Development of a PicoTDC-based card for the ALICE TOF detector

Presented by: Sandro Geminiani for the Picoteam

Picoteam: Dr. Davide Falchieri (Supervisor), Dr. Pietro Antonioli (Co-supervisor), Casimiro Baldanza, Dr. Marco Giacalone, Sandro Geminiani, Jacopo Succi, Dr. Carlo Veri

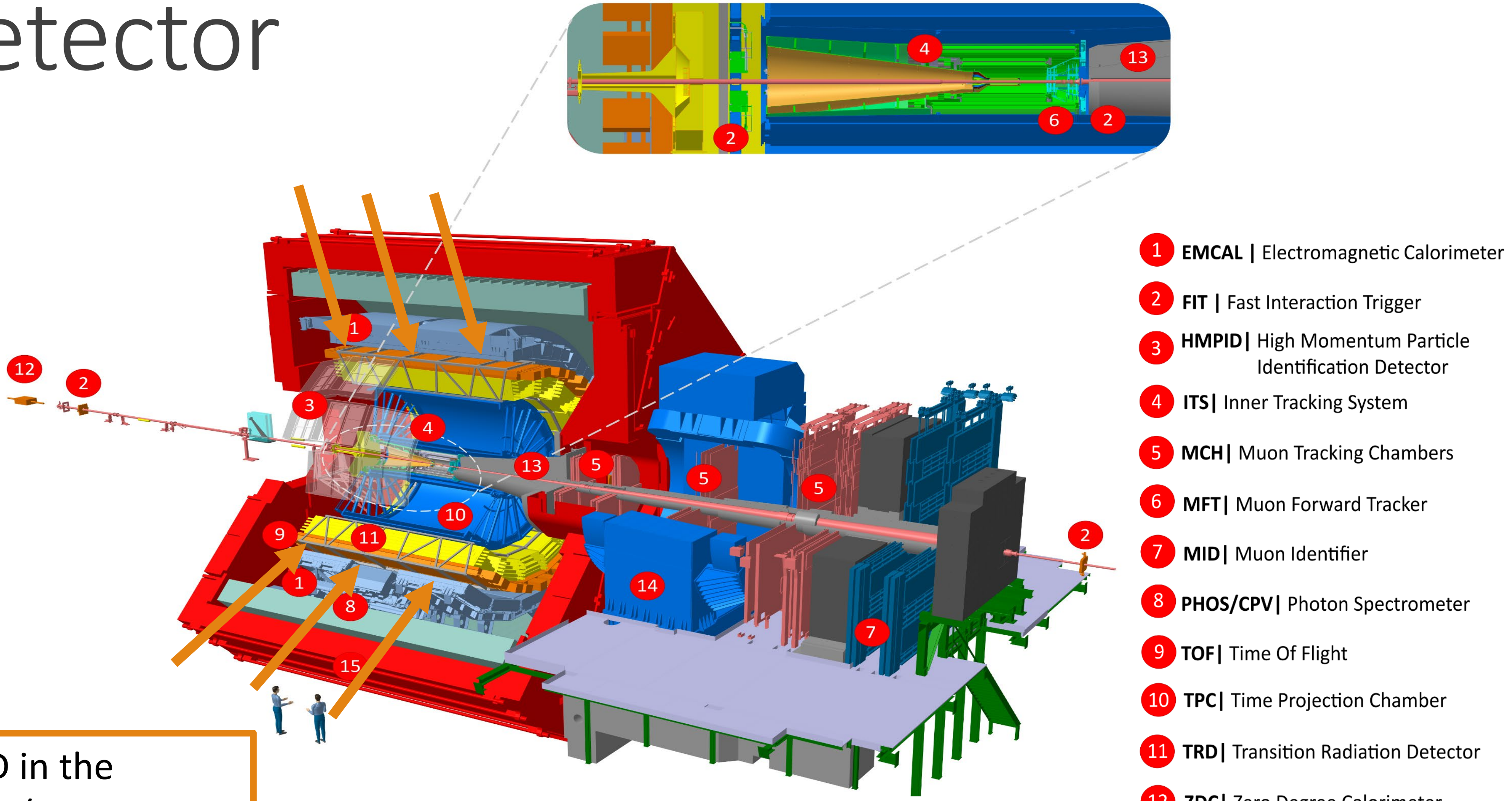
ALICE-ePIC meeting

Bologna – 28/08/2024

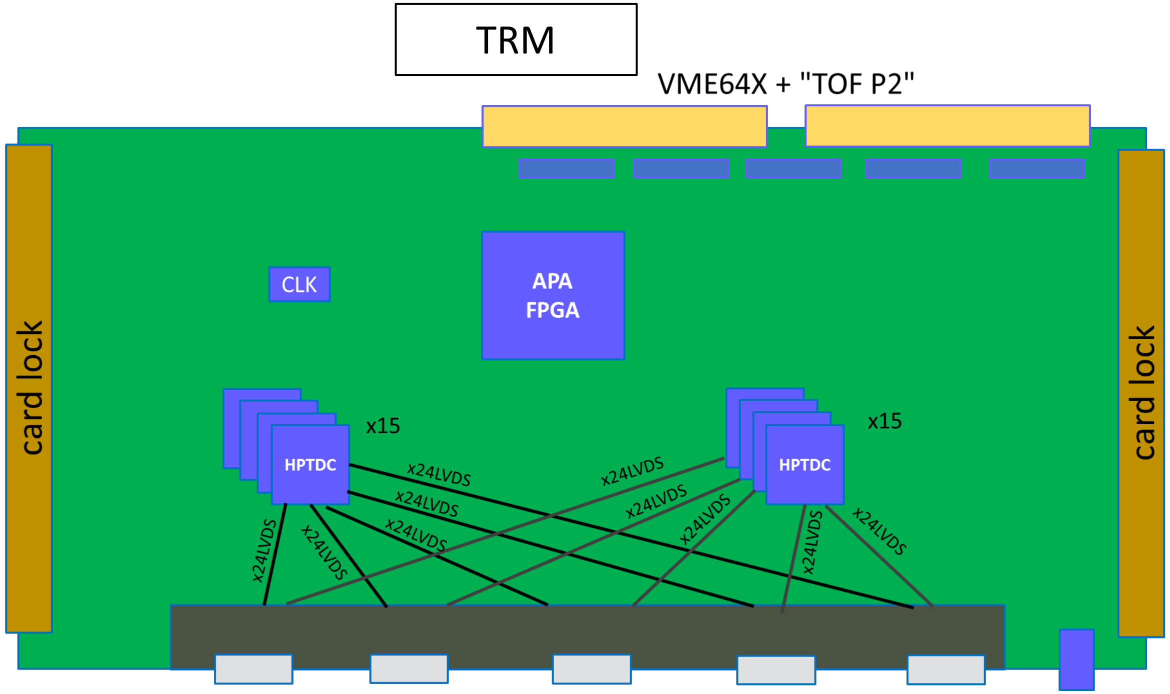
The ALICE detector

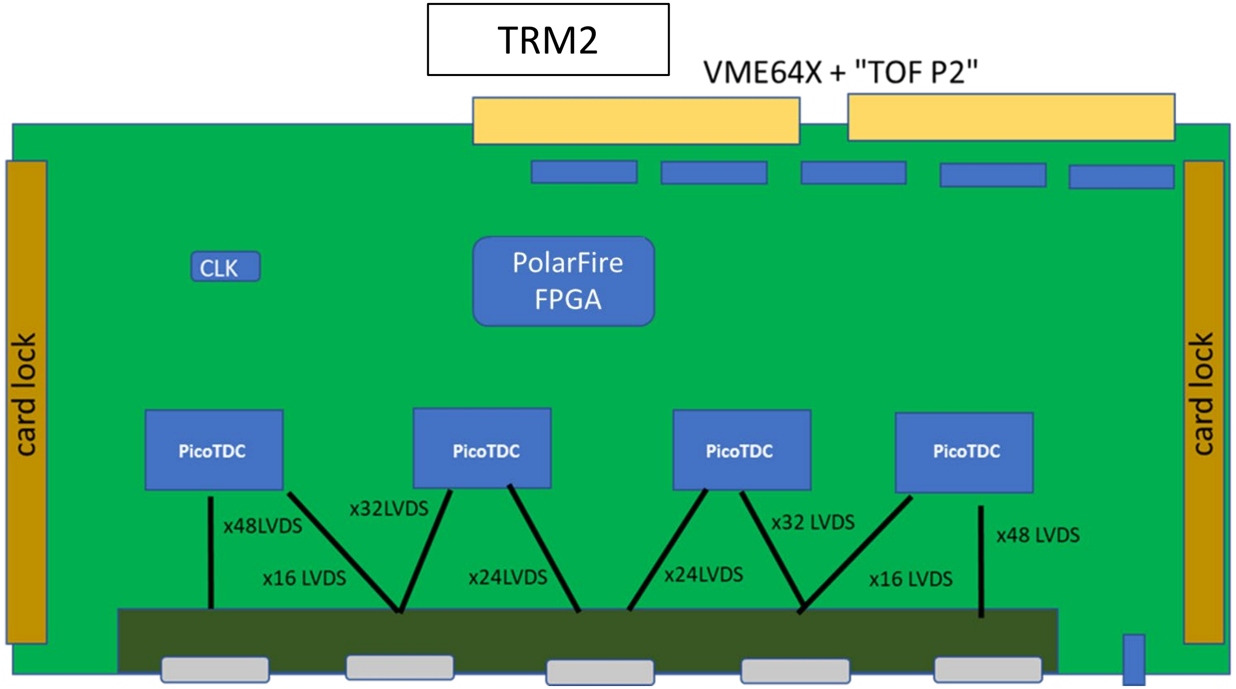
- Located at the **LHC** collider (CERN).
- Studying the **strong interactions** and the **Quark Gluon Plasma**, within relativistic heavy-ion collisions.
- Sophisticated **tracking** and **PID** systems.
- Before Run 3 (started in 2022): upgrade for **continuous readout**.

TOF detector: PID in the 0.3 GeV/c < p < 5 GeV/c range, (~**60 ps resolution**).



From TRM to TRM2 for time measurements in the TOF detector





The **TRM VME** card is the main element of the TOF readout system and it hosts:

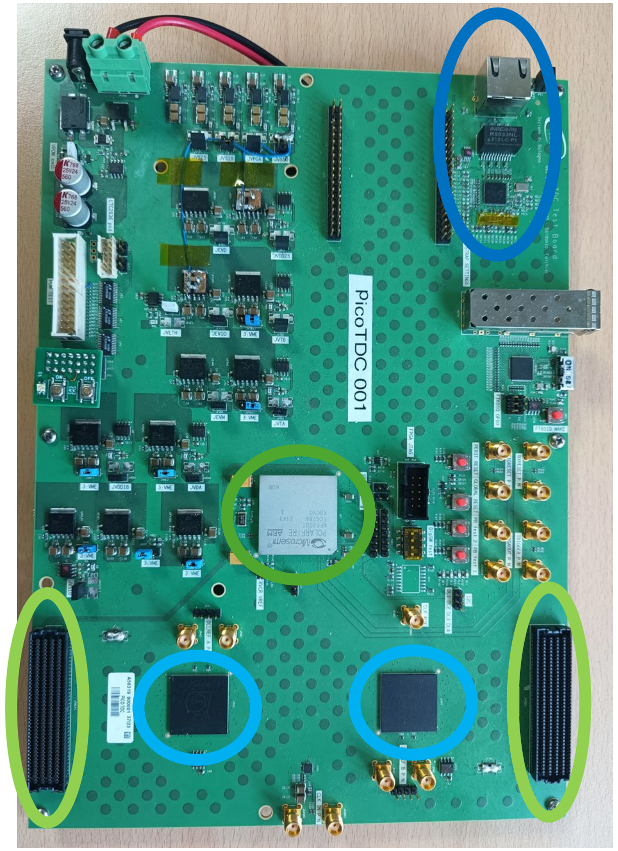
- An **Actel ProASIC FPGA** to manage the readout and board operations.
- **30 HPTDC** ASICs (24.4 ps LSB, 8 ch/chip) to provide time measurements.

