# µDAQ-Lite

# User's Manual

Eagle Technology – Cape Town, South Africa
Copyright © 2005
www.eagledaq.com

# μDAQ-Lite Remote Devices

## Data Acquisition and Process Control

**Copyright**

All rights reserved. No part of this publication may be reproduced, stored in a retrieval system, or transmitted, in any form or any means, electronic, mechanical, by photographing, recording, or otherwise without prior written permission.

Copyright © Eagle Technology, South Africa
November 2005
Revision 1.1

Information furnished in this manual is believed to be accurate and reliable; however no responsibility is assumed for its use, or any infringements of patents or other rights of third parties, which may result from its use.

Trademarks and Logos in this manual are the property of their respective owners.

**Product Warranty**

Eagle Technology, South Africa, warrants its products from defect in material and workmanship from confirmed date of purchase for a period of one year if the conditions listed below are met. The product warranty will call the Eagle Technology Data Acquisition Device short as **ETDAQD**.

- The warranty does not apply to an **ETDAQD** that has been previously repaired, altered, extended by any other company or individual outside the premises of Eagle Technology.
- That a qualified person configure and install the **ETDAQD**, and damages caused to a device during installation shall make the warranty void and null.
- The warranty will not apply to conditions where the **ETDAQD** has been operated in a manner exceeding its specifications.

Eagle Technology, South Africa, does not take responsibility or liability of consequential damages, project delays, damaging of equipment or capital loss as a result of its products.

Eagle Technology, South Africa, holds the option and final decision to repair or replace any **ETDAQD.** Proof of purchase must be supplied when requesting a repair.

# TABLE OF CONTENTS

# Table of Figures

# Table of Tables

# 1. Introduction

The µDAQ-Lite devices are Universal Serial Bus architecture data acquisition devices. They are part of the µDAQ series, more specifically digital I/O and Analog I/O for USB.

The µDAQ-Lite has support for eight digital input, eight digital output lines, one counter, two analog output, and eight analog inputs.

## Features

The µDAQ-Lite series has some very unique features and are listed below:

- USB Revision 1.1 compliant @ full speed.
- Powered by USB port.
- TTL compatible digital I/O ports.
- 16-bit counter-timer.
- Channel list and voltage range scanning @ 49 KHz.
- 12-bit Analog I/O resolution.
- Quick and effortlessly to install.

## Applications

The µDAQ-Lite series can be used in the following applications:
- Automation test equipment.
- TTL compatible status monitoring.
- Plant/Factory process control.
- Pulse counting.
- Controlling and monitoring of any TTL compatible equipment.
- Mobile computing.
- Laboratory applications

## Key Specifications

- 1 x 8-bit digital output port.
- 1 x 8-bit digital input port.
- 1 x 16-bit counter.
- Fully programmable digital input/output system.
- Fully programmable counter-timer system.
- 12-bit Resolution analog input system with a max range of ±10 volt.
- Fully programmable channel/gain list @ 49 KHz.
- 12-bit Resolution analog output system with a range of ±10 volt.

---

## Software Support

The μDAQ-Lite series is supported by EDRE SDK and has an extensive range of examples. The software will help you to get your hardware going very quickly. It also makes it easy to develop complicated control applications. All operating system drivers, utility and test software are supplied on the EDR Enhanced CD-Rom. The latest drivers can also be downloaded from the Eagle Technology website. For further support information see the Contact Details section.

## Contact Details

Below are the contact details of Eagle Technology.

**Eagle Technology**
PO Box 4376
Cape Town
8000
South Africa
Telephone +27 (021) 423 4943
Fax +27 (021) 424 4637
E-Mail eagle@eagle.co.za
Website http://www.eagledaq.com

# 2

# 2. Installation

This chapter describes how to install and configure the μDAQ-Lite device for the first time. Minimal configuration is necessary; almost all settings are done through software. The operating system will take care of all resource assignments.

## Package

μDAQ package will contain the following:
- μDAQ-Lite device.
- USB cable.
- Software CD-Rom.

## Operating System Support

The μDAQ-Lite series support the Windows Driver Models (WDM) driver types. The operating systems are listed in the table below.

| Board Type | Revision | Operating Systems | Driver Type |
|---|---|---|---|
| μDAQ-Lite | Revision 1 | Windows 2000/98/ME/XP | WDM PnP |

**Table 2-1 Operating System Support**

## Installation

This section will describe how to connect your USB device to your computer.
- Select any unused USB port and plug-in the A-side of the USB cable.
- Pug-in the B-side of the cable into the USB device.
- The operating system should immediately detect a new device was installed.

### PnP Installation

Installing the Windows 98/2000/XP device driver is a very straightforward task. Because it is plug and play Windows will detect a new device was installed. No setup is necessary. You simply have to supply Windows with a device driver.

