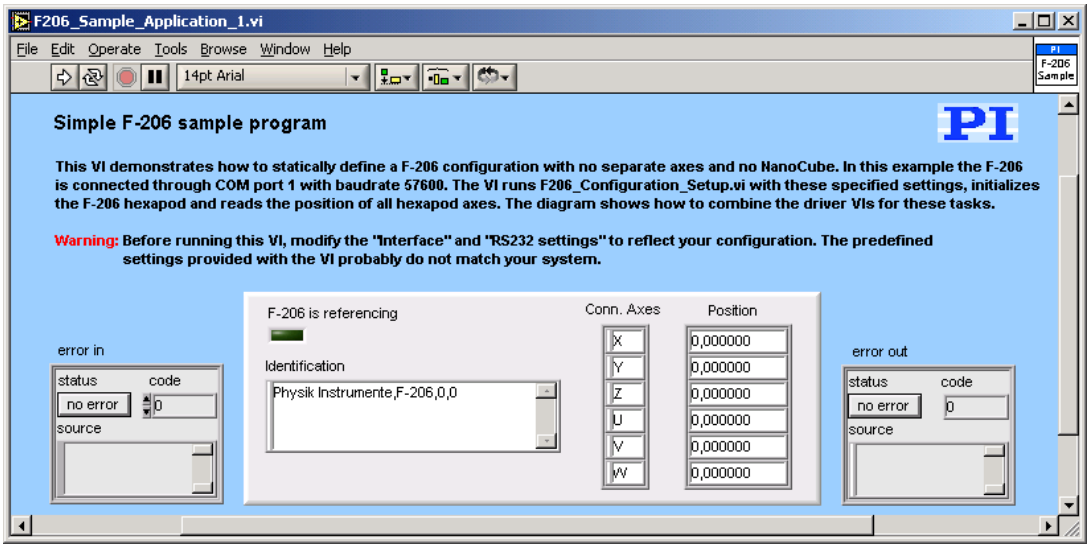
When using as a sub-VI, connect Stop refnum terminal to stop VI from caller.

As the initialization is a complex procedure which uses a large number of sub-VIs, F206_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.

3.8. F206_Sample_Application_1.vi

This VI demonstrates how to statically define an F-206 or C-887 + H-206 configuration with no separate axes and no NanoCube. In this example the controller is connected through COM port 1 with baudrate 57600. The VI runs F206_Configuration_Setup.vi with these specified settings, initializes the Hexapod and reads the position of all Hexapod axes. The diagram shows how to combine the driver VIs for these tasks.

Warning: Before running this VI, modify the "Interface" and "RS232 settings" to reflect your configuration. The predefined settings provided with the VI probably do not match your system.



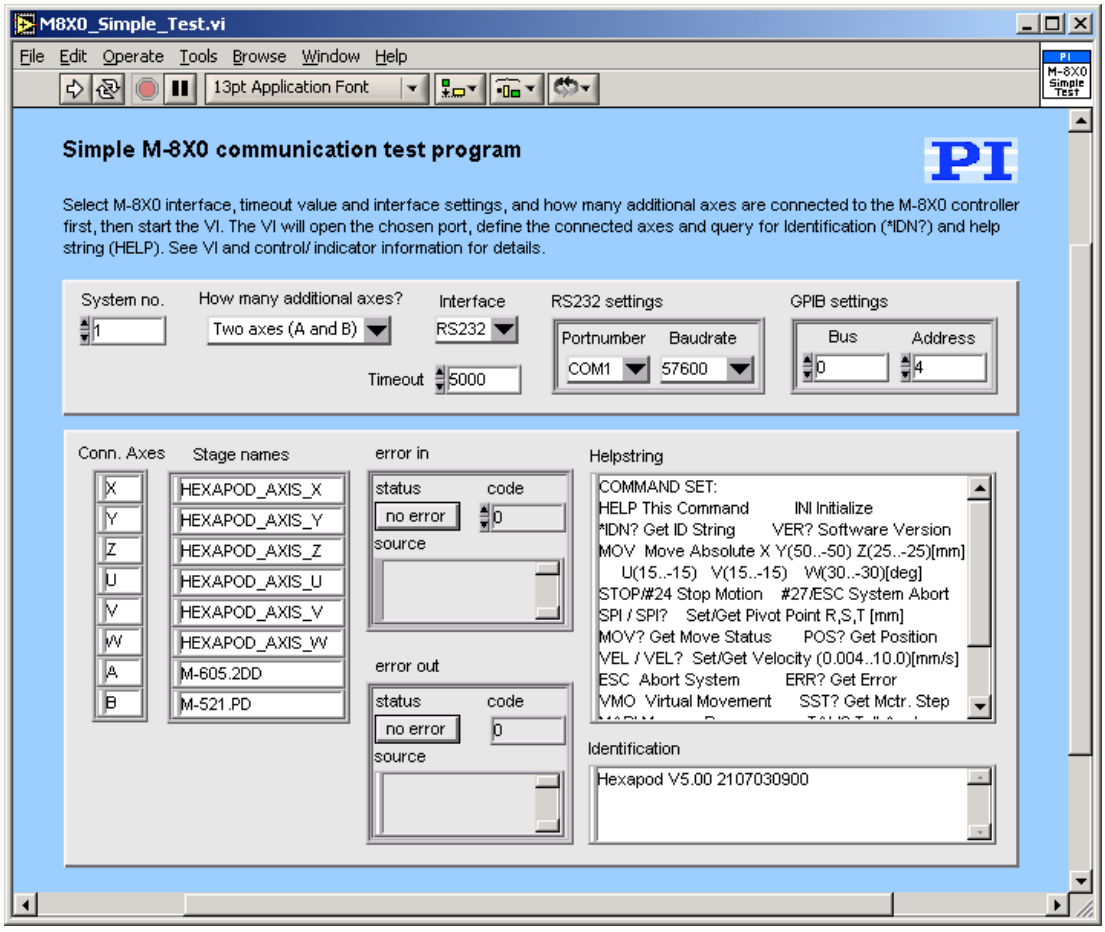
3.9. M8X0_Simple_Test.vi

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.

Select

- System number (= 1 in a one-system configuration),
- interface,
- timeout value,
- interface settings
- and if any additional axes are connected to the controller,

then start the VI. The VI will open the chosen port and query for the identification (*IDN?) and help string (HELP). Depending on the chosen value of How many additional axes? the connected axes ID's and connected stages are returned. The diagram shows how to combine the driver VIs for these tasks.



3.10. M8X0_Configuration_Setup.vi

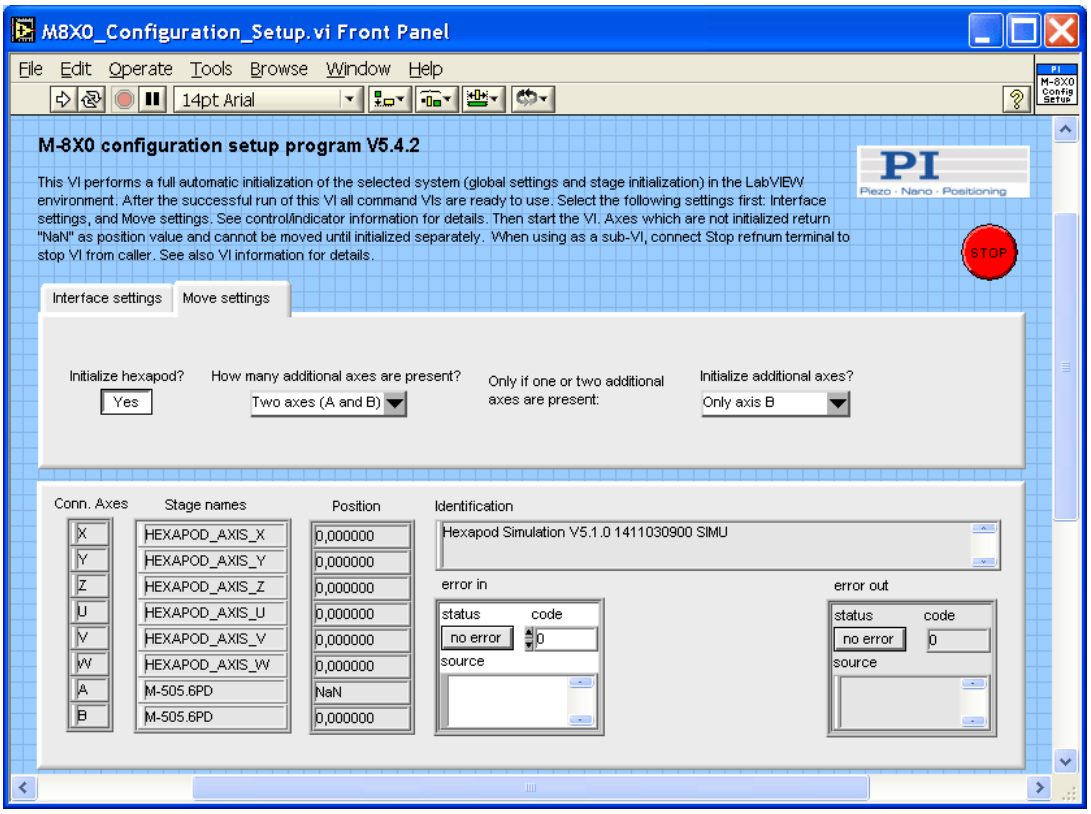
This VI is available for compatibility reasons only. For new applications, please use C887_Configuration_Setup.vi instead.

This VI performs a full 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.

It automatically detects if the connected Hexapod system is GCS 1.0 (M-840, M-850 etc.) or 2.0 (C-887 + H-810/811/824/840/850 etc.) and works for both versions. After a successful run of this VI, all command VIs are ready to use. First specify the following:

- System number (= 1 in a one-system-only configuration),
- Interface
- Timeout value
- Interface settings
- Whether the Hexapod is to be initialized
- How many additional axes are connected to the controller (you can drive up to two additional PI stages with the controller)
- Whether the additional axes are to be initialized.

Then start the VI.