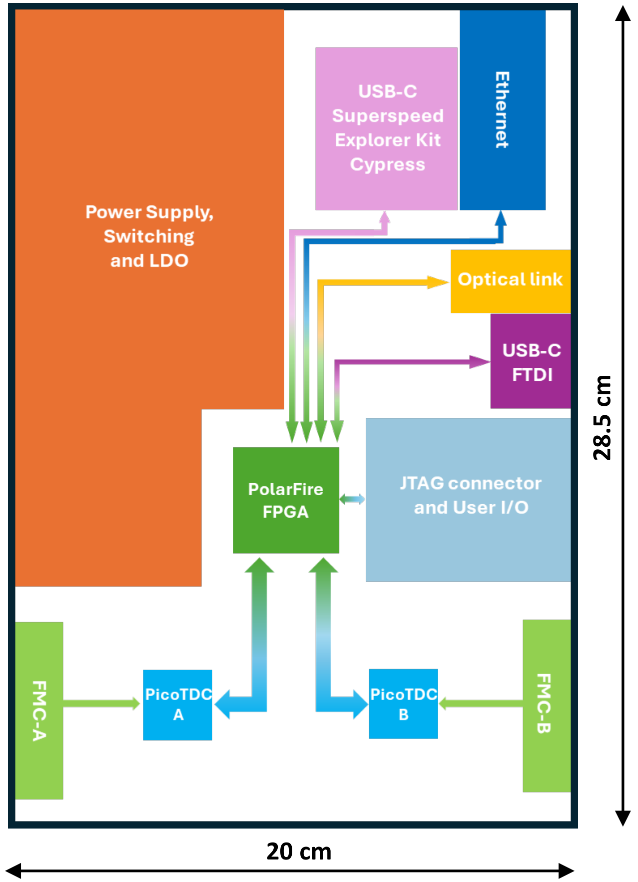
To replace damaged TRMs during LHC Run 4, a new **TRM2** project began, considering:

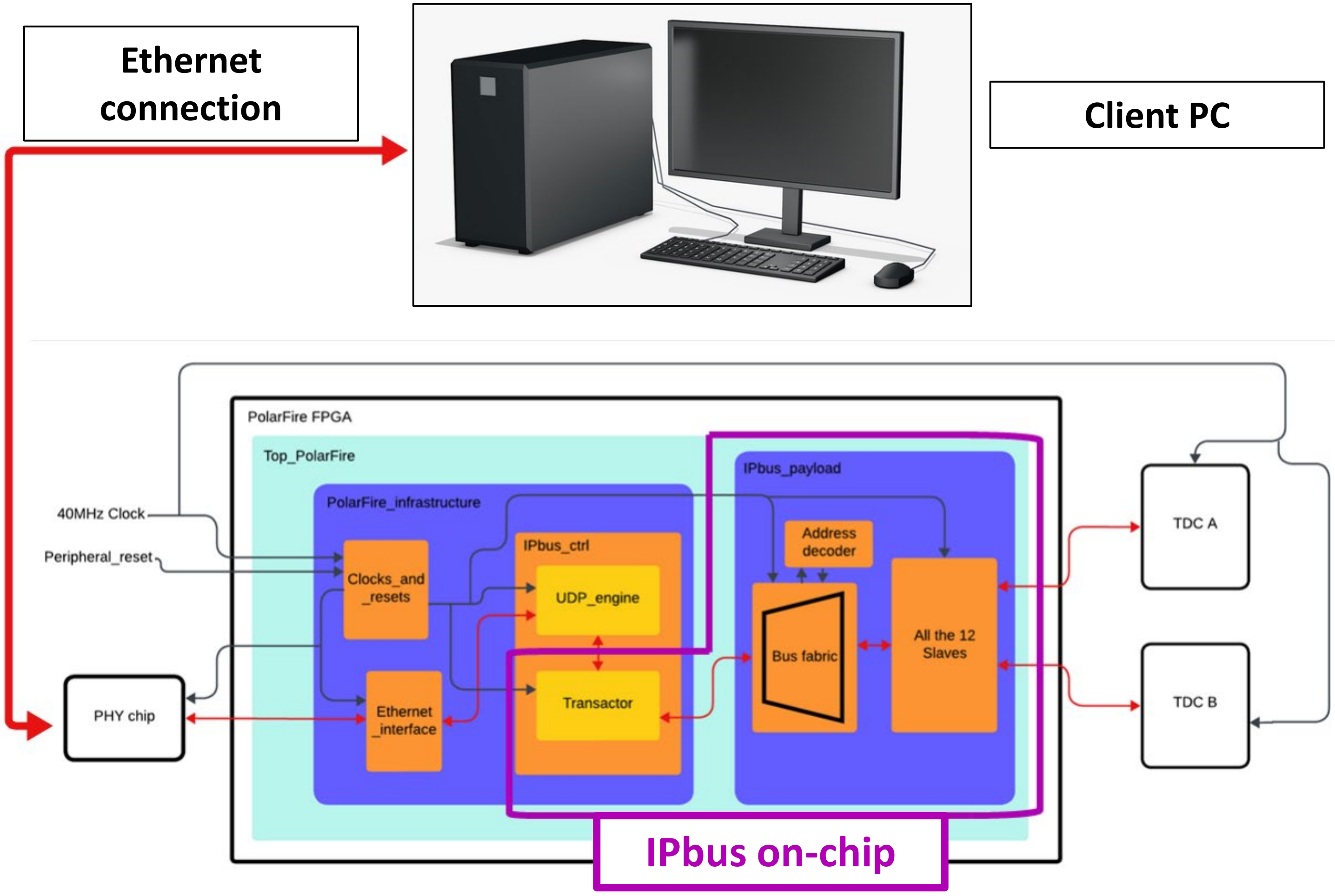
- A **PolarFire FPGA** to manage the readout and board operations.
- **4 PicoTDC** ASICs (12.2 ps or 3.05 ps LSB, 64 ch/chip) as successors of the HPTDCs.

The PicoTDC board

- **First step** for the **TRM2 development**.
- A **PolarFire FPGA** and **2 PicoTDCs**, connected to **2 high-density connectors** for plugging different FE boards and sensors.
- External links to the FPGA to control the board through USB and **Ethernet** communication protocols (an optical link is also provided).







My contribution: the firmware project

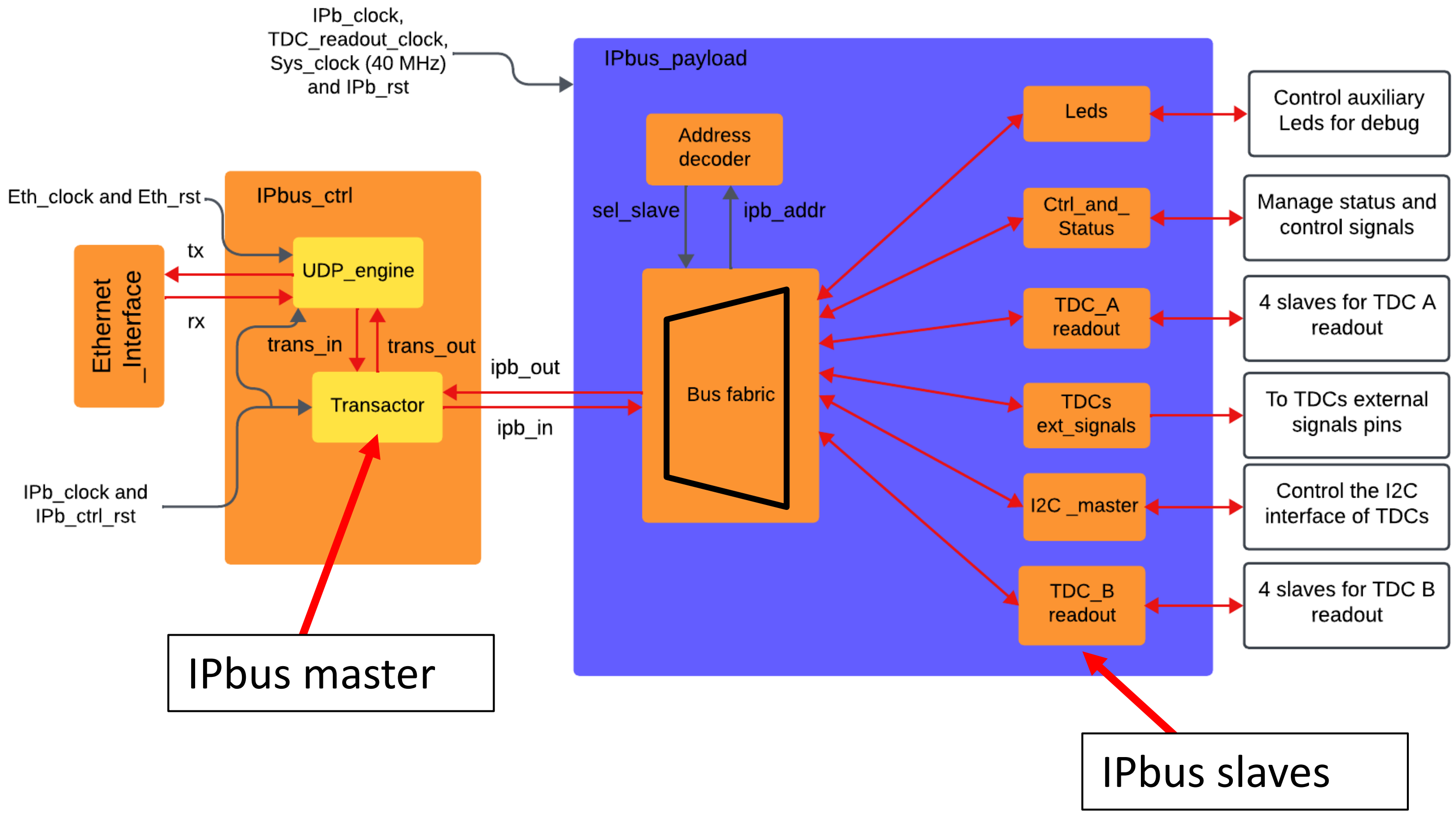
The firmware architecture is based on the **IPbus protocol** and provides a **1 Gb/s Ethernet solution.**

The **Ethernet_interface** is designed to manage the **Ethernet frames** in both directions.

The **UDP_engine** controls the **Transactor** through the information nested in the Ethernet frame.

The **Transactor**, the first element of the **IPbus on-chip**, controls the **IPbus read and write transactions** for the **IPbus slaves**.

IPbus on-chip



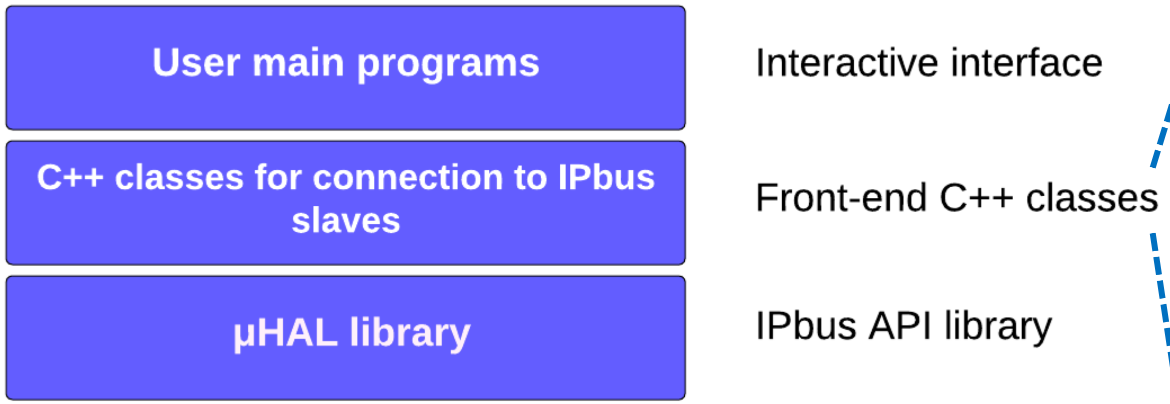
The **Transactor** acts as the **IPbus master** and controls the operations of all the **12 IPbus slaves** implemented.

