Implementation of QT Algorithm for EPD Outer Tiles
qt32b_l0_v7_a.mcs

Eleanor Judd

April 19, 2019

Description:

The EPD outer tile signals are digitized using QT32B boards. These boards have 32 ADC channels, each of which has an associated low-resolution TDC. There are no high-resolution TAC circuits. This algorithm forms 2 independent good hit counts, each of which is based on a configurable subset of channels. The good hit definition is a variation on the standard good-hit definition. It still includes an ADC requirement, but the user can decide whether or not to include a TDC requirement on a channel-by-channel basis. There is no slew correction logic in this algorithm because the resolution of the TDC values is too low to justify it.

An outline of the steps followed by this algorithm is given below:

1. Apply channel masks
2. Compare ADC and TDC values to thresholds
3. Apply configurable "Good Hit" Requirement
4. Apply both multiplicity masks
5. Count both sets of masked good hits.

The standard channel mask registers can be used for each channel to mask out that channel completely from the trigger but retain the data in the data-stream.

A good hit is then defined as one where the ADC value is greater than some **ADC_th**. The user can also specify for each channel whether or not the good hit definition includes the requirement that the corresponding TDC value is greater than **TDC_MIN** and less than **TDC_MAX**.

Two copies of the good hit list are made, and separate multiplicity masks are then applied to each list. The masked good hit bits in each list are counted to give a multiplicity value. Two independent multiplicities are needed because each EPD outer-tile QT board receives data from parts of 2 EPD rings. Counting the good hits in each ring separately makes it possible to implement ring-based trigger algorithms in the DSM boards, as well as ring-group-based algorithms. The results are delayed appropriately so they can be combined with the information that has been passed down from the preceding QT8 daughter card, and the final results are passed on to the next daughter card in the chain or the L0 FPGA on the mother board. Those final multiplicity values are 6-bit integers in the range 0 to 32. If the Multiplicity Masks have been set up to select just the channels in specific EPD rings, then the maximum value that actually occurs will be 24.

Inputs:

QT8A: 8 PMT ADC, 8 discriminator TDC
QT8B: 8 PMT ADC, 8 discriminator TDC
QT8C: 8 PMT ADC, 8 discriminator TDC
QT8D: 8 PMT ADC, 8 discriminator TDC

Registers (1 Set Per Daughter Card)**:**

Reg. 11: Channel Mask (8 bits)
Alg. Reg. 0 (Reg 13): EPD_QT_ADC_Threshold (12 bits)
Alg. Reg. 1 (Reg 14): EPD_QT_TDC_MIN (5 bits)
Alg. Reg. 2 (Reg 15): EPD_QT_TDC_MAX (5 bits)
Alg. Reg. 3 (Reg 16): EPD_QT_Dx_TDC_Mask (8 bits)
Alg. Reg. 4 (Reg 17): EPD_QT_Dx_Mult_Mask_1 (8 bits)
Alg. Reg. 5 (Reg 18): EPD_QT_Dx_Mult_Mask_2 (8 bits)
NOTE 1: All of the Mask registers are bitmasks.
For the Channel Mask a "0" turns a channel ON in the logic and a "1" turns it off.
For the TDC and Mult Masks the opposite convention is used. A "1" turns a bit ON in the logic, and a "0" turns it OFF.
NOTE 2: Registers with Dx (x = 1:4) in their name may have different values on each QT8 daughter card

LUT:

Timing adjustments/pedestal subtraction for each PMT

Algorithm Latch: 1 or 2

L0 Output to DSM:

(0-5)	Multiplicity-1
(6-11)	Multiplicity-2
(12-31)	'0'

Actions:

Tick	QT8A	QT8B	QT8C	QT8D
1	Latch inputs	Same as QT8A	Same as QT8A	Same as QT8A
2	Apply Channel_Mask	Same as QT8A	Same as QT8A	Same as QT8A
3	ADC > R0 -> ADC_GOOD TDC > R1 -> TDC_MIN_GOOD TDC < R2 -> TDC_MAX_GOOD	Same as QT8A	Same as QT8A	Same as QT8A
4	Combine GOOD info -> GOOD hits	Same as QT8A	Same as QT8A	Same as QT8A
5	Apply Mult_Mask1 -> GOOD-hits1 Apply Mult_Mask2 -> GOOD-hits2	Same as QT8A	Same as QT8A	Same as QT8A
6	Sum bits of GOOD-hits1 Sum bits of GOOD-hits2	Same as QT8A	Same as QT8A	Same as QT8A
7	Latch Sum-hits1 and Sum-hits2	Delay Sums	Delay Sums	Delay Sums
8	Delay Sum-hits1 and Sum-hits2	Delay	Delay	Delay
9	Latch out Sum-hits1 and Sum-hits2	Delay	Delay	Delay
10		Latch in Sums info from upstream QT8 Latch local Sums	Delay	Delay
11		Sum-hits1: Local + Upstream Sum-hits2: Local + upstream	Delay	Delay
12		Latch out Sum-hits1 and Sum-hits2	Delay	Delay
13			Latch in Sums info from upstream QT8 Latch local Sums	Delay
14			Sum-hits1: Local + Upstream Sum-hits2: Local + upstream	Delay
15			Latch out Sum-hits1 and Sum-hits2	Delay
16				Latch in Sums info from upstream QT8 Latch local Sums
17				Sum-hits1: Local + Upstream Sum-hits2: Local + upstream
18				Latch out Sum-hits1 and Sum-hits2