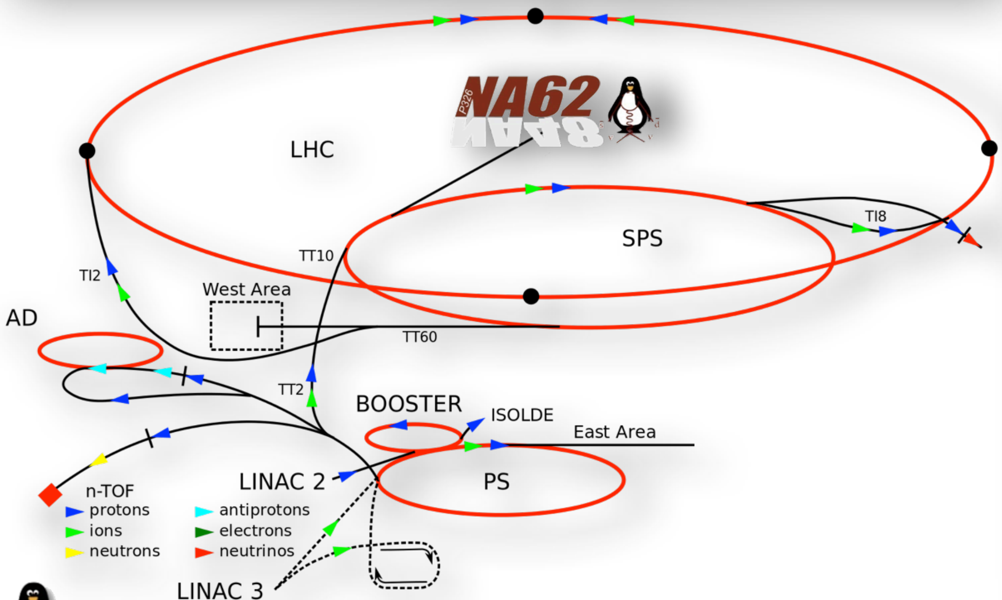
TWEPP 2016, Karlsruhe

Real-Time FPGA Design for the L0 Trigger of the RICH Detector of the NA62 Experiment at CERN SPS

Mattia Barbanera, Francesco Gonnella

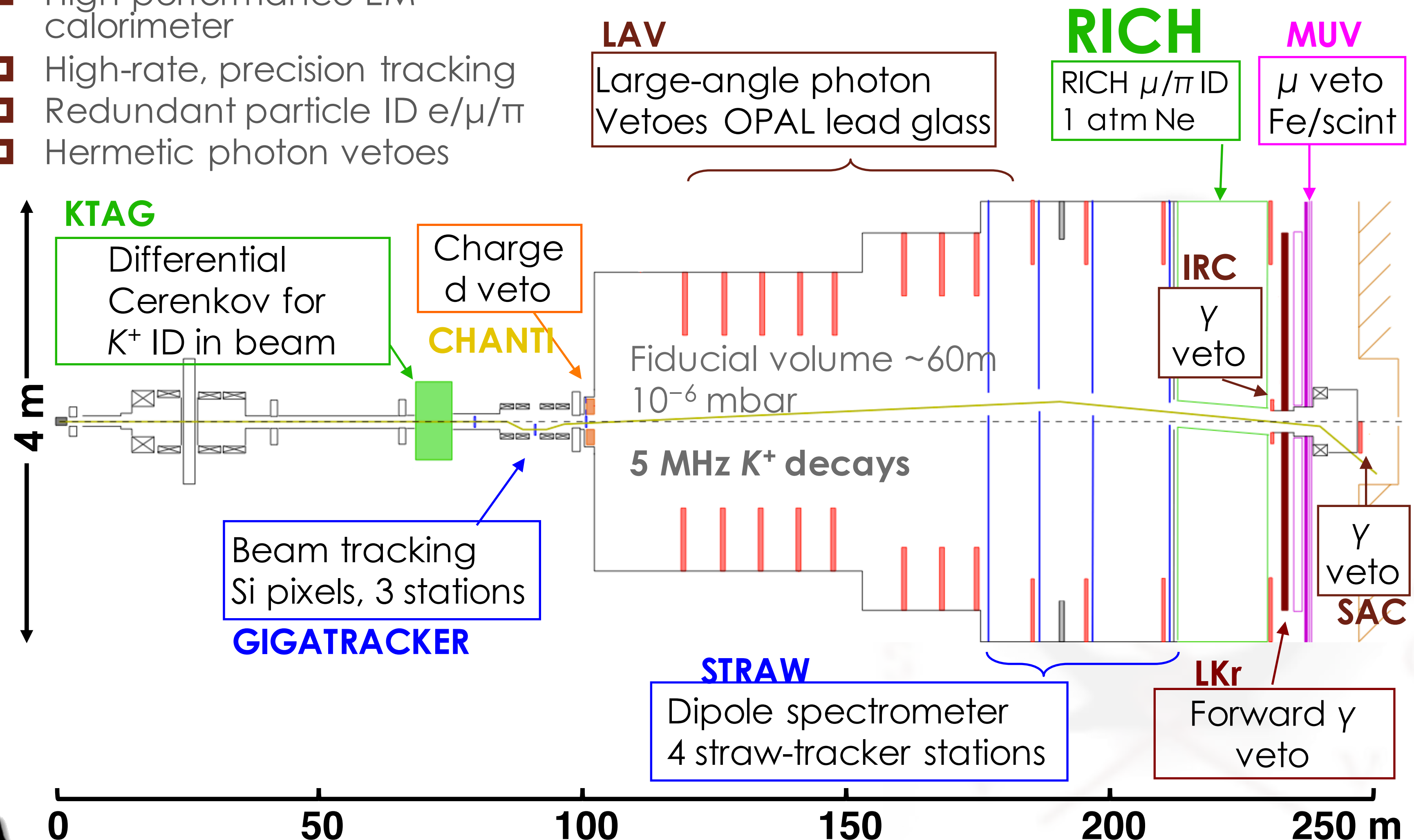
The NA62 experiment



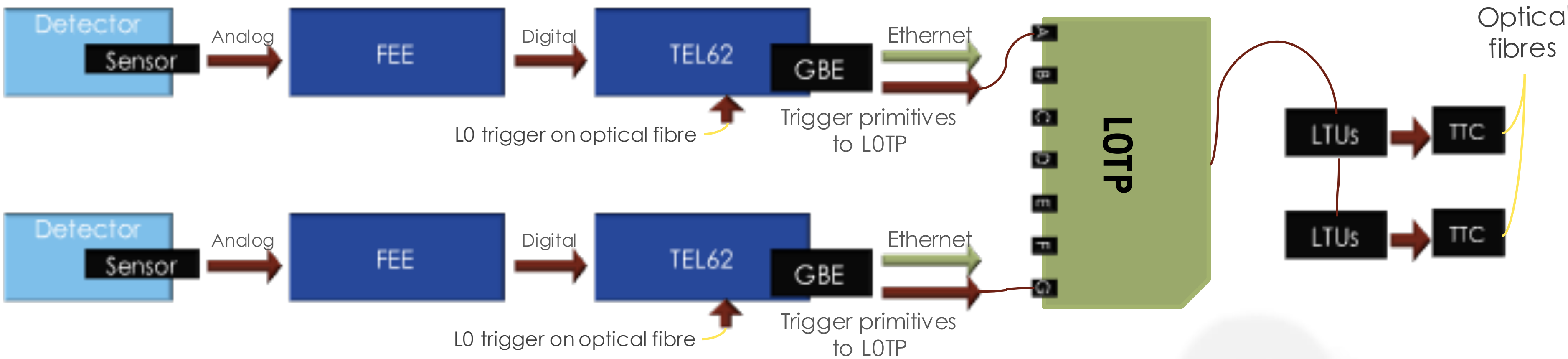
NA62 detector layout and principles

Measure the $K^+ \to \pi^+ \nu\bar{\nu}$ BR to 10% precision collecting O(100) events

- High-performance EM calorimeter
- High-rate, precision tracking
- Redundant particle ID e/μ/π
- Hermetic photon vetoes

