

MPD Dualword Event Format

**Benjamin Raydo
Paolo Musico
May 5, 2021**

1. Readout Data Format

MPD firmware used by the SSP readout can still be used in VME readout mode. It contains two APV words per 32bit word to increase the effective event rate limit when transporting to the SSP, but this benefit is also something the VME readout of the MPD can benefit from as well. This format doesn't support data processing (e.g. zero suppression) as the format assumes all 128 strips per APV will be reported each event (and the freed resources went towards increasing the number of events the MPD can buffer to help improve DAQ performance with SSP readout).

The repository is here:

https://github.com/braydo/Fpga_4_Fiber/tree/NoZeroSuppression

Firmware March 15, 2021 and later have this 32bit dual word format support (e.g.):

Mpd4/Fpga_4_Fiber_210315.jic

Note that these data types utilizes a 32bit format that is distinguishable from the older MPD 24bit data type present in some other firmware releases.

Data Word Categories

Data words from the module are divided into two categories: Data Type Defining (bit 31 = 1) and Data Type Continuation (bit 31 = 0). Data Type Defining words contain a 4-bit data type tag (bits 30 - 27) along with a type dependent data payload (bits 26 - 0). Data Type Continuation words provide additional data payload (bits 30 - 0) for the *last defined data type*. Continuation words permit data payloads to span multiple words and allow for efficient packing of various data types spanning multiple data words. Any number of Data Type Continuation words may follow a Data Type Defining word.

Data Type List

0	Block Header
1	Block Trailer
2	Event Header
3	Trigger Time
4	APV Data
5	Event Trailer
6	Reserved
7	Reserved
8	Reserved
9	Reserved
10	Reserved
11	Reserved
12	Reserved
13	Reserved
14	Data Not Valid (empty module)
15	Filler Word (non-data)

Data Type: Block Header

Type: 0x0
 Size: 1 word
 Description: Indicates the beginning of a block of events. (High-speed readout of a board or a set of boards is done in blocks of events)

31	30	29	28	27	26	25	24
1	0	0	0	0	SLOTID		
23	22	21	20	19	18	17	16
SLOTID		UNDEFINED			EVENT_PER_BLOCK		
15	14	13	12	11	10	9	8
EVENT_PER_BLOCK					UNDEFINED		
7	6	5	4	3	2	1	0
BLOCK_CNT							

BLOCK_CNT:

Event block number (used to align blocks when building events)

EVENT_PER_BLOCK:

Number of events in block

SLOTID:

Slot ID (set by VME64x backplane or Rotary switch when != 0)

Data Type: Block Trailer

Type: 0x1
 Size: 1 word
 Description: Indicates the end of a block of events. The data words in a block are bracketed by the block header and trailer.

31	30	29	28	27	26	25	24
1	0	0	0	1	SLOTID		
23	22	21	20	19	18	17	16
SLOTID		UNDEFINED			NUM_WORDS		
15	14	13	12	11	10	9	8
NUM_WORDS							
7	6	5	4	3	2	1	0
NUM_WORDS							

NUM_WORDS:

Total number of words in block of events

SLOTID:

Slot ID (set by VME64x backplane or Rotary switch when != 0)

Data Type: Event Header

Type: 0x2
 Size: 1 word
 Description: Indicates the start of an event. The included trigger number is useful to ensure proper alignment of event fragments when building events.

31	30	29	28	27	26	25	24
1	0	0	1	0	UNDEFINED		
23	22	21	20	19	18	17	16
UNDEFINED				TRIGGER_NUMBER			
15	14	13	12	11	10	9	8
TRIGGER_NUMBER							
7	6	5	4	3	2	1	0
TRIGGER_NUMBER							

TRIGGER_NUMBER:

Accepted event/trigger number

Data Type: Trigger Time

Type: 0x3

Size: 2 words

Description: Time of trigger occurrence relative to the most recent global reset. The time is measured by a 48bit counter that is clocked from the 40MHz system clock. The assertion of the global reset clears the counter. The de-assertion of global reset enables counter and thus sets t=0 for the module. The trigger time is necessary to ensure system synchronization and is useful in aligning event fragments when building events.