**Wait until Windows detects the new hardware**



**Select the Advanced option and click "Next"**
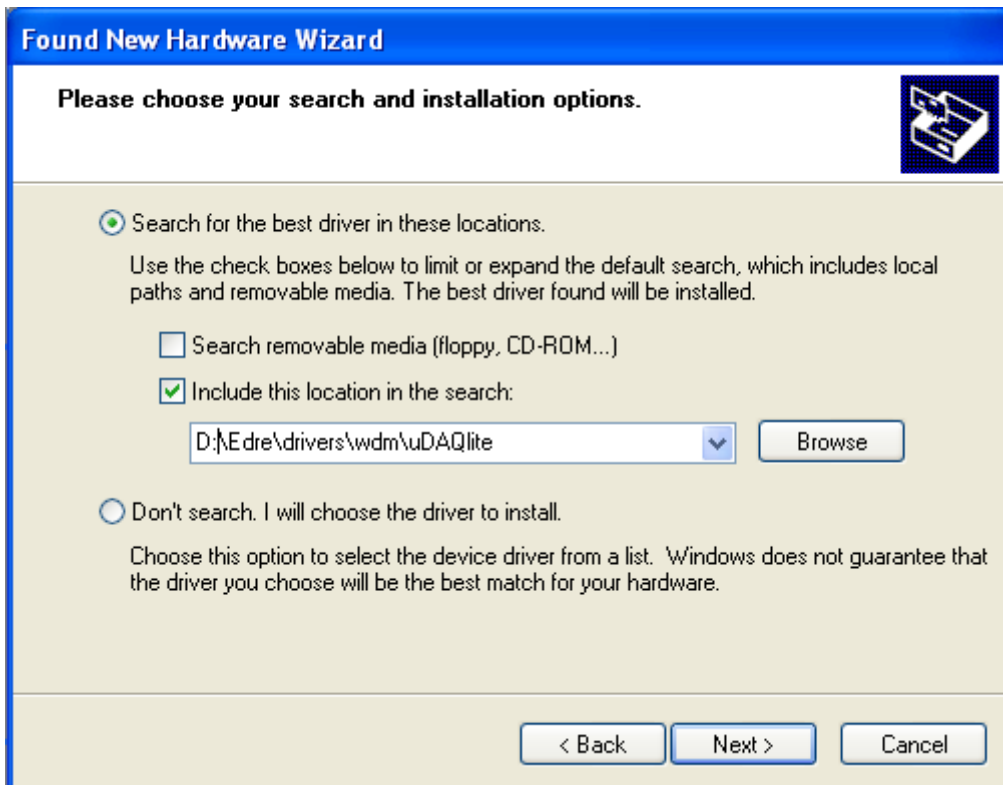
**Select "Search for the best driver in these locations"**
**You can enter the directory path on the CDROM where the driver can be found or select to "Browse" the CDROM. If you opt to browse, search for the uDAQlite.inf file on the Eagle CD-Rom.**
**The driver is normally located in the <CDROM>:\EDRE\DRIVERS\WDM\uDAQlite directory.**
**Click "Next"**
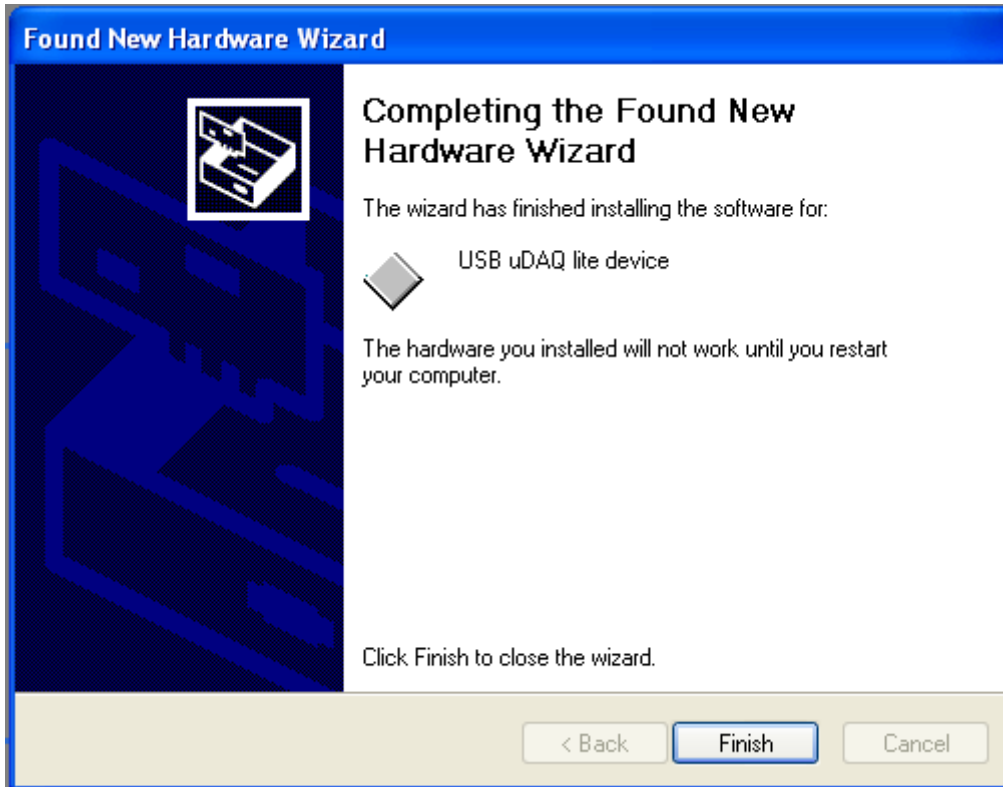
**When done you might have to restart your computer.**

## Post installation

When done with the driver installation the device manager can be open to make sure the installation was a success.

- First make sure that the driver is working properly by opening the *Device Manager*.
- Check under the Eagle Data Acquisition list if your board is listed and working properly. See picture below.
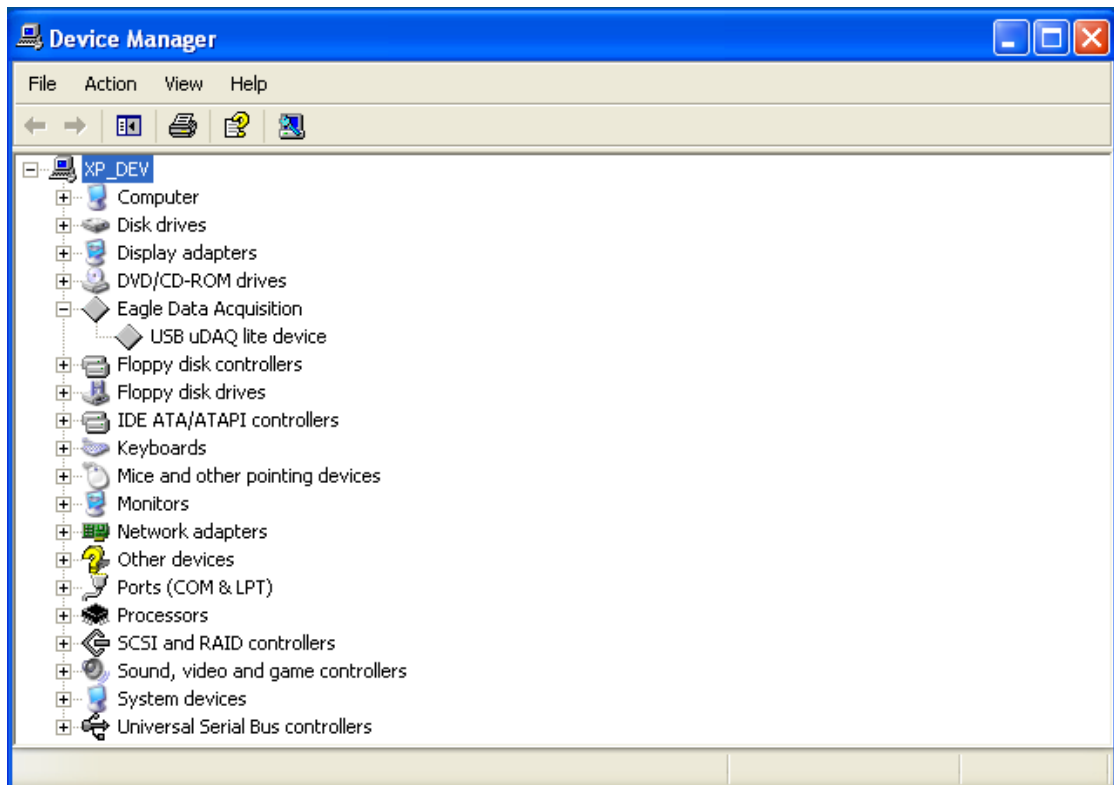
- Run **edreapi.exe** found on the Eagle CD-Rom and follow the on screen instructions. Edreapi.exe will install activex controls and libraries needed by applications controlling the hardware. (Normally located in <CDROM>:\EDRE\API directory)
- You can now verify that installation was successful by opening the control panel and then the "*EDR Enhanced Setup"* folder. This dialog should list all installed hardware. Verify your board's properties on this dialog. See picture below
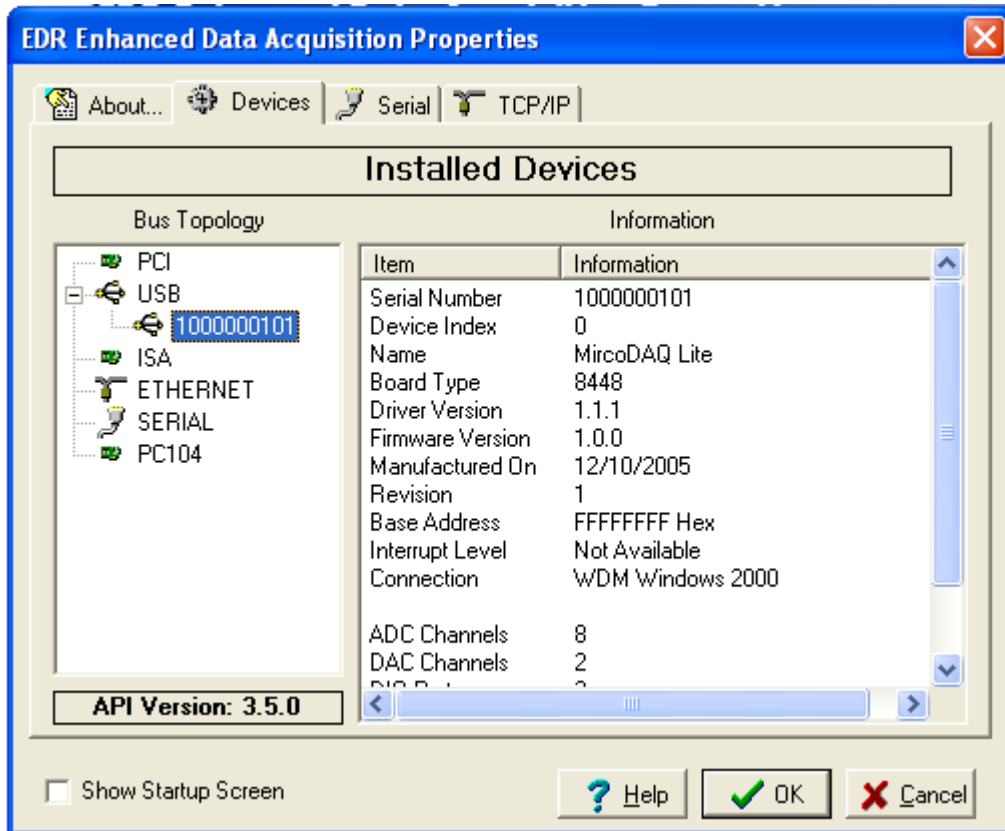
## Application Software

