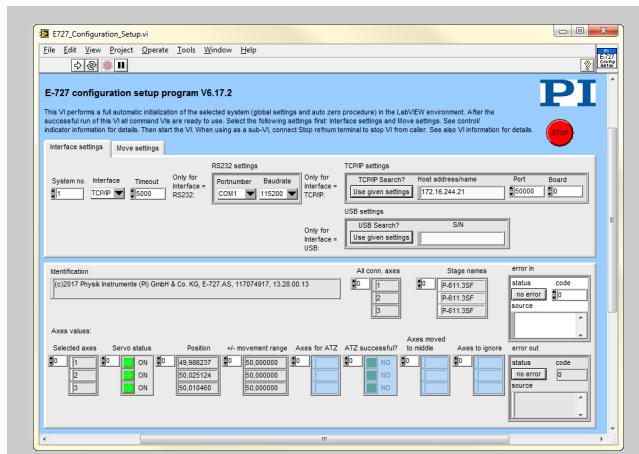


PZ260E Software Manual

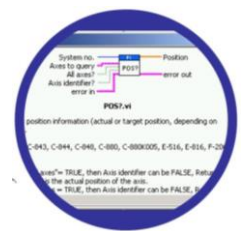
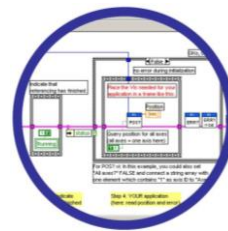
E-727 GCS Driver Library for NI LabVIEW

Release: 6.17.2 Date: 2018-06-29



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

- **E-727** Digital Multi-Channel Piezo Controller



(c)2002-2018 Physik Instrumente (PI)

GmbH & Co. KG

Auf der Römerstr. 1 · 76228 Karlsruhe, Germany

Tel. +49 721 4846-0

Support-software@pi.ws · www.pi.ws

Table of Contents

1. INTRODUCTION.....	3
1.1. PI GENERAL COMMAND SET (GCS).....	3
1.2. SCOPE OF THIS MANUAL.....	4
1.3. VI STRUCTURE	5
1.4. WORKING WITH TWO PI PRODUCTS WHICH UNDERSTAND PI'S GENERAL COMMAND SET (GCS) IN NI LABVIEW	7
1.5. FIRST STEPS FOR GCS-COMPATIBLE PI CONTROLLERS	7
2. HIGH LEVEL VIS	9
3. LOW LEVEL VIS	26
3.1. ANALOG CONTROL.LLB	26
3.2. COMMUNICATION.LLB	27
3.3. CONTROLLER ALGORITHMS.LLB	40
3.4. FILE HANDLING.LLB.....	43
3.5. GENERAL COMMAND.LLB.....	45
3.6. JOYSTICK.LLB	64
3.7. LIMITS.LLB	65
3.8. MACROS.LLB.....	68
3.9. OLD COMMANDS.LLB	73
3.10. OPTICAL OR ANALOG INPUT.LLB.....	74
3.11. PZT VOLTAGE.LLB	78
3.12. SCAN SUPPORT.LLB	82
3.13. SPECIAL COMMAND.LLB	83
3.14. SUPPORT.LLB	112
3.15. WAVEGENERATOR.LLB	126
4. CONTROLS	146
4.1. ANALOG CONTROL.LLB	146
4.2. COMMUNICATION.LLB	146
4.3. GENERAL COMMAND.LLB.....	147
5. ERROR CODES	147

© Copyright 2002 – 2018 Physik Instrumente (PI) GmbH & Co. KG

The following company names and brands are registered trademarks of Physik Instrumente (PI) GmbH & Co. KG: PI®, PICMA®, PILine®, PiezoWalk®, NEXACT®, NEXLINE®, NanoCube®, Picoactuator®, Plnano®, PIMag®, Q-Motion®

Notes on third-party brand names and trademarks:

Microsoft® and Windows® are either registered trademarks or trademarks of Microsoft Corporation in the United States and / or other countries.

LabVIEW, National Instruments and NI are trademarks 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.

The patents owned by PI can be found in our [patent list](#).

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](#)

File: E727_GCSLabVIEW_PZ260E_6172.doc

Disclaimer

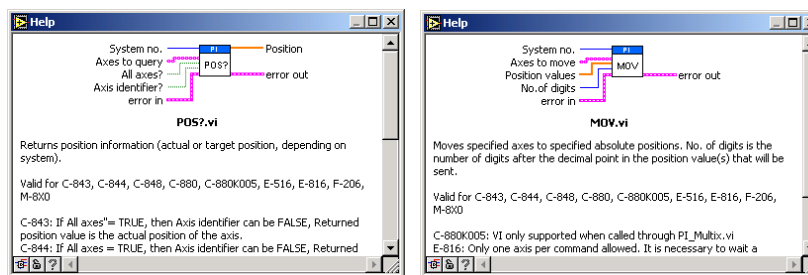
This software is provided “as is”. Physik Instrumente (PI) does not guarantee that this software is free of errors and will not be responsible for any damage arising from the use of this software. The user agrees to use this software on his own responsibility.

1. Introduction

The software for NI LabVIEW 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 NI 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 NI LabVIEW environment to display the *Context Help* window.



NI 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).

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-518 Display Module* or the *C-884 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 NI LabVIEW which connected system to talk to.

Translation Libraries

To control PI systems with a non-GCS 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. PI stage database files) 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 the `PI_LabVIEW_Driver_X-XXX_Setup` from the `CD-ROM\SingleSetups` folder that came with your system. The installation wizard 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 the PI stage database files, it may be because it is marked read-only. To see, open an Explorer, right-click the file `PIStages` file 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 driver set for NI LabVIEW can be used to control a non-GCS-compatible system just like any GCS-compatible system, and PCI/ISA-based controller boards (see Section "First Steps for GCS-Compatible PI Controllers" on p. 8 and the "`XXXX_Configuration_Setup.vi`" (with XXXX being the PI product number of your system) in section 2).

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. VI Structure

The folder structure of the drivers for NI LabVIEW 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, USB 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 NI 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					15
2	7	9	11	13	16
3					17
4					18
5	8	10	12	14	19
6					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 NI LabVIEW error numbers recommended by National Instruments, but rather those used by PI controllers. As a result, the error texts displayed by NI 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 \neq zero.

See also chapter 4 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 NI 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 NI 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 NI LabVIEW

When installing the NI LabVIEW programming support for two different PI products, there are two "Low Level" folders installed, one in each product-specific driver set for NI LabVIEW. 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 driver set installations for NI LabVIEW on one computer, it is important to make sure that NI 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 NI 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 NI LabVIEW's own "user.lib" sub-folder. If they are, NI 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 the PI GCS Merge Tool for NI LabVIEW to combine two or more PI driver sets. Please refer to PILabVIEWMergeTool_SM154E.pdf in the PI Manuals folder (C:\Users\Public\PI\Manuals) for more details.

1.5. First Steps for GCS-Compatible PI Controllers

1.5.1. E-727

Step 1: The E-727 controller is delivered pre-configured. Before you start, please check that the current configuration matches your stage connections.

Step 2: (advanced users can skip this step): To check communication between the E-727 controller and the host PC, run "E727_Simple_Test.vi". This VI will return the ID string of the E-727 controller, the axis IDs and stage names of the connected axes, the current position and the help string. 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:

WARNING: E727_Configuration_Setup.vi May Cause Move

When you start "E727_Configuration_Setup.vi" with Are stages connected? = TRUE and Perform Autozero? = TRUE or Move to Middle? = TRUE, the VI will perform an automated zero-point calibration of the connected stages and/or move all axes to their middle positions. It is therefore important to make sure that items connected to or mounted on the connected stages cannot be damaged by such a move. If a move is not desired, Are stages connected? can be switched off (see chapter 3).

To control one or more E-727 controllers with this driver set, run "E727_Configuration_Setup.vi". This VI performs all steps necessary for a full configuration of the driver VIs in the NI LabVIEW environment: e.g. the definition of the axis IDs, the automated zero-point calibration (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 "E727_Configuration_Setup.vi"; otherwise all global settings will be lost and the driver VIs will not work.

When programming your application, you can implement "E727_Configuration_Setup.vi" as an initialization VI in your software. See chapter 2 for a detailed description of "E727_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. See also "E727_Sample_Application_1.vi" as a sample VI showing how to implement your application using "E727_Configuration_Setup.vi".

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

Default position unit is μm , default velocity unit is $\mu\text{m/s}$.

Default IDs: AxisIDs are "1", "2", "3", "4"

Input Signal Channel IDs: "1", "2", "3", etc. Note: Input Signal Channels comprise the sensor channels for the sensors in the mechanics (Number Of Sensor Channels parameter, ID 0x0E000B03) and all analog input channels. In the E-727 firmware, the sensor channels are always represented by the first three input signal channels, while the analog input lines are always represented by the input signal channels four and higher.

Output Signal Channel IDs: "1", "2", "3", etc. Note: Output Signal Channels comprise the piezo voltage amplifiers dedicated to the actuators in the mechanics (Number Of Piezo Channels parameter, ID 0x0E000B04) and all analog output channels. In the E-727 firmware, the piezo amplifier channels are always represented by the first four output signal channels, while the analog output lines are always represented by the output signal channels five and higher.

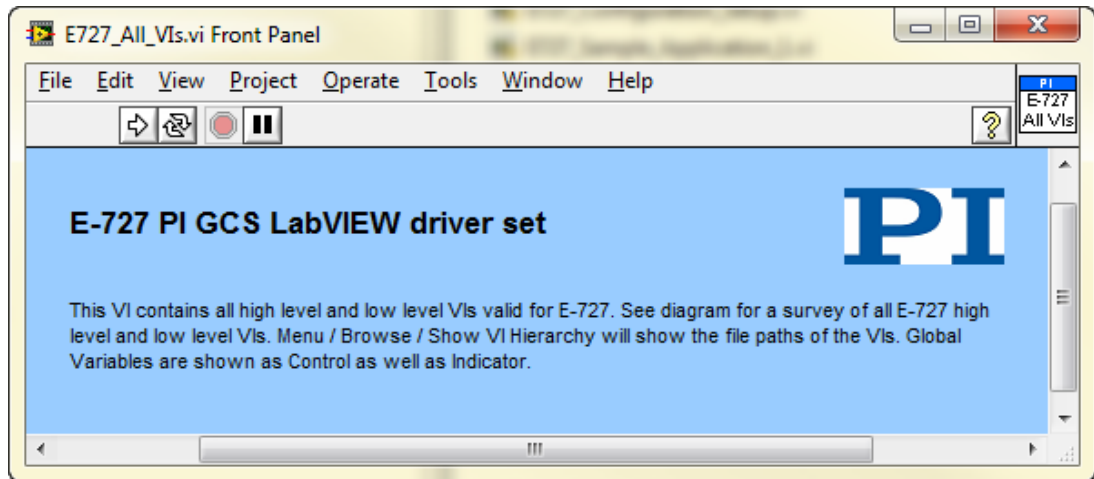
Digital Input and Output Line IDs: 1 to 3 (trigger only)

For more information on axes and channels, see the E-727 user manual (E727T0005).

GCS syntax version: 2.0

2. High Level VIs

2.1.1. E727_All_VIs.vi



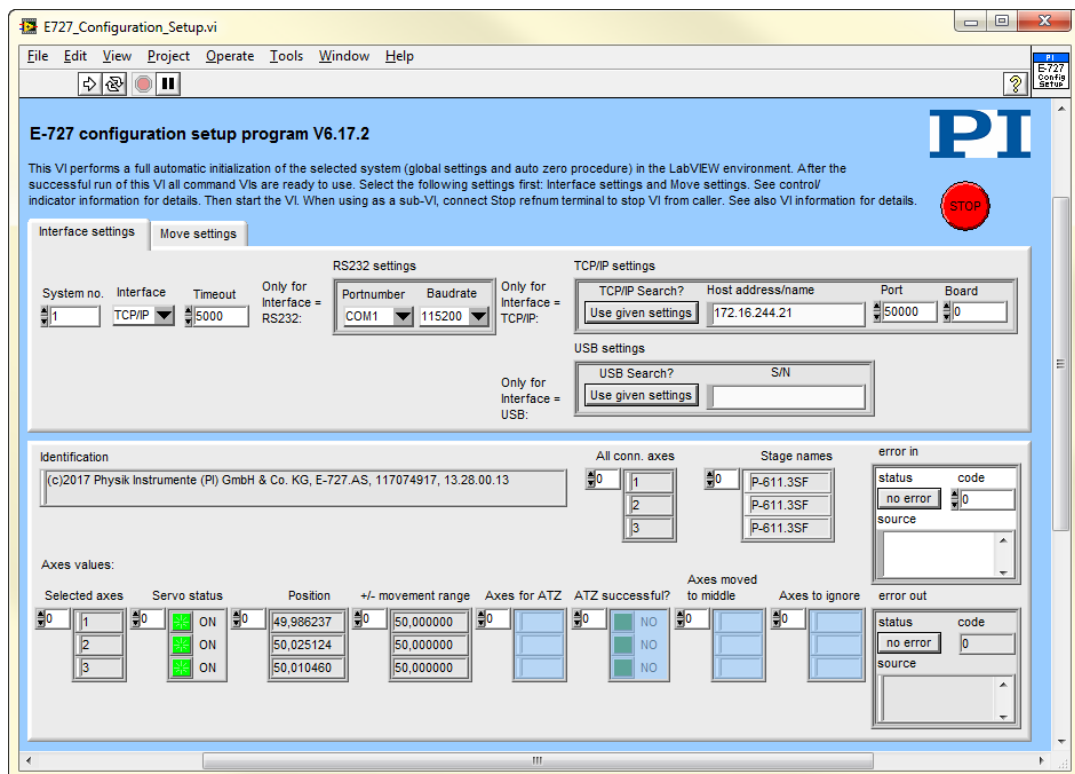
This VI contains all high level and low level VIs valid for E-727.

See diagram for a survey of all E-727 high level and low level VIs.

Menu / Browse / Show VI Hierarchy will show the file paths of the VIs.

Global Variables are shown as Control as well as Indicator.

2.1.2. E727_Configuration_Setup.vi



This VI performs a fully automatic initialization of the selected system (global settings and automatic zero-point calibration of the stages, if appropriate) 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 run 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, TCP/IP or USB). For interface = RS232, select the correct Portnumber and Baudrate. For interface = TCP/IP, select if you want to call a dialog to select the connected E-727 controller from a list (TRUE) or if you know the host address/name of the controller to connect (FALSE). For interface = USB, select the appropriate Serial Number of the controller to connect to (USB Search? FALSE) or if you want to search for the controller (USB Search? TRUE).
- Timeout value (in milliseconds)
- To which axes stages are connected and can be moved
- Whether a wave generator output is to be stopped (if you are not sure if any wave generator is running, leave this control TRUE).
- Whether the servo status of the connected axes is to change to TRUE. If FALSE is selected here, the servo status will not be changed.
- If auto zero is to be performed, whether the low voltage parameter is to define manually (Low voltage = I32 value) or automatically (Low voltage = NaN)
- If axes are to be moved to the middle positions of their travel range.

Then start the VI.

E727_Configuration_Setup.vi performs the following initialization tasks:

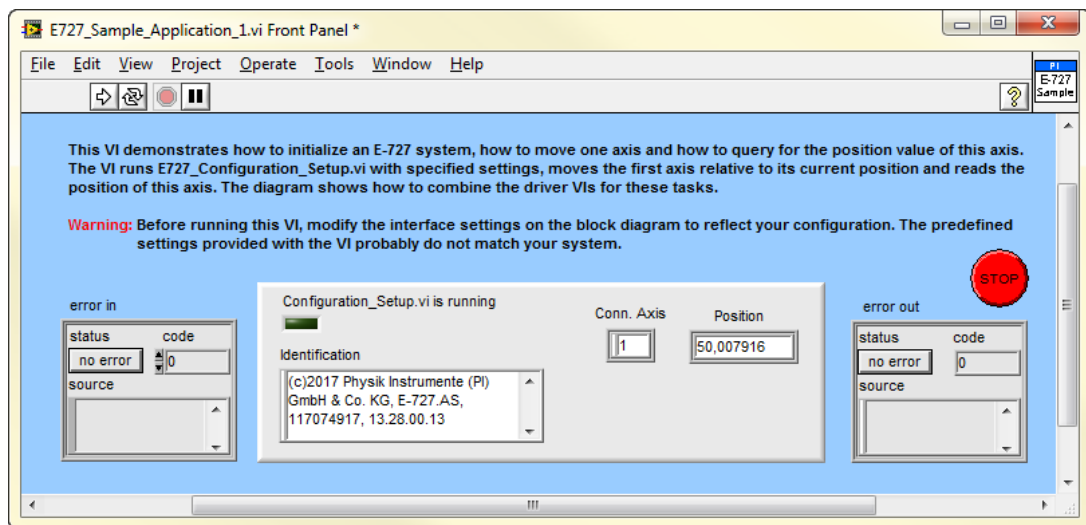
1. Initializes all Global variables.
2. Runs PI Open Interface of one system.vi to open a connection to the controller.
3. Runs *IDN?.vi to query for the controller identification string.
4. Defines the selected system to be "E-727"
5. Runs Define connected axes.vi with Read from controller = TRUE to query for the axis IDs of the connected axes.
6. Runs CST?.vi to query for the names of the connected stages.
7. If Stop wave generator output (if any) is TRUE, runs WGO.vi with WGO parameter = 0.
8. Runs SVO.vi to switch servo on (if Change servo status to ON? is TRUE)
9. Runs SVO?.vi to display servo status of axes
10. If Perform autozero? is TRUE, runs ATZ.vi
11. Runs POS?.vi to query for the position of the connected axes
12. Reads the position range (TMN?, TMX?) of the axes.
13. If Move to middle? is TRUE, moves axes to the middle positions of their range (MOV).
14. Runs POS?.vi to query for the position of the axes.
15. Runs ERR?.vi to query the controller for its error status.
16. Runs GCSTranslateError.vi to append the error message which corresponds with a GCS error number returned by ERR?.vi to Source from Error out.

Use this VI as the initialization VI for the E-727 in your application.

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

As the initialization is a complex procedure which uses a large number of sub-VIs, E727_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.

2.1.3. E727_Sample_Application_1.vi

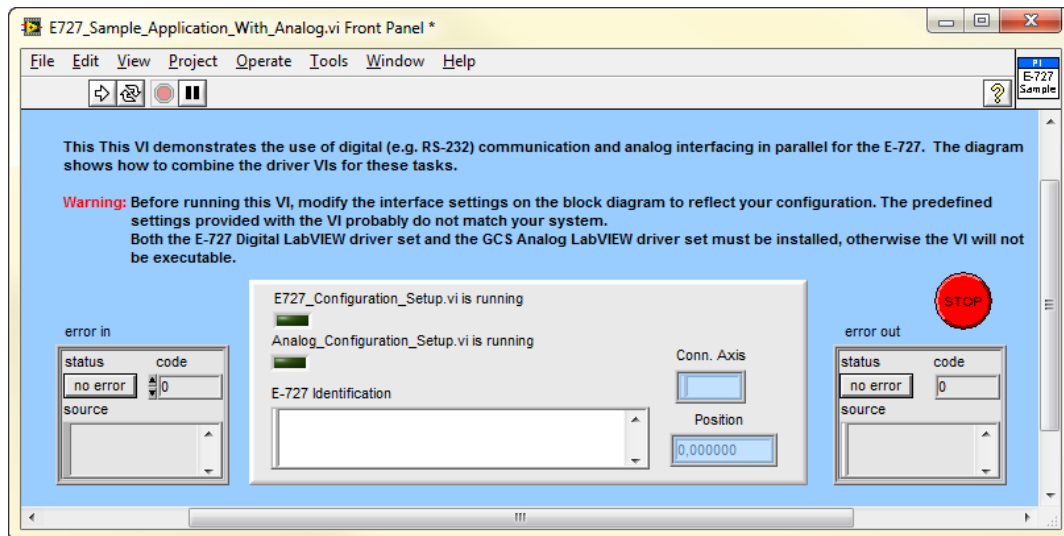


This VI demonstrates how to initialize an E-727 system, how to move one connected axis and how to query for the position value of this axis.

In this example the E-727 is connected through COM port 1 with baudrate 115200. The VI runs E727_Configuration_Setup.vi with these specified settings, moves the first connected axis relative to its current position and reads the position of this axis. The diagram shows how to combine the driver VIs for these tasks.

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

2.1.4. E727_Sample_Application_With_Analog.vi



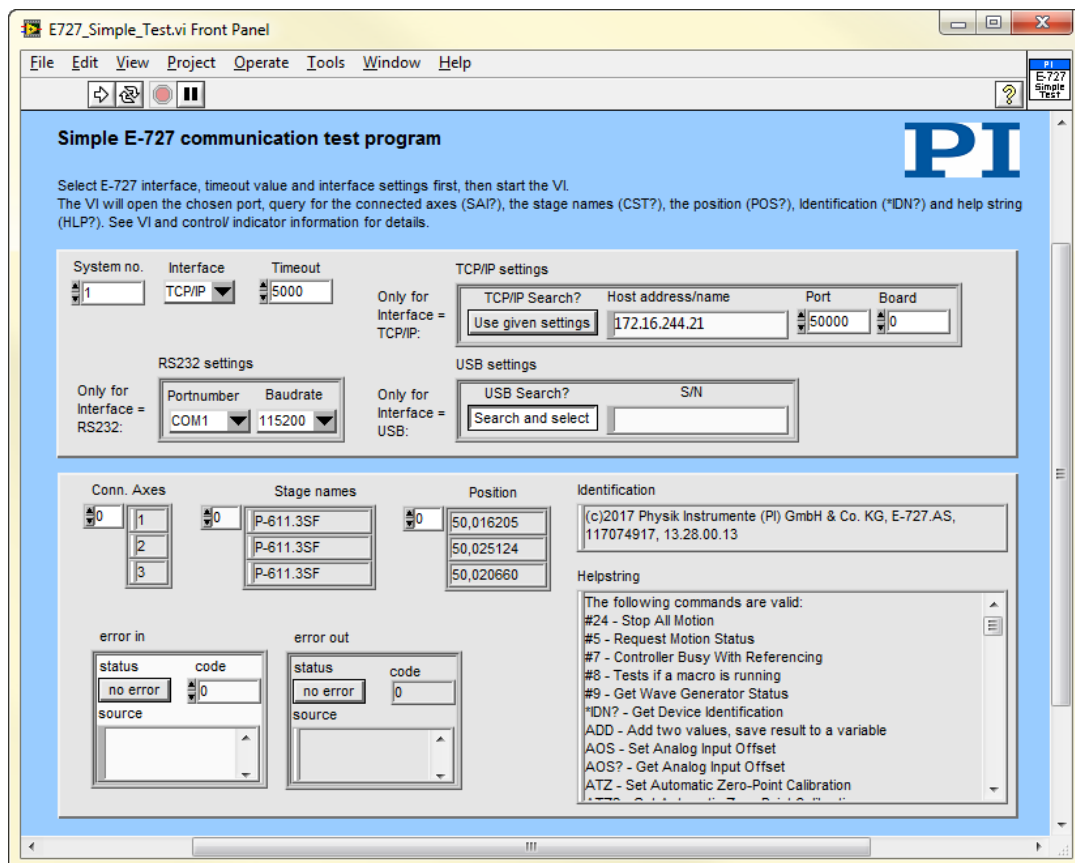
This VI demonstrates the use of digital (e.g. RS-232) communication and analog interfacing in parallel for the E-727. In this example the E-727 is connected through COM port 1 with baudrate 115200. The VI first initializes the E-727 via the selected digital interface by running E727_Configuration_Setup.vi. It then switches to analog control, using SwitchToAnalogOrDigital.vi and initializes the analog interface by running Analog_Configuration_Setup.vi. After these initialization steps, it moves one axis relative to its current position by outputting a voltage by the DAC card used, waits 10 ms and queries for the position of this axis. Finally, it switches back to digital control and closes both the analog and the digital connection to the controller.

As the E-727 provides no Sensor Monitor Output, connect the Analog Out line of the DAC card to the corresponding Analog In line of the card.

The diagram shows how to combine the driver VIs for these tasks.

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

2.1.5. E727_Simple_Test.vi



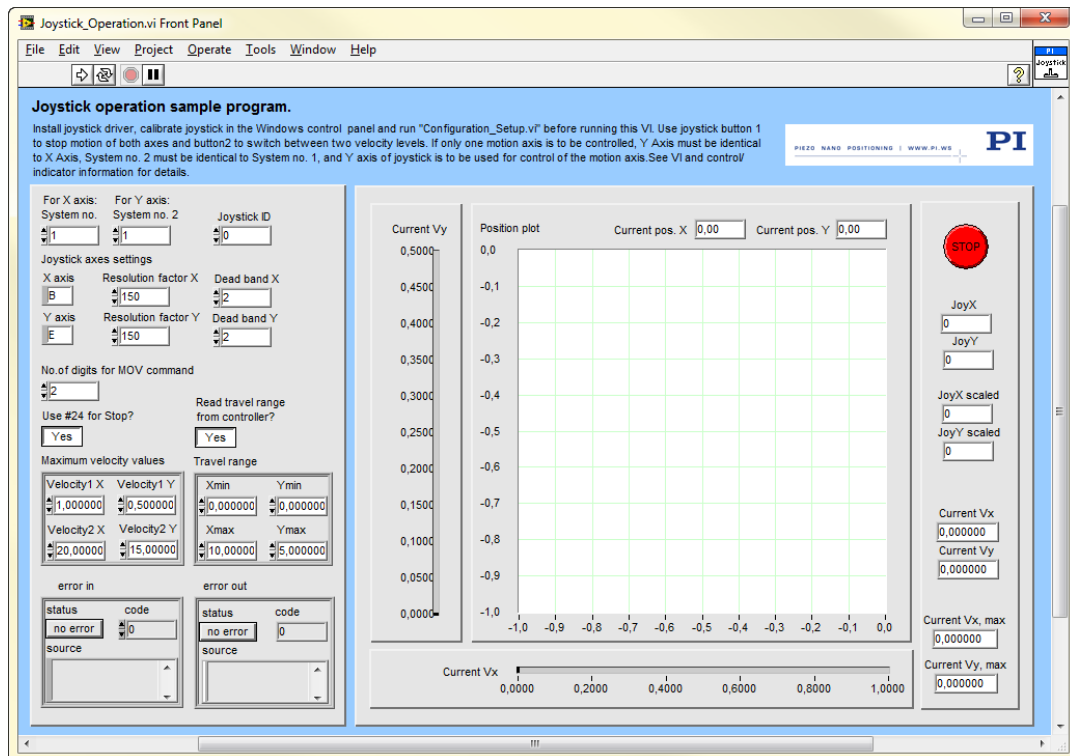
This simple test VI is a stand-alone sample application. Use the Help->Show Context Help menu sequence in the NI LabVIEW environment to display the Context Help window with the VI and control/indicator descriptions.

Specify

- System no. (= 1 in a one-system configuration),
- Interface (= RS232, TCP/IP or USB)
- Timeout value (in milliseconds)
- and interface settings first.

Then start the VI. The VI will open a connection to the E-727 and query the controller for its identification string (*IDN?), help string (HLP?), the IDs (SAI?) and stage names (CST?) of the connected axes and their current position values (POS?). The diagram shows how to combine the driver and support VIs for these tasks.

2.1.6. 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 host computer. The absolute value of the joystick position is converted into a 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-877, C-880, C-884, C-885, 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

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_Configuration_Setup_Rev2.vi" prior to running this VI.

C-877: Use #24 for stop = TRUE, Read from controller = TRUE;

Run "C877_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.

C-885: Use #24 for stop = TRUE, Read from controller = TRUE;

Run "C885_Configuration_Setup.vi" prior to running this VI.

C-891: Use #24 for stop = TRUE, Read from controller = TRUE;

Run "C891_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 $\mu\text{m/s}$.

Run "E517_Configuration_Setup.vi" prior to running this VI.

E-518: Use #24 for stop = TRUE, Read from controller = TRUE.

Default position unit is μm , default velocity unit is $\mu\text{m/s}$.

Run E518_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 $\mu\text{m/s}$.

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 $\mu\text{m/ms}$. 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 $\mu\text{m/s}$. 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 $\mu\text{m/s}$. Run "E725_Configuration_Setup.vi" prior to running this VI.

E-727: Use #24 for stop = TRUE, Read from controller = TRUE. Default position unit is μm , default velocity unit is $\mu\text{m/s}$. Run E727_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 $\mu\text{m/s}$. Run "E753_Configuration_Setup.vi" prior to running this VI.

E-754: Use #24 for stop = TRUE, Read from controller = TRUE. Default position unit is μm , default velocity unit is $\mu\text{m/s}$. Run "E754_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 $\mu\text{m/s}$. 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 $\mu\text{m/ms}$. 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.

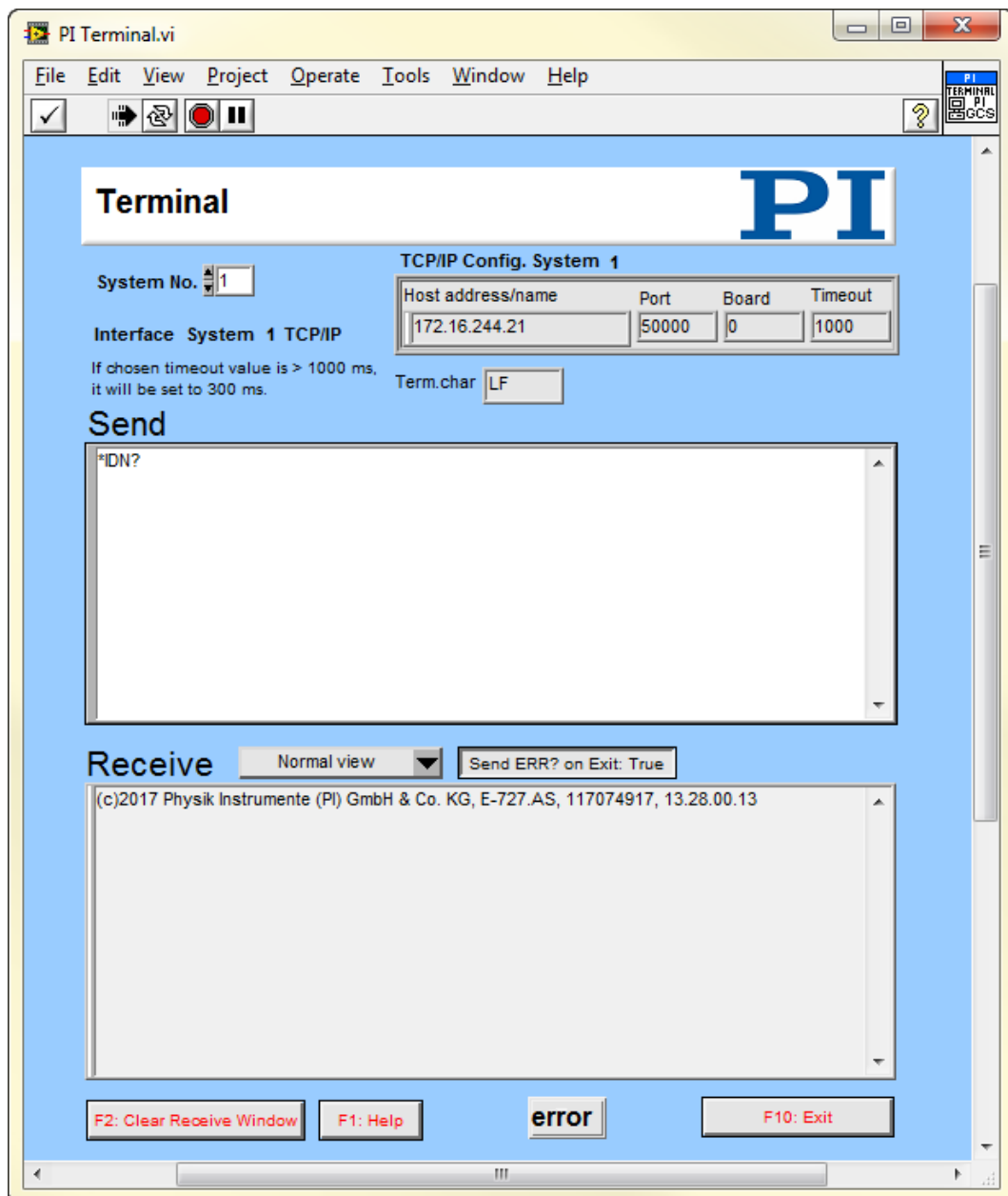
Hydra, Pollux: Use #24 for stop = TRUE, Read from controller = TRUE;

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.