Word 1:

31	30	29	28	27	26	25	24
1	0	0	1	1	UNDEFINED		
23	22	21	20	19	18	17	16
TRIGGER_TIME_H							
15	14	13	12	11	10	9	8
TRIGGER_TIME_H							
7	6	5	4	3	2	1	0
TRIGGER_TIME_H							

TRIGGER_TIME_H:

This is the upper 24bits of the trigger time

Word 2:

31	30	29	28	27	26	25	24
0	UNDEFINED						
23	22	21	20	19	18	17	16
TRIGGER_TIME_L							
15	14	13	12	11	10	9	8
TRIGGER_TIME_L							
7	6	5	4	3	2	1	0
TRIGGER_TIME_L							

TRIGGER_TIME_L:

This is the lower 24bits of the trigger time

Data Type: APV Data

Type: 0x4

Size: 1+64 words

Description: This data type contains a header + complete APV data frame: 1 header + 64 words follow with each of the words following the header containing 2 samples per 32bit word.

Word 1:

31	30	29	28	27	26	25	24
1	0	1	0	1	APVID		
23	22	21	20	19	18	17	16
APVID		SAMPLE_COUNTER			FRAME_COUNTER		
15	14	13	12	11	10	9	8
FRAME_COUNTER				APV_HEADER			
7	6	5	4	3	2	1	0
APV_HEADER							

APVID:

APVID from 0 to 15

SAMPLE_COUNTER:

Time sample 0 to 5

FRAME_COUNTER:

Sequential APV frame counter

APV_HEADER:

APV chip frame header

Word 2 to 65:

31	30	29	28	27	26	25	24
0	UNDEFINED					APV_SAMPLE1	
23	22	21	20	19	18	17	16
APV_SAMPLE1							
15	14	13	12	11	10	9	8
APV_SAMPLE1				APV_SAMPLE0			
7	6	5	4	3	2	1	0
APV_SAMPLE0							

APV_SAMPLE0:

13bit signed APV sample number (Word#-2)*2+0

APV_SAMPLE1:

13bit signed APV sample number (Word#-2)*2+1

Data Type: Event Trailer

Type: 0x5
Size: 1
Description: event trailer

Word 1:

31	30	29	28	27	26	25	24
1	0	1	0	1	UNDEFINED		
23	22	21	20	19	18	17	16
EVENT_LENGTH							
15	14	13	12	11	10	9	8
EVENT_LENGTH				UNDEFINED			
7	6	5	4	3	2	1	0
TRIGGER_TIME							

EVENT_LENGTH:

Total number of words in events

TRIGGER_TIME:

Trigger time fine resolution counter

Data Type: Data Not Valid

Type: 0x14

Size: 1 word

Description: Module has no data available for readout. This can if the module is being read out too quickly after receiving (event building is in process and no data words have been put into the buffer yet) a trigger or if the module doesn't have any events to report.

31	30	29	28	27	26	25	24
1	1	1	1	0	UNDEFINED		
23	22	21	20	19	18	17	16
UNDEFINED							
15	14	13	12	11	10	9	8
UNDEFINED							
7	6	5	4	3	2	1	0
UNDEFINED							

Data Type: Filler Word

Type: 0x15

Size: 1 word

Description: Non-data word appended to the block of events. This is used to force the total number of 32-bit words read out of a module to be a multiple of 2 or 4 when

31	30	29	28	27	26	25	24
1	1	1	1	1	UNDEFINED		
23	22	21	20	19	18	17	16
UNDEFINED							
15	14	13	12	11	10	9	8
UNDEFINED							
7	6	5	4	3	2	1	0
UNDEFINED							