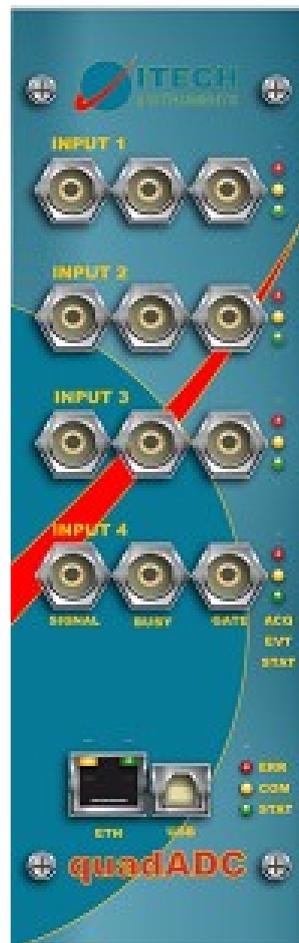


quadADC

Manual



ITECH Instruments, 2010

Table of Contents

1 PC Setup.....	3
1.1 Using the Ethernet interface.....	3
1.1.1 Ethernet connection when using a DHCP server.....	3
1.1.2 Ethernet connection without DHCP server.....	3
1.2 Using the USB interface.....	3
2 The ITECH Device configuration program.....	9
2.1 Device configuration dialog box.....	10
3 Front panel.....	12
3.1 BNC Connectors.....	12
3.2 Ethernet and USB device.....	12
3.3 LED's.....	12
4 Back panel.....	13
4.1 Multifunction connector.....	13

1 PC Setup

1.1 Using the Ethernet interface

1.1.1 Ethernet connection when using a DHCP server

If you have a DHCP server (most DSL routers include one of those) then nothing special has to be done. The PC and the quadADC will receive their IP settings from the DHCP server and the PC will find the quadADC automatically.

1.1.2 Ethernet connection without DHCP server

If the quadADC cannot find a DHCP server then it will set itself to a factory defined address in the range 169.254.x.y.¹

It is possible to simply set the PC on automatic TCP address. Windows will then select automatically an IP address in the same subnet. However this will take some time (1-2 minutes).

The other option is therefore to set the PC on an arbitrary address in the same subnet (e.g.169.254.1.1).

The IP address in the quadADC can be changed in order to adapt it to the standards at the customers site. ITECH delivers a program to do this setup. This program is described in chapter 2 of this manual.

1.2 Using the USB interface

The USB interface is used as “IP over USB” device. This means that the driver software will set up a virtual Ethernet card and then the communication will run using the TCP/IP protocol in the same way as it would run over a real Ethernet cable.

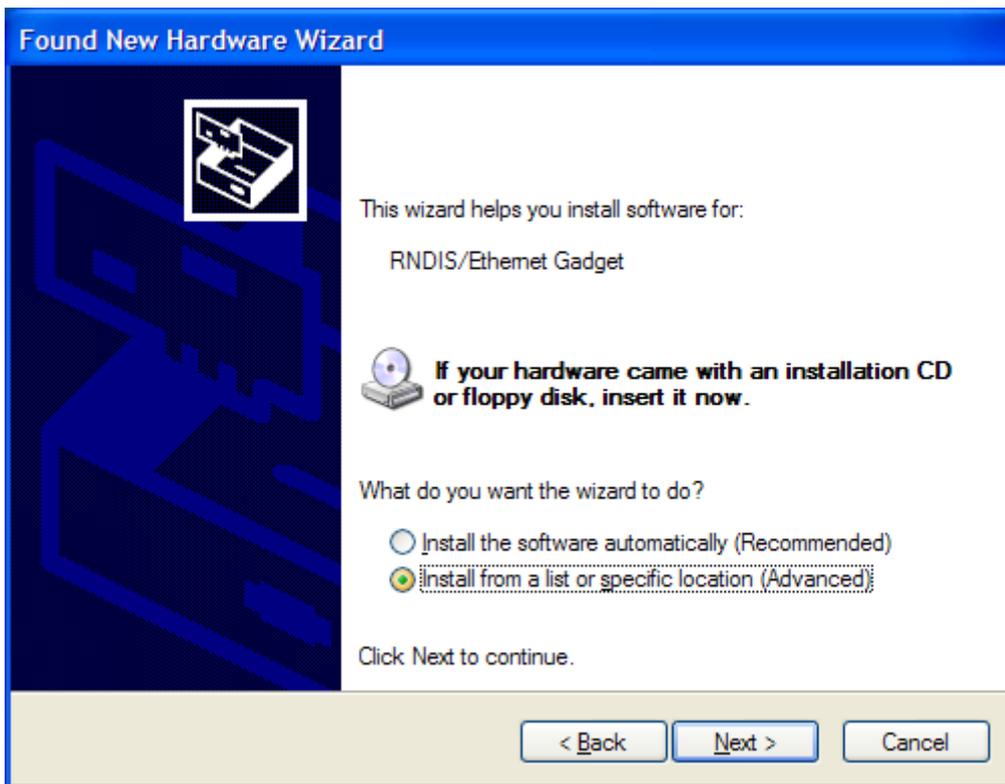
In the following, the process of setting up this USB network is described. The dialog boxes which appear there are produced by Windows and will therefore change with the version and language of the Windows operating system which you are using.

