Plans for future Trigger Processor

(preliminary thoughts about concept)

Benjamin Moritz Veit

12. Februar 2019

Federal Ministry of Education and Research

Requirements

Making the trigger decision on geometrical/energy criteria in real-time.

- Continuous readout of a subset of detectors.
- Event-building according to hit time (L0).

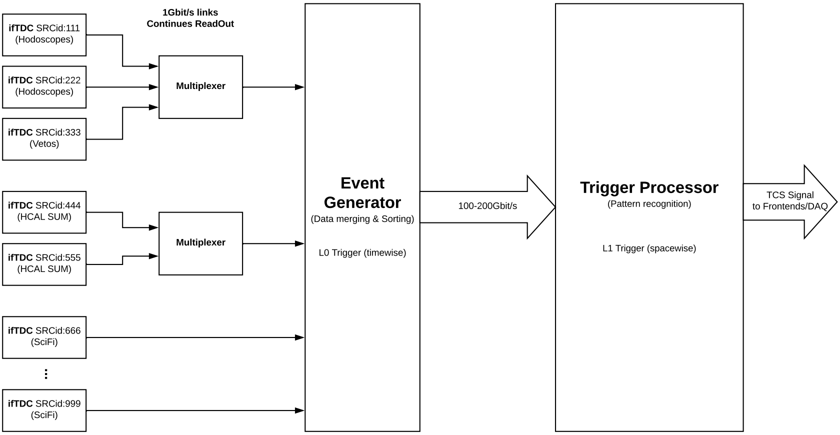
  (according to time resolution of these detectors)
- Flexible pattern recognition in almost real-time (L1).
  (depending on buffer size of front-ends)
  - Hodoscope Trigger (Target-Pointing/Energy loss)
  - Beam/Kink Trigger (PRM measurement)
  - CEDAR Trigger (Trigger on n-fold hits in det.)
  - Calorimeter Trigger (Energy Thr./Cluster recognition)
  - RICH (hadron information)
- Generating list of hits which belongs to possible tracks make trigger decision according to it (and sent them to some HLT).

Basically we need a continuous readout DAQ for the Trigger detectors and the capability to analyse the data in real-time.

Possible trigger detectors:

Type	# Channels	# Stations	Time Resolution	Rate/ch	current Readout
Hodoscopes	758 (550)	11	200-600 ps	$\approx$8 Mhz	TDC/Scaler
Vetos	62 (62)	5	400-800 ps	$\approx$5-15 Mhz	TDC/Scaler
SciFis	2600 (12)	7	$\approx$ 100 ps	$\approx$10 Mhz	TDC/Scaler
HCALs	984 (284)	2	$\approx$800 ps	?	TDC/SADC/Scal
ECALs	4568 (0)	3	$\approx$900 ps	?	MSADC
CEDARs	64 (0)	2	$\approx$350 ps	20Mhz	iFTDC
Sum	**6436 (908)**				
BMS	640	6	$\leq$100 ps	$\approx$ 15 Mhz	F1-TDC
Silicons	13k	7x4	$\approx$? ps	?	MSADC
TPC	?	1	$\approx$? ps	?	high latency
RICH	?	1	$\approx$? ps	?	?
RPD	?	?	$\approx$? ps	?	?
RPC	?	1	$\approx$? ps	?	?

Possible Readout Structure



Continues Readout I

- Protocol similar to current daq readout but with large TDC window (order of μs)

- The readout is initiated by a synchronous clock in parallel to the readout with normal TDC window when a trigger signal arrives via TCS.

Two options for read-out:

Sharing one fiber with the triggered data via UCF (see talk from Stefan Huber)

- UCF $\rightarrow$ complicated layering of data
- less cabling infrastructure
- limited bandwidth (2.5 Gbit/s)
- data stream gets "interrupted" when trigger occours
- data has to be separated at level of multiplexers

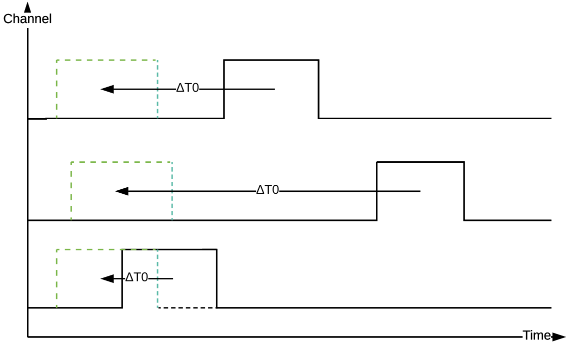
Continues Readout II

Dedicated fiber between front-end and trigger hardware

- Simple protocol like AURORA without layering
- Installation of fibers between front-ends and trigger hardware
- Bonding of remaining three SFP ports to gain bandwidth
- "Uninterrupted" data stream for trigger
- No need of multiplexer for high bandwidth connection directly between trigger processor and front-end

