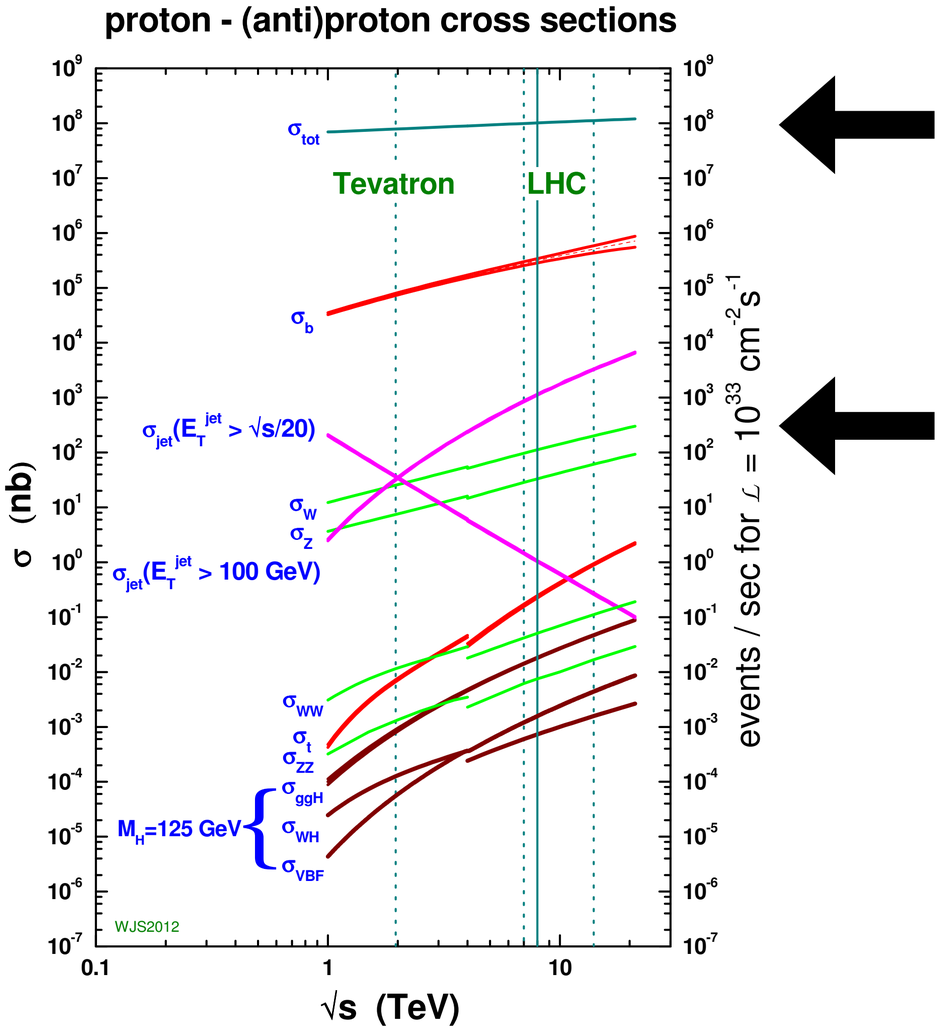
Triggering at the Large Hadron Collider

Irina Ene

PH290E, 21 April, 2021

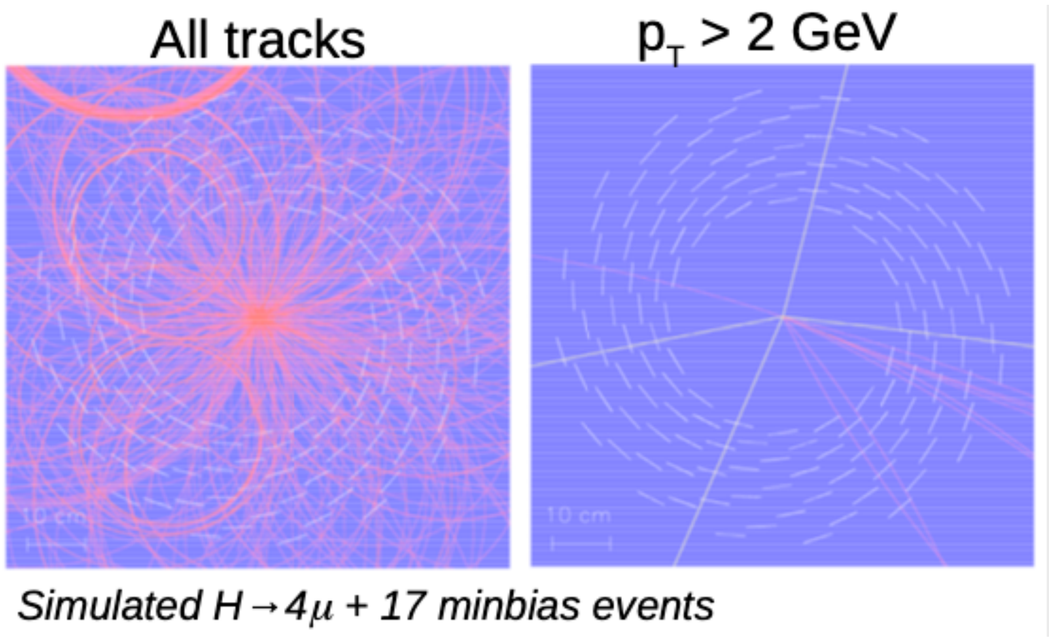
Event rates at LHC

- Typical collision “mundane”
  - ~ 1 GHz at $\mathscr{L} = 10^{34}\ \mathrm{cm}^{-2}\mathrm{s}^{-1}$
- “Interesting” physics happens at much lower rates
  - 6-8 orders of magnitude
- Higgs produced in 1 out of 10^9 collisions



How to identify interesting events?

- Mostly hadrons with low p_{T}
- **Interesting events usually have high p_{T} objects**
  - $H \to \gamma\gamma$ with $p_{\mathrm{T}}(\gamma) \sim 50 - 60$ GeV
  - $W \to e\bar{\nu}_e$ with $p_{\mathrm{T}}(e) \sim 30 - 40$ GeV
  - **Use as signatures**
    - electrons, photons, muons, jets, missing transverse energy

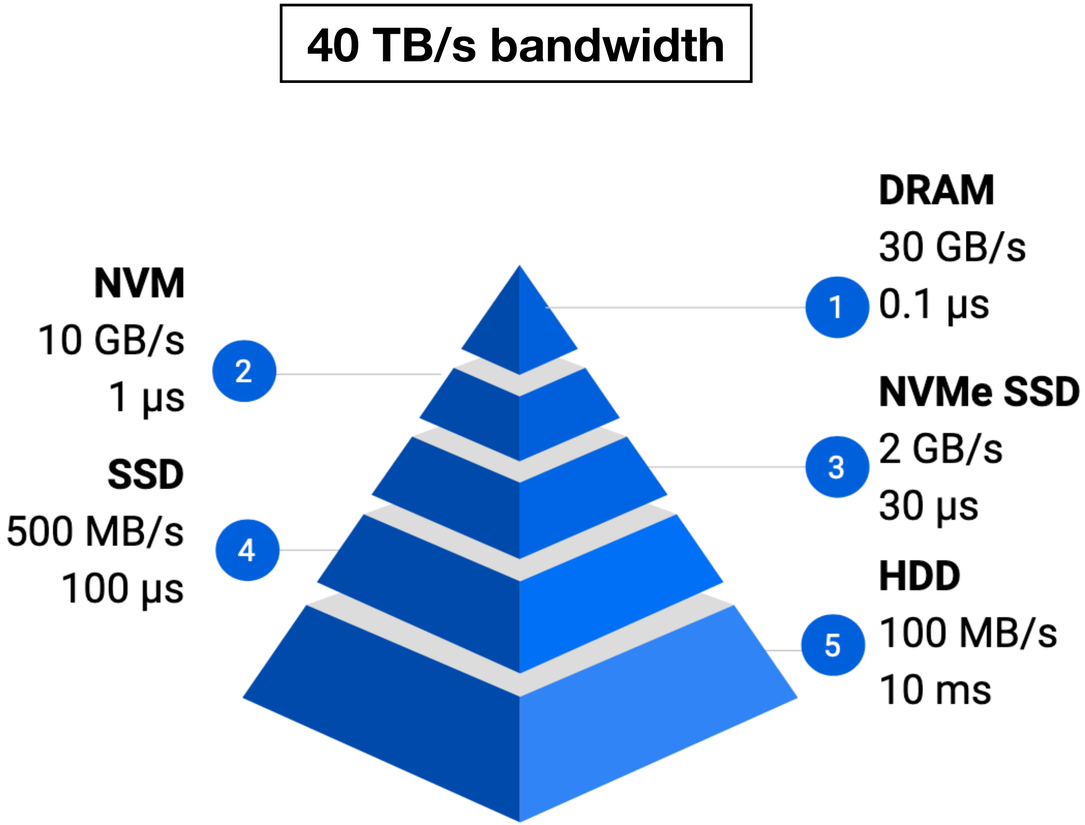


Simulated $H \to 4\mu$ + 17 minbias events

Frank Winklmeier slides

Data rates at LHC

- 25 ns bunch separation → **40 MHz** bunch crossing rate
  - Rate at which receiving new data
- ~ **1 MB** detector data per event
- Assume 100 efficient days (10^7 seconds) per year @ 40 TB/s
- $\sim 10^8$ TB per year to store (and analyze)
- **Bandwidth/storage would not be a problem for 1 kHz rate**