“M8X0_Configuration_Setup.vi” performs the following initialization tasks:

1. Runs “PI Open Interface of one system.vi” to open the chosen port.
2. Runs “*IDN?.vi” to query for the controller identification string.
3. Defines the selected system to be "M-8X0".
4. Runs “Define connected axes.vi” with Read from controller = FALSE and Connected axes depending on the selection of How many additional axes are present?.
5. Runs “CST?.vi” to query for the connected stages.
6. Initializes the Hexapod and separate axes (if any) according to your selections Initialize hexapod? and Initialize additional axes? by running “INI hexaxes and wait until finished.vi” or “FRF.vi”.
7. Runs “POS?.vi” to query for the position of all axes.
8. Runs “ERR?.vi” to query the controller for its error status.
9. Runs “GCSTranslateError.vi” to append the error message which corresponds with a GCS error number returned by “ERR?.vi” to Source from Error out.

Axes which are designated not to be initialized report NaN as position value and cannot be moved until initialized separately.

When using as a sub-VI, connect Stop refnum terminal to stop VI from caller.

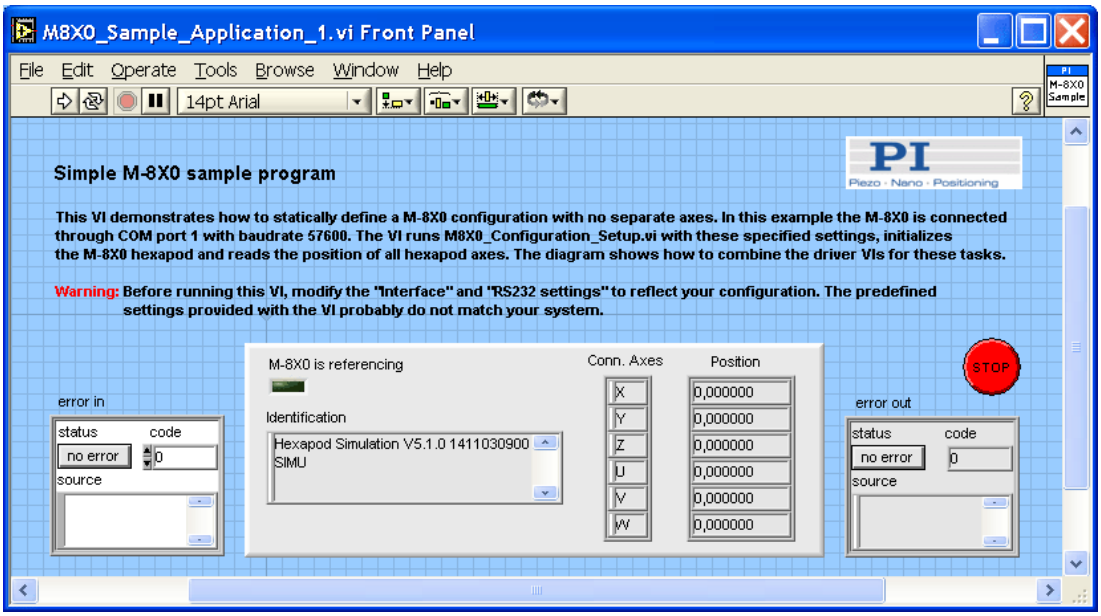
Use this VI as the initialization VI for the M-8X0 or C-887 + M-8X0 in your application.

As the initialization is a complex procedure which uses a large number of sub-VIs, M8X0_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.

3.11. M8X0_Sample_Application_1.vi

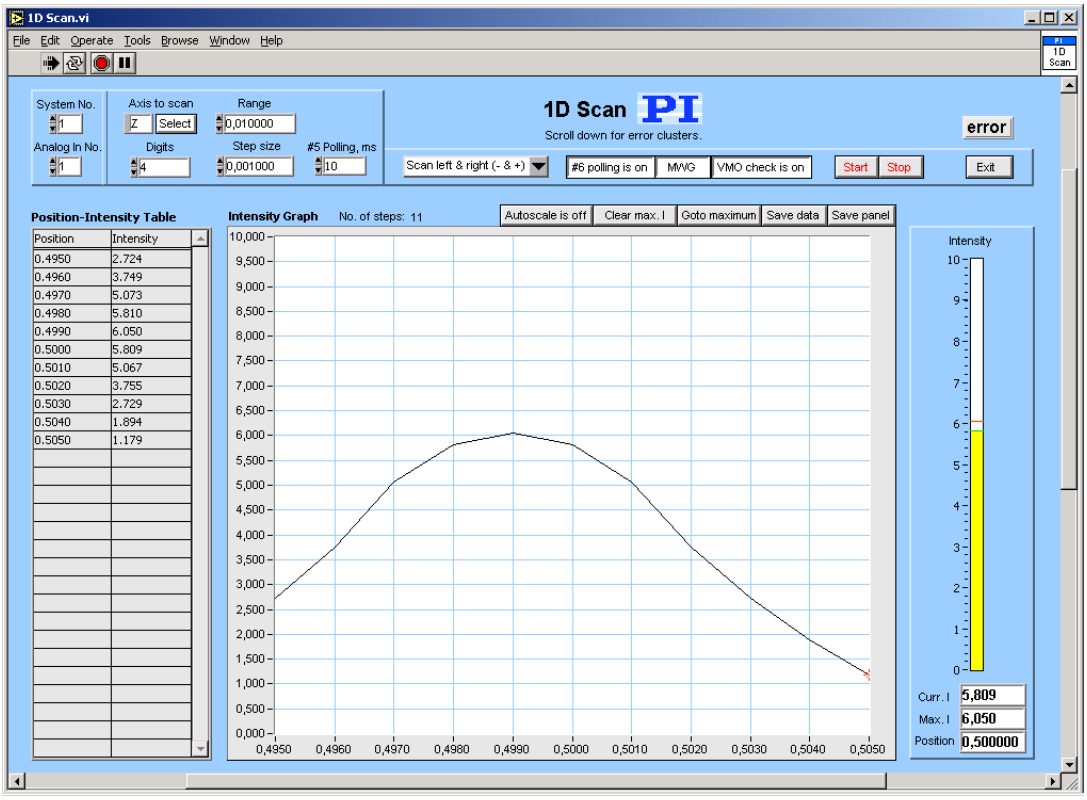
This VI demonstrates how to statically define an M-8X0 or C-887 + H-8X0 configuration with no separate axes. In this example the controller is connected through COM port 1 with baudrate 57600. The VI runs M8X0_Configuration_Setup.vi with these specified settings, initializes the Hexapod and reads the position of all Hexapod axes. The diagram shows how to combine the driver VIs for these tasks.

Warning: Before running this VI, modify the "Interface" and "RS232 settings" to reflect your configuration. The predefined settings provided with the VI probably do not match your system.



3.12. 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/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, C-884, C-887, E-761, E-861, E-871, F-206, M-8X0, Mercury_GCS

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

C-884: #6 polling? = FALSE, Use MWG? =FALSE, VMO check? =FALSE

E-761: #6 polling? = TRUE, Use MWG? = FALSE, VMO check? = FALSE

E-861: #6 polling? = FALSE, Use MWG? =FALSE, VMO check? =FALSE

E-871: #6 polling? = FALSE, Use MWG? =FALSE, VMO check? =FALSE

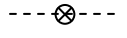
C-887, F-206: #6 polling? = TRUE, VMO check? = TRUE. For GCS syntax version = GCS 1.0 (Check with CSV?.vi. If CSV?.vi is not supported, syntax version is GCS 1.0), Use MWG? = TRUE, otherwise Use MWG? = FALSE,

M-8X0: #6 polling? = TRUE, Use MWG? = FALSE, VMO check? = TRUE

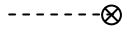
Mercury_GCS: #6 polling? = FALSE, Use MWG? =FALSE, VMO check? = FALSE

Output: 1D Array I, 1D Array X, Intensity, Position, Error out

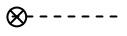
Remarks: Scan direction can be



0: *Left & right (- & +)* meaning that the scan starts at (Start position – ½ Range) and stops at (Start position + ½ Range),



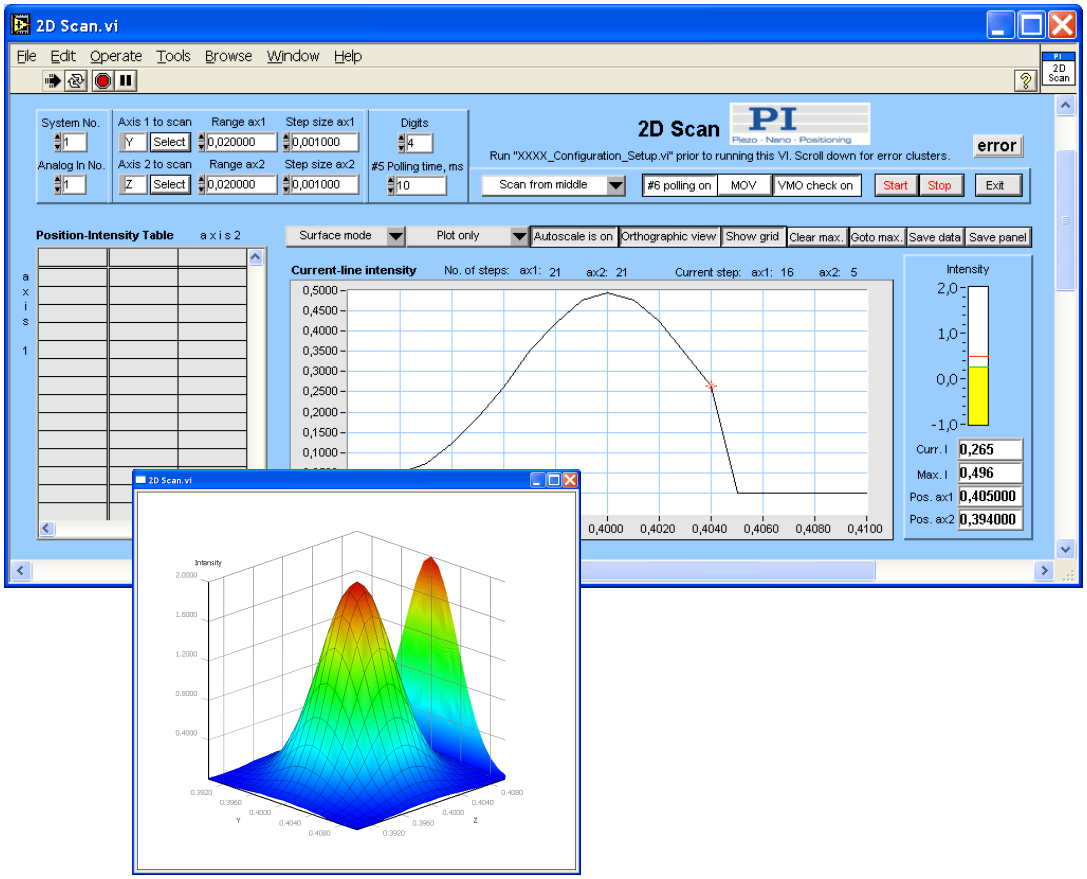
1: *To the left (-)* meaning that scan starts at (Start position – Range) and stops at Start position, or



2: *To the right (+)* meaning that scan starts at Start position and stops at (Start position + Range).

