Photon Detection System (PDS) @ ProtoDUNE II

Photon Detection System and Trigger Overview

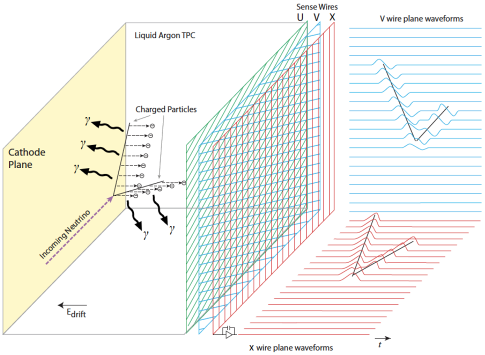
Manuel Arroyave[1]

[1] Fermilab

October 17, 2024

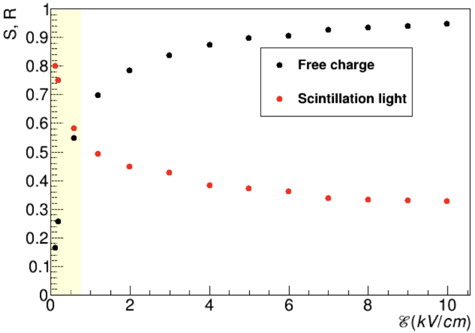
Purpose of the Photon Detection System (PDS) in DUNE

t_0 + TPC self-trigger capability



Diwan et al, 2016

Calorimetry



Marinho et al, 2022

Our Aim

We aim to develop a feasible plan for the PDS and DAQ to collaborate on enhancing the TPC self-trigger capability. This development involves several constraints; here are a few examples:

For DAQ

- TPs/local triggers should fit in the bandwidth/processing range of the DAQ; otherwise, they must be filtered/prescaled
- In case this is needed, we must provide information about the loss by using this filter/prescale
- The existing objects TPs TAs from PDS need to fit in the existing classes
- The proposed processes must be scalable for DUNE
- A large portion of the system should be testable during the NP02 run

Our Aim

We aim to develop a feasible plan for the PDS and DAQ to collaborate on enhancing the TPC self-trigger capability. This development involves several constraints; here are a few examples:

For PDS

- PDS is based on a 4-pi geometry
- We aim to trigger reasonable signals at ProtoDUNE, e.g., beam/atmospheric muons, while developing triggers for atmospheric/cosmic neutrinos for the FD
- Distinction between different events is desirable even if we do not use those for a trigger, e.g., discrimination between different energy ranges will greatly improve the trigger capabilities.

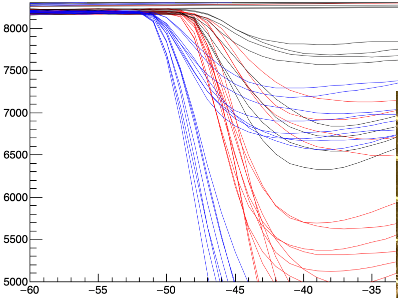
Status of the trigger chain of PDS

We HAVE a Firmware stable with TPs ready to be deployed

Trigger Primitives are the input to generate High-Level information to create TAs or TCs.
To develop algorithms for the DAQ-Trigger system, we must first understand the scintillation light signature patterns that help differentiate different events. A solid plan, backed by data and simulations, is essential for using these signatures effectively.

By the third quarter of 2024, this is the status of the PDS trigger features:

- Full characterization of the self-trigger algorithm in DAPHNE with high effitiency at 1.5 pe
- TriggerPrimitives: calculated in DAPHNE; we are characterizing the quality of the firmware for deployment during the following weeks
- TriggerActivity (clusters)
- TriggerCandidate (trigger)


8000
7500
7000
6500
6000
5500
5000
−60
−55
−50
−45
−40
−35

Terminology[1]

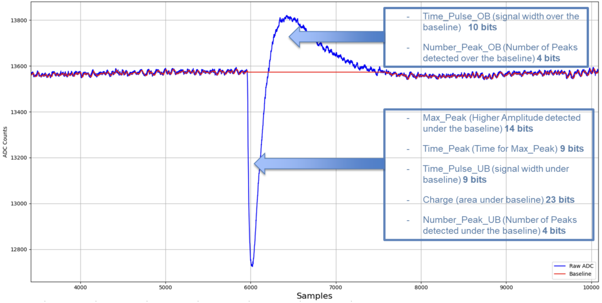
- Trigger Primitive (TP): The Simplest signal waveform representation (wire hit). These are generated using hit finding algorithms in the readout subsystem.
- Trigger Activity (TA): Cluster of hit(s) (TP(s)) that have been deemed fit to be sent up to the next level in the trigger hierarchy. Typically these will be tracks/showers or other outstanding physics activity within the box (sub-detector).
- Trigger Candidate (TC): Cluster of TAs across all sub-detectors.
- Trigger Decision (TD): A trigger request issued by Module Level Trigger (MLT) to the Data Flow Orchestrator (DFO) in order to request the raw data of the relevant detector channels over specified time windows from the readout subsystem that should be permanently stored for later analysis.
- Trigger Record (TR): An object in a stored file, containing the raw data, TPs, TAs, and TCs that have led to its construction.