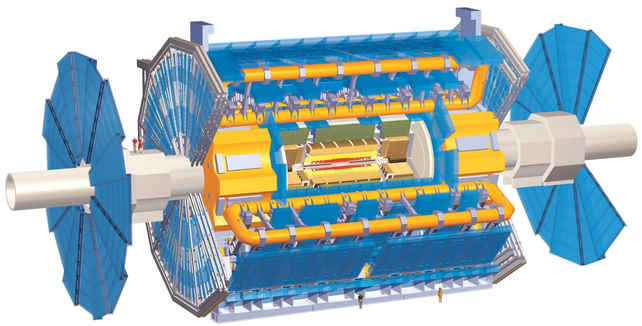


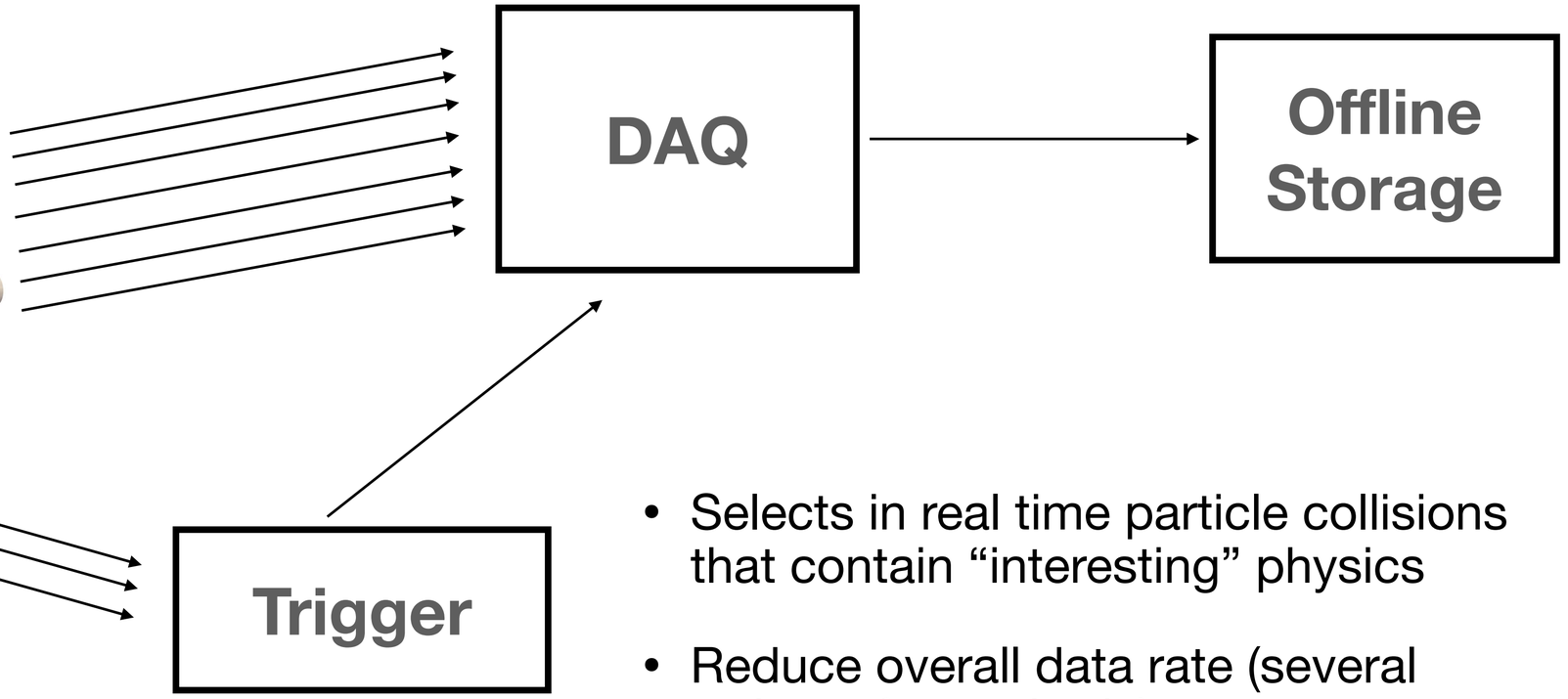
Adam Abed Abud slides

Trigger role

- Collects data from the detector
- Transfer data to storage for offline analysis

CGI ATLAS detector





- Selects in real time particle collisions that contain “interesting” physics
- Reduce overall data rate (several orders of magnitude)

Requirements for trigger system

- **High efficiency** → don’t want to lose processes of interest
- **Robust** → selection not biased in a way that affects the physics outcome
- **Large rate reduction** → limitations of DAQ and offline computers
- **Flexible** → to adapt to changing physics goals
- **Affordable** → resources are finite

Requirements for trigger system

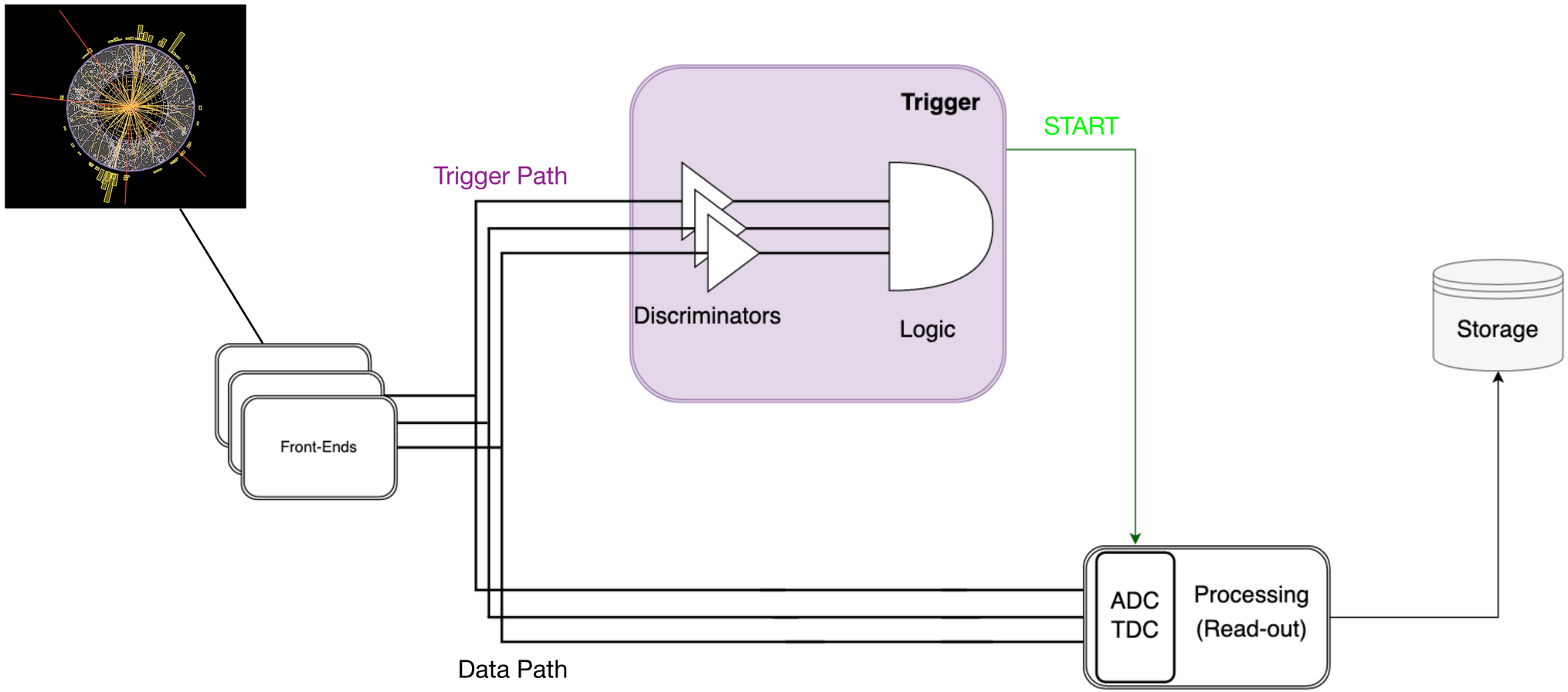
The earliest trigger… The graduate student

- High efficiency?
  Reflexes too slow ❌
- Large reduction of rate?
  Depends on your student… 🤞
- Robustness?
  Gets distracted, tired, hungry, … ❌
- Highly flexible?
  Depends on your student… 🤞
- Affordable?
  ✔️

7

Dinyar Rabady slides

A (very simple) trigger example



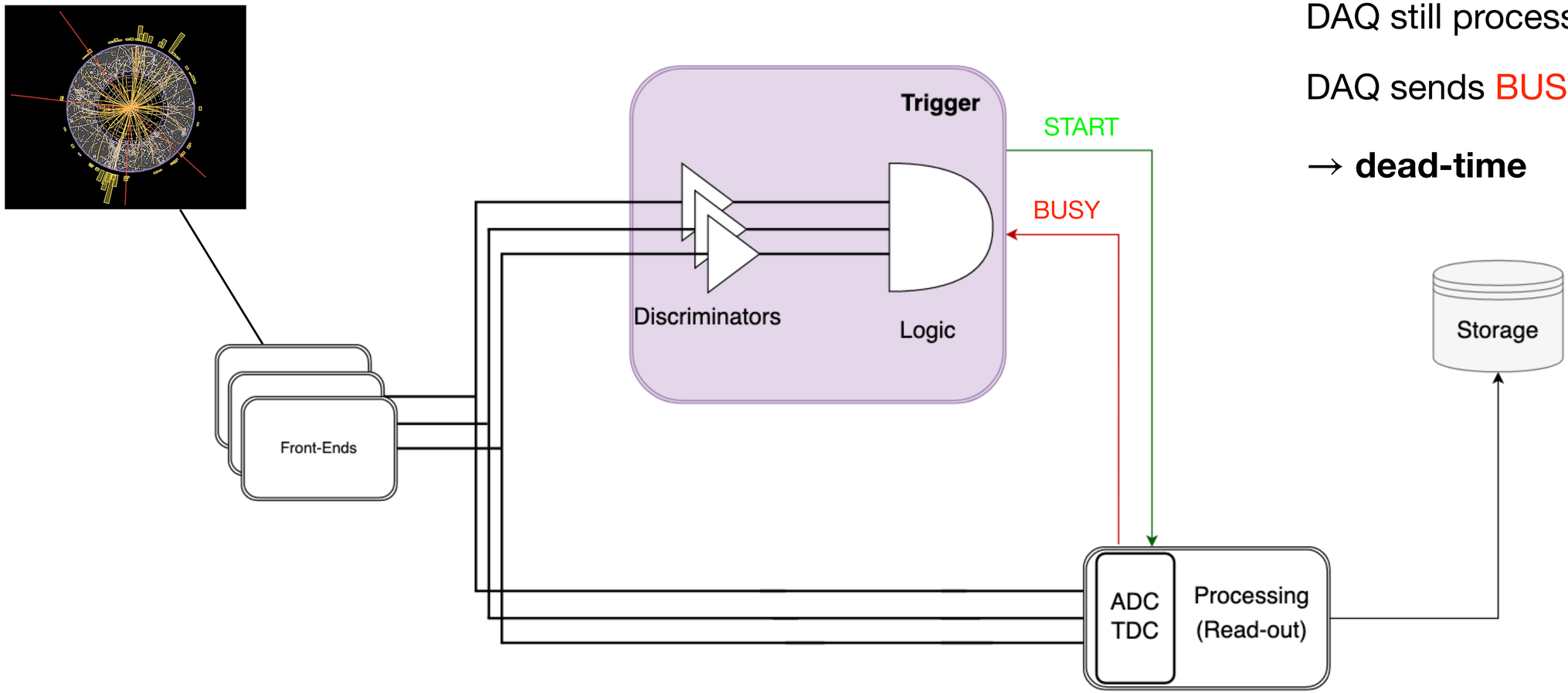
Inspired by Dinyar Rabady slides

A (very simple) trigger example

Trigger might send signal while DAQ still processing previous event

DAQ sends BUSY signal

→ **dead-time**



Inspired by Dinyar Rabady slides

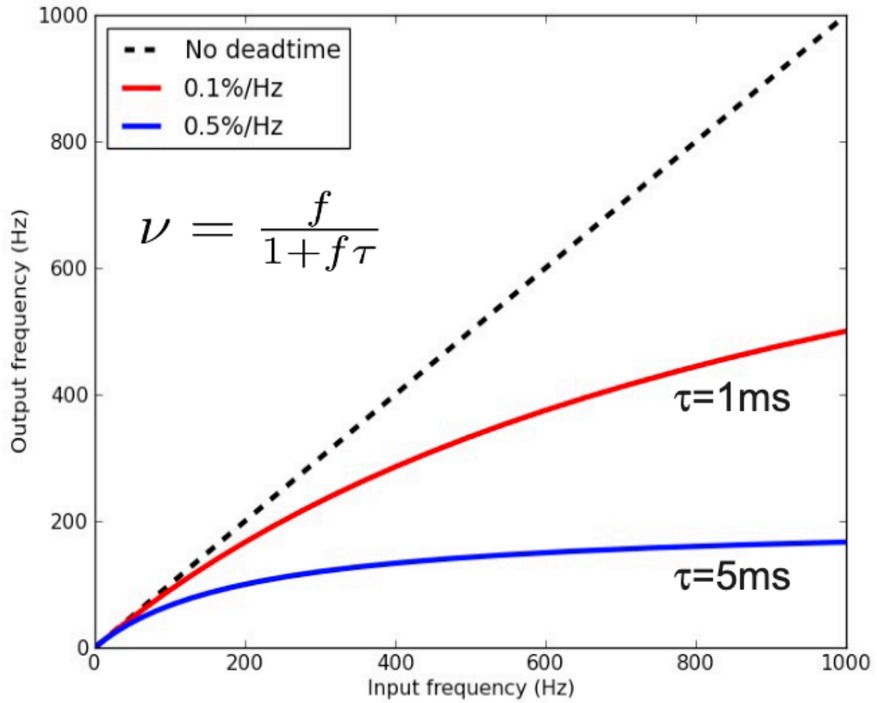
Dead-time

- **Dead-time** = fraction of acquisition time when no (new) events can be processed (even if they are interesting)
  - Happens when processing step takes finite time to complete
- We want high efficiency $\rightarrow f\tau \ll 1$