The slave addressing is performed through the **Bus fabric** and the **Address decoder**.

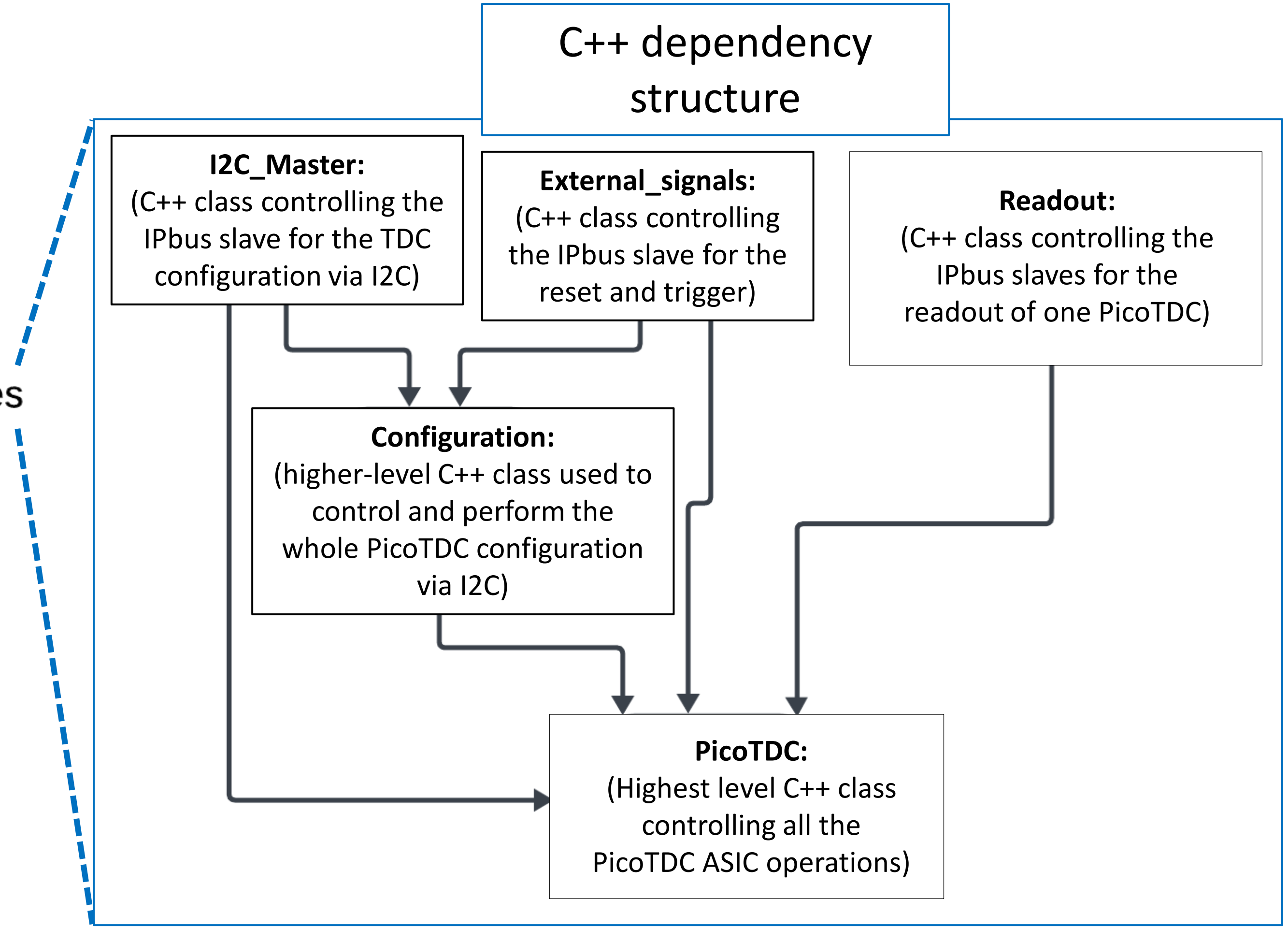
Almost all the slaves are used to **control the TDCs** for:

- **Configuration** via I2C connection
- Controlling **reset and trigger** signals
- **Readout**

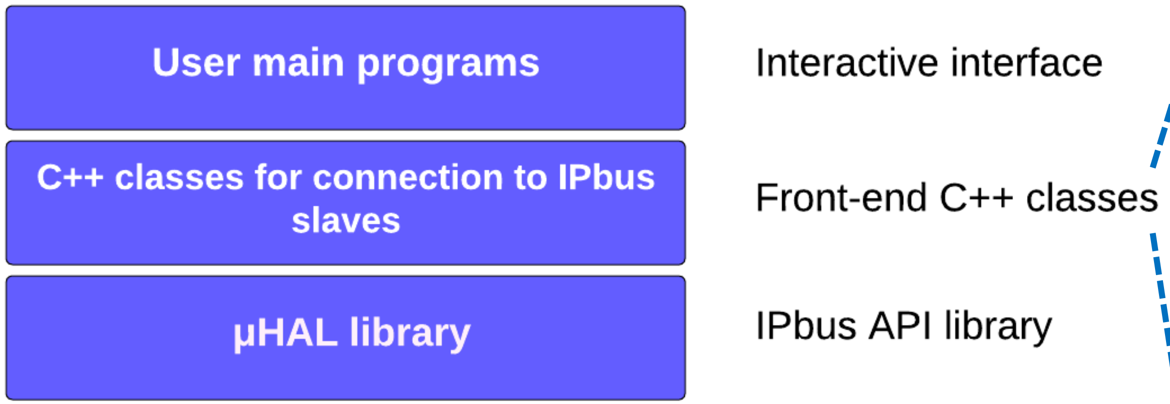
My contribution: the software project

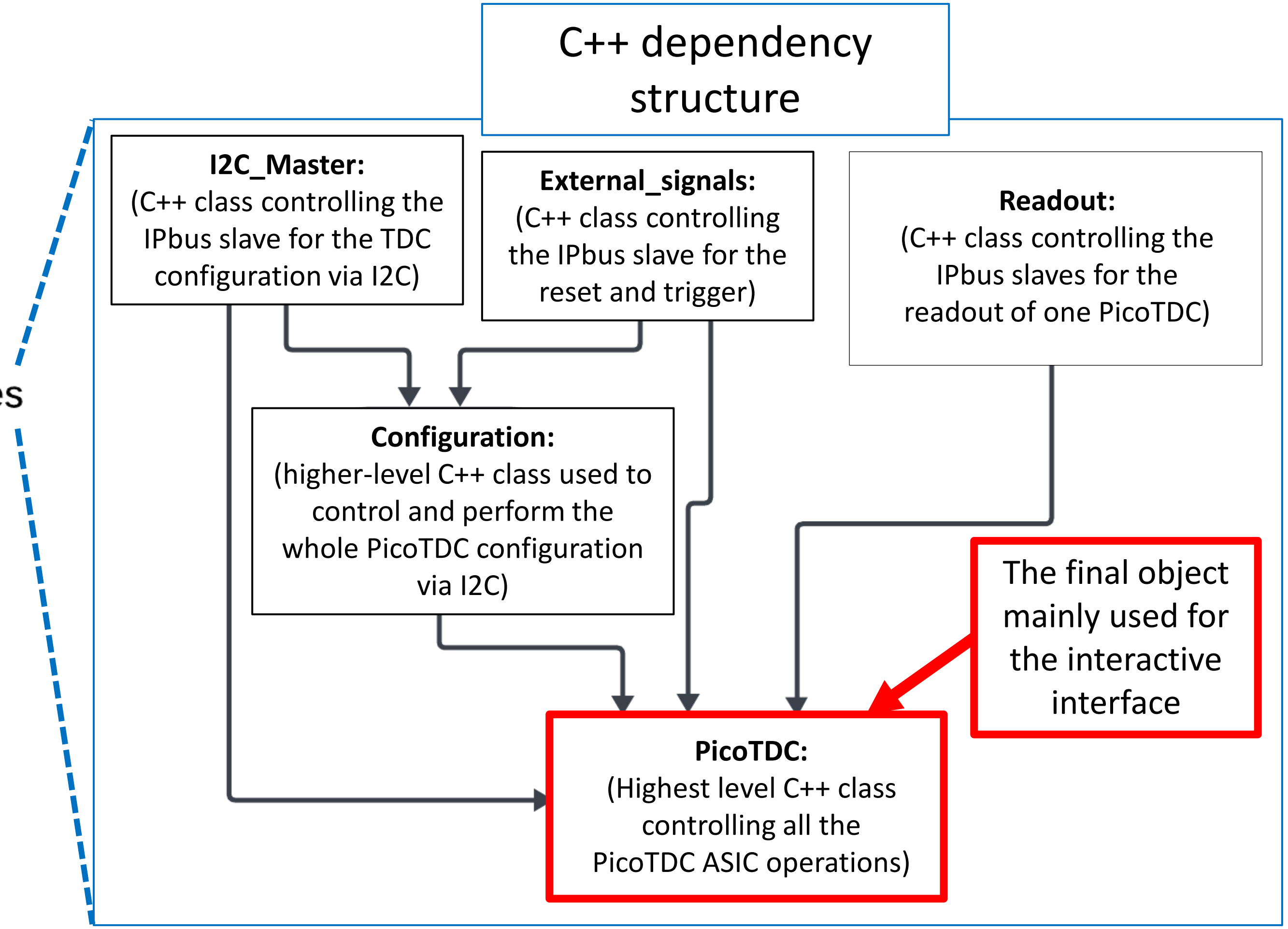


Considering a higher abstraction level from the bottom to the top, the **three-layer software structure** (C++ based) is shown, in which the first bottom layer is the **IPbus back-end library**.



My contribution:
the software project





Considering a higher abstraction level from the bottom to the top, the **three-layer software structure** (C++ based) is shown, in which the first bottom layer is the **IPbus back-end library**.

My contribution:
the software project

User main programs	Interactive interface
C++ classes for connection to IPbus slaves	Front-end C++ classes
µHAL library	IPbus API library

Considering a higher abstraction level from the bottom to the top, the **three-layer software structure** (C++ based) is shown, in which the first bottom layer is the **IPbus back-end library**.

The **final user interface** is designed to hide the system complexity and works through **prompt line commands.** It provides **two main user programs:**

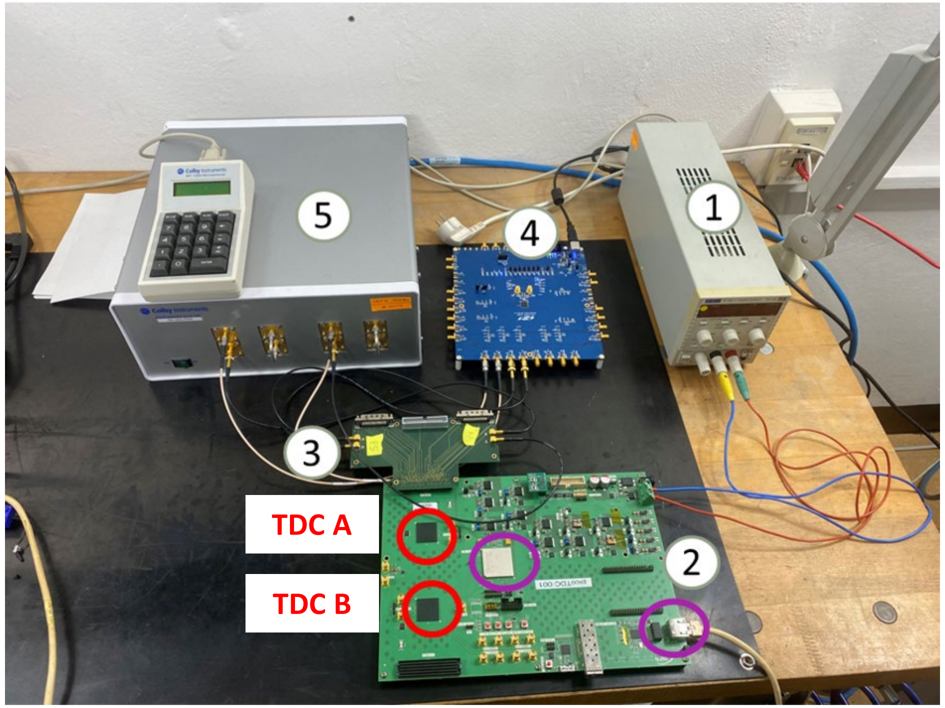
- ***PicoTOF***: user program to configure both the PicoTDCs.