---

[1] from the current trigger system *https://github.com/DUNE-DAQ/trigger*

Anatomy of the current TP from PDS

- TPs for PDS are computed in DAPHNe (Gateware)
- TAs, TCs, and TDs are computed in the DAQ servers
- Time alignment in the DAQ-Trigger system is achieved using the Time Stamp and the TP's relative delay to the frame.
- The format might fit the current format of the TPC TPs

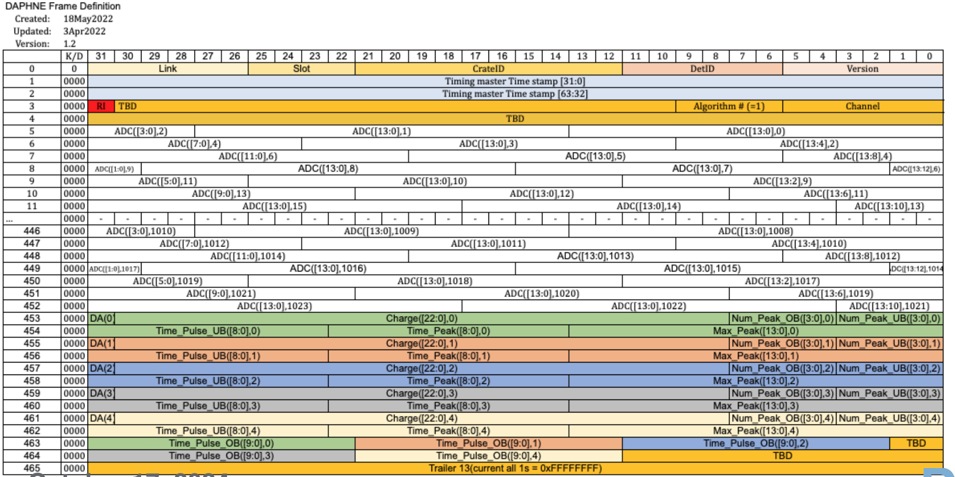


- Time_Pulse_OB (signal width over the baseline) **10 bits**
- Number_Peak_OB (Number of Peaks detected over the baseline) **4 bits**

- Max_Peak (Higher Amplitude detected under the baseline) **14 bits**
- Time_Peak (Time for Max_Peak) **9 bits**
- Time_Pulse_UB (signal width under baseline) **9 bits**
- Charge (area under baseline) **23 bits**
- Number_Peak_UB (Number of Peaks detected under the baseline) **4 bits**