- f = average input rate
- ν = readout rate
- τ = time to process one event

$\nu\tau$ = fraction of events lost

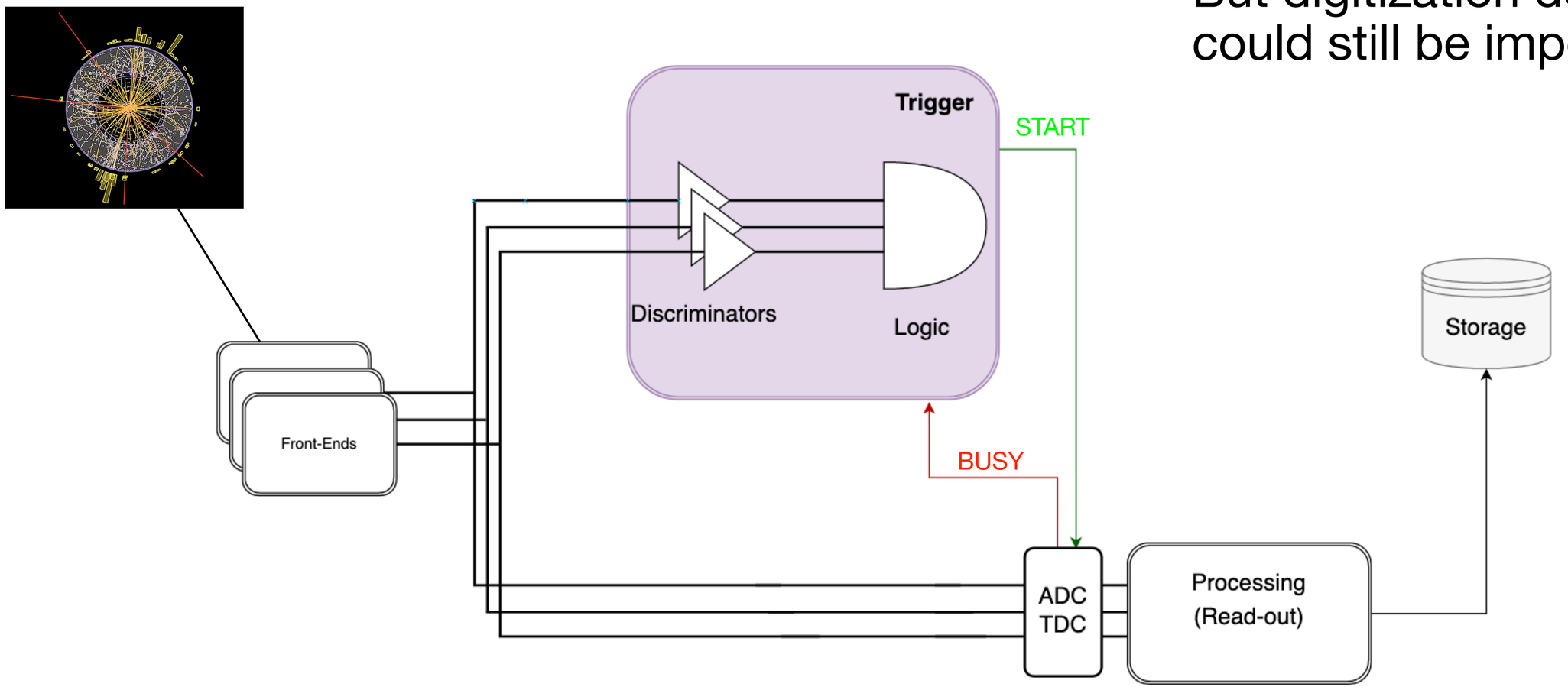
- $f - \nu = f\nu\tau \rightarrow \nu = \dfrac{f}{1 + f\tau} < f$



Brian Petersen slides

A (very simple) trigger example

But digitization dead-time could still be important



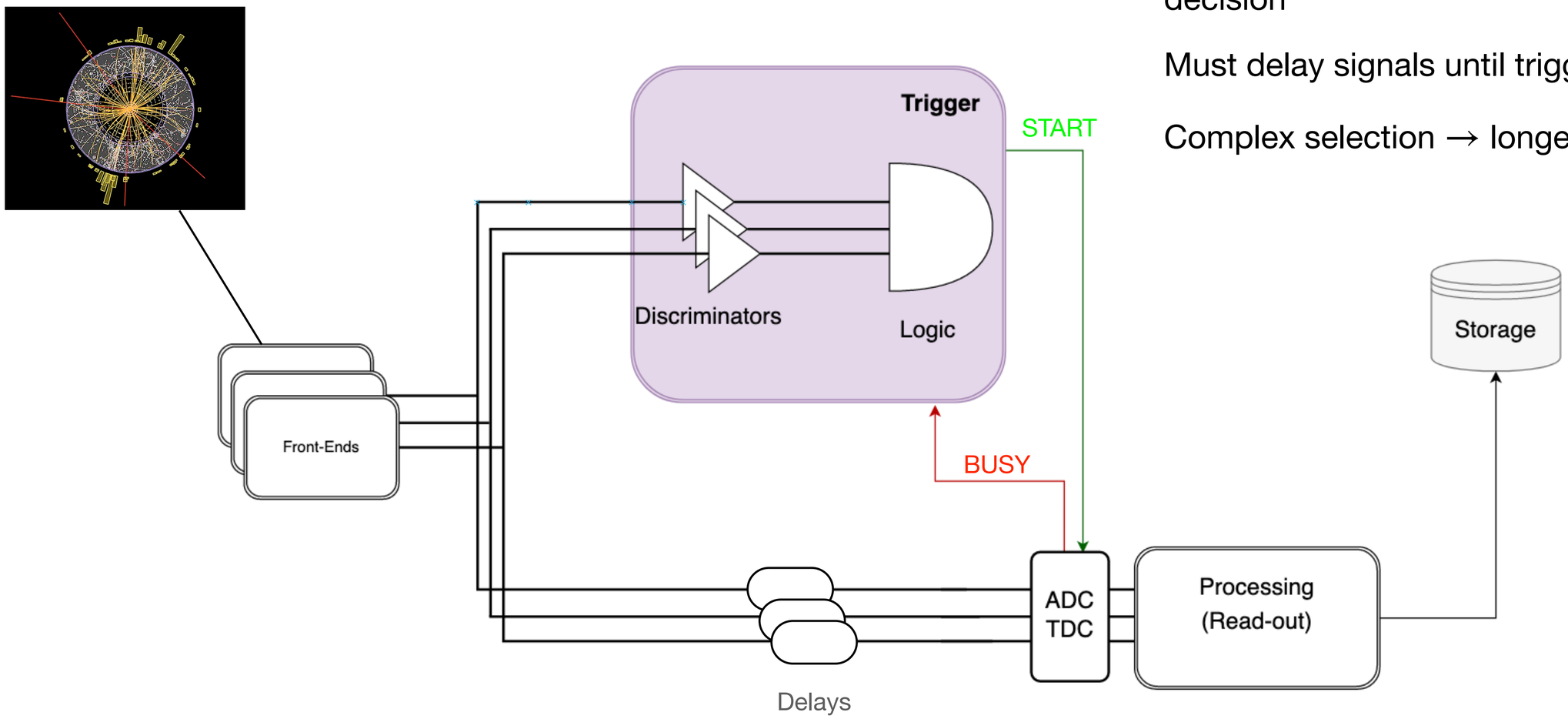
Inspired by Dinyar Rabady slides

A (very simple) trigger example

Latency = time to form and distribute trigger decision

Must delay signals until trigger decision is ready

Complex selection → longer latency



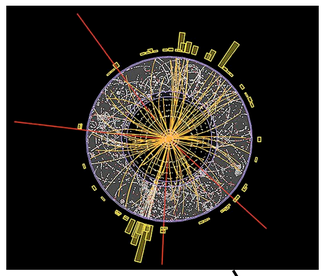
Analogue cables could probably work in a simple case