T0 Correction

Timing information of detectors has to be corrected per channel for different propagation times and characteristics of the channels:

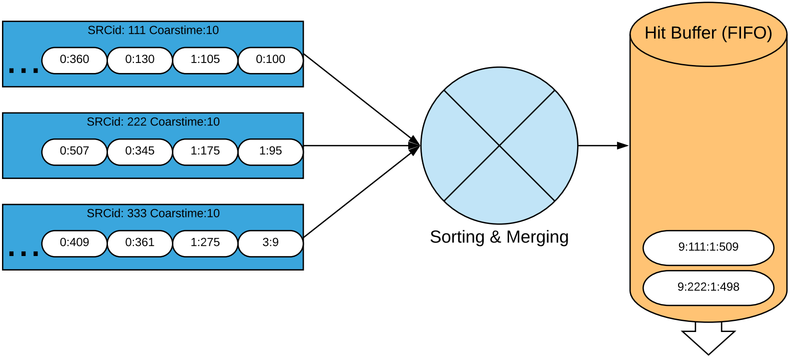


Could be done on level of TDCs, Multiplexer or Trigger processor.

Correction values should be set per channel via IPBus.

Eventbuilding: Sorting & Merging of Data

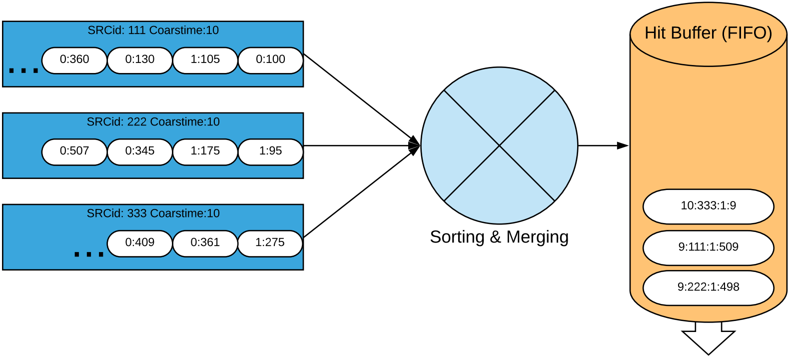
The data from different sources will be filled in one global FIFO buffer:



iFTDC sent frames with SRCid and Coarstime in header and the channel and hits in the data words.

Eventbuilding: Sorting & Merging of Data

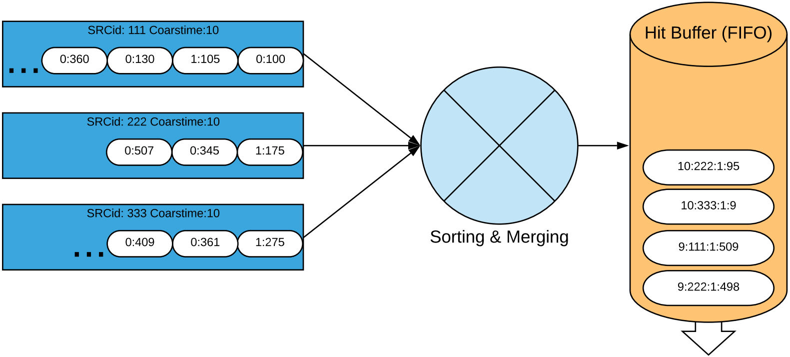
The data from different sources will be filled in one global FIFO buffer:



Hit with the smallest hit time is moved to the hit buffer (FIFO)