From the EDH DAPHNE Format

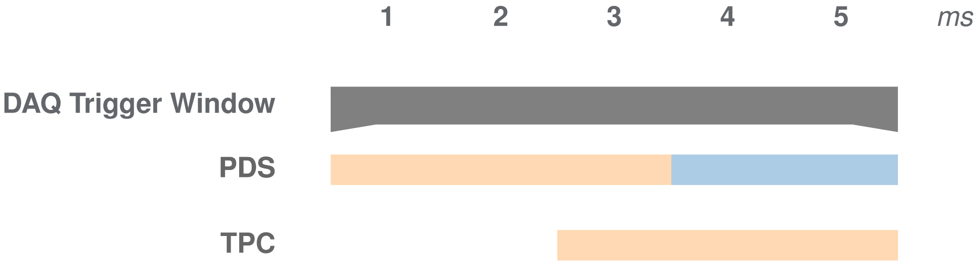
Anatomy of the current TP from PDS



DAPHNE Frame Definition
Created: 18May2022
Updated: 3Apr2022
Version: 1.2

<table>
<tr><td></td><td>K/D</td><td>31</td><td>30</td><td>29</td><td>28</td><td>27</td><td>26</td><td>25</td><td>24</td><td>23</td><td>22</td><td>21</td><td>20</td><td>19</td><td>18</td><td>17</td><td>16</td><td>15</td><td>14</td><td>13</td><td>12</td><td>11</td><td>10</td><td>9</td><td>8</td><td>7</td><td>6</td><td>5</td><td>4</td><td>3</td><td>2</td><td>1</td><td>0</td></tr>
<tr><td>0</td><td>0</td><td colspan="6">Link</td><td colspan="4">Slot</td><td colspan="10">CrateID</td><td colspan="6">DetID</td><td colspan="6">Version</td></tr>
<tr><td>1</td><td>0000</td><td colspan="32">Timing master Time stamp [31:0]</td></tr>
<tr><td>2</td><td>0000</td><td colspan="32">Timing master Time stamp [63:32]</td></tr>
<tr><td>3</td><td>0000</td><td>RI</td><td colspan="21">TBD</td><td colspan="4">Algorithm # (=1)</td><td colspan="6">Channel</td></tr>
<tr><td>4</td><td>0000</td><td colspan="32">TBD</td></tr>
<tr><td>5</td><td>0000</td><td colspan="4">ADC([3:0],2)</td><td colspan="14">ADC([13:0],1)</td><td colspan="14">ADC([13:0],0)</td></tr>
<tr><td>6</td><td>0000</td><td colspan="8">ADC([7:0],4)</td><td colspan="14">ADC([13:0],3)</td><td colspan="10">ADC([13:4],2)</td></tr>
<tr><td>7</td><td>0000</td><td colspan="12">ADC([11:0],6)</td><td colspan="14">ADC([13:0],5)</td><td colspan="6">ADC([13:8],4)</td></tr>
<tr><td>8</td><td>0000</td><td colspan="2">ADC([1:0],9)</td><td colspan="14">ADC([13:0],8)</td><td colspan="14">ADC([13:0],7)</td><td colspan="2">ADC([13:12],6)</td></tr>
<tr><td>9</td><td>0000</td><td colspan="6">ADC([5:0],11)</td><td colspan="14">ADC([13:0],10)</td><td colspan="12">ADC([13:2],9)</td></tr>
<tr><td>10</td><td>0000</td><td colspan="10">ADC([9:0],13)</td><td colspan="14">ADC([13:0],12)</td><td colspan="8">ADC([13:6],11)</td></tr>
<tr><td>11</td><td>0000</td><td colspan="14">ADC([13:0],15)</td><td colspan="14">ADC([13:0],14)</td><td colspan="4">ADC([13:10],13)</td></tr>
<tr><td>...</td><td>0000</td><td>-</td><td>-</td><td>-</td><td>-</td><td>-</td><td>-</td><td>-</td><td>-</td><td>-</td><td>-</td><td>-</td><td>-</td><td>-</td><td>-</td><td>-</td><td>-</td><td>-</td><td>-</td><td>-</td><td>-</td><td>-</td><td>-</td><td>-</td><td>-</td><td>-</td><td>-</td><td>-</td><td>-</td><td>-</td><td>-</td><td>-</td><td>-</td></tr>
<tr><td>446</td><td>0000</td><td colspan="4">ADC([3:0],1010)</td><td colspan="14">ADC([13:0],1009)</td><td colspan="14">ADC([13:0],1008)</td></tr>
<tr><td>447</td><td>0000</td><td colspan="8">ADC([7:0],1012)</td><td colspan="14">ADC([13:0],1011)</td><td colspan="10">ADC([13:4],1010)</td></tr>
<tr><td>448</td><td>0000</td><td colspan="12">ADC([11:0],1014)</td><td colspan="14">ADC([13:0],1013)</td><td colspan="6">ADC([13:8],1012)</td></tr>
<tr><td>449</td><td>0000</td><td colspan="2">ADC([1:0],1017)</td><td colspan="14">ADC([13:0],1016)</td><td colspan="14">ADC([13:0],1015)</td><td colspan="2">DC([13:12],1014</td></tr>
<tr><td>450</td><td>0000</td><td colspan="6">ADC([5:0],1019)</td><td colspan="14">ADC([13:0],1018)</td><td colspan="12">ADC([13:2],1017)</td></tr>
<tr><td>451</td><td>0000</td><td colspan="10">ADC([9:0],1021)</td><td colspan="14">ADC([13:0],1020)</td><td colspan="8">ADC([13:6],1019)</td></tr>
<tr><td>452</td><td>0000</td><td colspan="14">ADC([13:0],1023)</td><td colspan="14">ADC([13:0],1022)</td><td colspan="4">ADC([13:10],1021)</td></tr>
<tr><td>453</td><td>0000</td><td>DA(0</td><td colspan="23">Charge([22:0],0)</td><td colspan="4">Num_Peak_OB([3:0],0)</td><td colspan="4">Num_Peak_UB([3:0],0)</td></tr>
<tr><td>454</td><td>0000</td><td colspan="9">Time_Pulse_UB([8:0],0)</td><td colspan="9">Time_Peak([8:0],0)</td><td colspan="14">Max_Peak([13:0],0)</td></tr>
<tr><td>455</td><td>0000</td><td>DA(1</td><td colspan="23">Charge([22:0],1)</td><td colspan="4">Num_Peak_OB([3:0],1)</td><td colspan="4">Num_Peak_UB([3:0],1)</td></tr>
<tr><td>456</td><td>0000</td><td colspan="9">Time_Pulse_UB([8:0],1)</td><td colspan="9">Time_Peak([8:0],1)</td><td colspan="14">Max_Peak([13:0],1)</td></tr>
<tr><td>457</td><td>0000</td><td>DA(2</td><td colspan="23">Charge([22:0],2)</td><td colspan="4">Num_Peak_OB([3:0],2)</td><td colspan="4">Num_Peak_UB([3:0],2)</td></tr>
<tr><td>458</td><td>0000</td><td colspan="9">Time_Pulse_UB([8:0],2)</td><td colspan="9">Time_Peak([8:0],2)</td><td colspan="14">Max_Peak([13:0],2)</td></tr>
<tr><td>459</td><td>0000</td><td>DA(3</td><td colspan="23">Charge([22:0],3)</td><td colspan="4">Num_Peak_OB([3:0],3)</td><td colspan="4">Num_Peak_UB([3:0],3)</td></tr>
<tr><td>460</td><td>0000</td><td colspan="9">Time_Pulse_UB([8:0],3)</td><td colspan="9">Time_Peak([8:0],3)</td><td colspan="14">Max_Peak([13:0],3)</td></tr>
<tr><td>461</td><td>0000</td><td>DA(4</td><td colspan="23">Charge([22:0],4)</td><td colspan="4">Num_Peak_OB([3:0],4)</td><td colspan="4">Num_Peak_UB([3:0],4)</td></tr>
<tr><td>462</td><td>0000</td><td colspan="9">Time_Pulse_UB([8:0],4)</td><td colspan="9">Time_Peak([8:0],4)</td><td colspan="14">Max_Peak([13:0],4)</td></tr>
<tr><td>463</td><td>0000</td><td colspan="10">Time_Pulse_OB([9:0],0)</td><td colspan="10">Time_Pulse_OB([9:0],1)</td><td colspan="10">Time_Pulse_OB([9:0],2)</td><td colspan="2">TBD</td></tr>
<tr><td>464</td><td>0000</td><td colspan="10">Time_Pulse_OB([9:0],3)</td><td colspan="10">Time_Pulse_OB([9:0],4)</td><td colspan="12">TBD</td></tr>
<tr><td>465</td><td>0000</td><td colspan="32">Trailer 13(current all 1s = 0xFFFFFFFF)</td></tr>
</table>