¹ x and y depend on the Ethernet hardware address (MAC-ID) of the quadADC. The address will therefore be different for each device.

After connecting the device the new hardware wizard dialog box will pop up and ask if you want to connect to Windows Update. Select “No” and press Ok.

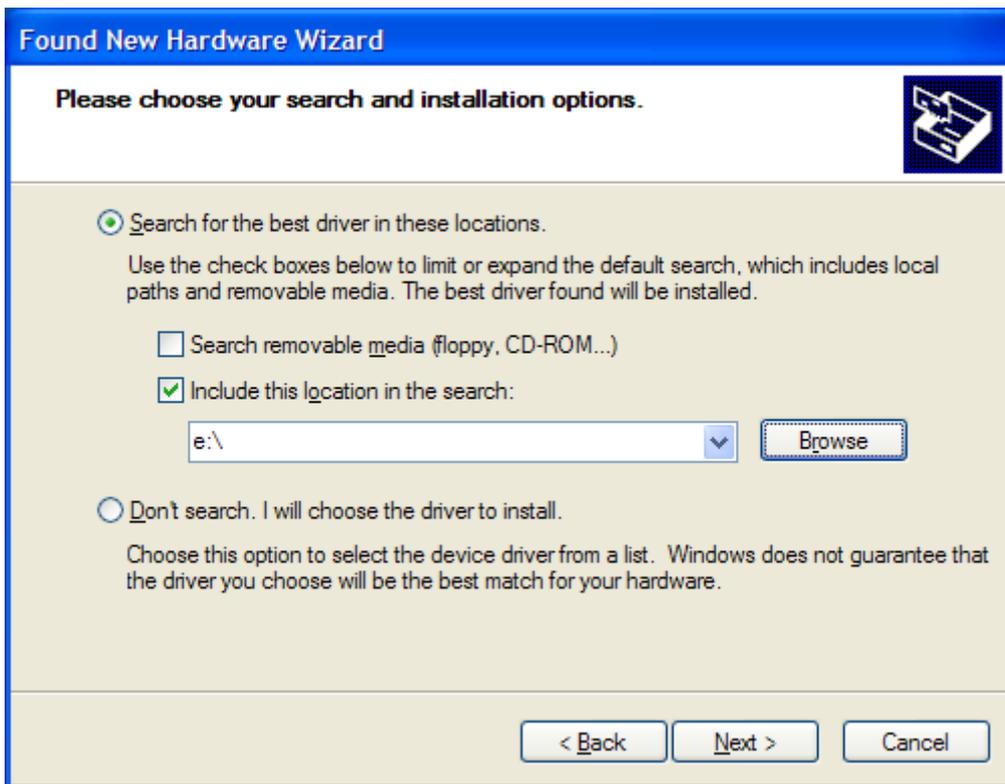


The next box names the device (RNDIS/Ethernet Gadget). Select “Install from a list or specific location” and press ok.



Now, windows will ask where the driver can be found. Make sure that the search path contains the file LINUX.INF. You should have received this device together with the WINNER nuclear spectroscopy software².

