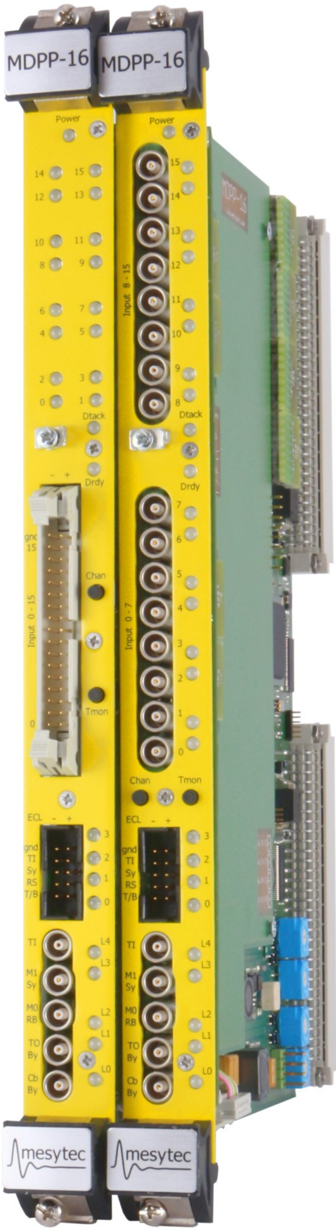
Software Module MDPP-32-QDC

V0010

32 channel VME pulse processor

The software module MDPP-32-QDC provides the functionality of a fast charge integrating ADC , a Zero-Crossing detector+TDC and a pulse shape discrimination unit. It also works with short pulses of plastic scintillators and provides neutron/gamma pulse shape discrimination with liquid scintillators

MDPP-32 with QDC software module:



- **Gain-polarity jumpers**

  determine: termination, polarity, input range and input configuration (differential / unipolar).

  Special QDC jumpers available to get best amplitude resolution for plastic scintillators.
- **Low noise variable gain input amplifiers.**

  Input signals for maximum range (highest spectrum channel)

  Plastic scintillators pulse width 5ns (QDC jumpers 2V): 200mV to 5V (20pC to 500pC)

  LYSO, Pulse width 30ns (any input jumper possible)

  15mV to 3V (10pC to 1.5nC) (other jumpers on request)
- **Timing resolution**

  down to 80ps channel to channel resolution,

  TF integration / differentiation time down to 15ns
- **Pulse shape discrimination**

  delivers two amplitudes: one is the standard integrated pulse amplitude (integration time 25ns to 6.4us), the second is a short integration of the rising edge of pulse. Integration as short as 12.5 ns to 1600 ns. This is fast enough to give good neutron / gamma separation for liquid scintillators

  Also works for Stilbene, CLYC, CsI.
- **Sample trail output**, up to 1000 samples per channel
- **AC-coupled and baseline restored**

  Offsets of the input signals have no effect. Even at highest rates, the amplitude keeps stable.
- **Dead time / rate capability**

  Channel dead time is 400ns, (integration time up to 300ns included). For longer integration time dead time is integration time + 100ns.
- **Two high resolution monitor outputs**

  for monitoring input signals and integrals of signals.
- **Two high resolution trigger inputs**

  24 ps resolution, start window, add time stamp
- **One high resolution trigger output** (1.5 ns resolution)
- **Installation and update via USB**

