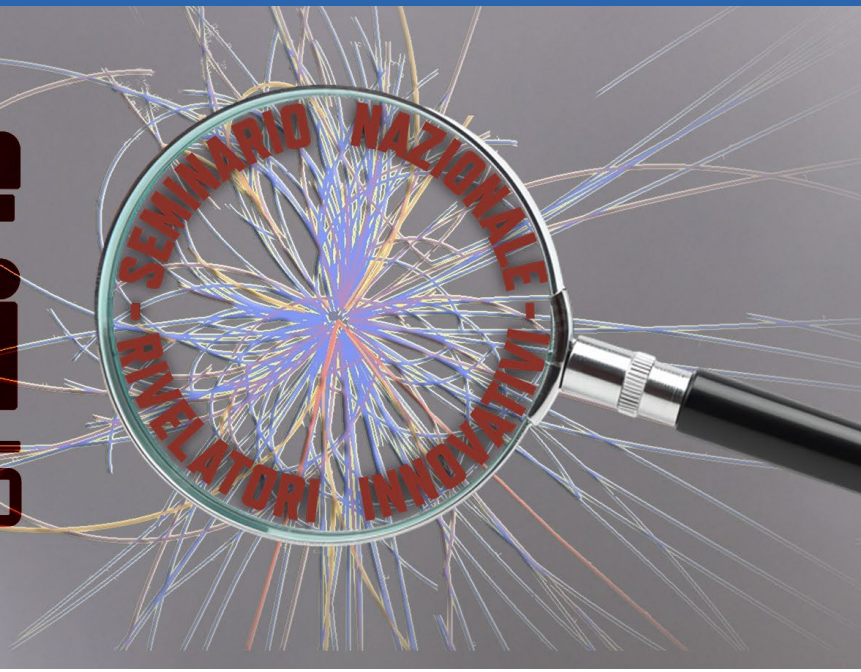
Finanziato dall'Unione europea NextGenerationEU
Ministero dell'Università e della Ricerca
Italiadomani PIANO NAZIONALE DI RIPRESA E RESILIENZA
FAIR Future Artificial Intelligence Research

VIII Seminario Nazionale Rivelatori Innovativi

SNRI25 - Genova, 24-28 novembre 2025



STREAM-AI

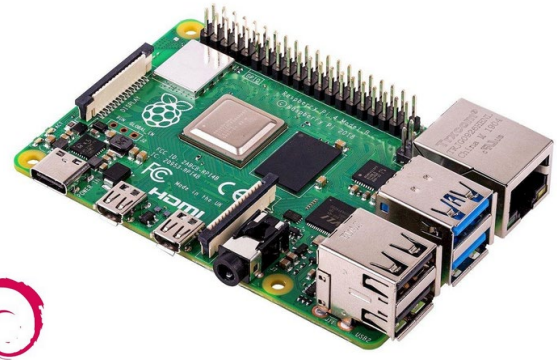
A streaming Read-Out data acquisition framework for HEP detectors

Marco Battaglieri battaglieri@ge.infn.it

Fabio Rossi farossi@ge.infn.it

STREAM-AI: Goals of the Lab

- General concept of Streaming Read-Out
  - Architecture
  - Functionality
  - Bottleneck
- Required nodes in a SRO system
  - Front-end
  - Data reduction
  - Frame Routing
  - High level analysis
- Functionality of each nodes
  - Characteristics
  - Algorithm



Raspbian

TensorFlow

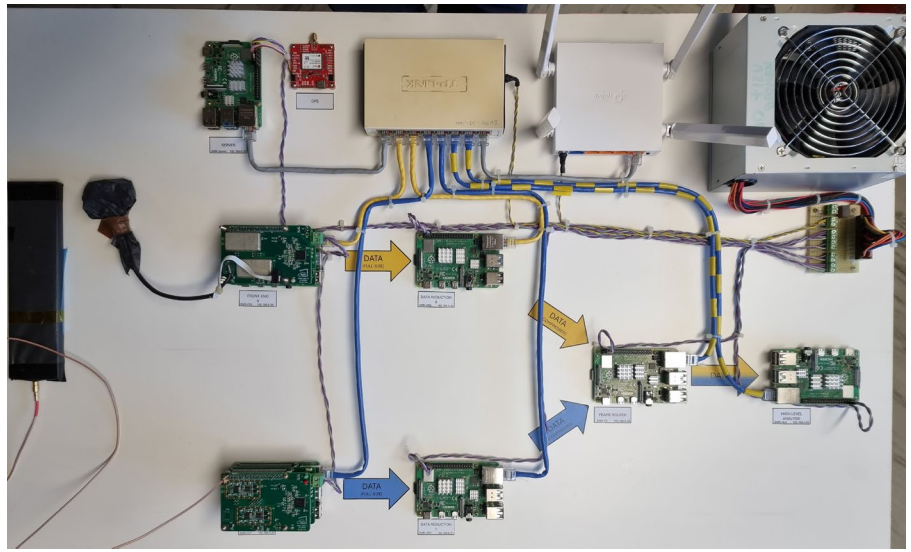
Keras

python™

General concept of Streaming Read-Out in a manageable environment

Finanziato dall'Unione europea NextGenerationEU
Ministero dell'Università e della Ricerca
Italiadomani PIANO NAZIONALE DI RIPRESA E RESILIENZA
FAIR Future Artificial Intelligence Research

OVERVIEW



Raspberry

- Reduced performance
- In scale system

HARDWARE

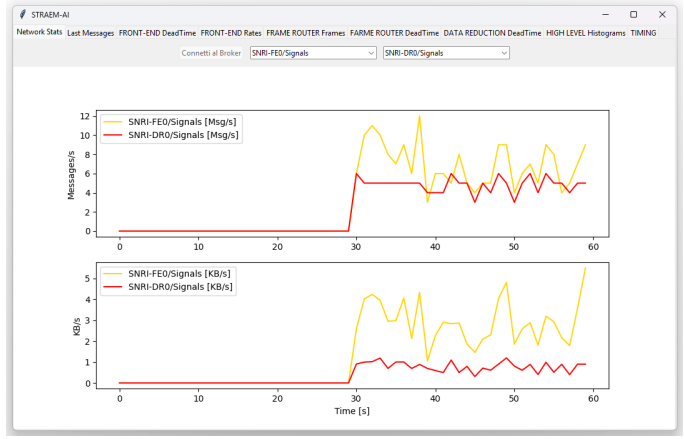
SOFTWARE

Useful for Class, Outreach and Research

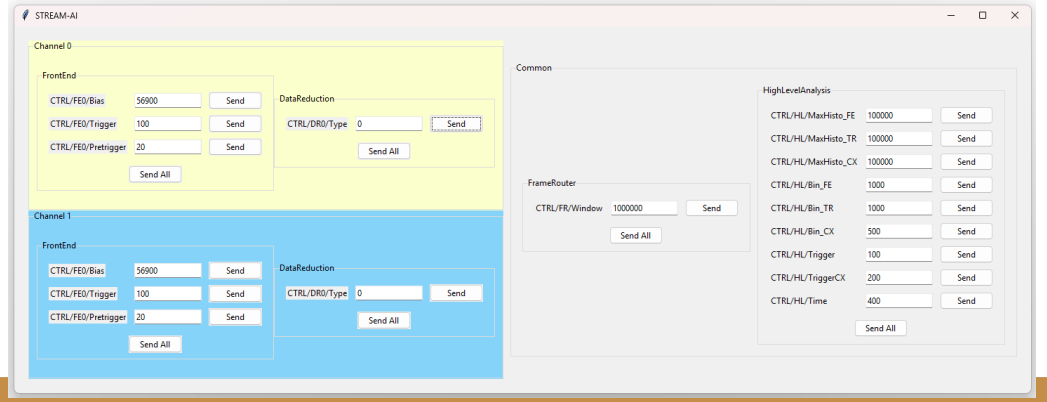
- Understand general SRO behavior
- Easy testbench for new ideas
- Study bottleneck

Reduced performance but not in functionality

Monitoring Interface



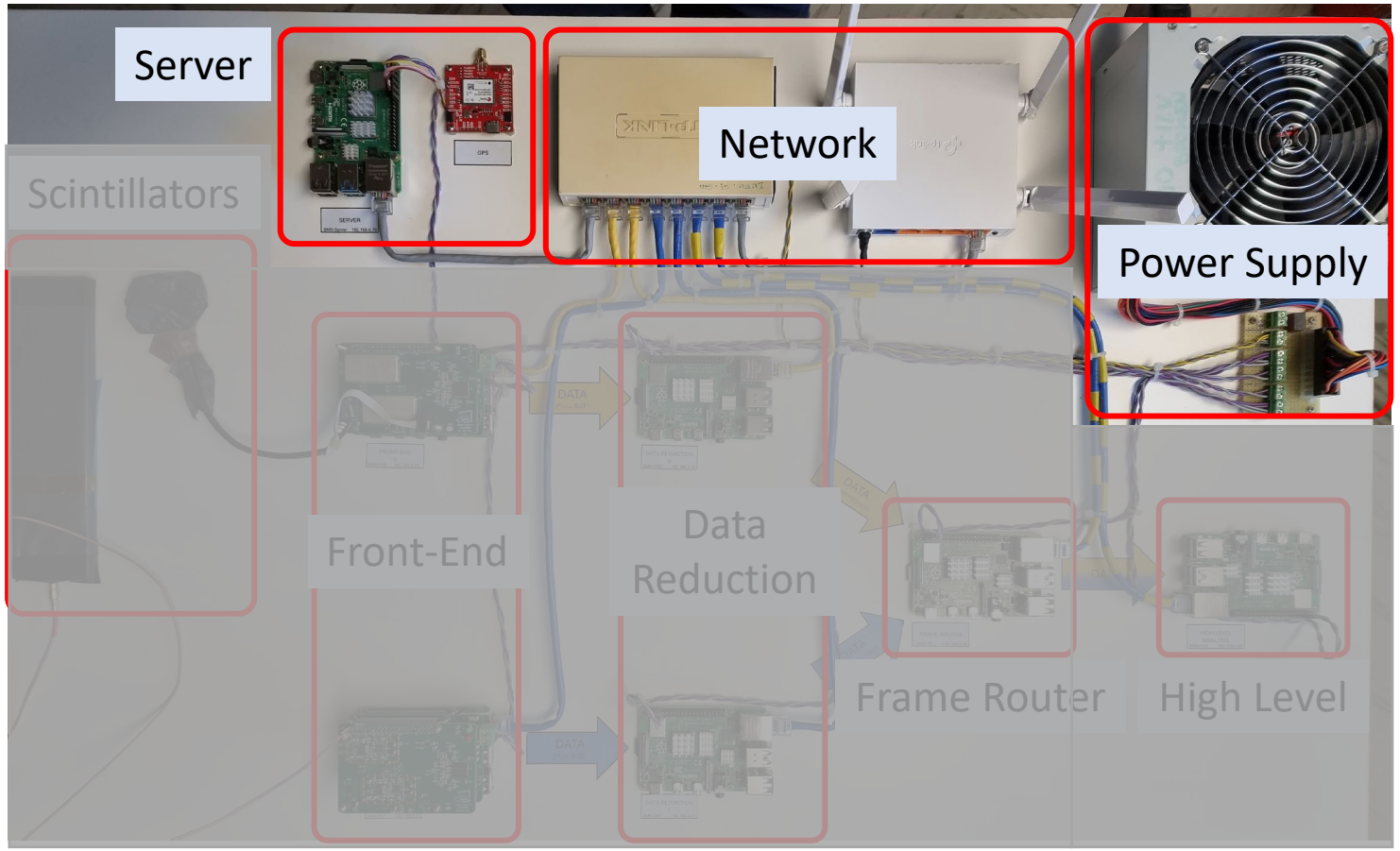
Control Interface



HARDWARE: Auxiliary

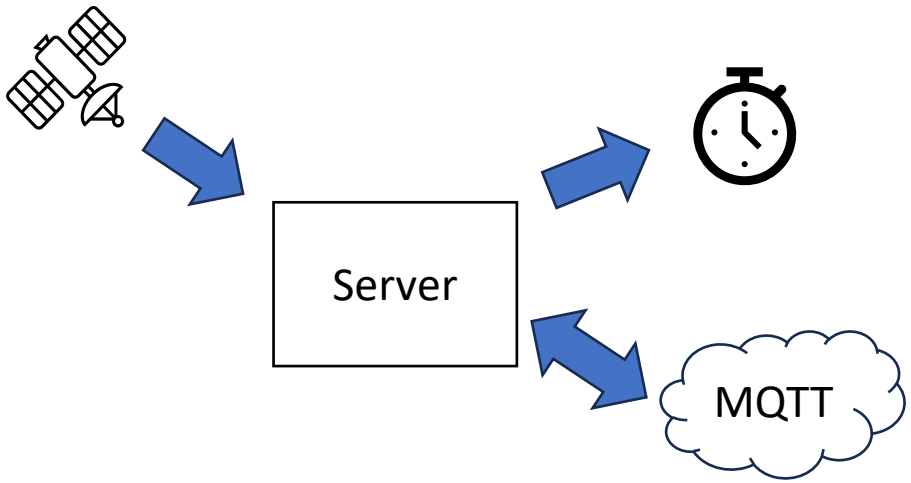
- Server
  - Timing NTP (chrony)
  - Global time with GPS
  - Broker MQTT
- Network
  - Star connection
  - Possible to access each nodes easily
  - Wi-Fi or Ethernet
- Power Supply
  - 5V, 12V, 9V

Synchronization, global timing and networking

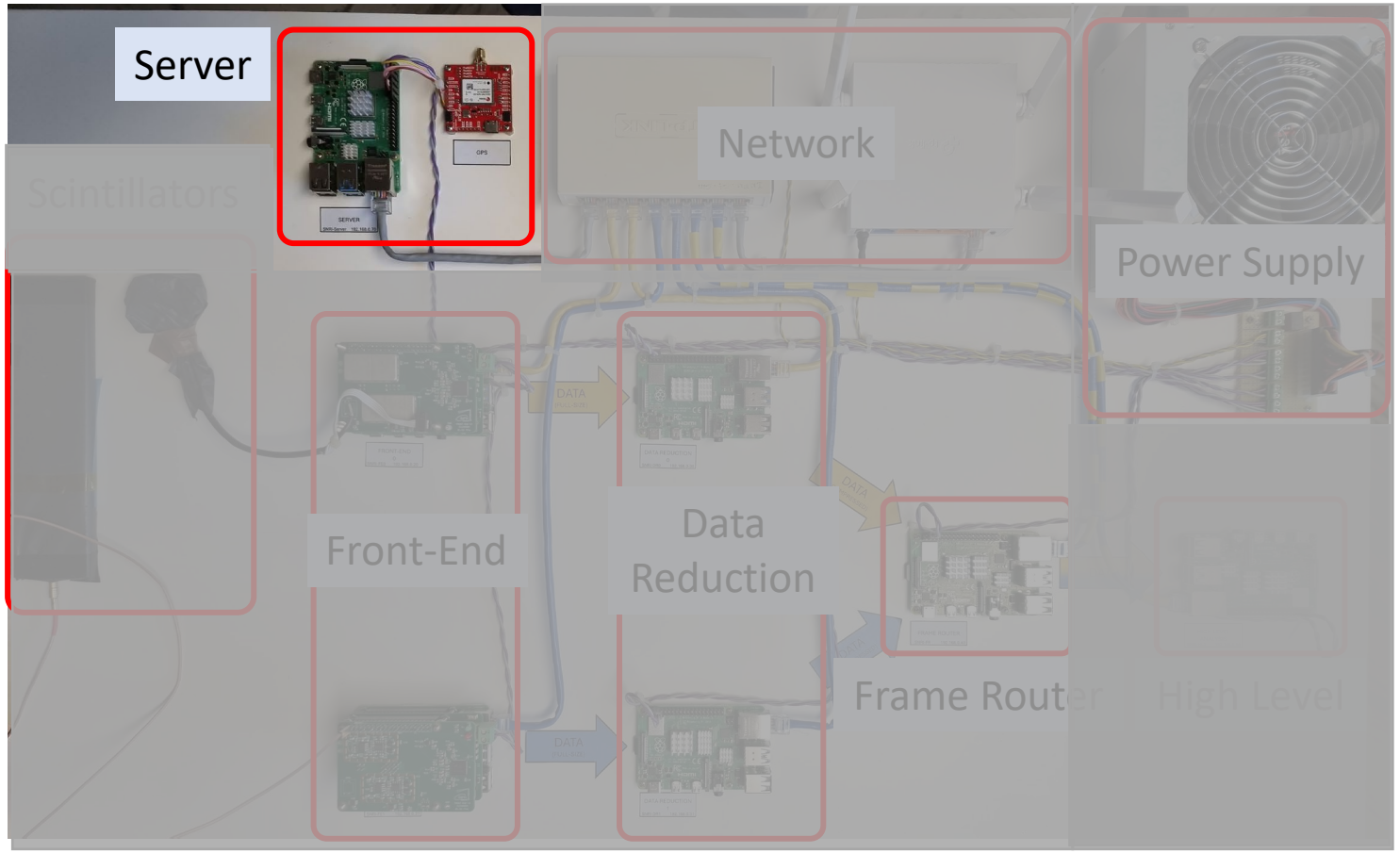


SERVER

- MQTT Broker
- NTP Server
- GPS Global Time (not in this implementation)



Synchronization, global timing and networking



SERVER: Check Synchronization on STREAM-AI Nodes

Monitoring synchronization

chronyc tracking

```
pi@SNRI-HL0:~/Desktop/SNRI/Codice $ chronyc tracking
Reference ID    : C0A80046 (192.168.0.70)
Stratum         : 11
Ref time (UTC)  : Wed May 14 21:32:23 2025
System time     : 0.000000000 seconds fast of NTP time
Last offset     : +0.000001274 seconds
RMS offset      : 0.000006710 seconds
Frequency       : 6.188 ppm fast
Residual freq   : -0.020 ppm
Skew            : 0.090 ppm
Root delay      : 0.000119269 seconds
Root dispersion : 0.000036483 seconds
Update interval : 64.4 seconds
Leap status     : Normal
```

Network jitter	chrony
10 µs	35 ± 8
100 µs	109 ± 14
1.0 ms	475 ± 93
10.0 ms	1603 ± 447

chrony performances on small LAN

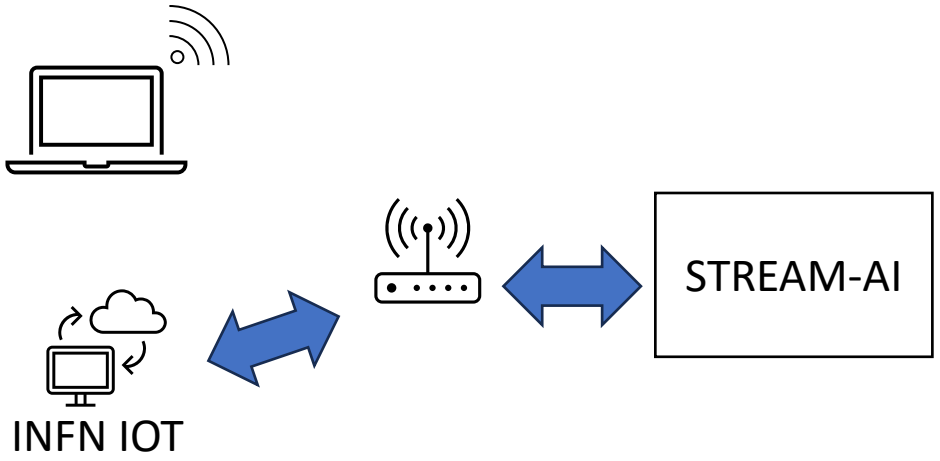
Restart and manual sync

- sudo systemctl restart chrony
  Restart the Chrony Service
- sudo chronyc -a makestep
  Sinchronize

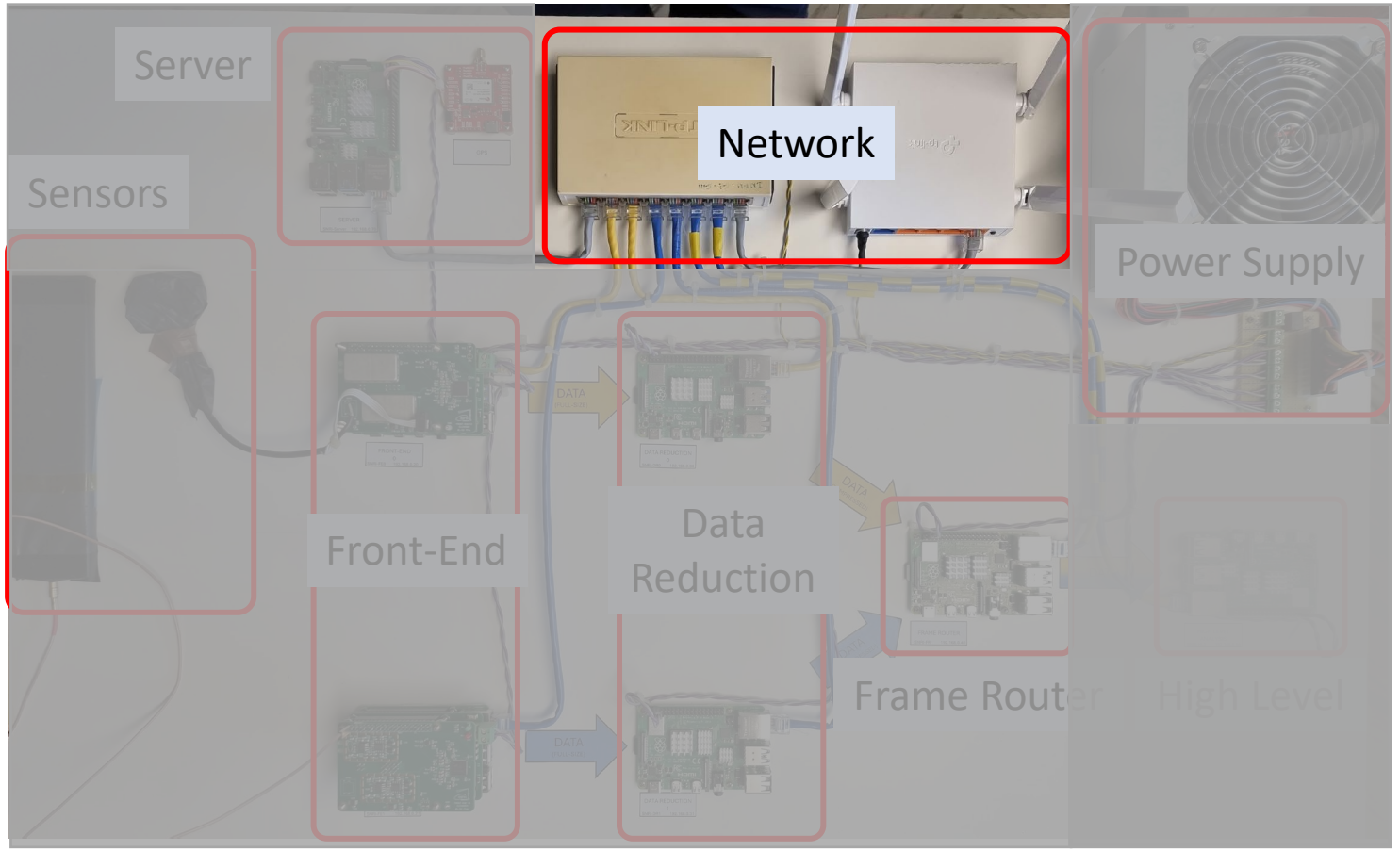
https://chrony-project.org/index.html

NETWORK

- LAN for STREAM-AI Interconnection
- WiFi for Laptop Control and Monitoring
- Ethernet connection to INFN IoT Net