The EDR Enhanced Software Development Kit CD-Rom comes with WaveView for Windows™. WaveView has support for Analog Inputs, Analog Outputs, Digital I/O.

WaveView can be found on the EDR Enhanced CD-Rom. (Normally located in <CDROM>:\EDRE\APPS\WVFW directory)

# 3

# 3. Interconnections

The μDAQ-Lite has connectors for digital I/O, counter-timers and analog I/O. The μDAQ-Lite make use of screw terminals for easy access. The screw terminal has a hole diameter of 1.15mm – 0.03mm.

## Pin Assignments

## Pin Descriptions

### Channel (CH0-7)

Channel can symbolize a digital input, digital output, analog input, or analog output.

### Analog Ground (AGND)

These lines are the analog reference ground the analog inputs and outputs.

### Reference (REF)

The reference line is used for calibration purpose only and should never be use as a supply voltage.

### Analog Trigger (TRIGGER)

This is the external gate for the analog input sampling system.

### Master Clock (MASTER)

When synchronizing two or more of the μDAQ-Lite devices this pin can be used to output a clock frequency. The signal are referenced to digital ground (DGND)

### Slave Clock (SLAVE)

This pin is used as the external clock source for analog sampling. It is used to control the convert timing of the analog to digital converter. The signal must be referenced to digital ground (DGND).

### Digital Ground (DGND)

All digital ground signals (DIO & counter) should be connected to this pin.

### Counter Input (IN)

The pin is used for the external clock source to the counter. The signal must be referenced to digital ground (DGND).

### Counter Output (OUT)

This is the output of the counter. The signal are be referenced to digital ground (DGND).

### Counter Gate (GATE)

The pin is used for the external gate to the counter. The signal must be referenced to digital ground (DGND).

# 4

# 4. Programming Guide

The μDAQ-Lite is supplied with a complete software development kit. EDR Enhanced (EDRE SDK) comes with drivers for many operating systems and a common application program interface (API). The API also serves as a hardware abstraction layer (HAL) between the control application and the hardware. The EDRE API makes it possible to write an application that can be used on all hardware with common sub-systems.

## EDR Enhanced API

The EDR Enhanced SDK comes with both ActiveX controls and a Windows DLL API. Examples are provided in many different languages and serve as tutorials. EDRE is also supplied with a software manual and user's guide.

The EDRE API hides the complexity of the hardware and makes it really easy to program the μDAQ-Lite device. It has got functions for each basic sub-system and is real easy to learn.
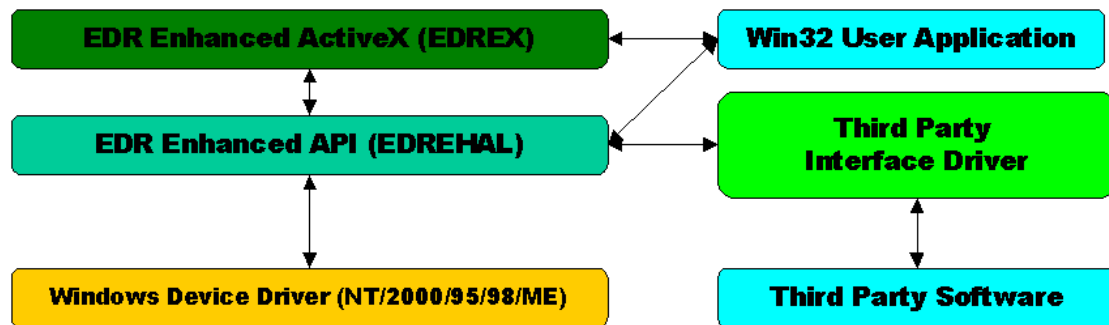


**Figure 4-1 EDR Enhanced Design**

## Digital Inputs/Outputs

The μDAQ-Lite device can has 8 digital input and 8 digital output lines. All digital lines conform to TTL specifications.

### Reading the Digital Inputs

A single call is necessary to read a digital input port. The '**Port**' parameter will always be equal to 0 because there is only one input port available.

**API-CALL**
*Long EDRE_DioRead(ulng Sn, ulng Port, ulng *Value)*
The serial number, port, and a pointer to variable to hold the result must be passed by the calling function. A return code will indicate if any errors occurred.

**ACTIVEX CALL**
*Long EDREDioX.Read(long Port)*
Only the port-number needs to be passed and the returned value will either hold an error or the value read. If the value is negative an error did occur.

### Writing to the Digital Outputs

A single call is necessary to write to a digital output port. The '**Port**' parameter will always be equal to 0 because there is only one output port available.

**API-CALL**
*Long EDRE_DioWrite(ulng Sn, ulng Port, ulng Value)*
The serial number, port, and a value must be passed by the calling function. A return code will indicate if any errors occurred.

**ACTIVEX CALL**
*Long EDREDioX.Write(long Port, long Value)*
The port number and value to be written needs to be passed and the returned value holds an error or the value read. If the value is negative an error did occur.

