

Algorithms for TOF and MTD DSM Tree  
RHIC 2016 Heavy Ion Run

Eleanor Judd

April 8, 2016

Change Log:

Date	Description
January 21, 2016	First version for the 2016 heavy ion run. The MT101 algorithm has been modified to add in bit meaning “at least one good MTD hit”. All the DAQ10k-related logic (test mode and ID selection) has been removed. There are now 2 MTD DSM boards; MT101 and MT102, both of which run the same algorithm. The TF201 algorithm has therefore been modified to combine the muon counts from the two boards. It also passes through all the cosmic bits and the new hit bits. To make room for these extra bits the PP2PP logic was removed since PP2PP is not taking data this run.
April 8, 2016	Second version. The TF201 algorithm has been modified. Two of the MTD Cosmic bits have been combined to make room for a 5 <sup>th</sup> TOF multiplicity threshold.

## Layer 0 DSM Boards: MIX\_TF001:006

The TOF layer-0 DSM boards each receive 20 5-bit multiplicity values from the TOF trays. The connections are made such that each layer-0 DSM receives TOF data from one 2-hour pie-slice of the detector. Each 5-bit number is actually a count of how many TOF trays, in a group of 24, were hit. Values greater than 24 are therefore unphysical and are ignored. The values are summed to calculate the total multiplicity. The summing is performed in stages. The result of one of the intermediate stages is 4 numbers, each of which covers  $\frac{1}{4}$  of the 2-hour pie-slice: 2 segments on the East side of the barrel and 2 on the West side. Each of those 4 numbers is compared to two thresholds. The 4 bits associated with the 1<sup>st</sup> threshold are passed to Layer-1 for use by the Ultra-Peripheral Collisions (UPC) program. The 4 bits associated with the 2<sup>nd</sup> threshold are passed to the TF202 DSM for use in the DAQ10k readout system. In parallel two signals (the RHIC clock and the run/stop signal) are passed through to the test header. This is being done to help debug a synchronization problem in the TOF layer-0 DSM boards.

NOTE: In order to avoid doing too many sums in parallel, this algorithm takes an extra tick of the RHIC clock, which corresponds to 4 extra ticks of the 4xRHIC clock that is used by the FPGA. This allows many sums to be performed in series, which is easier to implement.

RBT File: mix\_tf001\_2014\_c.rbt

Users: TF001:TF006

Inputs: Ch 0:6 = TOF trays  
Ch 7 = Unused

On each DSM channel:

(0:14) 3 5-bit TOF multiplicity values

(15) Unused

NOTE: Ch 6 receives just 2 input multiplicity values so it uses only bits 0:9

LUT: 1-to-1. Noisy, dead and non-instrumented channels are also zeroed out here

Registers:

R0: TOF\_upc\_th (8)

R1: TOF\_DAQ10k\_th (8)

Action

1<sup>st</sup> Latch inputs

2<sup>nd</sup> Zero out any channel with a value greater than 24

3<sup>rd</sup> Sum TOF channels 0:2, 3:4, 5:7, 8:9, 10:12, 13:14, 15:17 and 18:19

4<sup>th</sup> Combine these sums in pairs to make the sums of channels 0:4, 5:9, 10:14 and 15:19.

NOTE: Each of these sums covers 1 unit in  $\eta$  and  $\pi/6$  (i.e. 1/12 of the barrel = 1 hour) in  $\phi$ . The cabling of the TOF data into the DSM boards is such that the ordering is the same for all 6 TOF layer-0 DSM boards. WHEN VIEWED FROM THE EAST:

- Sum 0:4 = West, low hour
- Sum 5:9 = West, high hour
- Sum 10:14 = East, low hour