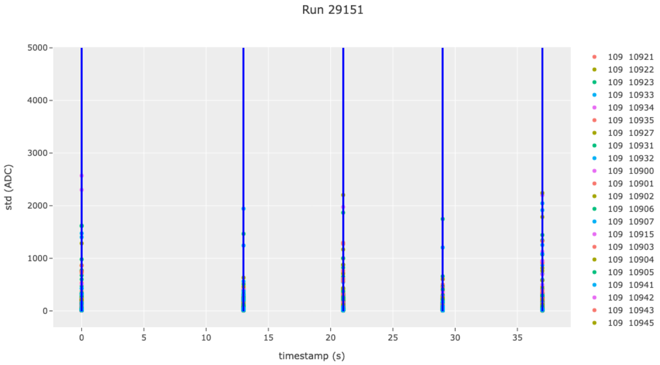
Delay between the timestamps from PDS vs TPC



The delay between a TP from PDS and a TP from TPC from the same event is in the range, $R = [0ms, 3ms]$

We can even have several TPs from one system while only one/none on the other

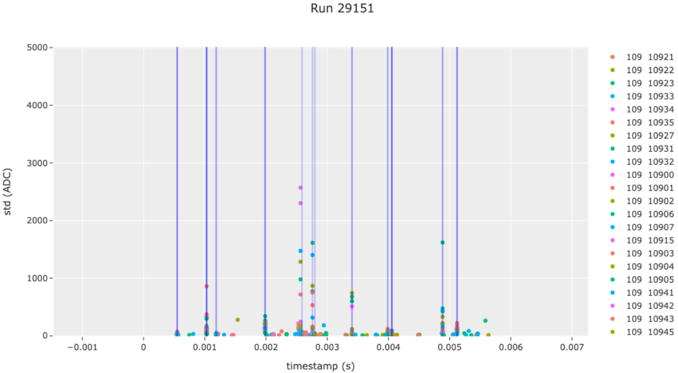
How does data look from the point of view of the DAQ



One HDF5 File - One APA

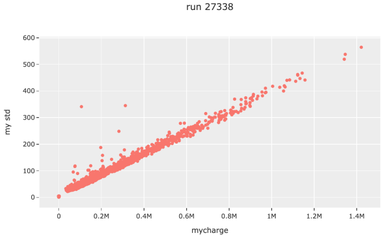
- Each point represents one waveform.
- Each color represents one channel.
- I'm assuming RMS is proportional to the charge in a collection of TP's in a Wf (see next slide)

How does one field of a TP look from the point of view of the DAQ



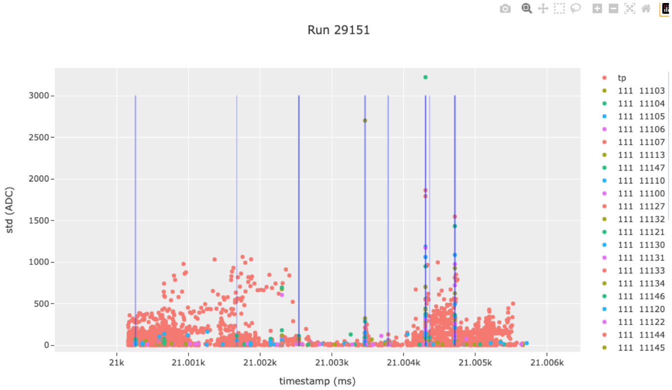
One Second of data taking - One APA

- I use the RMS since it's proportional to the charge in a collection of TPs, and those are not currently available for analysis