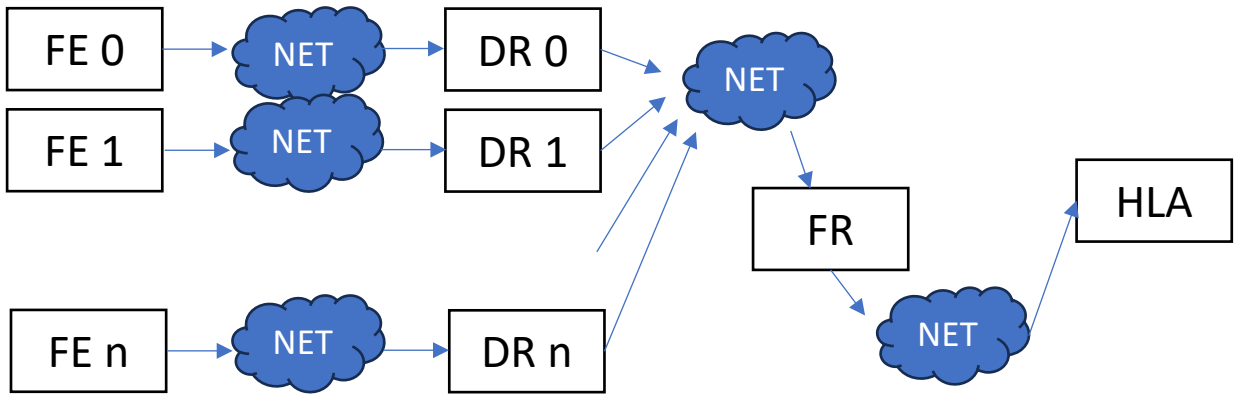


Lan, WiFi and IoT infrastructure connection



NETWORK: topology details

Real Streaming Read-Out system

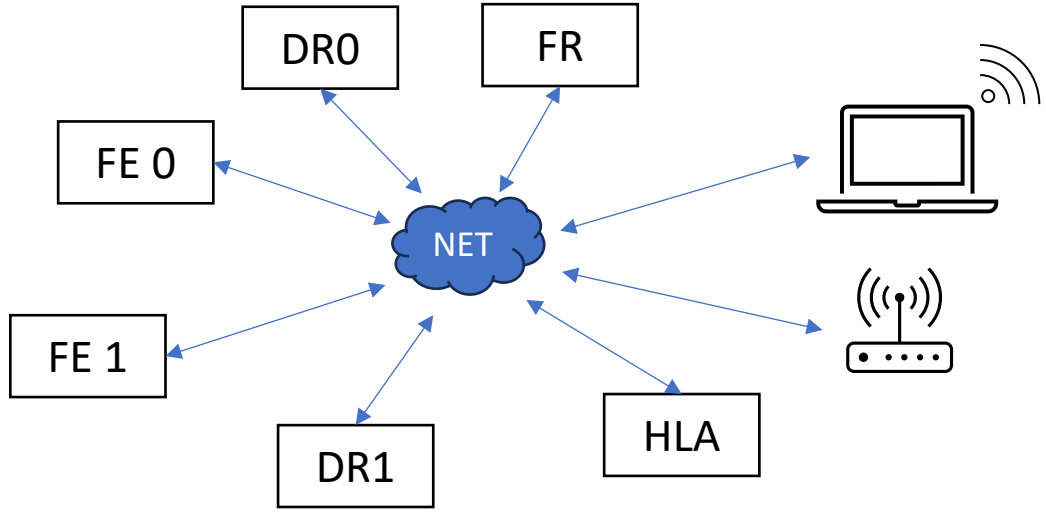


- **High data rate**

Each connection has point to point connection or dedicated net to avoid bottleneck. Multiple net could be used between DR and FR as well as between FR and HLA.

- **Optimized for speed**

STREAM-AI Solution



- **Low data rate**

Star connection allow a simple use of MQTT to access all nodes from others. Each node can receive data from each other.

- **Optimized for understand and debug**

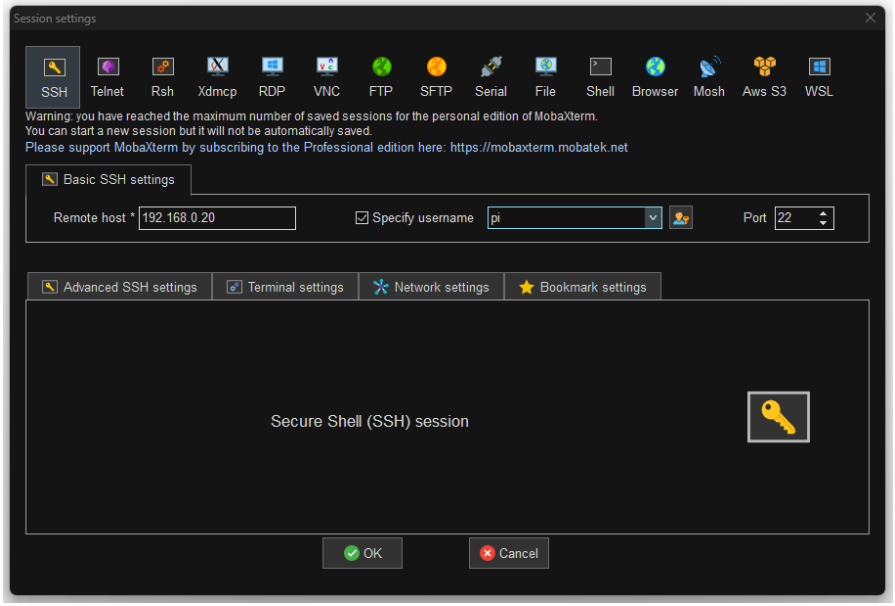
Main difference with respect to a real SRO – No impact on functionality

NETWORK: Addresses, Hostname and PW

Addresses

Host	Hostname	Username	Pw	IP Address
Front end 0	SNRI-FE0	pi	Snr12025	192.168.0.20
Front end 1	SNRI-FE1			192.168.0.21
Data Reduction 0	SNRI-DR0			192.168.0.30
Data Reduction 1	SNRI-DR1			192.168.0.31
Frame Router	SNRI-FR			192.168.0.40
High Level Analysis	SNRI-HL0			192.168.0.50
Server	SNRI-Server			192.168.0.70

MobaXterm Interface



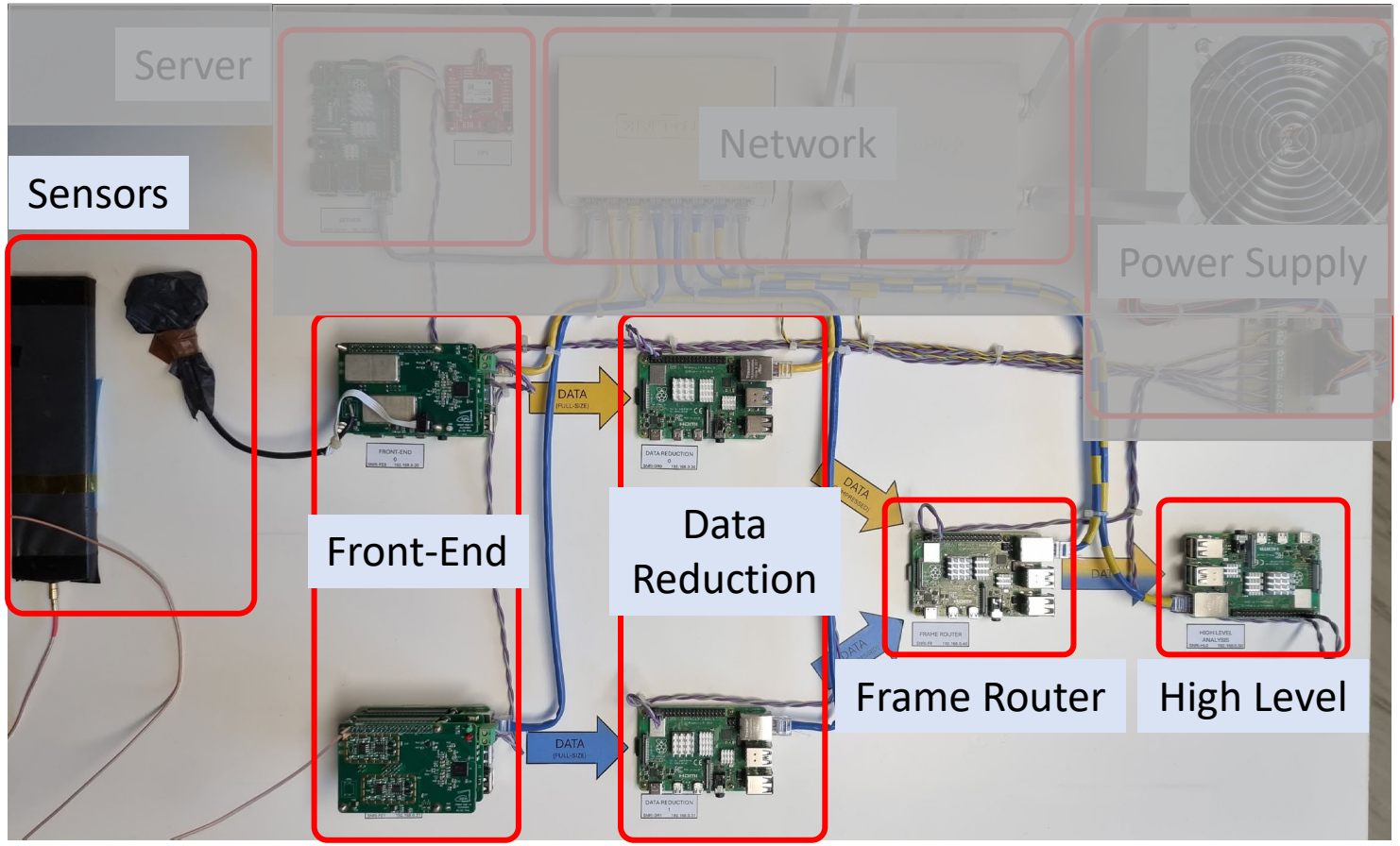
Project Folder:

cd **Desktop/SNRI/Codice**

DATA PATH

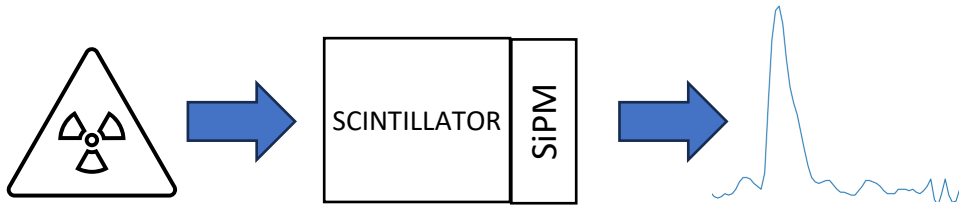
- Sensors
  - Sensitive to gamma radiation
- Front-End
  - 10MS/s 12 bit 2 channels
  - Bias 24 to 81 Vdc
  - Voltage, current or charge preamplifier
- Data Reduction
  - Autoencoder
  - GZIP
- Frame Router
  - Standard implementation
- High Level
  - High level trigger: check for coincidence

All nodes are fully customizable with C code

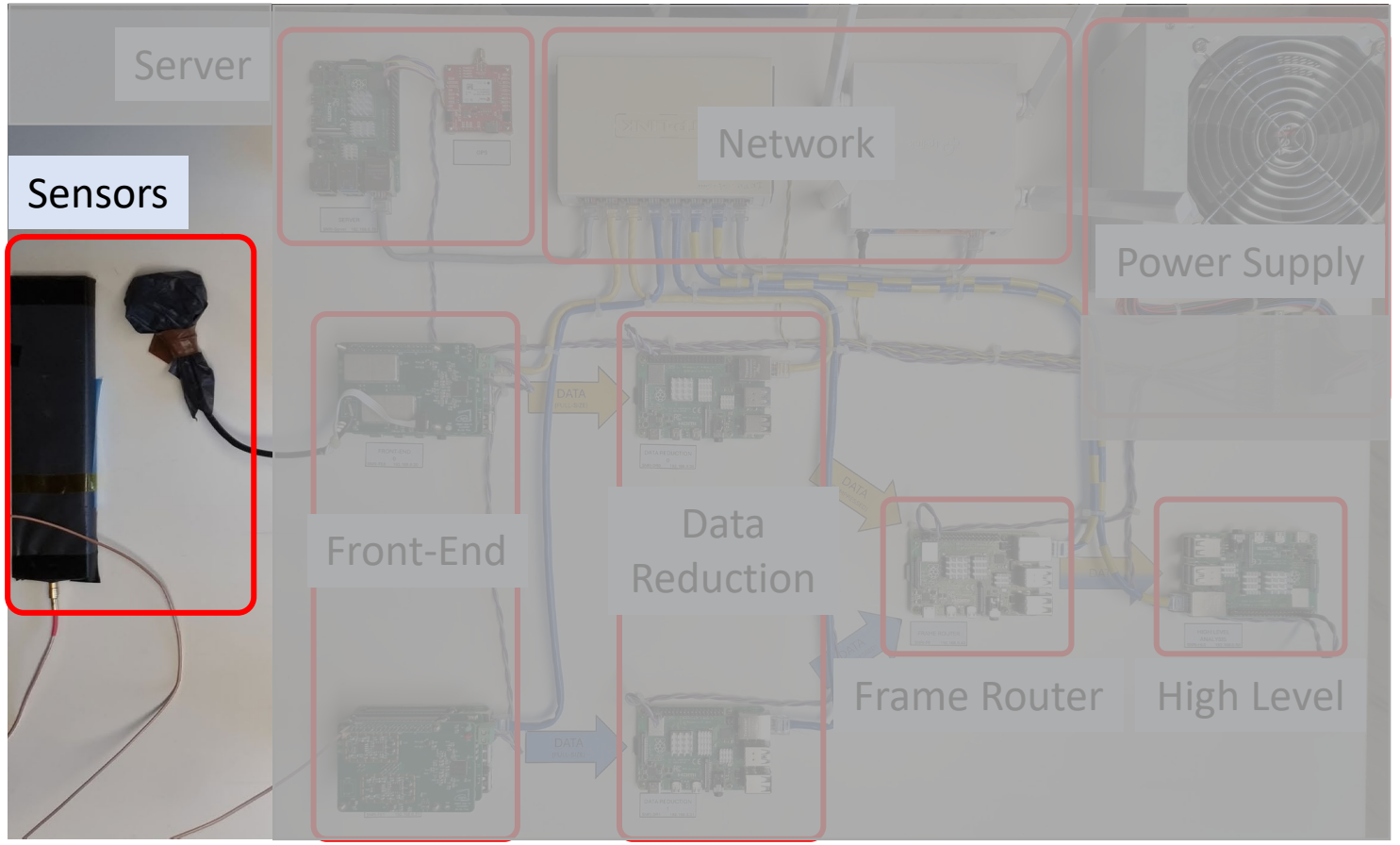


SENSORS

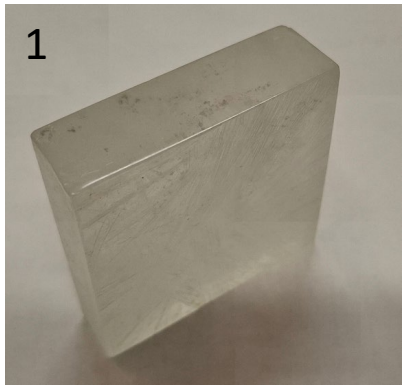
- Sensors
  - Different type of scintillator:
    - CsI(Tl)
    - BGO
  - SiPM (Silicon Photomultiplier)



Converts radiation in electrical pulses

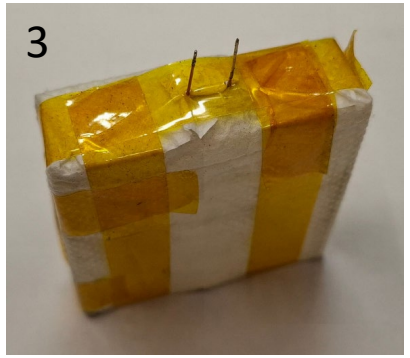


SENSORS: Details



CsI(Tl) Cristal

Cristal + SiPM



Reflective wrapping

Light protecting wrapping

Two scintillators are arranged one above the other to detect cosmic-ray coincidences.

CsI(Tl)

photons/MeV:	≈ 50’000
Decay Time:	≈ 1µs

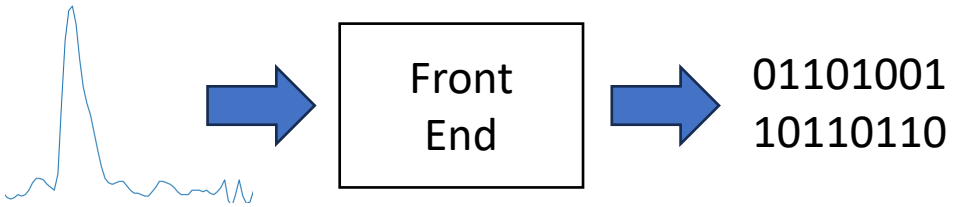
SiPM (S13360-6025CS)

Vop:	56.95V
Id:	0.286µA

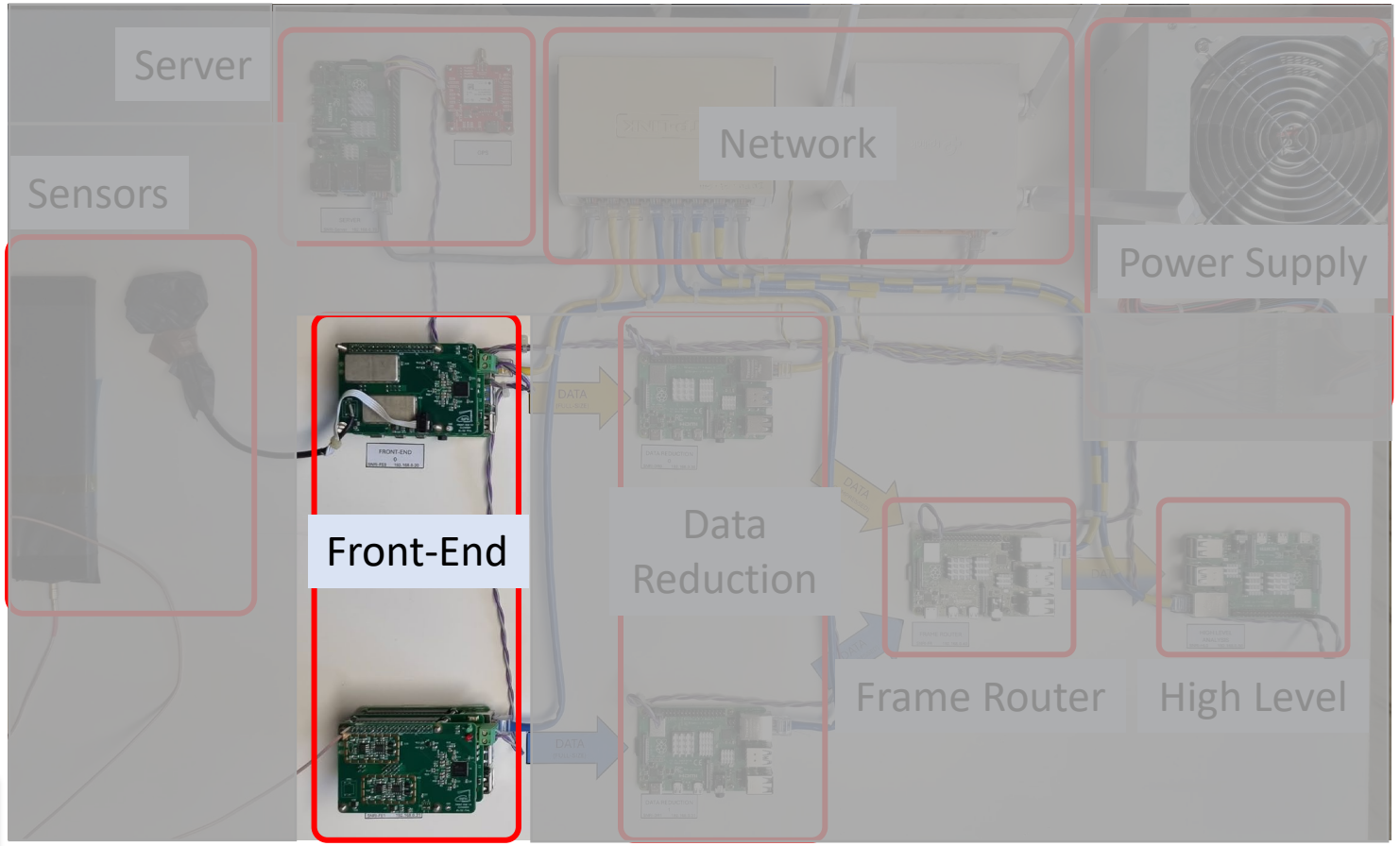
CRISTAL: convert gamma radiation to photons
SIPM: convert photons to electrical pulse

FRONT-END

- Signal conditioning
- Analog to Digital conversion
- Zero suppression
- Stream data



Signal conditioning, sampling and quantization

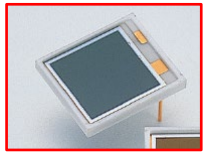


Finanziato dall'Unione europea NextGenerationEU
Ministero dell'Università e della Ricerca
Italiadomani PIANO NAZIONALE DI RIPRESA E RESILIENZA
FAIR Future Artificial Intelligence Research

