Imperial College London

# The LHC and the *really big data* challenge

Instrumentation course guest lecture

Andrew W. Rose, Imperial College London

awr01@imperial.ac.uk

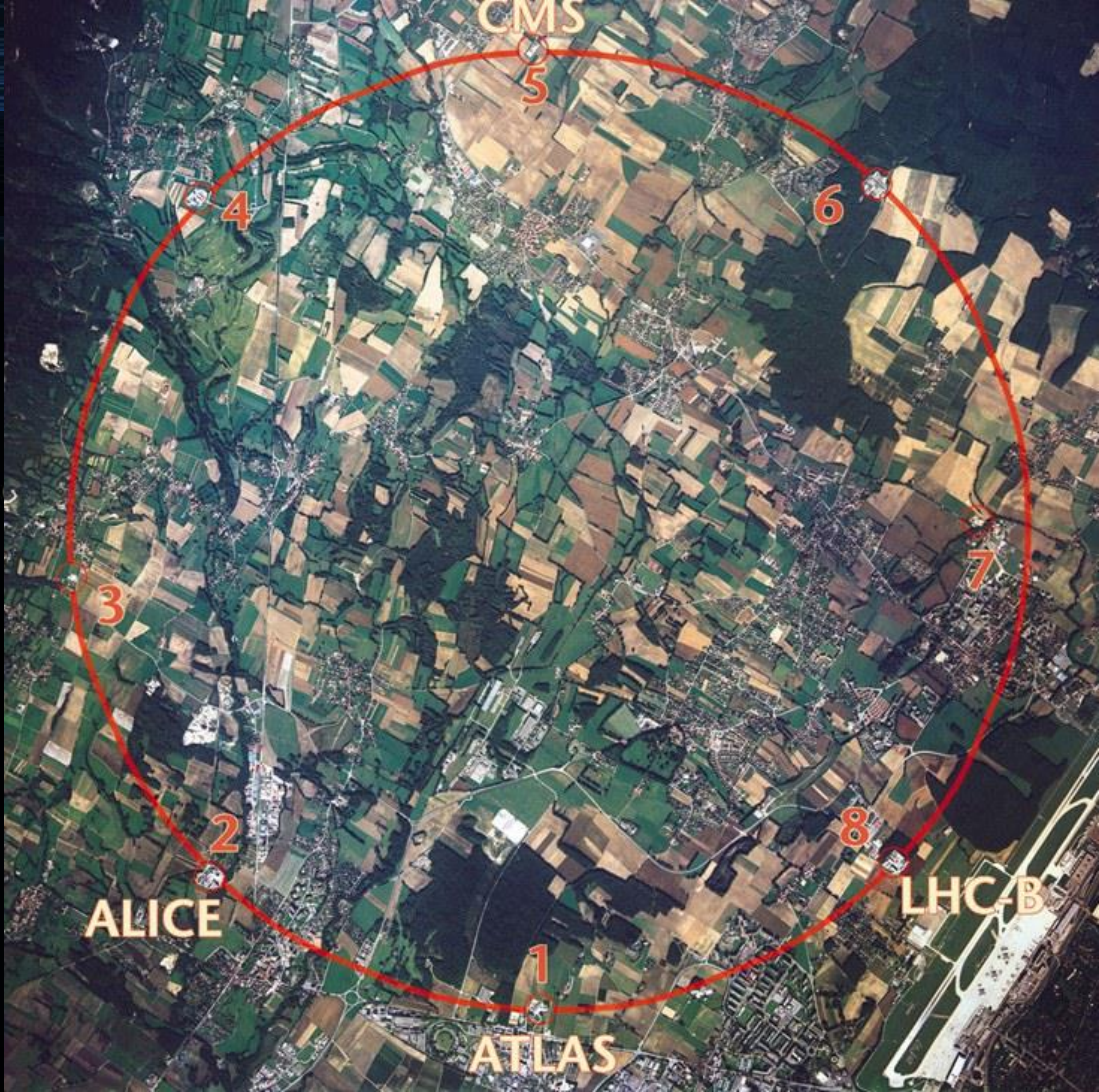
I'll upload a copy of these slides to www.hep.ph.ic.ac.uk/~awr01

# Introduction

- I hope you will all be familiar to some extent with the

  **Large Hadron Collider**

  at CERN, Switzerland but will assume minimal background knowledge
- The largest and most complex scientific endeavour in history
- Most famous for its observation of the Higgs boson



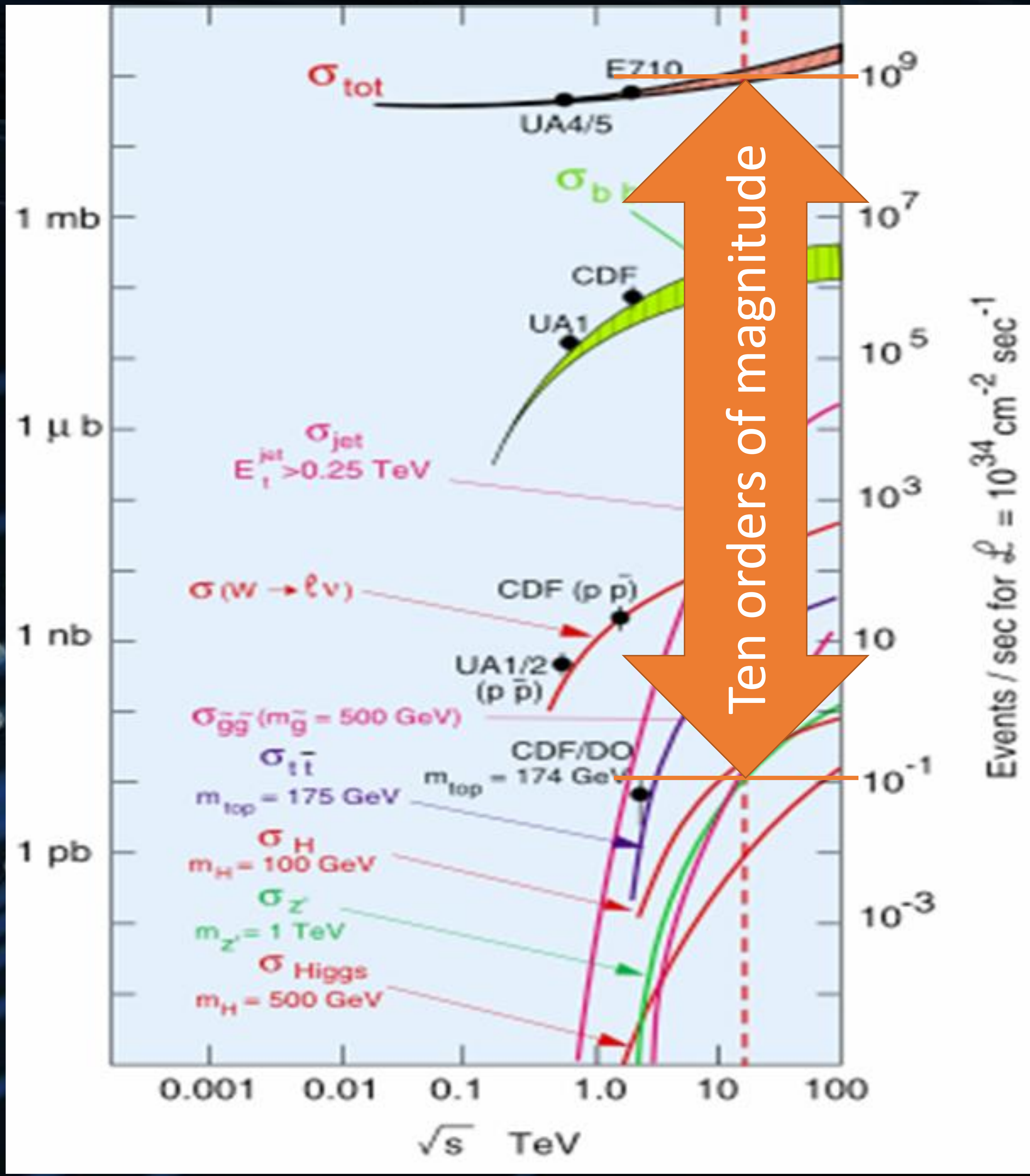
# Science: The basics

- A bold statement:

**Science is the art of knowing what to record, and when**

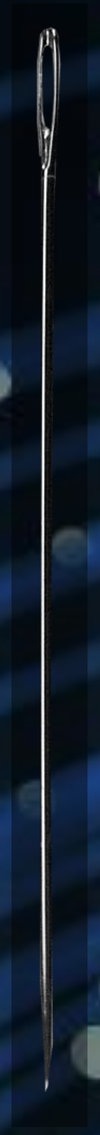
# Science: The basics

- Science is the art of knowing what to record, and when
- With CMS & ATLAS in "discovery mode", we care about the Higgs Boson or rarer
  - Higgs Boson production is ten orders of magnitude below the total interaction rate



# Science: The basics

- Science is the art of knowing what to record, and when
- With CMS & ATLAS in "discovery mode", we care about the Higgs Boson or rarer
  - Higgs Boson production is ten orders of magnitude below the total interaction rate
  - That is a needle in a haystack the same mass as the Empire State Building



# Science: The basics

- Science is the art of knowing what to record, and when
- With CMS & ATLAS in "discovery mode", we care about the Higgs Boson or rarer
  - Higgs Boson production is ten orders of magnitude below the total interaction rate
  - That is a needle in a haystack the same mass as the Empire State Building
  - Or if you collided protons once per second, that is one event every **317 years**

# Science: The basics

- Science is the art of knowing what to record, and when
- With CMS & ATLAS in “discovery mode”, we care about the Higgs Boson or rarer
  - Higgs Boson production is ten orders of magnitude below the total interaction rate
  - That is a needle in a haystack the same mass as the Empire State Building
  - Or if you collided protons once per second, that is one event every **317 years**