3.13. 2D Scan.vi

This VI performs a two-dimensional (2D) scan, varying the positions of two axes by the specified step sizes while an analog signal is monitored. Any combination of two axes can be selected. The analog signal is observed while scanning over a two-axis range. The direction of the scan can be selected. After the scan, all axes are returned to the start position. The range and step size for each axis can be specified. After the scan, axes 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 Board number (Board 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 IDs of the axes to scan directly into the Axis to scan controls. Then, select Range and Step size for each axis. Digits is the number of digits after the decimal point in the position values that will be sent. #5 Polling time 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 command MWG. See control descriptions below to determine if your system supports MWG. If VMO check is on, the VI will check whether all four corners of the scan area 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.

During the scan, Current-line intensity shows the intensity distribution for the line currently being scanned. When the scanning process has finished, the resulting intensity distribution is shown in a separate 3D graphics. This window stays always on top and can be minimized or closed when not needed. This graphics provides the following display options:

- Drag (click&hold left mouse button) any point to rotate graph
- Press Shift and drag any point to move graph
- Click&hold right mouse button to zoom in or out

Surface mode, Projections, and Autoscale can be chosen. If Autoscale is off, the analog level is scaled from 0-10. A scale grid can be shown (Show Grid) and the view can be changed from perspective to orthographic.

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 (axes, position and intensity information). Data will be saved in GCS Array format. With Save panel a screen copy of this VI and the graph window can be saved as a JPG file.

Position-Intensity Table will show the position values of both axes to scan and the corresponding intensity values observed during the scan. On the right side of the VI panel, the current and maximum intensity values and the current position values 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 (Axes to scan, Range, Step size, Digits, #5 polling time, #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-880, C-884, C-887, E-761, F-206, M-8X0
Input	System No. (1), Board No. (1), Axis 1 to scan (Y), Axis 2 to scan (Z), Range ax1 (0.02), Range ax2 (0.02), Step size ax1 (0.001), Step size ax2 (0.001), Digits (4), #5 polling time, ms (10), Scan direction (0: Scan from middle), #6 polling? (TRUE), Use MWG? (FALSE), VMO check? (TRUE), Exit (FALSE), Error in (no error) C-880: #6 polling? = TRUE, Use MWG? = FALSE, VMO check? = TRUE C-884: #6 polling? = FALSE, Use MWG? = FALSE, VMO check? = FALSE E-761: #6 polling? = TRUE, Use MWG? = FALSE, VMO check? = FALSE C-887, F-206: #6 polling? = TRUE, VMO check? = TRUE. For GCS syntax version = GCS 1.0 (Check with CSV?.vi. If CSV?.vi is not supported, syntax version is GCS 1.0), Use MWG? = TRUE, otherwise Use MWG? = FALSE, M-8X0: #6 polling? = TRUE, Use MWG? = FALSE, VMO check? = TRUE
Output	2D Array I, 1D Array X, 1D Array Y, Intensity, Pos. ax1, Pos. ax2, Error out

Remarks

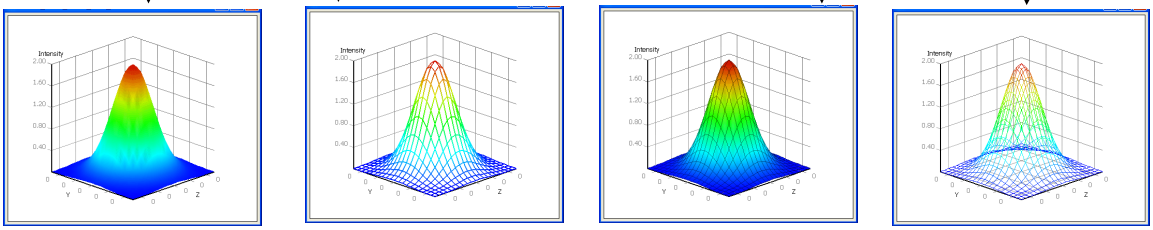
Surface mode can be

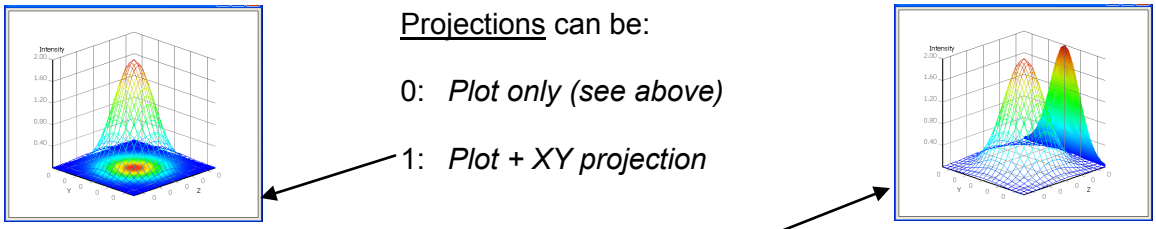
0: *Surface mode*

1: *Hidden line mode*

2: *Surface + line mode*

3: Wire mode





Projections can be:

0: *Plot only (see above)*

1: *Plot + XY projection*

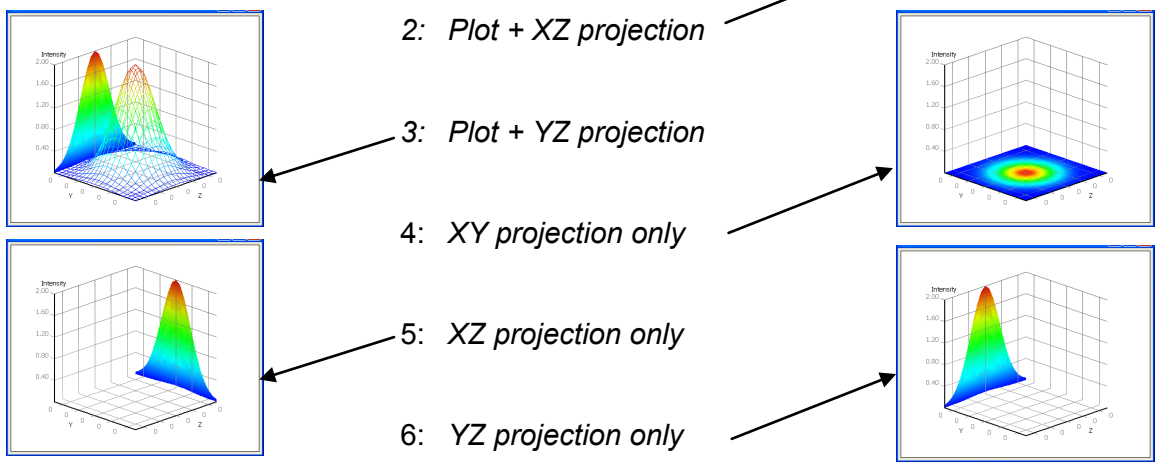
2: Plot + XZ projection

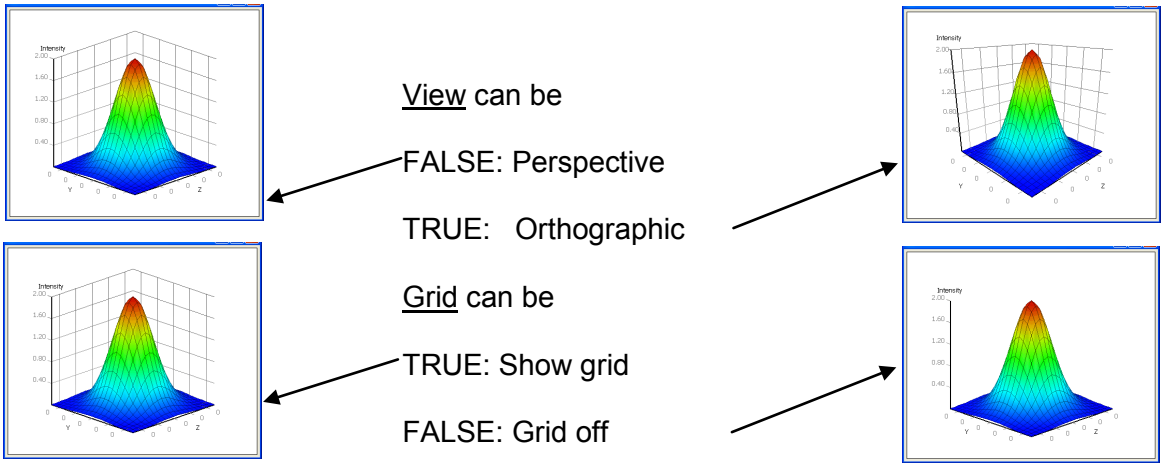
3: Plot + YZ projection

4: *XY projection only*

5: *XZ projection only*

6: *YZ projection only*





View can be

FALSE: Perspective

TRUE: Orthographic

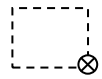
Grid can be

TRUE: Show grid

FALSE: Grid off

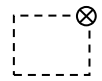
Scan direction can be:

0: *Scan from middle*
start: (Start position ax1 – ½ Range ax1, Start position ax2 – ½ Range ax2)
end: (Start position ax1 + ½ Range ax1, Start position ax2 + ½ Range ax2)



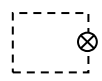
1: *Scan upper left*
start: (Start position ax1 – Range ax1, Start position ax2)
end: (Start position ax1, Start position ax2 + Range ax2)

2: *Scan upper right*
start: (Start position ax1, Start position ax2)
end: (Start position ax1 + Range ax1, Start position ax2 + Range ax2)



3: *Scan lower left*
start: (Start position ax1 – Range ax1, Start position ax2 – Range ax2)
end: (Start position ax1, Start position ax2)

4: *Scan lower right*
start: (Start position ax1, Start position ax2 – Range ax2)
end: (Start position ax1 + Range ax1, Start position ax2)