2.1.7. PI Terminal.vi



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

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

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

C-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. DaisyChain: Interface = DLL, DLL for Device = PI_GCS2_DLL, DLL Interface = TCPIP_DC or USB_DC, Parameter = Number of device in chain, Register DC: FALSE. Syntax: GCS 2.0; Term char = LF.

C-886: 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-887, F-206, M-8X0: For GCS syntax version = GCS 1.0 (older firmware or compatibility mode, can be set with CSV.vi), Interface = RS232, GPIB (not available for C-887 controller) or TCP/IP., The error status will not be cleared by this VI. The first ERR? query will report a hidden error with error code 1, which will

be cleared during system initialization (INI). RS232: Input and output handshake settings must be FALSE. Syntax: GCS 1.0; Term char = LF.

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

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 (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-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 = USB, Parameter = Serial no. of system to connect to. Syntax: GCS 2.0; Term char = LF.

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

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

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

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

E-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). Syntax: GCS 2.0; Term char = LF.

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

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

E-861: Single Device: Interface = RS232 or DLL, RS232: Input and output HW handshake must be FALSE. DLL (USB): DLL for Device = E-861, DLL Interface = USB, Parameter = Serial no. of system to connect to. DaisyChain: Interface = DLL, DLL for Device = E-861, DLL Interface = RS232_DC, Parameter = Number of device in chain (first device: 1), Register DC: FALSE. Syntax: GCS 2.0; Term char = LF.

E-870: Interface = DLL, DLL (USB): DLL for Device = PI_GCS2_DLL, DLL Interface = USB, Parameter = Serial no. of system to connect to.

Syntax: GCS 2.0; Term char = LF.

E-871: Single Device: Interface = RS232 or DLL, RS232: Input and output HW handshake must be FALSE. DLL (USB): DLL for Device = PI_GCS2_DLL, DLL Interface = USB, Parameter = Serial no. of system to connect to.

DaisyChain: Interface = DLL, DLL for Device = PI_GCS2_DLL, DLL Interface = RS232_DC or USB_DC, Parameter = Number of device in chain, Register DC: FALSE. Syntax: GCS 2.0; Term char = LF.

E-872: 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-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.

E-874: 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.

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

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

GCS_Mercury: Single Device: Interface = RS232 or DLL, RS232: Input and output HW handshake must be FALSE. DLL (USB): DLL for Device = PI_GCS2_DLL, DLL Interface = USB, Parameter = Serial no. of system to connect to.

DaisyChain: Interface = DLL, DLL for Device = PI_GCS2_DLL, DLL Interface = RS232_DC, Parameter = Number of device in chain, Register DC: FALSE. Syntax: GCS 2.0; Term char = LF.

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

If the chosen timeout value is greater than 1000 ms, it will automatically be set to 1000 ms for a fluid program operation.

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

All controller responses are displayed in the Receive response window, which can be cleared by pressing the Clear Receive Window button or F2.

The view style of the Receive window can be changed to Show all characters or Hex View using the menu ring above the Receive window.

Exit or F10 will terminate the terminal application.

To send the last command again, just press the ENTER key again. The next line will then show the following entry: "(Send: cmd)" with cmd being the command from the line before, which was resent.

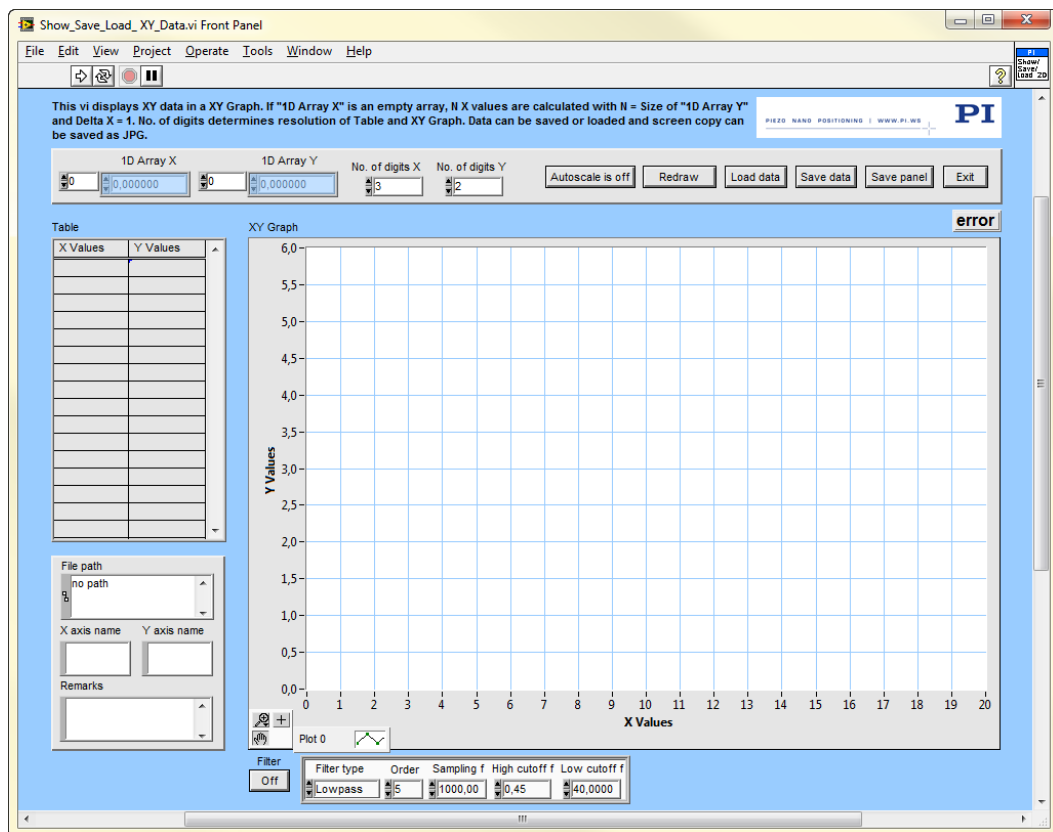
When the terminal application has just been started, pressing ENTER without entering a command will send "*IDN?" to the chosen system.

New commands can only be inserted into the last line of the Send window. The user can scroll through the history of the Send window using the scroll bar or the cursor up/down keys, but cannot change the history or resend commands by pressing ENTER unless in the last line. Pressing ENTER will always resend the last command, no matter where the cursor is positioned. Selecting text and using copy and paste (Ctrl+C, Ctrl+V) works for single lines, if only the contents of one single line (the command text) is selected and copied, not the full line (including the LineFeed) or multiple lines.

Many of PI's General Command Set compatible devices support single-byte commands. For example, the user can stop a fast scan of a C-880 or C-887 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.

2.1.8. Show_Save_Load_XY_Data.vi



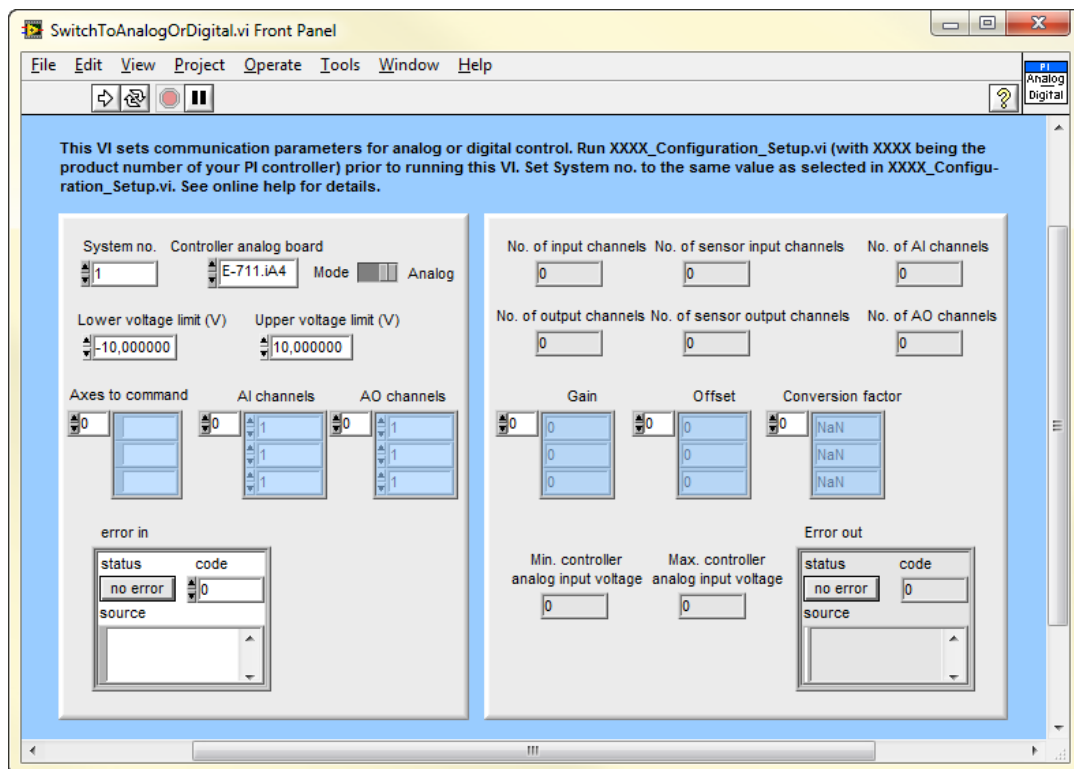
This VI displays XY data in an XY Graph. If 1D Array X is an empty array, N X values are calculated with $N = \text{Size of 1D Array Y}$ and $\Delta X = 1$. No. of digits determines the resolution of Table and XY Graph. Data can be saved or loaded and a screen copy can be saved as JPG.

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.

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, Analog systems, C-413, C-843, C-866, C-867, C-877, C-880, C-884, C-886, 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, E-874, 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.

2.1.9. SwitchToAnalogOrDigital.vi



This VI sets communication parameters for analog or digital control. Run XXXX_Configuration_Setup.vi (with XXXX being the product number of your PI controller) prior to running this VI.

- Set System no. to the same value as selected in XXXX_Configuration_Setup.vi.
- Select the name of the controller analog board (E-711.iA4).
- Mode can be Analog (T) or Digital (F).
- Lower voltage limit (V) and Upper voltage limit (V) must match the corresponding settings in Analog_Configuration_Setup.vi, representing the voltage range of the analog input of the controller. The VI checks if the selected values fit into the analog input voltage range of the controller.
- Axes to command must be a sub set of the axes connected to the controller.
- All channels are the analog input channels of the controller to use for analog control. In contrast to the firmware internal names of the analog input channels, select "1" here for the first analog input, "2" for the second and so on. The VI will automatically determine which input channel names correspond to the selected analog input channels.
- Similarly, select "1" for the first analog output channel (AO channels), "2" for the second analog output channel and so on, as the VI will automatically determine which output channel names correspond to the selected analog output channels.

If the current command level is 0 (CCL?), the VI switches to command level 1 (CCL) before making any further settings and back to 0 afterwards.

When Analog is selected as Mode, the VI connects the selected analog input channels to the selected axes (SPA), sets gain and offset for the analog input (SPA), sets the output type to sensor monitor output (SPA), connects the selected analog output channels to the selected axes (SPA), and sets the position report scaling value and offset for the analog output (SPA).

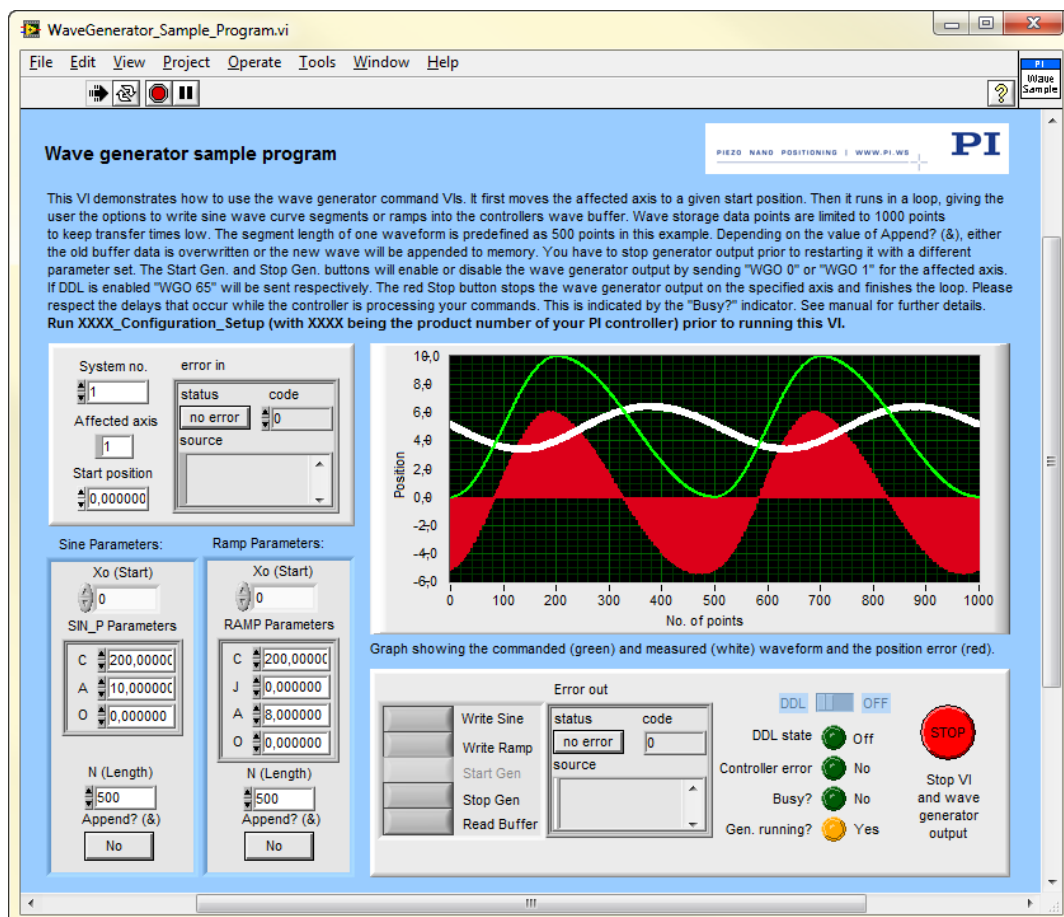
When Digital is selected as Mode, the VI disconnects the selected analog input channels from the axes (SPA) and sets the output type to control output (SPA).

The VI returns the total number of input and output channels available for this controller, the number of sensor input/output channels and the resulting number of analog input/output channels available for analog control. Also, it returns the Gain, Offset and Conversion factor settings resulting from the motion range (TMX?, TMN?) of the selected axes and sets the corresponding parameters (SPA). Finally, it returns the minimum and maximum analog input voltages of the controller as the given lower and upper voltage limits must fit into this range.

See XXXX_Sample_Application_With_Analog.vi (with XXXX being the product number of your PI controller) as a sample VI showing how to use this VI in combination with the corresponding Configuration_Setup VIs.

Valid for: C-413, E-709, E-712, E-725, E-727, E-754 (but must be present for C-885 also)

2.1.10. WaveGenerator_Sample_Program.vi



This VI demonstrates how to use the wave generator command VIs. It first moves the affected axis to a given start position. Then it runs in a loop, giving the user the options to write sine wave curve segments or ramps into the controllers wave buffer. Wave storage data points are limited to 1000 points to keep transfer times low. The segment length of one waveform - which normally defines the frequency - is predefined as 500 points in this example. Depending on the value of Append? (&), either the old buffer data is overwritten or the new wave will be appended to memory. Please take into account that you can append at most two waveforms since buffer size is limited to 1000 points as mentioned above.

Adjust the following controls to modify the shape of the waveforms.

N (Length): Waveform (i.e. the points to be modified) goes from 1 to Length. Datapoints from Length to 500 are filled with value of last wave point.

Xo (Start): Position where the first point of the waveform is located.

C: Center point of the wave. If $C = \text{Length}/2$ then the wave is symmetrical.

A: Amplitude of the waveform. $F(x) = A \cdot \sin(x) + O$

O: Offset of the waveform. $F(x) = A \cdot \sin(x) + O$

J: Jerk. Means curvature in the transition to or from standstill.

You have to stop generator output prior to restarting it with a different parameter set. The Start Gen. and Stop Gen. buttons will enable or disable the wave generator output by sending "WGO 0" or "WGO 1" for the affected axis. If DDL is enabled "WGO 65" will be sent respectively. The red Stop button stops the wave generator output on the specified axis and finishes the loop. Please respect the delays that occur while the controller is processing your commands. This is indicated by the "Busy?" indicator. See manual for further details.

In case of an error a button Clear Error will appear and the control buttons above will be disabled. Pressing this button will clear the error status and enable the control buttons again.

With DDL ON you can achieve significantly better position accuracy. Compare the target (green), response (white) and error (red) signals on the graph. Since this is a very sensitive mechanism we recommend starting with low amplitudes and adjusting the parameters with caution. The DDL button works only if the controller supports DDL. Contact PI for information on activation of DDL support.

See User Manual for details about the use of the wave generator.

Run XXXX_Configuration_Setup (with XXXX being the product number of your PI controller) prior to running this VI.

Valid for: E-712, E-725, E-727, E-753, E-754

3. Low Level VIs

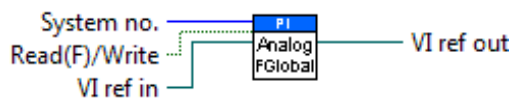
The following low-level VIs can be found in the "Low Level" folder:

3.1. Analog control.llb

3.1.1. Analog FGlobal.vi

This VI works as a functional global variable for VI references

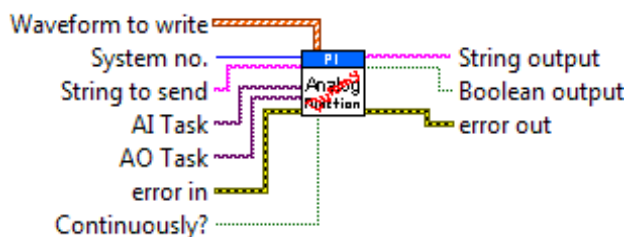
Valid for Analog Systems, but must be present for all other systems, too.



3.1.2. Analog Functions.vi

This vi calls Analog Functions (dyn).vi functions dynamically during runtime, depending on "String to send".

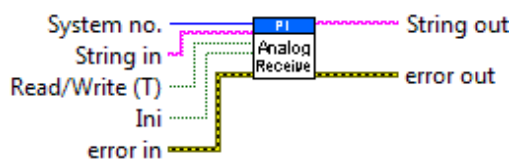
Valid for Analog Systems (but must be present for all other systems, too)



3.1.3. 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)



3.1.4. 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).

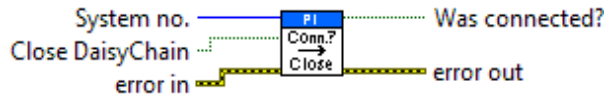


3.2. Communication.Ilb

3.2.1. 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.



3.2.2. 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-872, E-873, E-874, 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 othercontrols=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 othercontrols=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.

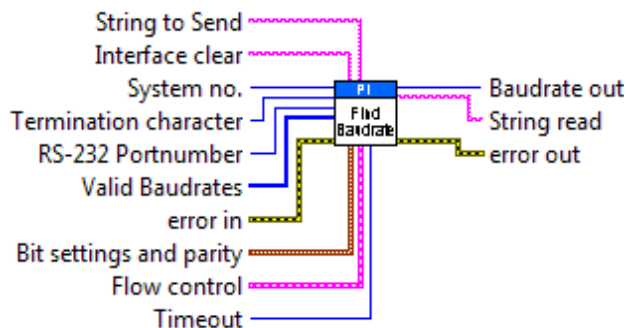
E-872: Input and output HW handshake must be FALSE. All other controls=default.

E-873: Input and output HW handshake must be FALSE. All other controls=default.

E-874: 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.



3.2.3. Find host address.vi

Performs an UDP broadcast and returns IP addresses of all controllers matching "Controller name". VI will also stop if Stop refnum or Local stop is TRUE.

Valid for C-413, C-702, C-867 C-884, C-885, C-886, C-887, C-891, E-517, E-518, E-712, E-725, E-727, E-753, E-754, E-871, E-872, E-873, E-874, F-206, M-8X0

C-413: Controller names = C-413, Mode = Find XPort by UDP, Port = 30718

C-413: Controller names = C-413, Mode = Find USB-K3 by UDP, Port = 1901

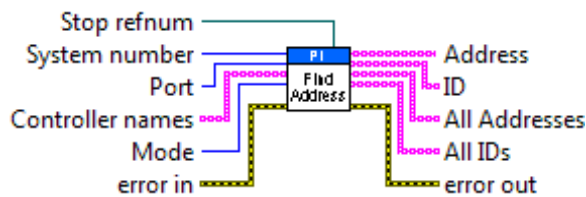
C-702: Controller names = C-702, Mode = Find controller by UDP, Port = 50000

C-867: Controller names = C-867, Mode = Find controller by UDP, Port = 50000

C-884: Controller names = C-884, Mode = Find controller by UDP, Port = 50000

C-885: Controller names = C-885, Mode = Find controller by UDP, Port = 50000

C-886: Controller names = C-886, Mode = Find controller by UDP, Port = 50000
 C-891: Controller names = C-891, Mode = Find XPort by UDP, Port = 30718
 E-517: Controller names = E-517, Mode = Find controller by UDP, Port = 50000
 E-518: Controller names = E-518, Mode = Find controller by UDP, Port = 50000
 E-712: Controller names = E-712, Mode = Find controller by UDP, Port = 50000
 E-725: Controller names = E-725, Mode = Find XPort by UDP, Port = 30718
 E-727: Controller names = E-727, Mode = Find controller by UDP, Port = 50000
 E-753: Controller names = E-753, Mode = Find controller by UDP, Port = 50000
 E-754: Controller names = E-754, Mode = Find controller by UDP, Port = 50000
 E-871: Controller names = E-871, Mode = Find XPort by UDP, Port = 30718
 E-872: Controller names = E-872, Mode = Find controller by UDP, Port = 50000
 E-873: Controller names = E-873, Mode = Find XPort by UDP, Port = 30718
 E-874: Controller names = E-874, Mode = Find USR-K3 by UDP, Port = 1901
 F-206: Controller names = F-206, F-HEX, Mode = Find XPort by UDP, Port = 30718
 C-887, M-8X0: Controller names = M-8X0, HEXAPOD, Mode = Find XPort by UDP, Port = 30718

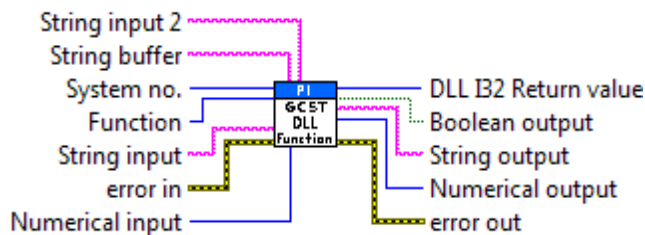


3.2.4. GCSTranslator DLL Functions.vi

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-886, 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-872, E-873, E-874, Hydra, Pollux, Mercury, Mercury_GCS (but must be present in Communication.llb for all other systems also)



3.2.5. Get subnet.vi

Calls system specific operating system function and returns subnet broadcast addresses of all installed network cards. Supports Windows (default case, calls IPCONFIG), Unix (calls IFCONFIG) and Mac.OS (calls IFCONFIG).

Valid for: C-413, C-702, C-867, C-884, C-885, C-886, C-887, C-891, E-517, E-518, E-712, E-725, E-727, E-753, E-754, E-871, E-872, E-873, E-874, F-206, M-8X0 (but must be present for all other systems except Analog systems, too)



3.2.6. Global DaisyChain.vi

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

Valid for all systems.



3.2.7. Global1.vi

A global variable which contains communication setup information.

Valid for all systems.



3.2.8. IFC.vi

Changes the current InterFace Configuration. After IFC is sent, the new interface configuration of the controller becomes active and the host must change the interface configuration too. Settings made with IFC are valid until the controller is powered down.

To save them to EPROM and make them the default use IFS instead.

Valid for C-702, C-867, C-884, C-885, E-517, E-518, E-709, E-712, E-725, E-727, E-753, E-754, E-871, E-872, E-873, E-874, Hydra

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

C-884: MACADR cannot be changed with IFC. To change IPADR the use of IFS is recommended instead.

C-885: MACADR cannot be changed with IFC. To change IPADR the use of IFS is recommended instead.

E-517: The current active RS-232 baud rate and GPIB (IEEE 488) address can also be set on the E-517 front panel in the COMMUNICATION display screen. This screen can be accessed by the trackball. The front panel settings interdepend with the interface parameter settings set with IFC. See E-517 User Manual for details.

E-709: Use IFC to change baud rate settings only.

E-712: Use IFC to change baud rate settings only, for all other settings use IFS instead.

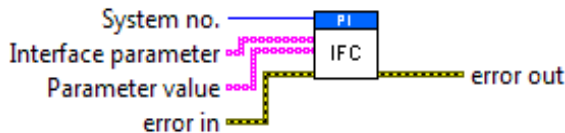
E-725, E-727: Use IFC to change baud rate settings only, for all other settings use IFS instead.

E-753: Use IFC to change baud rate settings only, for all other settings use IFS instead.

E-754: Use IFC to change baud rate settings only, for all other settings use IFS instead.

E-871: Use IFC to change baud rate settings only, for all other settings use IFS instead. Check HLP? answer to find out if IFC is supported.

E-872, E-873, E-874: Use IFC to change baud rate settings only, for all other settings use IFS instead.



3.2.9. IFC?.vi

Returns the current interface configuration.

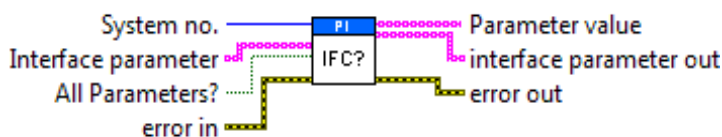
Valid for C-702, C-867, C-884, C-885, C-886, C-887, E-517, E-518, E-709, E-712, E-725, E-727, E-753, E-754, E-871, E-872, E-873, E-874, F-206, M-8X0, Hydra

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

E-517, E-518, E-753, E-754, E-872, E-873, E-874: Note that when the controller is part of a network with DHCP, the static IP address of the controller is returned, not the currently used IP address which was obtained from the DHCP server.

E-871: Note that when the controller is part of a network with DHCP, the static IP address of the controller is returned, not the currently used IP address which was obtained from the DHCP server. Check HLP? answer to find out if IFC? is supported.

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



3.2.10. IFS.vi

If Password is correct, the default parameter(s) for the interface are changed, but the current active parameters are not changed. Settings made with IFS are saved to EPROM and become active with the next startup/reboot. To change settings immediately (but temporarily) use IFC instead (if supported by your controller).

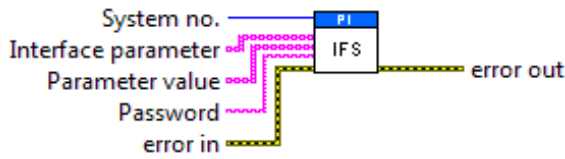
Valid for C-702, C-867, C-884, C-885, C-886, C-887, E-517, E-518, E-709, E-712, E-725, E-727, E-753, E-754, E-871, E-872, E-873, E-874, F-206, M-8X0

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

E-709: Interface parameter can only be RSBAUD.

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

E-871: Check HLP? answer to find out if IFS is supported.



3.2.11. IFS?.vi

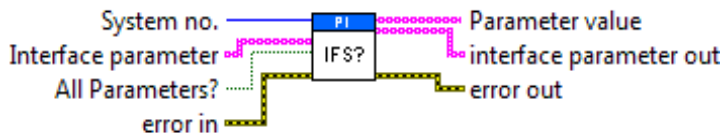
Returns the default InterFace configuration which is Stored in EPROM.

Valid for C-702, C-867, C-884, C-885, C-886, C-887, E-517, E-518, E-709, E-712, E-725, E-727, E-753, E-754, E-871, E-872, E-873, E-874, F-206, Hydra, M-8X0

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

E-871: Check HLP? answer to find out if IFS? is supported.

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



3.2.12. 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)

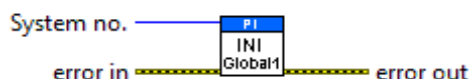
E-709: Only supported if E-709 is used inside C-867K012/K013.



3.2.13. Initialize Global1.vi

This VI initializes Global1 according to the given system no.

Valid for all systems.

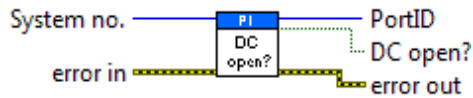


3.2.14. Is DaisyChain open.vi

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)

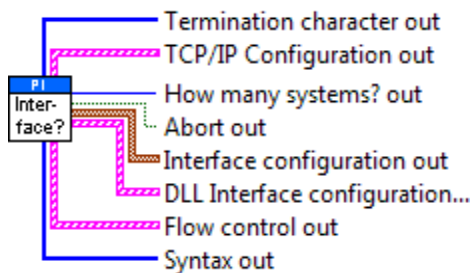
E-709: Only supported if E-709 is used inside C-867K012/K013.