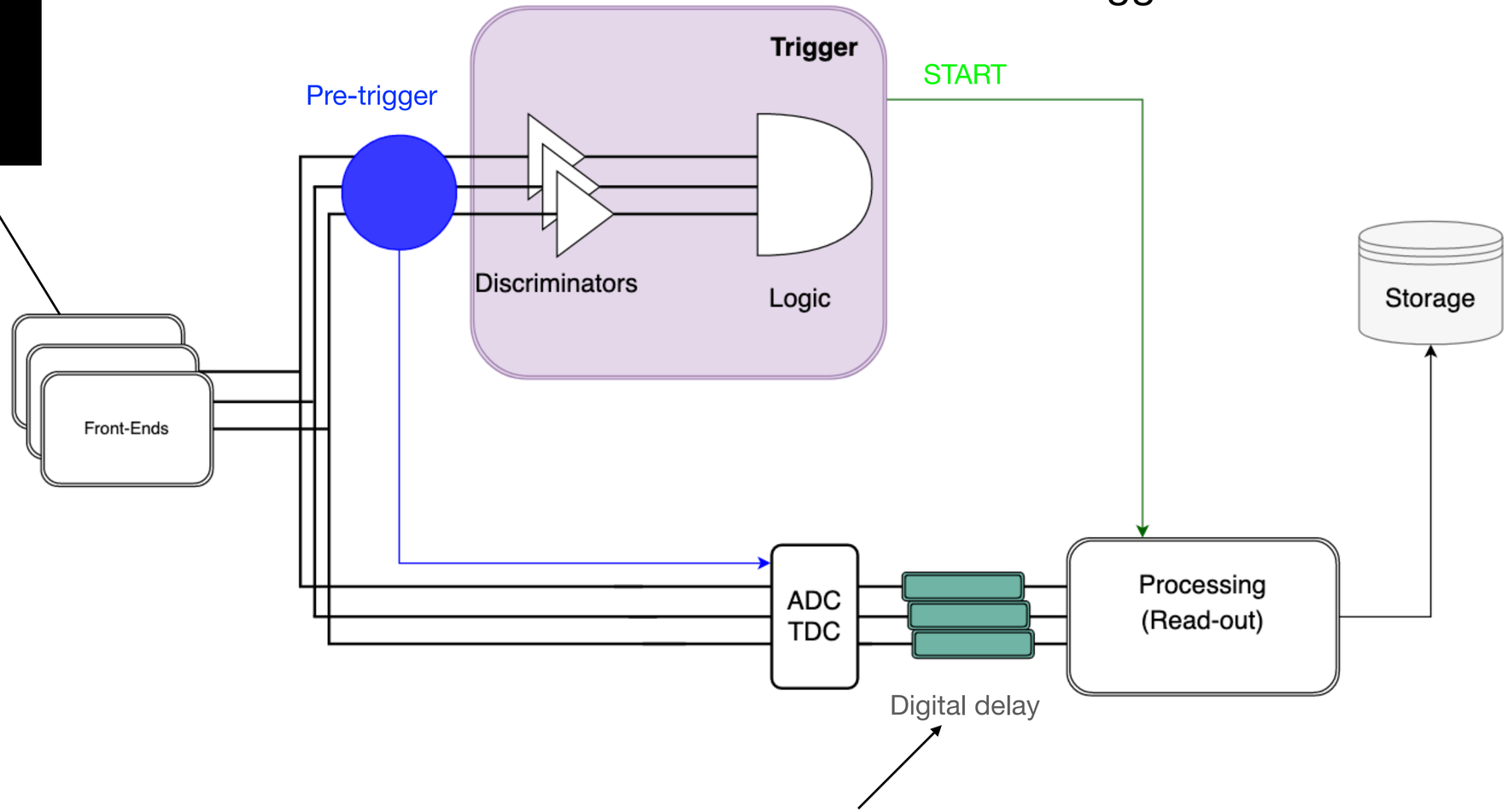
Inspired by Dinyar Rabady slides

A (very simple) trigger example

Pre-trigger: start digitizers without delay

Trigger decision can arrive later





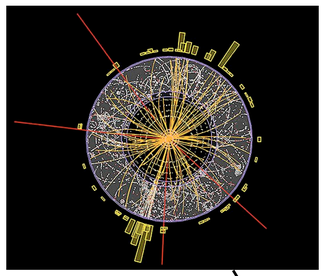
Inspired by Dinyar Rabady slides

A (very simple) trigger example

LHC clock = pre-trigger

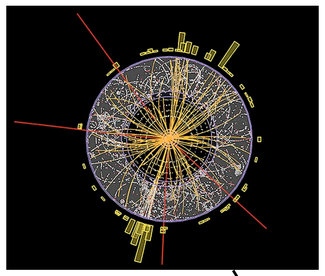
Pipeline to delay and temporarily store data

Trigger decision and data arrive at read-out at the same time



Inspired by Dinyar Rabady slides

A (very simple) digital trigger example



Digital Trigger System

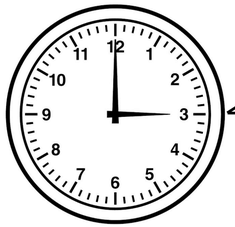
START

Storage

Front-Ends

ADC

LHC clock



Digital delay
PIPELINE

Processing
(Read-out)

Inspired by Dinyar Rabady slides

Implementation overview

Multi-level trigger

- Want to reduce rate from 40 MHz down to ~1 kHz
- **Multiple levels of triggers**, where at each level decrease output rate and increase processing times
  - Earlier levels: high rates, short latency
  - Higher levels: lower rates, longer latency
- High efficiency at all levels — rejected events are lost forever

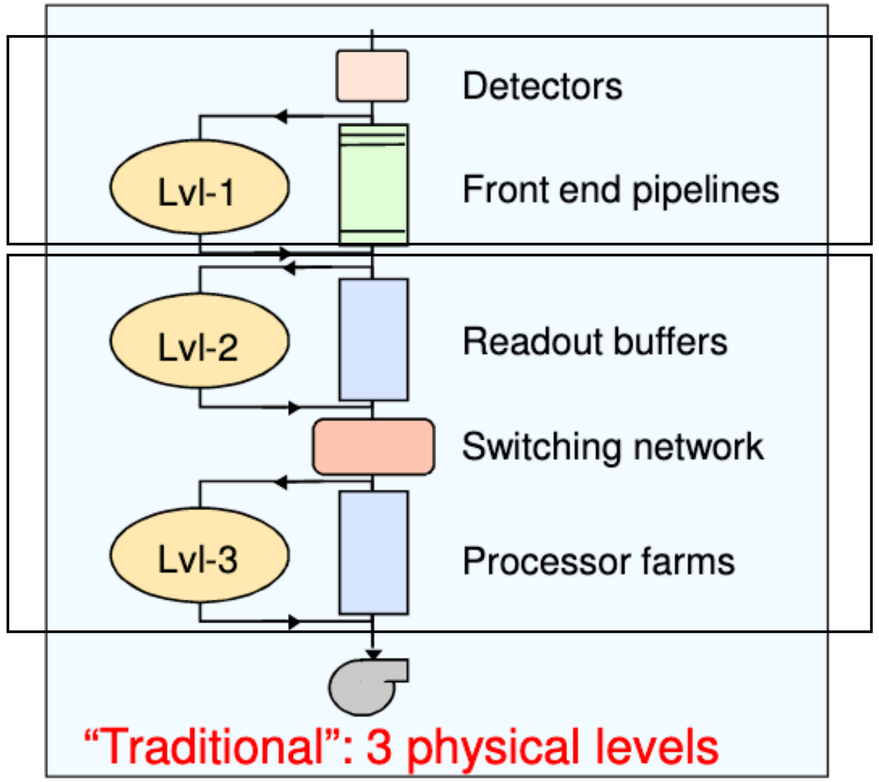
LHC Run-1

Experiment	# Trigger Levels
ATLAS	3
CMS	2
LHCb	3
ALICE	4

Implementation overview

Multi-level trigger

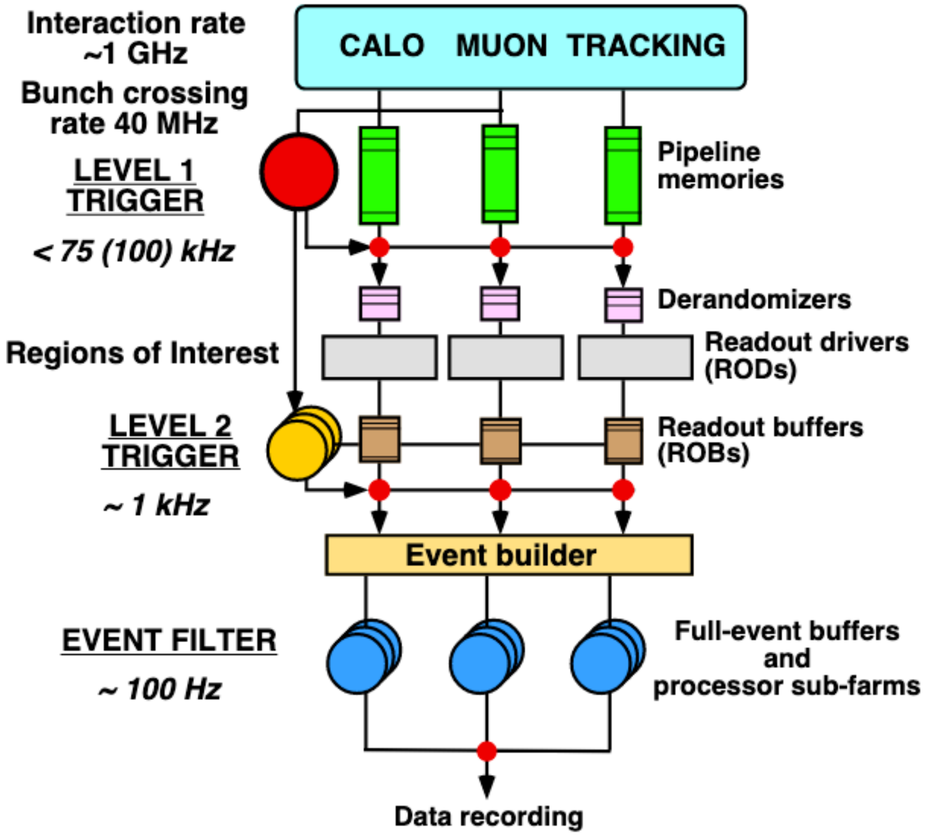
- **First-level trigger**
  - High rate, low latency
  - Custom electronics
  - Small subsample of detector data
  - Relatively simple algorithms
- **High-level trigger**
  - Lower rate, relaxed latency requirements
  - Commercial computing clusters
  - Flexible, more complex algorithms