5: *Scan left*
start: (Start position ax1 – Range ax1, Start position ax2 – ½ Range ax2)
end: (Start position ax1, Start position ax2 + ½ Range ax2)

6: *Scan right*
start: (Start position ax1, Start position ax2 – ½ Range ax2)
end: (Start position ax1 + Range ax1, Start position ax2 + ½ Range ax2)

7: *Scan above*
start: (Start position ax1 – ½ Range ax1, Start position ax2)
end: (Start position ax1 + ½ Range ax1, Start position ax2 + Range ax2)

8: *Scan below*
start: (Start position ax1 – ½ Range ax1, Start position ax2 – Range ax2)
end: (Start position ax1 + ½ Range ax1, Start position ax2)

3.14. 2D_Manual_Align.vi

This VI can be used to perform a two-dimensional (2D) manual alignment to the maximum of an analog signal by varying the positions of two axes by the specified step sizes while monitoring the analog signal. Any combination of two axes can be selected. The range and step size for each axis can be specified. The analog signal is observed while making one step per selected axis with specified step size using the corresponding buttons or function keys:

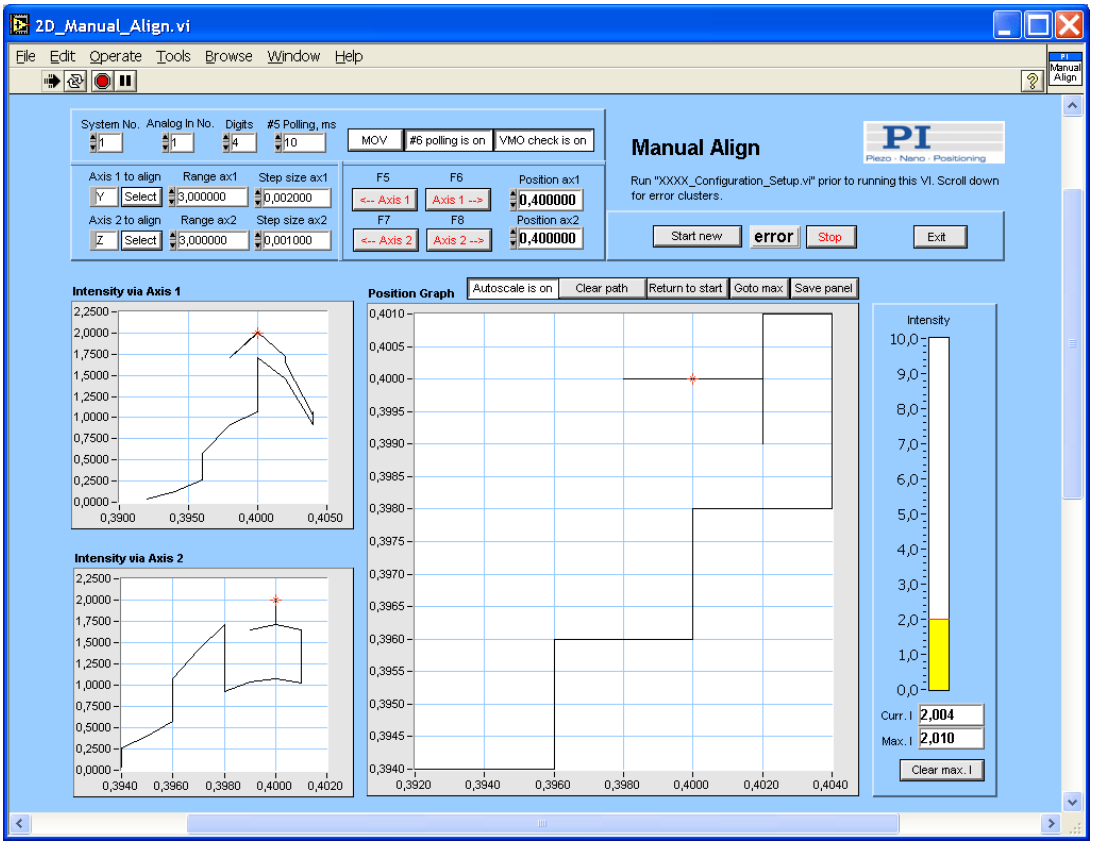
F5 (move axis 1 one step in negative direction),

F6 (axis 1 positive direction),

F7 (axis 2 negative direction)

F8 (axis 2 positive direction)

Typing in a certain position value for axes 1 or 2 is also possible. 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 (= 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 axes to align here or type the IDs of the axes to align directly into the Axes to align controls.

Then, select Range and Step size for each axis.

Digits is the number of digits after the decimal point in the position values that will be sent.

#5 Polling time 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.

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 align 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 all four corners of the maximum align area, defined by the Range values of both axes, 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/VMO?.

Press Goto max to go to the position where the maximum intensity was found during the scan. Press Return to start if you want to return to the start position. Start new clears all and checks axes, range etc. again.

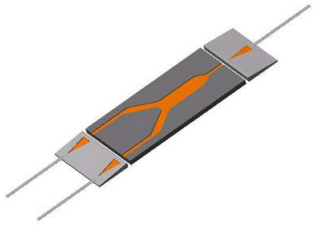
With Clear max. I the intensity value and the position value for the maximum intensity can be cleared. Clear Path clears the previous scan path in all graphs. Autoscale of the graphs can be switched on or off. With Save panel a screen copy of this VI can be saved as a JPG file.

Press Stop to stop the scan; press Exit to stop execution of this VI. When using as a sub-VI, connect the Stop refnum terminal to stop VI from caller.

Valid for	C-880, C-884, C-887, E-761, F-206, M-8X0
Input	System No. (1), Analog In No. (1), Digits (4), #5 polling time, ms (10), Use MWG? (FALSE), #6 polling? (TRUE), VMO check? (TRUE), Axis 1 to align (Y), Axis 2 to align (Z), Range ax1 (0.02), Range ax2 (0.02), Step size ax1 (0.001), Step size ax2 (0.001), Exit (FALSE), Return to start (F), Save data (F), Error in (no error) C-880: #6 polling? = TRUE, Use MWG? = FALSE, VMO check? = TRUE C-884: #6 polling? = FALSE, Use MWG? = FALSE, VMO check? = FALSE E-761: #6 polling? = TRUE, Use MWG? = FALSE, VMO check? = FALSE C-887, F-206: #6 polling? = TRUE, VMO check? = TRUE. For GCS syntax version = GCS 1.0 (Check with CSV?.vi. If CSV?.vi is not supported, syntax version is GCS 1.0), Use MWG? = TRUE, otherwise Use MWG? = FALSE, M-8X0: #6 polling? = TRUE, Use MWG? = FALSE, VMO check? = TRUE
Output	Intensity, Position ax1, Position ax2, Error out
Remarks	To make sure that the final position is the real maximum position of the analog signal distribution, run “2D Scan.vi” with Goto max = TRUE and a Step size larger than the Step size value used here before running this VI. See also Automated 2D_Align.vi for an automated alignment procedure.

3.15. 2Detector_Automated_Align.vi

This VI performs an automated alignment procedure with two optical channels (for instance, from a fiber array) seeking the position where the corresponding analog signals are maximized.



The procedure uses the firmware (internal) scan algorithms FSM, FAA (which calls FLM for GCS 2.0 automatically) and FAM (which calls FSM for GCS 2.0 automatically) and requires two optical boards (or one two-channel optical board) installed in the controller. The procedure consists of the following steps:

1. The signal connected to Analog In 1 is maximized using the FSM embedded fast alignment algorithm (2D linear scan) first with a given rough range and step size, then with a given fine range and step size
2. The signal connected to Analog In 2 is maximized by rotating the platform around the given Pivot Point (FAA algorithm), first with rough scan range and step size, then with a fine range and step size
3. The specified number of iterations (runs) is performed as follows:
   a. Fine-range FSM (linear) scan to Analog In 1 maximum is repeated
   b. Fine-range FAM (angular) scan to Analog In 2 maximum is repeated

In this way maximization of both channels is achieved even if the position of the pivot point is only roughly known.

Start the VI and select the System number (System No. = 1 in a one-system configuration) and both Analog In numbers (Analog In No. = number of the analog channel to observe). Pressing Select will open a user interface which shows a list box with all supported axes of the selected system. Choose the axes to align here or type the IDs of the axes to align directly into the Axes to align controls (must be a combination of X, Y and Z, must not contain the axis ID corresponding to the optical axis of the light path).

Iterations is the number of runs through Step 3 a and b to perform before the alignment procedure is stopped (recommended value: 4).

If #6 polling is on, the VI will poll for position changes when idle.

If VMO check is on, the VI will check whether all four corners of the maximum align area, defined by the Range values for FSM of both axes, and the given angular range for FAA are within the system workspace before starting the scan. It is assumed that all positions between these points are reachable.

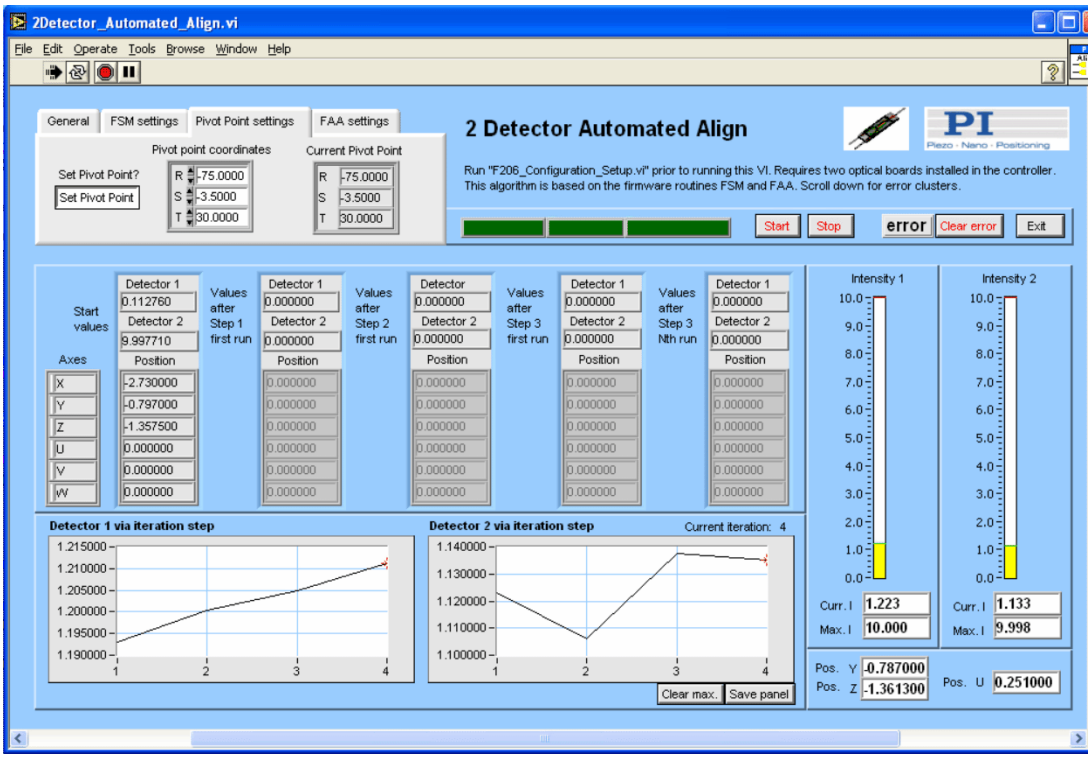
Select Range and Step size for rough and fine alignment with FSM, and a treshold level to reach.