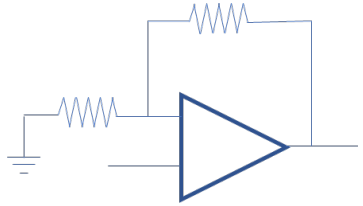
FRONT-END: Overview

HARDWARE

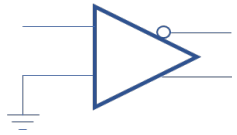
SiPM



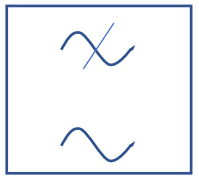
Voltage amplifier



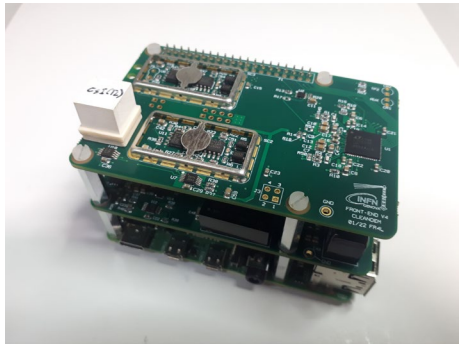
Single Ended to Differential



LP Filter



ADC



Hardware: Real Time

SOFTWARE

NET

Stream Data

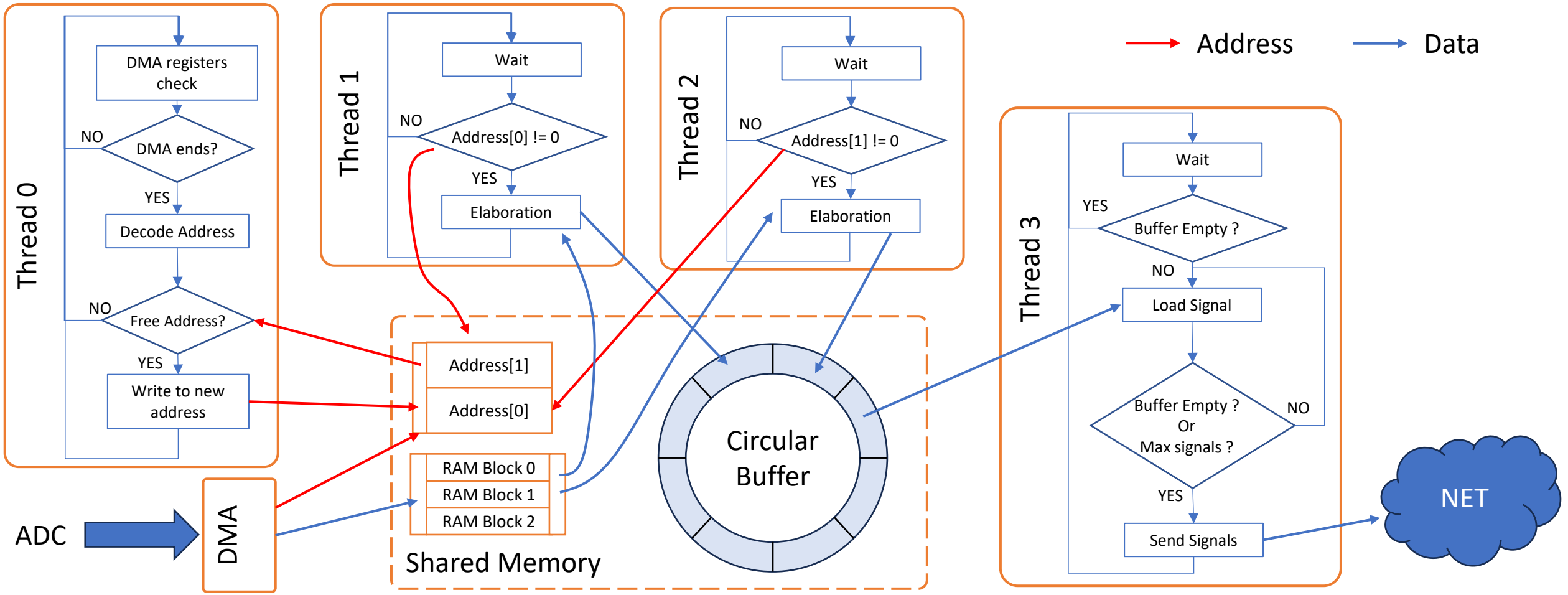
Zero suppression

Load Data

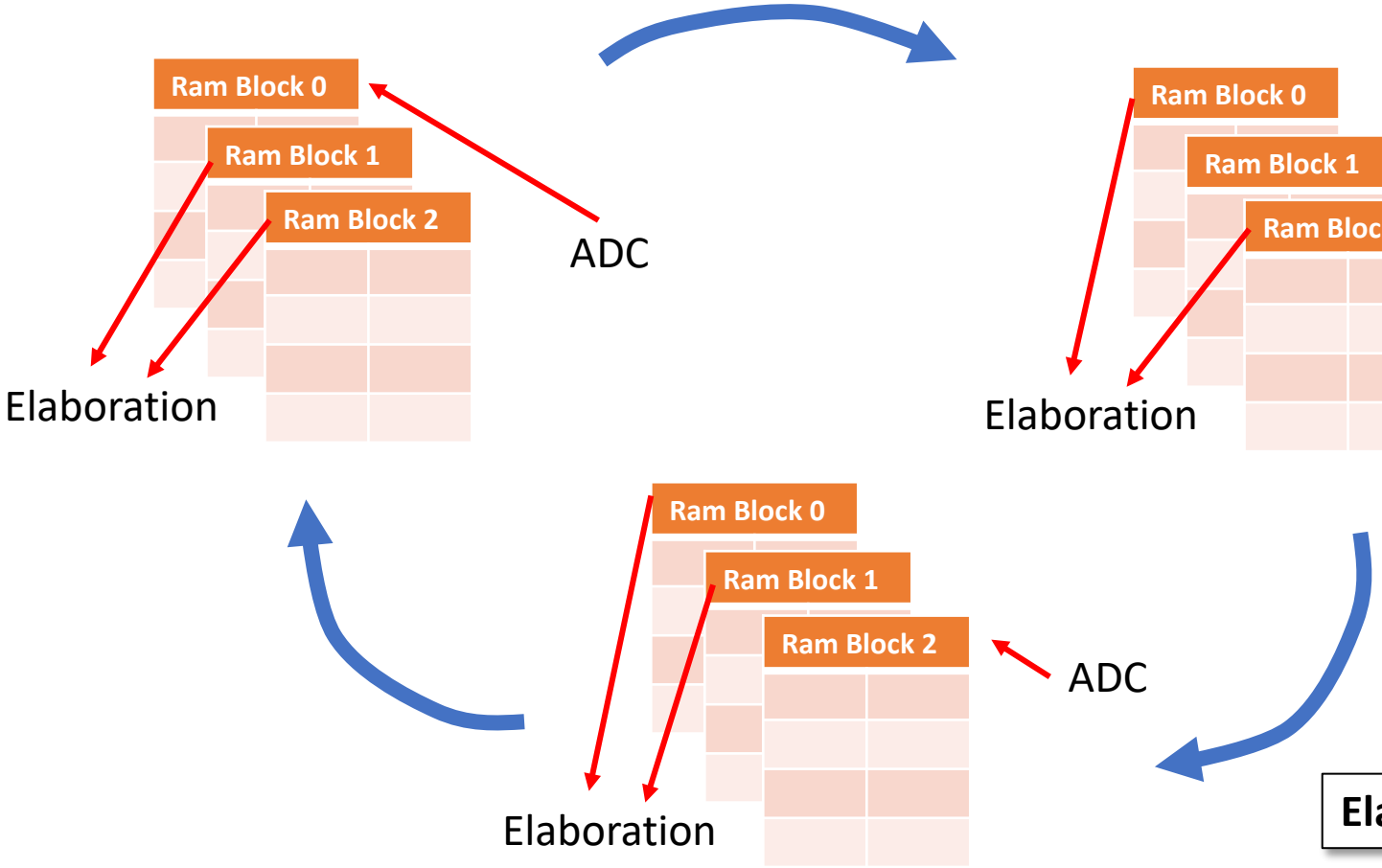
Raspbian

Software: Non-Real Time Operating System

FRONT-END: Details of algorithm



FRONT-END: Streaming Algorithm

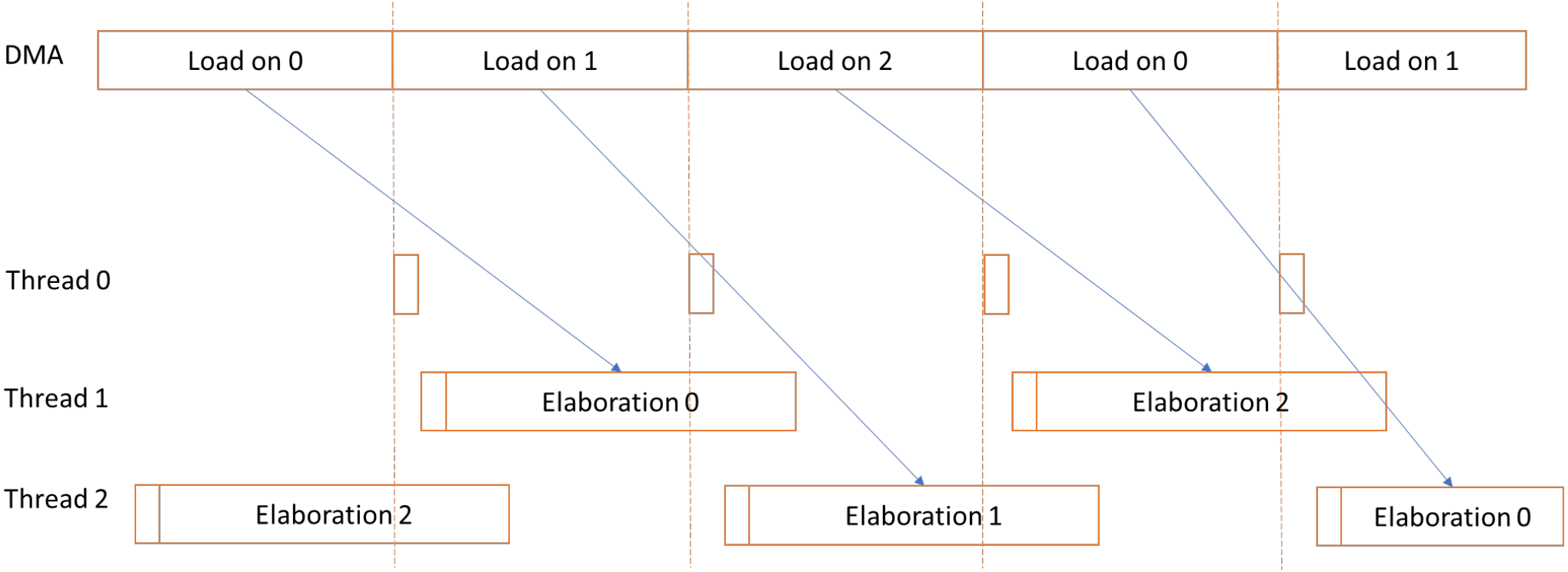


DMA write one block while Elaboration goes in parallel on the other two.

- Read samples from ADC in Real Time
- Constant sampling time
- No missing data
- Elaboration could be non real time

Elaboration could be executed on non RTOS system

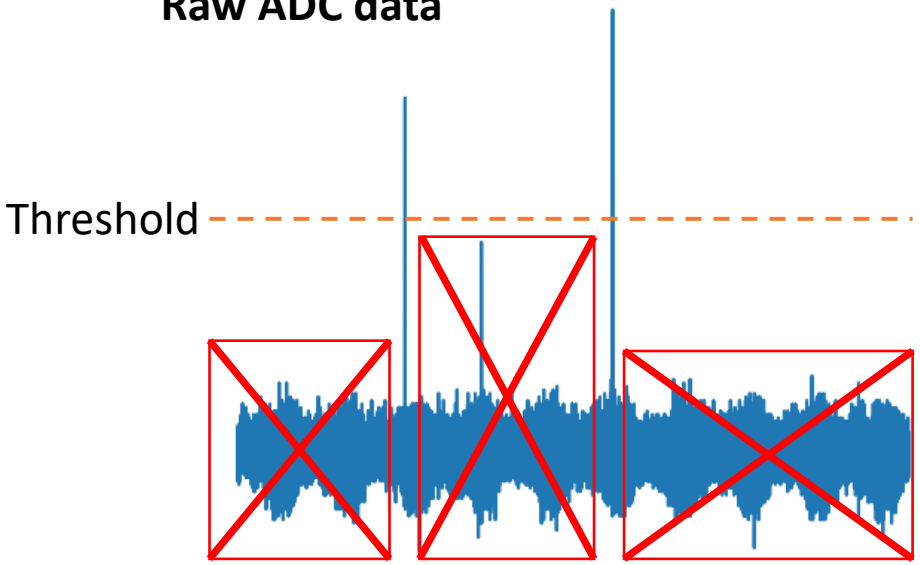
FRONT-END: Streaming Algorithm



Elaboration time is double respect to DMA write

FRONT-END: Zero Suppression

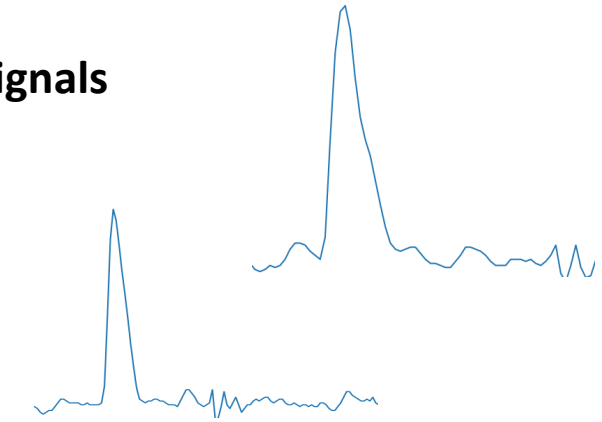
Raw ADC data



100ms: 1,5 Mbyte
Rate: 10MS/s
All the samples

Zero suppression

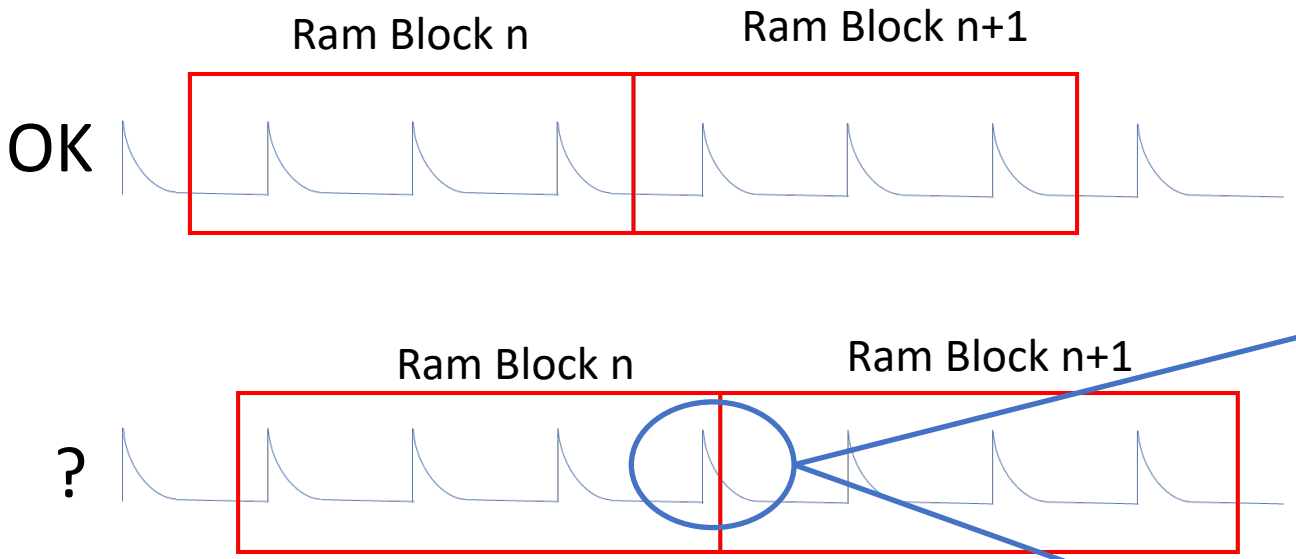
Raw Signals

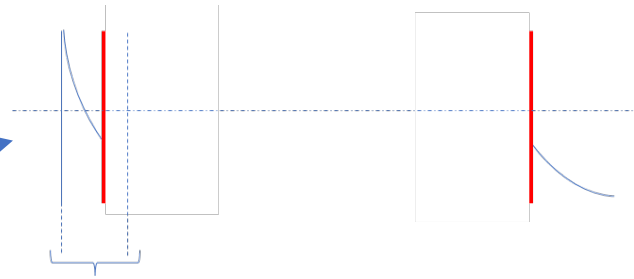


100ms: 150 Byte
Rate: gamma dependent (<< 10MS/s e.g. 100Hz)
Samples related to signals

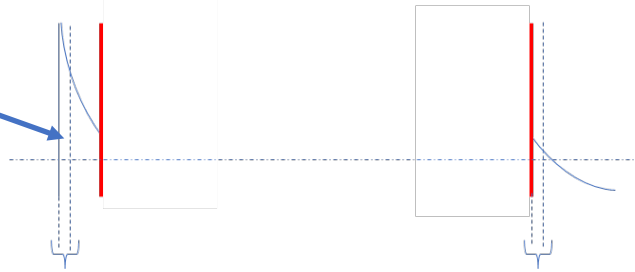
Keep only samples related to a signal, discard baseline.

FRONT-END: Issues of Algorithm





Num Events OK
Wrong integral



Wrong Number
Wrong integral

DMA Block Size
1MSample

ADC Frequency
10MS/s

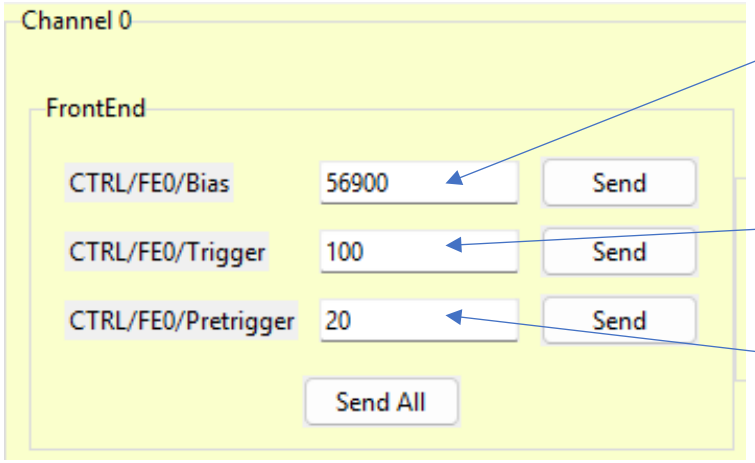
Discontinuities every
100ms

Neglected in current implementation

FRONT-END: Run streaming acquisition

sudo chrt –f 99 ./SNRI_main

Start code on the Front End with scheduling FIFO and highest priority



SiPM Bias Voltage [mV]

Modify carefully, high voltage can damage the SiPM

Trigger level [bit]

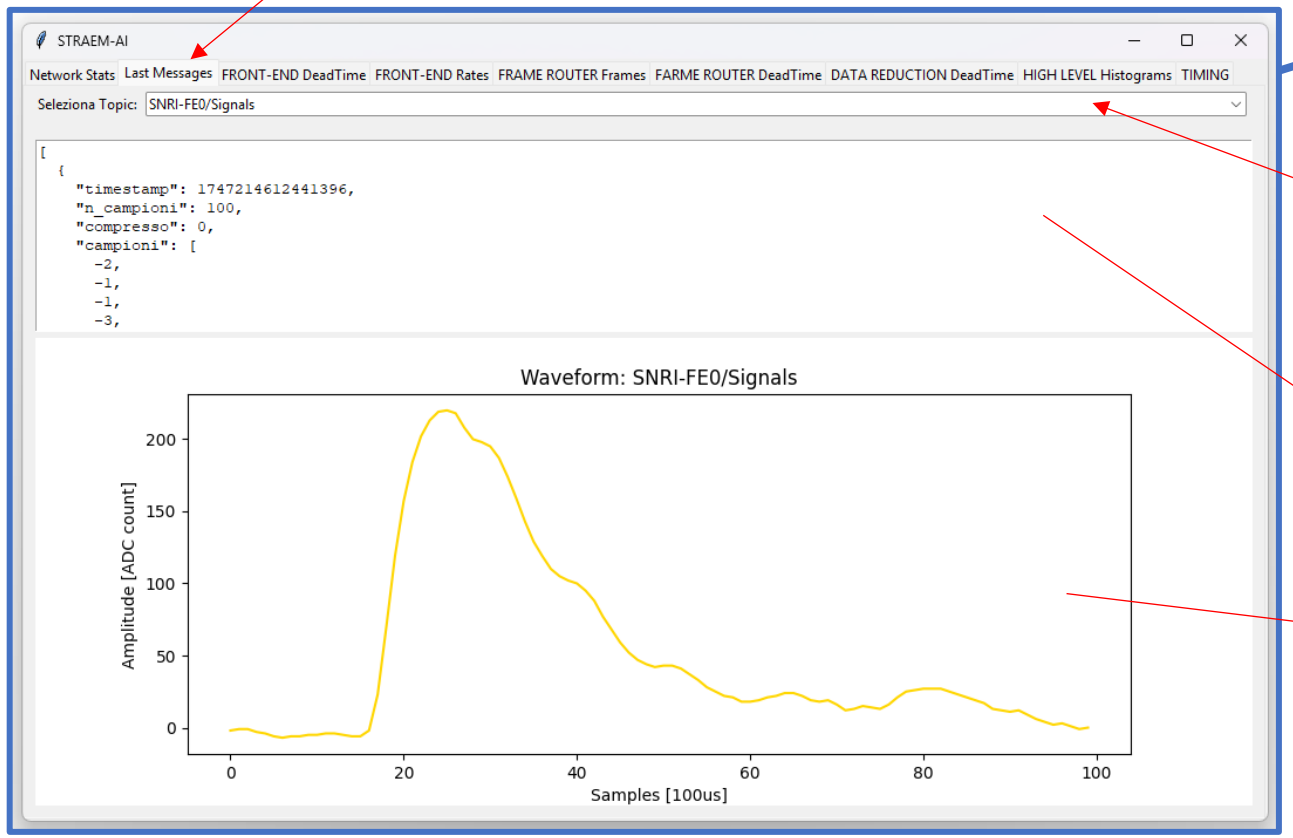
Pretrigger [samples]

```
pi@SNRI-FE0:~/Desktop/SNRI/Codice $ sudo chrt -f 99 ./SNRI_main
Sono core 0, Faccio l'acquisizione
Acquisizione: Inizializzazione
Sono core 1, Gestisco l'MQTT
Parametri: ok
Acquisition paramteres: ok
Calibration: ok
Acquisizione> Inizializzazione Board
Board Temperature = 31°C
SiPM Temperature = 31°C
SiPM Bias Voltage = 57.052V
Acquisizione: in funzione! 0
Prove: 1
Prova 0 di 1
-----------------------------------------
S       Events  Rate    Elab.   Persi   Volt
1.00    7       7.00    8       0       57.05
2.00    13      6.00    18      0       57.05
3.00    20      7.00    28      0       57.05
4.00    22      2.00    38      0       57.05
5.00    29      7.00    48      0       57.05
6.00    35      6.00    58      0       57.05
7.00    43      8.00    68      0       57.05
8.00    46      3.00    78      0       57.05
9.00    56      10.00   88      0       57.05
```

s	Events	Rate	Elab.	Persi	Volt
Elapsed time	Total number of acquired events	Instantaneous rate	Total number of elaborated block (100ms each)	Total number of lost block	Bias voltage of the SiPM

