

USER MANUAL

ProDAQ Signal Conditioning

ProDAQ 5716 16-Ch. Bridge Signal Conditioning Unit



PUBLICATION NUMBER: 5716-XX-UM-0020



Copyright, © 2013-2017, Bustec Production, Ltd.

Bustec Production, Ltd.
Bustec House, Shannon Business Park, Shannon, Co. Clare, Ireland
Tel: +353 (0) 61 707100, FAX: +353 (0) 61 707106

PROPRIETARY NOTICE

This document and the technical data herein disclosed, are proprietary to Bustec Production Ltd., and shall not, without express written permission of Bustec Production Ltd, be used, in whole or in part to solicit quotations from a competitive source or used for manufacture by anyone other than Bustec Production Ltd. The information herein has been developed at private expense, and may only be used for operation and maintenance reference purposes or for purposes of engineering evaluation and incorporation into technical specifications and other documents, which specify procurement of products from Bustec Production Ltd.. This document is subject to change without further notification. Bustec Production Ltd. Reserve the right to change both the hardware and software described herein.

Table of Contents

1. Introduction	7
1.1. Overview	7
2. Getting Started	9
2.1. Mounting the ProDAQ 5716 in a 19" Rack	9
2.1.1. The ProDAQ 5725 Rack-mount Kit.....	9
2.1.2. Installing the ProDAQ 5725 Rack-mount Kit.....	10
2.2. Preparing the ProDAQ 5716 for Tabletop Use	14
2.2.1. The ProDAQ 5726 Stackable Desktop Feet Set	14
2.2.2. Attaching the ProDAQ 5726 Stackable Desktop Feet Set	15
2.3. ProDAQ 5716 Connectors and Switches	16
2.3.1. Front Panel ProDAQ 5716-Ax.....	16
2.3.2. Front Panel ProDAQ 5716-Bx.....	16
2.3.3. Rear Panel ProDAQ 5716-xx.....	17
2.4. Connecting the ProDAQ 5716 to an ADC Function Card	18
2.5. Connecting your Sensors	19
2.5.1. Full-Bridge	19
2.5.2. Half-Bridge	19
2.5.3. Quarter-Bridge.....	20
2.5.4. Voltage/ICP	20
3. Theory of Operation	21
3.1. Full Bridge Mode	22
3.2. Half Bridge Mode	23
3.3. Quarter Bridge Mode	24
3.4. IEPE/ICP™ Mode	25
3.5. Differential Voltage Mode.....	25
3.6. Single-Ended Voltage Mode.....	26
4. The ProDAQ 5716 Soft Front Panel Plug-in	29
4.1.1. Channel Configuration	30
4.1.2. Calibration	31
4.1.3. Status	31
5. Programming the ProDAQ 5716.....	33
5.1. VXIplug&play Driver Organization.....	33
5.2. Connecting to the Function Card and Signal Conditioning Unit.....	34
6. VXIplug&play Driver Functions.....	37
6.1. VXIplug&play Driver Function Details.....	39
6.1.1. bu5716_autobalance	39
6.1.2. bu5716_channelIdentify.....	40
6.1.3. bu5716_close	41
6.1.4. bu5716_error_message.....	42
6.1.5. bu5716_excitCalibration	43
6.1.6. bu5716_getChannelConf.....	45
6.1.7. bu5716_getCoupling	47
6.1.8. bu5716_getDAC	48

6.1.9.	bu5716_getExcCalibCoeff	50
6.1.10.	bu5716_getExcitation.....	52
6.1.11.	bu5716_getExcitationMonitor	54
6.1.12.	bu5716_getFCLastError.....	55
6.1.13.	bu5716_getGain	56
6.1.14.	bu5716_getInputSrc.....	57
6.1.15.	bu5716_getMode	58
6.1.16.	bu5716_getPgaCalibCoeff	60
6.1.17.	bu5716_getShuntCal	62
6.1.18.	bu5716_getStatus	63
6.1.19.	bu5716_getVoltRefInfo	65
6.1.20.	bu5716_getVoltRefOutput.....	66
6.1.21.	bu5716_init	68
6.1.22.	bu5716_pgaCalibration	70
6.1.23.	bu5716_readTEDS	72
6.1.24.	bu5716_readTemperature	74
6.1.25.	bu5716_reset	75
6.1.26.	bu5716_resetCalibCoeff	76
6.1.27.	bu5716_revision_query	77
6.1.28.	bu5716_serialNumber.....	78
6.1.29.	bu5716_setChannelConf	79
6.1.30.	bu5716_setCoupling	81
6.1.31.	bu5716_setDAC.....	83
6.1.32.	bu5716_setExcitation.....	85
6.1.33.	bu5716_setExcitationMonitor	87
6.1.34.	bu5716_setGain.....	88
6.1.35.	bu5716_setInputSrc.....	89
6.1.36.	bu5716_setMode	90
6.1.37.	bu5716_setScuErrorMask.....	92
6.1.38.	bu5716_setVoltRefOutput.....	95
6.1.39.	bu5716_storeCalibCoeff	97
6.1.40.	bu5716_writeTEDS	98
7.	Specifications.....	101
7.1.	Available Versions.....	101
7.2.	Signal Conditioning.....	101
7.3.	Power Supply.....	102
7.4.	Environmental Specifications.....	102

Table of Figures

Figure 1 - ProDAQ 5716 Bridge Signal Conditioning Unit	7
Figure 2 - ProDAQ 5725 Rack-mount Kit Parts.....	9
Figure 3 - Assembling the Rack-mount Brackets	10
Figure 4 - Rack-mount Brackets Attachment Positions	10
Figure 5 - Rack-mount Brackets Attachment Positions for Reverse Mounting	11
Figure 6 - Assembling the rear support rails	12
Figure 7 - Final Steps to mount the ProDAQ 5720 in a 19" Rack	13
Figure 8 - ProDAQ 5726 Stackable Desktop Feet Set Parts.....	14
Figure 9 - Attaching the ProDAQ 5726 Stackable Desktop Feet Set.....	15
Figure 10 - ProDAQ 5716 Rear Panel	17
Figure 11 - Connecting to ADC Function Card(s).....	18
Figure 12 - Full-Bridge Sensor Configuration.....	19
Figure 13 - Half-Bridge Sensor Configuration	19
Figure 14 - Quarter-Bridge Sensor Configuration.....	20
Figure 15 - ProDAQ 5716 Block Diagram	21
Figure 16 – Full Bridge Configuration.....	22
Figure 17 – Half Bridge Configuration.....	23
Figure 18 – Quarter Bridge Configuration (350Ω shown)	24
Figure 19 – Fitting a User Resistor	24
Figure 20 – ICP™ Mode Configuration	25
Figure 21 – Differential Voltage Mode Configuration.....	26
Figure 22 – Single Ended Voltage Mode Configuration.....	27
Figure 23 - ProDAQ 3424 Soft Front Panel with ProDAQ 5716 Tab	29
Figure 24 - ProDAQ 5716 Channel Configuration Tab.....	30
Figure 25 - VXIplug&play Driver Organization.....	33
Figure 26 - Connecting to The ProDAQ 3416 and ProDAQ 5716	35

Reference Documents

Title	Number
ProDAQ 3416 User Manual	3416-XX-UM
ProDAQ 3424 User Manual	3424-XX-UM
ProDAQ 6100 User Manual	6100-XX-UM
ProDAQ 3180 Hardware Manual	3180-XX-HM

Glossary

Safety



This equipment contains voltage hazardous to human life and safety and is able to inflict personal injury. Disconnect the device from the AC line (mains) before opening the covers as described in chapter 3.4.



To operate this device, use a three-conductor power cord and an power outlet providing protective earth. Do not use a two-conductor extension cord or a three-prong/two-prong adapter.



If you replace the power cord provided, make sure that the replacement is rated for the power consumption stated in the specifications.

Do not position the device so that it is difficult to operate the disconnecting device.

If the equipment is used in a manner not specified by the manufacturer, its safety may be impaired.

Waste Electrical and Electronic Equipment (WEEE)



This product complies with the WEEE Directive 2002/96/EC marking requirement. The affixed product label indicates that you must not discard this electrical product in domestic household waste.

Product Category: Monitoring and Control Instrumentation

To return unwanted products, contact Bustec Ltd.

1. Introduction

1.1. Overview

The ProDAQ 5716 is a 1U high rack-mountable signal conditioning box for standard 19" racks. It provides signal conditioning for up to 16 strain gage or ICP sensors. Either one ProDAQ 3416 16-Channel ADC Function Card or up to two ProDAQ 3424 8-Channel 24-bit Sigma-Delta ADC Function Cards can be connected to the unit.



Figure 1 - ProDAQ 5716 Bridge Signal Conditioning Unit

The bridge signal conditioning provides a per-channel programmable excitation voltage of up to 20V by allowing the positive and negative excitation voltage to be programmed individually between 0..10V (positive side) and 0..-10V (negative side) with the current limited to 50mA per channel. By using remote sense lines per channel any losses due to lead resistance will be automatically compensated. Built-in high-precision resistors of 120 and 350 Ohm as well as sockets for user installable resistors allow for flexible quarter- and half-bridge completion.

Each input channel is equipped with a shunt calibration resistor and provisions for removing bridge offset voltage. A trifilar common mode filter will remove any unwanted high-frequency common mode signals. Each channel also contains selectable AC/DC coupling and ICP signal conditioning circuitry (optional). Both TEDS class 1 and class 2 are supported..

This page was intentionally left blank.

2. Getting Started

2.1. Mounting the ProDAQ 5716 in a 19" Rack

The ProDAQ 5725 Rack-mount Kit is available to mount the ProDAQ 5716 into standard 19" racks. The kit allows to mount the unit into racks with mounting depths between 440 mm and 810 mm.

2.1.1. The ProDAQ 5725 Rack-mount Kit

The ProDAQ 5725 Rack-mount Kit contains the following parts:

1. Front rack-mount bracket (2 pieces)
2. Rear support rail (2 pieces)
3. Rear rack-mount bracket (2 pieces)
4. Front handle (2 pieces)
5. M3 x 6mm countersunk screw (TORX, T10, 20 pieces)
6. M4 x 12mm panhead screw (TORX, T20, 4 pieces)
7. M4 washer (4 pieces)
8. M4 nut (4 pieces)

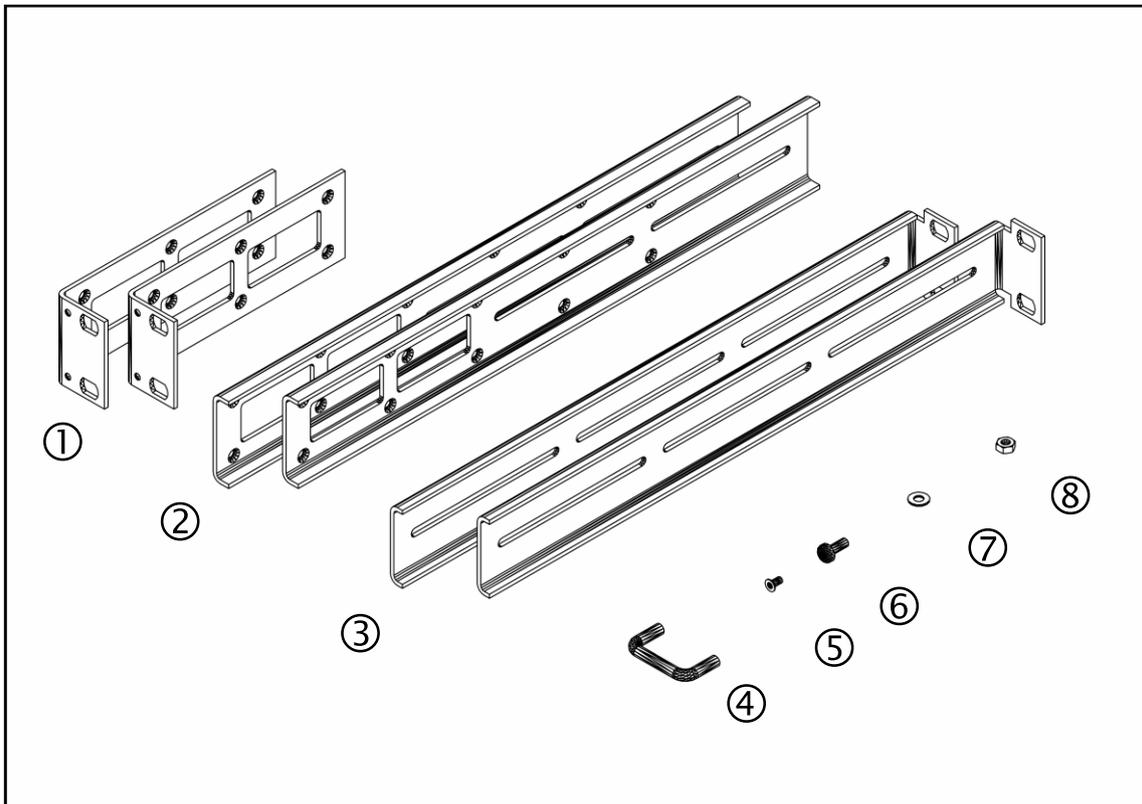


Figure 2 - ProDAQ 5725 Rack-mount Kit Parts

2.1.2. Installing the ProDAQ 5725 Rack-mount Kit

To install the rack-mount kit, first attach the handles to the rack-mount brackets by using two M3 x 6mm countersunk screws each (see Figure 3).

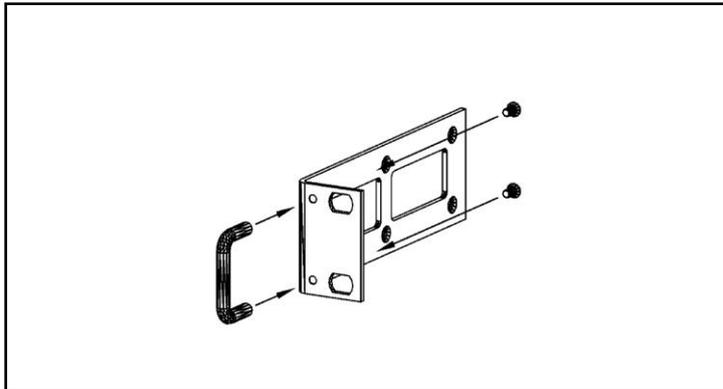


Figure 3 - Assembling the Rack-mount Brackets

Then attach one of the rack-mount brackets on each side of the ProDAQ 5720 by using four M3 x 6mm countersunk screws each. The brackets can be attached in two different positions, see Figure 4, ① and ②.

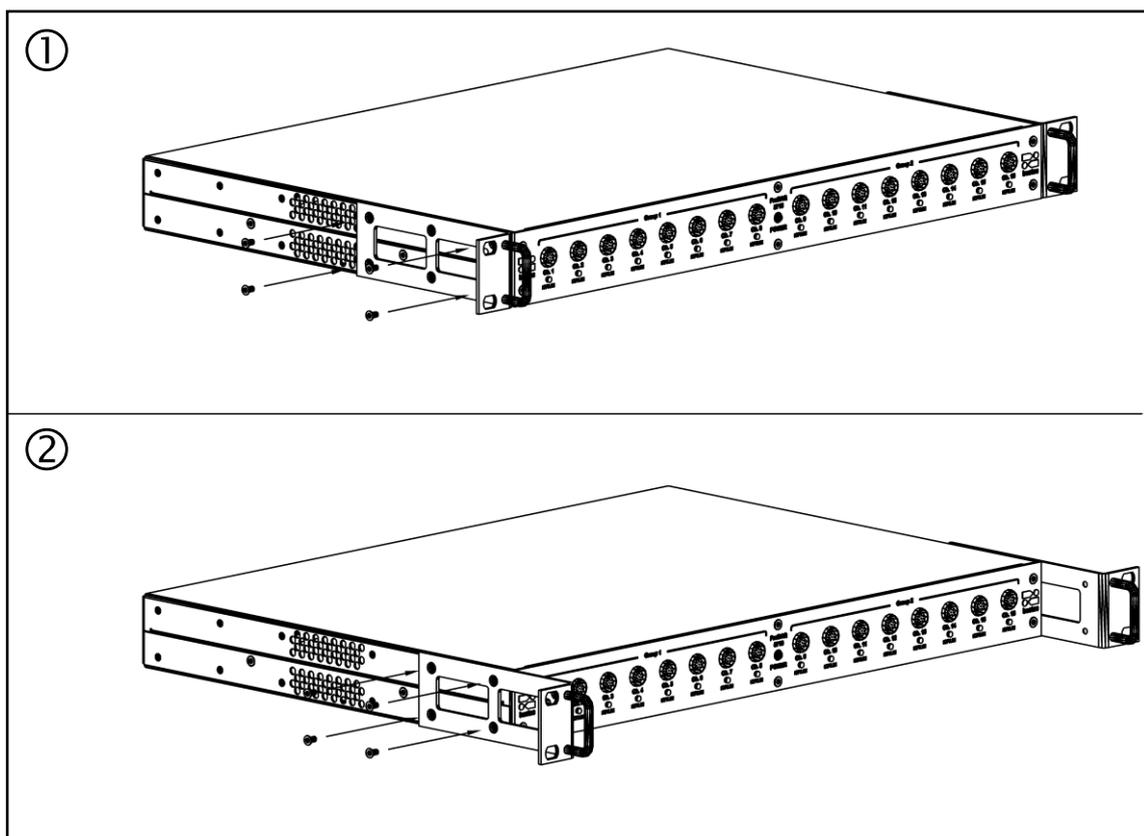


Figure 4 - Rack-mount Brackets Attachment Positions

Position 2 can be used to provide more space for connectors/cables or to allow for a deeper rack.

The mounting holes for the rack-mount kit on the ProDAQ 5720 are placed identically in both front and rear. If necessary, the unit can be mounted in a rack with its rear towards the front of the rack (reverse mounting, see Figure 5). As before the brackets can be attached in two different positions.

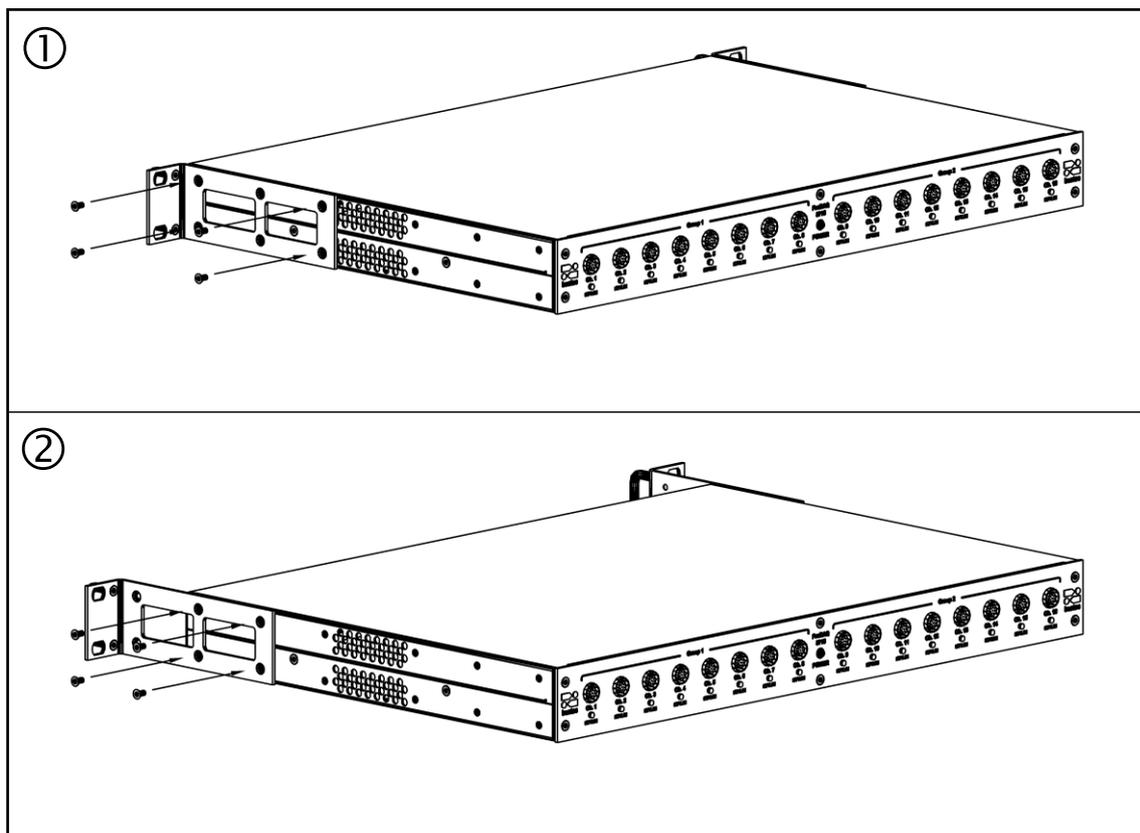


Figure 5 - Rack-mount Brackets Attachment Positions for Reverse Mounting

Next, attach one of the rear support rails to each side of the ProDAQ 5720 by using four M3 x 6mm countersunk screws each. The rear support rail can be mounted in four different positions to accommodate various rack depths and mounting positions (see Figure 6, positions 1 to 4).

Position 4 can only be used with the frontal rack-mount bracket is in position 2 (see Figure 4). When reverse mounting is used, Position 4 should not be used at all to not block off the airflow.

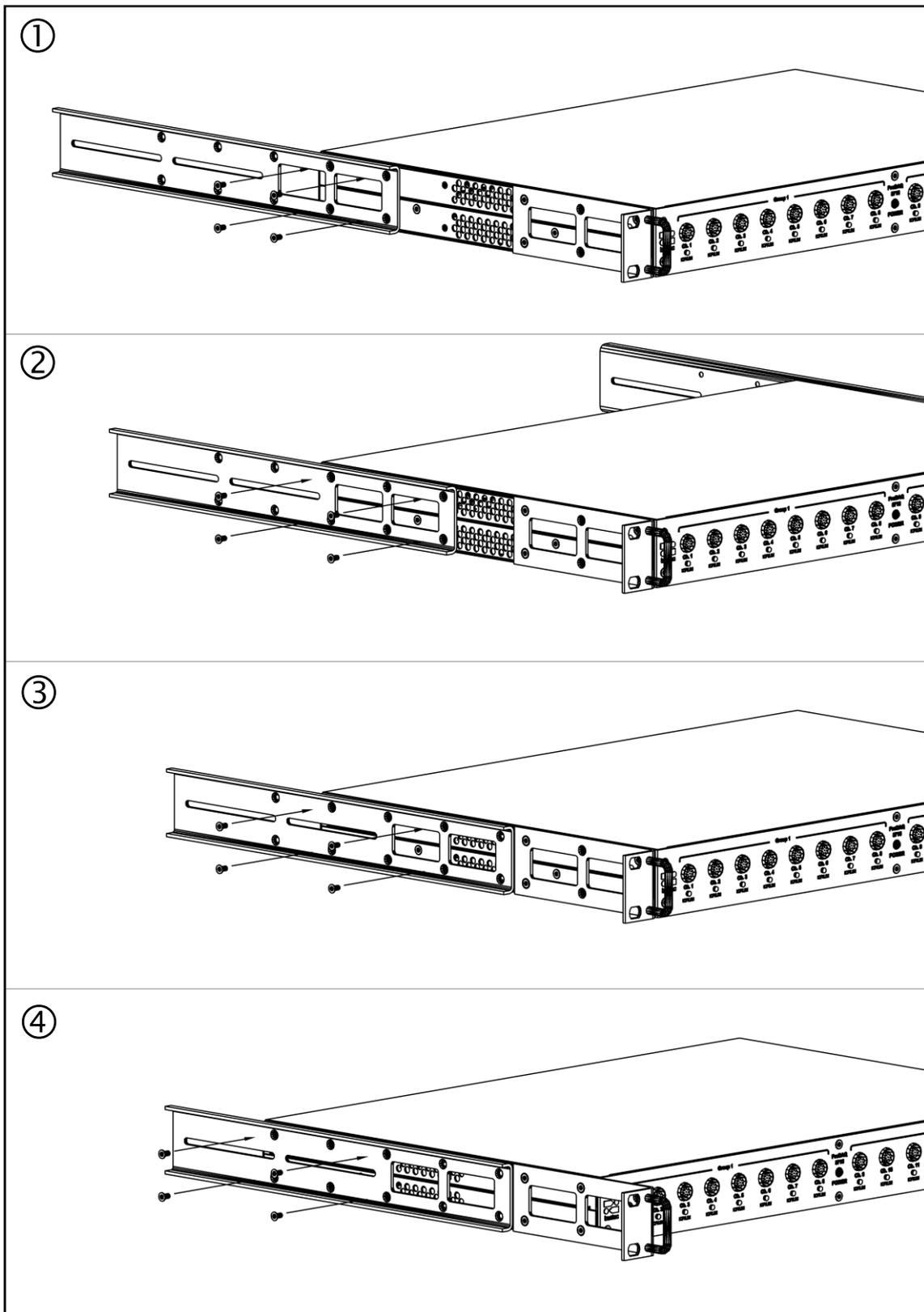


Figure 6 - Assembling the rear support rails

Next install the rear rack-mount brackets in your rack at the desired positions using the proper fixing materials for the rack used (see Figure 7, ①). Rack fixing materials are not part of the ProDAQ 5725 Rack-mount Kit.

