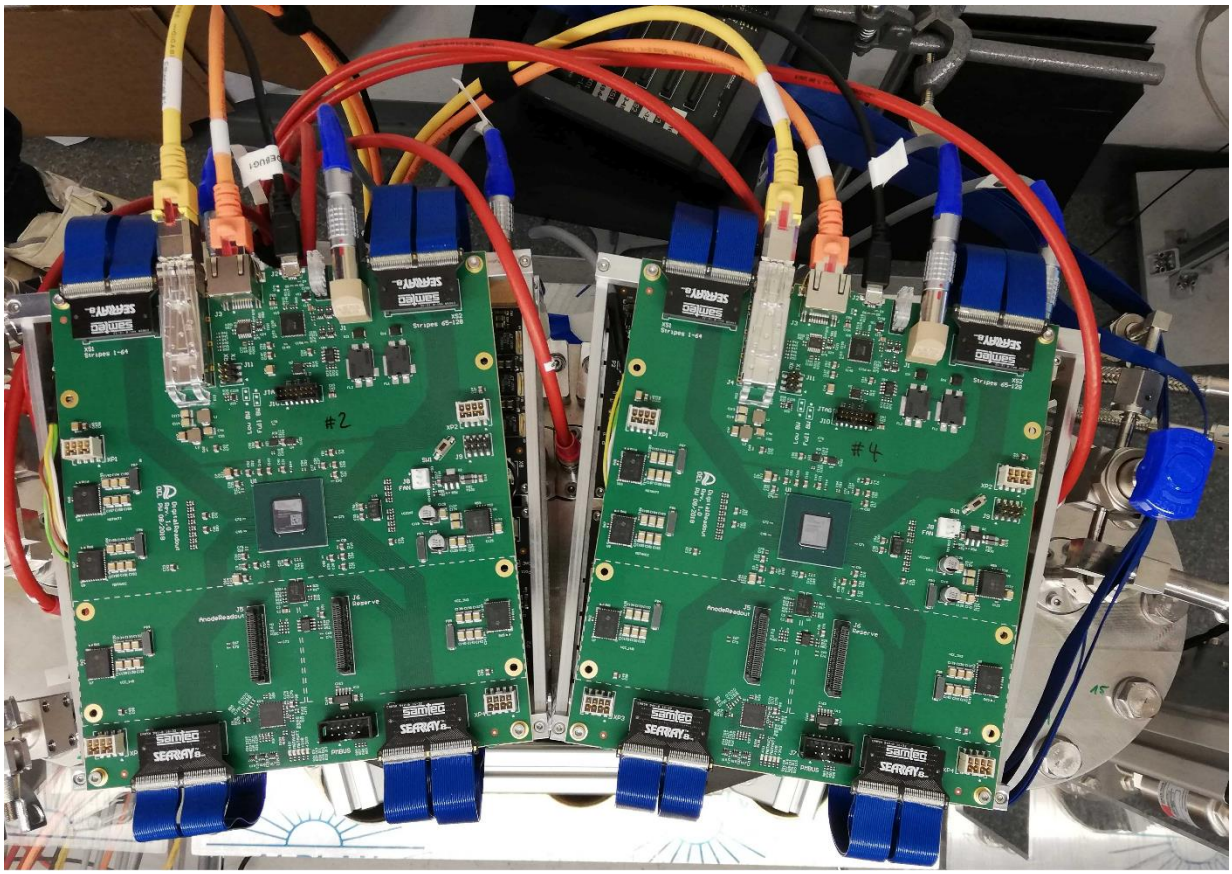
FRM II
Forschungs-Neutronenquelle
Heinz Maier-Leibnitz

MLZ
Heinz Maier-Leibnitz Zentrum

# ErwiN Detector – Readout Electronic



# Table of Contents

# List of Figures

# 1 Introduction

# 2 Hardware

## 2.1 Frontend Electronic

### 2.1.1 PreShape PCB

The *PreShape* board has 16 channels for single wire readout of the detector.

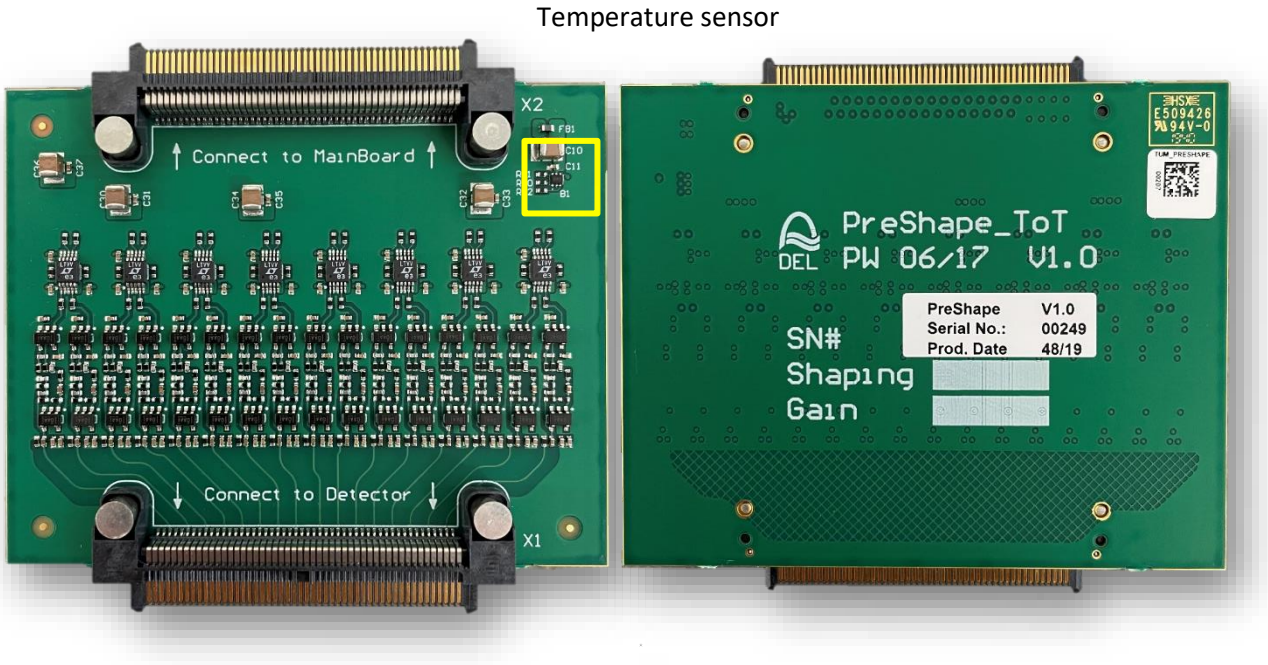



**Figure 1: PreShape board, top & bottom view**

Each channel consists of a charge sensitive amplifier, Gaussian shaper and a comparator for the time-over-threshold (ToT) conversion.

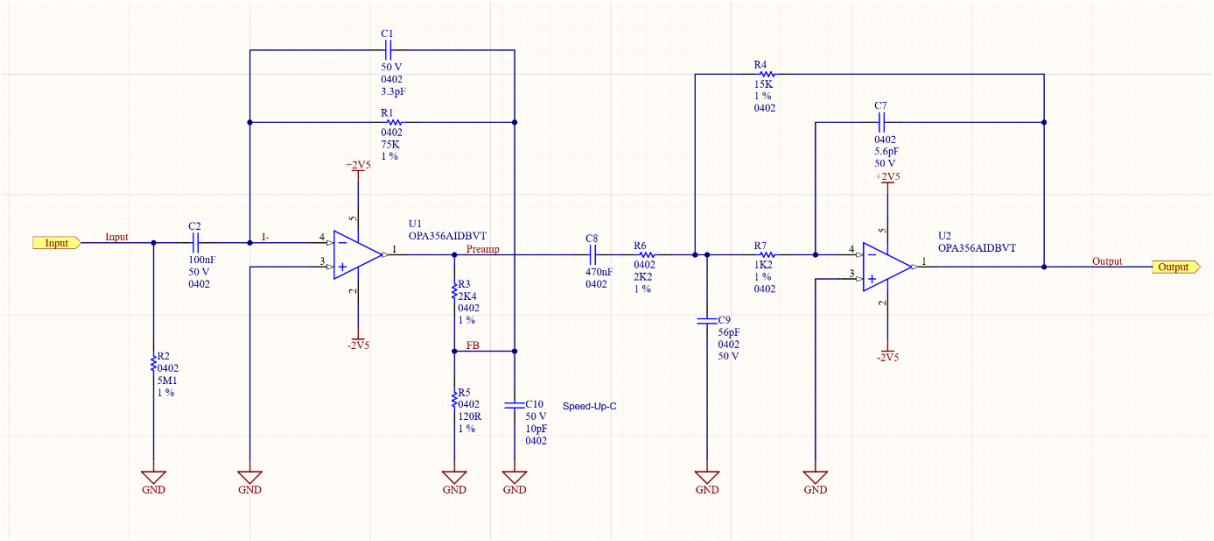



**Figure 2: PreShape analog part - charge sensitive amplifier + shaper**

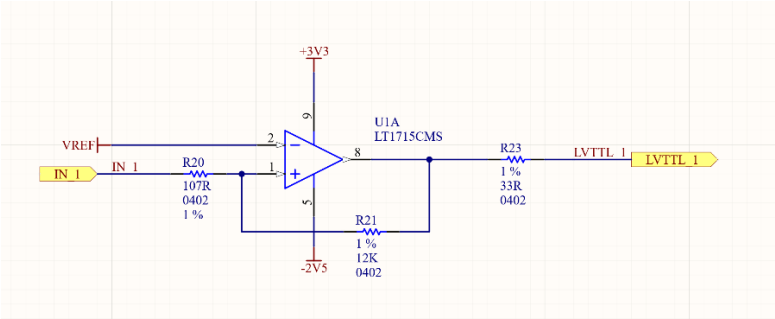



**Figure 3: PreShape digital part - ToT conversion**

The ToT signals are routed through the MainBoard to the DigitalReadout board, where they get processed into neutron events in the FPGA.

The reference voltage for the ToT conversion comes from the MainBoard.

In addition to the ToT signals, the signals after the analog section are also routed to the MainBoard where they are available for debugging.

For an overview of the temperature distribution inside the detector shielding, there is a temperature sensor on each *PreShape* board, which gets read out by the FPGA on the *DigitalReadout* board.

### 2.1.2 MainBoard PCB

The *MainBoard* PCB has four slots for *PreShape* boards on the bottom side.
The analog and ToT signals coming from the four *PreShape* boards are routed separately to one Samtec connector each.
For digital readout, the signals from the *XRD* connector are routed by a Samtec cable to the *DigitalReadout* board above.

The required voltages for the *PreShape* board are generated on the *MainBoard*.
The Lemo connector *XP1* is used for the power supply. The connected voltages are also available at the connectors *X7* and *X8* and can be used to loop through the voltages to another *MainBoard* or to connect sense lines.

The reference voltage for the ToT signal conversion on the *PreShape* board is also generated on the *MainBoard*. To set the voltage, there are two different configuration options. (See chapter 2.1.3)

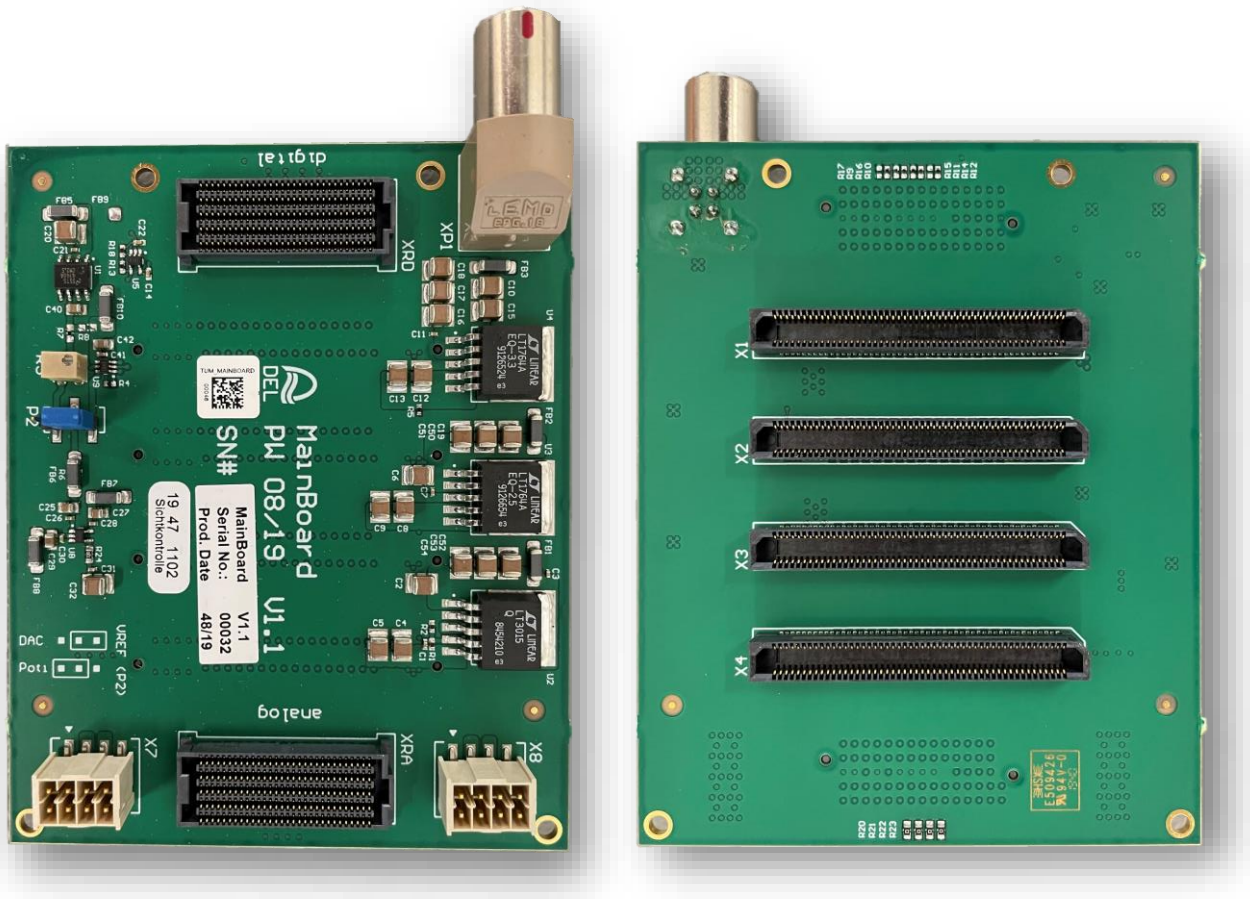



**Figure 4: MainBoard, top & bottom view**

### 2.1.3 Setting ToT threshold voltage

#### 2.1.3.1 Configuration 1 – Adjustment by potentiometer

The reference voltage for the ToT signal conversion can be set manually by the potentiometer *R3*.
By using this way of configuration, the threshold voltage can be adjusted between 0 - 825 mV.
To increase the threshold voltage, turn the adjusting screw of the potentiometer counterclockwise.
To decrease it, turn it clockwise.
The adjusted voltage can be measured at the capacitor C32.
In delivery state, the threshold voltage is set to 200mV.

The Jumper *P2* has to be plugged into the position for the potentiometer.