And one Higgs boson is pretty much

**USELESS**

We need loads of them to be able to study them…

# This is why the LHC collides…

~50 protons at a time

×

40 million times a second

# This is why the LHC collides…

~50 protons at a time

×

40 million times a second

---

1 Higgs boson every ~5 seconds

# This is why the LHC collides...

~50 protons at a time

×

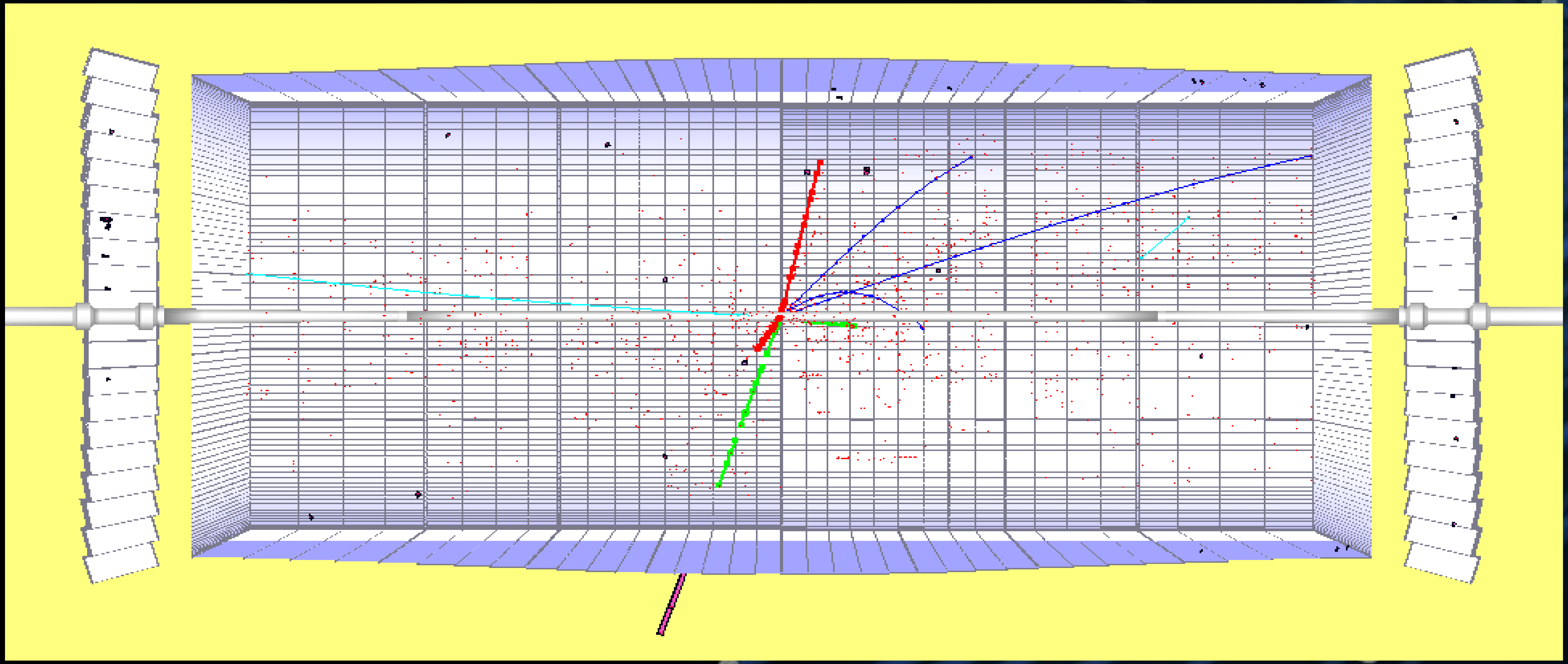
40 million times a second

---

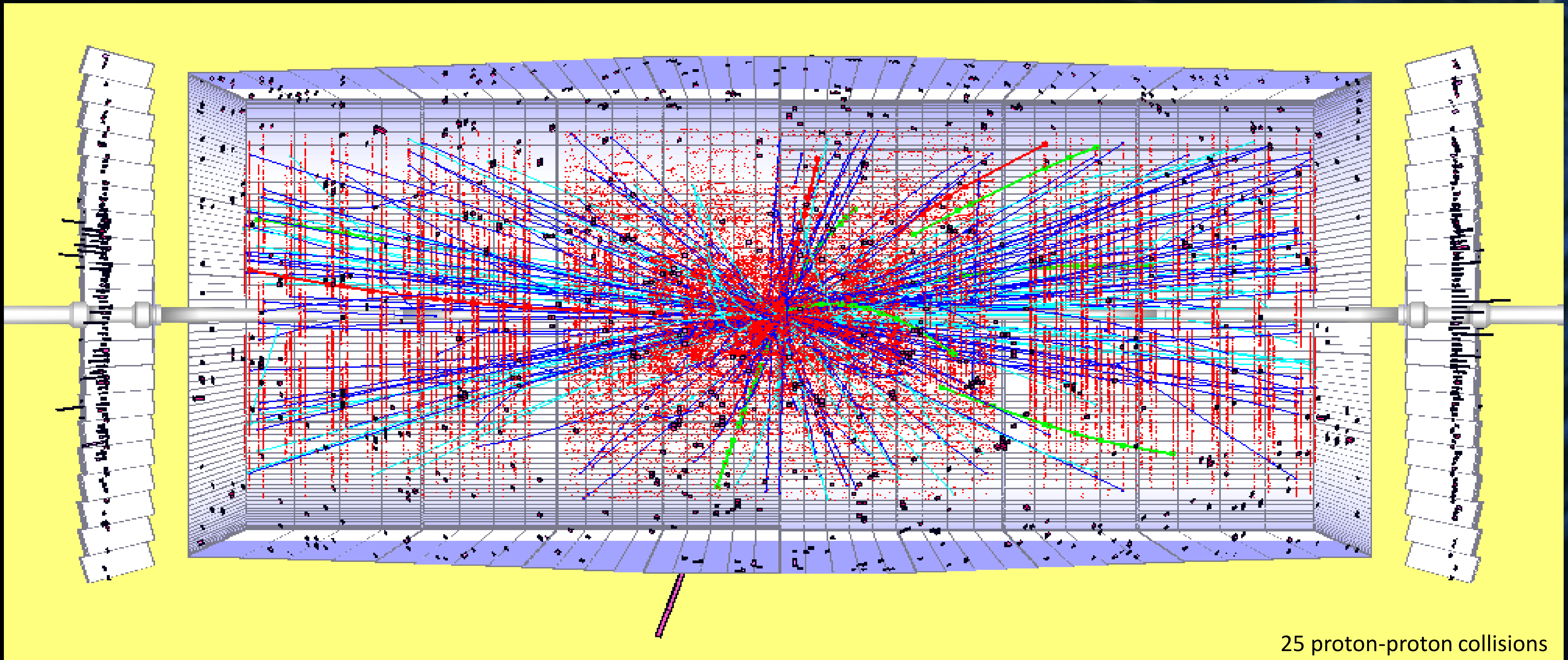
1 Higgs boson every ~5 seconds

But this causes problems

# One proton-proton collision



# Many proton-proton collisions



25 proton-proton collisions

# Many proton-proton collisions


Even if you get a Higgs boson, the

**RUBBISH**

makes it hard to find

# This is why the LHC collides…

~50 protons at a time

×

40 million times a second

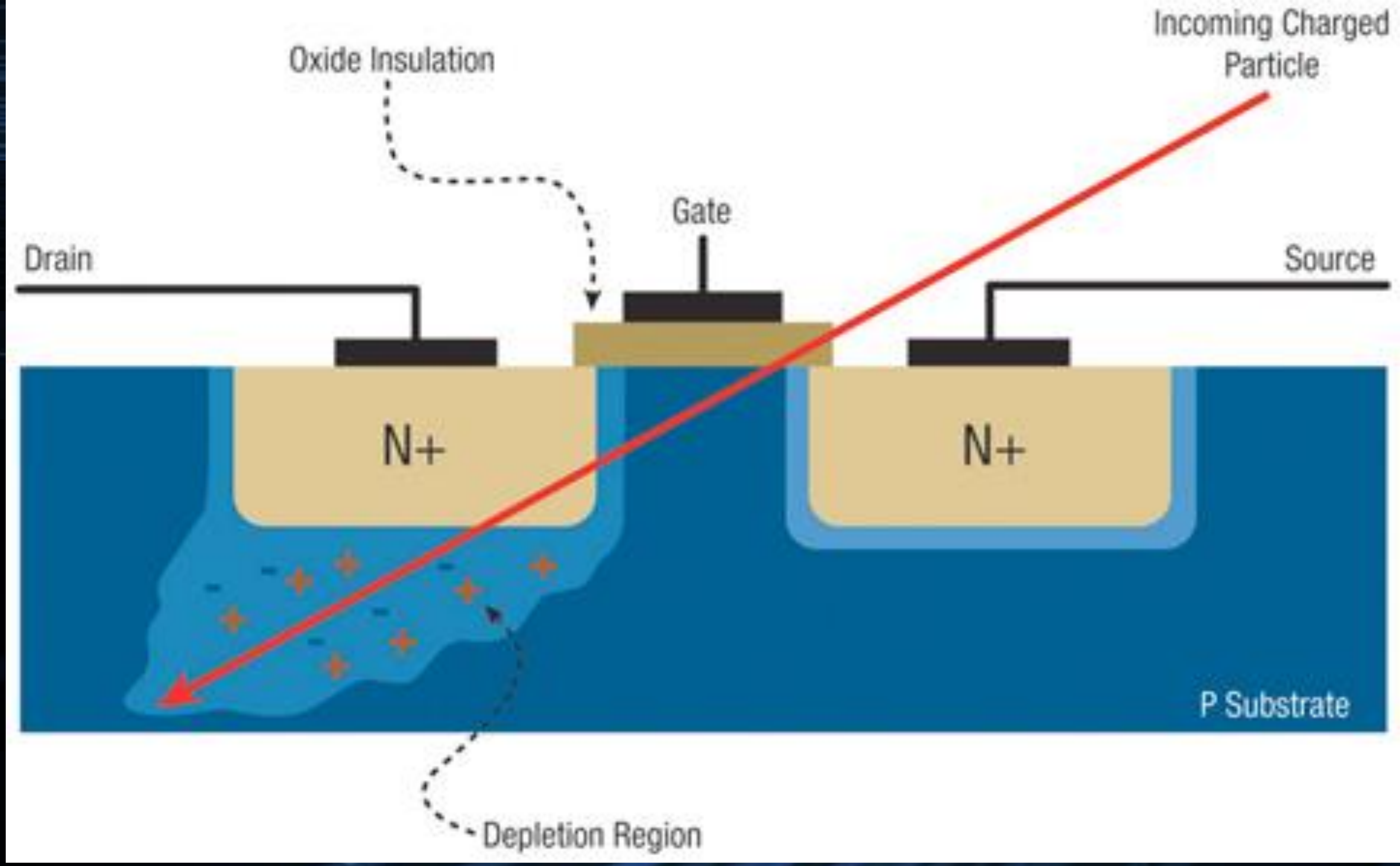
---

1 Higgs boson every ~5 seconds

This causes other problems too

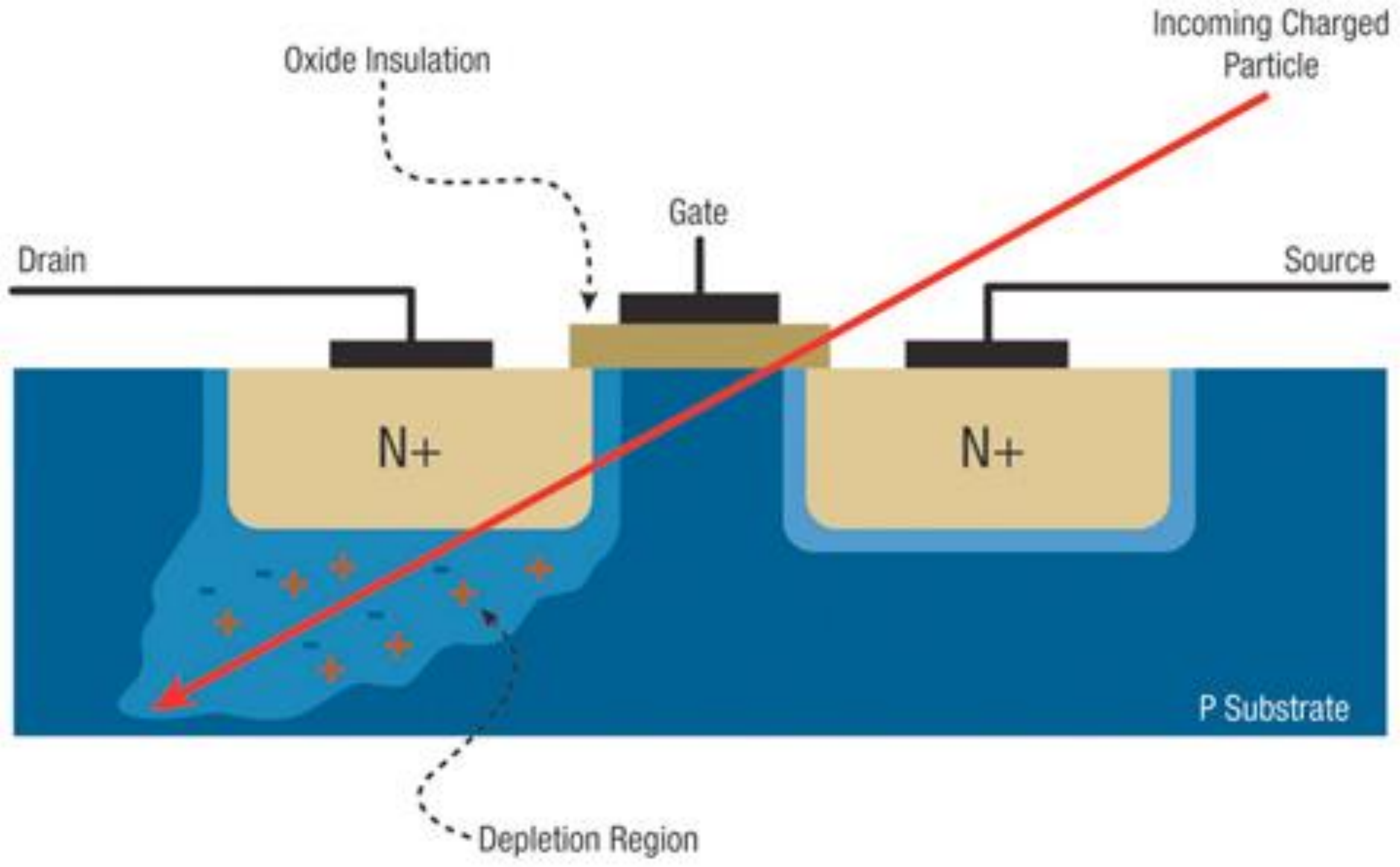
# Single Event Upsets

- When energetic charged particles pass through a transistor, they can change the state
  - And there are plenty of charged particles in the LHC experiments



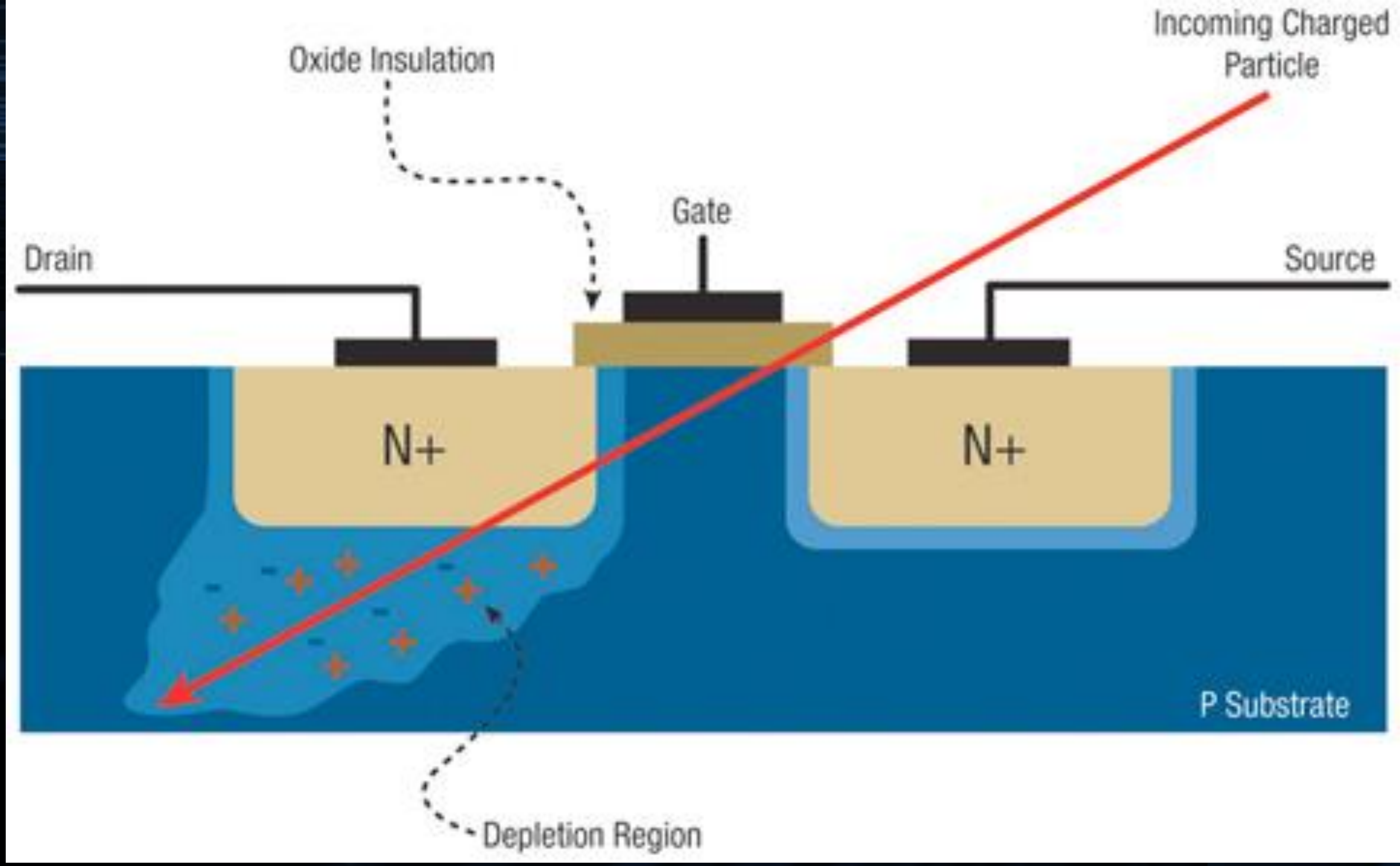
# Single Event Upsets

- When energetic charged particles pass through a transistor, they can change the state
  - And there are plenty of charged particles in the LHC experiments
- In a microprocessor
  - At best this corrupts the data
  - At worst it changes the program flow



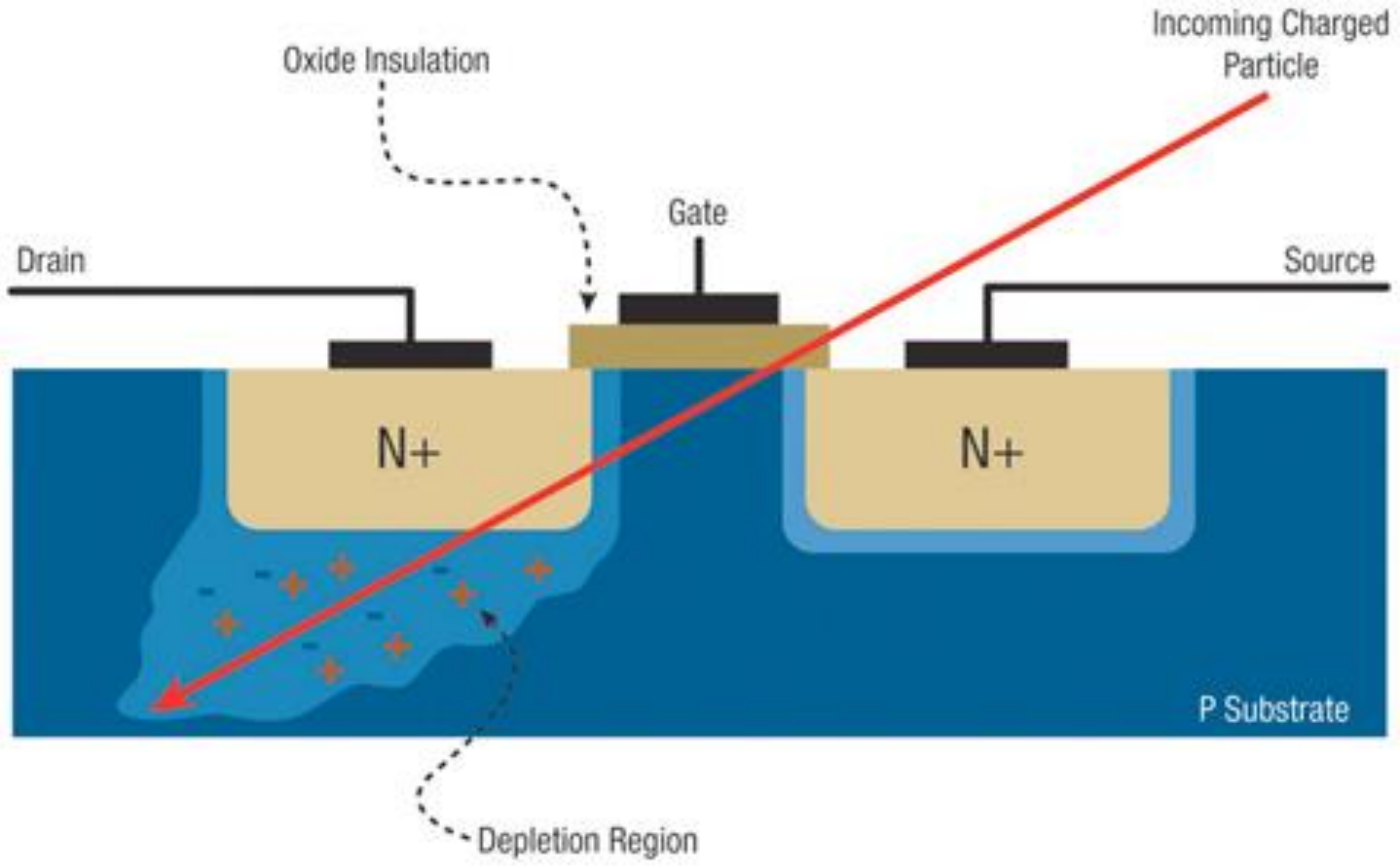
# Single Event Upsets

- When energetic charged particles pass through a transistor, they can change the state
  - And there are plenty of charged particles in the LHC experiments
- In a microprocessor
  - At best this corrupts the data
  - At worst it changes the program flow
- There are much, much worse effects
  - Entire circuits reconfigured
  - Transistor shorts power to ground – burn out the chip
  - Block the ability to reset or reconfigure the chip



# Single Event Upsets

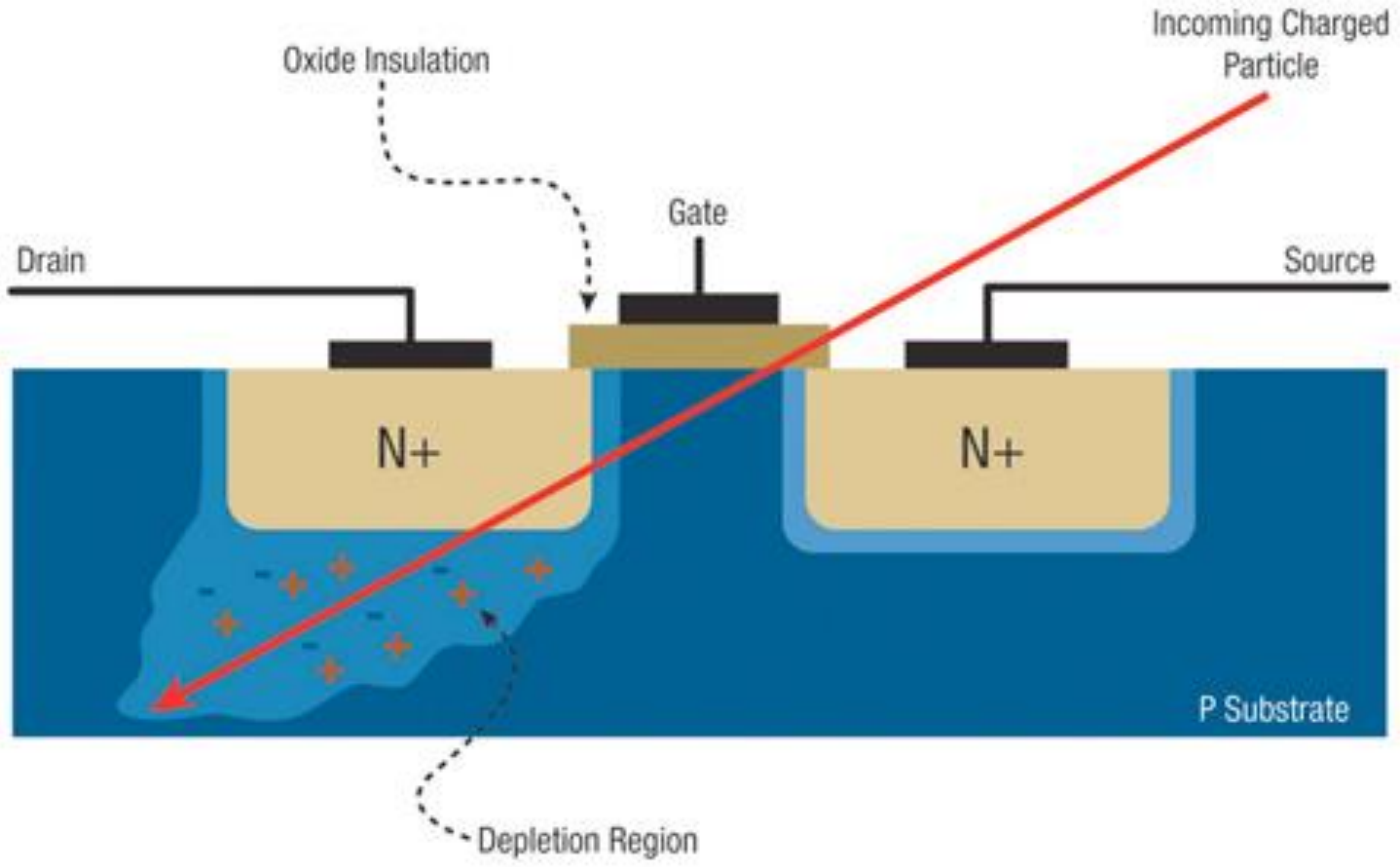
- Mitigate by having three copies of the logic and arbitrate by majority logic
  - Triple Redundant Logic
- Deep well transistor architectures to minimize charge collection
- Insulating substrates rather than semiconductors
  - Diamond, Sapphire, …
- Or wide band gap substrates
  - Silicon Carbide, Gallium Nitride
- Continual reconfiguration
  - Called "Scrubbing"