Select whether the pivot point is to be set, and if yes, select the appropriate Pivot point coordinates (should correspond to the location of the outermost fiber to which Analog In 1 must be connected).

Also, select range (for rough and fine alignment), threshold level and velocity for the FAA algorithm. Then start the VI.

On the right side of the VI panel, the current and maximum intensity values and the current position values are shown. Start values contains the intensity and position values when starting the VI. Values after Step 1 first run contain the intensity and position values after the first (rough, followed by fine) FSM scan. Values after Step 2 first run contain the intensity and position values after the first (rough, followed by fine) FAA scan. Values after Step 3 first run contain the intensity and position values after the following FSM and FAA fine scans. Values after Step 3 Nth run contain the intensity and position values after the nth FSM and FAA fine scan. Detector 1 and 2 via Iteration Step are graphical displays of the intensity values versus the number of iteration steps.

With Clear max. the intensity value and the position value for the maximum intensity can be cleared. With Save panel a screen copy of this VI can be saved as a JPG file.



Press Stop to stop the scan; press Exit to stop execution of this VI. To use this VI as a sub-VI for your application without user interaction, wire the desired alignment parameters (Axes to scan, Ranges, Step sizes, #6 polling?, VMO check?) to the corresponding terminals, and wire a TRUE constant to the Start scan and Exit terminals. After calling the VI, it will perform the alignment procedure and finish execution. When using as a sub-VI, connect the Stop refnum terminal to stop VI from caller.

Run C887_Configuration_Setup.vi (or, for older applications, F206_Configuration_Setup.vi) before starting this VI.

Valid for	C-887, F-206
Input	System No. (1), Analog In 1 (1), Analog In 2 (2), Axis 1 to scan (Y), Axis 2 to scan (Z), #6 polling? (TRUE), VMO check? (TRUE), FSM settings (Range FSM ax1 (0.3), Range FSM ax2 (0.3), Step size FSM (0.01), Treshold FSM, V (0.2), Range ax1 fine (0.1), Range ax2 fine (0.1), Step size fine (0.001)), Set Pivot Point? (TRUE), Pivot Point coordinates (0,0,0), FAA settings (Range FAA (0.3),

Range FAA fine (0.2), Threshold FAA (0.2), VEL for FAA, mm/s (0.2)), Start (FALSE), Exit (FALSE), Error in (no error)

C-887, F-206: #6 polling? = TRUE, VMO check? = TRUE, Range FSM ax1 must be equal to Range FSM ax2, Range ax1 fine must be equal to Range ax2 fine.

Output

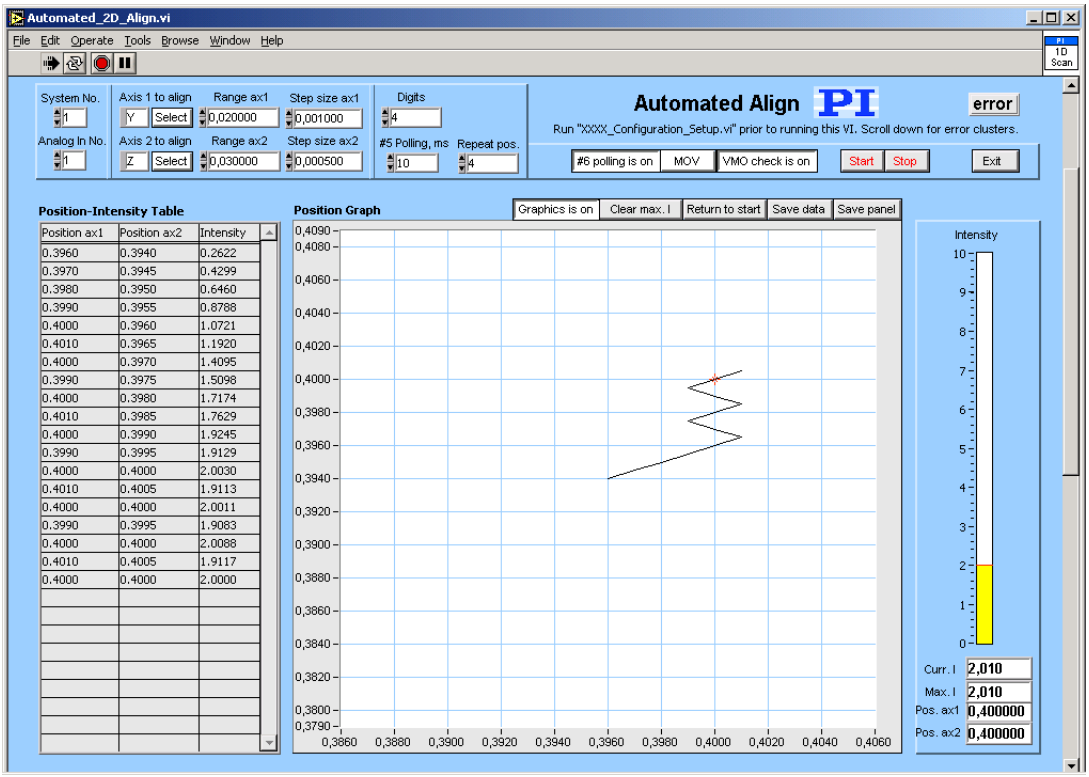
Start values (Detector 1, Detector 2, Position), Values after Step 3 Nth run (Detector 1, Detector 2, Position), Intensity 1, Intensity 2, Error out

Remarks

When starting this VI, there must be an observable gradient in the analog input, otherwise the algorithm will stop without finding the maximum.

3.16. Automated_2D_Align.vi

This VI performs a two-dimensional (2D) automated alignment seeking the maximum of an analog signal by varying the positions of two axes by the specified step sizes while monitoring the analog signal. Any combination of two axes can be selected. The range and step size for each axis can be specified. The analog signal is observed while making one step per selected axis with specified step size. After each step, the new analog signal value is compared with the analog signal value before the step. If it is smaller, the scanning motion direction for the next step of this axis is reversed. If it's larger, the next step for this axis will be performed in the same direction as before. If the same position is held for a given number of times, the alignment procedure is stopped. 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 Board number (Board 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 axes to align here or type the IDs of the axes to align directly into the Axes to align controls. Then, select Range and Step size for each axis. Digits is the number of digits after the decimal point in the position values that will be sent. #5 Polling time 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. Repeat pos. is the number of times a position value should be held before the alignment procedure is stopped. 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 align 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 all four corners of the maximum align area, defined by the Range values of both axes, 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.

During the alignment, the display of the align path can be toggled on or off. When the alignment process has finished, the commanded position values of the alignment path and the corresponding analog values are displayed in a Position-Intensity Table. With Clear max. the intensity value and the position value for the maximum intensity can be cleared.

Press Return to start if you want to return to the start position after the scan. Press Save data to save scan data (axes, position and intensity information). Data will be saved in GCS Table format. With Save panel a screen copy of this VI can be saved as a JPG file.

Position-Intensity Table will show the position values of both axes to align and the corresponding intensity values observed during the alignment. On the right side of the VI panel, the current and maximum intensity values and the current position values are shown.

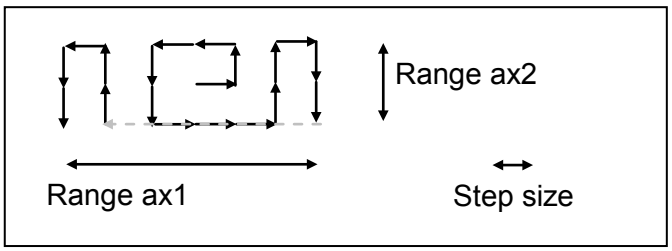
Press Stop to stop the scan; press Exit to stop execution of this VI. To use this VI as a sub-VI for your application without user interaction, wire the desired alignment parameters (Axes to scan, Range, Step size, Digits, #5 polling time, #6 polling?, Use MWG?, VMO check?) to the corresponding terminals, and wire a TRUE constant to the Start scan and Exit terminals. After calling the VI, it will perform the alignment and finish execution. When using as a sub-VI, connect the Stop refnum terminal to stop VI from caller.

Valid for	C-880, C-884, C-887, E-761, F-206, M-8X0
Input	System No. (1), Board No. (1), Axis 1 to align (Y), Axis 2 to align (Z), Range ax1 (0.02), Range ax2 (0.02), Step size ax1 (0.001), Step size ax2 (0.001), Digits (4), #5 polling time, ms (10), Repeat pos. (4), #6 polling? (TRUE), Use MWG? (FALSE), VMO check? (TRUE), Exit (FALSE), Return to start (F), Save data (F), Error in (no error) C-880: #6 polling? = TRUE, Use MWG? = FALSE, VMO check? = TRUE C-884: #6 polling? = FALSE, Use MWG? = FALSE, VMO check? = FALSE E-761: #6 polling? = TRUE, Use MWG? = FALSE, VMO check? = FALSE C-887, F-206: #6 polling? = TRUE, VMO check? = TRUE. For GCS syntax version = GCS 1.0 (Check with CSV?.vi. If CSV?.vi is not supported, syntax version is GCS 1.0), Use MWG? = TRUE, otherwise Use MWG? = FALSE, M-8X0: #6 polling? = TRUE, Use MWG? = FALSE, VMO check? = TRUE
Output	1D Array ax1, 1D Array ax2, 2D Array I, Intensity, Pos. ax1, Pos. ax2, Error out
Remarks	When starting this VI, there must be a gradient of the analog input value observable, otherwise the algorithm will stop without finding the maximum or will run out until it reaches the given range of the alignment area. To make sure that the final position is the real maximum position of the analog signal distribution, run “2D Scan.vi” with Goto max = TRUE and a Step size larger than the Step size value used here before running this VI.

