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

- **C-663.11 Mercury Step**
  Networkable Single-Axis Stepper Motor Controller with GCS Firmware

- **C-663.12 Mercury Step**
  Networkable Single-Axis Stepper Motor Controller with GCS Firmware

- **C-863.11 Mercury**
  Networkable Single-Axis DC-Motor Controller with GCS Firmware
Table of Contents

0. DISCLAIMER ............................................................................................................................... 4

1. INTRODUCTION ..................................................................................................................... 4
   1.1. PI GENERAL COMMAND SET (GCS) ............................................................................. 4
   1.2. SCOPE OF THIS MANUAL .......................................................................................... 5
   1.3. VI STRUCTURE ........................................................................................................... 6
   1.4. WORKING WITH TWO PI PRODUCTS WHICH UNDERSTAND PI’S GENERAL COMMAND SET (GCS) IN LABVIEW ...................................................................................... 8
   1.5. SOFTWARE UPDATES ................................................................................................ 9
   1.6. FIRST STEPS FOR GCS-COMPATIBLE PI CONTROLLERS ..................................... 10

2. LOW LEVEL VIS ..................................................................................................................... 12
   2.1. ANALOG CONTROL.LLB .......................................................................................... 12
   2.2. COMMUNICATION.LLB ............................................................................................ 13
   2.3. FILE HANDLING.LLB .............................................................................................. 23
   2.4. GENERAL COMMAND.LLB ..................................................................................... 25
   2.5. JOYSTICK.LLB ........................................................................................................ 44
   2.6. LIMITS.LLB ............................................................................................................... 49
   2.7. MACROS.LLB ............................................................................................................ 59
   2.8. OLD COMMANDS.LLB ............................................................................................ 65
   2.9. OPTICAL OR ANALOG INPUT.LLB .......................................................................... 66
   2.10. SCAN SUPPORT.LLB ................................................................................................ 70
   2.11. SPECIAL COMMAND.LLB ...................................................................................... 72
   2.12. SUPPORT.LLB .......................................................................................................... 107

3. HIGH LEVEL VIS .................................................................................................................... 121
   3.1. PI TERMINAL.VI ........................................................................................................ 121
   3.2. MERCURY_GCS_SIMPLE_TEST.VI .......................................................................... 127
   3.3. MERCURY_GCS_CONFIGURATION_SETUP.VI ............................................................ 128
   3.4. MERCURY_GCS_SAMPLE_APPLICATION_1.VI ............................................................ 131
   3.5. MERCURY_GCS_SAMPLE_APPLICATION_1_WITH_MOVE.VI .................................. 132
   3.6. MERCURY_GCS_SAMPLE_APPLICATION_2A.VI ......................................................... 133
   3.7. MERCURY_GCS_SAMPLE_APPLICATION_DAIryCHAIN.VI ...................................... 134
   3.8. MERCURY_GCS_SAMPLE_APPLICATION_USBDAIryCHAIN.VI .................................. 135
   3.9. JOYSTICK OPERATION.VI .......................................................................................... 136
   3.10. SHOW_SAVE_LOAD_XY_DATA.VI ........................................................................... 139
   3.11. 1D SCAN.VI .............................................................................................................. 140

4. APPENDIX A ........................................................................................................................ 143
Physik Instrumente (PI) GmbH & Co. KG is the owner of the following trademarks:
PI®, PIC®, PICMA®, PILine®, PiezoWalk®, NEXACT®, NEXLINE®, NanoCube®, Picoactuator®, PInano®, PIMag®
The following designations are protected company names or registered trademarks of third parties:
Windows, NI LabVIEW.
LabVIEW is a trademark of National Instruments. Neither the driver Software, nor any software programs or other goods or services offered by PI, are affiliated with, endorsed by, or sponsored by National Instruments.
PI owns the following patents or patent applications for the technology field Piezo Stepping Drive (PiezoWalk®, NEXACT®, NEXLINE®):

PI owns the following patents or patent applications for the technology field Multilayer Piezo Actuators (PICMA®):

PI owns the following patents or patent applications for the technology field Ultrasonic Piezo Motors (PILine®):
- Germany: DE102004024656A1, DE102004044184B4, DE102004059429B4, DE102005010073A1,
- DE102005039357B4, DE102005039358A1, DE102006041017B4, DE102008012992A1, DE102008023478A1,
- DE102008058484A1, DE102010022812A1, DE102010047280A1, DE102010055848, DE102011075985A1,
- DE102011082200A1, DE102011087542B3, DE102011087542B3, DE102011108175, DE102012201863B3,
- DE19522072C1, DE19938954A1

Europe:

USA:
- US8482185B2

Japan:

China:
- ZL200380108542.0, ZL200580015994.3, ZL200580029560.9, ZL2006800036995.6, ZL200680007223.4,
- ZL200680030007.1, ZL200680042853.5

International patent applications:

PI owns the following patents or patent applications for the technology field Magnetic Direct Drives (PIMag®):
- WO212146709A2, DE102012207082A1

PI owns the following patents or patent applications for the technology field Piezo Inertia Drives (PIShift, PiezoMike):

Software products that are provided by PI are subject to the General Software License Agreement of Physik Instrumente (PI) GmbH & Co. KG and may incorporate and/or make use of third-party software components. For more information, please read the General Software License Agreement and the Third Party Software Note linked below.

General Software License Agreement
Third Party Software Note
Release: 6.14.0
File: Mercury_GCS_LabVIEW_MS206E_6140.doc
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 operating 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 achieve 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. 9 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-precision 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. Vl 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.vi 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:

```
  1  2  3  4  5  6  7  8  9 10 11 12 13 14 15 16 17 18 19 20
```

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 4 on p. 143 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 wherever 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\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/](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/](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. Mercury with GCS firmware

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

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

Step 3:

WARNINGS

Mercury_GCS_Configuration_Setup.vi May Cause Move

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

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

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

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

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

Default IDs:

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

Analog input channels: 1 to 6. Analog input channels to be queried with TAV?.vi: 1 to 4. Lines 1 to 4 can also be used for digital input (see Mercury User manual for pinout)

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

Digital output lines: 1 to 4 (see Mercury User manual for pinout)

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

Default position unit is mm, default velocity unit is mm/s.

GCS syntax version: 2.0
2. Low Level VIs

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

2.1. Analog control.lib

2.1.1. Analog FGlobal.vi

This VI works as a functional global variable for VI references
Valid for Analog Systems, but must be present for all other systems, too.

2.1.2. Analog Functions.vi

This vi calls Analog Functions (dyn).vi functions dynamically during runtime, depending on "String to send".
Valid for Analog Systems (but must be present for all other systems, too)

2.1.3. Analog Receive String.vi

This VI works as an old style global variable for String out.
Valid for Analog Systems (but must be present for all other systems, too)

2.1.4. Global Analog.vi

A global variable which contains setup information for analog systems.
Valid for Analog systems (but must be present for all other systems, too).
2.2. Communication.lib

2.2.1. Close connection if open.vi

This VI checks if the connection to the selected system is already open and, if it is, it closes this connection.

Valid for all systems.

Valid for C-413, C-702, C848, C-867, C-877, C-880, C-880K005, C-884, C-887, C-891, E-516, E-517, E-518, E-709, E-712, E-725, E-727, E-753, E-754, E-755, E-816, E-861, E-871, E-873, F-206, M-8X0, Mercury_GCS (but must be present for C-885 also)

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. The baudrate is set via the DIP switches on the controller front panel. See the controller User manual for details.
C-877: Input and output HW handshake must be FALSE. All other controls = default. Check Controller User Manual for supported baud rates.
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 TRUE. All other controls = default. The baudrate is set via IFC/IFS.vi. See the controller User manual for details.
C-891: 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-518: Input and output HW handshake must be TRUE. Not available for Interface = 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.

2.2.2. Find baudrate.vi

Opens COM port of given system with valid baudrates until status of Error out is false.

Valid for C-413, C-702, C848, C-867, C-877, C-880, C-880K005, C-884, C-887, C-891, E-516, E-517, E-518, E-709, E-712, E-725, E-727, E-753, E-754, E-755, E-816, E-861, E-871, E-873, F-206, M-8X0, Mercury_GCS (but must be present for C-885 also)

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. The baudrate is set via the DIP switches on the controller front panel. See the controller User manual for details.
C-877: Input and output HW handshake must be FALSE. All other controls = default. Check Controller User Manual for supported baud rates.
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 TRUE. All other controls = default. The baudrate is set via IFC/IFS.vi. See the controller User manual for details.
C-891: 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-518: Input and output HW handshake must be TRUE. Not available for Interface = 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. All other controls=default.

E-725, E-727: 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-754: Input and output HW handshake must be TRUE. Not available for Interface = TCP/IP or DLL. 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. The baudrate is set via the DIP switches on the controller front panel. See the E-861 User manual for details.

E-871: Input and output HW handshake must be FALSE. All other controls=default. The baudrate is set via the DIP switches on the controller front panel. See the controller User manual for details.

E-873: 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. The baudrate is set via the DIP switches on the controller front panel. See the controller User manual for details.

---

### 2.2.3. GCSTranslator DLL Functions.vi

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.

Valid for C-413, C-843, C-843.PM, C-844, C-865, C-866, C-867, C-877, C-884, C-885, C-891, E-517, E-518, E-709, E-710, E-712, E-725, E-727, E-754, E-755, E-761, E-816, E-861, E-870, E-871, E-873, Hydra, Pollux, Mercury, Mercury_GCS (but must be present in Communication.llb for all other systems also)
2.2.4. **Global DaisyChain.vi**

Global DaisyChain is a global variable which contains setup information for DaisyChain systems.
Valid for all systems.

2.2.5. **Global1.vi**

A global variable which contains communication setup information.
Valid for all systems.

2.2.6. **Initialize Global DaisyChain.vi**

This VI initializes Global DaisyChain according to the given system no.
Valid for C-867, C-885, E-709, E-755, E-861, E-871, Mercury_GCS (but must be present for all other systems except Analog systems, too)
E-709: Only supported if E-709 is used inside C-867K012/K013.

2.2.7. **Initialize Global1.vi**

This VI initializes Global1 according to the given system no.
Valid for all systems.

2.2.8. **Is DaisyChain open.vi**

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.
Valid for: C-867, C-885, E-709, E-755, E-861, E-871, Mercury_GCS (but must be present for all other systems except Analog systems, too)
E-709: Only supported if E-709 is used inside C-867K012/K013.

2.2.9. **PI Ask for Communication Parameters.vi**

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.

Valid for all except Analog systems.

2.2.10. **PI Open Interface of one system.vi**

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?.

Valid for all except Analog systems.

- **C-413**: Interface = RS232, DLL, or TCP/IP (TCP/IP support available depending on controller version! If unsure whether your controller supports TCP/IP communication, please refer to the controller user manual). 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, TCP/IP 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-877: 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. 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 TRUE. 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-885: Interface = TCP/IP or 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.

C-891: Interface = RS232, TCP/IP or DLL. TCP/IP support available depending on controller version! If unsure whether your controller supports TCP/IP communication, please refer to your user manual. RS232: Input and output HW handshake must be FALSE. DLL: DLL for Device = C-891, 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-518: Interface = RS232, TCP/IP or DLL, RS232: Input and output HW handshake must be TRUE, DLL (USB): DLL for Device = PI_GCS2_DLL, DLL Interface = SB, 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-727: Interface = RS232, TCP/IP 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.

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

E-754: Interface = RS232, TCP/IP or DLL, RS232: Input and output HW handshake must be TRUE. DLL: DLL for Device = E-754, DLL Interface = USB, Parameter = Serial no. of system to connect to. 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.

E-873: 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.

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.

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.

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.

---

### 2.2.11. PI Open Interface.vi

Establishes communication with the connected systems. The interface and error statuses of all connected systems are cleared by this VI, which sends XXX (no command), *IDN? and ERR?.

Valid for all systems except analog systems.

See PI Open Interface of one system.vi for control settings.
2.2.12. **PI Receive String.vi**

Read string from selected system.
Valid for all systems.

```
inputs:
  System no.
  Strip spaces?
  bytes read

Outputs:
  String read
  bytes read
  error in
  error out
```

2.2.13. **PI Send String.vi**

Sends command with or without trailing termination character to selected system.
Valid for all systems.

```
inputs:
  System no.
  String to send
  Attach termination char.

Outputs:
  error in
  error out
```

2.2.14. **PI VISA Receive Characters.vi**

This vi reads n bytes (characters) via the chosen VISA interface. Sub-vi for "PI Receive String.vi".
Valid for C-413, C-702, C-848, C-867, C-877, C-880, C-880K005, C-884, C-885, C-887, C-891, E-516, E-517, E-518, E-712, E-725, E-727, E-753, E-754, E-709, E-816, E-861, E-871, E-873, F-206, M-8X0, Mercury_GCS (but must be present in Communication.lib for all other systems, too)

```
inputs:
  System no.
  Bytes to read

Outputs:
  String read
  Bytes read
  error in
  error out
```

2.2.15. **Select DaisyChain device.vi**

Performs a broadcast, returns names of all controllers matching "Controller name" and lets the user select the appropriate controller from a ring control. VI will also stop if Cancel is TRUE.
Valid for C-867, C-885, E-709, E-755, E-861, E-871, Mercury_GCS
C-867: Controller name = "C-867", DLL Interface = RS232_DC
C-885: Controller name = "C-885", DLL Interface = USB_DC or TCPIP_DC
E-709: Controller name = "E-709", DLL Interface = RS232_DC. Only supported if E-709 is used inside C-867K012/K013.
E-755: Controller name = "E-755", DLL Interface = RS232_DC
E-861: Controller name = "E-861", DLL Interface = RS232_DC or USB_DC
E-871: Controller name = "E-871", DLL Interface = RS232_DC or USB_DC
Mercury_GCS: Controller name = "Mercury_GCS", DLL Interface = RS232_DC or USB_DC

2.2.16. Select USB device.vi

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

Valid for C-413, C-866, C-867, C-877, C-884, C-885, C-891, E-517, E-518, E-709, E-712, E-725, E-727, E-754, E-816, E-861, E-870, E-871, E-873, Mercury_GCS

C-413: Controller name = "C-413", DLL Interface = USB
C-866: Controller name = "C-866", DLL Interface = USB
C-867: Controller name = "C-867", DLL Interface = USB
C-877: Controller name = "C-877", DLL Interface = USB
C-884: Controller name = "C-884", DLL Interface = USB
C-885: Controller name = "C-885", DLL Interface = USB
C-891: Controller name = "C-891", DLL Interface = USB
E-517: Controller name = "E-517", DLL Interface = USB
E-518: Controller name = E-518, DLL Interface = USB
E-709: Controller name = E-709, DLL Interface = USB
E-712: Controller name = E-712, DLL Interface = USB
E-725: Controller name = E-725, DLL Interface = USB
E-727: Controller name = E-727, DLL Interface = USB
E-754: Controller name = E-754, DLL Interface = USB
E-816: Controller name = "E-816", DLL Interface = USB
E-861: Controller name = "E-861", DLL Interface = USB
E-870: Controller name = "E-870", DLL Interface = USB
E-871: Controller name = "E-871", DLL Interface = USB
E-873: Controller name = "E-873", DLL Interface = USB
2.2.17. Set logging mode.vi

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.

Valid for all VIs.
2.3. File handling.llb

2.3.1. ArrayFile.vi

This VI checks the names of all arrays in a data file or deletes a given array from a data file.

Valid for: Analog systems, C-413, C-867, C-877, C-880, C-884, C-887, E-517, E-518, E-712, E-725, E-727, E-761, E-861, E-871, E-873, F-206, M-8X0, Mercury_GCS (but must be present for C-885 also). To support analog interfacing, VI must be present for E-709 and E-816 also.

2.3.2. File handler.vi

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.

Valid for: All systems.

2.3.3. GetDataFormat.vi

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.

Valid for: Analog systems, Analog systems, C-413, C-702, C843, C-866, C-867, C-877, C-880, C-884, C-887, C-891, E-517, E-518, E-709, E-710, E-712, E-725, E-727, E-753, E-754, E-755, E-761, E-861, E-871, E-873, F-206, M-8X0, Mercury_GCS (but must be present for C-885 also). To support analog interfacing, VI must be present for E-816 also.
2.3.4. **MatrixIO.vi**

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.

Valid for: Analog systems, C-413, C-702, C-843, C-866, C-867, C-877, C-880, C-884, C-887, C-891, E-517, E-518, E-709, E-710, E-712, E-725, E-727, E-753, E-754, E-755, E-761, E-861, E-871, E-873, F-206, M-8X0, Mercury_GCS(but must be present for C-885 also). To support analog interfacing, VI must be present for E-816 also.

---

2.3.5. **TableIO.vi**

This VI reads or writes data files in table format. Sub-VI for "DDL.vi", "DRR?.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"

Valid for: Analog systems, C-413, C-702, C-843, C-866, C-867, C-877, C-880, C-884, C-887, C-891, E-517, E-518, E-709, E-710, E-712, E-725, E-727, E-753, E-754, E-755, E-761, E-861, E-871, E-873, F-206, M-8X0, Mercury, Mercury_GCS(but must be present for C-885 also). To support analog interfacing, VI must be present for E-816 also.
2.4. General command.lib

2.4.1. *IDN?.vi

Returns system identification string.
Valid for all systems
E-816: This command cannot be issued to a slave.

2.4.2. CSV?.vi

Returns current GCS syntax version.
Valid for C-413, C-702, C-867, C-877, C-884, C-885, C-887, C-891, E-517, E-518,
E-709, E-712, E-725, E-727, E-753, E-754, E-761, E-861, E-870, E-871, E-873, F 206, Hydra, Pollux, M-8X0, Mercury_GCS
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 (check with CSV?.vi. If CSV?.vi
is not supported, syntax version is GCS 1.0).

2.4.3. Define connected axes.vi

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.
Valid for all systems.
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,
All other systems: Read from controller = TRUE, Invert order = FALSE
2.4.4. Define connected systems (Array).vi

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 first field of "Controller names" is valid and only the entry for the given system number is overwritten in Global2 (Array).vi.

Valid for all systems.

Analog system: Only supported when called by Analog_Configuration_Setup.vi

---

2.4.5. ERR?.vi

Returns error information. Controller error is TRUE if selected system reports an error code which is not 0. See appendix A of the GCS LabVIEW manual for a list of PI error codes and use "GCSTranslateError.vi" to translate error codes into error descriptions programmatically.

Valid for all systems.

Analog system: VI does not report any error.

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

---

2.4.6. Global2 (Array).vi

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

Valid for all systems.

---

2.4.7. HLP?.vi

Returns help string.

Valid for Analog systems, C-413, C-702, C-843, C-843.PM, C-848, C-865, C-866, C-867, C-877, C-880, C-884, C-885, C-887, C-891, E-516, E-517, E-518, E-709, E-712, E-725, E-727, E-710, E-753, E-754, E-755, E-761, E-861, E-870, E-871, E-873, F-206, Hydra, Pollux, M-8X0, Mercury, Mercury_GCS (but must be present for all other systems also).

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.
2.4.8. **HLT.vi**

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

Valid for C-413, C-702, C-843, C-843.PM, C-844, C-848, C-865, C-866, C-867, C-877, C-880, C-884, C-885, C-887, C-891, E-517, E-518, E-710, E-727, E-754, E-755, E-761, E-861, E-871, E-873, F-206, Hydra, Pollux, M-8X0, Mercury, Mercury_GCS

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-877: 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
C-885: 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.

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).

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
2.4.9. **HPA?.vi**

Returns a help string containing information about valid parameter IDs.

Valid for C-413, C-843, C-867, C-877, C-884, C-887, C-891, E-517, E-518, E-709, E-710, E-712, E-725, E-727, E-753, E-754, E-755, E-761, E-861, E-870, E-871, E-873, F-206, Hydra, Pollux, Mercury, Mercury_GCS, M-8X0 (but must be present for C-885 also)

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).

```
<table>
<thead>
<tr>
<th>System no.</th>
<th>Parameter help string</th>
</tr>
</thead>
<tbody>
<tr>
<td>error in</td>
<td>HPA?</td>
</tr>
<tr>
<td>error out</td>
<td></td>
</tr>
</tbody>
</table>
```

2.4.10. **Initialize Global2.vi**

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

Valid for all systems.

```
<table>
<thead>
<tr>
<th>System no.</th>
<th>PI</th>
</tr>
</thead>
<tbody>
<tr>
<td>error in</td>
<td>Global2</td>
</tr>
<tr>
<td>error out</td>
<td></td>
</tr>
</tbody>
</table>
```

2.4.11. **MAN?.vi**

Returns online manual entry for given command.