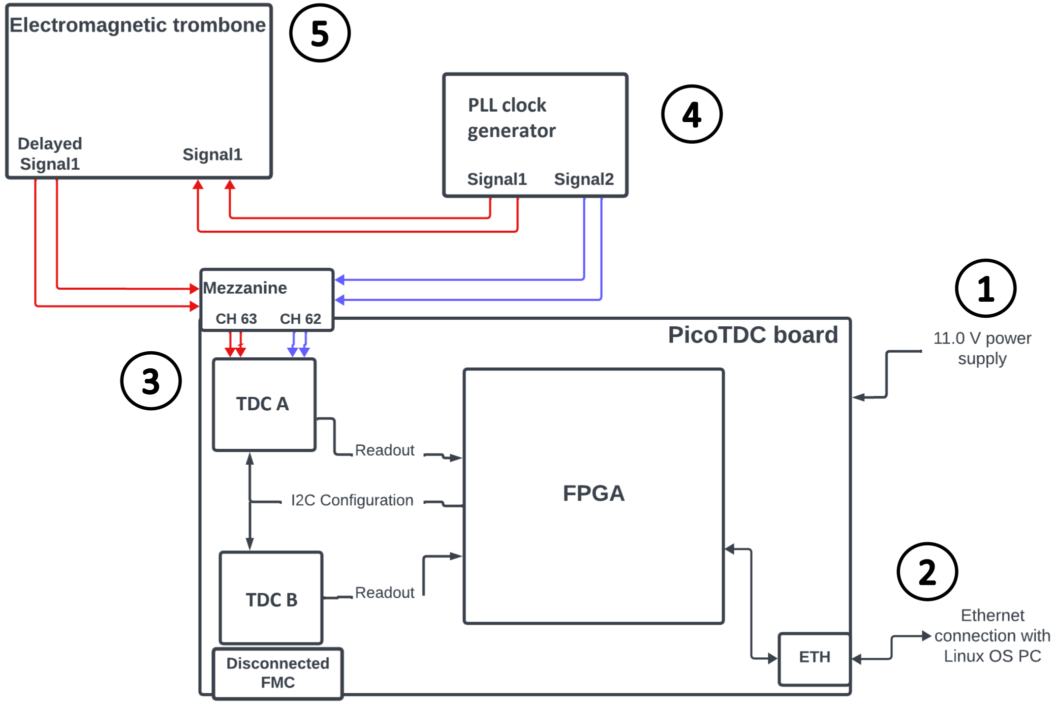
```
~$ ./PicoTOF --triggered --lw 400 360 --falling_edge n  --ch_en fine 62 63 --init A
```

- ***PicoRead***: user program for reading out data from the PicoTDC chips, considering a **trigger-based** readout.

```
$ ./PicoRead --chip A --events 10000000 --output file.ptdat
```

TDC resolution measurements





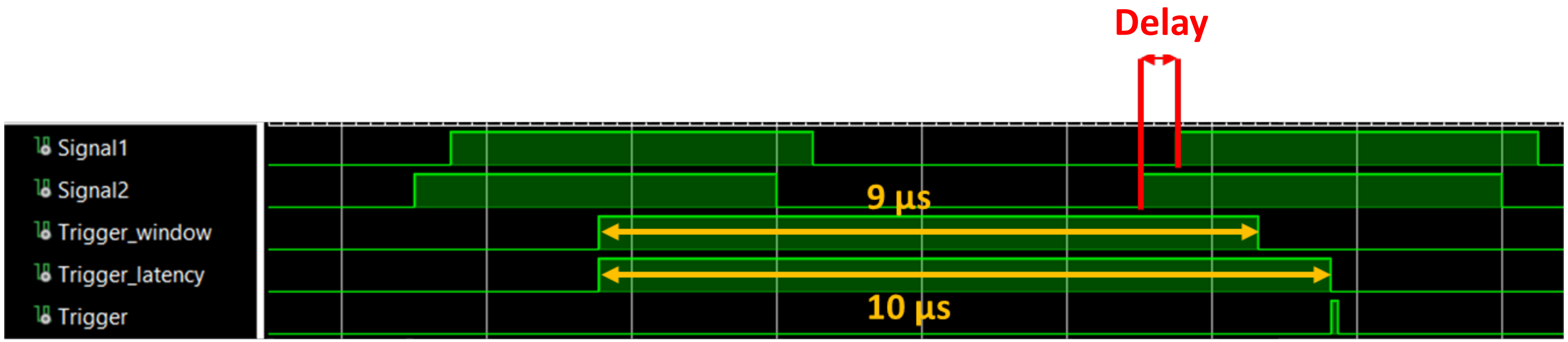
This is the setup used for the **TDC resolution estimation** performing a **two-channel time measurement,** considering **two clock signals** (100 kHz) and employing an **electromagnetic trombone** to shift one of the two signals by a **delay within the 0-600 ps range**.

Resolution measurements (2)

PicoTDC configuration:

```
:~$ ./PicoTOF --triggered --lw 400 360 --falling_edge n  --ch_en fine 62 63 --init A
```

- **Two channels** are enabled and they perform measurements only on the **rising edge**, considering the **3.05 ps finest binning** («falling_edge» and «ch_en»).
- **Triggered mode**, to have data only when a **software trigger** is supplied («triggered»).
- **Trigger latency** and **window** set to **10 µs** and **9 µs** respectively («lw» parameters expressed in 25 ns clock-cycle units).



Resolution measurements (2)

PicoTDC configuration:

```
~$ ./PicoTOF --triggered --lw 400 360 --falling_edge n  --ch_en fine 62 63 --init A
```

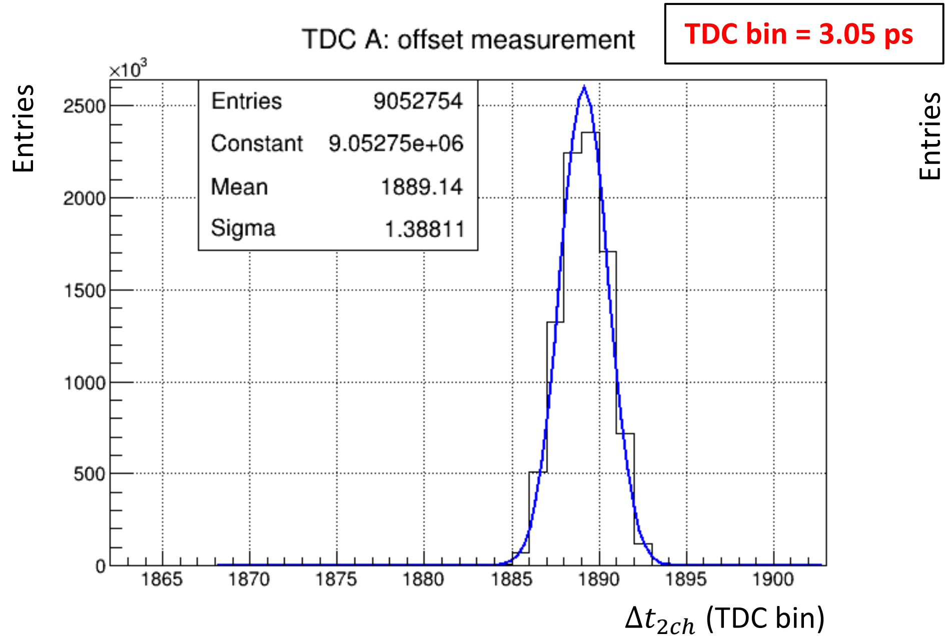
- **Two channels** are enabled and they perform measurements only on the **rising edge**, considering the **3.05 ps finest binning** («falling_edge» and «ch_en»).
- **Triggered mode**, to have data only when a **software trigger** is supplied («triggered»).
- **Trigger latency** and **window** set to **10 μs** and **9 μs** respectively («lw» parameters expressed in 25 ns clock-cycle units).

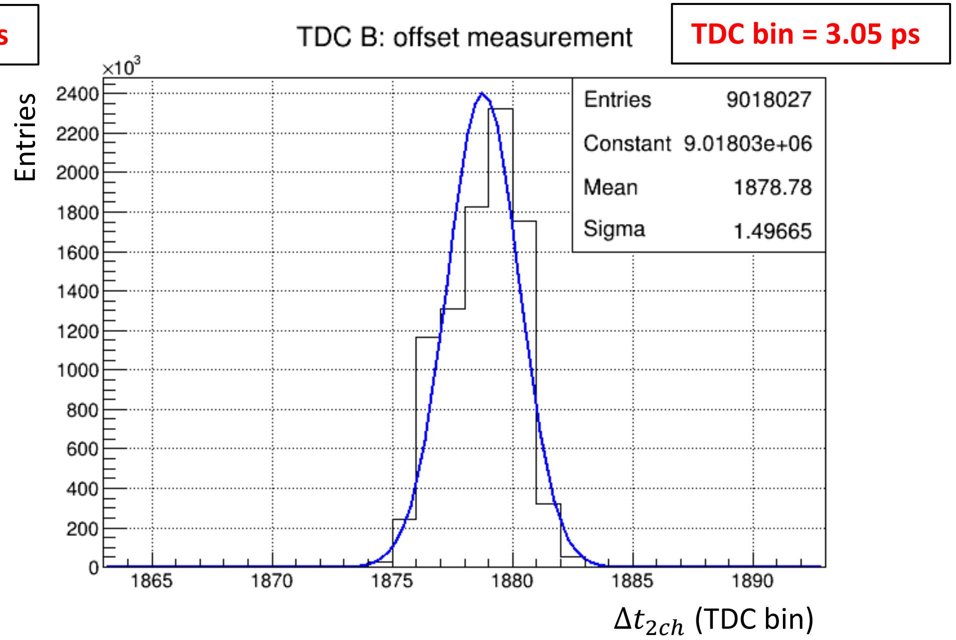
PicoTDC readout:

```
$ ./PicoRead --chip A --events 10000000 --output file.ptdat
```

- The **chip** is selected using the label A or B («chip»).
- Request for **10^7 triggered events** («events»).
- Saving all the read data from the selected chip in a **«ptdat» extension file** («output»).

Data analysis and results





These are the first measurements done for the **offset induced by the trombone** for both the TDCs, considering a **0 ps configured delay**.

Using a Gaussian fit, the values **1889.1 · 3.05 ps = 5761.9 ps** and **1878.8 · 3.05 ps = 5730.3 ps** are estimated respectively for TDCs A and B (**multiplying the measured mean by the TDC bin**).