mesytec GmbH & Co. KG

Wernher-von-Braun- Str. 1, D-85640 Putzbrunn, Germany

phone: +49 - 89 / 456007-30 fax: +49 - 89 / 456007-39

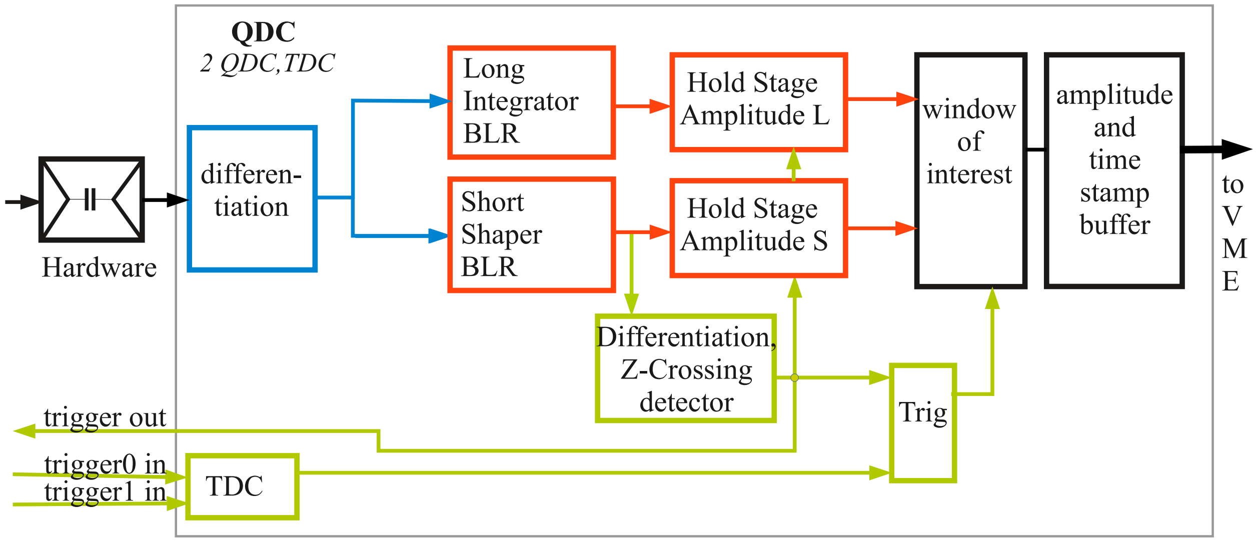
info@mesytec.com - www.mesytec.com

Software module: "QDC"

Delivers timing and amplitude and pulse shape analysis for signals from fast charge amplifying Detectors. For Example Photo multipliers, GEM, Channel plate..

Replaces Fast amplifier, CFD, QDC, TDC.

The following picture shows a schematic representation of the software:



The signal is amplified filtered digitized. Then it is split into two branches: short integration and long integration.
In the "short integration" branch the signal is differentiated with a time constant down to 12.5ns, followed by an integrator. This allows to extract a very short time interval at the beginning of the pulse, which even allows to perform pulse shape discrimination with liquid scintillators. The short shaped path is also used for timing. The short pulse is discriminated to get an amplitude independent zero crossing signal. This one is used to calculate the timing.
The "long integration" path is the QDC-(Charge to digital converter) part of the processing software. Usually the full input pulse is integrated here. Integration times of 25 ns to 1600 ns are possible. As the input is AC-coupled a baseline restorer is required, to preserve a stable pulse amplitude at very high rates. The integrated signal is sampled at a well defined time, determined by the timing branch.

Then the two amplitude and timing values are filtered by a **window of interest** and stored in a **buffer**.

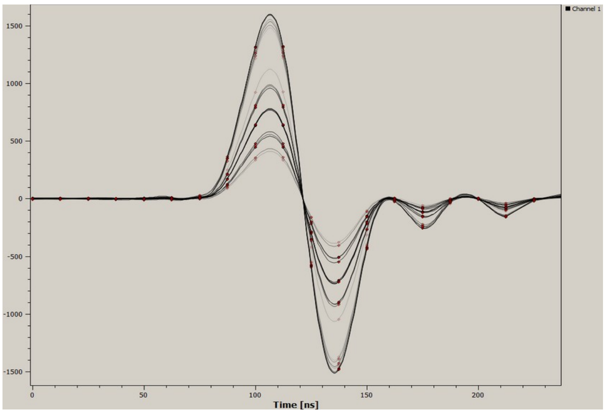
Short data:

- Amplitude resolution of up to 4 k (12 bit)
- Trigger to channel time resolution of **60 ps rms,** uniform at any delay.
- Channel to channel time resolution of **60 ps rms**, uniform at any delay.
- Trigger input with 24 ps timing resolution
- Can be operated self triggered or externally triggered
- Outputs internal raw trigger with 1.5 ns time resolution

