USER MANUAL

ProDAQ Data Acquisition Function Cards

ProDAQ 3510 16-Channel, 16-Bit, DAC Function Card

PUBLICATION NUMBER: 3510-XX-UM

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. reserves the right to change both the hardware and software described herein.

WARRANTY STATEMENT

Products sold by Bustec Production Ltd. are warranted to be free from defects in workmanship or materials. Bustec Production Ltd. at its discretion will either repair or replace any hardware products that prove to be defective during the warranty period. You are a valued customer. Our mission is to make any necessary repairs in a reliable and timely manner.

Duration of Warranty

The warranty period for Bustec Production Ltd. hardware is one year. Software and firmware products designed for use with this hardware are warranted not to fail to execute their programming instructions due to defect in materials or workmanship for a period of ninety days from the date of delivery to the initial end user.

Return of Product

Authorization is required from Bustec Production Ltd. before you send us your product for service or calibration. Call Bustec Production Ltd. support facility. If you are unsure where to call, contact Bustec Production Ltd. at 353-61-707100. We can also be reached at www.bustec.com.

Limitation of Warranty

Bustec Production Ltd. shall be released from all obligations under this warranty in the event repairs or modifications are made by persons other than authorized Bustec Production Ltd. service personnel or without the written consent of Bustec Production Ltd.

Bustec Production Ltd. expressly disclaims any liability to its customers, dealers and representatives and to users of its product. Also to any other person or persons, for special or consequential damages of any kind and from any cause whatsoever arising out of or in any way connected with the manufacture, sale, handling, repair, maintenance, replacement or use of said products.

Representations and warranties made by any person including dealers and representatives of Bustec Production Ltd., which are inconsistent or in conflict with the terms of this warranty (including but not limited to the limitations of the liability of Bustec Production Ltd. as set forth above), shall not be binding upon Bustec Production Ltd. unless reduced to writing and approved by an officer of Bustec Production Ltd.

Except as stated above, Bustec Production Ltd. makes no warranty, express or implied (either in fact or by operation of law), statutory or otherwise; and except to the extent stated above. Bustec Production Ltd. shall have no liability under any warranty, express or implied (either in fact or by operation of law), statutory or otherwise.

FOR YOUR SAFETY

Before undertaking any troubleshooting, maintenance or exploratory procedure, read carefully the **WARNINGS** and **CAUTION** notices.

This equipment contains voltage hazardous to human life and safety, and is capable of inflicting personal injury.

If this instrument is to be powered from the AC line (mains) through an autotransformer, ensure the common connector is connected to the neutral (earth pole) of the power supply.

Before operating the unit, ensure the conductor (green wire) is connected to the ground (earth) conductor of the power outlet. Do not use a two-conductor extension cord or a three-prong/two-prong adapter. This will defeat the protective feature of the third conductor in the power cord.

Maintenance and calibration procedures sometimes call for operation of the unit with power applied and protective covers removed. Read the procedures and heed warnings to avoid “live” circuit points.

Before operating this instrument:

1. Ensure the instrument is configured to operate on the voltage at the power source. See Installation Section.
2. Ensure the proper fuse is in place for the power source to operate.
3. Ensure all other devices connected to or in proximity to this instrument are properly grounded or connected to the protective third-wire earth ground.

If the instrument:

- fails to operate satisfactorily
- shows visible damage
- has been stored under unfavourable conditions
- has sustained stress

Do not attempt to operate until qualified personnel checks its performance.

TABLE OF CONTENTS

1. INTRODUCTION 5

1.1 GENERAL 5
1.2 OPERATION 5
1.3 TRIGGER OUTPUTS 6
1.4 FRONT PANEL CONNECTOR 7

2. INSTALLATION INSTRUCTIONS 8

2.1 UNPACKING AND INSPECTION 8
2.2 RESHIPMENT INSTRUCTIONS 8
2.3 INSTALLATION 9
2.4 REMOVAL 10

3. THEORY OF OPERATION 11

3.1 HARDWARE 11
3.2 OPERATION 12
3.3 CONVERSION 12
3.4 REGISTERS 13

4. SOFTWARE UTILITIES 18

4.1 INTRODUCTION 18
4.2 USER INTERFACE AND INSTALLATION 18
4.2.1 SOFTWARE INSTALLATION 18
4.2.2 SOFTWARE UTILIZATION 18
4.3 PROGRAMMING CONCEPTS 21
4.3.1 INSTRUMENT DRIVER OVERVIEW 21
4.3.2 ERROR/STATUS INFORMATION 22
4.3.3 CONNECTING TO THE INSTRUMENT 22
4.3.4 PROGRAMMING INSTRUMENT 23

1. Introduction

1.1 General

The ProDAQ 3510 Function Card is one of a range of function cards designed to provide full functionality when installed in one of the range of ProDAQ motherboard modules such as the model 3120.

This high-density function card contains 16 fully independent 16bit channels with each having a digital to analog converter specifically designed for DC and low frequency applications. There are 4 versions available providing up to 16 channels in a number of different output voltages, current or combinational formats. Each voltage output channel is provided with a standard filter (100Hz).

All outputs in voltage mode have continuous short circuit protection. Current mode outputs all have over voltage protection. If a high output current is used on multiple channels of a DAC function card, this may restrict the number of 3510 cards that can be fitted into a 3120 module.