Eventbuilding: Sorting & Merging of Data

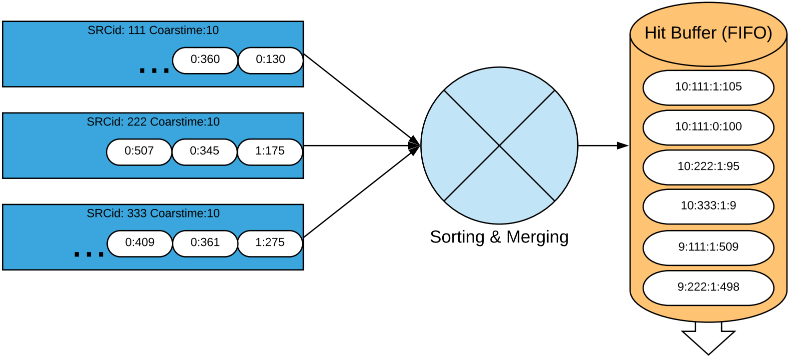
The data from different sources will be filled in one global FIFO buffer:



Next hit with the smallest hit time is moved to the hit buffer (FIFO)

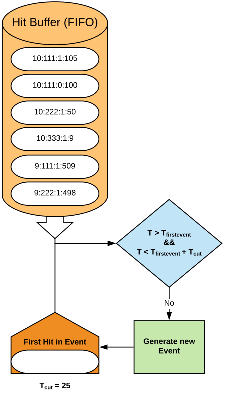
Eventbuilding: Sorting & Merging of Data

The data from different sources will be filled in one global FIFO buffer:



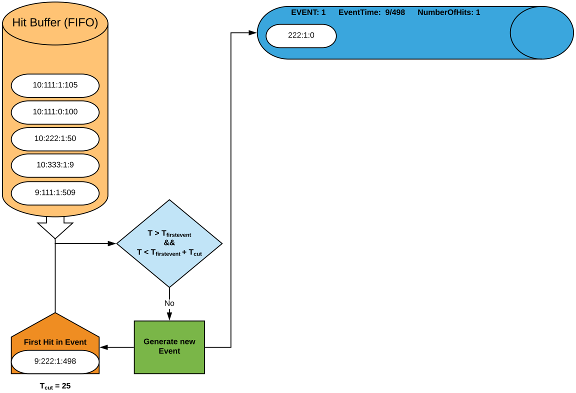
Hit information from different sources are now sorted in Hit Buffer!

Event Building



Sorted hits waiting in hit buffer.

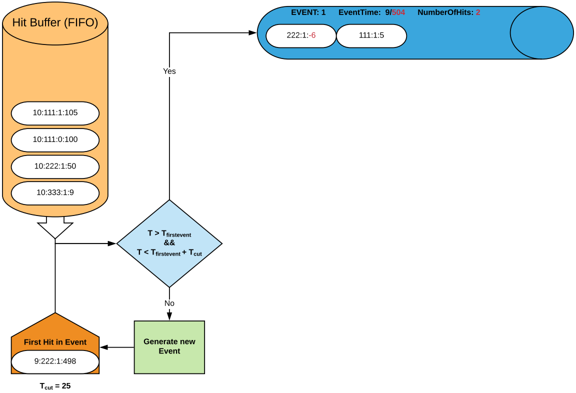
Event Building



No previous event!

→ Generate new event and remember time of first hit in event.

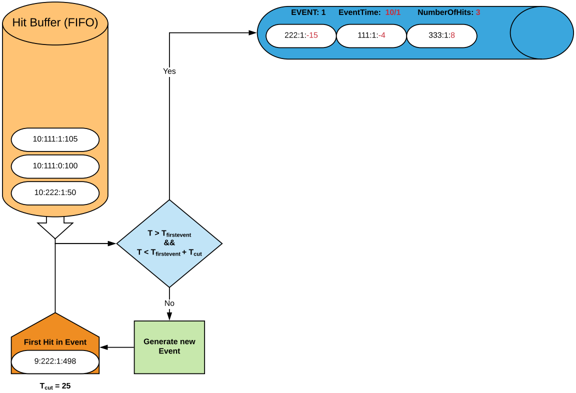
Event Building



Next hit satisfies condition!

$\rightarrow$ Add to event 1 (update EventTime, Nb. hits and hit time).

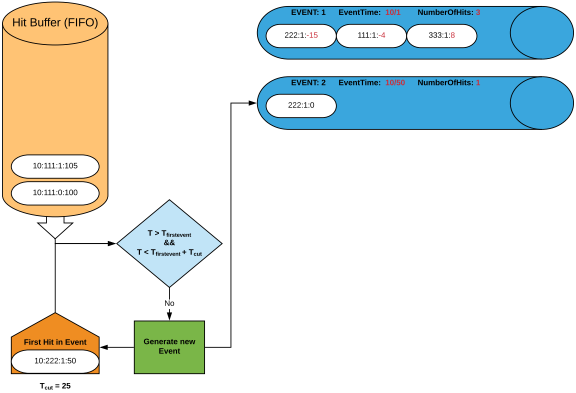
Event Building



Next hit satisfies condition!