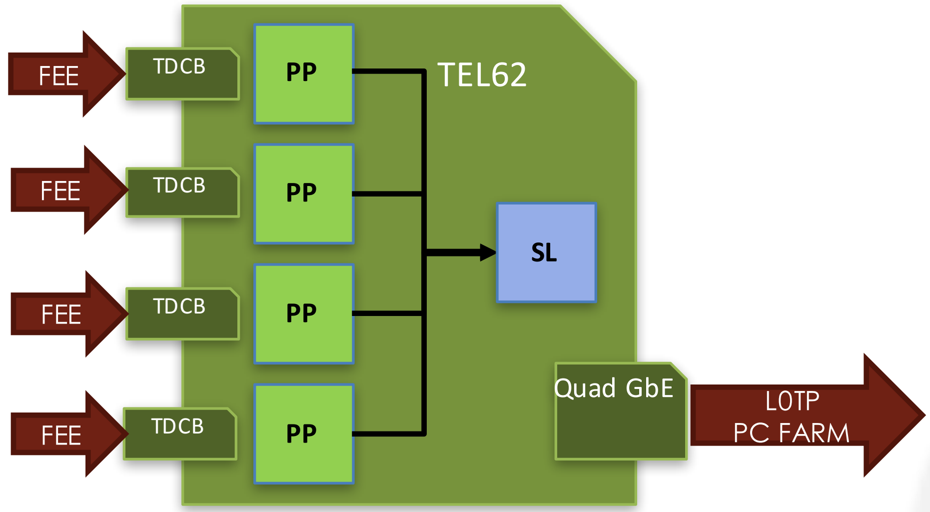


Trigger and Data AcQuisition



- Three-level trigger to reduce the amount of stored data, from 10MHz to 100kHz of events:
  - Level-0: hardware trigger; reduction factor 10
  - Level-1 and Level-2: software triggers; reduction factor 10
- Some sub-detectors participate in the level-0 trigger, producing fast and small-sized information to be sent to the L0TP: "**trigger primitives**"
  - Primitives contain time information (~100 ps LSB) and a **primitive ID** containing reduced information different for every detector
- L0TP produces the L0 trigger if predefined conditions are satisfied
  - Some detectors using TEL62 have dedicated FW creating trigger primitives

RICH L0-FW: General Working Principle

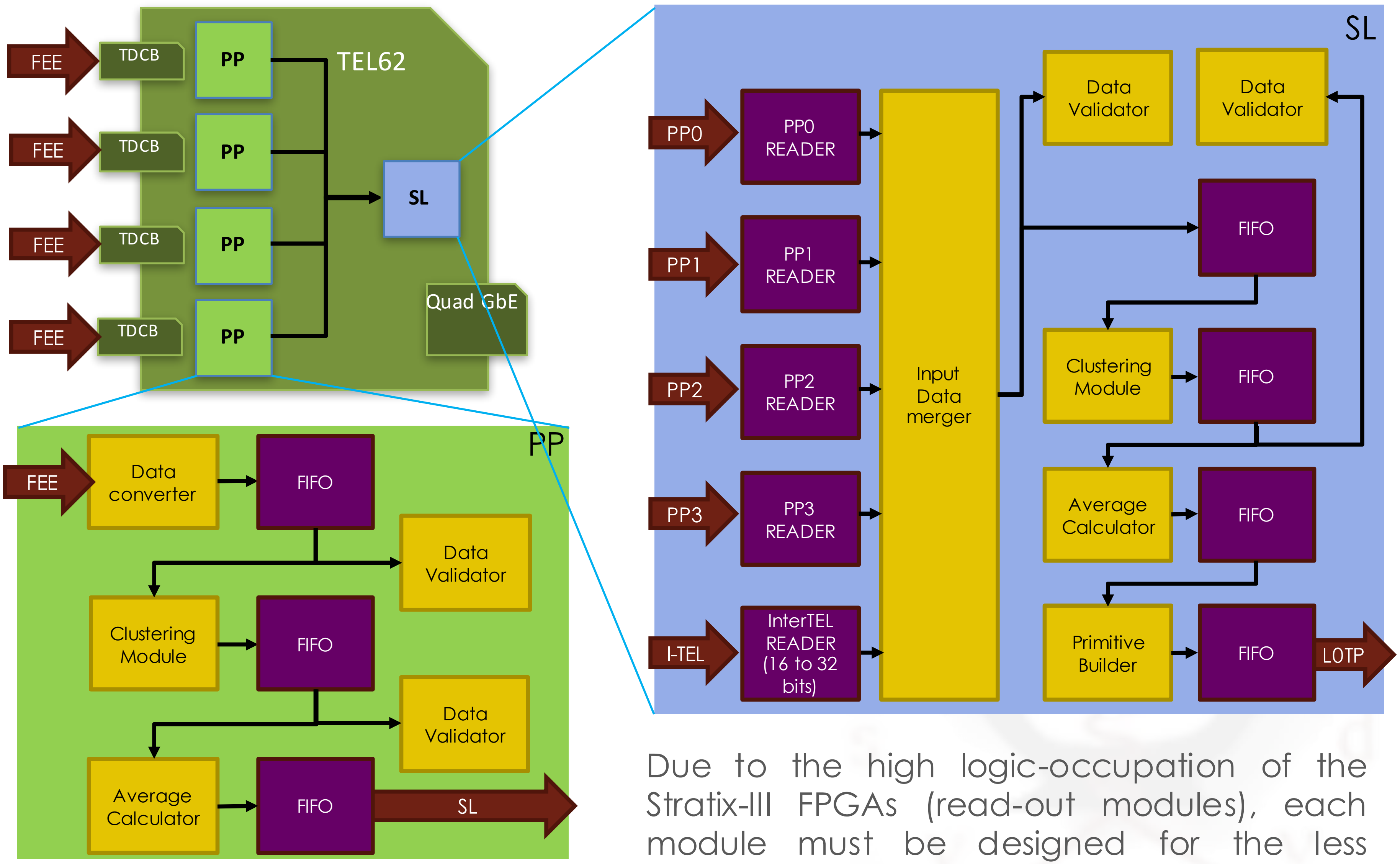


- The aim of the RICH L0 primitive generating firmware is to group together hits belonging to the same Cherenkov circle, creating time clusters (very general)
- In the PP a preliminary clustering is performed and in the SL clusters coming from the 4 PPs are merged together
- In the final stage of the SL clusters are used to produce primitives to be sent to the L0 Trigger Processor

RICH L0-FW: General Working Principle

- All modules are clocked at 160 MHz
- The delay in production of primitives must be less than 5 time frames of 6.4-us (basic time-division of the experiment)
- Inside the firmware a common data-format (**RICH format**) is used
  - All the modules must accept RICH format as input and output format, so that they can be freely moved inside the firmware
- A common clustering module is implemented, able to accept the RICH format as input and output
- It is possible to use a multiple TEL62 setup, by connecting in daisy-chain all the SLs with Inter-TEL boards (as foreseen by the collaboration in the next years)
  - In this case, only the last board sends primitives to L0TP