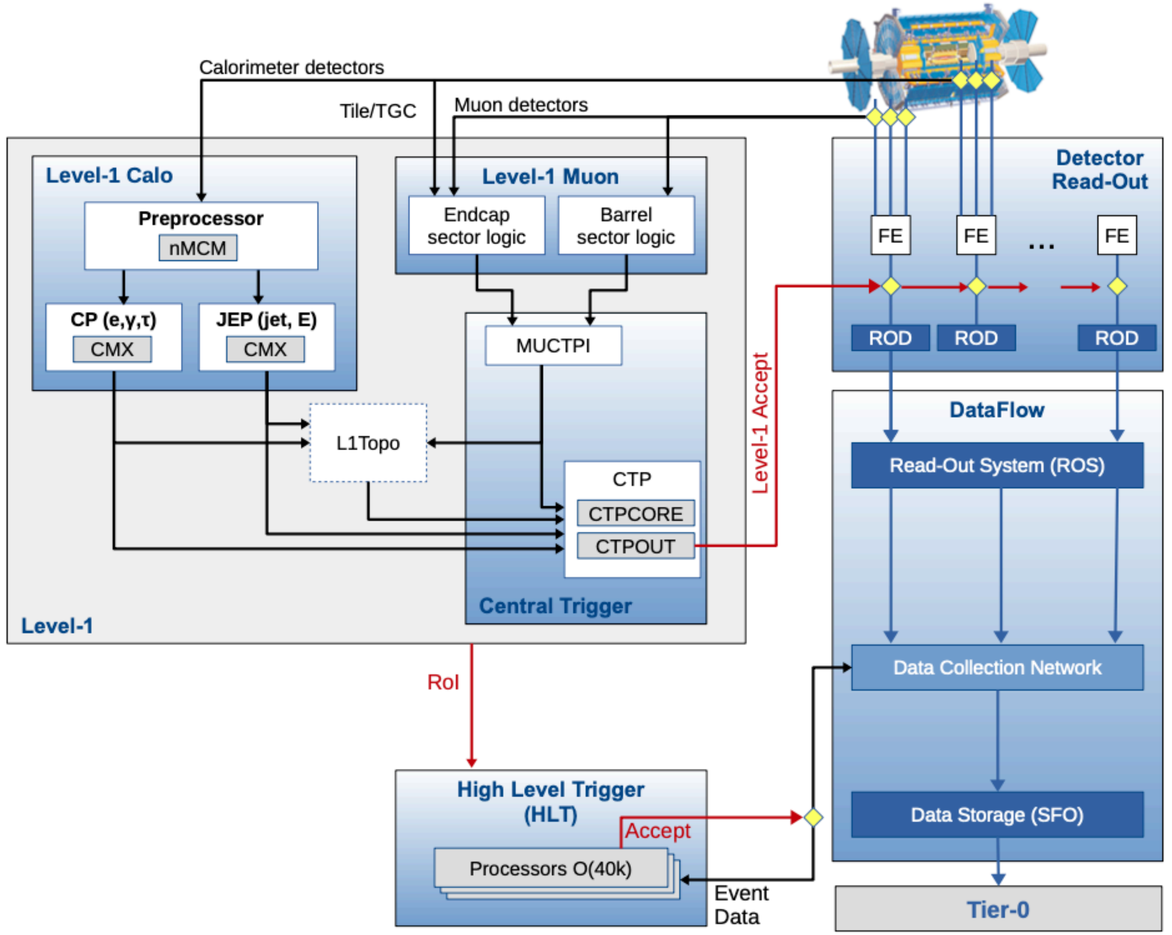
Leonard Apanasevich slides

Implementation overview

Example: ATLAS



ATLAS TDR

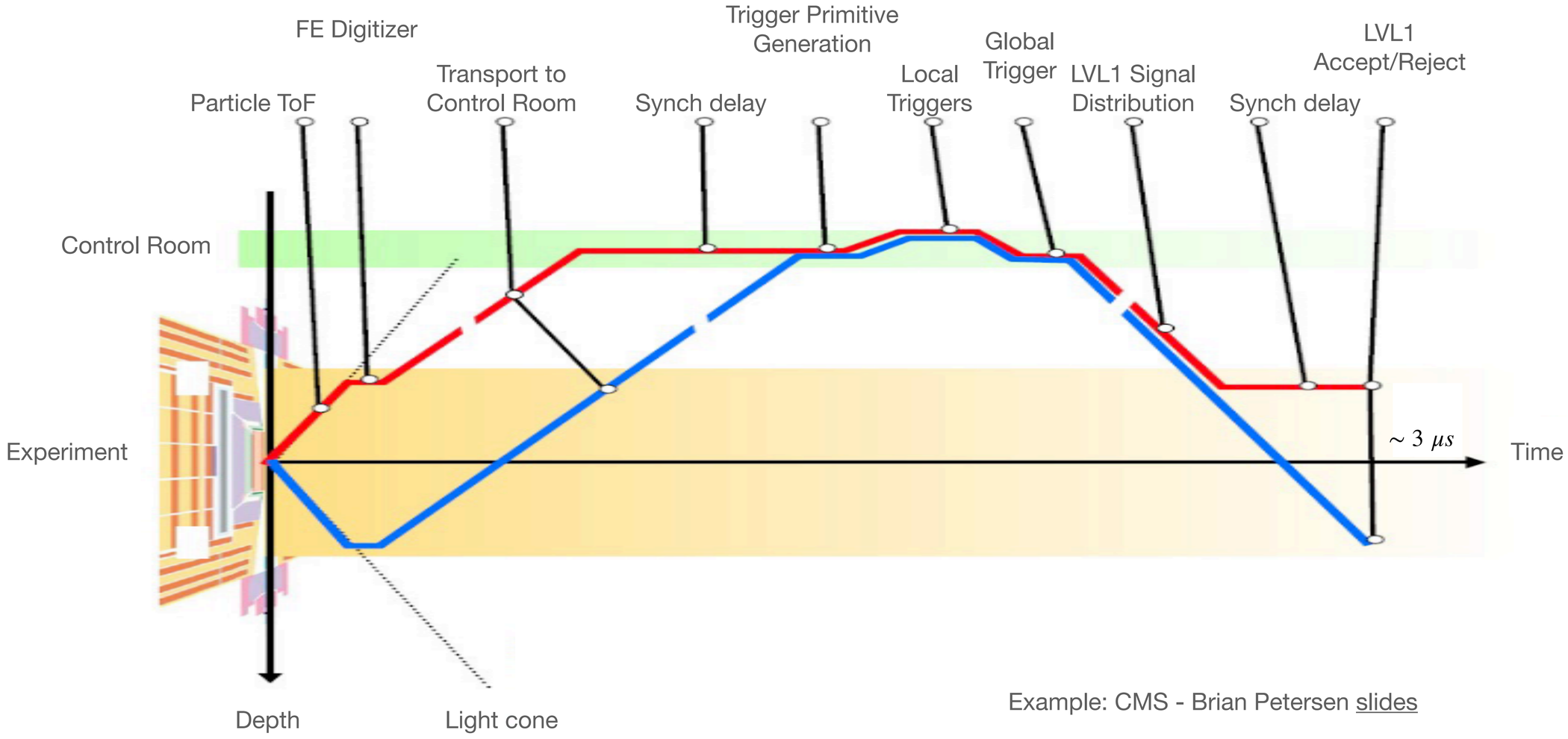


Frank Winklmeier slides

Level-1 Trigger

Latency

- Latency = time from bunch crossing until trigger decision reaches detector
- Lot of time spent on data transmission

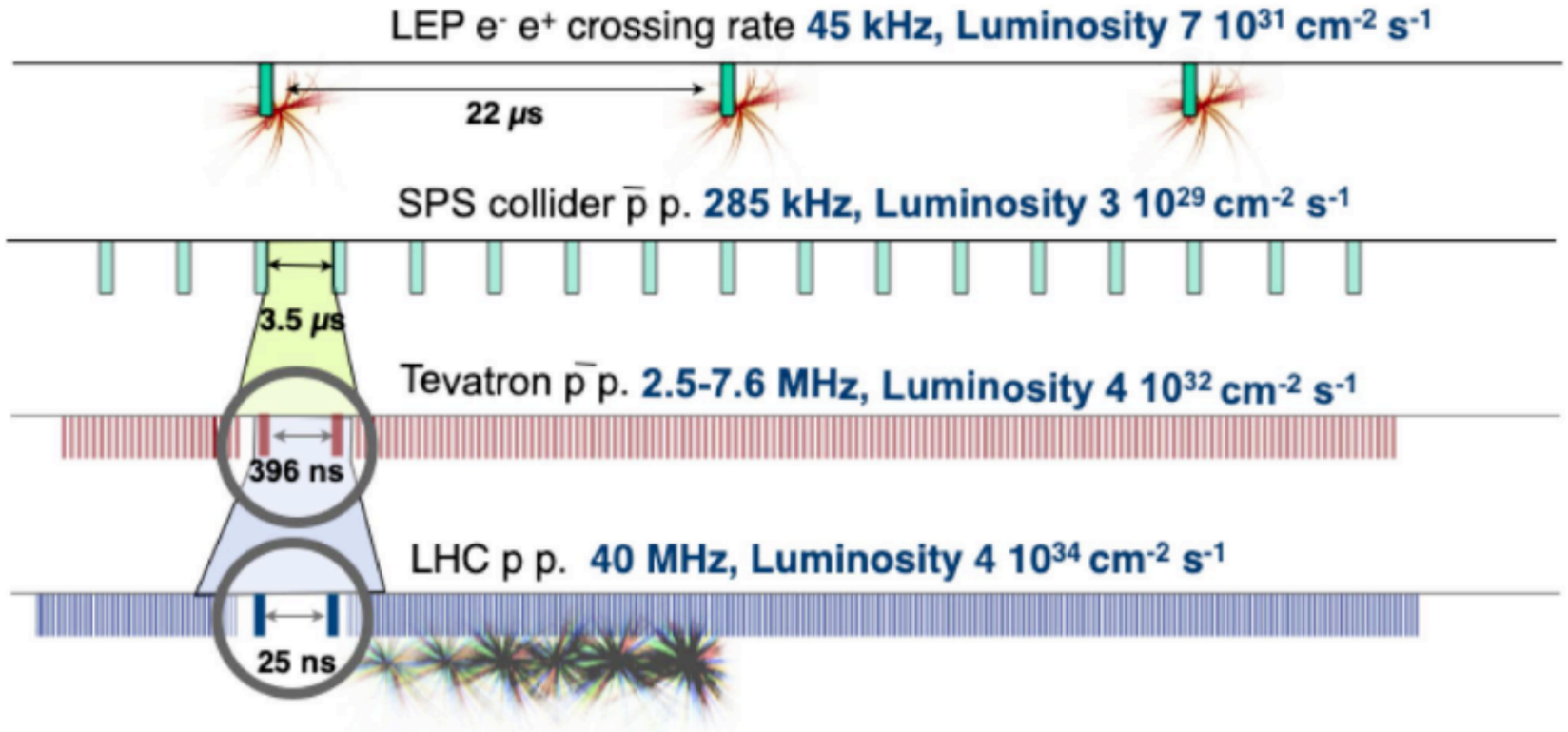


Example: CMS - Brian Petersen slides

Level-1 Trigger

Pipelines

- Can’t deliver a complex trigger decision in 25 ns and can’t process only one event at a time

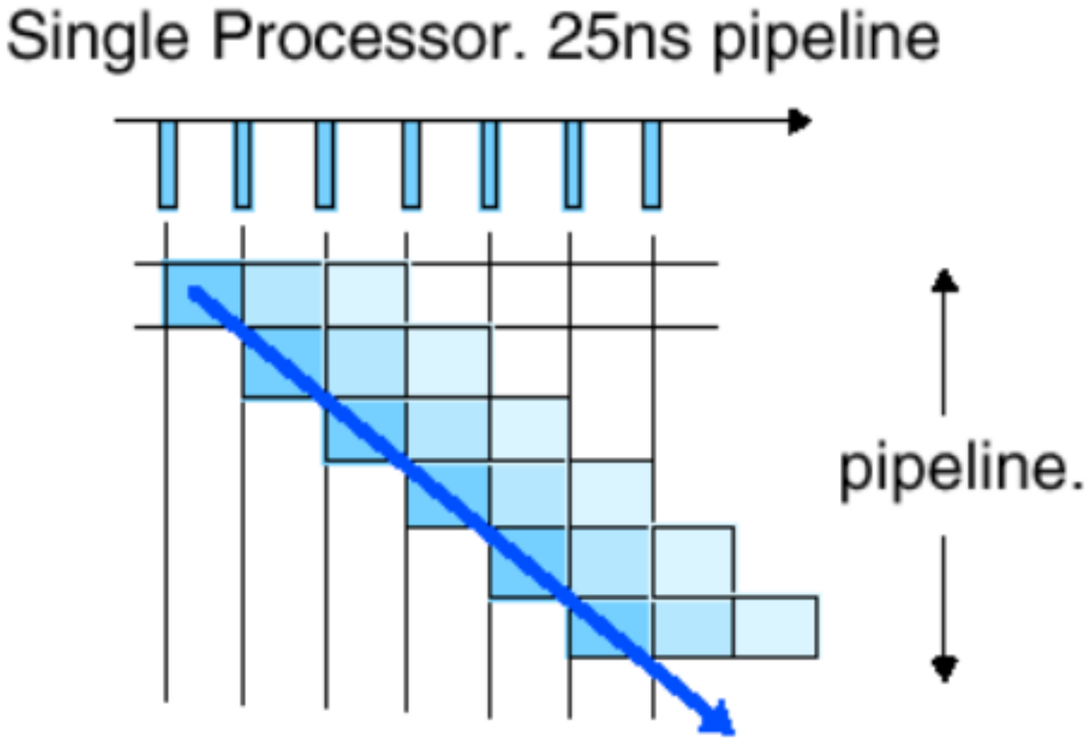


Dinyar Rabady slides

Level-1 Trigger

Pipelines

- Can’t deliver a complex trigger decision in 25 ns and can’t process only one event at a time
- **Pipeline** = multiple processing steps, events flow from step to step
  - First In First Out
- **Parallel** processing of the many inputs as much as possible
  - Processing step should fit within one bunch crossing period