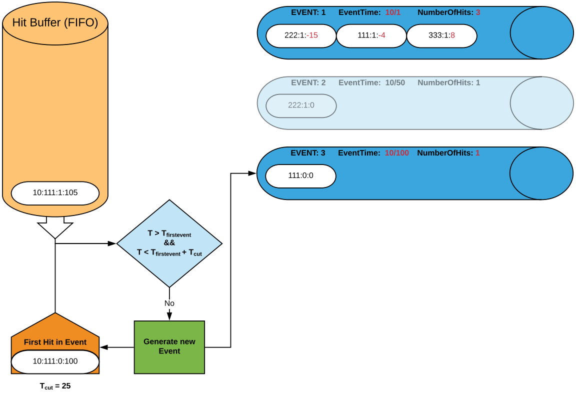
→ Add to event 1 (update EventTime, Nb. hits and hit time).

Event Building



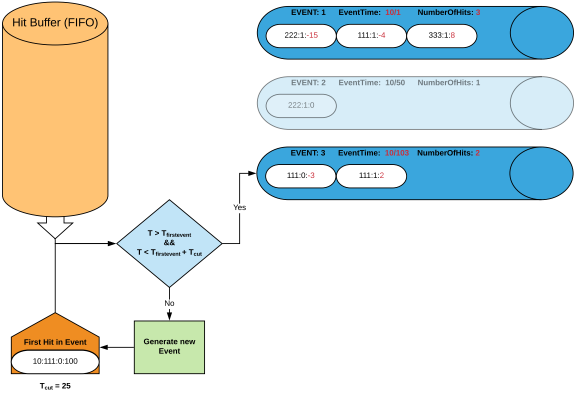
Next hit does not satisfy condition!
$\rightarrow$ Generate new event and remember time of first hit in event.

Event Building



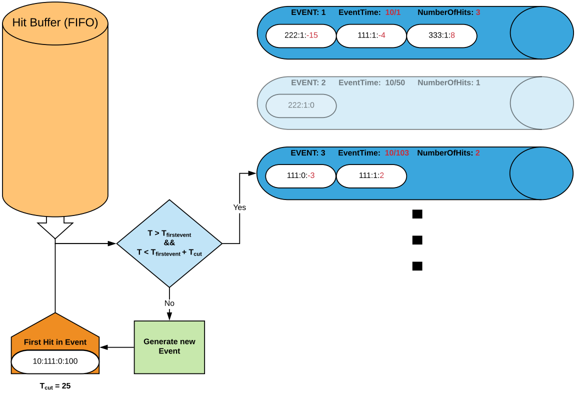
Next hit does not satisfy condition!
$\rightarrow$ Generate new event and remember time of first hit in event.
(do not sent previous event - only 1 hit!).

Event Building



Next hit satisfies condition!
→ Add to event 3 (update EventTime, Nb. hits and hit time).

Event Building



Hits now grouped in timewise events!
Ready to sent to trigger processor.

Event data format

Header 64 bit:

Size [bit]	Type	Comment
8	Event Number	
48	Event Time	Has to be evaluated
6	Number of Hits in Event	$\rightarrow$ 64 hits
2	Error & Flags	e.g. Event size exceeded

Data words 32 bit:

Size [bit]	Type	Comment
10	SRCid	$\rightarrow$ 1024 Frontends
8	Channel	$\rightarrow$ 256 Channels
13	Hit Time	$\rightarrow$ 8192 TDC ticks
1	Extended data frame	next word belongs to this hit

Do we need a fixed length of Events e.g. Header + 29Hits + CRC ?

Event data rates?

... if we have to archive a fixed trigger latency and sending zeros for events with no hits and we assume 100 MHz event rate:

(M2-Beamline intensity max. $\approx$ 4.2 10^8 particles/spill $\rightarrow \approx$ 90 MHz beam rate)

For header + maximum of 29hits/event:

$10^8 * 32 * 32 = 10^{11}$ bit/s $\rightarrow$ **100 Gb/s**

For header + maximum of 61hits/event:

$10^8 * 64 * 32 = 2 \times 10^{11}$ bit/s $\rightarrow$ **200 Gb/s**

For data transfer between eventbuilder and central trigger processor!