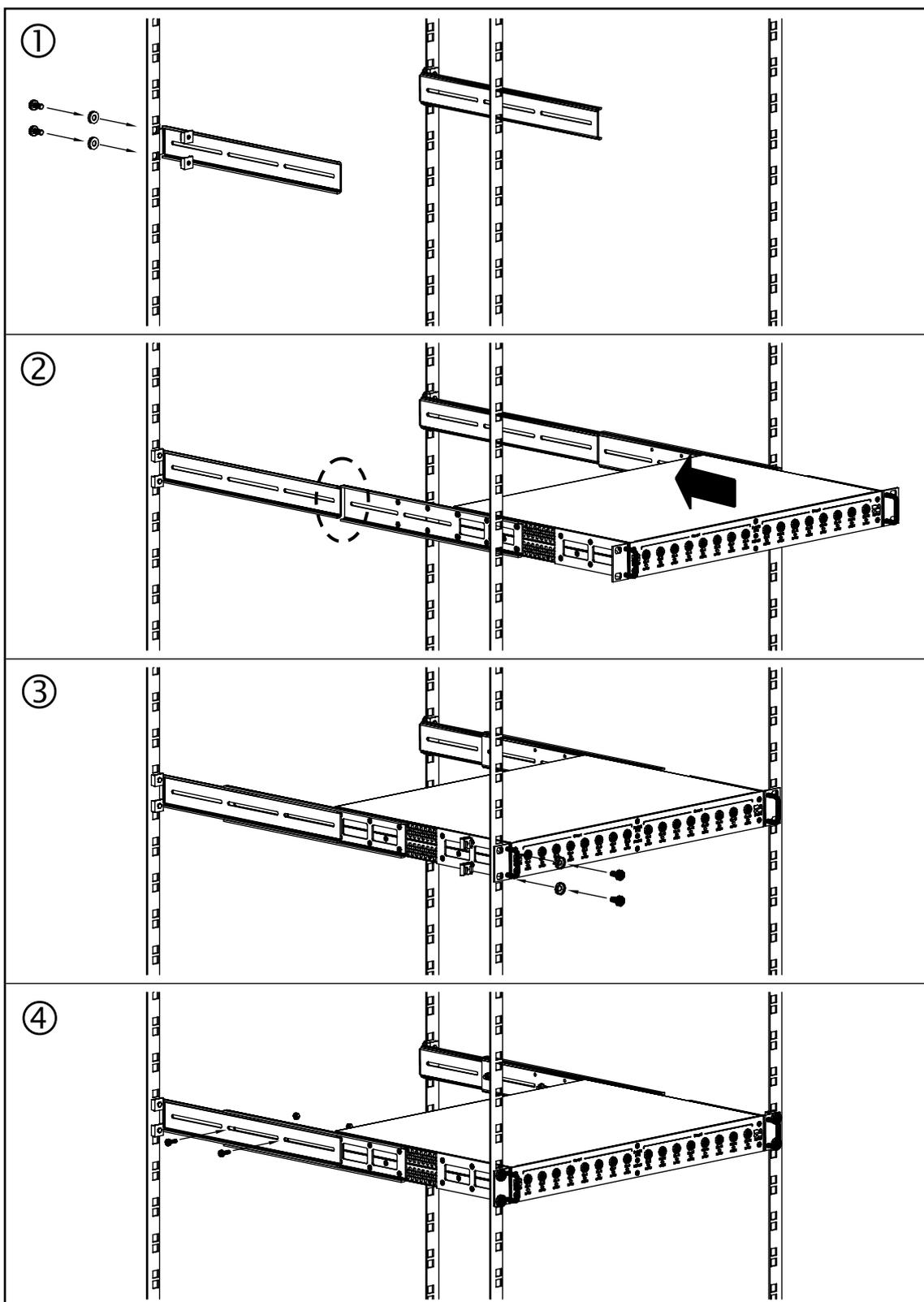


Figure 7 - Final Steps to mount the ProDAQ 5720 in a 19" Rack

Next install the ProDAQ 5716 into the rack at the desired position. To do so, slide the ProDAQ 5716 into the rack from the front (see Figure 7, ②). Take care that the rear support rails attached to the ProDAQ 5720 and the rear rack-mount brackets slide properly into each other. Attach the front rack-mount brackets to the rack using the proper fixing materials for the rack used (see Figure 7, ③).

Finally attach the rear support rail on each side to the already installed rear rack-mount bracket by using 2 pieces M4 x 12mm panhead screws, washers and nuts on each side. If only one slotted hole is accessible, position the screws that they are as far apart as possible (see Figure 7, ④).

WARNING

Do not use other screws than metric M3 for attaching the front rack-mount bracket or the rear support rail to the enclosure. Using any other screw style will damage the thread permanently.

WARNING

Do not use screws longer than 6mm to attach the front rack-mount bracket or the rear support rail to the enclosure. Using any other length might damage the unit and/or cause electrical shorts.

2.2. Preparing the ProDAQ 5716 for Tabletop Use

The ProDAQ 5726 Stackable Feet Set is available to provide support to use the ProDAQ 5716 on a desk etc. The feet set allows to stack several ProDAQ 5716 in a secured way.

2.2.1. The ProDAQ 5726 Stackable Desktop Feet Set

The ProDAQ 5726 Stackable Desktop Feet Set contains the following parts:

1. Stackable feet (4 pieces, with rubber feet attached)
2. M3 x 6mm countersunk screw (TORX, T10, 8 pieces)

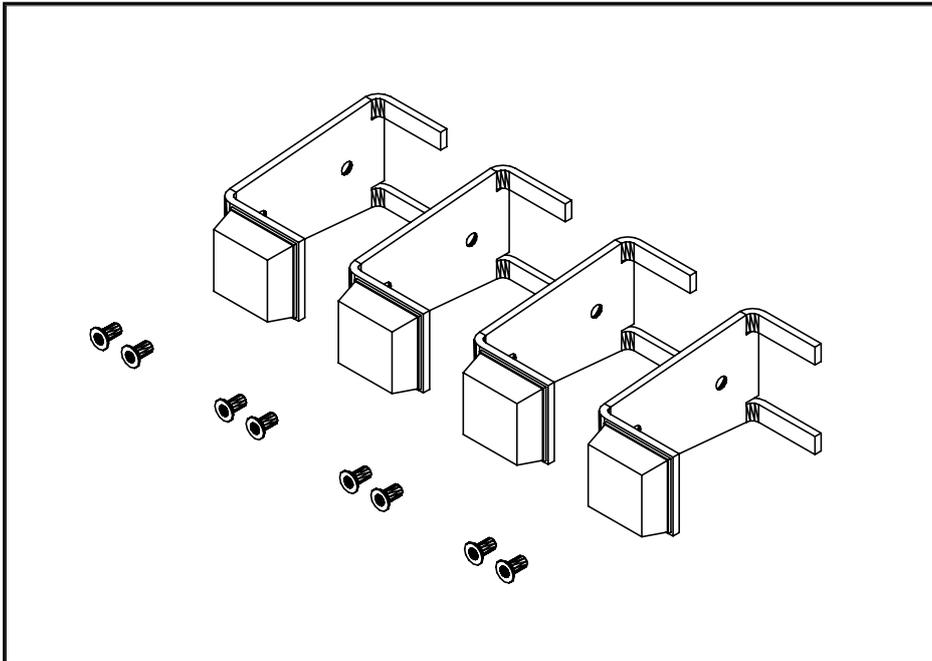


Figure 8 - ProDAQ 5726 Stackable Desktop Feet Set Parts

2.2.2. Attaching the ProDAQ 5726 Stackable Desktop Feet Set

To attach the stackable feet to the ProDAQ 5716 enclosure, use 2 pieces M3 x 6mm countersunk screws per feet as shown in Figure 9, ①.

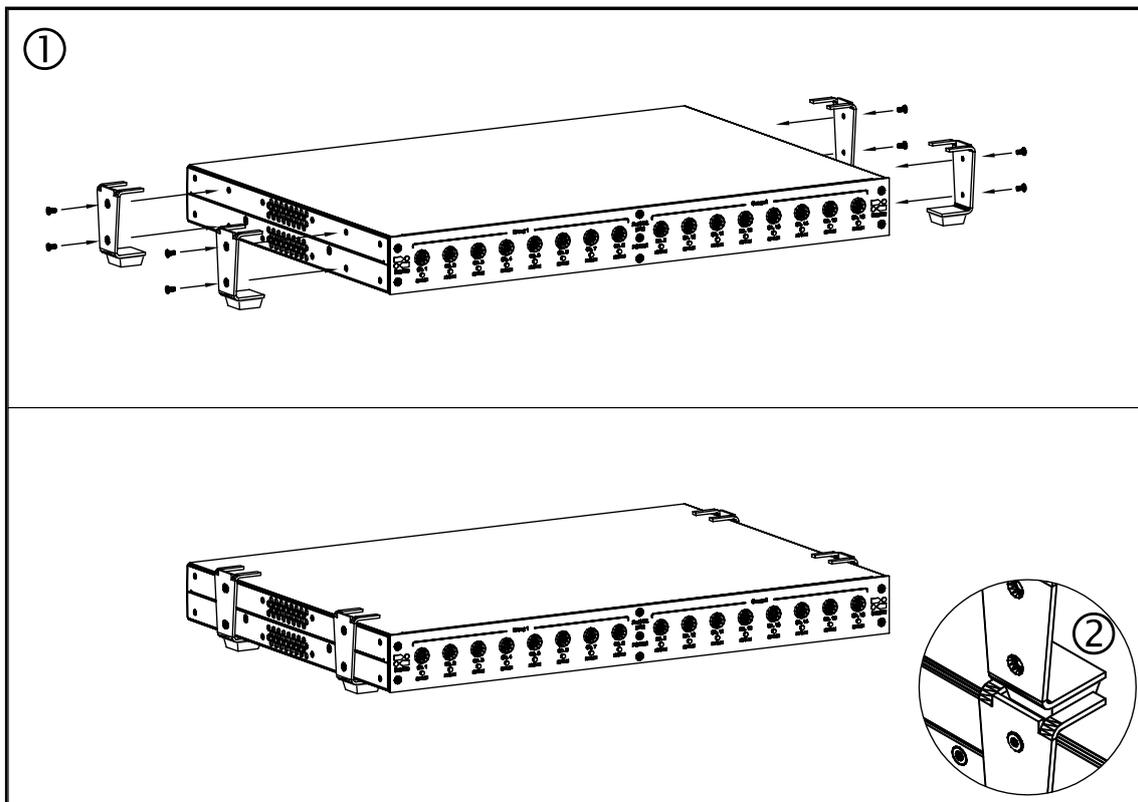


Figure 9 - Attaching the ProDAQ 5726 Stackable Desktop Feet Set

To securely stack several ProDAQ 5716, place the rubber feet of the upper ProDAQ 5716 into the brace provided by the feet below (see Figure 9, ②).

WARNING

Do not use screws longer than 6mm to attach the feet set to the enclosure. Using any other length might damage the unit and/or cause electrical shorts.

WARNING

Do not use other screws than metric M3 for attaching the feet set to the enclosure. Using any other screw style will damage the thread permanently.

2.3. ProDAQ 5176 Connectors and Switches

2.3.1. Front Panel ProDAQ 5716-Ax

The ProDAQ 5716 versions “AA” and “AB” feature an 8-pin LEMO series 1B socket for the connection of sensors.



Pin	Signal
1	EX+
2	SENSE-
3	SIG+
4	Shunt Cal- / TEDS return
5	Shunt Cal+ / TEDS
6	SIG-
7	EX-
8	SENSE+

Table 1 - ProDAQ 5716-Ax Sensor Connector Pin-out

2.3.2. Front Panel ProDAQ 5716-Bx

The ProDAQ 5716 versions “BA” and “BB” feature an 8-pin shielded RJ45 socket for the connection of sensors.



Pin	Signal
1	EX+
2	EX-
3	SIG-
4	Shunt Cal+ / TEDS
5	Shunt Cal- / TEDS return
6	SIG+
7	SENSE-
8	SENSE+

Table 2 - ProDAQ 5716-Bx Sensor Connector Pin-out

ICP/LIM LEDs

Each channel has a dual color LED which indicates certain conditions. If the particular channel is switched to ICP mode, the LED will lit green if the current is flowing and red if there is an error and no current flows. If the channel is configured for bridge measurement, a red LED indicates that the current limit circuit is active.

In addition the LED can be blinking green for channel identification purposes.

2.3.3. Rear Panel ProDAQ 5716-xx

The rear panel of the ProDAQ 5716 features the following switches and connectors:



Figure 10 - ProDAQ 5716 Rear Panel

Voltage Reference I/O	Dual 2mm socket to monitor the reference voltage or to provide a reference voltage from an external source.
ProDAQ 3424 #2 (Ch. 9-16)	SCSI-style connector to interface channels 9 to 16 of the unit to a ProDAQ 3424 function card.
ProDAQ 3416 (Ch. 1-16)	SCSI-style connector to interface the unit to a ProDAQ 3416 function card.
Trigger I/O 1-3	If the ProDAQ 5716 is used together with a ProDAQ 3416 ADC function card, the trigger I/O lines available on the ProDAQ 3416 connector are routed to these three SMB connectors.
ProDAQ 3424 #1 (Ch. 1-8)	SCSI-style connector to interface channels 1 to 8 of the unit to a ProDAQ 3424 function card.

2.4. Connecting the ProDAQ 5716 to an ADC Function Card

The ProDAQ 5716 is designed to interface to either one or two ProDAQ 3424 8-channel Sigma-Delta ADC function card(s) or one ProDAQ 3416 16-channel Sigma-Delta ADC function card. A standard ProDAQ 8010-Bx cable shall be used to connect between the ADC cards and the signal conditioning unit. Please note that it is not possible to connect both a ProDAQ 3416 and a ProDAQ 3424 to the ProDAQ 5716.

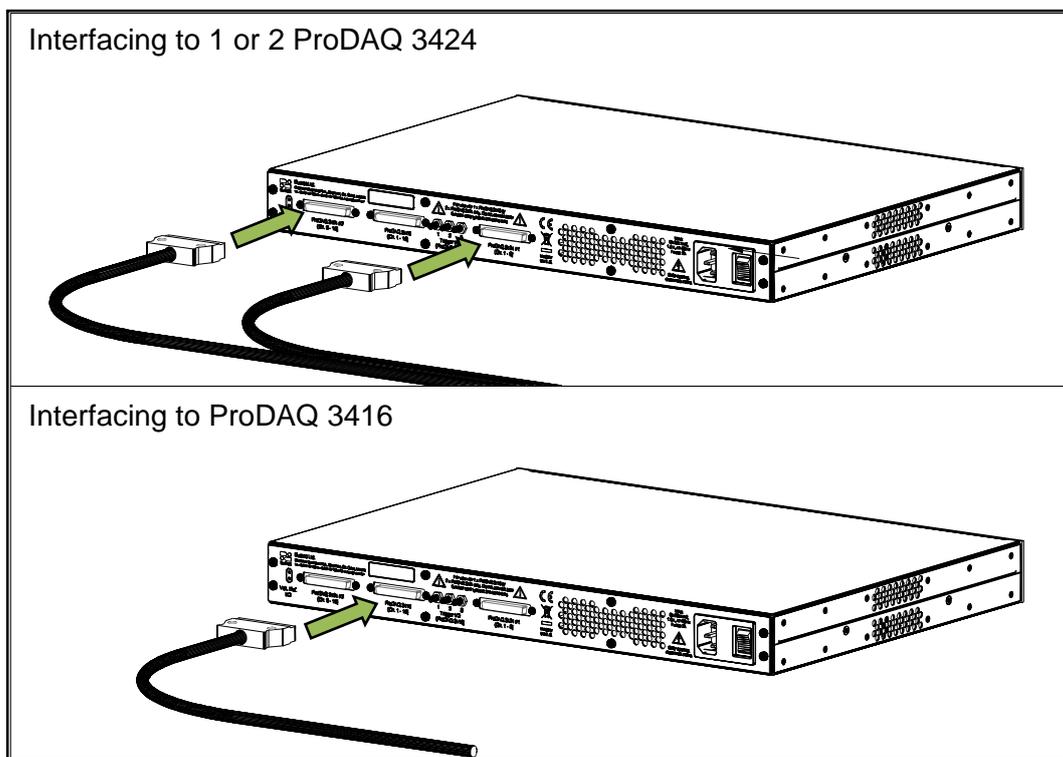


Figure 11 - Connecting to ADC Function Card(s)

If you want to use the ProDAQ 5716 together with one or two ProDAQ 3424 function cards, use the connectors labeled “ProDAQ 3424 #1 (Ch. 1 - 8)” and “ProDAQ 3424 #2 (Ch. 9 - 16)” on the rear. If you want to use a ProDAQ 3416 with the ProDAQ 5716, use the connector labeled “ProDAQ 3416 (Ch. 1 - 16)”.

Both ADC function cards feature an I²C port on their connector, which is used to control the ProDAQ 5716. If one or two ProDAQ 3424 function cards are used together with the ProDAQ 5716, the unit is virtually split into two units à 8 channels, with the respective 3424 controlling only its half. When used with a ProDAQ 3416, the complete unit is controlled via the one function card.

WARNING

It is not possible to use the ProDAQ 5716 with both the ProDAQ 3416 and the ProDAQ 3424 at the same time. Also take great care to connect the function cards to the correct outputs on the rear panel, as the pin-out of the connectors are different and an error could cause damage to both the ProDAQ 5716 and the ADC function card.

2.5. Connecting your Sensors

2.5.1. Full-Bridge

To connect a strain gage sensor employing all four bridge arms to the ProDAQ 5716, use the following configuration:

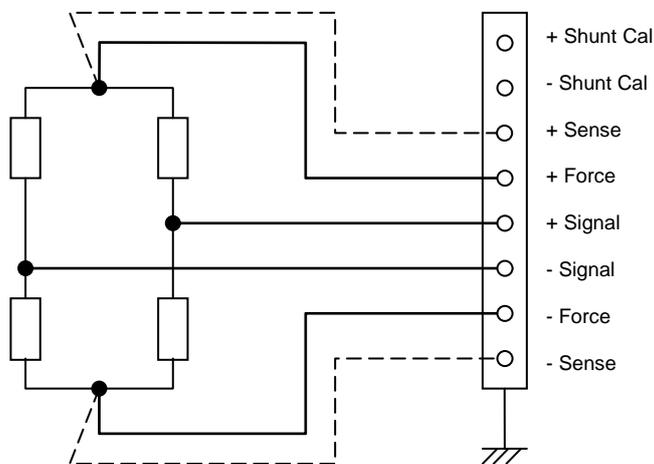


Figure 12 - Full-Bridge Sensor Configuration

For best performance, it is recommended to use additional remote sense wires (shown as “----”).

2.5.2. Half-Bridge

To connect a strain gage sensor employing only two arms of the bridge, use:

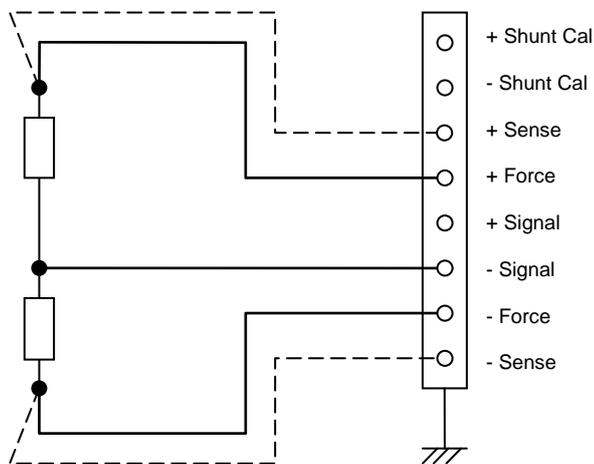


Figure 13 - Half-Bridge Sensor Configuration

The bridge completion will be done by internal resistors when configuring the particular channel for a half-bridge configuration. Again it is recommended to use a set of extra remote sense wires to achieve the highest accuracy (shown as “---”).

2.5.3. Quarter-Bridge

If your sensor consists only of one strain gage, use the following configuration:

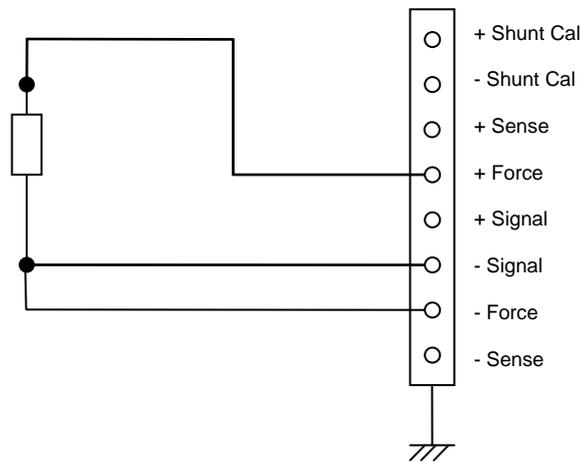


Figure 14 - Quarter-Bridge Sensor Configuration

The internal bridge completion needs to be configured depending on the resistance of your sensor.

2.5.4. Voltage/ICP

To measure voltages or use ICP/IEPE sensors, just connect to the +Signal/-Signal inputs of the ProDAQ 5716. All configuration for these modes are done via software.

3. Theory of Operation

The ProDAQ 5716 is a 16-channel, 1U, 19" rack-mount signal conditioning unit that is designed to provide excitation, completion and calibration for strain gage sensors or a constant current supply for ICP/IEPE sensors.

A block diagram of the ProDAQ 5716 is shown in Figure 15. The unit has 16 identical channels (only one is shown in the diagram). Each channel contains positive and negative voltage excitation with remote sensing, overvoltage protection, current limiting, quarter, half and full bridge completion, excitation monitoring, current limit indication, shunt calibration, bridge auto-balancing, voltage calibration, programmable gain, ICP, TEDS and AC/DC coupling.

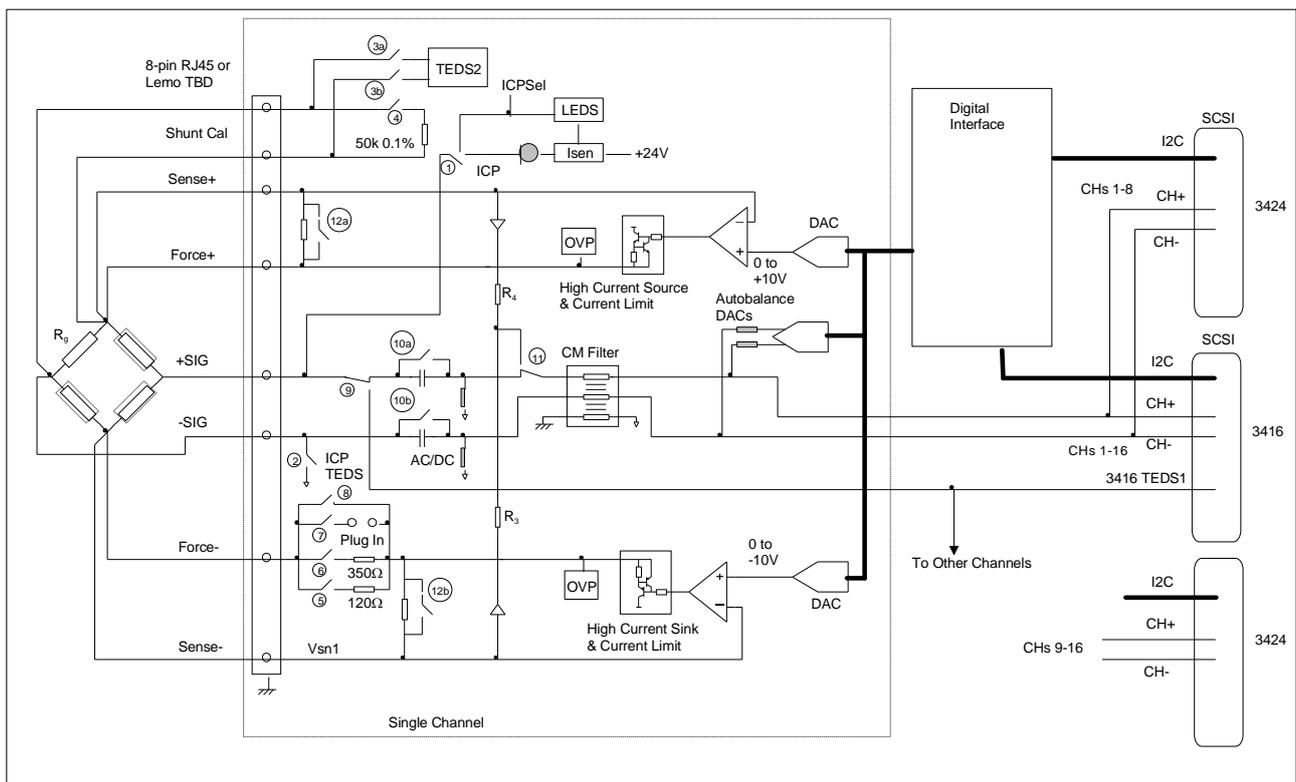


Figure 15 - ProDAQ 5716 Block Diagram

The bridge excitation type is constant voltage and is individually programmable per channel from 0 to $\pm 10V$ using on-board DACs. Each channel employs voltage sensing on both the positive and negative levels to minimize the error. If the voltage sensing is not used, errors caused by the cable resistance will influence the accuracy of the measurement. The excitation voltages for each channel are factory calibrated and the calibration coefficients stored on-board and automatically applied. The sense voltages can be switched to the ADC function card inputs for accurate on-line calibration.

