FADC250 Data Format

(Ed Jastrzembski – updated **12/4/13**)

We have identified the multiple types of data produced by the JLab 250 MHz Flash ADC module and defined a 32-bit word data format for readout over VME. This format is consistent with the 12 GeV standard defined for JLab custom data acquisition modules.

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 raw ADC samples. Any number of Data Type Continuation words may follow a Data Type Defining word. The scaler data type is an exception. It specifies the number of 32-bit data words that follow.

Data Type List

0 – block header
1 – block trailer
2 – event header
3 – trigger time
4 – window raw data
5 – (reserved)
6 – pulse raw data
7 – pulse integral
8 – pulse time
9 – (reserved)
10 – pulse pedestal
11 – (reserved)
12 – scaler data
13 – (reserved)
14 – data not valid (empty module)
15 – filler (non-data) word

Pulse Data

To reduce the amount of data generated at high trigger rates, the module has the capability of identifying pulses within the trigger window and reporting computed quantities (pulse integral, pulse time) instead of raw data samples or simple sums of raw samples. The algorithm for identifying a pulse may be complex and application dependent. Individual pulses may have characteristics that compromise the accuracy of the computed quantities. For example, a pulse that extends beyond the measurement window may significantly affect the computed integral value, while the computed time value (which relies on the leading edge) may be accurate. On the other hand, a pulse whose leading edge starts outside the measurement window may have an inaccurate time, but an acceptable integral value. To identify these or other conditions, we include a two-bit *quality factor* when reporting computed quantities for pulses. The algorithm designer may use the four classes to identify specific conditions that may be associated with the measurement, or give an overall rating of confidence in the measurement based of all characteristics of the pulse. Of course, the algorithm can ultimately choose to not report a particular measurement for a pulse.

The data format allows for up to four identified pulses in the trigger window for each channel. The pulse number links together pulse data types 6, 7, 8, and 10.

Data Types

Block Header (0) – indicates the beginning of a block of events. (High-speed readout of a board or set of boards is done in blocks of events.)

(31) = 1
(30 – 27) = 0
(26 – 22) = slot number (set by VME64x backplane)
(21 – 18) = module ID (‘1’ for FADC250)
(17 – 8) = event block number (used to align blocks when building events)
(7 – 0) = number of events in block

Block Trailer (1) – indicates the end of a block of events. The data words in a block are bracketed by the block header and trailer.

(31) = 1
(30 – 27) = 1
(26 – 22) = slot number (set by VME64x backplane)
(21 – 0) = total number of words in block of events

Event Header (2) – indicates the start an event. The included trigger number is useful to ensure proper alignment of event fragments when building events. The 22-bit trigger number is not a limitation, as it will be used to distinguish events within event blocks, or among events that are concurrently being built or transported.

(31) = 1
(30 – 27) = 2
(26 – 22) = slot number (set by VME64x backplane)
(21 – 0) = event number (trigger number)

Trigger Time (3) – time of trigger occurrence relative to the most recent global reset. Time in the ADC data processing chip is measured by a 48-bit counter that is clocked by the 250 MHz system clock. The global reset signal is distributed to every ADC processing chip. The assertion of the global reset clears the counters and inhibits counting. The de-assertion of global reset enables counting and thus sets t = 0 for the component. The trigger time is necessary to ensure system synchronization and is useful in aligning event fragments when building events. With careful clock, trigger, and global reset distribution it may be possible to achieve identical trigger times from all components of the system. However, even if t = 0 is not the same for all components, *changes* in trigger times can be monitored to ensure system synchronization is maintained. The six bytes of the trigger time

$$\text{Time} = T_A\ T_B\ T_C\ T_D\ T_E\ T_F$$

are reported in two words (Type Defining + Type Continuation):

<u>Word 1</u>:

(31) = 1
(30 – 27) = 3
(26 – 24) = reserved (read as 0)
(23 – 16) = T_D
(15 – 8) = T_E
(7 – 0) = T_F

<u>Word 2</u>:

(31) = 0
(30 – 24) = reserved (read as 0)
(23 – 16) = T_A
(15 – 8) = T_B
(7 – 0) = T_C

Window Raw Data (4) – raw ADC data samples for the trigger window. The first word identifies the channel number and window width. Multiple continuation words contain two samples each. The earlier sample is stored in the most significant half of the continuation word. Strict time ordering of the samples is maintained in the order of the continuation words. A *sample not valid* flag may be set for any sample; e.g. the last reported sample is not valid when the window consists of an odd number of samples.