RICH L0-FW: Working Scheme



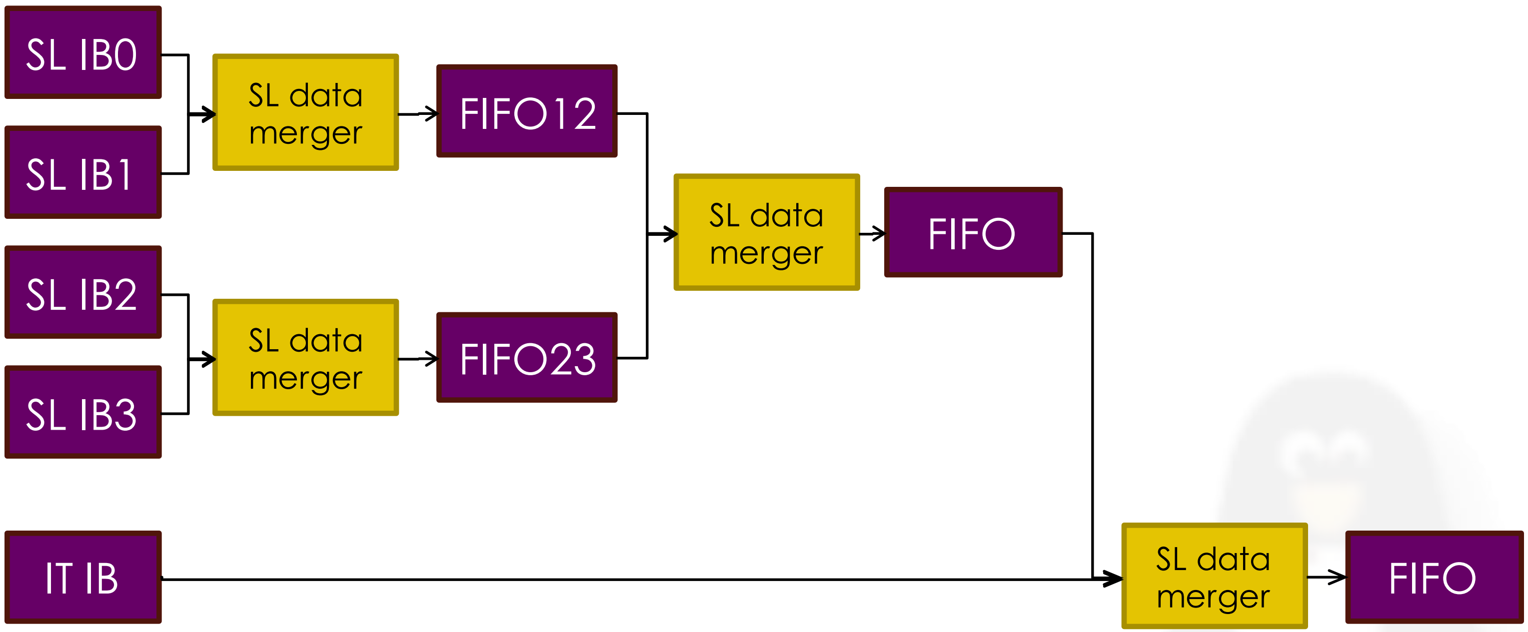
Due to the high logic-occupation of the Stratix-III FPGAs (read-out modules), each module must be designed for the less occupation and maximum clock-speed

Data Converter: RICH format

- Reads TDC data sorted in frames of 25 ns and converts them into RICH format: each hit is a cluster with N_{hits}=1 and CTS=0 (Cluster Time-Sum)
  - 400 ns time stamp, cluster fine-time and time-sum of the differences between the cluster time and the time of every hit belonging to the cluster
  - Being the sum signed, on average its value is small even if the cluster is made of a significant number of hits
- It produces 16 400-ns TimeStamps per 6.4-us frame
  - If there are no data corresponding to that TS, a fake cluster (speed-data) with N_{hits}=0 and SUM=0 is produced
- This module can handle 1024–16=1008 words per 6.4-us frame
- All the data can be split into 2 16-bit words to be sent through the Inter-TEL bus

31 30	29 28 27 26 25 24	23 22 21 20 19 18 17 16	15 14	13 12	11 10 9 8	7 6 5 4 3 2 1 0
T.S. 1/2	Timestamp (27:14) 400 ns		**T.S. 2/2**	Timestamp (13:0) 400 ns		
Data 1 1/2	N_{hits} (7:2)	Cluster Time-Sum (7:0) (signed) LSB = 100 ps	**Data 1 2/2**	N_{hits} (1:0)	Fine Time (11:0) LSB = 100 ps (up to 400 ns)	
Data 2 1/2	N_{hits} (7:2)	Cluster Time-Sum (7:0) (signed) LSB = 100 ps	**Data 2 2/2**	N_{hits} (1:0)	Fine Time (11:0) LSB = 100 ps (up to 400 ns)	

SL Input Data merger

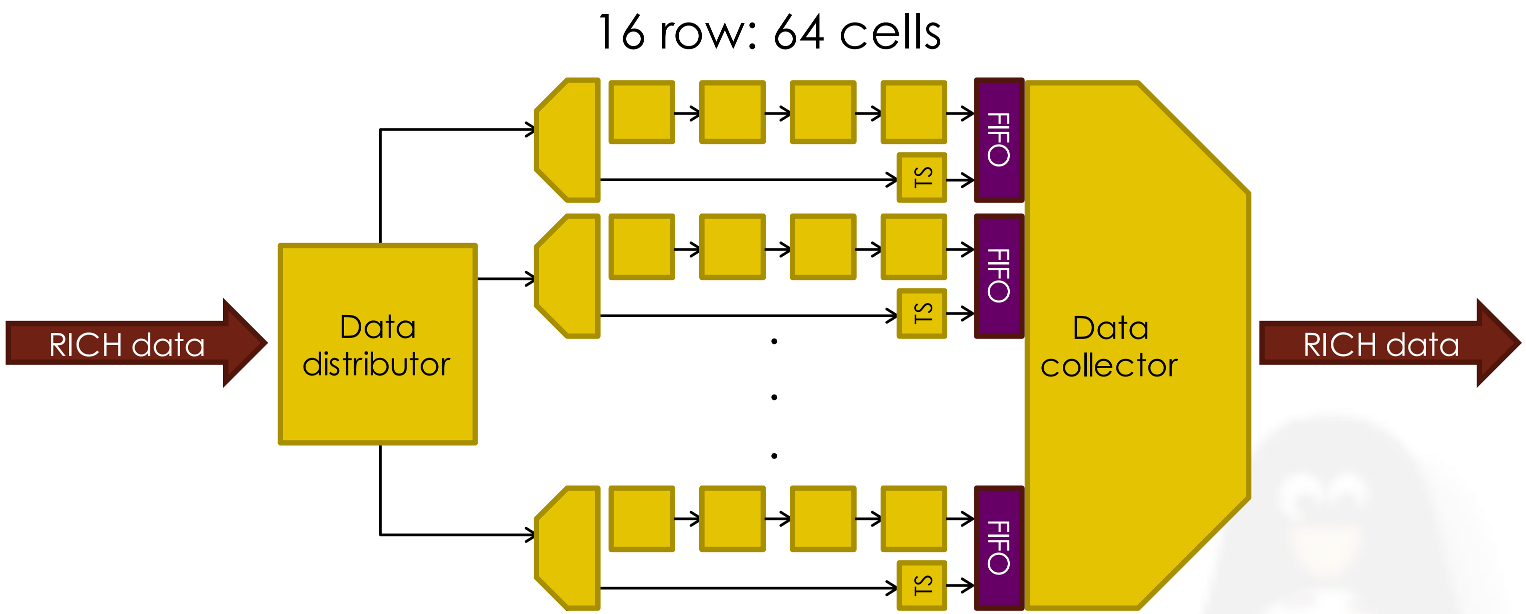


- "SL data merger" is purely combinatorial and merges the clusters from two sources
  - Clocked at the same frequency of the other modules (160 MHz)
  - Waits that both its input FIFO are not-empty
  - Compares the type of words (TS, data, speed-data) and their time
  - Produces data sorted in frames of 25 ns (as the input of PP), with no replication of TS or speed-data

SL average calculator

- The geometric mean of the time of the hits is achieved using the CTS field
  - Each time the clustering module has a hit (or a pre-cluster, in the SL) in input, the sum is updated:
    - $CTS_{new} = CTS_{old} + N_{new}(t_{seed} - t_{input})$
  - The average calculator computes the reference time as follows
    - $Seed_{new} = Seed_{old} + CTS_{old}/N_{old}$
  - The multiplicity remains the same and the sum is re-set to 0
    - $N_{new} = N_{old}$
    - $CTS_{new} = 0$
- Used in the PPs in order to have clusters with more precise reference time and to avoid overflows of CTS field in the SL
- The final time is computed with the calculator in the SL