FRONT-END: Expected signals

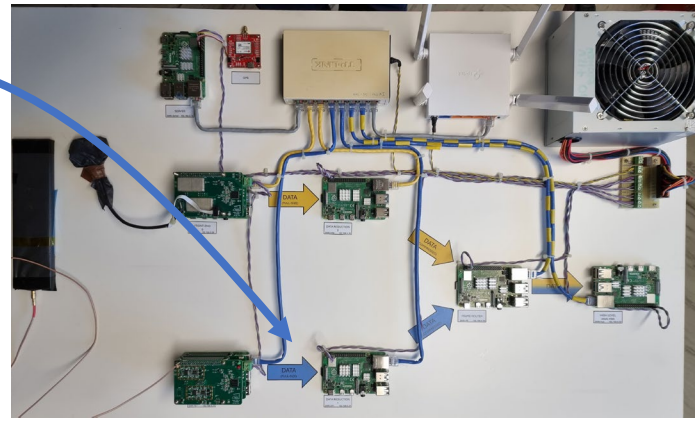
Last Message



Topic to monitor

Message content

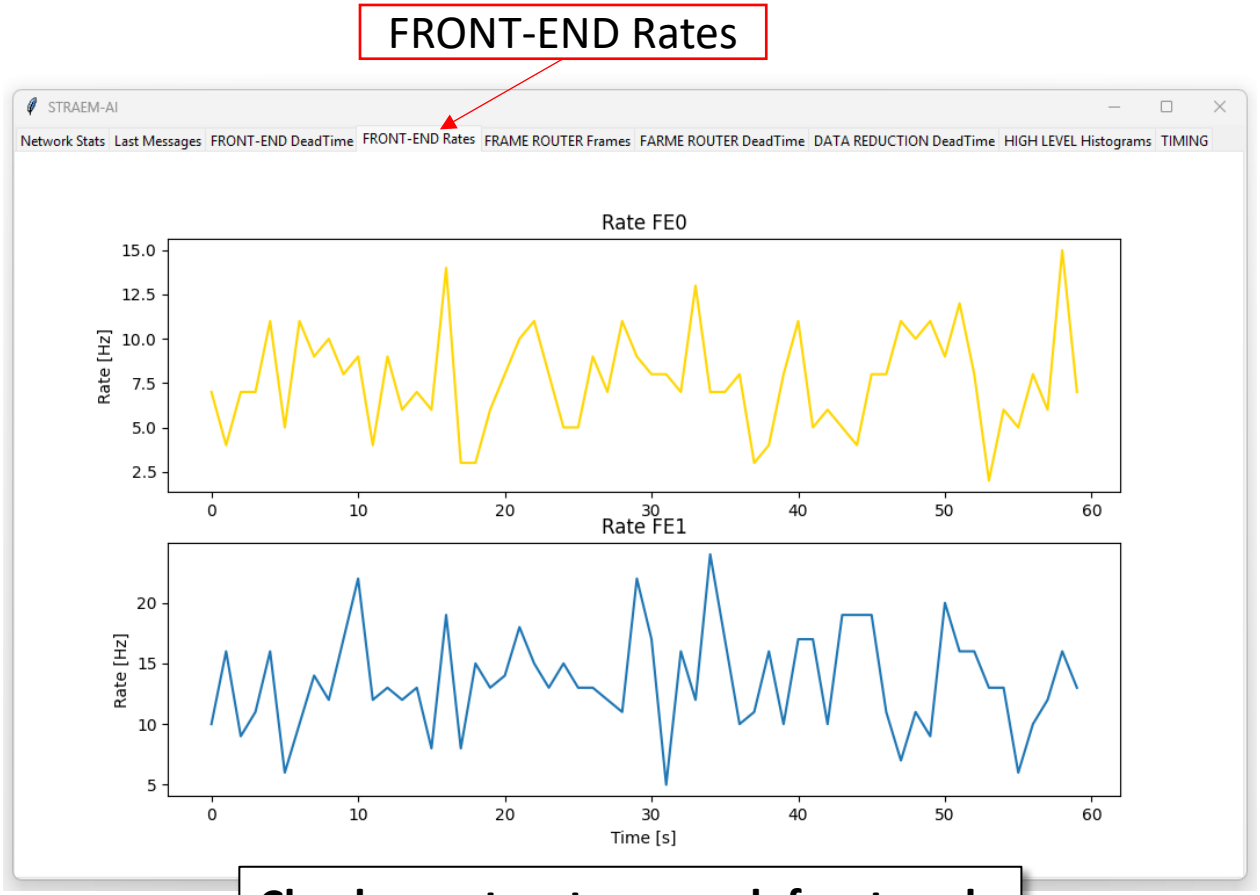
Display waveform



This tab works also with other nodes and display the waveform if there is any in message.

View single MQTT message: single acquired signals

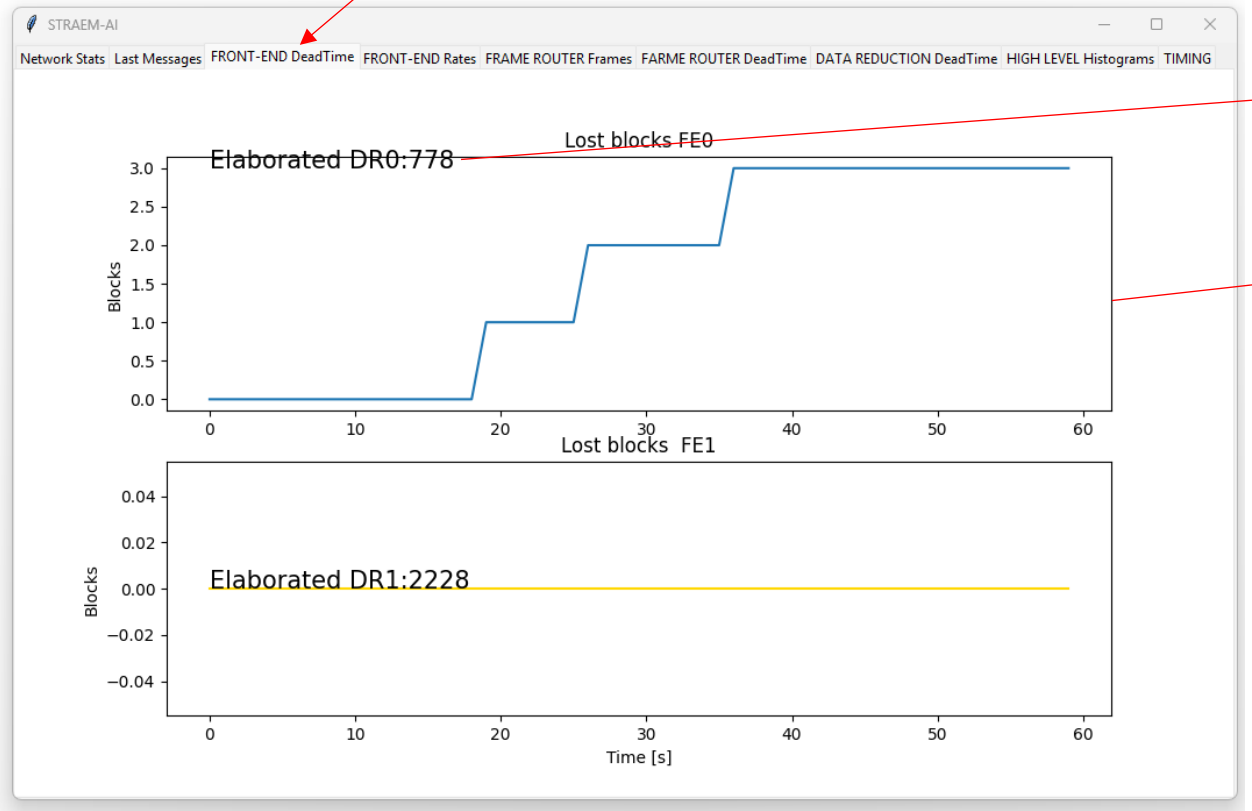
FRONT-END Rates



Check events rate on each front-ends

FRONT-END Dead-Time

FRONT-END DeadTime



Total number of elaborated blocks

Lost Blocks

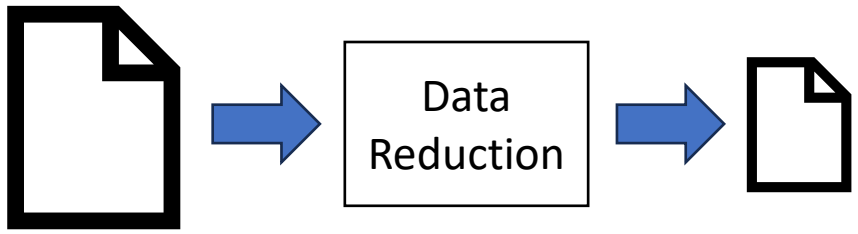
Tab show number of lost block

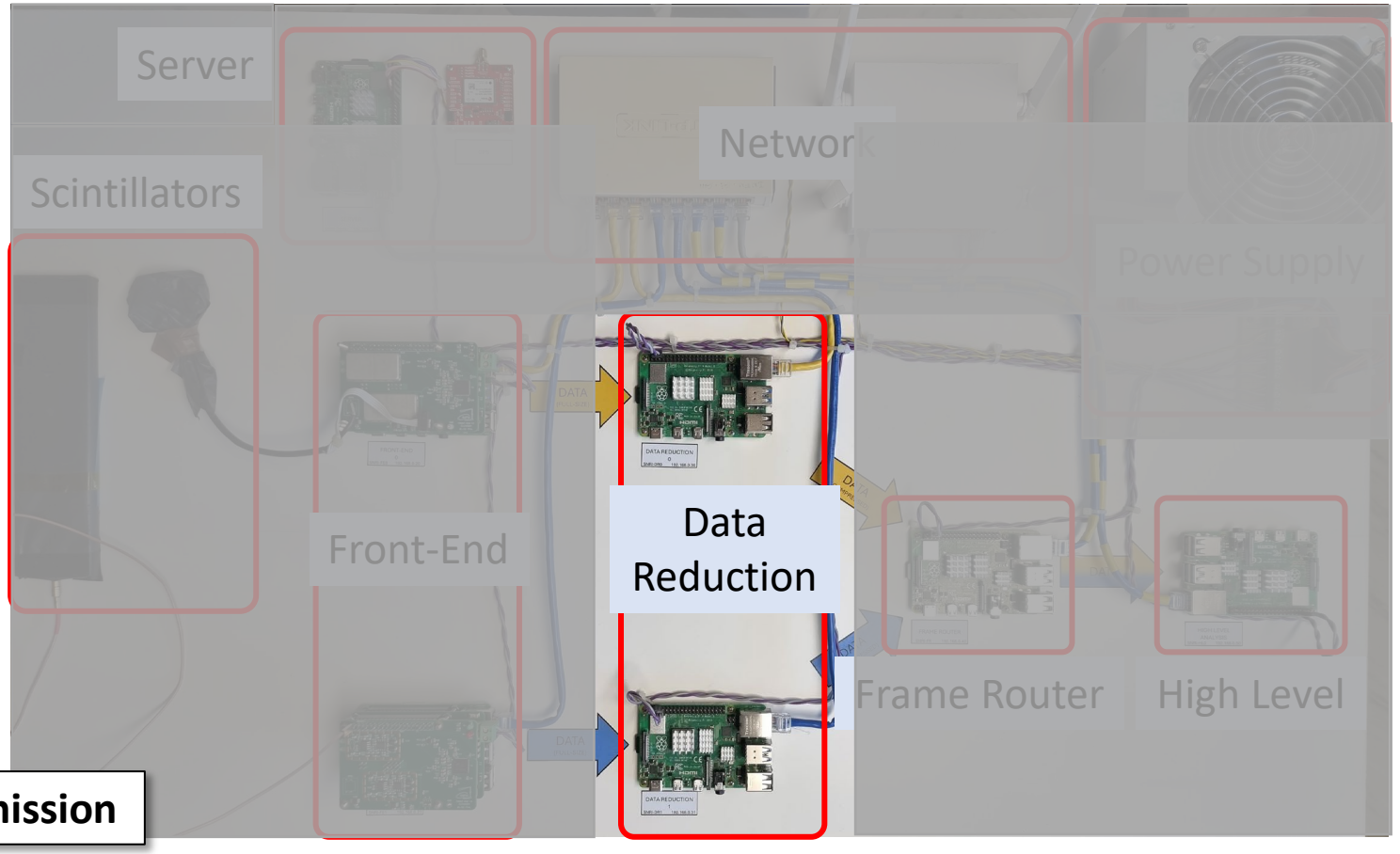
Each block contains 100ms of acquisition

Monitoring Dead Time from each Front-End

DATA REDUCTION

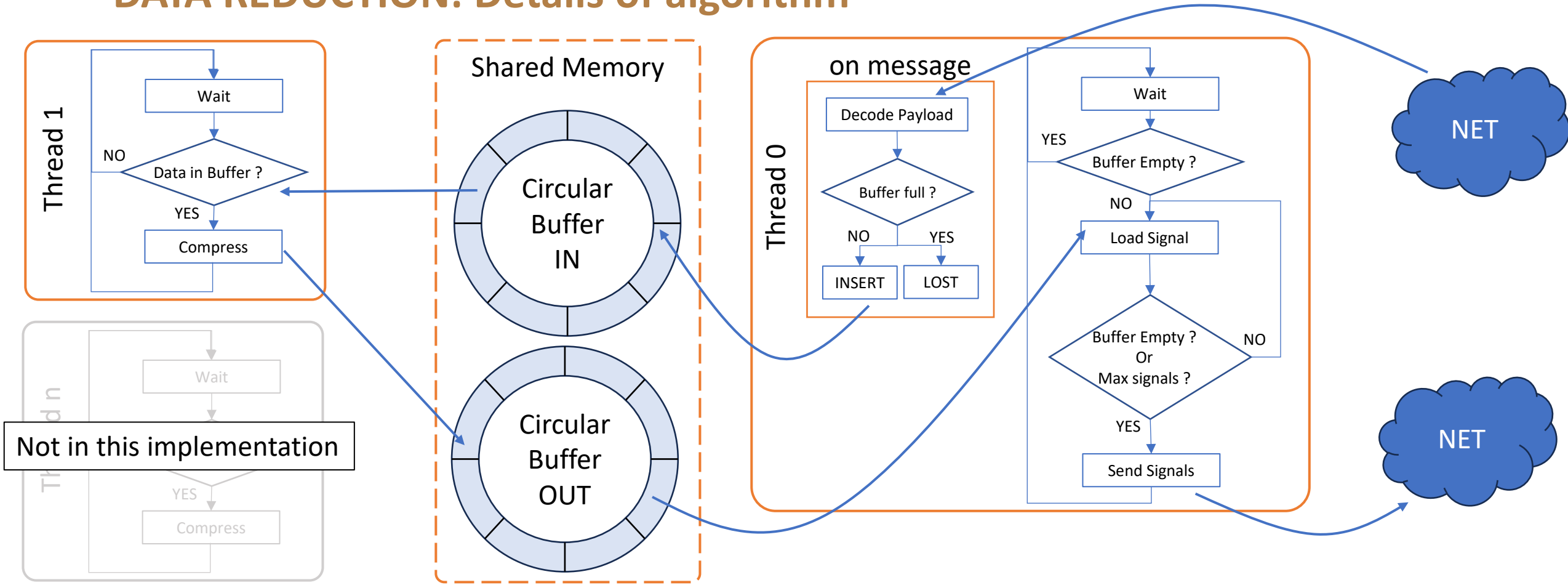
- Reduce bandwidth
- Avoid network bottleneck





Reduce the bandwidth required for signal transmission

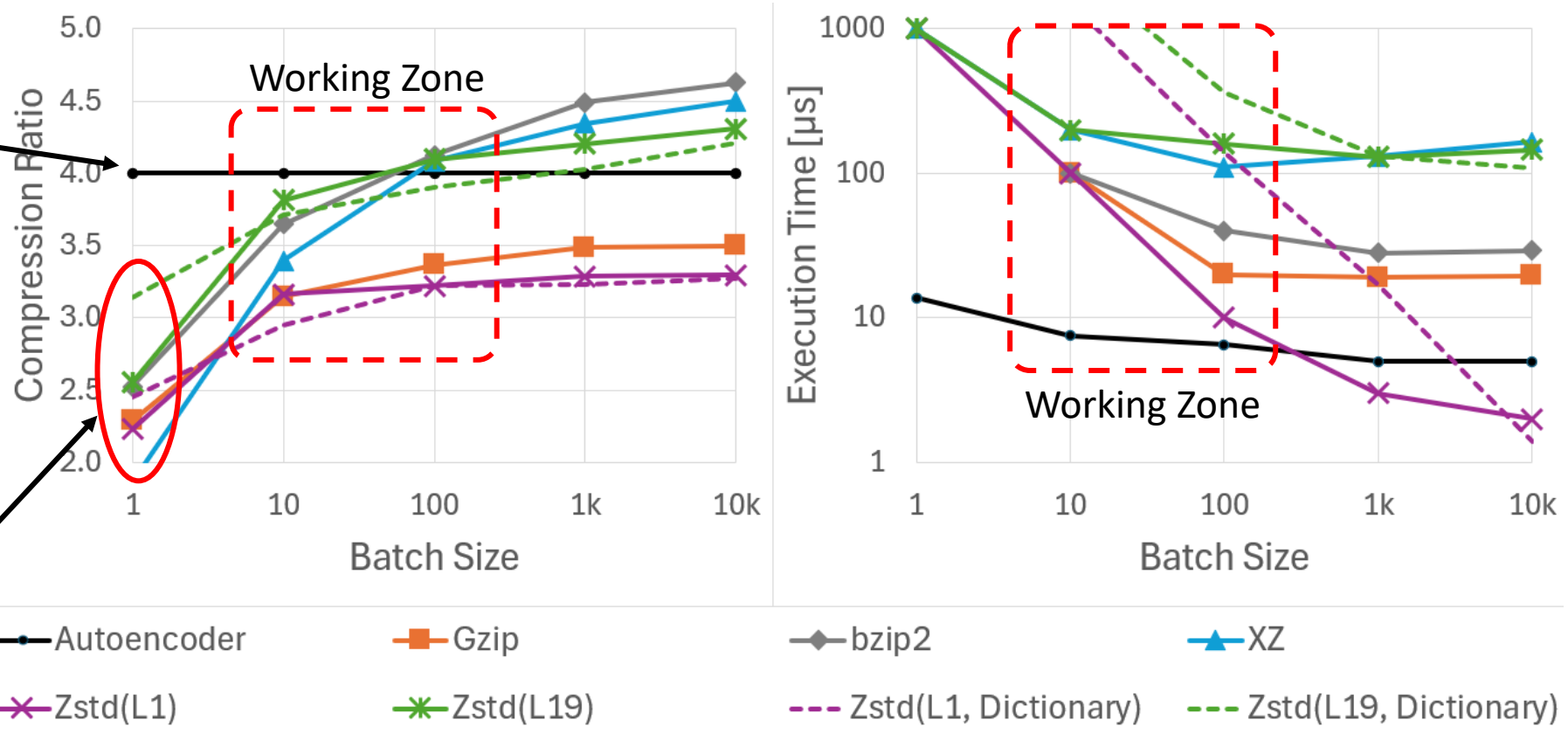
DATA REDUCTION: Details of algorithm



DATA REDUCTION: Choose the compression algorithm

Autoencoder compression ratio is a Parameter

Standard compression ratio depends on signals number



DATA REDUCTION: Lossy compression - Autoencoder

Machine Learning Algorithm

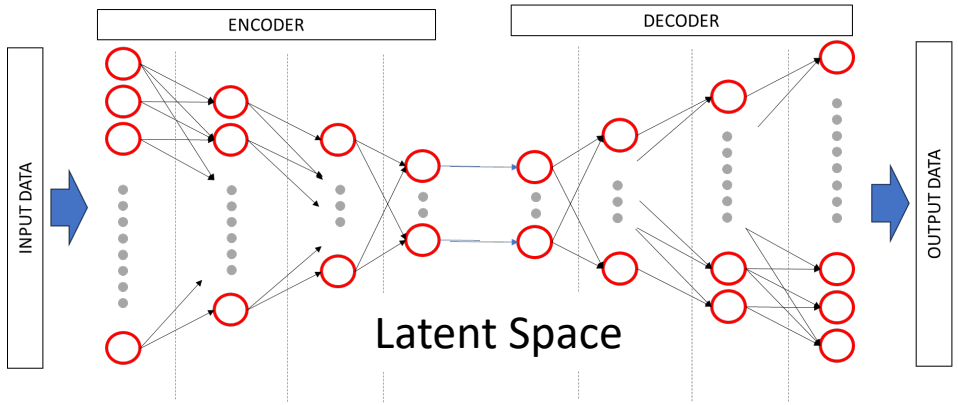
Dimensionality reduction

Unsupervised learning

Artificial Neural Network

Composed of two function:
- encoding
- decoding

FULLY CONNECTED AUTOENCODER WITH DENSE LAYER



python™ TensorFlow Keras

Lossy compression algorithm

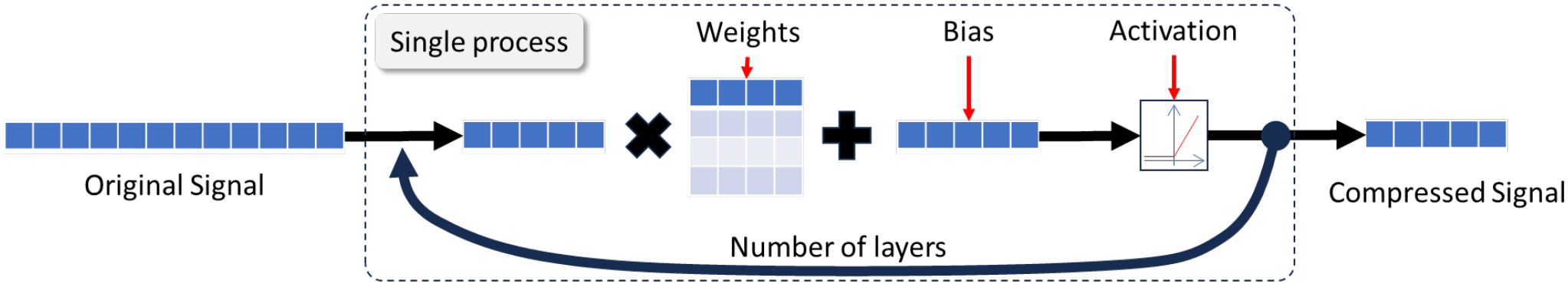
Rossi F., Battaglieri M., Ragusa E., Gastaldo P., Gavalian G.
"Artificial intelligence data reduction algorithm for streaming readout in high energy physics experiment"
The 2024 World Congress in Computer Science, Computer Engineering, and Applied Computing https://doi.org/10.1007/978-3-031-85856-7_5