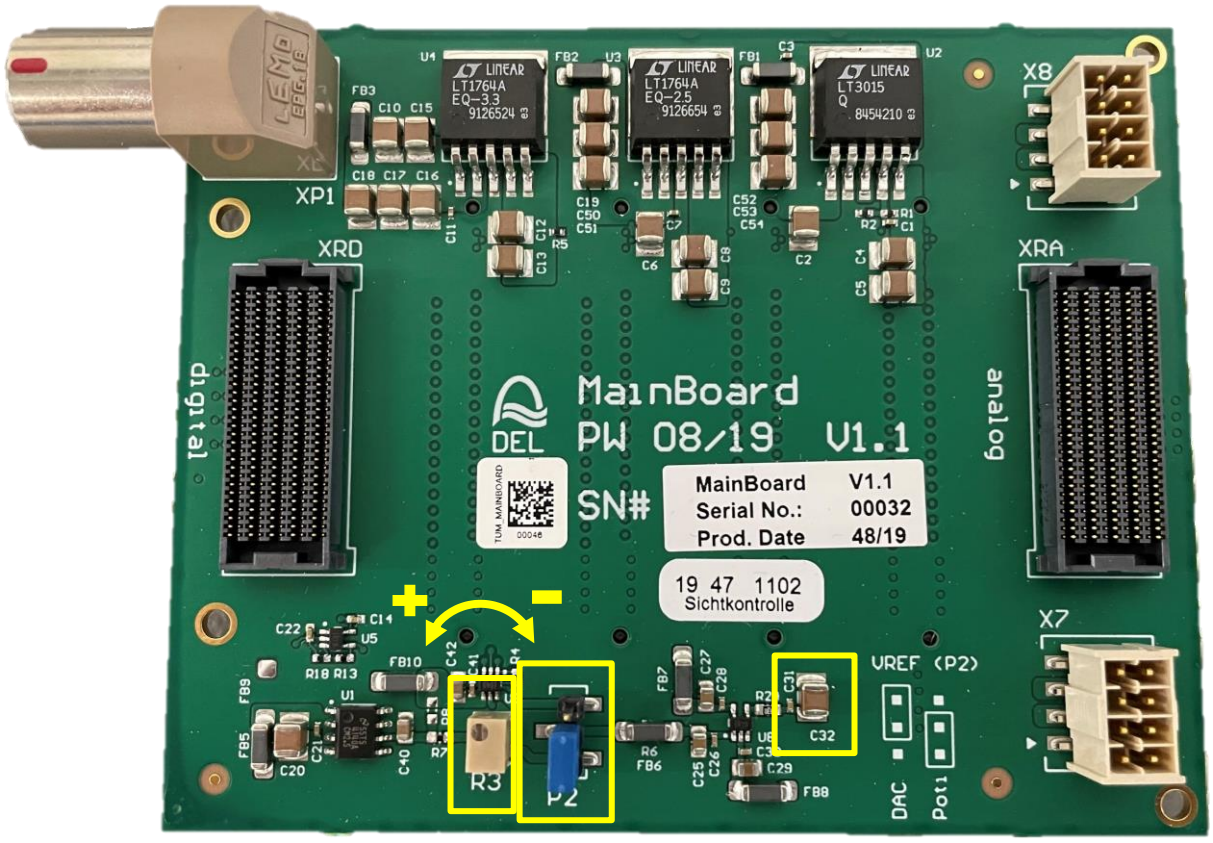



**Figure 5: Setting ToT threshold voltage - potentiometer**

### 2.1.3.2 Configuration 2 – Adjustment by DAC

To set the reference voltage by software, jumper *P2* has to be plugged in the position for the DAC. The voltage can then be set between 0 - 1 V by means of a 10-bit DAC.

$$V_{ref} = 3{,}3V \quad \Rightarrow 1\ \text{LSB} \triangleq 3{,}22\ \text{mV}$$

In delivery state, the default configuration value for the threshold voltage is set to 200mV.

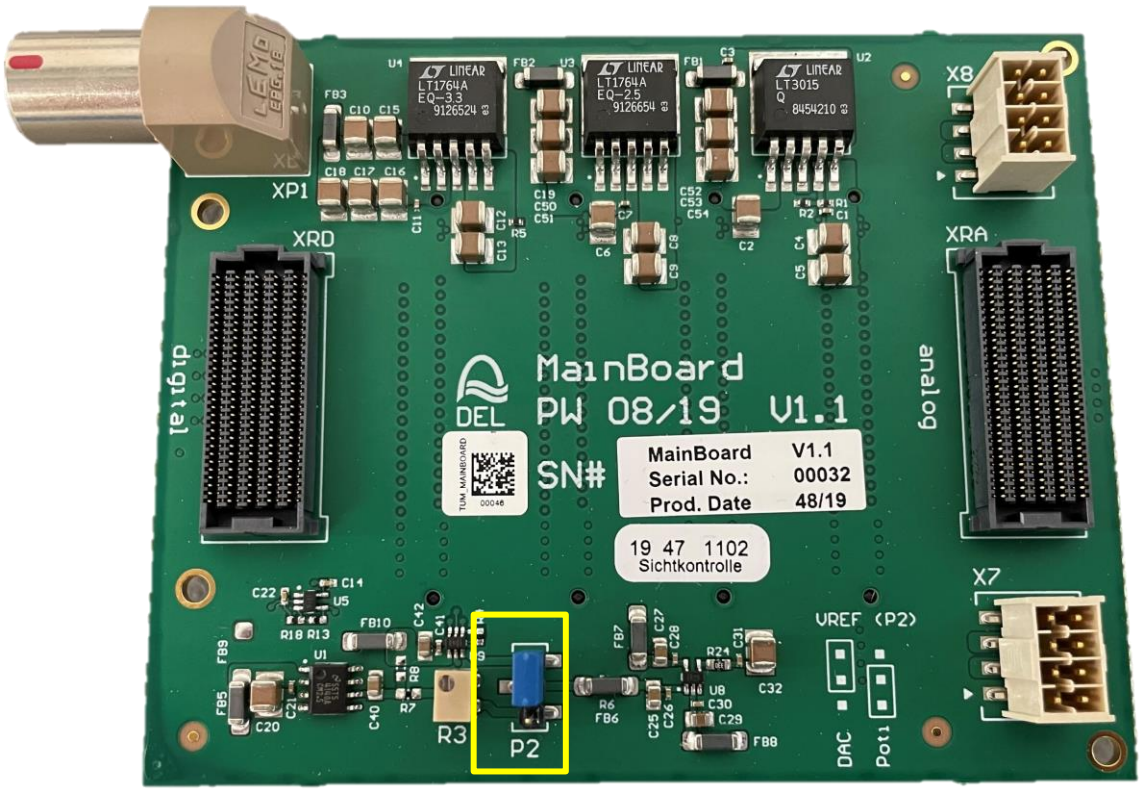
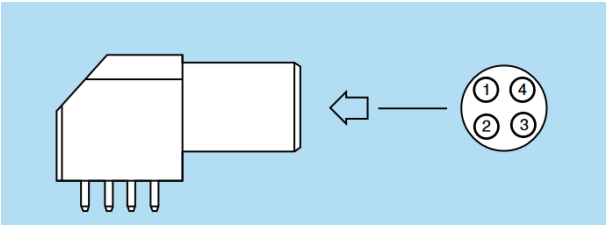


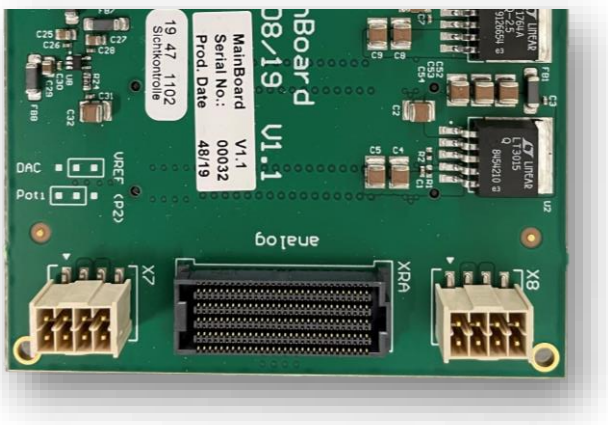

**Figure 6: Setting ToT threshold voltage - DAC**

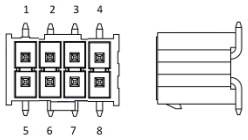
### 2.1.4 Power Supply

| PCB *MainBoard* | Connection cable |
|---|---|
| Connector *XP1* | |
| Lemo EPG.1B.304.HLN | Lemo FGG.1B.304.CYCD52Z |




| Pin assignment *XP1* | |
|---|---|
| 1 | GND |
| 2 | + 3,8 V |
| 3 | - 3,0 V |
| 4 | GND |
| Shield | GND |




| Pin assignment *X7 & X8* | |
|---|---|
| 1 | GND |
| 2 | Potential *XP1* – Pin 2 |
| 3 | Potential *XP1* – Pin 3 |
| 4 | GND |
| 5 | GND |
| 6 | Potential *XP1* – Pin 2 |
| 7 | Potential *XP1* – Pin 3 |
| 8 | GND |

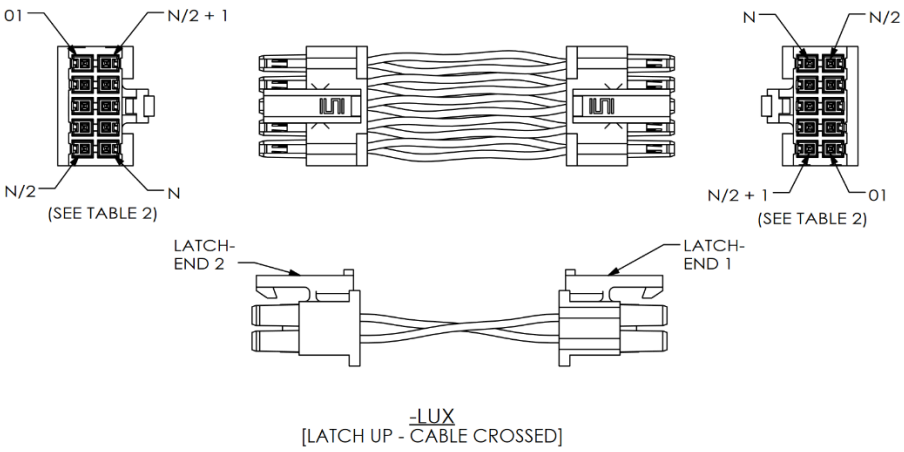



**Figure 7: Cable for looping the power supply voltages**
**MMSDT-04-20-L-03.25-D-K-LUX**

### 2.1.5 Assembling of the frontend modules

The detector is divided into nine segments. The readout electronics for each segment consists of four frontend modules.

Each frontend module consists of four *PreShape* boards with 16 channels each and one *MainBoard*. Together there are 128 channels for the readout of the X-axis and 128 channels of the Y-axis per segment.

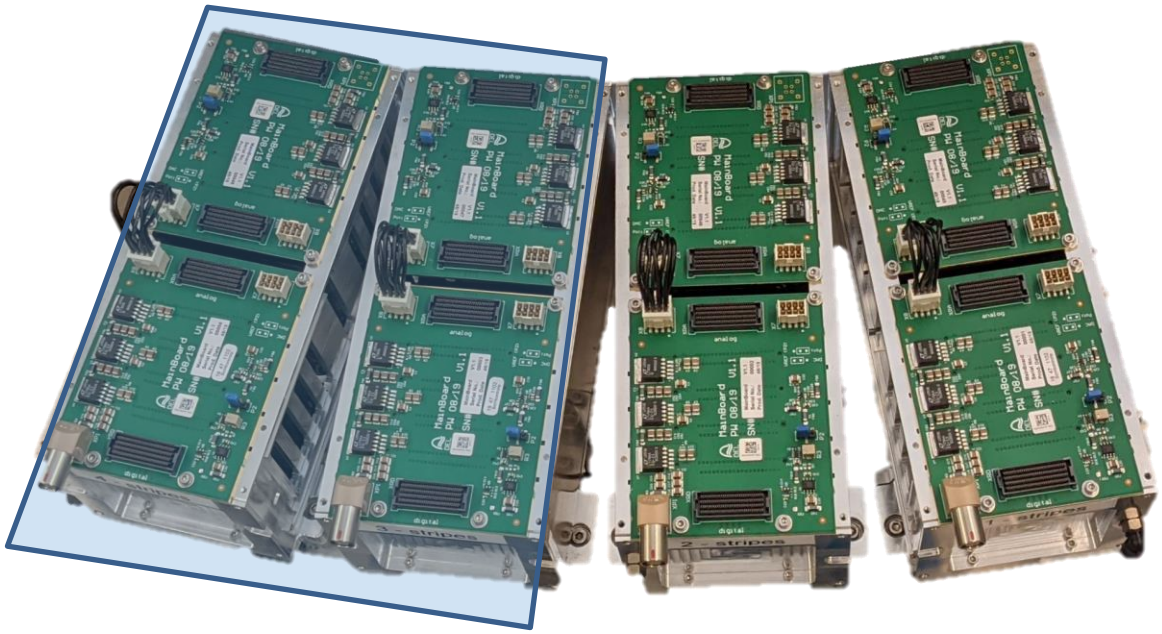


**Figure 8: Frontend module**

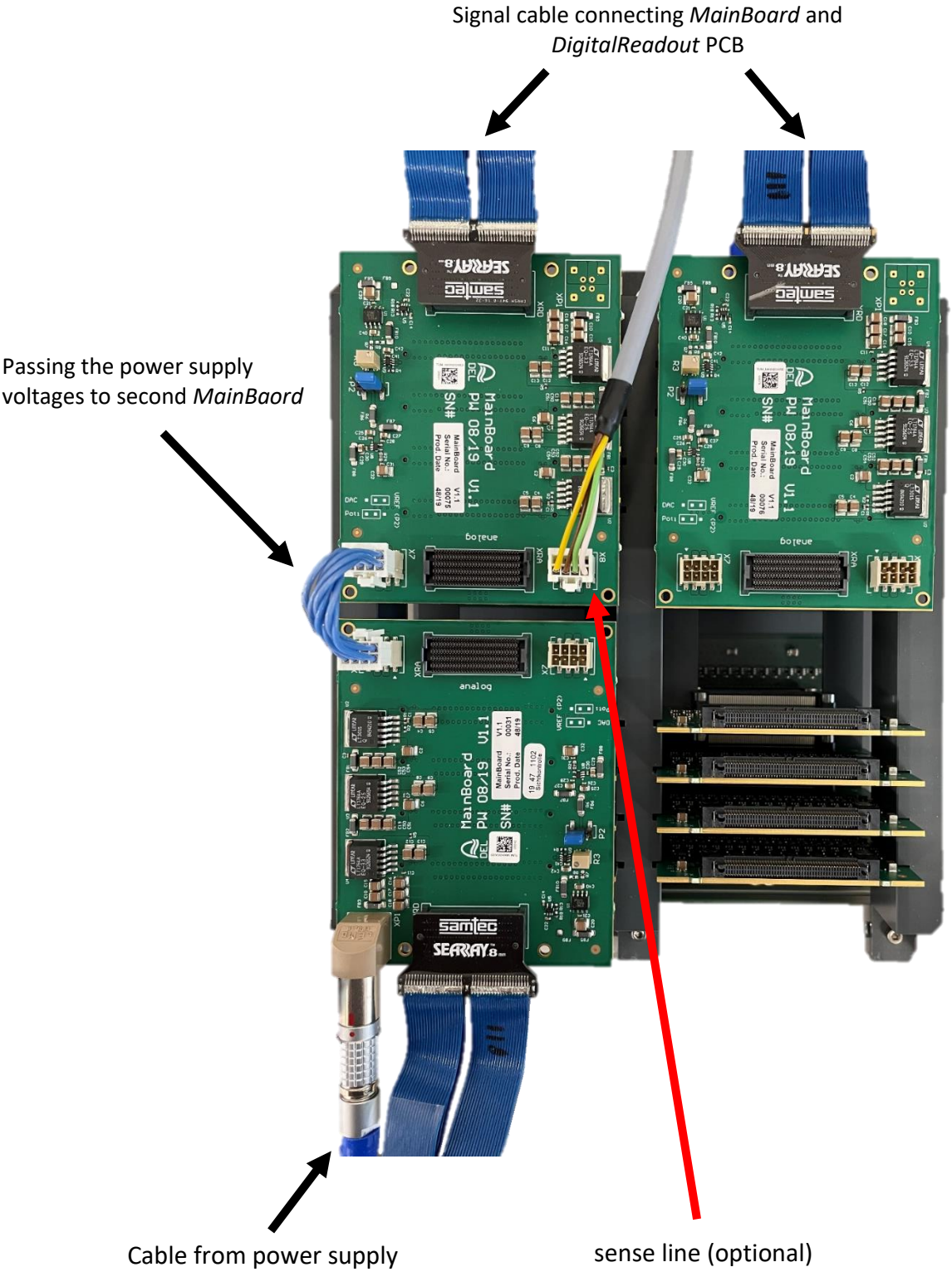
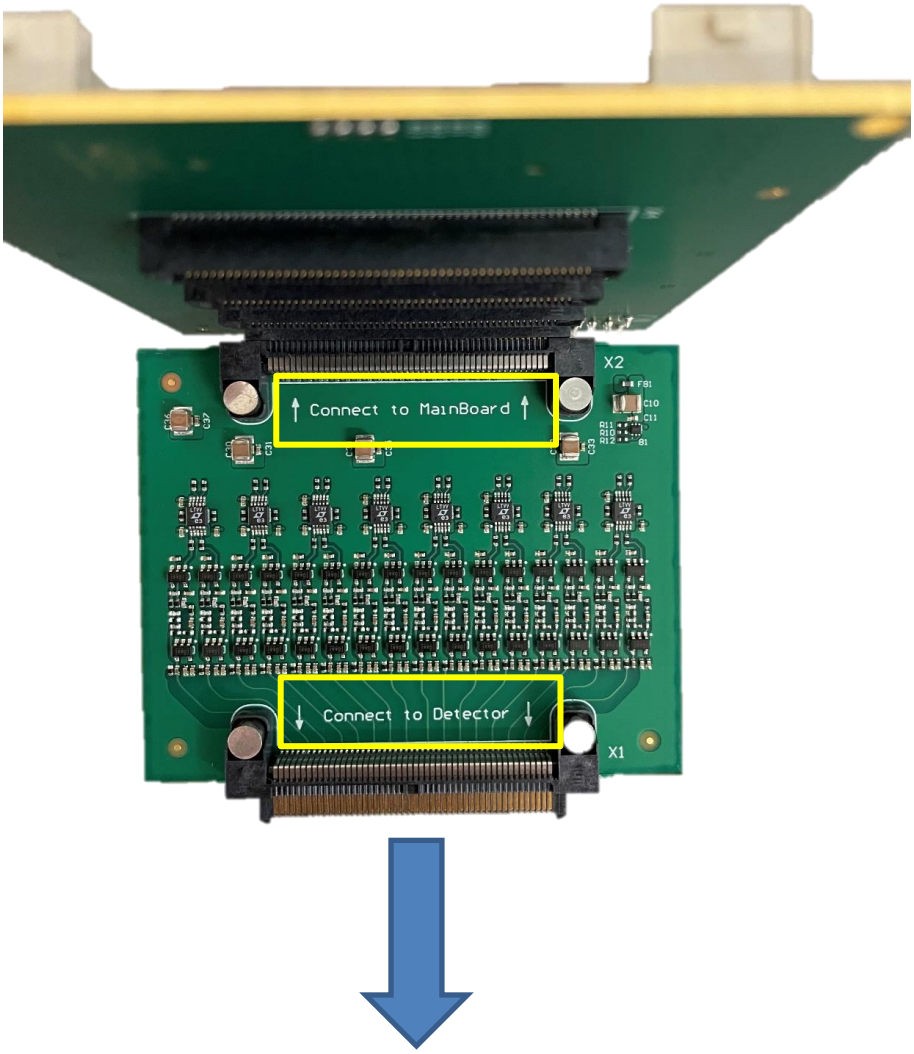



