Module Level Trigger Discussion

J.R. Klein, Penn, 8/31/2021

Module Level Trigger Interfaces

- MLT receives TCs from several (but not many) possible algorithmic streams
  - No requirement on relative timing of these except that they have to arrive in time that data can still be retrieved from upstream buffers
- MLT sends TDs to DFO to generate a request to UDAQ for data
- CCM configures MLT; operational monitoring also sent to CCM
- MLT can “veto” or hold off calibration source triggering (through CCM?)
- No explicit interface with Timing System (unless this is where veto goes)

Trigger Candidates and MLT

There are a few classes of TCs:

- TPC-generated
  - Derived from TPC trigger primitives; these are slow
- PDS-generated
  - Derived from PDS trigger primitives; these are fast (compared to TPC)
- “Hardware”-generated
  - Come through HSI from a variety of sources and times relative to detector
  - This includes beam information
- Self-generated
  - Random triggers, pulsed triggers, “Run Control” triggers

MLT Functionality

The bottom-line functionality is taking TCs and sending TDs

But this can be broken up into a few pieces

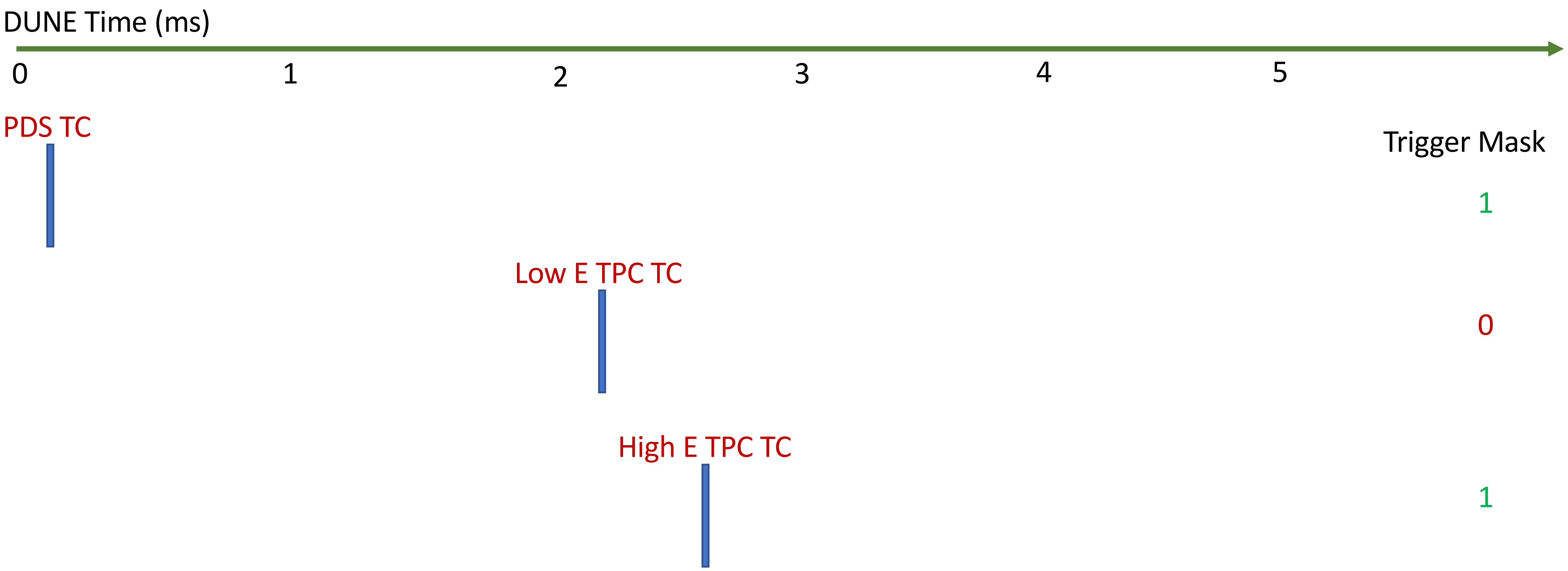
- Trigger *arbitration:*
  - Are we configured to accept this trigger candidate type?
  - Does this trigger need to be sent as a TD immediately, or can I wait?
  - Do I require a coincidence between TCs to generate a TD?
- Trigger merging
  - What time window should I use for this collection of triggers?
  - What fraction of the channels/APAs/CRPs should be included in TD?
  - ("Aggressive" ROIs/fiducialization have been moved to HLF)
- TD packaging and sending
  - Can I do this promptly or am I being held off by DFO?
- Deadtime Accounting