Each channel of the ProDAQ 5716 can provide up to 50mA of output current that is factory set with a current limit of 50 mA. In the case of channel failure an overvoltage protection circuit prevents the excitation voltage from rising above too high. The current limit circuit is monitored in order to indicate that limit has been reached.

Each channel has an autobalance circuit which is designed to eliminate the offset voltage between the two arms of the bridge. A shunt calibration resistor is available on two of the pins of the output

connector. The purpose of shunt calibration is to imbalance the bridge by a known quantity in order to simulate a strain. This is done by switching in a fixed precision resistance in parallel with a gage. It is used for calibration of the bridge.

Each channel of the ProDAQ 5716-XX contains a programmable gain amplifier with gains of 1, 10, 100 and 1000. A voltage reference located on the function card carrier can be used to calibrate the complete signal path. Alternatively a connector on the rear panel can be used to insert a custom calibration voltage.

The ProDAQ 5716 can also provide a constant current supply for ICP/IEPE sensors. This allows for the excitation of sensors that conform to this standard using a constant current biasing technique. LEDs indicate the status of the ICP supply current.

Each channel has the ability to communicate to external sensors via TEDS. It can accommodate both TEDS Class 1 and TEDS class 2. TEDS class 1 is used when interfacing to the positive input pin on a sensor while TEDS class 2 is used where the communication pins of the sensor are separate to the sensor pins.

3.1. Full Bridge Mode

Figure 16 shows the typical configuration for a full bridge. The excitation sense connections (Sense+ and Sense-) are optional but recommended for highest accuracy. If they are not used then excitation sensing is performed within the ProDAQ 5716. Without remote sensing the error is dependent upon the cable length. The shunt calibration connections are also optional.

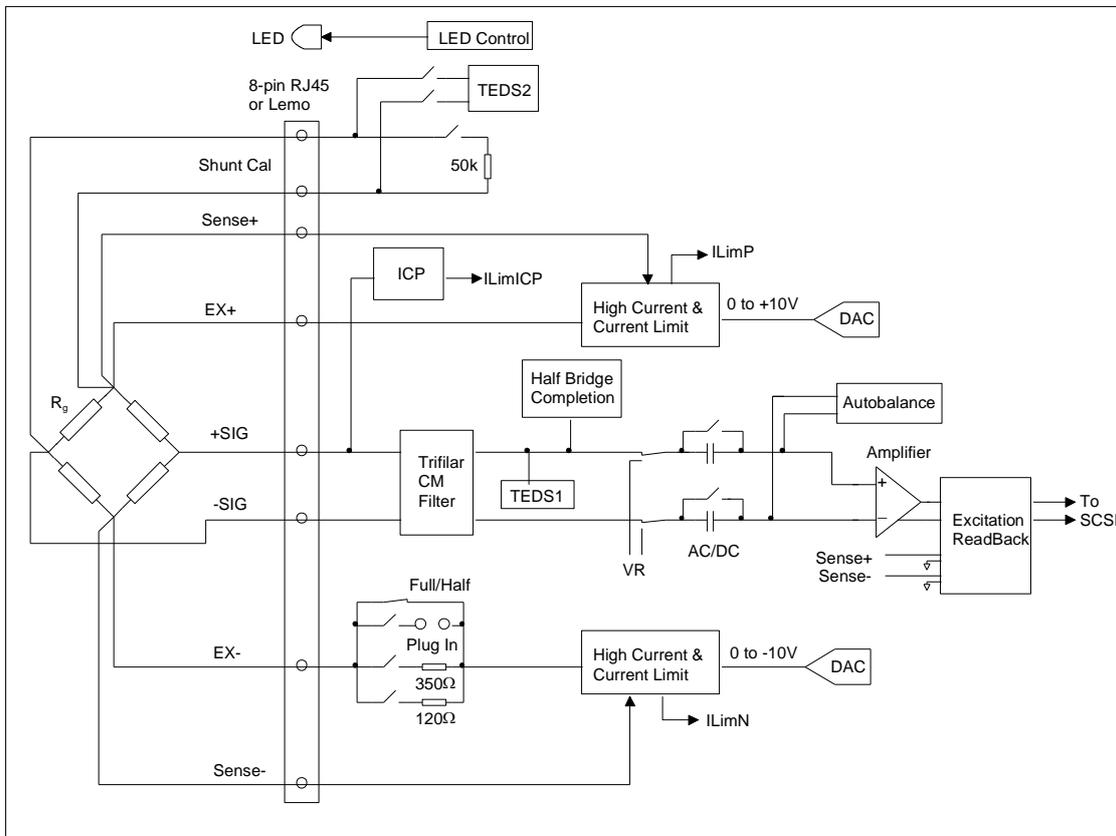


Figure 16 – Full Bridge Configuration

In full bridge mode the operator programs a positive excitation voltage (from 0 to +10V) and a negative excitation voltage (from 0V to -10V). The voltages are completely independent but for operation around 0V they should be set to the same level. For example, if an excitation voltage of +12V is required across the strain gauge then setting $\pm 6V$ will mean that the centre of the bridge is at a CM voltage of 0V. If the arms of the bridge are producing a small offset voltage due to mismatch this may be then corrected for using the Autobalance feature.

The excitation current on each channel is monitored in real time. A current limit circuit limits the current to 50mA and the front panel LED goes red to signal this limit.

The 50k shunt calibration resistor may be switched across one element of the bridge in order to imbalance the bridge and simulate a load. Also available on these pins is TEDS class 2.

The differential signal from the bridge connects to +SIG and -SIG of the 8-pin connector. A trifilar CM filter helps to reduce HF CM noise. The user chooses the type of coupling (AC or DC) although DC coupling is recommended. A suitable system gain should be then set. The system gain consists of a combination of the ProDAQ 5716's gain (1, 10, 100 or 1000) and ADC gain (for the 3416 this can be 1, 2, 5, 10, 20, 50, 100, 200, 500, 1000 or 2000). Note that the maximum specified system gain is 2000 and, although it is possible to program higher gains, performance is not guaranteed.

3.2. Half Bridge Mode

Figure 17 shows the typical configuration for a half bridge. Operation is very similar to the full-bridge mode. However, one arm of the bridge (+SIG) is connected internally and provides a reference to the sensor.

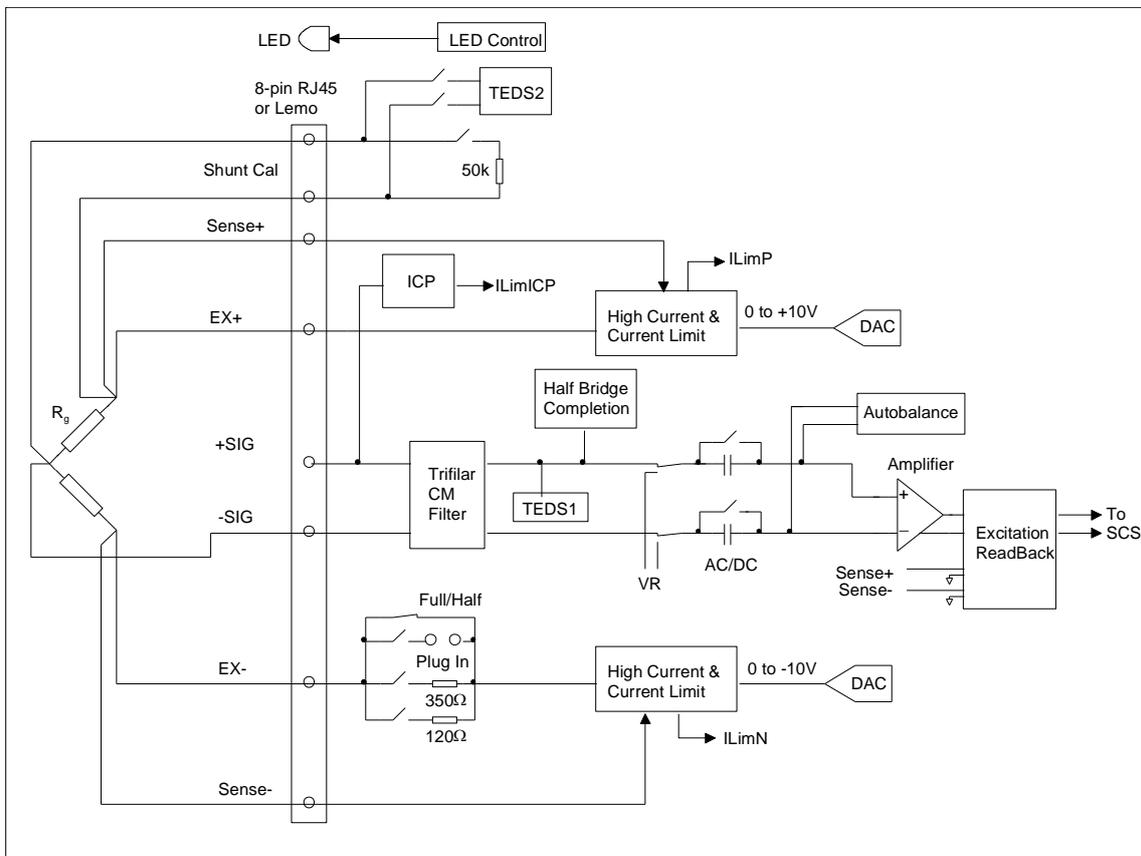


Figure 17 – Half Bridge Configuration

3.3. Quarter Bridge Mode

Figure 18 shows the typical configuration for a 350Ω quarter bridge. In this mode one arm of the bridge is internal (as in half bridge mode) and one element of the other arm is also internal. The internal element is 120Ω, 350Ω or User Selectable. In quarter bridge mode remote sensing is not available.

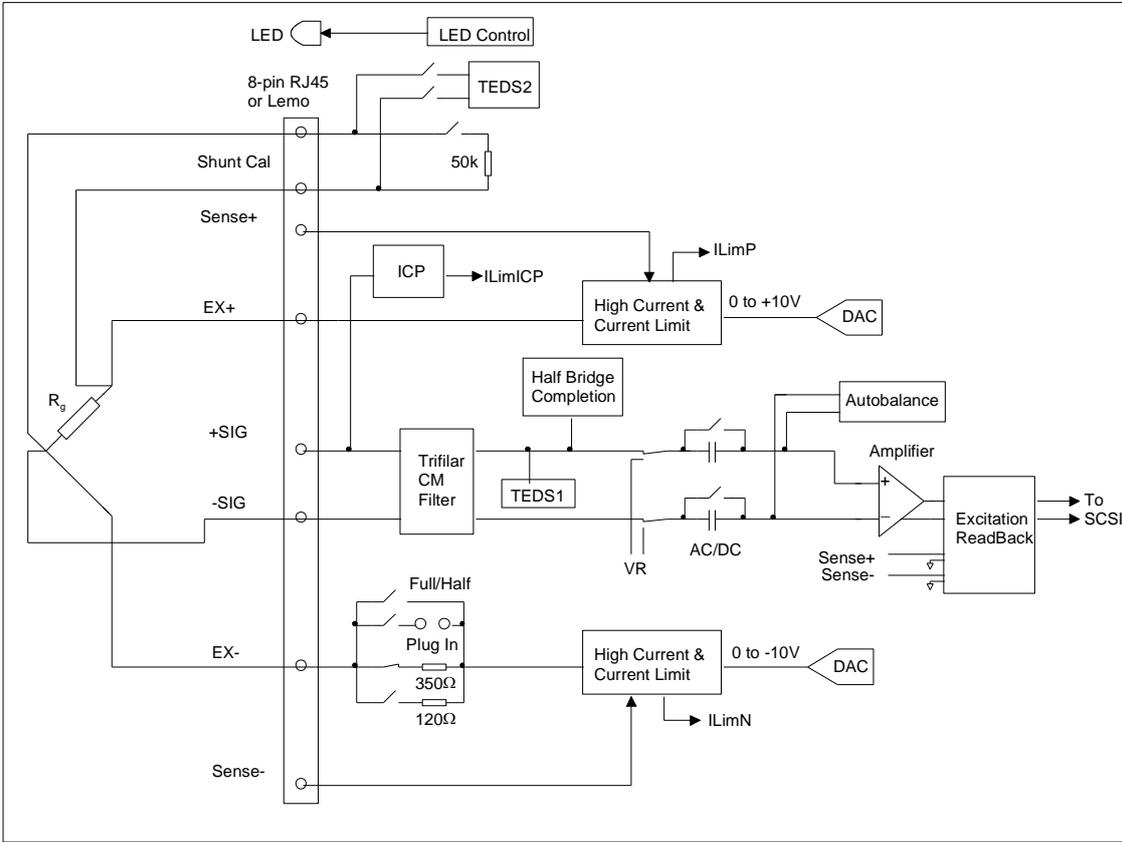


Figure 18 – Quarter Bridge Configuration (350Ω shown)

To fit a different resistor value to a channel the top of the 5716 must first be removed. Available on each channel is a 4-pin header on a 0.1" pitch, shown in Figure 19. The resistor, R, should be plugged in between pin 4 (pin 1 is the square pin) and either pin 3 (for 0.1" pitch), pin 2 (for 0.2" pitch) or pin 1 (for 0.3" pitch). In software the plug-in resistor option should be selected.

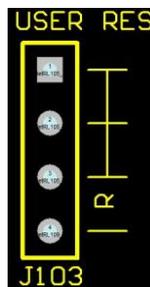


Figure 19 – Fitting a User Resistor

3.4. IEPE/ICP™ Mode

Figure 20 shows the configuration for IEPE/ICP™ mode. IEPE/ICP™ sensors have an integrated amplifier and are powered using a current source. The 5716 has, as standard, a 4.7mA current source. During normal operation the current source is connected to the sensor via +SIG. The return from the ICP™ sensor is connected to GND via –SIG. An internal relay does this automatically when ICP™ mode is selected.

Because the current source creates a relatively high DC voltage on +SIG it is necessary to AC couple the signal to the amplifier. When IEPE/ICP™ mode is chosen AC coupling is automatically set on +SIG with DC coupling on –SIG. During normal ICP™ operation the front panel LED is green. If a fault occurs and no current flows then the LED will go red and a software interrupt generated.

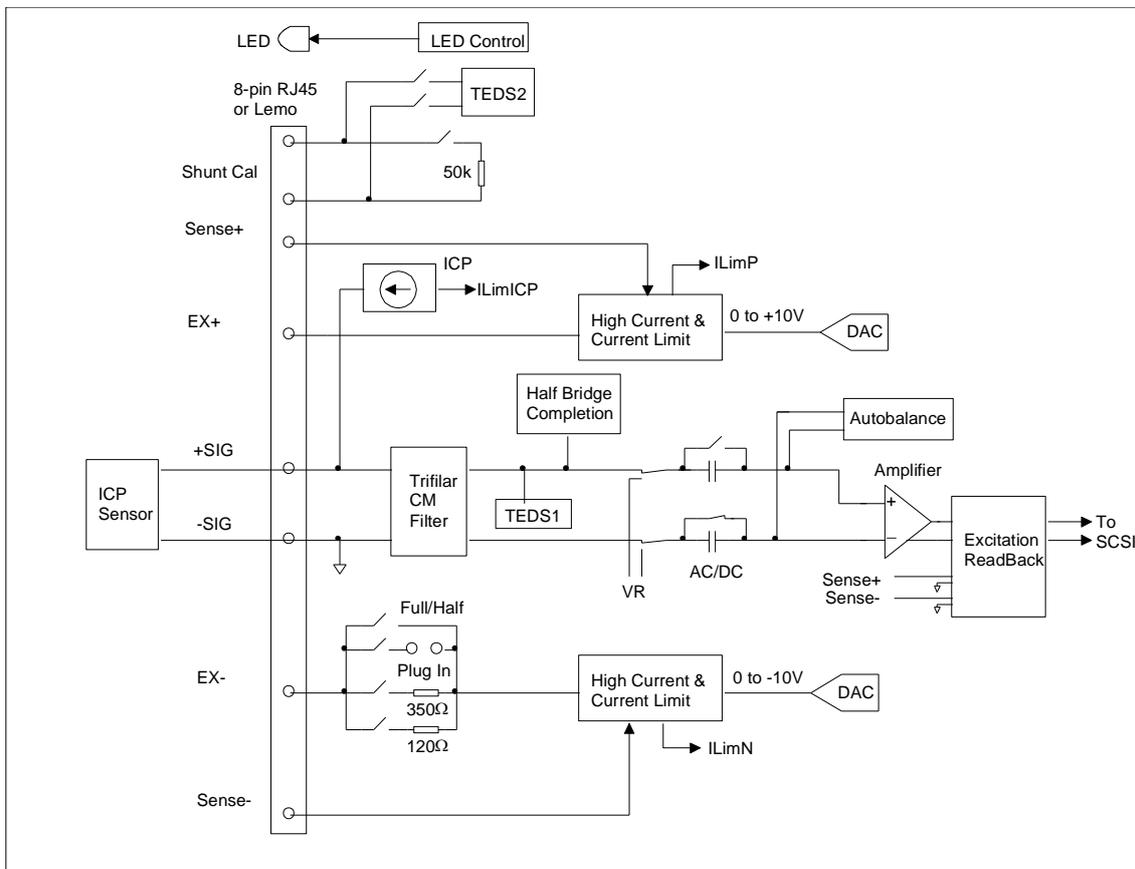


Figure 20 – ICP™ Mode Configuration

3.5. Differential Voltage Mode

Figure 21 shows the configuration for a differential voltage signal source. This type of source is often called a balanced signal. It has two outputs which are out of 180° out of phase and a common ground. The ground should be connected to the shield of the connector and the inputs connected to +SIG and –SIG, as shown in the diagram.

If AC coupling is selected it is possible to operate with the sensor at a CM voltage that is outside the $\pm 10V$ operating range of the 5716. The CM voltage in this case is $\pm 60V$. Do not select TEDS class 1 when a CM voltage is present on the sensor.

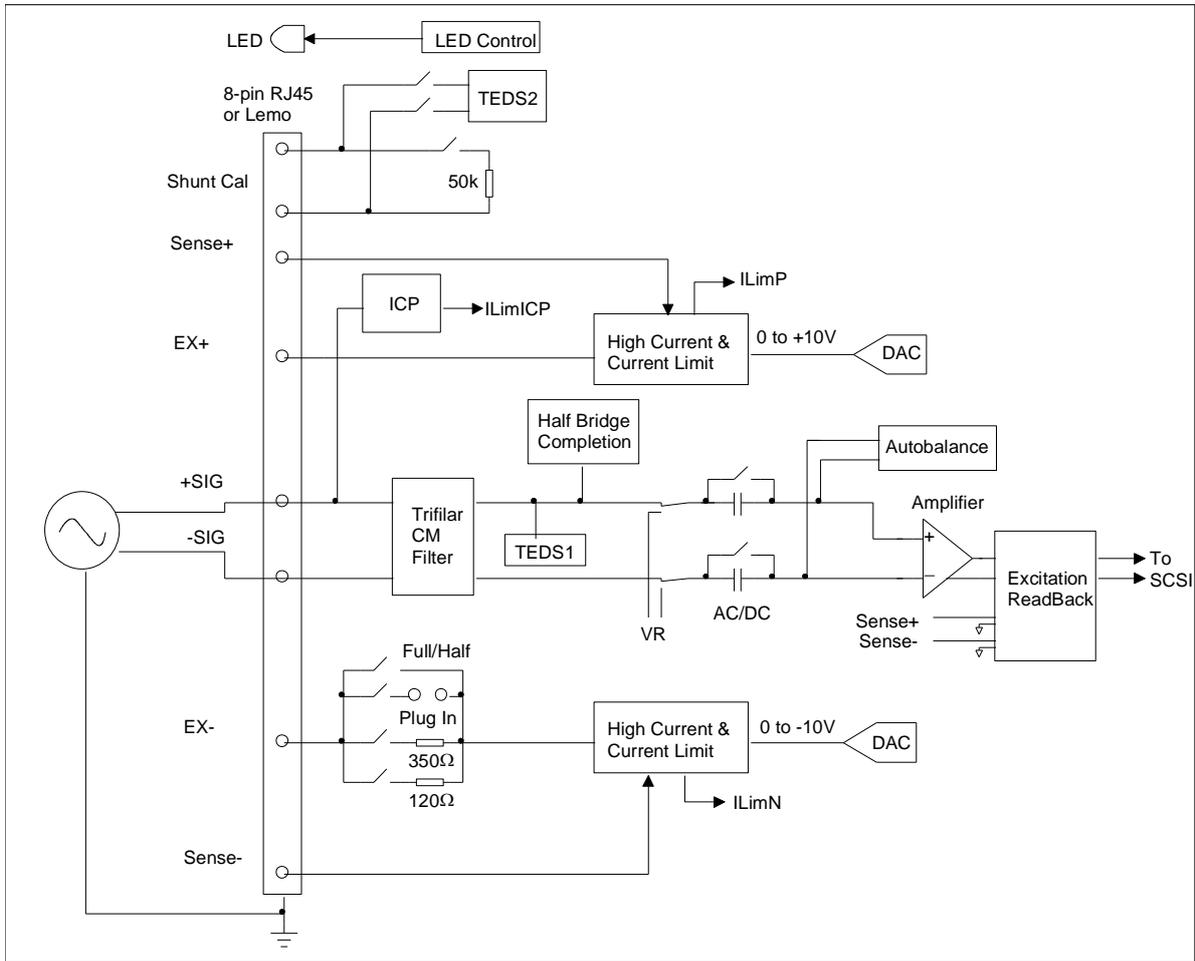


Figure 21 – Differential Voltage Mode Configuration

3.6. Single-Ended Voltage Mode

Figure 22 shows the configuration for a single ended voltage signal source. The signal should be connected to +SIG while the ground should be connected to -SIG. Optionally the ground may also be connected to the shield of the connector as shown in the diagram.

If AC coupling is selected it is possible to operate with the sensor at a CM voltage that is outside the $\pm 10V$ operating range of the 5716. The CM voltage in this case is $\pm 60V$. Do not select TEDS class 1 when a CM voltage is present on the sensor. Note that AC coupling only applies to +SIG and that the -SIG capacitor is shorted out.

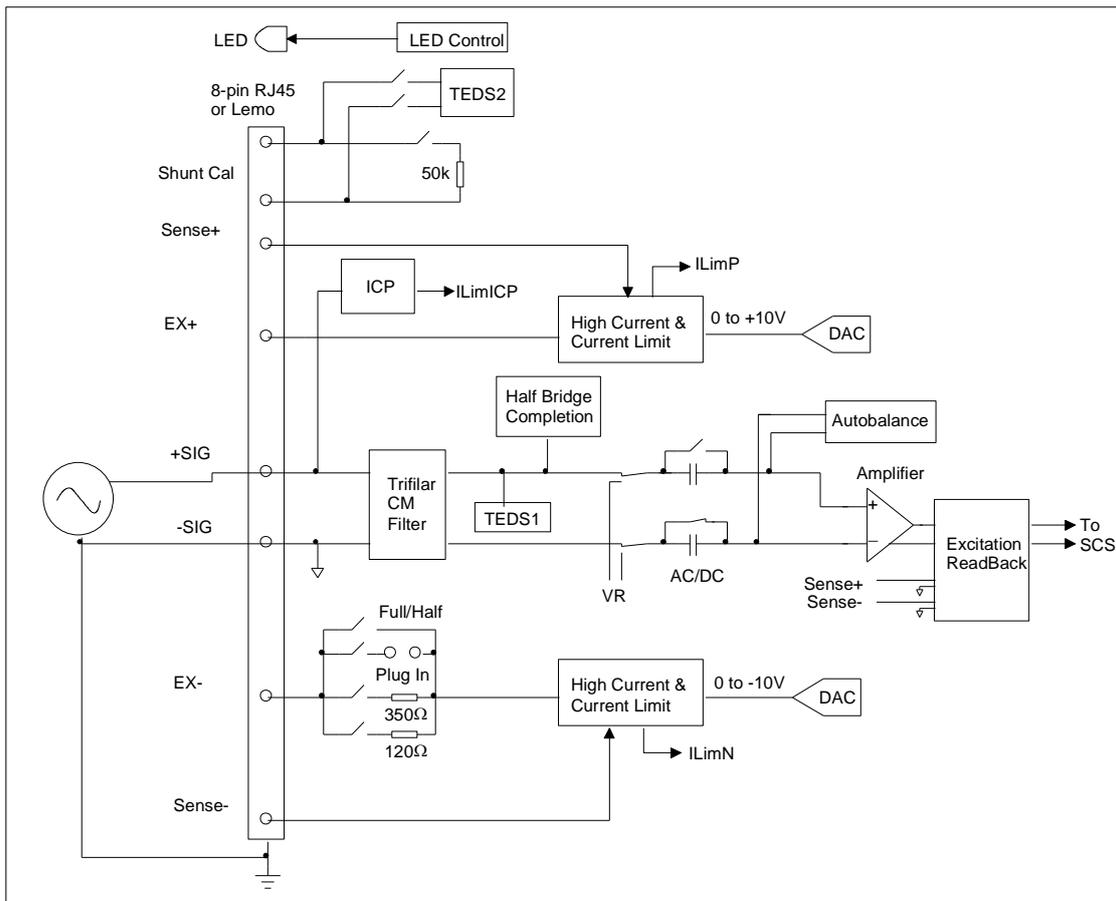


Figure 22 – Single Ended Voltage Mode Configuration

This page was intentionally left blank.