3.2.15. 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.



3.2.16. 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. DaisyChain: Interface = DLL, DLL for Device = PI_GCS2_DLL, DLL Interface = TCP/IP_DC or USB_DC, Parameter = Number of device in chain, Register DC: FALSE. Syntax: GCS 2.0; Term char = LF.

C-886: 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-872: 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-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.

E-874: 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.

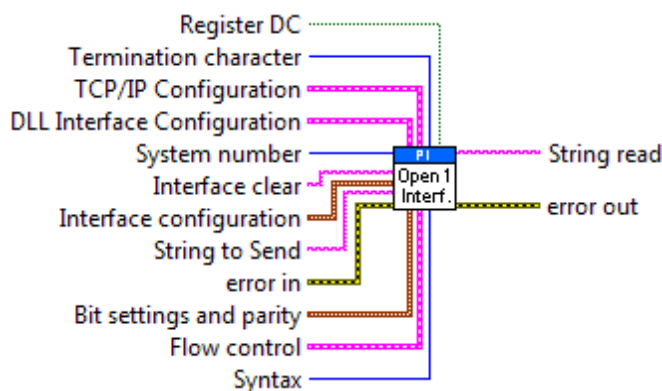
C-887 + H-810/11/24/40/50 (GCS 2.0): Interface = RS232 or TCP/IP, RS232: Input and output handshake settings must be FALSE, Syntax: GCS 2.0; Term char = LF.

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.

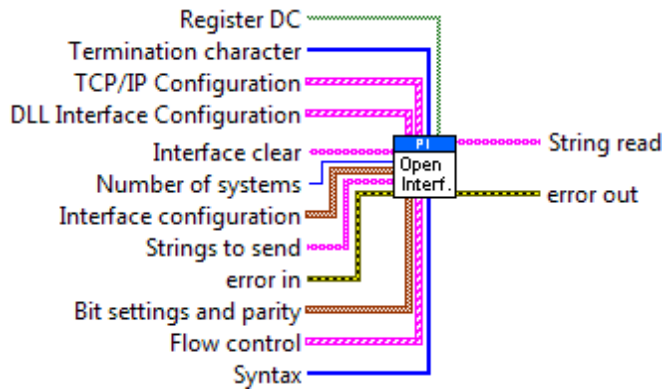


3.2.17. PI Open Interface.vi

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.

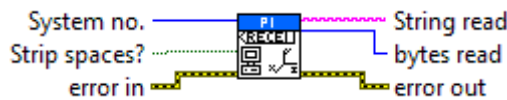
See PI Open Interface of one system.vi for control settings.



3.2.18. PI Receive String.vi

Read string from selected system.

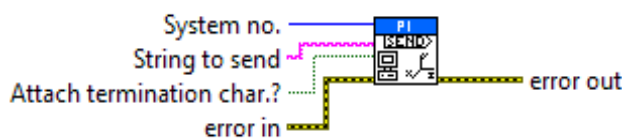
Valid for all systems.



3.2.19. PI Send String.vi

Sends command with or without trailing termination character to selected system.

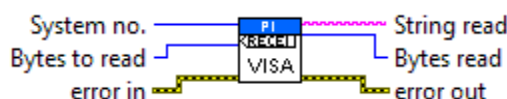
Valid for all systems.



3.2.20. 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-886, 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-872, E-873, E-874, F-206, M-8X0, Mercury_GCS (but must be present in Communication.llb for all other systems, too)



3.2.21. Select host address.vi

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

Valid for C-413, C-702, C-867, C-884, C-885, C-886, C-887, C-891, E-517, E-518, E-712, E-725, E-727, E-753, E-754, E-871, E-872, E-873, E-874, F-206, M-8X0

C-413: Controller names = C-413, Mode = Find XPort by UDP, Port = 30718

C-413: Controller names = C-413, Mode = Find USR-K3 by UDP, Port = 1901

C-702: Controller names = C-702, Mode = Find controller by UDP, Port = 50000

C-867: Controller names = C-867, Mode = Find controller by UDP, Port = 50000

C-884: Controller names = C-884, Mode = Find controller by UDP, Port = 50000

C-885: Controller names = C-885, Mode = Find controller by UDP, Port = 50000

C-886: Controller names = C-886, Mode = Find controller by UDP, Port = 50000

C-891: Controller names = C-891, Mode = Find XPort by UDP, Port = 30718

E-517: Controller names = E-517, Mode = Find controller by UDP, Port = 50000

E-518: Controller names = E-518, Mode = Find controller by UDP, Port = 50000

E-712: Controller names = E-712, Mode = Find controller by UDP, Port = 50000

E-725: Controller names = E-725, Mode = Find XPort by UDP, Port = 30718

E-725: Controller names = E-725, Mode = Find controller by UDP, Port = 50000

E-753: Controller names = E-753, Mode = Find controller by UDP, Port = 50000

E-754: Controller names = E-754, Mode = Find controller by UDP, Port = 50000

E-871: Controller names = E-871, Mode = Find XPort by UDP, Port = 30718

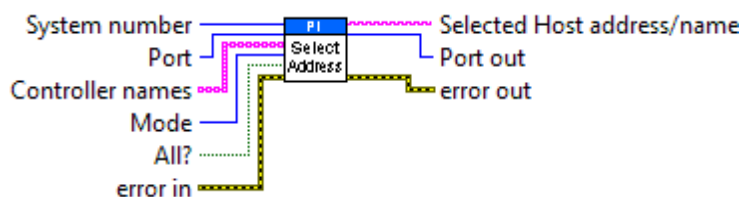
E-872: Controller names = E-872, Mode = Find controller by UDP, Port = 50000

E-873: Controller names = E-873, Mode = Find XPort by UDP, Port = 30718

E-874: Controller names = E-874, Mode = Find USR-K3 by UDP, Port = 1901

F-206: Controller names = F-206, F-HEX, Mode = Find XPort by UDP, Port = 30718

C-887, M-8X0: Controller names = M-8X0, HEXAPOD, Mode = Find XPort by UDP, Port = 30718

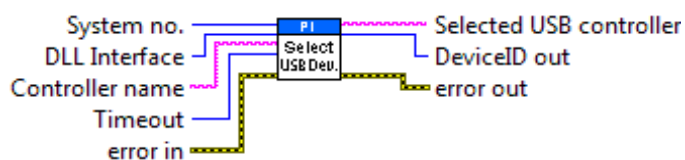


3.2.22. Select USB device.vi

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-886, 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-872, E-873, E-874, Mercury_GCS

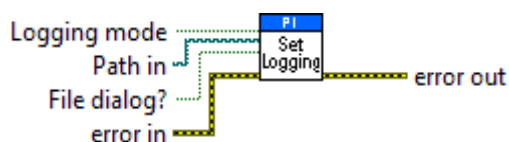
C-413: Controller name = "C-413", DLL Interface = USB
 C-866: Controller name = "C-866", DLL Interface = USB
 C-867: Controller name = "C-867", DLL Interface = USB
 C-877: Controller name = "C-877", DLL Interface = USB
 C-884: Controller name = "C-884", DLL Interface = USB
 C-885: Controller name = "C-885", DLL Interface = USB
 C-886: Controller name = "C-886", 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-872: Controller name = "E-872", DLL Interface = USB
 E-873: Controller name = "E-873", DLL Interface = USB
 E-874: Controller name = "E-874", DLL Interface = USB
 Mercury_GCS: Controller name = "Mercury_GCS", DLL Interface = USB



3.2.23. 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.

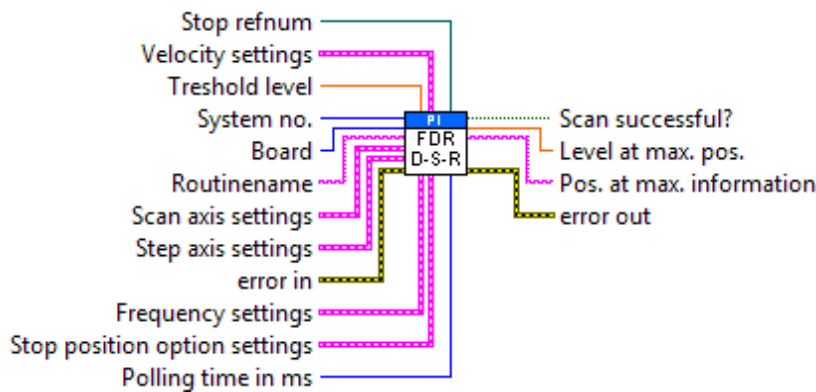


3.3. Controller Algorithms.IIb

3.3.1. FDR Def_Start_Results.vi

Defines a fast alignment area scan routine, starts scan, waits until scan has finished and queries results. If Stop refnum or Local stop is TRUE, VI sends #24 and stops.

Valid for C-887, E-712, E-727

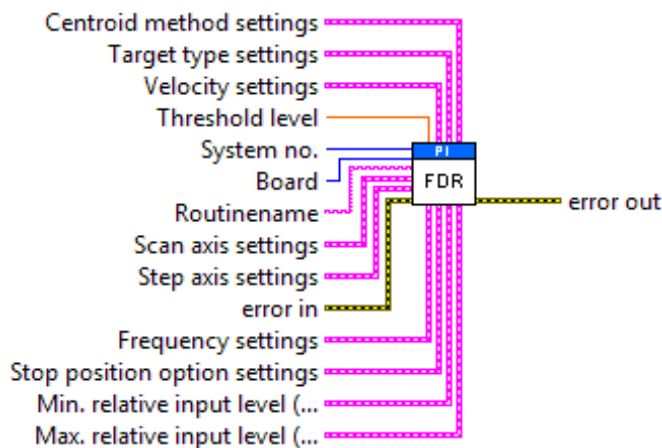


3.3.2. FDR.vi

Defines a fast alignment area scan routine. Use FRS to start the routine and FRR? to read out the results of the routine.

Valid for C-887, E-712, E-727

C-887, E-712, E-727: Check HLP? answer to find out if FDR is supported.

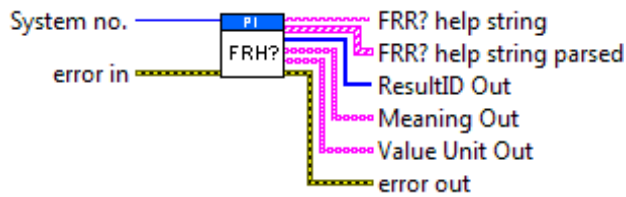


3.3.3. FRH?.vi

The command returns an ASCII help string with available query options ("result IDs") for the FRR? command.

Valid for C-887, E-712, E-727

C-887, E-712, E-727: Check HLP? answer to find out if FRH? is supported.



3.3.4. FRP.vi

Stops, pauses or resumes the given fast alignment routine(s).

Valid for C-887, E-712, E-727

C-887, E-727: Option can be

0 - stop given routine

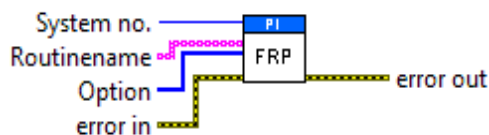
E-712: Option can be

0 - stop given routine

1 - pause given routine

2 - resume given routine

C-887, E-712, E-727: Check HLP? answer to find out if FRP is supported.



3.3.5. FRP?.vi

Returns current status of given fast alignment routine(s).

Valid for C-887, E-712, E-727

C-887, E-727: Status can be

0 - given routine has been stopped / is not running

2 - given routine is running

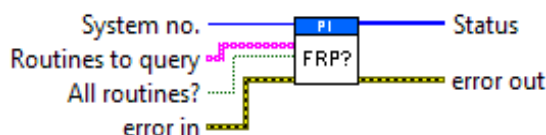
E-712: Status can be

0 - given routine has been stopped / is not running

1 - given routine has been paused

2 - given routine is running

C-887, E-712, E-727: Check HLP? answer to find out if FRP? is supported.



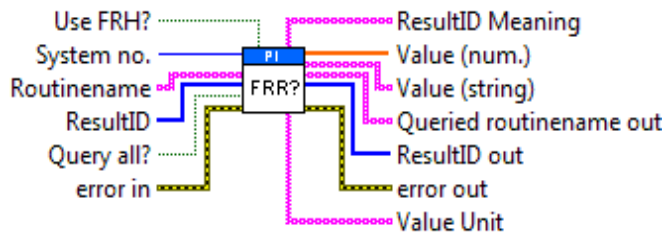
3.3.6. FRR?.vi

Get results of given fast alignment routine. If "Use FRH?" is TRUE, VI queries FRH? and returns meaning of ResultIDs and units of Values. Requires FRH? and Parse_FRH_Type_String.vi.

Valid for C-887, E-712, E-727

E-712, E-727: Query all? must be FALSE, Use FRH? can be TRUE.

C-887, E-712, E-727: Check HLP? answer to find out if FRR? is supported.

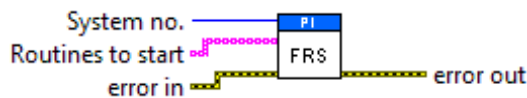


3.3.7. FRS.vi

Starts the given fast alignment routine (one or more). Routine must be predefined, e.g. with FDR or FDG.

Valid for C-887, E-712, E-727

C-887, E-712, E-727: Check HLP? answer to find out if FRS is supported.



3.3.8. SIC.vi

Set fast alignment Input Calculation.

Calculation Type:

0 = Calculation OFF

1 = Exponential Calculation

Calculated Input Signal = Parameter0 + Parameter1 *
Parameter2^(Parameter3*Voltage)

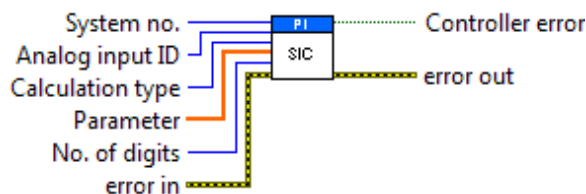
2 = Polynomial Calculation

Calculated Input Signal = Parameter0 + Parameter1*Voltage^1 +
Parameter2*Voltage^2 + Parameter3*Voltage^3 + Parameter4*Voltage^4)

Valid for C-887, E-712, E-727

C-887, E-712: Check HLP? answer to find out if SIC is supported.

E-727: Only calculation type 0 supported. Check HLP? answer to find out if SIC is supported.



3.3.9. SIC?.vi

Query Fast alignment Input Calculation

Signal input calculation Type:

0 = Calculation OFF

1 = Exponential Calculation:

Calculated Input Signal = Parameter0 + Parameter1 *
Parameter2^(Parameter3*Voltage)

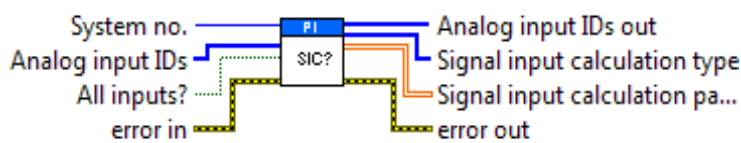
2 = Polynomial Calculation:

Calculated Input Signal = Parameter0 + Parameter1*Voltage^1 +
Parameter2*Voltage^2 + Parameter3*Voltage^3 + Parameter4*Voltage^4)

Valid for C-887, E-712, E-727

C-887, E-712: Check HLP? answer to find out if SIC? is supported.

E-727: Only calculation type 0 supported. Check HLP? answer to find out if SIC? is supported.

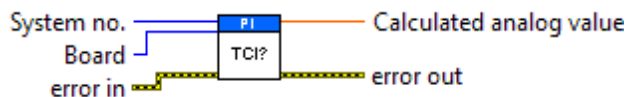


3.3.10. TCI?.vi

Queries Calculated fast alignment Input value. Answer is TAV? Value calculated with method set by SIC.

Valid for C-887, E-712, E-727

C-887, E-712, E-727: Check HLP? answer to find out if TCI? is supported.

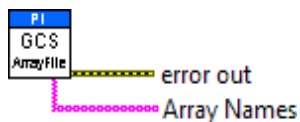


3.4. File handling.llb

3.4.1. 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, E-874, 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.

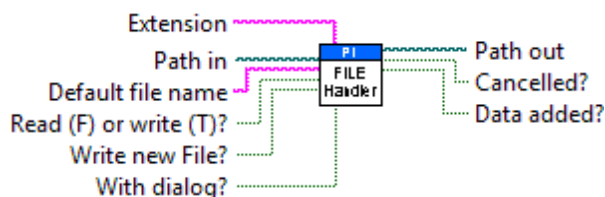


3.4.2. File handler.vi

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.

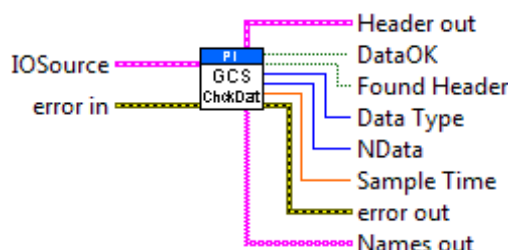
Valid for: All systems.



3.4.3. GetDataFormat.vi

This vi checks the format of a data file. Sub-VI for "Show_Save_Load_XY_Data.vi". See separate manual "GCSDData User SM146E.pdf" and control descriptions in the diagram for more information.

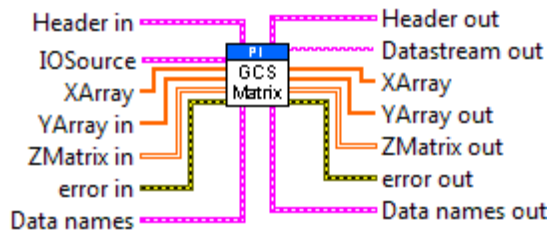
Valid for: Analog systems, Analog systems, C-413, C-702, C843, C-866, C-867, C-877, C-880, C-884, C-886, C-887, C-891, E-517, E-518, E-709, E-710, E-712, E-725, E-727, E-753, E-754, E-755, E-761, E-861, E-871, E-873, E-874, F-206, M-8X0, Mercury_GCS(but must be present for C-885 also). To support analog interfacing, VI must be present for E-816 also.



3.4.4. 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 "GCSDData_User_SM146E.pdf" and control descriptions in the diagram for more information.

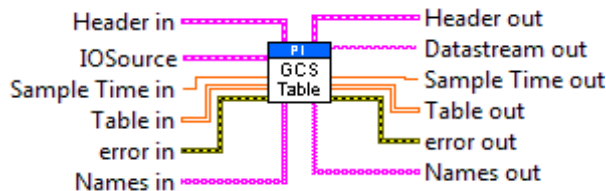
Valid for: Analog systems, C-413, C-702, C-843, C-866, C-867, C-877, C-880, C-884, C-886, C-887, C-891, E-517, E-518, E-709, E-710, E-712, E-725, E-727, E-753, E-754, E-755, E-761, E-861, E-871, E-873, E-874, F-206, M-8X0, Mercury_GCS(but must be present for C-885 also). To support analog interfacing, VI must be present for E-816 also.



3.4.5. TableIO.vi

This VI reads or writes data files in table format. Sub-VI for "DDL.vi""DRR?.vi", "HIT?", "GWD?.vi", "JLT?.vi", "TWS.vi" etc. See separate manual "GCSDATA User SM146E.pdf" and control descriptions in the diagram for more information. Sub-VI for "DRR?.vi"

Valid for: Analog systems, C-413, C-702, C-843, C-866, C-867, C-877, C-880, C-884, C-886, 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-872, E-873, E-874, 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.



3.5. General command.llb

3.5.1. *IDN?.vi

Returns system identification string.

Valid for all systems

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



3.5.2. CSV?.vi

Returns current GCS syntax version.

Valid for C-413, C-702, C-867, C-877, C-884, C-885, C-886, 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-872, E-873, E-874, 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).



3.5.3. 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

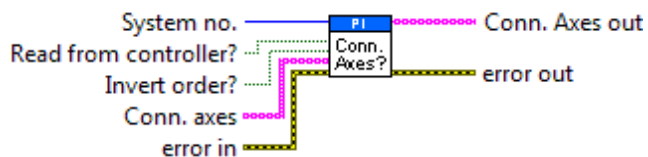
C-702: Read from controller = TRUE, Invert order = TRUE

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,
Connected axes = X,Y,Z,U,V,W, (A,B,K,L,M optional)

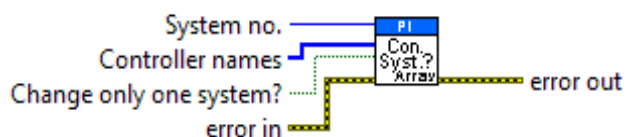
All other systems: Read from controller = TRUE, Invert order = FALSE

**3.5.4. 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

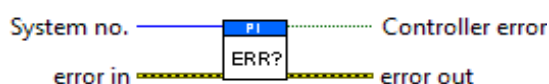
**3.5.5. ERR?.vi**

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.



3.5.6. 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.

**3.5.7. 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-886, 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-872, E-873, E-874, 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.

**3.5.8. 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-886, C-887, C-891, E-517, E-518, E-710, E-727, E-754, E-755, E-761, E-861, E-871, E-872, E-873, E-874, 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-886: If All axes? = TRUE, then Axis identifier? can be FALSE

C-891: If All axes? = TRUE, then Axis identifier? must be TRUE

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

E-518: If All axes? = TRUE, then Axis identifier? can be FALSE

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

E-727: If All axes? = TRUE, then Axis identifier? must be TRUE

E-754: If All axes? = TRUE, then Axis identifier? can be FALSE

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

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

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

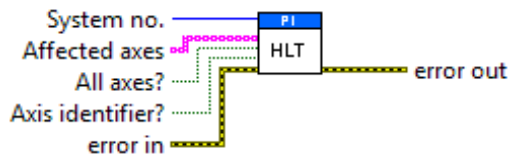
E-871, E-872, E-873, E-874: If All axes? = TRUE, then Axis identifier? can be FALSE

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

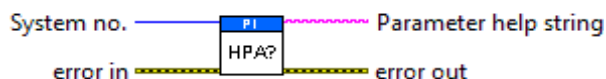


3.5.9. HPA?.vi

Returns a help string containing information about valid parameter IDs.

Valid for C-413, C-843, C-867, C-877, C-884, C-886, 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-872, E-873, E-874, 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).

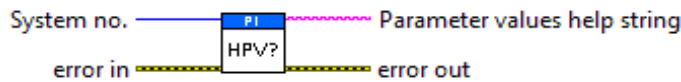


3.5.10. HPV?.vi

Returns a help string containing information about valid parameter values.

Valid for C-413, C-867, C-884, E-709, E-712, E-725, E-727, E-753, E-754, E-870, E-874 (but must be present for C-885 also)

C-867, C-884, E-709, E-712, E-725, E-753: Check HLP? answer to find out if HPV? is supported.



3.5.11. Initialize Global2.vi

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

Valid for all systems.



3.5.12. 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-886, 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, E-874, 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-886: Motion commands are not allowed when a wave generator is active.

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-755: Command not available for E-755.101.

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

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

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

E-871, E-873, E-874: This command works only in closed-loop operation. Motion commands like MOV are not allowed when a joystick is active on the axis.

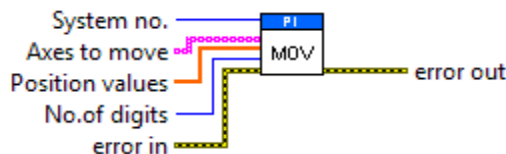
C-887, F-206, M-8X0: For GCS syntax version = GCS 1.0 (Check with CSV?.vi. If CSV?.vi is not supported, syntax version is GCS 1.0), no mix between F-206 axes X,Y,Z,U,V,W, separate axes A, B and NanoCube axes K, L, M in one call allowed. Subsequent MOV commands are handled in a buffer on controller, so 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.



3.5.13. 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-886, 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, E-874, 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

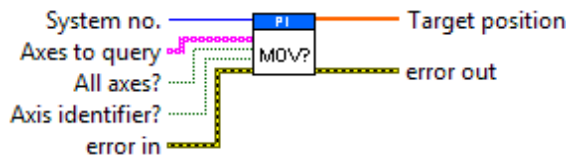
E-755: If All axes? = TRUE, then Axis identifier? can be FALSE. Command not available for E-755.101.

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



3.5.14. 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-886, 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, E-874, 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_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-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.llb .

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

E-725, 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-755: Command not available for E-755.101.

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

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

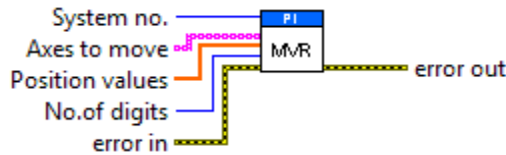
E-871, E-873, E-874: This command works only in closed-loop operation. Motion commands like MOV are not allowed when a joystick is active on the axis.

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.



3.5.15. ONT?.vi

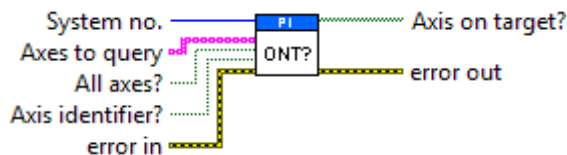
Indicates whether or not queried axis is at target position.

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-886, 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, E-874, 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.



3.5.16. 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-880K005, C-884, C-885, C-886, 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, E-874, 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 .

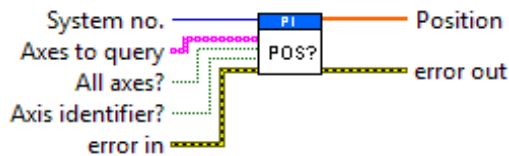
E-755: If All axes? = TRUE, then Axis identifier? can be FALSE. Command not available for E-755.101.

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.



3.5.17. PUN?.vi

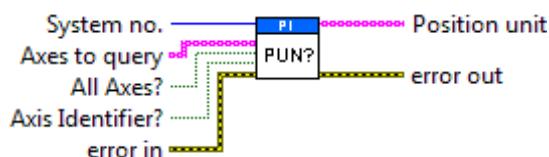
Returns the name of the connected stage for queried axes.

Valid for C-413, C-886, C-887, E-712, E-725, E-727, E-753, E-754, F-206, M-8X0

C-413, E-712, E-725, E-727, E-753: If All axes? = TRUE, then Axis identifier? can be FALSE. Check HLP? answer to find out if PUN? is supported.

C-886, E-754: 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 PUN? is supported. Only for GCS syntax version = GCS 2.0 or higher (check with CSV?.vi. If CSV?.vi is not supported, syntax version is GCS 1.0).



3.5.18. 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-886, 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-872, E-873, E-874, 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

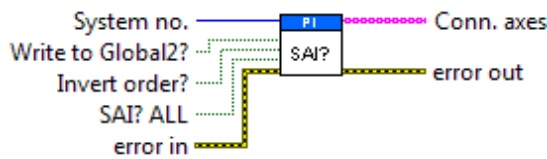
C-880: Invert order should be TRUE, SAI? ALL must be FALSE to read all configured axis IDs and must be TRUE to get all physically defined axis IDs

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



3.5.19. 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-886, 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-872, E-873, E-874, 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, C-886, E-516, E-517, E-518, C-891, E-710, E-712, E-725, E-727, E-753, E-754, E-755, E-861, E-870, E-871, E-872, E-873, E-874, 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. 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.

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.

C-891: Parameter no. format is TRUE (hex.). Use "HPA?.vi" to get valid parameter numbers or see the controller User Manual. 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.

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.

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

E-712: Parameter no. format is TRUE (hex.). Use "HPA?.vi" to get valid parameter numbers or see the E-712 User Manual. 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. 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.

E-754: Parameter no. format is TRUE (hex.). Use "HPA?.vi" to get valid parameter numbers or see the E-753 User Manual.

E-755: Parameter no. format is TRUE (hex.). Use "HPA?.vi" to get valid parameter numbers or see the E-755 User Manual.

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. 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. Each command is limited to set one parameter for only one channel.

E-871, E-872, E-873, E-874: 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.

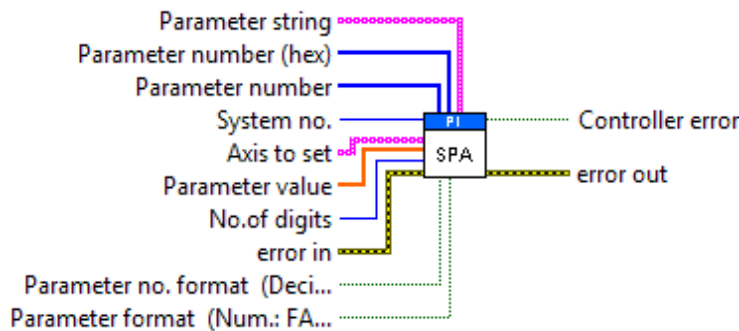
C-887, F-206, M-8X0: Parameter no. format is TRUE (hex.). Use "HPA?.vi" to get valid parameter numbers or see the controller User Manual. Check HLP?/HELP answer to find out if SPA is supported. Only for GCS syntax version = GCS 2.0 or higher (check with CSV?.vi. If CSV?.vi is not supported, syntax version 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.

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.

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



3.5.20. 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-886, 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-872, E-873, E-874, 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-872, E-873, E-874: 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.

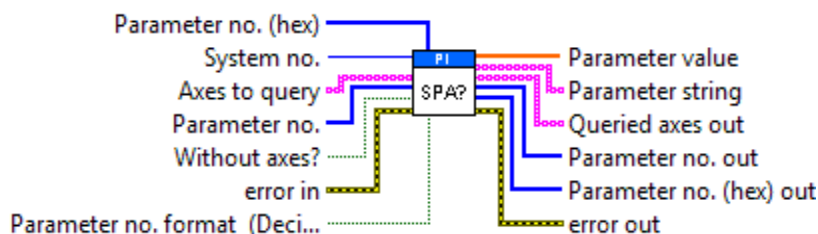
C-887, M-8X0: Check HLP?/HELP answer to find out if SPA? is supported.

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.



3.5.21. 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-886, C-887, C-891, E-516, E-517, E-518, E-709, E-712, E-725, E-727, E-753, E-754, E755, E-761, E-861, E-870, E-871, E-872, E-873, E-874, 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.

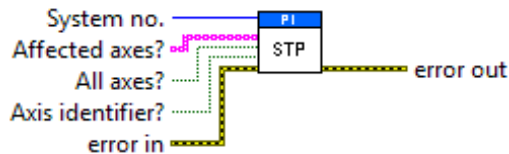
Analog: All axes? = TRUE, Axis identifier = FALSE. STP does not set any error code.

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.



3.5.22. 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-886, 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, E-874, 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.