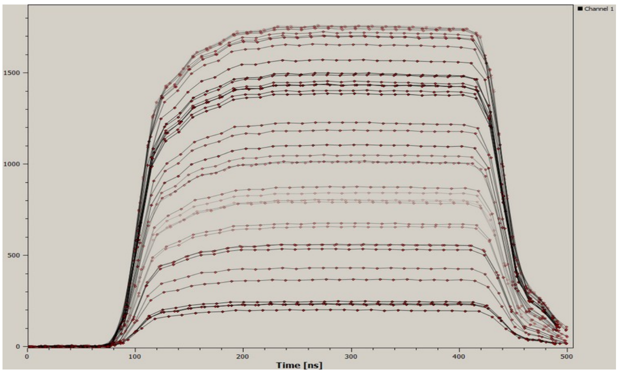
As easy to operate as all mesytec modules and fully data compatible.

Get sample trails

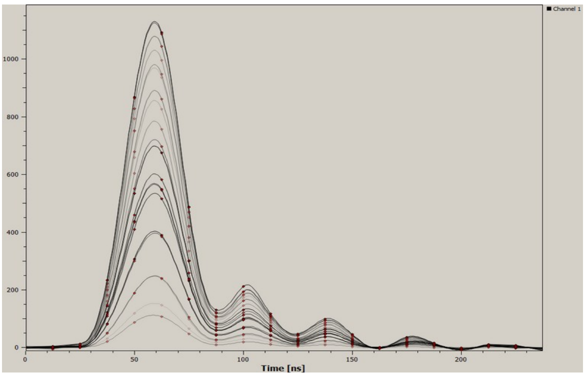
Sample trails are available for “SCP” and “QDC” software since Package SP0050.
In the updated software packets “SCP” for charge integrated signals and “QDC” for direct PMT and similar high gain sources, an extension is implemented allowing to add up to 1000 sample values per channel (12.5us trail length).
For a trail length of up to 500 samples, two traces per channel can be registered simultaneously, minimising the channel dead time. It is possible to specify a number of pre samples (before the input signal edge) and any number of samples (up to 1000) required. Up to 80 M Samples/s can be transmitted to the VME bus when 2ESST readout mode is used. This results in up to 320 MBytes/s of data transferred via VME bus.
The sample trails are simply added to the channel data. The powerful trigger system via window of interest, internal/external triggering is unchanged. The high resolution timing and amplitude(s) are emitted simultaneously.
Sample source can be set to direct ADC signal, and to several internally processed signals.
The following pictures show sample trails from “SCP” software with different sources selected. They can be displayed by mvme software in “event history” display.