We expect something like:

For **Beam/Kink trigger**: 7 Stations x 4 Planes x 2 Cluster size $\approxeq$ **56** hits/event!
For **CEDARS**: 2 Stations x 8 PMT x 2 Cluster size (pc fired) $\approxeq$ **32** hits/event!

Possible Hardware Platform for EventBuilder

ifDAQ XCKU Card

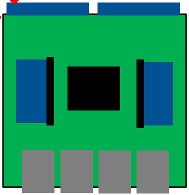
Enough SFP inputs cages (48x) and broadband AMC back-plane (240 Gb/s).

Development of Kintex UltraScale DAQ Module

OUTDATED!

Technical parameters :

- XCKU095-A1156
  - 1.2 M logic cells
  - 60 Mbit Block RAM
  - 64 x 16Gb/s
- **2 x 16 GB DDR4 SODIMM, combined data throughput 10 GB/s**
- AMC connector Interfaces
  - TCS
  - Ethernet(IPBUS)
  - 15 x 16Gb/s
- Front panel interfaces
  - 48 x 16Gb/s



Applications : Data Concentrator, Event Builder, Trigger

Belle2 DAQ/Trigger WS, Tapei 2017

See dedicated talk from Dominic Gaisbauer yesterday!

Pattern recognition

Seperated FPGA for pattern recognition (L1 Trigger)!

- Pattern recognition on event basis.
- More Information as before: more detectors, finer granularity.
  $\rightarrow$ e.g. better di-muon trigger
- Try to get hit candidates and fullfill the purpose of our current trigger matrices in an advanced way.
- Complexity of algorithm depends on available buffer width of front-ends.

Let's play a game!

Pattern recognition

Seperated FPGA for pattern recognition (L1 Trigger)!

- Pattern recognition on event basis.
- More Information as before: more detectors, finer granularity.
  $\rightarrow$ e.g. better di-muon trigger
- Try to get hit candidates and fullfill the purpose of our current trigger matrices in an advanced way.
- Complexity of algorithm depends on available buffer width of front-ends.

Let's play a game!

The Game of Life!

The Game of Life

The Game of Life is a cellular automaton devised by the British mathematician John Horton Conway in 1970 [1]

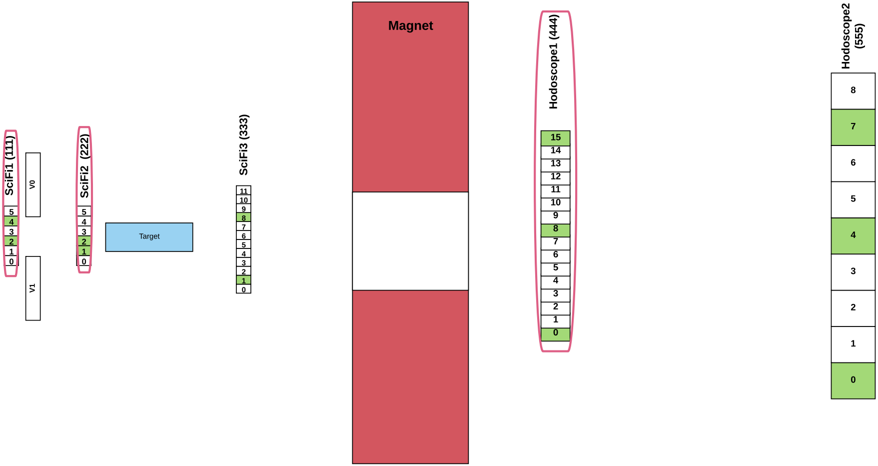
Original properties:

- Two-dimensional orthogonal grid of square cells.
- Each of which is in one of two possible states, alive or dead.
- Each generation is a pure function of the preceding one.
- The initial pattern constitutes the seed of the system.
- Births and deaths occur simultaneously.
- Fixed rules per field.

But we have slightly different rules...

[1] Gardner, Martin (October 1970). Mathematical Games – The fantastic combinations of John Conway's new solitaire game "life"

The Game Board



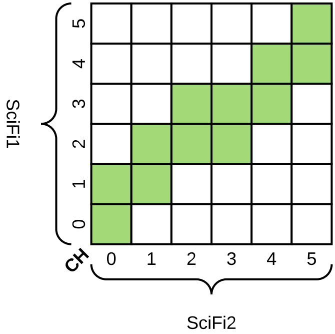
hits in detectors active seeding planes

Rules of our Game

- A new event can be played if there is no Veto activated.
- You start with +20 live points.
- The number of players is fixed (=Nb. of worker nodes).
- You are allowed to join the game if there are still active seeding tiles.
- Every player starts one after an other with one round delay.
- Moves are parallel.
- You move a field every round according to the move rules for this transition .
- You gain +10 points for every successful round.
- You lose -10 points if you not find a next hit.
- You field belongs to you if you need it for the way.
  - if not you have to release it!