4. The ProDAQ 5716 Soft Front Panel Plug-in

The ProDAQ 5716 does not have a stand-alone soft front panel. It comes with a plug-in dynamic link library for the ProDAQ 3416 or ProDAQ 3424 soft front panel, where it is shown as an additional tab.

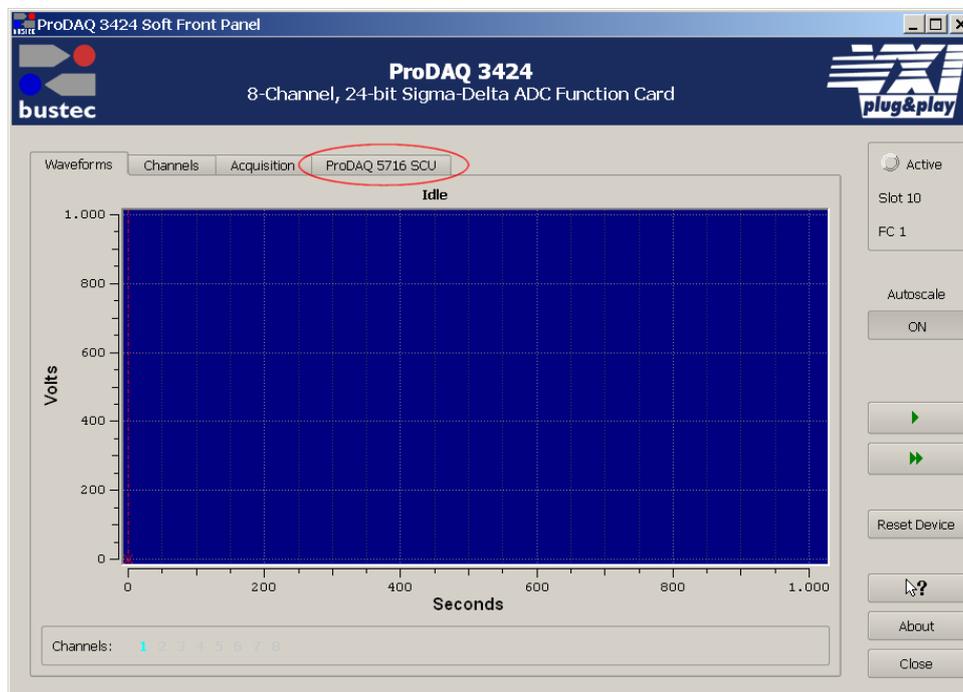


Figure 23 - ProDAQ 3424 Soft Front Panel with ProDAQ 5716 Tab

The plug-in is automatically loaded when the ProDAQ 3416 or 3424 Soft Front Panels detect a connected (and powered-on) ProDAQ 5716 on startup.

If you select the ProDAQ 5716 tab, it allows you to configure all channels controlled by the function card the soft front panel was started for. In case of the ProDAQ 3424 card and soft front panel, only channel number 1 to 8 are shown, as the function card only controls and uses one half of a ProDAQ 5716. Whether these eight channels correspond to the channels 1-8 or 9-16 on the ProDAQ 5716, depend on the connection you made between the ProDAQ 3424 and the ProDAQ 5716. If the ProDAQ 5716 is controlled by a ProDAQ 3416, tabs for all 16 channels are shown. To check the connection and identify the channel the particular settings refer to, you can use the "Identification On/Off" button on the lower part of the channel tabs. Switching the identification on will cause the per-channel LED on the ProDAQ 5716 front panel to blink.

The panel for each channel is divided into three groups: Configuration, Calibration and Status. The controls in the configuration group allow you to configure the channel for the type of measurement desired. If you change any setting, you will need to press the "Apply" button beneath the configuration choices. If you press "Apply to all", the same settings will be applied to all channels.

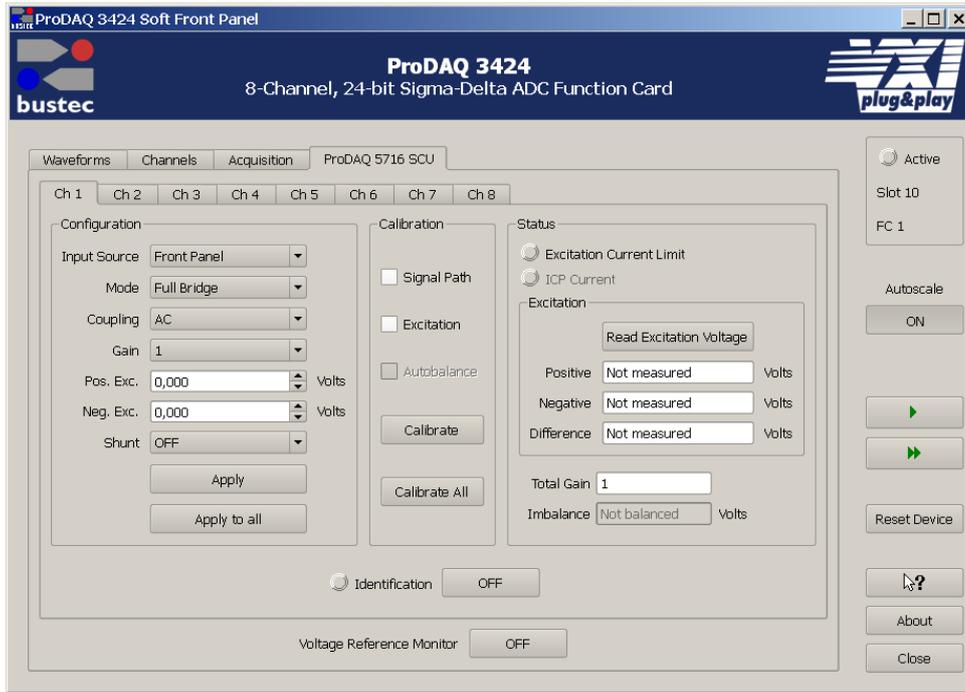


Figure 24 - ProDAQ 5716 Channel Configuration Tab

The controls in the calibration group allow the calibration of both the signal path and the excitation for the particular channel or all channels. The status group allows the read-out of the status of the channel.

4.1.1. Channel Configuration

The configuration group contains the following controls:

- Input Source** This combo box allows you to choose whether the channel input shall be connected to the front panel connector or the internal voltage reference bus.
- Mode** Selects the mode for the channel. It allows to configure the channel for full bridge, half bridge or the three different quarter bridge configurations (120 Ω, 350 Ω, User supplied resistor) as well as for ICP sensors or plain voltage measurements (differential or single-ended).
- Coupling** Allows to select AC or DC coupling.
- Gain** Configures the programmable gain amplifier for the channel. Gain factors of 1, 10, 100 and 1000 are possible.
- Pos.Exc. / Neg.Exc.** Sets the positive and negative excitation supply for bridge measurements.
- Shunt** Switches the internal shunt resistor on.

Some of the settings are mutual exclusive; e.g. when choosing ICP the bridge supplies are disabled and it is not possible to control them. To apply the settings to the channel, press the "Apply" button. To apply the settings to all channels, press "Apply to all".

4.1.2. Calibration

The calibration group of the channel's tab allow you to calibrate a channel. Depending on the type of measurement desired, different parts of the calibration process can be enabled or disabled. The three parts are:

Signal Path	Performs a calibration of the signal path from the ProDAQ 5716s SIG+/SIG- input pins to the ProDAQ 3416/3424 ADC. If a voltage reference is available on the function card carrier the ProDAQ 3416/3424 is installed in, it performs offset and gain calibration. If not, it only calibrates the offset.
Excitation	Uses the function cards ADC to measure the bridge excitation voltages and calibrate to the choose value. The accuracy of this calibration depends on the signal path calibration, so that this always should be done first. If both are enabled, the SFP will automatically perform the signal path calibration first.
Autobalance	Performs an autobalance on the bridge. To do so, the ProDAQ 5716 can insert a current into one arm of the bridge to add a voltage and balance the bridge. Again the accuracy of this process depends on the first two calibration steps.

To perform the chosen calibration, press the "Calibrate" button. To perform the chosen calibration on all channels, use "Calibrate All".

4.1.3. Status

The controls in the last group provide status information about the channel to the user.

Excitation Current Limit	If this control is on, it indicates that the channels current limiting circuit is active because there is a short or the resistance of the sensor is too low.
ICP Current	In ICP mode this control indicates with a green light that the current is flowing, or it chances to red if there is a fault and the current is not flowing.
Excitation	The controls in this group allow you to measure the actual excitation voltage. If the button "Read Excitation Voltage" is pressed, the input of the ADC function card is switch from the SIG+/SIG- inputs to the internal excitation voltage and used to measure it.
Total Gain	Displays the total gain enabled on both the ProDAQ 5716 and controlling ADC function card.
Imbalance	Shows the remaining imbalance after an autobalancing performed.

This page was intentionally left blank.

5. Programming the ProDAQ 5716

This chapter shows how to program the ProDAQ 5716 signal conditioning unit using the *VXIplug&play* driver. Complete examples can be found in the “Examples” subdirectory of the driver. All functions are explained in detail in the help file coming with the driver.

A ProDAQ 5716 unit is either controlled by one ProDAQ 3416 ADC function card or two ProDAQ 3424 ADC function cards. This is reflected in the software architecture - the ProDAQ 5716 driver utilizes either the ProDAQ 3416 or the ProDAQ 3424 driver to allow programming of the unit.

5.1. *VXIplug&play* Driver Organization

The *VXIplug&play* driver is organized in a hierarchical manner to allow the user to quickly choose the function calls to solve the task at hand without being confronted with unnecessary details. Besides the standard connection/disconnection and utility functions it contains different levels of functionality which provide single functions or sets of functions to solve a particular task:

Class/Panel Name:	Function Name:
Initialization	bu5716_init
Hardware Configuration	
Set Channel Configuration	bu5716_setChannelConf
Get Channel Configuration	bu5716_getChannelConf
Set Excitation Voltage	bu5716_setExcitation
Get Excitation Voltage	bu5716_getExcitation
Set SCU Error Mask	bu5716_setScuErrorMask
Calibration Functions	
Autobalance	bu5716_autobalance
Excitation Calibration	bu5716_excitCalibration
PGA Calibration	bu5716_pgaCalibration
Low-Level Access	
...	
Status Functions	
Get Channel Status	bu5716_getStatus
Identify Channel	bu5716_channelIdentify
Utility Functions	
Read Temperature	bu5716_readTemperature
Reset	bu5716_reset
Error Message	bu5716_error_message
Device Serial Number	bu5716_serialNumber
Revision Query	bu5716_revision_query
Get Function Card Last Error	bu5716_getFCLastError
Close	bu5716_close

Figure 25 - *VXIplug&play* Driver Organization

The section **Hardware Configuration** contains high-level functions to configure the unit. The section **Calibration Functions** contains high-level functions that can be used to calibrate the ProDAQ 3416/3424 and ProDAQ 5716. The section **Utility Functions** contains utility functions that can be used together with the high-level functions.

The section **Low-level Access** contains functions that directly change settings on a register level and are used by the higher level functions to implement their functionality. Using them directly in combination with the higher level functions might interfere with the functionality implemented and

should be avoided. In general the usage of the low-level functions will require an intimate knowledge of the ProDAQ 5716 hardware as well as the hardware of the ProDAQ 3416/3424 and the respective function card carrier. Before you attempt to implement your data acquisition or test application using them, it is recommended to study their usage in the higher level functions in the driver sources and/or contact Bustec for support.

A small exception are here the functions `bu5716_getStatus()` and `bu5716_channelIdentify()` which do not interfere and can be used everywhere.

The following paragraphs will explain the usage of the high level functions.

5.2. Connecting to the Function Card and Signal Conditioning Unit

The ProDAQ 5716 driver utilizes the ProDAQ 3416 or ProDAQ 3424 driver to access the signal conditioning card. Therefore first the ProDAQ 3416/3424 driver need to be initialized and connected to the correct function card (the one controlling the ProDAQ 5716) before the ProDAQ 5716 driver can be initialized. The following example shows the usage in case of using the ProDAQ 3416 ADC function card.

To initialize the ProDAQ 3416 driver and connect to the ProDAQ motherboard or function card carrier, the standard *VXIplug&play* initialization function `bu3416_init()` is used (see Figure 26, ①). (Please refer to the *VXIplug&play* standard VPP-4.3, section 4.3 for a detailed description of the address string.) After initializing the driver and connecting to the motherboard or carrier, the driver must be told which one of the function cards to work with. This is done by the function `bu3416_fcSelect()`. It takes as an argument the session established via the function `bu3416_init()`, the function card number and a boolean value specifying whether to reset the selected function card (see Figure 26, ②).

For your convenience, the driver contains a new function called `bu3416_paramInit()`, which combines the functionality of the `bu3416_init()` and `bu3416_fcSelect()` functions by extending the argument list of the standard initialization function with a parameter specifying the function card number (see Figure 26, ③).

For the driver functions to work properly, you will either have to use the function `bu3416_paramInit()` to open a session with the device, or you will have to call the function `bu3416_fcSelect()` after calling the function `bu3416_init()` and before any other driver function is called.

Now you can connect to the ProDAQ 5716 as well by using the function `bu5716_init()` with the session handle to the ProDAQ 3416 returned by the functions `bu3416_init()` or `bu3416_paramInit()`. As with the ProDAQ 3416 init function you have the choice whether to check the ID of the signal conditioning card to connect to as well as to reset the card (see Figure 26, ④). The function returns a new session handle, which must be used with all ProDAQ 5716 driver functions.

```

#include <visa.h>
#include <bu3416.h>
#include <bu5716.h>

main (int argc, char **argv)
{
    ViStatus status;
    ViSession session_3416;
    ViSession session_5716;
    ViChar descr[256];

#ifdef USE_PARAMINIT
① /* connect to a ProDAQ motherboard in a VXIbus system */
    if ((status = bu3416_init("VXI0::2::INSTR", VI_TRUE, VI_TRUE, &session_3416)) != VI_SUCCESS)
    {
        viStatusDesc (session_3416, status, descr);
        printf ("Error: bu3416_init() failed due to %s\n", descr);

        return -1;
    }
② /* use function card in position/slot 1 */
    if ((status = bu3416_fcSelect(session_3416, 1, VI_TRUE)) != VI_SUCCESS)
    {
        viStatusDesc (session_3416, status, descr);
        printf ("Error: bu3416_fcSelect failed due to %s\n", descr);

        return -1;
    }
#else
③ /* OR: connect to a 3416 in position 1 in a LXI function card carrier */
    if ((status = bu3416_paramInit("TCPIP::192.168.168.63::INSTR",
        1, VI_TRUE, VI_TRUE, &session_3416)) != VI_SUCCESS)
    {
        viStatusDesc (rm_session, status, descr);
        printf ("Error: bu3416_paramInit() failed due to %s\n", descr);

        return -1;
    }
#endif
④ /* connect to the 5716 controled by the 3416 */
    if ((status = bu5716_init(session_3416, VI_TRUE, VI_TRUE, &session_5821)) != VI_SUCCESS)
    {
        viStatusDesc (session_5716, status, descr);
        printf ("Error: bu5716_init() failed due to %s\n", descr);

        return -1;
    }
    /* ... */
}

```

Figure 26 - Connecting to The ProDAQ 3416 and ProDAQ 5716

To close the driver sessions with the ProDAQ 5716 and the ProDAQ 3416, the standard VXIplug&play functions `bu5716_close()` and `bu3416_close()` must be used, preferably in this sequence.

NOTE

Please note that only code snippets are shown here in the manual. For the complete example, refer to the 'Examples' folder in the drivers installation directory.

6. VXIplug&play Driver Functions

Introduction

This instrument driver provides programming support for the ProDAQ 5716 Bridge Signal Conditioning Unit. It contains functions for opening, configuring, calibrating and closing the instrument.

Assumptions

To successfully use this function card, it must be installed onto a ProDAQ VXIbus motherboard or a ProDAQ LXI function card carrier. The ProDAQ motherboard must in turn be installed in a VXIbus system which is connected via a suitable slot-0 controller to your computer. The LXI function card carrier must be connected via network to your computer. A suitable VISA library must be installed on your computer.

Error and Status Information

Each function in this instrument driver returns a status code that either indicates success or describes an error or warning condition. Your program should examine the status code from each call to an instrument driver function to determine if an error occurred.

The general meaning of the status code is as follows:

Value	Meaning
0	Success
Positive Values	Warnings
Negative Values	Errors

The description of each instrument driver function lists possible error codes and their meanings.

Function Tree Layout

Class/Panel Name:	Function Name:
Initialization	bu5716_init
Hardware Configuration	
Set Channel Configuration	bu5716_setChannelConf
Get Channel Configuration	bu5716_getChannelConf
Set Excitation Voltage	bu5716_setExcitation
Get Excitation Voltage	bu5716_getExcitation
Set SCU Error Mask	bu5716_setScuErrorMask
Calibration Functions	
Autobalance	bu5716_autobalance
Excitation Calibration	bu5716_excitCalibration
PGA Calibration	bu5716_pgaCalibration
Low-Level Access	
Set Input Source	bu5716_setInputSrc
Get Input Source	bu5716_getInputSrc
Set Mode	bu5716_setMode
Get Mode	bu5716_getMode
Set Gain	bu5716_setGain
Get Gain	bu5716_getGain
Set Coupling	bu5716_setCoupling
Get Coupling	bu5716_getCoupling
Set DAC	bu5716_setDAC
Get DAC	bu5716_getDAC
Set Shunt Cal	bu5716_setShuntCal
Get Shunt Cal	bu5716_getShuntCal
Set Excitation Monitor	bu5716_setExcitationMonitor
Get Excitation Monitor	bu5716_getExcitationMonitor
Voltage Reference Access	
Get Voltage Reference Info	bu5716_getVoltRefInfo
Set Voltage Reference Output	bu5716_setVoltRefOutput
Get Voltage Reference Output	bu5716_getVoltRefOutput
TEDS Interface	
Read TEDS Memory	bu5716_readTEDS
Write TEDS EEPROM	bu5716_writeTEDS
Low-Level Calibration	
Reset Calibration Coeff	bu5716_resetCalibCoeff
Get PGA Calibration Coeff	bu5716_getPgaCalibCoeff
Get Excit. Calibration Coeff	bu5716_getExcCalibCoeff
Store Calibration Coeff	bu5716_storeCalibCoeff
Status Functions	
Get Channel Status	bu5716_getStatus
Identify Channel	bu5716_channelIdentify
Utility Functions	
Read Temperature	bu5716_readTemperature
Reset	bu5716_reset
Error Message	bu5716_error_message
Device Serial Number	bu5716_serialNumber
Revision Query	bu5716_revision_query
Get Function Card Last Error	bu5716_getFCLastError
Close	bu5716_close

6.1. VXiplug&play Driver Function Details

The following functions are in alphabetical order.

6.1.1. bu5716_autobalance

```
ViStatus bu5716_autobalance (ViSession instrumentHandle,  
                             ViInt16 channel, ViPReal64 disbalance);
```

Purpose

This function performs bridge autobalance for the specified channel.

Parameter List

instrumentHandle

Variable Type ViSession

The Instrument Handle is used to identify the unique session or communication channel between the driver and the instrument.

If more than one instrument of the same model type is used, this handle will be used to differentiate between them.

channel

Variable Type ViInt16

This parameter specifies which channel will be autobalanced.

Possible values are:

1-16 (for ProDAQ 3416 master card) or
1-8 (for ProDAQ 3424 master card).

disbalance

Variable Type ViReal64 (passed by reference)

This parameter returns the final value of the bridge disbalance.

Return Value

If the function was successful, it will return a status of VI_SUCCESS, otherwise it will return a warning or error code. Passing the status code to the function "bu5716_error_message" will return a string describing the warning or error.

A driver function can return three different types of warnings or errors. The function "bu5716_error_message" will handle all three types of warning/error codes by passing them to the appropriate function if necessary ("bu3100_error_message" or "viStatusDesc"), to return the correct warning/error message.

VISA Warnings/Errors:

See section 3.3 of the VPP 4.3.2 document for a complete list of VISA status codes and their values. The VPP 4.3 document contains detailed descriptions of all VISA functions and the status codes returned by each of them.

BU3100 Warnings/Errors:

These are warning or error codes returned by the common motherboard interface library, which are used by the 5716 driver to access a ProDAQ motherboard. Warnings returned by the library will be in the range 0x3FFC0800 to 0x3FFC0900 and errors in the range 0xBFFC0800 to 0xBFFC0900. They are defined in the include file bu3100.h.

BU5716 Warnings/Errors:

Warning codes returned by the 5716 driver functions will be in the range 0x3FFC0B00 to 0x3FFC0FFF and error codes in the range 0xBFFC0B00 to 0xBFFC0FFF. They are defined in the include file bu5716.h.

6.1.2. bu5716_channelIdentify

```
ViStatus bu5716_channelIdentify (ViSession instrumentHandle,
                                ViInt16 channel, ViInt16 state);
```

Purpose

This function turns on/off the identification mechanism for the selected channel.

Parameter List

instrumentHandle

Variable Type ViSession

The Instrument Handle is used to identify the unique session or communication channel between the driver and the instrument.

If more than one instrument of the same model type is used, this handle will be used to differentiate between them.

channel

Variable Type ViInt16

This parameter specifies which channel will be identified.

Possible values are:

1-16 (for ProDAQ 3416 master card) or
1-8 (for ProDAQ 3424 master card).

state

Variable Type ViInt16

Channel identification state.

Possible values:

bu5716_OFF 0 - Turns off selected channel identification.
bu5716_ON 1 - Turns on selected channel identification.

Return Value

If the function was successful, it will return a status of VI_SUCCESS, otherwise it will return a warning or error code. Passing the status code to the function "bu5716_error_message" will return a string describing the warning or error.

A driver function can return three different types of warnings or errors. The function "bu5716_error_message" will handle all three types of warning/error codes by passing them to the appropriate function if necessary ("bu3100_error_message" or "viStatusDesc"), to return the correct warning/error message.

VISA Warnings/Errors:

See section 3.3 of the VPP 4.3.2 document for a complete list of VISA status codes and their values. The VPP 4.3 document contains detailed descriptions of all VISA functions and the status codes returned by each of them.

BU3100 Warnings/Errors:

These are warning or error codes returned by the common motherboard interface library, which are used by the 5716 driver to access a ProDAQ motherboard. Warnings returned by the library will be in the range 0x3FFC0800 to 0x3FFC0900 and errors in the range 0xBFFC0800 to 0xBFFC0900. They are defined in the include file bu3100.h.

BU5716 Warnings/Errors:

Warning codes returned by the 5716 driver functions will be in the range 0x3FFC0B00 to 0x3FFC0FFF and error codes in the range 0xBFFC0B00 to 0xBFFC0FFF. They are defined in the include file bu5716.h.

6.1.3. bu5716_close

```
ViStatus bu5716_close (ViSession instrumentHandle);
```

Purpose

This function closes the instrument and frees the resources allocated by the call to the initialization function `bu5716_init()`.

This function must be called once for every instrument handle returned by the initialize function, prior to terminating the application program.

Parameter List

instrumentHandle

Variable Type ViSession

The Instrument Handle is used to identify the unique session or communication channel between the driver and the instrument.

If more than one instrument of the same model type is used, this handle will be used to differentiate between them.

Return Value

If the function was successful, it will return a status of `VI_SUCCESS`, otherwise it will return a warning or error code. Passing the status code to the function `"bu5716_error_message"` will return a string describing the warning or error.

A driver function can return three different types of warnings or errors. The function `"bu5716_error_message"` will handle all three types of warning/error codes by passing them to the appropriate function if necessary (`"bu3100_error_message"` or `"viStatusDesc"`), to return the correct warning/error message.

VISA Warnings/Errors:

See section 3.3 of the VPP 4.3.2 document for a complete list of VISA status codes and their values. The VPP 4.3 document contains detailed descriptions of all VISA functions and the status codes returned by each of them.