- Sum 15:19 = East, high hour

- 5<sup>th</sup>      Combine these sums in pairs to make the sums of channels 0:9 and 10:19.  
 Compare each of the 4 sums to the UPC threshold in register 0. The logic looks for  
 sum(channels 0:4) > reg0, etc...  
 Compare each of the 4 sums to the DAQ10k threshold in register 1. The logic looks for  
 sum(channel 0:4) > reg1, etc...
- 6<sup>th</sup>      Combine these two sums to make the final sums of channels 0:19.  
 Also delay the UPC and DAQ10k threshold bits to the 8<sup>th</sup> step.
- 7<sup>th</sup>      Delay the final sum
- 8<sup>th</sup>      Latch Outputs

Output to TF101:

- (0:9) TOF multiplicity
- (10:13) UPC threshold bits
- (14:15) Unused

Output to TF202:

- (16:19) DAQ10k threshold bits
- (20:31) Unused

Output to Test Header:

- (0) Run/Stop signal from Operation Control FPGA
- (2:14) Unused
- (15) RHIC clock

## 1. Layer 1 DSM Board: MIX\_TF101

The TOF layer-1 DSM board receives a 10-bit multiplicity value and 4 UPC threshold bits from each of the six TOF layer-0 DSM boards. Each input multiplicity is compared to a threshold. In parallel with this, the values are also summed to calculate the total multiplicity. For the UPC program, the 24 input bits are masked so the UPC group can constrain which topologies they will trigger on. The masked bits are then searched to look for interesting combinations. The cabling is such that:

- TF001 covers sectors that start at 9 o'clock and 10 o'clock when viewed from the East
- TF002 covers sectors that start at 11 o'clock and 12 o'clock
- TF003 covers sectors that start at 1 o'clock and 2 o'clock
- TF004 covers sectors that start at 3 o'clock and 4 o'clock
- TF005 covers sectors that start at 5 o'clock and 6 o'clock
- TF006 covers sectors that start at 7 o'clock and 8 o'clock

NOTE: In order to avoid doing too many sums in parallel, this algorithm takes an extra tick of the RHIC clock, which corresponds to 4 extra ticks of the 4xRHIC clock that is used by the FPGA. This allows many sums to be performed in series, which is easier to implement.

RBT File: mix\_tf101\_2014\_b.rbt

Users: TF101

Inputs: Ch 0:5 = TF001:TF006  
Ch 6:7 = Unused

On each DSM channel:  
(0:9) TOF multiplicity  
(10:13) UPC threshold bits  
(14:15) Unused

LUT: 1-to-1

Registers:

R0: TOF-sector-th (10)  
R1: TOF\_upc\_East\_mask (12)  
R2: TOF\_upc\_West\_mask (12)

Action

- 1<sup>st</sup> Latch inputs
- 2<sup>nd</sup> Sum channels 0:1, 2:3 and 4:5  
Compare each of the 6 input multiplicity values to the threshold specified in register 0.  
Mask out the UPC bits using the values masks specified in register 1 (East) and register 2 (West). A “1” in the mask enables a UPC bit, “0” will disable that bit. BOTH masks are coded such that:
  - bit(0) enables/disables the sector that starts at 12 o’clock AS VIEWED FROM THE EAST
  - bit(1) enables/disables the sector that starts at 1 o’clock AS VIEWED FROM THE EAST
  - ...
  - bit(11) enables/disables the sector that starts at 11 o’clock AS VIEWED FROM THE EAST
- 3<sup>rd</sup> Combine the first two sums to make the sums of channels 0:3.  
Delay the sum of channels 4 and 5 to the 4<sup>th</sup> step.  
Delay the 6 threshold bits to the 8<sup>th</sup> step.  
Combine (OR) the East and West UPC bits for each  $\phi$  value and then look for all possible combinations separated by 4 hours, i.e.
  - 1 o’clock = East 1 o’clock OR West 1 o’clock
  - 2 o’clock = East 2 o’clock OR West 2 o’clock
  - ...
  - 12 o’clock = East 12 o’clock OR West 12 o’clock
$$\text{TOF\_UPC} = \begin{aligned} &12 \text{ o’clock AND } 4 \text{ o’clock OR} \\ &1 \text{ o’clock AND } 5 \text{ o’clock OR} \\ &\dots \\ &11 \text{ o’clock AND } 3 \text{ o’clock} \end{aligned}$$
- 4<sup>th</sup> Combine the two remaining sums to make the final total multiplicity sum of channels 0:5.  
Delay the TOF\_UPC bit to the 8<sup>th</sup> step.
- 5<sup>th</sup> Delay the final sum to the 8<sup>th</sup> step.