**Figure 9: Setup and cabling frontend module**




Samtec connector on FlexRigid PCB / detector multipin feedthrough

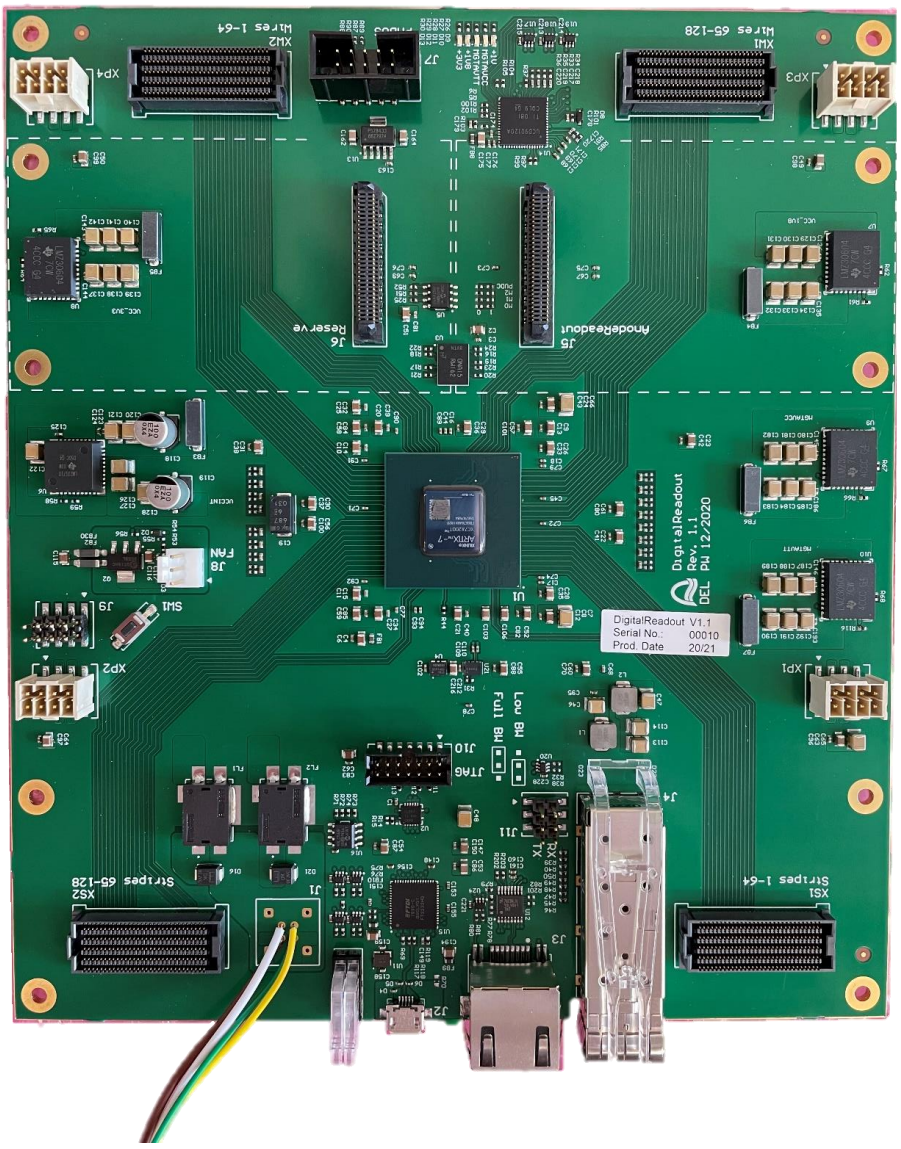
**Figure 10: Connection frontend module with multipin feedthrough**

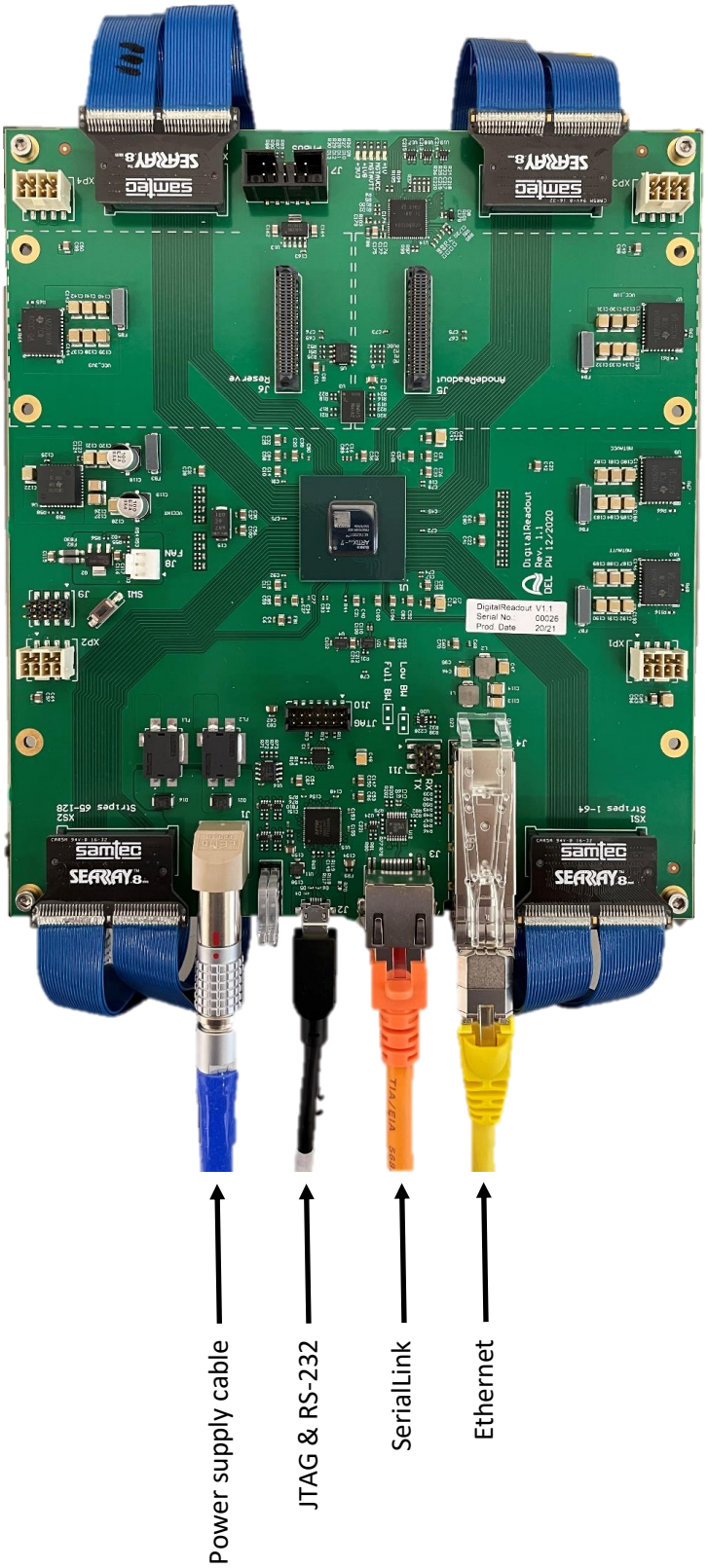
## 2.2 Digital Signal Readout

### 2.2.1 DigitalReadout PCB

The DigitalReadout board on top of every segment is used for the ToT signal sampling and the signal processing.
The ToT signals from the four frontend modules of every segment come through the XS1, XS1, XW1 and XW2 connectors and are then processed in the FPGA and sent to the *Correlation Unit* as pre-correlated events via the *SerialLink*.

In addition, the unit has a 1 Gbit/s (1 Gbit/s-only) network interface (SFP module, optionally optical or copper).
It is used as the configuration interface and for sending the values of the temperature sensors and the voltages on the main board to a service PC.

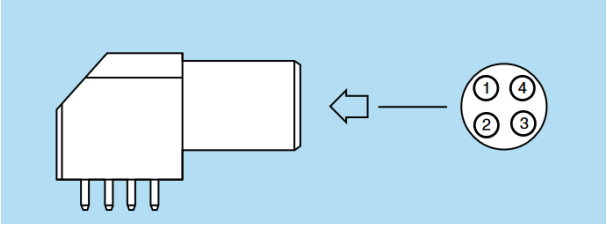

DigitalReadout V1.1
Serial No.: 00010
Prod. Date 20/21
DigitalReadout
Rev. 1.1
DEL PW 12/2020
Full BW
Low BW


Power supply cable
JTAG & RS-232
SerialLink
Ethernet

### 2.2.1.1 Power Supply

| PCB *DigitalReadout* | Connection cable |
|---|---|

| Connector J*1* | |
|---|---|
| Lemo EPG.1B.304.HLN | Lemo FGG.1B.304.CYCD52Z |




| Pin assignment *J1* | |
|---|---|
| 1 | GND |
| 2 | + 5,0 V |
| 3 | - 5,0 V |
| 4 | GND |
| Shield | GND |

| Pin assignment *XP1, XP2, XP3, XP4* | |
|---|---|
| 1 | GND |
| 2 | Potential *J1* – Pin 2 |
| 3 | +3,3 V |
| 4 | GND |
| 5 | GND |
| 6 | Potential *J1* – Pin 3 |
| 7 | +3,3 V |
| 8 | GND |

## 2.2.2 Correlation Unit

The *Correlation Unit* is the central module of the detector readout.
Here, the pre-correlated events of each segment are merged and then post-correlated again for the boundary areas between two segments.
The pre-correlated event data from the nine segments is send over the *SerialLink* to the *Correlation Unit* and is then send to the PC via the Ethernet interface (PC-IF).

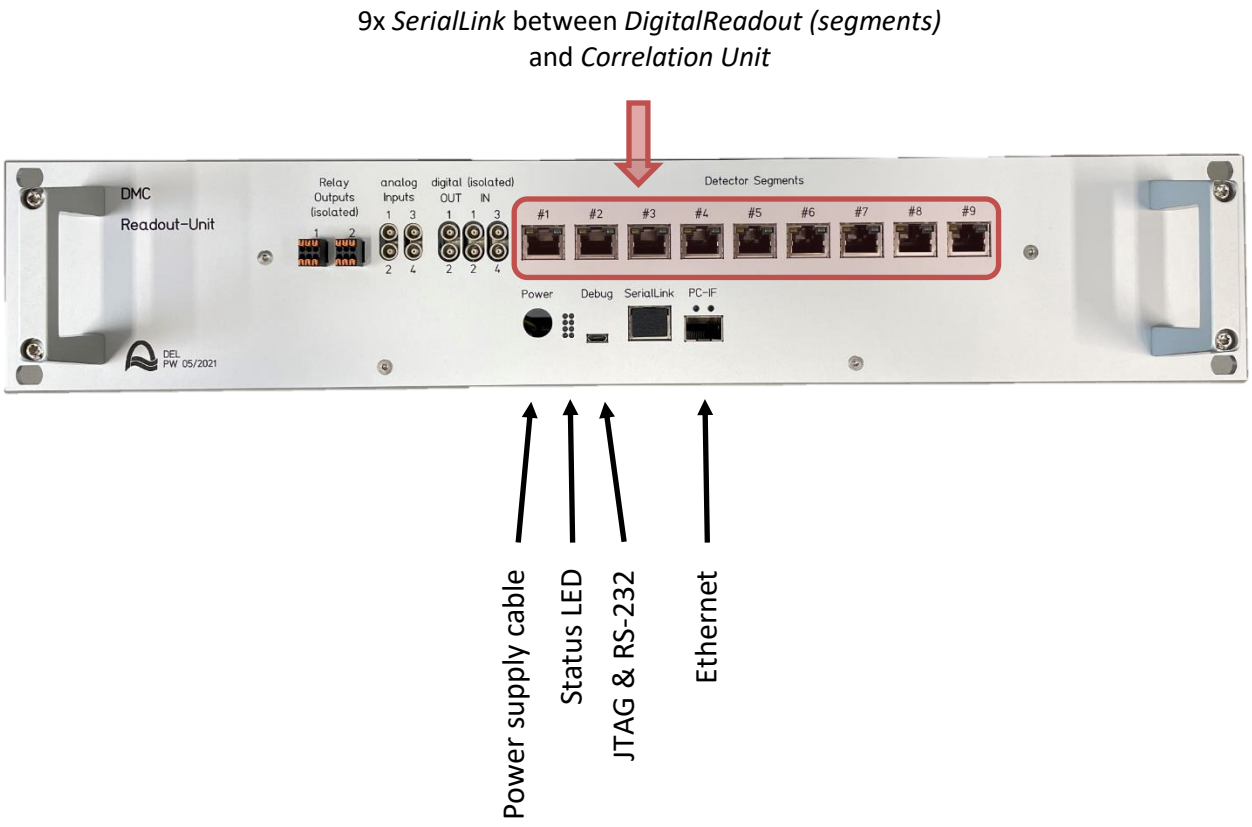
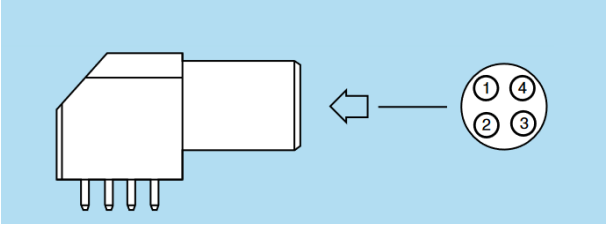



**Figure 11: Correlation Unit - Front plate view**

### 2.2.2.1 Power Supply

| Module *Correlation Unit* | Connection cable |
|---|---|
| Connector *J1* | |
| Lemo EPG.1B.304.HLN | Lemo FGG.1B.304.CYCD52Z |




| Pin assignment *J1* | |
|---|---|
| 1 | GND |
| 2 | + 5,0 V |
| 3 | - 5,0 V |
| 4 | GND |
| Shield | GND |

### 2.2.2.2 Front plate connectors

- ***SerialLink***

  A standard patch cable (no crossover cable) is required to connect the *DigitalReadout* boards and the *CorrelationUnit*.