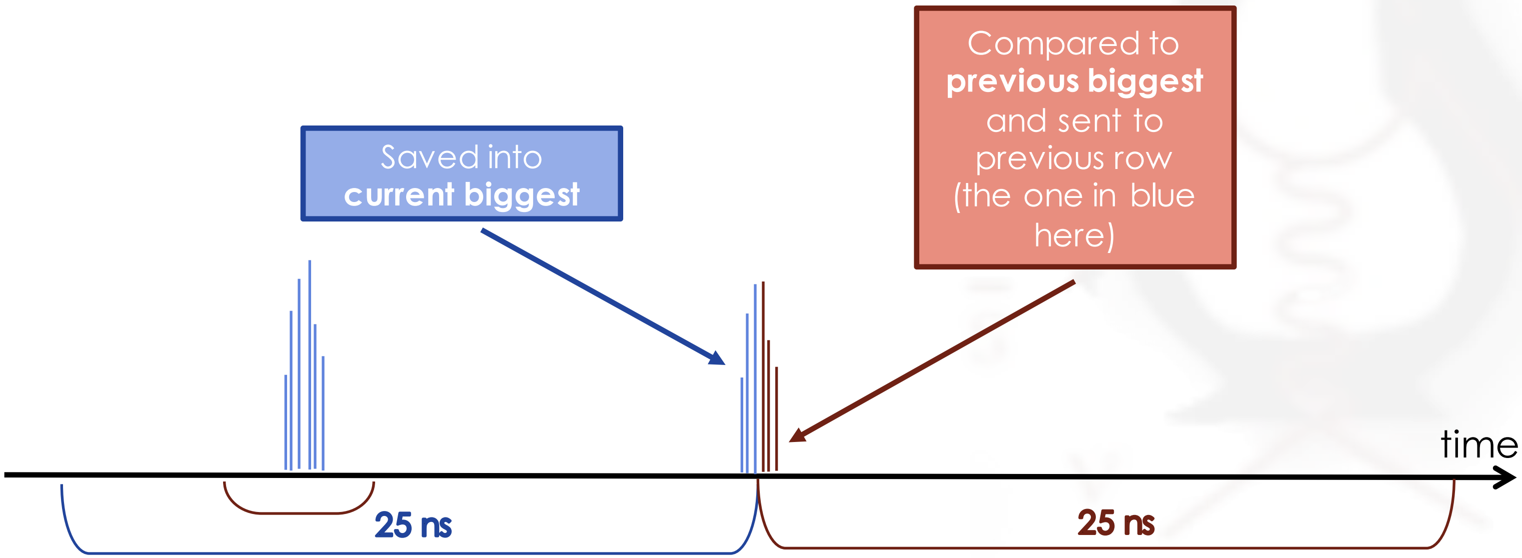
Clustering-module: working principle



- 4 cells to handle a 25 ns time slot means an instantaneous rate of clusters of 160 MHz
- 16 rows of cells are used to guarantee a through-put of 1, while handling many time frames
- Rows are used in cycle: in case a cluster must be formed with events in two adjacent frames, the data distributor sends the hit to the previous row

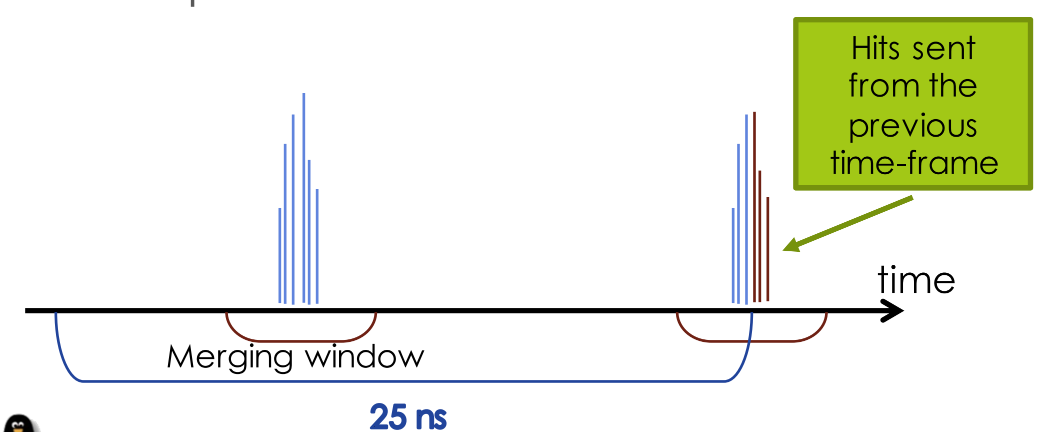
Clustering-module: Data Distributor

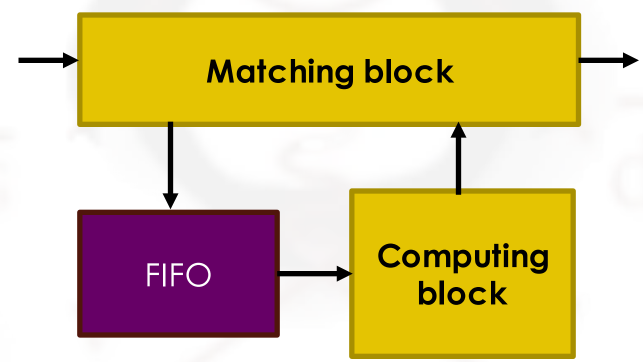
- Rearrange data in input into TS@25 ns (32 bit) and fine time@100 ps (8 bit)
- Delivers data to the proper row, splitting it into TS@25 ns (32 bit) and fine-time@100 ps (8+1 bit)
- Handles clusters split into two adjacent TS@25ns by sending them to the proper row
  - the 9th bit is set to one if the cluster belongs to the row used at the moment, to zero otherwise



Clustering-module: Cell

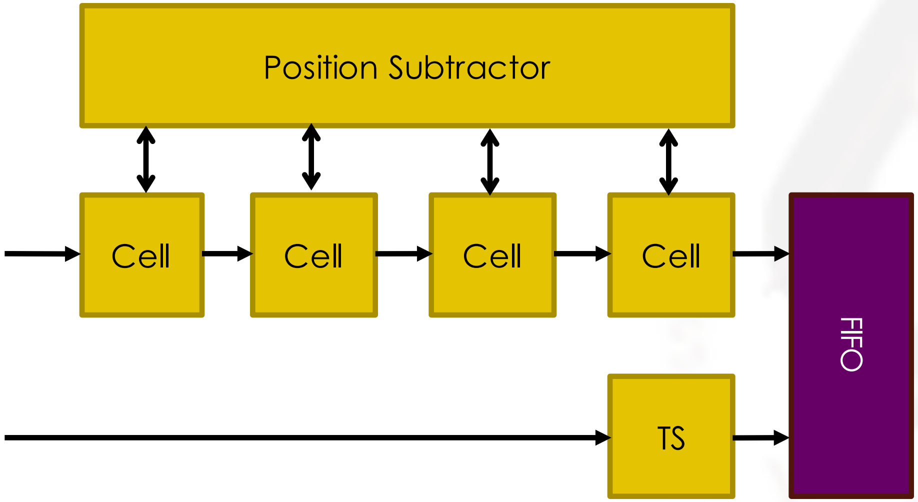
- Each cell stores the first $Time_0$ (9 bit) received
- If the $Time_1$ of successive cluster matches the stored $Time_0$, within a programmable time window, it merges the 2 clusters as follows:
  - $CTS_0 \mathrel{+}= N_1 * (Time_1 - Time_0) + CTS_1$
  - $N_0 \mathrel{+}= N_1$
- Divided in 2 block separated by a FIFO
  - The matching block, handling the comparison between $Time_0$ and $Time_i$
  - The computing block, handling the computations (containing an FPGA-embedded multiplier)
- When the flush-mode is enabled, it acts as a shift register, giving as output the stored cluster





Clustering-module: Sorting

- Dedicated electronics for the sorting of the clusters
- Each cell has an internal position field, that is appended to the output cluster
- The position field increases when a cluster with time bigger than the cluster seed passes through the cell
  - All the seeds that cannot fit in the row (i.e. from the 4th seed in a row) are subtracted from each cell