Valid for C-867, C-877, C-884, C-885, C-891, E-871, E-873, Mercury_GCS C-867, Mercury_GCS: Check HLP? answer to find out if MAN? is supported.

```
<table>
<thead>
<tr>
<th>System no.</th>
<th>PI</th>
</tr>
</thead>
<tbody>
<tr>
<td>Command</td>
<td>MAN?</td>
</tr>
<tr>
<td>error in</td>
<td>Online Manual</td>
</tr>
<tr>
<td>error out</td>
<td></td>
</tr>
</tbody>
</table>
```

2.4.12. **MOV.vi**

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.

Valid for Analog systems, C-413, C-702, C-843, C-843.PM, C-844, C-848, C-865, C-866, C-867, C-877, C-880, C-880K005, C-884, C-885, C-887, C-891, E-516, E-517, E-518, E-709, E-710, E-712, E-725, E-727, E-753, E-754, E-755, E-761, E-816, E-861, E-871, E-873, F-206, Hydra, Pollux, M 8X0, Mercury, Mercury_GCS 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-877: This command works only in closed-loop operation.

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.

C-885: This command works only in closed-loop operation.

C-891: This command works only in closed-loop operation. Motion commands are not allowed when the wave generator is active.

E-517, E-518: Motion commands like MOV are not allowed when the controller is in OFFLINE mode or when the wave generator output is active. When a macro is running on the controller, MOV will be executed not until the macro is finished or stopped. See "Control Value Generation" and "Control Modes" in the controller 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, E-727: Motion commands are not allowed when a wave generator is active or the analog input is used for target generation.

E-753, E-754: Motion commands are not allowed when the wave generator is active or the analog input is used for target generation.


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, E-873: 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.
2.4.13. MOV?.vi

Returns commanded target position.
Valid for C-413, C-702, C-843, C-843.PM, C-844, C-848, C-865, C-866, C-867, C-877, C-880, C-884, C-885, C-887, C-891, E-516, E-517, E-518, E-709, E-710, E-712, E-725, E-727, E-753, E-754, E-755, E-761, E-816, E-861, E-871, E-873, F-206, Hydra, Pollux, M-8X0, Mercury, Mercury_GCS
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), cCommand 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), cCommand has different implementation, please use MOV?_old.vi
All other systems: If All axes? = TRUE, then Axis identifier? can be FALSE

2.4.14. MVR.vi

Moves specified axes relative to current positions. No. of digits is the number of digits after the decimal point in the position value(s) that will be sent.
Valid for Analog systems, C-413, C-702, C-843, C-843.PM, C-844, C-848, C-865, C-866, C-867, C-877, C-880, C-880K005, C-884, C-887, C-891, E-516, E-517, E-518, E-709, E-710, E-712, E-725, E-727, E-753, E-754, E-755, E-761, E-816, E-861, E-871, E-873, F-206, Hydra, Pollux, M-8X0, Mercury, Mercury_GCS(but must be present for C-885 also)
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 MVR are not allowed when a joystick is active on the axis.
C-877: This command works only in closed-loop operation.
C-880K005: VI only supported when called through PI_Multi.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.

C-891: This command works only in closed-loop operation. Motion commands are not allowed when the wave generator is active.

E-517, E-518: Motion commands like MVR are not allowed when the controller is in OFFLINE mode or when the wave generator output is active. When a macro is running on the controller, MVR will be executed not until the macro is finished or stopped. See "Control Value Generation" and "Control Modes" in the controller 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.lib.

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

E-725, E-727: Motion commands are not allowed when a wave generator is active or the analog input is used for target generation.

E-753, E-754: Motion commands are not allowed when the wave generator is active or the analog input is used for target generation.


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, E-873: 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.

---

**2.4.15. MWG.vi**

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".

Valid for C-887, F-206, M-8X0 (but must be present for C-866, C-867, C-880, C-884, C-885, E-761, E-861, E-871, E-873 and Mercury_GCS also)
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, must be present anyway. For compatibility reasons, VI calls MOV.vi if GCS syntax version is > GCS 1.0.

2.4.16. ONT?.vi

Indicates whether or not queried axis is at target position.

Valid for C-413, C-702, C-843, C-843.PM, C-848, C-865, C-866, C-867, C-877, C-880, C-884, C-885, C-887, C-891, E-516, E-517, E-518, E-709, E-710, E-712, E-725, E-727, E-753, E-754, E-755, E-756, E-816, E-861, E-871, E-873, F-206, Hydra, Pollux, M-8X0, Mercury, Mercury_GCS (but must be present for all other systems also)

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.

2.4.17. POS?.vi

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

Valid for Analog systems, C-413, C-702, C-843, C-843.PM, C-844, C-848, C-865, C-866, C-867, C-877, C-880, C-884, C-885, C-887, C-891, E-516, E-517, E-518, E-709, E-710, E-712, E-725, E-727, E-753, E-754, E-755, E-761, E-816, E-861, E-871, E-873, F-206, Hydra, Pollux, M-8X0, Mercury, Mercury_GCS

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.

C-887, F-206: If All axes? = TRUE, then Axis identifier? can be FALSE. 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.

C-887, M-8X0: If All axes? = TRUE, then Axis identifier? can be FALSE. 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.

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

2.4.18. SAI?.vi

Returns axis identifiers of all connected axes and writes them into Global2 (Array).vi. Required by "Define connected axes.vi". If SAI? ALL is TRUE, all physically available axes are returned, no matter if configured or not. 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.

Valid for Analog systems, C-413, C-702, C-843, C-843.PM, C-844, C-848, C-865, C-866, C-867, C-877, C-880, C-880K005, C-884, C-885, C-887, C-891, E-516, E-517, E-518, E-709, E-710, E-712, E-725, E-727, E-753, E-754, E-755, E-761, E-816, E-861, E-871, E-873, F-206, Hydra, Pollux, M-8X0, Mercury, Mercury_GCS (but must be present in General command.llb for all other systems also)

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

2.4.19. SPA.vi

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 no. 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 an error code which is not 0.

Valid for C-413, C-702, C-843, C-843.PM, C-844, C-848, C-865, C-866, C-867, C-877, C-880, C-880K005, C-884, C-887, C-891, E-516, E-517, E-518, E-709, E-710, E-712, E-725, E-727, E-753, E-754, E-755, E-761, E-816, E-861, E-871, E-873, F-206, Hydra, Pollux, M-8X0, Mercury, Mercury_GCS (but must be present for C-885 also)

C-413, C-867, C-877, C-884, E-516, E-517, E-518, C-891, E-710, E-712, E-725, E-727, E-753, E-754, E-861, E-870, E-871, E-873, Hydra, Pollux, Mercury_GCS: The SPAcommand 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-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).

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).

C-843.PM: Parameter no. format is FALSE (decimal). See C-843 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, C-877: Parameter no. format is TRUE (hex.). Use "HPA?.vi" to get valid parameter numbers or see the ControllerUser 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). 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)
- 10: maximum velocity (0 to 1.79769313486231E308)
- 11: maximum allowed acceleration (0 to 1.79769313486231E308)
- 13: maximum allowed Jerk (-1.79E308 to 1.79E308)

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

C-891: Parameter no. format is TRUE (hex.). Use "HPA?.vi" to get valid parameter numbers or see the controller User Manual. See E-710 for warnings. Do not set more than 1 parameter at once.

E-516: Parameter no. format is FALSE (decimal). 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, E-518: Parameter no. format is TRUE (hex.). Use HPA?.vi to get valid parameter numbers or see the controller 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-754: 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.

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.

E-816: Parameter no. format is FALSE (decimal). See E-516 for a description of parameter numbers. This command cannot be issued to a slave. Each command limited to setting one parameter for only one axis.

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

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, E-873: 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. 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).

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

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.

### 2.4.20. SPA?.vi

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 no. 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 parameters 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.

Valid for C-413, C-702, C-843, C-843.PM, C-844, C-848, C-865, C-866, C-867, C-877, C-880, C-880K005, C-884, C-887, C-891, E-516, E-517, E-518, E-709, E-710, E-712, E-725, E-727, E-753, E-754, E-755, E-761, E-816, E-861, E-870, E-871, E-873, F-206, Hydra, Pollux, M-8X0, Mercury, Mercury_GCS (but must be present for C-885 also)

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). The following parameter number outputs a string:

60: stage name (maximum 14 characters)

C-843.PM: Parameter no. format is FALSE (decimal). The following parameter number outputs a string:

60: stage name (maximum 14 characters)

C-865: Parameter no. format is FALSE (decimal). The following parameter number outputs a string: 60: stage name (maximum 14 characters). Parameter number 25 is read-only.

C-866: Parameter no. format is FALSE (decimal). The following parameter number outputs a string: 60: stage name (maximum 14 characters). Parameter number 25 is read-only.

C-867, C-877: 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.

C-891: Parameter no. format is TRUE (hex.). Use "HPA?.vi" to get valid parameter numbers. Do not query more than 1 parameter no. at once (except with Without axes? = TRUE).

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

E-517, E-518: 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: Use HPA?.vi to get valid parameter numbers and see GCS DLL manual for a description of valid parameter numbers. Parameter no. format is TRUE (Hex).

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, E-727: 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-754: 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: Use HPA?.vi to get valid parameter numbers and see GCS DLL manual for a description of valid parameter numbers. Parameter no. format is TRUE (Hex). Do not query more than 10 parameter no. at once (except with Without axes? = TRUE).

E-816: Parameter no. format is FALSE (decimal). Only one parameter value for only one axis per command allowed. This command cannot be issued to a slave.

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. The following parameter number outputs a string:

60 (0x3C): stage name (maximum 16 characters)

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, E-873: 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-887, 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: Parameter no. format is FALSE (decimal). The following parameter number outputs a string: 60: stage name (maximum 14 characters)

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.

2.4.21. STP.vi

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.

Valid for Analog systems, C-413, C-702, C-843, C-843.PM, C-844, C-848, C-865, C-866, C-867, C-877, C-880, C-884, C-885, C-887, C-891, E-516, E-517, E-518, E-709, E-712, E-725, E-727, E-753, E-754, E755, E-861, E-870, E-871, E-873, 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.

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

All other systems: All axes? = TRUE, Axis identifier? = FALSE

E-517, E-518: 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-727, E-753, E-754: 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.
2.4.22. SVO.vi

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.

Valid for C-413, C-702, C-843, C-843.PM, C-844, C-848, C-865, C-866, C-867, C-877, C-880, C-884, C-885, C-887, C-891, E-516, E-517, E-518, E-709, E-710, E-712, E-725, E-727, E-753, E-754, E-755, E-761, E-816, E-861, E-871, E-873, F-206, Hydra, Pollux, M-8X0, Mercury, Mercury_GCS

E-516, E-517, E-518: 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-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. Check HLP?/HELP answer to find out if SVO is supported.

All other systems: Without axis ID = FALSE

2.4.23. SVO?.vi

Returns servo status of queried axes.

Valid for C-413, C-702, C-843, C-843.PM, C-844, C-848, C-865, C-866, C-867, C-877, C-880, C-884, C-885, C-887, C-891, E-516, E-517, E-518, E-709, E-710, E-712, E-725, E-727, E-753, E-754, E-755, E-761, E-816, E-861, E-871, E-873, F-206, Hydra, Pollux, M-8X0, Mercury, Mercury_GCS

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: 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. All axes? = TRUE, Axis identifier? = FALSE. Check HLP?/HELP answer to find out if SVO? is supported.

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

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 an error code which is not 0.

Valid for Analog systems, C-413, C-702, C-843, C-843.PM, C-844, C-848, C-865, C-866, C-867, C-877, C-880, C-880K005, C-884, C-885, C-887, C-891, E-516, E-517, E-518, E-709, E-710, E-712, E-725, E-727, E-753, E-754, E-755, E-761, E-861, E-873, F-206, Hydra, Pollux, M-8X0, Mercury, Mercury_GCS. To support analog interfacing, VI must be present for E-816 also.

Analog: Without axis ID? = FALSE; Velocity unit is \( \mu \text{m/sec} \)

C-413: Without axis ID? = FALSE. Velocity unit is mm/s. 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-877, C-884, C-891, Hydra, Pollux, Mercury_GCS: Without axis ID? = FALSE. Velocity unit is mm/s. 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.

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

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

C-885: Without axis ID? = FALSE. The VEL command saves the parameters in RAM only. For how to save the currently valid parameters to flash ROM, where they become the power-on defaults, please refer to the controller’s user manual.

E-516: Without axis ID? = FALSE. Velocity unit is \( \mu \text{m/s} \). 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-709, E-712, E-725, E-727, E-753, E-754, E-861: Without axis ID? = FALSE. Velocity unit is \( \mu \text{m/s} \) in closed-loop operation and V/s in open-loop operation. 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-710: Without axis ID? = FALSE. Velocity unit is \( \mu \text{m/ms} \).

E-755: Without axis ID? = FALSE. Velocity unit is \( \mu \text{m/s} \). Command not available for E-755.101.

E-761: Without axis ID? = FALSE. Velocity unit is \( \mu \text{m/ms} \). 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 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.

E-861: Without axis ID? = FALSE. 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.

E-873: Without axis ID? = FALSE. Check HLP? answer to find out if VEL is supported. 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.

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

2.4.25. VEL?.vi

Returns velocity setting for specified axes.

Valid for Analog systems, C-413, C-702, C-843, C-843.PM, C-844, C-848, C-865, C-866, C-867, C-877, C-880, C-880K005, C-884, C-885, C-887, C-891, E-516, E-517, E-518, E-709, E-710, E-712, E-725, E-727, E-753, E-754, E-755, E-761, E-861, E-873, F-206, Hydra, Pollux, M-8X0, Mercury, Mercury_GCS. To support analog interfacing, VI must be present for E-816 also.

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

C-702, C-848, Mercury™: If All axes? = TRUE, then Axis identifier? can be FALSE
C-843, C-843.PM, C-844, E-516: If All axes? = TRUE, then Axis identifier? must be TRUE
C-413, C-865, C-866, C-867, C-877, C-884, C-891, Hydra, Pollux, Mercury_GCS: If All axes? = TRUE, then Axis identifier? can be FALSE. Velocity unit is mm/s.
C-880: If All axes? = TRUE, then Axis identifier? can be FALSE. NanoCube axes will report velocity = 0
C-880K005: VI only supported when called through PI_Multix.vi
C-885: If All axes? = TRUE, then Axis identifier? can be FALSE.
E-517, E-518: 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-710: If All axes? = TRUE, then Axis identifier? must be TRUE. Velocity unit is µm/ms.
E-709, E-712, E-725, E-727, E-753, E-754, 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), only valid for velocity of axes A and/or B. All axes? must be FALSE. For platform velocity, use 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. For platform velocity: only first field of velocity array is valid

E-873: If All axes? = TRUE, then Axis identifier? can be FALSE. Check HLP? answer to find out if VEL? is supported.

2.4.26. VER?.vi

Returns firmware string.

Valid for C-702, C-848, C-866, C-867, C-877, C-880, C-884, C-885, C-887, C-891, E-516, E-517, E-518, E-727, E-754, E-755, E-761, E-871, E-873, F-206, Hydra, Pollux, M-8X0, Mercury, Mercury_GCS

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

2.4.27. VMO.vi

Virtual movement. Indicates whether a move to the specified position is possible or not. Stage will not be moved. is the number of digits after the decimal point in the position value(s) that will be sent.

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, C-885, E-761, E-861, E-871, E-873 and Mercury_GCS, also)

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 command VMO. For GCS 2.0 and higher, VI sends command VMO? instead.
2.5. Joystick.lib

2.5.1. Calculate joystick scaling.vi

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

Valid for C-413, C-702, C-843, C-843.PM, C-844, C-848, C-865, C-866, C-867, C-877, C-880, C-884, C-891, E-517, E-518, E-709, E-710, E-712, E-725, E-727, E-753, E-754, E-755, E-761, E-861, Hydra, Pollux, Mercury_GCS (but must be present for C-885 also)

2.5.2. JAS?.vi

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

Valid for C-867, E-709, E-861, E-870, Mercury, Mercury_GCS

C-867, E-709, E-861, E-870, Mercury, Mercury_GCS: The current amplitude gives a factor which is applied to the velocity set with VEL (closed-loop operation) or OVL (E-861 open-loop operation, E-870), the range is -1.0 to 1.0. Only one Joystick ID to query for one Joystick axis per command allowed. Use Without JoystickIDs? = TRUE to query all Joystick IDs.

E-709: Only supported if E-709 is used inside C-867K012/K013.

2.5.3. JAX.vi

Defines which axes to control with an axis of a joystick connected to the controller. If Axes to set is an empty array, the current settings are cleared and no axes are controlled. Joystick control is enabled or disabled with "JON.vi". Settings will not influence "Joystick_Operation_Sample_Program.vi" which works with a joystick connected to the host computer.

Valid for C-843, C-867, E-709, E-861, E-870, Mercury_GCS

E-709: Only supported if E-709 is used inside C-867K012/K013.
2.5.4. JAX?.vi

Returns which axes are assigned to be controlled by joystick axes of joysticks connected to the controller.

Valid for C-843, C-867, E-709, E-861, E-870, Mercury, Mercury_GCS

E-709: Only supported if E-709 is used inside C-867K012/K013.

2.5.5. JBS?.vi

Returns state (TRUE/FALSE) of queried buttons of joysticks connected to the controller.

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

2.5.6. JDT.vi

Sets default lookup table for joysticks connected to the controller, waits 100 ms and queries ERR?.

Controller error is TRUE if selected system reports error code 0. Settings will not influence "Joystick_Operation_Sample_Program.vi" which works with a joystick connected to the host computer.

Valid for C-867, E-709, E-861, E-870, Mercury, Mercury_GCS

C-867, E-709, E-861, Mercury_GCS: The following default lookup tables are provided:
1 = linear
2 = parabolic
E-709: Only supported if E-709 is used inside C-867K012/K013.

E-870: The following default lookup tables are provided:
1 = linear
2 = parabolic
5 = linear invert
6 = parabolic invert
Mercury: The following default lookup tables are provided:

1 = linear
3 = cubic

2.5.7. JLT.vi

Sets joystick lookup table for a joystick connected to the controller. The amplitudes
of the joystick axes are mapped to velocities of the controller axes. StartPoint is
the start point in the lookup table, starting with 1. Valid values for Factor are -1.0 to
1.0 and define the factor which is applied to the currently set velocity (VEL in
closed-loop operation, OVL in open-loop operation). No. of digits is the number of
digits after the decimal point in the Factor value(s) that will be sent. Controller error
is TRUE if selected system reports error code >= 0. Settings will not influence
"Joystick_Operation_Sample_Program.vi" which works with a joystick connected
to the host computer.

Valid for C-867, E-709, E-861, E-870, Mercury, Mercury_GCS

E-709: Only supported if E-709 is used inside C-867K012/K013. Only one lookup
table point for one axis per command allowed. To set more than one point you
must create a for-loop around this VI. The joystick table consists of 256 points. The
first point corresponds to the maximum joystick axis displacement in negative
direction, the 256th point to the maximum displacement in positive direction.

C-867, E-861, E-870, Mercury_GCS: Only one lookup table point for one joystick
axis per command allowed. To set more than one point you must create a for-loop
around this VI. The joystick table consists of 256 points. The first point

corresponds to the maximum joystick axis displacement in negative direction, the
256th point to the maximum displacement in positive direction.

2.5.8. JLT?.vi

Returns joystick lookup table for joysticks connected to the controller. For large N
values, communication timeout must be set long enough, otherwise a comm.error
may occur.

Valid for C-867, E-709, E-861, E-870, Mercury, Mercury_GCS

C-867, E-709, E-861, E-870, Mercury_GCS: JoystickIDs and Joystick axes must
be omitted in the JLT? command, while X0 and N are always required.

E-709: Only supported if E-709 is used inside C-867K012/K013.
2.5.9. **JON.vi**

Enables or disables joysticks connected to the controller. Controller error is TRUE if selected system reports an error code which is not 0. Settings will not influence "Joystick_Operation_Sample_Program.vi" which works with a joystick connected to the host computer.

Valid for C-843, C-867, E-709, E-861, Mercury, Mercury_GCS

C-843, C-867, E-861, Mercury, Mercury_GCS: Joystick mode can be 0 (OFF) or 1 (ON). Motion commands like MOV or OSM (if supported) are not allowed when a joystick is active on the axis.