BU3100 Warnings/Errors:

These are warning or error codes returned by the common motherboard interface library, which are used by the 5716 driver to access a ProDAQ motherboard. Warnings returned by the library will be in the range `0x3FFFC0800` to `0x3FFFC0900` and errors in the range `0xBFFC0800` to `0xBFFC0900`. They are defined in the include file `bu3100.h`.

BU5716 Warnings/Errors:

Warning codes returned by the 5716 driver functions will be in the range `0x3FFFC0B00` to `0x3FFFC0FFF` and error codes in the range `0xBFFC0B00` to `0xBFFC0FFF`. They are defined in the include file `bu5716.h`.

6.1.4. bu5716_error_message

```
ViStatus bu5716_error_message (ViSession instrumentHandle,
                              ViStatus errorReturnValue,
                              ViChar _VI_FAR errorMessage[]);
```

Purpose

Converts a numeric error code, returned by one of the functions of this driver into a descriptive error message string.

Parameter List

instrumentHandle

Variable Type ViSession

The Instrument Handle is used to identify the unique session or communication channel between the driver and the instrument.

If more than one instrument of the same model type is used, this handle will be used to differentiate between them.

errorReturnValue

Variable Type ViStatus

Accepts the error code, returned by one of the functions in this instrument driver. See bu5716.h for error codes.

errorMessage

Variable Type ViChar[]

Upon return from the function, this parameter holds a text error message which corresponds to the error code.

The VISA Warnings and VISA Errors are described in section 3.3 of the VPP 4.3.2 document and Appendix B of VPP 4.3.

Return Value

If the function was successful, it will return a status of VI_SUCCESS, otherwise it will return a warning or error code. Passing the status code to the function "bu5716_error_message" will return a string describing the warning or error.

A driver function can return three different types of warnings or errors. The function "bu5716_error_message" will handle all three types of warning/error codes by passing them to the appropriate function if necessary ("bu3100_error_message" or "viStatusDesc"), to return the correct warning/error message.

VISA Warnings/Errors:

See section 3.3 of the VPP 4.3.2 document for a complete list of VISA status codes and their values. The VPP 4.3 document contains detailed descriptions of all VISA functions and the status codes returned by each of them.

BU3100 Warnings/Errors:

These are warning or error codes returned by the common motherboard interface library, which are used by the 5716 driver to access a ProDAQ motherboard. Warnings returned by the library will be in the range 0x3FFC0800 to 0x3FFC0900 and errors in the range 0xBFFC0800 to 0xBFFC0900. They are defined in the include file bu3100.h.

BU5716 Warnings/Errors:

Warning codes returned by the 5716 driver functions will be in the range 0x3FFC0B00 to 0x3FFC0FFF and error codes in the range 0xBFFC0B00 to 0xBFFC0FFF. They are defined in the include file bu5716.h.

6.1.5. bu5716_excitCalibration

```
ViStatus bu5716_excitCalibration (ViSession instrumentHandle,
                                 ViInt16 channel,
                                 ViReal64 _VI_FAR posOffsets[],
                                 ViReal64 _VI_FAR negOffsets[],
                                 ViReal64 _VI_FAR posGains[],
                                 ViReal64 _VI_FAR negGains[]);
```

Purpose

This function performs the excitation calibration of the specified channel or all 16 channels.

Parameter List

instrumentHandle

Variable Type ViSession

The Instrument Handle is used to identify the unique session or communication channel between the driver and the instrument.

If more than one instrument of the same model type is used, this handle will be used to differentiate between them.

channel

Variable Type ViInt16

This parameter specifies which channel will be calibrated.

Possible values are:

1-16 (for ProDAQ 3416 master card) or
1-8 (for ProDAQ 3424 master card)

or

bu5716_CH_ALL 0 All available channels (16 for ProDAQ 3416
and 8 for ProDAQ 3424 master function card)
will be configured with the same parameters.

posOffsets

Variable Type ViReal64[]

This parameter returns the excitation calibration coefficients (positive offset) for all selected channels acquired during the excitation calibration process. The array should be allocated with the appropriate size, prior to the function call.

If only one channel is selected, the array should be allocated as a one-element array, however if all channels are selected, the array should be allocated according to the type of master card, i.e., ProDAQ 3424 requires an 8-element array, and the ProDAQ 3416 requires a 16-element array.

negOffsets

Variable Type ViReal64[]

This parameter returns the excitation calibration coefficients (negative offset) for all selected channels acquired during the excitation calibration process. The array should be allocated with the appropriate size, prior to the function call.

If only one channel is selected, the array should be allocated as a one-element array, however if all channels are selected, the array should be allocated according to the type of master card, i.e., ProDAQ 3424 requires an 8-element array, and the ProDAQ 3416 requires a 16-element array.

posGains

Variable Type ViReal64[]

This parameter returns the excitation calibration coefficients (positive gain) for all selected channels acquired during the excitation calibration process. The array should be allocated with the appropriate size, prior to the function call.

If only one channel is selected, the array should be allocated as a one-element array, however if all channels are selected, the array should be allocated according to the type of master card, i.e., ProDAQ 3424 requires an 8-element array, and the ProDAQ 3416 requires a 16-element array.

negGains

Variable Type ViReal64[]

This parameter returns the excitation calibration coefficients (negative gain) for all selected channels acquired during the excitation calibration process. The array should be allocated with the appropriate size, prior to the function call.

If only one channel is selected, the array should be allocated as a one-element array, however if all channels are selected, the array should be allocated according to the type of master card, i.e., ProDAQ 3424 requires an 8-element array, and the ProDAQ 3416 requires a 16-element array.

Return Value

If the function was successful, it will return a status of VI_SUCCESS, otherwise it will return a warning or error code. Passing the status code to the function "bu5716_error_message" will return a string describing the warning or error.

A driver function can return three different types of warnings or errors. The function "bu5716_error_message" will handle all three types of warning/error codes by passing them to the appropriate function if necessary ("bu3100_error_message" or "viStatusDesc"), to return the correct warning/error message.

VISA Warnings/Errors:

See section 3.3 of the VPP 4.3.2 document for a complete list of VISA status codes and their values. The VPP 4.3 document contains detailed descriptions of all VISA functions and the status codes returned by each of them.

BU3100 Warnings/Errors:

These are warning or error codes returned by the common motherboard interface library, which are used by the 5716 driver to access a ProDAQ motherboard. Warnings returned by the library will be in the range 0x3FFC0800 to 0x3FFC0900 and errors in the range 0xBFFC0800 to 0xBFFC0900. They are defined in the include file bu3100.h.

BU5716 Warnings/Errors:

Warning codes returned by the 5716 driver functions will be in the range 0x3FFC0B00 to 0x3FFC0FFF and error codes in the range 0xBFFC0B00 to 0xBFFC0FFF. They are defined in the include file bu5716.h.

6.1.6. bu5716_getChannelConf

```
ViStatus bu5716_getChannelConf (ViSession instrumentHandle,
                               ViInt16 channel, ViPInt16 mode,
                               ViPInt16 gain, ViPInt16 coupling,
                               ViPInt16 inputSource);
```

Purpose

This function returns the configuration of the specified channel.

Parameter List

instrumentHandle

Variable Type ViSession

The Instrument Handle is used to identify the unique session or communication channel between the driver and the instrument.

If more than one instrument of the same model type is used, this handle will be used to differentiate between them.

channel

Variable Type ViInt16

This parameter specifies for which channel the configuration will be returned.

Possible values are:

1-16 (for ProDAQ 3416 master card) or
1-8 (for ProDAQ 3424 master card)

or

bu5716_CH_ALL 0 All available channels (16 for ProDAQ 3416
and 8 for ProDAQ 3424 master function card)
will be configured with the same parameters.

mode

Variable Type ViInt16 (passed by reference)

This parameter returns the channel mode.

Possible values are:

bu5716_MODE_FB	0	Full-bridge configuration (default);
bu5716_MODE_HB	1	Half-bridge configuration, sensing connected;
bu5716_MODE_QB_120	2	Quarter-bridge configuration with 120 ohm completion resistors;
bu5716_MODE_QB_350	3	Quarter-bridge configuration with 350 ohm completion resistors;
bu5716_MODE_QB_USER	4	Quarter-bridge configuration with user selected plug-in completion resistors;
bu5716_MODE_ICP	5	Front-end configured to work with ICP (Integrated Circuit Piezoelectric) sensors;
bu5716_MODE_DI	6	Differential voltage;
bu5716_MODE_SE	7	Single-ended voltage.

gain

Variable Type ViInt16 (passed by reference)

This parameter returns the gain for the input channel.

Possible values are:

bu5716_GAIN_1	1	Gain 1
bu5716_GAIN_10	10	Gain 10
bu5716_GAIN_100	100	Gain 100
bu5716_GAIN_1000	1000	Gain 1000

coupling

Variable Type ViInt16 (passed by reference)

This parameter returns the channel's PGA (Programmable Gain Amplifier) coupling mode.

Possible values are:

bu5716_COUP_AC	0	Both PGA inputs are AC coupled.
bu5716_COUP_DC	1	Both PGA inputs are DC coupled.

inputSource

Variable Type ViInt16 (passed by reference)

This parameter returns the channel's input source.

Possible values are:

bu5716_INP_SRC_FP	0	Front-panel
bu5716_INP_SRC_VREF	1	Voltage Reference

Return Value

If the function was successful, it will return a status of VI_SUCCESS, otherwise it will return a warning or error code. Passing the status code to the function "bu5716_error_message" will return a string describing the warning or error.

A driver function can return three different types of warnings or errors. The function "bu5716_error_message" will handle all three types of warning/error codes by passing them to the appropriate function if necessary ("bu3100_error_message" or "viStatusDesc"), to return the correct warning/error message.

VISA Warnings/Errors:

See section 3.3 of the VPP 4.3.2 document for a complete list of VISA status codes and their values. The VPP 4.3 document contains detailed descriptions of all VISA functions and the status codes returned by each of them.

BU3100 Warnings/Errors:

These are warning or error codes returned by the common motherboard interface library, which are used by the 5716 driver to access a ProDAQ motherboard. Warnings returned by the library will be in the range 0x3FFC0800 to 0x3FFC0900 and errors in the range 0xBFFC0800 to 0xBFFC0900. They are defined in the include file bu3100.h.

BU5716 Warnings/Errors:

Warning codes returned by the 5716 driver functions will be in the range 0x3FFC0B00 to 0x3FFC0FFF and error codes in the range 0xBFFC0B00 to 0xBFFC0FFF. They are defined in the include file bu5716.h.

6.1.7. bu5716_getCoupling

```
ViStatus bu5716_getCoupling (ViSession instrumentHandle,  
                             ViInt16 channel, ViPInt16 coupling);
```

Purpose

This function returns the PGA coupling mode of the specified channel.

Parameter List

instrumentHandle

Variable Type ViSession

The Instrument Handle is used to identify the unique session or communication channel between the driver and the instrument.

If more than one instrument of the same model type is used, this handle will be used to differentiate between them.

channel

Variable Type ViInt16

This parameter specifies for which channel the configuration will be returned.

Possible values are:

1-16 (for ProDAQ 3416 master card) or
1-8 (for ProDAQ 3424 master card).

coupling

Variable Type ViInt16 (passed by reference)

This parameter returns the channel's PGA coupling mode.

Possible values are:

bu5716_COUP_AC 0 Both PGA inputs are AC coupled.
bu5716_COUP_DC 1 Both PGA inputs are DC coupled.

Return Value

If the function was successful, it will return a status of VI_SUCCESS, otherwise it will return a warning or error code. Passing the status code to the function "bu5716_error_message" will return a string describing the warning or error.

A driver function can return three different types of warnings or errors. The function "bu5716_error_message" will handle all three types of warning/error codes by passing them to the appropriate function if necessary ("bu3100_error_message" or "viStatusDesc"), to return the correct warning/error message.

VISA Warnings/Errors:

See section 3.3 of the VPP 4.3.2 document for a complete list of VISA status codes and their values. The VPP 4.3 document contains detailed descriptions of all VISA functions and the status codes returned by each of them.

BU3100 Warnings/Errors:

These are warning or error codes returned by the common motherboard interface library, which are used by the 5716 driver to access a ProDAQ motherboard. Warnings returned by the library will be in the range 0x3FFC0800 to 0x3FFC0900 and errors in the range 0xBFFC0800 to 0xBFFC0900. They are defined in the include file bu3100.h.

BU5716 Warnings/Errors:

Warning codes returned by the 5716 driver functions will be in the range 0x3FFC0B00 to 0x3FFC0FFF and error codes in the range 0xBFFC0B00 to 0xBFFC0FFF. They are defined in the include file bu5716.h.

6.1.8. bu5716_getDAC

```
ViStatus bu5716_getDAC (ViSession instrumentHandle, ViInt16 channel,
                       ViInt16 DAC, ViPReal64 value);
```

Purpose

This function returns the DAC output value of the specified channel.

Parameter List

instrumentHandle

Variable Type ViSession

The Instrument Handle is used to identify the unique session or communication channel between the driver and the instrument.

If more than one instrument of the same model type is used, this handle will be used to differentiate between them.

channel

Variable Type ViInt16

This parameter specifies for which channel the configuration will be returned.

Possible values are:

1-16 (for ProDAQ 3416 master card) or
1-8 (for ProDAQ 3424 master card).

DAC

Variable Type ViInt16

This parameter specifies which DAC value will be returned.

Possible values are:

bu5716_DAC_MAIN_POS	0	Main DAC positive pass
bu5716_DAC_MAIN_NEG	1	Main DAC negative pass
bu5716_DAC_AUTO_POS	2	Autobalance DAC positive pass
bu5716_DAC_AUTO_NEG	3	Autobalance DAC negative pass

value

Variable Type ViReal64 (passed by reference)

The value of the specified DAC.

Possible values are:

For main DAC positive pass: 0 - 10V;
For main DAC negative pass: -10 - 0V;
For autobalance DAC positive pass: 0 - 10V;
For autobalance DAC negative pass: -10 - 0V.

Return Value

If the function was successful, it will return a status of VI_SUCCESS, otherwise it will return a warning or error code. Passing the status code to the function "bu5716_error_message" will return a string describing the warning or error.

A driver function can return three different types of warnings or errors. The function "bu5716_error_message" will handle all three types of warning/error codes by passing them to the appropriate function if necessary ("bu3100_error_message" or "viStatusDesc"), to return the correct warning/error message.

VISA Warnings/Errors:

See section 3.3 of the VPP 4.3.2 document for a complete list of VISA status codes and their values. The VPP 4.3 document contains detailed descriptions of all VISA functions and the status codes returned by each of them.

BU3100 Warnings/Errors:

These are warning or error codes returned by the common motherboard interface library, which are used by the 5716 driver to access a ProDAQ motherboard. Warnings returned by the library will be in the range 0x3FFC0800 to 0x3FFC0900 and errors in the range 0xBFFC0800 to 0xBFFC0900. They are defined in the include file bu3100.h.

BU5716 Warnings/Errors:

Warning codes returned by the 5716 driver functions will be in the range 0x3FFC0B00 to 0x3FFC0FFF and error codes in the range 0xBFFC0B00 to 0xBFFC0FFF. They are defined in the include file bu5716.h.

6.1.9. bu5716_getExcCalibCoeff

```
ViStatus bu5716_getExcCalibCoeff (ViSession instrumentHandle,
                                  ViInt16 channel, ViReal64 posOffset,
                                  ViReal64 negOffset,
                                  ViReal64 posGain, ViReal64 negGain);
```

Purpose

This function retrieves the excitation calibration coefficients currently being used for the specified channel and gain setting.

Parameter List

instrumentHandle

Variable Type ViSession

The Instrument Handle is used to identify the unique session or communication channel between the driver and the instrument.

If more than one instrument of the same model type is used, this handle will be used to differentiate between them.

channel

Variable Type ViInt16

This parameter specifies for which channel the calibration coefficients will be returned.

Possible values are:

1-16 (for ProDAQ 3416 master card) or
1-8 (for ProDAQ 3424 master card).

posOffset

Variable Type ViReal64 (passed by reference)

Returns the calibration positive offset value.

negOffset

Variable Type ViReal64 (passed by reference)

Returns the calibration negative offset value.

posGain

Variable Type ViReal64 (passed by reference)

Returns the calibration positive gain value.

negGain

Variable Type ViReal64 (passed by reference)

Returns the calibration negative gain value.

Return Value

If the function was successful, it will return a status of VI_SUCCESS, otherwise it will return a warning or error code. Passing the status code to the function "bu5716_error_message" will return a string describing the warning or error.

A driver function can return three different types of warnings or errors. The function "bu5716_error_message" will handle all three types of warning/error codes by passing them to the appropriate function if necessary ("bu3100_error_message" or "viStatusDesc"), to return the correct warning/error message.

VISA Warnings/Errors:

See section 3.3 of the VPP 4.3.2 document for a complete list of VISA status codes and their values. The VPP 4.3 document contains detailed descriptions of all VISA functions and the status codes returned by each of them.

BU3100 Warnings/Errors:

These are warning or error codes returned by the common motherboard interface library, which are used by the 5716 driver to access a ProDAQ motherboard. Warnings returned by the library will be in the range 0x3FFC0800 to 0x3FFC0900 and errors in the range 0xBFFC0800 to 0xBFFC0900. They are defined in the include file bu3100.h.

BU5716 Warnings/Errors:

Warning codes returned by the 5716 driver functions will be in the range 0x3FFC0B00 to 0x3FFC0FFF and error codes in the range 0xBFFC0B00 to 0xBFFC0FFF. They are defined in the include file bu5716.h.

6.1.10. bu5716_getExcitation

```
ViStatus bu5716_getExcitation (ViSession instrumentHandle,
                              ViInt16 channel,
                              ViPReal64 positiveExcitationVoltage,
                              ViPReal64 negativeExcitationVoltage);
```

Purpose

This function returns the excitation voltage of the specified channel.

Parameter List

instrumentHandle

Variable Type ViSession

The Instrument Handle is used to identify the unique session or communication channel between the driver and the instrument.

If more than one instrument of the same model type is used, this handle will be used to differentiate between them.

channel

Variable Type ViInt16

This parameter specifies for which channel the excitation voltage will be returned.

Possible values are:

1-16 (for ProDAQ 3416 master card) or
1-8 (for ProDAQ 3424 master card)

or

bu5716_CH_ALL 0 All available channels (16 for ProDAQ 3416
and 8 for ProDAQ 3424 master function card)
will be configured with the same parameters.

positiveExcitationVoltage

Variable Type ViReal64 (passed by reference)

This parameter returns the bridge positive, excitation voltage for the specified channel.

Valid values are: 0.0 to +10.0 volts.

negativeExcitationVoltage

Variable Type ViReal64 (passed by reference)

This parameter returns the bridge negative, excitation voltage for the specified channel.

Valid values are: -10.0 to 0.0 volts.

Return Value

If the function was successful, it will return a status of VI_SUCCESS, otherwise it will return a warning or error code. Passing the status code to the function "bu5716_error_message" will return a string describing the warning or error.

A driver function can return three different types of warnings or errors. The function "bu5716_error_message" will handle all three types of warning/error codes by passing them to the appropriate function if necessary ("bu3100_error_message" or "viStatusDesc"), to return the correct warning/error message.

VISA Warnings/Errors:

See section 3.3 of the VPP 4.3.2 document for a complete list of VISA status codes and their values. The VPP 4.3 document contains detailed descriptions of all VISA functions and the status codes returned by each of them.

BU3100 Warnings/Errors:

These are warning or error codes returned by the common motherboard interface library, which are used by the 5716 driver to access a ProDAQ motherboard. Warnings returned by the library will be in the range 0x3FFC0800 to 0x3FFC0900 and errors in the range 0xBFFC0800 to 0xBFFC0900. They are defined in the include file bu3100.h.

BU5716 Warnings/Errors:

Warning codes returned by the 5716 driver functions will be in the range 0x3FFC0B00 to 0x3FFC0FFF and error codes in the range 0xBFFC0B00 to 0xBFFC0FFF. They are defined in the include file bu5716.h.

6.1.11. bu5716_getExcitationMonitor

```
ViStatus bu5716_getExcitationMonitor (ViSession instrumentHandle,
                                      ViPInt16 mode);
```

Purpose

This function returns whether the excitation voltage source is connected/disconnected to/from all channels.

Parameter List

instrumentHandle

Variable Type ViSession

The Instrument Handle is used to identify the unique session or communication channel between the driver and the instrument.

If more than one instrument of the same model type is used, this handle will be used to differentiate between them.

mode

Variable Type ViInt16 (passed by reference)

This parameter returns which excitation (EX) voltage will be monitored.

Possible values are:

bu5716_EXCMON_OFF	0	All channel inputs are connected to the front-panel or voltage reference;
bu5716_EXCMON_P	1	All channel positive inputs are connected to EX+ signal and negative inputs are connected to GND;
bu5716_EXCMON_N	2	All channel negative inputs are connected to EX- signal and positive inputs are connected to GND;
bu5716_EXCMON_PN	3	All channel positive inputs are connected to EX+ signal and negative inputs are connected to EX- signal.

Return Value

If the function was successful, it will return a status of VI_SUCCESS, otherwise it will return a warning or error code. Passing the status code to the function "bu5716_error_message" will return a string describing the warning or error.

A driver function can return three different types of warnings or errors. The function "bu5716_error_message" will handle all three types of warning/error codes by passing them to the appropriate function if necessary ("bu3100_error_message" or "viStatusDesc"), to return the correct warning/error message.

VISA Warnings/Errors:

See section 3.3 of the VPP 4.3.2 document for a complete list of VISA status codes and their values. The VPP 4.3 document contains detailed descriptions of all VISA functions and the status codes returned by each of them.

BU3100 Warnings/Errors:

These are warning or error codes returned by the common motherboard interface library, which are used by the 5716 driver to access a ProDAQ motherboard. Warnings returned by the library will be in the range 0x3FFC0800 to 0x3FFC0900 and errors in the range 0xBFFC0800 to 0xBFFC0900. They are defined in the include file bu3100.h.

BU5716 Warnings/Errors:

Warning codes returned by the 5716 driver functions will be in the range 0x3FFC0B00 to 0x3FFC0FFF and error codes in the range 0xBFFC0B00 to 0xBFFC0FFF. They are defined in the include file bu5716.h.

6.1.12. bu5716_getFCLastError

```
ViStatus bu5716_getFCLastError (ViSession instrumentHandle,  
                               ViPStatus fcErrorCode);
```

Purpose

The last error code returned by the function card driver which controls the ProDAQ 5716. Useful in the event of the error bu5716_MASTER_ERR in response to the ProDAQ 5716 driver function call.

Parameter List

instrumentHandle

Variable Type ViSession

The Instrument Handle is used to identify the unique session or communication channel between the driver and the instrument.

If more than one instrument of the same model type is used, this handle will be used to differentiate between them.

fcErrorCode

Variable Type ViStatus (passed by reference)

Last error code returned by the function card driver which controls the ProDAQ 5716. This code is related to the function card driver and should be passed only to bu3416_error_message() when using the ProDAQ 3416 as the master card or to bu3424_error_message() when using the ProDAQ 3424.

Return Value

If the function was successful, it will return a status of VI_SUCCESS, otherwise it will return a warning or error code. Passing the status code to the function "bu5716_error_message" will return a string describing the warning or error.

A driver function can return three different types of warnings or errors. The function "bu5716_error_message" will handle all three types of warning/error codes by passing them to the appropriate function if necessary ("bu3100_error_message" or "viStatusDesc"), to return the correct warning/error message.

VISA Warnings/Errors:

See section 3.3 of the VPP 4.3.2 document for a complete list of VISA status codes and their values. The VPP 4.3 document contains detailed descriptions of all VISA functions and the status codes returned by each of them.

BU3100 Warnings/Errors:

These are warning or error codes returned by the common motherboard interface library, which are used by the 5716 driver to access a ProDAQ motherboard. Warnings returned by the library will be in the range 0x3FFC0800 to 0x3FFC0900 and errors in the range 0xBFFC0800 to 0xBFFC0900. They are defined in the include file bu3100.h.

BU5716 Warnings/Errors:

Warning codes returned by the 5716 driver functions will be in the range 0x3FFC0B00 to 0x3FFC0FFF and error codes in the range 0xBFFC0B00 to 0xBFFC0FFF. They are defined in the include file bu5716.h.

6.1.13. bu5716_getGain

```
ViStatus bu5716_getGain (ViSession instrumentHandle, ViInt16 channel,
                        ViPInt16 gain);
```

Purpose

This function returns the PGA gain setting of the specified channel.

Parameter List

instrumentHandle

Variable Type ViSession

The Instrument Handle is used to identify the unique session or communication channel between the driver and the instrument.

If more than one instrument of the same model type is used, this handle will be used to differentiate between them.

channel

Variable Type ViInt16

This parameter specifies for which channel the configuration will be returned.

Possible values are:

1-16 (for ProDAQ 3416 master card) or
1-8 (for ProDAQ 3424 master card).

gain

Variable Type ViInt16 (passed by reference)

This parameter returns the channel's PGA gain.