## Counter

The counter sub-system is supported by functions to Write, Read, Configure and controlling the gate. There is only 1 counter. The table below shows the relation of the counter and its software assigned number.

| Counter | Assigned Number | Description |
|---------|-----------------|-------------|
| CT0 | 0 | Counter 0 |

**Table 4-1 Counter Assignment**
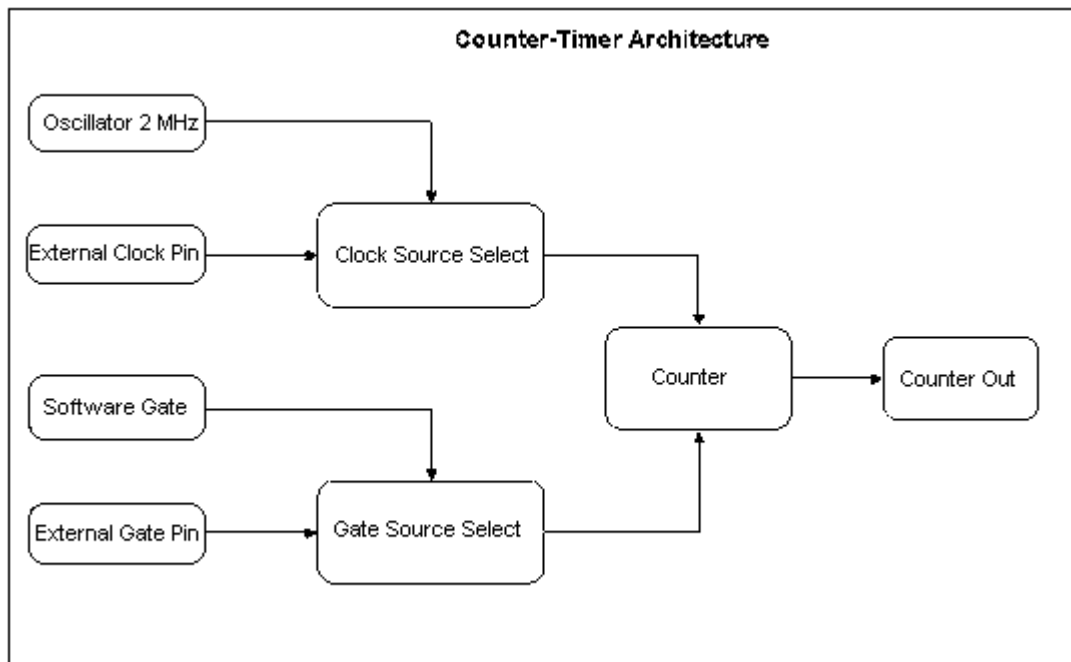


**Figure 4-2 Counter-Timer Architecture**

### Architecture

The clock source and gate can be selected via software. The clock source can either be internal or external. The gate can also be either inter or external. The internal gate is controlled via software as well.

### Writing the initial counter value

A single call is necessary to write a counter's initial load value.

**API-CALL**
*Long EDRE_CTWrite(uIng Sn, uIng Ct, uIng Value)*
The serial number, counter-number, and a value must be passed by the calling function. A return code will indicate if any errors occurred.

**ACTIVEX CALL**
*Long EDRECTX.Write(long Port, uIng Value)*
The counter-number and a value must be passed by the calling function. A return code will indicate if any errors occurred.

### Reading the counter value

A single call is necessary to read a counter.

**API-CALL**
*Long EDRE_CTRead(uIng Sn, uIng Ct, puIng Value)*
The serial number, counter-number, and a reference parameter must be passed by the calling function. A return code will indicate if any errors occurred.

**ACTIVEX CALL**
*Long EDRECTX.Read(long Port)*
The counter number must be passed by the calling function. If the return code is negative it means an error occurred, otherwise it will be the value read from the counter.

### Configuring a counter

A single call is necessary to configure a counter.

**API-CALL**
*Long EDRE_CTConfig(uIng Sn, uIng Ct, uIng Mode, uIng Type, uIng ClkSrc, uIng GateSrc)*
The serial number, counter-number, mode, type, clock source and gate source is needed to specify a counter's configuration. A return code will indicate if any errors occurred.

**ACTIVEX CALL**
*Long EDRECTX.Configure(long ct, long mode, long type, uIng source, uIng gate)*
The counter-number, mode, type, clock source and gate source is needed to specify a counter's configuration. A return code will indicate if any errors occurred.

The µDAQ-Lite only uses the counter clock and gate parameters. The table below shows the options for each parameter.

| Parameter | Description |
|---|---|
| Sn | Serial Number |
| Ct | Counter Number: |
| | 0 : Counter 0 |
| Mode | NOT USED |
| Type | NOT USED |
| Source | 0 : Internal (2 MHz) |
| | 1 : External (External connector) |
| Gate | 0 : Internal |
| | 1 : External (External connector) |

**Table 4-2 Counter Configuration**

### Controlling the counter gate

A single call is necessary to control a counter's gate.

**API-CALL**
*Long EDRE_CTSoftGate(uIng Sn, uIng Ct, uIng Gate)*
The serial number, counter-number and gate are needed to control a counter's gate. A return code will indicate if any errors occurred.

**ACTIVEX CALL**
*Long EDRECTX.SoftGate(uIng Ct, uIng Gate)*
The counter-number and mode is needed to control a counter's gate. A return code will indicate if any errors occurred.

These values are acceptable as a gate source.

| Value | Description |
|---|---|
| 0 | Gate disabled |
| 1 | Gate enabled |

**Table 4-3 Gate Configuration**

## Analog Output

The μDAQ-Lite has 2 x 12-bit analog output channels with a range of ±10 volt. These channels are very easy to program. A single command is used to write to them.

### Writing to a DAC channel

A single call is necessary to set a voltage on a DAC channel. The table below shows the relation between the software channel and the channel on the connector.

| Assigned Software Channel | Assigned Connector Pin |
|---|---|
| 0 | DAC0 |
| 1 | DAC1 |

**Table 4-4 Assigned DAC Channels**

**API-CALL**
*Long EDRE_DAWrite (ulng Sn, ulng Channel, long uVoltage)*
The serial number, DAC channel and micro-voltage is needed to set a DAC channel's voltage. A return code will indicate if any errors occurred.

**ACTIVEX CALL**
*Long EDREDAX.Write (ulng Channel, long uVoltage)*
The DAC channel and micro-voltage is needed to set a DAC channel's voltage. A return code will indicate if any errors occurred.

## Analog Input

The μDAQ-Lite has a very flexible analog input sub-system. Configuration includes dynamic range, gain and differential or single ended inputs. Each of these settings can be applied to an individual channel while scanning.

The analog inputs can operate in two modes, single read or scanning. Only one mode can be used at a single moment. The table below shows the relation between the software assigned channels and the connector.

| Assigned Software Channel | Input Type | Input Pin | Reference Pin |
|---|---|---|---|
| 0 | Single | ACH0 | AGND |
| … | … | … | … |
| 7 | Single | ACH7 | AGND |
| 0 | Differential | ACH0 | ACH1 |
| 1 | Differential | ACH2 | ACH3 |
| 2 | Differential | ACH4 | ACH5 |
| 3 | Differential | ACH6 | ACH7 |
| 4 | Differential | ACH1 | ACH0 |
| 5 | Differential | ACH3 | ACH2 |
| 6 | Differential | ACH5 | ACH4 |
| 7 | Differential | ACH7 | ACH6 |

**Table 4-5 Assigned Analog Input Channels**

### Reading a single voltage from a channel

To read a single ADC channel you need to specify the channel, voltage range and gain.