E-709: Only supported if E-709 is used inside C-867K012/K013. Motion commands like MOV are not allowed when a joystick is active on the axis. Joystick mode can be 0 (OFF) or 1 (ON).

```
System no.    JoystickID to set
Joystick mode  Controller error
```

2.5.10. **JON?.vi**

Returns activation status of queried joysticks connected to the controller.

Valid for C-843, C-867, E-709, E-861, Mercury, Mercury_GCS

E-709: Only supported if E-709 is used inside C-867K012/K013.

```
System no.  JoystickID to query
All joysticks? Joystick status
```

2.5.11. **Read joystick.vi**

Sub-VI for operation with a joystick connected to the game port of the host computer. Reads joystick position and button status for a standard 2-button 2-axis joystick.

Install joystick driver and calibrate joystick in the Windows control panel before running this VI.

Valid for C-413, C-702, C-843, C-843_PM, C-844, C-848, C-865, C-866, C-867, C-877, C-880, C-884, C-891, E-517, E-518, E-709, E-710, E-712, E-725, E-727, E-753, E-754, E-755, E-761, E-861, Hydra, Pollux, Mercury_GCS (but must be present for C-885 also)
2.5.12. Scale joystick data.vi

Sub-VI for operation with a joystick connected to the game port of the host computer. Scales joystick position. Use output value from "Calculate joystick scaling.vi" for Factor *. Dead band * is the maximum scaled position value that does not result in any motion.

Valid for C-413, C-702, C-843, C-843,PM, C-844, C-848, C-865, C-866, C-867, C-877, C-880, C-884, C-891, E-517, E-518, E-709, E-710, E-712, E-725, E-727, E-753, E-754, E-755, E-761, E-861, Hydra, Pollux, Mercury_GCS (but must be present for C-885 also)
2.6. Limits.lib

2.6.1. DFH.vi

Defines current position of the "DFH axes" as home position. Their position value is set to 0. Due to the change of the home position the values for the travel range limits (TMN?, TMX?) are also changed accordingly. The home position is reset to default by "ATZ.vi", "REF.vi", "MNL.vi", "MPL.vi" or similar command VIs.

Valid for C-702, C-843, C-843.PM, C-844, C-848, C-865, C-866, C-867, C-877, C-880, C-880K005, C-884, E-517, E-518, E-710, E-755, E-761, E-871, E-873, Hydra, Pollux, Mercury, Mercury_GCS (but must be present for C-885 also)

C-867, Mercury_GCS: If All axes? = TRUE, then Axis identifier? must be TRUE. Check HLP? answer to find out if DFH is supported.

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

E-517: If All axes? = TRUE, then Axis identifier? must be TRUE. The values are saved with WPA 100. The position shown in the E-517 display is not affected by DFH because it is that of the sensor channel (can also be queried with TSP?). DFH affects only the axis position which can be queried with POS?.

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

E-518, E-761: If All axes? = TRUE, then Axis identifier? must be TRUE. The values are saved with WPA 100.

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

```
<table>
<thead>
<tr>
<th>System no.</th>
<th>DFH axes</th>
<th>All axes?</th>
<th>Axis identifier?</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>P1</td>
<td></td>
<td>error in</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>error out</td>
</tr>
</tbody>
</table>
```

2.6.2. DFH?.vi

Returns the difference between the current home position and the absolute or initial zero point (default home position) for each of the queried axes.

Valid for C-702, C-843, C-843.PM, C-844, C-848, C-865, C-866, C-867, C-877, C-880, C-880K005, C-884, E-517, E-518, E-710, E-755, E-761, E-871, E-873, Hydra, Pollux, Mercury, Mercury_GCS (but must be present for C-885 also)

C-702, C-848, C-877, C-880, C-884, E-517, E-518, E-871, E-873, Hydra, Pollux, Mercury™, Mercury_GCS: If All axes? = TRUE, then Axis identifier? can be FALSE

C-843, C-843.PM, C-844, C-865, C-866, E-710, E-761: If All axes? = TRUE, then Axis identifier? must be TRUE

C-867, Mercury_GCS: If All axes? = TRUE, then Axis identifier? can be FALSE. Check HLP? answer to find out if DFH? is supported.

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

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

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

" EdgeID 1: negative limit switch, Parameter value is 0 when the default setting should be used (e.g. from pistages.dat), 1 when active high, -1 when active low

" EdgeID 2: positive limit switch, Parameter value is 0 when the default setting should be used (e.g. from pistages.dat), 1 when active high, -1 when active low

" EdgeID 3: reference switch, Parameter value is 0 when the default setting should be used (e.g. from pistages.dat), 1 when active high, -1 when active low

" EdgeID 4: autofind AxisIn (is one input line of the motion chip carrying an external sensor signal which changes its state at a certain position), Parameter value gives the signal state to the left of the edge (high or low)

" EdgeID 5: find arbitrary edge (i.e. change of the state of the AxisIn signal), Parameter value gives the direction of motion: 1=positive, -1=negative

Valid for C-843, C-867, C-877, C-884, E-861, E-871, E-873, Hydra, Pollux, Mercury_GCS (but must be present for C-885 also)

C-843: EdgeID can be 1 to 3. Parameter value can be -1, 0 or 1, depending on EdgeID. The firmware detects the presence or absence of reference switch and limit switches using controller parameters (ID 0x14 for reference switch; ID 0x32 for limit switches). According to the values of those parameters, the controller enables or disables FED motions to the appropriate signal edges. Adapt the parameter values to your hardware using SPA.

C-867, C-877, C-884, E-861, E-871, E-873, Mercury_GCS: EdgeID can be 1 to 3. Parameter value must be 0. The firmware detects the presence or absence of reference switch and limit switches using controller parameters (ID 0x14 for reference switch; ID 0x32 for limit switches). See User Manual for details and specialties. According to the values of those parameters, the controller enables or disables FED motions to the appropriate signal edges. Adapt the parameter values to your hardware using SPA or SEP.

Hydra, Pollux: EdgeID can be 2 or 3. Parameter value must be 0. The firmware detects the presence or absence of reference switch and limit switches using controller parameters (ID 0x14 for reference switch; ID 0x32 for limit switches). See User Manual for details and specialties. According to the values of those parameters, the controller enables or disables FED motions to the appropriate signal edges. Adapt the parameter values to your hardware using SPA (or SEP, if supported).
2.6.4. FNL.vi

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

Valid for C-843, C-843.PM, C-867, C-884, E-755, E-861, E-871, E-873, Hydra, Pollux, Mercury_GCS (but must be present for C-885 also)

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

C-867: Check HLP? answer to find out if FNL is supported. If not, you can use FRF instead and set parameter no. 0x70 to select which sensor is to use for referencing. See controller User Manual for details.

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

C-867, C-877, C-884, E-871, E-873, Hydra, Pollux, Mercury_GCS: If All axes? = TRUE, then Axis identifier? must be TRUE. Servo must be enabled with SVO for the commanded axis prior to using this command (closed-loop operation).

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

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

E-861: If All axes? = TRUE, then Axis identifier? must be TRUE. The reference mode must be set to "1" (factory default) with the RON command if referencing is to be done by performing a reference move.

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

2.6.5. FPL.vi

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

Valid for C-843, C-843.PM, C-867, C-884, E-755, E-861, E-871, E-873, Hydra, Pollux, Mercury_GCS (but must be present for C-885 also)
C-843, C-843.PM: If All axes? = TRUE, then Axis identifier? must be TRUE
C-867: Check HLP? answer to find out if FPL is supported. If not, you can use FRF instead and set parameter no. 0x70 to select which sensor is to use for referencing. See controller User Manual for details.
E-755: If All axes? = TRUE, then Axis identifier? can be FALSE. Command not available for E-755.101
C-867,C-877,C-884, E-871, E-873, Hydra, Pollux, Mercury_GCS: If All axes? = TRUE, then Axis identifier? must be TRUE. Servo must be enabled with SVO for the commanded axis prior to using this command (closed-loop operation).
The reference mode must be set to "1" (factory default) with the RON command if referencing is to be done by performing a reference move.
The positive limit switch of the mechanics is used to determine the positive physical limit of the travel range. The sum of VALUE_AT_REF_POS (parameter ID 0x16) and DISTANCE_REF_TO_P_LIM (parameter ID 0x2F) is set as the current position when the axis is at the positive limit switch. If the soft limits (MAX_TRAVEL_RANGE_POS, parameter ID 0x15, and MAX_TRAVEL_RANGE_NEG, parameter ID 0x30) are used to reduce the travel range, the limit switches can not be used for reference moves. FNL and FPL commands will provoke an error message, and only the reference switch can be used for a reference move (FRF command). See User Manual for details and specifics.
E-861: If All axes? = TRUE, then Axis identifier? must be TRUE. The reference mode must be set to "1" (factory default) with the RON command if referencing is to be done by performing a reference move.
The positive limit switch of the mechanics is used to determine the positive physical limit of the travel range. The sum of VALUE_AT_REF_POS (parameter ID 0x16) and DISTANCE_REF_TO_P_LIM (parameter ID 0x2F) is set as the current position when the axis is at the positive limit switch. If the soft limits (MAX_TRAVEL_RANGE_POS, parameter ID 0x15, and MAX_TRAVEL_RANGE_NEG, parameter ID 0x30) are used to reduce the travel range, the limit switches can not be used for reference moves. FNL and FPL commands will provoke an error message, and only the reference switch can be used for a reference move (FRF command).

2.6.6. **FRF.vi**

This VI starts a fast referencing of the specified axes. Use #7 polling to determine end of this referencing procedure.
Valid for C-413, C-843, C-843.PM, C-867,C-877,C-880 (only K006/K007 version), C-880K005, C-884, C-885, C-887, C-891, E-712, E-754, E-861, E-871, E-873, 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)
C-843, C-843.PM, C-880: If All axes? = TRUE, then Axis identifier? must be TRUE
C-880K005: VI only supported when called through PI_Multix.vi
C-413, C-867, C-877, C-884, E-871, E-873, Hydra, Pollux, Mercury_GCS: If All axes? = TRUE, then Axis identifier? must be TRUE. 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.

C-885: If All axes? = TRUE, then Axis identifier? can be FALSE. 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.

C-891: If All axes? = TRUE, then Axis identifier? must be TRUE. The system mechanics Reference type can be set by changing the parameter ID 0x70. For details please refer to the Controller User Manual.

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

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

### 2.6.7. FRF?.vi

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.

Valid for C-413, C-843, C-843.PM, C-866, C-867, C-877, C-880 (only C-880K006/C-880K007 version), C-880K005, C-884, C-885, C-887, C-891, E-712, E-754, E-755, E-861, E-871, E-873, F-206, Hydra, Pollux, M-8X0, Mercury_GCS

C-413, C-843, C-843.PM, C-866, C-867, C-880, C-884, C-885, C-891, E-754, E-861, E-871, E-873, 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).

2.6.8. GOH.vi

Moves specified axes to their home positions.
Valid for C-702, C-843, C-843.PM, C-844, C-848, C-865, C-866, C-867, C-877, C-880, C-880K005, C-884, E-517, E-518, E-710, E-755, E-761, E-861, E-871, E-873, Hydra, Pollux, Mercury, Mercury_GCS (but must be present for C-885 also)
C-702, C-843, C-843.PM, C-844, C-848, C-865, C-866, C-867,C-877,C-880, E-710, E-861, E-871, E-873, Mercury_GCS: If All axes? = TRUE, then Axis identifier? must be TRUE
C-880K005: VI only supported when called through PI_Multix.vi
E-755: If All axes? = TRUE, then Axis identifier? can be FALSE. Command not available for E-755.101.
E-517, E-518, E-761, Mercury™: If All axes? = TRUE, then Axis identifier? can be FALSE

2.6.9. LIM?.vi

Indicates whether queried axes have limit switches or not.
Valid for C-413, C-702, C-843, C-843.PM, C-848, C-865, C-866, C-867, C-877, C-880, C-884, C-887, C-891, E-712, E-755, E-861, E-871, E-873, F-206, Hydra, Pollux, M-8X0, Mercury™, Mercury_GCS (but must be present for C-885 also)
All systems: If All axes? = TRUE, then Axis identifier? can be FALSE
E-712: Check HLP? answer to find out if LIM? is supported.
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).
2.6.10. RON.vi

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

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

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

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

WARNINGS:

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

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

Valid for C-413, C-702, C-843, C-843.PM, C-848, C-865, C-866, C-867, C-877, C-880, C-880K005, C-884, C-885, C-891, E-712, E-861, E-871, E-873, Hydra, Pollux, Mercury™, Mercury_GCS

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

E-712: Check HLP? answer to find out if RON is supported and E-712 User Manual to find out for which type of stages RON can be used.

All other systems: Without axis ID = FALSE

<table>
<thead>
<tr>
<th>Without axis ID</th>
<th>System no.</th>
<th>Axes to command</th>
<th>Reference mode</th>
<th>error out</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

2.6.11. RON?.vi

Indicates whether queried axes have reference mode ON or OFF. See "RON.vi" above for description of reference mode.
Valid for C-413, C-702, C-843, C-843.PM, C-844, C-848, C-865, C-866, C-867, C-877, C-880, C-880K005, C-884, C-885, C-887, C-891, E-712, E-861, E-871, E-873, F-206, Hydra, Pollux, M-8X0, Mercury™, Mercury_GCS

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

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

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

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).

2.6.12. TMN?.vi

Valid for Analog systems, C-413, C-702, C-843, C-843.PM, C-844, C-848, C-865, C-866, C-867, C-877, C-880, C-884, C-885, C-887, C-891, E-517, E-518, E-709, E-710, E-712, E-725, E-727, E-753, E-754, E-755, E-761, E-861, E-871, E-873, F-206, Hydra, Pollux, M-8X0, Mercury, Mercury_GCS. To support analog interfacing, VI must be present for E-816 also.

Analog, C-413, C-702, C-843, C-844, C-848, C-865, C-866, C-867, C-877, C-880, C-884, C-885, C-887, C-891, E-517, E-518, E-709, E-712, E-725, E-727, E-753, E-754, E-755, E-761, E-861, E-871, E-873, F-206, Hydra, Pollux, M-8X0, Mercury, Mercury_GCS: If All axes? = TRUE, then Axis identifier? must be TRUE

C-413, C-867, C-877, C-884, C-891, E-861, E-871, E-873, Hydra, Pollux, Mercury_GCS: If All axes? = TRUE, then Axis identifier? can be FALSE. The minimum commandable position is defined by the MAX_TRAVEL_RANGE_NEG parameter ID 0x30 (SPA).

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

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

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.
2.6.13. TMX?.vi

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

Valid for Analog systems, C-413, C-702, C-843, C-843.PM, C-844, C-848, C-865, C-866, C-867, C-877, C-880, C-884, C-885, C-887, C-891, E-517, E-518, E-709, E-710, E-712, E-725, E-727, E-753, E-754, E-755, E-761, E-861, E-871, E-873, F-206, Hydra, Pollux, M-8X0, Mercury, Mercury_GCS. To support analog interfacing, VI must be present for E-816 also.

Analog, C-413, C-702, C-848, C-865, C-866, C-867, C-877, C-880, C-884, C-885, C-887, C-891, E-517, E-518, E-709, E-712, E-725, E-727, E-753, E-754, E-761, E-861, E-871, E-873, 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

C-413, C-847, C-877, C-884, C-861, E-871, E-873, Hydra, Pollux, Mercury_GCS: If All axes? = TRUE, then Axis identifier? can be FALSE. The maximum commandable position is defined by the MAX_TRAVEL_RANGE_POS parameter ID 0x15 (SPA).

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

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

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.

2.6.14. TRS?.vi

Indicates whether or not given axes have a reference sensor with direction sensing.
Valid for C-413, C-867, C-877, C-884, C-887, C-891, E-712, E-754, E-755, E-861, E-871, E-873, F-206, Hydra, Pollux, M-8X0, Mercury_GCS (but must be present for C-885 also)

C-413, C-867, C-877, C-884, C-891, E-755, E-861, E-871, E-873, Hydra, Pollux, 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.

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). If All axes? = TRUE, then Axis identifier? can be FALSE.
2.7. Macros.llb

2.7.1. #8.vi

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

Valid for C-702, C-848, C-867, C-877, C-880, C-884, C-887, C-891, E-516, E-517, E-518, E-727, E-754, E-816, E-861, E-871, E-873, F-206, M-8X0, Mercury_GCS (but must be present for C-885 also)

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).

System no. | PI | Macro running?
---|---|---
error in | #8 | String read

2.7.2. Define macro contents.vi

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.

Valid for C-702, C-844, C-848, C-867, C-877, C-880, C-884, C-887, C-891, E-517, E-518, E-727, E-754, E-816, E-861, E-871, E-873, F-206, M-8X0, Mercury, Mercury_GCS (but must be present for C-885 also)

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.

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).

System no. | PI | Macro contents
---|---|---
error in | define macro | error out

2.7.3. MAC BEG.vi

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.

Valid for C-702, C-844, C-848, C-867, C-877, C-880, C-884, C-887, C-891, E-516, E-517, E-518, E-727, E-754, E-816, E-861, E-871, E-873, F-206, M-8X0, Mercury, Mercury_GCS (but must be present for C-885 also)

C-702, C-848, C-867, C-877, C-880, C-884, C-887, C-891, E-516, E-517, E-518, E-727, E-754, E-816, E-871, E-873, F-206, M-8X0, Mercury_GCS: Macro name must be between 1 and 8 characters
E-516: Macro name must be between 3 and 8 characters. 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 E-516 is powered off.

E-816: Macro name must be between 3 and 8 characters. 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, F-206: Macro name must be between 3 and 8 characters.

Mercury: For valid macro names see GCS DLL Manual.

C-887, M-8X0: Macro name must be between 3 and 8 characters. 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).

---

### 2.7.4. MAC DEF.vi

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

Valid for C-867, C-877, C-884, C-887, C-891, E-516, E-517, E-518, E-727, E-754, E-816, E-861, E-871, E-873, F-206, M-8X0, Mercury_GCS(but must be present for C-885 also)

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).

---

### 2.7.5. MAC DEF?.vi

Get name of autostart macro.

Valid for C-867, C-877, C-884, C-887, C-891, E-516, E-517, E-518, E-727, E-754, E-816, E-861, E-871, E-873, F-206, M-8X0, Mercury_GCS(but must be present for C-885 also)

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).
2.7.6. MAC DEL.vi

Delete macro. If "With dialog" is TRUE, a dialog box pops up to confirm the deletion. "Controller error" is TRUE if selected system reports an error code which is not zero.

Valid for C-702, C-844, C-848, C-867, C-877, C-880, C-884, C-887, C-891, E-516, E-517, E-518, E-727, E-754, E-816, E-861, E-871, E-873, F-206, M-8X0, Mercury, Mercury_GCS (but must be present for C-885 also)

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.

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).

2.7.7. MAC END.vi

Stops current macro recording.

Valid for C-702, C-844, C-848, C-867, C-877, C-880, C-884, C-887, C-891, E-516, E-517, E-518, E-727, E-754, E-816, E-861, E-871, E-873, F-206, M-8X0, Mercury, Mercury_GCS (but must be present for C-885 also)

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.

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).

2.7.8. MAC ERR?.vi

Returns error information from macro execution
Valid for C-867, C-877, C-884, C-887, C-891, E-727, E-871, E-873, F-206, Mercury_GCS, M-8X0 (but must be present for C-885 also)

C-867, C-887, F-206, M-8X0: Check HLP?/HELP answer to find out if MAC ERR? is supported.

System no. | Error information
--- | ---
error in | error out

### 2.7.9. MAC NSTART.vi

Start macro N times.

Valid for C-702, C-867, C-877, C-884, C-887, C-891, E-516, E-517, E-518, E-727, E-754, E-816, E-861, E-871, E-873, F-206, M-8X0, Mercury, Mercury_GCS (but must be present for C-885 also)

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.

### 2.7.10. MAC START.vi

Start macro.

Valid for C-702, C-844, C-848, C-867, C-877, C-880, C-884, C-887, C-891, E-516, E-517, E-518, E-727, E-754, E-816, E-861, E-871, E-873, F-206, M-8X0, Mercury, Mercury_GCS (but must be present for C-885 also)

C-702, C-848, C-867, C-877, C-880, C-891, E-516, E-517, E-518, E-727, E-754, E-816 E-871, E-873: Use #8.vi to determine when macro execution has finished.

For C-702, C-848, C-880, E-516and E-517, E-518 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.