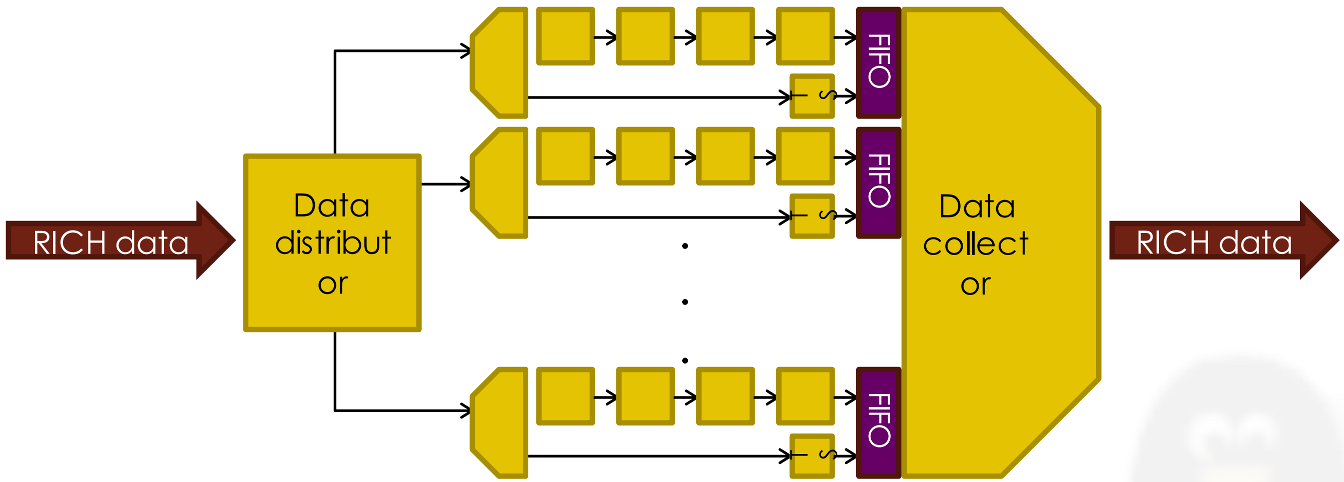


Clustering-module: Data Collector

- Retrieves data produced by the 16 rows of clustering cells, sorts them and re-converts them in RICH format

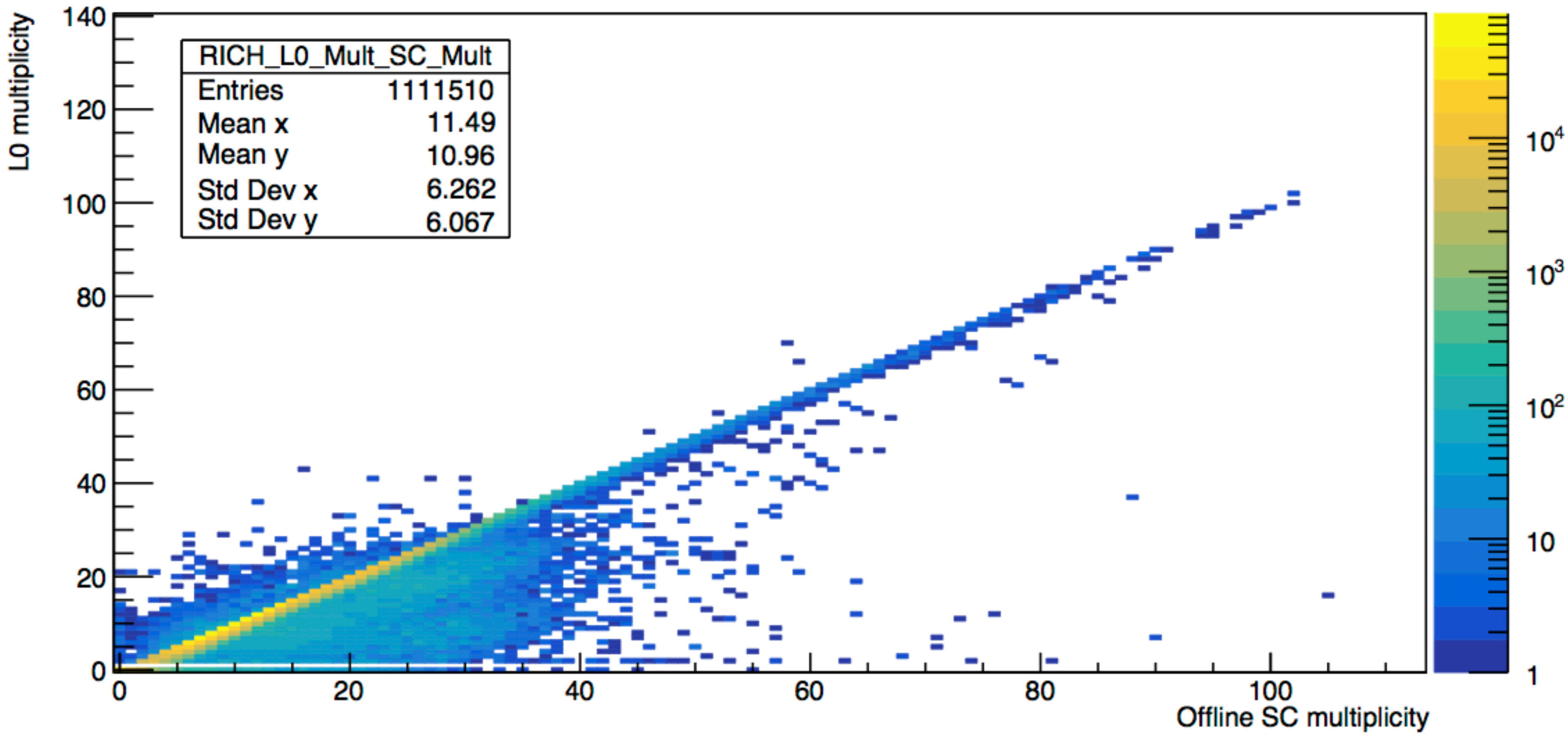
- **Reader** of the rows: in order to read 1 word per clock cycle, reacts to the empty signals of two consecutive rows
- **Sorter**: sorts the clusters by addressing a RAM with the position field computed by the cells. In order to stand the full rate, there are 8 RAMs
- **Cluster Discard**: discard the clusters that have multiplicity out of a predefined range
- **Formatting** module: formats the cluster in the RICH format