Short integration output, triangular shaped bipolar



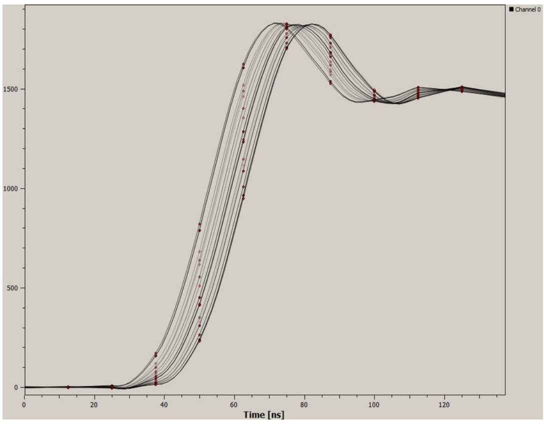
Long integration signal for amplitude



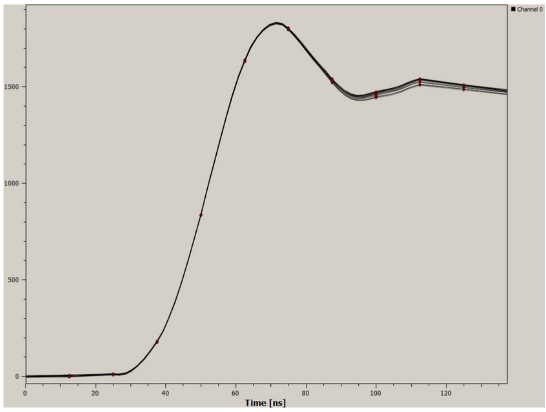
Directly from ADC

Resampling

Usually the input signals are asynchronous to the sampling frequency. This results in a jitter of the digitized signal by the sampling period in time. To get a stable display there are two possibilities:
1) MDPP can do a resampling. So it calculates new sampling points related to the precise time measured by the digital discriminator in the timing path.
2) In the sampling header, a “phase” showing the time difference between discriminator time and sample time is transmitted. In the mvme display the curve is shifted by this time to give a stable display as usually shown by an oscilloscope.



Display of 20 traces without resampling



Display with resampling in MDPP-16

Interpolation / reconstruction

A sin(x)/x interpolation (order 6 is used here) of traces requires four samples before and 3 samples behind the interpolated time slice between two samples. So in the mvme display (first approach) the first 3 and last 3 samples are interpolated linear. So for a time slice to be precisely interpolated there are 6 neighbour samples needed.
That’s not required for resampling in MDPP-16. For the integrated resampling filter any number of samples is available. So for data reduction it is even possible to transmit only two samples to determine the rise time of a signal.

Offset correction

The 4 samples before transmitted samples are used for offset subtraction: mean value of the 4 samples is subtracted from all following samples.

Only 8 parameters have to be set:

In Hardware:

Polarity of the signal,
set Jumper to correct position

Register Settings

Signal properties:

1. signal width [ns]
2. maximum signal amplitude [mV]

Analysis property

3. required long integration
4. required short integration
5. threshold

Hardware Property:

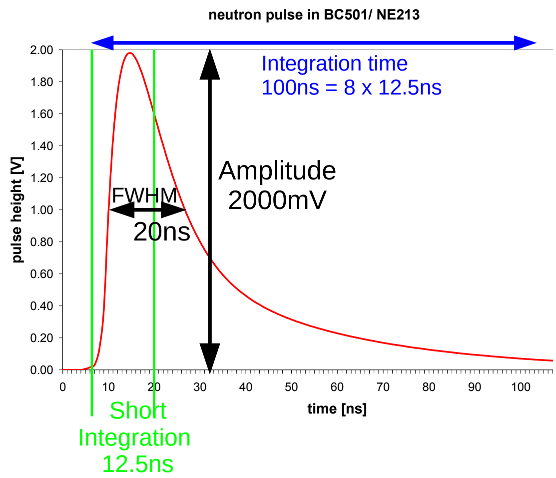
Gain jumper sensitivity [mV]
QDC-Gainjumper used ?

Output Data

Amplitude:

channel 0..31 Amplitude Long int. (16 bit)
channel 64..95 Amplitude Short int. (16 bit)
Timing: Difference to window start:
channel 32 to 63 channel time difference (16bit)
Chan 96,97 Trigger input 0,1 time diff. (16 bit)

Example: The following diagram shows the charge pulse as it is produced by a liquid scintillator when detecting a neutron interaction.



The initialization data for MDPP-32-QDC are:

Signal_width = 20
Input_Amplitude = 2000
Jumper_range = 3000 // for 3V-Jumper
QDC_Jumper = 0 // 0 = standard jumper
Integration_long = 8
Integration_short = 1
threshold0 0x200 // = 1/128 of full range 0xFFFF

For direct charge signals from PMTs connected to CsI crystals, a signal widt of 600ns, short integration of 400ns (= 32 clocks) and long integration of 2.0us to 4.0us (150 to 300 clocks) performs best.