How does data look from the point of view of the DAQ

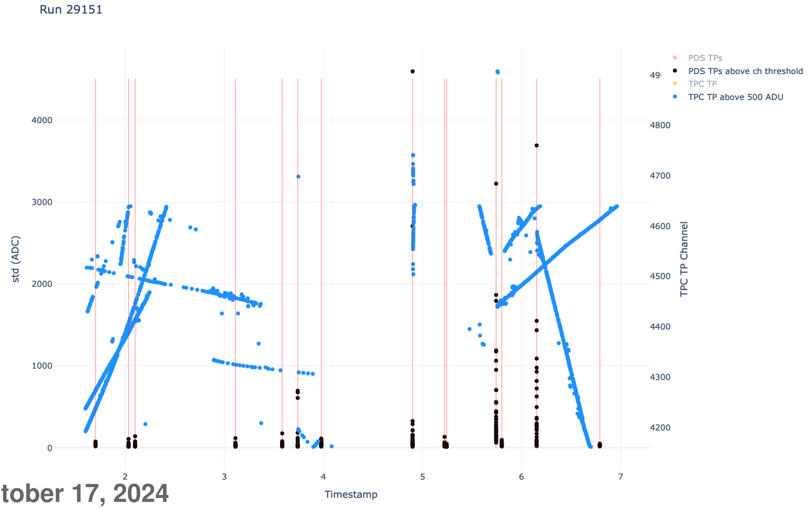
- These are all the waveforms from self-trigger above 5Pe
- We can set a threshold for a number of channels e.g. 5 with signals above a certain value, e.g.. $RMS \geq 100 ADC$
- In this case, we would have six trigger activities/clusters in one record.



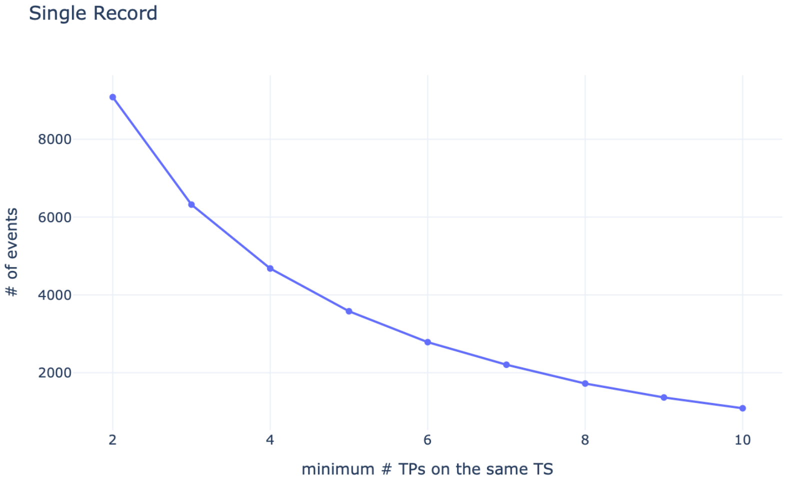
One fragment - One APA

TPC vs PDS data filtered

TPC vs PDS data filtered

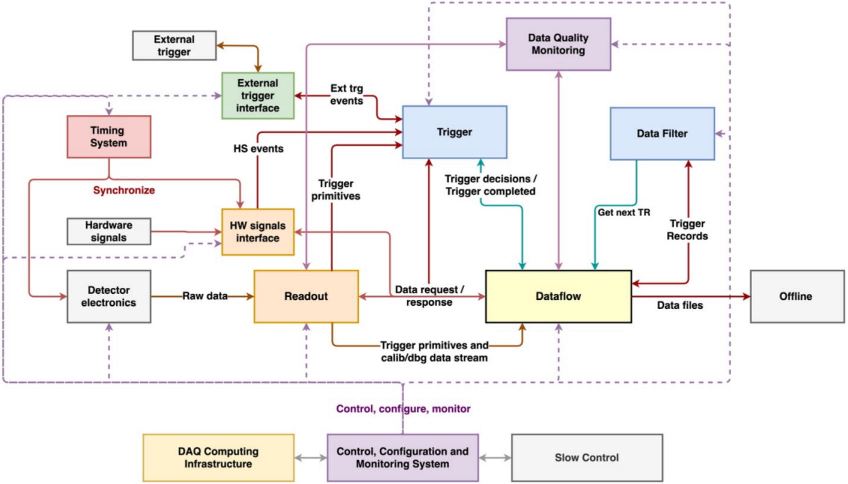


Rate of TP groups after time coincidence filter



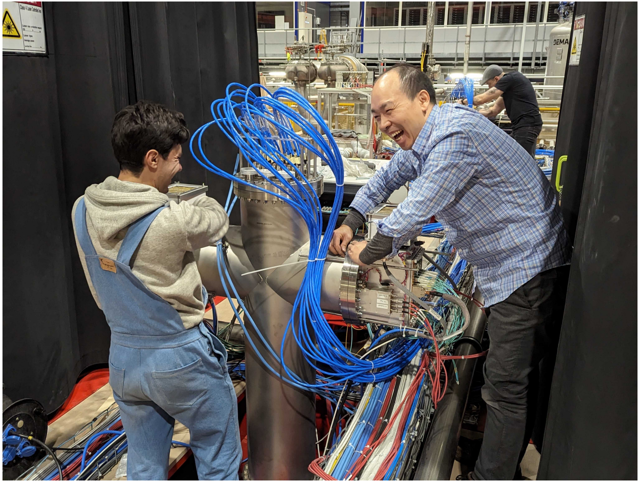
DEEP UNDERGROUND
NEUTRINO EXPERIMENT

Overview



Trigger's position relative to DAQ

DAPHNE



DAPHNE

DAPHNE is the FEB (Front End Board) of the PDS

It collects analog signals from the Photon Detectors PDs, and digitizes them, creating packets with metadata, raw waveforms, and trigger primitives -TPs.