- ***Digital inputs & outputs (galvanically isolated)***

  | | |
  |---|---|
  | 4x Monitor inputs: | 3,3V, LVCMOS / max. 5,5 V / 50 Ω terminated |
  | 2x Outputs: | 3,3V, LVCMOS / needs to be terminated with 50 Ω |

- ***Analog inputs***

  | | |
  |---|---|
  | 4x Inputs: | 0 - 10V, Attenuation 10:1 (voltage divider) |

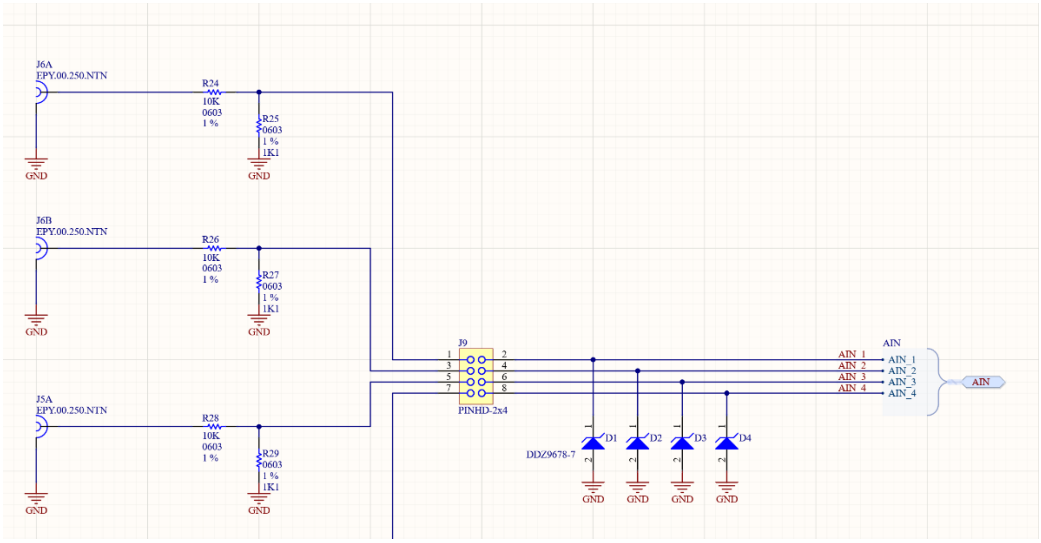



**Figure 12: Analog inputs**

- ***Relays***

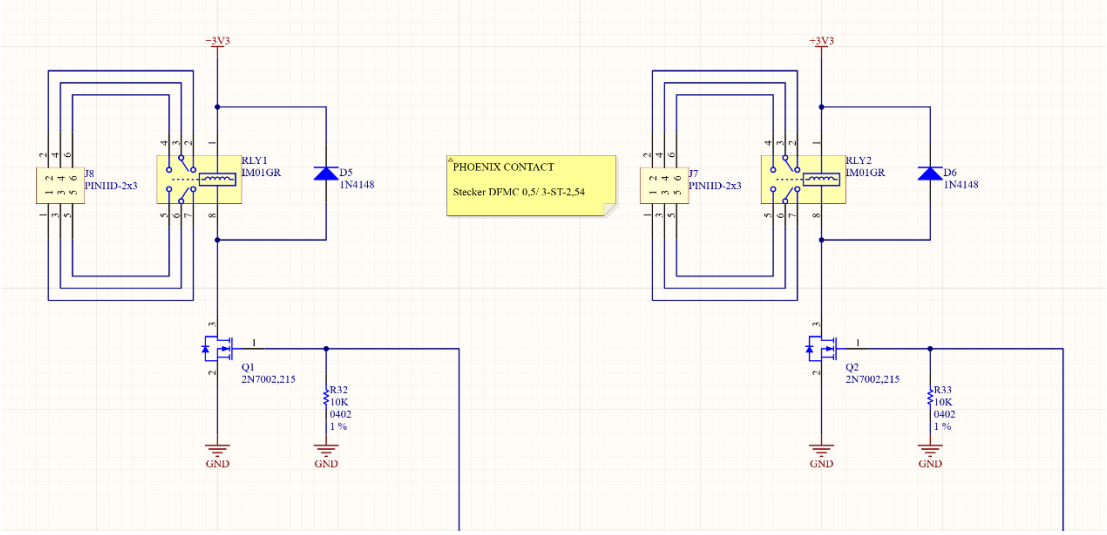



**Figure 13: Relay outputs**

# 3 Firmware

## 3.1 Correlation Unit - Block diagram

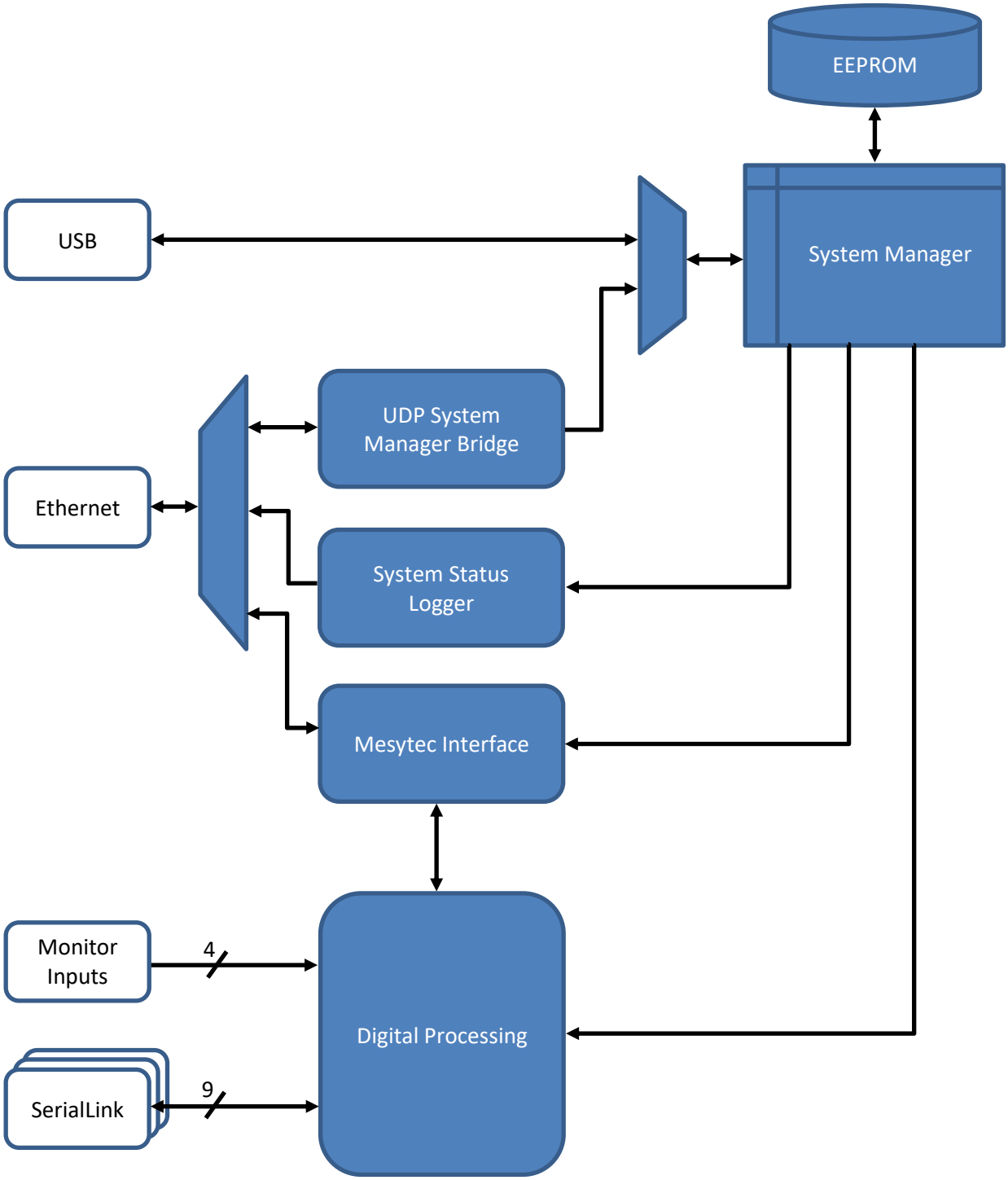



**Figure 14: Block diagram - Correlation Unit**

## 3.2 DigitalReadout - Block diagram

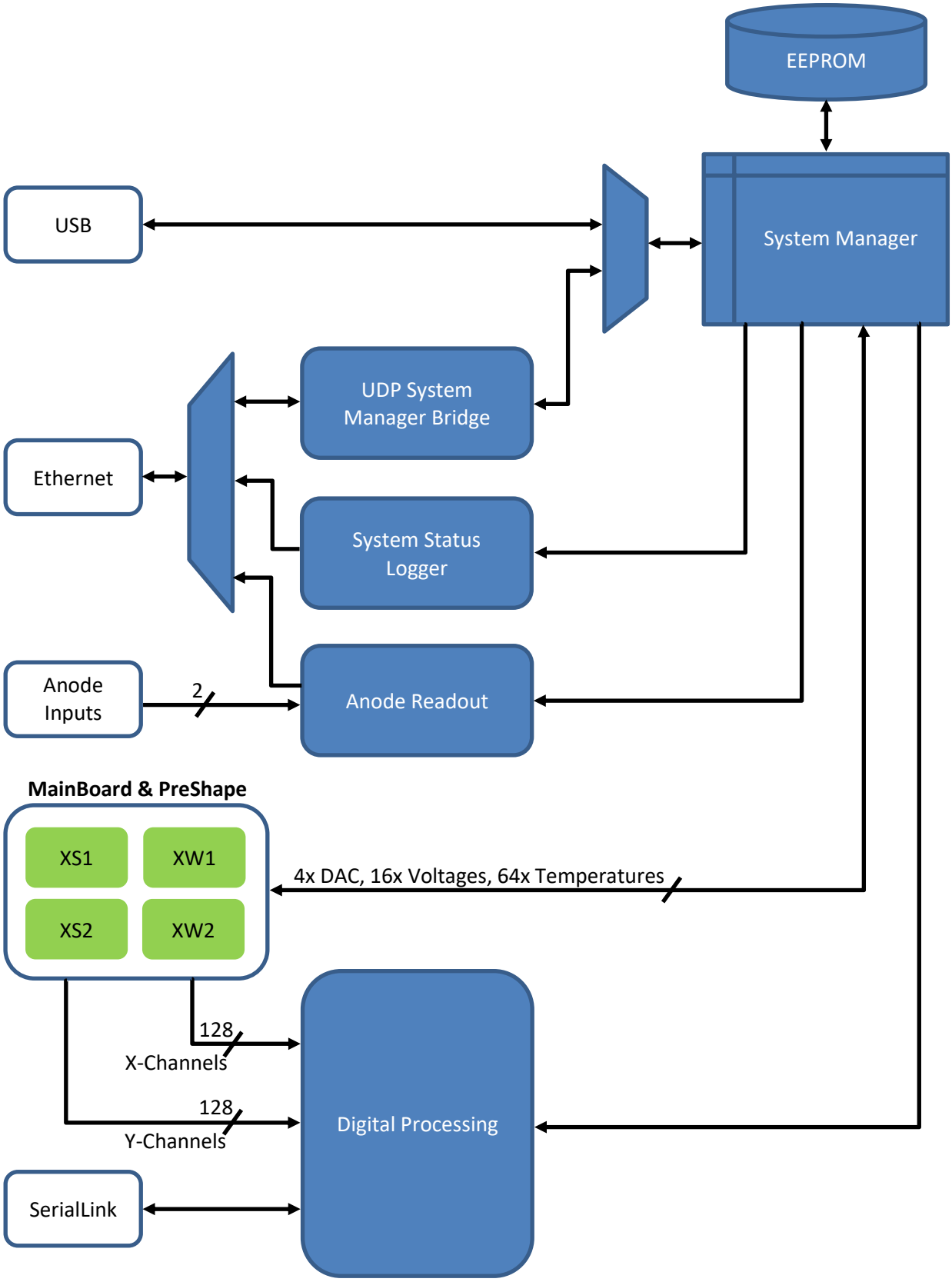



**Figure 15: Block diagram - DigitalReadout**

## 3.3 Register Overview

| | | | System Manager | QMesyDAQ | EEPROM | Correlation Unit | Digital Readout |
|---|---|---|---|---|---|---|---|
| **Device Ethernet Configuration** | MAC Address | 0x00 + 4-Bit subaddress | ✓ | | ✓ | ✓ | ✓ |
| | MAC Address – Set Register | 0x90 + 4-Bit subaddress | ✓ | | | ✓ | ✓ |
| | IP Address | 0x01 + 4-Bit subaddress | ✓ | | ✓ | ✓ | ✓ |
| | IP Address – Set Register | 0x91 + 4-Bit subaddress | ✓ | | | ✓ | ✓ |
| **DAQ Configuration** | Destination IP Address DAQ-Server | 0x02 + 4-Bit subaddress | ✓ | | ✓ | ✓ | |
| | Destination IP Address DAQ-Server – Set Register | 0x93 + 4-Bit subaddress | ✓ | | | ✓ | |
| | UDP Port Eventdata | 0x10 + 4-Bit subaddress | ✓ | | ✓ | ✓ | |
| | UDP Port Command | 0x11 + 4-Bit subaddress | ✓ | | ✓ | ✓ | |
| | MCPD ID | 0x40 + 4-Bit subaddress | ✓ | | ✓ | ✓ | |
| | Run ID | 0x82 + 4-Bit subaddress | ✓ | ✓ | | ✓ | |
| | Segment ID | 0x41 + 4-Bit subaddress | ✓ | | ✓ | | ✓ |
| | Segment Count | 0x42 + 4-Bit subaddress | ✓ | | ✓ | ✓ | |
| | ~~Slave Operation~~ | ~~0x4F + 4-Bit subaddress~~ | ~~✓~~ | | ~~✓~~ | ~~✓~~ | |
| | Frontend ToT Threshold | 0x30 + 4-Bit subaddress | ✓ | | ✓ | | ✓ |
| **System Status Logger** | Destination IP Address | 0x03 + 4-Bit subaddress | ✓ | | ✓ | ✓ | ✓ |
| | Destination IP Address – Set Register | 0x92 + 4-Bit subaddress | ✓ | | | ✓ | ✓ |
| | UDP Port | 0x12 + 4-Bit subaddress | ✓ | | ✓ | ✓ | ✓ |
| | Configuration | 0x20 + 4-Bit subaddress | ✓ | | ✓ | ✓ | ✓ |
| **System Manager** | UDP Port | 0x13 + 4-Bit subaddress | ✓ | | ✓ | ✓ | ✓ |
| **Anode Readout** | Destination IP Address | 0x04 + 4-Bit subaddress | ✓ | | ✓ | | ✓ |
| | Destination IP Address – Set Register | 0x94 + 4-Bit subaddress | ✓ | | | | ✓ |
| | UDP Port | 0x14 + 4-Bit subaddress | ✓ | | ✓ | | ✓ |
| | Control | 0x21 + 4-Bit subaddress | ✓ | | | | ✓ |
| | Threshold | 0x22 + 4-Bit subaddress | ✓ | | ✓ | | ✓ |
| **System Status Information** | Firmware Timestamp | 0x50 + 4-Bit subaddress | ✓ | | | ✓ | ✓ |
| | Firmware Version | 0x51 + 4-Bit subaddress | ✓ | | | ✓ | ✓ |
| | Firmware Git Hash | 0x52 + 4-Bit subaddress | ✓ | | | ✓ | ✓ |
| | System Uptime | 0x53 + 4-Bit subaddress | ✓ | | | ✓ | ✓ |
| | Device Type | 0x54 + 4-Bit subaddress | ✓ | | | ✓ | ✓ |
| | System Status | 0x70 + 4-Bit subaddress | ✓ | | | ✓ | ✓ |
| | DAQ Status | 0x71 + 4-Bit subaddress | ✓ | | | ✓ | |
| | FPGA Temperature | 0x60 + 4-Bit subaddress | ✓ | | | ✓ | ✓ |
| | FPGA Temperature Max | 0x61 + 4-Bit subaddress | ✓ | | | ✓ | ✓ |
| | PreShape Temperatures | 0x62 + 4-Bit subaddress | ✓ | | | | ✓ |
| | MainBoard PSU Voltages | 0x63 + 4-Bit subaddress | ✓ | | | | ✓ |

| | | | System Manager | QMesyDAQ | EEPROM | Correlation Unit | Digital Readout |
|---|---|---|---|---|---|---|---|
| **Command & Control** | EEPROM Command | 0x80 + 4-Bit subaddress | ✓ | | | ✓ | ✓ |
| | DAQ Command | 0x81 + 4-Bit subaddress | ✓ | | | ✓ | |
| **Debug & Raw Data** | ~~Digital Processing~~ | ~~0xA0 + 4-Bit subaddress~~ | ~~✓~~ | | | ~~✓~~ | ~~✓~~ |
| | X Channel Mask | 0xA1 + 4-Bit subaddress | ✓ | | | | ✓ |
| | Y Channel Mask | 0xA2 + 4-Bit subaddress | ✓ | | | | ✓ |
| | Segment Data Path Disable | 0xA3 + 4-Bit subaddress | ✓ | | | ✓ | |
| | Correlation Unit | 0xA4 + 4-Bit subaddress | ✓ | | | ✓ | |
| | Digital Readout | 0xA5 + 4-Bit subaddress | ✓ | | | | ✓ |
| **Front panel I/O** | Front panel Analog Inputs | 0xB0 + 4-Bit subaddress | ✓ | | | ✓ | |
| | Front panel Relay Outputs | 0xB1 + 4-Bit subaddress | ✓ | | | ✓ | |
| | Front panel Digital Outputs | 0xB2 + 4-Bit subaddress | ✓ | | | ✓ | |