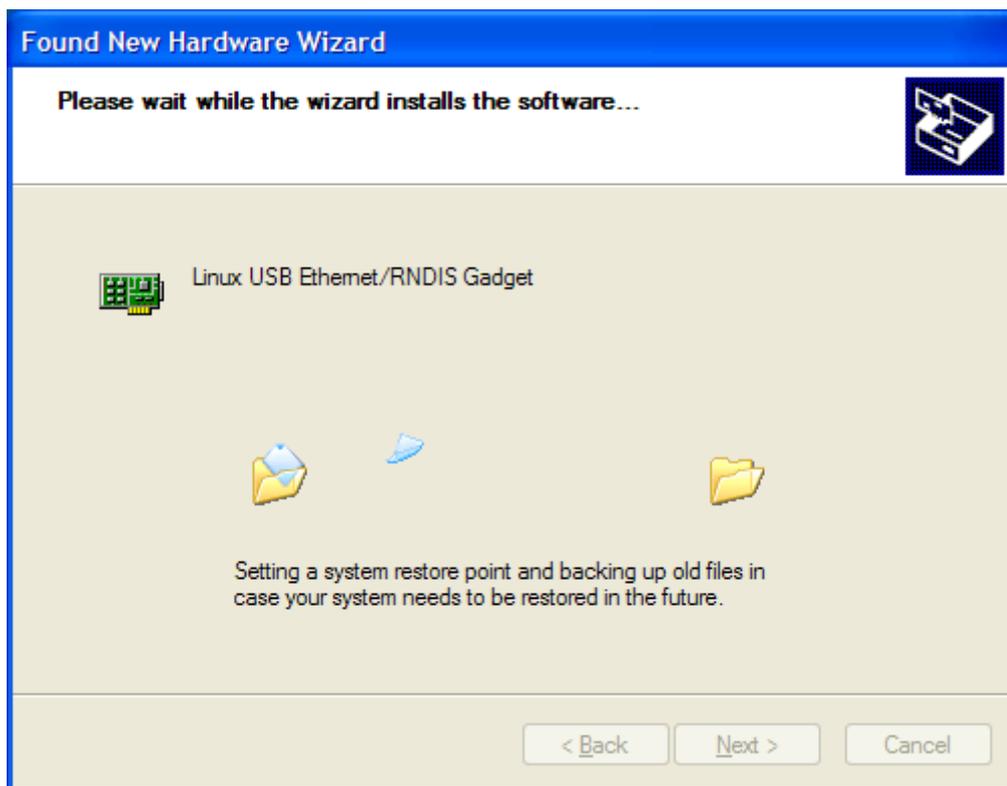
² Otherwise you can download it from <http://www.itech-instruments.com/support>



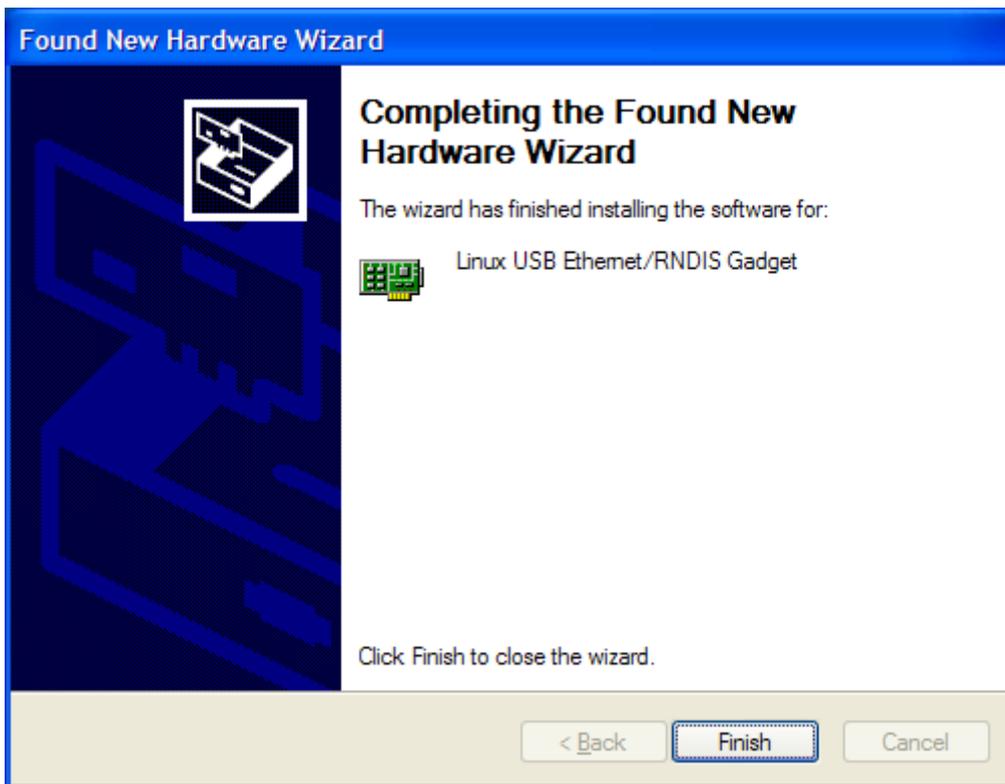
Windows will then display a warning message that the software did not pass the “Windows Logo” test. Select “Continue Anyway”.