**API-CALL**
*Long EDRE_ADSingle (uLng Sn, uLng Channel, uLng Gain, uLng Range, plong uVoltage)*

| Parameter | Type | Description | | |
|---|---|---|---|---|
| Sn | Unsigned long | Device serial number | | |
| Channel | Unsigned long | ADC channel to read | | |
| Gain | Unsigned long | Gain Codes | | |
| | | Value | Single-ended Gain | Differential-ended Gain |
| | | 0 | X ¼ (±10V) | X ⅛ (±20V) |
| | | 1 | | X ¼ (±10V) |
| | | 2 | | X ½ (±5V) |
| | | 3 | | X ⅝ (±4V) |
| | | 4 | | X 1 (±2.5V) |
| | | 5 | | X 1.25 (±2V) |
| | | 6 | | X 2 (±1.25V) |
| | | 7 | | X 2.5 (±1V) |
| Range | Unsigned long | Range Codes | | |
| | | Value | Range | |
| | | 0 | BIPOLAR, SINGLE-ENDED | |
| | | 1 | BIPOLAR, DIFFERENTIAL-ENDED | |
| uVoltage | Pointer to long | Returned micro voltage | | |
| Return | long | Error code | | |

**ACTIVEX CALL**
*Long EDREADX.SingleRead (long Channel)*
Make sure to set the *Gain* and *Range* properties of the ADC ActiveX control. This will in turn set the range and gain when reading the ADC channel.

## Configuring the ADC subsystem for scanning

This is the most complicated part of configuring the μDAQ-Lite for auto scanning. Make sure that you use the correct format when applying the channel list configuration. There are many loopholes and care should be taken when implementing code to configure the μDAQ-Lite.

### API-CALL
*Long EDRE_ADConfig (uIng Sn, puIng Freq, uIng ClkSrc, uIng Burst, uIng Range, puIng ChanList, puIng GainList, uIng ListSize)*

The following parameters must be specified when configuring the ADC sub-system.

| Parameter | Type | Description |
|---|---|---|
| Sn | Unsigned long | Device serial number |
| Frequency | Pointer to an unsigned long | ADC Sampling frequency |
| ClkSrc | Unsigned long | This parameter is used to configure the clock/convert source of the ADC sub-system. |

| Offset (bits) | Description |
|---|---|
| 0 | Clock Source (C0-C7)<br>0: Internal 2 MHz clock<br>1: External (SLAVE) |
| 8 | Trigger Source (G0-G7)<br>0: Disable<br>1: External Trigger (TRIGGER) |

Example Layout:

| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| G7 | G6 | G5 | G4 | G3 | G2 | G1 | G0 | C7 | C6 | C5 | C4 | C3 | C2 | C1 | C0 |

| Parameter | Type | Description |
|---|---|---|
| Burst | Unsigned long | Not Used |
| Range | Unsigned long | Not used |
| ChanList | Pointer to an unsigned long | This is a pointer to an array that contains the list of channels to be scanned. The array length should be the same length as the value of ListSize |
| GainList | Pointer to an unsigned long | GainList is an array that contains the gain/range settings for each channel in the scan list. The array length should the same as the ListSize value. |

| Offset (bits) | Description | | |
|---|---|---|---|
| 0 | Specifies the gain of the channel.(G) | | |
| | Value | Single-ended Gain | Differential-ended Gain |
| | 0 | X ¼ (±10V) | X ⅛ (±20V) |
| | 1 | | X ¼ (±10V) |
| | 2 | | X ½ (±5V) |
| | 3 | | X ⅝ (±4V) |
| | 4 | | X 1 (±2.5V) |
| | 5 | | X 1.25 (±2V) |
| | 6 | | X 2 (±1.25V) |
| | 7 | | X 2.5 (±1V) |
| 8 | Specifies the range of the channel. (R) | | |
| | Value | Range | |
| | 0 | BIPOLAR, SINGLE-ENDED | |
| | 1 | BIPOLAR, DIFFERENTAIL-ENDED | |

Example Layout:

| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| R7 | R6 | R5 | R4 | R3 | R2 | R1 | R0 | G7 | G6 | G5 | G4 | G3 | G2 | G1 | G0 |

| Parameter | Type | Description |
|---|---|---|
| ListSize | unsigned long | This is the length of the channel list. |
| Return | long | Error code |

### ACTIVEX CALL
*Long EDREADX.Configure (plong Channels, plong Gains, long ListSize)*

The *Frequency* and *ClockSource* ADC ActiveX control must be setup before calling the configure function. See the above table for the layout of the Channels and Gains lists.

*EDREADX.Frequency*

This is the sampling frequency of the ADC process. This parameter must be set before calling the Configure method. After calling the Configure method the Frequency property will be set to the actual sampling frequency.

### EDREADX.ClockSource
The clock source property is used to specify the clock settings for the ADC process.

| Offset (bits) | Description |
|---|---|
| 0 | Clock Source (C0-C7)<br>0: Internal 2 MHz clock<br>1: External Convert (SLAVE) |
| 8 | Trigger Source (G0-G7)<br>0: Disable<br>1: External Trigger (TRIGGER) |

Example Layout

| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| G7 | G6 | G5 | G4 | G3 | G2 | G1 | G0 | C7 | C6 | C5 | C4 | C3 | C2 | C1 | C0 |

## Starting and Stopping the ADC process
A single call is necessary to start or stop the ADC process

### API-CALL
### Long EDRE_ADStart (ulng Sn)
A serial number needs to be specified to start the ADC process. A returned error code will indicate if the function succeeded.

### ACTIVEX CALL
### Long EDREADX.Start ()
A call to the start method will start the ADC process of the device too which the ActiveX control is linked. A returned error code will indicate if the function succeeded.

### API-CALL
### Long EDRE_ADStop (ulng Sn)
A serial number needs to be specified to stop an ADC process. A returned error code will indicate if the function succeeded.

### ACTIVEX CALL
### Long EDREADX.Stop ()
A call to the start method will stop the ADC process of the device too which the ActiveX control is linked. A returned error code will indicate if the function succeeded.

## Driver buffer functions
A single call is necessary copy data from the driver buffer to a user buffer. The driver-buffer is a large circular buffer that can hold data for a period of time running at full speed. This buffer needs to be emptied regularly to make sure it does not overrun. The buffer can be queried with number of samples available and other status issues as well. There are two functions available to copy data, one for copying voltages, another to copy the raw data. The raw data is significantly faster as for the data does not have to be converted to voltages before copying it to the user buffer. The raw data also occupies less space than the micro voltage buffer. There are also functions to write data to disk as the user buffer get copied. Refer to the EDR Enhanced programming manual for a reference to these functions.

### API-CALL
### Long EDRE_ADGetData (ulng Sn, plong Buf, pulng BufSize)
### ACTIVEX CALL
### Long EDREADX.GetData (plong Buffer, plong Size)
To retrieve data from the driver buffer the serial number need to be supplied, a buffer to hold the data and the size of the buffer or requested number of samples. The driver will only copy

the number of available samples in multiple of the channel list. For the ActiveX call only the buffer and size need to be supplied.


## Querying the ADC subsystem

The driver can be queried to check the status of the ADC subsystem. The number of unread samples is one example. The appendix has a list of all possible query codes.

**API-CALL**
*Long EDRE_Query (ulng Sn, ulng QueryCode, ulng Param)*
A serial number, query code and parameter must be specified when doing a query.

**ACTIVEX CALL**
*Long EDREADX.GetUnread ()*
This function automatically queries the ADC driver buffer for the number of available samples.

# 5

# 5. Calibration

If the device needs to be calibrated, the software can be found on the EDR Enhanced SDK CD-Rom. Normally located in <CDROM>:\EDRE\APPS\uDAQlite_cal directory.
This application provides step-by-step information of how to calibrate your device. Make sure that you have a high precision multimeter and calibration voltage source. This will help to configure your device more accurately.