Monitor outputs

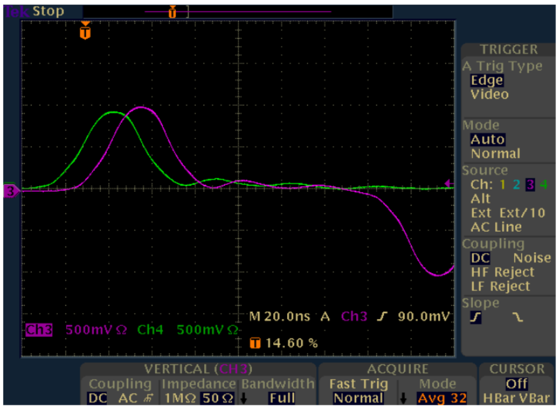
(Lemo 2 = mon 0, and Lemo 3 = mon 1)

Switching on the monitor: press pus button "chan", then select a wave form with "Tmon" button. The button "chan" allows to switch through the individual channels.

Wave forms:

Tmon 0,

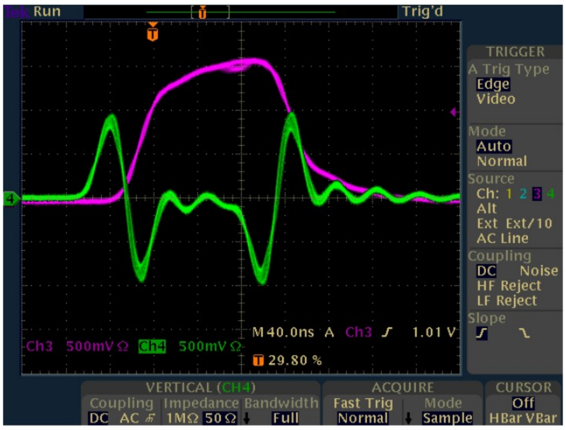
Green: mon0, signal before ADC,
Magenta: mon1, differentiated input signal



Tmon0, green=inp.,mag.=differentiated

Tmon 1:

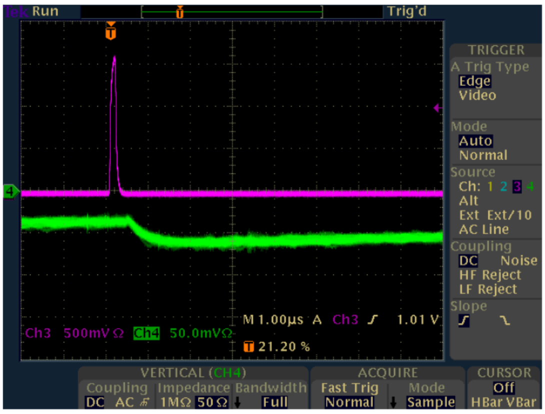
Green: mon 0, short integration signal. The first positive maximum is converted.
Magenta: mon 1, long integrated signal.
The flicker mark shows the sampling time.



Tmon1, green=Shortint; mag.=Longint

Tmon 2 :

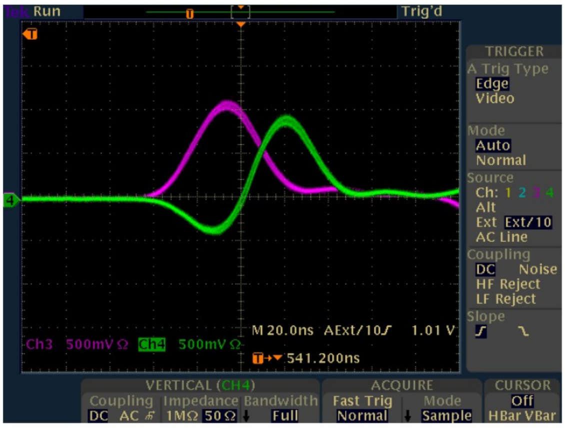
Green: mon0, baseline from BLR, multiplied x8.
Magenta: mon 1, Long integration signal.