6/7<sup>th</sup> No logic

8<sup>th</sup> Latch Outputs

Output to TF201:

(0:12)	TOF total multiplicity
(13:15)	Unused
(16:21)	6 sector threshold bits
(22)	TOF_UPC bit
(23:31)	Unused

## 2. Layer 0 QT Board: MXQ\_MT001:008

All eight MTD QT boards in the MXQ crate are using the same algorithm. That algorithm forms TAC pair sums and then finds the two largest sums to pass on to the DSM tree. Please see the documentation provided by Chris Perkins for a detailed description of this algorithm at [http://www.star.bnl.gov/public/trg/TSL/Software/qt\\_v6\\_c\\_doc.pdf](http://www.star.bnl.gov/public/trg/TSL/Software/qt_v6_c_doc.pdf)

## 3. Layer 0 QT Board: MXQ\_VP003:004

The two VPD QT boards in the MXQ crate are running essentially the same algorithm as the two BBC small-tile QT boards which are in the BBQ crate. The only difference is that this version discards the ADC sum and instead produces 2 copies of the TAC Max, one each for MT101 and MT102. Please see the documentation provided by Chris Perkins for a detailed description of the new algorithm at [http://www.star.bnl.gov/public/trg/TSL/Software/qt\\_v7\\_3\\_doc.pdf](http://www.star.bnl.gov/public/trg/TSL/Software/qt_v7_3_doc.pdf)

## 4. Layer 1 DSM Board: MIX\_MT101

The MT101 DSM receives its data through a TDSMI, so there are 10 12-bit input channels. The algorithm receives the 2 best TAC sums from each of 4 MTD QT boards. It also receives the fastest (largest) TAC values from the QT boards covering the East and West sides of the VPD (MXQ\_VP003/4). Note that the cables carrying the VPD ADC sums are not connected to MT101 so that data is not received. There are two separate logic streams flowing through the algorithm:

**Muons:** The VPD TAC sum is calculated. Then the algorithm finds the difference between each MTD TAC sum and that VPD TAC sum. All 8 TAC differences are checked to determine if they are inside a window. Those that are inside the window are counted, and the count is sent on to TF201.

**Cosmic Rays:** A hit bit is set if at least one of the MTD TAC sums is non-zero, and a cosmic bit is set if at least two are non-zero. Those bits are sent to TF201 for use in cosmic ray triggers. In addition the MTD TAC sums are also sorted to find the two largest non-zero values. The difference between those 2 largest TAC sums is calculated, and that difference is checked to determine if it is inside a window. That bit is also sent to TF201 for use as a more restrictive cosmic ray trigger.

NOTE: This algorithm takes an extra tick of the RHIC clock to implement all the logic.

RBT File: mix\_mt101\_2016\_a.rbt

Users: MT101, MT102

Inputs: Ch0:1 = MT001 = MTD  
Ch2:3 = MT002 = MTD  
Ch4:5 = MT003 = MTD  
Ch6:7 = MT004 = MTD  
Ch8 = VP003 = VPD East  
Ch9 = VP004 = VPD West

From the MTD QT boards:

Bits 0:9 = Best MTDE+MTDW TAC sum (value=0 means not enough good hits)  
Bits 10:11 = Unused  
Bits 12:21 = 2<sup>nd</sup> best MTDE+MTDW TAC sum (value=0 means not enough good hits)  
Bits 22:23 = Unused

From VP003/4 QT boards:

Bits 0:11 = Max TAC (value=0 implies no good hits)

LUT: 1:1

Registers:

R0: MTD-VPD-TACdiff-Min (11)  
R1: MTD-VPD-TACdiff-Max (11)  
R2: MTD-InputCh-Bitmask (8)  
Bit x = 0 turns off ChX, Bit x = 1 turn on ChX  
R3: MTD-Cosmic-Tdiff-Min (10)  
R4: MTD-Cosmic-Tdiff-Max (10)

Action:

- 1<sup>st</sup> Latch input
- 2<sup>nd</sup> Define Good\_VPD\_E = VPDE-TAC > 0  
Define Good\_VPD\_W = VPDW-TAC > 0  
Calculate 13-bit VPD-sum = VPDE TAC + VPDW TAC  
Truncate VPD-sum to 10-bits by dropping 3 LSB  
For each MTD ChX:  
Zero out MTD-TAC-X if R2(x) = 0
- 3<sup>rd</sup> Define Good\_VPD = Good\_VPD\_E and Good\_VPD\_W.  
For each MTD ChX:  
Define: Good\_ChX = MTD-TAC-X > 0  
Calculate: MTD-VPD-TACdiffX = 1024 + MTD-TAC-X - VPD-sum  
Split the MTD-TAC-X values into 2 groups (A = ch0:ch3 and B = ch4:ch7)  
For each group:  
Make the comparisons necessary to find the 2 largest values in each group, e.g. in group A look for ch0>ch1, ch0>ch2, ch0>ch3, ch1>ch2, ch1>ch3 and ch2>ch3
- 4<sup>th</sup> Count the Good\_ChX bits = cosmic\_count  
Within each group use the results of the comparisons to find the 2 largest values, e.g. in group A:  
Ch0 is largest if ch0>ch1 and ch0>ch2 and ch0>ch3  
...  
Ch3 is largest if (not ch0>ch3) and (not ch1>ch3) and (not ch2>ch3)

NOTE 1: If all channels have the same MTD-TAC-X value then this logic will select Ch3 as the largest.

NOTE 2: If the largest MTD-TAC-X value is zero (i.e. no good hits) then no channel is selected and the output of this stage is also zero.

For each MTD ChX:

Compare the TACdiffX value to the min and max values specified in R0 and R1

$TACdiffX-min = MTD-VPD-TACdiffX > R0$

$TACdiffX-max = MTD-VPD-TACdiffX < R1$

Zero out TACdiffX-min/max if EITHER Good\_VPD or Good\_ChX is FALSE.

Combine the min and max values to determine if the TAC difference is inside the window:

$TACdiff-in-window-X = TACdiffX-min \text{ and } TACdiffX-max$

- 5<sup>th</sup> Set the Hit bit to 1 if cosmic\_count > 0  
Set the Cosmic-Ray bit to 1 if cosmic\_count > 1  
Count how many TACdiff-in-window bits are true = num-muons  
Compare the largest MTD-TAC-X values from groups A and B to find the overall largest value.  
NOTE 1: If both channels have the same MTD-TAC-X value then this logic will select the one from group B as the largest.  
NOTE 2: If the largest MTD-TAC-X value is zero (i.e. no good hits) then no channel is selected and the output of this stage is also zero.  
Compare the remaining values to find the 2<sup>nd</sup> largest value. Same NOTES apply.
- 6<sup>th</sup> Calculate MTD-Largest-Tdiff = largest MTD-TAC-X – 2<sup>nd</sup> largest MTD-TAC-X.  
Zero out the result if either TAC-X value is zero.  
NOTE: The result of this subtraction is guaranteed to be positive so there is no need for the offset of 1024 that is used in the MTD-VPD subtraction logic.
- 7<sup>th</sup> Compare the Largest-Tdiff value to the min and max values specified in R6 and R7  
 $Tdiff-min = MTD-Largest-Tdiff > R3$   
 $Tdiff-max = MTD-Largest-Tdiff < R4$   
Combine the min and max values to determine if the TAC difference is inside the window. This is the timed-cosmic bit  
 $Tcosmic = Tdiff-min \text{ and } Tdiff-max$
- 8<sup>th</sup> Latch output

Output to TF201:

- (0:3) num\_muons
- (4:11) “MTD-VPD TAC difference in window” bits
- (12) Hit bit
- (13) Good\_VPD
- (14) Cosmic-Ray
- (15) Timed-Cosmic-Ray

## 5. Layer 0 QT Board: MXQ\_PP001

Please see the documentation provided by Chris Perkins for a description of the algorithm.