Designing radiation-hard electronics is a specialist skill

# Single Event Upsets

- Mitigate by having three copies of the logic and arbitrate by majority logic
  - Triple Redundant Logic
- Deep well transistor architectures to minimize charge collection
- Insulating substrates rather than semiconductors
  - Diamond, Sapphire, …
- Or wide band gap substrates
  - Silicon Carbide, Gallium Nitride
- Continual reconfiguration
  - Called "Scrubbing"




The best solution is to keep your most sensitive electronics as far away from radiation as possible

# This is why the LHC collides...

~50 protons at a time

×

40 million times a second

---

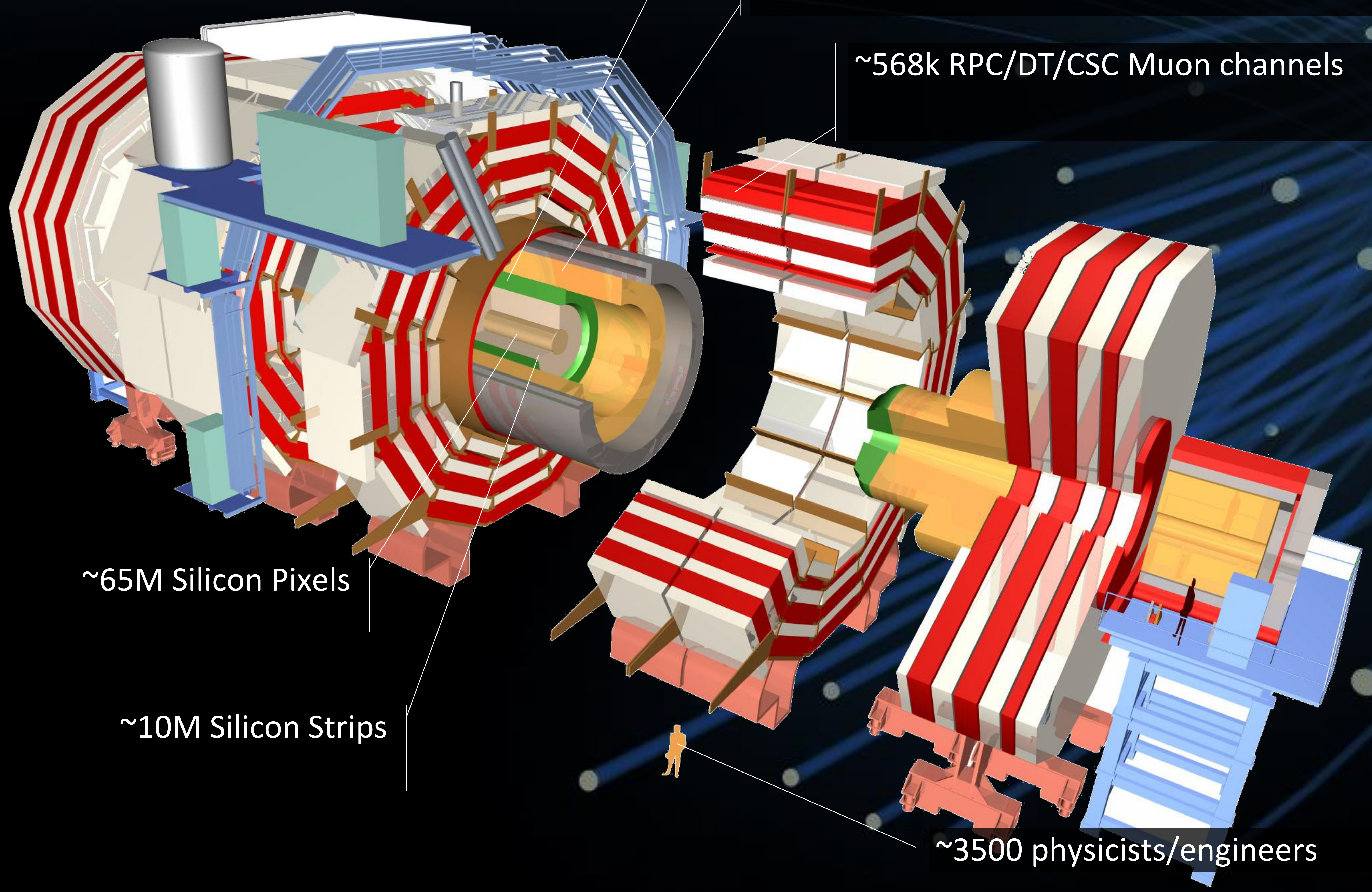
1 Higgs boson every ~5 seconds

But this causes problems

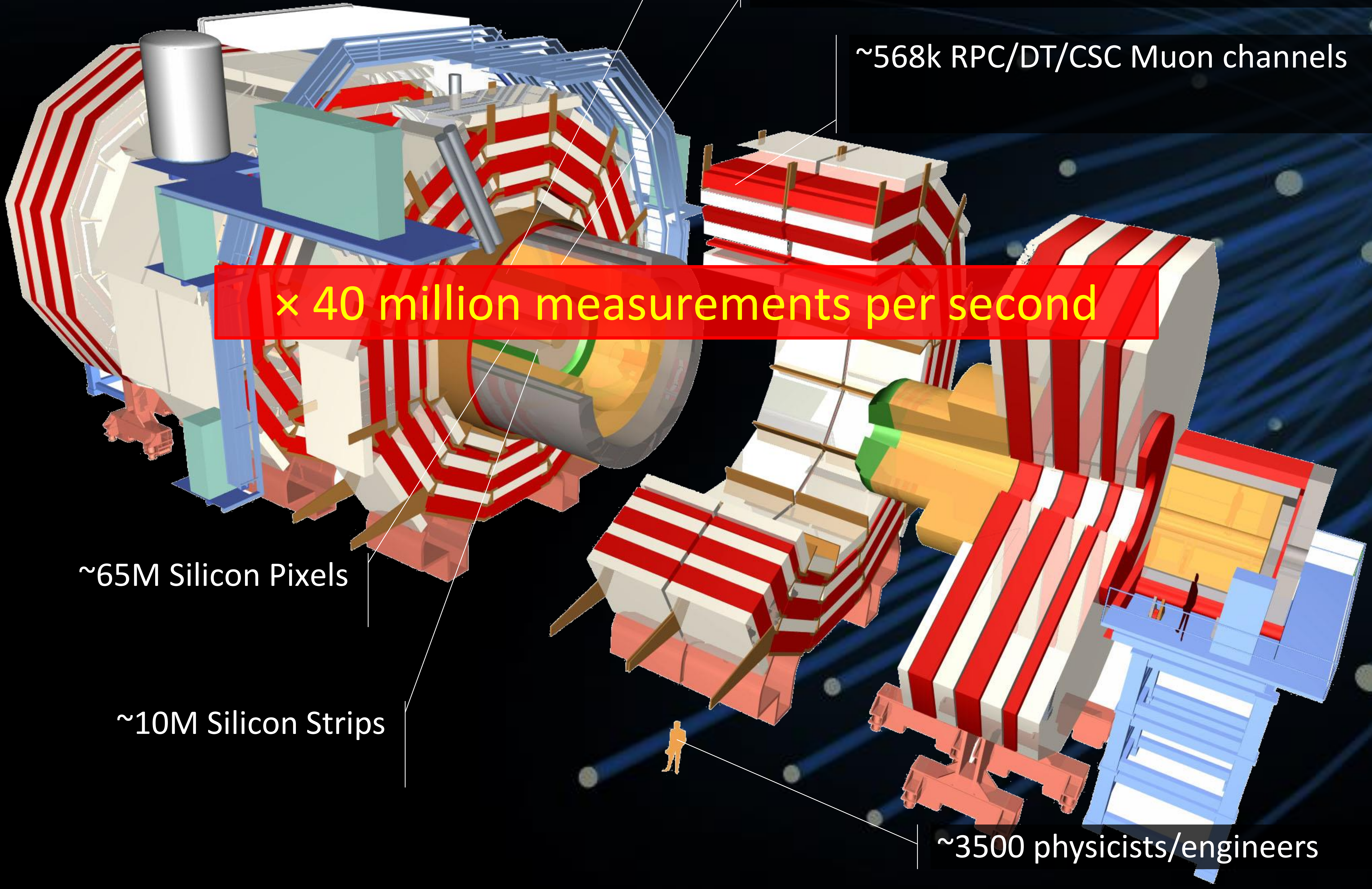
This causes problems too

# Big detectors for small particles

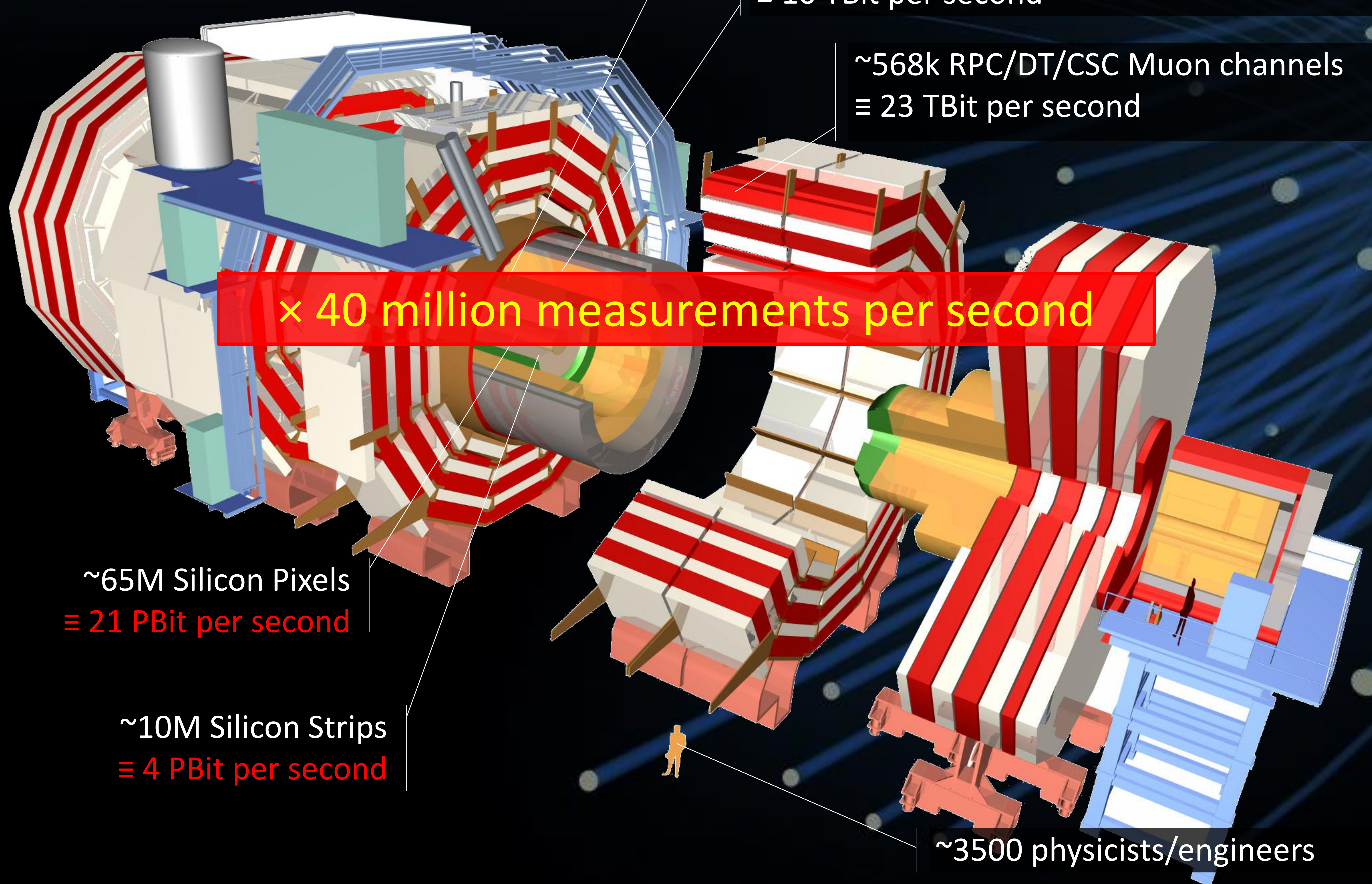
# Big detectors…



# Big detectors…



# … Bigger data


~80k $PbWO_4$ Ecal Crystals
≡ 40 TBit per second
~15k channel Brass/Plastic sampling HCAL
≡ 10 TBit per second
~568k RPC/DT/CSC Muon channels
≡ 23 TBit per second
× 40 million measurements per second
~65M Silicon Pixels
≡ 21 PBit per second
~10M Silicon Strips
≡ 4 PBit per second
~3500 physicists/engineers