Tmon2, green=Baseline, mag. =Longint

Tmon 3: Check **noise**

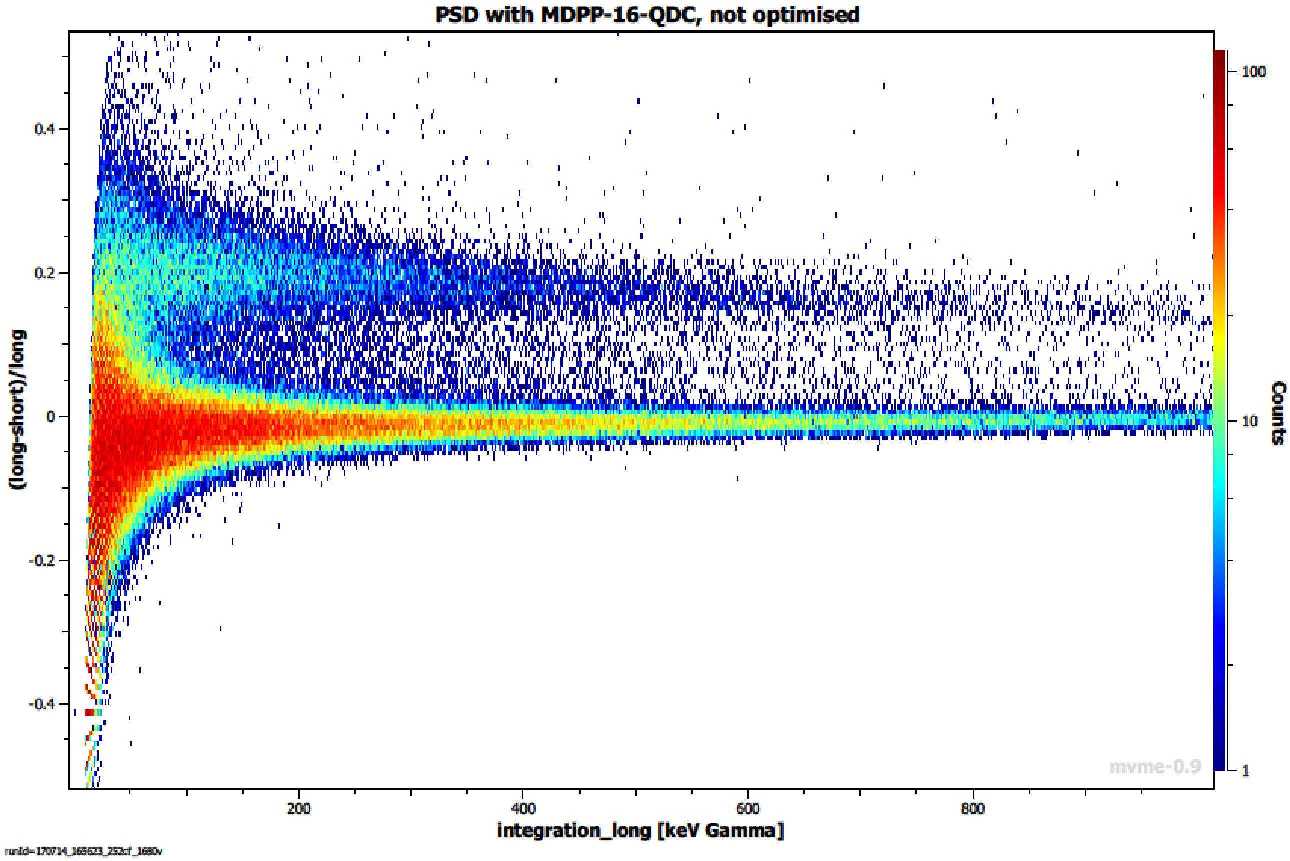
Green: mon 0, CFD delayed subtracted signal. Zero crossing triggers timing (delayed compared to mon1 signal).
Magenta: mon 1,timing filter signal.



Tmon3, green=Z-cross, mag.=TF-out

Measurement done with MDPP-16 QDC software.

(MDPP-32 behaves very similar.)



MDPP-32 register set, QDC Firmware.
Only registers which are different to SCP software modules are listed.