## Calibration Procedure – uDAQ-lite

1. Install the USB Calibration Software <CDROM>\EDRE\APPS\uDAQlite_cal
2. Run the USB Calibration Software.
3. Follow the step-by-step information on screen to tune your device.
4. Make sure to save the data to your device.

## Equipment

The following calibration equipment is required to calibrate the µDAQ-lite. If the calibration equipment does conform to these specifications it will not be possible to calibrate the device accurately.

## High Precision Multimeter

A high precision multimeter is required to measure output analog voltages. The HP3478A digital multimeter is an example of such a device. This device is used as standard test equipment to calibrate the µDAQ-lite. Make sure the device conform to its own calibration requirements and that it is serviced regularly. The device requirements are the following.

| Item | Specification |
|---|---|
| Voltage Range | -10V to 10V |
| Type | Analog Input |
| Relative Accuracy | 0.1 % of 1 bit in 16384 |
| Accuracy | < 1.2 µV |

**Table 5-1 Analog Multimeter Requirements**

## High Precision Voltage Source

A high precision voltage source is required to generate input analog voltages. The Burster Digistant Typ 4405 is am example of such a device. The device is used as standard test equipment to calibrate the µDAQ-lite. Make sure the device conform to its own calibration requirements and that it is serviced regularly. The device requirements are the following.

| Item | Specification |
|------|---------------|
| Voltage Range | 0V to 10V |
| Type | Analog Output |
| Relative Accuracy | 0.1 % of 1 bit in 16384 |
| Accuracy | < 1.2 μV |

**Table 5-2 Analog Source Requirements**

## Coaxial Calibration Cable

A specialized cable is needed to connect the calibration equipment to the device. The connection points are the following.

| Source | μDAQ and Rugged μDAQ Pin | Destination |
|--------|---------------------------|-------------|
| Analog Ground | AGND | 1. Voltage Generator Reference |
|  |  | 2. Voltage Meter Reference |
| Analog Input Channel 0 - 7 | ANALOG INPUTS CH0 - 7 | Voltage Generator Positive |
| Analog Output Channel 0 | ANALOG OUTPUTS CH0 | Voltage Meter Positive |
| Analog Output Channel 1 | ANALOG OUTPUTS CH1 | Voltage Meter Positive |

 **Table 5-3 Calibration Cable Connections**

The diagram below shows a typical connection cable. To reduce external noise effects on the process only use coaxial cables. Banana type plugs can be used to connect to the calibration instruments.
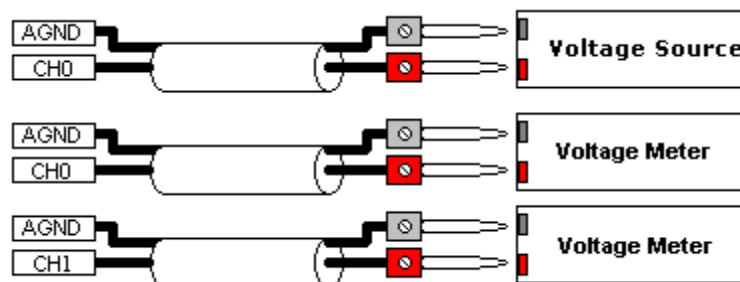


**Figure 5-1 Calibration Cable**

## Software

The μDAQ-lite require that the device software must be installed and operational. If not see the **Installation** chapter to setup the device. The next step is to install the calibration software for the μDAQ-lite.

### Software Installation

The software to calibrate the μDAQ-lite can be found on the Eagle Technology CDROM at *<EAGLECD>\EDRE\APPS\uDAQlite_cal*.
The installation application will place a short cut under Eagle Technology on the Windows Menu System.

## Operation

The software application will indicate the current step and a description of what to do. Use the buttons at the bottom to navigate. The slider bar is used to adjust the setting. The mouse roller button or the keyboard arrow keys can be used for fine adjustment. The indicator box will show any readings if they are relevant to the current step. If the device does not respond to the adjustments it can be caused by the wrong wiring connection. Factory settings can be reloaded by pushing the "*Load Factory Settings*" button.

## Procedure and Methodology

The table below indicates the purpose in calibrating the analog device. Follow the steps as below. The software will also show a short description of the process and what to do. End the end make sure to save the new settings to the device.

| Step | Type | Sub-System | Calibration Device | Description |
|---|---|---|---|---|
| 1 | Getting Ready | None | None | Press 'Next' to start the calibration process. Before starting allow for the device to settle to ambient temperature. |
| 2 | CH0 Bipolar, Single-ended offset | Analog Input | Short circuit | Connect channel 0 (CH0) to analog ground (AGND). Move the slider until you get the analog reading to hover around 0 V. |
| 3 | CH1 Bipolar, Single-ended offset | Analog Input | Short circuit | Connect channel 1 (CH1) to analog ground (AGND). Move the slider until you get the analog reading to hover around 0 V. |
| 4 | CH2 Bipolar, Single-ended offset | Analog Input | Short circuit | Connect channel 2 (CH2) to analog ground (AGND). Move the slider until you get the analog reading to hover around 0 V. |
| 5 | CH3 Bipolar, Single-ended offset | Analog Input | Short circuit | Connect channel 3 (CH3) to analog ground (AGND). Move the der until you get the analog reading to hover around 0 V. |
| 6 | CH4 Bipolar, Single-ended offset | Analog Input | Short circuit | Connect channel 4 (CH4) to analog ground (AGND). Move the slider until you get the analog reading to hover around 0 V. |
| 7 | CH5 Bipolar, Single-ended offset | Analog Input | Short circuit | Connect channel 5 (CH5) to analog ground (AGND). Move the slider until you get the analog reading to hover around 0 V. |
| 8 | CH6 Bipolar, Single-ended offset | Analog Input | Short circuit | Connect channel 6 (CH6) to analog ground (AGND). Move the slider until you get the analog reading to hover around 0 V. |
| 9 | CH7 Bipolar, Single-ended offset | Analog Input | Short circuit | Connect channel 7 (CH7) to analog ground (AGND). Move the slider until you get the analog reading to hover around 0 V. |
| 10 | CH0 Bipolar, Differential-ended offset | Analog Input | Short circuit | To calibrating differential channel 0 connect channel 0 (CH0) to channel 1 (CH1). Move the slider until you get the analog reading hovers around 0 V. |
| 11 | CH1 Bipolar, Differential-ended offset | Analog Input | Short circuit | To calibrating differential channel 1 connect channel 2 (CH2) to channel 3 (CH3). Move the slider until you get the analog reading hovers around 0 V. |
| 12 | CH2 Bipolar, Differential-ended offset | Analog Input | Short circuit | To calibrating differential channel 2 connect channel 4 (CH4) to channel 5 (CH5). Move the slider until you get the analog reading hovers around 0 V. |
| 13 | CH3 Bipolar, Differential-ended offset | Analog Input | Short circuit | To calibrating differential channel 3 connect channel 6 (CH6) to channel 7 (CH7). Move the slider until you get the analog reading hovers around 0 V. |
| 14 | Gain | Analog Inputs | Voltage Source | Connect a very accurate 10 V source across channel 0 (CH0) to analog ground (AGND). Move the slider until you get the analog reading to hover around 10 V. |
| 15 | CH0 offset | Analog Output | Multimeter | Connect a voltmeter that can measure accurately down to 1 mV over analog output 0 (CHO) and analog ground (AGND). Move the slider until the voltmeter reading hovers around 0V. |
| 16 | CH1 offset | Analog Output | Multimeter | Connect a voltmeter that can measure accurately down to 1 mV over analog output 1 (CH1) and analog ground (AGND). Move the slider until the voltmeter reading hovers around 0V. |
| 17 | CH0 gain | Analog Output | Multimeter | Connect a voltmeter that can measure accurately down to 1 mV over analog output 0 (CH0) and analog ground (AGND). Move the slider until the voltmeter reading hovers around 8V. |
| | CH1 gain | Analog Output | Multimeter | Connect a voltmeter that can measure accurately down to 1 mV over analog output 1 (CH1) and analog ground (AGND). Move the slider until the voltmeter reading hovers around 8V. |
| 18 | Save Calibration Values | All | None | To save these settings click on 'Save' or click on 'Exit' to discard changes. |