C-887, 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.
2.7.11. MAC?.vi

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

Valid for C-702, C-844, C-848, C-867, C-877, C-880, C-884, C-887, C-891, E-516, E-517, E-518, E-727, E-754, E-754, E-816, E-861, E-871, E-873, F-206, M-8X0, Mercury, Mercury_GCS (but must be present for C-885 also)

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

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).

```
   System no.  Macro names or contents
   Macro name  MAC?
   Get contents?  error out
   error in
```

2.7.12. MAT.vi

Carries out a mathematical operation or bit operation and saves the result as a variable.

See controller User Manual for details regarding local and global variables.

Valid for C-867, Mercury_GCS

```
   System no.  Controller error
   Variable name  MAT
   Argument 1
   Argument 2
   Operation
   error in
   No. of digits
```

2.7.13. RMC?.vi

List macros currently running.

Valid for C-867, C-877, C-884, C-887, C-891, E-727, E-861, E-871, E-873, F-206, M-8X0, Mercury_GCS (but must be present for C-885 also)

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

```
   System no.  Running macros
   error in
```

2.7.14. VAR.vi

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.

Valid for: C-867, C-877, C-884, C-887, C-891, E-727, E-871, E-873, F-206, M-8X0, Mercury_GCS (but must be present for C-885 also)
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).

C-867, Mercury_GCS : Check HLP? answer to find out if VAR is supported.

### 2.7.15. VAR?.vi

Returns the value of given variable name(s). See controller User Manual for details regarding local and global variables.

Valid for C-867, C-877, C-884, C-887, C-891, E-727, E-871, E-873, F-206, M-8X0, Mercury_GCS (but must be present for C-885 also)

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).

C-867, Mercury_GCS : Check HLP? answer to find out if VAR? is supported.
2.8. Old commands.llb

2.8.1. #5_old.vi

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

Valid for F-206, M-8X0 (but must be present for all other systems, too)
M-8X0: Only answer for overall system is valid.

---

2.8.2. Wait for hexapod system axes to stop.vi

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 occurred, or if Stop refnum or Local stop is TRUE. When using as a sub-VI, use Stop refnum to stop VI from caller. Required by "General wait for movement to stop.vi".

Valid for C-887, F-206, M-8X0 (but must be present for all other systems, too)
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

---
2.9. Optical or Analog Input.lib

2.9.1. MOV and TAV?.vi

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. Requires "Define connected systems.vi". Valid for C-865, C-866, C-867, C-880, C-884, C-887, E-712, E-761, E-861, E-871, E-873, F-206, M-8X0, Mercury™, Mercury_GCS (but must be present for C-885 also)

E-712: Check HLP? answer to find out if TAV? is supported.
E-761: Board = 4. The output is the current voltage at the analog input line, with gain and offset.

2.9.2. MWG and TAV?.vi

Moves stage to absolute position (MWG.vi), waits for the specified axes to stop using (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" and "Wait for hexapod system axes to stop.vi" to be present. Required by "1D Scan.vi" and "2D Scan.vi".

Valid for C-887, F-206 (but must be present for C-866, C-867, C-880, C-884, C-885, E-761, E-861, E-871, E-873, M-8X0 and Mercury_GCS, too)

2.9.3. TAC?.vi

Returns the number of installed analog channels.
Valid for C-702, C-866, C-867, C-880, C-884, C-887, E-861, E-871, E-873, F-206, M-8X0, Mercury™, Mercury_GCS (but must be present for C-885 also)

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).

System no. TAC? Analog channels error in error out

2.9.4. TAV?.vi

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.

Valid for C-865, C-866, C-867, C-880, C-884, C-887, E-712, E-761, E-861, E-871, E-873, F-206, M-8X0, Mercury™, Mercury_GCS (but must be present for C-885 also)

C-865, C-866: Query = Value, Board = 1. Range and Power unit are not valid.

C-867, C-884, E-861, E-871, E-873, Mercury_GCS: Query = Value, Board = 1 to 4. Range and Power unit are not valid.

C-880: Query = Value. Range and Power unit are not valid

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

E-761: Query = Value, Board = 4. Range and Power unit are not valid. The output is the current voltage at the analog input line, with gain and offset.

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. Range and Power unit are not valid

System no. TAV? Analog value Range Power unit error in error out

2.9.5. TSC?.vi

Returns the number of available sensor channels.

Valid for C-413, C-887, E-517, E-518, E-709, E-710, E-712, E-725, E-727, E-753, E-754, E-755, E-761 (but must be present for Analog systems, C-867, C-877, C-885, C-891, E-816, E-861, E-870, E-871, E-873, F-206, M-8X0 and Mercury_GCS also)

E-517: Using the Sensor Enable parameter, ID 0x02000000, you can change the E-517 configuration in case of hardware changes, e.g. if you install additional sensor and/or amplifier channels in the system. If this parameter is changed, the Number Of Sensor Channels parameter is adapted automatically. E.g. if parameter 0x02000000 is set to "disabled" for a sensor channel, this sensor channel is no longer included in the TSC? response. See "Configure Axes and Channels" in the E-517 User manual for details.
E-709: The response comprises all ADC channels of the device: the "genuine" sensor (sensor integrated in the mechanics) and the "general purpose" analog input.

C-413, E-712, E-725, E-727: 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-754: The response comprises all ADC channels of the device: the "genuine" sensor (capacitive or incremental sensors integrated in the mechanics) and the "general purpose" analog input.

2.9.6. TSP.vi

Sets sensor position for the specified sensor number.

Valid for Mercury_GCS

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

2.9.7. TSP?.vi

Returns sensor position for the specified sensor number.

Valid for C-413, E-517, E-518, E-709, E-710, E-712, E-725, E-727, E-753, E-754, E-755, E-761, Mercury_GCS (but must be present for C-885 also)

E-517, E-518: If All sensors? = TRUE, then Sensor identifier? can be FALSE. The input signal channels to be queried with TSP? are the sensor channels of the piezo control electronics, IDs = 1 to 3 (actually available IDs depend on the response to TSC?).

E-761: If All sensors? = TRUE, then Sensor identifier? can be FALSE.

E-709: If All sensors? = TRUE, then Sensor identifier? must be FALSE. 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: If All sensors? = TRUE, then Sensor identifier? must be TRUE. Command is available in command level 1 only (see "CCL.vi", "CCL?.vi")

C-413, E-712, E-725, E-727: If All sensors? = TRUE, then Sensor identifier? can be FALSE. 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: If All sensors? = TRUE, then Sensor identifier? can be FALSE. 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-754: If All sensors? = TRUE, then Sensor identifier? must be FALSE. Sensors to query comprises all ADC channels of the device: can be the "genuine" sensor (capacitive sensor integrated in the mechanics, identifier = 1, or incremental sensor, identifier = 3) and the "general purpose" analog input (identifier = 2).

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

Mercury_GCS: If All sensors? = TRUE, then Sensor identifier? can be FALSE. Check HLP? answer to find out if TSP? is supported.
2.10. **Scan support.llb**

2.10.1. **Axis names.vi**

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"

Valid for: Analog systems, C-413, C-702, C-866, C-867, C-877, C-880, C-884, C-887, C-891, E-517, E-518, E-709, E-710, E-712, E-725, E-727, E-753, E-754, E-755, E-761, E-861, E-871, E-873, F-206, M-8X0, Mercury_GCS (but must be present for C-885 also). To support analog interfacing, VI must be present for E-816 also.

```
Names
```

2.10.2. **Calculate 1D scan positions.vi**

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. Scan direction can be

- Left & right (- & +) meaning that the scan starts at (Start position - ½ Range) and stops at (Start position + ½ Range),
- To the left (-) meaning that scan starts (Start position - Range) and stops at Start position, or
- To the right (+) meaning that scan starts at Start position and stops at (Start position + Range).

Valid for: C-866, C-867, C-880, C-884, C-887, E-761, E-861, E-871, E-873, F-206, M-8X0, Mercury_GCS (but must be present for C-885 also)

```
Scan direction
Start position
Step size
Range
1D position array
Minimum position
Maximum position
No. of steps
1D intensity array
```

2.10.3. **Maximum Intensity?.vi**

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.

Valid for: C-866, C-867, C-880, C-884, C-887, E-761, E-861, E-871, E-873, F-206, M-8X0, Mercury_GCS (but must be present for C-885 also)
2.11. Special command.llb

2.11.1. #24.vi

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.

Valid for Analog systems, C-413, C-702, C-843, C-843.PM, C-844, C-848, C-865, C-866, C-867, C-877, C-880, C-880K005, C-884, C-885, C-887, C-891, E-516, E-517, E-518, E-709, E-712, E-725, E-727, E-753, E-754, E-755, E-761, E-816, E-861, E-870, E-871, E-873, 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.

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

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.

System no. #24 error in error out

2.11.2. #4.vi

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

Valid for C-867, C-877, C-884,C-885,C-887, C-891, E-861, E-871, E-873, F-206, Hydra, Pollux, M-8X0, Mercury_GCS

C-887, 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).

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).

System no. #4 Status information Status information (txt) error in error out
2.11.3. #5.vi

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".

Valid for Analog systems, C-413, C-702, C-843, C-843.PM, C-844, C-848, C-865, C-866, C-867, C-877, C-880, C-884, C-885, C-887, C-891, E-517, E-518, E-709, E-712, E-725, E-727, E-753, E-754, E-755, E-761, E-861, E-870, E-871, E-873, Hydra, Pollux, Mercury, Mercury_GCS (but must be present for all other systems also)

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

C-887, 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.

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) coding in answer is different, please use #5_old.vi.

<table>
<thead>
<tr>
<th>System no.</th>
<th>Axis moving?</th>
</tr>
</thead>
<tbody>
<tr>
<td>error in</td>
<td>#5</td>
</tr>
<tr>
<td></td>
<td>error out</td>
</tr>
</tbody>
</table>

---

2.11.4. #6.vi

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.

Valid for C-702, C-848, C-880, C-887, E-517, E-518, E-761, F-206, M-8X0 (but must be present for C-866, C-867, C-884, C-885, E-861, E-871, E-873 and Mercury_GCS also)

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.

C-887, 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.

<table>
<thead>
<tr>
<th>System no.</th>
<th>Pos change?</th>
</tr>
</thead>
<tbody>
<tr>
<td>error in</td>
<td>#6</td>
</tr>
<tr>
<td></td>
<td>error out</td>
</tr>
</tbody>
</table>

---

2.11.5. #7.vi

Sends the single ASCII character 7 and returns the ready status of the controller. Sub-VI for "Wait for answer of longlasting command.vi".
Valid for C-413, C-702, C-843, C-843.PM, C-844, C-848, C-865, C-866, C-867, C-877, C-880, C-880K005, C-884, C-885, C-887, C-891, E-517, E-518, E-709, E-710, E-712, E-725, E-727, E-754, E-755, E-761, E-861, E-870, E-871, E-873, Hydra, Pollux, Mercury, Mercury_GCS (but must be present for E-516, E-753, E-816, F-206, M-8X0 also)

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

C-887, F-206, 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).

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

System no. #7 Ready? String read error in

---

2.11.6. **ACC.vi**

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

Valid for C-843, C-867, C-877, C-884, C-891, E-861, E-873, Hydra, Pollux, Mercury_GCS (but must be present for C-885 also)

C-843, C-867, C-877, C-884, C-891, E-861, Hydra, Pollux, Mercury_GCS: Without axis ID? = FALSE. Acceleration unit is mm/s².

E-873: Check HLP? answer to find out if ACC is supported. Without axis ID? = FALSE. Acceleration unit is mm/s².

Without axis ID? System no. Axes to set Acceleration values No.of digits Controller error error out

---

2.11.7. **ACC?.vi**

Returns closed-loop acceleration setting for specified axes.

Valid for C-843, C-867, C-877, C-884, C-891, E-861, E-873, Hydra, Pollux, Mercury_GCS (but must be present for C-885 also)

C-843, C-867, C-877, C-884, C-891, E-861, Hydra, Pollux, Mercury_GCS: If All axes? = TRUE, then Axis identifier? can be FALSE. Acceleration unit is mm/s².

E-873: Check HLP? answer to find out if ACC? is supported. If All axes? = TRUE, then Axis identifier? can be FALSE. Acceleration unit is mm/s².
2.11.8. BRA.vi

Switches brake for given axes on or off.
Valid for C-843, C-843.PM, C-844, C-848, C-880, C-884, C-891, Mercury™,
Mercury_GCS (but must be present for C-885 also)

2.11.9. BRA?.vi

Returns list of axes with a brake and status of the brakes for queried axes.
Requires SPA?.vi to be present.
Valid for C-843, C-843.PM, C-844, C-848, C-880, C-884, C-891, Mercury™,
Mercury_GCS (but must be present for C-885 also)
C-843, C-843.PM, C-844, Mercury™: Invert order should be FALSE. Brake status?
is not valid.
C-848, C-880: Invert order should be TRUE. Brake status? is not valid
C-884, C-891, Mercury_GCS: Invert order is not valid. VI returns axes with brakes
and their brake status.

2.11.10. CCL.vi

If password is correct, this vi sets the command level of the controller and queries
ERR?. Controller error is TRUE if selected system report error code not equal to
zero. Use "HLP?.vi" to determine which commands are available in the current
command level.
Valid for C-413, C-867, C-877, C-887, C-891, E-517, E-518, E-709, E-710, E-712,
E-725, E-727, E-753, E-754, E-755, E-870, E-871, E-873, F-206, M-8X0,
Mercury_GCS (but must be present for C-885 also)
C-413, C-867, C-877, C-891, E-517, E-518, E-709, E-710, E-712, E-725, E-727, E-
753, E-754, E-755, E-870, E-871, E-873, F-206, M-8X0, Mercury_GCS: 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".
C-867, Mercury_GCS: 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).
2.11.11. CCL?.vi

Returns the current command level.

Valid for C-413, C-702, C-867, C-877, C-887, C-891, E-517, E-518, E-709, E-710, E-712, E-725, E-727, E-753, E-754, E-755, E-816, E-870, E-871, E-873, F-206, M-8X0, Mercury_GCS (but must be present for C-885 also)

C-867, Mercury_GCS: 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?s 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).

2.11.12. CST.vi

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.

Valid for C-702, C-843, C-843.PM, C-844, C-848, C-865, C-866, C-867, C-877, C-880, C-884, C-887, C-891, E-710, E-761, E-871, E-873, F-206, Hydra, Pollux, M-8X0, Mercury, Mercury_GCS (but must be present for C-885 also)

C-877, C-884, E-780, C-891, E-873: 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).

C-867, E-871, Mercury_GCS: 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). Only valid for separate axes A and/or B.

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

Returns the name of the connected stage for queried axes.
Valid for C-413, C-702, C-843, C-843.PM, C-844, C-848, C-865, C-866, C-867, C-877, C-880, C-884, C-885, C-887, C-891, E-709, E-710, E-712, E-725, E-727, E-753, E-754, E-755, E-761, E-861, E-871, E-873, F-206, Hydra, Pollux, M-8X0, Mercury, Mercury_GCS
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-877, C-884, C-885, C-891, E-709, E-710, E-712, E-725, E-727, E-753, E-754, E-755, E-761, E-861, E-871, E-873, Hydra, Mercury™, Mercury_GCS: If All axes? = TRUE, then Axis identifier? can be FALSE
C-887, F-206, M-8X0: Check HLP?/HELP answer to find out if CST? is supported
Pollux: If All axes? = TRUE, then Axis identifier? can be FALSE. See HydraPollux_GCS_DLL Manual for specialities of the Pollux controller family regarding stage name settings.

2.11.14. CTO.vi

This vi Configures the Trigger Output conditions for the given trigger output line, waits 100 ms and queries ERR?. The trigger output will always be periodically. The trigger output only becomes active when enabled with TRO. Note: Do not use DIO when an axis was configured with CTO. See User Manual or 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 not equal to 0.
Valid for: C-413, C-702, C-843, C-866, C-867, C-877, C-884, E-517, E-518, E-709, E-712, E-725, E-727, E-753, E-754, E-871, E-873, Mercury, Mercury_GCS(but must be present for C-885 also)
C-413: TriggerOut to set can be 1 to 5.
C-702: TriggerOut to set can be 1 to 8. DIO uses A-H to designate the same lines CTO calls 1-8.
C-843: TriggerOut to set can be 1 to 4 (4-axis card) or 2 (2-axis card).
C-866: TriggerOut to set can be 1.
C-867, C-877: Check ControllerUser Manual for available output trigger numbers. Check HLP? answer to find out if CTO is supported.
C-884, E-871, E-873: Check C-884 User Manual for available output trigger numbers.
E-517, E-518: TriggerOut to set can be 1 to 3. The trigger output is immediately active when the condition given by the CTO configuration is fulfilled (there is no TRO command). If the CTO Parameter ID is "Trigger Mode" and the Parameter value is "Generator Trigger", then the trigger points in the waveform must be set with TWS and/or with WGO.
The current trigger output configuration is saved with the WPA command, in addition to the current parameter values and other settings.

The width of a trigger pulse is 30 µs by default, except with the MinMaxThreshold trigger mode where the pulse width depends on the threshold settings. You can change the default pulse width using the Pulse Width parameter, ID 0xE000900. Possible values are in the range of 10 to 150 µs.

The assignment of the trigger lines to the axes of the controller is fixed (DIO_O1 belongs to the first axis (A by default), DIO_O2 to the second axis (B by default) and DIO_O3 to the third axis (C by default).

E-709: TriggerOut to set can be 1 to 2. The trigger output is immediately active when the condition given by the CTO configuration is fulfilled (there is no TRO command). If the CTO Parameter ID is "Trigger Mode" and the Parameter value is "Generator Trigger", then the trigger points in the waveform must be set with TWS.

E-712: TriggerOut to set can be 1 to 7. The trigger output is immediately active when the condition given by the CTO configuration is fulfilled (there is no TRO command). If the CTO Parameter ID is "Trigger Mode" and the Parameter value is "Generator Trigger", then the trigger points in the waveform must be set with TWS. By default, axis 1 is connected to TriggerOut line 1, axis 2 to line 2, axis 3 to line 3, ..., axis n to line n. If the number of TriggerOut lines exceeds the number of axes, the "surplus" lines are all connected to the last axis.

E-725: TriggerOut to set can be 1 to 7. The trigger output is immediately active when the condition given by the CTO configuration is fulfilled (there is no TRO command). If the CTO Parameter ID is "Trigger Mode" and the Parameter value is "Generator Trigger", then the trigger points in the waveform must be set with TWS. By default, axis 1 is connected to TriggerOut line 1, axis 2 to line 2, axis 3 to line 3, ..., axis n to line n. If the number of TriggerOut lines exceeds the number of axes, the "surplus" lines are all connected to the last axis.

E-727: TriggerOut to set can be 1 to 3. The trigger output is immediately active when the condition given by the CTO configuration is fulfilled (there is no TRO command). If the CTO Parameter ID is "Trigger Mode" and the Parameter value is "Generator Trigger", then the trigger points in the waveform must be set with TWS. By default, axis 1 is connected to TriggerOut line 1, axis 2 to line 2, axis 3 to line 3. If the number of TriggerOut lines exceeds the number of axes, the "surplus" lines are all connected to the last axis.

E-753: TriggerOut to set can be 1. The trigger output is immediately active when the condition given by the CTO configuration is fulfilled (there is no TRO command). If the CTO Parameter ID is "Trigger Mode" and the Parameter value is "Generator Trigger", then the trigger points in the waveform must be set with TWS.

E-754: TriggerOut to set can be 1. The trigger output is immediately active when the condition given by the CTO configuration is fulfilled (there is no TRO command). If the CTO Parameter ID is "Trigger Mode" and the Parameter value is "Generator Trigger", then the trigger points in the waveform must be set with TWS.

Mercury: TriggerOut to set can be 1 to N (N: number of connected axes, if supported by corresponding Mercury firmware).

Mercury_GCS: Check Mercury User Manual for available output trigger numbers.
2.11.15. CTO?.vi

Returns the Trigger Output configuration for the given trigger output line.
Valid for: C-413, C-702, C-843, C-866, C-867, C-877, C-884, E-517, E-518, E-709, E-712, E-725, E-727, E-753, E-754, E-871, E-873, Mercury, Mercury_GCS (but must be present for C-885 also)

C-413: TriggerOut to query can be 1 to 5.
C-702: TriggerOut to query can be 1 to 8.
C-843: TriggerOut to query can be 1 to 4 (4-axis card) or 2 (2-axis card).
C-866: TriggerOut to query can be 1.
C-867, C-877: Check Controller User Manual for available output trigger numbers.
Check HLP? answer to find out if CTO? is supported.
C-884, E-871, E-873: Check Controller User Manual for available output trigger numbers.
E-517, E-518: TriggerOut to query can be 1 to 3.
E-709: TriggerOut to query can be 1 to 2.
E-712: TriggerOut to query can be 1 to 7.
E-725: TriggerOut to query can be 1 to 7.
E-727: TriggerOut to query can be 1 to 3.
E-753: TriggerOut to query can be 1.
E-754: TriggerOut to query can be 1.
Mercury: TriggerOut to query can be 1 to N (N: number of connected axes, if supported by corresponding Mercury firmware).
Mercury_GCS: Check Mercury User Manual for available output trigger numbers.