Data FIFO, read data at address 0x0000 (access R/W D32, 64)

only even numbers of 32 bit-words will be transmitted. In case of odd number of data words, the last word before End of event mark will be a fill word (= 0).
FIFO size: 48 k - 512 = 48640 words with 32 bit length

Header (4 byte)

2 header signature	2 subheader	3	1 SA	8 module id	3 TDC_resolution → 0x6042	3	10 number of following data words, including EOE
b01	b00	b000	0	module id	bxxx	bxxx	number of 32 bit data words

Sampling mode:

b01	b00	TDC_resolution	1	module id	16 bits, number of 32 bit data words

Data (4 byte) DATA event

2 data-sig	2	4	1	7	16
b00	01	xxxx	overflow	channel number 0..31	ADC value long integration

Data (4 byte) DATA event

2 data-sig	2	4	1	7	16
b00	01	xxxx	overflow	channel number 94..125	ADC value short integration

Data (4 byte) DATA event

2 data-sig	2	5	7	16
b00	01	xxxxx	channel number 32..63	TDC time difference

Data (4 byte) DATA event

2 data-sig	2	5	7	16
b00	01	xxxxx	channel number 64..65	Trigger time difference External trigger input T0, T1

Data (4 byte) **Extended time stamp**

2 data-sig	2	12	16
b00	10	xxxx xxxx xxxx	16 high bits of time stamp

Data (4 byte), fill dummy (to fill MBLT64 word at odd data number)

2 data-sig	30
b00	0

End of Event mark (4 byte)

2	30
b11	event counter / time stamp

trigger counter/ time stamp 30 bit trigger counter or time stamp information, depending on register 0x6038 “marking type”: 0 = event counter, 1 = time stamp

Sample Transmission

From QDC Firmware packet FW3050 samples can be added to the data stream by setting Register 0x6044 to 16 or 24 (streaming mode) and specifying sampling parameters in reg 0x6146 to 0x614A.
Samples follow the time word:

2 data-sig	2	6	6	16
b00	b01	xxxxxx	channel number 16..31	TDC time difference
b00	b11	Sample header[27:0]		
b00	b11	Sample1[13:0], Sample0[13:0]		
..	..	..		
b00	b11	Sample_2N[13:0], Sample_2N-1[13:0]		

Sample header: { 0, sample_config[7:0], phase[8:0], following_sample_words[9:0] }

Following_sample_words[9:0]: (= N) up to 500 sample pairs may follow = 1000 samples.

Phase[8:0]: a phase with range 0 to 511 is transmitted, showing the distance between trigger point to the ADC sampling time. If “resampling” is active this word is not needed. If resampling is switched off, it can be used to shift samples for jitter free display of wave forms.

Sample config[7:0]: { not_offset, not_resample, 0, 0, 0, 0, source[1:0]}
source = 0 Samples directly from ADC
SCP
source = 1 Samples from reconstructed input signal
source = 2 Samples from timing filter shaper
source = 3 Samples from main shaper

QDC

source = 1 Samples from short integration
source = 2 Samples from long integration

Extract samples:

-extract time word, extract channel address from time word
-next word with data[31:28] == 0x3 is the sample header
-following words with data[31:28] == 0x3 are the samples for the detected channel

QDC software allows two “streaming modes” from Firmware revision FW3050.

Setting Register 0x6044 to 4 switches on compact streaming mode, 0x6044 = 8 the standard streaming mode. In streaming mode the timing of MDPP-16 is synchronised to the VME back plane clock.
All 16 channels and the 2 trigger inputs are now completely independent and output an independent event frame. Window of interest is switched off. The event frame is 2 words long

Compact streaming mode:

Header (4 byte)

2 header signature	6 module id	6 Address[4:0] {trig_flag,0,address}	2 Pile_up, overflow	16 Amplitude
b01	module id	T0AAAA	P,O	Long integrated Ampl[15:0]

For Trigger events (address 0b10000 and 100001) the number of events, which were skipped due to buffer overflow since last trigger is emitted in at the place of “amplitude”.