<u>Word 1</u>:

(31) = 1
(30 – 27) = 4
(26 – 23) = channel number (0 – 15)
(22 – 12) = reserved (read as 0)
(11 – 0) = window width (in number of samples)

<u>Words 2 - N</u>:

(31) = 0
(30) = reserved (read as 0)
(29) = sample x not valid
(28 – 16) = ADC sample x (includes overflow bit)
(15 – 14) = reserved (read as 0)
(13) = sample x + 1 not valid

(12 – 0) = ADC sample x + 1 (includes overflow bit)

Pulse Raw Data (6) – raw ADC data samples for an identified pulse. Raw data from an interval of the trigger window that includes the pulse is provided. The first word indicates the channel number, pulse number, and first sample number. The first sample number is relative to the beginning of the trigger window (sample 0). Up to 4 pulses may be identified for each channel. Multiple continuation words contain two raw samples each. The earlier sample is stored in the most significant half of the continuation word. Strict time ordering of the samples is maintained in the order of the continuation words. A *sample not valid* flag may be set for any sample; for example, the last reported sample is tagged as not valid when the pulse interval consists of an odd number of samples.

<u>Word 1</u>:

(31) = 1
(30 – 27) = 6
(26 – 23) = channel number (0 – 15)
(22 – 21) = pulse number (0 – 3)
(20 – 10) = reserved (read as 0)
(9 – 0) = first sample number for pulse

<u>Words 2 - N</u>:

(31) = 0
(30) = reserved (read as 0)
(29) = sample x not valid
(28 – 16) = ADC sample x (includes overflow bit)
(15 - 14) = reserved (read as 0)
(13) = sample x + 1 not valid
(12 – 0) = ADC sample x + 1 (includes overflow bit)

Pulse Integral (7) – integral of an identified pulse within the trigger window. The pulse integral may be a simple sum of raw data samples over the pulse duration, or the result of a complex fit to pulse shape. Pedestal subtraction may be included.

(31) = 1
(30 – 27) = 7
(26 – 23) = channel number (0 – 15)
(22 – 21) = pulse number (0 – 3)
(20 – 19) = measurement quality factor (0 – 3)
(18 – 0) = pulse integral

Pulse Time (8) – time associated with an identified pulse within the trigger window.

(31) = 1
(30 – 27) = 8
(26 – 23) = channel number (0 – 15)
(22 – 21) = pulse number (0 – 3)
(20 – 19) = measurement quality factor (0 – 3)
(18 - 16) = reserved (read as 0)

(15 – 0) = pulse time

Pulse Pedestal (10) – pedestal and peak value associated with an identified pulse within the trigger window.

(31) = 1
(30 – 27) = 10
(26 – 23) = channel number (0 – 15)
(22 – 21) = pulse number (0 – 3)
(20 – 12) = pedestal (average of first 4 samples of window)
(11 – 0) = pulse peak value

Scaler Header (12) – indicates the beginning of a block of scaler data words. The number of scaler data words that will immediately follow it is provided in the header. The scaler data words are 32 bits wide and so have no bits available to identify them. Currently there are 18 scaler words reported: 16 from individual channels, a timer, and a trigger count. The scalers and time represent values recorded at the indicated trigger count. Scaler data must be enabled into the data stream by the user.

(31) = 1
(30 – 27) = 12
(26 – 6) = reserved (read as 0)
(5 – 0) = number of scaler data words to follow (18 = current)

Data Not Valid (14) – module has no valid data available for read out.

(31) = 1
(30 – 27) = 14
(26 – 22) = slot number (set by VME64x backplane)
(21 – 0) = undefined

Filler Word (15) – non-data word appended to the block of events. Forces the total number of 32-bit words read out of a module to be a multiple of 2 or 4 when 64-bit VME transfers are used. **This word should be ignored**.

(31) = 1
(30 – 27) = 15
(26 – 22) = slot number (set by VME64x backplane)
(21 – 0) = undefined