Data analysis and results (2)

TDC A:		
$Delay_{exp}$ (ps)	$Delay_{meas}$ (ps)	$\sigma_{time(2ch)}$ (ps)
10	10.5	4.3
20	20.2	4.3
100	101.0	4.4
200	200.9	4.5
300	301.4	4.4
400	401.9	4.0
500	501.3	3.0
600	601.8	3.9

TDC B:		
$Delay_{exp}$ (ps)	$Delay_{meas}$ (ps)	$\sigma_{time(2ch)}$ (ps)
10	8.7	4.6
20	18.7	4.6
100	100.7	4.6
200	200.5	4.6
300	300.8	4.5
400	401.0	3.8
500	500.4	2.9
600	600.8	3.6

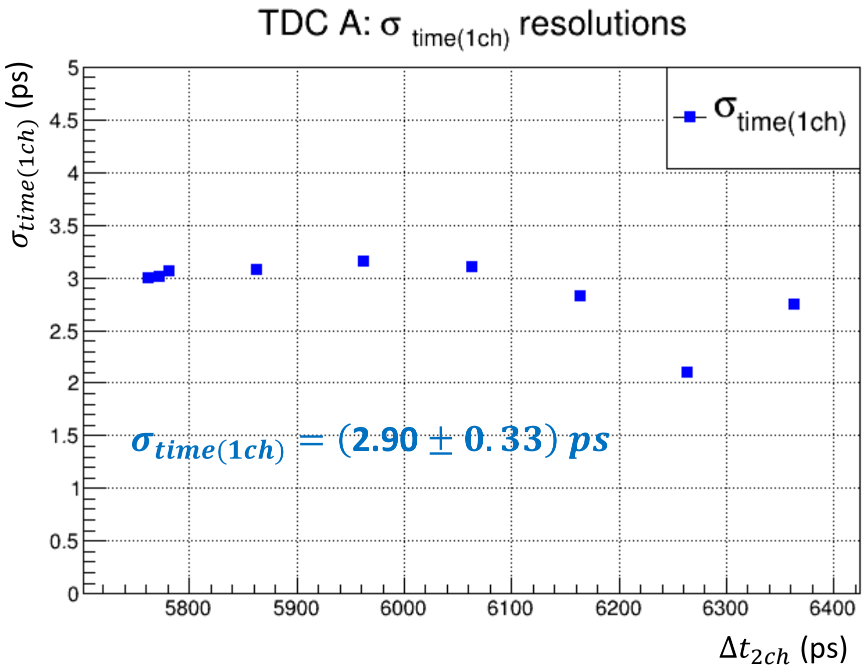
The analysis results for the estimated delays show **excellent agreement (within 2 ps)** with the expected delays:

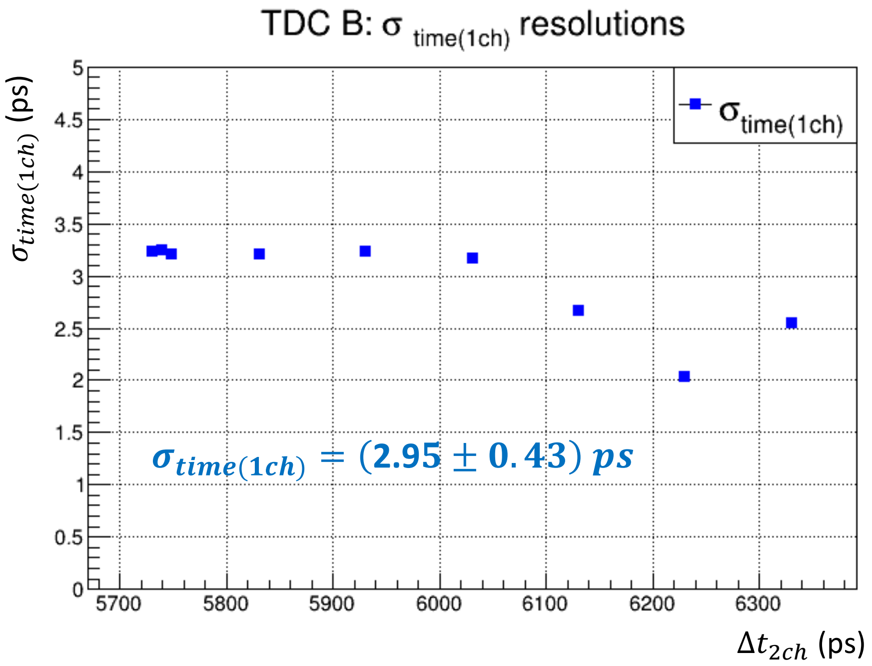
$$\mathbf{Delay_{meas}} = (\mathbf{Mean_{meas}} - \mathbf{Mean_{offset}}) \cdot \mathbf{3.05\ ps}$$

For **a 2 channels time measurement**, the $\boldsymbol{\sigma_{time(2ch)}}$ resolution values are estimated using:

$$\boldsymbol{\sigma_{time(2ch)}} = \mathbf{Sigma_{meas}} \cdot \mathbf{3.05\ ps}$$

Data analysis and results (3)





The $\boldsymbol{\sigma}_{\mathbf{time(1ch)}}$ **resolution** value for each dataset is estimated, using:

$$\boldsymbol{\sigma}_{\mathbf{time(1ch)}} = \frac{\boldsymbol{\sigma}_{\mathbf{time(2ch)}}}{\sqrt{\mathbf{2}}}$$

The **1-channel resolution** measured for both TDCs, **within a time interval of 600 ps**, is found considering the **mean** and the **standard deviation** for all 9 measurements.

Conclusions and outlook

Reliable operations and good performances of the board hardware:

1. Measured delays in agreement within 2 ps with the expected ones.
2. 1-channel TDC resolution of (2.95 ± 0.43) ps (maximum achievable resolution is $\mathrm{LSB}/\sqrt{12} = 0.88$ ps)
3. The DAQ system was reliable during all 36 hours of measurements.

The PicoTDC board and the DAQ system:

- Provide a **good test environment** for the **TRM2** development.
- Are **good DAQ resources** for **sensor tests** within test beams or laboratory analysis.

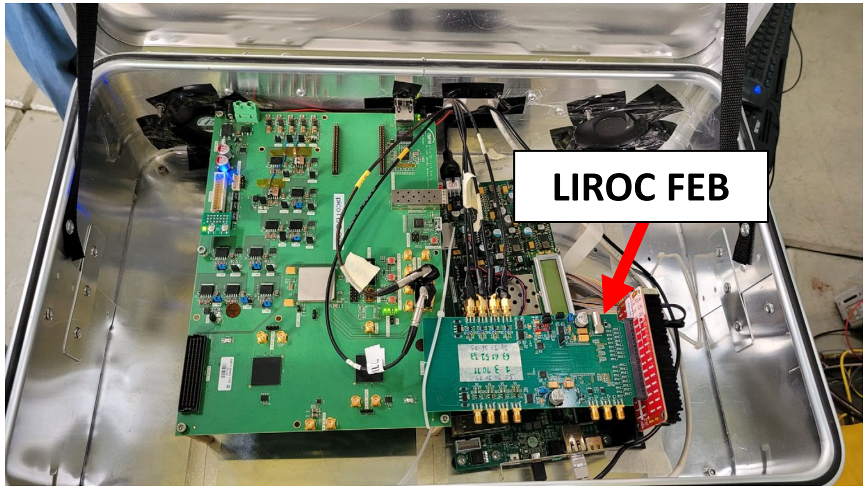
Conclusions and outlook

Reliable operations and good performances of the board hardware:

1. Measured delays in agreement within 2 ps with the expected ones.
2. 1-channel TDC resolution of (2.95 ± 0.43) ps (maximum achievable resolution is $\mathrm{LSB}/\sqrt{12} = 0.88$ ps)
3. The DAQ system was reliable during all 36 hours of measurements.

The PicoTDC board and the DAQ system:

- Provide a **good test environment** for the **TRM2** development.
- Are **good DAQ resources** for **sensor tests** within test beams or laboratory analysis.

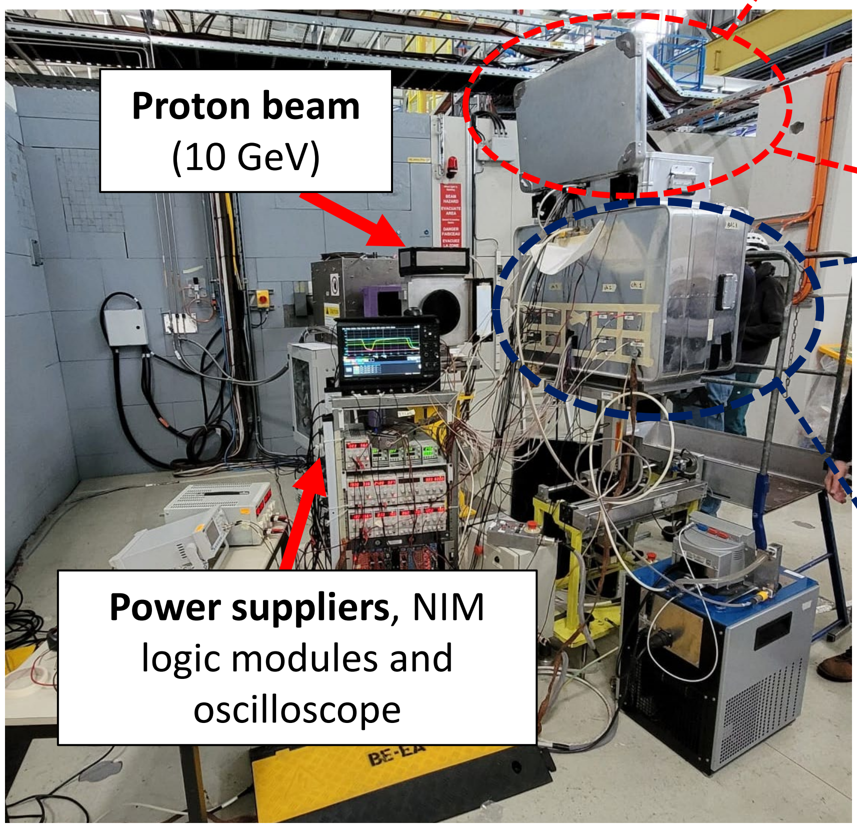


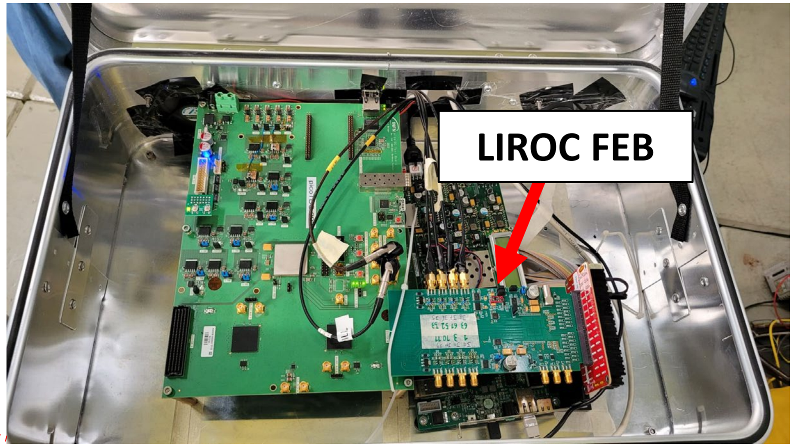
At the end of April 2024, **I used the DAQ system and the board in a test beam at CERN PS**, for the ALICE3 group of Bologna.

The board was connected in a complete DAQ chain getting signals from SIPM and LGAD sensors, the devices under test.

The system was stable and reliable throughout the test beam!