Interfaces

- Analog Interface: DAPHNE Reads the analog signals from PD Cold Electronics
- Timing Interface: Takes the clk for the FPGA
- DAQ: DAPHNE sends data to the DAQ: readout/monitoring
- Slow Control: *not implemented*

DAPHNE

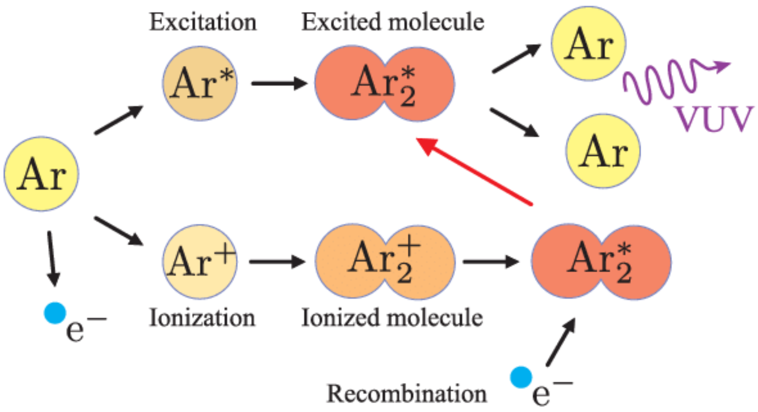
DAPHNE is the FEB (Front End Board) of the PDS

It collects analog signals from the Photon Detectors PDs, and digitizes them, creating packets with metadata, raw waveforms, and trigger primitives -TPs.

Specs

- 62.5 MSps, 40 Channels, 1 DAPHNE/APA
- Self-trigger and full streaming capability
- 4 links, 4.8 Gbps available to send data, either self-trigger or full streaming

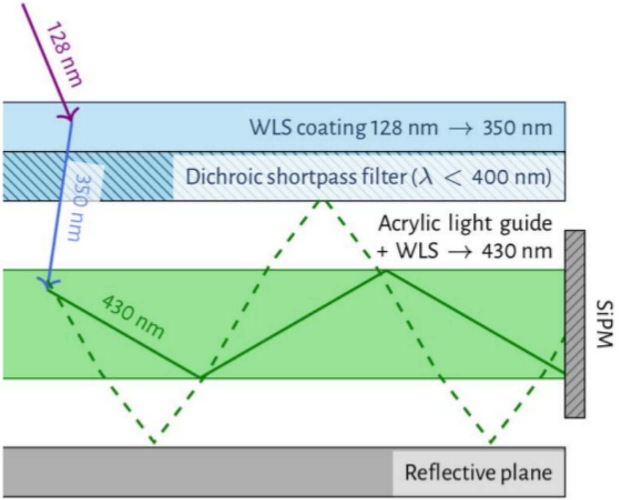
The LAr scintillation



Boccone Vittorio, 2007

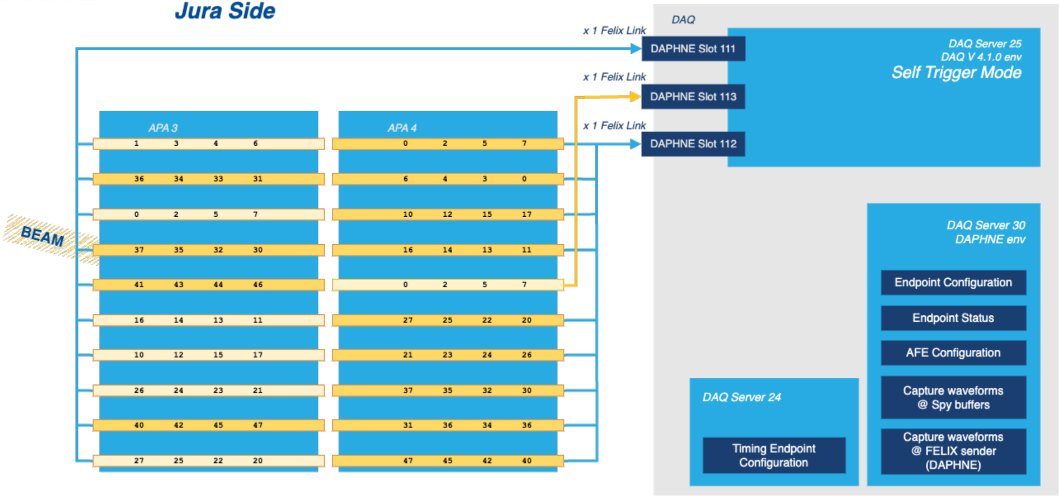
Trapping photons...