MLT Functionality

Trigger coincidences

- We would like to record all TCs that are in coincidence with each other
- Sometimes coincidences may be *required* for a TD
- Coincidences also affect how TCs are merged to create TDs
- Useful if this is not just for TCs that are enabled to create a TD
  - E.g., coincidence of a low-energy TC and a high-energy TC
  - Even if low E TC would not create a trigger on its own
  - This will help us understand and monitor efficiencies
- Coincidence “gates” are complex
  - Must include known offsets (PDS vs. TPC, for example)
  - Can’t be so long as to be pointless (if, e.g., PDS TC rate is high)

Simple Logic



DUNE Time (ms)

0 1 2 3 4 5

Trigger Mask

PDS TC

1

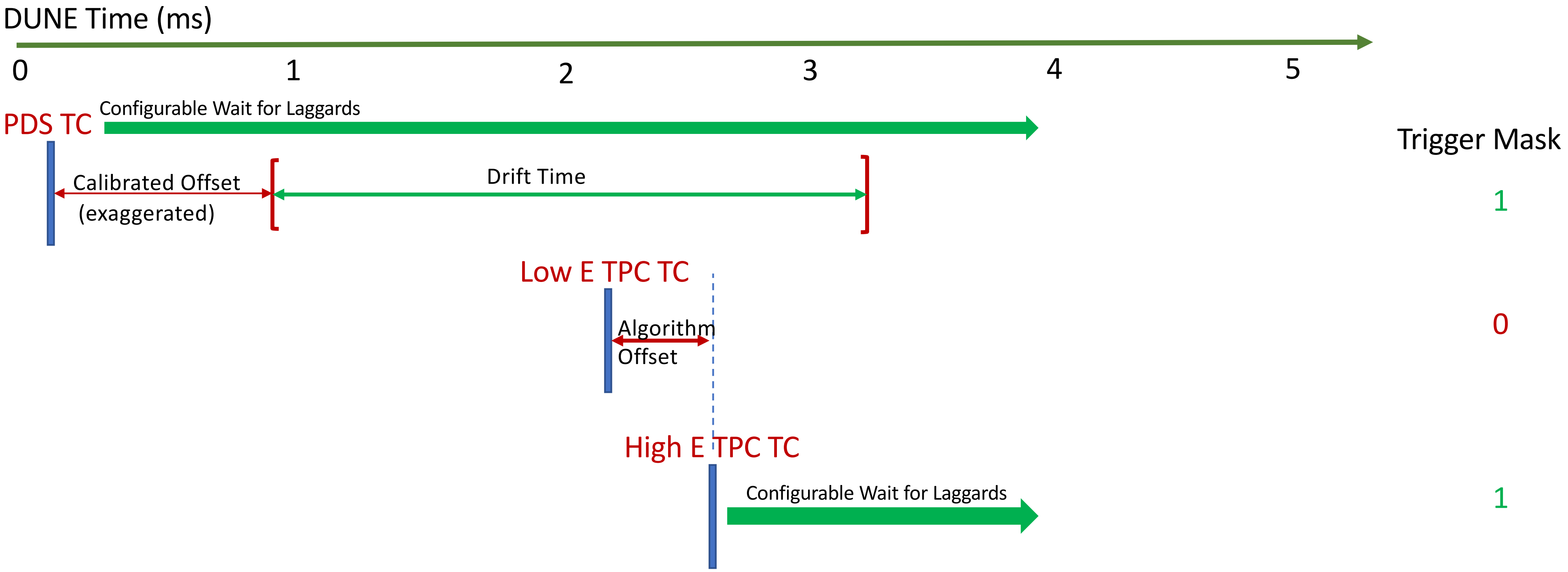
Low E TPC TC

0

High E TPC TC

1

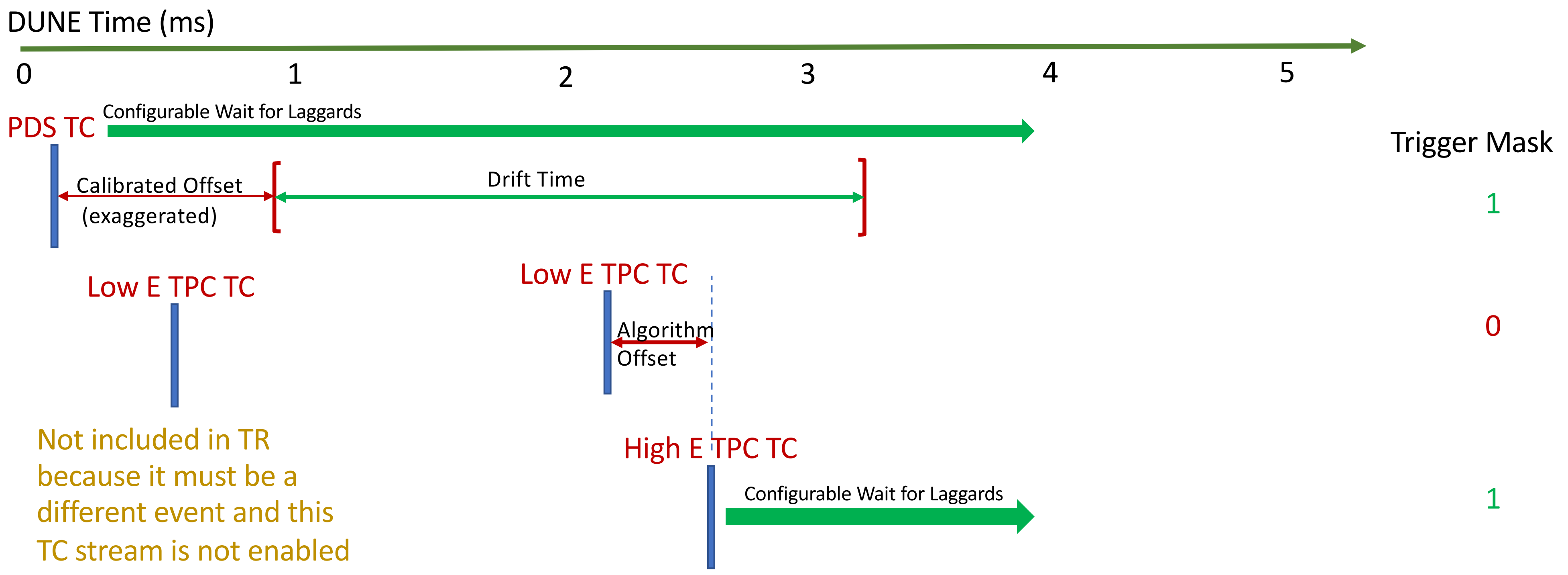
Simple Logic



Trigger Record would record that 3 TCs were in coincidence, even though one here (LE) could not have created a TD on its own
Coincidence is not <u>required</u> here for a TD; it just happens to occur