2.11.16. DEC.vi

Sets closed-loop deceleration and checks for error. If Without axis ID? is TRUE, then Axes to set is ignored and first field of Deceleration values array is used for deceleration command. Number of digits is the number of digits after the decimal point in the deceleration value(s) that will be sent. Controller error is TRUE if selected system reports error code which is not 0.
Valid for C-843, C-867, C-877, C-884, C-891, E-861, E-873, Mercury_GCS (but must be present for C-885 also)
C-843, C-867, C-877, C-884, C-891, E-861, Mercury_GCS: Without axis ID? = FALSE. Deceleration unit is mm/s².
E-873: Check HLP? answer to find out if DEC is supported. Without axis ID? = FALSE. Deceleration unit is mm/s².
2.11.17. DEC?.vi

Returns closed-loop deceleration setting for specified axes.

Valid for C-843, C-867, C-877, C-884, C-891, E-861, E-873, Hydra, Pollux, Mercury_GCS (but must be present for C-885 also)

C-843, C-867, C-877, C-884, C-891, E-861, Hydra, Pollux, Mercury_GCS: If All axes? = TRUE, then Axis identifier? can be FALSE. Deceleration unit is mm/s².

E-873: Check HLP? answer to find out if DEC? is supported. If All axes? = TRUE, then Axis identifier? can be FALSE. Deceleration unit is mm/s².

2.11.18. DIO.vi

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.

Valid for C-413, C-702, C-843, C-843.PM, C-848, C-865, C-866, C-867, C-877, C-880, C-884, C-887, C-891, E-861, E-871, E-873, F-206, Hydra, Mercury™, Mercury_GCS, M-8X0 (but must be present for C-885 also)

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, C-891, E-861, E-871, E-873, Mercury_GCS: For DO mode format = Boolean, DO's to command can be 1-4 and only one DO per command allowed. For DO mode format = hexadecimal, DO's to command and DO mode are not valid and DO pattern must be set correctly (all DO's are set with one single command).

C-877: For DO mode format = Boolean, DO's to command can be 1-4 and only one DO per command allowed. For DO mode format = hexadecimal, DO's to command and DO mode are not valid and DO pattern must be set correctly (all DO's are set with one single command). Check HLP? answer to find out if DIO is supported

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.

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.

2.11.19. DIO?.vi

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.

Valid for C-413, C-702, C-843, C-843.PM, C-848, C-865, C-866, C-867, C-877, C-880, C-884, C-887, C-891, E-517, E-518, E-709, E-712, E-727, E-754, E-761, E-861, E-871, E-873, F-206, Hydra, Mercury, Mercury_GCS, M-8X0 (but must be present for C-885 also)

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 . 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, E-873, 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 1-4. Query pattern? can be TRUE.

C-877: If All DI's = TRUE, then DI identifier must be FALSE. Invert order for TVI? is not valid. DI's to query are 1-4. Query pattern? can be TRUE. Check HLP?/HELP answer to find out if DIO is supported.

C-891: If All DI's = TRUE, then DI identifier must be FALSE. Invert order for TVI? is not valid. DI's to query per command allowed and DI's to query can be 1 to 4. Query pattern? can be TRUE.

E-517, E-518: 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, E-727: If All DI's = TRUE, then DI identifier can be FALSE. Invert order for TVI? is not valid. Query pattern? is not valid. Check HLP? answer to find out if CTI? is supported.

E-754: If All DI's = TRUE, then DI identifier can be FALSE. Invert order for TVI? is not valid. DI's to query are 1-2. Query pattern? is not valid.

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

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).

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.

2.11.20. DRC.vi

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 an error code which is not 0.

Valid for C-413, C-702, C-843, C-866, C-867, C-884, C-887, C-891, E-517, E-518, E-709, E-710, E-712, E-725, E-727, E-753, E-754, E-755, E-861, E-871, E-873, F-206, M-8X0, Mercury_GCS(but must be present for C-885 also)

C-843: Trigger option must be 0. 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: Trigger option must be 0. 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-877: See User Manual for available record options, number of data recorder tables and points per table. Trigger option must be 0.

C-884: See User Manual for available record options. Trigger option must be 0. The C-884 has eight data recorder tables with 8192 points per table.
C-891: See User Manual for available record options. Trigger option must be 0. The C-891 has four data recorder tables with 4096 points per table. The number of tables can be changed to 1, 2 or 8 by setting the appropriate parameter value, see User Manual for details.

E-517, E-518: Trigger option must be 0. 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. Trigger option must be 0. By default, the number of data recorder tables is 4. It can be reduced by setting the appropriate parameter value, see User Manual for details.

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

C-413, E-712, E-725, E-753, E-754: Trigger option must be 0. 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: Trigger option must be 0. See User Manual for available record options. The E-861 has two data recorder tables with 1024 points per table.

E-871, E-873: Trigger option must be 0. See User Manual for available record options. The E-871, E-873 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: Trigger option must be 0. See User Manual for available record options. The Mercury_GCS has two data recorder tables with 1024 points per table.

2.11.21. DRC?.vi

Returns the data recording configuration (Source ID and Rec. option) for the queried record table.

GCS 2.0: Trigger option is not valid.

Valid for C-413, C-702, C-843, C-866, C-867, C-877, C-884, C-887, C-891, E-517, E-518, E-709, E-710, E-712, E-725, E-727, E-753, E-754, E-755, E-861, E-871, E-873, F-206, M-8X0, Mercury_GCS (but must be present for C-885 also)

C-413, C-843, C-866, C-867, C-877, C-887, C-891, E-517, E-518, E-709, E-710, E-712, E-725, E-727, E-753, E-754, E-861, E-871, E-873, F-206, M-8X0, Mercury_GCS: 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).

2.11.22. DRL?.vi

This VI returns the number of recorded data values for the given record tables.
Valid for C-413, C-867, C-877, C-884, C-887, C-891, E-518, E-709, E-712, E-725, E-727, E-753, E-754, E-871, E-873, F-206, M-8X0, Mercury_GCS (but must be present for C-885 also)
C-867, Mercury_GCS: 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).

2.11.23. DRR? and display data.vi

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 comm.error may occur. If Sample time is zero, it is set to 1.0 for displaying data in the 2D graph only.
Valid for Analog systems, C-413, C-702, C-843, C-866, C-867, C-877, C-884, C-887, C-891, E-517, E-518, E-709, E-710, E-712, E-725, E-727, E-753, E-754, E-755, E-761, E-861, E-871, E-873, F-206, M-8X0, Mercury_GCS(but must be present for C-885 also). To support analog interfacing, VI must be present for E-816 also.
Analog: Rec. table IDs, x0, N and Nmax are not valid. Without parameter? must be TRUE.
C-413: X0 >= 1. Nmax = 4096. 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-702: Nmax = 262144. X0 >= 0.
C-843: X0 >= 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. 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. If N = -1 all points of the last record are returned.
C-866: \( X_0 \geq 1. \) Nmax = 32,256. If \( N = -1 \) all points of the last record are returned.

C-867: \( X_0 \geq 1. \) Nmax = 8192. The number of tables is 4.

C-877: \( X_0 \geq 1. \) Nmax = 8192. The number of tables is 4.

C-884: \( X_0 \geq 1. \) Nmax = 8192. The number of tables is 8.

C-891: \( X_0 \geq 1. \) Nmax = 16384. The C-891 has four data recorder tables with 4096 points per table. It can be changed to 1, 2 or 8 by setting the appropriate parameter value, see User Manual for details.

E-517, E-518: \( X_0 \geq 1. \) Nmax = 8192. The number of tables is 3.

E-709: \( X_0 \geq 1. \) Nmax = 4096. 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-710: \( X_0 \geq 1. \) Nmax = 32,256.

E-712: \( X_0 \geq 1. \) Nmax = 262,144. 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, E-727: \( X_0 \geq 1. \) Nmax = 262,144. 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 controllerUser Manual for details.

E-753: \( X_0 \geq 1. \) Nmax = 65,536. 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-754: \( X_0 \geq 1. \) Nmax = 1,048,576. The 1,048,576 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-754 User Manual for details.

E-755: \( X_0 \geq 1. \) Nmax = 4,096.

E-761: \( X_0 \geq 0. \) 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-861: \( X_0 \geq 1. \) Nmax = 1024. Two data recorder tables with 1024 points per table are provided.

E-871, E-873: \( X_0 \geq 1. \) Nmax = 1024. The number of tables is 2.

C-887, F-206: \( X_0 \geq 1. \) See C-887 User Manual for Nmax default value (can be changed with SPA.vi). 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: 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).

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 2.0 and higher, see controller User Manual for available data recorder tables and points.

Mercury_GCS: Xo >= 1. Nmax = 1024. Two data recorder tables with 1024 points per table are provided.

2.11.24. DRR?.vi

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.

Valid for Analog systems, C-413, C-702, C-843, C-866, C-867, C-877, C-884, C-887, C-891, E-517, E-518, E-709, E-710, E-712, E-725, E-727, E-753, E-754, E-755, E-761, E-861, E-871, E-873, F-206, M-8X0, Mercury_GCS (but must be present for C-885 also). To support analog interfacing, VI must be present for E-816 also.

Analog: Rec. table IDs, x0, N and Nmax are not valid. Without parameter? must be TRUE.

C-413: X0 >= 1. Nmax = 4096. 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-702: Nmax = 262144. X0 >= 0.

C-843: X0 >= 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. 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. If N = -1 all points of the last record are returned.

C-866: X0 >= 1. Nmax = 32,256. If N = -1 all points of the last record are returned.

C-877: Xo >= 1. Nmax = 8192. The number of tables is 4.

C-877: Xo >= 1. Nmax = 8192. The number of tables is 4.
C-884: \( X_0 \geq 1 \). \( N_{\text{max}} = 8192 \). The number of tables is 8.

C-891: \( X_0 \geq 1 \). \( N_{\text{max}} = 16384 \). The C-891 has four data recorder tables with 4096 points per table. It can be changed to 1, 2 or 8 by setting the appropriate parameter value, see User Manual for details.

E-517, E-518: \( X_0 \geq 1 \). \( N_{\text{max}} = 8192 \). The number of tables is 3.

E-709: \( X_0 \geq 1 \). \( N_{\text{max}} = 4096 \). 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-710: \( X_0 \geq 1 \). \( N_{\text{max}} = 32,256 \).

E-712: \( X_0 \geq 1 \). \( N_{\text{max}} = 262,144 \). 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, E-727: \( X_0 \geq 1 \). \( N_{\text{max}} = 262,144 \). 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: \( X_0 \geq 1 \). \( N_{\text{max}} = 65,536 \). 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 controller User Manual for details.

E-754: \( X_0 \geq 1 \). \( N_{\text{max}} = 1,048,576 \). The 1,048,576 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-754 User Manual for details.

E-755: \( X_0 \geq 1 \). \( N_{\text{max}} = 4,096 \).

E-761: \( X_0 \geq 0 \). 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-861: \( X_0 \geq 1 \). \( N_{\text{max}} = 1024 \). Two data recorder tables with 1024 points per table are provided.

E-871, E-873: \( X_0 \geq 1 \). \( N_{\text{max}} = 1024 \). The number of tables is 2.

C-887, F-206: \( X_0 \geq 1 \). See controller User Manual for \( N_{\text{max}} \) default value (can be changed with SPA.vi). 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: 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 controller User Manual for Nmax default value (can be changed with SPA.vi).

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: Xo >= 1. Nmax = 1024. Two data recorder tables with 1024 points per table are provided.

![Diagram of data recorder](image)

### 2.11.25. DRT.vi

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 not equal to 0.

Valid for C-413, C-702, C-843, C-866, C-867, C-877, C-884, C-887, C-891, E-518, E-709, E-712, E-725, E-727, E-753, E-754, E-755, E-861, E-871, E-873, F-206, M-8X0, Mercury_GCS (but must be present for C-885 also)

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-877, C-884, C-891, C-891, E-518, E-709, E-712, E-754, E-861, E-871, E-873, 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.

E-727, E-754: See User Manual for available trigger sources and values. The specified trigger source is set for all data recorder tables.

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 for available trigger sources and values. DataRecorderTable = 0 (the specified trigger source is set for all data recorder tables).
2.11.26. **DRT?.vi**

Returns the Data Recorder Trigger source and value for the queried data recorder tables.

Valid for C-413, C-702, C-843, C-866, C-867, C-877, C-884, C-887, C-891, E-518, E-709, E-712, E-725, E-727, E-753, E-754, E-755, E-861, E-871, E-873, F-206, M-8X0, Mercury_GCS (but must be present for C-885 also)

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).

2.11.27. **HDR?.vi**

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 correspondingly section content.

Valid for C-413, C-866, C-867, C-877, C-884, C-887, C-891, E-517, E-518, E-709, E-712, E-725, E-727, E-753, E-754, E-861, E-871, E-873, F-206, M-8X0, Mercury_GCS (but must be present for C-885 also)

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).

2.11.28. **MVE.vi**

Vector path move to specified absolute position. Specified axes to move start moving simultaneously. The current settings for velocity, acceleration and deceleration define the maximum possible values. The actual velocity is defined by the slowest axis. No other motion commands are allowed during vector move. No. of digits is the number of digits after the decimal point in the position value(s) that will be sent.

Valid for C-843, C-867, C-884, Hydra, Mercury_GCS (but must be present for C-885 also)

C-867: Check HLP? answer to determine if MVE is supported.
2.11.29. POS.vi

Assigns new position value to current position without moving the stage. Command can only be used when the reference mode is switched off (see "RON.vi"). No. of digits is the number of digits after the decimal point in the position value(s) that will be sent.

Warning: If the current position is incorrectly set on an axis with reference mode OFF, the stage can be driven into the mechanical hard stop when moving to a position which is thought to be within the travel range of the stage, but actually is not.

Valid for C-413, C-702, C-843, C-843.PM, C-848, C-865, C-866, C-867, C-877, C-880, C-880K005, C-884, C-885, C-891, E-712, E-755, E-861, E-871, E-873, Hydra, Pollux, Mercury™, Mercury_GCS

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

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

E-755: Command not available for E-755.101

E-861: With open-loop systems, this command is not useful because there is no position sensor.

2.11.30. RBT.vi

Reboots the controller. Controller behaves like after a cold start.

Valid for C-413, C-702, C-867, C-877, C-884, C-885, C-887, C-891, E-517, E-518, E-709, E-712, E-725, E-727, E-753, E-754, E-755, E-761, E-861, E-870, E-871, E-873, F-206, Hydra, Pollux, M-8X0, Mercury_GCS

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).

C-885: Reboots master only.

2.11.31. RPA.vi

Replaces the current values of the given parameters to restore for "Affected axes" in the controller RAM with the values from non-volatile memory, waits 5000 ms and queries ERR?. For axis-related parameters, "Affected axes" is the axis name, for piezo- or sensor-related parameters, the channel number, otherwise a parameter-
related code. If parameter no. is in decimal format, use "Parameter to restore" input, for hexadecimal parameter numbers use "Parameter to restore (hex)" input and switch "Parameter no. format" to TRUE. Do not mix decimal and hex parameter numbers in one call. See GCS DLL manual for available parameter numbers. If "Affected axes" is an empty array, RPA is sent without axis (item) and parameter specification and controller restores all values for all axes (items). Controller error is TRUE if selected system reports error code not equal to 0. Valid for C-413, C-867, C-877, C-884, E-517, E-518, E-709, E-710, E-712, E-725, E-727, E-753, E-754, E-755, E-761, E-861, E-870, E-871, E-873, Mercury_GCS (but must be present for C-885 also)

E-710, E-761: If Affected axes = empty array, all parameters for all axes are restored. Parameter no. format is TRUE (hex.). Use "HPA?.vi" to get valid parameter numbers or see User Manual.

C-413, E-517, E-518, E-712, E-725, E-727, E-753, E-754, E-755: If Affected axes = empty array, all parameters for all axes are restored. Parameter no. format is TRUE (hex.). Use "HPA?.vi" to get valid parameter numbers or see User Manual.

C-867, C-877, C-884, E-709, E-861, E-870, E-871, E-873, Mercury_GCS: You can reset either all parameters or one single parameter with RPA. If Affected axes = empty array, all parameters for all axes are restored. Parameter no. format is TRUE (hex.). Use "HPA?.vi" to get valid parameter numbers or see User Manual.

### 2.11.32. RTR.vi

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. Controller error is TRUE if selected system report error code not equal to zero.

Valid for C-413, C-843, C-867, C-877, C-884, C-887, C-891, E-517, E-518, E-709, E-712, E-725, E-727, E-753, E-754, E-761, E-861, E-871, E-873, F-206, M 8X0, Mercury_GCS (but must be present for C-885 also)

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).

### 2.11.33. RTR?.vi

Returns the current table rate.

Valid for C-413, C-843, C-867, C-877, C-884, C-887, C-891, E-517, E-518, E-709, E-712, E-725, E-727, E-753, E-754, E-761, E-861, E-871, E-873, F-206, M 8X0, Mercury_GCS (but must be present for C-885 also)
2.11.34. **SAI.vi**

Set axis identifier. With this command the axis identifiers of the connected axes can be changed.

GCS 1.0: Only one character is allowed as axis ID. Valid axis IDs can be listed with TVI?.vi.

GCS 2.0: Characters that Axis IDs can contain can be listed with TVI?.vi.

Please run "Define connected axes.vi" with new axis IDs after renaming axes.

Valid for C-702, C-843, C-843.PM, C-844, C-848, C-865, C-866, C-867, C-880, C-884, C-891, E-517, E-518, E-709, E-710, E-761, E-871, E-873, Hydra, Pollux, Mercury, Mercury_GCS (but must be present for C-885 also)

E-517, E-518: Maximum length for axis name is 8 characters. SAI affects the Axis Name parameter, ID 0x07000600, in volatile memory (RAM). To save the currently valid value to non-volatile memory, where it becomes the power-on default, you must use WPA. Changes not saved with WPA will be lost when the controller is powered down or rebooted.

E-709: Maximum length for axis name is 4 characters. SAI affects the Axis Name parameter, ID 0x07000600, in volatile memory (RAM). To save the currently valid value to non-volatile memory, where it becomes the power-on default, you must use WPA. Changes not saved with WPA will be lost when the controller is powered down or rebooted.

C-867, C-877, C-884, C-891, E-761, E-871, E-873, Mercury_GCS: The settings are automatically written to non-volatile memory.

Hydra, Pollux: To save the currently valid axis identifier to non-volatile memory, where it becomes the power-on default, you must use WPA. Changes not saved with WPA will be lost when the controller is powered down or rebooted.

---

2.11.35. **SEP.vi**

If Password is correct, this VI sets parameters for commanded axes to EPROM, waits 100 ms and queries ERR?. For axis-related parameters, Axis to set is the axis name; for piezo- or sensor-related parameters, the channel number; otherwise a parameter-related code. If parameter number is in decimal format, use Parameter number input, for hexadecimal parameter numbers use Parameter number (hex.) input and switch Parameter no. format to TRUE. For numeric parameter values use Parameter value input, for parameter strings use Parameter string input and switch Parameter format to TRUE. Do not mix decimal and hex. parameter numbers or numeric and string parameter values in one call. Parameter numbers which can be set depend on current CCL level. Use "HPA?.vi" to get valid
parameter numbers or see GCS DLL manual for available parameter numbers and values. No. of digits is the number of digits after the decimal point in the numeric parameter value(s) that will be sent. Hidden error is TRUE if selected system reports error code which is not 0.

Note: This command sets the same parameters as SPA, but SPA writes them only to volatile memory, while SEP only writes to non-volatile memory. After parameters were set with SEP, use RPA to activate them (write them to volatile memory), or they become active after next power up.

Valid for C-413, C-867, C-877, C-884, C-891, E-517, E-518, E-709, E-710, E-712, E-725, E-727, E-753, E-754, E-755, E-761, E-861, E-870, E-871, E-873, Mercury_GCS (but must be present for C-885 also)

C-413: Parameter no. format is TRUE (hex.). Required command level depends on parameter ("CCL.vi") . Do not set more than 4 parameter no. at once.