3.17. Find_threshold_with_2_systems.vi

This VI scans the input and output of any optical device (having input and output on opposite sites, for instance wave guides) until a certain amount of signal ("threshold level") is found. The algorithm is based on the firmware (internal) scan algorithm FSM and requires two systems (controllers and mechanics) with one optical board installed in the second controller.

If First step? is TRUE, both systems are moved to the specified start position. The gain of the optical input is set and the intensity read. First, System 2 performs an FSM scan. If the treshold is not found by this scan, System 1 performs a step followed by another FSM scan of system 2 and so on. The steps made by System 1 are part of a linear spiral up to Range ax2, followed by a meander scan on both sides of the spiral, up to Range ax1:



Motion path of system 1

If the treshold level was found during the scan, the algorithm stops at this position, otherwise it proceeds with the scan until Range ax1 is reached, then it returns to the original position.

Start the VI and select the System numbers for System 1 (System No. = 1 in a two-system configuration) and System 2 (System No. = 2 in a two-system configuration). Select also the Analog In number for System 2 (Analog In No. = number of the analog channel to observe) and the Gain setting. Pressing Select will open a user interface which shows a list box with all supported axes of the selected systems. For both systems choose the axes to scan here or type the IDs of the axes to scan directly into the Axes to scan and FSM axes controls (must be a combination of X, Y and Z, must not contain the axis ID corresponding to the optical axis of the light path).

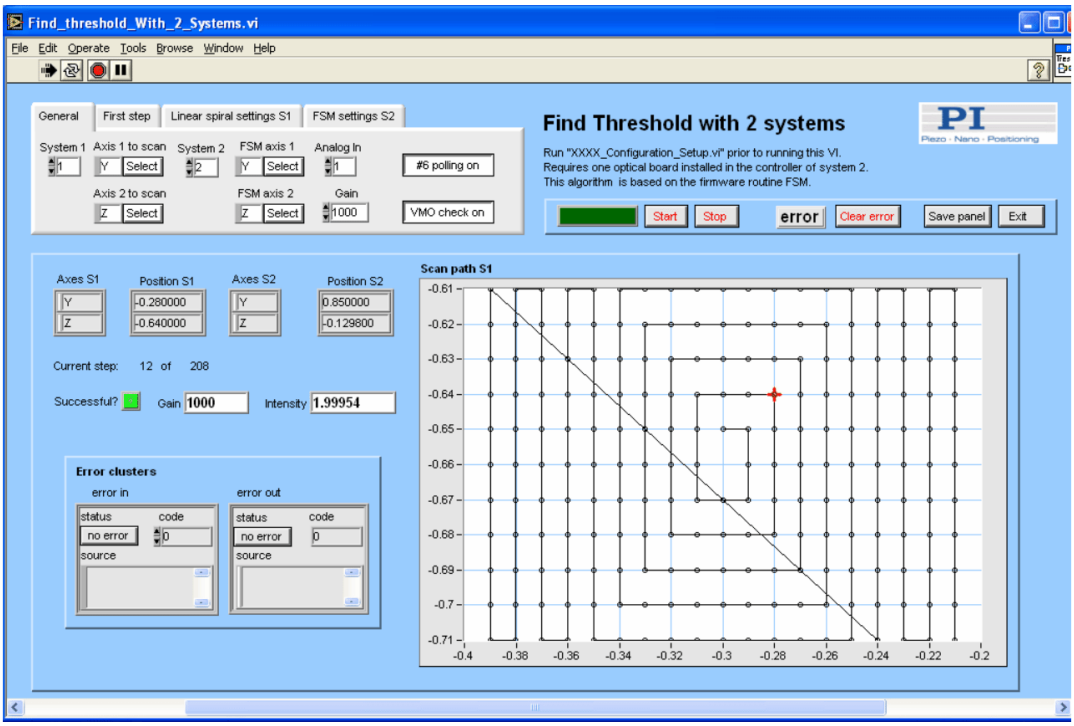
If #6 polling is on, the VI will poll for position changes when idle. If VMO check is on, the VI will check whether all four corners of the maximum scan area, defined by the Range values, are within the system workspace before starting the scan. It is assumed that all positions between these points are reachable.

Select First Step settings: if a move to a given start position of both systems is to be performed, if First Step? Is TRUE, select also the axes to move and the corresponding position values for the First Step position.

For the linear spiral that System 1 performs, select Range and Step size (see graphics above). For the FSM algorithm that System 2 executes, select Range, Step size and Threshold level. Then start the scan by pressing Start.

The Axes and Position indicators display the position valus of the axes that are being moved. Scan Path S1 shows the linear spiral that System 1 will perform if the Threshold level is not reached before it is completed. Successful? reports if the scan was successful, i.e. if the threshold level has been found. Gain and Intensity are the values when the algorithm has finished.

With Save panel a screen copy of this VI can be saved as a JPG file.



Press Clear Error to clear any error status if an error occurred and you want to proceed without stopping the VI. Press Stop to stop the scan; press Exit to stop execution of this VI. To use this VI as a sub-VI for your application without user interaction, wire the desired alignment parameters (Axes to scan, Analog In, Gain, #6 polling?, VMO check?, First Step settings, Linear Spiral settings, FSM settings) to the corresponding terminals, and wire a TRUE constant to the Start scan and Exit terminals. After calling the VI, it will perform the alignment and finish execution. When using as a sub-VI, connect the Stop refnum terminal to stop VI from caller.

Run C887_Configuration_Setup (or, for older applications, F206_Configuration_Setup.vi) before starting this VI.

Valid for	C-887, F-206
Input	System 1(1), Axis 1 to scan (Y), Axis 2 to scan (Z), System 2 (2), FSM axis 1 (Y), FSM axis 2 (Z), Analog In (1), Gain (1000), #6 polling? (TRUE), VMO check? (TRUE), First Step settings (First step? (FALSE), Axes S1 (empty string array), Position S1 (empty num. array, Axes S2 (empty string array), Position S2 (empty num. array)), Linear Spiral settings (Range ax 1 (0.2), Range ax2 (0.1), Step size (0.01)), FSM settings (Range FSM ax1 (0.3), Range FSM ax2 (0.3), Step size FSM (0.01), Treshold FSM, V (0.2), Start (FALSE), Exit (FALSE), Error in (no error) C-887, F-206: #6 polling? = TRUE, VMO check? = TRUE, Range FSM ax1 must be equal to Range FSM ax2
Output	Successful?. Intensity, Gain, Axes S1, Position S1, Axes S2, Position S2, Error out
Remarks	Make sure that System 2 is the controller which has an Optical Board with the given Analog In No. installed.

3.18. Omega_Device_Automated_Align.vi

This VI allows the automated alignment of an optical device or fiber arrangement with input and output on the same side (so-called “Omega Device”) to the maximum of an analog signal.

The special difficulty in aligning such a device is that it is a coupled alignment (in contrast to a 2 Channel alignment where each channel has it’s own input and output and can be aligned separately using 2 optical boards or power meters): no output signal is available until both input AND output are aligned. This sample program shows how to perform such an alignment quickly and easily. The algorithm is based on the firmware (internal) scan algorithms FIO, FSM and FAA (which calls FLM automatically for GCS 2.0) and requires one optical board installed in the controller.

If First step? is TRUE, the system is moved to the specified start position. Then, the VI uses the firmware FIO fast scan algorithm to find first light (“threshold level”). It then optimizes the signal by running the FSM algorithm with given range and step size, followed by a rotation of the platform around the given Pivot Point (FAA algorithm, two consecutive runs with rough and fine scan ranges, respectively). The Pivot Point must be set to either the location of the input or the output channel (the exact position needs to be known only roughly).

For an optimal alignment result, FSM and FAA are repeated with the given fine scan parameters and an auto ranging function is performed after each step. When the alignment with the C-887/F-206 is finished, an optional linear fine alignment with a 3-axis NanoCube piezo stage can be performed.

Start the VI and select the

- System number (System No. = 1 in a one-system configuration),
- Analog In number (Analog In No. = number of the analog channel to observe) and
- Gain of the Analog Input.

Pressing Select will open a user interface which shows a list box with all supported axes of the selected system. Choose the axes to scan here or type the IDs of the axes to scan directly into the Axes to scan controls (must be a combination of X, Y and Z, must not contain the axis ID corresponding to the optical axis of the light path).

If #6 polling is on, the VI will poll for position changes when idle. If VMO check is on, the VI will check whether all four corners of the maximum align area, defined by the Range values for FSM of both axes, and the given angular range for FAA are within the system workspace before starting the scan. It is assumed that all positions between these points are reachable.

Select First Step settings: whether a move to a given start position is to be performed and whether First Step? Is TRUE; also select the axes to move and the corresponding position values for the First Step position.