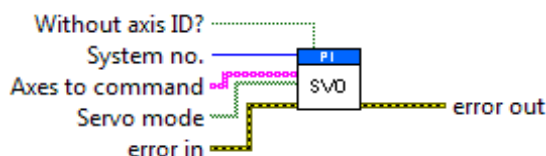
E-755: Without axis ID = FALSE. When the servo mode is switched off, RNP is automatically performed for the corresponding Nexline channel, which could take a few seconds. Command not available for E-755.101.

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

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

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



3.5.23. 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-886, 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, E-874, 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

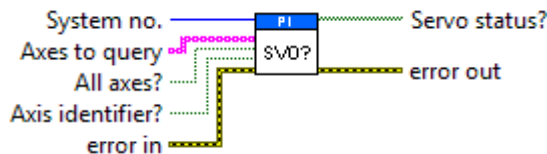
C-865, C-866: If All axes? = TRUE, then Axis identifier? must be TRUE

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



3.5.24. 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-886, 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, E-874, 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 $\mu\text{m}/\text{sec}$

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 $\mu\text{m}/\text{s}$. 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 -517, E-518: 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-709, E-712, E-725, E-727, E-753, E-754, E-861: Without axis ID? = FALSE. Velocity unit is $\mu\text{m/s}$. 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.

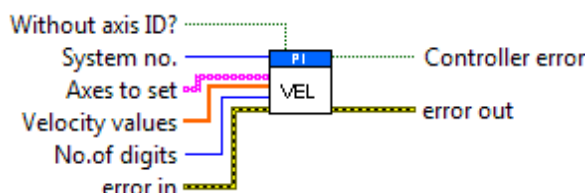
E-873, E-874: Without axis ID? = FALSE. Check HLP? answer to find out if VEL is supported. The VEL command saves the parameters in RAM only. To save the currently valid parameters to flash ROM, where they become the power-on defaults, you must run WPA.vi. Parameter changes not saved with WPA will be lost when the controller is powered off.

C-886: VEL command is only valid for velocity of additional axes. To set velocity of the parallel kinematics, please use the VLS command instead.

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



3.5.25. 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-886, 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, E-874, 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 $\mu\text{m/s}$

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.

C-886: If All axes? = TRUE, then Axis identifier? can be FALSE. VEL? only reports the velocity for additional axes. To query the velocity of the parallel kinematics (system velocity), please use the VLS? command instead.

E-517, E-518: If All axes? = TRUE, then Axis identifier? can be FALSE. Velocity unit is $\mu\text{m/s}$ 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 $\mu\text{m/ms}$.

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 $\mu\text{m/s}$.

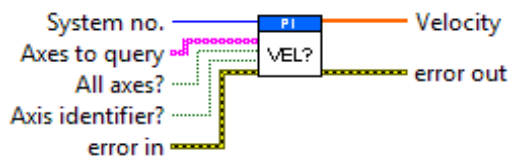
E-755: If All axes? = TRUE, then Axis identifier? can be FALSE. Velocity unit is $\mu\text{m/s}$. Command not available for E-755.101.

E-761: If All axes? = TRUE, then Axis identifier? can be FALSE. Velocity unit is $\mu\text{m/ms}$.

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, E-874: If All axes? = TRUE, then Axis identifier? can be FALSE. Check HLP? answer to find out if VEL? is supported.

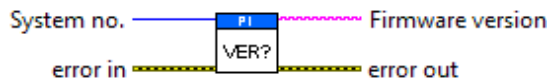


3.5.26. VER?.vi

Returns firmware string.

Valid for C-702, C-848, C-866, C-867, C-877, C-880, C-884, C-885, C-886, C-887, C-891, E-516, E-517, E-518, E-727, E-754, E-755, E-761, E-871, E-872, E-873, E-874, F-206, Hydra, Pollux, M-8X0, Mercury, Mercury_GCS

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

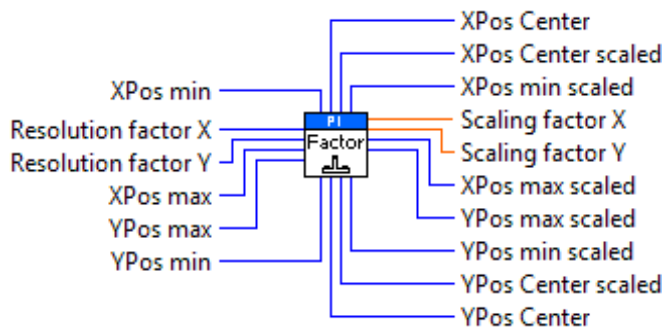


3.6. Joystick.Ilb

3.6.1. 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)

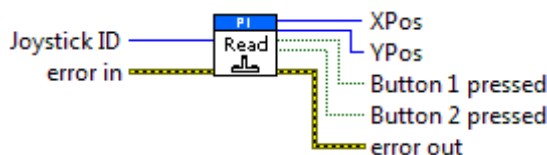


3.6.2. Read joystick.vi

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.

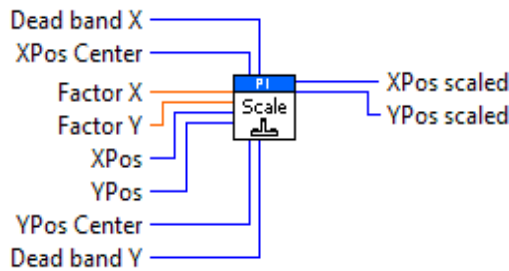
Valid for C-413, C-702, C-843, C-843.PM, C-844, C-848, C-865, C-866, C-867, C-877, C-880, C-884, C-891, E-517, E-518, E-709, E-710, E-712, E-725, E-727, E-753, E-754, E-755, E-761, E-861, Hydra, Pollux, Mercury_GCS(but must be present for C-885 also)



3.6.3. 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)



3.7. Limits.IIb

3.7.1. ATZ.vi

Performs an automatic zero-point calibration for the specified linear axes (see User Manual for details), waits until this procedure has finished and returns whether ATZ was successful or not. Select NaN as Low voltage parameter if you want the controller to use the stored values it has. VI will also stop if Stop refnum or Local stop is TRUE. The home position is reset to default by ATZ. ATZ works independent of servo mode.

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

C-413 : The result of the AutoZero procedure (new offset value) is present in RAM only. To save the result to EEPROM, where it becomes the power-on default, you must run WPA.vi.

E-709: The result of the AutoZero procedure (new offset value) is present in RAM only. To save the result to EEPROM, where it becomes the power-on default, you must run WPA.vi. For stages with ID-chip the option "Power Up Read ID-Chip" must be disabled by default to make the AutoZero result available in the future. See E-709 User Manual for details.

E-712: The result of the AutoZero procedure (new offset value) is present in RAM only. To save the result to EEPROM, where it becomes the power-on default, you must run WPA.vi. For stages with ID-chip the option "Power Up Read ID-Chip" must be disabled by default to make the AutoZero result available in the future. See E-712 User Manual for details.

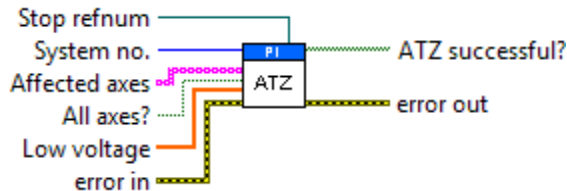
E-725, E-727: The result of the AutoZero procedure (new offset value) is present in RAM only. To save the result to EEPROM, where it becomes the power-on default, you must run WPA.vi. For stages with ID-chip the option "Power Up Read ID-Chip" must be disabled by default to make the AutoZero result available in the future. See controllerUser Manual for details.

E-753: The result of the AutoZero procedure (new offset value) is present in RAM only. To save the result to EEPROM, where it becomes the power-on default, you must run WPA.vi. For stages with ID-chip the option "Power Up Read ID-Chip" must be disabled by default to make the AutoZero result available in the future. See E-753 User Manual for details.

E-754: The result of the AutoZero procedure (new offset value) is present in RAM only. To save the result to EEPROM, where it becomes the power-on default, you must run WPA.vi. For stages with ID-chip the option "Power Up Read ID-Chip"

must be disabled by default to make the AutoZero result available in the future. See E-754 User Manual for details.

E-761: Be aware that the result of the AutoZero procedure (new offset value) is automatically written to non-volatile memory (EPROM). For stages with ID-chip the option "Read ID-Chip always" must be disabled by default to make the AutoZero result (new offset value) available in the future. See E-761 User Manual for details.

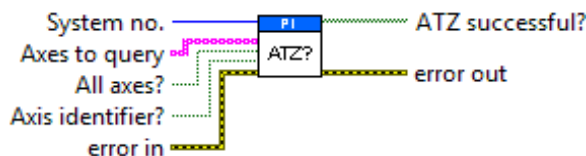


3.7.2. ATZ?.vi

Returns if ATZ was successful or not for queried axes.

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

C-413, E-709, E-712, E-725, E-727, E-753, E-754: If All axes? = TRUE, then Axis identifier? can be FALSE



3.7.3. TMN?.vi

Returns minimum (low-end) travel limit (if present, position of negative limit switch, or value of negative soft limit, if set, whichever is higher).

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-886, 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, E-874, 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, E-874, 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

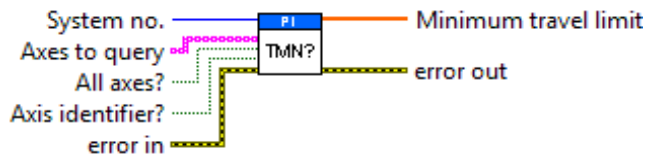
C-413, C-867, C-877, C-884, C-891, E-861, E-871, E-873, E-874, Hydra, Pollux, Mercury_GCS: If All axes? = TRUE, then Axis identifier? can be FALSE. The minimum commandable position is defined by the MAX_TRAVEL_RANGE_NEG parameter ID 0x30 (SPA).

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.



3.7.4. 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-886, 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, E-874, 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, E-874, F-206, Hydra, Pollux, M-8X0, Mercury, Mercury_GCS: If All axes? = TRUE, then Axis identifier? can be FALSE

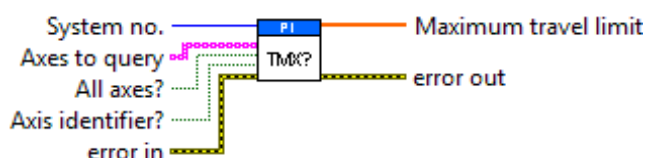
C-843, C-843.PM, C-844, E-710: If All axes? = TRUE, then Axis identifier? must be TRUE

C-413, C-867, C-877, C-884, E-861, E-871, E-873, E-874, 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.



3.8. Macros.IIb

3.8.1. #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-886, C-887, C-891, E-516, E-517, E-518, E-727, E-754, E-816, E-861, E-871, E-872, E-873, E-874, 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).



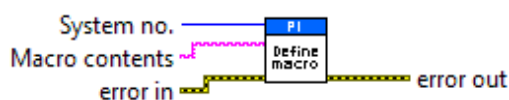
3.8.2. Define macro contents.vi

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-886, C-887, C-891, E-517, E-518, E-727, E-754, E-816, E-861, E-871, E-872, E-873, E-874, 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).



3.8.3. 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-886, C-887, C-891, E-516, E-517, E-518, E-727, E-754, E-816, E-861, E-871, E-872, E-873, E-874, 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-886, C-887, C-891, E-516, E-517, E-518, E-727, E-754, E-816, E-871, E-872, E-873, E-874, 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).



3.8.4. MAC DEF.vi

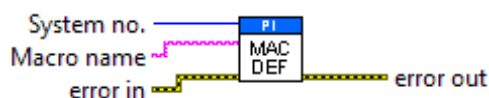
Define autostart macro. To disable the autostart macro, "Macro name" must be an empty string.

Valid for C-867, C-877, C-884, C-886, C-887, C-891, E-516, E-517, E-518, E-727, E-754, E-816, E-861, E-871, E-872, E-873, E-874, 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).



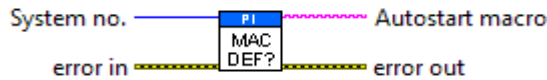
3.8.5. MAC DEF?.vi

Get name of autostart macro.

Valid for C-867, C-877, C-884, C-886, C-887, C-891, E-516, E-517, E-518, E-727, E-754, E-816, E-861, E-871, E-872, E-873, E-874, 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).



3.8.6. 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-886, C-887, C-891, E-516, E-517, E-518, E-727, E-754, E-816, E-861, E-871, E-872, E-873, E-874, 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 BEG 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).



3.8.7. MAC END.vi

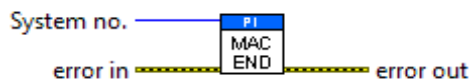
Stops current macro recording.

Valid for C-702, C-844, C-848, C-867, C-877, C-880, C-884, C-886, C-887, C-891, E-516, E-517, E-518, E-727, E-754, E-816, E-861, E-871, E-872, E-873, E-874, 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).

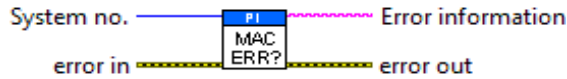


3.8.8. MAC ERR?.vi

Returns error information from macro execution

Valid for C-867, C-877, C-884, C-886, C-887, C-891, E-727, E-871, E-872, E-873, E-874, 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.



3.8.9. MAC FREE?.vi

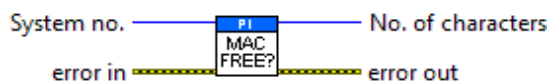
Returns the amount of free memory for macro recording (in number of characters).

Valid for C-867, C-886, C-887, E-516, E-517, E-518, E-727, E-754, E-816, F-206, M-8X0 (but must be present for C-885 also)

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

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

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



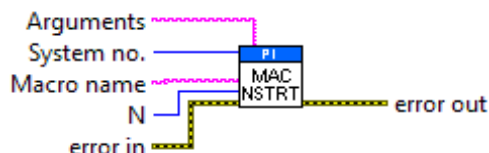
3.8.10. MAC NSTART.vi

Start macro N times.

Valid for C-702, C-867, C-877, C-884, C-886, C-887, C-891, E-516, E-517, E-518, E-727, E-754, E-816, E-861, E-871, E-872, E-873, E-874, 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.



3.8.11. MAC START.vi

Start macro.

Valid for C-702, C-844, C-848, C-867, C-877, C-880, C-884, C-886, C-887, C-891, E-516, E-517, E-518, E-727, E-754, E-816, E-861, E-871, E-872, E-873, E-874, F-206, M-8X0, Mercury, Mercury_GCS (but must be present for C-885 also)

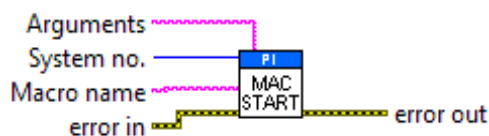
C-702, C-848, C-867, C-877, C-880, C-886, C-891, E-516, E-517, E-518, E-727, E-754, E-861 E-871, E-872, E-873, E-874: Use #8.vi to determine when macro execution has finished. For C-702, C-848, C-880, E-516 and E-517, E-518 Arguments is not supported.

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.



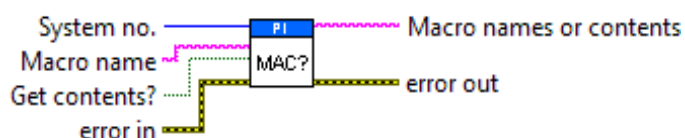
3.8.12. MAC?.vi

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-886, C-887, C-891, E-516, E-517, E-518, E-727, E-754, E-754, E-816, E-861, E-871, E-872, E-873, E-874, 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).

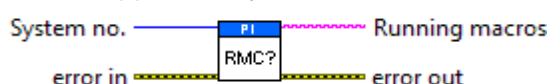


3.8.13. RMC?.vi

List macros currently running.

Valid for C-867, C-877, C-884, C-886, C-887, C-891, E-727, E-861, E-871, E-872, E-873, E-874, 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).



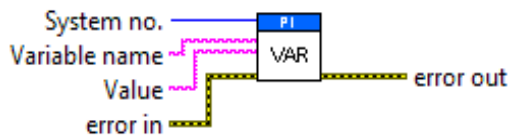
3.8.14. VAR.vi

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-886, C-887, C-891, E-727, E-871, E-872, E-873, E-874, 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.



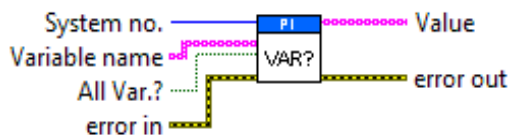
3.8.15. VAR?.vi

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-886, C-887, C-891, E-727, E-871, E-872, E-873, E-874, 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.



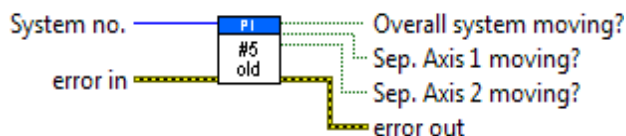
3.9. Old commands.llb

3.9.1. #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.



3.9.2. Wait for hexapod system axes to stop.vi

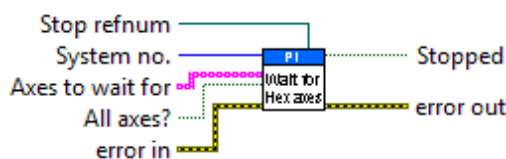
This vi waits for the specified axes of a PI hexapod system (hexapod axes X, Y, Z, U, V, W and separate axes A, B) to stop using #5 polling. If a NanoCube axis (K, L or M) is commanded, the VI will return immediately. If one of the hexapod axes (X, Y, Z, U, V or W) is commanded, it will wait for all 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



3.10. Optical or Analog Input.IIb

3.10.1. TAD?.vi

Returns AD value for the specified sensor number.

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, E-870, F-206, M-8X0 (but must be present for C-885 also)

C-413 : If All sensors? = TRUE, then Sensor identifier? must be FALSE. Sensors to query comprises all ADC channels of the device: can be a "genuine" sensor (sensor integrated in the mechanics) or a "general purpose" analog input.

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

E-709: 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 for 4-channel controller only and in command level 1 only (see "CCL.vi" and "CCL?.vi")

E-712: If All sensors? = TRUE, then Sensor identifier? must be FALSE. Sensors to query comprises all ADC channels of the device: can be a "genuine" sensor (sensor integrated in the mechanics) or a "general purpose" analog input.

E-725, E-727: If All sensors? = TRUE, then Sensor identifier? must be FALSE. Sensors to query comprises all ADC channels of the device: can be a "genuine" sensor (sensor integrated in the mechanics) or a "general purpose" analog input.

E-753: 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) and the "general purpose" analog input (identifier = 2).

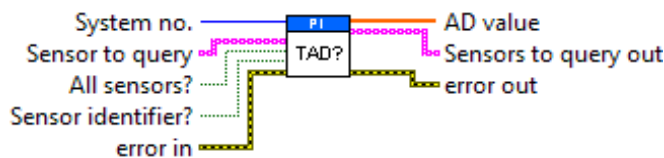
E-754: If All sensors? = TRUE, then Sensor identifier? must be FALSE. Sensors to query comprises all ADC channels of the device: can be the "genuine" sensor (capacitive sensor integrated in the mechanics, identifier = 1, or incremental sensor, identifier = 3) and the "general purpose" analog input (identifier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

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

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

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

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



3.10.2. 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-727, E-761, E-861, E-871, E-873, E-874, 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, E-874, Mercury_GCS: Query = Value, Board = 1 to 4. Range and Power unit are not valid.

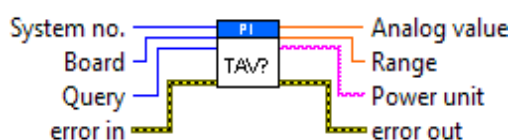
C-880: Query = Value. Range and Power unit are not valid

E-712, E-727: 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



3.10.3. TNS?.vi

Returns normalized sensor value for the specified sensor number.

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

C-413 : If All sensors? = TRUE, then Sensor identifier? must be FALSE. Sensors to query comprises all ADC channels of the device: can be a "genuine" sensor (sensor integrated in the mechanics) or a "general purpose" analog input.

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" and "CCL?.vi")

E-712: If All sensors? = TRUE, then Sensor identifier? must be FALSE. Sensors to query comprises all ADC channels of the device: can be a "genuine" sensor (sensor integrated in the mechanics) or a "general purpose" analog input.

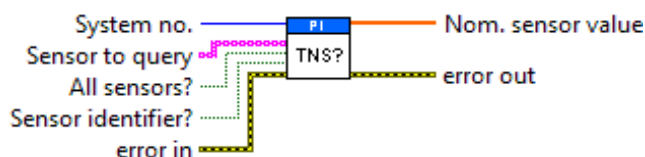
E-725, E-727: If All sensors? = TRUE, then Sensor identifier? must be FALSE. Sensors to query comprises all ADC channels of the device: can be a "genuine" sensor (sensor integrated in the mechanics) or a "general purpose" analog input.

E-753: 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) 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? must be FALSE. Command not available for E-755.101.

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



3.10.4. 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-884, C-885, C-891, E-516, E-816, E-861, E-870, E-871, E-872, E-873, E-874, 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 (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.



3.10.5. 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")

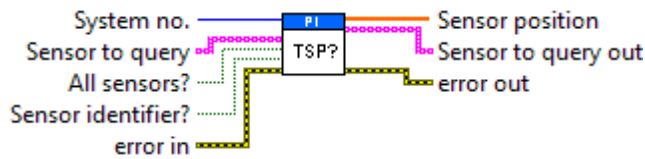
C-413, E-712, E-725, E-727: If All sensors? = TRUE, then Sensor identifier? can be FALSE. Sensors to query comprises all ADC channels of the device: can be a "genuine" sensor (sensor integrated in the mechanics) or a "general purpose" analog input.

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.



3.11. PZT voltage.llb

3.11.1. DPO.vi

DDL processing parameter correction for specified axis. Calculates internal DDL processing parameters. DPO is required when servo parameters have changed for an axis.

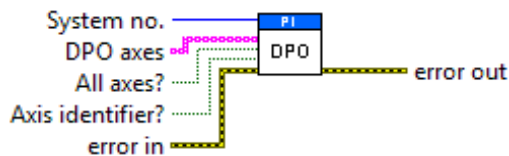
Valid for E-710, E-712, E-725, E-727, E-753, E-754

E-710: If All axes = TRUE, then Axis identifier must be TRUE. Command is available in command level 1 only (see "CCL.vi" and "CCL?.vi")

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

E-725, E-727: If All axes? = TRUE, then Axis identifier? can be FALSE

E-753, E-754: If All axes = TRUE, then Axis identifier? can be FALSE.



3.11.2. DTC.vi

Clears DDL table.

Valid for E-710, E-712, E-725, E-727, E-753, E-754, E-761

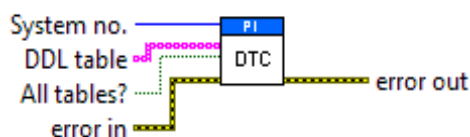
E-710: "All tables?" must be FALSE.

E-712: All tables? must be FALSE.

E-725, E-727: All tables? must be FALSE.

E-753, E-754: "All tables?" must be FALSE.

E-761: "All tables?" must be FALSE.



3.11.3. OVF?.vi

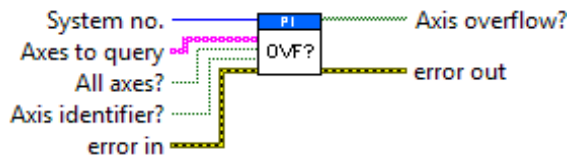
Returns overflow information for queried axes.

Valid for C-413, E-516, E-517, E-518, E-709, E-712, E-725, E-727, E-753, E-754, E-761, E-816(but must be present for C-885 also)

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

C-413, E-517, E-518, E-709, E-712, E-725, E-727, E-753, E-754, E-761: If All axes? = TRUE, then Axis identifier? can be FALSE

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



3.11.4. SVA.vi

Sets absolute PZT voltage for specified axes. Servo must be switched off (using "SVO.vi" before using this command. No. of digits is the number of digits after the decimal point in the voltage value(s) that will be sent.

Valid for C-413, C-867, 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-874 (but must be present for C-885 also)

E-517, E-518: Motion commands like SVA 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, SVA 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: Motion commands are not allowed when a wave generator is active or the analog input is used for target generation.

E-712: For axes driven by conventional piezo actuators, PZT voltage is a dimensionsless value whose range corresponds approximately to the mechanics travel range in μm .

For axes driven by NEXLINE® drives, PZT voltage is given in volts. SVA affects only the shearing segments of the NEXLINE® stack actuators. Hence SVA will cause motion in analog mode only.

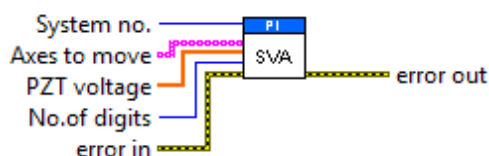
Motion commands are not allowed when a wave generator is active or the analog input is used for target generation.

E-725, E-727, E-753, E-754: PZT voltage is a dimensionsless value whose range corresponds approximately to the mechanics travel range in μm .

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

E-761: PZT voltage is a dimensionsless value whose range corresponds approximately to the mechanics travel range in μm . Motion commands are not allowed when a wave generator is active.

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.



3.11.5. SVA?.vi

Returns commanded PZT voltage for queried axes.

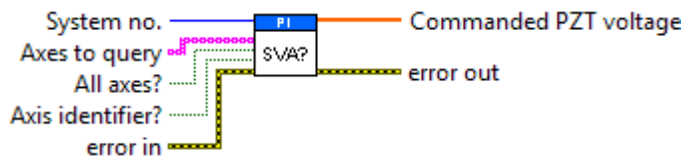
Valid for C-413, E-516, E-517, E-518, E-709, E-710, E-712, E-725, E-727, E-753, E-754, E-755, E-761, E-816, E-874 (but must be present for C-885 also)

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

C-413, E-517, E-518, E-709, E-712, E-725, E-727, E-753, E-754, E-755, E-761, E-874: If All axes= TRUE, then Axis identifier? can be FALSE.

E-712, E-725, E-727, E-753, E-754, E-761: The response is a dimensionsless value whose range corresponds approximately to the mechanics travel range in μm .

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



3.11.6. SVR.vi

Sets relative PZT voltage for specified axes. Servo must be switched off (using "SVO.vi") before using this command. No. of digits is the number of digits after the decimal point in the voltage value(s) that will be sent.

Valid for C-413, 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 (but must be present for C-885 also)

E-517, E-518: Motion commands like SVR 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, SVR 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: Motion commands are not allowed when a wave generator is active or the analog input is used for target generation.

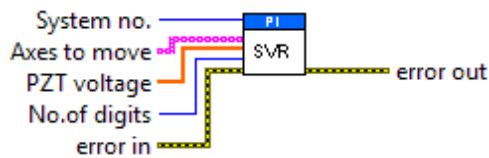
E-712: SVR is valid for axes driven by conventional piezo actuators only. PZT voltage is a dimensionsless value whose range corresponds approximately to the mechanics travel range in μm . Motion commands are not allowed when a wave generator is active or the analog input is used for target generation.

E-725, E-727: PZT voltage is a dimensionsless value whose range corresponds approximately to the mechanics travel range in μm . Motion commands are not allowed when a wave generator is active or the analog input is used for target generation.

E-753, E-754: PZT voltage is a dimensionsless value whose range corresponds approximately to the mechanics travel range in μm . Motion commands are not allowed when the wave generator is active or the analog input is used for target generation.

E-761: PZT voltage is a dimensionsless value whose range corresponds approximately to the mechanics travel range in μm . Motion commands are not allowed when a wave generator is active.

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.



3.11.7. VCO.vi

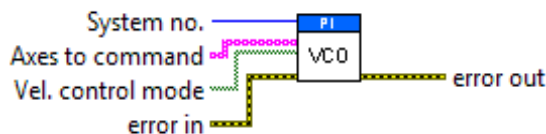
Sets velocity-control mode for specified axes.

Valid for Analog systems, E-516, E-517, E-518, E-727, E-754, E-761 (but must be present for C-885 also). To support analog interfacing, VI must be present for E-709, E-712, E-725 and E-816 also.

Analog: Velocity control mode can only be set for all axes equally. Therefore only first field of Vel.control mode is valid.

E-516, E-517, E-518, E-727, E-754: The VCO 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-761: The VCO command saves the parameters in RAM only. To save the currently valid parameters to flash ROM, where they become the power-on defaults, you must run WPA.vi with "Affected axes" as an empty array. Parameter changes not saved with WPA will be lost when the PC is powered off or the E-761 is rebooted.



3.11.8. VCO?.vi

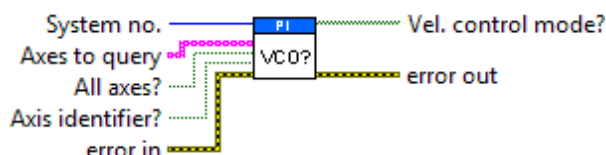
Returns velocity-control mode status for queried axes.

Valid for Analog systems, E-516, E-517, E-518, E-727, E-754, E-761 (but must be present for C-885 also). To support analog interfacing, VI must be present for E-709, E-712, E-725 and E-816 also.

Analog: If All axes = TRUE, then Axis identifier can be FALSE. Only first field of Vel.control mode is valid.

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

E-517, E-518, E-727, E-754, E-761: If All axes = TRUE, then Axis identifier can be FALSE



3.11.9. VOL.vi

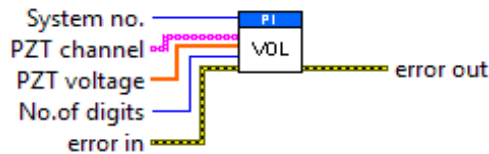
Sets absolute PZT voltage for specified piezo channel. No. of digits is the number of digits after the decimal point in the voltage value(s) that will be sent. If the

commanded voltage exceeds the voltage limits of the piezo channel, then the command is not executed.

Valid for Analog systems, E-761 (but must be present for C-885 also). To support analog interfacing, VI must be present for E-709, E-712, E-725, E-727 and E-816 also.

Analog system: PZT channel is identical with axis ID.

E-761: PZT channel can be 1 to 4.



3.11.10. VOL?.vi

Returns current PZT voltage for queried axes / piezo channels.

Valid for Analog systems, C-413, 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 (but must be present for C-885 also)

Analog: If All axes? = TRUE, then Axis identifier? can be FALSE. VI reads control voltage.

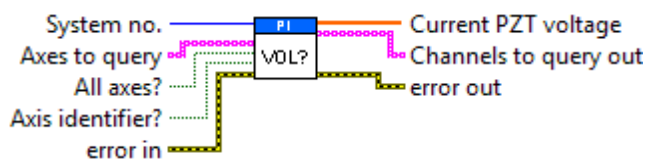
E-516: If All axes? = TRUE, then Axis identifier? must be TRUE

E-710: If All axes? = TRUE, then Axis identifier? must be TRUE. Axes to query are piezo channel numbers.