Note that Wall Clock Time at MLT may look very different

Simple Logic

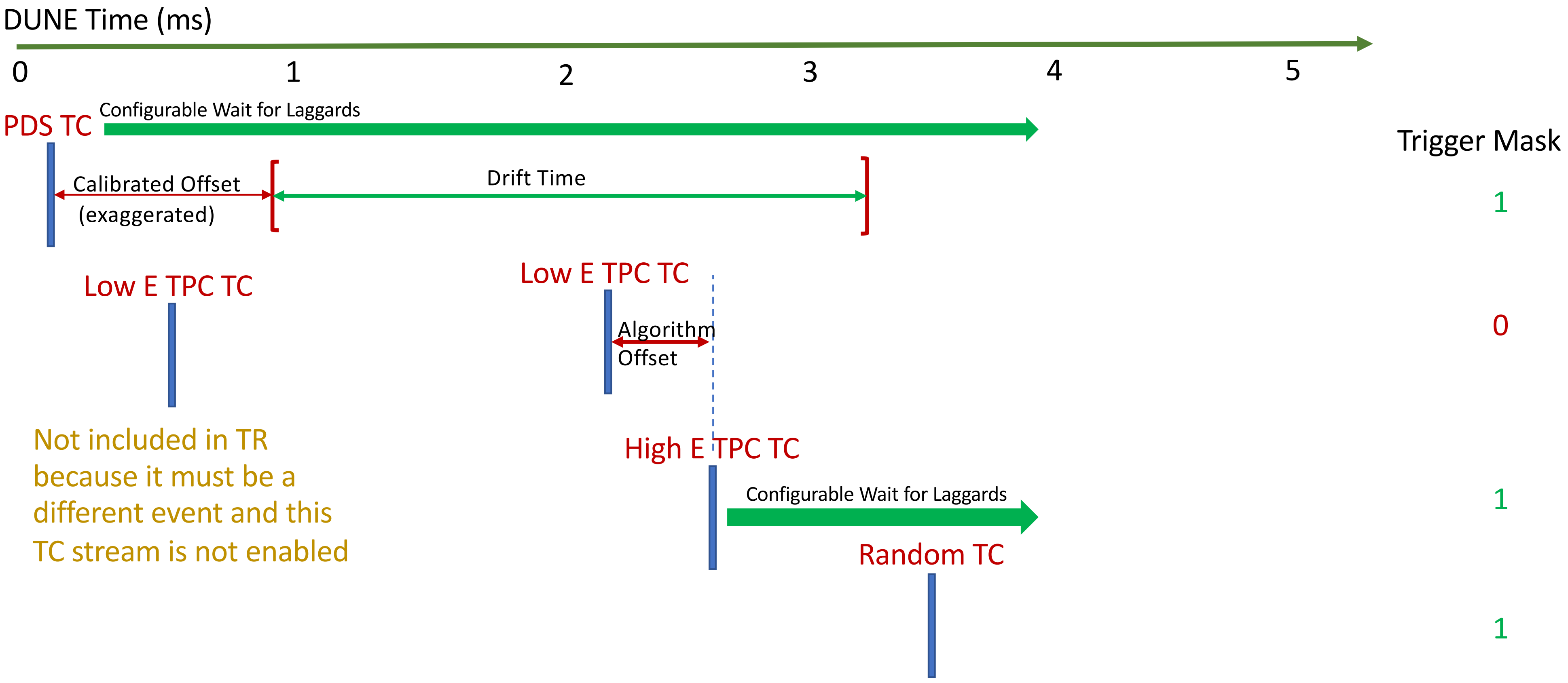


Not included in TR because it must be a different event and this TC stream is not enabled

Trigger Record would still record that 3 TCs were in coincidence, because first Low E TC must be something else

- TD merge would have to decide how much of detector to read out
- Readout windows will be different for TPC and PDS data

Simple Logic



DUNE Time (ms)

0 1 2 3 4 5

PDS TC

Configurable Wait for Laggards

Calibrated Offset (exaggerated)