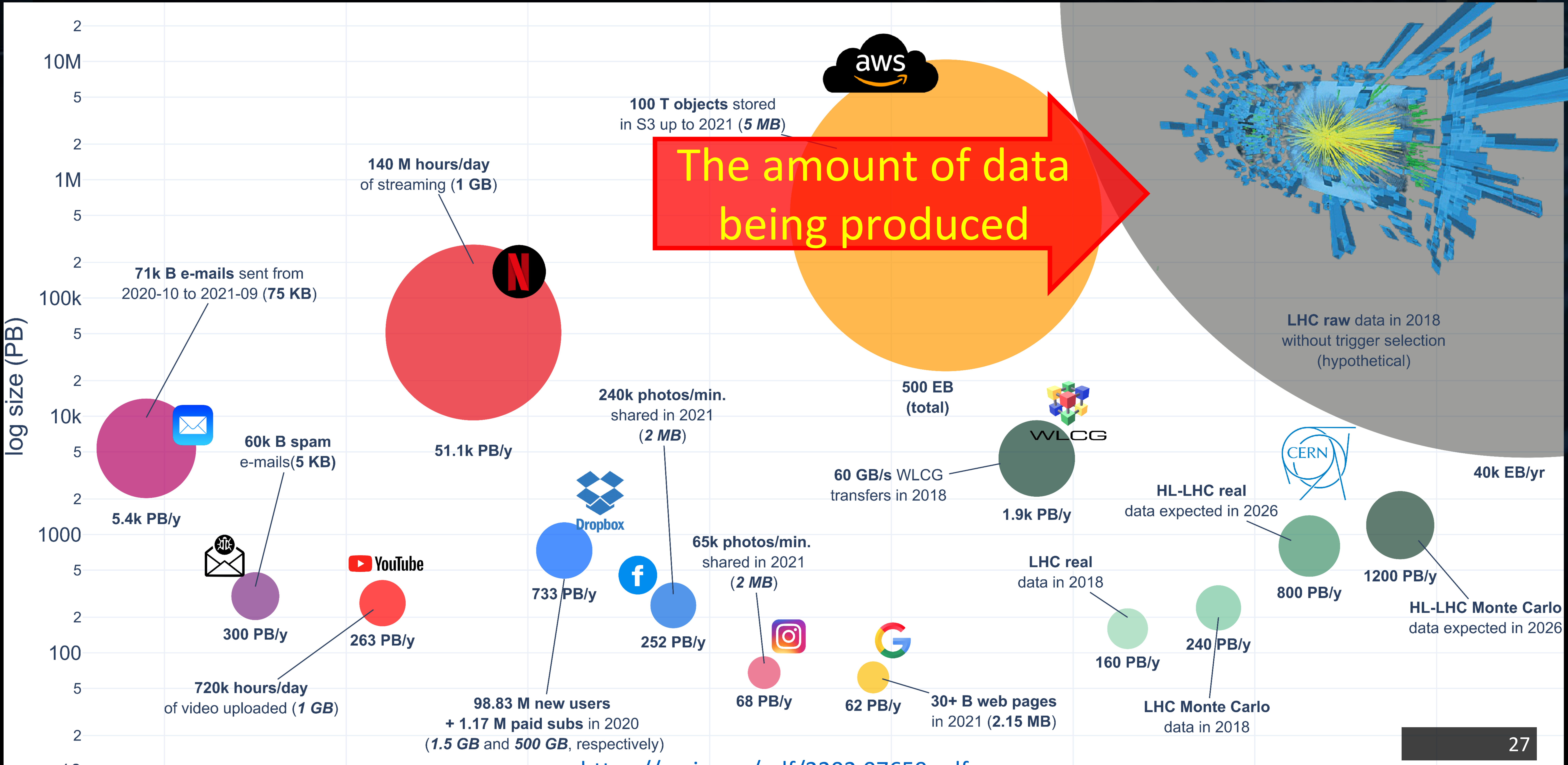
# … Bigger data


**What does that even mean?**

# What does that even mean?


https://www.cisco.com/

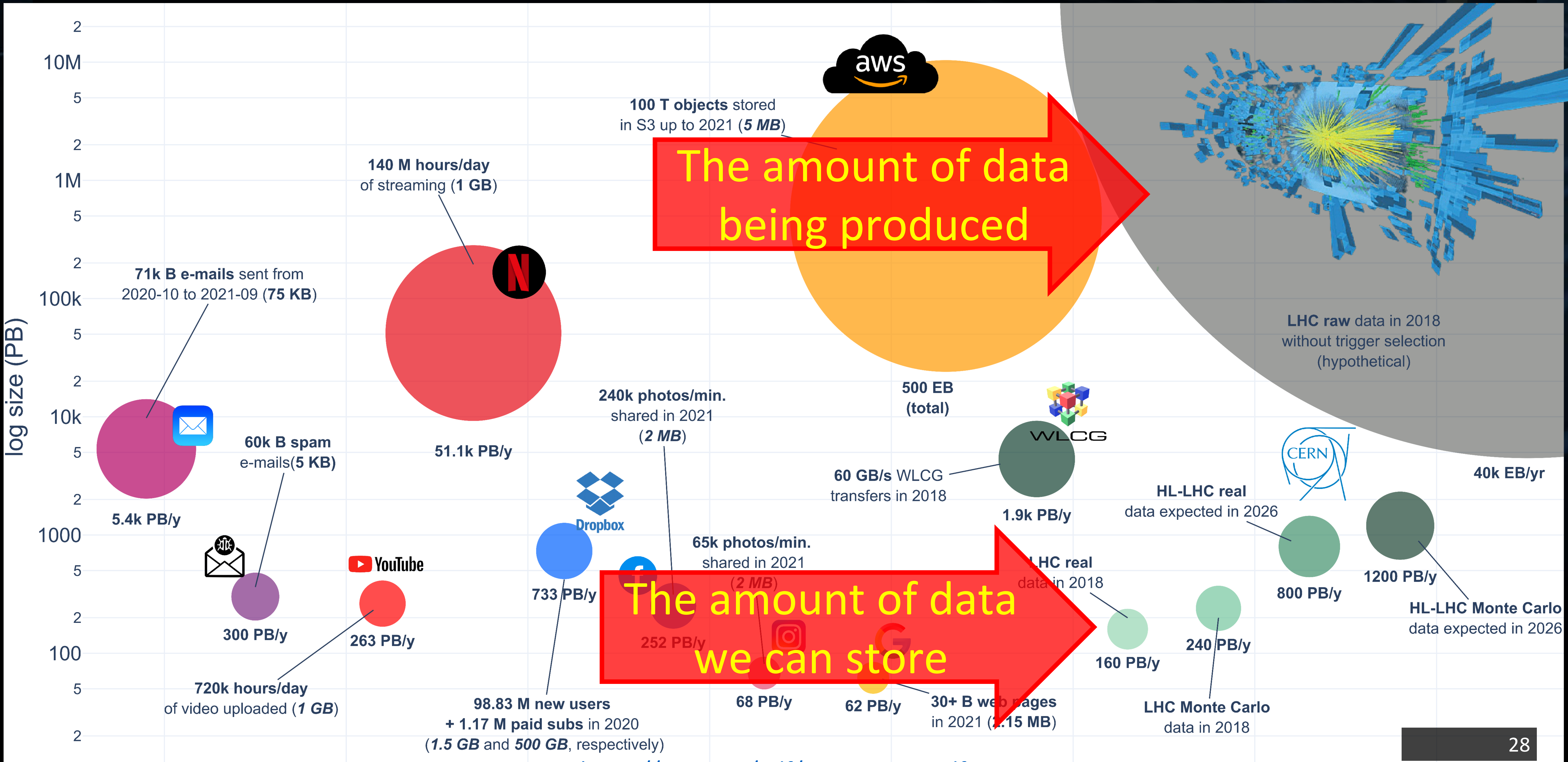
# What does that even mean?






https://arxiv.org/pdf/2202.07659.pdf

# What does that even mean?




https://arxiv.org/pdf/2202.07659.pdf

# What does that even mean?


**But do we need to store it all?**

**The Higgs is rare, after all**

https://arxiv.org/pdf/2202.07659.pdf

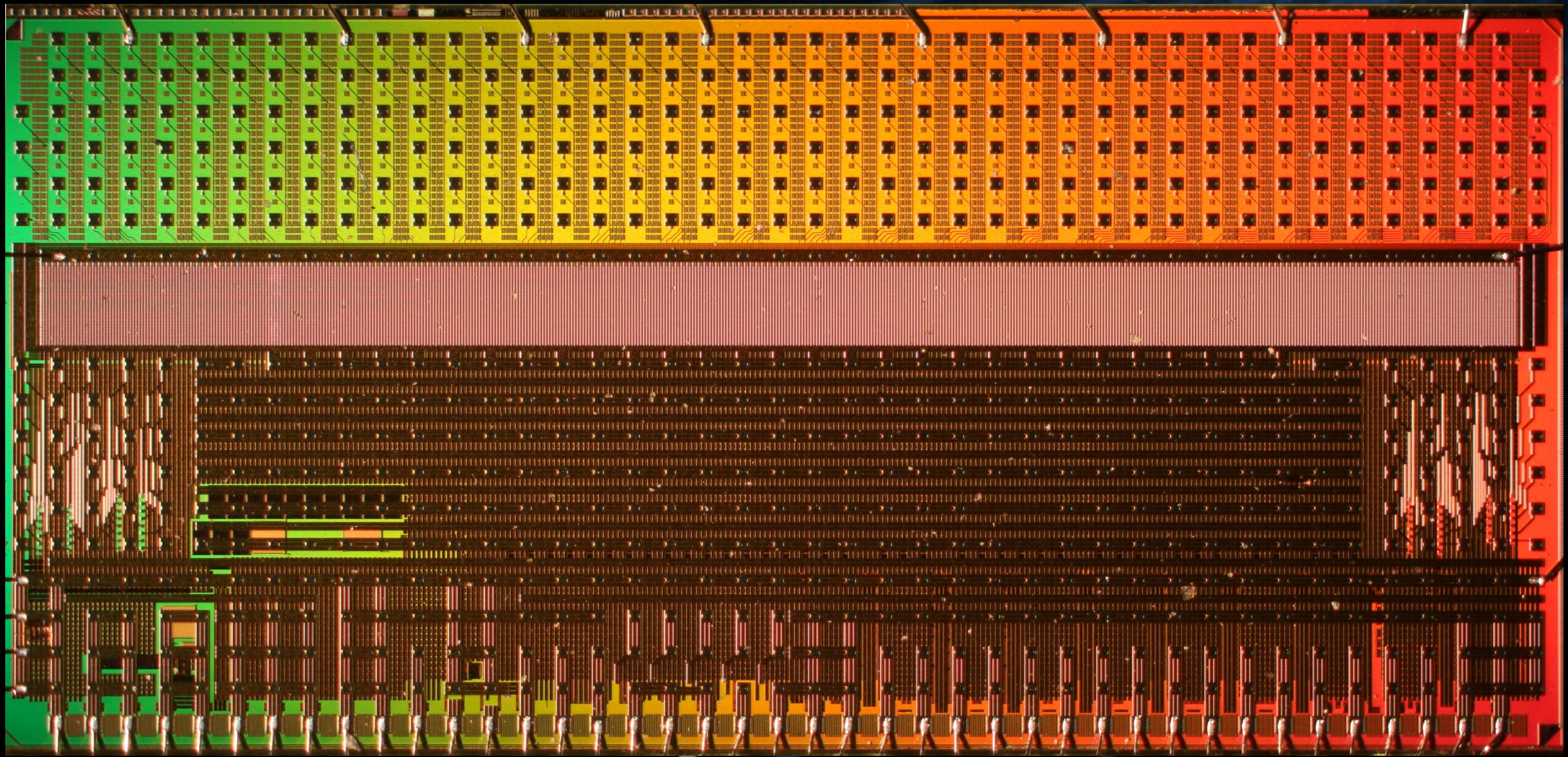
# So what do we do?

- We keep the data safe on the detector

# So what do we do?

- We keep the data safe on the detector

Designing radiation-hard electronics is a specialist skill



CMS CBC3 ASIC
Designed at Imperial College London
Layout at Rutherford Appleton Lab

# So what do we do?

- We keep the data safe on the detector for several microseconds

# So what do we do?

- We keep the data safe on the detector for several microseconds
- We quickly analyse some small fraction of the data and select the interesting crossings to keep

# So what do we do?

- We keep the data safe on the detector for several microseconds
- We quickly analyse some small fraction of the data and select the interesting crossings to keep

**Called Triggering**

# So what do we do?

- We keep the data safe on the detector for several microseconds
- We quickly analyse some small fraction of the data and select the ~~interesting crossings to keep~~

uninteresting crossings to discard

The LHC is a discovery machine, after all

**Called Triggering**

# So what do we do?

- We keep the data safe on the detector for several microseconds
- We quickly analyse some small fraction of the data and select the interesting crossings to keep

uninteresting crossings to discard

**~30Tbps**

The LHC is a discovery machine, after all

**Called Triggering**

# But!

- If you get it wrong
- If your system fails
- If you take too long

} you throw away your valuable physics

# But!

- If you get it wrong
- If your system fails
- If you take too long

} you throw away your valuable physics

**And that is a *really, really* expensive mistake**

# So what do we do?

- We keep the data safe on the detector for several microseconds
- We quickly analyse some small fraction of the data and select the interesting crossings to keep

# So what do we do?

- We keep the data safe on the detector for several microseconds
- We quickly analyse some small fraction of the data and select the interesting crossings to keep

- **But the radiation in the experiment is so intense that the programmable electronics is kept 100m away**
- **The speed of light costs you 1µs to get the data off detector**
- **And 1µs to get the decision back to the detector**
- **Leaves 1.2µs for all data processing**

# So what do we do?

- We keep the data safe on the detector for several microseconds
- We quickly analyse some small fraction of the data and select the interesting crossings to keep

- **But the radiation in the experiment is so intense that the programmable electronics is kept 100m away**
- **The speed of light costs you 1µs to get the data off detector**
- **And 1µs to get the decision back to the detector**
- **Leaves 1.2µs for all data processing**

4800 instructions on a 4GHz CPU

# But recall...

~50 protons at a time

×

40 million times a second

---

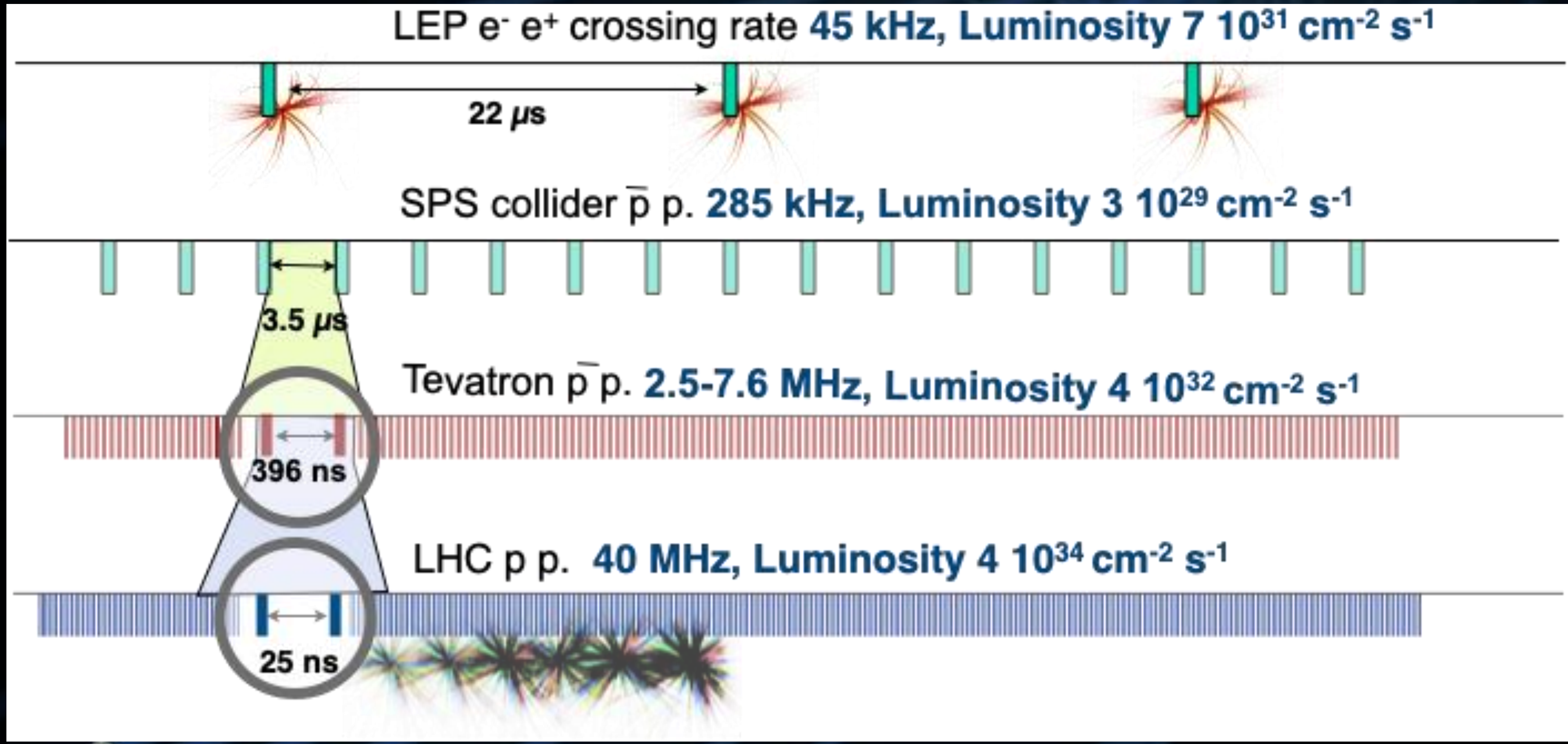
1 Higgs boson every ~5 seconds

But this causes problems

This causes other problems too

# A note on timescales

- At 40MHz BX rate, a 4GHz CPU could perform 100 CPU operations (not enough to be useful) before the next data arrives
- What technology can we use?

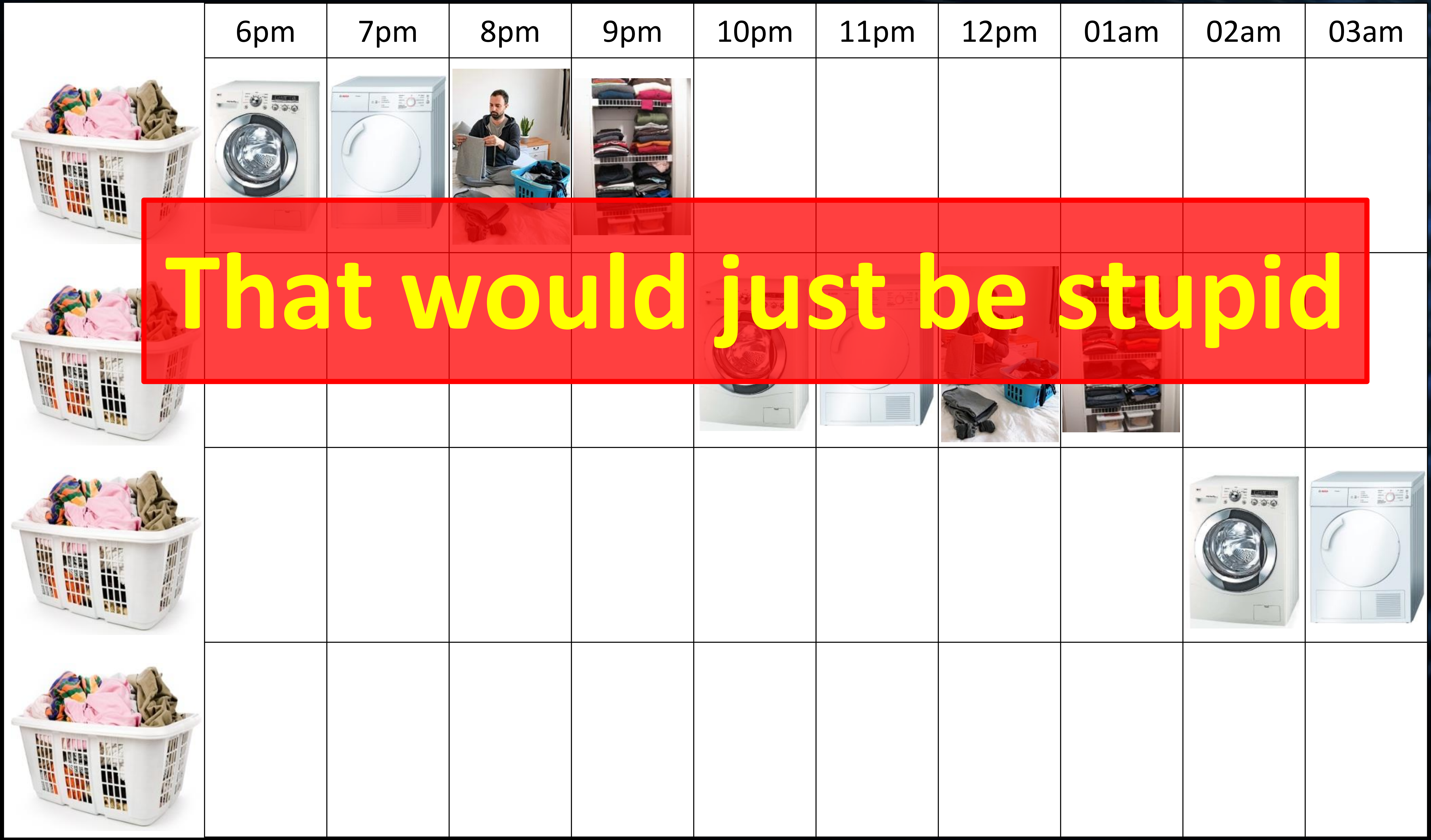


# Sequential processing

| | 6pm | 7pm | 8pm | 9pm | 10pm | 11pm | 12pm | 01am | 02am | 03am |
|---|---|---|---|---|---|---|---|---|---|---|
|  | | | | | | | | | | |
|  | | | | | | | | | | |
|  | | | | | | | | | | |
|  | | | | | | | | | | |