C-413, E-517, E-518, E-709, E-712, E-725, E-727, E-753, E-754, E-755: If All axes? = TRUE, then Axis identifier? can be FALSE. Axes to query are piezo channel numbers.

E-761: If All axes? = TRUE, then Axis identifier? can be FALSE. Axes to query are piezo channel numbers, which can be 1 to 4.

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



3.12. Scan support.Ilb

3.12.1. 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-886, 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, E-874, 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.



3.13. Special command.llb

3.13.1. #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-886, 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-872, E-873, E-874, 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.

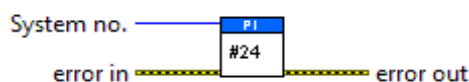
Analog systems: #24 does not set any error code. When used with any digital controller, does not influence connection between selected analog input channel and axis.

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.



3.13.2. #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-886, 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-872, E-873, E-874, 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.



3.13.3. #7.vi

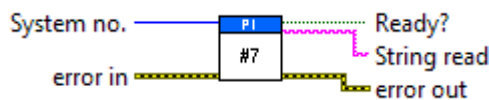
Sends the single ASCII character 7 and returns the ready status of the controller. Sub-VI for "Wait for answer of longlasting command.vi".

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-886, 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-872, E-873, E-874, 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).

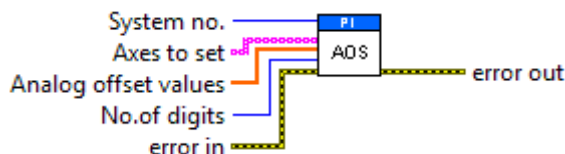
E-712: This VI is to be used during reference moves only. Check HLP? answer to find out if #7 is supported.



3.13.4. AOS.vi

Set Analog Input Offset. This command adds an offset value to an input value of an analog input, which is configured as analog target position for the selected axis. The target position of the selected axis is Target = Analog Input + Offset. No. of digits is the number of digits after the decimal point in the position value(s) that will be sent.

Valid for C-413, E-709, E-712, E-725, E-727, E-753, E-754 (but must be present for C-885 also)

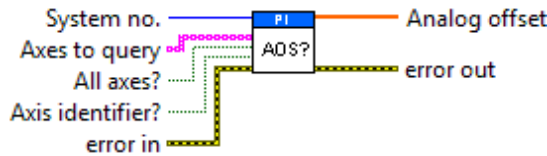


3.13.5. AOS?.vi

Returns analog input offset, which was set by AOS or by a parameter command.

Valid for C-413, E-709, E-712, E-725, E-727, E-753, E-754 (but must be present for C-885 also)

C-413, E-709, E-712, E-725, E-727, E-753, E-754: If All axes? = TRUE, then Axis identifier? can be FALSE



3.13.6. CCL.vi

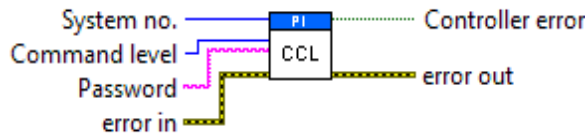
If password is correct, this vi sets the command level of the controller and queries ERR?. Controller error is TRUE if selected system report error code not equal to zero. Use "HLP?.vi" to determine which commands are available in the current command level.

Valid for C-413, C-867, C-877, C-884, C-886, 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-872, E-873, E-874, F-206, M-8X0, Mercury_GCS (but must be present for C-885 also)

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-870, E-871, E-872, E-873, E-874, 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, C-884, 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).



3.13.7. CCL?.vi

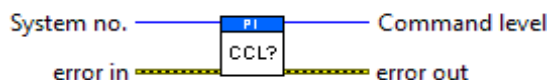
Returns the current command level.

Valid for C-413, C-702, C-867, C-877, C-884, C-886, 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-872, E-873, E-874, F-206, M 8X0, Mercury_GCS (but must be present for C-885 also)

C-867, C-884, 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).



3.13.8. CST?.vi

Returns the name of the connected stage for queried axes.

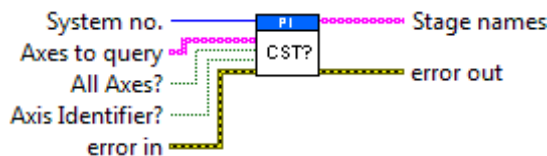
Valid for C-413, C-702, C-843, C-843.PM, C-844, C-848, C-865, C-866, C-867, C-877, C-880, C-884, C-885, C-886, C-887, C-891, E-709, E-710, E-712, E-725, E-727, E-753, E-754, E-755, E-761, E-861, E-871, E-872, E-873, E-874, F-206, Hydra, Pollux, M-8X0, Mercury, Mercury_GCS

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, E-874, 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.



3.13.9. 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, E-874, 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, E-874: 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 μ s 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 0x0E000900. Possible values are in the range of 10 to 150 μ s.

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.

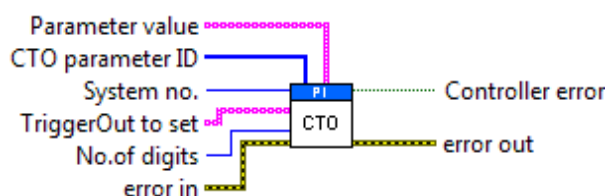
E-727: TriggerOut to set can be 1 to 3. The trigger output is immediately active when the condition given by the CTO configuration is fulfilled (there is no TRO command). If the CTO Parameter ID is "Trigger Mode" and the Parameter value is "Generator Trigger", then the trigger points in the waveform 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.



3.13.10. 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, E-874, 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, E-874: 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.

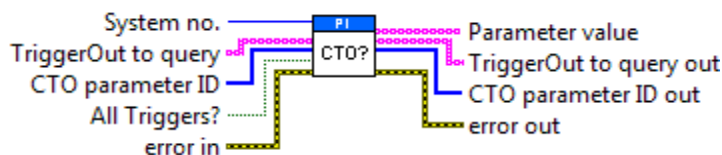
E-727: TriggerOut to query can be 1 to 3.

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.



3.13.11. DEL.vi

Delays the command interpreter of specified system for given

"Delay time". DEL is used within macros primarily. Do not mistake MAC DEL which deletes macros for DEL which delays.

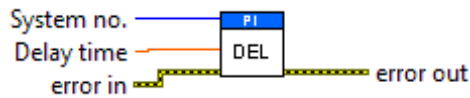
Valid for C-702, C-848, C-880, C-884, C-886, C-887, C-891, E-516, E-517, E-518, E-727, E-816, E-871, E-872, E-873, E-874, F-206, M-8X0, Mercury, Mercury_GCS (but must be present for C-885 also)

All systems: Delay time unit is ms.

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

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

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



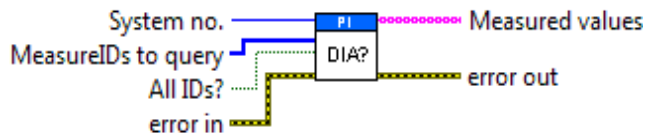
3.13.12. DIA?.vi

Returns diagnosis information. Use "HDI?.vi" to find out valid MeasureIDs and their meanings.

Valid for C-867, C-886, C-887, C-891, E-727, E-754, E-870

C-867, C-887, C-891: Check HLP? answer to find out if DIA? is supported.

E-870: If All ID's? = FALSE, only one MeasureID to query per command allowed.



3.13.13. 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-872, E-873, E-874, 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, E-874, 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. . If All DI's = FALSE, only one DI 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.

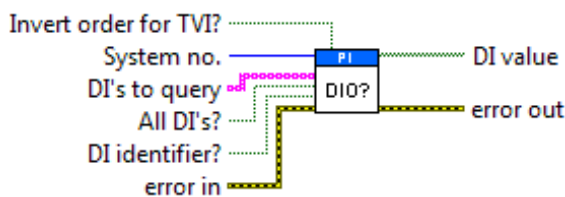
E-761: All DI's = FALSE. DI's to query are "1". Query pattern? is not valid. Note that the E-761 has no genuine digital input lines, but the analog input is internally interpreted as digital input for triggering tasks (see E-761 User Manual), and its signal state can be queried by this command. If the 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).

E-872: If All DI's = TRUE, then DI identifier must be FALSE. Invert order for TVI? is not valid. DI's can be queried with TIO?.vi. Query pattern? can be TRUE.

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.



3.13.14. 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-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, E-874, 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-727, 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.

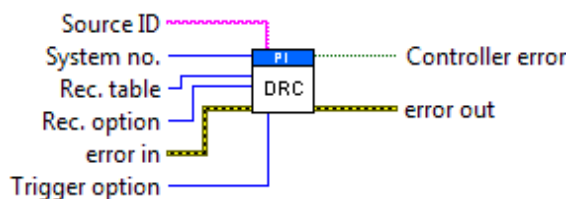
E-861: Trigger option must be 0. See User Manual for available record options. The E-861 has two data recorder tables with 1024 points per table.

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.

E-874: Trigger option must be 0. See User Manual for available record options. The E-874 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.



3.13.15. DRC?.vi

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

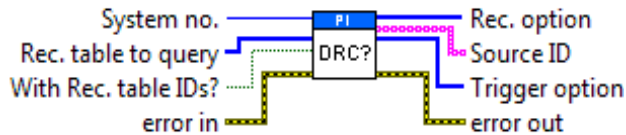
GCS 2.0: Trigger option is not valid.

Valid for C-413, C-702, C-843, C-866, C-867, C-877, C-884, C-886, 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, E-874, 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-886, 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, E-874, 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).



3.13.16. DRL?.vi

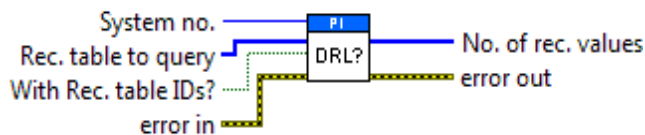
This VI returns the number of recorded data values for the given record tables.

Valid for C-413, C-867, C-877, C-884, C-886, C-887, C-891, E-518, E-709, E-712, E-725, E-727, E-753, E-754, E-871, E-873, E-874, 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).



3.13.17. 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-886, 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, E-874, 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: $X0 \geq 1$. 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: $X0 \geq 1$. $N_{max} = 32,256$. If $N = -1$ all points of the last record are returned.

C-867: $X0 \geq 1$. $N_{max} = 8192$. The number of tables is 4.

C-877: $X0 \geq 1$. $N_{max} = 8192$. The number of tables is 4.

C-884: $X0 \geq 1$. $N_{max} = 8192$. The number of tables is 8.

C-891: $X0 \geq 1$. $N_{max} = 16384$. 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: $X0 \geq 1$. $N_{max} = 8192$. The number of tables is 3.

E-709: $X0 \geq 1$. $N_{max} = 4096$. 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: $X0 \geq 1$. $N_{max} = 32,256$.

E-712: $X0 \geq 1$. $N_{max} = 262,144$. 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: $X0 \geq 1$. $N_{max} = 262,144$. 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 controller User Manual for details.

E-753: $X0 \geq 1$. $N_{max} = 65,536$. 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 E-753 User Manual for details.

E-754: $X0 \geq 1$. $N_{max} = 1,048,576$. 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: $X0 \geq 1$. $N_{max} = 4,096$.

E-761: $X0 \geq 0$. 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: $X0 \geq 1$. $N_{max} = 1024$. Two data recorder tables with 1024 points per table are provided.

E-871, E-873, E-874: $X_0 \geq 1$. $N_{max} = 1024$. The number of tables is 2.

C-887, F-206: $X_0 \geq 1$. See C-887 User Manual for N_{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.

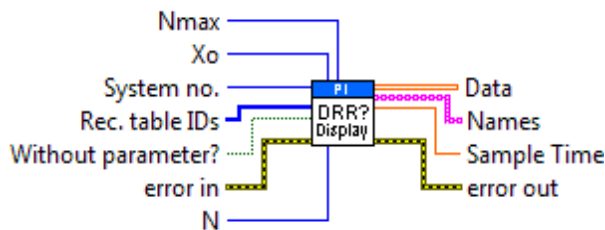
C-887, M-8X0: For GCS syntax version = 1.0 (Check with CSV?.vi. If CSV?.vi is not supported, syntax version is GCS 1.0), Rec. table IDs, x_0 , N and N_{max} are not valid and Without parameter? must be TRUE.

For GCS syntax version = 2.0, $X_0 \geq 1$. See C-887 User Manual for N_{max} 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: $X_0 \geq 1$. $N_{max} = 1024$. Two data recorder tables with 1024 points per table are provided.



3.13.18. DRR?.vi

Returns N recorded data points. If N is greater than N_{max} , multiple queries are sent. For large N_{max} 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-886, 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, E-874, 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, x_0 , N and N_{max} are not valid. Without parameter? must be TRUE.

C-413: $X_0 \geq 1$. $N_{max} = 4096$. 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: $N_{max} = 262144$. $X_0 \geq 0$.

C-843: $X_0 \geq 1$. Check C-843 User Manual for valid N_{max} 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: $X0 \geq 1$. $N_{max} = 32,256$. If $N = -1$ all points of the last record are returned.

C-867: $X0 \geq 1$. $N_{max} = 8192$. The number of tables is 4.

C-877: $X0 \geq 1$. $N_{max} = 8192$. The number of tables is 4.

C-884: $X0 \geq 1$. $N_{max} = 8192$. The number of tables is 8.

C-891: $X0 \geq 1$. $N_{max} = 16384$. 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: $X0 \geq 1$. $N_{max} = 8192$. The number of tables is 3.

E-709: $X0 \geq 1$. $N_{max} = 4096$. 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: $X0 \geq 1$. $N_{max} = 32,256$.

E-712: $X0 \geq 1$. $N_{max} = 262,144$. 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: $X0 \geq 1$. $N_{max} = 262,144$. 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 controller User Manual for details.

E-753: $X0 \geq 1$. $N_{max} = 65,536$. 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: $X0 \geq 1$. $N_{max} = 1,048,576$. 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: $X0 \geq 1$. $N_{max} = 4,096$.

E-761: $X0 \geq 0$. 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: $X0 \geq 1$. $N_{max} = 1024$. Two data recorder tables with 1024 points per table are provided.

E-871, E-873, E-874: $X0 \geq 1$. $N_{max} = 1024$. The number of tables is 2.

C-887, F-206: $X0 \geq 1$. See controller User Manual for N_{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.

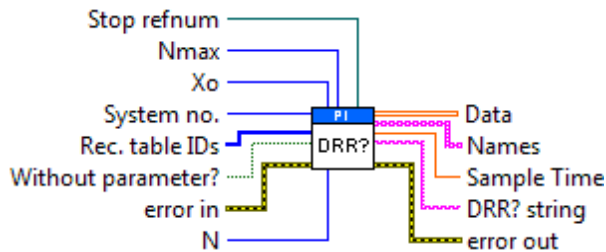
C-887, M-8X0: For GCS syntax version = 1.0 (Check with CSV?.vi. If CSV?.vi is not supported, syntax version is GCS 1.0), Rec. table IDs, xo, N and Nmax are not valid and Without parameter? must be TRUE.

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.



3.13.19. 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-886, 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, E-874, F-206, M-8X0, Mercury_GCS (but must be present for C-885 also)

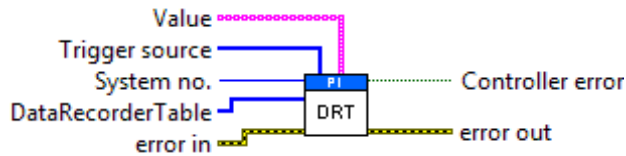
C-866: See C-866_GCS_Commands_SM150E.pdf for available trigger sources and values. DataRecorderTable = 0 (the specified trigger source is set for all data recorder tables).

C-413, C-843, C-867, C-877, C-884, C-891, E-518, E-709, E-712, E-754, E-861, E-871, E-873, E-874, 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).



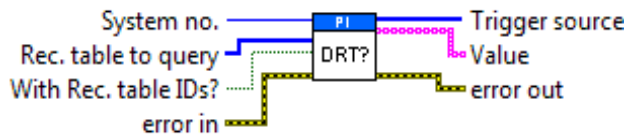
3.13.20. DRT?.vi

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-886, 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, E-874, 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).

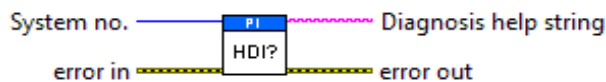


3.13.21. HDI?.vi

Returns help on diagnosis information, received with DIA?.

Valid for C-867, C-886, C-887, C-891, E-727, E-754, E-870

C-867, C-887, C-891: Check HLP? answer to find out if HDI? is supported.

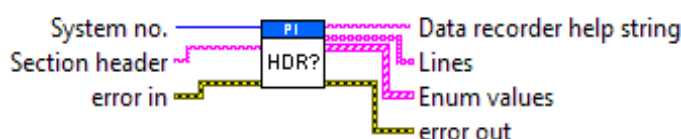


3.13.22. HDR?.vi

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 corresponding section content.

Valid for C-413, C-866, C-867, C-877, C-884, C-886, 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, E-874, 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).



3.13.23. IMP.vi

Performs a single impulse-move (two equal moves in opposite directions in quick succession) from the current position with specified impulse size (amplitude) and records a fixed number of actual positions at specified intervals thereafter, which can be read out with IMP?.vi (GCS 1.0) or DRR?.vi (GCS 2.0). If supported, "Delay" sets the number of servo loops between each position recording.

Valid for C-413, C-886, C-887, E-517, E-518, E-709, E-710, E-712, E-725, E-727, E-753, E-754, E-755, E-761, F-206 (but must be present for C-885 also)

E-517, E-518: Controller saves up to 8,192 position values. Typically, IMP is used in open-loop mode. For a single step-move, see STE.vi. Use DRR?.vi to read recorded values back. The number of servo cycles used for data recording depends on the setting made with RTR. Motion commands like IMP are not allowed when the controller is in OFFLINE mode or when the wave generator output is active. When a macro is running on the controller, IMP 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. Typically, IMP is used in open-loop mode. For a single step-move, see "STE.vi". Use DRR?.vi to read recorded values back. The number of servo cycles used for data recording depends on the setting made with RTR. Motion commands are not allowed when the wave generator is active or the analog input is used for target generation.

E-710: Delay = 0. Controller saves 32,256 position values. Width of impulse and sampling interval taken from "Table rate" parameter, set with "SPA_Hex.vi".
Caution: "Table rate" parameter influences Wave Generator also, not only IMP. For a single step-move, see "STE.vi". Use IMP?.vi to read position values back.

E-712: Delay = 0. Controller saves up to 262,144 position values. Typically, IMP is used in open-loop mode. For a single step-move, see "STE.vi". Use DRR?.vi to read recorded values back. The number of servo cycles used for data recording depends on the setting made with RTR. Motion commands are not allowed when a wave generator is active or the analog input is used for target generation.

E-725, E-727: Delay = 0. Controller saves up to 262,144 position values. Typically, IMP is used in open-loop mode. For a single step-move, see "STE.vi". Use DRR?.vi to read recorded values back. The number of servo cycles used for data recording depends on the setting made with RTR. Motion commands are not allowed when a wave generator is active or the analog input is used for target generation.

E-753: Controller saves up to 65,536 position values. Typically, IMP is used in open-loop mode. For a single step-move, see "STE.vi". Use DRR?.vi to read recorded values back. The number of servo cycles used for data recording depends on the setting made with RTR. Motion commands are not allowed when the wave generator is active or the analog input is used for target generation.

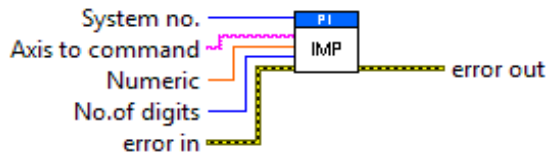
E-754: Delay = 0. Controller saves up to 1,048,576 position values. Typically, IMP is used in open-loop mode. For a single step-move, see "STE.vi". Use DRR?.vi to read recorded values back. The number of servo cycles used for data recording depends on the setting made with RTR. Motion commands are not allowed when a wave generator is active or the analog input is used for target generation.

E-755: Delay= 0. Controller saves 4,096 position values. Typically, IMP is used in open-loop mode. For a single step-move, see "STE.vi". Use DRR?.vi to read recorded values back.

E-761: Delay is impuls width in servo loops. Default value is 0 (for one servo loop). Controller saves 8,192 position values. Typically, IMP is used in open loop mode.

For a single step-move, see "STE.vi". The number of servo cycles used for data recording depends on the setting made with RTR.vi. Use DRR?.vi or IMP?.vi to read position values back.

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



3.13.24. JOG.vi

Starts motion with the given (constant) velocity values for the specified axes. The sign of the velocity values determines the direction of motion.

When motion started with JOG is executed, the target value is changed continuously according to the given velocity (can be checked with "MOV?.vi").

Motion started with JOG is executed in addition to motion started with other move commands (e.g. MOV.vi or MVR.vi).

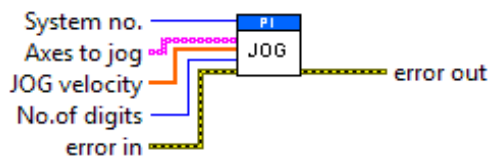
As long as motion of the axis is caused by JOG only, the axis stays on target (i.e. ONT?.vi responds with 1 since the target is continuously adapted to the actual motion).

Motion started by JOG is stopped in the following cases:

- The velocity is set to 0 with JOG.
- #24, STP or HLT is sent: these commands set the velocity for JOG to 0.
- A travel range limit is reached: the velocity for JOG remains unchanged, no error is set, the target value is set equal to the limit value.

JOG can be changed while the axis is moving.

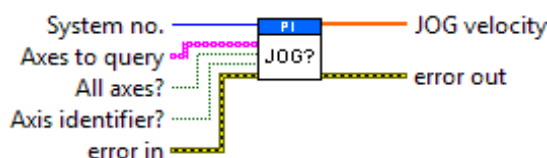
Valid for E-727, E-754



3.13.25. JOG?.vi

Get the velocity and direction for motion caused by JOG.

Valid for E-727, E-754



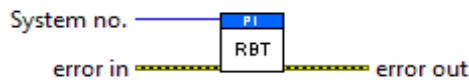
3.13.26. RBT.vi

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-886, 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-872, E-873, E-874, 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.

**3.13.27. RPA.vi**

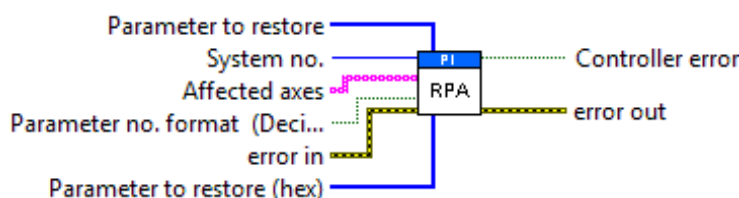
Replaces the current values of the given parameters to restore for "Affected axes" in the controller RAM with the values from non-volatile memory, waits 5000 ms and queries ERR?. For axis-related parameters, "Affected axes" is the axis name, for piezo- or sensor-related parameters, the channel number, 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, E-874, 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 the E7XX_GCS_DLL 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, E-874, 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.

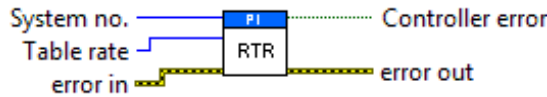
**3.13.28. RTR.vi**

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, E-874, 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).

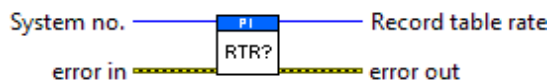


3.13.29. RTR?.vi

Returns the current table rate.

Valid for C-413, C-843, C-867, C-877, C-884, C-886, 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, E-874, 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).



3.13.30. 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-872, E-873, E-874, 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-872, E-873, E-874, 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")

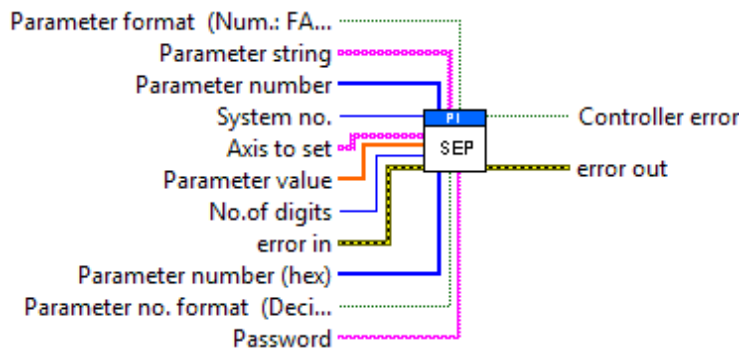
E-710: Parameter no. format is TRUE (hex.). Command is available in command level 1 only (see "CCL.vi", "CCL?.vi"). Command writes parameters to Eprom and RAM

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.



3.13.31. 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-872, E-873, E-874, 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).

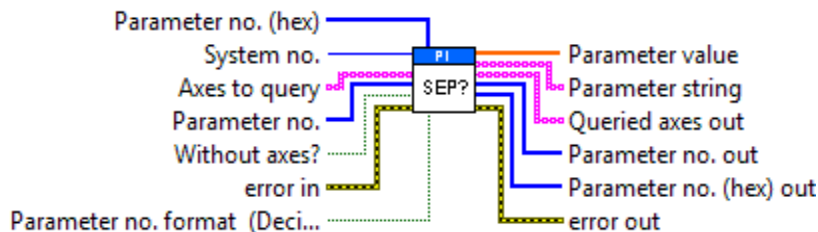
C-867, C-877, C-891, E-709, E-871, E-872, E-873, E-874, Mercury_GCS: Only one parameter value for only one axis per command allowed. Use "Without axes?" = TRUE for a query of all parameters.

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.



3.13.32. SSN?.vi

Returns controller serial number.

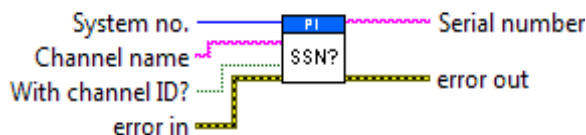
Valid for C-702, C-848, C-866, C-880, C-886, C-887, E-517, E-518, E-709, E-727, E-754, E-755, E-761, E-816, F-206, M-8X0(but must be present for C-885 also)

E-816: With channel ID = TRUE. This command cannot be issued to a slave.

All other systems: With channel ID = FALSE

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

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



3.13.33. 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-886, C-887, F-206, M-8X0 (but must be present in Special command.llb for all other systems, too)

C702: If All axes = TRUE, then Axis identifier can be FALSE

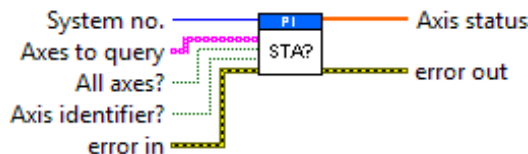
C-848: If All axes = TRUE, then Axis identifier can be FALSE

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

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

C-886: Command is equivalent to #4. For details see User Manual of the controller.

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.



3.13.34. 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-886, 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, E-874, 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 servo cycles used for data recording depends on the setting made with RTR. Motion commands are not allowed when the wave generator is active or the analog input is used for target generation.

E-710: Controller saves 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 servo cycles used for data recording depends on the setting made with RTR. Motion commands are not allowed when a wave generator is active or the analog input is used for target generation.

E-725, 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 servo cycles used for data recording depends on the setting made with RTR. Motion commands are not allowed when a wave generator is active or the analog input is used for target generation.

E-753: Controller saves up to 65,536 position values. Use DRR?.vi or DRR? and display data.vi to read recorded values back. The number of servo cycles used for data recording depends on the setting made with RTR. Motion commands are not allowed when the wave generator is active or the analog input is used for target generation.

E-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.

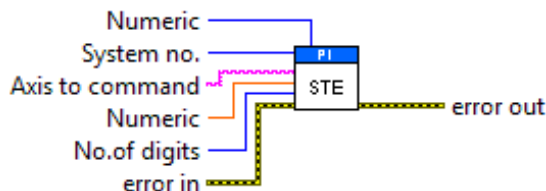
Motion commands like STE are not allowed when the joystick is active for the axis. Use DRC to define record options. Use DRR?.vi or DRR? and display data.vi to read recorded values back. You can also use MVR in combination with DRC to record values of a step motion.

E-871, E-873, E-874: 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.



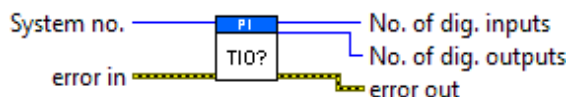
3.13.35. TIO?.vi

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-872, E-873, E-874, 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.

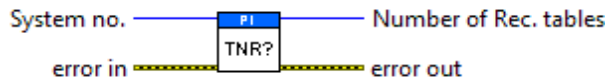


3.13.36. TNR?.vi

Returns the number of recording tables.

Valid for C-413, C-702, C-866, C-867, C-877, C-884, C-886, 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, E-874, 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).

**3.13.37. 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-872, E-873, E-874, 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).

**3.13.38. TVI?.vi**

GCS 1.0: Get valid axis identifiers. Should be called before axes are renamed with SAI.vi.

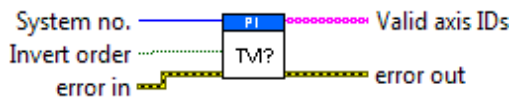
GCS 2.0: Get valid characters for axis IDs.

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-872, E-873, E-874, 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-872, E-873, E-874, Hydra, Pollux, Mercury_GCS: Invert order must be FALSE. Returns valid characters for axis IDs.



3.13.39. 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-886, 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-872, E-873, E-874, 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-886: See the user manual of the controller for valid passwords and affected settings.

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, E-872: 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 controller 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.

E-874: 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

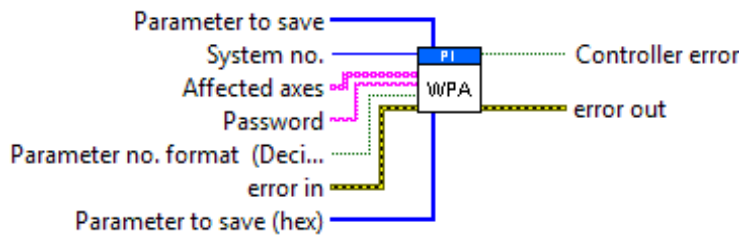
Parameters can be changed in volatile memory with SPA.

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.



3.14. Support.IIb

3.14.1. 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

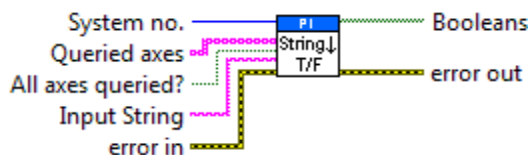


3.14.2. Assign booleans from string to axes.vi

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.