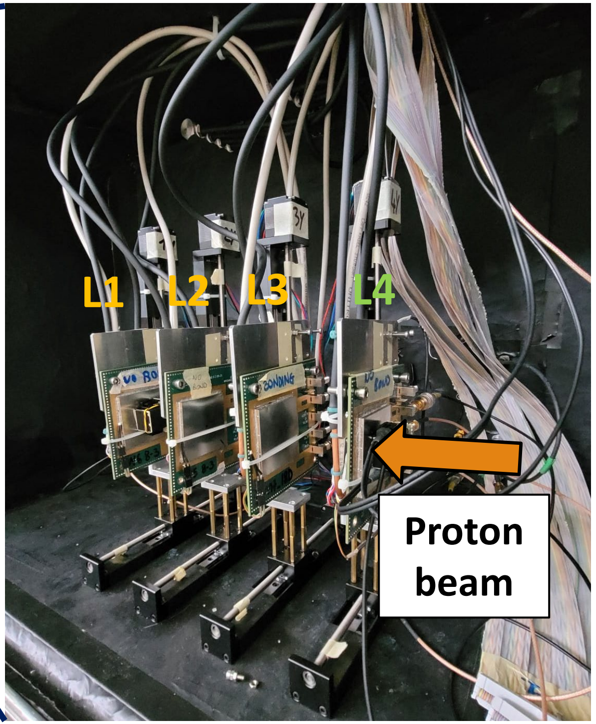
Test beam April 2024 (PS at CERN)





Readout electronics box including:

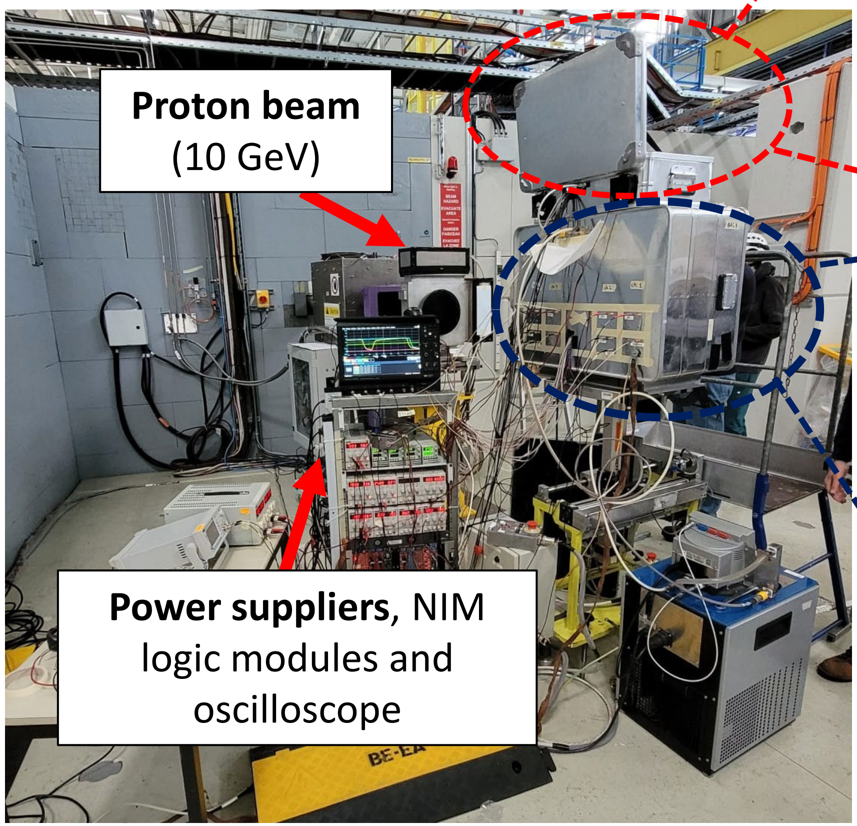
- **The DAQ chain**: the **PicoTDC board** and the **LIROC FEB** (amplification + discrimination)

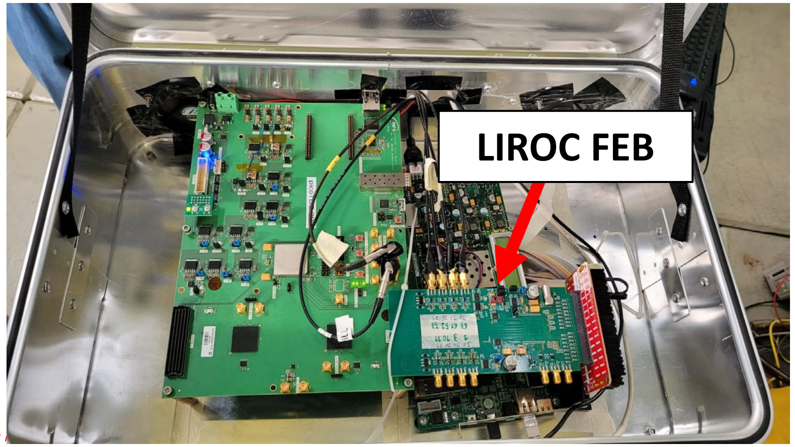


Sensors box including **four layers** (**LGADs** and **SiPMs**):

- **L4:** LGAD signal used as **the trigger signal**
- **L3, L2, L1:** sensors connected to **TDC input channels**

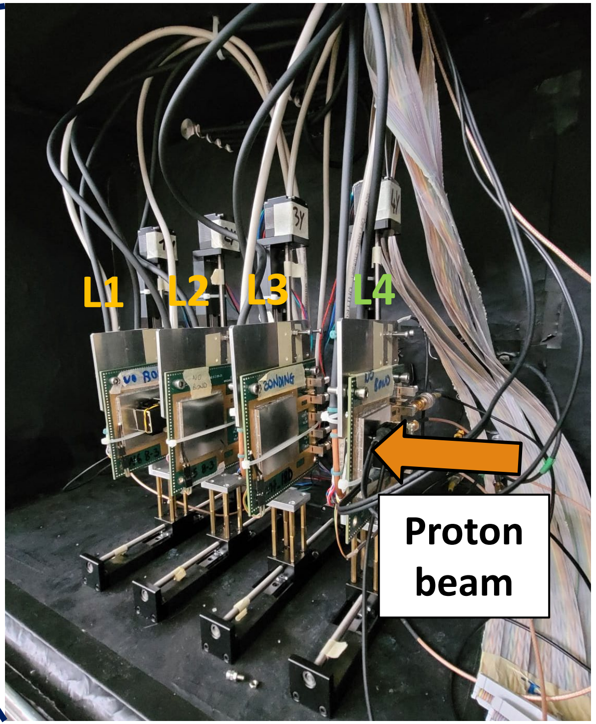
Test beam April 2024 (PS at CERN)





Readout electronics box including:

- **The DAQ chain**: the **PicoTDC board** and the **LIROC FEB** (amplification + discrimination)



At TB of June/July 2024, **five layers** were employed and **both the TDCs on board** were used!

Sensors box including **four layers** (**LGADs** and **SiPMs**):

- **L4:** LGAD signal used as **the trigger signal**
- **L3, L2, L1:** sensors connected to **TDC input channels**

April 2024 TB team

June/July 2024 TB team

Thank you all!

April 2024 TB team

June/July 2024 TB team

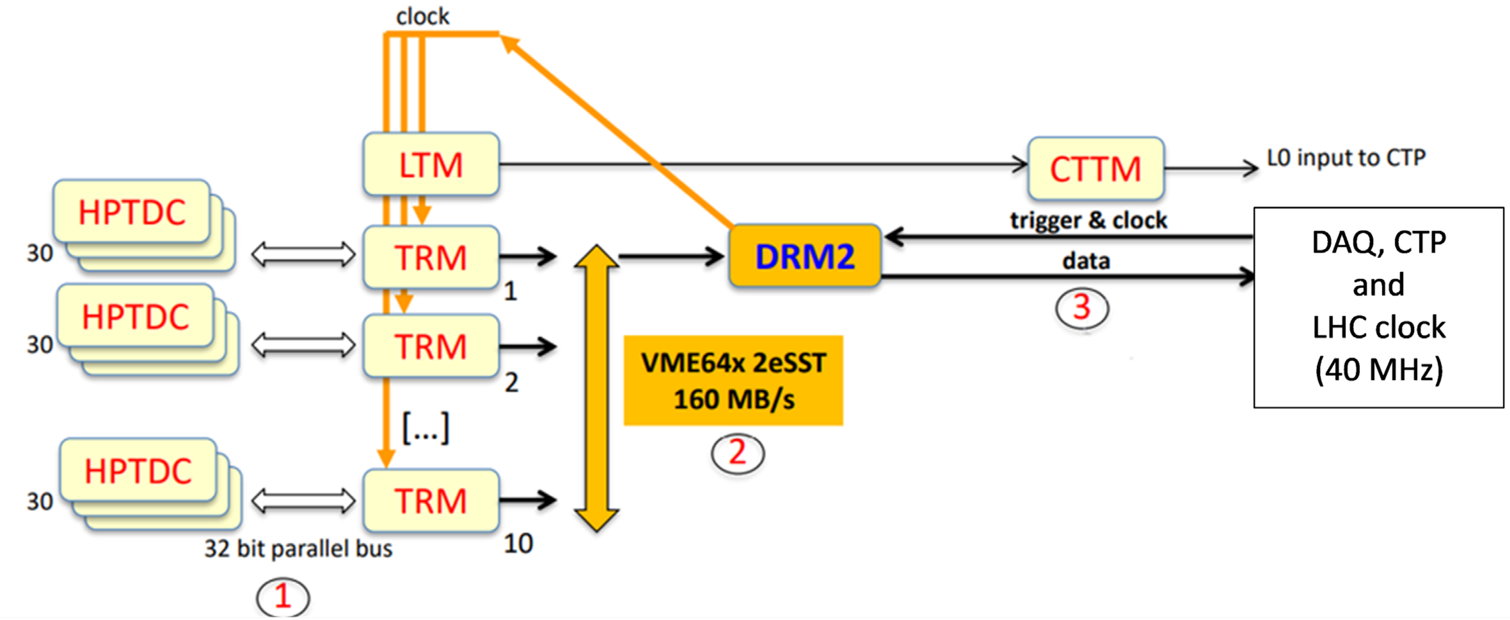
Thank you all! →

Thanks to the entire Picoteam for the work done in developing both the board and the DAQ system!

Thank you for your attention!

Backup slides

Electronics readout system of the TOF detector

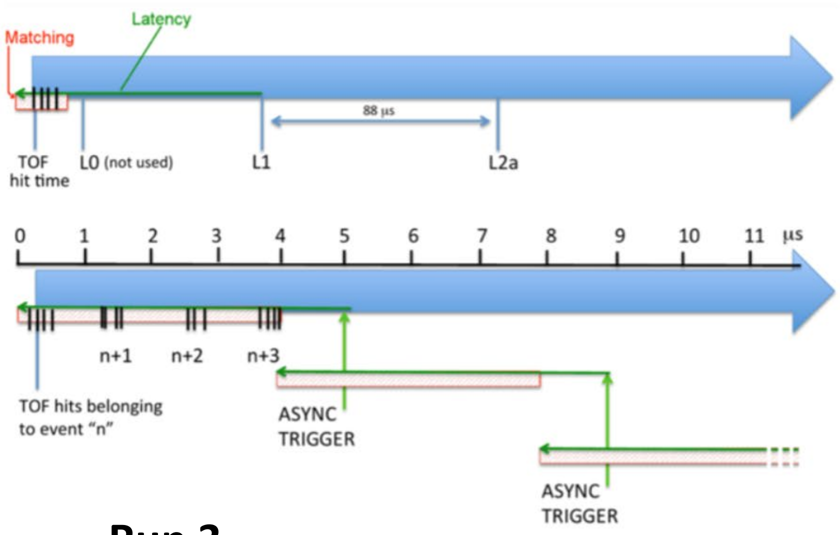


This is the electronics readout system of the TOF detector, employing:

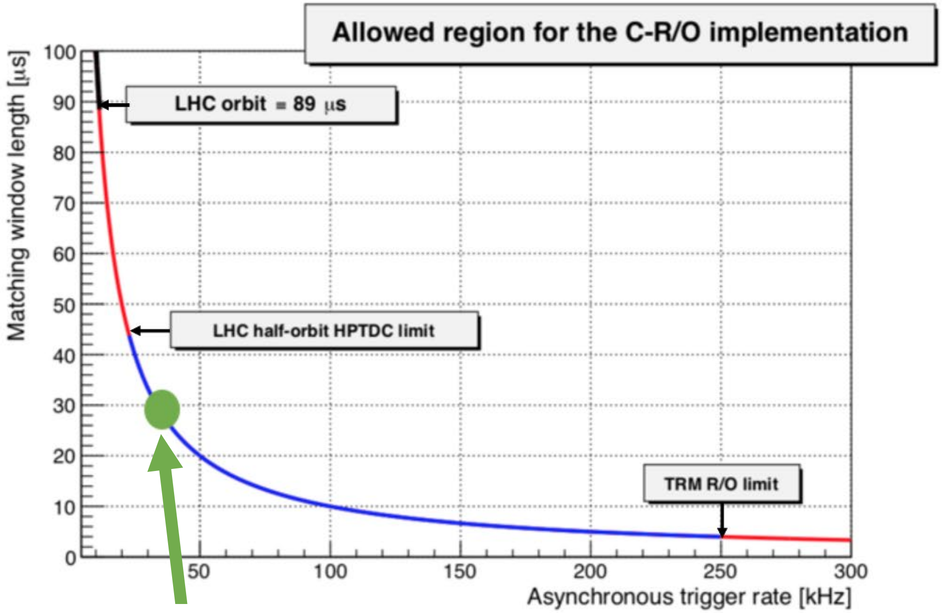
- The **TRM** cards to provide timing measurements.
- The **DRM2** card as the VME master to perform the TRMs readout.
- The **LTM** as an independent interface for trigger purposes.