# Sequential processing

# Sequential processing

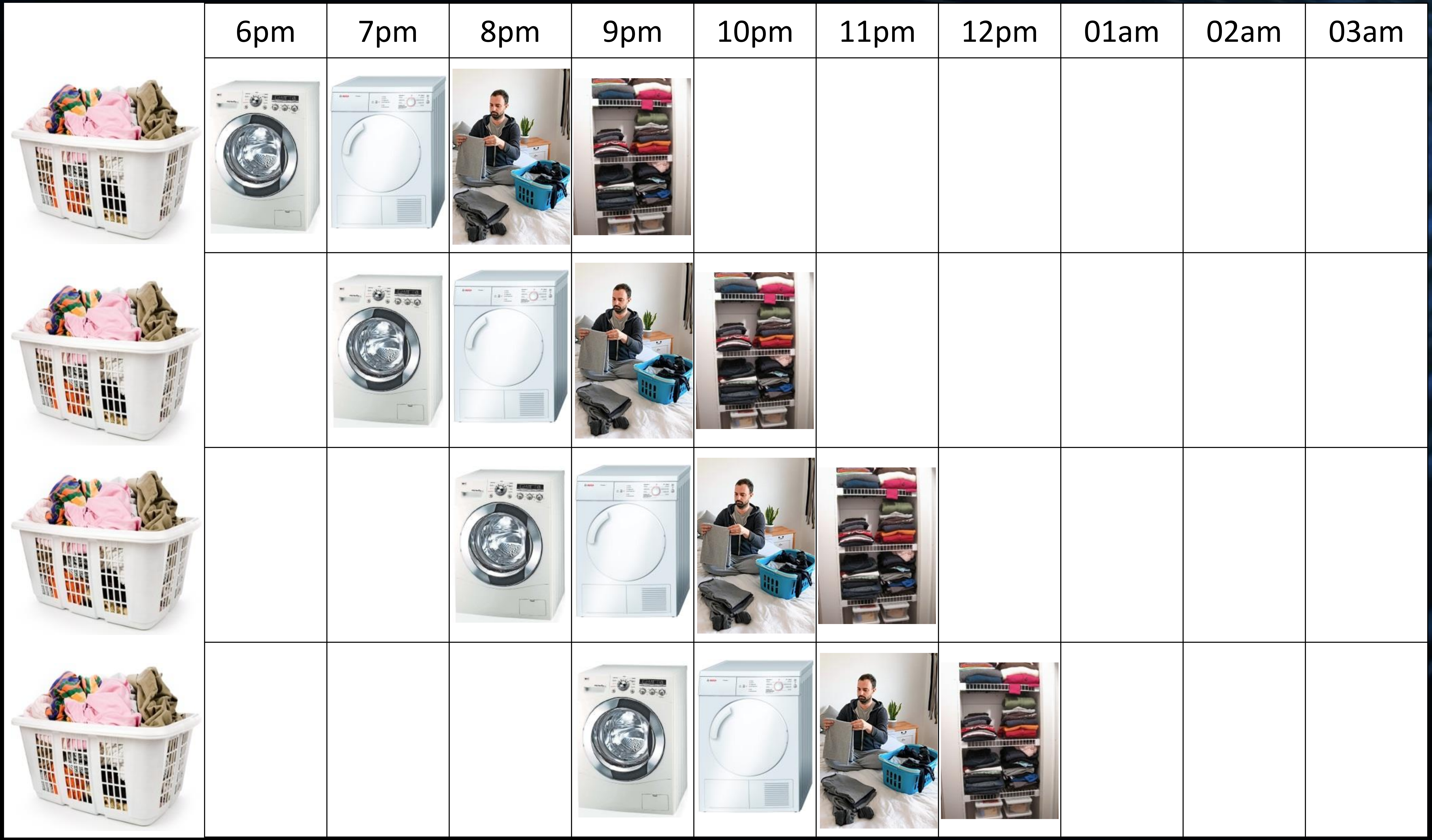


# Sequential processing


**That would just be stupid**

But this is essentially what a PC does

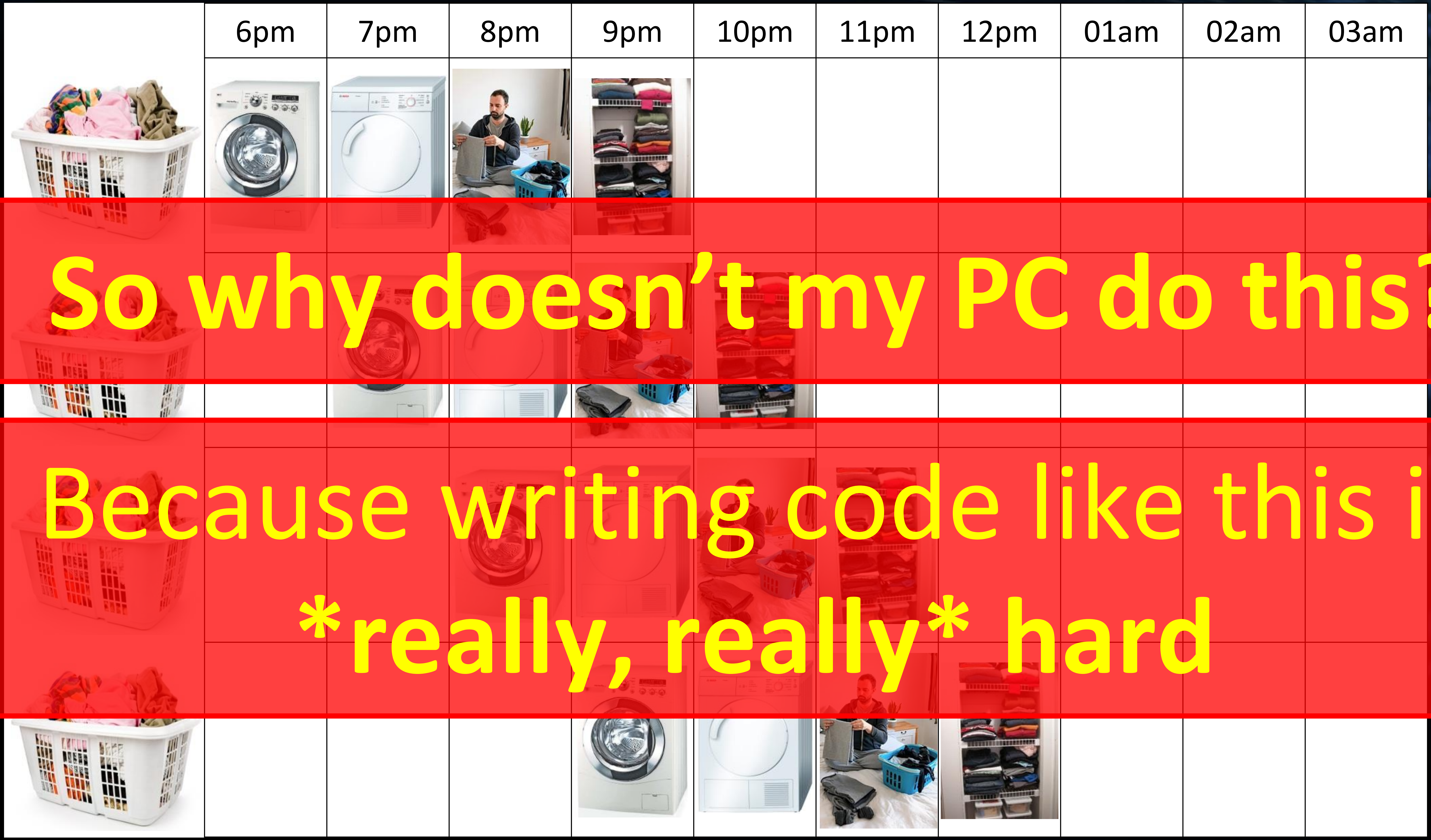
# Pipelined processing



# Pipelined processing


**So why doesn’t my PC do this?**

# Pipelined processing

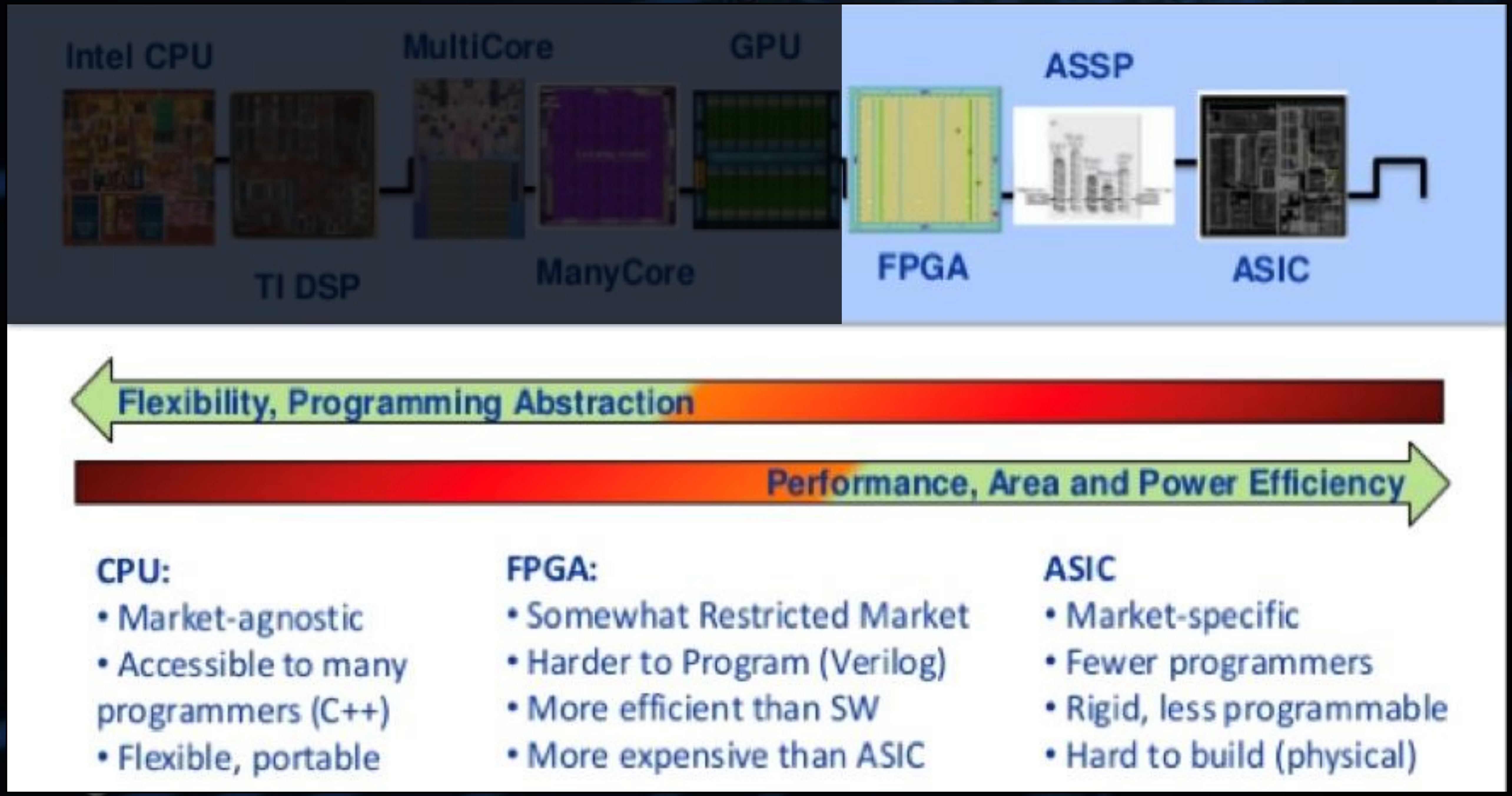



**So why doesn’t my PC do this?**

Because writing code like this is ***really, really* hard**

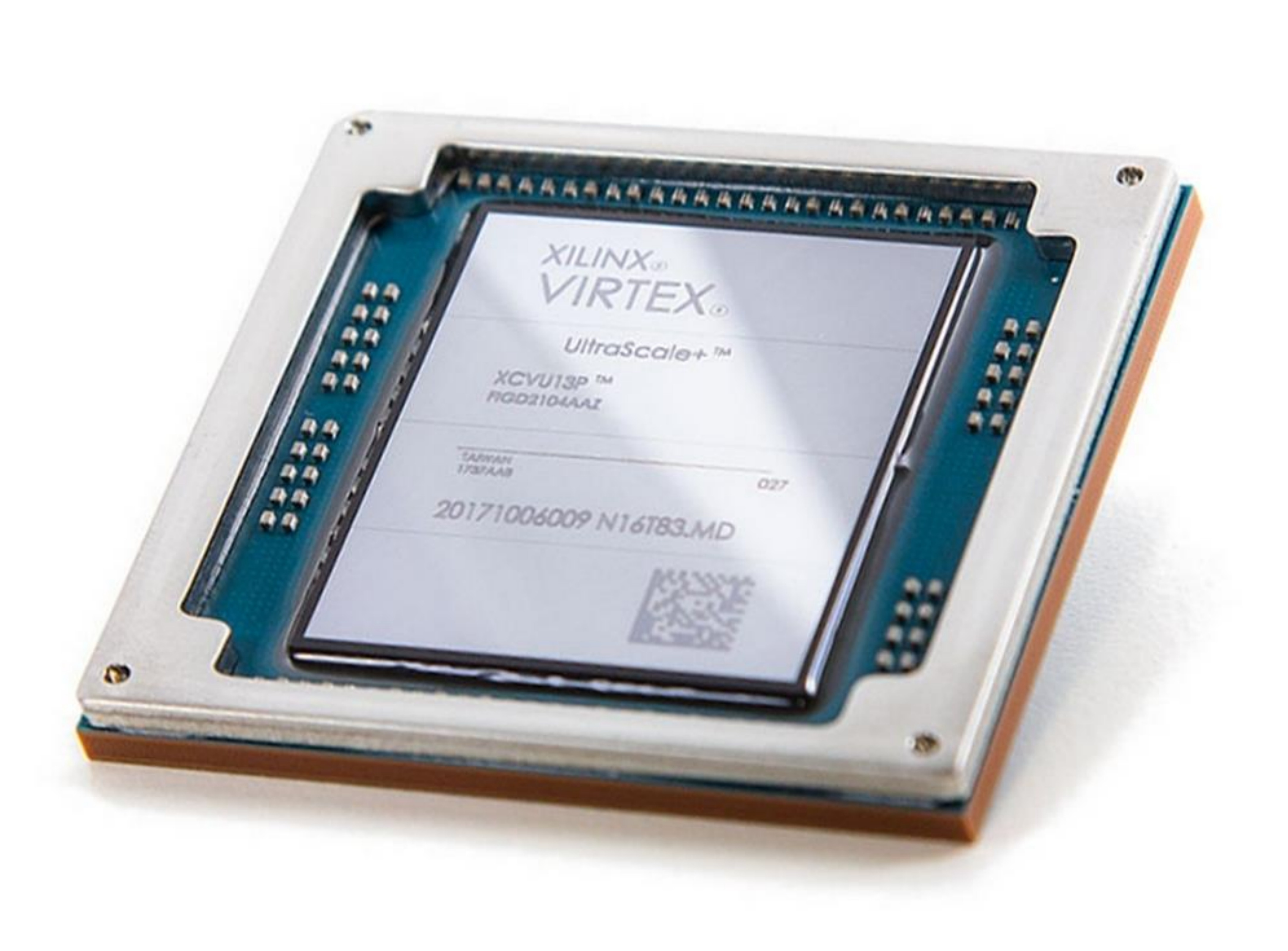
# What technology can we use?