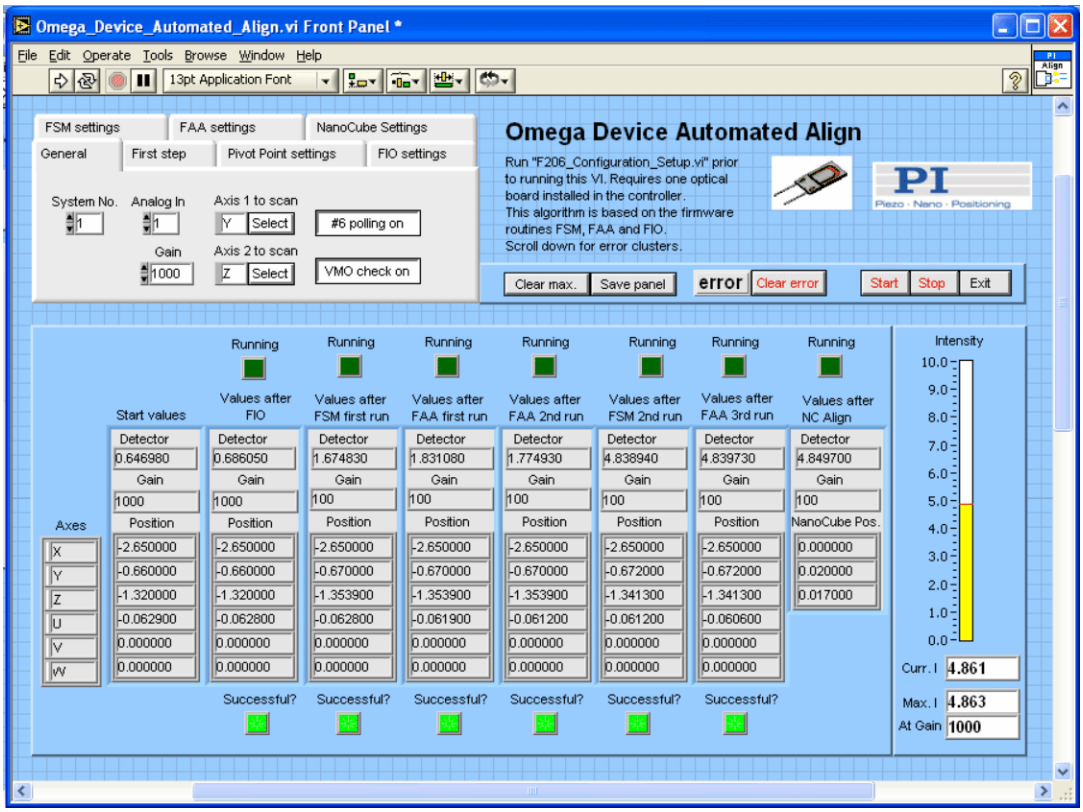
Select whether the Pivot Point is to set, and if yes, select the appropriate Pivot Point coordinates (should correspond to the location of the input or the output).

Select FIO settings: Linear and Angular range, Step size and Threshold. See description of FIO.vi for details.

Then, select Range and Step size for rough and fine alignment with FSM, and a threshold level to reach (FSM settings).

Also select range (for rough and fine alignment), threshold level and velocity for the FAA algorithm. Then start the VI.

If you want to add a NanoCube alignment, set “NanoCube Align?” to TRUE and select Range and Step size. NanoCube axes to command are determined automatically depending on the selected C-887/F-206 axes to align. Finally, select whether you want the VI to perform a NanoCube Scan to visualize the intensity distribution after the alignment.



On the right side of the VI panel, the current and maximum intensity values and the current position values and gain setting are shown. Start values contains the intensity, gain and position values when the VI was started.

Running? indicates that the corresponding step is running, Successful? indicates whether it was successful or not.

Values after FIO contain the intensity, gain and position values after the FIO scan. Values after FSM first run and Values after FAA first run contain the intensity, gain and position values after the first rough FSM and FAA scan. Values after FAA 2nd run and Values after FSM 2nd run contain the intensity, gain and position values after the first fine FAA and FSM scan. Values after FSM 3rd run contain the intensity, gain and position values after the final FSM fine scan.

With Clear max. the intensity value and the position value for the maximum intensity can be cleared. With Save panel a screen copy of this VI can be saved as a JPG file.

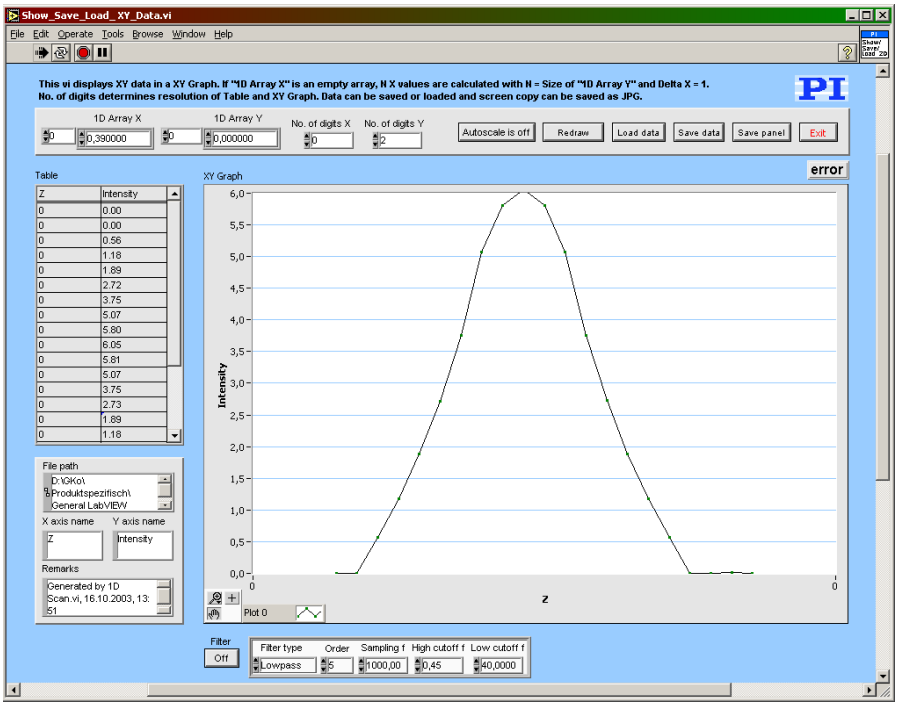
Press Clear Error to clear any error status if an error has occurred and you want to proceed without stopping the VI. Press Stop to stop the scan; press Exit to stop execution of this VI. To use this VI as a sub-VI for your application without user interaction, wire the desired alignment parameters (Axes to scan, Analog In and Gain settings, #6 polling?, VMO check?, First Step settings, Pivot Point settings, FIO, FSM and FAA settings, and NanoCube settings) to the corresponding terminals, and wire a TRUE constant to the Start scan and Exit terminals. After calling the VI, it will perform the alignment and finish execution. When using as a sub-VI, connect the Stop refnum terminal to stop VI from caller.

Run C887_Configuration_Setup.vi (or, for older applications, F206_Configuration_Setup.vi) before starting this VI.

Valid for	C-887, F-206
Input	System No. (1), Analog In (1), Gain (1000), Axis 1 to scan (Y), Axis 2 to scan (Z), #6 polling? (TRUE), VMO check? (TRUE), First Step settings (First step? (FALSE), Axes (empty string array), Position (empty num. array)), Set Pivot Point? (TRUE), Pivot Point coordinates (0,0,0), FIO settings (Linear range FIO, mm (0.3), Angular range FIO, deg (0.3), Step size FIO lin. spiral (0.01), Threshold FIO, V (0.2)), FSM settings (Range FSM ax1 (0.3), Range FSM ax2 (0.3), Step size FSM (0.01), Treshold FSM, V (0.2), Range ax1 fine (0.1), Range ax2 fine (0.1), Step size fine (0.001)), FAA settings (Range FAA (0.3), Range FAA fine (0.2), Threshold FAA (0.2), VEL for FAA, mm/s (0.2)), NanoCube settings (NanoCube Align? (F), Range NC ax1 (0.04), Range NC ax2 (0.04), Step size, mm (0.0005), NanoCube Scan? (F)), Start (FALSE), Exit (FALSE), Error in (no error) C-887, F-206: #6 polling? = TRUE, VMO check? = TRUE, Range FSM ax1 must be equal to Range FSM ax2, Range ax1 fine must be equal to Range ax2 fine.
Output	Start values (Detector, Position, Gain), Values after FAA 3rd run (Detector, Position, Gain), Values after NC align (Detector, Position, Gain), Intensity, Gain, Error out
Remarks	When starting this VI, there must be a gradient of the analog input value observable, otherwise the algorithm will stop without finding the maximum.

3.19. 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.



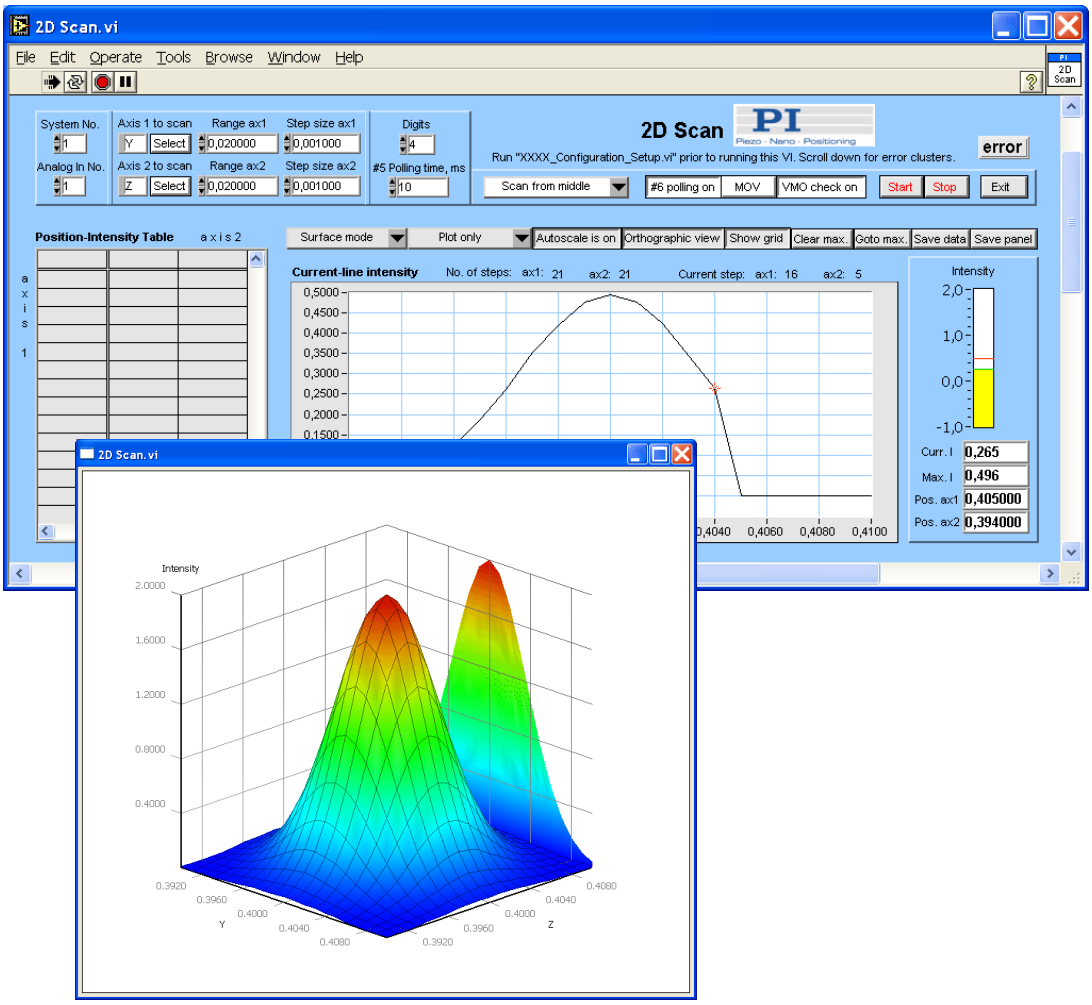
If data (1D Array X, 1D Array Y) are sent to the VI via the corresponding connectors, the VI will display the corresponding graphics after being called. To load data at runtime, press the Load data button. A dialog will pop up where a data file to open can be selected. The VI can read data in GCSArray, 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-413, C-843, C-866, C-867, C-880, C-884, C-887, E-517, E-709, E-712, E-725, E-753, E-755, E-761, E-861, E-871, F-206, M-8X0, Mercury_GCS. 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	

3.20. Show_Save_Load_XYZ_Data.vi

This VI displays XYZ data in a 3D Graph by opening a separate window which stays always on top. If 1D Array X or 1D Array Y is an empty array, N X values and M Y values are calculated with N = Size of dimension 0 of 2D Array Z, M = Size of dimension 1 of 2D Array Z and Delta X, Delta Y = 1. No. of digits determines the resolution of X-Y-Z Data table and XYZ Graph. Data can be saved or loaded and a screen copy can be saved as JPG.