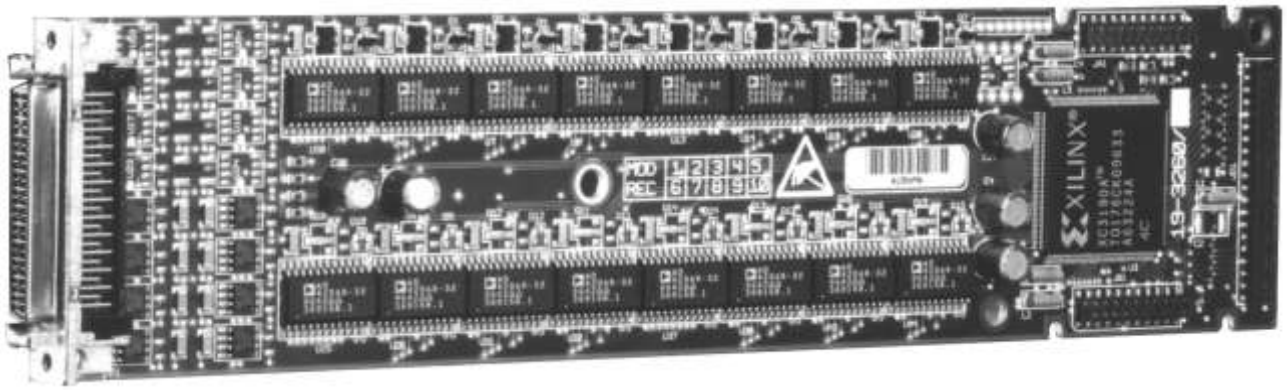


Figure 1: The ProDAQ 16 Channel, 16-bit Digital to Analog Converter

Modules are supplied with VXI*plug&play* drivers conforming to the WiN95 or WINNT standards. ProDAQ modules are able to safely and simply expand your existing hardware systems because they are compatible with all the current popular software environments available. The supplied driver automatically detects and software configures all of the cards fitted within a 3120 ProDAQ module, simplifying system configuration.

1.2 Operation

This function card is designed as an option to the ProDAQ range of data acquisition VXIbus motherboards. Function cards have no functionality without the motherboard.

Each function card occupies an address range of 64k 16bit words aligned to a 32bit VME address space boundary.

This card functions as a digital to analog output unit when fitted within a ProDAQ VXIbus module. It is designed for slow process control with a maximum speed of 100Hz per channel.

The card is factory configured as either all voltage or current or 8 voltage and 8 current outputs to provide 16 outputs (in 2 groups of 8) with a range of up to +10V, +16V or current output up to 24mA.

The DAC is programmed, by software, to deliver the correct output of 0 to 24mA, 4 to 24mA, 0 to 20mA or voltage. For flexibility the DACs are grouped into two groups of eight channels.

The outputs are protected against damage by protection resistors for current outputs or by a buffer output stage in the case of the voltage mode. A 2-pole output filter is fitted as standard to the voltage channels with characteristics set by the selection of factory installed components on the PCB.

The circuitry consists of 16 DAC chips with some active and passive support circuitry. The DAC is serially controlled by an FPGA. The DACs are logically organized in two groups:

G1 DAC's 1, 2, 3, 4, 5, 6, 7 and 8

G2 DAC's 9, 10, 11, 12, 13, 14, 15 and 16

A clock signal, serial data and a load signal control the DAC's data input. The output operation modes (voltage or the different current outputs) are defined by two static signals. In addition, a clear line is used to set the output to the lower boundary of the selected range.

Failures in the output are detected by the DAC and signaled to the FPGA, where they can be read by software. Depending on the programming of the output trigger, the Logical-OR of all failures can be sent to the front panel output.

1.3 Trigger Outputs

An output trigger signal (to the front panel, to the motherboard or both) may be selected from:

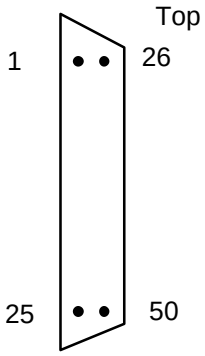
- Software; i.e. write a bit into a control and status register.
- A constant frequency generated by dividing the system clock with a programmable constant.
- The trigger input from the motherboard.
- Alarm output for any DAC detected a failure (current version only - no current output).

Note:

- Only one trigger source at a time can be active.
- The polarity of the trigger signal is programmable.
- The front panel output is an open collector TTL type, pulled high via a resistor.

1.4 Front Panel Connector

The front panel is fitted with a high-density female 50pin SCSI type connector. Pin assignment of the 50pin SCSI connector is as follows:

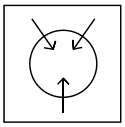


Signal	Row A	Row B	Signal
NC	1	26	Trigger out
GND	2	27	GND
GND	3	28	GND
GND	4	29	GND
GND	5	30	GND
GND	6	31	GND
GND	7	32	GND
GND	8	33	GND
GND	9	34	GND
Out 16	10	35	GND
Out 15	11	36	GND
Out 14	12	37	GND
Out 13	13	38	GND
Out 12	14	39	GND
Out 11	15	40	GND
Out 10	16	41	GND
Out 9	17	42	GND
Out 8	18	43	GND
Out 7	19	44	GND
Out 6	20	45	GND
Out 5	21	46	GND
Out 4	22	47	GND
Out 3	23	48	GND
Out 2	24	49	GND
Out 1	25	50	GND

2. Installation instructions

2.1 Unpacking and Inspection

1. Before unpacking the ProDAQ function card, check the exterior of the shipping carton for any signs of damage. All irregularities should be noted on the shipping bill.
2. Remove the function card from its carton, preserving the factory packaging as much as possible.
3. Inspect the function card for any defect or damage. Immediately notify the carrier if any damage is apparent.
4. Have a qualified person check the instrument for safety before use.