TOF detector continuous readout

Run 1-Run 2

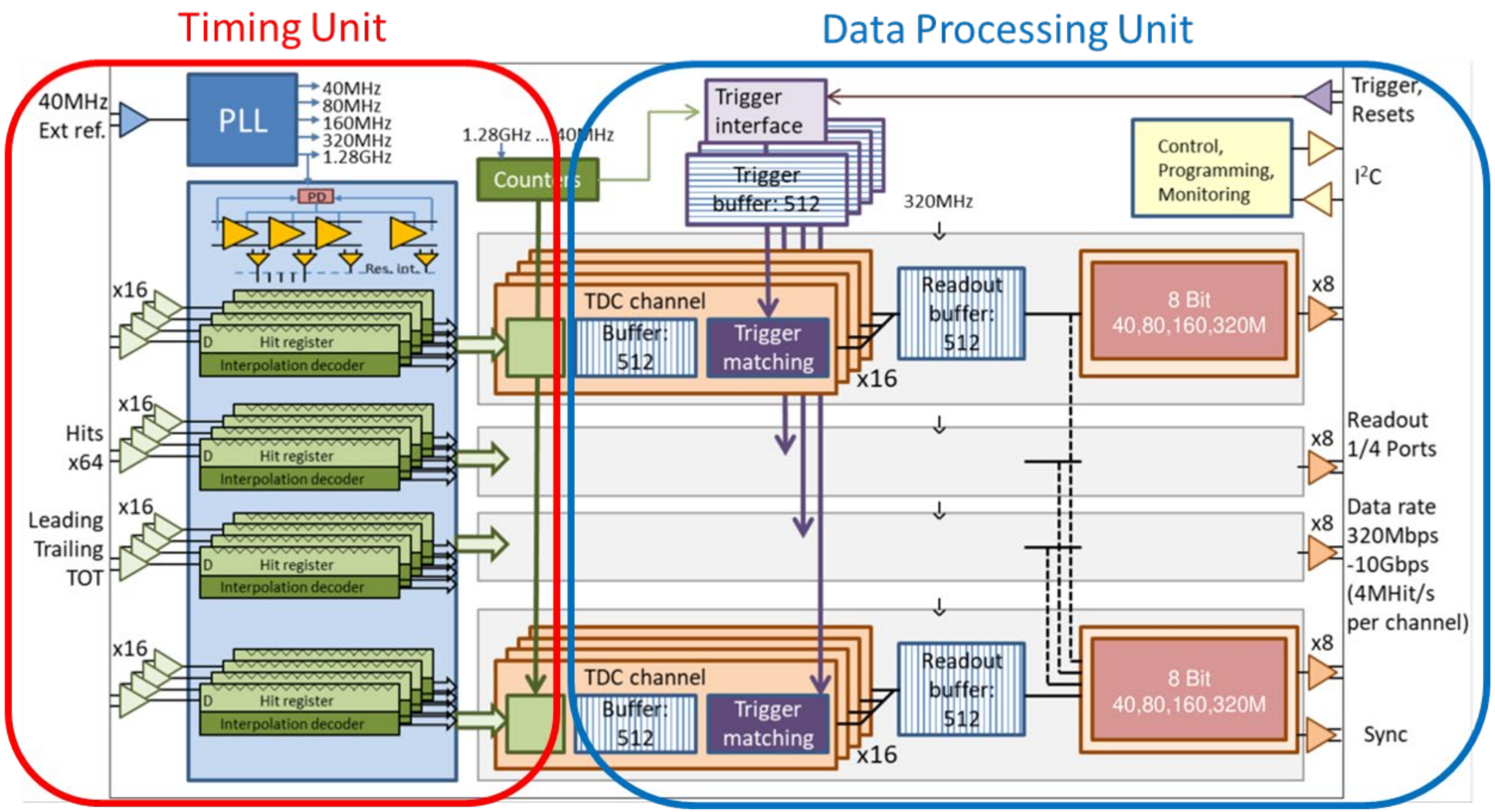


Run 3

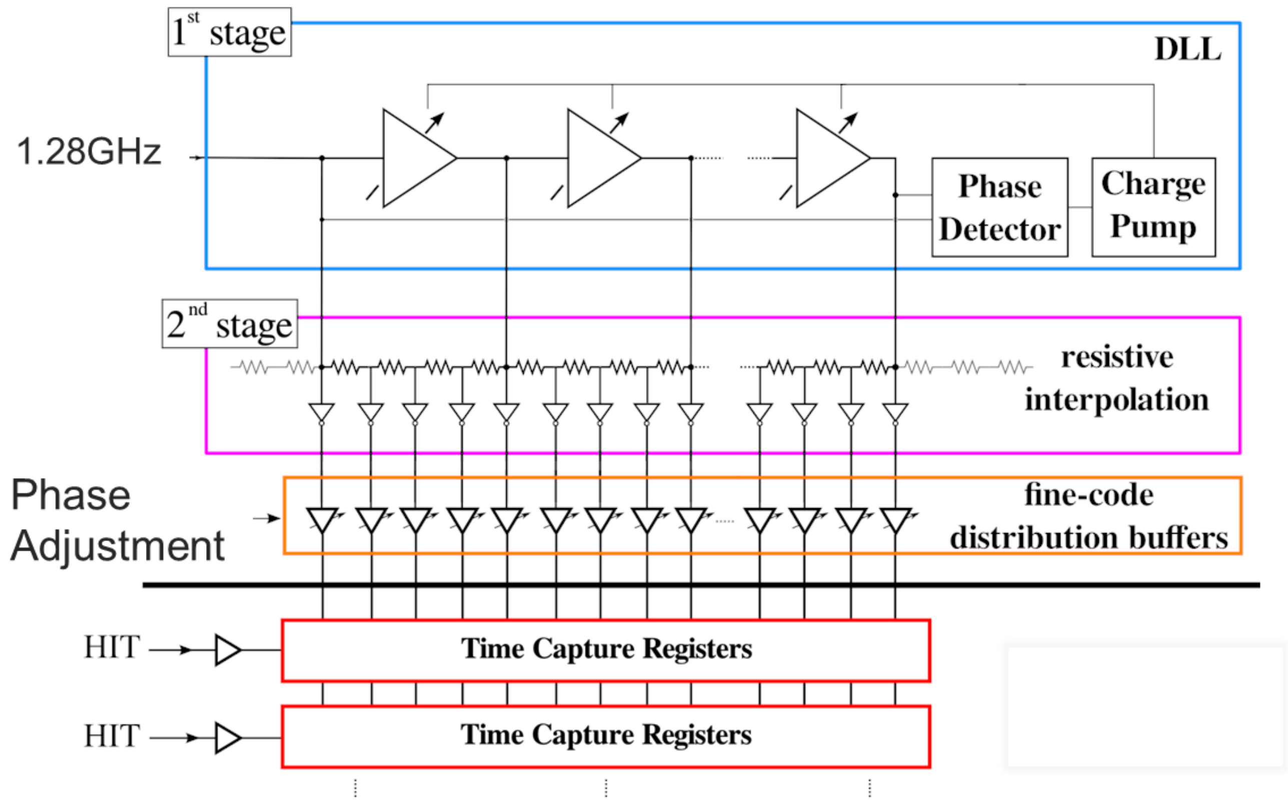


Run 3 trigger working point

PicoTDC architecture



2-stage DLL used within the PicoTDC

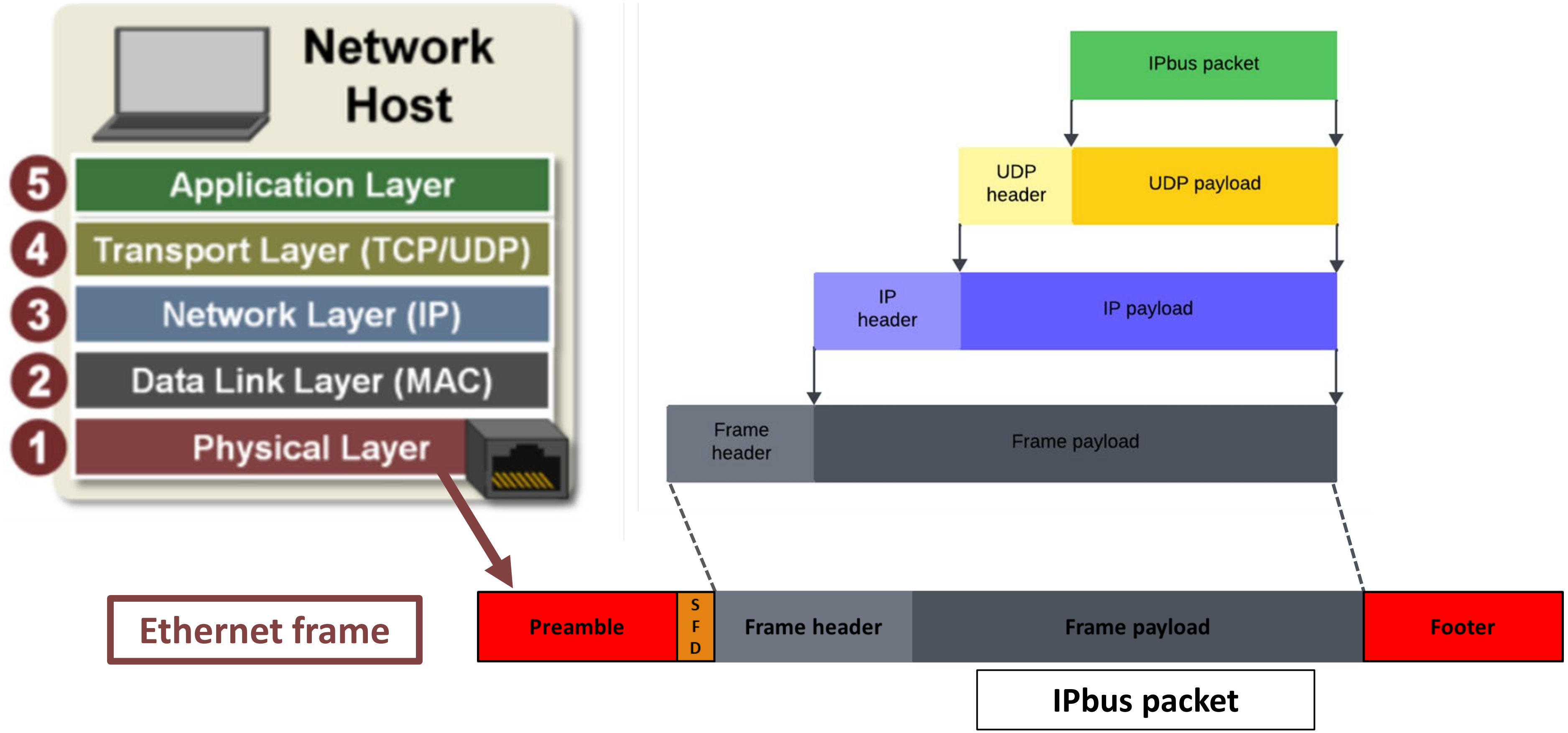


12.2ps delay

3.05ps delay

32 steps each with 0.6ps resolution

UDP/IP model



Each node in an Ethernet network of devices provides this layer structure, and each layer has its communication protocol. **The final Ethernet frame, sent on a physical connection, contains each layer's information**.