Alessandro Cardini slides

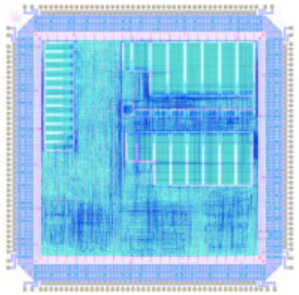
Level-1 Trigger

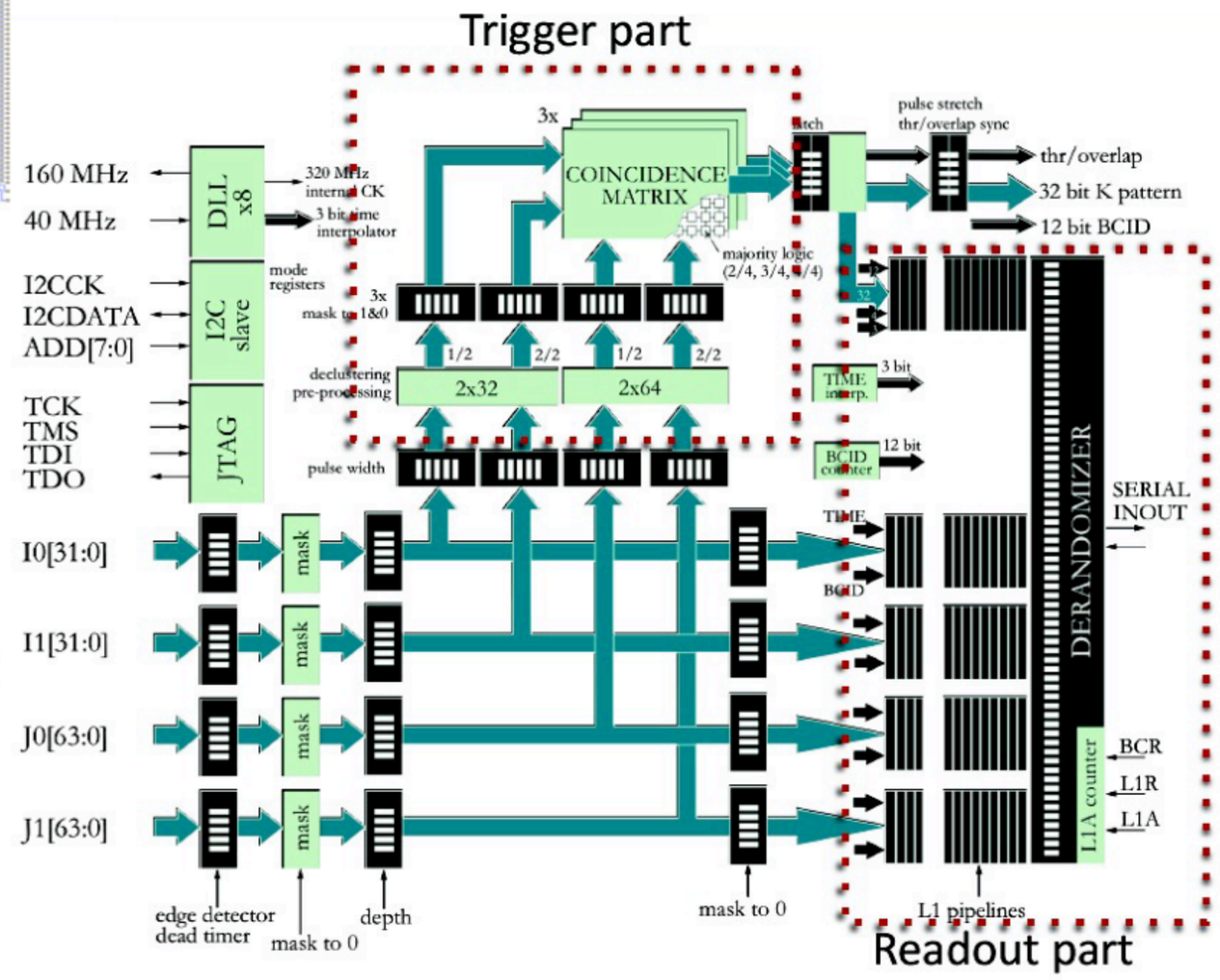
Electronics

- **Very fast custom-made electronics (parallel processing)**
  - **ASIC**s (Application-Specific Integrated Circuits)
    - Very fast
    - Not reprogrammable
    - Long development cycle
    - Very expensive for low volume
  - **FPGA**s (Field-Programmable Gate Arrays)
    - Still very fast (O(100 MHz))
    - Reprogrammable
    - Hard to program
    - Very expensive for high volume

Level-1 Trigger

Electronics: ATLAS Muon Barrel Trigger ASIC

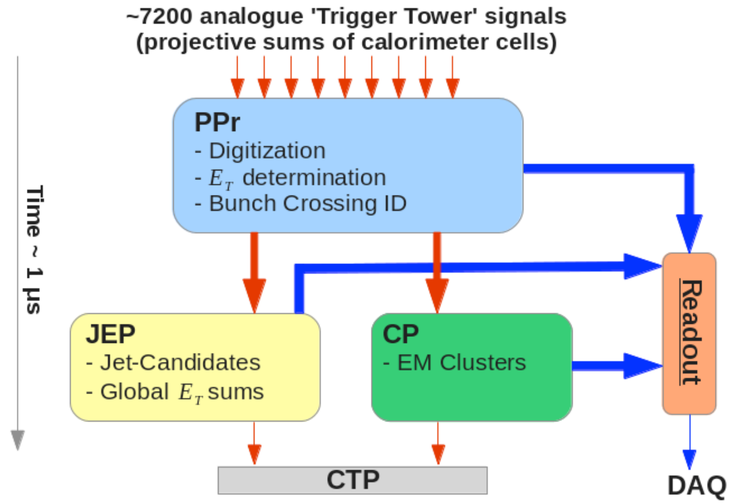




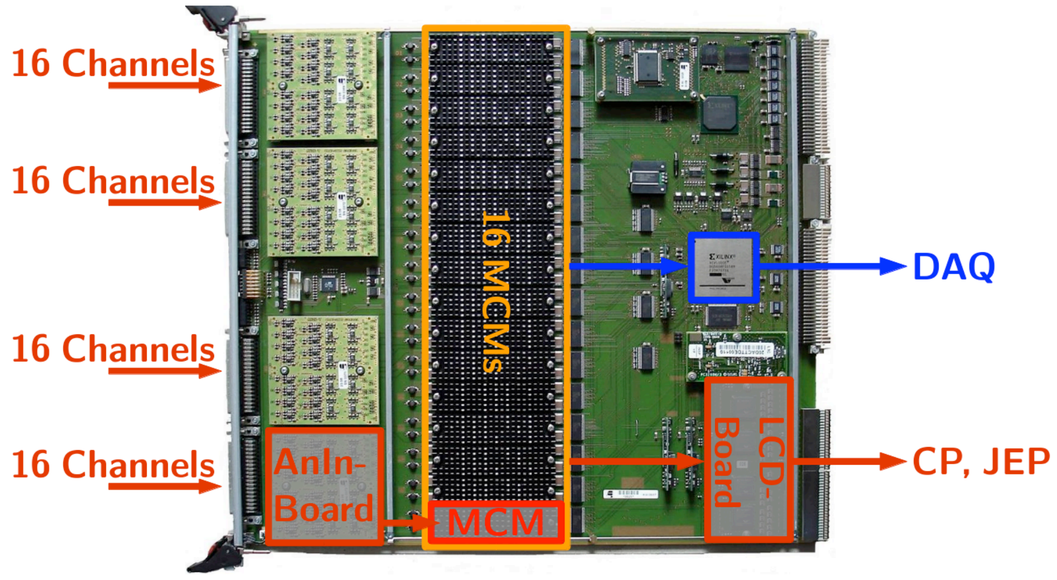
Level-1 Trigger

Electronics: ATLAS L1Calo PreProcessor

KIP (Heidelberg University) website

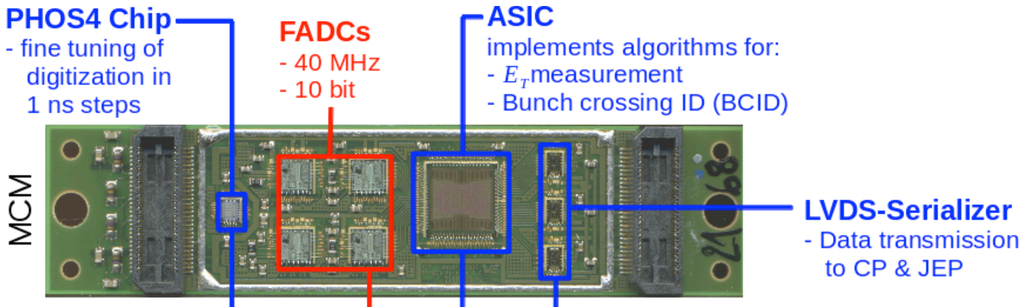


~120 X PreProcessor module (PPM)