DATA REDUCTION: C code implementation of Autoencoder

SLOW

Simple for training

Extract trained model parameters
and
use with C code

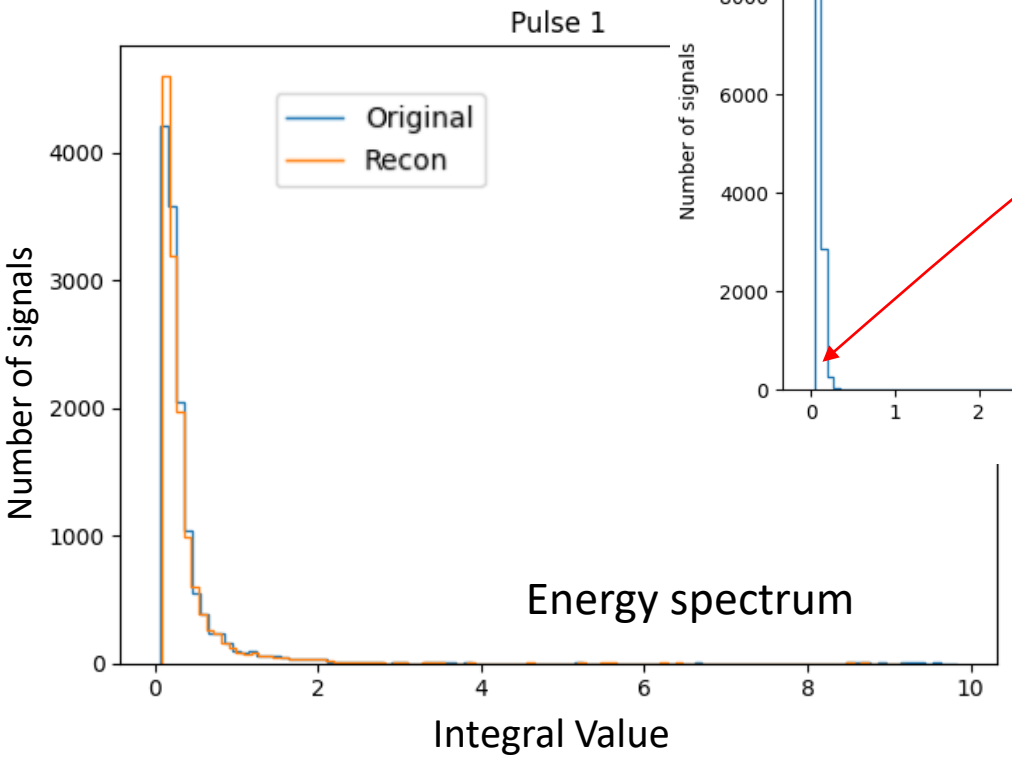


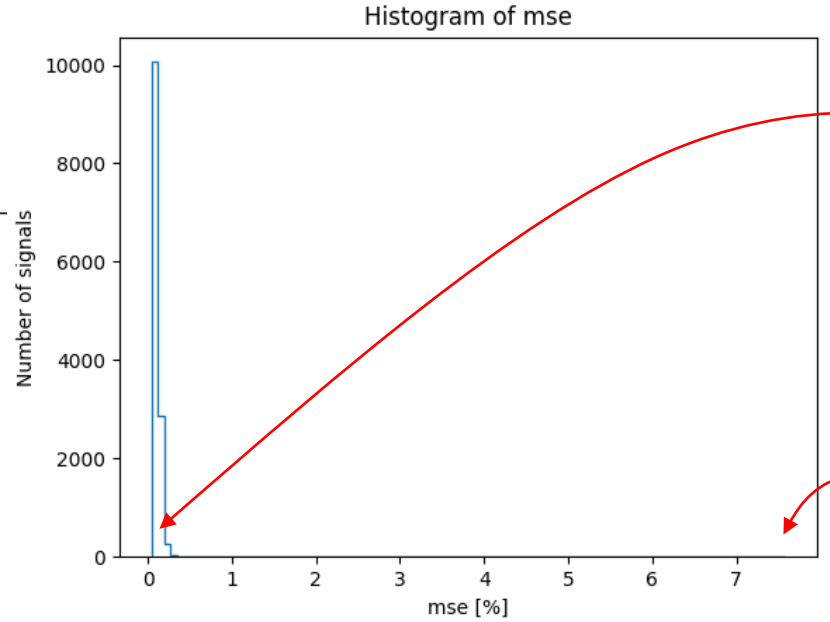
Details of the implementation on High Performance Server

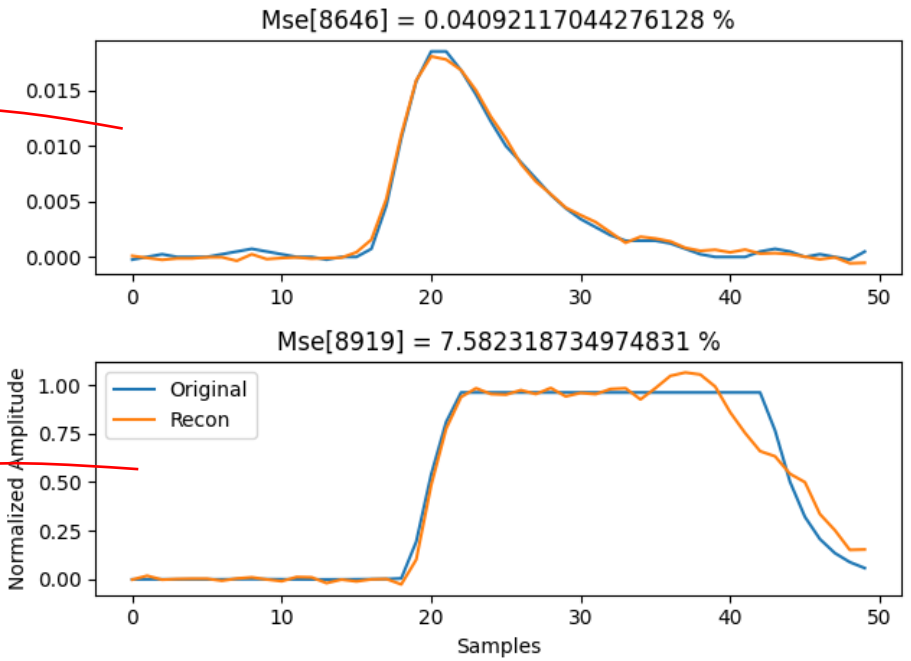
Rossi F., Battaglieri M., Gavalian G., Ragusa E., Gastaldo P.
"Real Time implementation of Artificial Intelligence compression algorithm for High-Speed Streaming Readout signals"
Conference on Computing in High Energy and Nuclear Phisics CHEP 2024

Speed-up compression with C code

DATA REDUCTION: Impact on data (BGO)



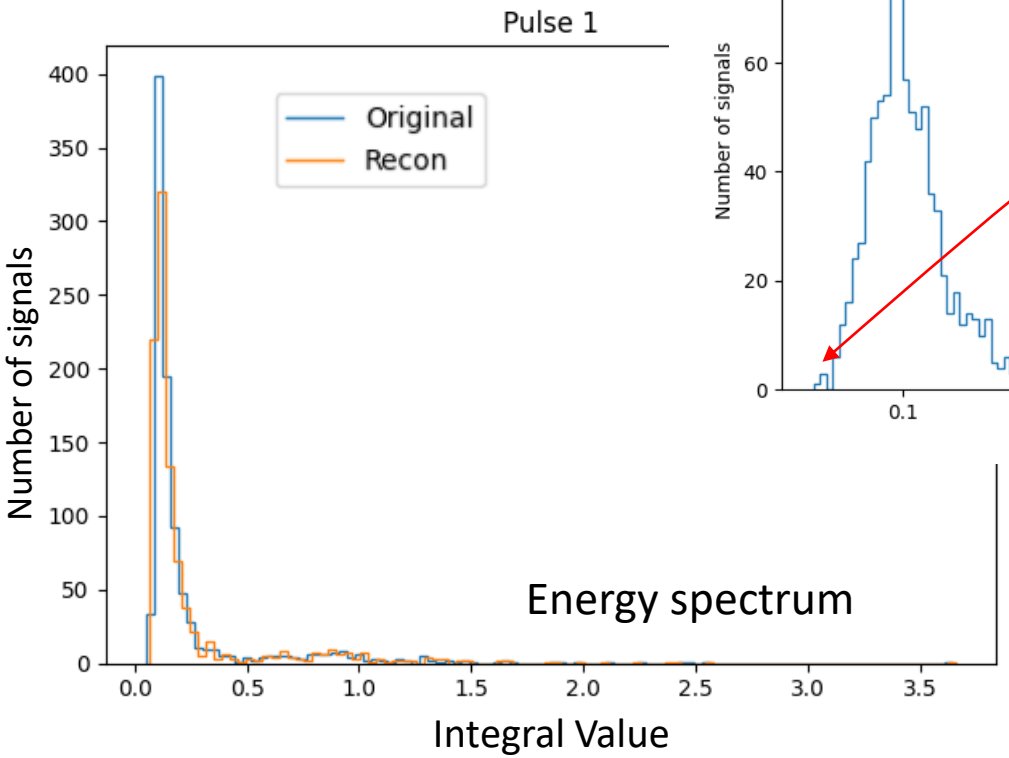


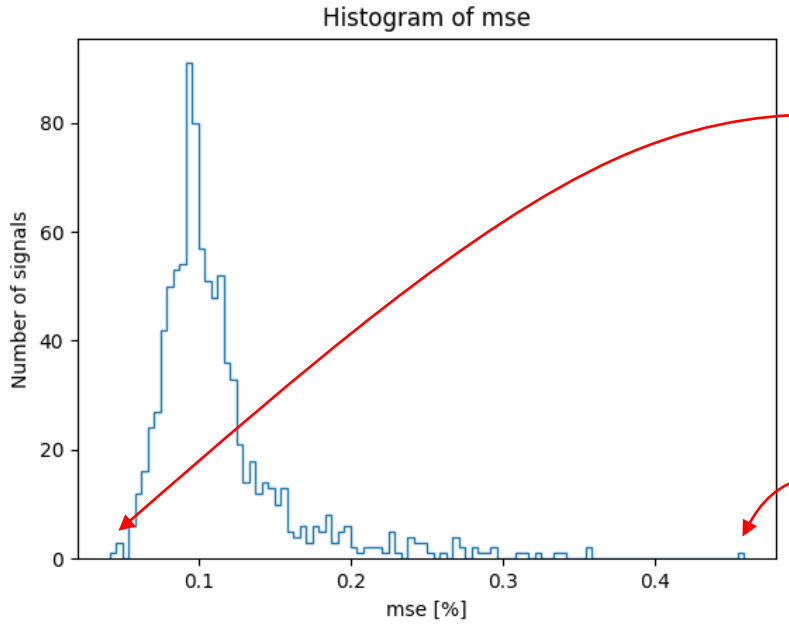


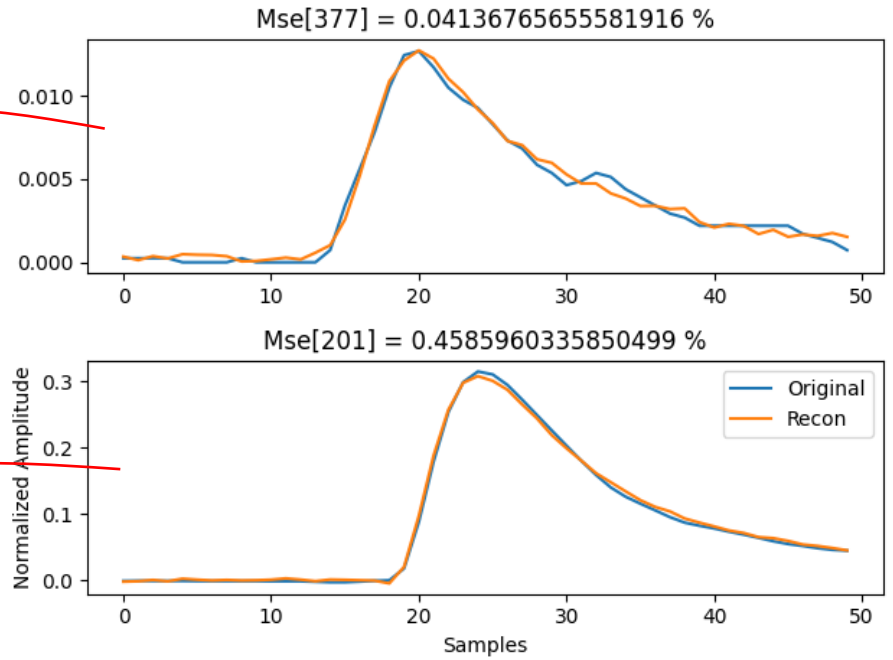
Dimension:	1x1x1 cm^3
Frequency:	10Ms/s
Quantization:	12bits
Signal:	50 samples

Compression factor = 10

DATA REDUCTION: Impact on data (CsI(Tl))







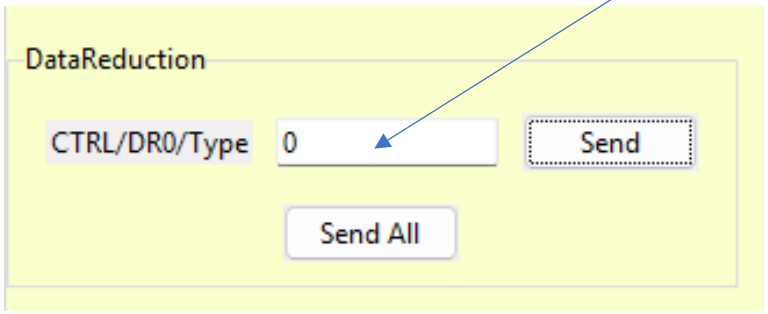
Dimension:	5x5x1 cm^3
Frequency:	10Ms/s
Quantization:	12bits
Signal:	50 samples

Compression factor = 8

DATA REDUCTION: Start streaming compression

sudo ./SNRI_main

Start code



Type of compression

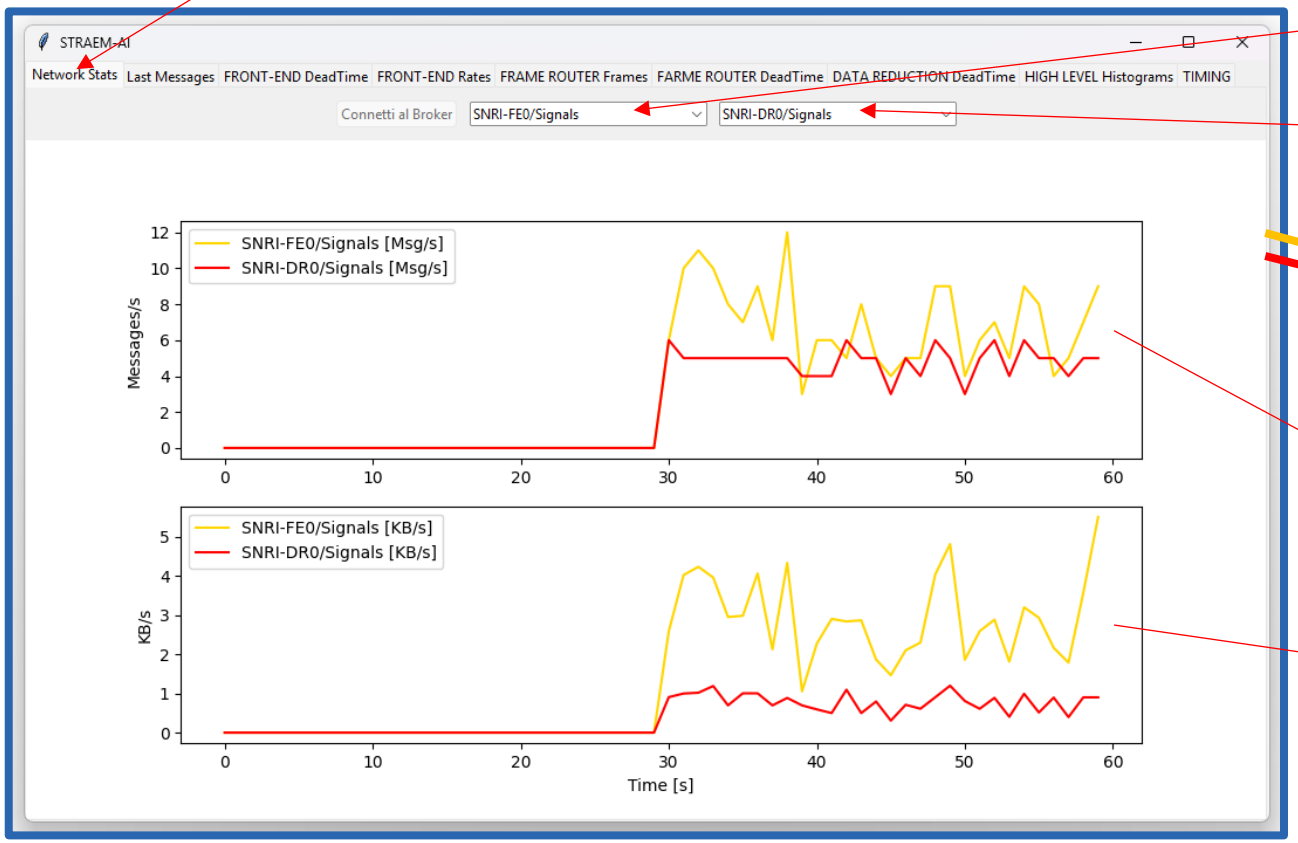
0 – No compression
1 – AI
2 - Gzip

```
pi@SNRI-DR0:~/Desktop/SNRI/Codice $ sudo ./SNRI_main
Sono core 0, Faccio l'elaborazione
Sono core 1, Gestisco l'MQTT
Ottenuti 1 Threads
{"PersiBuffer": "0","PersiFull": "0","Elaborati": "13"}
{"PersiBuffer": "0","PersiFull": "0","Elaborati": "15"}
{"PersiBuffer": "0","PersiFull": "0","Elaborati": "16"}
{"PersiBuffer": "0","PersiFull": "0","Elaborati": "17"}
{"PersiBuffer": "0","PersiFull": "0","Elaborati": "18"}
{"PersiBuffer": "0","PersiFull": "0","Elaborati": "19"}
{"PersiBuffer": "0","PersiFull": "0","Elaborati": "20"}
{"PersiBuffer": "0","PersiFull": "0","Elaborati": "21"}
{"PersiBuffer": "0","PersiFull": "0","Elaborati": "22"}
{"PersiBuffer": "0","PersiFull": "0","Elaborati": "25"}
{"PersiBuffer": "0","PersiFull": "0","Elaborati": "30"}
```

PersiBuffer	PersiFull	Elaborati
Lost Signal for input buffer full	Lost signals for output buffer full	Total number of compressed signals

DATA REDUCTION: Expected behavior

Network Stats

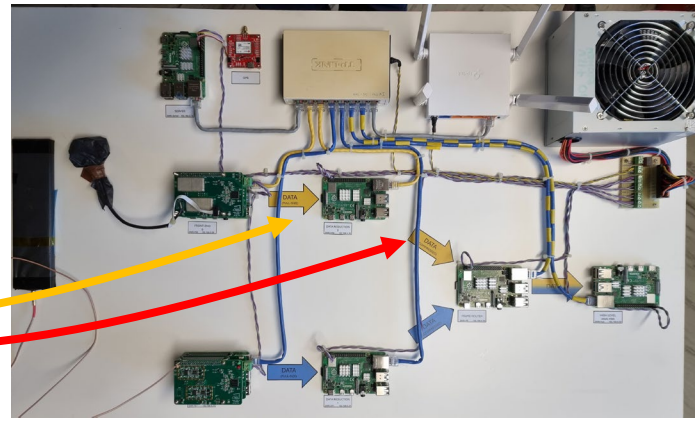


First Topic

Second Topic

MQTT [Messages/s]

Total band [Kbyte/s]

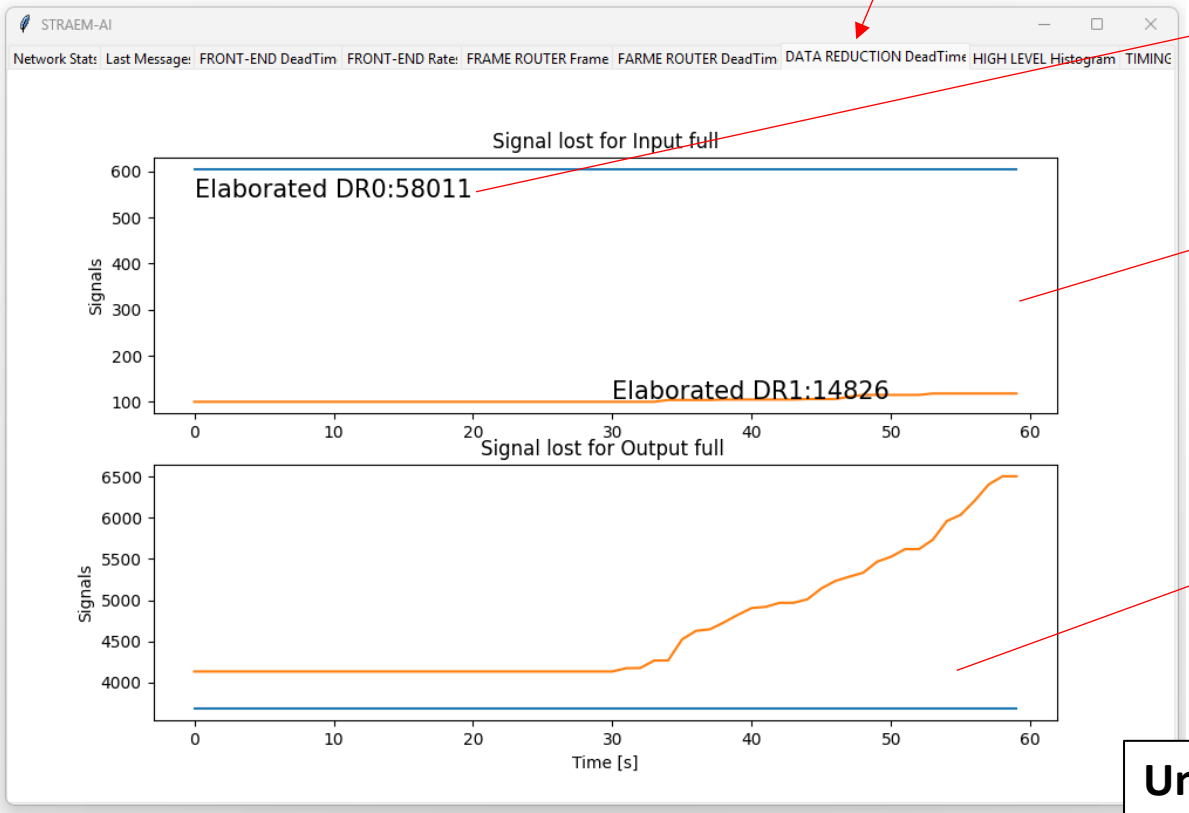


The actual compression ratio may be affected by the data representation. In this code, MQTT messages are implemented using char.