Windows will then install the driver. Depending on you PC this may take a while.



A dialog box will inform you about the correct installation of the driver.



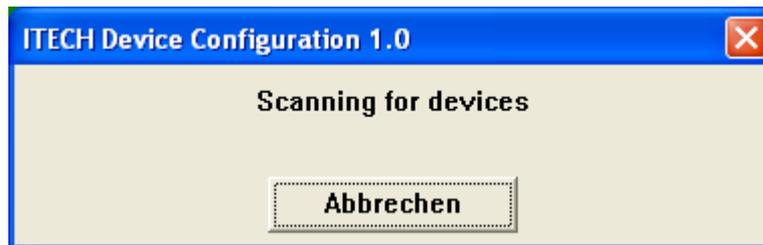
The quadADC includes a minimal DHCP server. Therefore the IP address of the PC will be set automatically. Just leave the IP settings on “Obtain an IP address automatically”.

2 The ITECH Device configuration program

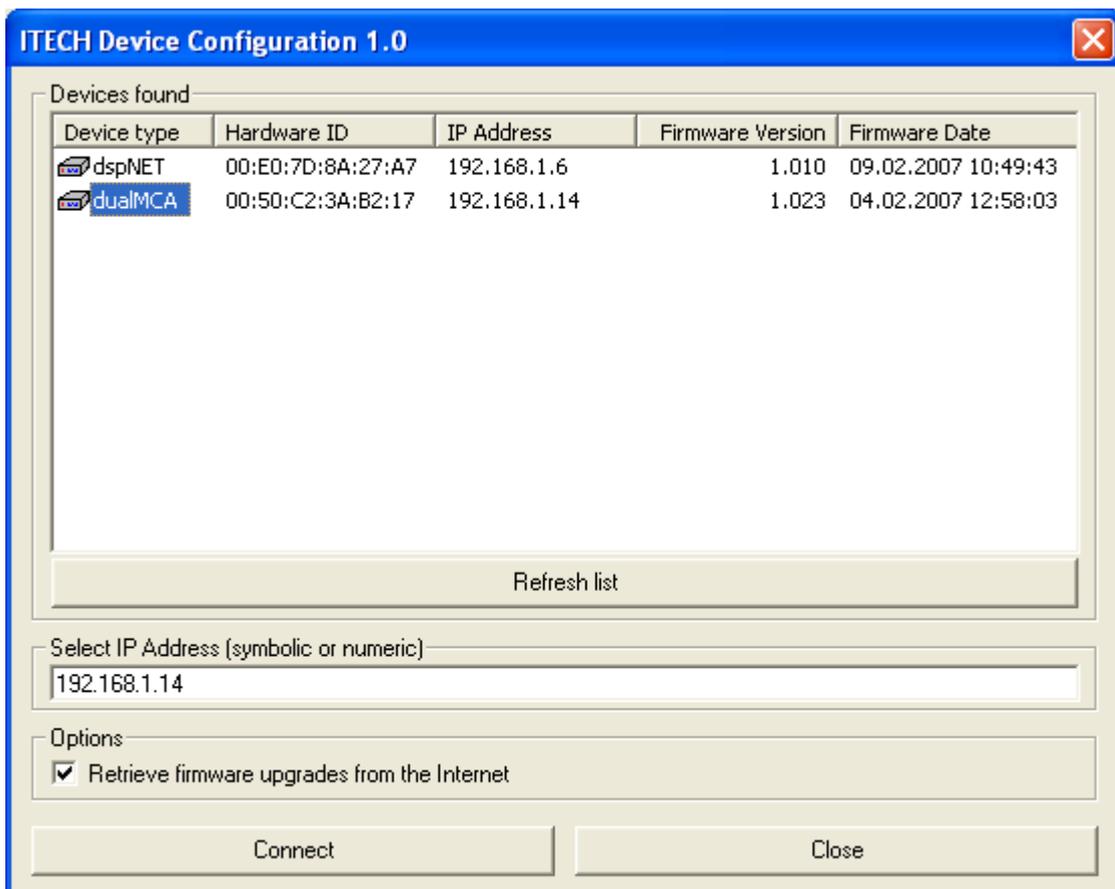
The ITECH Device Configuration program is used to select the IP addresses in ITECH Ethernet connected devices. It is also used to install new firmware versions into the device.