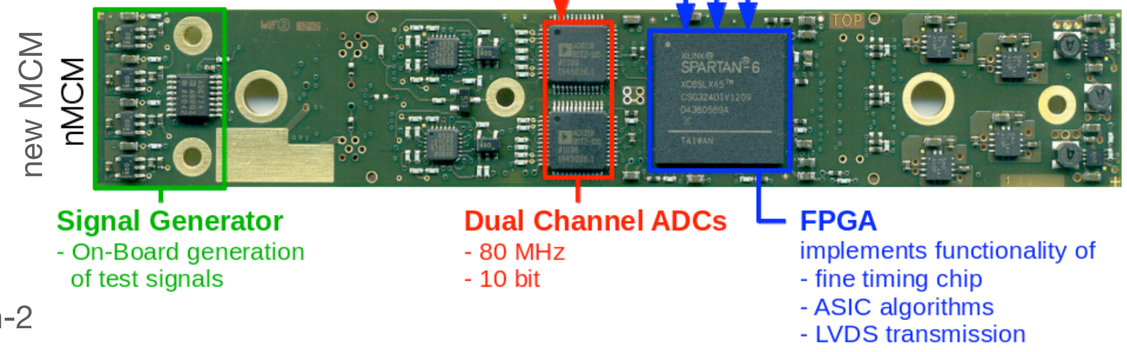


Multi-Chip Module

Run-1



Run-2



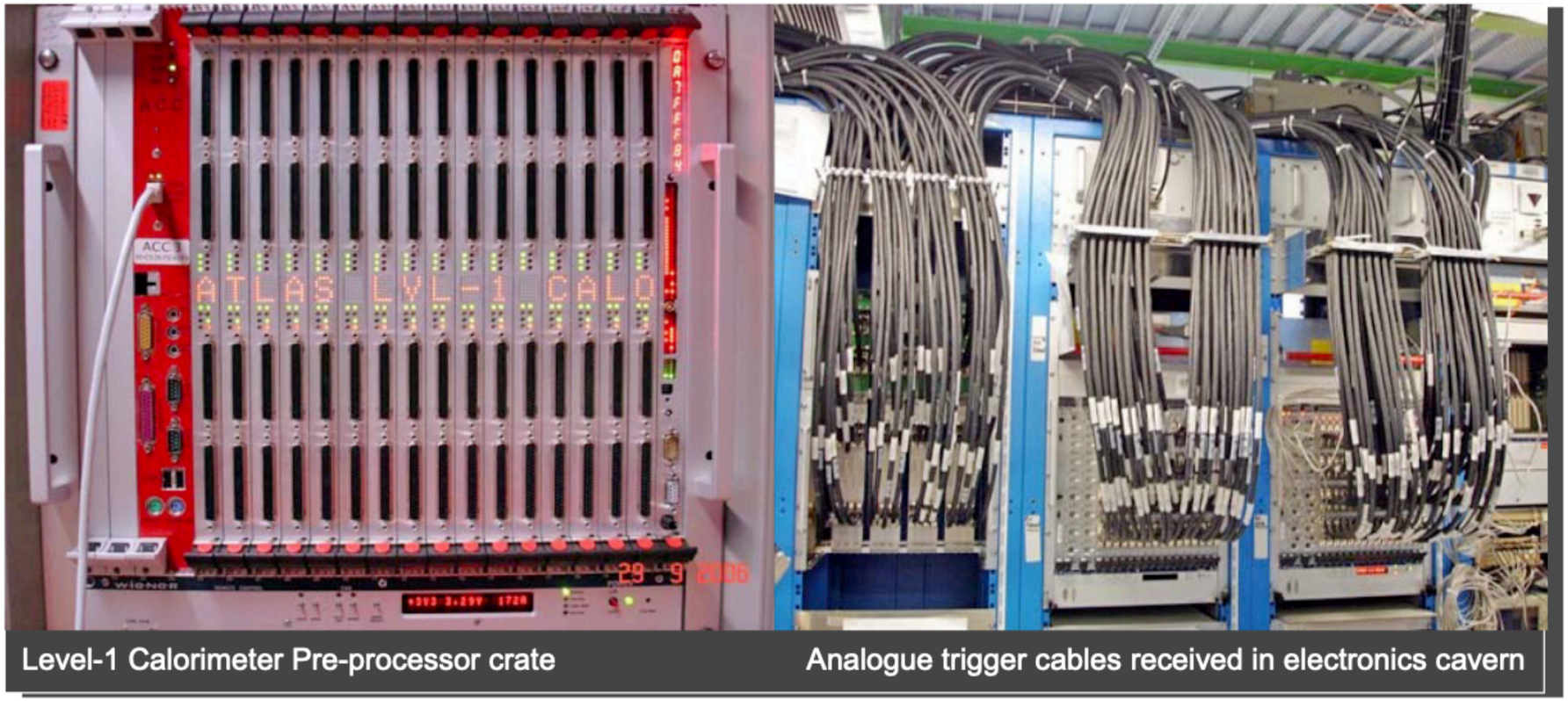
Level-1 Trigger

Electronics: ATLAS L1Calo PreProcessor

PreProcessor Module

Brian Peterson slides

~8 VME crates



Level-1 Calorimeter Pre-processor crate

Analogue trigger cables received in electronics cavern

Full L1 calorimetery system: ~27 VME crates

Good cable management is very important!

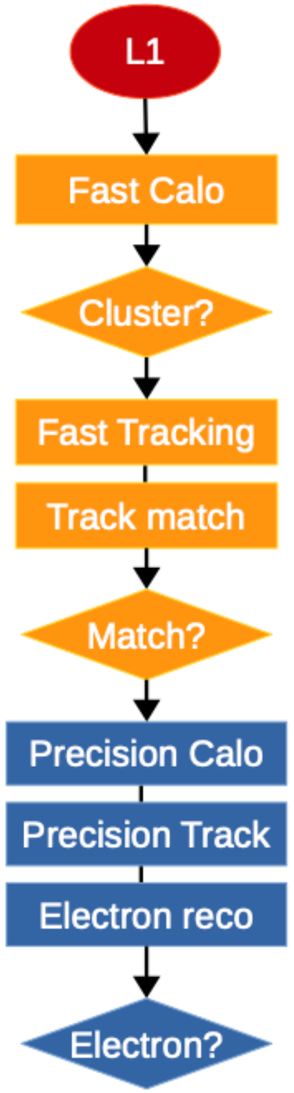
High-Level Trigger

Example: ATLAS

- Software-based running on large PC farm
- O(10k) cores running in parallel (events are independent)
- HLT processing ≪ 1s → can't use offline reconstruction (~10 s per event)
- Step-wise processing with **early rejection** (stop processing as soon as one step fails)
  - Fast reconstruction
    - offline tools + trigger specific configs
    - guided by L1 RoIs (confirm L1 result); combine with info from other detectors (tracker)
  - Precision reconstruction
    - very close to offline reconstruction
    - detector data at full granularity
- If HLT accept, the event data is written out

Typical HLT node:
2x12-core Intel Xeon Haswell
→ 96 cores/box
48 GB RAM, 10Gb Ethernet
4 motherboards in 2U box

Trigger Chain



Frank Winklmeier slides

Summary

- Challenging environment at LHC greatly influences design of trigger system
  - Large event rate
  - Large data volumes
  - Large rejection factors
- Key issues: dead-time, latency, synchronization, pipelined trigger/read-out
- **Multi-level trigger architecture**
  - first-level: fast, parallel custom electronics → trigger decision every 25 ns
  - high-level: large commercial computing farm to process events with offline-like algorithms

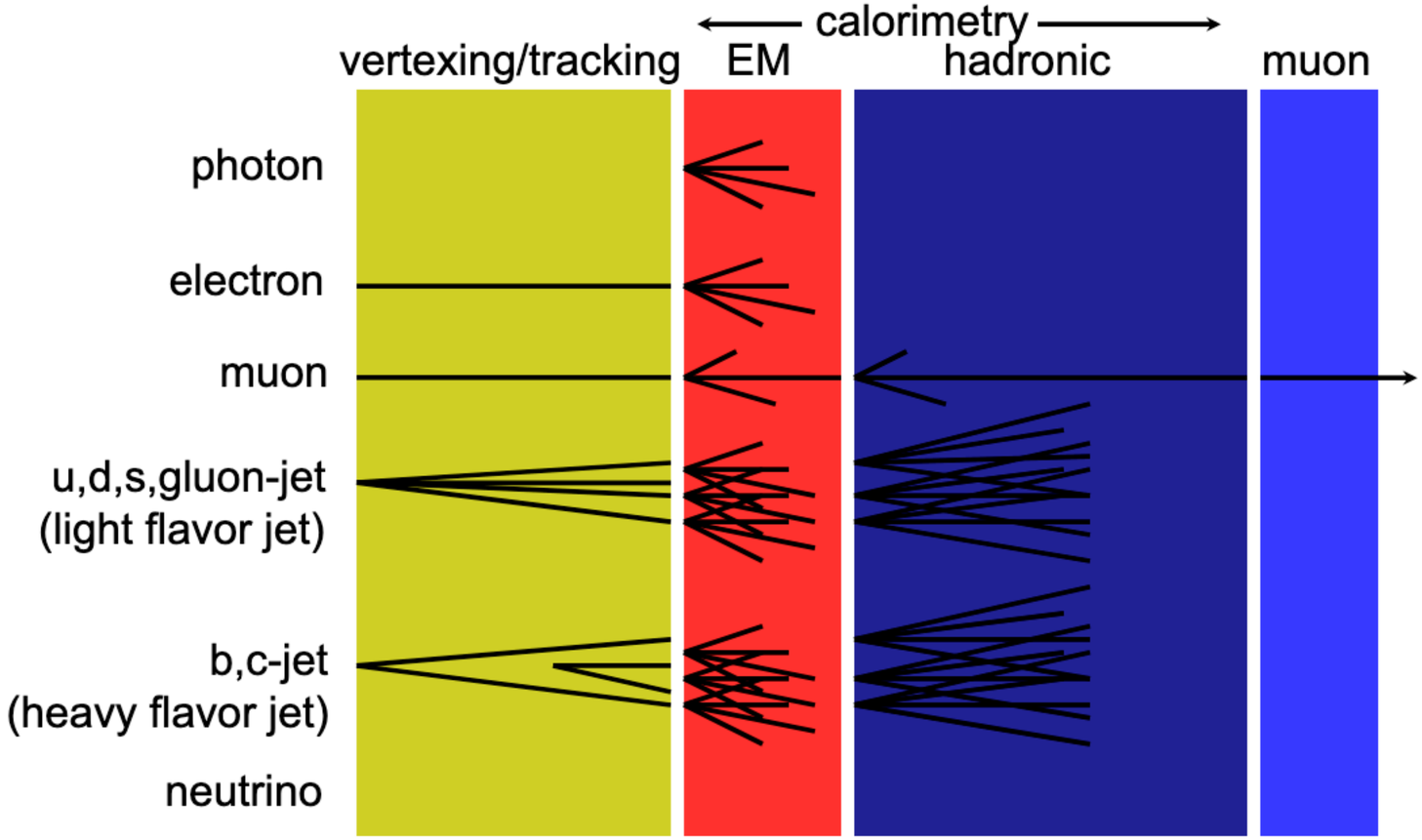
Resources

- “Triggering at High Luminosity Colliders” - https://arxiv.org/abs/0704.2548
- “ATLAS detector and physics performance” TDR - https://cds.cern.ch/record/391176/files/cer-0317330.pdf
- ATLAS RPC and L1 muon barrel trigger https://arxiv.org/pdf/2103.01029.pdf
- “Storage Systems for DAQ” - https://indico.cern.ch/event/828931/contributions/3469925/
- “Trigger architectures and hardware” - https://indico.cern.ch/event/828931/contributions/3469931/
- “Introduction to the CMS Trigger” - https://indico.cern.ch/event/58768/contributions/2057235/
- “Introduction to Trigger for Physics Workshop” - https://indico.cern.ch/event/558579/contributions/2253609/
- “Trigger and Data Acquisition” - https://indico.cern.ch/event/115062/
- “Trigger and Data Acquisition at the Large Hadron Collider” - http://www.le.infn.it/lhcschool/talks/Cardini-1.pdf
- “The ATLAS Trigger & Data Acquisition System” - https://indico.cern.ch/event/860971/contributions/3626496/
- “LHC experiments - Trigger, Data-Taking and Computing” - https://indico.mpp.mpg.de/event/5470/
- “Electronic, Trigger and Data Acquisition” - https://indico.cern.ch/event/190068/