## 3.4 Power-up / Register initialization

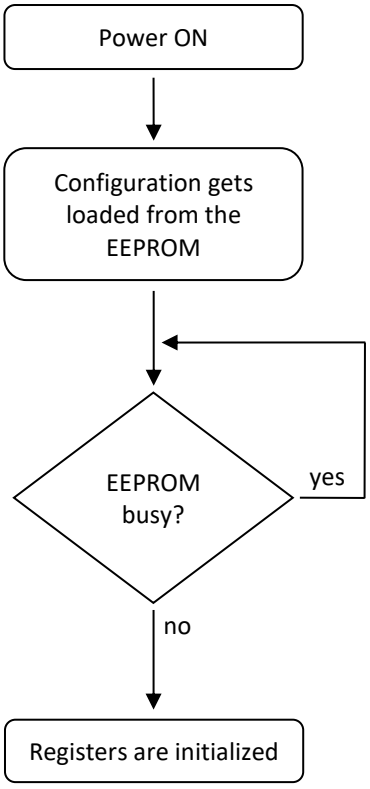



**Figure 16: Power-up / register initialization**

After a power-up, the registers are initialized with the values stored in the EEPROM.
During this process, there should be no write accesses to the registers.

## 3.5 Register Description

atten

### 3.5.1 Device Ethernet Configuration

#### 3.5.1.1 MAC Address

Register: **MAC Address**
Address: 0x00 + 4-Bit subaddress
Type: Read-only
Description: Subaddress 0x0, 0x1 and 0x2 contain
the 48-Bit MAC address of the Ethernet interface.

Default value: 02:23:20:21:22:30

| XX | : | XX | : | XX | : | XX | : | XX | : | XX |
|---|---|---|---|---|---|---|---|---|---|---|
| Byte 5 | | Byte 4 | | Byte 3 | | Byte 2 | | Byte 1 | | Byte 0 |

Subaddress: 0x0
Content: 16-Bit

| Bit # | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | MAC Address **Byte 1** | | | | | | | | MAC Address **Byte 0** | | | | | | | |

Subaddress: 0x1
Content: 16-Bit

| Bit # | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | MAC Address **Byte 3** | | | | | | | | MAC Address **Byte 2** | | | | | | | |

Subaddress: 0x2
Content: 16-Bit

| Bit # | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | MAC Address **Byte 5** | | | | | | | | MAC Address **Byte 4** | | | | | | | |

### 3.5.1.2 MAC Address – Set Register

| | |
|---|---|
| Register: | **MAC Address – Set Register** |
| Address: | 0x90 + 4-Bit subaddress |
| Type: | Read / Write |
| Description: | Subaddress 0x0, 0x1 and 0x2 contain<br>the 48-Bit MAC address of the Ethernet interface.<br>Subaddress 0xF contains the control bit (Bit 0) to apply the change. |

To change the device mac address, write the new value to this register.
By writing 0x1 to subaddress 0xF, the new value will be applied to the *MAC Address Register* too.

### 3.5.1.3 IP Address

Register: **IP Address**
Address: 0x01 + 4-Bit subaddress
Type: Read-only
Description: Subaddress 0x0 and 0x1 contain
the 32-Bit IP address of the device Ethernet interface.

Default value: 192.168.2.10

XXX . XXX . XXX . XXX
Byte 3 Byte 2 Byte 1 Byte 0

Subaddress: 0x0
Content: 16-Bit


Subaddress: 0x1
Content: 16-Bit


### 3.5.1.4 IP Address – Set Register

Register: **IP Address – Set Register**
Address: 0x91 + 4-Bit subaddress
Type: Read / Write
Description: Subaddress 0x0 and 0x1 contain
the 32-Bit IP address of the device Ethernet interface.
Subaddress 0xF contains the control bit (Bit 0) to apply the change.

To change the device ip address, write the new value to this register.
By writing 0x1 to subaddress 0xF, the new value will be applied to the *IP Address Register* too.

## 3.5.2 DAQ Configuration

### 3.5.2.1 Destination IP Address DAQ-Server

Register: **Destination IP Address DAQ-Server**
Address: 0x02 + 4-Bit subaddress
Type: Read-only
Description: Subaddress 0x0 and 0x1 contain the 32-Bit IP address of the Ethernet interface of the DAQ-Server where the correlated eventdata will be send to.

Default value: 192.168.2.3

XXX : XXX : XXX : XXX
Byte 3 Byte 2 Byte 1 Byte 0

Subaddress: 0x0
Content: 16-Bit


Subaddress: 0x1
Content: 16-Bit


### 3.5.2.2 Destination IP Address DAQ-Server – Set Register

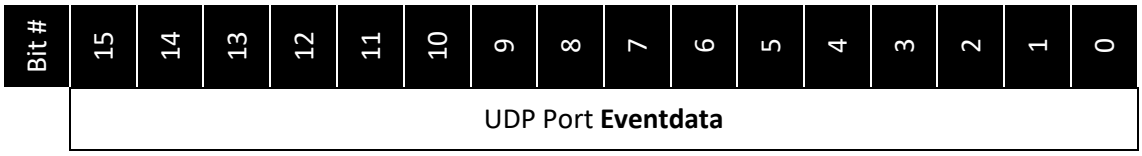
Register: **Destination IP Address DAQ-Server – Set Register**
Address: 0x93 + 4-Bit subaddress
Type: Read / Write
Description: Subaddress 0x0 and 0x1 contain the 32-Bit IP address of the Ethernet interface of the DAQ-Server where the correlated eventdata will be send to.
Subaddress 0xF contains the control bit (Bit 0) to apply the change.

To change the destination ip address for the DAQ server, write the new value to this register.
By writing 0x1 to subaddress 0xF, the new value will be applied to the *Destination IP Address DAQ-Server Register* too.

### 3.5.2.3 UDP Port Eventdata

Register: **UDP Port Eventdata**
Address: 0x10 + 4-Bit subaddress
Type: Read / Write
Description: Subaddress 0x0 contains the 16-Bit port number used for the eventdata transfer.

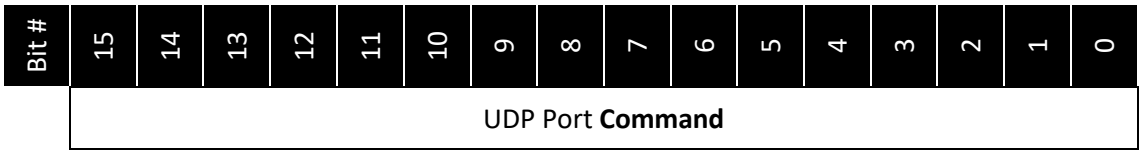
Subaddress: 0x0
Content: 16-Bit unsigned Integer




### 3.5.2.4 UDP Port Command

Register: **UDP Port Command**
Address: 0x11 + 4-Bit subaddress
Type: Read / Write
Description: Subaddress 0x0 contains the 16-Bit port number used for the command transfer.

Subaddress: 0x0
Content: 16-Bit unsigned Integer



### 3.5.2.5 MCPD ID

Register: **MCPD ID**
Address: 0x40 + 4-Bit subaddress
Type: Read / Write
Description: Subaddress 0x0 contains the 8-Bit MCPD ID.

Subaddress: 0x0
Content: 8-Bit unsigned Integer

| Bit # | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | x | x | x | x | x | x | x | x | **MCPD ID** | | | | | | | |

### 3.5.2.6 Run ID

Register: **Run ID**
Address: 0x82 + 4-Bit subaddress
Type: Read / Write
Description: Subaddress 0x0 contains the 16-Bit Run ID.

Subaddress: 0x0
Content: 16-Bit unsigned Integer

| Bit # | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | **Run ID** | | | | | | | | | | | | | | | |

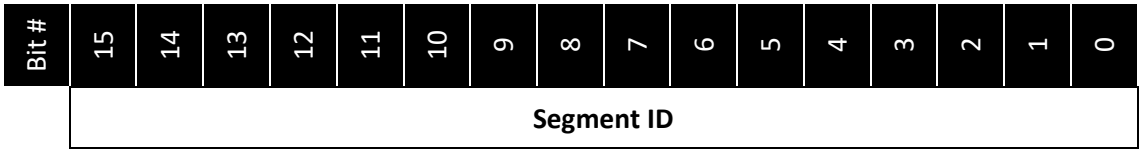
### 3.5.2.7 Segment ID

Register: **Segment ID**
Address: 0x41 + 4-Bit subaddress
Type: Read / Write
Description: Subaddress 0x0 contains the 16-Bit Segment ID.

For the detector with nine *DigitalReadout* segments the ***Segment ID*** has to be configured from zero to eight.

Looking at the neutron entrance window from the front, the DigitalReadout board of the segment on the left side has ***Segment ID*** 0.

Subaddress: 0x0
Content: 16-Bit unsigned Integer



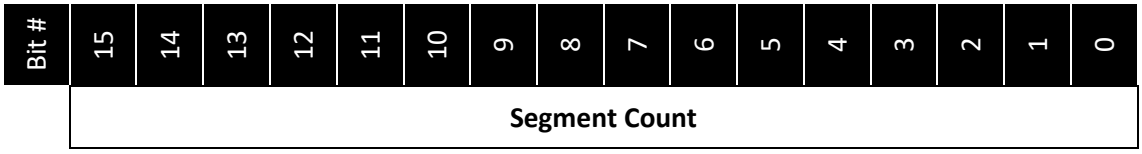

### 3.5.2.8 Segment Count

Register: **Segment Count**
Address: 0x42 + 4-Bit subaddress
Type: Read / Write
Description: Subaddress 0x0 contains the 16-Bit Segment Count.

For the use with QMesyDAQ as the readout software, this registers needs to be configured as the following:

2-Segment detector => segment count = 2 (Prototype)
9-Segment detector => segment count = 9

Subaddress: 0x0
Content: 16-Bit unsigned Integer



### 3.5.2.9 Frontend ToT Threshold

Register: **Frontend ToT Threshold**
Address: 0x30 + 4-Bit subaddress
Type: Read / Write
Description: The threshold voltage for the Time-over-Threshold conversion can be set by writing to this register.

$$ToT\ threshold\ voltage\ (mV) = \frac{value * 3{,}3V * 1000}{1024}$$

*Notice: The threshold voltage range is limited by the firmware between 50-1000mV!*

In order to set the threshold voltage for a complete segment, you have to write to the following subaddresses:

| *MainBoard Address Map* | |
|---|---|
| Subaddress | *MainBoard* connected to *DigitalReadout* PCB |
| 0x0 | Connector XS1 |
| 0x1 | Connector XS2 |
| 0x2 | Connector XW1 |
| 0x3 | Connector XW2 |

Content: 10-Bit unsigned Integer

| Bit # | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | x | x | x | x | x | x | **ToT Threshold** | | | | | | | | | |

## 3.5.3 System Status Logger

### 3.5.3.1 Destination IP Address

Register: **System Status Logger - Destination IP Address**
Address: 0x03 + 4-Bit subaddress
Type: Read-only
Description: Subadress 0x0 and 0x1 contain the 32-Bit IP address of the Ethernet interface of the PC where the data will be send to.

Default value: 192.168.2.3

XXX : XXX : XXX : XXX
Byte 3 Byte 2 Byte 1 Byte 0

Subaddress: 0x0
Content: 16-Bit


Subaddress: 0x1
Content: 16-Bit


### 3.5.3.2 Destination IP Address – Set Register

Register: **System Status Logger - Destination IP Address – Set Register**
Address: 0x92 + 4-Bit subaddress
Type: Read / Write
Description: Subaddress 0x0 and 0x1 contain
the 32-Bit destination ip address of the *System Status Logger*.
Subaddress 0xF contains the control bit (Bit 0) to apply the change.

To change the destination ip address, write the new value to this register.
By writing 0x1 to subaddress 0xF, the new value will be applied to the *Destination IP Address Register* too.

### 3.5.3.3 UDP Port

Register: **System Status Logger - UDP Port**
Address: 0x12 + 4-Bit subaddress
Type: Read / Write
Description: Subaddress 0x0 contains the 16-Bit port number used for the System Status Logger.

Subaddress: 0x0
Content: 16-Bit unsigned Integer

| Bit # | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | UDP Port **System Status Logger** | | | | | | | | | | | | | | | |

### 3.5.3.4 Configuration

Register: **System Status Logger - Configuration**
Address: 0x20 + 4-Bit subaddress
Type: Read / Write
Description: This is the configuration register to enable or disable the periodically UDP messages from the *System Status Logger*.

Update interval:
500ms ≙ "00"
1s ≙ "01"
5s ≙ “10"
10s ≙ "11"

Subaddress: 0x0
Content:

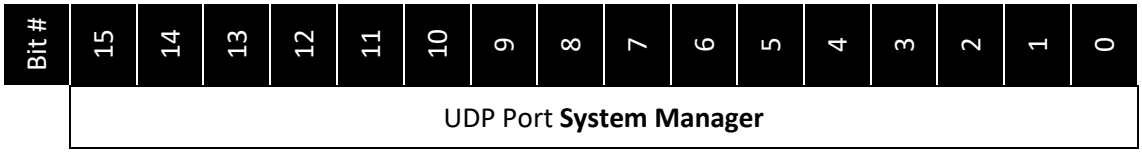
| Bit # | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | x | x | x | x | x | x | x | x | x | x | x | x | x | **Update Interval** | | **Enable** |

## 3.5.4 System Manager

### 3.5.4.1 UDP Port System Manager

Register: **UDP Port System Manager**
Address: 0x13 + 4-Bit subaddress
Type: Read / Write
Description: Subaddress 0x0 contains the 16-Bit port number used for the interface to the internal System Manager.

Subaddress: 0x0
Content: 16-Bit unsigned Integer



## 3.5.5 Anode Readout

### 3.5.5.1 Destination IP Address

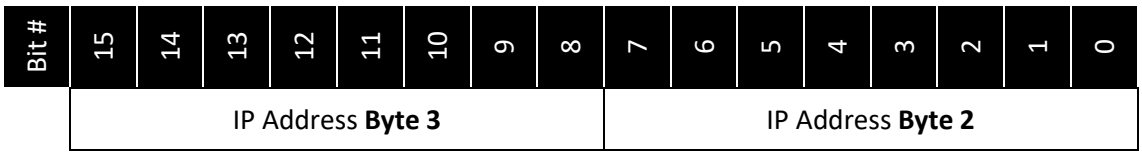
Register: **Anode Readout - Destination IP Address**
Address: 0x04 + 4-Bit subaddress
Type: Read-only
Description: Subadress 0x0 and 0x1 contain the 32-Bit IP address of the Ethernet interface of the PC where the data will be send to.