PicoTOF «init» option for TDCs:

```
if (vm.count("init")){

  const string chip = vm["init"].as<string>();
  //Selection of the chip that must be initialized
  if(chip == "A"){
          TDCA.Initialize_Ext(1, my_block); //init
          my_readA.Reset_FIFOS(4);//reset of all the readout fifos
          }
  else if(chip == "B"){
          TDCB.Initialize_Ext(2, my_block); //init
          my_readB.Reset_FIFOS(4);//reset of all the readout fifos
          }
  else if(chip == "AandB"){
          TDCA.Initialize_Ext(1, my_block);//init
          my_readA.Reset_FIFOS(4);   //reset of all the readout fifos
          TDCB.Initialize_Ext(2, my_block);//init
          my_readB.Reset_FIFOS(4);//reset of all the readout fifos
          }
  else
      cout<<"The only options allowed are A,B and A&B, skipping\n";

}
```

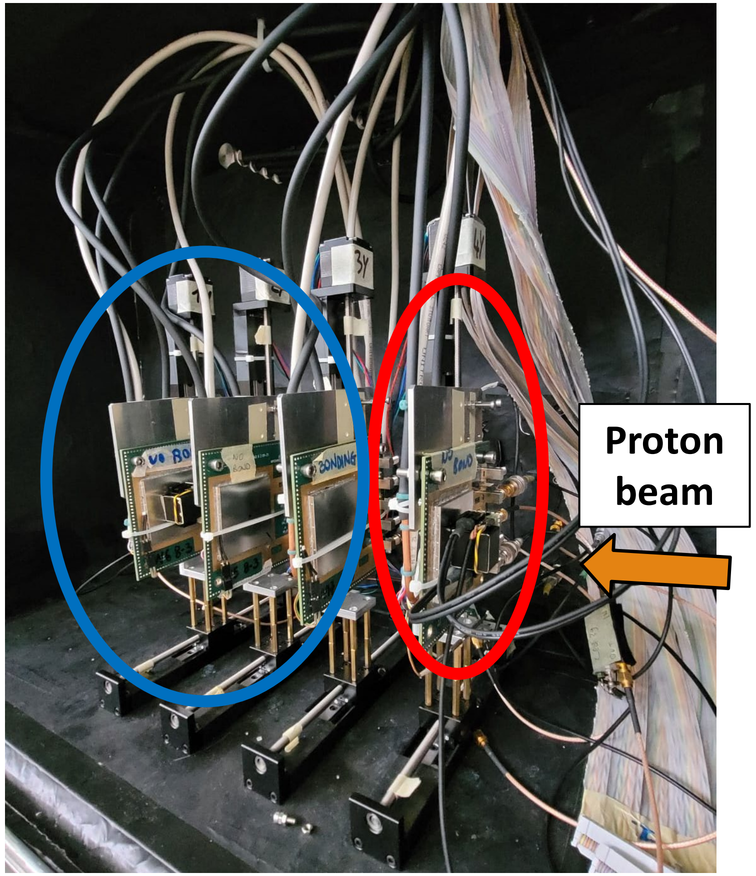
PicoTOF main loop for trigger-based readout:

```
while (iev < opt.events) { //Number of requested events (opt.events)
  sigMgr.SetOneImpulseTrigger(trigger_lane);   // trigger lane defined by opt.chip
                                               // Send a software trigger to the selected chip
  buff = ptread.Read_One_FIFO(0);   // Read  data

  outb[0]=0x12345678;              // event header
  outb[1]=iev;                     // event id
  outb[2]=0x0;                     // separator
  outb[3]=buff.size()*4;           // data buffer syze

  if ((buff.size()+4)>MAX_OUT_BUF) {
    std::cout<<"Too many data from FIFO per event, increase output buffer"<<std::endl;
    exit(EXIT_FAILURE);
  } else {
    // in future here we may want to insert a chip header....
    int i=0;
    for (uint32_t word : buff) {outb[4+i]=word; i++;}
  }
  fout.write((char*)&outb,outb[3]+HEADER_WORDS*4); //write the data within an fout ofstream
                                                   // the fout directory is defined by opt.output
```

Test beam April 2024 (PS at CERN)

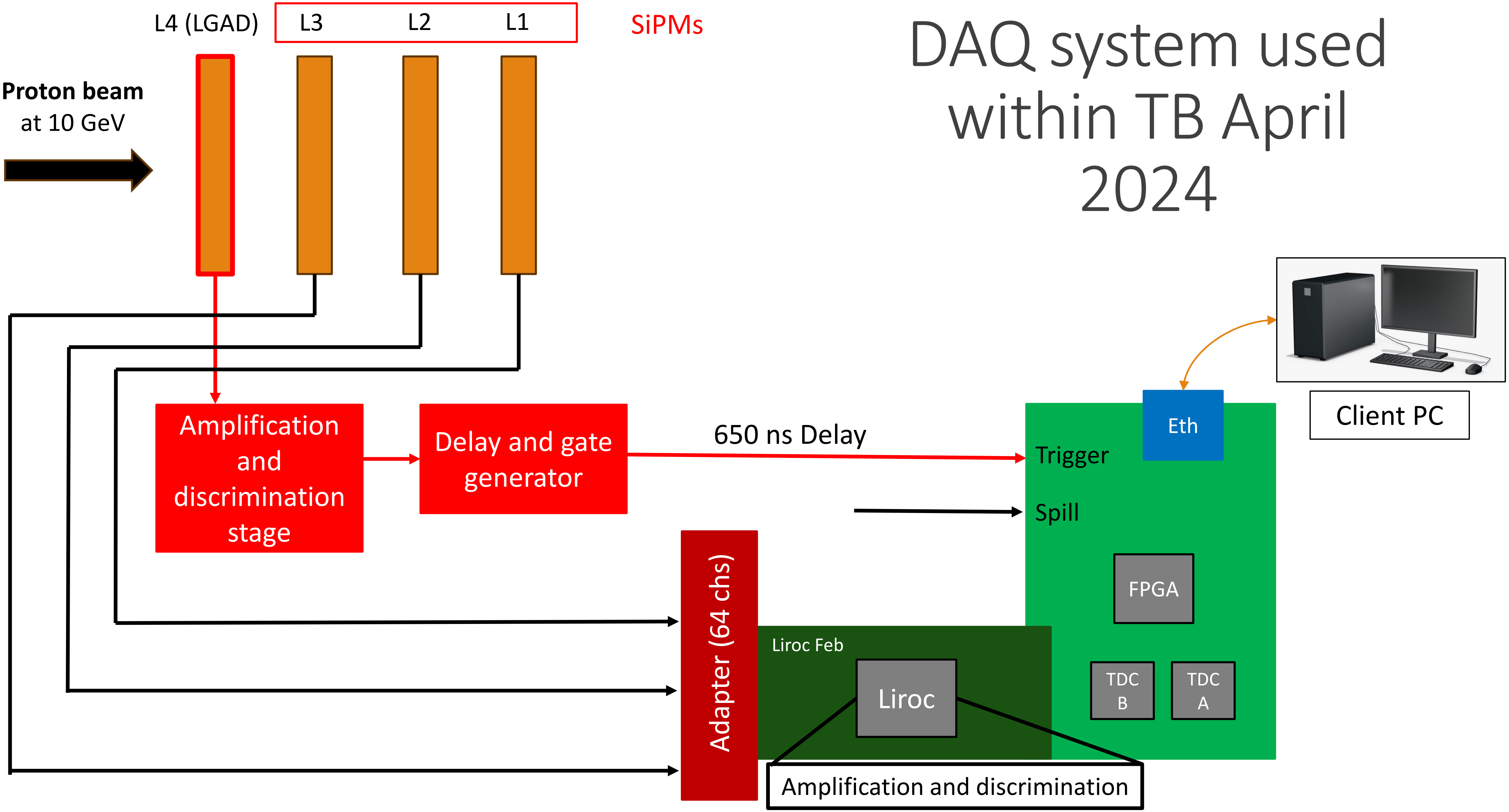


The **sensors box** included four layers for different sensors, among LGADs and SiPMs :

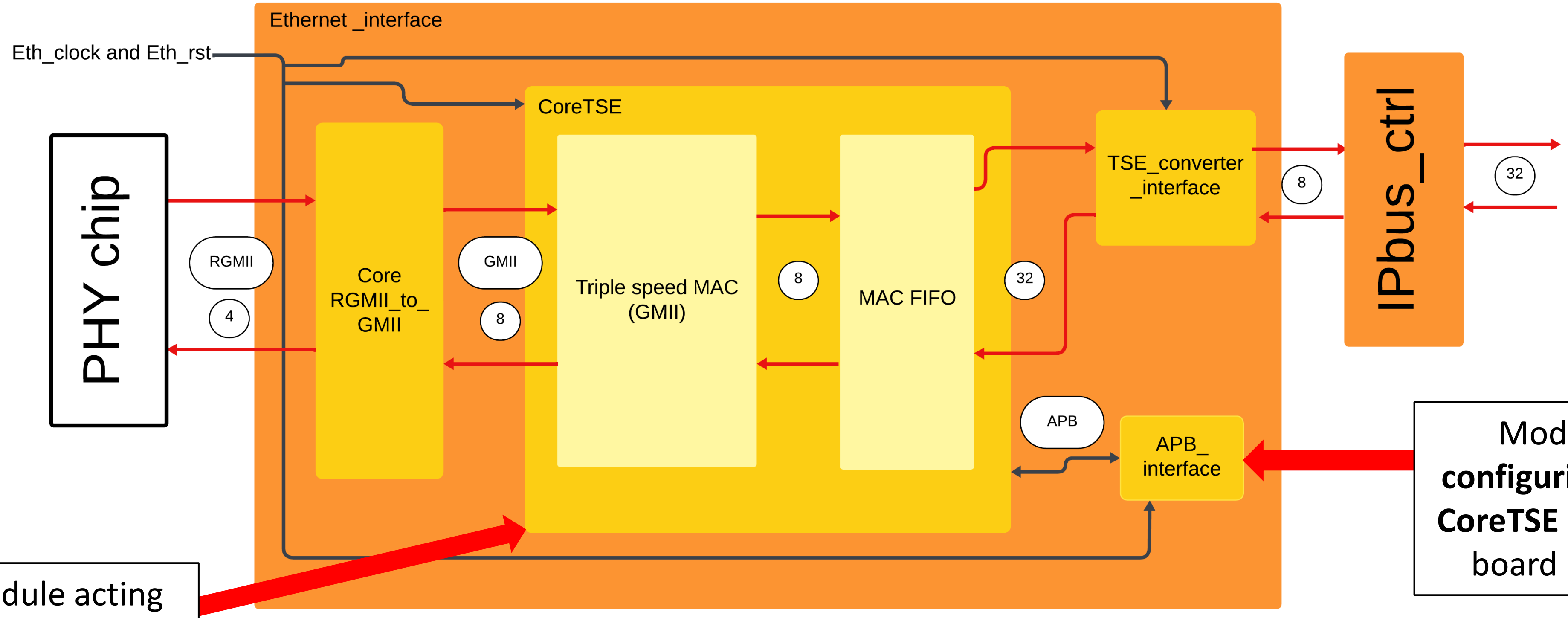
- **L4 (trigger)**: LGAD for a clean signal.
- **Other 3 layers**: sensors connected to TDC input channels.

A trigger latency of 650 ns and a matching window of 250 ns were configured for one PicoTDC readout.

DAQ system used within TB April 2024



The Ethernet_interface module



Module acting as the **Link Layer**

Module **configuring the CoreTSE** at each board reset