Firmware:

MQDC-32

For gating with experiment trigger the firmware MQDC32_V16_FW0200 (or higher) is recommended.

The modifications:

1. experiment trigger always creates an event frame; this helps to keep a synchronous event structure when several mudules are read out with close synchronisation.
2. new register 0x601C: allow to change interrupt source from data amount in FIFO to event count in FIFO
3. new register 0x601E event count threshold, irq may be emitted when events in FIFO exceed this threshold
4. register 0x606E nim0 output can now be set to value 4, which allows to signal when the event count exceeds the threshold 0x601E.
5. When ECL1 is set as experiment gate, and ECL is not used for gates, (NIM2 and 3 as gate1 and 0) then ECL3 is also experiment gate input ("OR" of the two inputs is used internally). This simplifies ECL bus mechanics.

MTDC-32, MADC-32 with firmware FW0200 event counting is added:

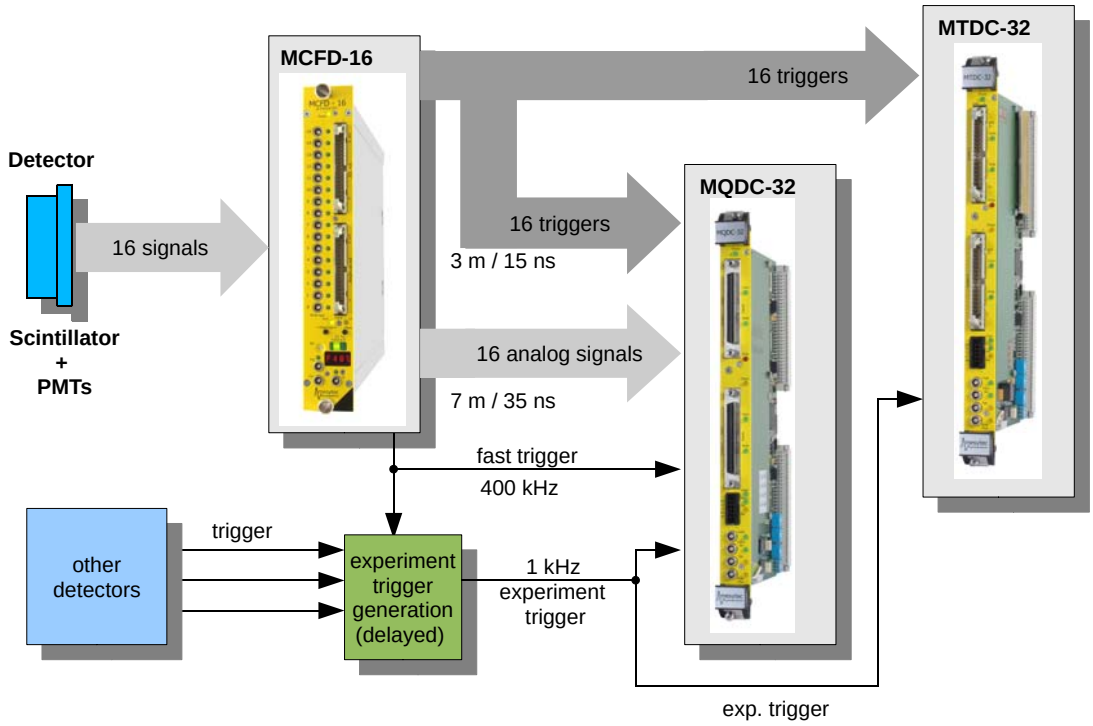
1. new register 0x601C: allow to change interrupt source from data amount in FIFO to event count in FIFO
2. new register 0x601E event count threshold, irq may be emitted when events in FIFO exceed this threshold
3. register 0x606E nim0 output can now be set to value 4, which allows to signal when the event count exceeds the threshold 0x601E.