NOTE:

Proper ESD handling procedures must always be used when packing, unpacking, or installing any function card. Failure to do so may cause damage to the unit.

2.2 Reshipment Instructions

1. Use the original packing material when returning the function card to Bustec Production Ltd. for servicing. The original shipping carton and the instrument's plastic foam will provide the necessary support for safe reshipment.
2. If the original packing material is unavailable, wrap the function card in anti- static plastic sheeting and use plastic spray foam to surround and protect the instrument.
3. Reship in either the original or a new shipping carton.

2.3 Installation

Installing and removing the particular Function Card requires use of an extraction tool. It is used to help align the card and the pins on the motherboard.

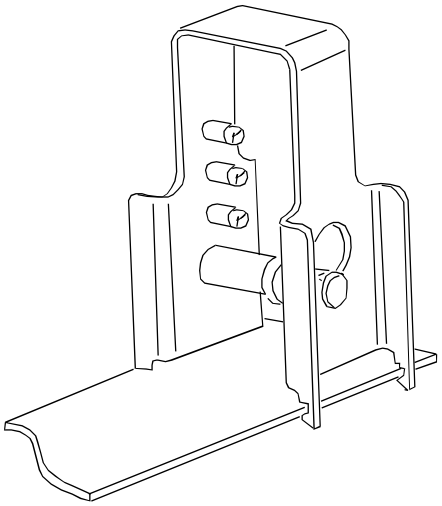


Figure 2: The Extraction Tool

The individual Function Card has four small cutouts (two per side) on the rear portion of the card. The extraction tool fits into these cutouts.

CAUTION

There are three places (40 pin and two 22 pin connectors) on the motherboard where the Function Card must be plugged in. These pins may bend or break when inserting the Function Card if it is not aligned properly.

The Function Card is placed into the slot with the connector (50pin SCSI connector) facing the front of the module.

Then, align the back edge (using the extraction tool) with the pins on the motherboard and gently press the Function Card down onto the pins.

There are two screws and two washers that go through the front panel and lock the front of the Function Card. Additionally, there are three other screws and two washers per screw that go on the top of the card and lock it down.

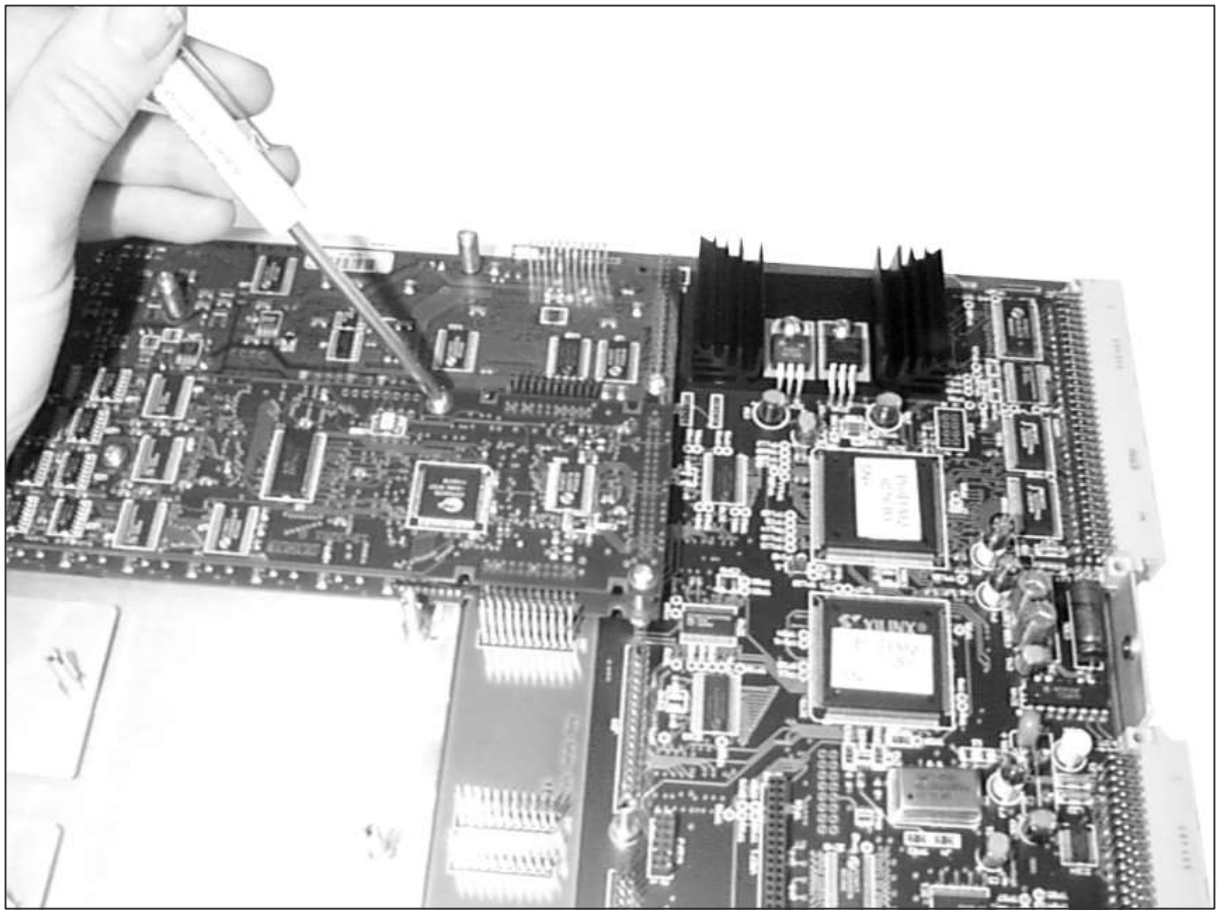


Figure 3: Locking Down A Screw

This procedure is for installing a Function Card that is to be mounted in either slot 2, slot 4, slot 6, or slot 8. For installing a function card in slots 1, 3, 5, or 7, the procedure is the same except that the screws used to fasten it to the chassis are studs (with a male screw on one end and a female screw on the other).