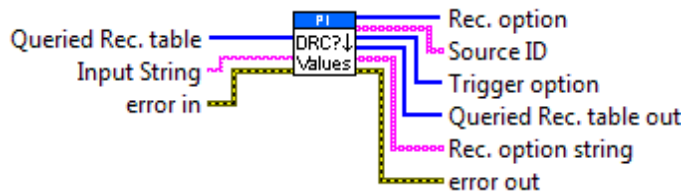


3.14.3. Assign DRC values.vi

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.

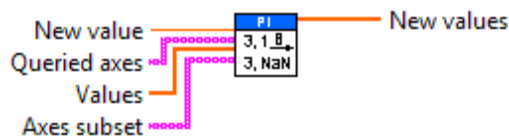
Valid for C-413, C-702, C-843, C-866, C-867, C-877, C-884, C-886, C-887, C-891, E-517, E-518, E-709, E-710, E-712, E-725, E-727, E-753, E-754, E-755, E-861, E-871, E-872, E-873, E-874, F-206, M-8X0, Mercury_GCS(but must be present for C-885 also)



3.14.4. Assign NaN for chosen axes.vi

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

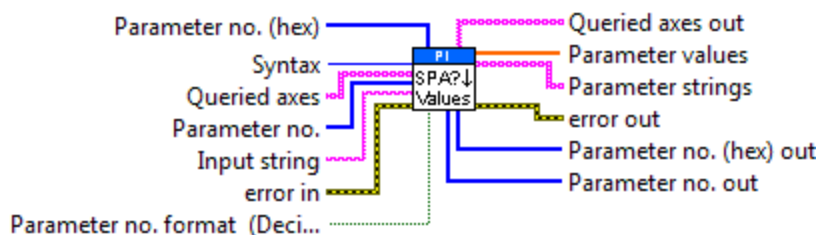
Valid for Analog systems, C-413, C-702, C-843, C-843.PM, C-848, C-865, C-866, C-867, C-877, C-880, C-884, C-886, C-887, C-891, E-517, E-518, E-709, E-712, E-725, E-727, E-753, E-754, E-755, E-861, E-871, E-873, E-874, F-206, Hydra, Pollux, M-8X0, Mercury, Mercury_GCS (but must be present for C-885 also). To support analog interfacing, VI must be present for E-816 also.



3.14.5. 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".

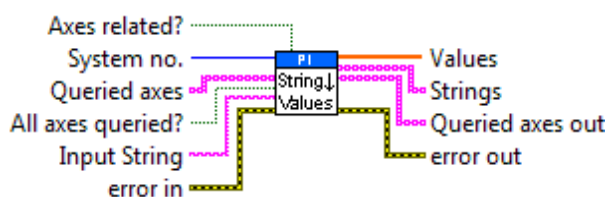
Valid for C-413, C-702, C-843, C-843.PM, C-844, C-848, C-865, C-866, C-867, C-877, C-880, C-880K005, C-884, C-886, C-887, C-891, E-516, E-517, E-518, E-709, E-710, E-712, E-725, E-727, E-753, E-754, E-755, E-761, E-816, E-861, E-870, E-872, E-871, E-873, E-874, Hydra, Pollux, Mercury, Mercury_GCS (but must be present for C-885 also)



3.14.6. 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.

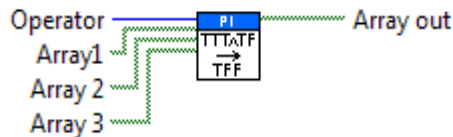
Valid for all systems.



3.14.7. 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.

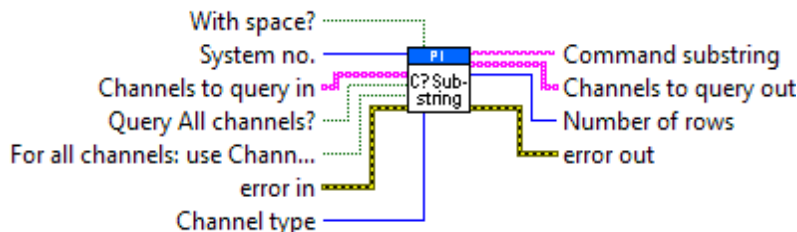
Valid for all systems



3.14.8. 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-872, E-873, E-874, F-206, Mercury_GCS, M-8X0 (but must be present for C-885 also)

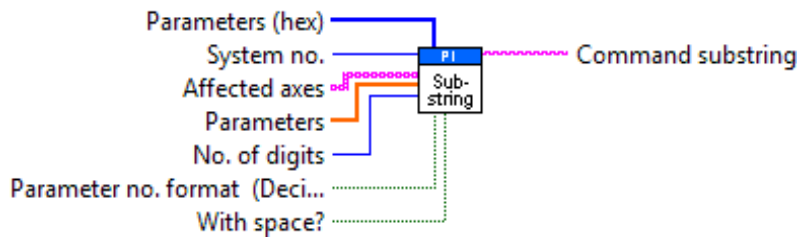


3.14.9. Build command substring.vi

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".

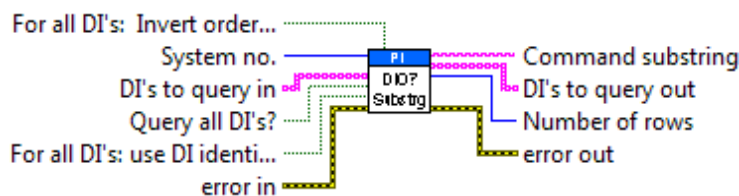
Valid for all systems



3.14.10. 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-872, E-873, E-874, F-206, Hydra, Pollux, Mercury, Mercury_GCS, M-8X0 (but must be present for C-885 also)

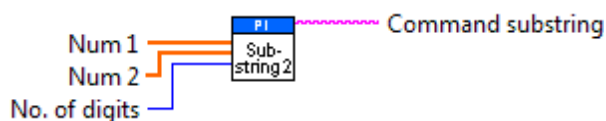


3.14.11. Build num command substring.vi

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.



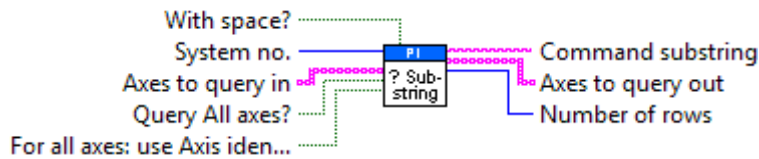
3.14.12. Build query command substring.vi

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.c. for systems which accept commands like POS? without axis IDs). If "With space?" is true or system supports GCS 2.0, a space character is added between the axes identifiers.

Example: If axes A;B;C;D are connected to the system to command, Axes to query in is A;B;D, Query all axes? is TRUE and Use Axis identifier is TRUE, resulting

Command substring is "ABCD", Number of rows is 4 and Axes to query out is A;B;C;D.

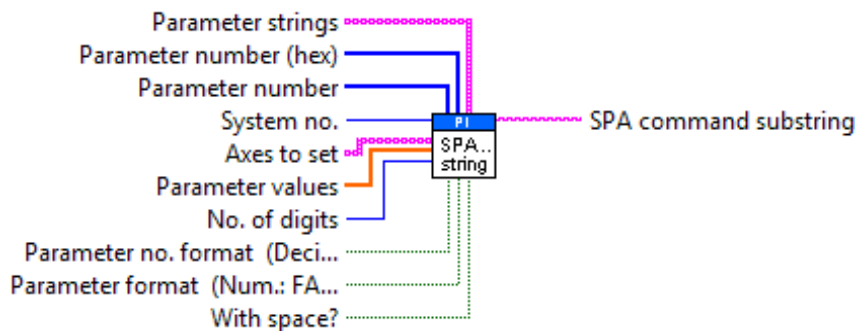
Valid for all systems



3.14.13. Build SPA command substring.vi

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".

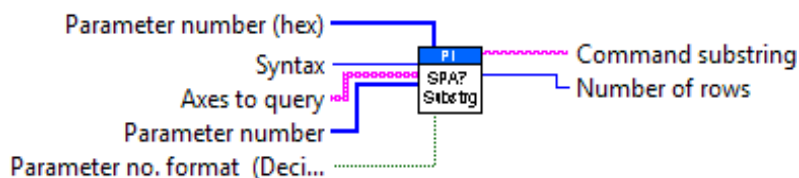
Valid for C-413, C-702, C-843, C-843.PM, C-844, C-848, C-865, C-866, C-867, C-877, C-880, C-880K005, C-884, C-886, C-887, C-891, E-516, E-517, E-518, E-709, E-712, E-725, E-727, E-753, E-754, E-816, E-861, E-870, E-871, E-872, E-873, E-874, F-206, Hydra, Pollux, M-8X0, Mercury, Mercury_GCS (but must be present for C-885, E-710, E-755 and E-761 also)



3.14.14. 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".

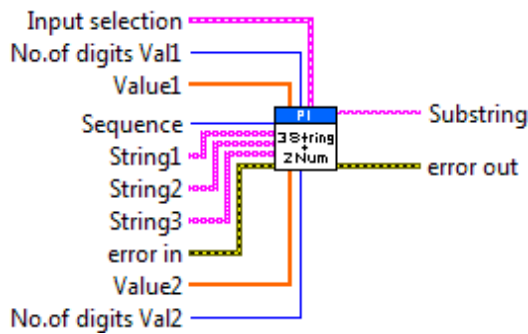
Valid for C-413, C-702, C-843, C-843.PM, C-844, C-848, C-865, C-866, C-867, C-877, C-880, C-880K005, C-884, C-886, C-887, C-891, E-516, E-517, E-518, E-709, E-710, E-712, E-725, E-727, E-753, E-754, E-755, E-761, E-816, E-861, E-870, E-871, E-872, E-873, E-874, Hydra, Pollux, Mercury, Mercury_GCS (but must be present for C-885 also)



3.14.15. 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

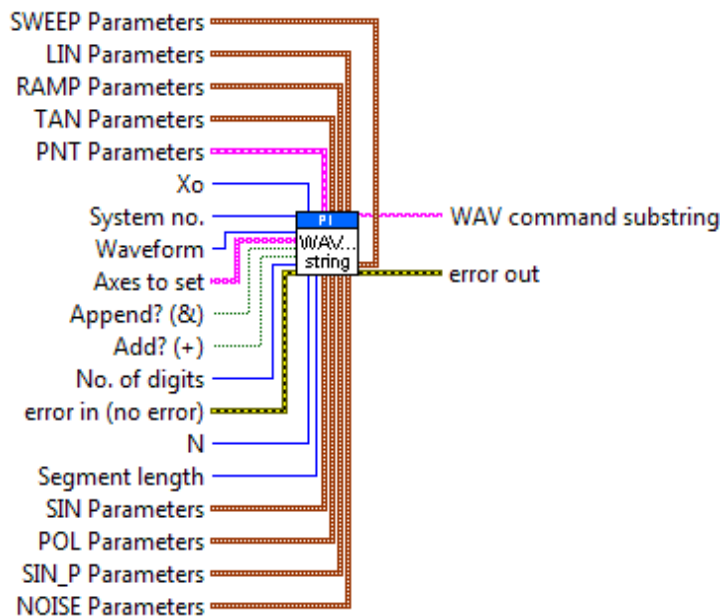
Valid for all systems



3.14.16. Build WAV command substring.vi

This vi builds a command substring for the WAV 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 "WAV.vi".

Valid for C-413, C-886, 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-761 (but must be present for C-885 also)



3.14.17. 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

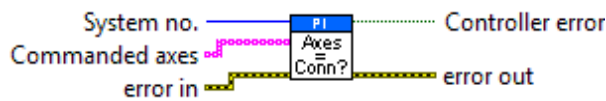


3.14.18. Commanded axes connected?.vi

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.

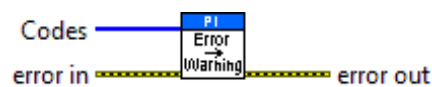
Valid for all systems.



3.14.19. 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".

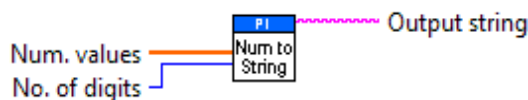
Valid for all systems.



3.14.20. 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



3.14.21. Convert num value to syntax selection.vi

This VI converts a numerical value to the corresponding GCS syntax version.

Valid for: all systems.



3.14.22. Count occurrences in string.vi

This vi counts, how often an expression occurs in a string.

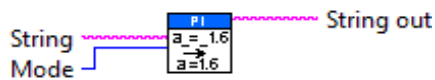
Valid for all systems



3.14.23. Cut out additional spaces.vi

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

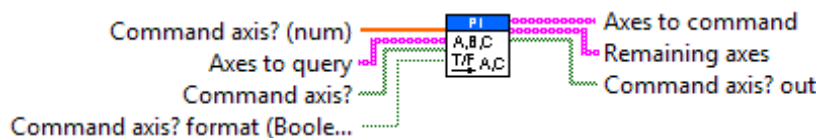
Valid for all systems



3.14.24. 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.

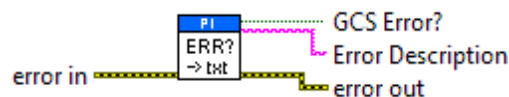
Valid for all systems



3.14.25. 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



3.14.26. General wait for movement to stop.vi

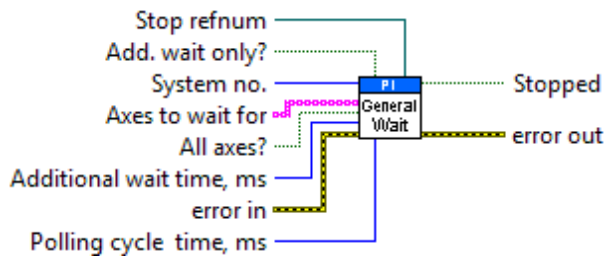
This VI waits for the specified axes to stop. An additional wait time can be specified. The wait method depends on the system to command. "XXX_Configuration_Setup.vi" (with XXX being the product name of your system) must be run before running this vi. If Add. wait only? is TRUE, VI waits the given Additional 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.

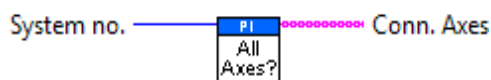
C-887, F-206, M-8X0: For GCS syntax version = GCS 1.0 (Check with CSV?.vi. If CSV?.vi is not supported, syntax version is GCS 1.0), VI will not wait for INI procedure to complete.



3.14.27. 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.

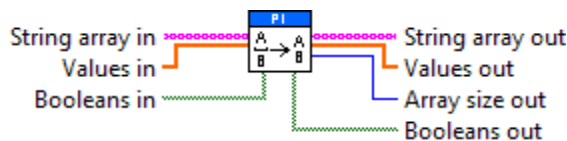
Valid for all systems.



3.14.28. Get arrays without blanks.vi

Returns the string array and related values and boolean arrays without white space string fields.

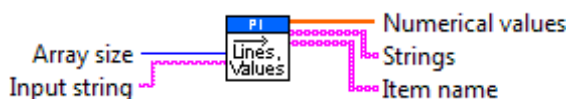
Valid for all systems



3.14.29. 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".

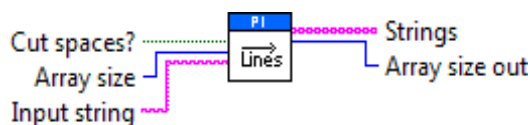
Valid for all systems



3.14.30. 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".

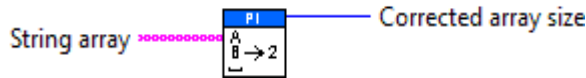
Valid for all systems.



3.14.31. Get string array size without blanks.vi

This VI returns the size of a string array without counting white space strings.

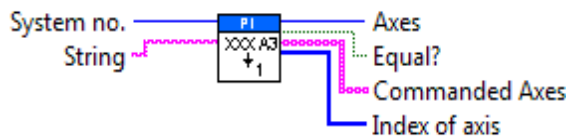
Valid for all systems



3.14.32. 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.

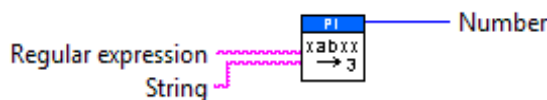
Valid for all systems



3.14.33. How often does string contain regular expression?.vi

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

Valid for all systems



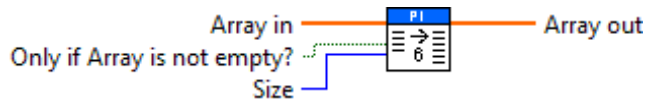
3.14.34. Increase array size.vi

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



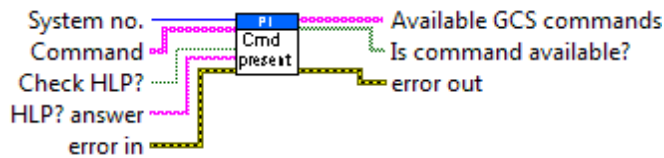
3.14.35. Is command present in HLP answer?.vi

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".

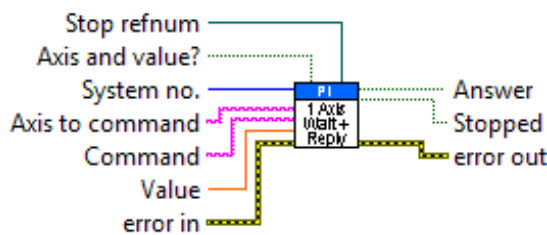
Valid for all systems



3.14.36. Longlasting one-axis command.vi

This VI sends a command (like REF, MNL or MPL), polls with #7 for controller-ready signal and returns original (boolean) command response.

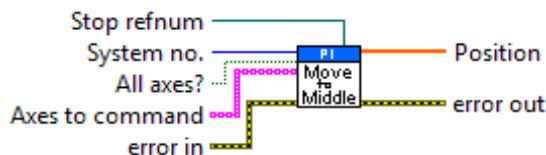
Valid for C-413, C-702, C-843, C-843.PM, C-844, C-848, C-865, C-866, C-877, C-880, C-880K005, C-884, C-887, E-710, E-761, E-871, E-873, E-874, F-206, Hydra, Pollux, Mercury (but must be present for C-885, E-709, E-712, E-725, E-727, E-753 and E-754 also)



3.14.37. Move axes to their middle position.vi

Moves axes to the middle position of their travel range, waits until motion has finished and queries position. VI also stops if Stop refnum or Local stop is TRUE. When using as a sub-VI, use Stop refnum to stop VI from caller.

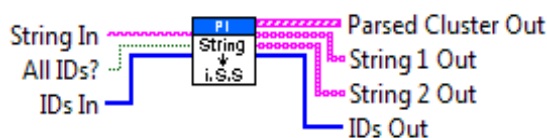
Valid for C-413, C-867, C-877, C-884, C-885, C-891, E-727, E-871, E-873, E-874, Mercury_GCS



3.14.38. Parse_FRH_Type_String.vi

Returns ID, String 1 and String 2 for String In. String In must be in FRH? reply format (ID1 = String1 TAB String2 Space LF ID2 = String 3 TAB String 4 Space LF end of help LF). If "All IDs?" is FALSE, parsed strings are returned for IDs In only.

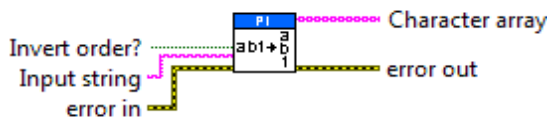
Valid for C-887, E-712, E-727



3.14.39. Return single characters from string.vi

Get single characters from input string.

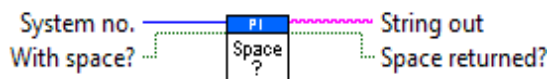
Valid for all systems



3.14.40. 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.

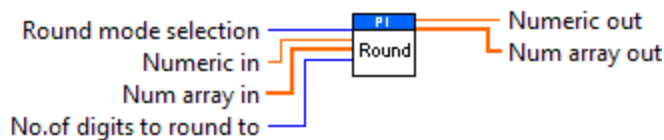
Valid for all systems.



3.14.41. Round with options.vi

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

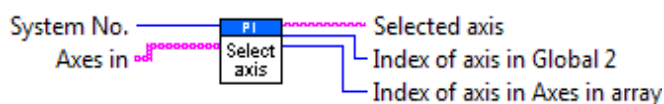
Valid for all systems



3.14.42. 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.

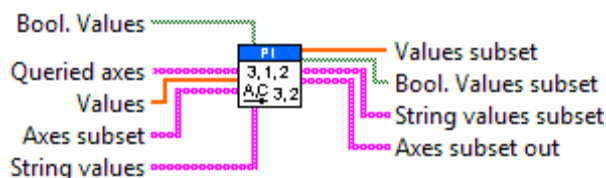
Valid for all systems



3.14.43. Select values for chosen axes.vi

This VI returns only values for the given axes subset.

Valid for all systems.



3.14.44. 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



3.14.45. 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



3.14.46. 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.

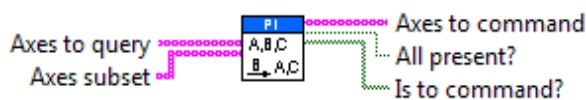
Valid for: all systems



3.14.47. 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".

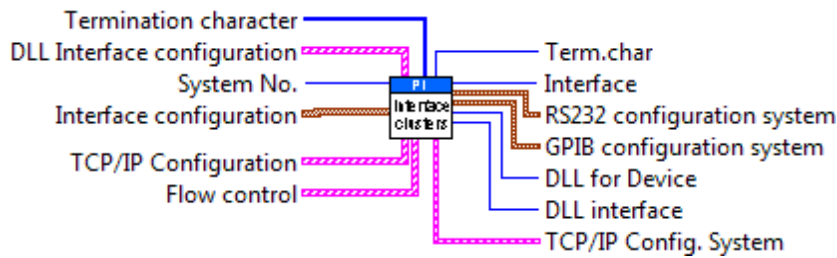
Valid for all systems



3.14.48. Unbundle/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".

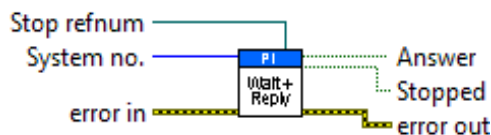
Valid for all systems.



3.14.49. Wait for answer of longlasting command.vi

This VI waits for the answer of commands like REF, MPL, MNL or scanning routines using #7 polling and stops if answer has come, Refnum sStop refnum or Local stop is TRUE, or if a communications error occurred. Sub-VI for Long-lasting, one-axis commands and controller-algorithm commands. Requires "#7.vi" to be present. When using as a sub-VI, use Stop refnum to stop VI from caller.

Valid for C-413, C-702, C-843, C-843.PM, C-844, C-848, C-865, C-866, C-877, C-880, C-880K005, C-884, C-887, C-891, E-710, E-761, E-871, E-873, E-874, Hydra, Pollux, Mercury (but must be present for C-885, E-516, E-709, E-712, E-725, E-727, E-753, E-754, F-206 and M-8X0, too)



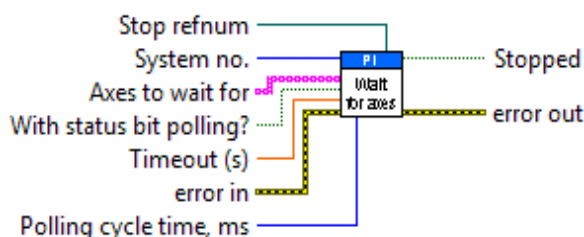
3.14.50. Wait for axes to stop.vi

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-872, E-873, E-874, Hydra, Pollux, Mercury, Mercury_GCS (but must be present in Support.llb for all other systems also)

C-880: With status bit polling? = TRUE

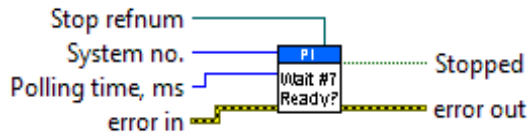
All other systems: With status bit polling? = FALSE



3.14.51. 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 occurred. 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-886, 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-872, E-873, E-874, F-206, Hydra, Pollux, Mercury, Mercury_GCS, M-8X0 (but must be present for E-816 also)



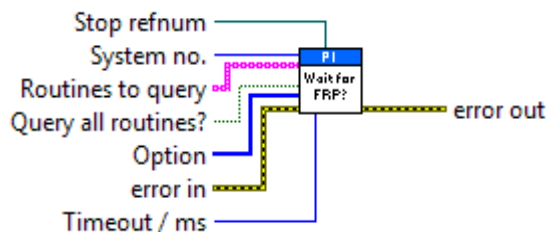
3.14.52. Wait for FRP?.vi

This vi queries FRP? until the desired option or the timeout has been reached.

If the Stop button is pressed, #24 will be sent to the controller and all motion including fast routine scan procedures will be stopped.

Valid for C-887, E-712, E-727

C-887, E-712, E-727: Check HLP? answer to find out if FRP? is supported.



3.15. WaveGenerator.IIb

3.15.1. #9.vi

Polls to determine whether a wave generator is running for any of the connected axes by sending the single ASCII character 9. Connected axes are read from Global2.vi and displayed on the front panel for assignment.

Valid for C-413, C-843, C-843.PM, C-886, 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-761 (but must be present for C-885 also)

C-843: #9 polls the state of the User Profile mode.

C-887: Check HLP? answer to find out if #9 is supported.



3.15.2. DDL.vi

Loads the specified values to the specified DDL Table. Please refer to the GCS DLL Manual for units and restrictions.

Valid for E-710, E-712, E-725, E-727, E-753, E-754

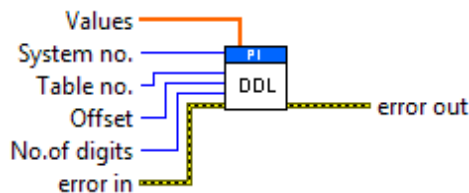
E-710: Table number can be 1 to 8. First stored value in table has index 1, index 0 must be DDL repeat number. Maximum 32768 values can be stored.

E-712: Table number can be 1 to the maximum number of axes (query with TLT? for the number of DDL tables). First stored value in table has index 1, so Offset must be greater than zero. Maximum 262,144 values can be stored. Due to the maximum numbers of parameters per command only 30 DDL table values can be set per one command call.

E-725, E-727: Table number can be 1 to the maximum number of axes (query with TLT? for the number of DDL tables).. First stored value in table has index 1, so Offset must be greater than zero. Maximum 262,144 values can be stored. Due to the maximum numbers of parameters per command only 30 DDL table values can be set per one command call.

E-753: Table number can be 1. First stored value in table has index 1, so Offset must be greater than zero. Maximum 65,536 values can be stored. Due to the maximum numbers of parameters per command only 30 DDL table values can be set per one command call.

E-754: Table number can be 1. First stored value in table has index 1, so Offset must be greater than zero. Maximum 1,046,076 values can be stored. Due to the maximum numbers of parameters per command only 30 DDL table values can be set per one command call.



3.15.3. DDL?.vi

Returns N values from specified DDL Table (in the controller memory). For large N values, communication timeout must be set long enough, otherwise a comm.error may occur.

Valid for E-710, E-712, E-725, E-727, E-753, E-754

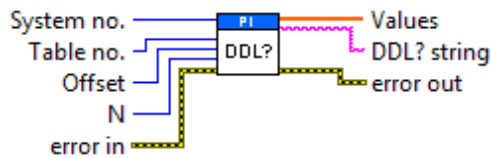
E-710: Table number can be 1 to 8. First stored value in table has index 1, index 0 is the DDL repeat number. Maximum 32768 values can be stored.

E-712: Table number can be 1 to the maximum number of axes (query with TLT? for the number of DDL tables). Offset is start point and must be greater than zero, because first stored value in table has index 1. Maximum 262,144 values can be stored.

E-725, E-727: Table number can be 1 to the maximum number of axes (query with TLT? for the number of DDL tables).. Offset is start point and must be greater than zero, because first stored value in table has index 1. Maximum 262,144 values can be stored.

E-753: Table number can be 1. Offset is start point and must be greater than zero, because first stored value in table has index 1. Maximum 65,536 values can be stored.

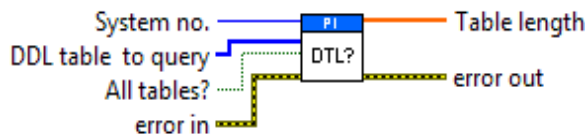
E-754: Table number can be 1. Offset is start point and must be greater than zero, because first stored value in table has index 1. Maximum 1,046,076 values can be stored.



3.15.4. DTL?.vi

Returns DDL table length.

Valid for E-712, E-725, E-727, E-753, E-754



3.15.5. GWD?.vi

Returns N waveform sequence points. Nmax is the maximum number of points that can be read at once. If $N > N_{max}$, then the VI will query GWD? more than once to read all N values.

Valid for C-413, C-886, 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-761 (but must be present for C-885 also)

C-413: Query axis is identical with wave table number, which can be 1 to 8. Nmax must be less than or equal to the length of the defined wave. Xo must be greater than zero.

C-886: Query axis is identical with wave table number. Nmax must be less than or equal to the length of the defined wave. Xo must be greater than zero.

C-887: Query axis is identical with wave table number, which can be 1 to 100. Nmax must be less than or equal to the length of the defined wave. Xo must be greater than zero. Check HLP? answer to find out if GWD? is supported.

C-891: Query axis is identical with wave table number, which can be 1 to 40. Nmax must be less than or equal to the length of the defined wave. Xo must be greater than zero. The GWD? command is present for compatibility reasons only. The response to GWD? only contains the header, but no waveform values.

E-516: $N_{max} = 5$, N can be 8192 maximum

E-517, E-518: Query axis is identical with wave table number, which can be 1 to 3. Nmax must be less than or equal to the length of the defined wave. Xo must be greater than zero. The response to GWD? does not contain any offset to the wave generator output set with WOS.

E-709: Query axis is identical with wave table number, which can be 1 to 4. Nmax must be less than or equal to the length of the defined wave. The recommended Nmax value is 1000. For larger values it might be necessary to increase the timeout value in E709_Configuration_Setup.vi. Xo must be greater than zero.

E-710: Query axis is identical with wave table number. Nmax must be less than the length of the defined wave.

E-712: Query axis is identical with wave table number, which can be 1 to 90. Nmax must be less than or equal to the length of the defined wave. Xo must be greater than zero.

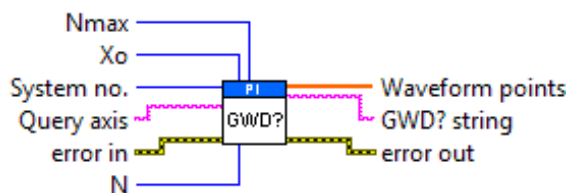
E-725: Query axis is identical with wave table number, which can be 1 to 30. Nmax must be less than or equal to the length of the defined wave. Xo must be greater than zero.