You die when you have 0 live points!

Move Rules

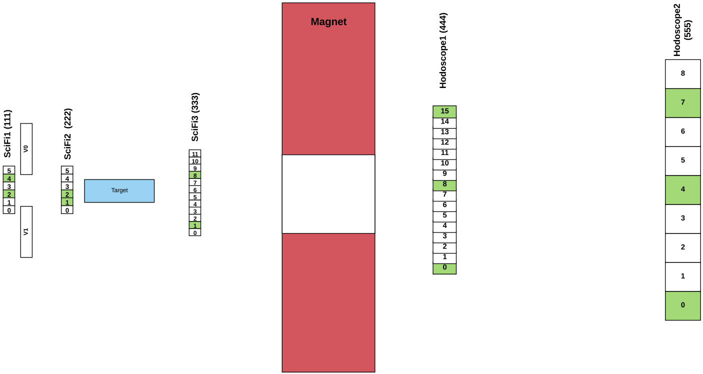


The move rules (like our traditional trigger matrices) descripe the allowed transitions from one plane to another!

(green fields show allowed combinations)

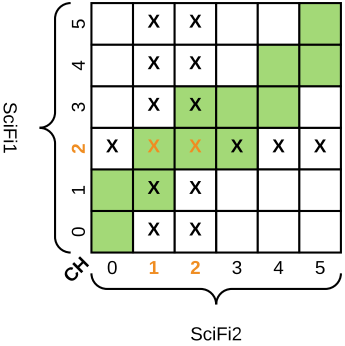
Move 1 : Player 1 starts

Event can be played since no active Veto field!



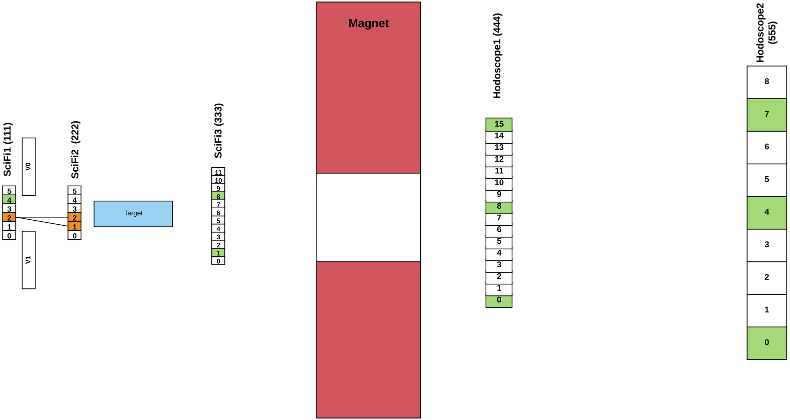
Can choose on of the active fields in seed plane 1 and play a move according to move rules between SciFi1 and SciFi2.

Move Rules transition plane 1-2



Start from CH2@SciFi1 allows transition to CH1&2@SciFi2 ✓ +10 points

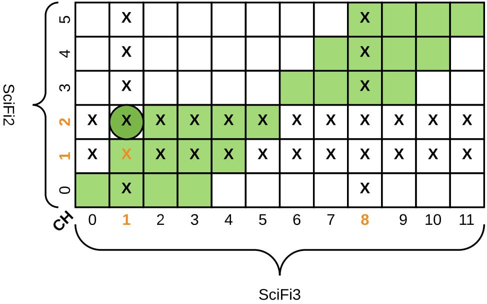
End of Move 1



Player 1 moves from CH2@SciFi1 to CH1&2@SciFi2 $\rightarrow$

P1 = 30 live points

Move 2 : Player 1

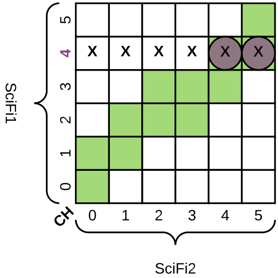


Starts from CH1&2@SciFi2 (predicted positions)