2.4 Removal

Removing a Function Card is the reverse of the install procedure.

3. Theory of operation

3.1 Hardware

This Function card is a 16bit DAC where each channel has its own DAC.

The speed of the DAC is 3.5 ms to settle the current output to 0.01%. The voltage version has an additional low-pass filter output amplifier with a 100 Hz (-3dB) cut-off frequency.

For the current output version the output current range is user programmable, whereas it is a hardware configurable option in case of the voltage output version.

The outputs are protected against damage by resistors in the case of the current output version, or by a buffer output stage in the case of the voltage version.

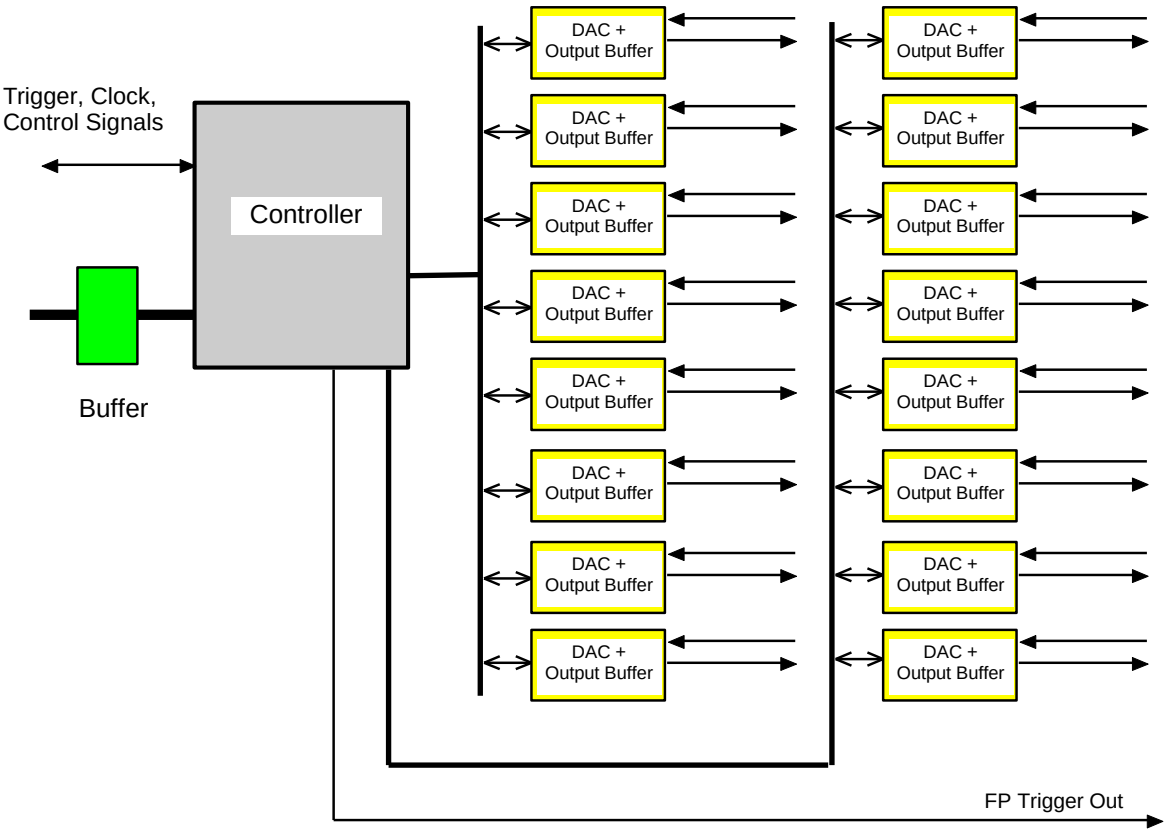


Figure 4: Simplified DAC block diagram

3.2 Operation

The circuitry consists of 16 DAC chips plus active and passive support circuitry. The DAC is serially controlled by a FPGA. The DAC's are logically organized in four groups G0=DAC 1,2,3,4; G1=5,6,7,8; G2=9,10,11,12; G3=13,14,15,16.

Each DAC data input is controlled by a clock signal, serial data and a load signal. The output operation modes (voltage or the different current output) are defined by static signals. In addition a clear line can be employed to set the output to the lower margin output.

Failures in the output (current version only) are detected by the DAC and signalled to the controller, where the failure can be read by software. Depending on the programming of the output trigger, the "Logical OR" of all failures can be sent to the front panel output. The front panel trigger output is an open collector output, with a 16k pull-up resistor.

3.3 Conversion

To understand the required sequence of operations the following figure shows the internal timing diagram to convert a digital data into the DAC output signal.