E-727: Query axis is identical with wave table number, which can be 1 to 40. Nmax must be less than or equal to the length of the defined wave. Xo must be greater than zero.

E-753: Query axis is identical with wave table number, which can be 1 to 10. Nmax must be less than or equal to the length of the defined wave. Xo must be greater than zero.

E-754: Query axis is identical with wave table number, which can be 1 to 10. Nmax must be less than or equal to the length of the defined wave. Xo must be greater than zero.

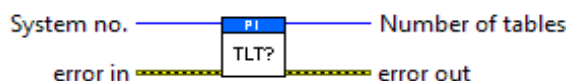
E-761: Query axis is identical with wave table number, which can be 1 to 4. Nmax must be less than the length of the defined wave. The content of a wave table is not completely erased when a new waveform is written to this table. Only the number of points given by the new waveform is written beginning with the first point in the table, but any subsequent data points will keep the old values from the former waveform. This affects only the response to the GWD? query and not the wave generator output which will only send the new waveform points.



3.15.6. TLT?.vi

Returns the number of DDL tables available in the controller.

Valid for E-710, E-712, E-725, E-727, E-753, E-754, E-761



3.15.7. TWC.vi

Clears the trigger wave settings for the waveform.

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



3.15.8. TWG?.vi

Returns the number of available wave generators.

Valid for C-413, C-891, C-886, C-887, E-517, E-518, E-709, E-710, E-712, E-725, E-727, E-753, E-754, E-761 (but must be present for C-843 and C-885 also)

C-413: Number of wave generators = 2

C-887: Number of wave generators = No. of connected axes. Check HLP? answer to find out if TWG? is supported.

C-891: Number of wave generators = 1

E-517, E-518: Number of wave generators = 3

E-709: Number of wave generators = 1

E-710: Number of wave generators = 2

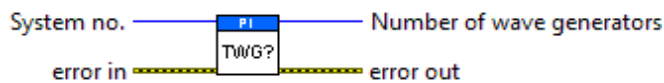
E-712: Number of wave generators = No. of connected axes

E-725, E-727: Number of wave generators = No. of connected axes

E-753: Number of wave generators = 1

E-754: Number of wave generators = 1

E-761: Number of wave generators = 4



3.15.9. TWS.vi

GCS 1.0: Sets trigger values for point(s) on the waveform and checks for error. Trigger value is bit-mapped.

GCS 2.0: Defines the trigger state of a certain trigger line for a certain waveform point to high or low.

Controller error is TRUE if selected system reports error code not equal to 0.

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

E-517, E-518: Trigger value is the trigger line to use, can be 1 to 3. Wavepoint can be 1 to 8,192. The power-on default state of all points is low. Afterwards, the signal state of the trigger output line can be switched to "low" for all points using the TWC command. It is recommended to use TWC before trigger actions are set with TWS. During the waveform output, the TWS settings will only be used if the configuration of the appropriate trigger line was set with CTO before (set Parameter value "Generator Trigger" for CTO Parameter ID "Trigger Mode"). During the waveform output, the TWS settings will only be used if the configuration of the appropriate trigger line was set with CTO before (set Parameter value "Generator Trigger" for CTO Parameter ID "Trigger Mode"). If you start the wave generator with the WGO start options given by bit 3, 4 or 5, the corresponding pulses are output in addition to the output pulses defined with TWS. The number of wave generator output cycles during which trigger pulses are to be output can be set using the Number Of Trigger Cycles parameter, ID 0x18000100. The width of a trigger pulse is 30 µs by default. You can change the default pulse width using the Pulse Width parameter, ID 0x0E000900. Possible values are in the range of 10 to 150 µs.

E-709: Trigger value is the trigger line to use, can be 1 or 2. Line 2 is identical with line 1 here as there is only one trigger table for all wave tables. Wavepoint can be 1 to 16,332. Switch can be 0 (low) or 1 (high). The power-on default state of all points is low. Afterwards, the signal state of the trigger output line can be switched to "low" for all points using the TWC command. It is recommended to use TWC before trigger actions are set with TWS. During the waveform output, the TWS settings will only be used if the configuration of the appropriate trigger line was set with CTO before (set Parameter value "Generator Trigger" for CTO Parameter ID "Trigger Mode"). Only one trigger line per command allowed.

E-710: Wavepoint can be 1-16000. Switch is not valid. Trigger value is bitmapped:

- Bit 0: trigger line 1: 0 not active, 1 active
- Bit 1: trigger line 2: 0 not active, 1 active
- Bit 2: trigger line 3: 0 not active, 1 active
- Bit 3: trigger line 4: 0 not active, 1 active
- Bit 8: If = 0, then the Trigger values apply to corresponding Wavepoints only.

If = 1, then the Trigger value applies to all points between the last point set by this command and the corresponding Wavepoint point.

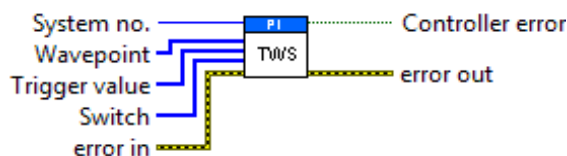
E-712: Trigger value is the trigger line to use, can be 1 to 7. Wavepoint can be 1 to 262,144. Switch can be 0 (low) or 1 (high). The power-on default state of all points is low. Afterwards, the signal state of the trigger output line can be switched to "low" for all points using the TWC command. It is recommended to use TWC before trigger actions are set with TWS. During the waveform output, the TWS settings will only be used if the configuration of the appropriate trigger line was set with CTO before (set Parameter value "Generator Trigger" for CTO Parameter ID "Trigger Mode"). Due to the maximum number of parameters per command 10 trigger lines can be set per command call.

E-725: Trigger value is the trigger line to use, can be 1 to 7. Wavepoint can be 1 to 262,144. Switch can be 0 (low) or 1 (high). The power-on default state of all points is low. Afterwards, the signal state of the trigger output line can be switched to "low" for all points using the TWC command. It is recommended to use TWC before trigger actions are set with TWS. During the waveform output, the TWS settings will only be used if the configuration of the appropriate trigger line was set with CTO before (set Parameter value "Generator Trigger" for CTO Parameter ID "Trigger Mode").

E-727: Trigger value is the trigger line to use, can be 1 to 3. Wavepoint can be 1 to 262,144. Switch can be 0 (low) or 1 (high). The power-on default state of all points is low. Afterwards, the signal state of the trigger output line can be switched to "low" for all points using the TWC command. It is recommended to use TWC before trigger actions are set with TWS.

E-753: Trigger value is the trigger line to use, can be 1. Wavepoint can be 1 to 65,536. Switch can be 0 (low) or 1 (high). During the waveform output, the TWS settings will only be used if the configuration of the appropriate trigger line was set with CTO before (set Parameter value "Generator Trigger" for CTO Parameter ID "Trigger Mode").

E-754: Trigger value is the trigger line to use, can be 1. Wavepoint can be 1 to 1,048,576. Switch can be 0 (low) or 1 (high). During the waveform output, the TWS settings will only be used if the configuration of the appropriate trigger line was set with CTO before (set Parameter value "Generator Trigger" for CTO Parameter ID "Trigger Mode").



3.15.10. TWS?.vi

Returns trigger points set by TWS. 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.

Valid for E-517, E-518, E-712, E-725, E-727, E-753, E-754

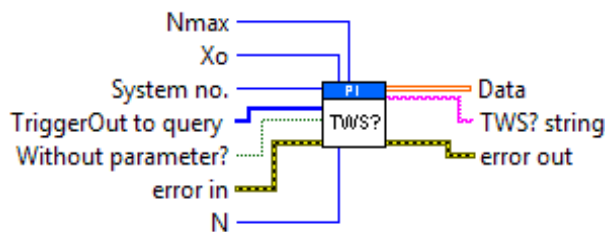
E-517, E-518: Nmax = 8192.

E-712: Nmax = 262,144.

E-725, E-727: Nmax = 262,144.

E-753: Nmax = 262,144. Check HLP? answer to find out if TWS? is supported.

E-754: Nmax = 1,048,576.



3.15.11. WAV.vi

Defines or modifies a stored waveform for one or more axes. Please refer to the GCS DLL manual or to the User manual for a description of waveform parameters, units and restrictions. No. of digits is the number of digits after the decimal point in the 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-886, 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-761(but must be present for C-885 also)

C-413: Affected axis is identical with wave table number. 8 waves can be defined in total. Use WSL.vi to connect wave table to wave generator. Waveforms are automatically stored in non-volatile memory. Note that the number of write cycles is limited. See User Manual for details.

Waveform can be: SIN_P, RAMP, LIN or PNT.

Valid for PNT: Affected axes, No. of digits, Xo, N, Append?, Segment length, Nmax for PNT, PNT Parameters (Ao to An-1). Nmax for PNT must be <=10 (4096 is maximum size for all wave tables in total but input buffer size is 128 characters).

C-886: Affected axis is identical with wave table number. 100 waves can be defined in total. Use WSL.vi to connect wave table to wave generator.

Waveform can be: SIN_P, PNT, LIN, or RAMP. Check MAN? WAV answer to find out which waveforms are supported.

Valid for PNT: Affected axes, No. of digits, Xo, N, Append?, Segment length, Nmax for PNT, PNT Parameters (Ao to An-1). Nmax for PNT must be < 20 (1 Million is maximum size for all wave tables in total but input buffer size is limited). Segment length must be zero for PNT.

C-887: Affected axis is identical with wave table number. 100 waves can be defined in total. Use WSL.vi to connect wave table to wave generator.

Waveform can be: SIN_P, PNT, LIN, or RAMP. Check MAN? WAV answer to find out which waveforms are supported..

Valid for PNT: Affected axes, No. of digits, Xo, N, Append?, Segment length, Nmax for PNT, PNT Parameters (Ao to An-1). Nmax for PNT must be < 20 (4 Million is maximum size for all wave tables in total but input buffer size is limited). Segment length must be zero for PNT. Check HLP? answer to find out if WAVis supported.

C-891: Affected axis is identical with wave table number. 40 waves can be defined in total. Use WSL.vi to connect wave table to wave generator. Waveform can be: SIN_P, RAMP, or LIN.

E-516: Waveform can be: SIN, TAN, POL or PNT.

The following parameters are equivalent (see GCS DLL manual):

Affected axes = szAxes, Xo = nStart, N = nLength, Add? = Add,

PNT Parameters = pPoints (PNT parameters is an array consisting of single data points which will be sent in groups of Nmax for PNT data points.)

Valid for PNT: Affected axes, No. of digits, Xo, N, Add?, Nmax for PNT, PNT Parameters (Ao to An-1)

E-517, E-518: Affected axis is identical with wave table number, which can be 1 to 3. One wave table number per command call allowed only.

Waveform can be: SIN_P, RAMP or LIN. For SIN, TAN, POL and PNT Segment length must be zero.

Valid for PNT: Affected axes, No. of digits, Xo, N, Add?, Nmax for PNT, PNT Parameters (Ao to An-1) Nmax for PNT must be <= 20 (8,192 is maximum size for all wave tables in total but input buffer size is 256 characters).

E-709: Affected axis is identical with wave table number. 4 waves can be defined in total. Use WSL.vi to connect wave table to wave generator. Waveforms are automatically stored in non-volatile memory. Note that the number of write cycles is limited. See User Manual for details.

Waveform can be: SIN_P, RAMP, or LIN.

E-710: Affected axis is identical with wave table number.

Waveform can be: SIN_P, RAMP, LIN or PNT.

Valid for PNT: Affected axes, No. of digits, Xo, N, Append?, Segment length, Nmax for PNT, PNT Parameters (Ao to An-1). Nmax for PNT must be <= Wave storage max. value set with WMS.vi. Segment length must be zero for PNT.

The length of the wave to define must be less than the value defined with "WMS.vi".

E-712: Affected axis is identical with wave table number. 90 waves can be defined in total. Use WSL.vi to connect wave table to wave generator.

Waveform can be: SIN_P, RAMP, LIN, NOISE or PNT.

Valid for PNT: Affected axes, No. of digits, Xo, N, Append?, Segment length, Nmax for PNT, PNT Parameters (Ao to An-1). Nmax for PNT must be <= 20 (262,144 is maximum size for all wave tables in total but input buffer size is 256 characters). Segment length must be zero for PNT.

E-725: Affected axis is identical with wave table number. 30 waves can be defined in total. Use WSL.vi to connect wave table to wave generator.

Waveform can be: SIN_P, RAMP, LIN, NOISE or PNT.

Valid for PNT: Affected axes, No. of digits, Xo, N, Append?, Segment length, Nmax for PNT, PNT Parameters (Ao to An-1). Nmax for PNT must be <= 20

(262,144 is maximum size for all wave tables in total but input buffer size is 256 characters). Segment length must be zero for PNT.

E-727: Affected axis is identical with wave table number. 40 waves can be defined in total. Use WSL.vi to connect wave table to wave generator.

Waveform can be: SIN_P, RAMP, LIN, NOISE or PNT.

Valid for PNT: Affected axes, No. of digits, Xo, N, Append?, Segment length, Nmax for PNT, PNT Parameters (Ao to An-1). Nmax for PNT must be ≤ 20 (262,144 is maximum size for all wave tables in total but input buffer size is 256 characters). Segment length must be zero for PNT.

E-753: Affected axis is identical with wave table number. 10 waves can be defined in total. Use WSL.vi to connect wave table to wave generator.

Waveform can be: SIN_P, RAMP, LIN, NOISE or PNT.

Valid for PNT: Affected axes, No. of digits, Xo, N, Append?, Segment length, Nmax for PNT, PNT Parameters (Ao to An-1). Nmax for PNT must be ≤ 20 (65,536 is maximum size for all wave tables in total but input buffer size is 256 characters). Segment length must be zero for PNT.

E-754: Affected axis is identical with wave table number. 10 waves can be defined in total. Use WSL.vi to connect wave table to wave generator.

Waveform can be: SIN_P, PNT, RAMP, LIN, NOISE or SWEEP.

Valid for PNT: Affected axes, No. of digits, Xo, N, Append?, Segment length, Nmax for PNT, PNT Parameters (Ao to An-1). Nmax for PNT must be ≤ 20 (1,048,576 is maximum size for all wave tables in total but input buffer size is limited). Segment length must be zero for PNT.

Valid for SWEEP: Affected axes, Append?, SWEEP Parameters. See controller user manual for details.

E-761: Affected axis is identical with wave table number.

Waveform can be: SIN_P, RAMP, LIN, PNT or POL.

Valid for PNT: Affected axes, No. of digits, Xo, N, Append?, Segment length, Nmax for PNT, PNT Parameters (Ao to An-1). Nmax for PNT must be \leq Wave storage max. value set with WMS.vi, and can be 50 maximum. Segment length must be zero for PNT.

General: The following parameters are equivalent (see GCS DLL manual):

Affected axes = szWaveTableIds, Xo = iOffsetOfFirstPointInWaveTable, N = iNumberOfPoints, Append? = iAddAppendWave, C = iCenterPointOfWave, J = iNumberOfSpeedUpDownPointsInWave, A = dAmplitudeOfWave, O = dOffsetOfWave (is only valid if a wave segment is being concatenated to an existing wave, i.e. Append? = TRUE), Segment length = iSegmentLength, PNT Parameters = pdWavePoints (PNT parameters is an array consisting of single data points which will be sent in groups of Nmax for PNT data points.)

Valid for SIN_P: Affected axes, No. of digits, Xo, N, Append?, Segment length, SIN_P Parameters (C, A and O)

Valid for SIN: Affected axes, No. of digits, Xo, N, Add?, SIN Parameters (A, N, Xo, Phi and B)

Valid for RAMP: Affected axes, No. of digits, Xo, N, Append?, Segment length, RAMP Parameters (C, J, A and O)

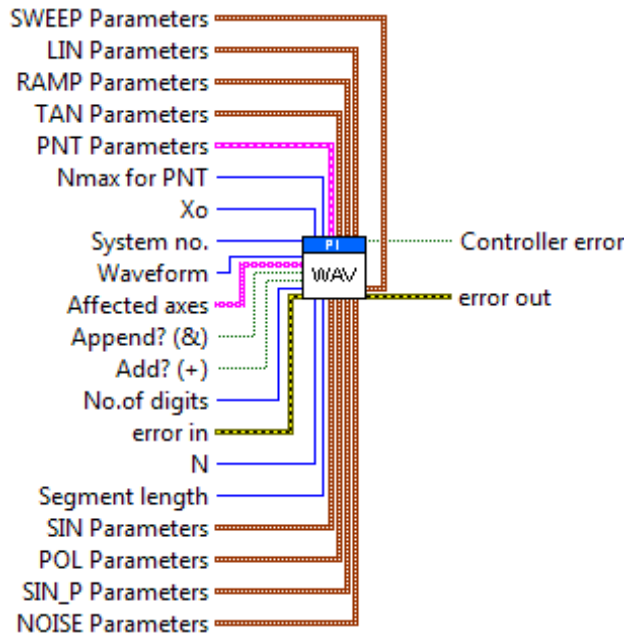
Valid for LIN: Affected axes, No. of digits, Xo, N, Append?, Segment length, LIN Parameters (J, A and O)

Valid for NOISE: Affected axes, No. of digits, Append?, Segment length, NOISE Parameters (A and O)

Valid for TAN: Affected axes, No. of digits, Xo, N, Add?, TAN Parameters (A, N, Xo, Phi and B)

Valid for POL: Affected axes, No. of digits, Xo, N, Add?, POL Parameters (Xo, Ao to A5)

Valid for PNT: see above.



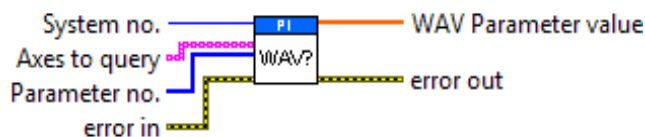
3.15.12. WAV?.vi

Returns waveform parameter values for queried axes and parameter numbers.

Valid for C-413, C-886, C-887, C-891, E-517, E-518, E-709, E-710, E-712, E-725, E-727, E-753, E-754, E-761 (but must be present for C-885 also)

All systems: Affected axis is identical with wave table number. The following parameter number is valid:

- 1: Number of waveform points for currently defined wave



3.15.13. WCL.vi

Clears waveform associated with specified axis.

Valid for C-413, C-886, C-887, E-517, E-518, E-709, E-710, E-712, E-725, E-727, E-753, E-754, E-761 (but must be present for C-885 also)

C-413: WCL axes is identical with wave table number which can be 1 to 8. All axes? must be FALSE.

C-886: WCL axes is identical with wave table number which can be 1 to 100. All axes? must be FALSE.

C-887: WCL axes is identical with wave table number which can be 1 to 100. All axes? must be FALSE. Check HLP? answer to find out if WCL is supported.

E-517, E-518: WCL axes is identical with wave table number which can be 1 to 3. All axes? must be FALSE.

E-709: WCL axes is identical with wave table number which can be 1 to 4. All axes? must be FALSE.

E-710: Does also clear DDL table. All axes must be FALSE, only one axis per command allowed.

E-712: WCL axes is identical with wave table number which can be 1 to 90. All axes? must be FALSE.

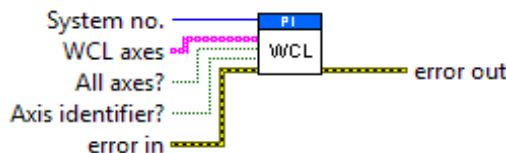
E-725: WCL axes is identical with wave table number which can be 1 to 30. All axes? must be FALSE.

E-727: WCL axes is identical with wave table number which can be 1 to 40. All axes? must be FALSE.

E-753: WCL axes is identical with wave table number which can be 1 to 10. All axes? must be FALSE.

E-754: WCL axes is identical with wave table number which can be 1 to 10. All axes? must be FALSE.

E-761: All axes must be FALSE.



3.15.14. WGC.vi

Sets number of wave generator cycles. Controller error is TRUE if selected system reports an error code which is not zero.

Valid for C-413, C-886, C-887, C-891, E-516, E-517, E-518, E-709, E-712, E-725, E-727, E-753, E-754, E-761 (but must be present for C-885 also)

C-413: Affected axes is identical with wave generator ID, which can be 1 to 2. If Cycles = 0 the wave is performed periodically until it is stopped otherwise.

C-886: Affected axes is identical with wave generator ID, which can be 1 to No. of connected axes. If Cycles = 0 the wave is performed periodically until it is stopped otherwise. Cycles set for any wave generator ID are valid for all wave generators (cannot be set differently).

C-887: Affected axes is identical with wave generator ID, which can be 1 to No. of connected axes. If Cycles = 0 the wave is performed periodically until it is stopped otherwise. Cycles set for any wave generator ID are valid for all wave generators (cannot be set differently). Check HLP? answer to find out if WGC is supported.

C-891: Affected axes is identical with wave generator ID, which can be 1. If Cycles = 0 the wave is performed periodically until it is stopped otherwise.

E-517, E-518: Affected axes is identical with wave generator ID, which can be 1 to 3. If Cycles = 0 the wave is performed periodically until it is stopped otherwise.

WGC saves the new value in RAM only. The WPA command saves the currently valid value to non-volatile memory, where it becomes the power-on default.

Settings not saved with WPA will be lost on power down or reboot.

E-709: Affected axes is identical with wave generator ID, which can be 1. If Cycles = 0 the wave is performed periodically until it is stopped otherwise.

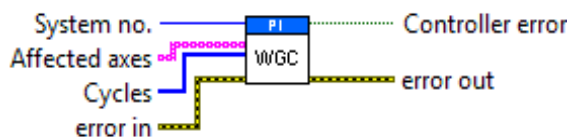
E-712: Affected axes is identical with wave generator ID, which can be 1 to No. of connected axes. If Cycles = 0 the wave is performed periodically until it is stopped otherwise. Cycles set for any wave generator ID are valid for all wave generators (cannot be set differently).

E-725, E-727: Affected axes is identical with wave generator ID, which can be 1 to No. of connected axes. If Cycles = 0 the wave is performed periodically until it is stopped otherwise. Cycles set for any wave generator ID are valid for all wave generators (cannot be set differently).

E-753: Affected axes is identical with wave generator ID, which can be 1. If Cycles = 0 the wave is performed periodically until it is stopped otherwise.

E-754: Affected axes is identical with wave generator ID, which can be 1. If Cycles = 0 the wave is performed periodically until it is stopped otherwise.

E-761: Affected axes is identical with wave table number, which can be 1 to 4. If Cycles = 0 the wave is performed periodically until it is stopped otherwise. E-761: WGC saves the new value in RAM only. The WPA command saves the currently valid value to non-volatile memory, where it becomes the power- on default. Settings not saved with WPA will be lost when the PC is powered off or the E-761 is rebooted.



3.15.15. WGC?.vi

Returns number of wave generator cycles set for specified axes.

Valid for C-413, C-886, C-887, C-891, E-516, E-517, E-518, E-709, E-712, E-725, E-727, E-753, E-754, E-761 (but must be present for C-885 also)

C-413: Axes to query is identical with wave generator ID, which can be 1 to 2. All axes? = must be FALSE.

C-886: Axes to query is identical with wave generator ID, which can be 1 to No. of connected axes. All axes? = must be FALSE.

C-887: Axes to query is identical with wave generator ID, which can be 1 to No. of connected axes. All axes? = must be FALSE. Check HLP? answer to find out if WGC? is supported.

C-891: Axes to query is identical with wave generator ID, which can be 1. All axes? = must be FALSE.

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

E-517, E-518: Axes to query is identical with wave generator ID, which can be 1 to 3. All axes? must be FALSE.

E-709: Axes to query is identical with wave generator ID, which can be 1. All axes? must be FALSE.

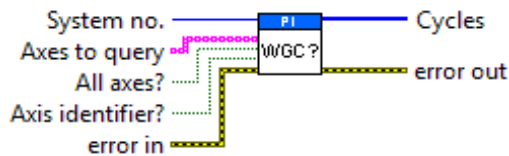
E-712: Axes to query is identical with wave generator ID, which can be 1 to No. of connected axes. All axes? must be FALSE.

E-725, E-727: Axes to query is identical with wave generator ID, which can be 1 to No. of connected axes. All axes? must be FALSE.

E-753: Axes to query is identical with wave generator ID, which can be 1. All axes? must be FALSE.

E-754: Axes to query is identical with wave generator ID, which can be 1. All axes? must be FALSE.

E-761: Axes to query is identical with wave table number, which can be 1 to 4. All axes must be FALSE



3.15.16. WGO.vi

Enables, disables, and sets wave generator output mode. Parameter is bit-mapped. Controller error is TRUE if selected system reports an error code which is not zero.

Valid for C-413, C-886, 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-761 (but must be present for C-885 also)

C-413: Affected axis is identical with wave generator number, which can be 1 to 2. See C-413 User Manual for a description of valid parameters. Use DRC to configure data recording. Recording ends when the record table content has reached the maximum number of points (4096 in total). Starting the wave generator is not allowed when the analog input is used for target generation.

C-886: Affected axis is identical with wave generator number, which can be 1 to No. of connected axes. See controller User Manual for a description of valid parameters. Recording ends when the record table content has reached the maximum number of points (1 Million in total). Starting the wave generator is not allowed when axes are moving.

C-887: Affected axis is identical with wave generator number, which can be 1 to No. of connected axes. See Controller User Manual for a description of valid parameters. Use DRC to configure data recording. Recording ends when the record table content has reached the maximum number of points (4 Million in total). Starting the wave generator is not allowed when axes are moving. Check HLP? answer to find out if WGO is supported.

C-891: Affected axis is identical with wave generator number, which can be 1. See Controller User Manual for a description of valid parameters. Use DRC to configure data recording. Recording ends when the record table content has reached the maximum number of points (16384 in total).

E-516: See User Manual for parameter definition.

E-517, E-518: Affected axis is identical with wave generator number, which can be 1 to 3. See controller User Manual for a description of valid parameters. Use DRC to configure data recording. Recording ends when the record table content has reached the maximum number of points (8,192 in total).

E-709: Affected axis is identical with wave generator number, which can be 1. See E-709 User Manual for a description of valid parameters. Use DRC to configure data recording. Recording ends when the record table content has reached the maximum number of points (4096 in total). Starting the wave generator is not allowed when the analog input is used for target generation.

E-710: Affected axis is identical with wave generator number. Only two wave generators can run simultaneously. See E7XX_GCS_DLL Manual for parameter definition.

E-712: Affected axis is identical with wave generator number, which can be 1 to No. of connected axes. See E-712 User Manual for a description of valid parameters. Use DRC to configure data recording. Recording ends when the record table content has reached the maximum number of points (262,144 in total). Starting the wave generator is not allowed when the analog input is used for target generation.

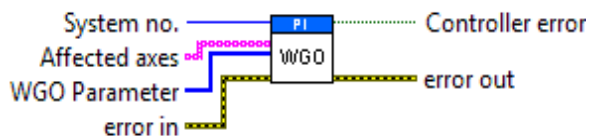
E-725: Affected axis is identical with wave generator number, which can be 1 to 3. See E-725 User Manual for a description of valid parameters. Use DRC to configure data recording. Recording ends when the record table content has reached the maximum number of points (262,144 in total). Starting the wave generator is not allowed when the analog input is used for target generation.

E-727: Affected axis is identical with wave generator number, which can be 1 to 4. See E-727 User Manual for a description of valid parameters. Use DRC to configure data recording. Recording ends when the record table content has reached the maximum number of points (262,144 in total). Starting the wave generator is not allowed when the analog input is used for target generation.

E-753: Affected axis is identical with wave generator number, which can be 1. See E-753 User Manual for a description of valid parameters. Use DRC to configure data recording. Recording ends when the record table content has reached the maximum number of points (65,536 in total). Starting the wave generator is not allowed when the analog input is used for target generation.

E-754: Affected axis is identical with wave generator number, which can be 1. See controller User Manual for a description of valid parameters. Use DRC to configure data recording. Recording ends when the record table content has reached the maximum number of points (1,048,576 in total). Starting the wave generator is not allowed when the analog input is used for target generation.

E-761: Affected axis is identical with wave generator number, which can be 1 to 4. Four wave generators can run simultaneously. See E7XX_GCS_DLL Manual for parameter definition. Each time the wave generator is started recording starts automatically as follows: recorder table 1: axis 1 actual position, recorder table 2: axis 2 actual position, recorder table 3: axis 3 actual position, recorder table 4: analog input voltage (same value as read with TAV?.vi, i.e. contains gain and offset for the analog input, see E-761 User Manual. Recording ends when the record table content has reached the maximum number of points (8192 per table).



3.15.17. WGO?.vi

Returns WGO parameter for specified axes.

Valid for C-413, C-886, 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-761 (but must be present for C-885 also)

C-413: Axis to query is identical with wave generator number, which can be 1 to 2. If All axes? = TRUE, then Axis identifier? must be FALSE.

C-886: Axis to query is identical with wave generator number, which can be 1 to No. of connected axes. If All axes? = TRUE, then Axis identifier? must be FALSE.

C-887: Axis to query is identical with wave generator number, which can be 1 to No. of connected axes. If All axes? = TRUE, then Axis identifier? must be FALSE. Check HLP? answer to find out if WGO? is supported.

C-891: Axis to query is identical with wave generator number, which can be 1. If All axes? = TRUE, then Axis identifier? must be FALSE.

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

E-517, E-518: Axis to query is identical with wave generator number, which can be 1 to 3. If All axes? = TRUE, then Axis identifier? must be FALSE.

E-709: Axis to query is identical with wave generator number, which can be 1. If All axes? = TRUE, then Axis identifier? must be FALSE.

E-710: Affected axis is identical with wave generator number. If All axes = TRUE, then Axis identifier must be FALSE

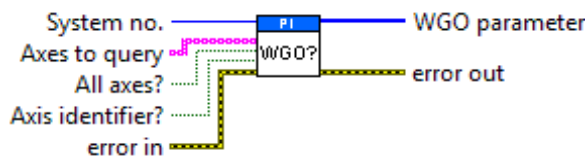
E-712: Axis to query is identical with wave generator number, which can be 1 to No. of connected axes. If All axes? = TRUE, then Axis identifier? must be FALSE.

E-725, E-727: Axis to query is identical with wave generator number, which can be 1 to No. of connected axes. If All axes? = TRUE, then Axis identifier? must be FALSE.

E-753: Axis to query is identical with wave generator number, which can be 1. If All axes? = TRUE, then Axis identifier? must be FALSE.

E-754: Axis to query is identical with wave generator number, which can be 1. If All axes? = TRUE, then Axis identifier? must be FALSE.

E-761: Affected axis is identical with wave generator number, which can be 1 to 4. All axes must be FALSE



3.15.18. WGR.vi

Starts a new recording.