**Table 5-4 Calibration Procedure**

## Calibration Validity and Operating Conditions

The μDAQ-lite will perform as specified when operating under normal conditions as set in the specification appendix. However there are conditions where the device can behave outside these preset specifications. The following has an effect on the accuracy of the device. It would be good practice to recalibrate the device specifically for this environment.

| No | Condition | Remedy |
|---|---|---|
| 1 | One year since last calibration. | It is advisable that the device be recalibrated every year if highest accuracy is required. |
| 2 | Harsh operating conditions. | If the unit operates in a harsh area, like factories, it is advisable that the unit recalibrated each year (1) and service every five (5) years |
| 3 | High/Low temperatures | Extreme temperature can effect to operation of the device. The identities of the analog circuit will certainly changes under extreme temperatures. The solution would be to recalibrate the unit within these conditions |
| 4 | Above average humidity | If the device operates in high humidity it can cause the unit to degrade in performance over time. The device needs to be calibrated and serviced more frequently. |
| 5 | Exceeding analog input/output specifications | If the device was driven outside its operating region it can affect the accuracy of the device. It would be best practice to recalibrate or in severe case to service the device. |
| 6 | Lightning strike | In the case of such an event the device need to be checked and tested by the manufacturer to prevent costly secondary damage. |
| 7 | Exceeding power input | If the power input was to high or of the wrong type the unit can be severely damaged |
| 8 | Extensive long storage | Make sure the device is stored in a static free environment inside the original packaging. If the device was stored for an extensive period it will need to be recalibrated. |

**Table 5-5 Operating Conditions Voiding Calibration**

## Reset Factory Settings

If you wish to reload the factory calibration settings you can simply click on *"Load Factory Settings"*. This will load the calibration setting that was stored in the device at manufacturing. To save these settings simply click on *"Save"*. These setting will be saved and the program will exit. The new calibration values will be loaded once the device has been restarted.

# A
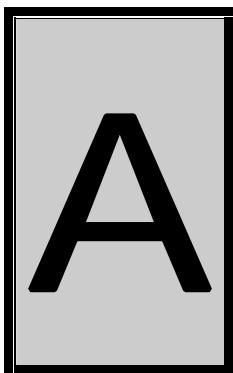
# A.Specifications

## Absolute maximum ratings

| Parameter | Symbol | Condition | Rating | Unit |
|---|---|---|---|---|
| Digital Input Voltage | Vdi | | -0.5 to 5.0 | V |
| Digital Output Voltage | Vdo | | -0.5 to 5.0 | V |
| Digital Output Current | Vdoc | Ta = 25°C with respect to ground | ±20.0 | mA |
| Analog Input Voltage | Vai | | ±35 | V |
| Analog Output Voltage | Vao | | ±10 | V |
| Analog Output Current | Vao | | ±2.0 | mA |
| Storage Temperature | Tstg | - | -50 to 150 | °C |
| Operating Temperature | Tstg | - | 0 to 70 | °C |
| Power Dissipation | Pd | Ta = 25°C | 10.0 | W |

## Digital Input/Output Characteristics

| Parameter | Symbol | Condition | Min. | Typ. | Max. | Unit |
|---|---|---|---|---|---|---|
| Input High | Vih | | 2.0 | | | V |
| Input Low | Vil | | | | 0.8 | V |
| Output High | Voh | Ta = 25°C with respect to ground | 4.9 | 5.0 | | V |
| Output Low | Vol | | | | 0.0 | 0.1 | V |
| Output Source/Sink Current | Io | | | | 20.0 | mA |
| Input Source/Sink Current | Ii | | | | 2.0 | mA |

## Counter-Timer Characteristics

| Parameter | Condition | Min. | Max. | Spec | Unit |
|---|---|---|---|---|---|
| Input High | | 2.0 | 5.25 | | V |
| Input Low | | -0.5 | 0.8 | | V |
| Output High | Ta = 25°C with respect to ground | 2.4 | | | V |
| Output Low | | | 0.8 | | V |
| Clock Source | | | 2 | 50% duty cycle | MHz |
| Resolution | | | | 16 (up counter) | Bit |
| Counter Reload | | 1.5 | 2 | | μSeconds |

| External Clock | Ta = 25°C with respect to ground | | 500 | Falling Edge | KHz |
| Output Pulse (High) | | 0.9 | 1.0 | | μSeconds |
| Output Frequency | | | 100 | | KHz |

## Analog Output Characteristics

| Parameter | Condition | Spec | Unit |
|---|---|---|---|
| Number of channels | Ta = 25°C with respect to analog ground | 2 | - |
| Resolution | | 12 | Bits |
| Maximum Output | | +10 | V |
| Minimum Output | | -10 | V |
| Output current | | ±5 | mA |
| Zero offset error | | 2 | mV |
| Full scale error | | 30 | mV |

## Analog Input Characteristics

| Parameter | Condition | Spec | Unit |
|---|---|---|---|
| Number of channels | Ta = 25°C with respect to analog ground | 8 single-ended, 4 differential-ended | - |
| Acquisition speed | | 49 | KHz |
| Resolution | | 12-bit | Bits |
| Input range, single-ended | | ±10 | V |
| Input range, differential-ended | | ±20, ±10, ±5, ±4, ±2.5, ±2, ±1.25, ±1 | V |
| Input impedance | | 2 | MΩ |
| Trigger source | | Software or external (TRIGGER) | |
| Clock source | | Internal or external (master or slave configuration) | - |

## Conversion Characteristics

### External Clock – SLAVE pin

| Parameter | Condition | Min. | Max. | Spec | Unit |
|---|---|---|---|---|---|
| Input High | Ta = 25°C with respect to ground | 2.0 | 5.25 | | V |
| Input Low | | -0.5 | 0.8 | | V |
| Maximum Rate | | | 35 | | KHz |
| Conversion | | | | Rising Edge | |
| Clock Source | | | | 50% duty cycle | |

### External Trigger – TRIGGER pin

| Parameter | Condition | Min. | Max. | Spec | Unit |
|---|---|---|---|---|---|
| Input High | Ta = 25°C with respect to ground | 2.0 | 5.25 | | V |
| Input Low | | -0.5 | 0.8 | | V |
| External Trigger | | 1.0 | | Rising Edge | μSecond |

### Master Output – MASTER pin

| Parameter | Condition | Min. | Max. | Spec | Unit |
|---|---|---|---|---|---|
| Pulse | Ta = 25°C with respect to ground | 0.9 | 1.0 | | μSeconds |
| Output High | | 4.5 | 5.0 | | V |
| Output Low | | 0 | 0.8 | | V |

## Bus Interface

| | |
|---|---|
| **Bus Type** | Universal Serial Bus Revision 1.1 |
| **Bus Speed** | USB Full Speed – 12 Mega bit per second. |
| **Controller** | USB Serial Interface Endpoint Compliant |
| **Voltage** | 5V |

## Power Requirements

| Device | Typical | Power Source |
|---|---|---|
| UDAQ-lite | 350 mA | USB Power |

## Environmental / Physical

| | |
|---|---|
| **Relative Humidity** | 0% to 90% (non-condensing) |
| **Operating Temperature** | 0°C to 70°C |
| **Housing** | Plastic Casing |
| **Dimension** | Height: 35mm |
| | Width: 80mm |
| | Length: 148mm |

## Connectors

The screw terminal has a hole diameter of 1.15mm – 0.03mm.