And how do we collect the photons?



Brizzolari, 2024

DAPHNE



Jura Side

DAQ

x 1 Felix Link

DAPHNE Slot 111

x 1 Felix Link

DAPHNE Slot 113

x 1 Felix Link

DAPHNE Slot 112

DAQ Server 25
DAQ V 4.1.0 env
Self Trigger Mode

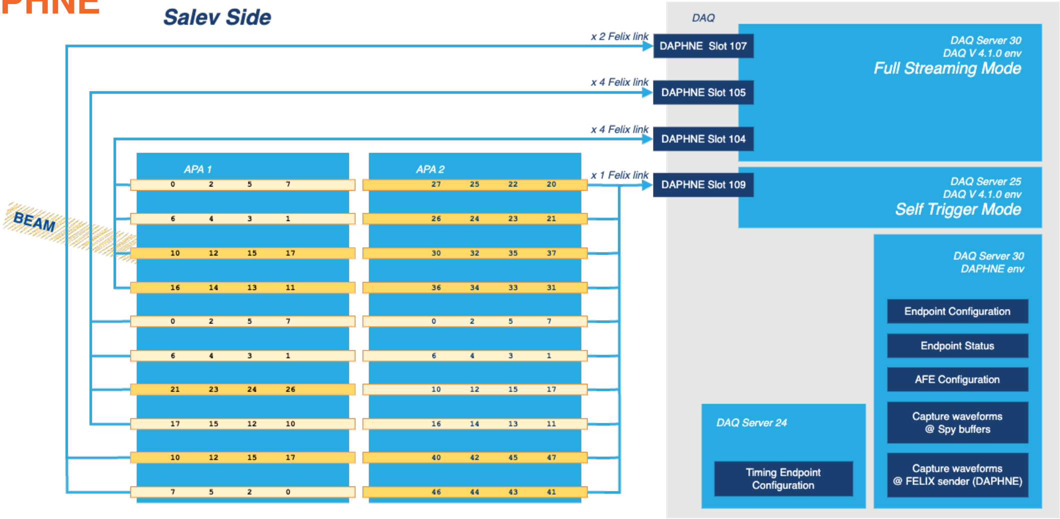
DAQ Server 30
DAPHNE env

- Endpoint Configuration
- Endpoint Status
- AFE Configuration
- Capture waveforms @ Spy buffers
- Capture waveforms @ FELIX sender (DAPHNE)

DAQ Server 24

- Timing Endpoint Configuration

DAPHNE



Salev Side

DAQ

x 2 Felix link DAPHNE Slot 107

x 4 Felix link DAPHNE Slot 105

x 4 Felix link DAPHNE Slot 104

x 1 Felix link DAPHNE Slot 109

DAQ Server 30
DAQ V 4.1.0 env
Full Streaming Mode

DAQ Server 25
DAQ V 4.1.0 env
Self Trigger Mode

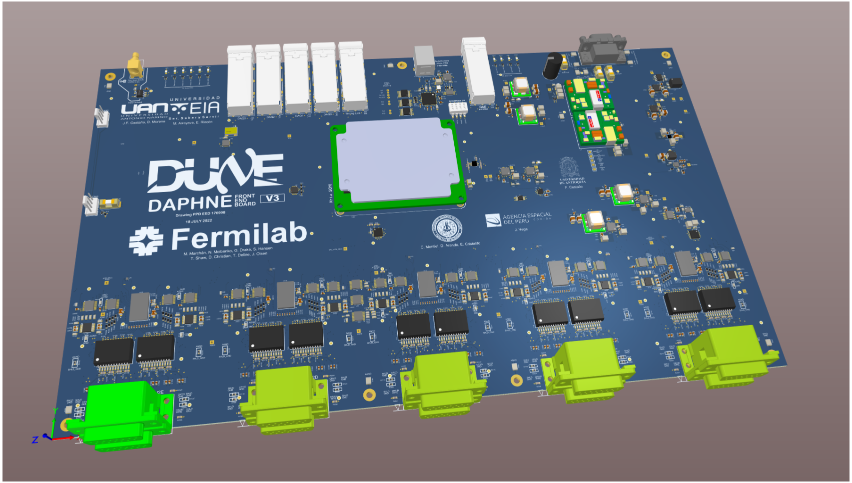
DAQ Server 30
DAPHNE env

- Endpoint Configuration
- Endpoint Status
- AFE Configuration
- Capture waveforms @ Spy buffers
- Capture waveforms @ FELIX sender (DAPHNE)