Possible values are:

bu5716_GAIN_1	1	Gain 1
bu5716_GAIN_10	10	Gain 10
bu5716_GAIN_100	100	Gain 100
bu5716_GAIN_1000	1000	Gain 1000

Return Value

If the function was successful, it will return a status of VI_SUCCESS, otherwise it will return a warning or error code. Passing the status code to the function "bu5716_error_message" will return a string describing the warning or error.

A driver function can return three different types of warnings or errors. The function "bu5716_error_message" will handle all three types of warning/error codes by passing them to the appropriate function if necessary ("bu3100_error_message" or "viStatusDesc"), to return the correct warning/error message.

VISA Warnings/Errors:

See section 3.3 of the VPP 4.3.2 document for a complete list of VISA status codes and their values. The VPP 4.3 document contains detailed descriptions of all VISA functions and the status codes returned by each of them.

BU3100 Warnings/Errors:

These are warning or error codes returned by the common motherboard interface library, which are used by the 5716 driver to access a ProDAQ motherboard. Warnings returned by the library will be in the range 0x3FFC0800 to 0x3FFC0900 and errors in the range 0xBFFC0800 to 0xBFFC0900. They are defined in the include file bu3100.h.

BU5716 Warnings/Errors:

Warning codes returned by the 5716 driver functions will be in the range 0x3FFC0B00 to 0x3FFC0FFF and error codes in the range 0xBFFC0B00 to 0xBFFC0FFF. They are defined in the include file bu5716.h.

6.1.14. bu5716_getInputSrc

```
ViStatus bu5716_getInputSrc (ViSession instrumentHandle,
                             ViInt16 channel, ViPInt16 inputSource);
```

Purpose

This function returns the input source of the specified channel.

Parameter List

instrumentHandle

Variable Type ViSession

The Instrument Handle is used to identify the unique session or communication channel between the driver and the instrument.

If more than one instrument of the same model type is used, this handle will be used to differentiate between them.

channel

Variable Type ViInt16

This parameter specifies for which channel the configuration will be returned.

Possible values are:

1-16 (for ProDAQ 3416 master card) or
1-8 (for ProDAQ 3424 master card).

inputSource

Variable Type ViInt16 (passed by reference)

This parameter returns the channel's input source.

Possible values are:

bu5716_INP_SRC_FP 0 Front-panel (default)
bu5716_INP_SRC_VREF 1 Voltage Reference

Return Value

If the function was successful, it will return a status of VI_SUCCESS, otherwise it will return a warning or error code. Passing the status code to the function "bu5716_error_message" will return a string describing the warning or error.

A driver function can return three different types of warnings or errors. The function "bu5716_error_message" will handle all three types of warning/error codes by passing them to the appropriate function if necessary ("bu3100_error_message" or "viStatusDesc"), to return the correct warning/error message.

VISA Warnings/Errors:

See section 3.3 of the VPP 4.3.2 document for a complete list of VISA status codes and their values. The VPP 4.3 document contains detailed descriptions of all VISA functions and the status codes returned by each of them.

BU3100 Warnings/Errors:

These are warning or error codes returned by the common motherboard interface library, which are used by the 5716 driver to access a ProDAQ motherboard. Warnings returned by the library will be in the range 0x3FFC0800 to 0x3FFC0900 and errors in the range 0xBFFC0800 to 0xBFFC0900. They are defined in the include file bu3100.h.

BU5716 Warnings/Errors:

Warning codes returned by the 5716 driver functions will be in the range 0x3FFC0B00 to 0x3FFC0FFF and error codes in the range 0xBFFC0B00 to 0xBFFC0FFF. They are defined in the include file bu5716.h.

6.1.15. bu5716_getMode

```
ViStatus bu5716_getMode (ViSession instrumentHandle, ViInt16 channel,
                        ViPInt16 mode);
```

Purpose

This function returns the mode of the specified channel.

Parameter List

instrumentHandle

Variable Type ViSession

The Instrument Handle is used to identify the unique session or communication channel between the driver and the instrument.

If more than one instrument of the same model type is used, this handle will be used to differentiate between them.

channel

Variable Type ViInt16

This parameter specifies for which channel the configuration will be returned.

Possible values are:

1-16 (for ProDAQ 3416 master card) or
1-8 (for ProDAQ 3424 master card).

mode

Variable Type ViInt16 (passed by reference)

This parameter returns the channel mode.

Possible values are:

bu5716_MODE_FB	0	Full-bridge configuration (default);
bu5716_MODE_HB	1	Half-bridge configuration, sensing connected;
bu5716_MODE_QB_120	2	Quarter-bridge configuration with 120 ohm completion resistors;
bu5716_MODE_QB_350	3	Quarter-bridge configuration with 350 ohm completion resistors;
bu5716_MODE_QB_USER	4	Quarter-bridge configuration with user selected plug-in completion resistors;
bu5716_MODE_ICP	5	Front-end configured to work with ICP sensors;
bu5716_MODE_DI	6	Differential voltage;
bu5716_MODE_SE	7	Single-ended voltage.

Return Value

If the function was successful, it will return a status of VI_SUCCESS, otherwise it will return a warning or error code. Passing the status code to the function "bu5716_error_message" will return a string describing the warning or error.

A driver function can return three different types of warnings or errors. The function "bu5716_error_message" will handle all three types of warning/error codes by passing them to the appropriate function if necessary ("bu3100_error_message" or "viStatusDesc"), to return the correct warning/error message.

VISA Warnings/Errors:

See section 3.3 of the VPP 4.3.2 document for a complete list of VISA status codes and their values. The VPP 4.3 document contains detailed descriptions of all VISA functions and the status codes returned by each of them.

BU3100 Warnings/Errors:

These are warning or error codes returned by the common motherboard interface library, which are used by the 5716 driver to access a ProDAQ motherboard. Warnings returned by the library will be in the range 0x3FFC0800 to 0x3FFC0900 and errors in the range 0xBFFC0800 to 0xBFFC0900. They are defined in the include file bu3100.h.

BU5716 Warnings/Errors:

Warning codes returned by the 5716 driver functions will be in the range 0x3FFC0B00 to 0x3FFC0FFF and error codes in the range 0xBFFC0B00 to 0xBFFC0FFF. They are defined in the include file bu5716.h.

6.1.16. bu5716_getPgaCalibCoeff

```
ViStatus bu5716_getPgaCalibCoeff (ViSession instrumentHandle,
                                  ViInt16 channel, ViInt16 idealGain,
                                  ViPReal64 offset, ViPReal64 gain);
```

Purpose

This function retrieves the PGA calibration coefficients currently being used for the specified channel and gain setting.

Parameter List

instrumentHandle

Variable Type ViSession

The Instrument Handle is used to identify the unique session or communication channel between the driver and the instrument.

If more than one instrument of the same model type is used, this handle will be used to differentiate between them.

channel

Variable Type ViInt16

This parameter specifies for which channel the calibration coefficients will be returned.

Possible values are:

1-16 (for ProDAQ 3416 master card) or
1-8 (for ProDAQ 3424 master card).

idealGain

Variable Type ViInt16

This parameter specifies the gain for which the calibration coefficients will be retrieved.

Possible values are:

bu5716_GAIN_1	1	Gain 1
bu5716_GAIN_10	10	Gain 10
bu5716_GAIN_100	100	Gain 100
bu5716_GAIN_1000	1000	Gain 1000

offset

Variable Type ViReal64 (passed by reference)

This parameter returns the calibration offset value.

gain

Variable Type ViReal64 (passed by reference)

This parameter returns the calibration gain value.

Return Value

If the function was successful, it will return a status of VI_SUCCESS, otherwise it will return a warning or error code. Passing the status code to the function "bu5716_error_message" will return a string describing the warning or error.

A driver function can return three different types of warnings or errors. The function "bu5716_error_message" will handle all three types of warning/error codes by passing them to the appropriate function if necessary ("bu3100_error_message" or "viStatusDesc"), to return the correct warning/error message.

VISA Warnings/Errors:

See section 3.3 of the VPP 4.3.2 document for a complete list of VISA status codes and their values. The VPP 4.3 document contains

detailed descriptions of all VISA functions and the status codes returned by each of them.

BU3100 Warnings/Errors:

These are warning or error codes returned by the common motherboard interface library, which are used by the 5716 driver to access a ProDAQ motherboard. Warnings returned by the library will be in the range 0x3FFC0800 to 0x3FFC0900 and errors in the range 0xBFFC0800 to 0xBFFC0900. They are defined in the include file bu3100.h.

BU5716 Warnings/Errors:

Warning codes returned by the 5716 driver functions will be in the range 0x3FFC0B00 to 0x3FFC0FFF and error codes in the range 0xBFFC0B00 to 0xBFFC0FFF. They are defined in the include file bu5716.h.

6.1.17. bu5716_getShuntCal

```
ViStatus bu5716_getShuntCal (ViSession instrumentHandle,
                             ViInt16 channel, ViPBoolean connect);
```

Purpose

This function returns the state of the calibration shunt resistor of the specified channel.

Parameter List

instrumentHandle

Variable Type ViSession

The Instrument Handle is used to identify the unique session or communication channel between the driver and the instrument.

If more than one instrument of the same model type is used, this handle will be used to differentiate between them.

channel

Variable Type ViInt16

This parameter specifies for which channel the configuration will be returned.

Possible values are:

1-16 (for ProDAQ 3416 master card) or
1-8 (for ProDAQ 3424 master card).

connect

Variable Type ViBoolean (passed by reference)

This parameter specifies whether the calibration shunt is connected or disconnected.

Possible values are:

VI_FALSE 0 The calibration shunt is disconnected.
VI_TRUE 1 The calibration shunt is connected.

Return Value

If the function was successful, it will return a status of VI_SUCCESS, otherwise it will return a warning or error code. Passing the status code to the function "bu5716_error_message" will return a string describing the warning or error.

A driver function can return three different types of warnings or errors. The function "bu5716_error_message" will handle all three types of warning/error codes by passing them to the appropriate function if necessary ("bu3100_error_message" or "viStatusDesc"), to return the correct warning/error message.

VISA Warnings/Errors:

See section 3.3 of the VPP 4.3.2 document for a complete list of VISA status codes and their values. The VPP 4.3 document contains detailed descriptions of all VISA functions and the status codes returned by each of them.

BU3100 Warnings/Errors:

These are warning or error codes returned by the common motherboard interface library, which are used by the 5716 driver to access a ProDAQ motherboard. Warnings returned by the library will be in the range 0x3FFC0800 to 0x3FFC0900 and errors in the range 0xBFFC0800 to 0xBFFC0900. They are defined in the include file bu3100.h.

BU5716 Warnings/Errors:

Warning codes returned by the 5716 driver functions will be in the range 0x3FFC0B00 to 0x3FFC0FFF and error codes in the range 0xBFFC0B00 to 0xBFFC0FFF. They are defined in the include file bu5716.h.

6.1.18. bu5716_getStatus

```
ViStatus bu5716_getStatus (ViSession instrumentHandle,  
                          ViPInt16 icpStatus, ViPInt16 exlStatus,  
                          ViPInt16 icpFlags, ViPInt16 exlFlags);
```

Purpose

This function returns the status of the ICP current and excitation current limit detection.

Parameter List

instrumentHandle

Variable Type ViSession

The Instrument Handle is used to identify the unique session or communication channel between the driver and the instrument.

If more than one instrument of the same model type is used, this handle will be used to differentiate between them.

icpStatus

Variable Type ViInt16 (passed by reference)

This parameter returns a bitmask of the actual current states on ICP. Each bit represents the following channels:

```
Bit0 - channel 1  
Bit1 - channel 2  
...  
Bit15 - channel 15
```

Possible values:

```
0 - No valid ICP current detected.  
1 - ICP current detected.
```

NOTE: When the device is controlled by the ProDAQ 3424, only the eight lowest bits are valid.

exlStatus

Variable Type ViInt16 (passed by reference)

This parameter returns a bitmask of the actual current limit state on excitation. Each bit represents the following channels:

```
Bit0 - channel 1  
Bit1 - channel 2  
...  
Bit15 - channel 15
```

Possible values:

```
0 - No Excitation Current Limit condition detected.  
1 - Excitation Current Limit condition detected.
```

NOTE: When the device is controlled by the ProDAQ 3424, only the eight lowest bits are valid.

icpFlags

Variable Type ViInt16 (passed by reference)

This parameter returns a bitmask of the current errors on ICP. Each bit represents the following channels:

```
Bit0 - channel 1  
Bit1 - channel 2  
...  
Bit15 - channel 15
```

Possible values:

```
0 - No new ICP current errors detected since last readout.  
1 - ICP current error occurred.
```

NOTE: When the device is controlled by the ProDAQ 3424, only the eight lowest bits are valid.

exlFlags

Variable Type ViInt16 (passed by reference)

This parameter returns a bitmask of the current errors on excitation. Each bit represents the following channels:

Bit0 - channel 1
Bit1 - channel 2
...
Bit15 - channel 15

Possible values:

0 - No Excitation Current Limit condition detected.
1 - Excitation Current Limit condition detected.

NOTE: When the device is controlled by the ProDAQ 3424, only the eight lowest bits are valid.

Return Value

If the function was successful, it will return a status of VI_SUCCESS, otherwise it will return a warning or error code. Passing the status code to the function "bu5716_error_message" will return a string describing the warning or error.

A driver function can return three different types of warnings or errors. The function "bu5716_error_message" will handle all three types of warning/error codes by passing them to the appropriate function if necessary ("bu3100_error_message" or "viStatusDesc"), to return the correct warning/error message.

VISA Warnings/Errors:

See section 3.3 of the VPP 4.3.2 document for a complete list of VISA status codes and their values. The VPP 4.3 document contains detailed descriptions of all VISA functions and the status codes returned by each of them.

BU3100 Warnings/Errors:

These are warning or error codes returned by the common motherboard interface library, which are used by the 5716 driver to access a ProDAQ motherboard. Warnings returned by the library will be in the range 0x3FFC0800 to 0x3FFC0900 and errors in the range 0xBFFC0800 to 0xBFFC0900. They are defined in the include file bu3100.h.

BU5716 Warnings/Errors:

Warning codes returned by the 5716 driver functions will be in the range 0x3FFC0B00 to 0x3FFC0FFF and error codes in the range 0xBFFC0B00 to 0xBFFC0FFF. They are defined in the include file bu5716.h.

6.1.19. bu5716_getVoltRefInfo

```
ViStatus bu5716_getVoltRefInfo (ViSession instrumentHandle,
                               ViInt32 voltRefModule, ViPInt32 nVolts,
                               ViReal64 _VI_FAR voltages[]);
```

Purpose

This function returns the voltages available on the specified voltage reference module.

Parameter List

instrumentHandle

Variable Type ViSession

The Instrument Handle is used to identify the unique session or communication channel between the driver and the instrument. If more than one instrument of the same model type is used, this handle will be used to differentiate between them.

voltRefModule

Variable Type ViInt32

This parameter specifies for which voltage reference module the information should be returned.

Possible values are:

bu5716_VREF_SRC_FC	1	Voltage reference module located on the motherboard module of the master function card.
--------------------	---	---

nVolts

Variable Type ViInt32 (passed by reference)

Returns the number of voltages available on the selected voltage reference module.

voltages

Variable Type ViReal64[]

This array contains the list of all possible voltages generated by the voltage reference module. It should be allocated with size 20 prior to the function call.

Return Value

If the function was successful, it will return a status of VI_SUCCESS, otherwise it will return a warning or error code. Passing the status code to the function "bu5716_error_message" will return a string describing the warning or error.

A driver function can return three different types of warnings or errors. The function "bu5716_error_message" will handle all three types of warning/error codes by passing them to the appropriate function if necessary ("bu3100_error_message" or "viStatusDesc"), to return the correct warning/error message.

VISA Warnings/Errors:

See section 3.3 of the VPP 4.3.2 document for a complete list of VISA status codes and their values. The VPP 4.3 document contains detailed descriptions of all VISA functions and the status codes returned by each of them.

BU3100 Warnings/Errors:

These are warning or error codes returned by the common motherboard interface library, which are used by the 5716 driver to access a ProDAQ motherboard. Warnings returned by the library will be in the range 0x3FFC0800 to 0x3FFC0900 and errors in the range 0xBFFC0800 to 0xBFFC0900. They are defined in the include file bu3100.h.

BU5716 Warnings/Errors:

Warning codes returned by the 5716 driver functions will be in the range 0x3FFC0B00 to 0x3FFC0FFF and error codes in the range 0xBFFC0B00 to 0xBFFC0FFF. They are defined in the include file bu5716.h.

6.1.20. bu5716_getVoltRefOutput

```
ViStatus bu5716_getVoltRefOutput (ViSession instrumentHandle,
                                  ViInt32 voltRefModule,
                                  ViPReal64 voltage,
                                  ViPBoolean monitor);
```

Purpose

This function returns the current output voltage of the voltage reference module.

Parameter List

instrumentHandle

Variable Type ViSession

The Instrument Handle is used to identify the unique session or communication channel between the driver and the instrument.

If more than one instrument of the same model type is used, this handle will be used to differentiate between them.

voltRefModule

Variable Type ViInt32 (passed by reference)

This parameter returns which voltage reference module is currently used to generate the voltage reference.

Possible values are:

bu5716_VREF_SRC_OFF	0	All voltage reference modules are disconnected;
bu5716_VREF_SRC_FC	1	Voltage reference module located on the motherboard module of the master function card;
bu5716_VREF_SRC_ONB	2	Short to ground is used;
bu5716_VREF_SRC_EXT	3	External voltage reference is used.

voltage

Variable Type ViReal64 (passed by reference)

This parameter returns the current output voltage of the voltage reference module.

monitor

Variable Type ViBoolean (passed by reference)

This parameter returns whether the voltage reference output is routed to the ProDAQ 5716 rear panel connector (for monitoring purposes).

Possible values are:

VI_FALSE	Voltage reference output is disconnected from the ProDAQ 5716 rear panel connector.
VI_TRUE	Voltage reference output is connected to the ProDAQ 5716 rear panel connector.

Return Value

If the function was successful, it will return a status of VI_SUCCESS, otherwise it will return a warning or error code. Passing the status code to the function "bu5716_error_message" will return a string describing the warning or error.

A driver function can return three different types of warnings or errors. The function "bu5716_error_message" will handle all three types of warning/error codes by passing them to the appropriate function if necessary ("bu3100_error_message" or "viStatusDesc"), to return the correct warning/error message.

VISA Warnings/Errors:

See section 3.3 of the VPP 4.3.2 document for a complete list of VISA status codes and their values. The VPP 4.3 document contains

detailed descriptions of all VISA functions and the status codes returned by each of them.

BU3100 Warnings/Errors:

These are warning or error codes returned by the common motherboard interface library, which are used by the 5716 driver to access a ProDAQ motherboard. Warnings returned by the library will be in the range 0x3FFC0800 to 0x3FFC0900 and errors in the range 0xBFFC0800 to 0xBFFC0900. They are defined in the include file bu3100.h.

BU5716 Warnings/Errors:

Warning codes returned by the 5716 driver functions will be in the range 0x3FFC0B00 to 0x3FFC0FFF and error codes in the range 0xBFFC0B00 to 0xBFFC0FFF. They are defined in the include file bu5716.h.

6.1.21. bu5716_init

```
ViStatus bu5716_init (ViSession masterHandle, ViBoolean IDQuery,
                    ViBoolean resetDevice,
                    ViPSession instrumentHandle);
```

Purpose

Initializes the instrument and returns an "Instrument Handle". The instrument handle must be used with all of the other functions of this driver.

The initialize call allows the instrument to be queried to ensure that it is a Bustec data acquisition system. It also resets the module to the power-up state if the "Reset" parameter is True (ON).

This function interrogates the motherboard registers to ascertain in which locations there are function cards fitted and then checks those locations to identify the type of function card fitted.

Note that for each "bu5716_init()" call, a new unique instrument handle is returned. Thus, if four calls are made to the initialize call in succession, four unique instrument handles will be returned.

For each instrument handle returned by the "bu5716_init()" function, the "bu5716_close()" function should be called to free up the resources allocated by "bu5716_init()". The call(s) to "bu5716_close()" should be made before the application program terminates.

Parameter List

masterHandle

Variable Type ViSession

This control specifies the instrument handle of the master function card (For instance, ProDAQ 3416 or ProDAQ 3424 function cards). The VXI Plug&Play driver of the master function card should be initialized prior to this function call.

IDQuery

Variable Type ViBoolean

Specifies whether to send an ID Query to the instrument during the initialization procedure.

Valid Range: 1 = Yes
 0 = No

Default Value: 1 - Yes

NOTE: Under normal circumstances the ID Query ensures that the instrument initialized over the bus is the type supported by this driver. However, circumstances may arise where it is undesirable to send an ID Query to the instrument. In those cases, set this control to Skip Query and this function will initialize the bus and the command arrays in the driver, without doing an ID Query.

resetDevice

Variable Type ViBoolean

Specifies whether the instrument is to be reset to its power-on settings during the initialization procedure.

Valid Range: 1 = Yes
 0 = No

Default Value: 1 - Yes

NOTE: If you do not want the instrument reset, set this control to No while initializing the instrument.

instrumentHandle

Variable Type ViSession (passed by reference)

The Instrument Handle is used to identify the unique session or communication channel between the driver and the instrument.

If more than one instrument of the same model type is used, this handle will be used to differentiate between them.

NOTE: A new (unique) handle will be returned EACH time the initialize function is called. The `bu5716_close()` call should be used for EVERY handle returned by the `bu5716_init()` function.

Return Value

If the function was successful, it will return a status of `VI_SUCCESS`, otherwise it will return a warning or error code. Passing the status code to the function `"bu5716_error_message"` will return a string describing the warning or error.

A driver function can return three different types of warnings or errors. The function `"bu5716_error_message"` will handle all three types of warning/error codes by passing them to the appropriate function if necessary (`"bu3100_error_message"` or `"viStatusDesc"`), to return the correct warning/error message.

VISA Warnings/Errors:

See section 3.3 of the VPP 4.3.2 document for a complete list of VISA status codes and their values. The VPP 4.3 document contains detailed descriptions of all VISA functions and the status codes returned by each of them.

BU3100 Warnings/Errors:

These are warning or error codes returned by the common motherboard interface library, which are used by the 5716 driver to access a ProDAQ motherboard. Warnings returned by the library will be in the range `0x3FFC0800` to `0x3FFC0900` and errors in the range `0xBFFC0800` to `0xBFFC0900`. They are defined in the include file `bu3100.h`.

BU5716 Warnings/Errors:

Warning codes returned by the 5716 driver functions will be in the range `0x3FFC0B00` to `0x3FFC0FFF` and error codes in the range `0xBFFC0B00` to `0xBFFC0FFF`. They are defined in the include file `bu5716.h`.

6.1.22. bu5716_pgaCalibration

```
ViStatus bu5716_pgaCalibration (ViSession instrumentHandle,
                               ViInt16 channel, ViInt16 voltRefModule,
                               ViReal64 voltage,
                               ViReal64 _VI_FAR offsets[],
                               ViReal64 _VI_FAR gains[]);
```

Purpose

This function performs the PGA calibration of the specified channel. The calibration is performed for the current setting of the PGA, i.e. 1, 10, 100, or 1000.

Parameter List

instrumentHandle

Variable Type ViSession

The Instrument Handle is used to identify the unique session or communication channel between the driver and the instrument.

If more than one instrument of the same model type is used, this handle will be used to differentiate between them.

channel

Variable Type ViInt16

This parameter specifies which channel will be calibrated.

Possible values are:

1-16 (for ProDAQ 3416 master card) or
1-8 (for ProDAQ 3424 master card)

or

bu5716_CH_ALL 0 All available channels (16 for ProDAQ 3416 and 8 for ProDAQ 3424 master function card) will be configured with the same parameters.

voltRefModule

Variable Type ViInt16

This parameter specifies which voltage reference module will be used for calibration.

Possible values are:

bu5716_VREF_SRC_FC 1 Voltage reference module located on the motherboard module of the master function card.

bu5716_VREF_SRC_EXT 3 User voltage reference is used which is connected to monitor input. User must know exactly, the value of voltage and enter this value to the Voltage parameter.

voltage

Variable Type ViReal64

This parameter specifies the output voltage of the voltage reference module which will be used for the gain calibration. The available set of voltages depend on the type of the selected voltage reference module. If the voltage reference module is unable to generate the requested message, the function will return an error status.

offsets

Variable Type ViReal64[]

Returns the calibration offset values.

OR

This parameter returns the calibration coefficients (offset) for all

selected channels acquired during the calibration process.
The array should be allocated with the appropriate size, prior to the function call.

gains

Variable Type ViReal64[]

Returns the calibration gain values.

OR

This parameter returns the calibration coefficients (gain) for all selected channels acquired during the calibration process.
The array should be allocated with the appropriate size, prior to the function call.

Return Value

If the function was successful, it will return a status of VI_SUCCESS, otherwise it will return a warning or error code. Passing the status code to the function "bu5716_error_message" will return a string describing the warning or error.