This program can currently be used with the dualMCA, the dspNET, the quadADC, the quadMCS and the quadADC and will be extended for upcoming devices.

When the program is started it first scans for devices present in the current subnet. During the scan the following dialog box is shown.



When the scan is completed a dialog box with the devices found is shown.



The list view shows the devices found together with the detected firmware versions and the IP addresses under which those devices are visible. Below the box is a “Refresh” button which will invoke a new device scan.

The software however can only detect devices in the local subnet. It will of course not search the entire Internet. Therefore there is a IP Address input field so that the user can manually enter the IP of a device reachable over the Internet. It is also possible to use a symbolic name here if the devices is registered by some name server.

In the options box one can select if the program should try to download new firmware upgrades from the ITECH Instruments Web server. Those will be stored in the directory where the configuration program resides if this directory is writable. If this is not the case (e.g. if the software is run from a CD ROM) then the firmware files will be stored in a subdirectory named itech.firmware of the users temporary directory (usually C:\Documents and Settings\User_Name\Local Settings\Temp\itech.firmware)

If the automatic download is not possible then one also can retrieve the firmware images manually (for instance by e-mail) and put them in one of the places shown above.

2.1 Device configuration dialog box

After selecting a device one can press “Connect” in order to connect to this device. After a successful connection the following dialog box will appear:

The screenshot shows a dialog box titled "Device configuration". It is divided into three main sections. The first section, "Device", contains fields for "Device type" (dualMCA), "Hardware ID" (00:50:C2:3A:B2:17), and "Current Firmware version" (1.023), with a button below labeled "Upload a new firmware version". The second section, "Ethernet IP address", has three radio button options: "automatic using a DHCP server", "automatic if DHCP server is found, otherwise manually selected" (which is selected), and "always manually selected IP". Below these are input fields for "IP Selection" (10.0.0.101) and "Gateway". The third section, "USB IP address", has an input field for "IP Selection" (10.0.1.101). At the bottom of the dialog are "OK" and "Abbrechen" buttons.

In the first block the device type and current firmware version are described. A new firmware version can be installed by pressing the corresponding button.

Ethernet IP address

In the next box one can select the IP address used for the Ethernet interface of the module³. There are three different modes:

1) automatic using a DHCP server⁴

³ Each device/computer in a network must use a different IP address.

⁴ A DHCP server distributes the Ethernet addresses to the different computers. Most DSL or WLAN routers include

In this mode the module will wait for a DHCP server during the boot process.

Attention: Do not activate this mode unless you are sure that you have a working DHCP server. The device will not start without server!

2) automatic if DHCP server is found, otherwise manually

In this mode, the device will try to find a DHCP server. If none is found then the boot process continues using the given IP address. This is the default mode for the device.

3) always manually selected IP

In this mode the device will ignore any DHCP server and will always use the selected IP address. Please note that the devices are usually delivered with an IP in the range 169.254.x.y.

USB IP Address

The USB interface of the quadADC uses “IP over USB” as protocol. The default IP address is in the range 10.x.y.58. The x and y depend on the serial number of the device which makes sure that every device is in its own subnet.

such a DHCP server.

3 Front panel

3.1 BNC Connectors

For each inputs the quadADC features three BNC connectors:

- The leftmost connector is the signal input. It accepts gaussian or triangular shaped inputs in the range 0-10V
- The middle connector is a dead time connector. The dead time as seen by the amplifier can be connected here and then will be used for dead time calculation in addition to the internal dead time.
- The last connector is a gate input. The quadADC will add the status of gate as another bit to the measured ADC value. This permits acquiring the spectrum in coincidence and anticoincidence mode at the same time. The PC software will later decide which of the two spectra - or the sum of the spectra in case of “coincidence off” - will be read out. A second gate input per channel is available on the backside connector.