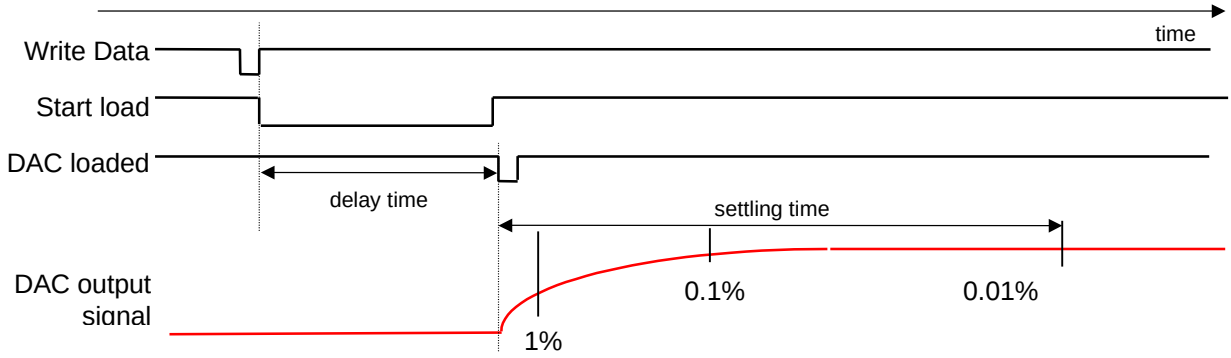


Figure 5: Illustration of the DAC timing. (Not to scale)

After the data is written to the internal shift register, the data is loaded serially into the DAC, which takes ~6.8µs. Then the DAC starts the conversion and after 3.5ms the DAC value has been settled to 0.01%. In case of the current output version, it depends on the output impedance when the output signal has reached that level.

3510 card channels with a voltage output have a low pass output filter fitted with a corner frequency of 100Hz (f_{-3dB}). This increases the settling time to 120ms for a 0.01% precision.

After the loading of one DAC channel, the software can safely load another DAC without interfering with the settling of the previously loaded DAC. If a fast controller is used then the DAC status should be monitored to ensure that there is no loading activity of the previous channel.

3.4 Registers

An output trigger signal (to the front panel or to the MB) can be selected from the following:

- Software; i.e. write a bit into a control and status register.
- A constant frequency generated by dividing the system clock with a programmable constant.
- The trigger input from the motherboard.
- Alarm output for any DAC detected failure (current version)**.**
- The polarity of the trigger signal is programmable.
- The front panel output is a TTL open collector type.
- Only one trigger source at a time can be active.

Address	Name	Function
0	FCID	ID register =0x9010
1	FCSER	serial Number
2	GCSR	general control and status register
3	FCLEN	size of installed FIFO = 0
5	OTRI	output trigger control
7	DIVCLK	system clock divider not used by DAC (for testing only)
8	MODE	R/W left for Software purpose bit 0 if set DAC serial Output disabled (for testing only)
E	FAIL	failure indicator from the DAC's (RO) 0 →15 if a bit is 0 then the responding channel has an error
F	DACMODE	DAC mode R/W
80 → 8F	DAC1 → 16	write to DAC 1 → 16 store the information and loads the DAC serially
80	RDAC	read back the content of the DAC shift register

FCID Function Card Id = 9010 (read only)

FCver Version number, content defined by Bustec Production Ltd. (read only).

GCSR General Control and Status Register

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Operation	-	-	-	-	-	-	-	-	-	-	-	-	-	RO	-	RW
Initial	x	x	x	x	x	x	x	x	x	x	x	x	x	0	x	0
Content	Not Used													Active		Reset

Active As long as the FPGA downloads data into the DAC chip(s) the bit is set to 1.

Reset 1 resets the internal control logic

FCLEN FIFO length (read only).

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Initial	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Content	FIFO length = 0															

OTRI Output trigger control

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Initial	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Content	Stat	Not used		DAC Fail	Pol	FPo	MBP	MBL	SWT	MBT	DivC	not used				

DivC use Clock Divider
MBT trigger Input from MB
SWT software trigger
MBL enable trigger to MB, signal is a level
MBP enable trigger pulse to MB. Pulse is 5 x clock signal width = 125ns
FPo enabled the Trigger to front panel output (plain signal)
Pol change polarity of output
DACFail enable failure OR
Stat status of the trigger line to the MB

DIVCLK

Defines the divider factor by which the system clock is divided. This may be used as a clock signal. However the software does not use it.

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Operation	-	-	-	-	-	-	-	-	RW	RW	RW	RW	RW	RW	RW	RW
Initial	x	x	x	x	x	x	x	x	0	0	0	0	0	0	0	0
Content	Not Used								Divider							

Mode

Read write register

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Operation	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW
Initial	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Content	Software purpose															DisD

DisD Disable data output to DAC

Fail

Indicates that when the current set by the DAC is not reached by the hardware.

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Init.	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Content	DAC 16	DAC 15	DAC 14	DAC 13	DAC 12	DAC 11	DAC 10	DAC 9	DAC 8	DAC 7	DAC 6	DAC 5	DAC 4	DAC 3	DAC 2	DAC 1

DACmode

DAC mode R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Init.	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Con-tents	OP TF3	Clr G3	Range G3		OP TF2	Clr G2	Range G2		OPT F1	Clr G1	Range G0		OPT F0	Clr G0	Range G0	

Clear 1-sets the output to the bottom of the span

0-normal operation

NOTE: The read back bit value is inverted for all clear signals! The default value after HW initialisation is 1 = clear.

Range Select In four groups of four AD420

RS 2	RS1	
0	0	Voltage output
0	1	4mA - 20mA
1	0	0mA - 20mA
1	1	0mA - 24mA

NOTE: The read back bit value is inverted for all RS2, RS1 signals! The default value after HW initialization is 0-24mA.