Clustering-module: Performance



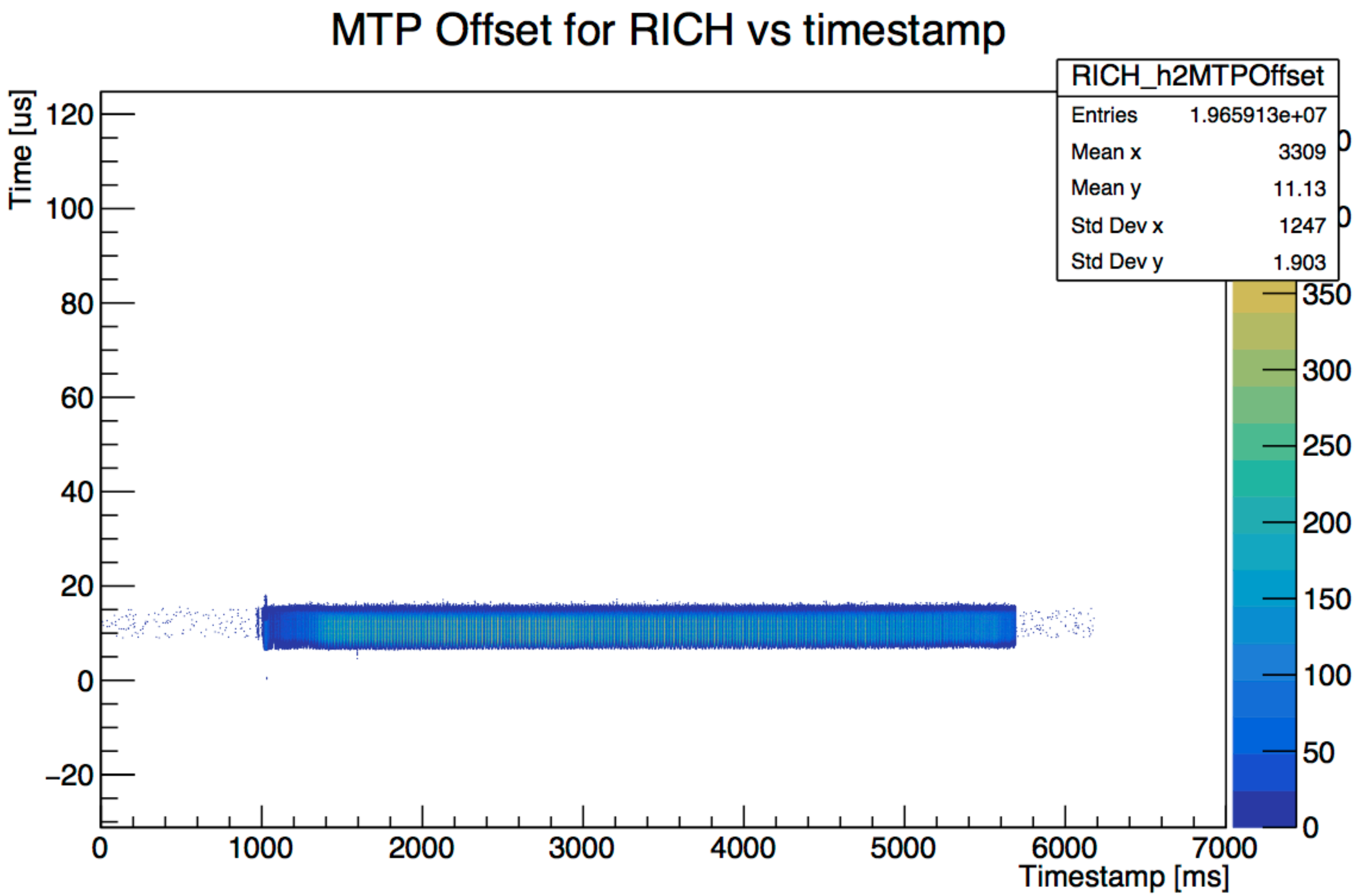
- The **throughput** of the clustering module (like any other module in the RICH firmware) must be kept at **1 word per clock cycle**
- For this reason we need 16 rows of clustering:
  - Two rows are filled “at the same time” to take care of border effect
  - Once the second row is completed, the first can be read out
  - 16 rows are needed to compensate the latency of the cell:
    - The latency of one cell is given by: ***2d + m + f*** where *d* is the depth of the row, *m* is the latency of the multiplier and *f* is the latency of the internal FIFO
    - In our case 2*4 + 3 + 3 = **14** < 16, so there will always be an empty row to fill

L0 multiplicity vs offline multiplicity



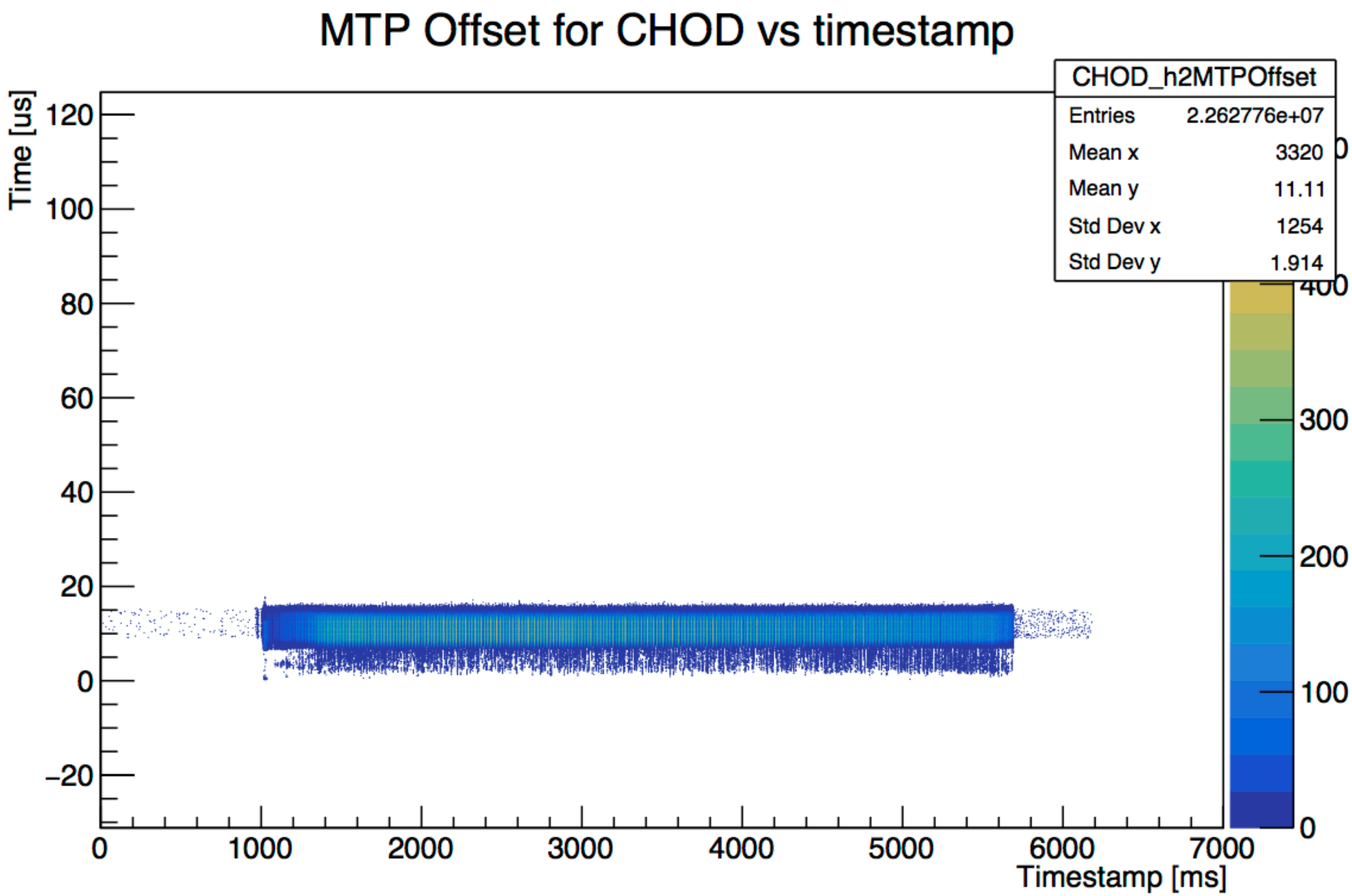
- Correlation between offline multiplicity and FW multiplicity
  - The line corresponding to FW multiplicity 0 represents the inefficiency of the FW algorithm

Delay of primitive production



- Delay is stable and between 2 and 3 time-frames of 6.4 us each
  - FEE sends data in time-frames of 6.4 us