Valid for: C-413, C-886, C-887, C-891, E-517, E-518, E-709, E-710, E-712, E-725, E-727, E-753, E-754, E-761 (but must be present for C-885 also)

C-413: Use DRC to configure data recording. Recording ends when the record table content has reached the maximum number of points (4096 in total).

C-886: Recording ends when the record table content has reached the maximum number of points (8192 per table).

C-887: Use DRC to configure data recording. Recording ends when the record table content has reached the maximum number of points (4 Million in total). Check HLP? answer to find out if WGR is supported.

C-891: Use DRC to configure data recording. Recording ends when the record table content has reached the maximum number of points (16384 in total).

E-517, E-518: Use DRC to configure data recording. Recording ends when the record table content has reached the maximum number of points (8,192 in total). If

more than one wave generator is running, recording starts at the waveform start point which occurs first.

E-709: Use DRC to configure data recording. Recording ends when the record table content has reached the maximum number of points (4096 in total).

E-712: Use DRC to configure data recording. Recording ends when the record table content has reached the maximum number of points (262,144 in total).

E-725, E-727: Use DRC to configure data recording. Recording ends when the record table content has reached the maximum number of points (262,144 in total).

E-753: Use DRC to configure data recording. Recording ends when the record table content has reached the maximum number of points (65,536 in total).

E-754: Use DRC to configure data recording. Recording ends when the record table content has reached the maximum number of points (1,048,576 in total).

E-761: Recording will be started for all recorder tables when the wave generator is running for an arbitrary axis. 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?.vi, i.e. contains gain and offset for the analog input, see E-761 User Manual). Recording starts always with the next start point of the waveform, i.e. there might be a short delay between sending WGR and the start of the record. If more than one wave generator is running, recording starts at the waveform start point which occurs first. Recording ends when the record table content has reached the maximum number of points (8192 per table).



3.15.19. WOS.vi

Sets Wave Generator Output Offset. The current wave generator output is created by adding the output offset value to the current wave value:

Generator Output = Output Offset + Current Wave Value.

The value is also modified by the generator when the WGO command was executed with iStartMod bit 8 set: At the end of each cycle the output offset value is equate with the current Generator Output. No. of digits is the number of digits after the decimal point in the offset value(s) that will be sent.

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

C-413: Generator ID can be 1 to 2.

E-517, E-518: Generator ID can be 1 to 3. You can set the offset also using the CFG wave type (see WAV command) or by directly changing the Wave Offset parameter, ID 0x1300010b, with SPA.

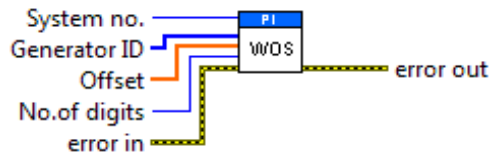
E-709: Generator ID can be 1

E-712: Generator ID can be 1 to No. of connected axes

E-725, E-727: Generator ID can be 1 to No. of connected axes

E-753: Generator ID can be 1

E-754: Generator ID can be 1



3.15.20. WOS?.vi

Returns wave generator output offset.

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

C-413: Generator ID can be 1 to 2.

E-517, E-518: Generator ID can be 1 to 3.

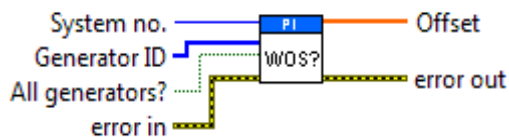
E-709: Generator ID can be 1

E-712: Generator ID can be 1 to No. of connected axes

E-725, E-727: Generator ID can be 1 to No. of connected axes

E-753: Generator ID can be 1

E-754: Generator ID can be 1



3.15.21. WSL.vi

Connects a wave table to a generator or disconnects the selected generator from any wave table.

Valid for C-413, C-886, C-887, C-891, E-709, E-517, E-518, E-712, E-725, E-727, E-753, E-754 (but must be present for C-885 also)

C-413: Generator ID can be 1 to 2, WaveTableID can be 1 to 8.

C-886: Generator ID can be 1 to No. of connected axes, WaveTableID can be 1 to 100.

C-887: Generator ID can be 1 to No. of connected axes, WaveTableID can be 1 to 100. Check HLP? answer to find out if WSL is supported.

C-891: Generator ID can be 1, WaveTableID can be 1 to .

E-709: Generator ID can be 1, WaveTableID can be 1 to 4.

E-517: Generator ID can be 1 to 3., WaveTableID can be 1 to 3. Check HLP? answer to find out if WSL is supported.

E-51: Generator ID can be 1 to 3. WaveTableID can be 1 to 3.

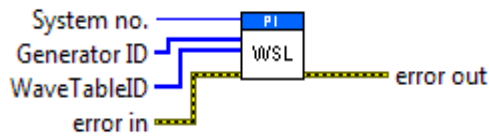
E-712: Generator ID can be 1 to No. of connected axes, WaveTableID can be 1 to 90

E-725: Generator ID can be 1 to No. of connected axes, WaveTableID can be 1 to 30

E-727: Generator ID can be 1 to No. of connected axes, WaveTableID can be 1 to 40

E-753: Generator ID can be 1, WaveTableID can be 1 to 10

E-754: Generator ID can be 1, WaveTableID can be 1 to 10



3.15.22. WSL?.vi

Returns current setting of wave table selection. If WaveTableID is zero no wave table is connected to the generator.

Valid for C-413, C-886, C-887, C-891, E-709, E-517, E-518, E-712, E-725, E-727, E-753, E-754 (but must be present for C-885 also)

C-413: Generator ID can be 1 to 2.

C-886: Generator ID can be 1 to No. of connected axes.

C-887: Check HLP? answer to find out if WSL? is supported. Generator ID can be 1 to No. of connected axes.

C-891: Generator ID can be 1.

E-709: Generator ID can be 1

E-517: Generator ID can be 1 to 3. Check HLP? answer to find out if WSL? is supported.

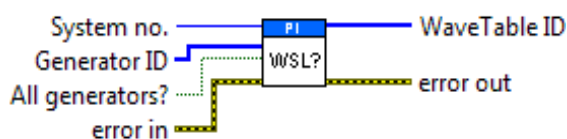
E-51: Generator ID can be 1 to 3.

E-712: Generator ID can be 1 to No. of connected axes

E-725, E-727: Generator ID can be 1 to No. of connected axes

E-753: Generator ID can be 1

E-754: Generator ID can be 1



3.15.23. WTR.vi

Set Wave Generator Table Rate. This VI will change the sampling interval of the wave generator. When Generator ID is zero all generators are selected, otherwise each generator is selected individually. Table Rate sets the duration of the wave table points as a multiple of the servo interval time. When a Table rate higher than 1 is set, Interpolation type can be used to join each wave table point by an interpolation.

Valid for: C-413, C-886, C-887, C-891, E-517, E-518, E-709, E-712, E-725, E-727, E-753, E-754 (but must be present for C-885 also)

C-413 : The following Interpolation types can be selected:

0 = no interpolation

Generator ID must be 0 because values can only be set for all generators.

C-886: The following Interpolation types can be selected:

0 = no interpolation, 1 = straight line

Table rate values and Interpolation type for any Generator ID are valid for all wave generators (cannot be set differently).

C-887: The following Interpolation types can be selected:

0 = no interpolation, 1 = straight line

Table rate values and Interpolation type for any Generator ID are valid for all wave generators (cannot be set differently). Check HLP? answer to find out if WTR is supported.

C-891: The following Interpolation types can be selected:

0 = no interpolation, 1 = straight line.

Generator ID can be 0 or 1 as there is only one generator.

E-517, E-518: Interpolation type must be 0. Generator ID can be 1 to 3. WTR sets the value of the Wave Generator Table Rate parameter, ID 0x13000109, in volatile memory. You can set the wave table rate also using the CFG wave type (see WAV command) or by directly changing the parameter with SPA.

E-709: The following Interpolation types can be selected:

0 = no interpolation

Generator ID can be 0 or 1 as there is only one generator.

E-712: The following Interpolation types can be selected:

0 = no interpolation, 1 = straight line

Generator ID must be 0 because values can only be set for all generators.

E-725, E-727: The following Interpolation types can be selected:

0 = no interpolation, 1 = straight line

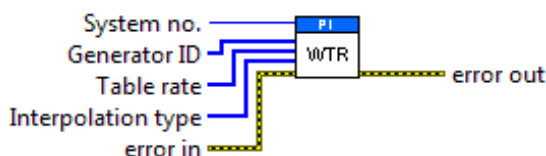
Generator ID must be 0 because values can only be set for all generators.

E-753: The following Interpolation types can be selected:

0 = no interpolation, 1 = straight line. Generator ID must be 0 because values can only be set for all generators.

E-754: The following Interpolation types can be selected:

0 = no interpolation, 1 = straight line. Generator ID must be 0 because values can only be set for all generators.



3.15.24. WTR?.vi

Returns the current settings of the wave generator table rate.

Valid for C-413, C-886, C-887, C-891, E-517, E-518, E-709, E-712, E-725, E-727, E-753, E-754 (but must be present for C-885 also)

C-413 : Generator ID can be 1 to 2

C-886: Values queried for one Generator ID are valid for all generators.

C-887: Values queried for one Generator ID are valid for all generators. Check HLP? answer to find out if WTR? is supported.

C-891: Generator ID can be 1.

E-517, E-518: Generator ID can be 1 to 3.

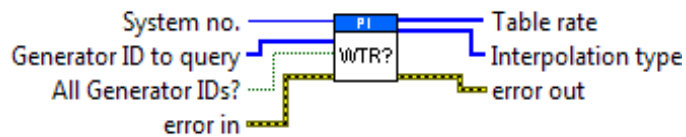
E-709: Generator ID can be 0 or 1 because there is only one generator.

E-712: Generator ID must be 0 because values are valid for all generators.

E-725, E-727: Generator ID must be 0 because values are valid for all generators.

E-753: Generator ID must be 0 because values are valid for all generators.

E-754: Generator ID must be 0 because values are valid for all generators.



4. Controls

4.1. Analog control.llb

4.1.1. Available Analog Commands.ctf

Type definition for available analog commands.

Valid for Analog systems (but must be present for all other systems, too)



4.2. Communication.llb

4.2.1. Available DLL interfaces.ctf

Type definition for hardware interfaces available when communicating with a system through a PI GCS DLL.

Valid for, C-843, C-843.PM, C-865, C-866, C-867, C-877, C-884, C-885, C-886, 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-872, E-873, E-874, Hydra, Pollux, Mercury, Mercury_GCS (but must be present for all other systems also)



4.2.2. Available DLLs.ctf

Type definition for available GCS DLLs for communicating with a system.

Valid for, C-843, C-843.PM, C-865, C-866, C-867, C-877, C-884, C-885, C-886, 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-872, E-873, E-874, Hydra, Pollux, Mercury, Mercury_GCS (but must be present for all other systems also)



4.2.3. Available interfaces.ctf

Type definition for available interfaces for communicating with a system.

Valid for All systems



4.2.4. Syntax.ctl

Type definition for GCS version.

Valid for All systems



4.2.5. Termination character.ctl

Type definition for termination character.

Valid for All systems



4.3. General command.llb

4.3.1. Controller names.ctl

Type definition for control Controller names.

Valid for All systems



5. Error Codes

Error codes are not unambiguous, but can result from a PI error message or LabVIEW internal error code. In addition to the list below see National Instruments error codes.

100	PI LabVIEW driver reports error. See source control for details
0	No error occurred during function call
-1	Error during com operation (could not be specified)
-2	Error while sending data
-3	Error while receiving data
-4	Not connected (no port with given ID open)
-5	Buffer overflow
-6	Error while opening port
-7	Timeout error
-8	There are more lines waiting in buffer
-9	There is no interface or DLL handle with the given ID
-10	Event/message for notification could not be opened
-11	Function not supported by this interface type

- 12 Error while sending \"echoed\" data
- 13 IEEE488: System error
- 14 IEEE488: Function requires GPIB board to be CIC
- 15 IEEE488: Write function detected no listeners
- 16 IEEE488: Interface board not addressed correctly
- 17 IEEE488: Invalid argument to function call
- 18 IEEE488: Function requires GPIB board to be SAC
- 19 IEEE488: I/O operation aborted
- 20 IEEE488: Interface board not found
- 21 IEEE488: Error performing DMA
- 22 IEEE488: I/O operation started before previous operation completed
- 23 IEEE488: No capability for intended operation
- 24 IEEE488: File system operation error
- 25 IEEE488: Command error during device call
- 26 IEEE488: Serial poll-status byte lost
- 27 IEEE488: SRQ remains asserted
- 28 IEEE488: Return buffer full
- 29 IEEE488: Address or board locked
- 30 RS-232: 5 data bits with 2 stop bits is an invalid combination, as is 6, 7, or 8 data bits with 1.5 stop bits
- 31 RS-232: Error configuring the COM port
- 32 Error dealing with internal system resources (events, threads, ...)
- 33 A DLL or one of the required functions could not be loaded
- 34 FTDIUSB: invalid handle
- 35 FTDIUSB: device not found
- 36 FTDIUSB: device not opened
- 37 FTDIUSB: IO error
- 38 FTDIUSB: insufficient resources
- 39 FTDIUSB: invalid parameter
- 40 FTDIUSB: invalid baud rate
- 41 FTDIUSB: device not opened for erase
- 42 FTDIUSB: device not opened for write
- 43 FTDIUSB: failed to write device
- 44 FTDIUSB: EEPROM read failed
- 45 FTDIUSB: EEPROM write failed
- 46 FTDIUSB: EEPROM erase failed
- 47 FTDIUSB: EEPROM not present
- 48 FTDIUSB: EEPROM not programmed
- 49 FTDIUSB: invalid arguments
- 50 FTDIUSB: not supported

- 51 FTDIUSB: other error
- 52 Error while opening the COM port: was already open
- 53 Checksum error in received data from COM port
- 54 Socket not ready, you should call the function again
- 55 Port is used by another socket
- 56 Socket not connected (or not valid)
- 57 Connection terminated (by peer)
- 58 Can't connect to peer
- 59 Operation was interrupted by a nonblocked signal
- 60 No Device with this ID is present
- 61 Driver could not be opened (on Vista: run as administrator!)
- 62 Host not found
- 63 Device already connected
- 1001 Unknown axis identifier
- 1002 Number for NAV out of range--must be in [1,10000]
- 1003 Invalid value for SGA--must be one of {1, 10, 100, 1000}
- 1004 Controller sent unexpected response
- 1005 No manual control pad installed, calls to SMA and related commands are not allowed
- 1006 Invalid number for manual control pad knob
- 1007 Axis not currently controlled by a manual control pad
- 1008 Controller is busy with some lengthy operation (e.g. reference move, fast scan algorithm)
- 1009 Internal error--could not start thread
- 1010 Controller is (already) in macro mode--command not valid in macro mode
- 1011 Controller not in macro mode--command not valid unless macro mode active
- 1012 Could not open file to write or read macro
- 1013 No macro with given name on controller, or macro is empty
- 1014 Internal error in macro editor
- 1015 One or more arguments given to function is invalid (empty string, index out of range, ...)
- 1016 Axis identifier is already in use by a connected stage
- 1017 Invalid axis identifier
- 1018 Could not access array data in COM server
- 1019 Range of array does not fit the number of parameters
- 1020 Invalid parameter ID given to SPA or SPA?
- 1021 Number for AVG out of range--must be >0
- 1022 Incorrect number of samples given to WAV
- 1023 Generation of wave failed

- 1024 Motion error: position error too large, servo is switched off automatically
- 1025 Controller is (already) running a macro
- 1026 Configuration of PZT stage or amplifier failed
- 1027 Current settings are not valid for desired configuration
- 1028 Unknown channel identifier
- 1029 Error while reading/writing wave generator parameter file
- 1030 Could not find description of wave form. Maybe WG.INI is missing?
- 1031 The WGWaveEditor DLL function was not found at startup
- 1032 The user cancelled a dialog
- 1033 Error from C-844 Controller
- 1034 DLL necessary to call function not loaded, or function not found in DLL
- 1035 The open parameter file is protected and cannot be edited
- 1036 There is no parameter file open
- 1037 Selected stage does not exist
- 1038 There is already a parameter file open. Close it before opening a new file
- 1039 Could not open parameter file
- 1040 The version of the connected controller is invalid
- 1041 Parameter could not be set with SPA--parameter not defined for this controller!
- 1042 The maximum number of wave definitions has been exceeded
- 1043 The maximum number of wave generators has been exceeded
- 1044 No wave defined for specified axis
- 1045 Wave output to axis already stopped/started
- 1046 Not all axes could be referenced
- 1047 Could not find parameter set required by frequency relation
- 1048 Command ID given to SPP or SPP? is not valid
- 1049 A stage name given to CST is not unique
- 1050 A uuencoded file transfered did not start with \"begin\" followed by the proper filename
- 1051 Could not create/read file on host PC
- 1052 Checksum error when 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
- 1067 ProfileGenerator: No profile has been created yet
- 1068 ProfileGenerator: Generated profile exceeds limits of one or both axes
- 1069 ProfileGenerator: Unknown parameter ID in Set/Get Parameter command
- 1070 ProfileGenerator: Parameter out of allowed range
- 1071 User Profile Mode: Out of memory
- 1072 User Profile Mode: Cluster is not assigned to this axis
- 1073 Unknown cluster identifier
- 1074 The installed device driver doesn't match the required version. Please see the documentation to determine the required device driver version.
- 1075 The library used doesn't match the required version. Please see the documentation to determine the required library version.
- 1076 The interface is currently locked by another function. Please try again later.
- 1077 Version of parameter DAT file does not match the required version. Current files are available at www.pi.ws.
- 1078 Cannot write to parameter DAT file to store user defined stage type.
- 1079 Cannot create parameter DAT file to store user defined stage type.
- 1080 Parameter DAT file does not have correct revision.
- 1081 User stages DAT file does not have correct revision.
- 1082 Timeout Error. Some lengthy operation did not finish within expected time.
- 1083 A function argument has an unexpected datatype.
- 1084 Length of data arrays is different.
- 1085 Parameter value not found in parameter DAT file.
- 1086 Macro recording is not allowed in this mode of operation.
- 1087 command cancelled by user input.
- 1088 Controller sent too few GCS data sets

- 1089 Controller sent too many GCS data sets
- 1090 Communication error while reading GCS data
- 1091 Wrong number of input arguments.
- 1092 Change of command level has failed.
- 1093 Switching off the servo mode has failed.
- 1094 A parameter could not be set while performing CST: CST was not performed (parameters remain unchanged).
- 1095 Connection could not be reestablished after reboot.
- 1096 Sending HPA? or receiving the response has failed.
- 1097 HPA? response does not comply with GCS2 syntax.
- 1098 Response to SPA? could not be received.
- 1099 Version of PAM file cannot be handled (too old or too new)
- 1100 PAM file does not contain required data in PAM-file format
- 1101 Information does not contain all required data
- 1102 No value for parameter available
- 1103 No PAM file is open
- 1104 Invalid value
- 1105 Unknown parameter
- 1106 Response to SEP? could not be received.
- 1107 Response to SPA? could not be received.
- 1108 Error while performing CST: One or more parameters were not set correctly.
- 1109 PAM file has duplicate entry with different values.
- 1110 File has no signature
- 1111 File has invalid signature
- 10000 PI stage database: String containing stage type and description has invalid format.
- 10001 PI stage database: Database does not contain the selected stage type for the connected controller.
- 10002 PI stage database: Establishing the connection has failed.
- 10003 PI stage database: Communication was interrupted (e.g. because database was deleted).
- 10004 PI stage database: Querying data failed.
- 10005 PI stage database: System already exists. Rename stage and try again.
- 10006 PI stage database: Response to HPA? contains unknown parameter IDs.
- 10007 PI stage database: Inconsistency between database and response to HPA?.
- 10008 PI stage database: Stage has not been added.
- 10009 PI stage database: Stage has not been removed.

-10010	Controller does not support all stage parameters stored in PI stage database. No parameters were set.
-10011	The version of PISTAGES3.DB stage database is out of date. Please update via PIUpdateFinder. No parameters were set.
-10012	Mismatch between number of parameters present in stage database and available in controller interface. No parameters were set.
-10013	Mismatch between number of parameters present in stage database and available in controller interface. Some parameters were ignored.
-10014	One or more parameters could not be set correctly on the controller.
-10015	One or more parameter definitions are not present in stage database. Please update PISTAGES3.DB via PIUpdateFinder. Missing parameters were ignored.
0	No error
1	Parameter syntax error
2	Unknown command
3	Command length out of limits or command buffer overrun
4	Error while scanning
5	Unallowable move attempted on unreferenced axis, or move attempted with servo off
6	Parameter for SGA not valid
7	Position out of limits
8	Velocity out of limits
9	Attempt to set pivot point while U,V and W not all 0
10	Controller was stopped by command
11	Parameter for SST or for one of the embedded scan algorithms out of range
12	Invalid axis combination for fast scan
13	Parameter for NAV out of range
14	Invalid analog channel
15	Invalid axis identifier
16	Invalid stage name
17	Parameter out of range
18	Invalid macro name
19	Error while recording macro
20	Macro not found
21	Axis has no brake
22	Axis identifier specified more than once
23	Illegal axis or channel
24	Incorrect number of parameters
25	Invalid floating point number
26	Parameter missing

27	Soft limit out of range
28	No manual pad found
29	No more step-response values
30	No step-response values recorded
31	Axis has no reference sensor
32	Axis has no limit switch
33	No relay card installed
34	Command not allowed for selected stage(s)
35	No digital input installed
36	No digital output configured
37	No more MCM responses
38	No MCM values recorded
39	Controller number invalid
40	No joystick configured
41	Invalid axis for electronic gearing, axis can not be slave
42	Position of slave axis is out of range
43	Slave axis cannot be commanded directly when electronic gearing is enabled
44	Calibration of joystick failed
45	Referencing failed
46	OPM (Optical Power Meter) missing
47	OPM (Optical Power Meter) not initialized or cannot be initialized
48	OPM (Optical Power Meter) Communication Error
49	Move to limit switch failed
50	Attempt to reference axis with referencing disabled
51	Selected axis is controlled by joystick
52	Controller detected communication error
53	Command is not allowed while the affected axis is in motion.
54	Unknown parameter
55	No commands were recorded with REP
56	Password invalid
57	Data Record Table does not exist
58	Source does not exist; number too low or too high
59	Source Record Table number too low or too high
60	Protected Param: current Command Level (CCL) too low
61	Command execution not possible while Autozero is running
62	Autozero requires at least one linear axis
63	Initialization still in progress
64	Parameter is read-only
65	Parameter not found in non-volatile memory

- 66 Voltage out of limits
- 67 Not enough memory available for requested wave curve
- 68 Not enough memory available for DDL table; DDL can not be started
- 69 Time delay larger than DDL table; DDL can not be started
- 70 The requested arrays have different lengths; query them separately
- 71 Attempt to restart the generator while it is running in single step mode
- 72 Motion commands and wave generator activation are not allowed when analog target is active
- 73 Motion commands are not allowed when wave generator output is active; use WGO to disable generator output
- 74 No sensor channel or no piezo channel connected to selected axis (sensor and piezo matrix)
- 75 Generator started (WGO) without having selected a wave table (WSL).
- 76 Interface buffer did overrun and command couldn't be received correctly
- 77 Data Record Table does not hold enough recorded data
- 78 Data Record Table is not configured for recording
- 79 Open-loop commands (SVA, SVR) are not allowed when servo is on
- 80 Hardware error affecting RAM
- 81 Not macro command
- 82 Macro counter out of range
- 83 Joystick is active
- 84 Motor is off
- 85 Macro-only command
- 86 Invalid joystick axis
- 87 Joystick unknown
- 88 Move without referenced stage
- 89 Command not allowed in current motion mode
- 90 No tracing possible while digital IOs are used on this HW revision. Reconnect to switch operation mode.
- 91 Move not possible, would cause collision
- 92 Stage is not capable of following the master. Check the gear ratio(SRA).
- 93 This command is not allowed while the affected axis or its master is in motion.
- 94 Servo cannot be switched on when open-loop joystick control is enabled.
- 95 This parameter cannot be changed in current servo mode.
- 96 Unknown stage name
- 97 Invalid length of value (too much characters)

98	AutoZero procedure was not successful
99	Sensor voltage is off
100	PI LabVIEW driver reports error. See source control for details.
200	No stage connected to axis
201	File with axis parameters not found
202	Invalid axis parameter file
203	Backup file with axis parameters not found
204	PI internal error code 204
205	SMO with servo on
206	uudecode: incomplete header
207	uudecode: nothing to decode
208	uudecode: illegal UUE format
209	CRC32 error
210	Illegal file name (must be 8-0 format)
211	File not found on controller
212	Error writing file on controller
213	VEL command not allowed in DTR Command Mode
214	Position calculations failed
215	The connection between controller and stage may be broken
216	The connected stage has driven into a limit switch, some controllers need CLR to resume operation
217	Strut test command failed because of an unexpected strut stop
218	While MOV! is running position can only be estimated!
219	Position was calculated during MOV motion
220	FIFO buffer underrun during interpolation
221	FIFO buffer overflow during interpolation
230	Invalid handle
231	No bios found
232	Save system configuration failed
233	Load system configuration failed
301	Send buffer overflow
302	Voltage out of limits
303	Open-loop motion attempted when servo ON
304	Received command is too long
305	Error while reading/writing EEPROM
306	Error on I2C bus
307	Timeout while receiving command
308	A lengthy operation has not finished in the expected time
309	Insufficient space to store macro
310	Configuration data has old version number

311	Invalid configuration data
333	Internal hardware error
400	Wave generator index error
401	Wave table not defined
402	Wave type not supported
403	Wave length exceeds limit
404	Wave parameter number error
405	Wave parameter out of range
406	WGO command bit not supported
500	The \"red knob\" is still set and disables system
501	The \"red knob\" was activated and still disables system - reanimation required
502	Position consistency check failed
503	Hardware collision sensor(s) are activated
504	Strut following error occurred, e.g. caused by overload or encoder failure
505	One sensor signal is not valid
506	Servo loop was unstable due to wrong parameter setting and switched off to avoid damage.
507	digital connection to external spi slave device is lost
508	Move attempt not permitted due to customer or limit settings
509	Emergency stop caused by trigger input
530	A command refers to a coordinate system that does not exist
531	A command refers to a coordinate system that has no parent node
532	Attempt to delete or change a coordinate system that is in use
533	Definition of a coordinate system is cyclic
536	Coordinate system cannot be defined as long as Hexapod is in motion
537	Coordinate system type is not intended for manual enabling
539	A coordinate system cannot be linked to itself
540	Coordinate system definition is erroneous or not complete (replace or delete it)
542	The coordinate systems are not part of the same chain
543	Unused coordinate system must be deleted before new coordinate system can be stored
544	With this coordinate system type SPI usage is not supported
545	Soft limits invalid due to changes in coordinate system
546	Coordinate system is write protected
547	Coordinate system cannot be changed because its content is loaded from a configuration file
548	Coordinate system may not be linked

549	A KSB-type coordinate system can only be rotated by multiples of 90 degrees
551	This query is not supported for this coordinate system type
552	This combination of work and tool coordinate systems does not exist
553	The combination must consist of one work and one tool coordinate system
554	This coordinate system type does not exist
555	BasMac: unknown controller error
556	No coordinate system of this type is enabled
557	Name of coordinate system is invalid
558	File with stored CS systems is missing or erroneous
559	File with leveling CS is missing or erroneous
601	Not enough memory
602	Hardware voltage error
603	Hardware temperature out of range
604	Position error of any axis in the system is too high
606	Maximum value of input signal has been exceeded
607	Value is not integer
608	Fast alignment process cannot be paused because it is not running
609	Fast alignment process cannot be restarted/resumed because it is not paused
650	Parameter could not be set with SPA - SEP needed?
651	Phase finding error
652	Sensor setup error
653	Sensor communication error
654	Motor amplifier error
655	Overcurrent protection triggered by I2T-module
656	Overcurrent protection triggered by amplifier module
657	Safety stop triggered
658	Sensor off?
700	Command not allowed in external mode
710	External mode communication error
715	Invalid mode of operation
716	Firmware stopped by command (#27)
717	External mode driver missing
718	Missing or incorrect configuration of external mode
719	External mode cycletime invalid
720	Brake is activated
731	Command not allowed while surface detection is running
732	Last surface detection failed

733	Fieldbus is active and is blocking GCS control commands
1000	Too many nested macros
1001	Macro already defined
1002	Macro recording not activated
1003	Invalid parameter for MAC
1004	Deleting macro failed
1005	Controller is busy with some lengthy operation (e.g. reference move, fast scan algorithm)
1006	Invalid identifier (invalid special characters, ...)
1007	Variable or argument not defined
1008	Controller is (already) running a macro
1009	Invalid or missing operator for condition. Check necessary spaces around operator.
1010	No answer was received while executing WAC/MEX/JRC/...
1011	Command not valid during macro execution
1024	Motion error: position error too large, servo is switched off automatically
1025	Maximum motor output reached
1063	User Profile Mode: Command is not allowed, check for required preparatory commands
1064	User Profile Mode: First target position in User Profile is too far from current position
1065	Controller is (already) in User Profile Mode
1066	User Profile Mode: Block or Data Set index out of allowed range
1071	User Profile Mode: Out of memory
1072	User Profile Mode: Cluster is not assigned to this axis
1073	Unknown cluster identifier
1090	There are too many open tcpip connections
2000	Controller already has a serial number
4000	Sector erase failed
4001	Flash program failed
4002	Flash read failed
4003	HW match code missing/invalid
4004	FW match code missing/invalid
4005	HW version missing/invalid
4006	FW version missing/invalid
4007	FW update failed
4008	FW Parameter CRC wrong
4009	FW CRC wrong
5000	PicoCompensation scan data is not valid
5001	PicoCompensation is running, some actions can not be executed

- during scanning/recording
- 5002 Given axis can not be defined as PPC axis
 - 5003 Defined scan area is larger than the travel range
 - 5004 Given PicoCompensation type is not defined
 - 5005 PicoCompensation parameter error
 - 5006 PicoCompensation table is larger than maximum table length
 - 5100 Common error in Nexline firmware module
 - 5101 Output channel for Nexline can not be redefined for other usage
 - 5102 Memory for Nexline signals is too small
 - 5103 RNP can not be executed if axis is in closed loop
 - 5104 relax procedure (RNP) needed
 - 5200 Axis must be configured for this action
 - 5300 Frequency analysis failed
 - 5301 Another frequency analysis is running
 - 6000 Invalid preset value of absolute sensor
 - 6001 Error while writing to sensor
 - 6002 Error while reading from sensor
 - 6003 Checksum error of absolute sensor
 - 6004 General error of absolute sensor
 - 6005 Overflow of absolute sensor position



End of document