CH1@SciFi2 →CH1@SciFi3 ✓ +10 points

release CH1@SciFi2 (not needed as way point - can be seed field for next player)

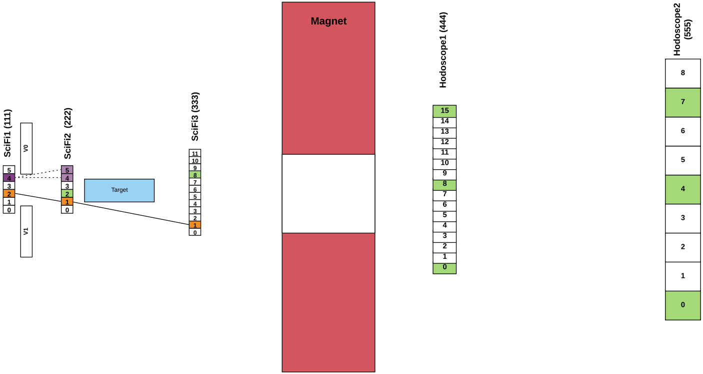
Move 2 : Start Player2 (In Parallel)



Player 2 starts from CH4@SciFi1 but do not find any allowed transition to SciFi2 -10 points

Using CH4&5@SciFi2 as predicted start point since player has still live.

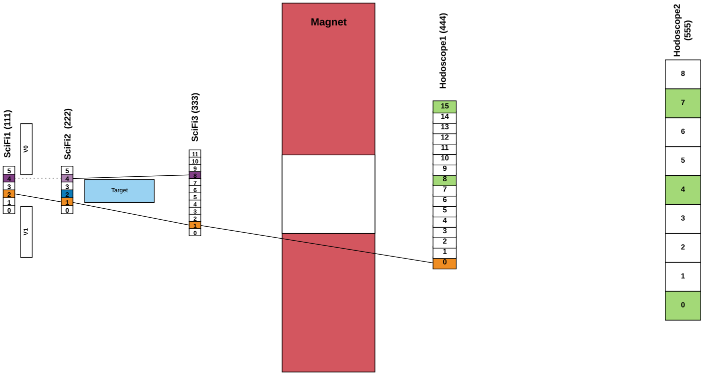
End of Move 2



Player 1 moves to CH1@SciFi3 and releases CH2@SciFi2!
Player 2 possible fields 4&5@SciFi2

Live points: P1=40 P2=10

End of Move 3



SciFi1 (111)
V0
V1
5
4
3
2
1
0
SciFi2 (222)
5
4
3
2
1
0
Target
SciFi3 (333)
11
10
9
8
7
6
5
4
3
2
1
0
Magnet
Hodoscope1 (444)
15
14
13
12
11
10
9
8
7
6
5
4
3
2
1
0
Hodoscope2 (555)
8
7
6
5
4
3
2
1
0

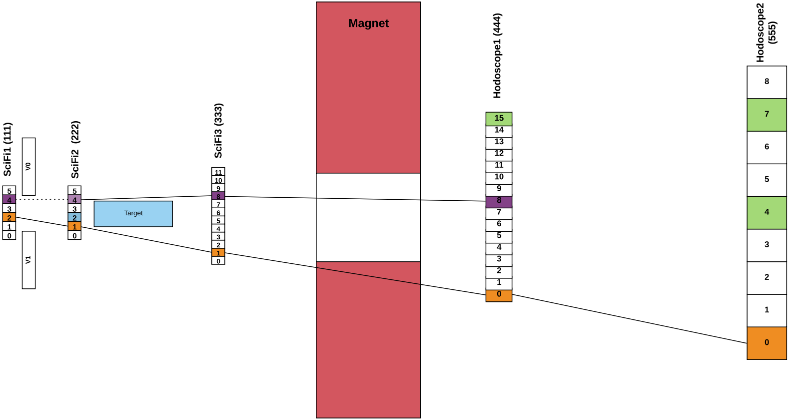
Summary Move 3

- **Player 1:**
  Moves from CH1@SciFi3 to CH0@Hodo1 ✓ +10 points
- **Player 2:**
  Found hit in SciFi2 in last round but according to moving pattern he would be on field 4&5@SciFi2
  $\leftarrow$ Use this as start points for transition SciFi2$\rightarrow$SciFi3
  Moves from CH4@SciFi2 to CH8@SciFi3 ✓ +10 points
- **Player 3:**
  Start using CH2@SciFi2 as next active seeding field.
  Does not find hit in SciFi3 -10 points