Delay of primitive production

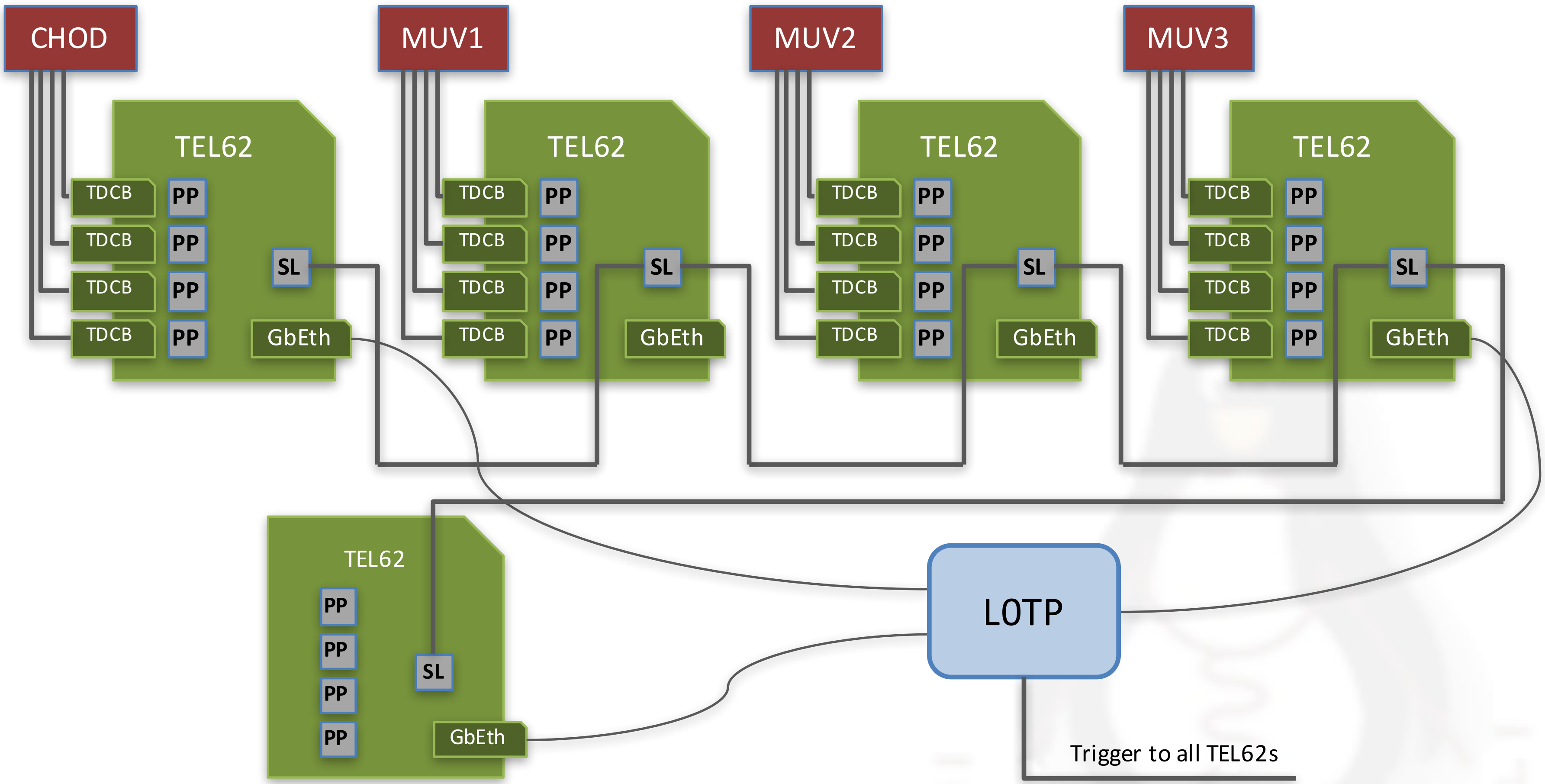


- Because of its generality, the FW has been employed also for L0 of CHOD detector
  - With a higher rate (~15%), the delay tends to diminish

Conclusions

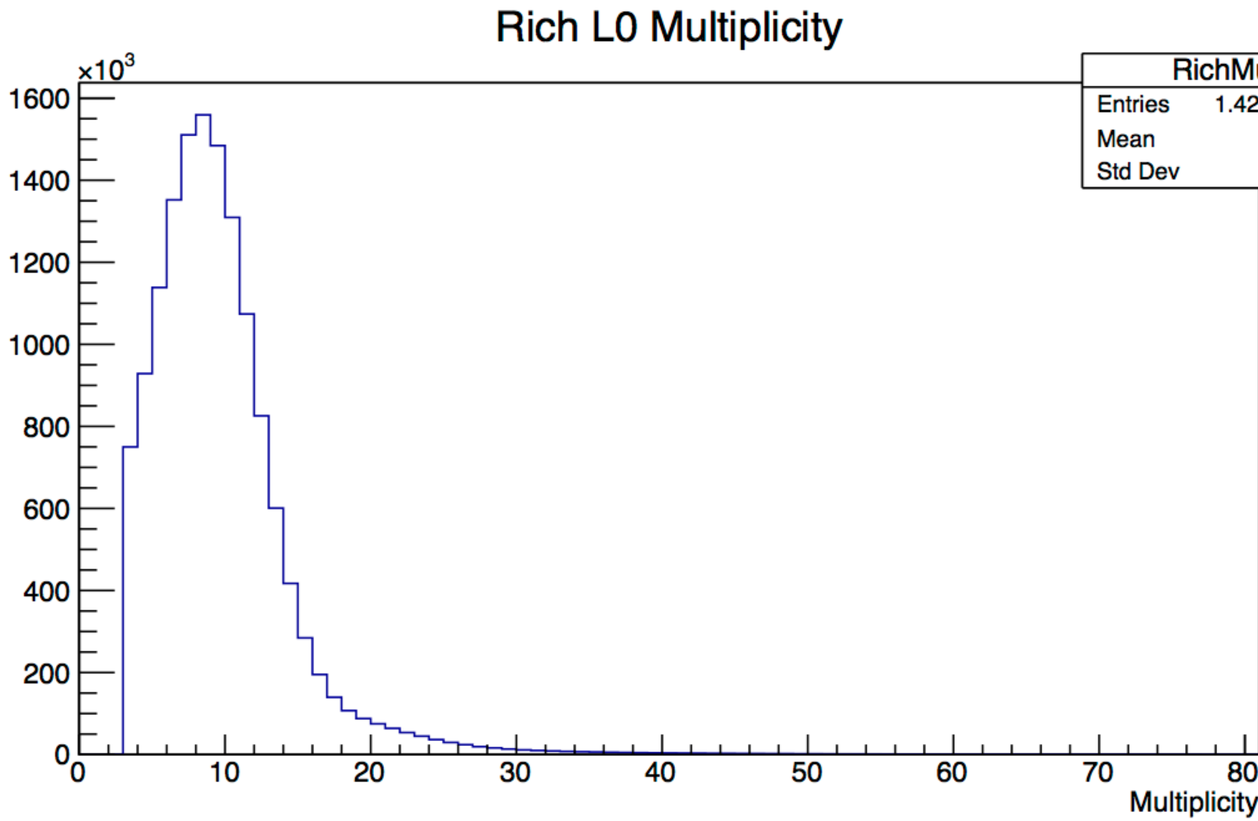
- A Firmware for RICH Level-0 has been developed and it's working with an efficiency of 98.76%
- The system can stand the full rate of the detector
  - Real rate of the detector is twice the one of the MC: a single Gb-Ethernet cannot stand the primitive rate
- The maximum delay of primitive production is 3 time-frames of 6.4 us each
  - The higher is the rate, the faster is the production (up to the saturation of the GbE link)
- Because of it's generality, it has been employed also for the L0 of CHOD detector and it's ready for the use of Inter-TEL boards, foreseen by the NA62 collaboration for the years to come