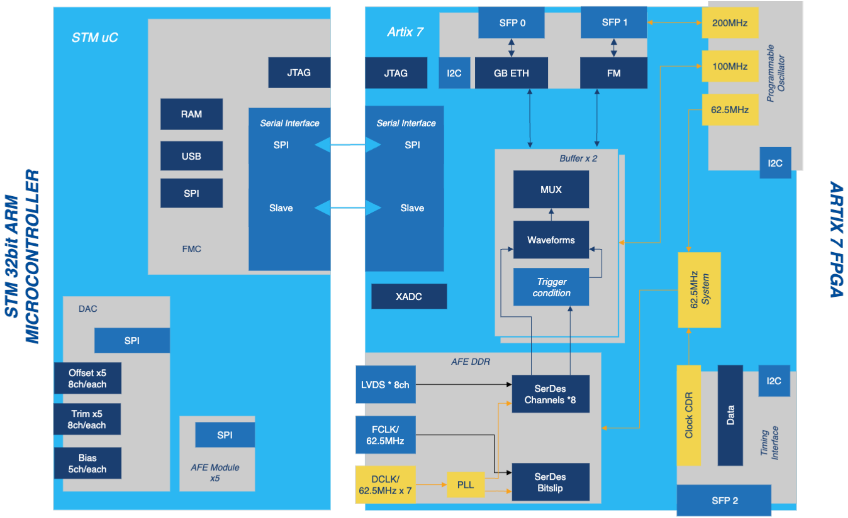
DAQ Server 24

- Timing Endpoint Configuration

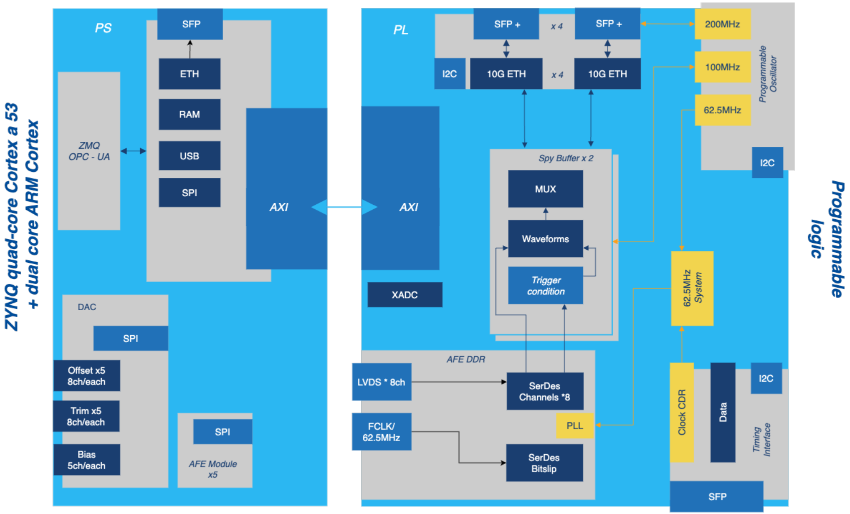
DAPHNE


UNIVERSIDAD
UAN
UNIVERSIDAD ANTONIO NARIÑO
EIA
Ser, Saber y Servir
J.F. Castaño, D. Moreno
M. Arroyave, E. Rincón
DAQ3
DAQ2
DAQ1
DAQ0
DUNE
DAPHNE FRONT END BOARD V3
Drawing PPD EED 176998
18 JULY 2022
Fermilab
M. Marchán, N. Moibenko, G. Drake, S. Hansen
T. Shaw, D. Christian, T. Deline, J. Olsen
Kria SOM
C. Montiel, D. Aranda, E. Cristaldo
AGENCIA ESPACIAL DEL PERU CONIDA
J. Vega
UNIVERSIDAD DE ANTIOQUIA
F. Castaño
Y
Z

DAPHNE


STM 32bit ARM MICROCONTROLLER
STM uC
JTAG
RAM
USB
SPI
Serial Interface
SPI
Slave
FMC
DAC
SPI
Offset x5 8ch/each
Trim x5 8ch/each
Bias 5ch/each
SPI
AFE Module x5
Artix 7
SFP 0
SFP 1
JTAG
I2C
GB ETH
FM
Serial Interface
SPI
Slave
XADC
Buffer x 2
MUX
Waveforms
Trigger condition
AFE DDR
LVDS * 8ch
SerDes Channels *8
FCLK/ 62.5MHz
DCLK/ 62.5MHz x 7
PLL
SerDes Bitslip
200MHz
100MHz
62.5MHz
Programmable Oscillator
I2C
62.5MHz System
Clock CDR
Data
I2C
Timing Interface
SFP 2
ARTIX 7 FPGA

DAPHNE


ZYNQ quad-core Cortex a 53
+ dual core ARM Cortex
PS
SFP
ETH
RAM
USB
SPI
ZMQ
OPC - UA
AXI
DAC
SPI
Offset x5
8ch/each
Trim x5
8ch/each
Bias
5ch/each
SPI
AFE Module
x5
PL
SFP +
x 4
SFP +
I2C
10G ETH
x 4
10G ETH
200MHz
100MHz
62.5MHz
Programmable
Oscillator
I2C
AXI
Spy Buffer x 2
MUX
Waveforms
Trigger
condition
XADC
62.5MHz
System
AFE DDR
LVDS * 8ch
SerDes
Channels *8
PLL
FCLK/
62.5MHz
SerDes
Bitslip
Clock CDR
Data
I2C
Timing
Interface
SFP
Programmable
logic