# B
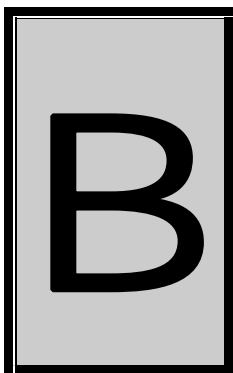
# B.Configuration Constants

## Query Codes

| Name | Value | Description |
|---|---|---|
| APIMAJOR | 1 | Query EDRE API major version number. |
| APIMINOR | 2 | Query EDRE API minor version number. |
| APIBUILD | 3 | Query EDRE API build version number. |
| APIOS | 4 | Query EDRE API OS type. |
| APINUMDEV | 5 | Query number of devices installed. |
| BRDTYPE | 10 | Query a board's type. |
| BRDREV | 11 | Query a board's revision. |
| BRDYEAR | 12 | Query a board's manufactured year. |
| BRDMONTH | 13 | Query a board's manufactured month. |
| BRDDAY | 14 | Query a board's manufactured day. |
| BRDSERIALNO | 15 | Query a board's serial number. |
| DRVMAJOR | 20 | Query a driver's major version number. |
| DRVMINOR | 21 | Query a driver's minor version number. |
| DRVBUILD | 22 | Query a driver's build version number. |
| ADNUMCHAN | 100 | Query number of ADC channel. |
| ADNUMSH | 101 | Query number of samples-and-hold channels. |
| ADMAXFREQ | 102 | Query maximum sampling frequency. |
| ADBUSY | 103 | Check if ADC system is busy. |
| ADFIFOSIZE | 104 | Get ADC hardware FIFO size. |
| ADFIFOOVER | 105 | Check for FIFO overrun condition. |
| ADBUFFSIZE | 106 | Check software buffer size. |
| ADBUFFOVER | 107 | Check for circular buffer overrun. |
| ADBUFFALLOC | 108 | Check if software buffer is allocated. |
| ADUNREAD | 109 | Get number of samples available. |
| ADEXTCLK | 110 | Get status of external clock line – PCI30FG. |
| ADEXTTRIG | 111 | Get status of external trigger line – PCI30FG. |
| ADBURST | 112 | Check if burst mode is enabled. |
| ADRANGE | 113 | Get ADC range. |
| DANUMCHAN | 200 | Query number of DAC channels. |
| DAMAXFREQ | 201 | Query maximum DAC output frequency. |
| DABUSY | 202 | Check if DAC system is busy. |
| DAFIFOSZ | 203 | Get DAC FIFO size. |
| CTNUM | 300 | Query number of counter-timer channels. |
| CTBUSY | 301 | Check if counter-timer system is busy. |
| DIONUMPORT | 400 | Query number of digital I/O ports. |
| DIOQRYPORT | 401 | Query a specific port for capabilities. |
| DIOPORTWIDTH | 402 | Get a specific port's width. |
| INTNUMSRC | 500 | Query number of interrupts sources. |
| INTSTATUS | 501 | Queries interrupt system's status. |
| INTBUSCONNECT | 502 | Connect interrupt system to bus. |
| INTISAVAILABLE | 503 | Check if an interrupt is available. |
| INTNUMTRIG | 504 | Check number times interrupted |

## Error Codes

| Name | Value | Description |
| --- | --- | --- |
| EDRE_OK | 0 | Function successfully. |
| EDRE_FAIL | -1 | Function call failed. |
| EDRE_BAD_FN | -2 | Invalid function call. |
| EDRE_BAD_SN | -3 | Invalid serial number. |
| EDRE_BAD_DEVICE | -4 | Invalid device. |
| EDRE_BAD_OS | -5 | Function not supported by operating system. |
| EDRE_EVENT_FAILED | -6 | Wait on event failed. |
| EDRE_EVENT_TIMEOUT | -7 | Event timed out. |
| EDRE_INT_SET | -8 | Interrupt in use. |
| EDRE_DA_BAD_RANGE | -9 | DAC value out of range. |
| EDRE_AD_BAD_CHANLIST | -10 | Channel list size out of range. |
| EDRE_BAD_FREQUECY | -11 | Frequency out of range. |
| EDRE_BAD_BUFFER_SIZE | -12 | Data passed by buffer incorrectly sized |
| EDRE_BAD_PORT | -13 | Port value out of range. |
| EDRE_BAD_PARAMETER | -14 | Invalid parameter value specified. |
| EDRE_BUSY | -15 | System busy. |
| EDRE_IO_FAIL | -16 | IO call failed. |
| EDRE_BAD_ADGAIN | -17 | ADC-gain out of range. |
| EDRE_BAD_QUERY | -18 | Query value not supported. |
| EDRE_BAD_CHAN | -19 | Channel number out of range. |
| EDRE_BAD_VALUE | -20 | Configuration value specified out of range. |
| EDRE_BAD_CT | -21 | Counter-timer channel out of range. |
| EDRE_BAD_CHANLIST | -22 | Channel list invalid. |
| EDRE_BAD_CONFIG | -23 | Configuration invalid. |
| EDRE_BAD_MODE | -24 | Mode not valid. |
| EDRE_HW_ERROR | -25 | Hardware error occurred. |
| EDRE_HW_BUSY | -26 | Hardware busy. |
| EDRE_BAD_BUFFER | -27 | Buffer invalid. |
| EDRE_REG_ERROR | -28 | Registry error occurred. |
| EDRE_OUT_RES | -29 | Out of resources. |
| EDRE_IO_PENDING | -30 | Waiting on I/O completion |

## Digital I/O Return Query Codes Codes

| Name | Value | Description |
| --- | --- | --- |
| DIOOUT | 0 | Port is an output. |
| DIOIN | 1 | Port is an input. |
| DIOINOROUT | 2 | Port can be configured as in or out. |
| DIOINANDOUT | 3 | Port is an input and an output. |

# C
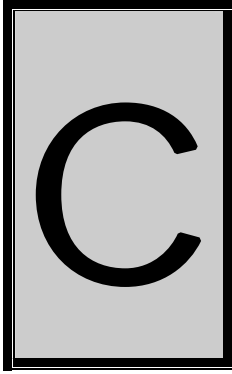
# C.Ordering Information

For ordering information please contact Eagle Technology directly or visit our website www.eagledaq.com. They can also be emailed at eagle@eagle.co.za.