Drift Time

Trigger Mask

1

Low E TPC TC

Low E TPC TC

Algorithm Offset

0

Not included in TR because it must be a different event and this TC stream is not enabled

High E TPC TC

Configurable Wait for Laggards

1

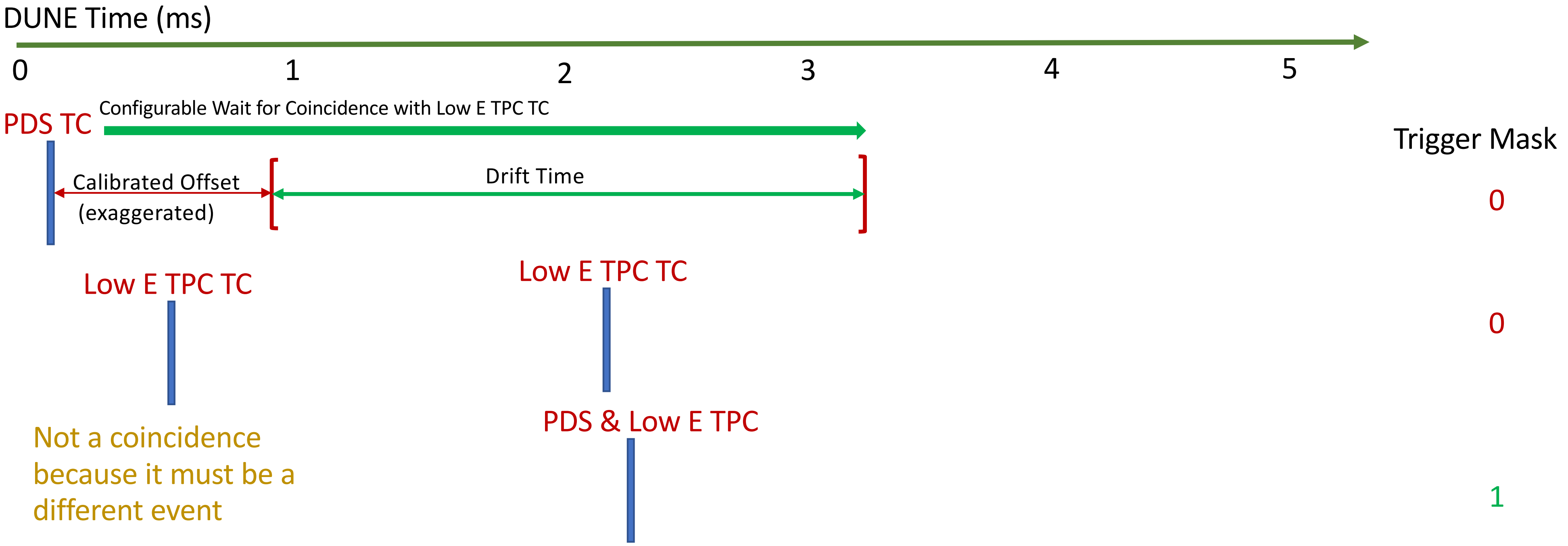
Random TC

1

Trigger Record would record that 4 TCs were in coincidence, because random’s time window would include prior event data

Slightly More Complex Logicçç

Imagine that coherent noise means we require a coincidence with PDS for Low E triggering



Not a coincidence because it must be a different event

Trigger Record would still record that a PDS & LE TPC satisfied coincidence trigger; record bits and coincidence bit

- TD merge would have to decide how much of detector to read out
- Readout windows will be different for TPC and PDS data

Summary Suggestions

- We need a configurable trigger “mask” for MLT, which includes coincidence logic
- TPC is so slow that we should defined a “coincidence” of triggers by their overlapping readout windows, not necessarily by physics
- We need a configurable window that may depend on trigger type to decide how long to wait for other associated triggers
- We record trigger “bits” even when they are masked out
  - Will be useful for trigger efficiency measurements