Verify data reduction performance real-time

Data Reduction: Dead-Time

DATA REDUCTION DeadTime



Total number of elaborated signals

Lost signal for input buffer full.

Node cannot accept more signals from Front-End

Lost signal for output buffer full

Node cannot send more signals to Frame Router

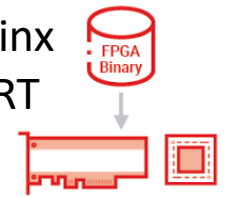
Understand and monitor the performances of the data reduction

Speed up Implementation of Data Reduction Node

4 x NVIDIA Tesla V100 GPU

INFN CNAF

Xilinx XRT



XILINX VITIS AI

ALVEO V70 FPGA

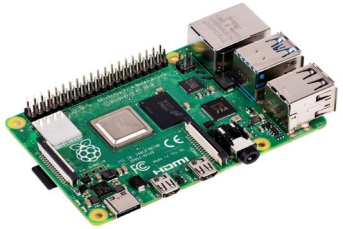
Data Reduction

Receive Signal

Compress (Encoder)

Transmission

Raspberry Pi 4 Rev. B



Low cost hardware (Educational Tool)

LAN (TCP/IP)

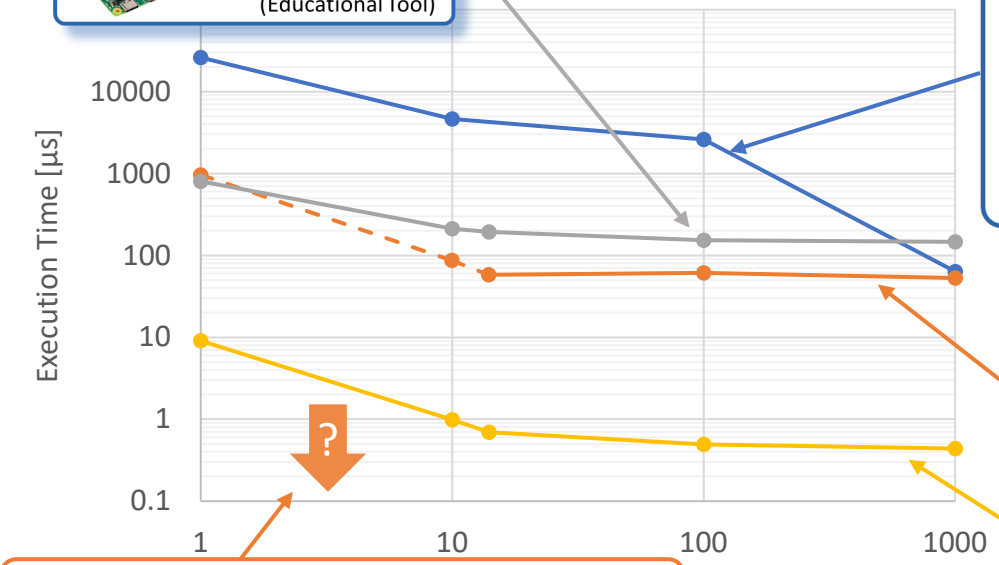
High performance DELL C6400 server (4 x AMD EPYC 7413 24-Core Processor)

Finanziato dall'Unione europea NextGenerationEU
Ministero dell'Università e della Ricerca
Italiadomani PIANO NAZIONALE DI RIPRESA E RESILIENZA
FAIR Future Artificial Intelligence Research

Test for BDX Experiment

Raspberry Pi 4 Rev. B

Low cost hardware (Educational Tool)



4 x NVIDIA Tesla V100 GPU

INFN CNAF

Xilinx XRT

XILINX VITIS AI

ALVEO V70 FPGA

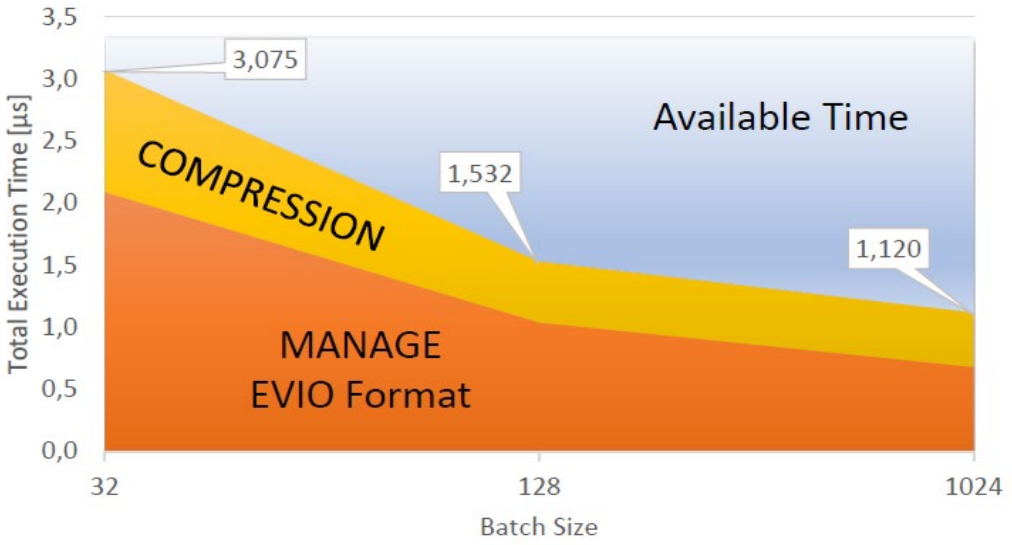
ALINX VD100

Further Studies

hls 4 ml

High performance DELL C6400 server
(4 x AMD EPYC 7413 24-Core Processor)

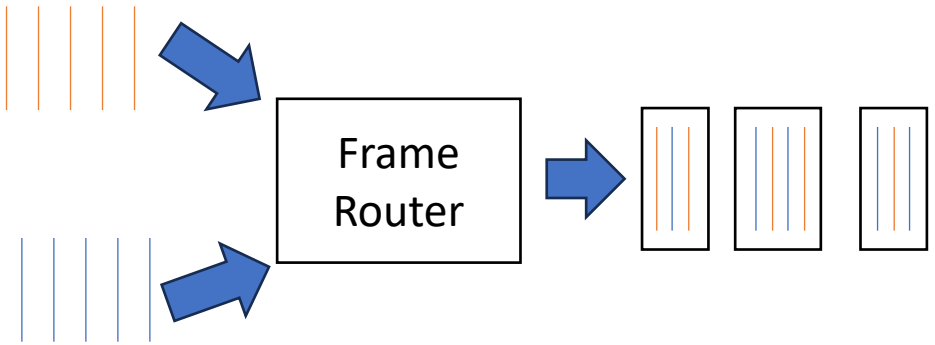
BDX Use Case

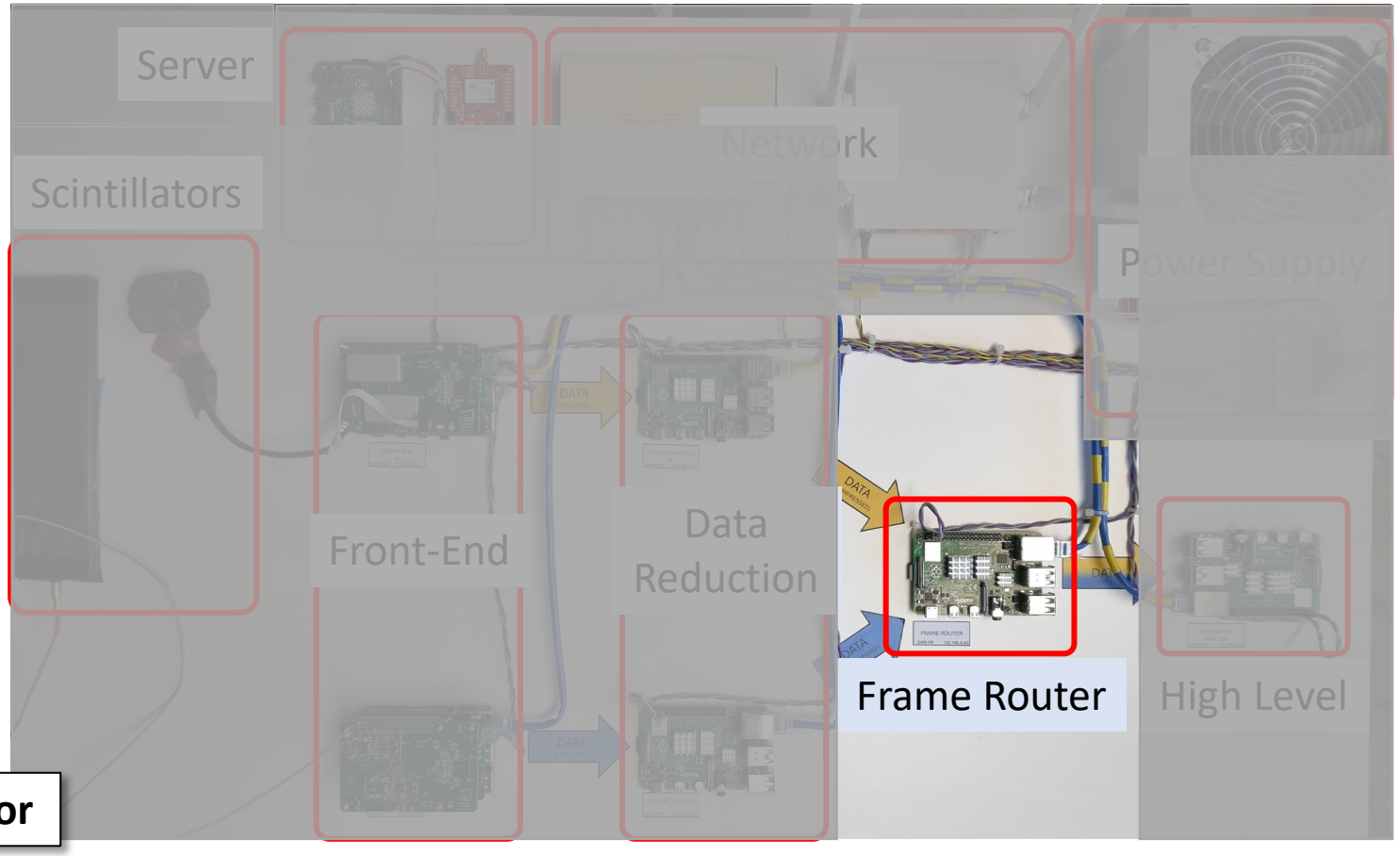


Rate up to 300KHz

FRAME ROUTER

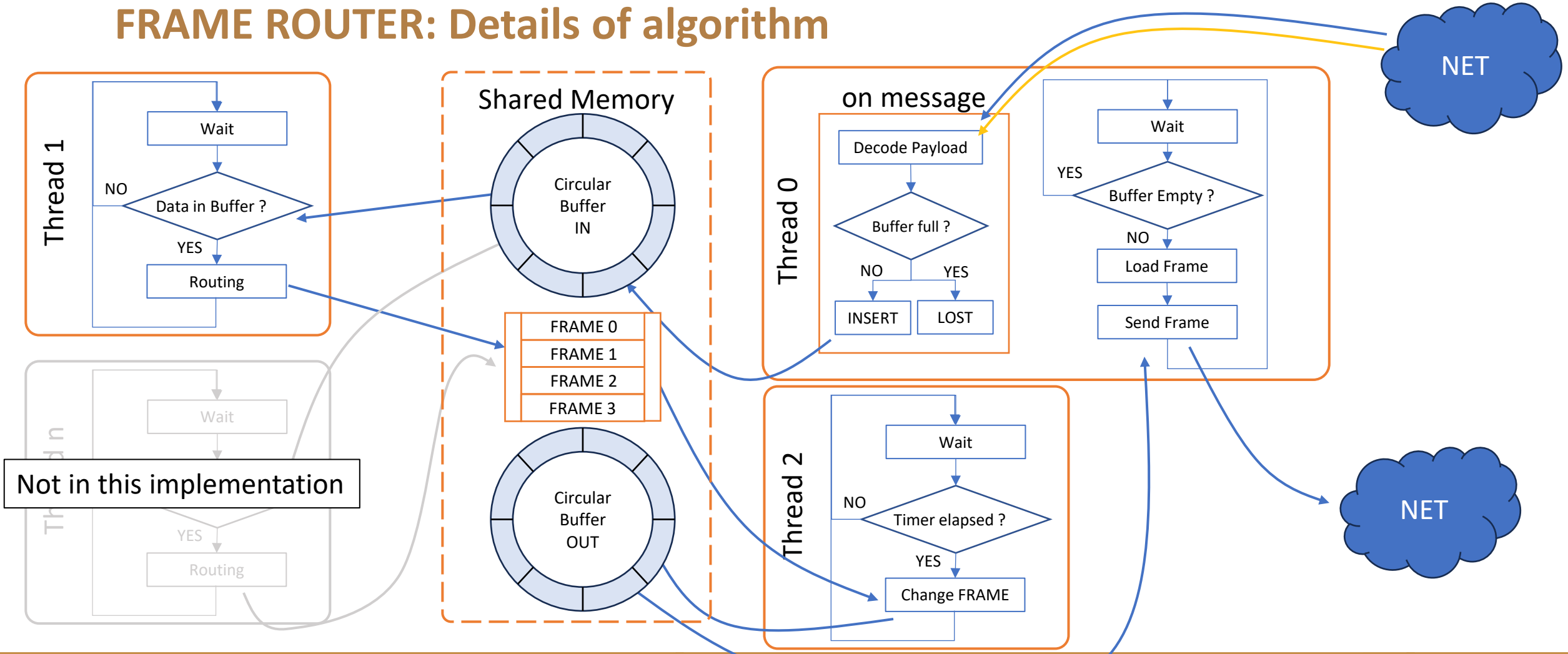
- Merge signals from two stream
- Build time frame with all the signal with similar timestamp
- Send to one (or more) elaboration node





Each frame contains the entire state of the detector

FRAME ROUTER: Details of algorithm



FRAME ROUTER: Details of algorithm

Initialized with the first message TS

Frame name	Frame 0		Frame 1		Frame 2		Frame 3	
Start timestamp of frame	TS IN	N-1	TS IN	N	TS IN	N+1	TS IN	N+2
Finish timestamp of frame	TS OUT	N	TS OUT	N+1	TS OUT	N+2	TS OUT	N+3
Timestamp of frame creation	TS APT	-1	TS APT	0	TS APT	1	TS APT	2
Timestamp for closing frame	TS CRA	3	TS CRA	4	TS CRA	5	TS CRA	6

Parametric frame number

Required to take into account messages with timestamp older that the first one

Thread 1

Compare TS of incoming messages to insert in the right frame.

Thread 2

Compare global timestamp with the closure time of each frames. Remove if elapsed and create a new one.

- **Few FRAMES:**
  - Low memory impact
  - High probability of lost signals
- **Many FRAMES:**
  - High memory impact
  - Low probability of lost signals

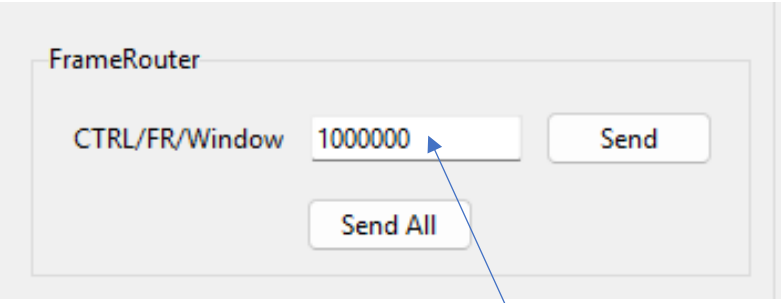
FRAME ROUTER: Start frame routing

sudo ./SNRI_main

Start code

First Received signal used for synchronization

Info of each closed frame

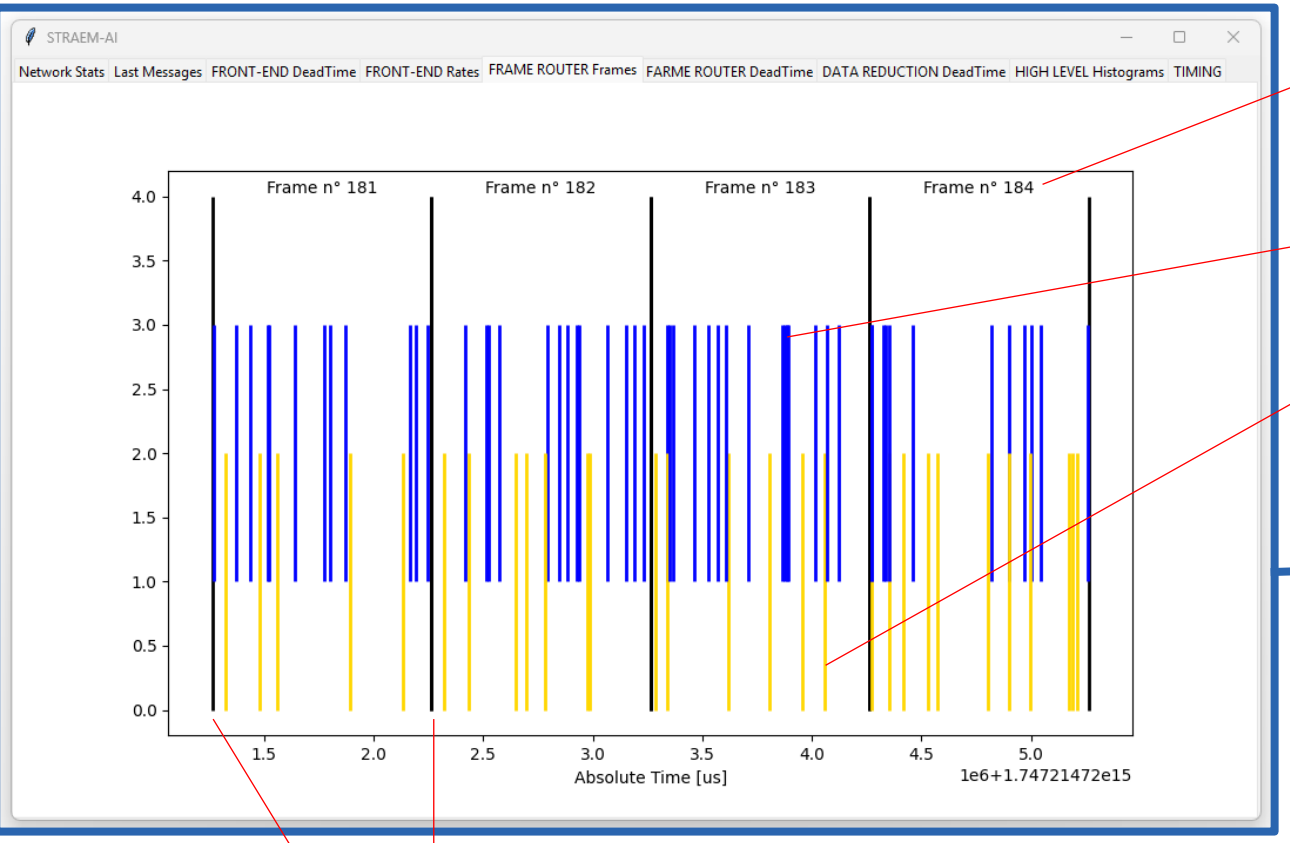


Frame windows duration [µs]

This implementation wok with duration between 0.5 and 2 seconds.

```
pi@SNRI-FR:~/Desktop/SNRI/Codice $ sudo ./SNRI_main
Acquisition paramteres: ok
Sono core 0, Faccio il Routing
Routing: Inizializzazione
Sono core 1, Gestisco l'MQTT
Signal:
        timestamp: 1747262895245065
        Front-End: SNRI-FE0
        Number of Samples: 100
        Samples: [0, -2, -2, -2, -4, -4, -4, -4, -5, -6, -6, -6, -5, -4, -2,
, 26, 21, 20, 19, 17, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0
Sono sub 1, carico all'interno dei thread
Sono sub 0, cambio i frame
Frame n. : 0
        timestamp_in       : 1747262894245065
        timestamp_out      : 1747262895245065
        timestamp_apertura : 1744667614763697
        timestamp_chiusura : 1744667615763697
        contiene segnali   : 0
Frame n. : 2
        timestamp_in       : 1747262895245065
        timestamp_out      : 1747262896245065
        timestamp_apertura : 1744667615763699
        timestamp_chiusura : 1744667616763699
        contiene segnali   : 12
Frame n. : 4
        timestamp_in       : 1747262896245065
        timestamp_out      : 1747262897245065
        timestamp_apertura : 1744667616763700
        timestamp_chiusura : 1744667617763700
        contiene segnali   : 7
Frame n. : 3
        timestamp_in       : 1747262897245065
        timestamp_out      : 1747262898245065
        timestamp_apertura : 1744667615764087
        timestamp_chiusura : 1744667618764087
```

FRAME ROUTER: Expected behavior

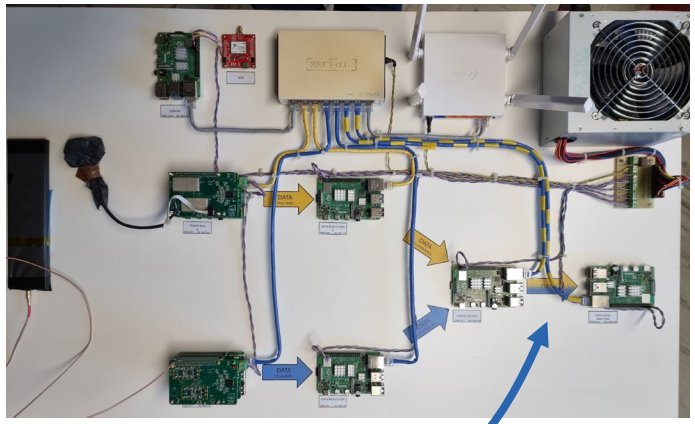


Frame number

FE1 Signals

FE0 Signals

Frame boundaries

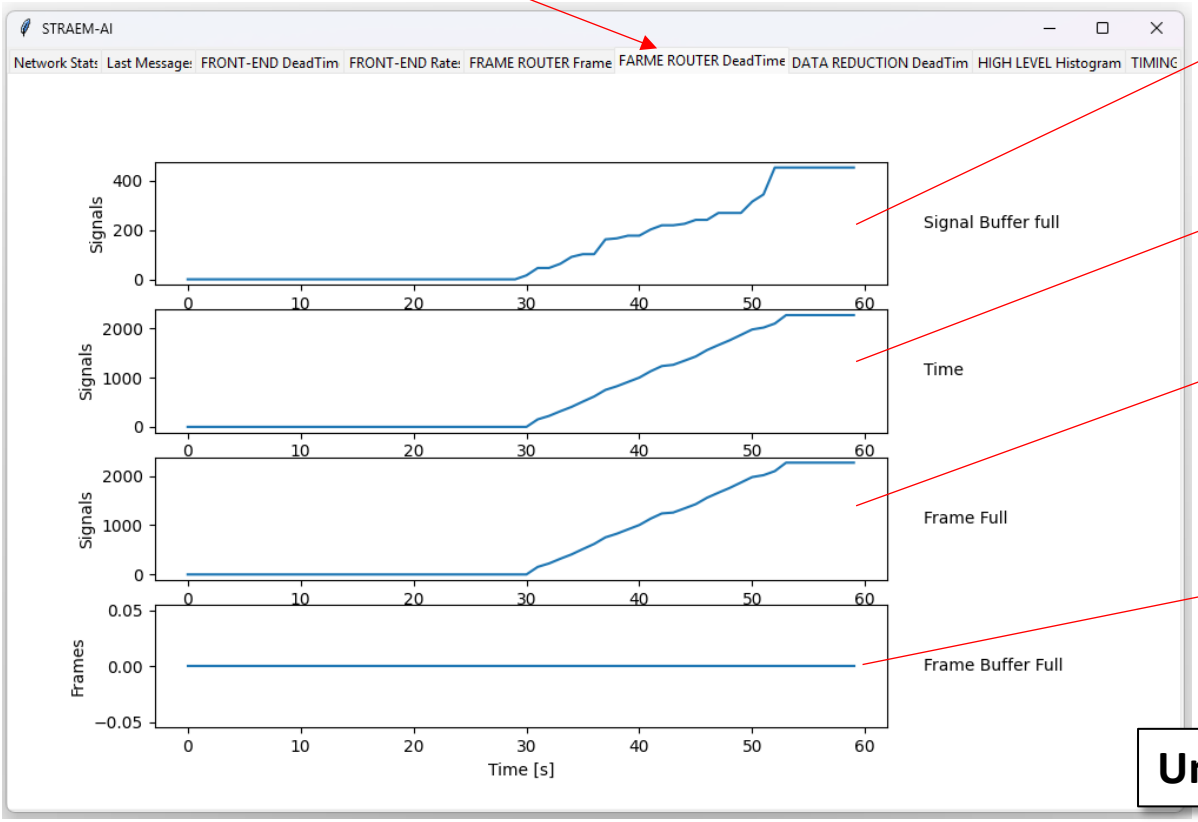


Lines represent only the events not its amplitude. Different height are useful only for visualize signals when overlapped.