C-867, C-877, C-891, E-871, E-873, Mercury_GCS: Parameter no. format is TRUE (hex.). You can write only one single parameter per SEP command. See User manual for more information.

C-884: Parameter no. format is TRUE (hex.). You can write up to four single parameters per SEP command. See User manual for more information.

E-517, E-518: Parameter no. format is TRUE (hex.). Requires command level 1 ("CCL.vi") for parameter 0x02000000 (used to enable/disable axes which is only required if the hardware configuration is changed), 0x04000E00 and 0x04000E01 (used to configure unit and format of the LCD display on the E-517 front panel).

E-709: Parameter no. format is TRUE (hex.). You can write only one single parameter per SEP command. Required command level depends on parameter ("CCL.vi")

E-710: Parameter no. format is TRUE (hex.). Command is available in command level 1 only (see "CCL.vi", "CCL?.vi"). Command writes parameters to Eprom and RAM

E-755, E-761: Parameter no. format is TRUE (hex.)

E-712, E-725, E-727, E-753, E-754: Parameter no. format is TRUE (hex.). Requires command level 1 ("CCL.vi"). Do not set more than 10 parameters at once.

E-861: Parameter no. format is TRUE (hex.). You can write only one single parameter per SEP command. The GEMAC parameters (ID 0x7000010 to ID 0x700001F) can not be changed with SEP. Use SPA and WPA instead to save their values to non-volatile memory. See "GEMAC Parameter Adjustment" in the E-861 User manual for more information.

E-870: Parameter no. format is TRUE (hex.). You can write only one single parameter per SEP command.
2.11.36. SEP?.vi

Returns parameter values from non-volatile memory for queried axes and parameter numbers. For axis-related parameters, Axes to query is the axis name; for piezo- or sensor-related parameters, the channel number; otherwise a parameter-related code. If parameter number is in decimal format, use "Parameter no." input, for hexadecimal parameter numbers use "Parameter no. (hex)" input and switch "Parameter no. format" to TRUE. Do not mix decimal and hex. parameter numbers in one call. If Without axes? is TRUE, all available parameter for all axes/designators are returned. For parameter numbers which output a string use Parameter string output. See GCS DLL Manual for available parameter numbers.

Valid for C-413, C-867, C-877, C-884, C-891, E-517, E-518, E-709, E-710, E-712, E-725, E-727, E-753, E-754, E-755, E-761, E-861, E-870, E-871, E-873, Mercury_GCS(but must be present for C-885 also)

All systems: Use "HPA?.vi" to get valid parameter numbers. Parameter no. format is TRUE (hex).

C-413: Do not query more than 6 parameter no. at once (except with Without axes? = TRUE).
C-867, C-877, C-891, E-709, E-871, E-873, Mercury_GCS: Only one parameter value for only one axis per command allowed. Use "Without axes?" = TRUE for a query of all parameters.
C-884: Use Without axes? = TRUE for a query of all parameters.
E-712, E-725,E-727,E-753, E-754: Do not query more than 10 parameter no. at once (except with Without axes? = TRUE).
E-861: Only one parameter value for only one axis per command allowed. Use "Without axes?" = TRUE for a query of all parameters.
E-870: Only one parameter value for only one axis per command allowed. Use Without axes? = TRUE for a query of all parameters.

2.11.37. SMO.vi

Sets the motor output directly and checks for error. Command will only be executed if channel is in servo-off mode (SVO.vi).

Caution: In servo-off mode limit switches are not enabled!

No. of digits is the number of digits after the decimal point in the motor output value(s) that will be sent. Controller error is TRUE if selected system reports an error code which is not 0.

Valid for C-702, C-843, C-843.PM, C-844, C-848, C-865, C-866, C-867, C-877, C-880, C-884, C-891, E-871, E-873, Mercury_GCS(but must be present for C-885 also)

All systems: No. of digits is 0.

E-873: Check HLP? answer to find out if SMO is supported.
2.11.38. **SMO?.vi**

Returns the current motor output. In servo-on mode, the actual value as set by the regulator is reported. In servo-off mode, the value set by the last SMO command is reported.

Valid for C-702, C-843, C-843.PM, C-844, C-848, C-865, C-866, C-867, C-877, C-880, C-884, C-891, E-871, E-873, Hydra, Mercury_GCS (but must be present for C-885 also)

C-702, C-848, C-865, C-866, C-867, C-877, C-880, C-884, C-891, E-871, Hydra, Mercury_GCS: If All axes? = TRUE, then Axis identifier? must be FALSE

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

E-873: Check HLP? answer to find out if SMO is supported. If All axes? = TRUE, then Axis identifier? can be FALSE.

2.11.39. **SRG?.vi**

Returns register values for queried axes and register numbers. See Motion Processor or Controller User Manual for a description of bit-coded answer. If Query all? is TRUE, Axes to query and Register no. are ignored and command is sent without any parameters to query all register numbers for all axes.

Valid for C-413, C-843, C-843.PM, C-866, C-867, C-877, C-884, C-885, C-891, E-712, E-754, E-861, E-871, E-873, Hydra, Pollux, Mercury™, Mercury_GCS

C-843, C-843.PM: Register numbers are:
- 1: Event status register
- 2: Activity status register
- 3: Signal status register

C-866: Register numbers are:
- 1: Event status register
- 2: Activity status register
- 3: Signal status register
- 4: Signal sense

C-413, C-867, C-877, C-884, C-885, C-891, E-712, E-754, E-861, E-871, E-873, Hydra, Pollux, Mercury, Mercury_GCS: Register number is:
- 1: Status register

E-712: Check HLP? answer to find out if SRG? is supported.
2.11.40. SST.vi

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.

Valid for C-702, C-848, C-867, C-880, C-887, E-871, E-873, F-206, M-8X0, Mercury_GCS (but must be present for C-885 also)

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").

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, E-873: 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)


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

2.11.41. SST?.vi

Returns step-size setting (e.g. for manual control pad, see also "SST.vi").

Valid for C-702, C-848, C-867, C-880, C-887, E-871, E-873, F-206, M-8X0, Mercury_GCS (but must be present for C-885 also)

C-702, C-867, C-880, E-871, E-873: If All axes = TRUE, then Axis identifier can be FALSE

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

C-887, F-206, M-8X0: All axes = TRUE, Axis identifier = FALSE. Only values for axes X,Y,Z,U,V,W are valid

Mercury_GCS: Check HLP? answer to find out if SST? is supported.
2.11.42. STA?.vi

Returns axis status (integer). Required by "General wait for movement to stop.vi" and "Wait for axes to stop.vi".
Valid for C-702, C-848, C-880, C-880K005, C-887, F-206, M-8X0 (but must be present in Special command.lib for all other systems, too)
C702: 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. Check HLP?/HELP answer to find out if STA? is supported. Command is equivalent to #4. For details see User Manual of the controller.

2.11.43. STE.vi

Performs a single step-move from the current position with specified step size (amplitude). If supported, "Delay" defines the number of servo loops between position recording (GCS 2.0: Delay must be 0). Controller saves a definite number of position values, which can be read out with STE?.vi (GCS 1.0) or DRR? (GCS 2.0). No. of digits is the number of digits after the decimal point in the step size values that will be sent. For an impulse-move, see "IMP.vi".
Valid for Analog systems, C-413, C-843, C-843,PM, C-848, C-865, C-866, C-867, C-877, C-880, C-884, C-887, C-891, E-517, E-518, E-709, E-710, E-712, E-725, E-727, E-753, E-754, E-755, E-761, E-861, E-871, E-873, F-206, M-8X0, Mercury_GCS(but must be present for C-885 also). To support analog interfacing, VI must be present for E-816 also.
Analog: Delay = 0. 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. Delay = 0. 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. Delay = 0. 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. Delay = 0. 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-877: See User Manual for available record options, number of data recorder tables and points per table. 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.

C-891: See Controller User Manual for details. 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, E-518: 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 servocycles used for data recording depends on the setting made with RTR.

Motion commands like STE are not allowed when the controller is in OFFLINE mode or when the wave generator output is active. When a macro is running on the controller, STE will be executed not until the macro is finished or stopped. See "Control Value Generation" and "Control Modes" in the controller 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 servocycles 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 32256 position values. Delay = 0. "Table Rate" parameter, set with "SPA.vi", is used as sampling interval instead of Delay. Caution: "Table Rate" parameter influences Wave Generator also, 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 servocycles 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, E-727: 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 servocycles 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 servocycles 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-754: Controller saves up to 1,048,576 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: Delay = 0. Controller saves 4096 position values. Use DRR?.vi or DRR? and display data.vi to read recorded values back.

E-761: Controller saves 8192 position values. Delay = 0. The number of servo cycles used for data recording depends on the setting made with RTR.vi. 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, E-873: 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: Controller saves 1,024 position values (number can be changed by setting the appropriate parameter value, see User Manual). Use DRR?.vi or DRR? and display data.vi to read recorded values back. 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).

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.

2.11.44. TCV?.vi

Returns current commanded velocity for specified axes. In contrast to VEL?, during acceleration and deceleration TCV? reports the value currently produced by the trajectory generator. The sign of the returned value depends on the commanded direction of motion. If the velocity of the axis is controlled by an axis of a Human Interface Device (i.e. HI control is activated, see HIA and HIN), TCV? returns the current value according to the deflection of the HID axis, and VEL? returns the maximum velocity set by the corresponding parameter or VEL.
Valid for C-867, C-877, C-884, C-891, Mercury_GCS (but must be present for C-885 also)

C-867, Mercury_GCS: If All axes? = TRUE, then Axis identifier? can be FALSE. Velocity unit is mm/s. Check HLP? answer to find out if TCV? is supported.

C-877, C-884, C-891: If All axes? = TRUE, then Axis identifier? can be FALSE. Velocity unit is mm/s.

2.11.45. TIO?.vi

Returns the number of digital inputs and outputs available in the controller.

Valid for C-413, C-702, C-843, C-843,PM, C-848, C-866, C-867, C-877, C-880, C-884, C-891, E-517, E-518, E-709, E-727, E-712, E-754, E-761, E-861, E-871, E-873, Hydra, Mercury, Mercury_GCS (but must be present for C-885, C-887, E-816, F-206, M-8X0 also)

C-877, E-709, E-712: Check HLP? answer to find out if TIO? is supported.

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 DIO?vi.

2.11.46. TNR?.vi

Returns the number of recording tables.

Valid for C-413, C-702, C-866, C-867, C-877, C-884, C-887, C-891, E-517, E-518, E-709, E-710, E-712, E-725, E-727, E-753, E-754, E-755, E-761, E-861, E-871, E-873, F-206, M-8X0, Mercury_GCS (but must be present for C-885 also)

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).

2.11.47. TPC?.vi

Returns the number of available piezo channels.

Valid for C-413, C-887, E-517, E-518, E-709, E-710, E-712, E-725, E-727, E-753, E-754, E-755, E-761 (but must be present for Analog systems, C-867, C-877, C-884, C-885, C-891, E-516, E-816, E-861, E-870, E-871, E-873, F-206 and M-8X0 and Mercury_GCS also)

E-517, E-518: Using the Sensor Enable parameter, ID 0x02000000, you can change the controller 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 controller User manual for details.

E-709, E-712, E-725, E-727: Returns all Output Signal Channels (piezo channels + analog output channels).

<table>
<thead>
<tr>
<th>System no.</th>
<th>Number of piezo channels</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td>error in</td>
<td>error out</td>
</tr>
</tbody>
</table>

### 2.11.48. TRO.vi

Enables or disables the TRigger Output mode which was set with CTO. Do not use DIO on output lines for which the trigger output is activated with TRO.

Valid for C-413, C-702, C-843, C-866, C-867, C-877, C-884, E-871, E-873, Mercury_GCS (but must be present for C-885 also)

C-702: TriggerOut to command can be 1 to 8.

C-843: TriggerOut to command can be 1 to 4 (4-axis card) or 2 (2-axis card).

C-866: TriggerOut to command can be 1.

C-867, C-877: Check HLP? answer to find out if TRO is supported. Check controller User Manual for available output trigger numbers.

C-413, C-884, E-871, E-873: Check controller User Manual for available output trigger numbers.

Mercury: TriggerOut to command can be 1 to N (N: number of connected axes, if supported by corresponding Mercury firmware).

Mercury_GCS: Check Mercury User Manual for available output trigger numbers.

### 2.11.49. TRO?.vi

Returns the trigger output mode enable status for the given trigger output lines.

Valid for C-413, C-702, C-843, C-866, C-867, C-877, C-884, E-871, E-873, Mercury, Mercury_GCS (but must be present for C-885 also)

C-702: TriggerOut to query can be 1 to 8.

C-843: TriggerOut to query can be 1 to 4 (4-axis card) or 2 (2-axis card).

C-866: TriggerOut to query can be 1.

C-867, C-877: Check HLP? answer to find out if TRO? is supported. Check C-867 User Manual for available output trigger numbers.

C-413, C-884, E-871, E-873: Check controller User Manual for available output trigger numbers.

Mercury: TriggerOut to query can be 1 to N (N: number of connected axes, if supported by corresponding Mercury firmware).

Mercury_GCS: Check Mercury User Manual for available output trigger numbers.
2.11.50. TVI?.vi

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.

Valid for C-702, C-843, C-843.PM, C-844, C-848, C-865, C-866, C-867, C-877, C-880, C-884, C-891, E-517, E-518, E-710, E-761, E-871, E-873, Hydra, Pollux, Mercury, Mercury_GCS (but must be present for C-413, C-885, C-887, E-709, E-712, E-727, E-754, E-816, E-861, F-206 and M-8X0 also)

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-877, C-884, C-891, E-517, E-518, E-871, E-873, Hydra, Pollux, Mercury_GCS: Invert order must be FALSE. Returns valid characters for axis IDs.

2.11.51. VST?.vi

Returns the names of all stages which can be connected to the controller.

Valid for C-702, C-843, C-843.PM, C-844, C-848, C-865, C-866, C-867, C-877, C-880, C-884, C-887, C-891, E-710, E-761, E-871, E-873, F-206, Hydra, Pollux, M-8X0, Mercury, Mercury_GCS (but must be present for C-885 also)

C-877, C-884, C-891, E-873: Command is available via USB interface only.

C-867, 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).

Mercury_GCS: Command is available via USB interface only. Check HLP? answer to find out if VST? is supported.

2.11.52. WPA.vi

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, E-727: 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 no. 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 (item) and parameter specification.

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.

Valid for C-413, C-867, C-877, C-884, C-887, C-891, E-516, E-517, E-518, E-709, E-710, E-712, E-725, E-727, E-753, E-754, E-755, E-761, E-816, E-861, E-870, E-871, E-873, F-206, Hydra, Pollux, Mercury_GCS, M-8X0 (but must be present for C-885 also)

C-413: 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).

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, C-877, C-891: 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). 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: 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). 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: 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).
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.

E-516: Affected axes and Parameter to save = empty array. 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, E-518: If Affected axes = empty array, all parameters for all axes are saved. Parameter no. format is TRUE (hex). 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, 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 (E-517 only. GCS syntax version, i.e. E-517 or E-516 mode; requires command level 1 ("CCL.vi").

E-709: If Affected axes = empty array, all parameters for all axes are saved. Parameter no. format is TRUE (hex). 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: If Affected axes = empty array, all parameters for all axes are saved. See E-710 GCS DLL Manual for a list of available parameters. Command is available in command level 1 only (see "CCL.vi" and "CCL?.vi"). Parameter no. format is TRUE (hex.).

E-712: If Affected axes = empty array, all parameters for all axes are saved. Parameter no. format is TRUE (hex). 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"). Do not save more than 10 parameters at once.

E-725, E-727: 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, E-727 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: If Affected axes = empty array, all parameters for all axes are saved. Parameter no. format is TRUE (hex). 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-754: If Affected axes = empty array, all parameters for all axes are saved. Parameter no. format is TRUE (hex). 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-754 is powered off or rebooted. Parameters can be changed in volatile memory with SPA, AOS, ATZ, DPO, IFC, RTR, VEL, WOS and WTR.

E-755: If Affected axes = empty array, all parameters for all axes are saved. Parameter no. format is TRUE (hex). 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: If Affected axes = empty array, all parameters for all axes (items) are saved. See E-7XX GCS DLL Manual for a list of available parameters. Parameter no. format is TRUE (hex.). 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-816: Affected axes and Parameter to save = empty array. This command cannot be issued to a slave.

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. If Affected axes = empty array, the currently valid values of all parameters affected by the specified password are saved (see below). Parameter no. format is TRUE (hex).

E-870: If Affected axes = empty array, all parameters for all axes are saved. Parameter no. format is TRUE (hex). 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, E-873: If Affected axes = empty array, all parameters for all axes are saved. Parameter no. format is TRUE (hex). 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.

Hydra, Pollux: 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). 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.

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). 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.
2.12. Support.llb

2.12.1. Analyse input string for terminal.vi

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.

Valid for all systems

2.12.2. Assign booleans from string to axes.vi

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".

Valid for all systems.

2.12.3. Assign DRC values.vi

This VI assigns values (Source ID and Rec. option) from input string to queried Rec. tables. Sub-VI for DRC?.vi.

GCS 2.0: Trigger option is not valid.

Valid for C-413, C-702, C-843, C-866, C-867, C-877, C-884, C-887, C-891, E-517, E-518, E-709, E-710, E-712, E-725, E-727, E-753, E-754, E-755, E-861, E-871, E-873, F-206, M-8X0, Mercury_GCS (but must be present for C-885 also)

2.12.4. Assign DRT values from string.vi

This VI assigns numerical values / strings from input string to queried axes and parameter numbers. Sub-VI for "DRT?.vi".
Valid for C-413, C-702, C-866, C-867, C-877, C-884, C-887, C-891, E-755, E-861, E-871, E-873, Mercury_GCS (but must be present for C-885 also)

2.12.5. **Assign NaN for chosen axes.vi**

This VI returns "NaN" for the given axes subset.

Valid for Analog systems, C-413, C-702, C-843, C-843.PM, C-848, C-865, C-866, C-867, C-877, C-880, C-884, C-887, C-891, E-517, E-518, E-709, E-712, E-725, E-727, E-753, E-754, E-755, E-861, E-871, E-873, F-206, Hydra, Pollux, M-8X0, Mercury, Mercury_GCS (but must be present for C-885 also). To support analog interfacing, VI must be present for E-816 also.

2.12.6. **Assign SPA values from string to axes.vi**

This VI assigns numerical values / strings from input string to queried axes and hex. parameter numbers. Sub-VI for "SPA?.vi".

Valid for C-413, C-702, C-843, C-843.PM, C-844, C-848, C-865, C-866, C-867, C-877, C-880, C-880K005, C-884, C-887, C-891, E-516, E-517, E-518, E-709, E-710, E-712, E-725, E-727, E-753, E-754, E-755, E-761, E-816, E-861, E-870, E-871, E-873, Hydra, Pollux, Mercury, Mercury_GCS (but must be present for C-885 also)

2.12.7. **Assign values from string to axes.vi**

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.

Valid for all systems.
2.12.8. Boolean array calculations.vi

This vi performs a boolean operation of up to three boolean input arrays. The difference to LabVIEW's 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.
Valid for all systems.

2.12.9. Build channel query command substring.vi

This vi builds a query command substring for channel query commands. If "Query 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.g. for systems which accept commands like VMA? without channel IDs). If "With space?" is true, a space character is added between the channel identifiers.
Valid for Analog systems, C-413, C-867, C-877, C-884, C-887, C-891, E-516, E-517, E-518, E-709, E-710, E-712, E-725, E-727, E-753, E-754, E755, E-761, E-816, E-861, E-870, E-871, E-873, F-206, Mercury_GCS, M-8X0 (but must be present for C-885 also).

2.12.10. Build command substring.vi

This VI builds a command substring by combining axis identifier and parameter. If parameter no. 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".
Valid for all systems

2.12.11. Build DIO? query command substring.vi

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.

Valid for C-413, C-702, C-843, C-843.PM, C-848, C-865, C-866, C-867, C-877, C-880, C-884, C-887, C-891, E-517, E-518, E-709, E-712, E-727, E-754, E-761, E-816, E-861, E-871, E-873, F-206, Hydra, Pollux, Mercury, Mercury_GCS, M-8X0 (but must be present for C-885 also)

2.12.12. Build num command substring.vi

This vi builds a command substring by combining Num 1, Space and Num 2. 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".

Valid for all systems.

2.12.13. Build query command substring.vi

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.g. 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.

Valid for all systems

```
```

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".