InterTEL configuration

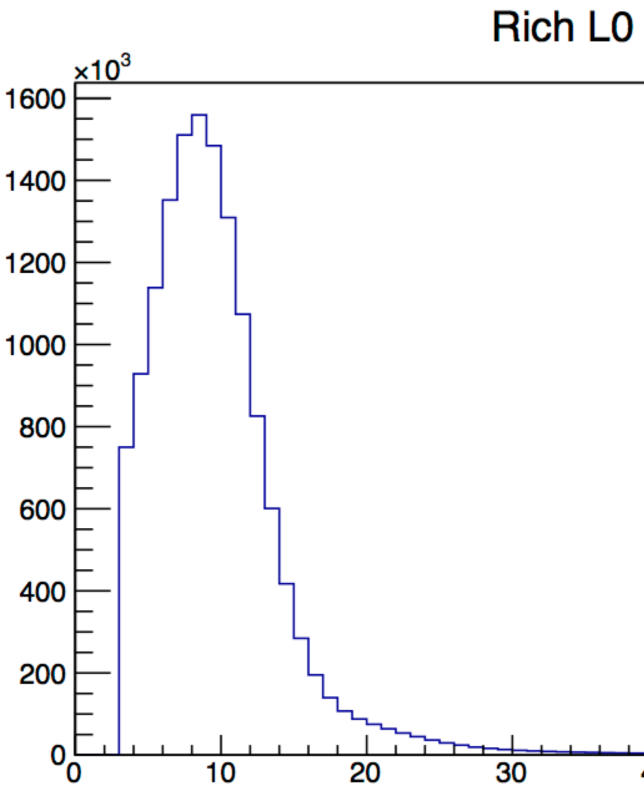


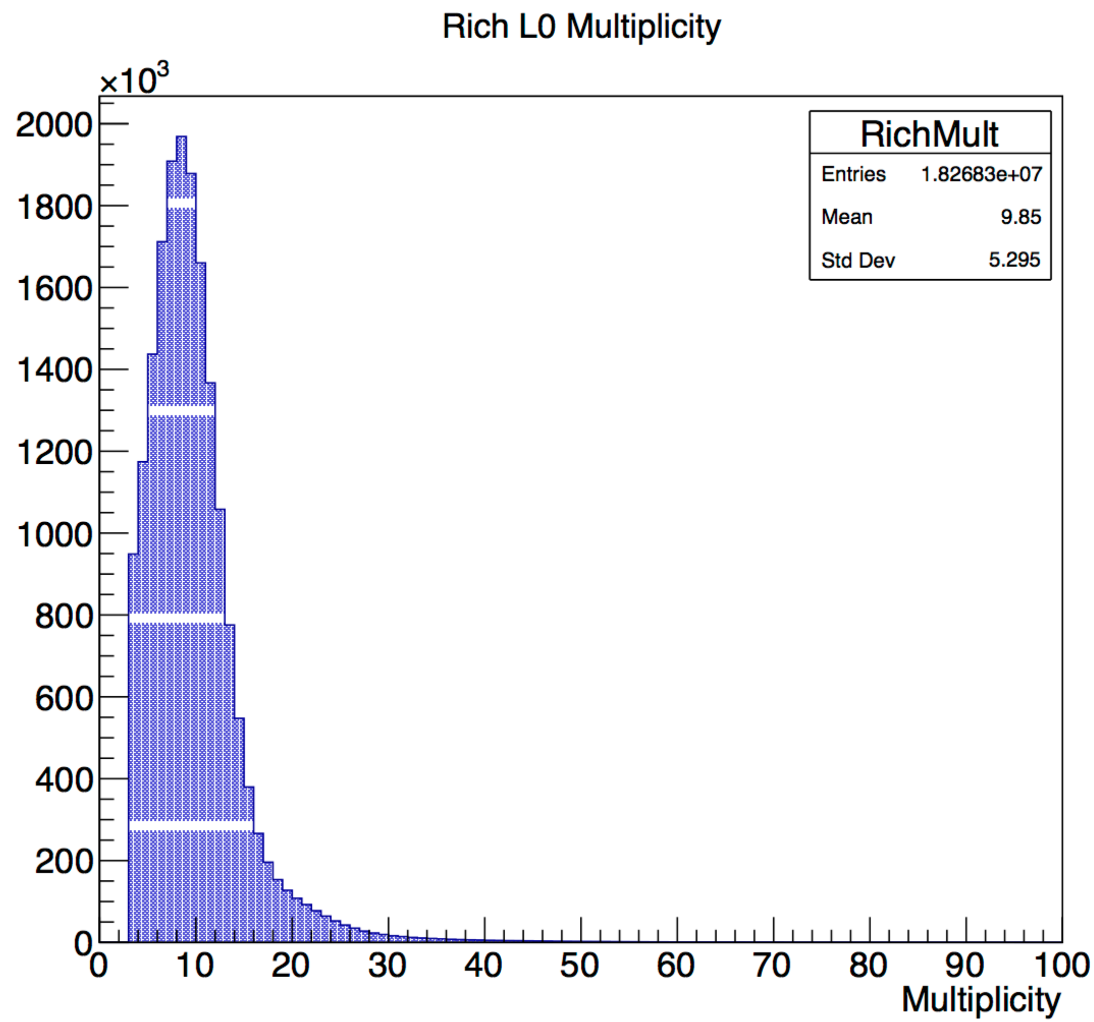
Daisy-chain Heavy Neutrino trigger architecture

RICH L0 multiplicity



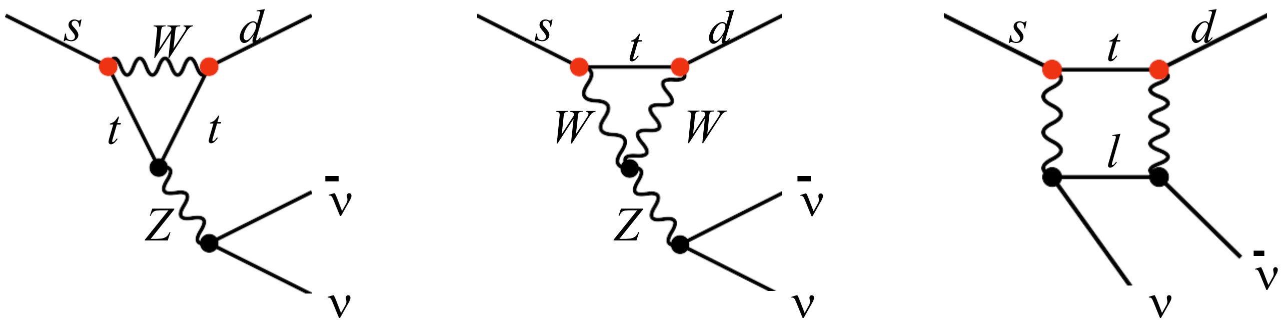
RICH L0 multiplicity





Ultra rare kaon-decays

$K^+ \rightarrow \pi^+ \nu\nu$: theoretically pure and almost experimentally unexplored



Decay	Branching Ratio ($\times 10^{11}$)				
	Theory (SM)			Experiment	
$K^+ \longrightarrow \pi^+ \nu\bar{\nu}$	9.11	0.72	[1]	17.3 +11.5 -10.5	[2]
$K^0 \longrightarrow \pi^0 \nu\bar{\nu}$	3.00	±0.30	[1]	<2600 (90%CL)	[3]

NA62 goal is to measure this BR to 10% precision

in addition: $|V_{td}|$ to ≤ 10% accuracy

[1] **A. J. Buras, D. Buttazzo, J. Girrbach-Noe and R. Knegjens,** *arXiv:1503.02693*

[2] **A. V. Artamonov *et al.* (E949 Collaboration)**
B. Phys.Rev.Lett.101, 191802, 2008.

[3] **J. K. Ahn *et al.* (E391a Collaboration)** PR D81 (2010)072004

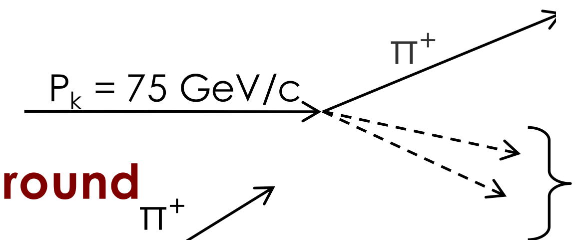
These processes are very sensitive probes for new physics:

- They are highly suppressed
- They are predicted with very high accuracy

In-flight kaon decay at 75 GeV/c

- **Large missing momentum**

Kinematic Signature

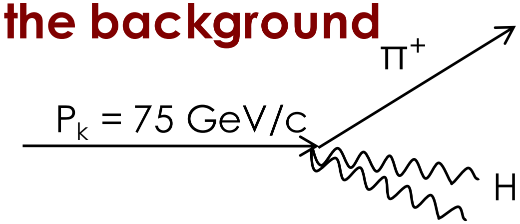


Require:
P_π= 15 - 35 GeV/c

Neutrinos
carry ≥ 40 GeV/c momentum

Consequence for the background

$K \to \pi^+ \pi^0$
$\hookrightarrow \gamma\gamma$



- **Cuts on the missing mass**

$$m^2_{miss} = (P_K - P_\pi)^2$$

defines 2 regions where the signal is not dominated by background

Rejects ≈ 92% of Kaon decays