## 6. Layer 1 PP2PP DSM Board: MIX\_PP101

The PP101 DSM board receives 16 “good hit” bits from the PP001 QT board. Those bits are combined to make “good detector” bits, which are themselves combined to make the trigger and scaler bits. All the bits are passed on to TF201, which separates the trigger bits from the scaler bits and sends them to the appropriate place. A mask register is available to turn each “good hit” bit on or off in case of problems.

RBT File: mix\_pp101\_2015\_a.rbt

Users: PP101

Inputs Ch 0 = PP001  
Ch 1:7 = Unused

From: The PP2PP QT board

- (0) E2U1 = Roman Pot East-2 (Back) Up 1
- (1) E2U2 = Roman Pot East-2 (Back) Up 2
- (2) E2D1 = Roman Pot East-2 (Back) Down 1
- (3) E2D2 = Roman Pot East-2 (Back) Down 2
- (4) W2U1 = Roman Pot West-2 (Back) Up 1
- (5) W2U2 = Roman Pot West-2 (Back) Up 2
- (6) W2D1 = Roman Pot West-2 (Back) Down 1
- (7) W2D2 = Roman Pot West-2 (Back) Down 2
- (8) E1D1 = Roman Pot East-1 (Front) Down 1
- (9) E1D2 = Roman Pot East-1 (Front) Down 2
- (10) E1U1 = Roman Pot East-1 (Front) Up 1
- (11) E1U2 = Roman Pot East-1 (Front) Up 2
- (12) W1D1 = Roman Pot West-1 (Front) Down 1
- (13) W1D2 = Roman Pot West-1 (Front) Down 2
- (14) W1U1 = Roman Pot West-1 (Front) Up 1
- (15) W1U2 = Roman Pot West-1 (Front) Up 2

LUT: 1-to-1

Registers:

R0: RPmask (16)

NOTE: The bits are arranged in the SAME ORDER as the input data coming from PP001

Bit 0 = E2U1 on(1) or off (0)

Bit 1 = E2U2 on(1) or off (0)

...

Bit 15 = W1U2 on(1) or off (0)

Action

1<sup>st</sup> Latch inputs

2<sup>nd</sup> Combine the 2 hits from the same counter, and then combine the 2 counters on each side, to make the detector bits. Mask each hit with Reg0 before it is used.

E2U = (E2U1 and Reg0(0)) or (E2U2 and Reg0(1))

E2D = (E2D1 and Reg0(2)) or (E2D2 and Reg0(3))

W2U = (W2U1 and Reg0(4)) or (W2U2 and Reg0(5))

W2D = (W2D1 and Reg0(6)) or (W2D2 and Reg0(7))  
Same for E1D, E1U, W1D, W1U

EU = E1U or E2U  
ED = E1D or E2D  
WU = W1U or W2U  
WD = W1D or W2D

3<sup>rd</sup> Delay a copy of the detector bits (EU, ED, WU and WD) to the 4<sup>th</sup> step.  
Combine the detector bits to make the trigger bits

EA (Elastic Trigger A) = EU and WD  
EB (Elastic Trigger B) = ED and WU  
IU (Inelastic Trigger Up) = EU and WU  
ID (Inelastic Trigger Down) = ED and WD

ET (Elastic Trigger) = EA or EB  
IT (Inelastic Trigger) = IU or ID  
EOR (East OR) = EU or ED  
WOR (West OR) = WU or WD

4<sup>th</sup> Latch Outputs

Output to TF201:

(0:3) ET, IT, EOR, WOR  
(4:7) EU, ED, WU, WD  
(8:15) Unused

## 7. Layer 2 TOF DSM Board: L1-TF201

All the information from the TOF and MTD and detectors is brought into the TOF layer 2 DSM. The MTD Hit and Cosmic Ray bits and the TOF\_UPC topology bit are simply passed through to the TCU. The TOF multiplicity is compared to five thresholds. The 6 TOF sector threshold bits are combined in pairs to make cosmic ray bits. The two MTD muons counts are added together and then used to set one bit meaning “at least one muon” and a second bit meaning “at least two muons”

RBT File: 11\_tf201\_2016\_b.rbt

Users: TF201

Inputs: Ch 0 = MT101  
Ch 1 = Unused  
Ch 2:3 = TF101  
Ch 4:5 = PP101 (Not used)  
Ch 6 = MT102  
Ch 7 = Unused

From MTD Layer 1 DSMS - MT101 (Ch0) and MT102 (Ch6)

(0:3) Num\_muons  
(4:11) Unused  
(12) Hit  
(13) Unused

- (14) Cosmic-Ray
- (15) Timed-Cosmic-Ray

From TOF Layer 1 DSM - TF101 on Ch. 2  
 (0:12) TOF total multiplicity  
 (13:15) Unused

From TOF Layer 1 DSM - TF101 on Ch. 3  
 (0:5) TOF sector threshold bits  
 (6) TOF\_UPC topology bit  
 (7:15) Unused

LUT: 1-to-1

Registers:

- R0: TOFmult0 (13)
- R1: TOFmult1 (13)
- R2: TOFmult2 (13)
- R3: TOFmult3 (13)
- R4: TOFmult4 (13)

Action

- 1<sup>st</sup> Latch inputs
- 2<sup>nd</sup> Delay the MTD Hit, Timed-Cosmic-Ray bits and the TOF\_UPC topology bit to the 4<sup>th</sup> step.  
 OR together the 2 MTD Cosmic-Ray bits  
 Add together the 2 MTD muon counts to get the total num\_muons  
 Combine the TOF sector bits from opposite sides of the barrel: i.e.  
     TOFsector0\_3 = TOFsector(0) and TOFsector(3)  
     TOFsector1\_4 = TOFsector(1) and TOFsector(4)  
     TOFsector2\_5 = TOFsector(2) and TOFsector(5)  
 Compare the TOF total multiplicity to the thresholds specified in registers 0 to 4.
- 3<sup>rd</sup> Delay the MTD Cosmic bit, the TOF multiplicity bits and the TOF sector bits to the 4<sup>th</sup> step.  
 Set the MTD threshold bits:  
     If Num\_muons >= 1 then MTD\_th1 = 1  
     If Num\_muons >= 2 then MTD\_th2 = 1
- 4<sup>th</sup> Latch Outputs

Output to TCU:

Bit	Name	Description
Bit 0	MTD_th1	At least one muon in the MTD
Bit 1	MTD_th2	At least two muons in the MTD
Bit 2	MTD2-T-Cosmic	Timed cosmic bit from MT102
Bit 3	TOF_UPC	TOF UPC topology bit
Bit 4	TOFmult0	TOF total multiplicity > th0
Bit 5	TOFmult1	TOF total multiplicity > th1
Bit 6	TOFmult2	TOF total multiplicity > th2
Bit 7	TOFmult3	TOF total multiplicity > th3
Bit 8	MTD1-T-Cosmic	Timed cosmic bit from MT101
Bit 9	TOFsector0_3	TOF sectors 0 and 3 multiplicity > th
Bit 10	TOFsector1_4	TOF sectors 1 and 4 multiplicity > th
Bit 11	TOFsector2_5	TOF sectors 2 and 5 multiplicity > th
Bit 12	MTD2-Hit	Hit bit from MT102
Bit 13	MTD-Cosmic	Cosmic bit from MT101 or MT102
Bit 14	MTD1-Hit	Hit bit from MT101
Bit 15	TOFmult4	TOF total multiplicity > th4