Default value: 192.168.2.3

XXX : XXX : XXX : XXX
Byte 3 Byte 2 Byte 1 Byte 0

Subaddress: 0x0
Content: 16-Bit


Subaddress: 0x1
Content: 16-Bit




### 3.5.5.2 Destination IP Address – Set Register

Register: **IP Address – Set Register**
Address: 0x94 + 4-Bit subaddress
Type: Read / Write
Description: Subaddress 0x0 and 0x1 contain
the 32-Bit destination ip address for the anode readout.
Subaddress 0xF contains the control bit (Bit 0) to apply the change.

To change the destination ip address, write the new value to this register.
By writing 0x1 to subaddress 0xF, the new value will be applied to the *Destination IP Address Register* too.

### 3.5.5.3 UDP Port

Register: **Anode Readout - UDP Port**
Address: 0x14 + 4-Bit subaddress
Type: Read / Write
Description: Subaddress 0x0 contains the 16-Bit port number used for the histogram transfer.

Subaddress: 0x0
Content: 16-Bit unsigned Integer

| Bit # | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | UDP Port **Anode Readout** | | | | | | | | | | | | | | | |

### 3.5.5.4 Control

Register: **Anode Readout - Control**
Address: 0x21 + 4-Bit subaddress
Type: Read / Write
Description:

Enable => Activates histogramming
Clear => Clears histogram

Update Interval => UI_500MS ≙ "00"
UI_1S ≙ "01"
UI_5S ≙ “10"
UI_10S ≙ "11"

Subaddress: 0x0 ≙ Channel 1
0x1 ≙ Channel 2

Content:

| Bit # | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | x | x | x | x | x | x | x | x | x | x | x | x | **Update Interval** | | **Clear** | **Enable** |

### 3.5.5.5 Threshold

Register: **Anode Readout - Threshold**
Address: 0x22 + 4-Bit subaddress
Type: Read / Write
Description:

Subaddress: 0x0 ≙ Channel 1
0x1 ≙ Channel 2

Content: 12-Bit unsigned Integer

| Bit # | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | x | x | x | x | **Threshold** | | | | | | | | | | | |

## 3.5.6 System Status Information

### 3.5.6.1 Firmware Timestamp

Register: **Firmware Timestamp**
Address: 0x50 + 4-Bit subaddress
Type: Read-only
Description: Contains the 32-bit timestamp generated during the bitstream generation by Xilinx Vivado. (XAPP1232)

Subaddress 0x0 contains the 16-LSBs.
Subaddress 0x1 contains the 16-MSBs.

ddddd_MMM_yyyyyy_hhhhh_mmmmmm_ssssss

(bit 31) ……(bit 0)

| | |
|---|---|
| ddddd | = 5 bits to represent 31 days in a month |
| MMMM | = 4 bits to represent 12 months in a year |
| yyyyyy | = 6 bits to represent 0 to 63 (to note year 2000 to 2063) |
| hhhhh | = 5 bits to represent 24 hours in a day |
| mmmmmm | = 6 bits to represent 60 minutes in an hour |
| ssssss | = 6 bits to represent 60 seconds in a minute |

### 3.5.6.2 Firmware Version

Register: **Firmware Version**
Address: 0x51 + 4-Bit subaddress
Type: Read-only
Description: Contains version information of the firmware in the FPGA.

Version: Major . Minor . Patch – Commits

Subaddress: 0x0
Content: 16-Bit unsigned Integer


Subaddress: 0x1
Content: 16-Bit unsigned Integer


Subaddress: 0x2
Content: 16-Bit unsigned Integer


Subaddress: 0x3
Content: 16-Bit unsigned Integer

### 3.5.6.3 Firmware Git Hash

| | |
|---|---|
| Register: | **Firmware Git Hash** |
| Address: | 0x52 + 4-Bit subaddress |
| Type: | Read-only |
| Description: | Contains the 160-Bit Git repository hash of the firmware in the FPGA. |
| | Subaddress 0x0 contains the 16-LSBs. |
| | Subaddress 0x9 contains the 16-MSBs. |

### 3.5.6.4 System Uptime

| | |
|---|---|
| Register: | **System Uptime** |
| Address: | 0x53 + 4-Bit subaddress |
| Type: | Read-only |
| Description: | Contains a 32-bit up-counter value. |
| | The value represents the uptime in seconds since the last power-up / FPGA configuration. |
| | Subaddress 0x0 contains the 16-LSBs. |
| | Subaddress 0x1 contains the 16-MSBs. |

### 3.5.6.5 Device Type

Register: **Device Type**
Address: 0x54 + 4-Bit subaddress
Type: Read-only
Description: Subaddress 0x0 contains the device type.

Device Type (DT): '0' ≙ DigitalReadout (detector segment)
'1' ≙ Correlation Unit

Subaddress: 0x0
Content: 1-Bit device type identifier

| Bit # | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | **DT** |

### 3.5.6.6 System Status

Register: **System Status**
Address: 0x70 + 4-Bit subaddress
Type: Read-only
Description: Subaddress 0x0 contains a 16-Bit vector.

Bit zero is a logic Or of Bits [5:1].

| | |
|---|---|
| Over Temperature: | Trigger = 85°C<br>Reset = 50°C |
| User Temperature Alarm: | Trigger = 65°C<br>Reset = 50°C |
| VCCINT Alarm: | Upper Level = 1,03V<br>Lower Level = 0,97V |
| VCCBRAM Alarm: | Upper Level = 1,05V<br>Lower Level = 0,95V |
| VCCAUX Alarm: | Upper Level = 1,89V<br>Lower Level = 1,75V |
| EEPROM busy: | This bit indicates that there is an access to the EEPROM.<br>This could be after a power-up or because of initiating a *STORE* or *LOAD* command through the *EEPROM Control* register. |

Subaddress: 0x0
Content: 16-Bit

| Bit # | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | x | x | x | x | x | x | x | x | x | **EEPROM busy** | **Over temperature** | **User Temp Alarm** | **VCCINT Alarm** | **VBRAM Alarm** | **VCCAUX Alarm** | **Alarm** |

### 3.5.6.7 DAQ Status

Register: **DAQ Status**
Address: 0x71 + 4-Bit subaddress
Type: Read-only
Description: Bit 0 indicates whether the DAQ is running or not.

Subaddress: 0x0
Content: 16-Bit

| Bit # | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | **DAQ running** |

### 3.5.6.8 FPGA Temperature

Register: **FPGA Temperature**
Address: 0x60 + 4-Bit subaddress
Type: Read-only
Description: Subaddress 0x0 contains the Temperature of the FPGA.

Subaddress: 0x0
Content: 12-Bit XADC value (Xilinx)

| Bit # | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | x | x | x | x | **Temperature FPGA** | | | | | | | | | | | |

### 3.5.6.9 FPGA Temperature Max

Register: **FPGA Temperature Max**
Address: 0x61 + 4-Bit subaddress
Type: Read-only
Description: Subaddress 0x0 contains the highest measured temperature of the FPGA since the last power up.

Subaddress: 0x0
Content: 12-Bit XADC value (Xilinx)

| Bit # | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | x | x | x | x | **Max. Temperature FPGA** | | | | | | | | | | | |

$$Temperature\ (°C) = \frac{XADC\ value * 503{,}975}{4096} - 273{,}15$$

### 3.5.6.10 PreShape Temperature

Register: **PreShape Temperature**
Address: 0x62 + 4-Bit subaddress
Type: Read-only
Description: Each subaddress contains a 12-Bit value from the temperature sensor TMP112. It has to be converted to the temperature value itself.

Subaddress: 2-Bit *MainBoard* + 2-Bit Slot on the Frontend Unit

| *MainBoard Address Map* | |
|---|---|
| Address | *MainBoard* connected to *DigitalReadout* PCB |
| 00 | Connector XS1 |
| 01 | Connector XS2 |
| 10 | Connector XW1 |
| 11 | Connector XW2 |

| *PreShape PCB Address Map* | |
|---|---|
| Address | *PreShape* PCB connected to *MainBoard* PCB |
| 00 | Connector X1 |
| 01 | Connector X2 |
| 10 | Connector X3 |
| 11 | Connector X4 |

E.g., address for the temperature sensor on *PreShape* slot X2, connected to *DigitalReadout* on XW1.

Subaddress => b'1001'

Content: 12-Bit value from temperature sensor TMP112 + 1-Bit sensor connected

Sensor Connection (SC): '1' ≙ not connected
'0' ≙ connected

One LSB represents 0,0625°C.
Negative numbers are represented in binary twos complement format.

| Bit # | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | **SC** | x | x | x | **Temperature PreShape** | | | | | | | | | | | |

### 3.5.6.11 MainBoard PSU Voltages

Register: **MainBoard PSU Voltages**
Address: 0x63 + 4-Bit subaddress
Type: Read-only
Description: Each subaddress contains the 12-Bit value from the integrated XADC of the FPGA.

Subaddress: 2-Bit *MainBoard* + 2-Bit Power Rail

| *MainBoard Address Map* | |
|---|---|
| Address | *MainBoard* connected to *DigitalReadout* PCB |
| 00 | Connector XS1 |
| 01 | Connector XS2 |
| 10 | Connector XW1 |
| 11 | Connector XW2 |

| *Power Rail Address Map* | |
|---|---|
| Address | Power Rail on *MainBoard* PCB |
| 00 | +2,5V (positive voltage / power supply for amplifiers) |
| 01 | -2,5V (negative voltage / power supply for amplifiers) |
| 10 | +3,3V (power supply for the comparators) |
| 11 | +Vref (threshold voltage for Time-over-Threshold conversion) |

E.g., address for the Power Rail +2,5V on the *MainBoard* connected to *DigitalReadout* on XW1.

Subaddress => b'1000'

Content: 12-Bit XADC value (Xilinx)

Conversion:

Rail +2,5V: Voltage = (UInt16) value / 4095 * 6
Rail -2,5V: Voltage = (UInt16) value / 4095 * 5
Rail +3,3V: Voltage = (UInt16) value / 4095 * 6
Rail +Vref: Voltage = (UInt16) value / 4095

| Bit # | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | x | x | x | x | **Voltage** | | | | | | | | | | | |

## 3.5.7 Command & Control

### 3.5.7.1 EEPROM Command

Register: **EEPROM Command**
Address: 0x80 + 4-Bit subaddress
Type: Read / Write
Description: EEPROM command register to initiate a Load configuration from EEPROM into the internal registers of the FPGA or to store the actual values of the internal registers into the EEPROM. The values in the EEPROM will be used to configure the registers on power up.

Command Type: LOAD ≙ "000"
STORE ≙ "001"

The execute bit will be reset automatically. To wait for the end of execution, you have to read the *EEPROM busy* bit in the *System Status* register.

Subaddress: 0x0
Content:

| Bit # | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | x | x | x | x | x | x | x | x | x | x | x | x | **Command Type** | | | **Execute** |

### 3.5.7.2 DAQ Command

Register: **DAQ Command**
Address: 0x81 + 4-Bit subaddress
Type: Read / Write
Description: By writing to the DAQ Command register, it is possible to control the DAQ system through the system manager.
After writing to the register, the bits will be reset automatically.

Subaddress: 0x0
Content:

| Bit # | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | X | x | x | x | x | x | x | x | x | x | x | x | **Continue** | **Stop** | **Start** | **Reset** |

## 3.5.8 Debug & Raw Data

### 3.5.8.1 Channel Mask X

Register: **Channel Mask X**
Address: 0xA1 + 4-Bit subaddress
Type: Read / Write
Description: Through this register, it is possible to disable certain X-channels in the signal-processing module.
The application is for debugging.

By writing a logical '1' to a Bit of this register, the X-channel will be disabled.

Default: All channels are enabled.

Subaddress: 0x0
Content:

| Bit # | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | CH (16 * subaddress)+15 | CH (16 * subaddress)+14 | CH (16 * subaddress)+13 | CH (16 * subaddress)+12 | CH (16 * subaddress)+11 | CH (16 * subaddress)+10 | CH (16 * subaddress)+9 | CH (16 * subaddress)+8 | CH (16 * subaddress)+7 | CH (16 * subaddress)+6 | CH (16 * subaddress)+5 | CH (16 * subaddress)+4 | CH (16 * subaddress)+3 | CH (16 * subaddress)+2 | CH (16 * subaddress)+1 | CH (16 * subaddress) |

### 3.5.8.2 Channel Mask Y

Register: **Channel Mask Y**
Address: 0xA2 + 4-Bit subaddress
Type: Read / Write
Description: Through this register, it is possible to disable certain Y-channels in the signal-processing module.
The application is for debugging.

By writing a logical '1' to a Bit of this register, the Y-channel will be disabled.

Default: All channels are enabled.

Subaddress: 0x0
Content:

| Bit # | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | CH (16 * subaddress)+15 | CH (16 * subaddress)+14 | CH (16 * subaddress)+13 | CH (16 * subaddress)+12 | CH (16 * subaddress)+11 | CH (16 * subaddress)+10 | CH (16 * subaddress)+9 | CH (16 * subaddress)+8 | CH (16 * subaddress)+7 | CH (16 * subaddress)+6 | CH (16 * subaddress)+5 | CH (16 * subaddress)+4 | CH (16 * subaddress)+3 | CH (16 * subaddress)+2 | CH (16 * subaddress)+1 | CH (16 * subaddress) |

### 3.5.8.3 Segment Data Path Disable

Register: **Segment Data Path Disable**
Address: 0xA3 + 4-Bit subaddress
Type: Read / Write
Description: By writing to this register, it is possible to disable the transmission of the eventdata of each individual segment.

Default: All data paths are enabled.

Subaddress: 0x0
Content:

| Bit # | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | x | x | x | x | x | x | x | **Segment 9** | **Segment 8** | **Segment 7** | **Segment 6** | **Segment 5** | **Segment 4** | **Segment 3** | **Segment 2** | **Segment 1** |

### 3.5.8.4 Correlation Unit

Register: **Correlation unit Raw Data**
Address: 0xA4 + 4-Bit subaddress
Type: Read / Write
Description: This register enables the Raw Data mode. The Eventdata structure in Raw Data mode is defined later in this paragraph.

Note! Both Correlation Unit and Digital Readout Raw Data registers have to be set to the same value (Normal or Raw Data) to provide consistent data.

Mode : Normal (correlated data) ≙ b'0'
Raw Data ≙ b'1'

Default: Normal Raw data mode off

Subaddress: 0x0
Content: 1-Bit

| Bit # | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | x | x | x | x | x | x | x | X | x | x | x | x | x | x | x | **Mode** |

**Eventdata Structure in Raw data mode**

In Raw Data mode the Correlation Unit sends the unprocessed cathode hits collected by the Digital Readout Card to the DAQ PC. Each Eventdata encodes a single hit, either of one of the 128 wires (X) or 128 stripes (Y), its Time over Threshold and Timestamp.

Header, event timestamp and ToT in Raw Data mode are in unit of 12.5ns being the ToT sampling frequency equals to 80Mhz. NB: the QMesyDAQ timestamp is in unit of 100ns.