Valid for C-413, C-702, C-843, C-843.PM, C-844, C-848, C-865, C-866, C-867, C-877, C-880, C-880K005, C-884, C-887, C-891, E-516, E-517, E-517, E-518, E-709, E-712, E-725, E-727, E-753, E-754, E-816, E-861, E-870, E-871, E-873, F-206, Hydra, Pollux, M-8X0, Mercury, Mercury_GCS (but must be present for C-885, E-710, E-755 and E-761 also)

```
2.12.15. Build SPA query command substring.vi
```

This VI builds an SPA? Command substring. Axes and parameters are combined into a substring. Number of rows is size of Axes to query array. Sub-VI for "SPA?.vi" and "SEP?.vi".

Valid for C-413, C-702, C-843, C-843.PM, C-844, C-848, C-865, C-866, C-867, C-877, C-880, C-880K005, C-884, C-887, C-891, E-516, E-517, E-517, E-518, E-709, E-710, E-712, E-725, E-727, E-753, E-754, E-816, E-861, E-870, E-871, E-873, Hydra, Pollux, M-8X0, Mercury, Mercury_GCS (but must be present for C-885, E-710, E-755 and E-761 also)
2.12.16. **Build stringplusnum substring.vi**

This vi builds a command substring by combining up to three strings and two values. Sub-vi for FCG.vi, CTC.vi and DAS.vi

Valid for all systems

![Diagram of Build stringplusnum substring.vi](image)

2.12.17. **Combine axes arrays.vi**

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.

Valid for all systems

![Diagram of Combine axes arrays.vi](image)

2.12.18. **Commanded axes connected?.vi**

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.

Valid for all systems

![Diagram of Commanded axes connected?.vi](image)

2.12.19. **Commanded stage name available?.vi**

This VI checks if Commanded stages is a subset of all available stages and returns Controller error TRUE if this is not the case. Available stages are defined by "VST?.vi".

Valid for C-702, C-843, C-843.PM, C-844, C-848, C-865, C-866, C-867, C-877, C-880, C-884, C-887, C-891, E-710, E-761, E-871, E-873, F-206, Hydra, Pollux, Mercury, M-8X0, Mercury_GCS (but must be present for C-885 also)

![Diagram of Commanded stage name available?.vi](image)
2.12.20. Convert error to warning.vi

If "code" is one of the code numbers given in "Codes", resets error status to "no error" and adds "Warning: " to "source".
Valid for all systems.

<table>
<thead>
<tr>
<th>Error in</th>
<th>Error out</th>
</tr>
</thead>
</table>

2.12.21. Convert num array to string.vi

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.
Valid for all systems.

<table>
<thead>
<tr>
<th>Num. Values</th>
<th>Output string</th>
</tr>
</thead>
<tbody>
<tr>
<td>No. of digits</td>
<td></td>
</tr>
</tbody>
</table>

2.12.22. Convert num value to syntax selection.vi

This VI converts a numerical value to the corresponding GCS syntax version.
Valid for: all systems.

<table>
<thead>
<tr>
<th>GCS syntax version</th>
<th>Syntax</th>
</tr>
</thead>
</table>

2.12.23. Count occurrences in string.vi

This vi counts, how often an expression occurs in a string.
Valid for all systems.

<table>
<thead>
<tr>
<th>Input string</th>
<th>Occurrences</th>
</tr>
</thead>
<tbody>
<tr>
<td>Expression</td>
<td></td>
</tr>
</tbody>
</table>

2.12.24. Cut out additional spaces.vi

Searches for spaces in "String" and cuts them out if the following character is not LF.
Valid for all systems.

<table>
<thead>
<tr>
<th>String</th>
<th>String cut</th>
</tr>
</thead>
</table>

2.12.25. Define axes to command from boolean array.vi

This vi returns only those axes IDs from the "Axes to query" array in the "Axes to command" array which have a boolean value TRUE in the "Command axis?" array or a numerical value of 1 in the "Command axis? (num)" array, depending on
"Command axis? format", and returns all remaining axes in the "Remaining axes" array.

Valid for all systems

2.12.26. GCSTranslateError.vi

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".

Valid for all systems

2.12.27. General wait for movement to stop.vi

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.

Valid for all systems.

Analog: If VOL was commanded or velocity control mode (VCO) is OFF, Add. wait only? must be TRUE, otherwise FALSE.

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.
2.12.28. Get all axes.vi

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.
Valid for all systems.

System no. --------- Conn. Axes

2.12.29. Get arrays without blanks.vi

Returns the string array and related values and boolean arrays without white space string fields.
Valid for all systems

String array in ------- String array out
<table>
<thead>
<tr>
<th>Values in</th>
<th>Values out</th>
</tr>
</thead>
<tbody>
<tr>
<td>Booleans in</td>
<td>Array size out</td>
</tr>
<tr>
<td></td>
<td>Booleans cut</td>
</tr>
</tbody>
</table>

2.12.30. Get lines and values from string.vi

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".
Valid for all systems

Array size ------- Numerical values
<table>
<thead>
<tr>
<th>lines, values</th>
<th>Strings</th>
</tr>
</thead>
<tbody>
<tr>
<td>Item name</td>
<td></td>
</tr>
</tbody>
</table>

2.12.31. Get lines from string.vi

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" and "Define macro contents (with or without) delay.vi".
Valid for all systems

Cut spaces? ------- Strings
<table>
<thead>
<tr>
<th>Array size</th>
<th>Array size out</th>
</tr>
</thead>
<tbody>
<tr>
<td>Input string</td>
<td></td>
</tr>
</tbody>
</table>

2.12.32. Get string array size without blanks.vi

This VI returns the size of a string array without counting white space strings.
Valid for all systems

String array --------- Corrected array size
<p>| |</p>
<table>
<thead>
<tr>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>6 -&gt; 2</td>
</tr>
</tbody>
</table>
2.12.33. **Get total number of commanded axes.vi**

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.

Valid for all systems

2.12.34. **How often does string contain regular expression?.vi**

This VI returns a count of the occurances of a regular expression in a string.

Valid for all systems

2.12.35. **Increase array size.vi**

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.

Valid for all systems

2.12.36. **Is command present in HLP answer?.vi**

This VI checks whether the specified commands are present in the HLP? answer. If "Check HLP?" is TRUE, HLP?.vi will be called to get HLP? information. If "Check HLP?" is FALSE, a valid HLP? answer has to be provided in "HLP answer".

Valid for all systems

2.12.37. **Return single characters from string.vi**

Get single characters from input string.

Valid for all systems
2.12.38. Return space.vi

This VI returns a space character in "String out" if "With space?" is TRUE or GCS syntax version is higher than 1.0.
Valid for all systems.

2.12.39. Round with options.vi

Rounds "Numeric in" and "Num array" according to "No. of digits to round to" and "Round mode selection".
Valid for all systems

2.12.40. Select axis.vi

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.
Valid for all systems

2.12.41. Select values for chosen axes.vi

This VI returns only values for the given axes subset.
Valid for all systems.

2.12.42. Select with boolean array input.vi

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.
Valid for all systems

T string  String array out
F string
T/F
Size

2.12.43. Selection to String array.vi

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).
Valid for all systems

Selection array  String array
String input

2.12.44. Set RON and return RON status.vi

Sets RON mode ON or OFF and returns which of the connected axes have RON mode ON and which have RON mode OFF.

If reference mode is OFF, no referencing is required for the axis. Only relative moves can be commanded (using "MVR.vi"), unless the actual position is set with POS.vi. Afterwards, relative and absolute moves can be commanded.

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

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

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

Valid for C-413, C-702, C-843, C-843.PM, C-848, C-865, C-866, C-867, C-877, C-880, C-884, C-885, C-891, E-712, E-861, E-871, E-873, Hydra, Pollux, Mercury™, Mercury_GCS

2.12.45. String with ASCII code conversion.vi

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.
2.12.46. **Subtract axes array subset from axes array.vi**

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".

Valid for all systems

![Axes diagram](image)

2.12.47. **Unbunde/bundle interface clusters for PI Terminal.vi**

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".

Valid for all systems

![Interface configuration diagram](image)

2.12.48. **Wait for axes to stop.vi**

This VI waits for the specified axes to stop using #5 polling. It also stops if a communication error occurred, 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.

Valid for C-413, C-702, C-843, C-843.PM, C-844, C-848, C-865, C-867, C-877, C-880, C-884, C-885, C-887, C-891, E-517, E-518, E-709, E-712, E-725, E-727, E-753, E-755, E-861, E-870, E-871, E-873, Hydra, Pollux, Mercury, Mercury_GCS (but must be present in Support.llb for all other systems also)

C-880: With status bit polling? = TRUE

All other systems: With status bit polling? = FALSE

![Stop VI diagram](image)
2.12.49. Wait for controller ready.vi

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.

Valid for C-413, C-702, C-843, C-843.PM, C-848, C-865, C-866, C-867, C-877, C-880, C-884, C-885, C-887, C-891, E-517, E-518, E-709, E-710, E-712, E-727, E-754, E-755, E-761, E-861, E-870, E-871, E-873, F-206, Hydra, Pollux, Mercury, Mercury_GCS, M-8X0 (but must be present for E-816 also)
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.

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: \textbf{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: \textbf{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: \textbf{Interface} = RS232, Input and output HW handshake must be TRUE. DaisyChain: \textbf{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: \textbf{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: \textbf{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: \textbf{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: \textbf{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, M-8X0: For GCS syntax version = GCS 1.0 (older firmware or compatibility mode, can be set with CSV.vi), \textbf{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), \textbf{Interface} = RS232 or TCP/IP. RS232: Input and output handshake settings must be FALSE. Syntax: GCS 2.0; Term char = LF

Hydra: \textbf{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: \textbf{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: \textbf{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: \textbf{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: \texttt{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: \texttt{cmd})" with \texttt{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. Mercury_GCS_Simple_Test.vi

This simple test VI is a stand-alone sample application. Use the Help→Show Context Help menu sequence in the NI LabVIEW environment to display the Context Help window with the VI and control/indicator descriptions.

Specify

- System number (= 1 in a one-system configuration),
- Interface (RS232, RS232 DaisyChain, USB or USB DaisyChain),
- For Interface = RS232 or RS232 DaisyChain: appropriate port number and RS232 baud rate, or
- For Interface = USB or USB DaisyChain: appropriate Serial Number of the controller to connect to or if you want to search for the controller, and
- For Interface = RS232 DaisyChain or USB DaisyChain: select if you want to register the DaisyChain if it is not already open only (FALSE) or in any case (TRUE), if you want to call a dialog to select the connected Mercury controller from a list (TRUE) or if you know the Device ID of the controller to connect (FALSE, Device ID must match the device ID of the controller in the DaisyChain network). If one device of a DaisyChain is already connected with the PI LabVIEW driver set, do not switch Register DC to TRUE; and
- Timeout value (in milliseconds).

Then start the VI. The VI will open the chosen communication port and query for the identification string of the connected Mercury controller. The diagram shows how to combine the driver and support VIs for these tasks.
3.3. **Mercury_GCS_Configuration_Setup.vi**

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

After the successful running of this VI, all command VIs are ready to use. Specify the correct parameters first:

- **System No.:** 1 in a one-system-only configuration.
- **Interface (RS232, RS232 DaisyChain, USB or USB DaisyChain)**
- **For Interface = RS232 or RS232 DaisyChain:** RS-232 settings: Choose the port number the Mercury is connected to and an appropriate baud rate.
- **For Interface = USB or USB DaisyChain:** appropriate Serial Number of the controller to connect to or if you want to search for the controller.
- **For Interface = RS232 DaisyChain or USB DaisyChain:** select if you want to register the DaisyChain if it is not already open only (FALSE) or in any case (TRUE), if you want to call a dialog to select the connected Mercury controller from a list (TRUE) or if you know the Device ID of the controller to connect (FALSE, Device ID must match the device ID of the controller in the DaisyChain network). If one device of a DaisyChain is already connected with the PI LabVIEW driver set, do not switch Register DC to TRUE.
- **Timeout:** choose an appropriate timeout value here (default is 5000 ms).
- **Reference mode** (only if needed, see description of RON.vi for details and warnings). You can also select here whether axes should be referenced always (Check FRF? is FALSE) or only if needed (Check FRF? is TRUE and FRF? returns FALSE).
- Whether or not axis is connected and can be moved (depends on mechanical setup).
- If axis is connected and can be moved, whether or not servo is to be switched on. Note that servo must be ON if referencing is to be done by a reference move, i.e. if the reference mode is ON, and to move an axis in closed-loop.
- If axis is connected, can be moved and servo is ON, which limit switch to use for referencing an axis that is to be referenced but has no reference switch. See User Manual for details and specialities.
- If axis is connected, can be moved, servo is ON and referencing was successful, whether or not axis is to be moved to the middle position of its travel range.

Then start the VI.

“Mercury_GCS_Configuration_Setup.vi” performs the following initialization tasks:

1. Runs “PI Open Interface of one system.vi” to open a connection to the Mercury controller.
2. Runs “*IDN?.vi” to query for the controller identification string.
3. Defines the selected system to be "Mercury_GCS".
4. Runs “Define connected axes.vi” with Read from controller = TRUE and Invert Order = FALSE.
5. Runs “CST?.vi” to query for the connected stage (use PIMikroMove to select or configure stages connected to Mercury). If Connected? is FALSE (because no stage is connected to the controller or axis cannot be moved), proceeds with step 18, otherwise step 6.

6. Runs “SVO.vi” to switch servo on (if Switch servo on? is TRUE)

7. Runs “SVO?.vi” to display servo status of axis

8. Sets the reference mode for the axis, if specified (Reference mode tab).

9. Reads whether axis has reference mode OFF.

10. If not, determines if axis has a reference switch (TRS?).

11. If yes, moves axis to the reference position (FRF) and queries if referencing was successful (FRF?). See User Manual for details and specialities.

12. If axis has no reference switch but has reference mode ON, reads if axis has a limit switch (LIM?).

13. If yes, moves axis to the positive or negative limit position (FPL/FNL) as specified and queries if referencing was successful (FRF?). See User Manual for details and specialities.

14. Reads the position range (TMN?, TMX?).

15. If Move to middle? is TRUE, moves axis to the middle position of its range (MOV) if it has been referenced before by a reference move (FRF/FPL/FNL).

16. Waits for motion to stop.

17. Runs “POS?.vi” to query for the position.

18. Runs “ERR?.vi” to query the controller for its error status.

19. Runs “GCSTranslateError.vi” to append the error message which corresponds with a GCS error number returned by “ERR?.vi” to Source from Error out.

20. If axis has referencing mode OFF, referencing was not successful, or axis was designated not to be moved, it will report NaN as position value. An axis with RON OFF can only be moved relative until its position is set with “POS.vi”. See description of "RON.vi" for details. If Axis was designated not to be moved during
runtime of Mercury_GCS_Configuration_Setup it cannot be moved until referenced separately.

As the initialization is a complex procedure which uses a large number of sub-VIs, Mercury_GCS_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. **Mercury_GCS_Sample_Application_1.vi**

This VI demonstrates how to define a Mercury (with GCS firmware) configuration with a connected axis. In this example the Mercury is connected through COM port 1 with baudrate 38400. The VI runs Mercury_GCS_Configuration_Setup.vi with these specified settings and reads the position of all axes. The diagram shows how to combine the driver VIs for these tasks.

**Warning:** Before running this VI, modify “Interface” and “RS232 settings” (or add “USB settings” if Interface = USB was selected) to reflect your configuration. The predefined settings provided with the VI probably do not match your system.
3.5. **Mercury_GCS_Sample_Application_1_with_Move.vi**

This VI demonstrates how to define a Mercury (with GCS firmware) configuration with a connected axis. In this example the Mercury is connected through COM port 1 with baudrate 38400. The VI runs Mercury_GCS_Configuration_Setup.vi with these specified settings, moves the first connected axis to the absolute position 0.01 mm, waits until motion stops and reads the position of all axes. The diagram shows how to combine the driver VIs for these tasks.

**Warning:** Before running this VI, modify “Interface” and “RS232 settings” (or add “USB settings” if Interface = USB was selected) to reflect your configuration. The predefined settings provided with the VI probably do not match your system.
3.6. **Mercury_GCS_Sample_Application_2a.vi**

This VI demonstrates how to define a Mercury configuration with two Mercury controllers (with GCS firmware) directly connected to the host PC (no DaisyChain) and with stages connected. In this example, the first Mercury is connected through COM port 1 with baudrate 38400 and the second through USB with S/N 1000. The VI runs Mercury_GCS_Configuration_Setup.vi for both controllers with these pre-defined settings and reads the position of the axes. The diagram shows how to combine the driver VIs for these tasks.

**Warning:** This VI has Interface settings for each controller that probably do not match your current configuration. Please modify the "Interface", "RS232 settings" or "USB settings" in the diagram before running the VI.
3.7. **Mercury_GCS_Sample_Application_DaisyChain.vi**

This VI demonstrates how to define a Mercury DaisyChain configuration with two Mercury controllers (with GCS firmware) in the chain and stages connected. In this example, the first Mercury is connected through COM port 1 with baudrate 38400. The VI runs Mercury_GCS_Configuration_Setup.vi for both controllers with pre-defined settings and reads the position of the axes. The diagram shows how to combine the driver VIs for these tasks.

**Warning:** This VI has "Interface" settings for the first controller in the chain that probably do not match your current configuration. Please modify the "Interface", "RS232 settings" or "USB settings" in the diagram before running the VI. All controllers in the chain must use the same baudrate (which can be set with the DIP switches on the Mercury front panel).
3.8. **Mercury_GCS_Sample_Application_USB_DaisyChain.vi**

This VI demonstrates how to define a Mercury DaisyChain configuration with two Mercury controllers (with GCS firmware) in the chain and stages connected. In this example, the first Mercury is connected via USB DaisyChain connection. The VI runs Mercury_GCS_Configuration_Setup.vi for both controllers with pre-defined settings and reads the position of the axes. The diagram shows how to combine the driver VIs for these tasks.

**Warning:** This VI has “Interface” settings for the first controller in the chain that probably do not match your current configuration. Please modify the "Interface", "RS232 settings" or "USB settings" in the diagram before running the VI. All controllers in the chain must use the same baudrate (which can be set with the DIP switches on the Mercury front panel).
3.9. Joystick Operation.vi

This VI can be used to control 2 closed-loop axes (which can belong to one or two connected systems) with a standard 2-button, 2-axis joystick connected to the game port of the host computer. The absolute value of the joystick position is converted into velocity values for the two stages connected to the system being commanded. Two velocity levels for each axis can be specified, e.g. one for fast, rough positioning and one for slow, fine positioning. Joystick button 2 switches between these levels. The sign of the joystick position determines whether the move command issued contains the positive or negative travel limit (read automatically if Read travel range from controller? is TRUE) of the corresponding axis. When the joystick is “in the middle position", the velocity of the corresponding axis is set to zero.

Dead band is the maximum size of the scaled joystick position value that does not result in any motion.

Resolution factor determines the joystick resolution.

No. of digits is the number of digits after the decimal point in the position values that will be sent by the MOV command.

If joystick button 1 is pressed, a stop command (STP or #24, depending on Use 24 for stop, is sent to the controller. The diagram shows how to combine the driver and support VIs for these tasks.

If only one motion axis is to be controlled, Y Axis must be identical to X Axis, System no. 2 must be identical to System no. 1, and Y axis of joystick is to be used for control of the motion axis.

Important: Install joystick driver and calibrate joystick in the Windows Control Panel before running this VI.

Valid for: C-413, C-702, C-843, C-843.PM, C-844, C-848, C-865, C-866, C-867, C-880, C-884, E-517, E-709, E-710, E-712, E-725, E-753, E-
Input

System number (1), System no. 2 (1), Joystick ID (0), X axis (empty string), Y axis (empty string), Resolution factor X (150), Resolution factor Y (150), Dead band X (2), Dead band Y (2), No. of digits (2), Use #24 for stop (TRUE), Read from controller (T), Maximum velocity values (2, 20, 0.5, 15), Travel range (0, 10, 0, 5), Error in (no error)

C-413: Use #24 for stop = TRUE, Read from controller = TRUE.
Default position unit is mm, default velocity unit is mm/s.
Run “C413_Configuration_Setup.vi” prior to running this VI.

C-702: Use #24 for stop = TRUE, Read from controller = TRUE
Run “C702_Configuration_Setup.vi” prior to running this VI.

C-843: Use #24 for stop = TRUE, Read from controller = TRUE;
Run “C843_Configuration_Setup.vi” prior to running this VI.

C-843.PM: Use #24 for stop = TRUE, Read from controller = TRUE;
Run “C843_PM_Configuration_Setup.vi” prior to running this VI.