  (possible fields in SciFi3: 2-5)

Live points: P1=50 P2=20 P3=10

End of Move 4



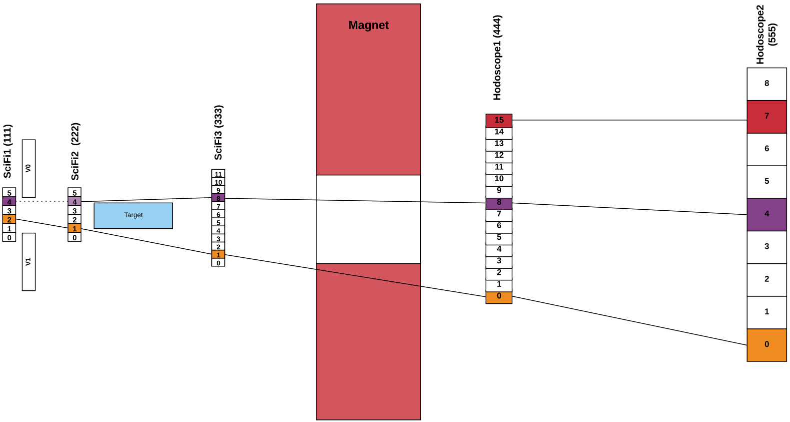
SciFi1 (111)
V0
V1
5
4
3
2
1
0
SciFi2 (222)
Target
SciFi3 (333)
11
10
9
8
7
6
5
4
3
2
1
0
Magnet
Hodoscope1 (444)
15
14
13
12
11
10
9
8
7
6
5
4
3
2
1
0
Hodoscope2 (555)
8
7
6
5
4
3
2
1
0

Move 4

- **Player 1:**
  Moves from CH0@Hodo1 to CH0@Hodo2 ✓ +10 points
- **Player 2:**
  Moves from CH8@SciFi3 to CH8@Hodo1 ✓ +10 points
- **Player 3:**
  Using CH2-5@SciFi3 as stating point but did not find any hit in Hodo1 -10 points → Dying!

Living points: P1=60 P2=30 P3=0

End of Move 5



Live points: P1=60 P2=40 P3=0 P3=30

Information at End of Game

"Easy" parallel algorithm with fixed latency.

Scores: P1=60 P2=40 P3=0 P3=30

- **Live points** indicates probability of good track:
  → Use weight functions to make trigger decision.
- **Number of track candidates:**
  → e.g. improved di-lepton identification.
- Each player has a **list of hits which are track candidates**.
  → Could be sent to further computation stages:
  - Hit Information translate via LUT to hit coordinates.
  - Could be directly used to calculate variables like: Φ, Θ, Q^2, Y, ...
  - Performing online track fitting

Accepted trigger event lets to generation of Trigger pulse after delay to obtain fixed latency and backwards compatibility to current TCS hardware... needed?

Many open Questions ...

- How to deal with different plans from detector station (Feature extraction on stage of the station to extract "pixel" information).
- What is the best frame-size for TDC frames.
- How to deal with overlapping events (in time).
- How to do it with 2D hit information.
- How to deal with different time resolution between readouts.
- Is the computation power enough to do pattern recognition in time budget.
- How to assure fixed latency.

Simulation for the trigger algorithm needed!

Possible Hardware Platforms

Also **ifDAQ XCKU Card** which is connected over fibre to the event-builder.

OR:

XCKU-PCIe Card (like: Xilinx KCU116, Alveo - U250 , ...)

- "Cheaper" since no custom hardware.
- PCIe interface could be used for monitoring/debugging.
- PCIe interface could be used to interface with CPUs or GPUs (DMA).
  - Low price high performance calculation power (high level trigger).
  - But high latency for transferring data to CPU/GPU memory (O(ms)). **Depending on maximal trigger latency!**
  - Uses AMC to FMC adapter (or QSFP28) to transfer data between event-builder and trigger processor.

Summary

Long story short:

We need a dedicated continuous read-out DAQ for a subset of detectors which is able to deal with particle rates up to 90 MHz to build events depended on the hit time and make an pattern recognition in almost real time (depends max. trigger latency).

Test bench for a whole "trigerless" readout of the spectrometer!

Three main Components:

- **Continuous readout TDCs** with resolution down to 100 ps.
- A **event-builder** who merge and sort the data to build timewise events.
- A **trigger-processor** which makes the pattern recognition event by event at full rate.

Lot of work to be done ...!