In order to univocally identify X and Y-Cathode number, the X and Y-Position of the QMesyDAQ Eventdata protocol as defined in par. 3.6.1.1 has to be decoded as follow:

when Y-Position = 0 => Number of X-Cathode = X-Position (7 LSB)

when Y-Position ≠ 0 (X-Position = 0) => Number of Y-Cathode = Y-Position – 512 (7 LSB)

MSB LSB

| | | | | |
|---|---|---|---|---|
| ID (1 bit) = 0 | Amplitude(8) | Y-Position (10) | X-Position (10) | Timestamp (19) |

*ID:* ID = 0 signaling a neutron event.
1 bit

*Amplitude:* Time over Threshold in unit of 12,5ns. 8 bit

*Y-Position:* When Y-Position ≠ 0 => Identifies a Y-Cathode
Y-Cathode = Y-Position - 512
Y-Cathode range is 7 bit, 128 stripes
When Y-Position = 0 => Identifies a X-Cathode

*X-Position:* When Y-Position = 0 => X-Cathode = X-Position
X-Cathode range is 7 bit, 128 wires

*Timestamp:* Timing offset to the corresponding header timestamp
event time = header timestamp + event timestamp
Time unit = 12,5 ns

E.g.
Y-Position = 537dec , X-Position = 0dec => Y-Cathode = 537-512 = 25
Y-Position = 0dec , X-Position = 44dec => X-Cathode = 44

Register: **Correlation Unit Data Generator**
Address: 0xA4 + 4-Bit subaddress
Type: Read / Write
Description: By writing to these registers, it is possible to enable the internal data generator. The generator send Eventdata with a position (Y-position & X-position ) linearly increased. The time generation is pseudorandom. The event production can be furthermore delayed by a pseudorandom delay that emulates the jitter of the digital processing.

Rate (approximately):

| | | | |
|---|---|---|---|
| Off | ≙ b'0000' | 3000 Hz/Seg | ≙ b'1000' |
| 25 Hz/Seg | ≙ b'0010' | 6000 Hz/Seg | ≙ b'1001' |
| 50 Hz/Seg | ≙ b'0010' | 12 kHz/Seg | ≙ b'1010' |
| 90 Hz/Seg | ≙ b'0011' | 24 kHz/Seg | ≙ b'1011' |
| 190 Hz/Seg | ≙ b'0100' | 48 kHz/Seg | ≙ b'1100' |
| 380 Hz/Seg | ≙ b'0101' | 98 kHz/Seg | ≙ b'1101' |
| 760 Hz/Seg | ≙ b'0110' | 194 kHz/Seg | ≙ b'1110' |
| 1500 Hz/Seg | ≙ b'0111' | 388 kHz/Seg | ≙ b'1111' |

Delay mask (in Tck = 12,5 ns)
e.g. Mask ≙ bx'00000000' => no delay, immediate generation
Mask ≙ 0x'00000111' => event generation delayed up to 7 Tck

Default: Data generator off

Subaddress: 0x1
Content: 9-Bit

| Bit # | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | **Delay Mask** | | | | | | | | **x** | **x** | **x** | **x** | **Rate** | | | |

### 3.5.8.5 Digital Readout

Register: **Digital Readout Raw Data**
Address: 0xA5 + 4-Bit subaddress
Type: Read / Write
Description: This register enables the Raw Data mode. The Eventdata structure in Raw Data mode is defined in par. 3.5.8.4

Note! Both Correlation Unit and Digital Readout Raw Data registers have to be set to the same value (Normal or Raw Data) to provide consistent data.

Mode : Normal (correlated data) ≙ b'0'
Raw Data ≙ b'1'

Default: Normal

Subaddress: 0x0
Content: 1-Bit

| Bit # | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | x | x | x | X | x | x | X | x | x | x | x | x | x | x | x | **Mode** |

Register: **Digital Readout Pattern Generator**
Address: 0xA5 + 4-Bit subaddress
Type: Read / Write
Description: By writing to these registers, it is possible to enable the internal pattern generator. The generator injects signal at the input of the digital readout in order to emulate neutron-like events.

Rate (approximately):

| | | | |
|---|---|---|---|
| Off | ≙ b'0000' | 2300 Hz/Seg | ≙ b'1000' |
| 20 Hz/Seg | ≙ b'0010' | 4600 Hz/Seg | ≙ b'1001' |
| 40 Hz/Seg | ≙ b'0010' | 9400 Hz/Seg | ≙ b'1010' |
| 80 Hz/Seg | ≙ b'0011' | 19 kHz/Seg | ≙ b'1011' |
| 150 Hz/Seg | ≙ b'0100' | 37 kHz/Seg | ≙ b'1100' |
| 300 Hz/Seg | ≙ b'0101' | 74 kHz/Seg | ≙ b'1101' |
| 570 Hz/Seg | ≙ b'0110' | 146 kHz/Seg | ≙ b'1110' |
| 1150 Hz/Seg | ≙ b'0111' | 285 kHz/Seg | ≙ b'1111' |

Time Generation:

| | |
|---|---|
| Random | ≙ b'0' |
| Fixed Frequency | ≙ b'1' |

Multiplicity (even or odd address):

| | |
|---|---|
| 5/even, 4/odd | ≙ b'00' |
| 3/even, 2/odd | ≙ b'01' |
| 1/even, 1/odd | ≙ b'10' |
| 1/even, 1/odd | ≙ b'11' |

Noise Generation:

| | |
|---|---|
| No noise | ≙ b'0' |
| Noise | ≙ b'1' |

Position Generation:

| | |
|---|---|
| Random | ≙ b'0' |
| Linear | ≙ b'1' |

Default: Pattern generator off

Subaddress: 0x1
Content: 9-Bit

| Bit # | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | x | x | x | x | x | x | X | **Position Gen** | **Noise Gen** | **Multiplicity** | | **Time Gen** | **Rate** | | | |

## 3.5.9 Front panel I/O

### 3.5.9.1 Analog Inputs

Register: **Analog Inputs**
Address: 0xB0 + 4-Bit subaddress
Type: Read-only
Description: The Correlation Unit has four analog input channels with a 10:1 attenuation.

Subaddress:

| Analog Inputs Address Map | |
|---|---|
| Address | *Correlation Unit* Front panel |
| 0x0 | Analog Input 1 |
| 0x1 | Analog Input 2 |
| 0x2 | Analog Input 3 |
| 0x3 | Analog Input 4 |

Content: 12-Bit XADC value (Xilinx)

| Bit # | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | x | x | x | x | **Analog input value** | | | | | | | | | | | |

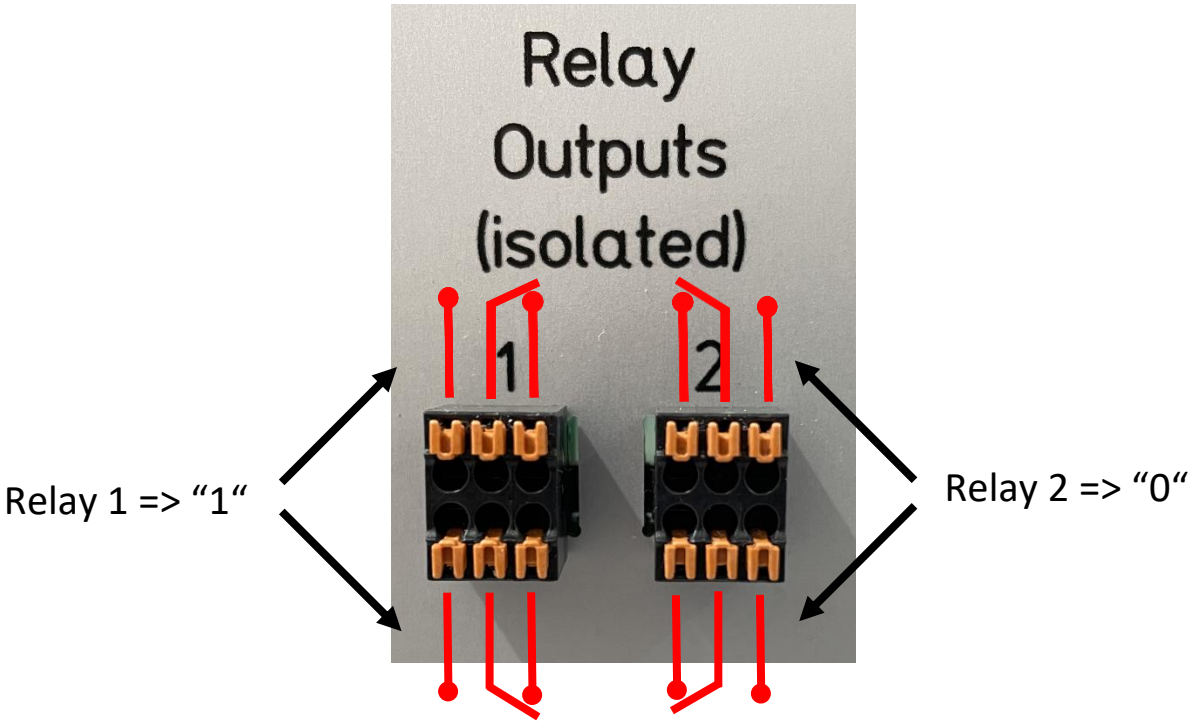
$$Analog\ Input\ Voltage\ (V) = \frac{value * 10V}{4096}$$

### 3.5.9.2 Relay Outputs

Register: **Relay Outputs**
Address: 0xB1 + 4-Bit subaddress
Type: Read / Write
Description: The Correlation Unit has two relay outputs.

Subaddress: 0x0
Content: 16-Bit

| Bit # | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | x | x | x | x | x | x | x | x | x | x | x | x | x | x | **Relay 2** | **Relay 1** |



### 3.5.9.3 Digital Outputs

Register: **Digital Outputs**
Address: 0xB2 + 4-Bit subaddress
Type: Read / Write
Description: The Correlation Unit has two digital outputs.

Subaddress: 0x0
Content: 16-Bit

| Bit # | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | x | x | x | x | x | x | x | x | x | x | x | x | x | x | **Digital Output 2** | **Digital Output 1** |

## 3.6 Protocols

### 3.6.1 Mesytec – Protocol

See Documentation www.mesytec.com

Communication between DAQ / control computers and the Correlation Unit is based on the standardized UDP / IP protocol family.

There are two principle kinds of communications in the detector readout system:

- The Correlation Unit receive and answer command buffers. Usually commands are emitted by a single control PC. Each command is echoed to the commanding computer, delivering a command response and a status description as well as a variable amount of information.

- During data acquisition, the Correlation Unit will emit data packages autonomously. The address of the recipient is set up during the initialization process of the system. Thus, it is possible to send data packages and command answers to different recipients.

For experiments with high data rates, it might be useful to have data taking distributed on more than one pc.

Both types of communication packets are wrapped in UDP packets.

Using UDP communication allows efficient data transmission without too much protocol overhead. But it has to be remembered that there’s no intrinsic protection against packet loss (like in TCP e. g.) Thus some simple control mechanisms have been implemented into the mesytec protocol in order to achieve sufficient data security.

### 3.6.1.1 Data Buffers

During data acquisition, the Correlation Unit transmit a continuous stream of event buffers for each detector segment.
Each event buffer consists of a buffer header (21 x 16 bit = 42 bytes) and a variable number of events. Each event has a length of 48 bits = 6 bytes.
The total length of an event buffer varies between 42 bytes (header only) up to 1.500 bytes (limited by unfragmented Ethernet frame length).

Buffer Structure:

| | | |
|---|---|---|
| Buffer Length (in 16bit words) | | Word 0 |
| Buffer Type ≙ 0x0002 | | |
| Header Length | | |
| Buffer Number | | |
| Run-ID | | |
| MCPD-ID | Status | |
| Header Timestamp Low | | |
| Header Timestamp Mid | | |
| Header Timestamp High | | |
| Parameter 0 Low | | |
| Parameter 0 Mid | | |
| Parameter 0 High | | |
| Parameter 1 Low | | |
| Parameter 1 Mid | | |
| Parameter 1 High | | |
| Parameter 2 Low | | |
| Parameter 2 Mid | | |
| Parameter 2 High | | |
| Parameter 3 Low | | |
| Parameter 3 Mid | | |
| Parameter 3 High | | Word 20 |
| Event 0 Low | | Word 21 |
| Event 0 Mid | | |
| Event 0 High | | |
| Event 1 Low | | |
| Event 1 Mid | | |
| Event 1 High | | |
| … | | |
| … | | |
| … | | |
| … | | |
| … | | |
| … | | |
| Event n Low | | |
| Event n Mid | | |
| Event n High | | Word 20+3*(n+1) |

**Header data dictionary:**

| | |
|---|---|
| *Buffer Type:* | 16 bit type descriptor.<br>Bits 0 … 14 carry a version information.<br>Bit 15 = 0: data buffer<br>(Bit 15 = 1: command buffer) |
| *Header Length:* | Length of header information in 16 bit words |
| *Buffer Number:* | Simple 16 bit counter for data buffers. Incremented automatically by the Correlation Unit to allow for loss monitoring. |
| *Buffer Length:* | Total length in multiple of 16 bit words, stretching from *Buffer Type* to last data word. |
| *Run ID:* | Simple 16 bit run counter, set by software, to allow for integrity Control. |
| *MCPD-ID:* | 8 bit ID of sending segment module, to be assigned during hardware initialization. |
| *Status:* | 8 bit wide bit field for sync and start/stop status of the sending MCPD-8<br>currently only bits 0, 1 used:<br>bit 0: 1 = DAQ running, 0 = DAQ stopped<br>bit 1: 1 = sync o.k., 0 = sync error (do not use!) |
| *Header Timestamp:* | 48 bits current status of the synchronized system timer (100 ns timing resolution)<br>Represents the value of the system timer at the moment of buffer opening.<br>All events in a buffer carry a positive 19 bit offset timing information relative to this header timestamp. |
| *Parameter 0 … 3:* | MCPD-8 allows the transmission of selected counter / input values with each buffer. So Parameter 0 … 3 represent the values of the selected counters / inputs at the moment of buffer opening (t = header timestamp)<br>Also the digital input status and ADC values can be mapped here. |
| *Event 0 … n:* | (n+1) x 48 bit event information<br>Event structure is explained in detail below. |

**Event structure:**

Each event has a fixed 48 bit length. The contents differs according to the event id.

ID = 0: Neutron data event
ID = 1: Trigger data event

**Neutron data events (ID = 0):**

MSB LSB

| | | | | |
|---|---|---|---|---|
| ID (1 bit) = 0 | Amplitude(8) | Y-Position (10) | X-Position (10) | Timestamp (19) |

*ID:* ID = 0 signaling a neutron event.
1 bit

*Amplitude:* Amplitude (energy) of the neutron event
8 bit

*Y-Position:* Y-Position of the neutron event
10 bit

*X-Position:* X-Position of the neutron event
10 bit

*Timestamp:* Timing offset to the corresponding header timestamp
event time = header timestamp + event timestamp
19 bit

**Trigger events (ID = 1):**

Several trigger sources (counters, timers, digital inputs) can initiate a data taking event. Triggered by levels or defined overflows of the trigger sources, data are taken from the data sources and written to an event structure. Possible trigger and data sources are timers, counters, and ADC values.

MSB LSB

| | | | | |
|---|---|---|---|---|
| ID (1 bit) = 1 | TrigID (3 bit) | DataID (4) | Data (21 bit) | Timestamp (19) |

*ID:* ID = 1 signaling a “not neutron” event
The event is generated from the four digital monitor inputs.
1 bit

*TrigID:* Trigger ID characterizing the event trigger source
0: Monitor input 1 … 4

*DataID:* DataID characterizing the data source.
Data taking was initiated by the trigger source identified in TrigID, at the time “header timestamp + event timestamp”
0 … 3: Monitor / Chopper input 1 … 4

*Data:* not used
21 bit

*Timestamp:* timing offset to the corresponding header timestamp
event time = header timestamp + event timestamp
19 bit

Notice: The trigger events come all and only from the data stream of segment 1!

### 3.6.1.2 Command Buffers

Each command buffer consists of a buffer header (10 x 16 bit = 20 bytes) and a trailing data block of variable length. The contents of the data block depends on the individual commands.