3.2 Ethernet and USB device

The front panel also contains the Ethernet and USB device connector. The wiring of these connector is according to the standards.

3.3 LED's

The quadADC has three LED's per input and another three LED's next too the USB connector.

The signification of the connectors next to the inputs is the following.

- Red: This LED indicates that the acquisition has been started on this input
- Yellow: This LED blinks shortly for each detected event. It therefore provides an rough estimation of the current count rate.
- Green: This LED indicates that the input boards has been detected correctly by the internal processor. It should always be lighted.

The signification of the three LED's next to the USB connector is the following:

- Red: This LED blinks when the devices is selected in the Itech Device Configuration program or in the InterWinner Hardware Setup program. The purpose is to identify a device if more than one device is present in the network. The LED will be light permanently if a severe internal error is detected.
- Yellow: This LED blinks while the quadADC is communicating with the computer.
- Green: This LED confirms correct functioning of the processor board at should always be on.

4 Back panel

4.1 Multifonction connector

The quadADC has a D-SUB37 (female) connector at the back side. This connector can be used as serial interface, gate or trigger input, sample changer control etc. The following table shows the wiring used.

Pin	Name	Purpose
1	GATE0	Gate input 1
2	GATE1	Gate input 2
3	GATE2	Gate input 3
4	GATE3	Gate input 4
5	TRIG0	Trigger #1
6	TRIG1	Trigger #2
7	TRIG2	Trigger #3
8	TRIG3	Trigger #4
9	OUT0	Sample changer output #1-1
10	OUT1	Sample changer output #1-2
11	OUT2	Sample changer output #2-1
12	OUT3	Sample changer output #2-2
13	OUT4	Sample changer output #3-1
14	OUT5	Sample changer output #3-2
15	OUT6	Sample changer output #4-1
16	OUT7	Sample changer output #4-2
17	IN0	Sample changer input #1-1
18	IN1	Sample changer input #1-2
19	+5V	Power supply
20,22,24,26,32,37	GND	Ground
21	IN2	Sample changer input #2-1

23	IN3	Sample changer input #2-2
25	IN4	Sample changer input #3-1
27	IN5	Sample changer input #3-2
28	IN6	Sample changer input #4-1
29	IN7	Sample changer input #4-2
30	TXD1	Serial interface
31	RXD1	Serial interface
33	CTS1	Serial interface
34	RTS2	Serial interface
35	TXD0	Serial debug interface
36	RXD0	Serial debug interface

4.2 USB host connector

The quadADC has a USB host connector on the backside. This is foreseen for future extensions or special applications.

5 LIST MODE data format

At the beginning of the file is a header structure of 32 bytes:

```
struct LISTMODEHEADER
{ // 8 DWORD = 32 Bytes
  DWORD    dwMagic1;    // Always 3327483392
  DWORD    dwMagic2;    // Always 1267282944
  DWORD    DevType;     // Device type: 3 for quadADC, 9=Orion, 10=Venus
  DWORD    Opt;         // Option dword, usually 0
  DWORD    T;           // Time of creation
  float    fwVersion;   // Version of the device firmware used
  DWORD    res[2];      // Reserved
};
```

The events follow this header. They are arranged as 32 bit words.

B31=0: Event

11 Bit Clock in internal units (0.5 μ s for the quadADC)

2 Bit: Number of the input

1 Bit Gate-A/PUR: Status of the PUR input

1 Bit Gate-B (on the DB25): Status of the second gate input

16 Bit channel/energy of the event

B31=1 B30=0 B29=0 B28=0: Clock event. This event appears every 1024 μ s.

12 Bit Dead-Time ADC0: Dead time of ADC0 since the last clock event

4 Bit Real-Time: (Bits 14-11)

12 Bit Dead-Time ADC1: Dead time of ADC1 since the last clock event

B31=1 B30=0 B29=0 B28=1: Second clock event. Always after the first clock event.

12 Bit Dead-Time ADC2: Dead time of ADC2 since the last clock event

4 Bit Real-Time: (Bits 18-15)

12 Bit Dead-Time ADC3: Dead time of ADC3 since the last clock event

B31=1 B30=0 B29=1: Trigger event

B28=0: reserved

B27=0: reserved

11 Bit Clock in internal units

8 Bit Status of the inputs

8 Bit Status of the inputs before this event