If data (1D Array X, 1D Array Y, 2D Array Z) are sent to the VI via the corresponding connectors, the VI will show 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. The 3D graphic is a separate window and provides the following display options:

- Drag (click&hold left mouse button) any point to rotate graph
- Press Shift and drag any point to move graph
- Click&hold right mouse button to zoom in or out

Suface mode (Surface, Hiden Line, Surface+Line or Wire), Projections (plot only, plot + XY/XZ/YZ projections, XY/XZ/YZ projections only), and Autoscale can be chosen. If Autoscale is off, the analog level is scaled from 0-10. A scale grid can

be shown (Show Grid) and the view can be changed from perspective to orthographic.

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, Z axis name and Remarks. With Save panel a screen copy of this VI and the graph window can be saved as a JPG file.

X-Y-Z data will show the values of the X and Y axis and the corresponding Z value.

Press Exit to stop execution of this VI.

Valid for	C-880, C-884, C-887, E-761, F-206, M-8X0
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	For surface modes and projection modes, see description of “2D Scan.vi”.

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 6.3.1	-
C-413	6.5.0	1.0.0
C-702	4.0.0	1.4.0
C-843	2.01 – 2.02 2.05 – 2.06 3.1.2., 3.1.2a 3.4.3 3.6.1 5.7.4	MC-DLL 1.0.2.2 MC-DLL 1.0.2.3 MC-DLL 1.0.2.3 MC-DLL 1.0.2.8 MC-DLL 1.0.2.8 GCS_DLL 1.3.1 GCS_DLL 2.0.0
C-843.PM	3.1.0 3.4.3a 3.6.2	MC-DLL 1.0.2.5 MC-DLL 1.0.2.5 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 5.7.2 6.4.0	1.1.0.0 C-867: 2.0.0.0 C-867.160: 1.0.0.0 For USB: PI_GCS2_DLL.dll V2.16.0.0
C-880	1.1 1.2 2.04 2.05 – 2.06 3.2.0	2.00 2.10 2.20 2.21 2.40
C-880K005	2.06	1.0
C-880K006	2.06	1.0
C-880K007	2.06	1.0
C-884	6.2.0	1.0.0
C-887	6.0.0 6.6.0	1.1.1 1.5.0
E-516	1.0 – 2.02 2.05 – 2.06	DSP V3.01, MCU V5 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
	6.3.1	For USB: PI_GCS2_DLL.dll V2.16.0.0
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-channel	3.4.4 (a, b)	2.13
E-712	5.3.1	1.0.1.0
	5.8.2	10.00.06.00
	6.31	For USB: PI_GCS2_DLL.dll V2.16.0.0
E-725	5.5.0	02.00.03.00
		For USB: E7XX_GCS2_DLL.dll V2.2.0
	6.3.1	For USB: PI_GCS2_DLL.dll V2.16.0.0
E-753	5.2.0	1.0.0
	6.3.1	For USB: PI_GCS2_DLL.dll V2.16.0.0
E-755	5.1.0	2.0.4.1 E7XX_GCS2_DLL.dll V1.1.0
E-761	3.5.0	1.0.0
	5.4.1	2.0.1
E-861	5.4.0	6.0
	5.8.1	7.0, GCSTranslator.dll V 1.6.0.2
E-816	2.01 – 2.06	2.02
	5.3.0	2.1.1
	5.7.1	3.2.0
E-870	6.1.0	1.0.0
E-871	6.3.0	01.000
F-206	1.1 – 2.06	Fhx0035 and higher
	5.4.2	Fhx0035 and higher, FHXF6 recommended
Hydra/ Pollux	6.1.0	Firmware Hydra: 3.3.1HydraPollux_GCS_DLL: 1.0.0
	6.4.1	Firmware Hydra: 3.3.1 Firmware Pollux: 4.2.3 Firmware Pollux NT: 2.3.4 HydraPollux_GCS_DLL: 2.0.0
M-840	2.03 – 2.06	Hex0037 and higher
	2.2.0	Hex0037 and higher, Hex0045 recomm.
	3.0.1	Hex0037 and higher, Hex0050 recomm.
	3.1.1	Hex0037 and higher, Hex0051 recomm.
	5.4.2	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
Mercury_ GCS	5.9.0	PI_GCS2_DLL 2.9.0, Firmware 1.2.0.0

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	Invalid 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	Command is not allowed while the affected axis is in motion.
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
92	Stage is not capable of following the master. Check the gear ratio(SRA).
93	This command is not allowed while the affected axis or its master is in motion.
94	Servo cannot be switched on when open-loop joystick control is enabled.
95	This parameter cannot be changed in current servo mode.
96	Unknown stage name
97	Invalid length of value (too much characters)
98	AutoZero procedure was not successful
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, some controllers need 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
500	The "red knob" is still set and disables system
501	The "red knob" was activated and still disables system - reanimation required
502	Position consistency check failed
503	Hardware collision sensor(s) are activated
504	Strut following error occurred, e.g. caused by overload or encoder failure

505	One sensor signal is not valid
506	Servo loop was unstable due to wrong parameter setting and switched off to avoid damage.
507	digital connection to external spi slave device is lost
530	A command refers to a coordinate system that does not exist
531	A command refers to a coordinate system that has no parent node
532	Attempt to delete or change a coordinate system that is in use
533	Definition of a coordinate system is cyclic
536	Coordinate system cannot be defined as long as Hexapod is in motion
537	Coordinate system type is not intended for manual enabling
539	A coordinate system cannot be linked to itself
540	Coordinate system definition is erroneous or not complete (replace or delete it)
542	The coordinate systems are not part of the same chain
543	Unused coordinate system must be deleted before new coordinate system can be stored
544	With this coordinate system type SPI usage is not supported
545	Soft limits invalid due to changes in coordinate system
546	Coordinate system is write protected
547	Coordinate system cannot be changed because its content is loaded from a configuration file
548	Coordinate system may not be linked
549	A KSB-type coordinate system can only be rotated by multiples of 90 degrees
551	This query is not supported for this coordinate system type
552	This combination of work and tool coordinate systems does not exist
553	The combination must consist of one work and one tool coordinate system
554	This coordinate system type does not exist
555	BasMac: unknown controller error
556	No coordinate system of this type is enabled
557	Name of coordinate system is invalid
558	File with stored CS systems is missing or erroneous
559	File with leveling CS is missing or erroneous
601	Not enough memory
602	Hardware voltage error
603	Hardware temperature out of range
604	Position error of any axis in the system is too high
606	Maximum value of input signal has been exceeded
607	Value is not integer
700	Command not allowed in external mode
710	External mode communication error
715	Invalid mode of operation

716	Firmware stopped by command (#27)
717	External mode driver missing
718	Missing or incorrect configuration of external mode
719	External mode cycletime invalid
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)
1006	Invalid identifier (invalid special characters, ...)
1007	Variable or argument not defined
1008	Controller is (already) running a macro
1009	Invalid or missing operator for condition. Check necessary spaces around operator.
1010	No answer was received while executing WAC/MEX/JRC/...
1011	Command not valid during macro execution
1024	Motion error: position error too large, servo is switched off automatically
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
1071	User Profile Mode: Out of memory
1072	User Profile Mode: Cluster is not assigned to this axis
1073	Unknown cluster identifier
1090	There are too many open tcpip connections
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
4008	FW Parameter CRC wrong
4009	FW CRC wrong
5000	PicoCompensation scan data is not valid

5001	PicoCompensation is running, some actions can not be executed during scanning/recording
5002	Given axis can not be defined as PPC axis
5003	Defined scan area is larger than the travel range
5004	Given PicoCompensation type is not defined
5005	PicoCompensation parameter error
5006	PicoCompensation table is larger than maximum table length
5100	Common error in Nexline firmware module
5101	Output channel for Nexline can not be redefined for other usage
5102	Memory for Nexline signals is too small
5103	RNP can not be executed if axis is in closed loop
5104	relax procedure (RNP) needed
5200	Axis must be configured for this action
6000	Invalid preset value of absolute sensor
6001	Error while writing to sensor
6002	Error while reading from sensor
6003	Checksum error of absolute sensor
6004	General error of absolute sensor
6005	Overflow of absolute sensor position
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!)
-62	Host not found
-63	Device already connected
-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: position error too large, servo is switched off automatically
-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.
-1077	Version of parameter DAT file does not match the required version. Current files are available at www.pi.ws.
-1078	Cannot write to parameter DAT file to store user defined stage type.
-1079	Cannot create parameter DAT file to store user defined stage type.
-1080	Parameter DAT file does not have correct revision.
-1081	User stages DAT file does not have correct revision.
-1082	Timeout Error. Some lengthy operation did not finish within expected time.
-1083	A function argument has an unexpected datatype.
-1084	Length of data arrays is different.
-1085	Parameter value not found in parameter DAT file.

6. Index

N

O

P

R

S

T

U

V

W

End of document