The total length of a command buffer varies between 20 bytes (header only) up to 1.500 bytes (limited by unfragmented Ethernet frame length)

Buffer Structure:

| | | |
|---|---|---|
| Buffer Length (in 16bit words) | | Word 0 |
| Buffer Type | | |
| Header Length | | |
| Buffer Number | | |
| Cmd | | |
| MCPD-ID | Status | |
| Header Timestamp Low | | |
| Header Timestamp Mid | | |
| Header Timestamp High | | Word 8 |
| Command Checksum | | Word 9 |
| Data 0 | | Word 21 |
| Data 1 | | |
| Data 2 | | |
| … | | |
| … | | |
| … | | |
| Trailing data | | |
| with | | |
| Variable length | | Word (buffer length -1) |

**Header data dictionary:**

| | |
|---|---|
| *Buffer Type:* | 16 bit type descriptor.<br>Bits 0 … 14 carry a version information.<br>Bit 15 = 0: data buffer<br>Bit 15 = 1: command buffer |
| *Header Length:* | Length of header information in 16 bit words |
| *Buffer Number:* | Simple 16 bit counter to allow loss monitoring.<br>Separate counters for data and cmd buffers.<br>A control software could increment with each cmd issued.<br>The Correlation Unit will increment its own counter with each cmd answered. |
| *Buffer Length:* | Total length in multiple of 16 bit words, stretching from<br>*Buffer Type* to last data word. Only counts useful data words.<br>Padding bytes added to fulfill minimum Ethernet buffer size will<br>not be counted. |
| *CMD-ID:* | 16 bit value representing the command that is answered / issued in<br>this buffer. Please see the following chapter for a detailed<br>description of the individual commands. |
| *MCPD-ID:* | 8 bit ID of sending segment module, to be assigned during<br>hardware initialization. |
| *Status:* | 8 bit wide bit field for sync and start/stop status of the sending<br>MCPD-8<br>currently only bits 0, 1 used:<br>bit 0: 1 = DAQ running, 0 = DAQ stopped<br>bit 1: 1 = sync o.k., 0 = sync error (do not use!)<br>Can be left blank in buffers sent by control pc. |
| *Header Timestamp:* | 48 bits current status of the synchronized system timer<br>(100 ns timing resolution)<br>generated shortly before Ethernet transmission. It allows to have a<br>timed log of command communication.<br>Can be left blank in buffers sent by control pc. |
| *Command Checksum:* | 16 bit XOR checksum covering all words from *Buffer Type*<br>(Word 0) to last data word (Word buffer length –1).<br>Set checksum field to 0x0000 before calculation. |

### 3.6.1.3 Command Reference

General Command Format:

The structure of a command buffer is displayed here again. Its layout is identical for all commands.

| | | |
|---|---|---|
| Buffer Length (in 16bit words) | | Word 0 |
| Buffer Type | | |
| Header Length | | |
| Buffer Number | | |
| Cmd | | |
| MCPD-ID | Status | |
| Header Timestamp Low | | |
| Header Timestamp Mid | | |
| Header Timestamp High | | Word 8 |
| Command Checksum | | Word 9 |
| Data 0 | | Word 21 |
| Data 1 | | |
| Data 2 | | |
| … | | |
| … | | |
| … | | |
| Trailing data | | |
| with | | |
| Variable length | | Word (buffer length -1) |

In the following, only data from Word 10 on (Data 0) are displayed. The given command number has to be entered in the header field “Cmd” (Word 4)

Each command buffer has a trailing 0xFFFF as last word.

Command answer:
Every command buffer will be answered by the Correlation Unit. Set values instead of requested values will be inserted into the appropriate fields. If a command fails in the Correlation Unit, the Cmd number will have bit 15 set.

**Implemented Commands:**

| Cmd # | Command |
|---|---|
| 0 | Reset |
| 1 | Start DAQ |
| 2 | Stop DAQ |
| 3 | Continue DAQ |
| ~~6~~ | ~~Set Slave Operation~~ |
| 7 | Set Master Clock |
| 8 | Set Run ID |
| 22 | Read MCPD-8 fast tx capabilities |
| 36 | Read ID |
| 51 | Retrieve version information |

### 3.6.1.3.1 Reset DAQ

| **Reset DAQ** | **Cmd = 0** |
|---|---|
| **Word** | **Contents** |
| 10 | 0xFFFF |

Running DAQ will be stopped.
All counters and timers will be reset to 0.

Answer buffers look like follows:

| **Reset DAQ (answer)** | **Cmd = 0** |
|---|---|
| **Word** | **Contents** |
| 10 | 0xFFFF |

#### 3.6.1.3.2 Start DAQ

| **Start DAQ** | **Cmd = 1** |
|---|---|
| **Word** | **Contents** |
| 10 | 0xFFFF |

Start DAQ starts the data acquisition system.
Neutron and trigger events will be filled into data buffers.

Answer buffers look like follows:

| **Start DAQ (answer)** | **Cmd = 1** |
|---|---|
| **Word** | **Contents** |
| 10 | 0xFFFF |

#### 3.6.1.3.3 Stop DAQ

| **Stop DAQ** | **Cmd = 2** |
|---|---|
| **Word** | **Contents** |
| 10 | 0xFFFF |

Stop DAQ stops the data acquisition system.

Answer buffers look like follows:

| **Stop DAQ (answer)** | **Cmd = 2** |
|---|---|
| **Word** | **Contents** |
| 10 | 0xFFFF |

### 3.6.1.3.4 Continue DAQ

| **Continue DAQ** | **Cmd = 3** |
|---|---|
| **Word** | **Contents** |
| 10 | 0xFFFF |

Continue DAQ restarts the data acquisition system.

Answer buffers look like follows:

| **Continue DAQ (answer)** | **Cmd = 3** |
|---|---|
| **Word** | **Contents** |
| 10 | 0xFFFF |

### 3.6.1.3.5 Set Master Clock

| **Set Master Clock** | **Cmd = 7** |
|---|---|
| **Word** | **Contents** |
| 10 | Master clock, bits 0 … 15 |
| 11 | Master clock, bits 16 … 31 |
| 12 | Master clock, bits 32 … 47 |
| 13 | 0xFFFF |

Master clock can be set to any value if desired. Normally, a reset is initiated before a new run and all counters are set to zero during this reset automatically.
Only if another run start time than zero is desired, this registers must be set.

Answer buffers look like follows:

| **Set Master Clock** | **Cmd = 7** |
|---|---|
| **Word** | **Contents** |
| 10 | 0x0000 (not implemented) |
| 11 | 0x0000 (not implemented) |
| 12 | 0x0000 (not implemented) |
| 13 | 0xFFFF |

### 3.6.1.3.6 Set Run ID

| **Set Run ID** | **Cmd = 8** |
|---|---|
| **Word** | **Contents** |
| 10 | Run ID |
| 11 | 0xFFFF |

Set value for the header field “Run ID”
Can be set to any desired value.

Answer buffers look like follows:

| **Set Run ID (answer)** | **Cmd = 8** |
|---|---|
| **Word** | **Contents** |
| 10 | Run ID |
| 11 | 0xFFFF |

### 3.6.1.3.7 Read MCPD-8 fast tx capabilities

| **Read MCPD-8 fast tx capabilities** | **Cmd = 22** |
|---|---|
| **Word** | **Contents** |
| 10 | 0xFFFF |

Set value for the header field “Run ID”
Can be set to any desired value.

Answer buffers look like follows:

| **Read MCPD-8 fast tx capabilities (answer)** | **Cmd = 22** |
|---|---|
| **Word** | **Contents** |
| 10 | Capabilities bitmap = 0x0007 |
| 11 | Current setting = 0x0002 |
| 12 | 0xFFFF |

### 3.6.1.3.8 Read ID

| Retrieve version information | Cmd = 36 |
|---|---|
| **Word** | **Contents** |
| 10 | 0xFFFF |

Returns an ID to identify the detector type.

Answer buffers look like follows:

| Retrieve version information (answer) | Cmd = 36 |
|---|---|
| **Word** | **Contents** |
| 10 | 0x006E |
| 11 | 0x006E |
| 12 | 0x006E |
| 13 | 0x006E |
| 14 | 0x006E |
| 15 | 0x006E |
| 16 | 0x006E |
| 17 | 0x006E |
| 18 | 0x006E |
| 19 | 0x006E |
| 20 | 0xFFFF |

### 3.6.1.3.9 Retrieve Version Information

| Retrieve version information | Cmd = 51 |
|---|---|
| **Word** | **Contents** |
| 10 | 0xFFFF |

Returns version information of the Correlation Unit FPGA firmware.

Answer buffers look like follows:

| Retrieve version information (answer) | Cmd = 51 |
|---|---|
| **Word** | **Contents** |
| 10 | Git Version – Major |
| 11 | Git Version – Minor |
| 12 | Git Version – Patch (8 bit) & Commits (8 bit) |
| 13 | 0xFFFF |

### 3.6.2 UDP System Manager Bridge - Protocol

| | |
|---|---|
| Buffer Length (in 16 bit words) | Word 0 |
| Buffer Type (0x0006) | |
| Header Length | |
| Buffer Number | |
| Address | |
| Data (only Writes) | Word 5 |

Buffer Length: Total length in multiple of 16 bit words, stretching from „buffer type“ to last data word

Buffer Type: 16 bit type descriptor
0x0006 ≙ UDP System Manager Command Buffer

Header Length: Length information in 16 bit words.

Buffer Number: Simple 16 bit counter for data buffers.
Incremented automatically to allow for loss monitoring

Address: 1-Bit Command Type + 15-Bit address

Command: ‘0’ ≙ Read
‘1’ ≙ Write

#### 3.6.2.1 Read Access

Request:

| | | |
|---|---|---|
| Buffer Length | 0x0005 | Word 0 |
| Buffer Type | 0x0006 | |
| Header Length | 0x0004 | |
| Buffer Number | 0x0000 | |
| (MSB) 1-Bit Command Type + 15-Bit address | 0x0011 | Word 4 |

Answer:

| | | |
|---|---|---|
| Buffer Length | 0x0006 | Word 0 |
| Buffer Type | 0x0006 | |
| Header Length | 0x0004 | |
| Buffer Number | 0x0000 | |
| (MSB) 1-Bit Command Type + 15-Bit address | 0x0011 | |
| Data | 0xC0A8 | Word 5 |

### 3.6.2.2 Write Access

Request:

| | | |
|---|---|---|
| Buffer Length | 0x0006 | Word 0 |
| Buffer Type | 0x0006 | |
| Header Length | 0x0004 | |
| Buffer Number | 0x0027 | |
| (MSB) 1-Bit Command Type + 15-Bit address | 0x8821 | |
| Data | 0xAC1C | Word 5 |

Answer:

| | | |
|---|---|---|
| Buffer Length | 0x0006 | Word 0 |
| Buffer Type | 0x0006 | |
| Header Length | 0x0004 | |
| Buffer Number | 0x0027 | |
| (MSB) 1-Bit Command Type + 15-Bit address | 0x0821 | |
| Data | 0xAC1C | Word 5 |

### 3.6.3 System Status Logger – Data Buffers

#### 3.6.3.1 Correlation Unit

UDP Payload:

| | |
|---|---|
| FPGA Temperature | Word 0 |
| FPGA Temperature Max | |
| 0x0000 | |
| 0x0000 | |
| 0x0000 | |
| 0x0000 | |
| 0x0000 | |
| 0x0000 | |
| 0x0000 | |
| … | |
| … | |
| … | |
| 0x0000 | Word 37 |

### 3.6.3.2 DigitalReadout

UDP Payload:

| | |
|---|---|
| FPGA Temperature | Word 0 |
| FPGA Temperature Max | |
| Temperature PreShape (DigitalReadout Con. XS1 / MainBoard Con. X1) | |
| Temperature PreShape (DigitalReadout Con. XS1 / MainBoard Con. X2) | |
| Temperature PreShape (DigitalReadout Con. XS1 / MainBoard Con. X3) | |
| Temperature PreShape (DigitalReadout Con. XS1 / MainBoard Con. X4) | |
| Temperature PreShape (DigitalReadout Con. XS2 / MainBoard Con. X1) | |
| Temperature PreShape (DigitalReadout Con. XS2 / MainBoard Con. X2) | |
| Temperature PreShape (DigitalReadout Con. XS2 / MainBoard Con. X3) | |
| Temperature PreShape (DigitalReadout Con. XS2 / MainBoard Con. X4) | |
| Temperature PreShape (DigitalReadout Con. XW1 / MainBoard Con. X1) | |
| Temperature PreShape (DigitalReadout Con. XW1 / MainBoard Con. X2) | |
| Temperature PreShape (DigitalReadout Con. XW1 / MainBoard Con. X3) | |
| Temperature PreShape (DigitalReadout Con. XW1 / MainBoard Con. X4) | |
| Temperature PreShape (DigitalReadout Con. XW2 / MainBoard Con. X1) | |
| Temperature PreShape (DigitalReadout Con. XW2 / MainBoard Con. X2) | |
| Temperature PreShape (DigitalReadout Con. XW2 / MainBoard Con. X3) | |
| Temperature PreShape (DigitalReadout Con. XW2 / MainBoard Con. X4) | |
| MainBoard Voltage Rail +2,5V (DigitalReadout Con. XS1) | |
| MainBoard Voltage Rail -2,5V (DigitalReadout Con. XS1) | |
| MainBoard Voltage Rail +3,3V (DigitalReadout Con. XS1) | |
| MainBoard Voltage Rail +Vref (DigitalReadout Con. XS1) | |
| MainBoard Voltage Rail +2,5V (DigitalReadout Con. XS2) | |
| MainBoard Voltage Rail -2,5V (DigitalReadout Con. XS2) | |
| MainBoard Voltage Rail +3,3V (DigitalReadout Con. XS2) | |
| MainBoard Voltage Rail +Vref (DigitalReadout Con. XS2) | |
| MainBoard Voltage Rail +2,5V (DigitalReadout Con. XW1) | |
| MainBoard Voltage Rail -2,5V (DigitalReadout Con. XW1) | |
| MainBoard Voltage Rail +3,3V (DigitalReadout Con. XW1) | |
| MainBoard Voltage Rail +Vref (DigitalReadout Con. XW1) | |
| MainBoard Voltage Rail +2,5V (DigitalReadout Con. XW2) | |
| MainBoard Voltage Rail -2,5V (DigitalReadout Con. XW2) | |
| MainBoard Voltage Rail +3,3V (DigitalReadout Con. XW2) | |
| MainBoard Voltage Rail +Vref (DigitalReadout Con. XW2) | |
| ToT Threshold Register for MainBoard connected to DigitalReadout Con. XS1 | |
| ToT Threshold Register for MainBoard connected to DigitalReadout Con. XS2 | |
| ToT Threshold Register for MainBoard connected to DigitalReadout Con. XW1 | |
| ToT Threshold Register for MainBoard connected to DigitalReadout Con. XW2 | Word 37 |

### 3.6.4 Anode Readout – Data Buffers

| | |
|---|---|
| Buffer Length (in 16 bit words) | Word 0 |
| Buffer Type (0x000A) | |
| Header Length | |
| Buffer Number | |
| Channel Number | |
| Fragment Number | |
| Uptime | |
| Counts ADC Channel 0 ( 16 bit LSB) | |
| Counts ADC Channel 0 ( 16 bit MSB) | |
| Counts ADC Channel 1 ( 16 bit LSB) | |
| Counts ADC Channel 1 ( 16 bit MSB) | |
| … | |
| … | |
| … | |
| Counts ADC Channel 353 ( 16 bit LSB) | |
| Counts ADC Channel 353 ( 16 bit MSB) | |

Buffer Length: Total length in multiple of 16 bit words, stretching from „buffer type“ to last data word

Buffer Type: 16 bit type descriptor
Bits 15 … 8 carry a version information.
Bits 7 … 0 buffer type => 0x0A ≙ Anode Readout Data Buffer.

Header Length: Length information in 16 bit words.

Buffer Number: Simple 16 bit counter for data buffers.
Incremented automatically to allow for loss monitoring

Channel Number: Anode Readout has two input channels.
Input Channel 1 ≙ 0x0000
Input Channel 2 ≙ 0x0001

Fragment Number: The histogram data of each channel is 4096 channels x 32 bits.
The data will be send in 12 UDP packets.
UDP packet with fragment number 0-10 has each 354 channel data.
UDP packet with fragment number 11 has 202 channel data.

Uptime: System Uptime

### 3.6.5 USB Interface

FTDI FT2232H (See FTDI homepage for compatible drivers)
Used FT245 Interface

Address: 1-Bit Command Type + 15-Bit address

Command: '0' ≙ Read
'1' ≙ Write

#### 3.6.5.1 Read Access

Request:

| | |
|---|---|
| (MSB) 1-Bit Command Type + 15-Bit address | 0x0011 |

Answer:

| | |
|---|---|
| Data | 0xC0A8 |

The lower Byte of the 16-Bit Word is send first.

#### 3.6.5.2 Write Access

Request:

| | | |
|---|---|---|
| (MSB) 1-Bit Command Type + 15-Bit address | 0x8821 | Word 0 |
| Data | 0xAC1C | Word 1 |

No Answer.

The lower Byte of the 16-Bit Word is send first.