Option Flags

OPTFLAG 3 (R14) 0-All channels are either voltage or current

1-(no pull-down installed) mixed Mode, first 8 are current, second 8 are voltage

In the following table 0 indicates a pull-down is installed!

OPTFLAG2 R13	OPTFLAG1 R12	OPTFLAG0 R11	output variant
Not fitted = 1	not fitted = 1	not fitted = 1	current version (Std Opt)
Not fitted = 1	not fitted = 1	fitted = 0	0-5V
Not fitted = 1	Fitted = 0	not fitted = 1	0-10V $^{+}$ (Std Opt)
Not fitted = 1	Fitted = 0	fitted = 0	0-16V (Std Opt)
fitted = 0	not fitted = 1	not fitted = 1	+/- 5V
fitted = 0	not fitted = 1	fitted = 0	+/- 10V * (Std Opt)
fitted = 0	Fitted = 0	not fitted = 1	+/- 16V
fitted = 0	Fitted = 0	fitted = 0	special custom version

* Standard Version for bipolar voltage version

$^{+}$ Standard Version for unipolar voltage version

Note: The default output after reset in the voltage mode is bottom of the range. Software will set it to zero if a bipolar range is detected.

The connected equipment should be able to withstand voltages up to ± 30 V without damage!

DAC1 - DAC16 These registers have only write access

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Initial	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Content	New DAC value to be loaded into a DAC															

A write in one of the DAC registers loads automatically the new value into the DAC.

4. Software utilities

4.1 Introduction

Plug and play software was developed in compliance with the ProDAQ software line. Supported software package encompasses the instrument driver, Soft Front Panel, documentation and examples. The VXI*plug&play* soft front panel is a graphical user interface application developed for the instrument (Trigger/Clock function card). It is used to verify instrument operation and functionality when the instrument is first integrated into a system. It provides instrument control in a user-friendly environment, being both Windows 95 and NT compatible. The user interface uses the installed driver to control and operate the instrument. The soft font panel may be also used as a discussion on the top-level driver functions developed and their use in an application environment.

4.2 User interface and installation

4.2.1 Software installation

With the function card a VXI*plug&play* Disk is delivered. It contains the software required to operate the function card in the ProDAQ environment. After the 16-Ch. DAC Function Card has been installed into the 3120 or 3150 motherboard, the VXI*plug&play* software may be used to communicate with the motherboard. To install the software, first power on the mainframe, then perform the following operations:

1. Start Windows (95 or NT) on your computer if it is not already running.
2. Ensure no ProDAQ software is currently running on your computer.
3. Insert the ProDAQ 3510 installation disk #1 into the 3 ½" floppy disk drive.
4. Launch the SETUP.EXE program.
5. Follow the instructions presented by the SETUP program.

After the SETUP program has completed, the executable Soft Front Panel program may be run. The drivers are available for WIN 95 or WIN NT. In the following table win**xx** stands for the particular version being used. If the system is a Windows NT then the VXI*plug&play* path is \Vxipnp\WinNT

Description	File	Hard Disk Destination
Instrument Driver		
Driver Source	bu3510.c	\Vxipnp\win**xx**\bu3510
Header File	bu3510.h	\Vxipnp\win**xx**\include\
Function Panel	bu3510.fp	\Vxipnp\win**xx**\bu3510\
Microsoft Windows DLL	bu3510_32.dll	\Vxipnp\win**xx**\bin\
Common Interface Library Windows DLL	bu3100_32.dll	\Vxipnp\win**xx**\bin\
Microsoft Windows import Library	bu3510.lib	\Vxipnp\win**xx**\lib\msc\
Common Interface Microsoft Windows import Library	bu3100.lib	\Vxipnp\win**xx**\lib\msc\
Microsoft Visual Basic function declaration file	bu3510.bas	\Vxipnp\win**xx**\include\
Driver documentation	bu3510.doc	\Vxipnp\win**xx**\bu3510\
Driver Windows help	bu3510.hlp	\Vxipnp\win**xx**\bu3510\
Soft Front Panel executable file	bu3510.exe	\vxipnp\win**xx**\bu3510\
Qt library shared DLL	qt-mt303.dll	%winsysdir%\

4.2.2 Software utilization

The purpose of Soft Front Panel is to demonstrate instrument's abilities. The soft font panel may be also used as a discussion on the top-level driver functions developed and their use in an application environment.

After the start of the Soft Front Panel application, the user will be presented with a dialog box showing all available ProDAQ 3510 instruments in a system, allowing the selection of one instrument, which will be operated. Due to imposed limitations, there is no possibility to control simultaneously two or more instruments fitted to the same motherboard. If there is only one instrument available, this dialog box will not appear and this instrument will be automatically selected for operation. In order to run the user interface for the chosen 3510, select the appropriate position from the list and press OK button.

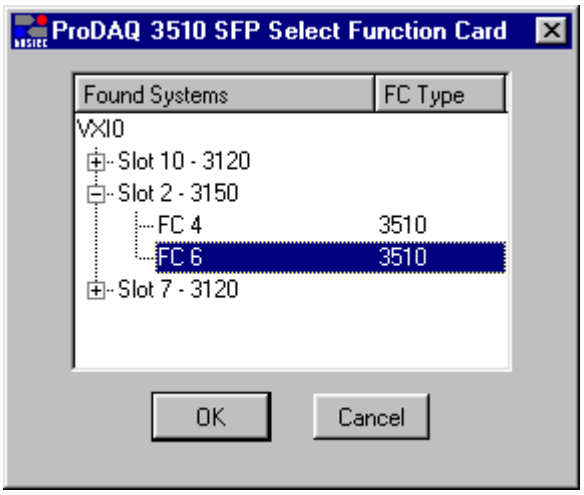


Figure 6: Function Card selection

This will invoke the main Soft Front Panel window as shown in Figure 7.