- Application-specific integrated circuits (ASICs): design encoded into silicon
- Field-programmable gate arrays (FPGAs)



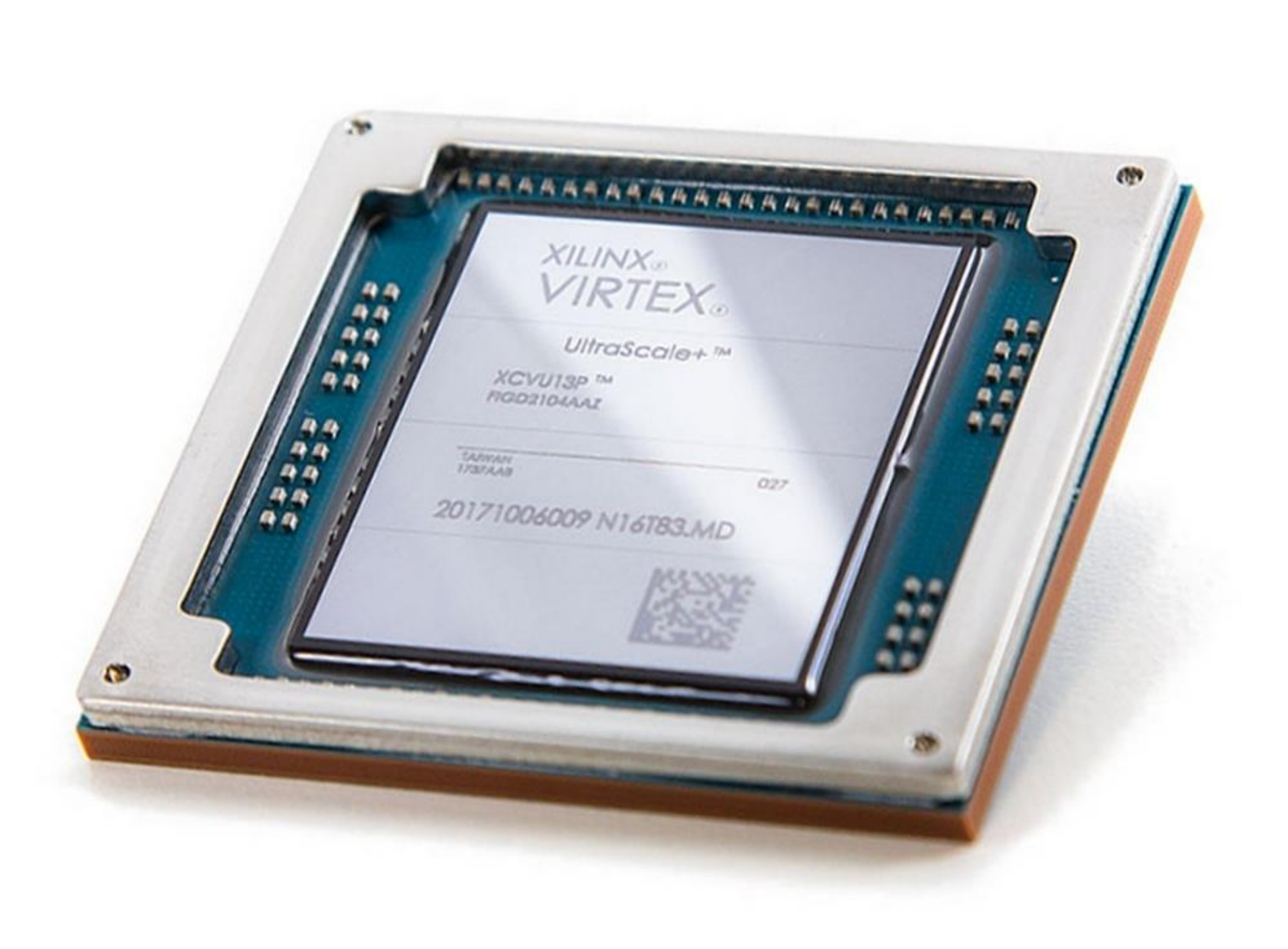
# Field-Programmable Gate Arrays

- Programmable circuit-on-chip



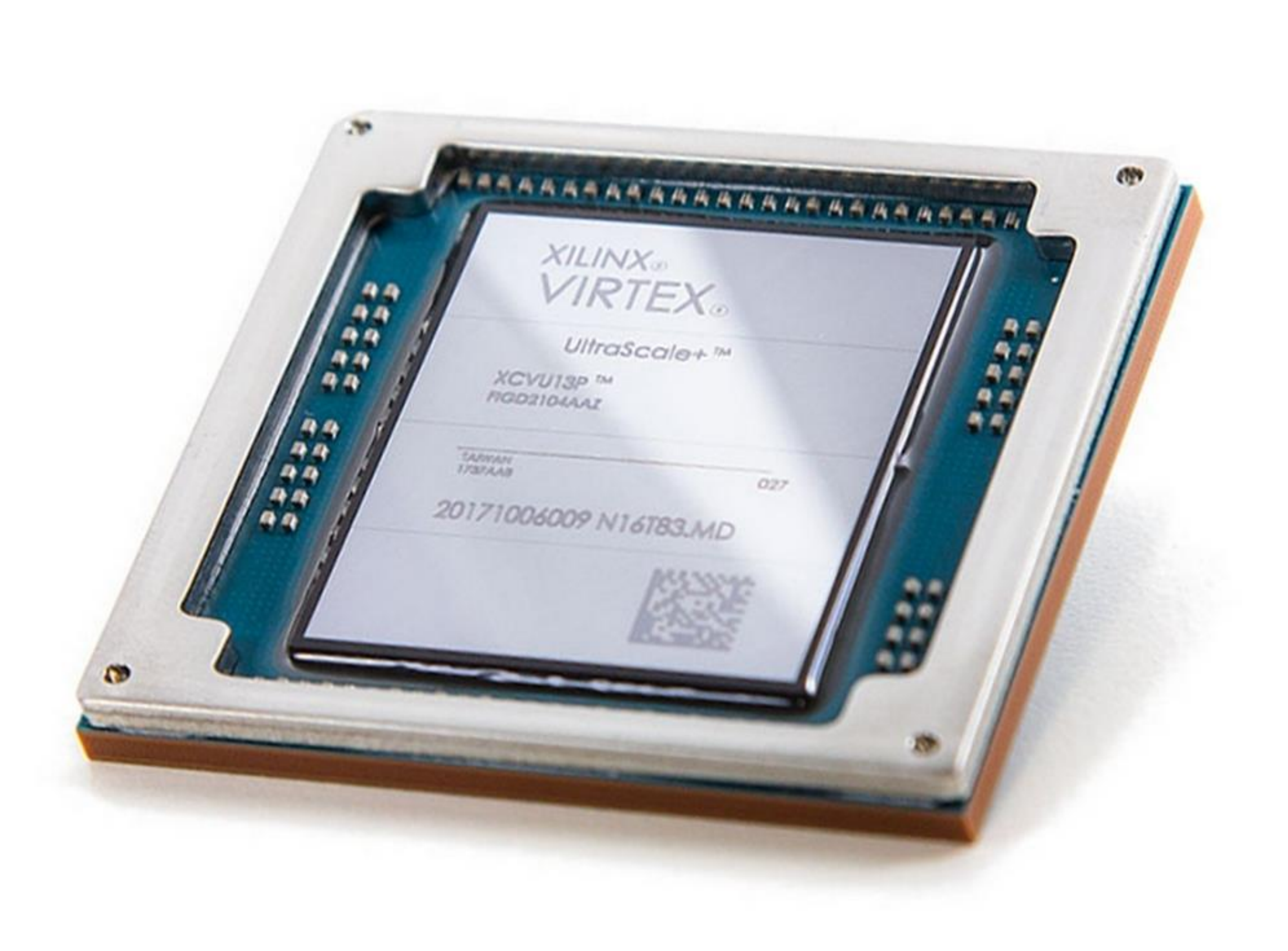
# Field-Programmable Gate Arrays

- Programmable circuit-on-chip
- Upwards of 9 million logic cells & up to 12000 “math” cells
  - All clocked at ~500MHz
- Up to $O(10^{15})$ operations/second
- Fully parallel and fully pipelineable



# Field-Programmable Gate Arrays

- Programmable circuit-on-chip
- Upwards of 9 million logic cells & up to 12000 “math” cells
  - All clocked at ~500MHz
- Up to $O(10^{15})$ operations/second
- Fully parallel and fully pipelineable

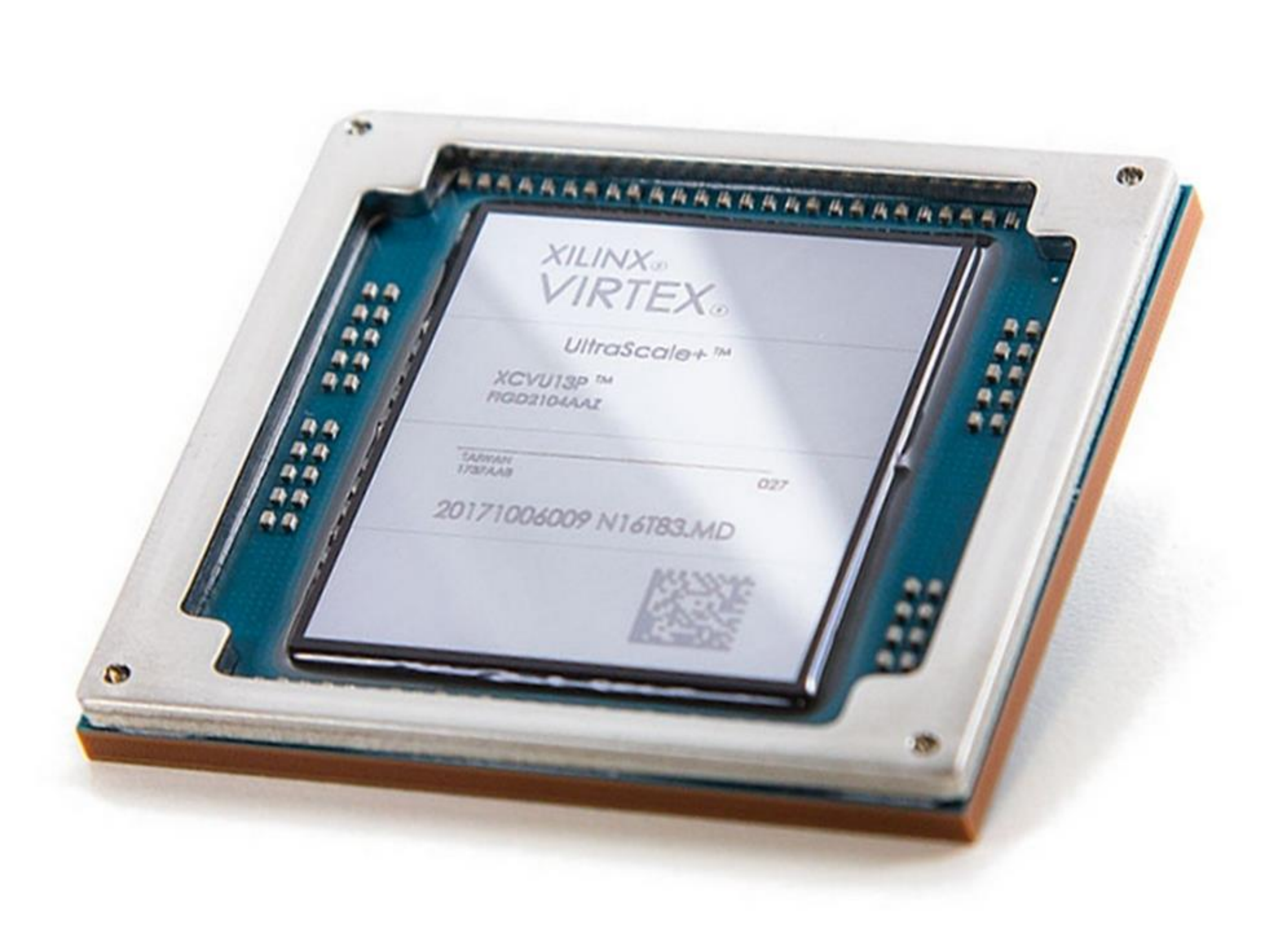



Rather useful when you are dealing with 30Tbps

# Field-Programmable Gate Arrays

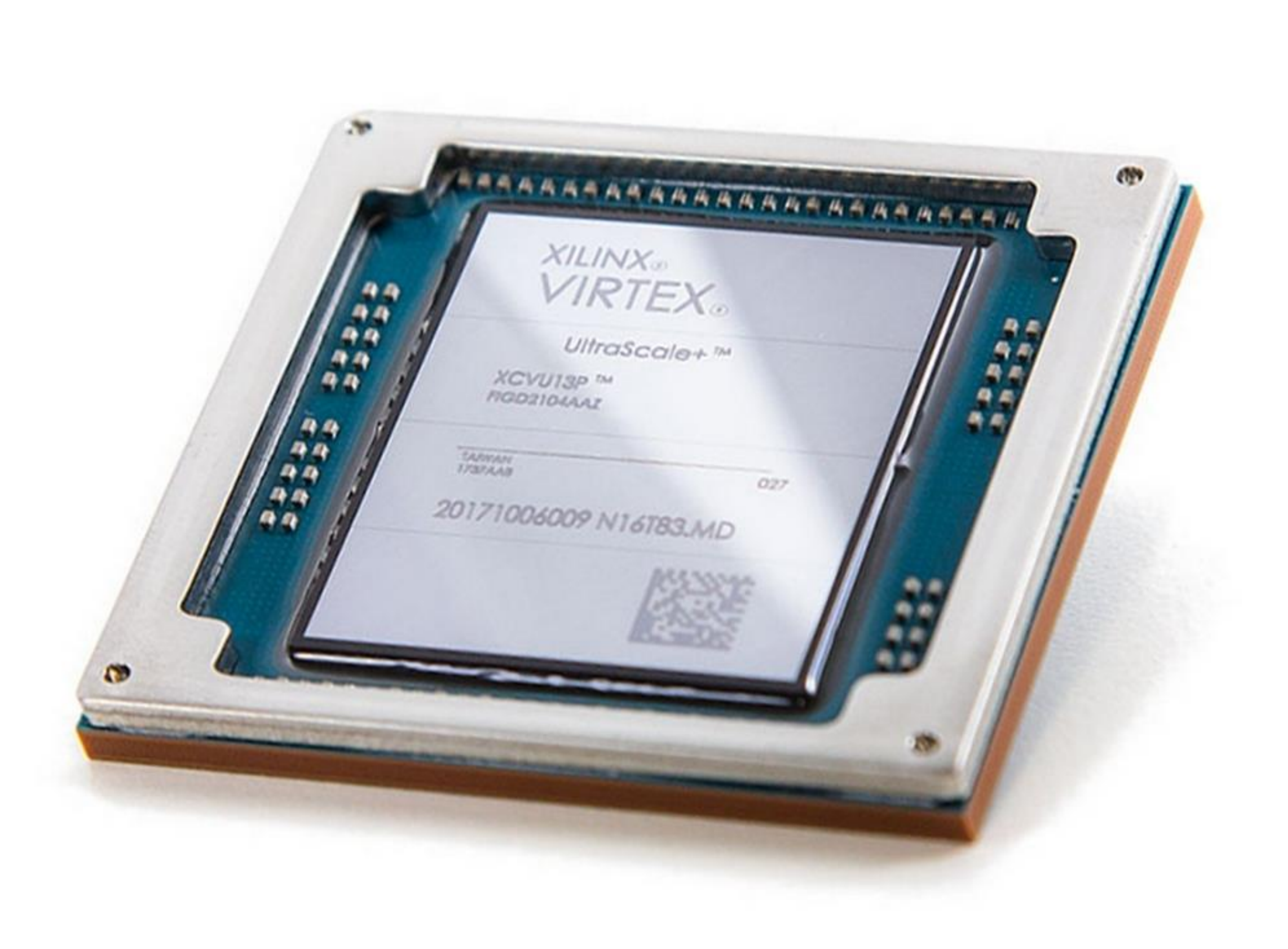
- Programmable circuit-on-chip
- Upwards of 9 million logic cells & up to 12000 “math” cells
    - All clocked at ~500MHz
- Up to $O(10^{15})$ operations/second
- Fully parallel and fully pipelineable

Rather useful when you are getting new data every 25ns



# Field-Programmable Gate Arrays

- Programmable circuit-on-chip
- Upwards of 9 million logic cells & up to 12000 “math” cells
    - All clocked at ~500MHz
- Up to $O(10^{15})$ operations/second
- Fully parallel and fully pipelineable

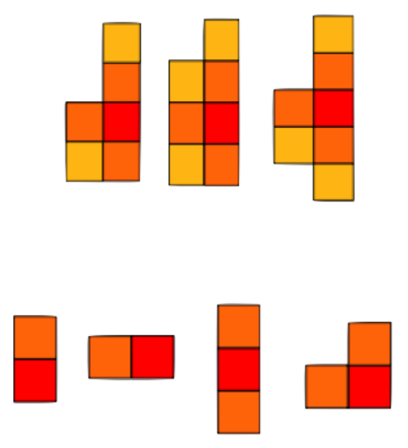



Rather useful when you are dealing with 30[illegible]bps

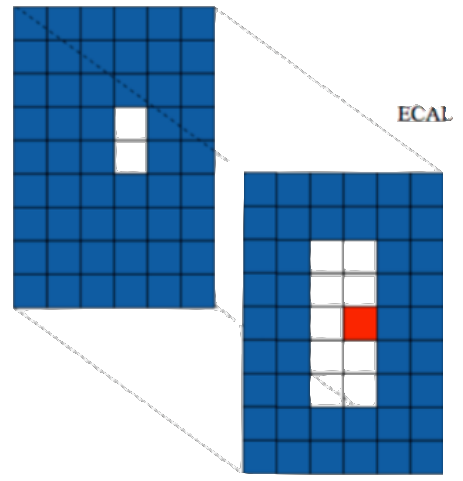
**and *really, really* hard to program efficiently**