Backup

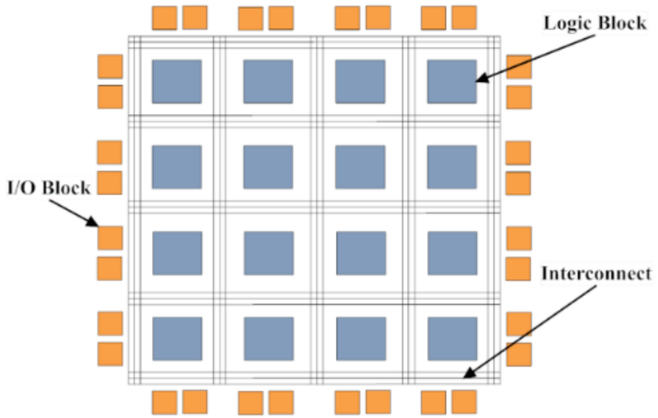
Particle ID

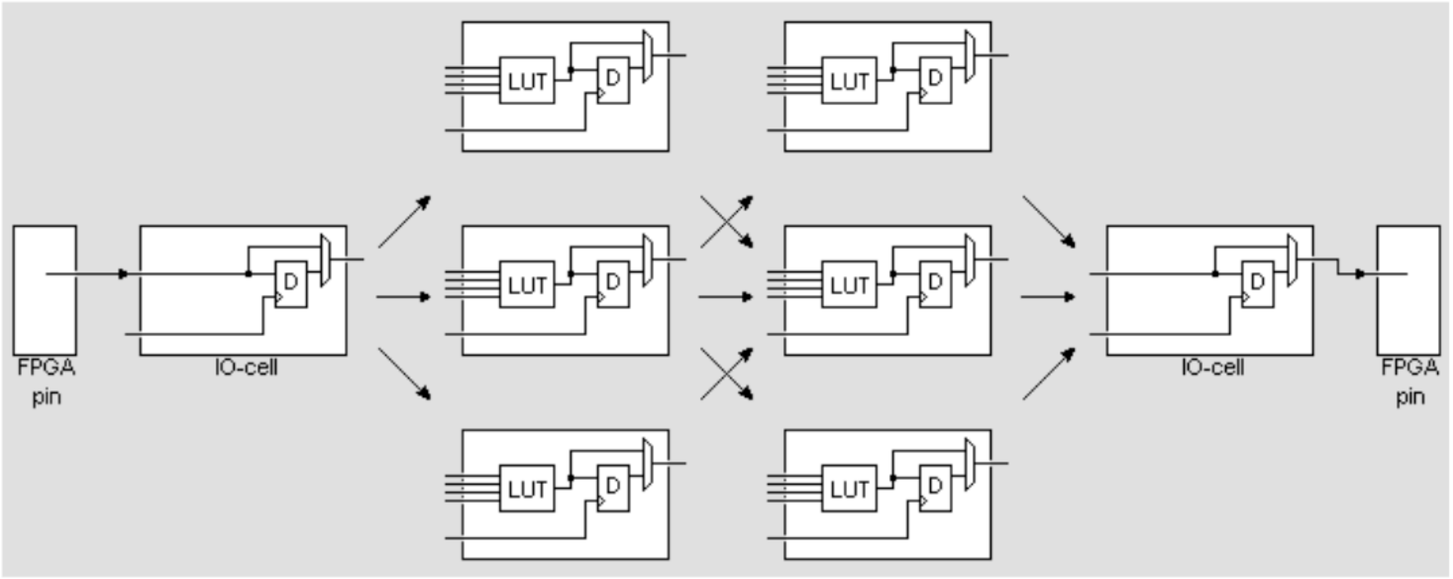
https://indico.cern.ch/event/77805/attachments/1057246/1507528/CERN100120-v2.pdf



Level-1 Trigger

FPGA Design





- **Example of a WAIT statement (Programming Language VS. HDL)**
  - In programming language (e.g. C) (Unix, #include <unistd.h>)

```
sleep(5); // sleep 5 seconds
```

  - In HDL (e.g. VHDL):
    - Not synthesizable (only for simulation test benches)

```
wait for 5 sec;  -- handy for TB clocks
```

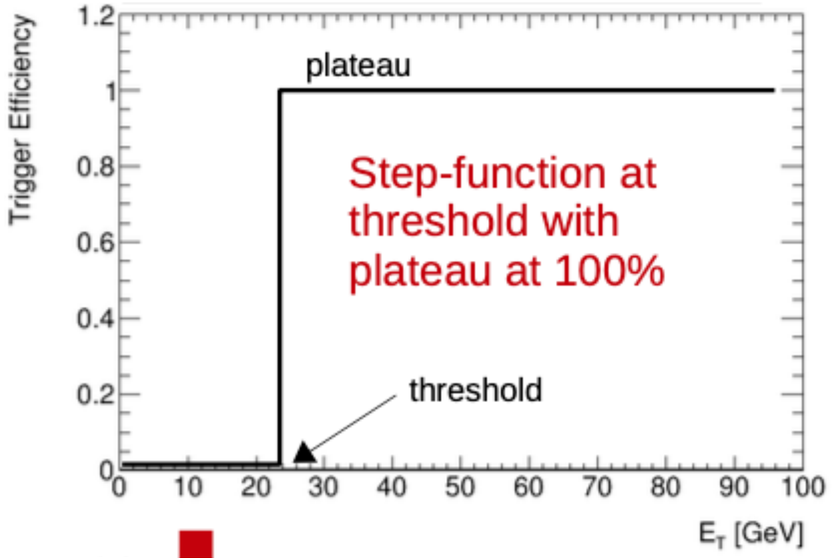
    - Synthesizable (for simulation and/or FPGA implementation)

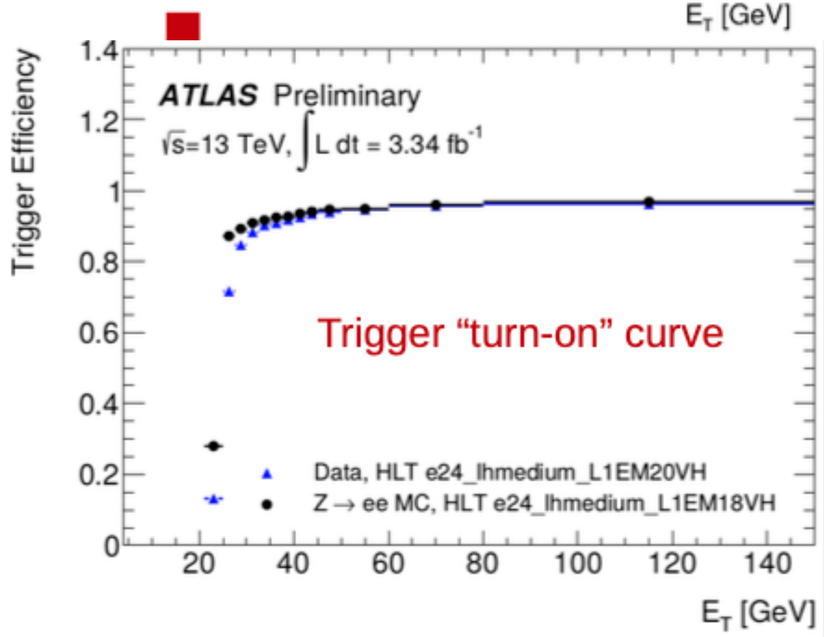
```
simple_delay_counter : process (delay_rst, delay_clk, delay_ena)
begin  -- process
  if delay_rst = '1' then
    s_count    <= delay_ld_value;
    s_delay_done <= '0';
  elsif rising_edge(delay_clk) then
    if delay_ena = '1' then
      if delay_ld = '1' then
        s_count <= delay_ld_value;
      else
        s_count <= s_count - 1;
      end if;
    end if;
    if s_count = 0 then
      s_delay_done <= '1';
    else
      s_delay_done <= '0';
    end if;
  end if;
 end process;
```

Trigger efficiencies

https://indico.cern.ch/event/558579/contributions/2253609/attachments/1359832/2058795/20161025-TriggerWorkshop-Intro.pdf

- Selection efficiency $\epsilon_{\text{trigger}} = \frac{N_{\text{trigger}}}{N_{\text{offline}}}$
  - as high as possible
  - bias free
  - known as precise as possible
- Measure by
  - tag-and-probe
    - trigger on one particle, see how well you do on second
  - boot-strap
    - use looser trigger
  - orthogonal
    - trigger on one signature, measure a different one
  - simulation

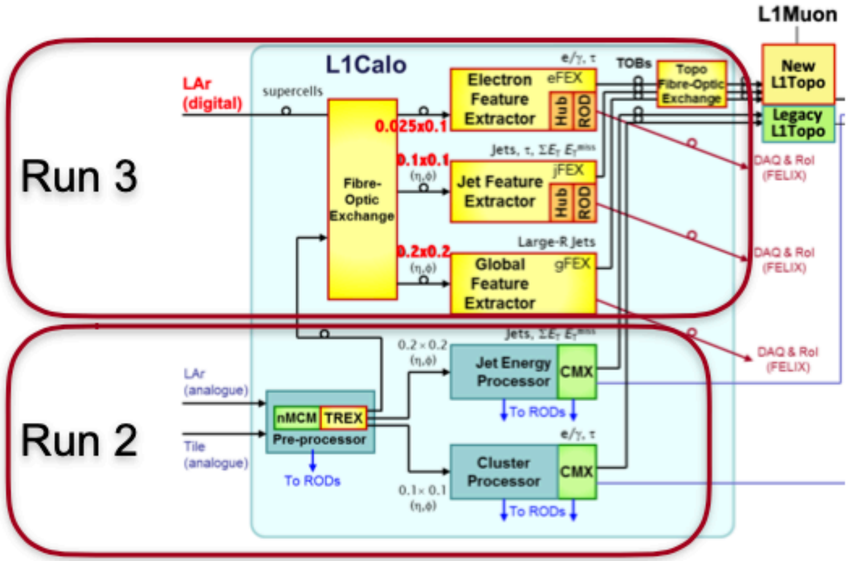




ATLAS Trigger for Run-3

https://indico.cern.ch/event/860971/contributions/3626496/attachments/1971759/3280808/Trigger_CBernius_Induct200120.pdf

- L1Calo Run 3 upgrade with digital processors **(FEXs)**
  - **better isolation**
  - **better pile-up and background rejection**
- Will **free up** part of the **L1 bandwidth** to be used by other triggers
- Challenge to commission the new L1Calo (and L1Topo) system in the beginning of Run 3 with the Run 2 system running in parallel



- In Run 2: AthenaMP (MultiProcessing)
- Run 3 software is based on Multi-Threading (MT) which means **algorithms using data from one event can be parallelized** and multi-events can also run in parallel (**AthenaMT/release 22/master**)