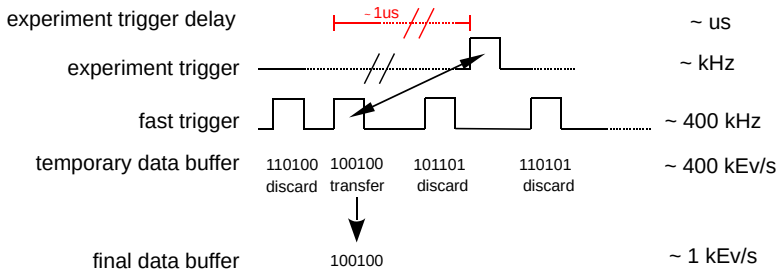
MCFD16

When RC-bus is used, MCFD should run with processor firmware MCFD16_V2_18

Example1

Signal processing of a 16 channel scintillator , light converted with a multi anode photo multiplier.





The 16 channel from PMT are connected vie 16 lemo cables to the inputs of an MCFD-16 discriminator.

The fast OR output is directly used as a free gate for MQDC-32 (direct short Lemo connection). This helps to keep the required cable delays as short as possible.

Trigger 1..3 output of the MCFD-16 can be used to create an experiment trigger, which is fed to the experiment trigger input of the MQDC-32 and is also used as trigger for the MTDC-32. The delay of experiment trigger is not critical.

This can also be done with many modules. In this case the fast OR from MCFDs feed the individual gate inputs (best is NIM inputs) of the MQDC-banks (operate MQDC in split bank).

The common experiment trigger can be distributed to MTDC-32 via ECL -T0 input and to the MQDC via ECL1 experiment trigger input (3. from top).

Readout is done event by event. In the present firmware the experiment gate does only produce an event data structure and increments the event counter when coincident converted data are found. This will be modified in the next firmware revision (now ready, FW0200). Then an empty frame is sent when no

conversion is found.

Connection: The QDC ECL outputs are wired to the QDC gate inputs and then to the TDC inputs. So QDC gate inputs should be unterminated, TDC inputs are terminated (TDC input jumpers set to "terminated").

The Analog signals from CFD output are wired to the analog input of the QDC. CFD output and QDC input should be differential, the QDC input jumpers must be differential and set to negative polarity. The cable length must be adapted to the required delay of the analog signal.

The experiment trigger is generated from signals of all CFDs and maybe triggers of other detectors, and is the converted to the main trigger in ECL. This signal is distributed to all TDCs and QDCs. As the experiment gate of QDCs do not require a very strict timing, it may also be a second driver output of the event gate.

The QDC event gate should have the length required for the coincidence in the experiment, but at least 25ns.

Register setup MQDC:

All IOs are counted from bottom to top, starting with 0.

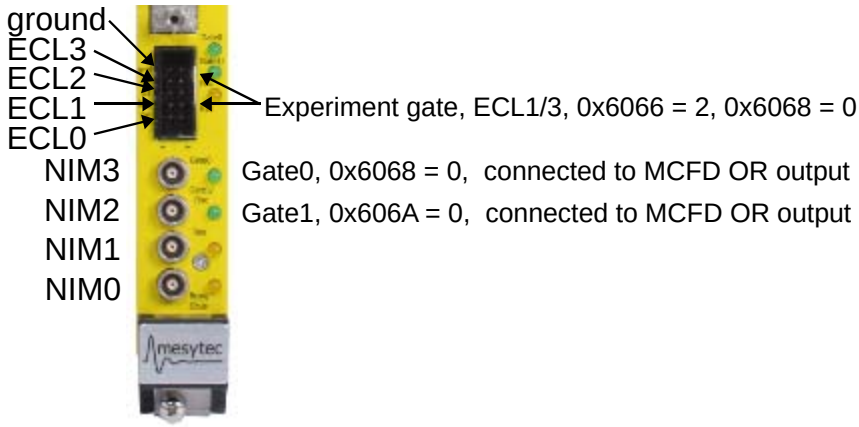
Thresholds are not required, when individual gates from CFD are used. Then the CFD thresholds determine when a gate is created and QDC conversion is started.

IO setting

ECL buses should always include ground, but should **not include** the ECL outputs.

So easiest way is to use an 8 wire flat cable from ECL1 to ground and to connect it through all modules and finally to the ECL driver. MQDC32, MTDC32 and MADC32 have the same ECL and NIM connector layout.

When experiment gate is used as trigger 0 of MTDC-32, the ECL bus has to be reduced after MQDCs from 8 wires to 4 wires (2x ground, 2x experiment gate).