Rather useful when you are getting new data every 25ns

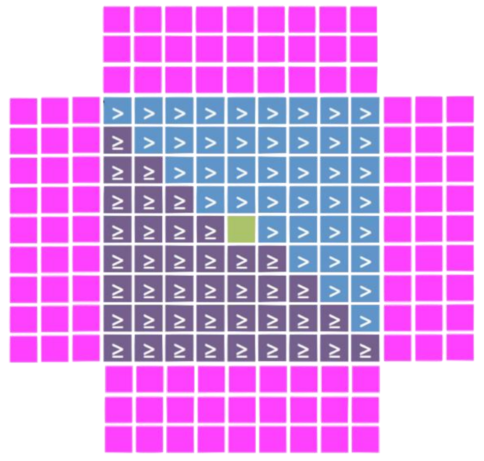
# The CMS Calorimeter trigger



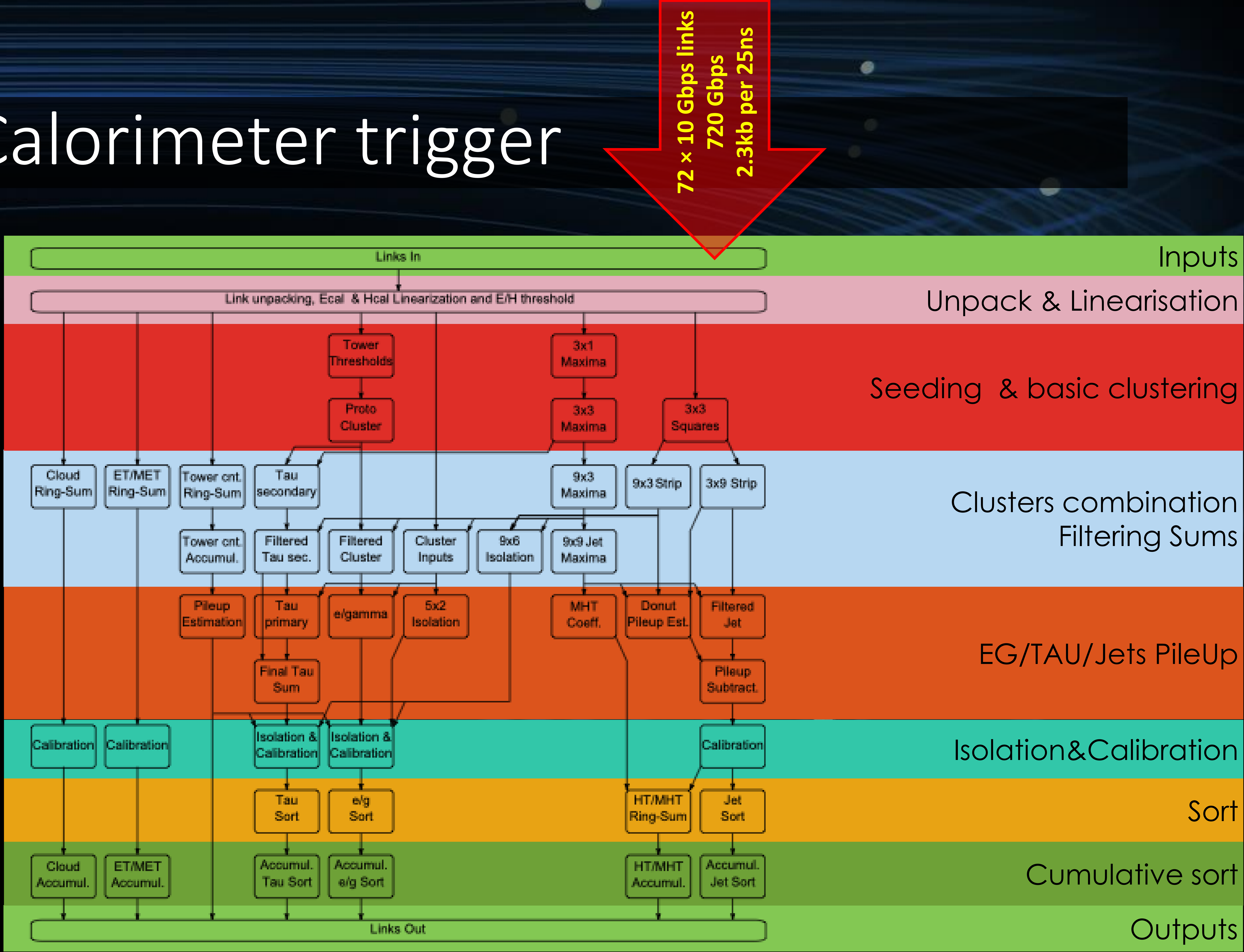
Dynamic clustering



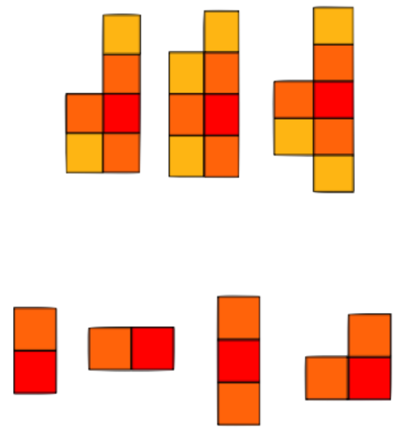

Jet building with pileup subtraction



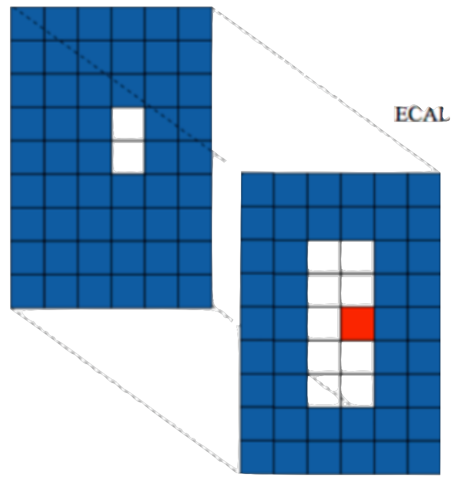
Shape veto, H/E, isolation, calibration



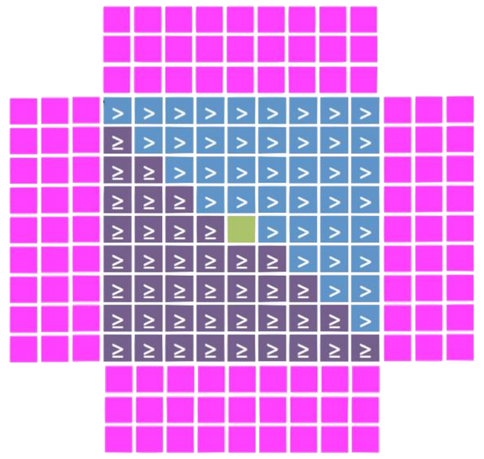
# The CMS Calorimeter trigger

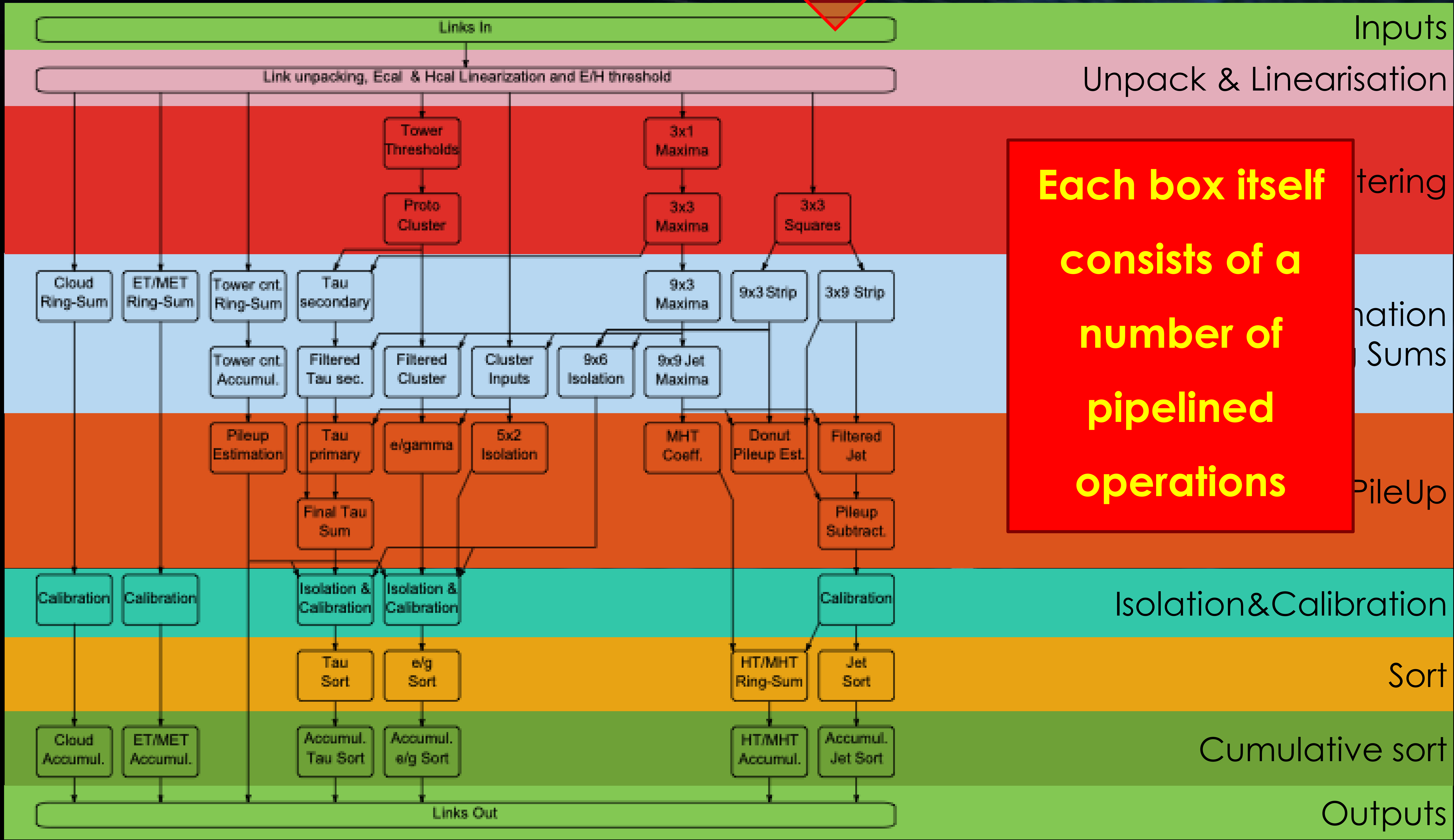
Dynamic clustering




Jet building with pileup subtraction



Shape veto, H/E, isolation, calibration



# What type of algorithms are we talking about?

- Classical algorithms
  - Clustering in 2D or 3D
  - Pattern identification/matching
  - Kalman filtering

**Remember that they have to be implemented in a**

**fully parallel**

**and**

**fully pipelined**

**form**

# What type of algorithms are we talking about?

- Classical algorithms
  - Clustering in 2D or 3D
  - Pattern identification/matching
  - Kalman filtering
- In future
  - Particle-flow – full reconstruction of particles and events
  - Machine-learning, BDTs and neural-nets in chip

**Remember that they have to be implemented in a**

**fully parallel**

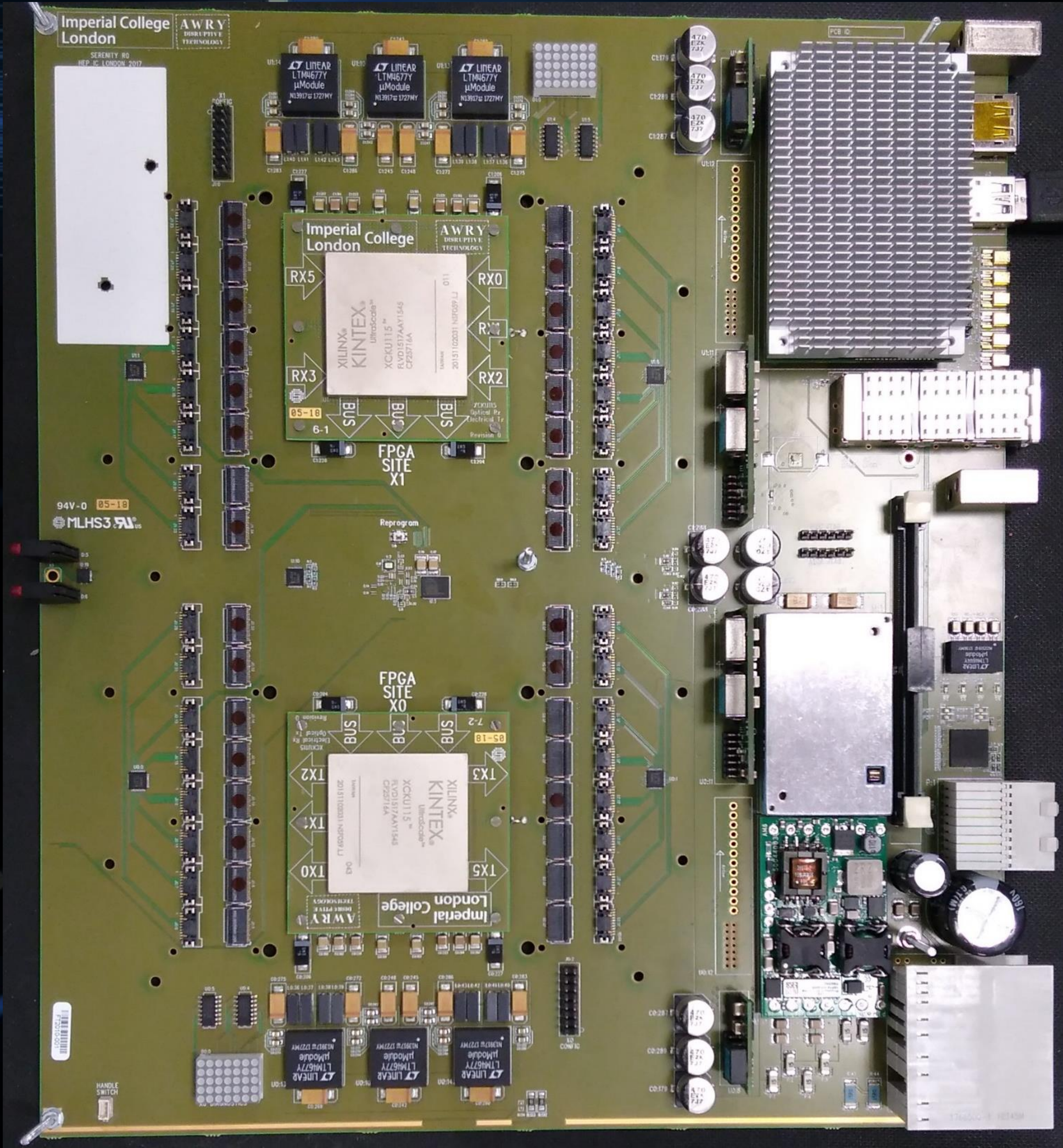
**and**

**fully pipelined**

**form**

# Field-Programmable Gate Arrays

- A chip needs to be attached to something!
- Imperial's latest platform: Serenity



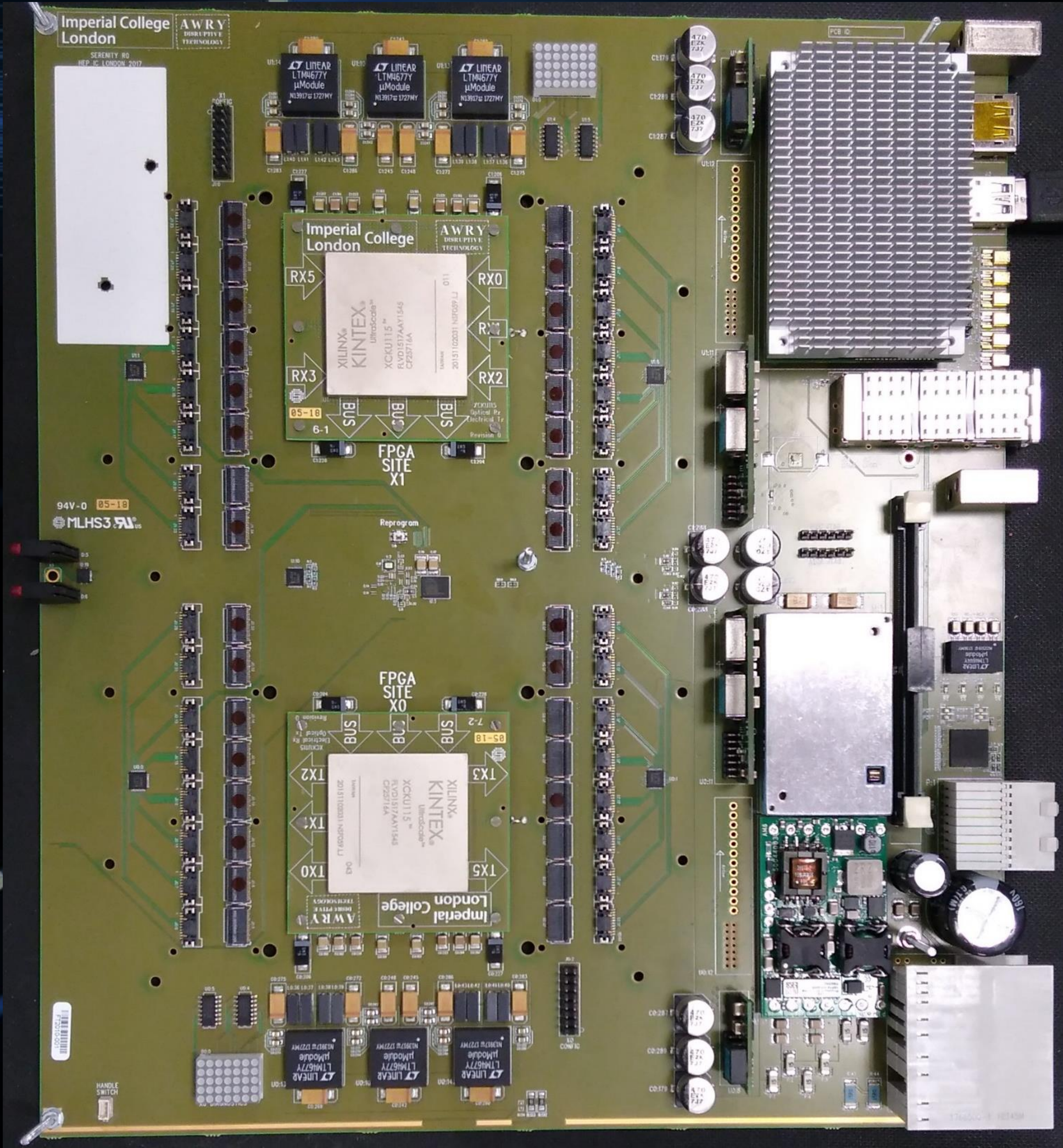
# Field-Programmable Gate Arrays

- A chip needs to be attached to something!
- Imperial's latest platform: Serenity
  - 208 optical transmit links &
    208 optical receive links @ 28.5Gbps =