End of Event mark (4 byte)

2	30
b11	Fine time stamp (set in reg 0x6042)

Standard streaming mode

Setting Register 0x6044 to 8 or 24(with samples)
Has the same data format as standard mode described on the page before, but only one channel (long + short amplitude) or trigger (address 0b10000 and 100001, skipped events[15:0]) is emitted in a packet. The “module time stamp” is the precise time stamp set in register 0x6042. Also extended time stamp with extra 16 bits is possible. As for both streaming modes, time is the precision time with resolution set with reg 0x6042

Registers

	operation mode				
0x6044	output_format	2	RW	3	0 = standard window of interest mode. from FW3050: 4 = compact Streaming mode 8 = standard streaming mode 16 = window of interest with samples 24 = standard streaming with samples.
0x6046[1]	adc_resolution[1]			4	0 = 64k **for new setups Always use 64k setting !** 1 = 32k 2 = 16k 3 = 8k 4 = 4k (default)

Channel addressing (select channel which are set)

0x6100	select_chan_quadruples	4	RW	8	channel to be modified: 0..7 channel quads; 0 = chan 0,1,2,3 1 = chan 4,5,6,7 8 = all channels (set to common values)

Channel settings for Quads of channels,

****** After writing a register in this page, 20 us wait time is required ********

Address	Parameter				
0x6110	Signal_width	10	RW	16	[FWHM, ns],
0x6112	Input_Amplitude	16	RW	1024	[mV], input amplitude 0 to peak in mV max value is the jumper range
0x6114	Jumper_range	16	RW	3072	[mV], Range printed on jumper top
0x6116	QDC_Jumper	1	RW	0	1= yes, 0=no
0x6118	Integration_long	9	RW	16	[12.5 ns], 2..508[1] in clock steps of 12.5ns
0x611A	Integration_short	7	RW	2	[12.5 ns], 1..120[1] in clock steps of 12.5ns must be < than long integration Should be <= 2x Signal width to avoid post triggers after integr_long time.
0x611C	threshold0	15	RW	0xFF	1...0xFFFF; example: 0.8% = 0x200;
0x611E	threshold1	15	RW	0xFF	
0x6120	threshold2	15	RW	0xFF	

0x6122	threshold3	15	RW	0xFF	
0x612A	long_gain_correction	12	RW	1024	256: divide by 4, 4096 multiply by 4, 1024 neutral;
0x612E	short_gain_correction	12	RW	1024	

0x6146	pre_samples	10	RW	4	0..1000, samples before trigger
0x6148	tot_samples	10	RW	12	0...1000, total number of samples
0x614A	sample_config	8	RW	0	{no_offset_corr, no_resamp,0,0,0,0,source[1:0]} source source = 0 Samples directly from ADC source = 1 Samples from short int source = 2 Samples from long int

Note(1), Since March 2021

How to set channel parameters

Signal_width This is the width of the input pulse at half the peak amplitude in ns. The pulse must be measured with a **terminated** oscilloscope. Do not measure at the monitor output of MDPP-32 ! it does not have the band width to properly measure the pulse width;

Input_amplitude: This is the amplitude of the input pulse measured from base line to peak in mV. Offsets of the signal have no effec. The pulse must be measured with a terminated oscilloscope. Do not measure at the monitor output of MDPP-32 ! it does not have the band width to properly measure the pulse width;

Jumper range: The value is printed on the jumper. multiply the value by 1000 to get the mV unit.

QDC_Jumper: set 1 if you used a QDC-Jumper;
Those jumpers are obsolete. MDPP-32 hardware has been changed to give best results with standard jumpers.

Integration_long: is the integration time to get the full charge of the input pulse.
It is specified in multiples of internal clocks, so 12.5ns. Allowed range is 2 (25ns) to 127 (1.6us)

Integration_short: is the integration time to get the first fast part of the input pulse.
It is specified in multiples of internal clocks, so 12.5ns. Allowed range is 1 (12.5ns) to 31 (387ns)
It must be smaller than the long integration time.

Threshold 0 to 3: The threshold parameter can be set separately for the four addressed channels.
Foll range is 64k (65535) = 0xFFFF; So a 1% of full range threshold is 65535/100 = 655;

Gain_corrections: The internal gains and hardware gain are calculated based on the signal width and amplitude. This should give a quite good start value. Details of the signals will have an effect on the real amplitude. So there are 2 scaling factors to correct the gains. The default of 1024 is the neutral setting, a lower value decreases the amplitude in a spectrum, a higher one increases it. It is possible to decrease the gain by a factor of 4 (->256), or increase it by a factor of 4 (-> 4095).