A driver function can return three different types of warnings or errors. The function "bu5716_error_message" will handle all three types of warning/error codes by passing them to the appropriate function if necessary ("bu3100_error_message" or "viStatusDesc"), to return the correct warning/error message.

VISA Warnings/Errors:

See section 3.3 of the VPP 4.3.2 document for a complete list of VISA status codes and their values. The VPP 4.3 document contains detailed descriptions of all VISA functions and the status codes returned by each of them.

BU3100 Warnings/Errors:

These are warning or error codes returned by the common motherboard interface library, which are used by the 5716 driver to access a ProDAQ motherboard. Warnings returned by the library will be in the range 0x3FFC0800 to 0x3FFC0900 and errors in the range 0xBFFC0800 to 0xBFFC0900. They are defined in the include file bu3100.h.

BU5716 Warnings/Errors:

Warning codes returned by the 5716 driver functions will be in the range 0x3FFC0B00 to 0x3FFC0FFF and error codes in the range 0xBFFC0B00 to 0xBFFC0FFF. They are defined in the include file bu5716.h.

6.1.23. bu5716_readTEDS

```
ViStatus bu5716_readTEDS (ViSession instrumentHandle, ViInt16 channel,
                          ViInt16 TEDS, ViChar _VI_FAR ROM[],
                          ViChar _VI_FAR OTPROM[],
                          ViChar _VI_FAR EEPROM[]);
```

Purpose

This function reads the data from 1-Wire ROM, One Time Programmable ROM (OTP ROM) and EEPROM memory of the TEDS sensor.

Parameter List

instrumentHandle

Variable Type ViSession

The Instrument Handle is used to identify the unique session or communication channel between the driver and the instrument.

If more than one instrument of the same model type is used, this handle will be used to differentiate between them.

channel

Variable Type ViInt16

This parameter specifies from which channel the TEDS sensor will be read.

Possible values are:

1-16 (for ProDAQ 3416 master card) or
1-8 (for ProDAQ 3424 master card).

TEDS

Variable Type ViInt16

Selects which type of TEDS will be read.

Possible values are:

bu5716_TEDS_CLASS_1 0 TEDS Class 1 will be accessed.
bu5716_TEDS_CLASS_2 1 TEDS Class 2 will be accessed.

ROM

Variable Type ViChar[]

After a successful reading, this buffer will contain 8 bytes read from the TEDS 1-Wire ROM. The buffer should be allocated with the appropriate size prior to the function call. If the user passes NULL to this parameter, the 1-Wire ROM will not be read.

OTPROM

Variable Type ViChar[]

After a successful reading, this buffer will contain 8 bytes read from the TEDS One Time Programmable (OTP) ROM. The buffer should be allocated with the appropriate size prior to the function call. If the user passes NULL to this parameter, the OTP ROM will not be read.

EEPROM

Variable Type ViChar[]

After a successful reading, this buffer will contain 32 bytes read from the TEDS EEPROM. The buffer should be allocated with the appropriate size prior to the function call. If the user passes NULL to this parameter, the TEDS EEPROM will not be read.

Return Value

If the function was successful, it will return a status of VI_SUCCESS, otherwise it will return a warning or error code.

Passing the status code to the function "bu5716_error_message" will return a string describing the warning or error.

A driver function can return three different types of warnings or errors. The function "bu5716_error_message" will handle all three types of warning/error codes by passing them to the appropriate function if necessary ("bu3100_error_message" or "viStatusDesc"), to return the correct warning/error message.

VISA Warnings/Errors:

See section 3.3 of the VPP 4.3.2 document for a complete list of VISA status codes and their values. The VPP 4.3 document contains detailed descriptions of all VISA functions and the status codes returned by each of them.

BU3100 Warnings/Errors:

These are warning or error codes returned by the common motherboard interface library, which are used by the 5716 driver to access a ProDAQ motherboard. Warnings returned by the library will be in the range 0x3FFC0800 to 0x3FFC0900 and errors in the range 0xBFFC0800 to 0xBFFC0900. They are defined in the include file bu3100.h.

BU5716 Warnings/Errors:

Warning codes returned by the 5716 driver functions will be in the range 0x3FFC0B00 to 0x3FFC0FFF and error codes in the range 0xBFFC0B00 to 0xBFFC0FFF. They are defined in the include file bu5716.h.

6.1.24. bu5716_readTemperature

```
ViStatus bu5716_readTemperature (ViSession instrumentHandle,
                                ViPReal64 temperature1,
                                ViPReal64 temperature2);
```

Purpose

This function reads the temperature from the onboard ProDAQ 5716 temperature sensors.

Parameter List

instrumentHandle

Variable Type ViSession

The Instrument Handle is used to identify the unique session or communication channel between the driver and the instrument.

If more than one instrument of the same model type is used, this handle will be used to differentiate between them.

temperature1

Variable Type ViReal64 (passed by reference)

This control returns the temperature from onboard temperature sensor 1 in degrees Celsius.

temperature2

Variable Type ViReal64 (passed by reference)

This control returns the temperature from onboard temperature sensor 2 in degrees Celsius. If the ProDAQ 5716 is controlled by the ProDAQ 3424, this parameter will be ignored.

Return Value

If the function was successful, it will return a status of VI_SUCCESS, otherwise it will return a warning or error code. Passing the status code to the function "bu5716_error_message" will return a string describing the warning or error.

A driver function can return three different types of warnings or errors. The function "bu5716_error_message" will handle all three types of warning/error codes by passing them to the appropriate function if necessary ("bu3100_error_message" or "viStatusDesc"), to return the correct warning/error message.

VISA Warnings/Errors:

See section 3.3 of the VPP 4.3.2 document for a complete list of VISA status codes and their values. The VPP 4.3 document contains detailed descriptions of all VISA functions and the status codes returned by each of them.

BU3100 Warnings/Errors:

These are warning or error codes returned by the common motherboard interface library, which are used by the 5716 driver to access a ProDAQ motherboard. Warnings returned by the library will be in the range 0x3FFC0800 to 0x3FFC0900 and errors in the range 0xBFFC0800 to 0xBFFC0900. They are defined in the include file bu3100.h.

BU5716 Warnings/Errors:

Warning codes returned by the 5716 driver functions will be in the range 0x3FFC0B00 to 0x3FFC0FFF and error codes in the range 0xBFFC0B00 to 0xBFFC0FFF. They are defined in the include file bu5716.h.

6.1.25. bu5716_reset

```
ViStatus bu5716_reset (ViSession instrumentHandle);
```

Purpose

This function resets the function card to its power-on state.

Parameter List

instrumentHandle

Variable	Type	ViSession
----------	------	-----------

The Instrument Handle is used to identify the unique session or communication channel between the driver and the instrument.

If more than one instrument of the same model type is used, this handle will be used to differentiate between them.

Return Value

If the function was successful, it will return a status of VI_SUCCESS, otherwise it will return a warning or error code. Passing the status code to the function "bu5716_error_message" will return a string describing the warning or error.

A driver function can return three different types of warnings or errors. The function "bu5716_error_message" will handle all three types of warning/error codes by passing them to the appropriate function if necessary ("bu3100_error_message" or "viStatusDesc"), to return the correct warning/error message.

VISA Warnings/Errors:

See section 3.3 of the VPP 4.3.2 document for a complete list of VISA status codes and their values. The VPP 4.3 document contains detailed descriptions of all VISA functions and the status codes returned by each of them.

BU3100 Warnings/Errors:

These are warning or error codes returned by the common motherboard interface library, which are used by the 5716 driver to access a ProDAQ motherboard. Warnings returned by the library will be in the range 0x3FFC0800 to 0x3FFC0900 and errors in the range 0xBFFC0800 to 0xBFFC0900. They are defined in the include file bu3100.h.

BU5716 Warnings/Errors:

Warning codes returned by the 5716 driver functions will be in the range 0x3FFC0B00 to 0x3FFC0FFF and error codes in the range 0xBFFC0B00 to 0xBFFC0FFF. They are defined in the include file bu5716.h.

6.1.26. bu5716_resetCalibCoeff

```
ViStatus bu5716_resetCalibCoeff (ViSession instrumentHandle,
                                ViInt16 mode, ViInt16 type);
```

Purpose

Resets the calibration coefficients with default values or with values taken from the onboard EEPROM memory.

Parameter List

instrumentHandle

Variable Type ViSession

The Instrument Handle is used to identify the unique session or communication channel between the driver and the instrument.

If more than one instrument of the same model type is used, this handle will be used to differentiate between them.

mode

Variable Type ViInt16

This parameter specifies whether the calibration coefficients will be reset with default values or with values stored in the onboard EEPROM.

Possible values are:

bu5716_CALIB_COEFF_EEPROM	0	EEPROM values will be taken.
bu5716_CALIB_COEFF_DEFAULT	1	Default values will be taken.

type

Variable Type ViInt16

Specifies which type of calibration coefficient will be set.

Possible values are:

bu5716_PGA_CALIB_DATA	1	PGA calibration coefficient;
bu5716_EXC_CALIB_DATA	2	Excitation calibration coefficient;
bu5716_ALL_CALIB_DATA	3	PGA & Excitation calibration coefficient.

Return Value

If the function was successful, it will return a status of VI_SUCCESS, otherwise it will return a warning or error code. Passing the status code to the function "bu5716_error_message" will return a string describing the warning or error.

A driver function can return three different types of warnings or errors. The function "bu5716_error_message" will handle all three types of warning/error codes by passing them to the appropriate function if necessary ("bu3100_error_message" or "viStatusDesc"), to return the correct warning/error message.

VISA Warnings/Errors:

See section 3.3 of the VPP 4.3.2 document for a complete list of VISA status codes and their values. The VPP 4.3 document contains detailed descriptions of all VISA functions and the status codes returned by each of them.

BU3100 Warnings/Errors:

These are warning or error codes returned by the common motherboard interface library, which are used by the 5716 driver to access a ProDAQ motherboard. Warnings returned by the library will be in the range 0x3FFC0800 to 0x3FFC0900 and errors in the range 0xBFFC0800 to 0xBFFC0900. They are defined in the include file bu3100.h.

BU5716 Warnings/Errors:

Warning codes returned by the 5716 driver functions will be in the range 0x3FFC0B00 to 0x3FFC0FFF and error codes in the range 0xBFFC0B00 to 0xBFFC0FFF. They are defined in the include file bu5716.h.

6.1.27. bu5716_revision_query

```
ViStatus bu5716_revision_query (ViSession instrumentHandle,  
                               ViChar _VI_FAR driverRevision[],  
                               ViChar _VI_FAR instrumentFirmwareRevision[]);
```

Purpose

Returns information about the hardware and software revisions.

Parameter List

instrumentHandle

Variable Type ViSession

The Instrument Handle is used to identify the unique session or communication channel between the driver and the instrument.

If more than one instrument of the same model type is used, this handle will be used to differentiate between them.

driverRevision

Variable Type ViChar[]

Returns the instrument driver revision.

instrumentFirmwareRevision

Variable Type ViChar[]

Returns firmware revision string.

Return Value

If the function was successful, it will return a status of VI_SUCCESS, otherwise it will return a warning or error code. Passing the status code to the function "bu5716_error_message" will return a string describing the warning or error.

A driver function can return three different types of warnings or errors. The function "bu5716_error_message" will handle all three types of warning/error codes by passing them to the appropriate function if necessary ("bu3100_error_message" or "viStatusDesc"), to return the correct warning/error message.

VISA Warnings/Errors:

See section 3.3 of the VPP 4.3.2 document for a complete list of VISA status codes and their values. The VPP 4.3 document contains detailed descriptions of all VISA functions and the status codes returned by each of them.

BU3100 Warnings/Errors:

These are warning or error codes returned by the common motherboard interface library, which are used by the 5716 driver to access a ProDAQ motherboard. Warnings returned by the library will be in the range 0x3FFC0800 to 0x3FFC0900 and errors in the range 0xBFFC0800 to 0xBFFC0900. They are defined in the include file bu3100.h.

BU5716 Warnings/Errors:

Warning codes returned by the 5716 driver functions will be in the range 0x3FFC0B00 to 0x3FFC0FFF and error codes in the range 0xBFFC0B00 to 0xBFFC0FFF. They are defined in the include file bu5716.h.

6.1.28. bu5716_serialNumber

```
ViStatus bu5716_serialNumber (ViChar VI_FAR subversion[],
                             ViPInt32 serialNumber,
                             ViSession instrumentHandle);
```

Purpose

This function returns the ProDAQ 5716 device subversion and serial number.

Parameter List

subversion

Variable Type ViChar[]

This parameter returns the ProDAQ 5716 subversion in the form of a string, i.e., "AA" or "AB".

serialNumber

Variable Type ViInt32 (passed by reference)

This parameter returns the serial number of the ProDAQ 5716.

instrumentHandle

Variable Type ViSession

The Instrument Handle is used to identify the unique session or communication channel between the driver and the instrument.

If more than one instrument of the same model type is used, this handle will be used to differentiate between them.

Return Value

If the function was successful, it will return a status of VI_SUCCESS, otherwise it will return a warning or error code. Passing the status code to the function "bu5716_error_message" will return a string describing the warning or error.

A driver function can return three different types of warnings or errors. The function "bu5716_error_message" will handle all three types of warning/error codes by passing them to the appropriate function if necessary ("bu3100_error_message" or "viStatusDesc"), to return the correct warning/error message.

VISA Warnings/Errors:

See section 3.3 of the VPP 4.3.2 document for a complete list of VISA status codes and their values. The VPP 4.3 document contains detailed descriptions of all VISA functions and the status codes returned by each of them.

BU3100 Warnings/Errors:

These are warning or error codes returned by the common motherboard interface library, which are used by the 5716 driver to access a ProDAQ motherboard. Warnings returned by the library will be in the range 0x3FFFC0800 to 0x3FFFC0900 and errors in the range 0xBFFC0800 to 0xBFFC0900. They are defined in the include file bu3100.h.

BU5716 Warnings/Errors:

Warning codes returned by the 5716 driver functions will be in the range 0x3FFFC0B00 to 0x3FFFC0FFF and error codes in the range 0xBFFC0B00 to 0xBFFC0FFF. They are defined in the include file bu5716.h.

6.1.29. bu5716_setChannelConf

```
ViStatus bu5716_setChannelConf (ViSession instrumentHandle,
                               ViInt16 channel, ViInt16 mode,
                               ViInt16 gain, ViInt16 coupling,
                               ViInt16 inputSource);
```

Purpose

This function configures the specified channel or all 16 channels.

Parameter List

instrumentHandle

Variable Type ViSession

The Instrument Handle is used to identify the unique session or communication channel between the driver and the instrument.

If more than one instrument of the same model type is used, this handle will be used to differentiate between them.

channel

Variable Type ViInt16

This parameter specifies which channel will be configured.

Possible values are:

1-16 (for ProDAQ 3416 master card) or
1-8 (for ProDAQ 3424 master card)

or

bu5716_CH_ALL 0 All available channels (16 for ProDAQ 3416
and 8 for ProDAQ 3424 master function card)
will be configured with the same parameters.

mode

Variable Type ViInt16

This parameter specifies the channel mode.

Possible values are:

bu5716_MODE_FB	0	Full-bridge configuration (default);
bu5716_MODE_HB	1	Half-bridge configuration;
bu5716_MODE_QB_120	2	Quarter-bridge configuration with 120 ohm completion resistors;
bu5716_MODE_QB_350	3	Quarter-bridge configuration with 350 ohm completion resistors;
bu5716_MODE_QB_USER	4	Quarter-bridge configuration with user selected plug-in completion resistors;
bu5716_MODE_ICP	5	Front-end configured to work with ICP sensors;
bu5716_MODE_DI	6	Differential voltage;
bu5716_MODE_SE	7	Single-ended voltage.

gain

Variable Type ViInt16

This parameter specifies the gain for the input channel.

Possible values are:

bu5716_GAIN_1	1	Gain 1
bu5716_GAIN_10	10	Gain 10
bu5716_GAIN_100	100	Gain 100
bu5716_GAIN_1000	1000	Gain 1000

coupling

Variable Type ViInt16

This parameter specifies the channel's PGA coupling mode.

Possible values are:

```
bu5716_COUP_AC    0    Both PGA inputs are AC coupled (default).
bu5716_COUP_DC    1    Both PGA inputs are DC coupled.
```

NOTE: If mode is set to ICP mode, this parameter will be ignored and coupling will be set to ICP coupling.

If mode is set to single-ended voltage, only positive site coupling is changed; negative is connected to ground.

inputSource

Variable Type ViInt16

This parameter specifies the channel's input source.

Possible values are:

```
bu5716_INP_SRC_FP    0    Front-panel (default)
bu5716_INP_SRC_VREF  1    Voltage Reference
```

Return Value

If the function was successful, it will return a status of VI_SUCCESS, otherwise it will return a warning or error code. Passing the status code to the function "bu5716_error_message" will return a string describing the warning or error.

A driver function can return three different types of warnings or errors. The function "bu5716_error_message" will handle all three types of warning/error codes by passing them to the appropriate function if necessary ("bu3100_error_message" or "viStatusDesc"), to return the correct warning/error message.

VISA Warnings/Errors:

See section 3.3 of the VPP 4.3.2 document for a complete list of VISA status codes and their values. The VPP 4.3 document contains detailed descriptions of all VISA functions and the status codes returned by each of them.

BU3100 Warnings/Errors:

These are warning or error codes returned by the common motherboard interface library, which are used by the 5716 driver to access a ProDAQ motherboard. Warnings returned by the library will be in the range 0x3FFC0800 to 0x3FFC0900 and errors in the range 0xBFFC0800 to 0xBFFC0900. They are defined in the include file bu3100.h.

BU5716 Warnings/Errors:

Warning codes returned by the 5716 driver functions will be in the range 0x3FFC0B00 to 0x3FFC0FFF and error codes in the range 0xBFFC0B00 to 0xBFFC0FFF. They are defined in the include file bu5716.h.

6.1.30. bu5716_setCoupling

```
ViStatus bu5716_setCoupling (ViSession instrumentHandle,
                             ViInt16 channel, ViInt16 coupling);
```

Purpose

This function sets the PGA coupling mode of the specified channel or all 16 channels.

Parameter List

instrumentHandle

Variable Type ViSession

The Instrument Handle is used to identify the unique session or communication channel between the driver and the instrument.

If more than one instrument of the same model type is used, this handle will be used to differentiate between them.

channel

Variable Type ViInt16

This parameter specifies which channel will be configured.

Possible values are:

1-16 (for ProDAQ 3416 master card) or
1-8 (for ProDAQ 3424 master card)

or

bu5716_CH_ALL 0 All available channels (16 for ProDAQ 3416
and 8 for ProDAQ 3424 master function card)
will be configured with the same parameters.

coupling

Variable Type ViInt16

This parameter specifies the channel's PGA coupling mode.

Possible values are:

bu5716_COUP_AC 0 Both PGA inputs are AC coupled (default).
bu5716_COUP_DC 1 Both PGA inputs are DC coupled.

NOTE: If mode is set to ICP mode, this parameter will be ignored and coupling will be set to ICP coupling.

If mode is set to single-ended voltage, only positive site coupling is changed; negative is connected to ground.

Return Value

If the function was successful, it will return a status of VI_SUCCESS, otherwise it will return a warning or error code. Passing the status code to the function "bu5716_error_message" will return a string describing the warning or error.

A driver function can return three different types of warnings or errors. The function "bu5716_error_message" will handle all three types of warning/error codes by passing them to the appropriate function if necessary ("bu3100_error_message" or "viStatusDesc"), to return the correct warning/error message.

VISA Warnings/Errors:

See section 3.3 of the VPP 4.3.2 document for a complete list of VISA status codes and their values. The VPP 4.3 document contains detailed descriptions of all VISA functions and the status codes returned by each of them.

BU3100 Warnings/Errors:

These are warning or error codes returned by the common motherboard interface library, which are used by the 5716 driver to access a ProDAQ motherboard. Warnings returned by the library will be in the range 0x3FFC0800 to 0x3FFC0900 and errors in the range 0xBFFC0800 to

0xBFFC0900. They are defined in the include file bu3100.h.

BU5716 Warnings/Errors:

Warning codes returned by the 5716 driver functions will be in the range 0x3FFC0B00 to 0x3FFC0FFF and error codes in the range 0xBFFC0B00 to 0xBFFC0FFF. They are defined in the include file bu5716.h.

6.1.31. bu5716_setDAC

```
ViStatus bu5716_setDAC (ViSession instrumentHandle, ViInt16 channel,
                       ViInt16 DAC, ViReal64 value);
```

Purpose

This function sets the output value of the specified DAC of one or all 16 channels.

Parameter List

instrumentHandle

Variable Type ViSession

The Instrument Handle is used to identify the unique session or communication channel between the driver and the instrument.

If more than one instrument of the same model type is used, this handle will be used to differentiate between them.

channel

Variable Type ViInt16

This parameter specifies which channel will be configured.

Possible values are:

1-16 (for ProDAQ 3416 master card) or
1-8 (for ProDAQ 3424 master card)

or

bu5716_CH_ALL 0 All available channels (16 for ProDAQ 3416
and 8 for ProDAQ 3424 master function card)
will be configured with the same parameters.

DAC

Variable Type ViInt16

This parameter specifies which DAC value will be set.

Possible values are:

bu5716_DAC_MAIN_POS	0	Main DAC positive pass
bu5716_DAC_MAIN_NEG	1	Main DAC negative pass
bu5716_DAC_AUTO_POS	2	Autobalance DAC positive pass
bu5716_DAC_AUTO_NEG	3	Autobalance DAC negative pass

value

Variable Type ViReal64

The value to be set for the specified DAC.

Possible values are:

For main DAC positive pass: 0 - 10V;
For main DAC negative pass: -10 - 0V;
For autobalance DAC positive pass: 0 - 10V;
For autobalance DAC negative pass: -10 - 0V.

Return Value

If the function was successful, it will return a status of VI_SUCCESS, otherwise it will return a warning or error code. Passing the status code to the function "bu5716_error_message" will return a string describing the warning or error.

A driver function can return three different types of warnings or errors. The function "bu5716_error_message" will handle all three types of warning/error codes by passing them to the appropriate function if necessary ("bu3100_error_message" or "viStatusDesc"), to return the correct warning/error message.

VISA Warnings/Errors:

See section 3.3 of the VPP 4.3.2 document for a complete list of VISA status codes and their values. The VPP 4.3 document contains detailed descriptions of all VISA functions and the status codes

returned by each of them.

BU3100 Warnings/Errors:

These are warning or error codes returned by the common motherboard interface library, which are used by the 5716 driver to access a ProDAQ motherboard. Warnings returned by the library will be in the range 0x3FFC0800 to 0x3FFC0900 and errors in the range 0xBFFC0800 to 0xBFFC0900. They are defined in the include file bu3100.h.

BU5716 Warnings/Errors:

Warning codes returned by the 5716 driver functions will be in the range 0x3FFC0B00 to 0x3FFC0FFF and error codes in the range 0xBFFC0B00 to 0xBFFC0FFF. They are defined in the include file bu5716.h.

6.1.32. bu5716_setExcitation

```
ViStatus bu5716_setExcitation (ViSession instrumentHandle,
                              ViInt16 channel,
                              ViReal64 positiveExcitationVoltage,
                              ViReal64 negativeExcitationVoltage);
```

Purpose

This function sets the excitation voltage for one or all channels.

Parameter List

instrumentHandle

Variable Type ViSession

The Instrument Handle is used to identify the unique session or communication channel between the driver and the instrument.

If more than one instrument of the same model type is used, this handle will be used to differentiate between them.

channel

Variable Type ViInt16

This parameter specifies on which channel the excitation voltage will be set.

Possible values are:

1-16 (for ProDAQ 3416 master card) or
1-8 (for ProDAQ 3424 master card)

or

bu5716_CH_ALL 0 All available channels (16 for ProDAQ 3416
and 8 for ProDAQ 3424 master function card)
will be configured with the same parameters.

positiveExcitationVoltage

Variable Type ViReal64

This parameter specifies the bridge positive, excitation voltage for the specified channel (or all channels).

Valid values are: 0.0 to +10.0 volts.

negativeExcitationVoltage

Variable Type ViReal64

This parameter specifies the bridge negative, excitation voltage for the specified channel (or all channels).

Valid values are: -10.0 to 0.0 volts.

Return Value

If the function was successful, it will return a status of VI_SUCCESS, otherwise it will return a warning or error code. Passing the status code to the function "bu5716_error_message" will return a string describing the warning or error.

A driver function can return three different types of warnings or errors. The function "bu5716_error_message" will handle all three types of warning/error codes by passing them to the appropriate function if necessary ("bu3100_error_message" or "viStatusDesc"), to return the correct warning/error message.

VISA Warnings/Errors:

See section 3.3 of the VPP 4.3.2 document for a complete list of VISA status codes and their values. The VPP 4.3 document contains detailed descriptions of all VISA functions and the status codes returned by each of them.

BU3100 Warnings/Errors:

These are warning or error codes returned by the common motherboard interface library, which are used by the 5716 driver to access a ProDAQ motherboard. Warnings returned by the library will be in the range 0x3FFC0800 to 0x3FFC0900 and errors in the range 0xBFFC0800 to 0xBFFC0900. They are defined in the include file bu3100.h.