Set NIM3 = gate0, NIM2 = gate1

0x6068 = 0 //Gate_select → Gate 0 and 1 from NIM-inputs

0x606A = 0 // NIM_gat1_osc, Lemo2 is → gate1 input

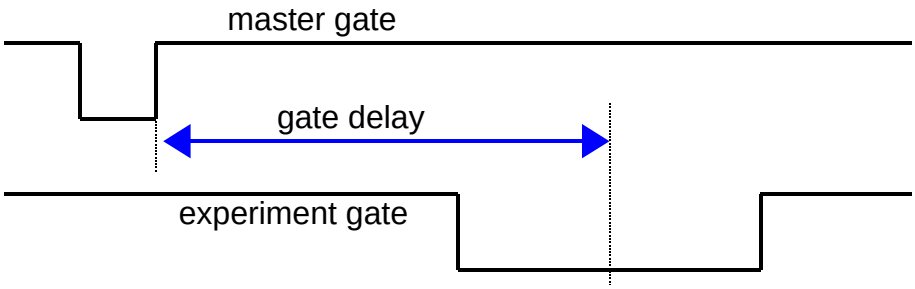
Set ECL1 for Experiment gate. This also activates the experiment gating mode.

0x6066 = 2 // ECL_FC_Reset -> ECL1 is input for experiment gate

ECL terminators: For all MQDC modules except the last one
0x6062 = b00000 = 0 // ECL_term: ECL1..3 all inputs unterminated
For the last module, terminate Experiment gate input:
0x6062 = b00100 = 4 // ECL_term: ECL2 and3 unterminated, input 1 terminated

Split bank operation, one MCFD-16 supplies signals to one bank of MQDC-32
0x6040 = 1 // bank_operation → operate banks independently

Experiment gate delay (exp. gate width determines the coincidence time, delay is adjusted here)
0x6054 = 1000 // exp_trig_delay bank0 -> experiment trigger comes 1000ns delayed
0x6056 = 1000 // exp_trig_delay bank1 -> experiment trigger comes 1000ns delayed



For the experiment gate the minimum low time (active) = 25ns, the minimum high time = 50ns.

Then register setting for FIFO control and readout is required

Register setup MTDC:

The MTDC-32 is operated in connected bank mode. All timings over all modules are differences between the time stamp. So Jitter of the trigger 0 will cancel out as long as the same trigger is fed to all modules. Here trigger1 can be used as an additional timing channel.

IOs

0x6062 = 0 //ECL_term -> bit 0 for: "trig0"
Last module terminates (0x6062 = 1)

0x6068 = 1 // Trigger 0 from ECL3-input

Window of interest

0x6050 = 0x4000 - 1000 //bank0_win_start. trigger0 is delayed by 1000ns
// so shift window of interest back in time
0x6054 = 500 //bank0_win_width. 500ns window width

Trigger source

0x6058 = 1 // trigger 0 input triggers the window of interest

Data conversion

0x6042 = 4 // set resolution to 32ps

0x605C = 0 // only transmit the first hit per channel in the window of interest.

Then register setting for FIFO control and readout is required

FIFO Setup, all modules

One TDC can be initialized to signal data via Interrupt:

0x6010 = 1 // IRQ = 1, vector = 0 (default)

0x6018 = 1 // signal an interrupt when data detected

All modules can then be initialized the same way

0x6036 = 0 // event by event transfer

0x603A = 1 // start DAQ (default)

0x603c = 0 // clear FIFO

0x6034 = 0 // readout reset, allow new event

Readout loop

at IRQ:

read each module until Berr

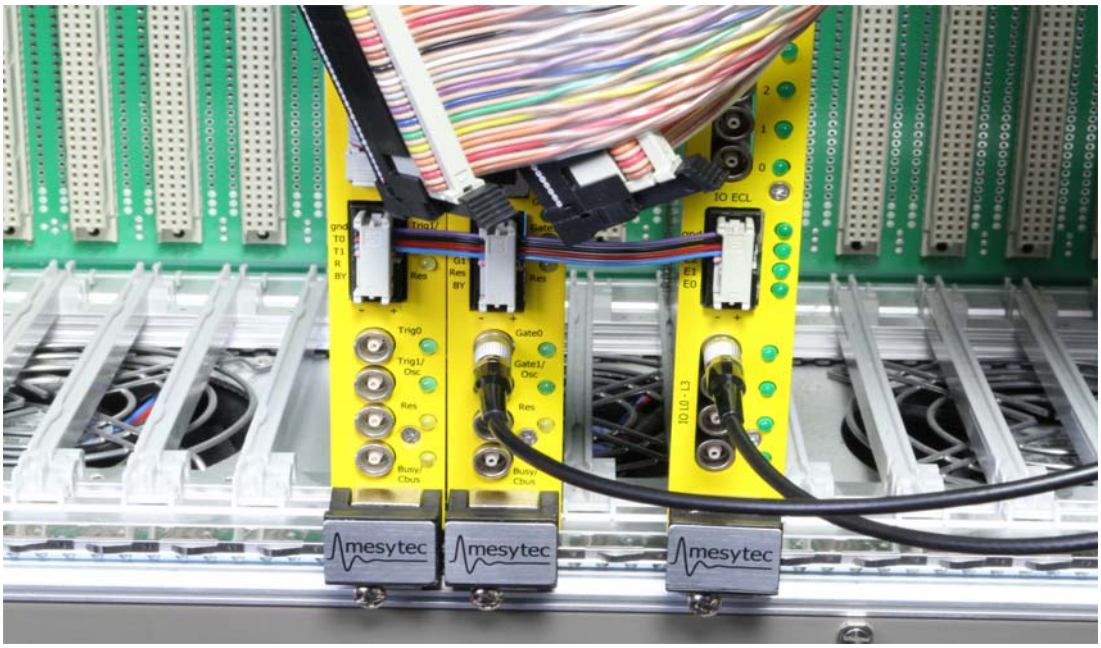
write 0x6034 for next event.

Example 2

With new firmware FW0200

Setup as before, 32 channel timing and amplitude readout with 2x MCFD-16, 1x MTDC-32, 1x MQDC-32. readout with buffered events while keeping them synchronous from both modules. Readout in CBLT.

In the lower picture a crimp connector is added to the standard interface cable to connect to the MTDC input.



The picture shows the ECL bus which provides the experiment gate to MQDC and MTDC.
it is terminated at the end.

Initialization list

MTDC input jumper: differential, set to front position -> terminated

```
Base_address = 0x0000
0x6010 0x001          // signal event available with IRQ1
0x6036 0x000b         // limit output by number of events
0x601A 0x001          // send one event before continue with CBLT chain
0x6020 0xA2           // this module is first in CBLT chain
0x6042 5              // TDC resolution 31ps
0x6050 15000          // start window of interest at -1.384us
0x6054 1500           // window width is 1.5us
0x6058 0x1            // start window of interest with trigger0
0x6062 1              // terminate ECL3 input
0x6068 1              // use ECL inputs for triggers (Trig0 is at ECL3)
```

MQDC: use differential 100Ohm input jumpers, front position for negative analog signals.
Gate inputs not terminated (default).

```
Base_address = 0x0100
0x6010 0x01           // signal by IRQ1
0x6036 0x000b         // limit output by number of events
0x601A 0x002          // send two events (one of each bank) before sending Berr.
0x601C 0x0            // init IRQ with event threshold
```

0x601E 0x3 // init IRQ when at least 3 events in the buffer
0x6020 0x8A // CBLT last module
0x6040 1 // split banks
0x6066 2 // use ECL1 and 3 for experiment gate
0x6068 0 // use NIM3 and 2 for Gate 0 and 1;

Both modules, multi cast
Base_address 0xbb00 // default multi cast base address (when not explicitly set)
0x6090 3 // reset event counter
0x603C 0x0 // reset fifo
0x6034 0x0 // reset event

One event:
read out by waiting on IRQ, and reading CBLT chain (default address 0xAA000000) to Berr. Event number show here is 0x1802;

0: 40010001 //**MQDC bank0,** empty frame
1: c0001802 // EOE, event counter
2: 40010003 //**MQDC bank1**
3: 04000a90 // channel0 amplitude
4: 00000000 // fill word
5: c0001802 // EOE, event counter
6: 40005003 // **MTDC**
7: 0420ad00 // trigger0 event at address 32
8: 0400accd // channel 0 timing
9: c0001802 // EOE, event counter