The soft front panel consist of the main window only and its operation is very simple. The window is divided into four groups where each group contains four channels. The group can be voltage or current type and the controls differ accordingly. For each channel there is possible to set the value in defined range. For the current type group there is also possible to select a range of output current using combo box provided, for the voltage type the range is fixed. Additionally there is an error indication feature for each current type channel. It consists of LED control, which turns red when an error occurres.

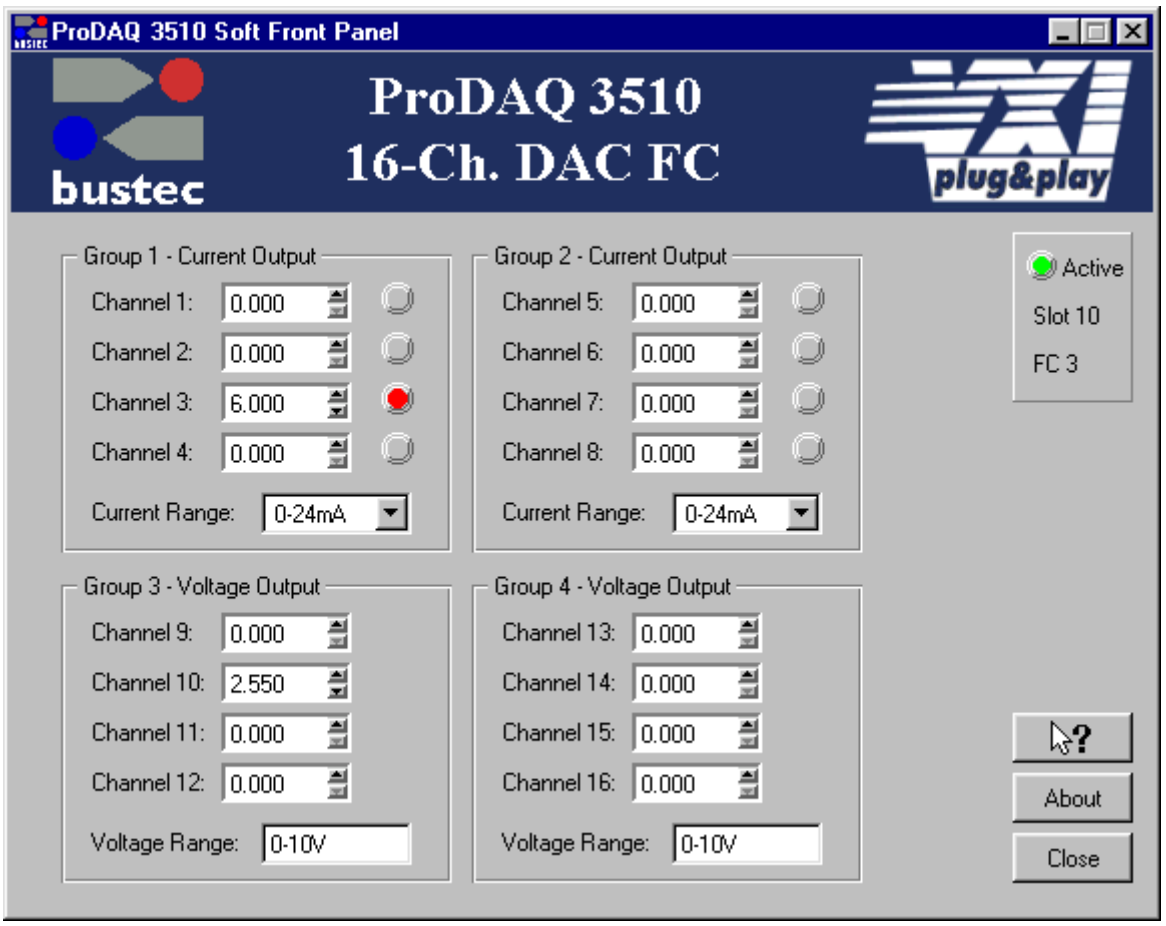


Figure 7: Soft front panel main window

At any time, on all panels, there is a context help available invoked by pressing the ? button, located at the window right-top corner, and then selection a control that help is needed for.

4.3 Programming concepts

4.3.1 Instrument driver overview

To use the instrument driver for the 16-Ch. DAC Function Card, one ProDAQ Motherboard, e.g. the 3120 or 3150, has to be used. In new 2.x version of drivers, a common interface library was implemented to act as an intermediate layer between the motherboard hardware and the driver, handling the communication to the different motherboards in a transparent way. In turn now every function card driver acts as a standalone VXIplug&play compatible driver, using its own instrument handle to communicate to the instrument. There is no longer the need to have a driver for the motherboard installed, although this is recommended. The common library is included to the installation package for every ProDAQ VXIplug&play driver.

The Instrument driver for the ProDAQ 3510 provides the following functionality.