C-844: Use #24 for stop = TRUE, Read from controller = TRUE
Run “C844_Configuration_Setup.vi” prior to running this VI.

C-848: Use #24 for stop = TRUE, Read from controller = TRUE
Run “C848_Configuration_Setup.vi” prior to running this VI.

C-865: Use #24 for stop = TRUE, Read from controller = TRUE;
Run “C865_Configuration_Setup.vi” prior to running this VI.

C-866: Use #24 for stop = TRUE, Read from controller = TRUE;
Run “C866_Configuration_Setup.vi” prior to running this VI.

C-867: Use #24 for stop = TRUE, Read from controller = TRUE;
Run “C867_Multiaxis_Configuration_Setup.vi” prior to running this VI.

C-880: Use #24 for stop = TRUE, Read from controller = TRUE
Run “C880_Configuration_Setup.vi” prior to running this VI.

C-884: Use #24 for stop = TRUE, Read from controller = TRUE;
Run “C884_Configuration_Setup.vi” prior to running this VI.

E-517: Use #24 for stop = TRUE, Read from controller = TRUE.
Default position unit is μm, default velocity unit is μm/s.
Run “E517_Configuration_Setup.vi” prior to running this VI.

E-709: Use #24 for stop = TRUE, Read from controller = TRUE.
Default position unit is μm, default velocity unit is μm/s.
Run “E709_Configuration_Setup.vi” prior to running this VI.

E-710: Use #24 for stop = FALSE, Read from controller = TRUE.
Default position unit is μm, default velocity unit is μm/ms. Joystick button 1 is not supported (controller does not support #24 or STP).
Run “E710_Configuration_Setup.vi” prior to running this VI.

E-712: Use #24 for stop = TRUE, Read from controller = TRUE.
Default position unit is μm, default velocity unit is μm/s.
Run “E712_Configuration_Setup.vi” prior to running this VI.

E-725: Use #24 for stop = TRUE, Read from controller = TRUE.
Default position unit is μm, default velocity unit is μm/s.
Run “E725_Configuration_Setup.vi” prior to running this VI.

E-753: Use #24 for stop = TRUE, Read from controller = TRUE.
Default position unit is μm, default velocity unit is μm/s.
Run “E753_Configuration_Setup.vi” prior to running this VI.
E-755: Use #24 for stop = TRUE, Read from controller = TRUE. Default position unit is µm, default velocity unit is µm/s. Run “E755_Configuration_Setup.vi” prior to running this VI. VI does not work with E-755.101.

E-761: Use #24 for stop = TRUE, Read from controller = TRUE. Default position unit is µm, default velocity unit is µm/ms. Run “E761_Configuration_Setup.vi” prior to running this VI.

E-861: Use #24 for stop = TRUE, Read from controller = TRUE; Run “E861_Configuration_Setup.vi” prior to running this VI.

Hydra, Pollux: Use #24 for stop = TRUE, Read from controller = TRUE; Run “HydraPollux_Configuration_Setup.vi” prior to running this VI.

Mercury_GCS: Use #24 for stop = TRUE, Read from controller = TRUE; Run “Mercury_GCS_Configuration_Setup.vi” prior to running this VI.

Use the Help→Show Context Help menu sequence in the LabVIEW environment to display the Context Help window with the VI and control/indicator descriptions.
3.10. **Show_Save_Load_XY_Data.vi**

This VI displays XY data in an XY Graph. If 1D Array X is an empty array, N X values are calculated with \( N = \text{Size of 1D Array Y} \) 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, 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:
3.11. **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.

![Diagram of 1D Scan](image)

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? = TRUE  
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).
### 4. 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.

<table>
<thead>
<tr>
<th>Error Code</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>100</td>
<td>PI LabVIEW driver reports error. See source control for details.</td>
</tr>
<tr>
<td>0</td>
<td>No error</td>
</tr>
<tr>
<td>1</td>
<td>Parameter syntax error</td>
</tr>
<tr>
<td>2</td>
<td>Unknown command</td>
</tr>
<tr>
<td>3</td>
<td>Command length out of limits or command buffer overrun</td>
</tr>
<tr>
<td>4</td>
<td>Error while scanning</td>
</tr>
<tr>
<td>5</td>
<td>Unallowable move attempted on unreferenced axis, or move attempted with servo off</td>
</tr>
<tr>
<td>6</td>
<td>Parameter for SGA not valid</td>
</tr>
<tr>
<td>7</td>
<td>Position out of limits</td>
</tr>
<tr>
<td>8</td>
<td>Velocity out of limits</td>
</tr>
<tr>
<td>9</td>
<td>Attempt to set pivot point while U,V and W not all 0</td>
</tr>
<tr>
<td>10</td>
<td>Controller was stopped by command</td>
</tr>
<tr>
<td>11</td>
<td>Parameter for SST or for one of the embedded scan algorithms out of range</td>
</tr>
<tr>
<td>12</td>
<td>Invalid axis combination for fast scan</td>
</tr>
<tr>
<td>13</td>
<td>Parameter for NAV out of range</td>
</tr>
<tr>
<td>14</td>
<td>Invalid analog channel</td>
</tr>
<tr>
<td>15</td>
<td>Invalid axis identifier</td>
</tr>
<tr>
<td>16</td>
<td>Invalid stage name</td>
</tr>
<tr>
<td>17</td>
<td>Parameter out of range</td>
</tr>
<tr>
<td>18</td>
<td>Invalid macro name</td>
</tr>
<tr>
<td>19</td>
<td>Error while recording macro</td>
</tr>
<tr>
<td>20</td>
<td>Macro not found</td>
</tr>
<tr>
<td>21</td>
<td>Axis has no brake</td>
</tr>
<tr>
<td>22</td>
<td>Axis identifier specified more than once</td>
</tr>
<tr>
<td>23</td>
<td>Illegal axis</td>
</tr>
<tr>
<td>24</td>
<td>Incorrect number of parameters</td>
</tr>
<tr>
<td>25</td>
<td>Invalid floating point number</td>
</tr>
<tr>
<td>26</td>
<td>Parameter missing</td>
</tr>
<tr>
<td>27</td>
<td>Soft limit out of range</td>
</tr>
<tr>
<td>28</td>
<td>No manual pad found</td>
</tr>
<tr>
<td>29</td>
<td>No more step-response values</td>
</tr>
<tr>
<td>30</td>
<td>No step-response values recorded</td>
</tr>
<tr>
<td>31</td>
<td>Axis has no reference sensor</td>
</tr>
<tr>
<td>Number</td>
<td>Error Message</td>
</tr>
<tr>
<td>--------</td>
<td>---------------</td>
</tr>
<tr>
<td>32</td>
<td>Axis has no limit switch</td>
</tr>
<tr>
<td>33</td>
<td>No relay card installed</td>
</tr>
<tr>
<td>34</td>
<td>Command not allowed for selected stage(s)</td>
</tr>
<tr>
<td>35</td>
<td>No digital input installed</td>
</tr>
<tr>
<td>36</td>
<td>No digital output configured</td>
</tr>
<tr>
<td>37</td>
<td>No more MCM responses</td>
</tr>
<tr>
<td>38</td>
<td>No MCM values recorded</td>
</tr>
<tr>
<td>39</td>
<td>Controller number invalid</td>
</tr>
<tr>
<td>40</td>
<td>No joystick configured</td>
</tr>
<tr>
<td>41</td>
<td>Invalid axis for electronic gearing, axis can not be slave</td>
</tr>
<tr>
<td>42</td>
<td>Position of slave axis is out of range</td>
</tr>
<tr>
<td>43</td>
<td>Slave axis cannot be commanded directly when electronic gearing is enabled</td>
</tr>
<tr>
<td>44</td>
<td>Calibration of joystick failed</td>
</tr>
<tr>
<td>45</td>
<td>Referencing failed</td>
</tr>
<tr>
<td>46</td>
<td>OPM (Optical Power Meter) missing</td>
</tr>
<tr>
<td>47</td>
<td>OPM (Optical Power Meter) not initialized or cannot be initialized</td>
</tr>
<tr>
<td>48</td>
<td>OPM (Optical Power Meter) Communication Error</td>
</tr>
<tr>
<td>49</td>
<td>Move to limit switch failed</td>
</tr>
<tr>
<td>50</td>
<td>Attempt to reference axis with referencing disabled</td>
</tr>
<tr>
<td>51</td>
<td>Selected axis is controlled by joystick</td>
</tr>
<tr>
<td>52</td>
<td>Controller detected communication error</td>
</tr>
<tr>
<td>53</td>
<td>MOV! motion still in progress</td>
</tr>
<tr>
<td>54</td>
<td>Unknown parameter</td>
</tr>
<tr>
<td>55</td>
<td>No commands were recorded with REP</td>
</tr>
<tr>
<td>56</td>
<td>Password invalid</td>
</tr>
<tr>
<td>57</td>
<td>Data Record Table does not exist</td>
</tr>
<tr>
<td>58</td>
<td>Source does not exist; number too low or too high</td>
</tr>
<tr>
<td>59</td>
<td>Source Record Table number too low or too high</td>
</tr>
<tr>
<td>60</td>
<td>Protected Param: current Command Level (CCL) too low</td>
</tr>
<tr>
<td>61</td>
<td>Command execution not possible while Autozero is running</td>
</tr>
<tr>
<td>62</td>
<td>Autozero requires at least one linear axis</td>
</tr>
<tr>
<td>63</td>
<td>Initialization still in progress</td>
</tr>
<tr>
<td>64</td>
<td>Parameter is read-only</td>
</tr>
<tr>
<td>65</td>
<td>Parameter not found in non-volatile memory</td>
</tr>
<tr>
<td>66</td>
<td>Voltage out of limits</td>
</tr>
<tr>
<td>67</td>
<td>Not enough memory available for requested wave curve</td>
</tr>
<tr>
<td>68</td>
<td>Not enough memory available for DDL table; DDL can not be started</td>
</tr>
<tr>
<td>69</td>
<td>Time delay larger than DDL table; DDL can not be started</td>
</tr>
<tr>
<td>70</td>
<td>The requested arrays have different lengths; query them separately</td>
</tr>
<tr>
<td>71</td>
<td>Attempt to restart the generator while it is running in single step mode</td>
</tr>
</tbody>
</table>
Motion commands and wave generator activation are not allowed when analog target is active

Motion commands are not allowed when wave generator output is active; use WGO to disable generator output

No sensor channel or no piezo channel connected to selected axis (sensor and piezo matrix)

Generator started (WGO) without having selected a wave table (WSL).

Interface buffer did overrun and command couldn't be received correctly

Data Record Table does not hold enough recorded data

Data Record Table is not configured for recording

Open-loop commands (SVA, SVR) are not allowed when servo is on

Hardware error affecting RAM

Not macro command

Macro counter out of range

Joystick is active

Motor is off

Macro-only command

Invalid joystick axis

Joystick unknown

Move without referenced stage

Command not allowed in current motion mode

No tracing possible while digital IOs are used on this HW revision. Reconnect to switch operation mode.

Move not possible, would cause collision

Stage is not capable of following the master. Check the gear ratio (SRA).

This command is not allowed while the affected axis or its master is in motion.

Servo cannot be switched on when open-loop joystick control is enabled.

This parameter cannot be changed in current servo mode.

Unknown stage name

Invalid length of value (too much characters)

PI LabVIEW driver reports error. See source control for details.

No stage connected to axis

File with axis parameters not found

Invalid axis parameter file

Backup file with axis parameters not found

PI internal error code 204

SMO with servo on

uudecode: incomplete header

uudecode: nothing to decode

uudecode: illegal UUE format

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
Servo loop was unstable due to wrong parameter setting and switched off to avoid damage.

digital connection to external spi slave device is lost

A command refers to a coordinate system that does not exist

A command refers to a coordinate system that has no parent node

Attempt to delete or change a coordinate system that is in use

Definition of a coordinate system is cyclic

Coordinate system cannot be defined as long as Hexapod is in motion

Coordinate system type is not intended for manual activation

A coordinate system cannot be linked to itself

Coordinate system definition is erroneous or not complete (replace or delete it)

The coordinate systems are not part of the same chain

Unused coordinate system must be deleted before new coordinate system can be stored

With this coordinate system type SPI usage is not supported

Soft limits invalid due to changes in coordinate system

Coordinate system is write protected

Coordinate system cannot be changed because its content is loaded from a configuration file

Coordinate system may not be linked

A KSB-type coordinate system can only be rotated by multiples of 90 degrees

This query is not supported for this coordinate system type

This combination of work and tool coordinate systems does not exist

The combination must consist of one work and one tool coordinate system

This coordinate system type does not exist

BasMac: unknown controller error

No coordinate system of this type is activated

Name of coordinate system is invalid

File with stored CS systems is missing or erroneous

File with leveling CS is missing or erroneous

Not enough memory

Hardware voltage error

Hardware temperature out of range

Position error of any axis in the system is too high

Maximum value of input signal has been exceeded

Value is not integer

Command not allowed because fieldbus target is active

Error in fieldbus-interface occurred

Too many nested macros

Macro already defined
<table>
<thead>
<tr>
<th>Code</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>1002</td>
<td>Macro recording not activated</td>
</tr>
<tr>
<td>1003</td>
<td>Invalid parameter for MAC</td>
</tr>
<tr>
<td>1004</td>
<td>Deleting macro failed</td>
</tr>
<tr>
<td>1005</td>
<td>Controller is busy with some lengthy operation (e.g. reference move, fast scan algorithm)</td>
</tr>
<tr>
<td>1006</td>
<td>Invalid identifier (invalid special characters, ...)</td>
</tr>
<tr>
<td>1007</td>
<td>Variable or argument not defined</td>
</tr>
<tr>
<td>1008</td>
<td>Controller is (already) running a macro</td>
</tr>
<tr>
<td>1009</td>
<td>Invalid or missing operator for condition. Check necessary spaces around operator.</td>
</tr>
<tr>
<td>1010</td>
<td>No answer was received while executing WAC/MEX/JRC/...</td>
</tr>
<tr>
<td>1011</td>
<td>Command not valid during macro execution</td>
</tr>
<tr>
<td>1024</td>
<td>Motion error: position error too large, servo is switched off automatically</td>
</tr>
<tr>
<td>1063</td>
<td>User Profile Mode: Command is not allowed, check for required preparatory commands</td>
</tr>
<tr>
<td>1064</td>
<td>User Profile Mode: First target position in User Profile is too far from current position</td>
</tr>
<tr>
<td>1065</td>
<td>Controller is (already) in User Profile Mode</td>
</tr>
<tr>
<td>1066</td>
<td>User Profile Mode: Block or Data Set index out of allowed range</td>
</tr>
<tr>
<td>1071</td>
<td>User Profile Mode: Out of memory</td>
</tr>
<tr>
<td>1072</td>
<td>User Profile Mode: Cluster is not assigned to this axis</td>
</tr>
<tr>
<td>1073</td>
<td>Unknown cluster identifier</td>
</tr>
<tr>
<td>1090</td>
<td>There are too many open tcpip connections</td>
</tr>
<tr>
<td>2000</td>
<td>Controller already has a serial number</td>
</tr>
<tr>
<td>4000</td>
<td>Sector erase failed</td>
</tr>
<tr>
<td>4001</td>
<td>Flash program failed</td>
</tr>
<tr>
<td>4002</td>
<td>Flash read failed</td>
</tr>
<tr>
<td>4003</td>
<td>HW match code missing/invalid</td>
</tr>
<tr>
<td>4004</td>
<td>FW match code missing/invalid</td>
</tr>
<tr>
<td>4005</td>
<td>HW version missing/invalid</td>
</tr>
<tr>
<td>4006</td>
<td>FW version missing/invalid</td>
</tr>
<tr>
<td>4007</td>
<td>FW update failed</td>
</tr>
<tr>
<td>4008</td>
<td>FW Parameter CRC wrong</td>
</tr>
<tr>
<td>4009</td>
<td>FW CRC wrong</td>
</tr>
<tr>
<td>5000</td>
<td>PicoCompensation scan data is not valid</td>
</tr>
<tr>
<td>5001</td>
<td>PicoCompensation is running, some actions can not be executed during scanning/recording</td>
</tr>
<tr>
<td>5002</td>
<td>Given axis can not be defined as PPC axis</td>
</tr>
<tr>
<td>5003</td>
<td>Defined scan area is larger than the travel range</td>
</tr>
<tr>
<td>5004</td>
<td>Given PicoCompensation type is not defined</td>
</tr>
<tr>
<td>5005</td>
<td>PicoCompensation parameter error</td>
</tr>
<tr>
<td>Code</td>
<td>Description</td>
</tr>
<tr>
<td>-------</td>
<td>-----------------------------------------------------------------------------</td>
</tr>
<tr>
<td>5006</td>
<td>PicoCompensation table is larger than maximum table length</td>
</tr>
<tr>
<td>5100</td>
<td>Common error in Nexline firmware module</td>
</tr>
<tr>
<td>5101</td>
<td>Output channel for Nexline cannot be redefined for other usage</td>
</tr>
<tr>
<td>5102</td>
<td>Memory for Nexline signals is too small</td>
</tr>
<tr>
<td>5103</td>
<td>RNP can not be executed if axis is in closed loop</td>
</tr>
<tr>
<td>5104</td>
<td>relax procedure (RNP) needed</td>
</tr>
<tr>
<td>5200</td>
<td>Axis must be configured for this action</td>
</tr>
<tr>
<td>6000</td>
<td>Invalid preset value of absolute sensor</td>
</tr>
<tr>
<td>6001</td>
<td>Error while writing to sensor</td>
</tr>
<tr>
<td>6002</td>
<td>Error while reading from sensor</td>
</tr>
<tr>
<td>6003</td>
<td>Checksum error of absolute sensor</td>
</tr>
<tr>
<td>6004</td>
<td>General error of absolute sensor</td>
</tr>
<tr>
<td>6005</td>
<td>Overflow of absolute sensor position</td>
</tr>
<tr>
<td>0</td>
<td>No error occurred during function call</td>
</tr>
<tr>
<td>-1</td>
<td>Error during com operation (could not be specified)</td>
</tr>
<tr>
<td>-2</td>
<td>Error while sending data</td>
</tr>
<tr>
<td>-3</td>
<td>Error while receiving data</td>
</tr>
<tr>
<td>-4</td>
<td>Not connected (no port with given ID open)</td>
</tr>
<tr>
<td>-5</td>
<td>Buffer overflow</td>
</tr>
<tr>
<td>-6</td>
<td>Error while opening port</td>
</tr>
<tr>
<td>-7</td>
<td>Timeout error</td>
</tr>
<tr>
<td>-8</td>
<td>There are more lines waiting in buffer</td>
</tr>
<tr>
<td>-9</td>
<td>There is no interface or DLL handle with the given ID</td>
</tr>
<tr>
<td>-10</td>
<td>Event/message for notification could not be opened</td>
</tr>
<tr>
<td>-11</td>
<td>Function not supported by this interface type</td>
</tr>
<tr>
<td>-12</td>
<td>Error while sending &quot;echoed&quot; data</td>
</tr>
<tr>
<td>-13</td>
<td>IEEE488: System error</td>
</tr>
<tr>
<td>-14</td>
<td>IEEE488: Function requires GPIB board to be CIC</td>
</tr>
<tr>
<td>-15</td>
<td>IEEE488: Write function detected no listeners</td>
</tr>
<tr>
<td>-16</td>
<td>IEEE488: Interface board not addressed correctly</td>
</tr>
<tr>
<td>-17</td>
<td>IEEE488: Invalid argument to function call</td>
</tr>
<tr>
<td>-18</td>
<td>IEEE488: Function requires GPIB board to be SAC</td>
</tr>
<tr>
<td>-19</td>
<td>IEEE488: I/O operation aborted</td>
</tr>
<tr>
<td>-20</td>
<td>IEEE488: Interface board not found</td>
</tr>
<tr>
<td>-21</td>
<td>IEEE488: Error performing DMA</td>
</tr>
<tr>
<td>-22</td>
<td>IEEE488: I/O operation started before previous operation completed</td>
</tr>
<tr>
<td>-23</td>
<td>IEEE488: No capability for intended operation</td>
</tr>
<tr>
<td>-24</td>
<td>IEEE488: File system operation error</td>
</tr>
<tr>
<td>-25</td>
<td>IEEE488: Command error during device call</td>
</tr>
<tr>
<td>-26</td>
<td>IEEE488: Serial poll-status byte lost</td>
</tr>
</tbody>
</table>
-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
-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 transferring 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
End of document