BU5716 Warnings/Errors:

Warning codes returned by the 5716 driver functions will be in the range 0x3FFC0B00 to 0x3FFC0FFF and error codes in the range 0xBFFC0B00 to 0xBFFC0FFF. They are defined in the include file bu5716.h.

6.1.33. bu5716_setExcitationMonitor

```
ViStatus bu5716_setExcitationMonitor (ViSession instrumentHandle,
                                     ViInt16 mode);
```

Purpose

This function connects/disconnects all channels to/from the excitation voltage source.

Parameter List

instrumentHandle

Variable Type ViSession

The Instrument Handle is used to identify the unique session or communication channel between the driver and the instrument.

If more than one instrument of the same model type is used, this handle will be used to differentiate between them.

mode

Variable Type ViInt16

This parameter specifies which excitation (EX) voltage will be monitored.

Possible values are:

bu5716_EXCMON_OFF	0	All channel inputs are connected to the front-panel or voltage reference;
bu5716_EXCMON_P	1	All channel positive inputs are connected to EX+ signal and negative inputs are connected to GND;
bu5716_EXCMON_N	2	All channel negative inputs are connected to EX- signal and positive inputs are connected to GND;
bu5716_EXCMON_PN	3	All channel positive inputs are connected to EX+ signal and negative inputs are connected to EX- signal.

Return Value

If the function was successful, it will return a status of VI_SUCCESS, otherwise it will return a warning or error code. Passing the status code to the function "bu5716_error_message" will return a string describing the warning or error.

A driver function can return three different types of warnings or errors. The function "bu5716_error_message" will handle all three types of warning/error codes by passing them to the appropriate function if necessary ("bu3100_error_message" or "viStatusDesc"), to return the correct warning/error message.

VISA Warnings/Errors:

See section 3.3 of the VPP 4.3.2 document for a complete list of VISA status codes and their values. The VPP 4.3 document contains detailed descriptions of all VISA functions and the status codes returned by each of them.

BU3100 Warnings/Errors:

These are warning or error codes returned by the common motherboard interface library, which are used by the 5716 driver to access a ProDAQ motherboard. Warnings returned by the library will be in the range 0x3FFC0800 to 0x3FFC0900 and errors in the range 0xBFFC0800 to 0xBFFC0900. They are defined in the include file bu3100.h.

BU5716 Warnings/Errors:

Warning codes returned by the 5716 driver functions will be in the range 0x3FFC0B00 to 0x3FFC0FFF and error codes in the range 0xBFFC0B00 to 0xBFFC0FFF. They are defined in the include file bu5716.h.

6.1.34. bu5716_setGain

```
ViStatus bu5716_setGain (ViSession instrumentHandle, ViInt16 channel,
                        ViInt16 gain);
```

Purpose

This function sets the gain of PGA for the specified channel or all 16 channels.

Parameter List

instrumentHandle

Variable Type ViSession

The Instrument Handle is used to identify the unique session or communication channel between the driver and the instrument.

If more than one instrument of the same model type is used, this handle will be used to differentiate between them.

channel

Variable Type ViInt16

This parameter specifies which channel will be configured.

Possible values are:

1-16 (for ProDAQ 3416 master card) or
1-8 (for ProDAQ 3424 master card)

or

bu5716_CH_ALL 0 All available channels (16 for ProDAQ 3416
and 8 for ProDAQ 3424 master function card)
will be configured with the same parameters.

gain

Variable Type ViInt16

This parameter specifies the channel's PGA gain.

Possible values are:

bu5716_GAIN_1 1 Gain 1
bu5716_GAIN_10 10 Gain 10
bu5716_GAIN_100 100 Gain 100
bu5716_GAIN_1000 1000 Gain 1000

Return Value

If the function was successful, it will return a status of VI_SUCCESS, otherwise it will return a warning or error code. Passing the status code to the function "bu5716_error_message" will return a string describing the warning or error. A driver function can return three different types of warnings or errors. The function "bu5716_error_message" will handle all three types of warning/error codes by passing them to the appropriate function if necessary ("bu3100_error_message" or "viStatusDesc"), to return the correct warning/error message.

VISA Warnings/Errors:

See section 3.3 of the VPP 4.3.2 document for a complete list of VISA status codes and their values. The VPP 4.3 document contains detailed descriptions of all VISA functions and the status codes returned by each of them.

BU3100 Warnings/Errors:

These are warning or error codes returned by the common motherboard interface library, which are used by the 5716 driver to access a ProDAQ motherboard. Warnings returned by the library will be in the range 0x3FFC0800 to 0x3FFC0900 and errors in the range 0xBFFC0800 to 0xBFFC0900. They are defined in the include file bu3100.h.

BU5716 Warnings/Errors:

Warning codes returned by the 5716 driver functions will be in the range 0x3FFC0B00 to 0x3FFC0FFF and error codes in the range 0xBFFC0B00 to 0xBFFC0FFF. They are defined in the include file bu5716.h.

6.1.35. bu5716_setInputSrc

```
ViStatus bu5716_setInputSrc (ViSession instrumentHandle,
                             ViInt16 channel, ViInt16 inputSource);
```

Purpose

This function sets the input source for the specified channel or all 16 channels.

Parameter List

instrumentHandle

Variable Type ViSession

The Instrument Handle is used to identify the unique session or communication channel between the driver and the instrument.

If more than one instrument of the same model type is used, this handle will be used to differentiate between them.

channel

Variable Type ViInt16

This parameter specifies which channel will be configured.

Possible values are:

1-16 (for ProDAQ 3416 master card) or
1-8 (for ProDAQ 3424 master card)

or

bu5716_CH_ALL 0 All available channels (16 for ProDAQ 3416
and 8 for ProDAQ 3424 master function card)
will be configured with the same parameters.

inputSource

Variable Type ViInt16

This parameter specifies the channel's input source.

Possible values are:

bu5716_INP_SRC_FP 0 Front-panel (default)
bu5716_INP_SRC_VREF 1 Voltage Reference

Return Value

If the function was successful, it will return a status of VI_SUCCESS, otherwise it will return a warning or error code. Passing the status code to the function "bu5716_error_message" will return a string describing the warning or error.

A driver function can return three different types of warnings or errors. The function "bu5716_error_message" will handle all three types of warning/error codes by passing them to the appropriate function if necessary ("bu3100_error_message" or "viStatusDesc"), to return the correct warning/error message.

VISA Warnings/Errors:

See section 3.3 of the VPP 4.3.2 document for a complete list of VISA status codes and their values. The VPP 4.3 document contains detailed descriptions of all VISA functions and the status codes returned by each of them.

BU3100 Warnings/Errors:

These are warning or error codes returned by the common motherboard interface library, which are used by the 5716 driver to access a ProDAQ motherboard. Warnings returned by the library will be in the range 0x3FFC0800 to 0x3FFC0900 and errors in the range 0xBFFC0800 to 0xBFFC0900. They are defined in the include file bu3100.h.

BU5716 Warnings/Errors:

Warning codes returned by the 5716 driver functions will be in the range 0x3FFC0B00 to 0x3FFC0FFF and error codes in the range 0xBFFC0B00 to 0xBFFC0FFF. They are defined in the include file bu5716.h.

6.1.36. bu5716_setMode

```
ViStatus bu5716_setMode (ViSession instrumentHandle, ViInt16 channel,
                        ViInt16 mode);
```

Purpose

This function sets the mode of the specified channel or all 16 channels.

Parameter List

instrumentHandle

Variable Type ViSession

The Instrument Handle is used to identify the unique session or communication channel between the driver and the instrument.

If more than one instrument of the same model type is used, this handle will be used to differentiate between them.

channel

Variable Type ViInt16

This parameter specifies which channel will be configured.

Possible values are:

1-16 (for ProDAQ 3416 master card) or
1-8 (for ProDAQ 3424 master card)

or

bu5716_CH_ALL 0 All available channels (16 for ProDAQ 3416
and 8 for ProDAQ 3424 master function card)
will be configured with the same parameters.

mode

Variable Type ViInt16

This parameter specifies the channel mode.

Possible values are:

bu5716_MODE_FB	0	Full-bridge configuration (default);
bu5716_MODE_HB	1	Half-bridge configuration, sensing connected;
bu5716_MODE_QB_120	2	Quarter-bridge configuration with 120 ohm completion resistors;
bu5716_MODE_QB_350	3	Quarter-bridge configuration with 350 ohm completion resistors;
bu5716_MODE_QB_USER	4	Quarter-bridge configuration with user selected plug-in completion resistors;
bu5716_MODE_ICP	5	Front-end configured to work with ICP sensors;
bu5716_MODE_DI	6	Differential voltage;
bu5716_MODE_SE	7	Single-ended voltage.

Return Value

If the function was successful, it will return a status of VI_SUCCESS, otherwise it will return a warning or error code. Passing the status code to the function "bu5716_error_message" will return a string describing the warning or error.

A driver function can return three different types of warnings or errors. The function "bu5716_error_message" will handle all three types of warning/error codes by passing them to the appropriate function if necessary ("bu3100_error_message" or "viStatusDesc"), to return the correct warning/error message.

VISA Warnings/Errors:

See section 3.3 of the VPP 4.3.2 document for a complete list of VISA status codes and their values. The VPP 4.3 document contains detailed descriptions of all VISA functions and the status codes returned by each of them.

BU3100 Warnings/Errors:

These are warning or error codes returned by the common motherboard interface library, which are used by the 5716 driver to access a ProDAQ motherboard. Warnings returned by the library will be in the range 0x3FFC0800 to 0x3FFC0900 and errors in the range 0xBFFC0800 to 0xBFFC0900. They are defined in the include file bu3100.h.

BU5716 Warnings/Errors:

Warning codes returned by the 5716 driver functions will be in the range 0x3FFC0B00 to 0x3FFC0FFF and error codes in the range 0xBFFC0B00 to 0xBFFC0FFF. They are defined in the include file bu5716.h.

6.1.37. bu5716_setScuErrorMask

```
ViStatus bu5716_setScuErrorMask (ViSession instrumentHandle,
                                 ViInt16 icpMask, ViInt16 exlMask);
```

Purpose

This function sets the SCU_ERR line source.

Parameter List

instrumentHandle

Variable Type ViSession

The Instrument Handle is used to identify the unique session or communication channel between the driver and the instrument.

If more than one instrument of the same model type is used, this handle will be used to differentiate between them.

icpMask

Variable Type ViInt16

This parameter is a bitmask of the channels which will affect the SCU_ERR line when an ICP error occurs.

Each bit represents the following channel:

```
Bit0 - channel 1
Bit1 - channel 2
...
Bit15 - channel 15
```

Possible values:

```
0 - Disabled as a source of SCU_ERR line
1 - Enabled as a source of SCU_ERR line
```

NOTE: When the device is controlled by the ProDAQ 3424, only the eight lowest bits are used.

exlMask

Variable Type ViInt16

This parameter is a bitmask of the channels which will affect the SCU_ERR line when an excitation error occurs.

Each bit represents the following channel:

```
Bit0 - channel 1
Bit1 - channel 2
...
Bit15 - channel 15
```

Possible values:

```
0 - Disabled as a source of SCU_ERR line
1 - Enabled as a source of SCU_ERR line
```

NOTE: When the device is controlled by the ProDAQ 3424, only the eight lowest bits are used.

Return Value

If the function was successful, it will return a status of VI_SUCCESS, otherwise it will return a warning or error code. Passing the status code to the function "bu5716_error_message" will return a string describing the warning or error.

A driver function can return three different types of warnings or errors. The function "bu5716_error_message" will handle all three types of warning/error codes by passing them to the appropriate function if necessary ("bu3100_error_message" or "viStatusDesc"), to return the correct warning/error message.

VISA Warnings/Errors:

See section 3.3 of the VPP 4.3.2 document for a complete list of VISA status codes and their values. The VPP 4.3 document contains detailed descriptions of all VISA functions and the status codes returned by each of them.

BU3100 Warnings/Errors:

These are warning or error codes returned by the common motherboard interface library, which are used by the 5716 driver to access a ProDAQ motherboard. Warnings returned by the library will be in the range 0x3FFC0800 to 0x3FFC0900 and errors in the range 0xBFFC0800 to 0xBFFC0900. They are defined in the include file bu3100.h.

BU5716 Warnings/Errors:

Warning codes returned by the 5716 driver functions will be in the range 0x3FFC0B00 to 0x3FFC0FFF and error codes in the range 0xBFFC0B00 to 0xBFFC0FFF. They are defined in the include file bu5716.h.

bu5716_setShuntCal

```
ViStatus bu5716_setShuntCal (ViSession instrumentHandle,
                             ViInt16 channel, ViBoolean connect);
```

Purpose

This function connects/disconnects the calibration shunt resistor of the specified DAC of one or all 16 channels.

Parameter List

instrumentHandle

Variable Type ViSession

The Instrument Handle is used to identify the unique session or communication channel between the driver and the instrument.

If more than one instrument of the same model type is used, this handle will be used to differentiate between them.

channel

Variable Type ViInt16

This parameter specifies which channel will be configured.

Possible values are:

1-16 (for ProDAQ 3416 master card) or
1-8 (for ProDAQ 3424 master card)

or

bu5716_CH_ALL 0 All available channels (16 for ProDAQ 3416
and 8 for ProDAQ 3424 master function card)
will be configured with the same parameters.

connect

Variable Type ViBoolean

This parameter specifies whether the calibration shunt is connected or disconnected.

Possible values are:

VI_FALSE 0 The calibration shunt is disconnected.
VI_TRUE 1 The calibration shunt is connected.

Return Value

If the function was successful, it will return a status of VI_SUCCESS, otherwise it will return a warning or error code. Passing the status code to the function "bu5716_error_message" will return a string describing the warning or error.

A driver function can return three different types of warnings or errors. The function "bu5716_error_message" will handle all three types of warning/error codes by passing them to the appropriate function if necessary ("bu3100_error_message" or "viStatusDesc"), to return the correct warning/error message.

VISA Warnings/Errors:

See section 3.3 of the VPP 4.3.2 document for a complete list of VISA status codes and their values. The VPP 4.3 document contains detailed descriptions of all VISA functions and the status codes returned by each of them.

BU3100 Warnings/Errors:

These are warning or error codes returned by the common motherboard interface library, which are used by the 5716 driver to access a ProDAQ motherboard. Warnings returned by the library will be in the range 0x3FFC0800 to 0x3FFC0900 and errors in the range 0xBFFC0800 to 0xBFFC0900. They are defined in the include file bu3100.h.

BU5716 Warnings/Errors:

Warning codes returned by the 5716 driver functions will be in the range 0x3FFC0B00 to 0x3FFC0FFF and error codes in the range 0xBFFC0B00 to 0xBFFC0FFF. They are defined in the include file bu5716.h.

6.1.38. bu5716_setVoltRefOutput

```
ViStatus bu5716_setVoltRefOutput (ViSession instrumentHandle,
                                  ViInt32 voltRefModule,
                                  ViReal64 voltage, ViBoolean monitor);
```

Purpose

This function sets the specified voltage reference module output voltage.

Parameter List

instrumentHandle

Variable Type ViSession

The Instrument Handle is used to identify the unique session or communication channel between the driver and the instrument.

If more than one instrument of the same model type is used, this handle will be used to differentiate between them.

voltRefModule

Variable Type ViInt32

This parameter specifies which voltage reference module will be used to generate the voltage reference.

Possible values are:

bu5716_VREF_SRC_OFF	0	All voltage reference modules are disconnected;
bu5716_VREF_SRC_FC	1	Voltage reference module located on the motherboard module of the master function card;
bu5716_VREF_SRC_ONB	2	With this option, only connection to ground is possible. Voltage parameter must be set to 0.0;
bu5716_VREF_SRC_EXT	3	External voltage reference is used.

WARNING:

Please always make sure that if you use external voltage reference, to disconnect it or set Monitor parameter to VI_FALSE when changing the voltage reference source.

voltage

Variable Type ViReal64

This parameter specifies the output voltage of the voltage reference module. The available set of voltages depend on the type of the selected voltage reference module.

If the voltage reference module is unable to generate the requested message, the function will return an error status. If the external voltage reference module is used, this parameter will contain the actual voltage provided by the external voltage reference source.

monitor

Variable Type ViBoolean

This parameter specifies whether the voltage reference output will be routed to the ProDAQ 5716 rear panel connector (for monitoring purposes). This parameter will be ignored if the external voltage reference is selected as a source.

Possible values are:

VI_FALSE	Voltage reference output is disconnected from the ProDAQ 5716 rear panel connector.
VI_TRUE	Voltage reference output is connected to the ProDAQ 5716 rear panel connector.

WARNING:

Please always make sure when using monitor, that the power source or voltage reference is NOT connected to the rear panel connector.

Return Value

If the function was successful, it will return a status of `VI_SUCCESS`, otherwise it will return a warning or error code. Passing the status code to the function `"bu5716_error_message"` will return a string describing the warning or error.

A driver function can return three different types of warnings or errors. The function `"bu5716_error_message"` will handle all three types of warning/error codes by passing them to the appropriate function if necessary (`"bu3100_error_message"` or `"viStatusDesc"`), to return the correct warning/error message.

VISA Warnings/Errors:

See section 3.3 of the VPP 4.3.2 document for a complete list of VISA status codes and their values. The VPP 4.3 document contains detailed descriptions of all VISA functions and the status codes returned by each of them.

BU3100 Warnings/Errors:

These are warning or error codes returned by the common motherboard interface library, which are used by the 5716 driver to access a ProDAQ motherboard. Warnings returned by the library will be in the range `0x3FFC0800` to `0x3FFC0900` and errors in the range `0xBFFC0800` to `0xBFFC0900`. They are defined in the include file `bu3100.h`.

BU5716 Warnings/Errors:

Warning codes returned by the 5716 driver functions will be in the range `0x3FFC0B00` to `0x3FFC0FFF` and error codes in the range `0xBFFC0B00` to `0xBFFC0FFF`. They are defined in the include file `bu5716.h`.

6.1.39. bu5716_storeCalibCoeff

```
ViStatus bu5716_storeCalibCoeff (ViSession instrumentHandle,
                                ViInt16 type);
```

Purpose

This function stores the calibration coefficients into the onboard EEPROM memory.

CAUTION: The factory calibration coefficients will be overwritten.

Parameter List

instrumentHandle

Variable Type ViSession

The Instrument Handle is used to identify the unique session or communication channel between the driver and the instrument.

If more than one instrument of the same model type is used, this handle will be used to differentiate between them.

type

Variable Type ViInt16

Selects which type of calibration coefficient will be stored into EEPROM.

Possible values are:

bu5716_PGA_CALIB_DATA	1	PGA calibration coefficient;
bu5716_EXC_CALIB_DATA	2	Excitation calibration coefficient;
bu5716_ALL_CALIB_DATA	3	PGA & Excitation calibration coefficient.

Return Value

If the function was successful, it will return a status of VI_SUCCESS, otherwise it will return a warning or error code. Passing the status code to the function "bu5716_error_message" will return a string describing the warning or error.

A driver function can return three different types of warnings or errors. The function "bu5716_error_message" will handle all three types of warning/error codes by passing them to the appropriate function if necessary ("bu3100_error_message" or "viStatusDesc"), to return the correct warning/error message.

VISA Warnings/Errors:

See section 3.3 of the VPP 4.3.2 document for a complete list of VISA status codes and their values. The VPP 4.3 document contains detailed descriptions of all VISA functions and the status codes returned by each of them.

BU3100 Warnings/Errors:

These are warning or error codes returned by the common motherboard interface library, which are used by the 5716 driver to access a ProDAQ motherboard. Warnings returned by the library will be in the range 0x3FFC0800 to 0x3FFC0900 and errors in the range 0xBFFC0800 to 0xBFFC0900. They are defined in the include file bu3100.h.

BU5716 Warnings/Errors:

Warning codes returned by the 5716 driver functions will be in the range 0x3FFC0B00 to 0x3FFC0FFF and error codes in the range 0xBFFC0B00 to 0xBFFC0FFF. They are defined in the include file bu5716.h.

6.1.40. bu5716_writeTEDS

```
ViStatus bu5716_writeTEDS (ViSession instrumentHandle, ViInt16 channel,
                          ViInt16 TEDS, ViInt16 address, ViInt16 count,
                          ViChar _VI_FAR data[]);
```

Purpose

This function writes the block of data to the TEDS EEPROM memory.

Parameter List

instrumentHandle

Variable Type ViSession

The Instrument Handle is used to identify the unique session or communication channel between the driver and the instrument.

If more than one instrument of the same model type is used, this handle will be used to differentiate between them.

channel

Variable Type ViInt16

This parameter specifies the channel number from which the TEDS sensor will be read.

Possible values are:

1-16 (for ProDAQ 3416 master card) or
1-8 (for ProDAQ 3424 master card).

TEDS

Variable Type ViInt16

Specifies which type of TEDS will be written to.

Possible values are:

bu5716_TEDS_CLASS_1	0	TEDS Class 1 will be accessed.
bu5716_TEDS_CLASS_2	1	TEDS Class 2 will be accessed.

address

Variable Type ViInt16

Specifies the starting address in the TEDS EEPROM memory where the data should be written to.

Valid values range from 0 to 31.

The sum of parameters Address and Count should not exceed the value 32.

count

Variable Type ViInt16

Specifies the number of bytes to be written to the TEDS EEPROM memory.

Valid values range from 1 to 32.

The sum of parameters Address and Count should not exceed the value 32.

data

Variable Type ViChar[]

This buffer should contain [Count] number of bytes of data to be written to the TEDS EEPROM memory.

Return Value

If the function was successful, it will return a status of VI_SUCCESS, otherwise it will return a warning or error code.

Passing the status code to the function "bu5716_error_message" will return a string describing the warning or error.

A driver function can return three different types of warnings or errors. The function "bu5716_error_message" will handle all three types of warning/error codes by passing them to the appropriate function if necessary ("bu3100_error_message" or "viStatusDesc"), to return the correct warning/error message.

VISA Warnings/Errors:

See section 3.3 of the VPP 4.3.2 document for a complete list of VISA status codes and their values. The VPP 4.3 document contains detailed descriptions of all VISA functions and the status codes returned by each of them.

BU3100 Warnings/Errors:

These are warning or error codes returned by the common motherboard interface library, which are used by the 5716 driver to access a ProDAQ motherboard. Warnings returned by the library will be in the range 0x3FFC0800 to 0x3FFC0900 and errors in the range 0xBFFC0800 to 0xBFFC0900. They are defined in the include file bu3100.h.

BU5716 Warnings/Errors:

Warning codes returned by the 5716 driver functions will be in the range 0x3FFC0B00 to 0x3FFC0FFF and error codes in the range 0xBFFC0B00 to 0xBFFC0FFF. They are defined in the include file bu5716.h.

7. Specifications

7.1. Available Versions

Versions	5716-AA	1U, 19" rack-mount Bridge Signal Conditioning Unit, IEPE and TEDS support, LEMO connectors
	5716-BA	1U, 19" rack-mount Bridge Signal Conditioning Unit, IEPE and TEDS support, RJ45 connectors

7.2. Signal Conditioning

Sensor Types	Strain Gages (resistive bridges), IEPE, Voltage
Number of channels	16 (1 x 16 or 2 x 8)
Bridge Configurations	Full-Bridge, Half-Bridge, Quarter-Bridge
Bridge Completion	120 Ω , 350 Ω , Custom (User installable)
Excitation Type	Constant Voltage
Excitation Range	-10V to 0V (Negative Supply), 0V to +10V (Positive Supply)
Excitation Current	max. 50 mA per channel
Excitation Resolution	16-bit (152 μ V)
Excitation Accuracy	0.05% max
Sensing	Remote or Internal
Bridge Balancing	Automatic
Shunt Calibration	50 k Ω , 0.1% per channel
Input Signal Type	Differential or Single-ended
Gain	1, 10, 100, 1000 programmable per channel
Input Resistance	10 M Ω
Signal Path Coupling	AC or DC, programmable per channel
Input Signal Full Power Bandwidth	Gain 1, 10, 100: >200 kHz Gain 1000: 60 kHz
Common Mode Filter	Trifilar Choke
IEPE (ICP™) Support	Yes
IEPE (ICP™) Current	4.7 mA typ.
TEDS Support	Class 1 and 2

7.3. Power Supply

Input Voltage Range	90-264 V AC, 50-60 Hz
Power	90 W max.

7.4. Environmental Specifications

Temperature	0 °C to +50 °C (operational) -40 °C to +70 °C (storage only)
Humidity	5% - 95% (non-condensing)
Warm-up Time	30 min minimum.

(This page was intentionally left blank)



Bustec Production, Ltd.
Bustec House, Shannon Business Park, Shannon, Co. Clare, Ireland
Tel: +353 (0) 61 707100, FAX: +353 (0) 61 707106

Bustec, Inc.
6137W 10930N, Highland, Utah 84003, USA
Tel: +1 (385) 207 8232, Fax: +1 (208) 485 2543