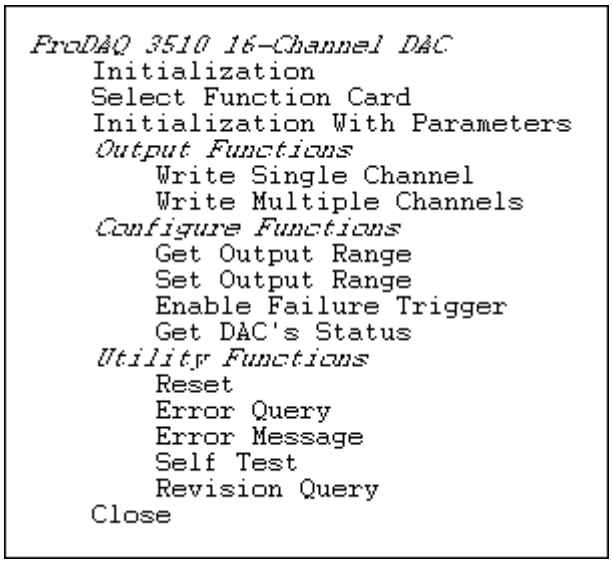


Figure 8: Instrument Driver function tree

A full description of the instrument driver functions can be referenced in the driver help file.

4.3.2 Error/Status Information

Every instrument driver function has the same return type format. Returning either a completion code or an error code.

```
ViStatus _VI_FUNC bu3510_functionName ( Parameters… );
```

In order to identify the successful operation of any function these codes can be used. The following example illustrates this principle.

```
ViSession vi;
ViStatus error;
ViChar msg[512];
   :
   :
error = bu3510_reset(vi);
if(error < VI_SUCCESS)
{
   bu3510_error_message (vi, error, msg);
   /* stop execution */
}
else if(error > VI_SUCCESS)
{
   bu3510_error_message (vi, error, msg);
   /* print a warning and continue execution */
}
```

If an error occurs, a value less than VI_SUCCESS is returned. The function bu3510_error_message converts the error code into a readable string. All driver functions operate along the same principles, so any errors in hardware access are easily determined.

If a warning occurs, a value greater than VI_SUCCESS is returned. The same function bu3510_error_message can be used to convert the warning code into readable string.

4.3.3 Connecting to the instrument

A typical initialization sequence is as following:

```
bu3510_init ("VXI::1::INSTR", VI_TRUE, VI_TRUE, &viSession);
bu3510_fcSelect (viSession, 2);  /*use a function card in position 2*/
```

The call of function `bu3510_fcSelect` is obligatory and has to be invoked after `bu3510_init()` function but before any other `bu3510_` function. Although, for convenience, another function is provided which encompasses functionality of those two function calls:

```
bu3510_paramInit ("VXI::1::INSTR",  2, VI_TRUE, VI_TRUE, &viSession);
```

There is a strong requirement that function `bu3510_close` should be called when the instrument is no longer used. Each ProDAQ driver obtains a lock to the motherboard resource, which is released by `bu3510_close` function afterwards. This also means that it is not possible to access two function cards, either the same or different located on the same motherboard, from two separate system processes, since they would use the same resource descriptor but different instances of common library DLL. Therefore, the unmatched `bu3510_close` call can lock the resource as long as common interface library is loaded.

4.3.4 Programming instrument

There are mainly two functions used during normal FC operation: `bu3510_writeChannel()` and `bu3510_setOutputRange()`.

The `bu3510_writeChannel()` function is used to set required output value for selected channel. To set multiple channels the `bu3510_writeChannels()` function can be used, which takes as a parameter array of values for channels selected by the channel mask.

Each channel has range of values possible to set, range is applied for all channels within group using `bu3510_setOutputRange()` function. The `bu3510_setOutputRange()` function can be used only for current type groups as voltage type group is fixed by hardware.

To obtain a group range the `bu3510_setOutputRange()` function is used, it returns symbolic value containing information about a group type and range. These constants are defined in bu3510.h header file and explained in driver's documentation.

```
bu3510_setOutputRange(viSession, 3, bu3510_CURRENT_20MA);
bu3510_writeChannel(viSession, 2, 0.008);
bu3510_writeChannel(viSession, 5, 0.012);
```

The example above assumes that group 1st and 2nd are current type and sets their range to 0 - 20mA. So first 8 channels are set to range 0 - 20mA.

Next the second channel output value is set to 8mA and 5th channel's output value is set to 12mA.

To find out if an error occurred on the selected output the `bu3510_getDACStatus()` function can be used:

```
ViBoolean dacStatus;
bu3510_getDACStatus(viSession, 2, &dacStatus);
```

If an error occurred the `bu3510_getDACStatus()` function sets `dacStatus` parameter value to VI_TRUE.

There is also another possible way to recognize error without checking it every time. The `bu3510_enableFailureTrigger()` function configures output trigger to be active during error presence and additionally function can install interrupt handler and connects output trigger to interrupt line.

The interrupt service routine will react on DAC failure, which appears on current type output in case of open circuit or improper load connected to the output.


```
void _VI_FUNCH isrFunc(viSession vi, void *para)
{
  /* Check which channel(s) caused error and take an action */
}

bu3510_enableFailureTrigger(viSession, bu3510_TRIG_DEST_MB,
                                       bu3510_OTR_MB_LEVEL,
                                       isrFunc, VI_NULL);
```

To uninstall previously installed interrupt service routine the same function is called with VI_NULL value in place of interrupt service routine parameter.

```
bu3510_enableFailureTrigger(viSession, bu3510_TRIG_DEST_NONE,
                                       bu3510_OTR_MB_LEVEL,
                                       VI_NULL, VI_NULL);
```

The sample above uninstalls interrupt service routine and additionally disables output trigger by choosing none for the trigger destination. Second parameter has no meaning in this case but must be supplied correctly.