Visually and qualitatively understand how the frame router works

Frame Router: DeadTime

FRAME ROUTER DeadTime



Lost signals for Input buffer full

Node cannot accept more signal from Front-End

Lost signal because its timestamp is no more in the open frame

Signals timestamp is no more in the open frames

Signal lost due to the frame full

Signals array in frame is full, no more signals can add

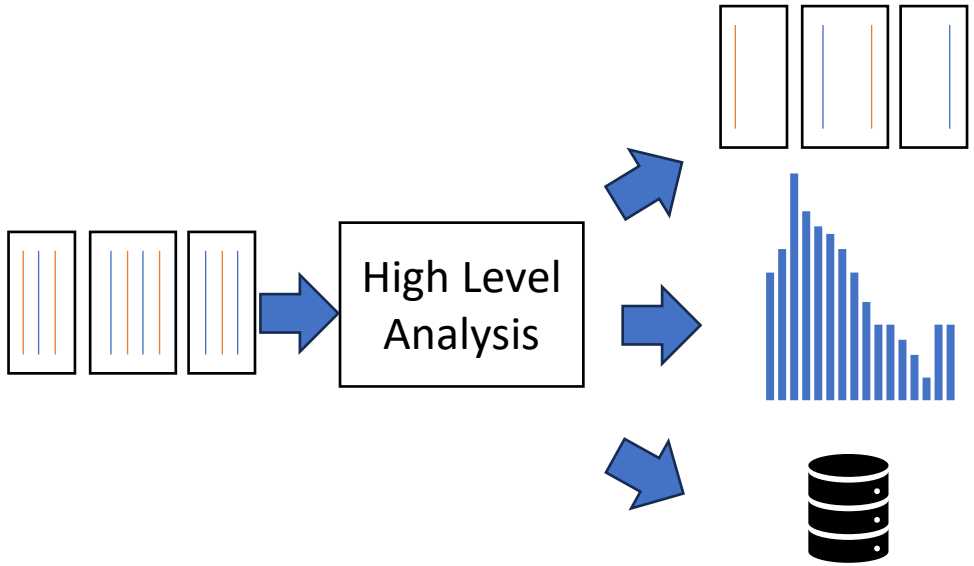
Frame lost for output buffer full

Node cannot send more frames to HIGH LEVEL ANALISYS

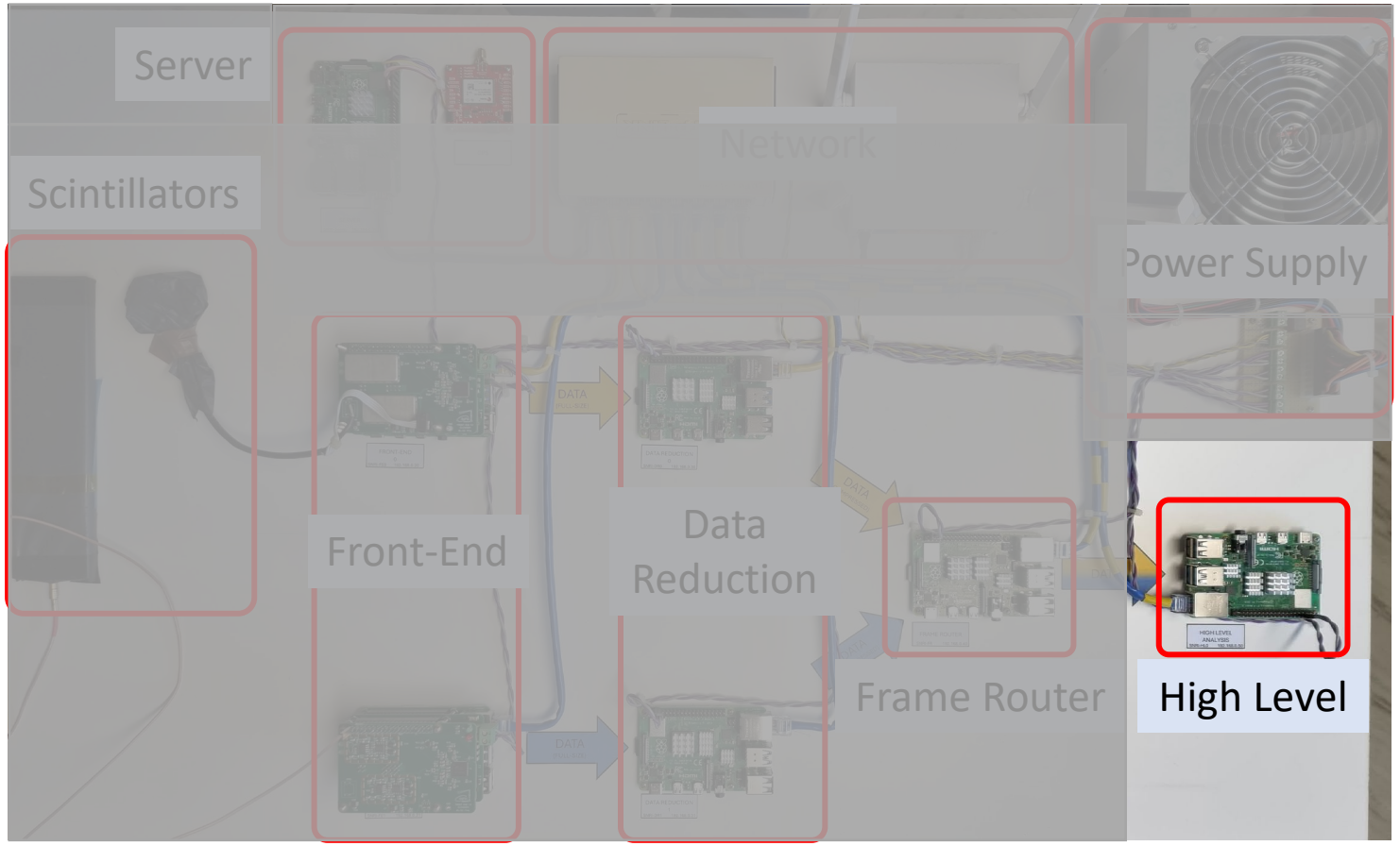
Understand and monitor the performances of the frame router

HIGH LEVEL ELABORATION

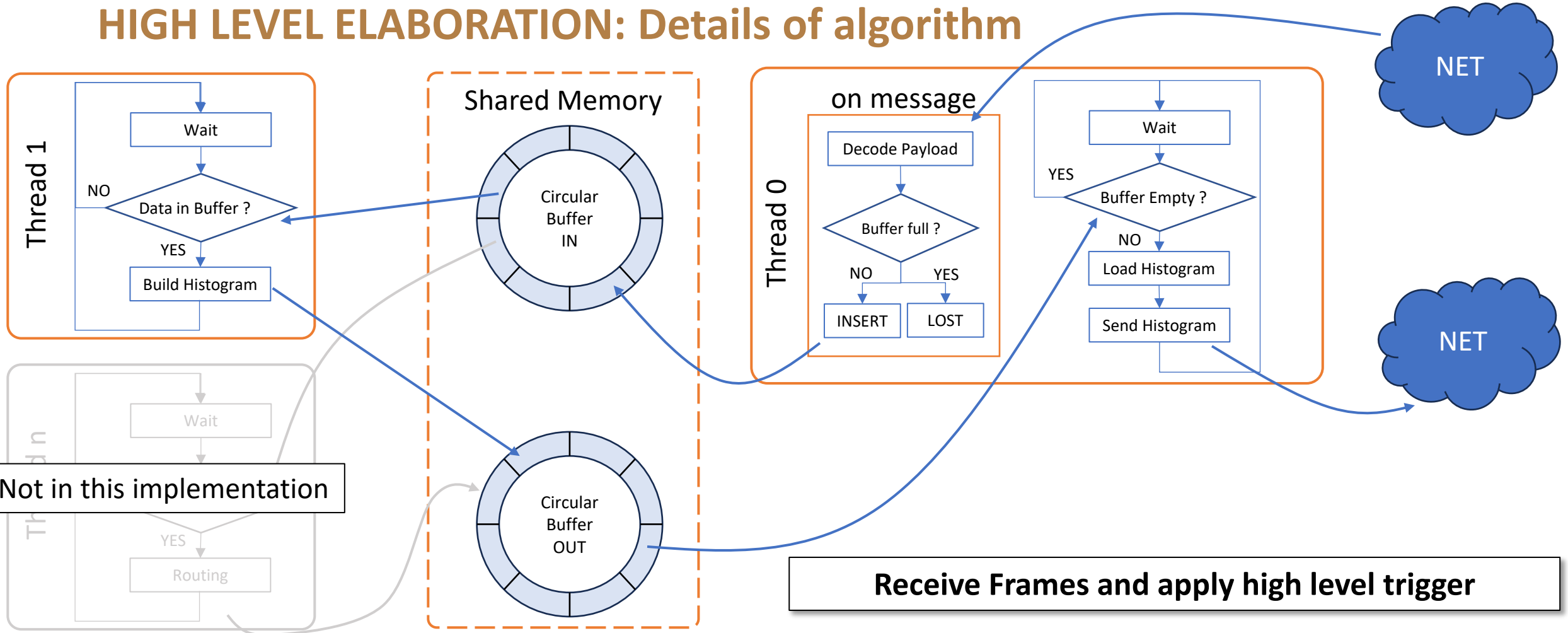
- Apply high level trigger
- Could be distributed on one or more nodes
- Make decision on a time frame
- Make decision knowing the entire state of the detector



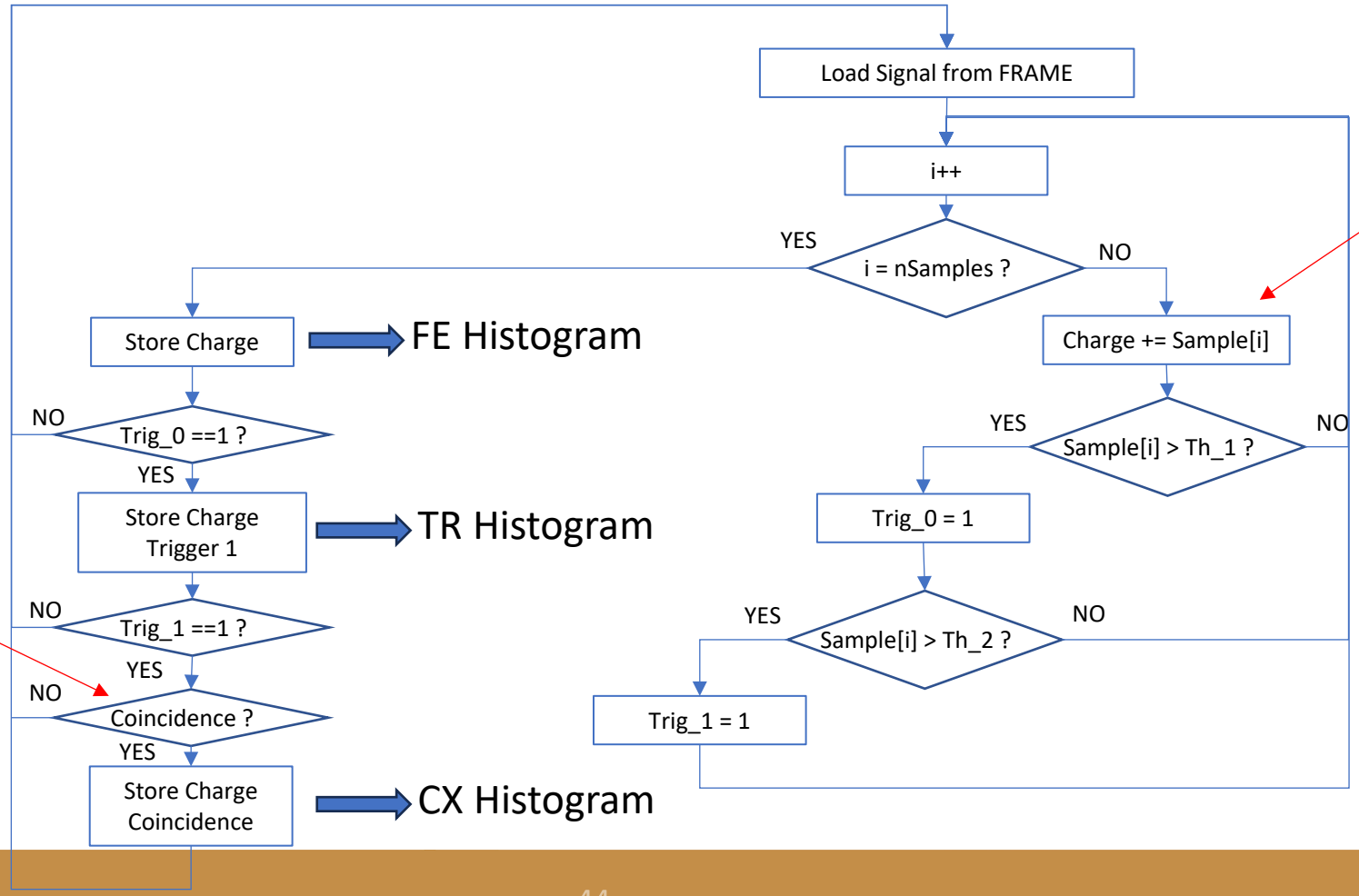
High Level trigger, Histograms, save on disk



HIGH LEVEL ELABORATION: Details of algorithm



HIGH LEVEL ELABORATION: Thread 1 algorithm in details

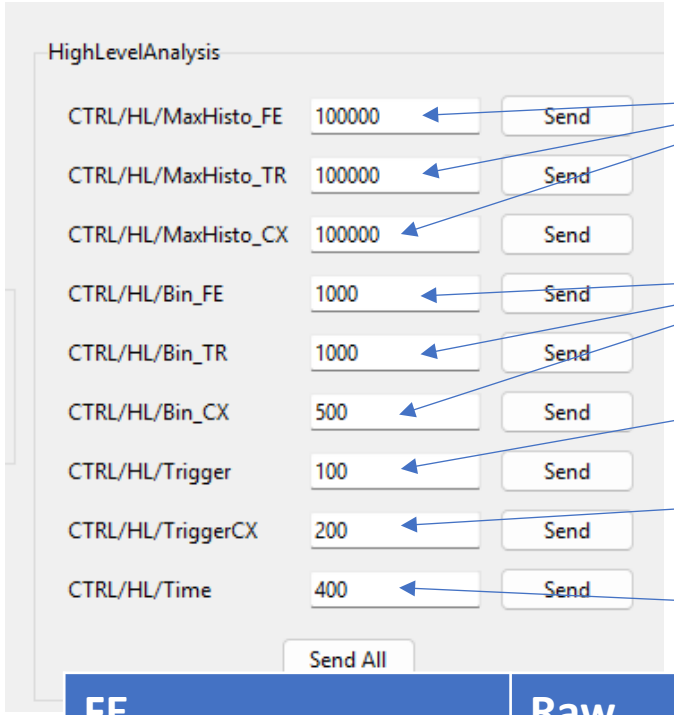


Compute Integral
Sum of all samples in this version

A trade-off was made regarding execution time. In this version, the threshold of the signals used for comparison with the current one is not checked.

HIGH LEVEL ANALISYS: Start Analysis

sudo ./SNRI_main



Time of histogram computation.

Histograms max value

Histograms number of bin

Trigger level on single channel [bit]

Trigger level for coincidence [bit]

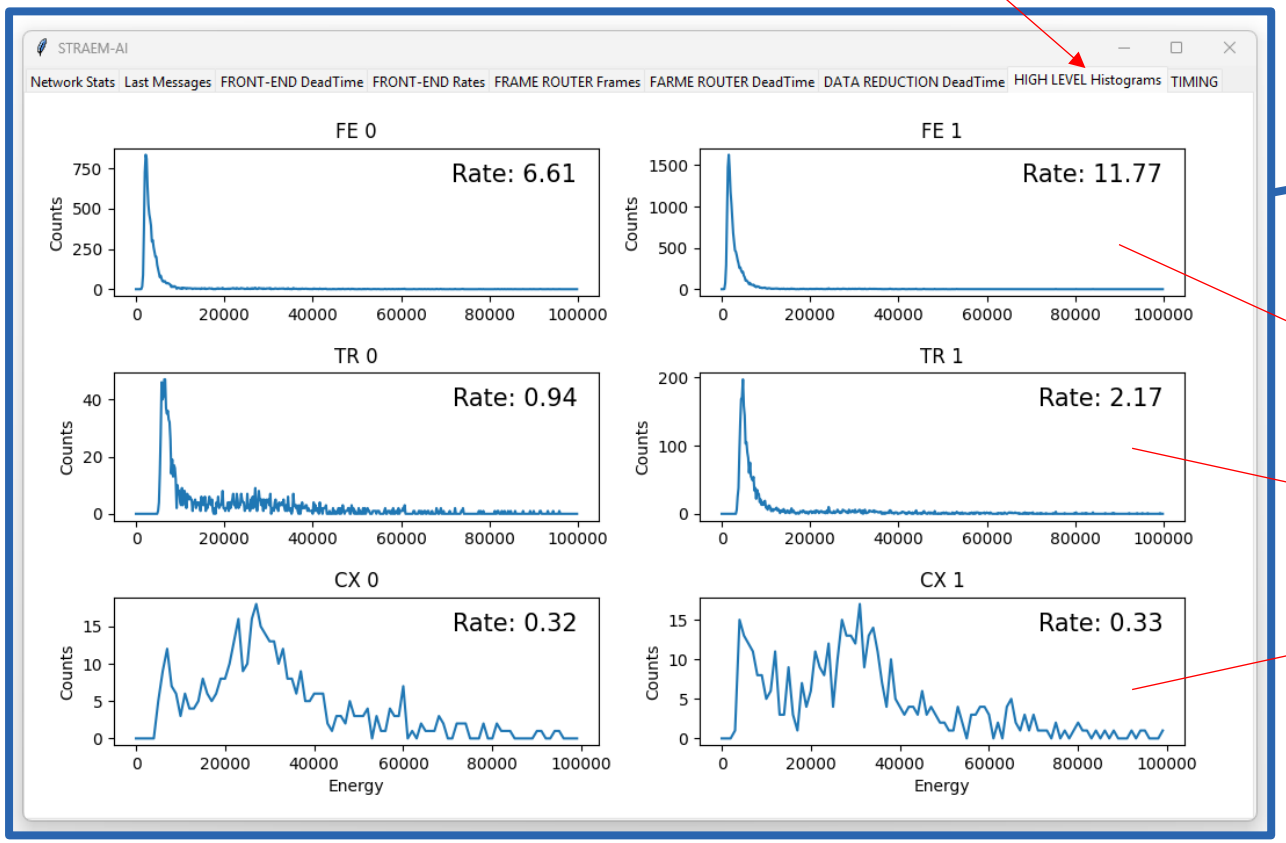
Max time difference for coincidence [µs]

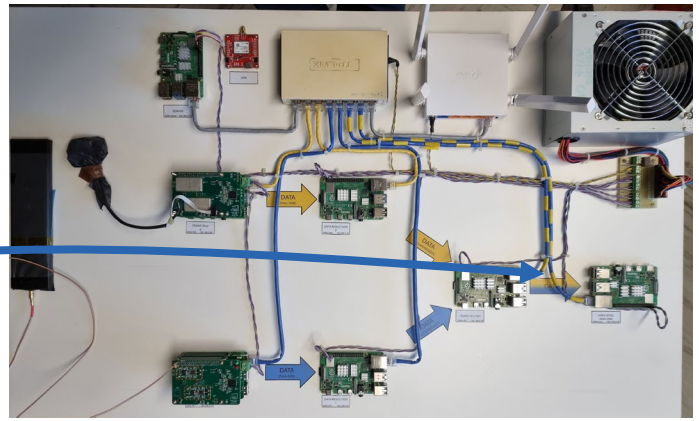
```
pi@SNRI-HL0:~/Desktop/SNRI/Codice $ sudo ./SNRI_main
Sono core 0, Faccio l'elaborazione
Sono core 1, Gestisco l'MQTT
Parametri: ok
Acquisition paramteres: ok
Calibration: ok
Ottenuti 1 Threads
Histogram computed: 7.29
        FE      Raw     Trig    Coin
        0       0.00    0.00    0.00
        1       0.00    0.00    0.00
Histogram computed: 8.23
        FE      Raw     Trig    Coin
        0       0.85    0.85    0.00
        1       0.36    0.36    0.00
Histogram computed: 9.16
        FE      Raw     Trig    Coin
        0       1.97    1.97    0.00
        1       0.98    0.98    0.00
Histogram computed: 10.09
        FE      Raw     Trig    Coin
        0       2.68    2.68    0.00
        1       1.59    1.59    0.00
Histogram computed: 11.01
        FE      Raw     Trig    Coin
        0       3.18    3.18    0.00
        1       2.27    2.27    0.00
Histogram computed: 11.93
        FE      Raw     Trig    Coin
        0       4.02    4.02    0.00
        1       2.60    2.60    0.00
Histogram computed: 12.85
        FE      Raw     Trig    Coin
```

FE	Raw	Trig	Coin
Front End identification number	Histogram of raw acquired data	Histogram of signal after trigger	Histogram after coincidence

Histograms

HIGH LEVEL Histograms





Histogram of all the signals acquired

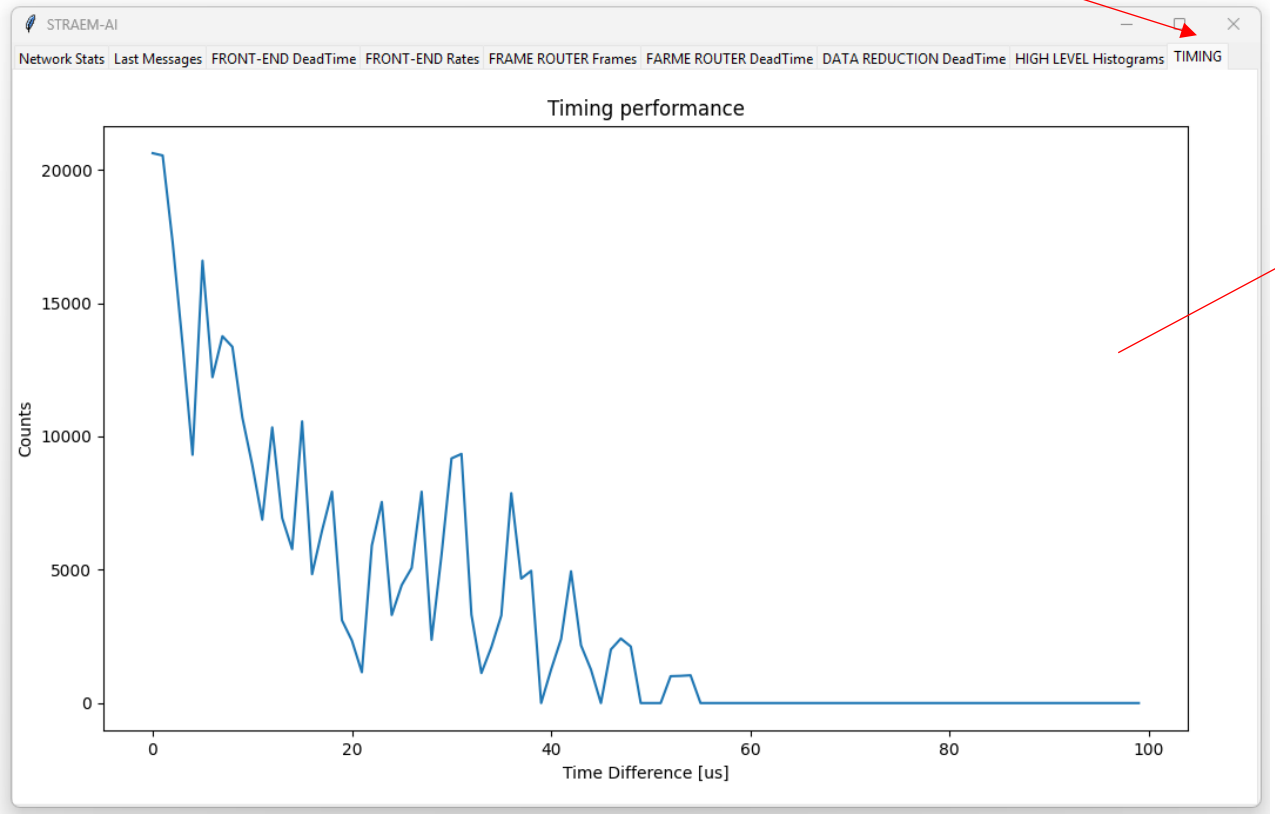
Histogram after high level trigger

Histogram after coincidence

Verify trigger, coincidence and spectroscopic features

Timing

TIMING



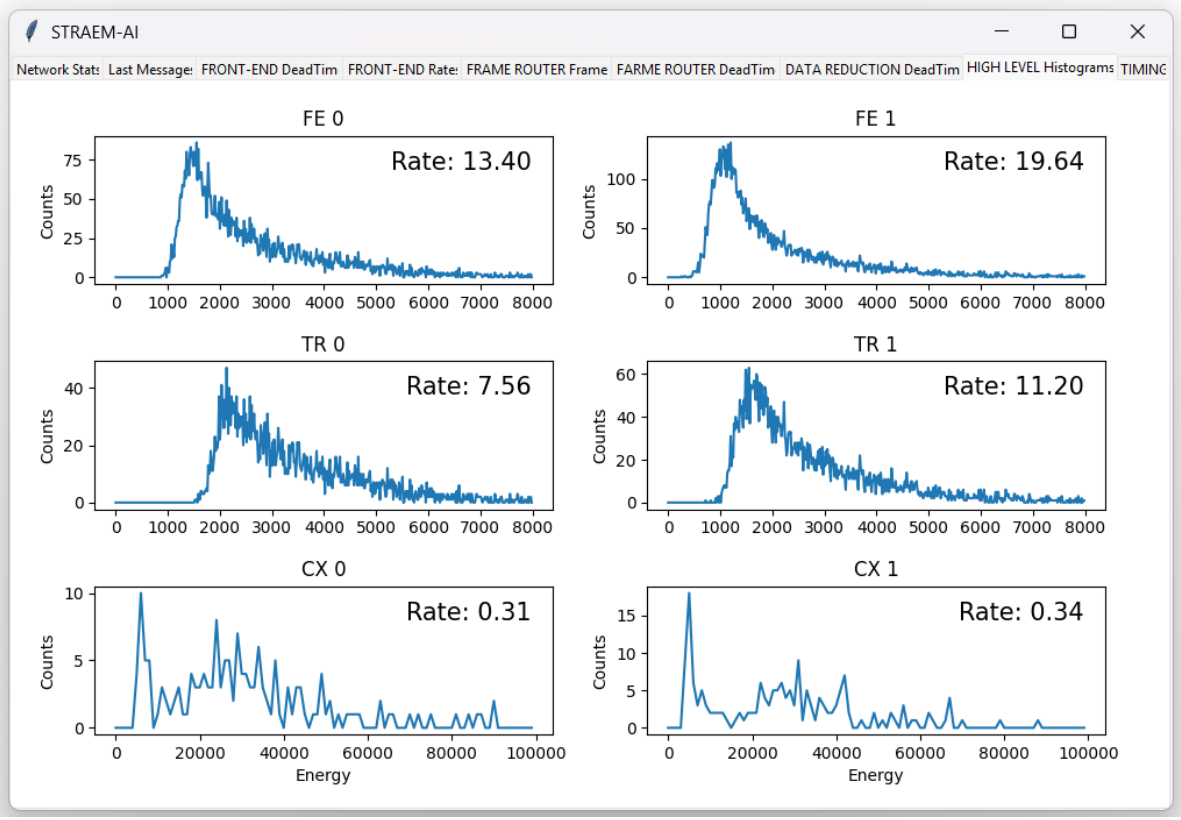
Time difference between 2 coincident signals

This time chart cannot monitor the real coincident timing.
It represent the accuracy in the timing of this implementation. Coincidence time for real signals are less than the resolution of STREAM-AI.

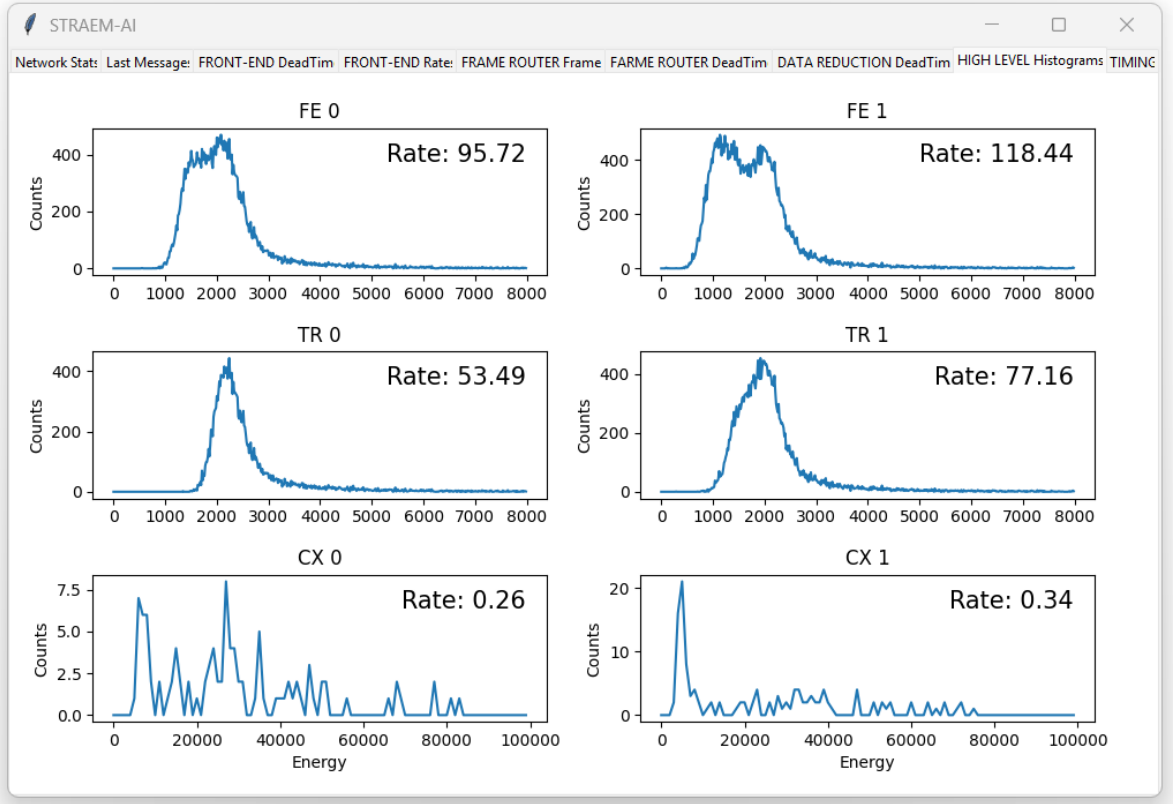
Monitor the synchronization

CsI(Tl) Cristal: Measure with Cs137 source

Background

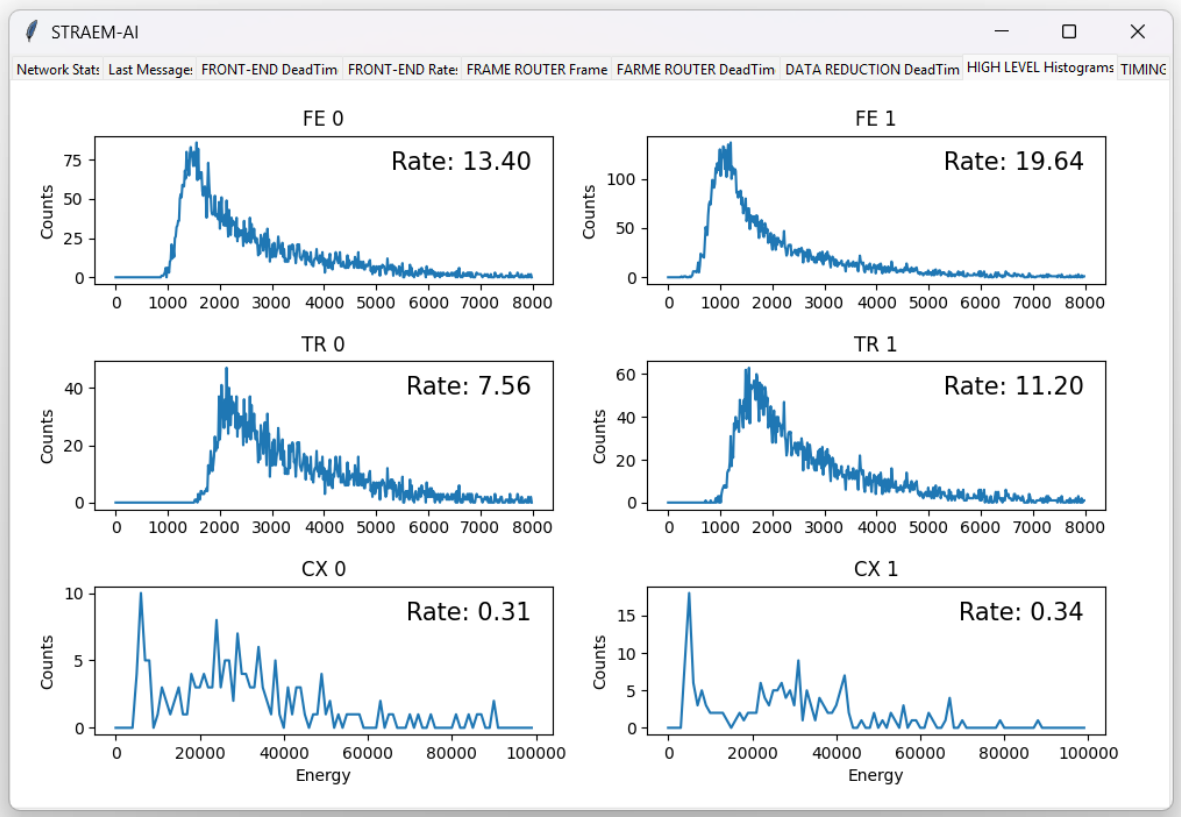


Cs137

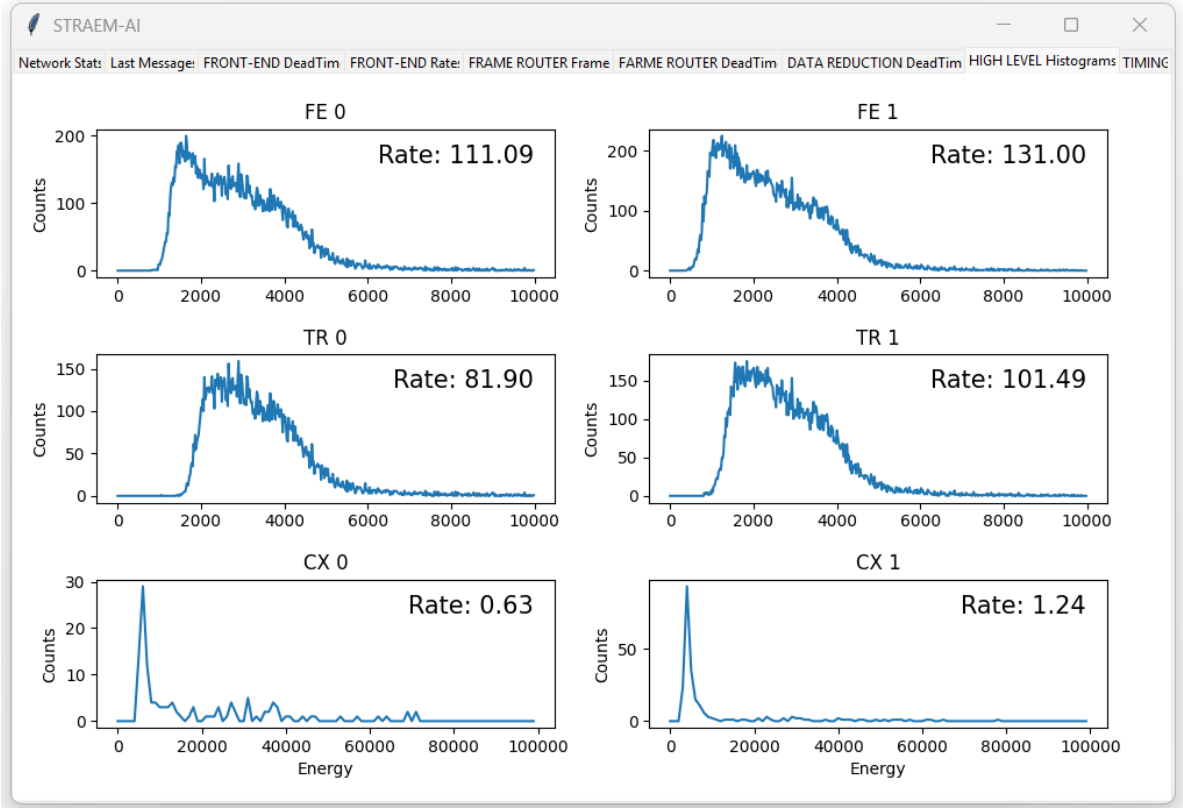


CsI(Tl) Cristal: Measure with Co60 source

Background

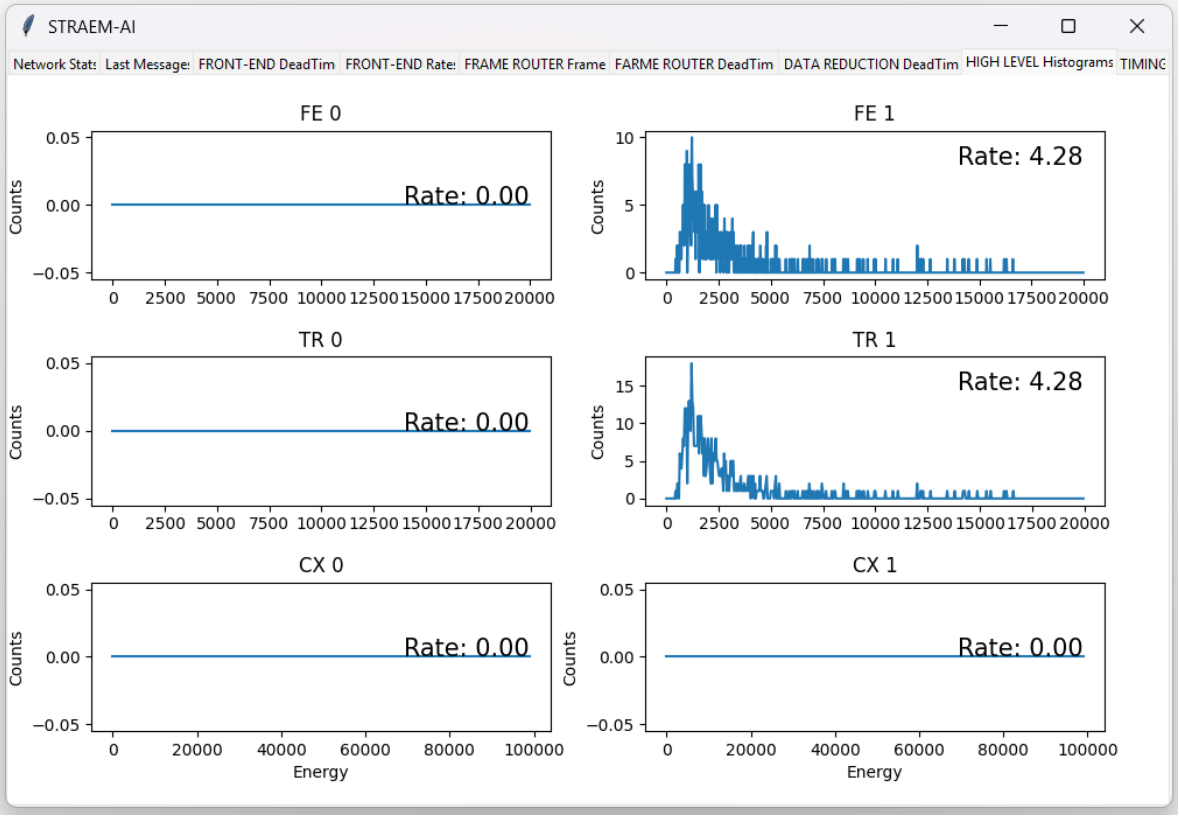


Co60

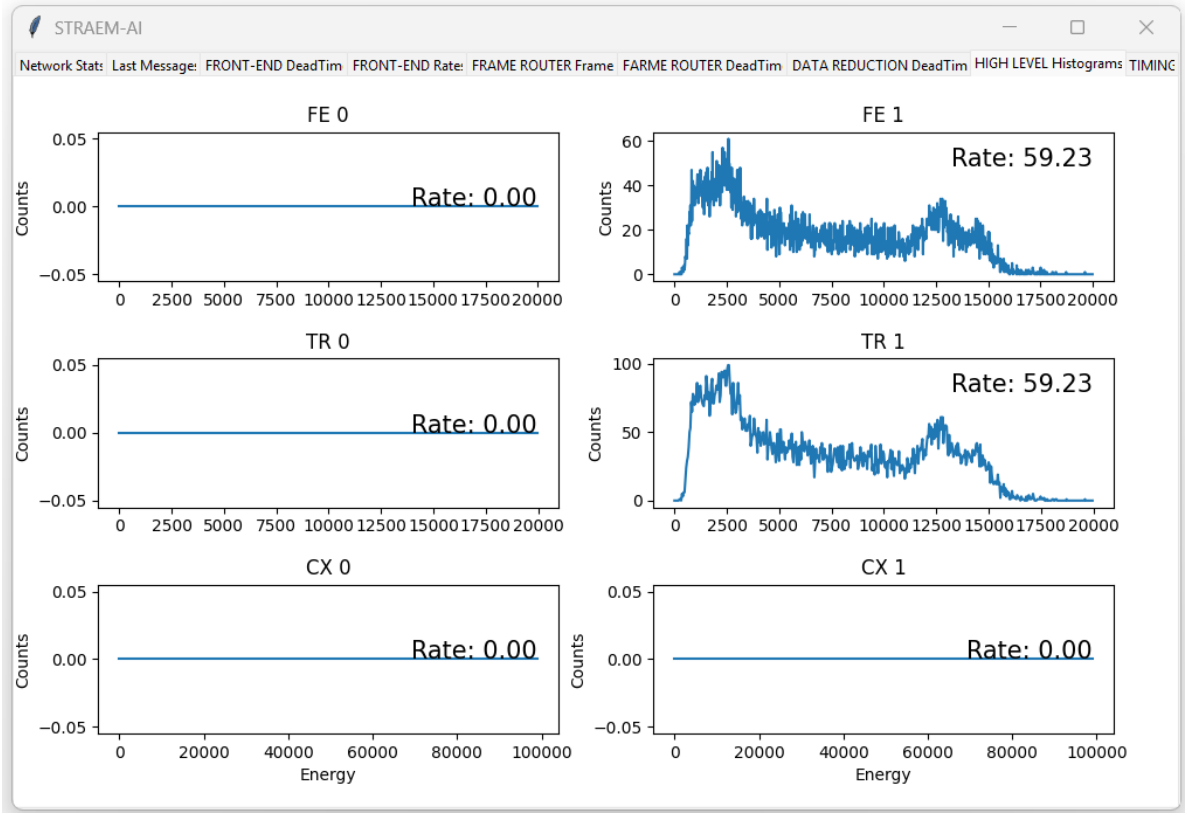


BGO Cristal: Measure with Co60

Background



Co60



https://fondazione-fair.it

https://www.ge.infn.it

ACKNOWLEDGMENT

Authors have received support from:

FAIR - Future Artificial Intelligence Research, funded by the European Union Next-Generation EU (Italy) Research) – spoke 6.