**5.9 + 5.9 Tbps**

  - Skewable to

**7.0 + 4.8 Tbps**

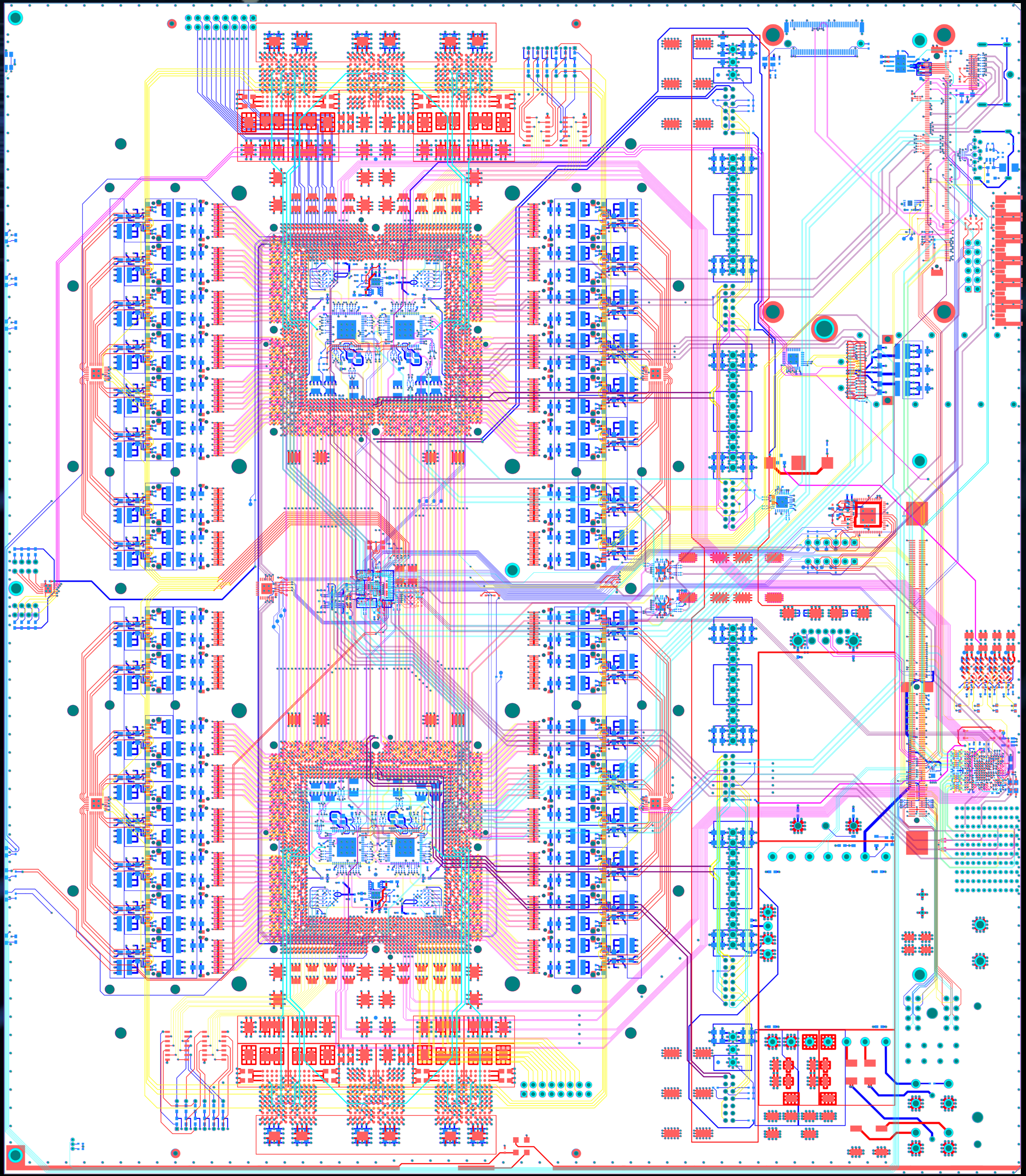


# Field-Programmable Gate Arrays

- A chip needs to be attached to something!
- Imperial's latest platform: Serenity

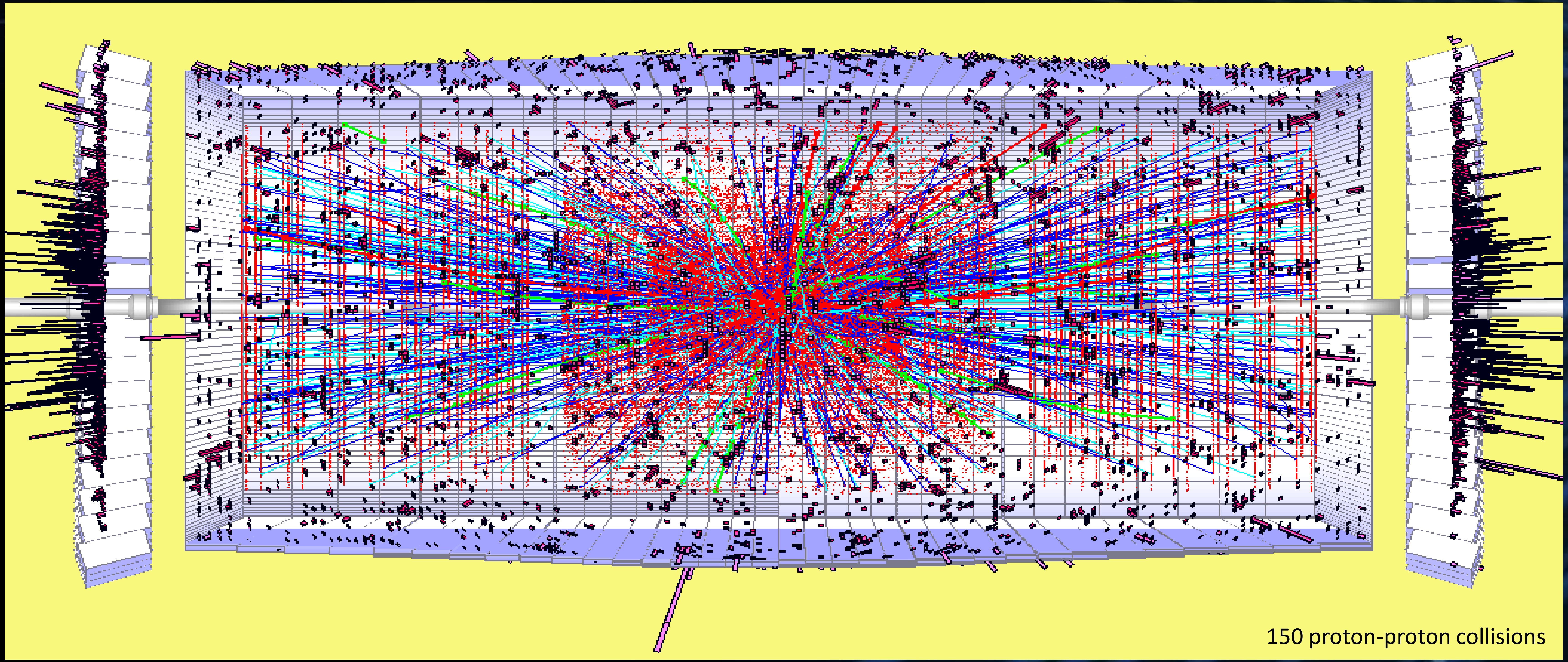
**A real challenge:**

- **30Gbps links requires ~analogue design**
- **Each chip drawing over 100A of core power**
- **Heat management a huge challenge**



- 6 ground planes – need to control noise
- 4 power planes – need to supply 9 voltages per chip at up to 100A
- 6 signal layers – these are big chips

# The future: High-luminosity LHC



150 proton-proton collisions

- Up to 250 proton-proton collisions per event
- Really, really hard to find the interesting event under all that rubbish
- Need to sift through 300Tbps to find the interesting events

# Conclusion

- To find the interesting events at the LHC we must search through vast amounts of data

# Conclusion

- To find the interesting events at the LHC we must search through vast amounts of data
- To do this requires work at the cutting edge of electronics, programming and physics

# Conclusion

- To find the interesting events at the LHC we must search through vast amounts of data
- To do this requires work at the cutting edge of electronics, programming and physics
    - And in the future, also Machine Learning

# Conclusion

- To find the interesting events at the LHC we must search through vast amounts of data
- To do this requires work at the cutting edge of electronics, programming and physics
  - And in the future, also Machine Learning
- And this will be even more challenging at the HL-LHC
  - Lots of fun to be had

# Thanks for listening

Any questions?

# Spares

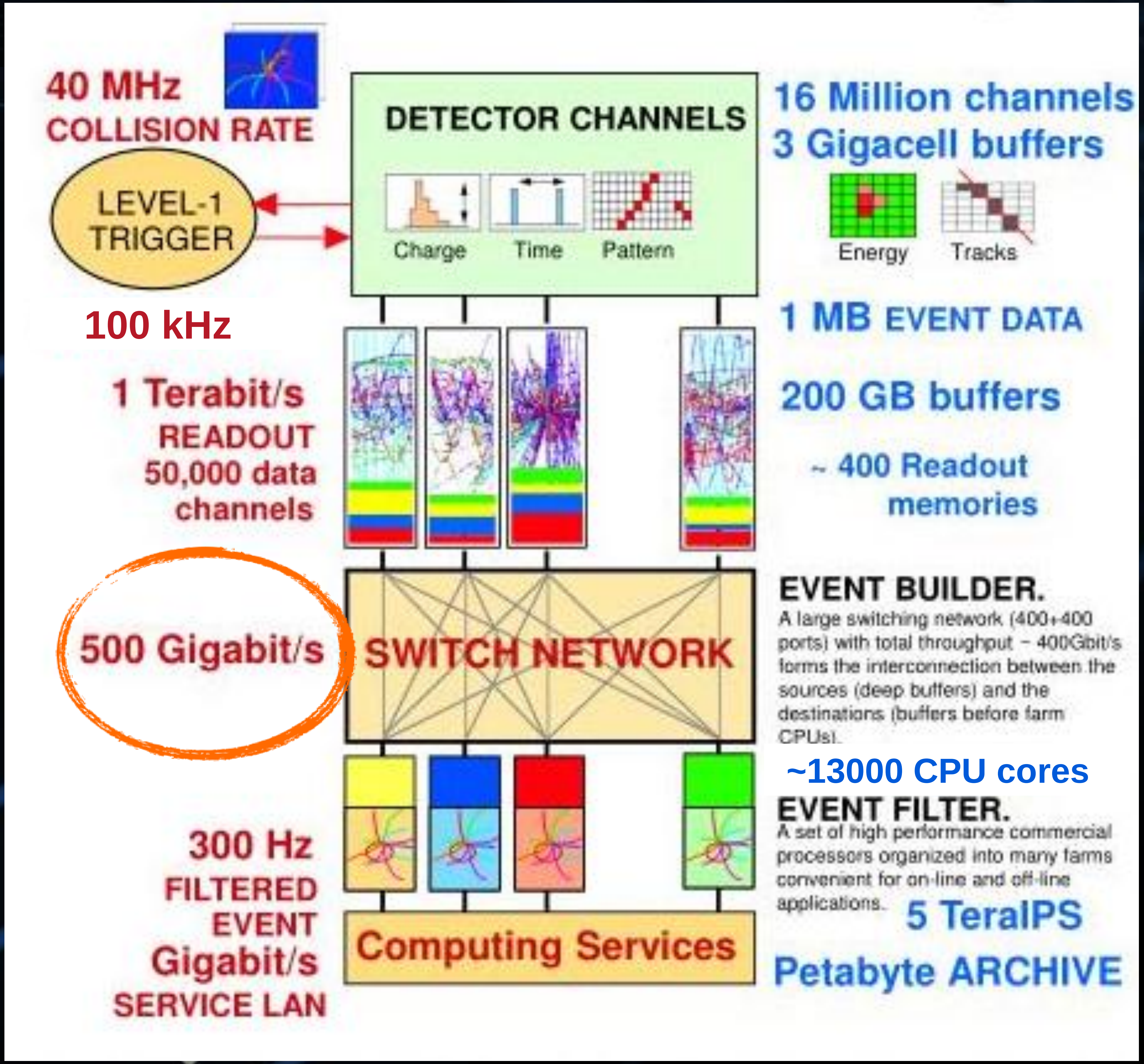
# How much data?

# Layered triggers

- In our FPGAs we accept events at 100kHz (out of the 40MHz)
  - Reduced the total data volume by a factor of 400
- Small enough to get through an ethernet network into PCs

# Layered triggers

- In our FPGAs we accept events at 100kHz (out of the 40MHz)
  - Reduced the total data volume by a factor of 400
- Small enough to get through an ethernet network into PCs
  - Although, of course, "small" here is relative



# Spares: Introduction to detectors


ONE DOES NOT SIMPLY
SEE A HIGGS BOSON

# Heavy and unstable

- If a heavy and unstable state is produced by a proton-proton collision, it decays quickly into more stable particles

# Heavy and unstable

- If a heavy and unstable state is produced by a proton-proton collision, it decays quickly into more stable particles
- And I mean ***REALLY*** quickly:

| | |
|---|---|
| Higgs Boson | $1.6 \times 10^{-22}$ seconds |
| W/Z Boson | $3 \times 10^{-25}$ seconds |
| Top Quark | $5 \times 10^{-25}$ seconds |
| Tau Lepton | $2.9 \times 10^{-13}$ seconds |

- Doesn't get anywhere near a detector

# So we can't see the Higgs (or most other particles) directly

- But you can't "see" the other particles either (in a conventional sense)
- So how are particle detectors built?

# A parallel question

- How can you tell the properties of a car and how fast it is going from the outside?

# Minimal disruption

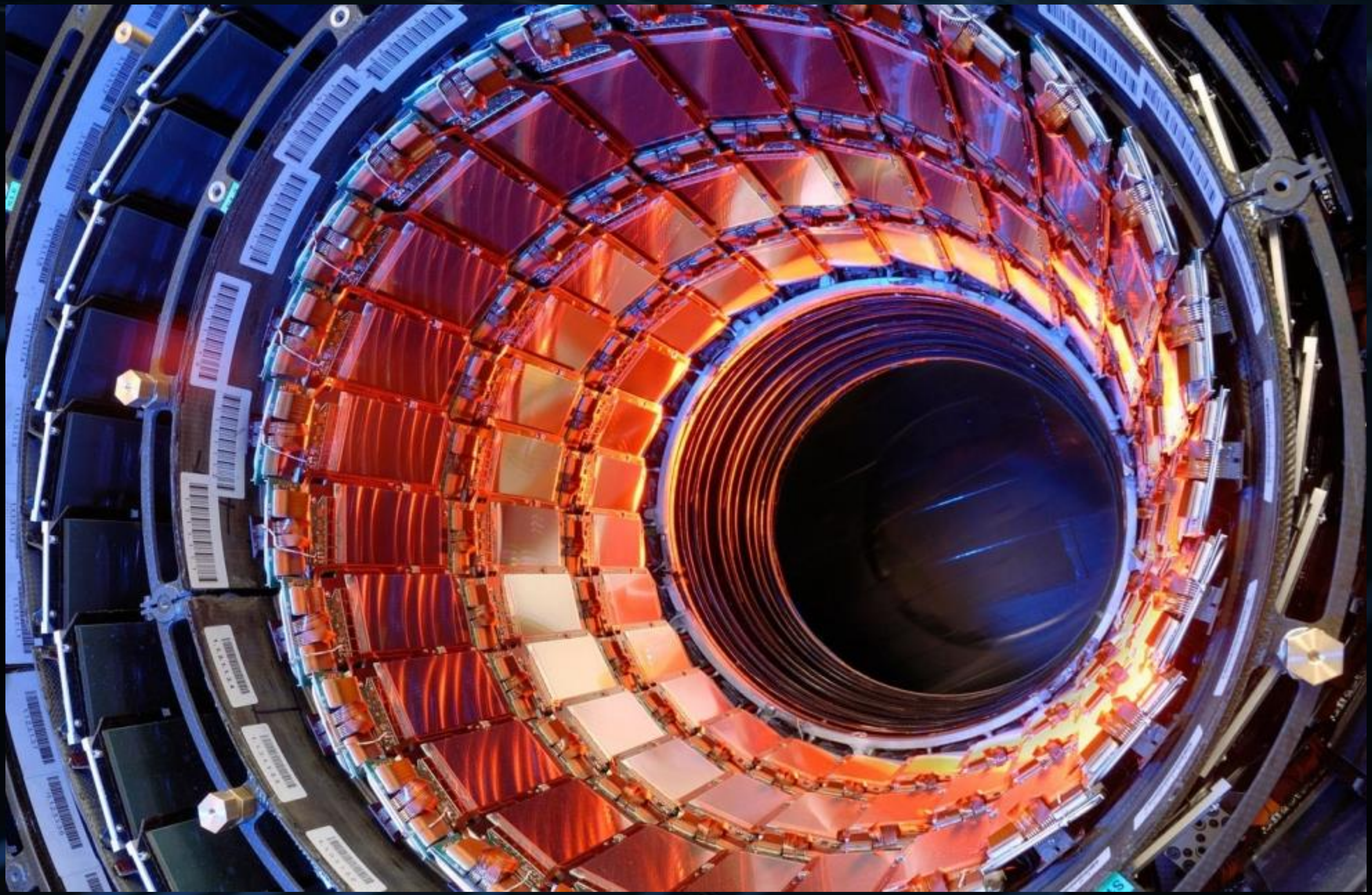
- Take a couple of snapshots
- Work out how it got between them and how long it took to do so

# Minimal disruption

- Take a couple of snapshots
- Work out how it got between them and how long it took to do so
- Does not affect the speed/momentum/energy of the car

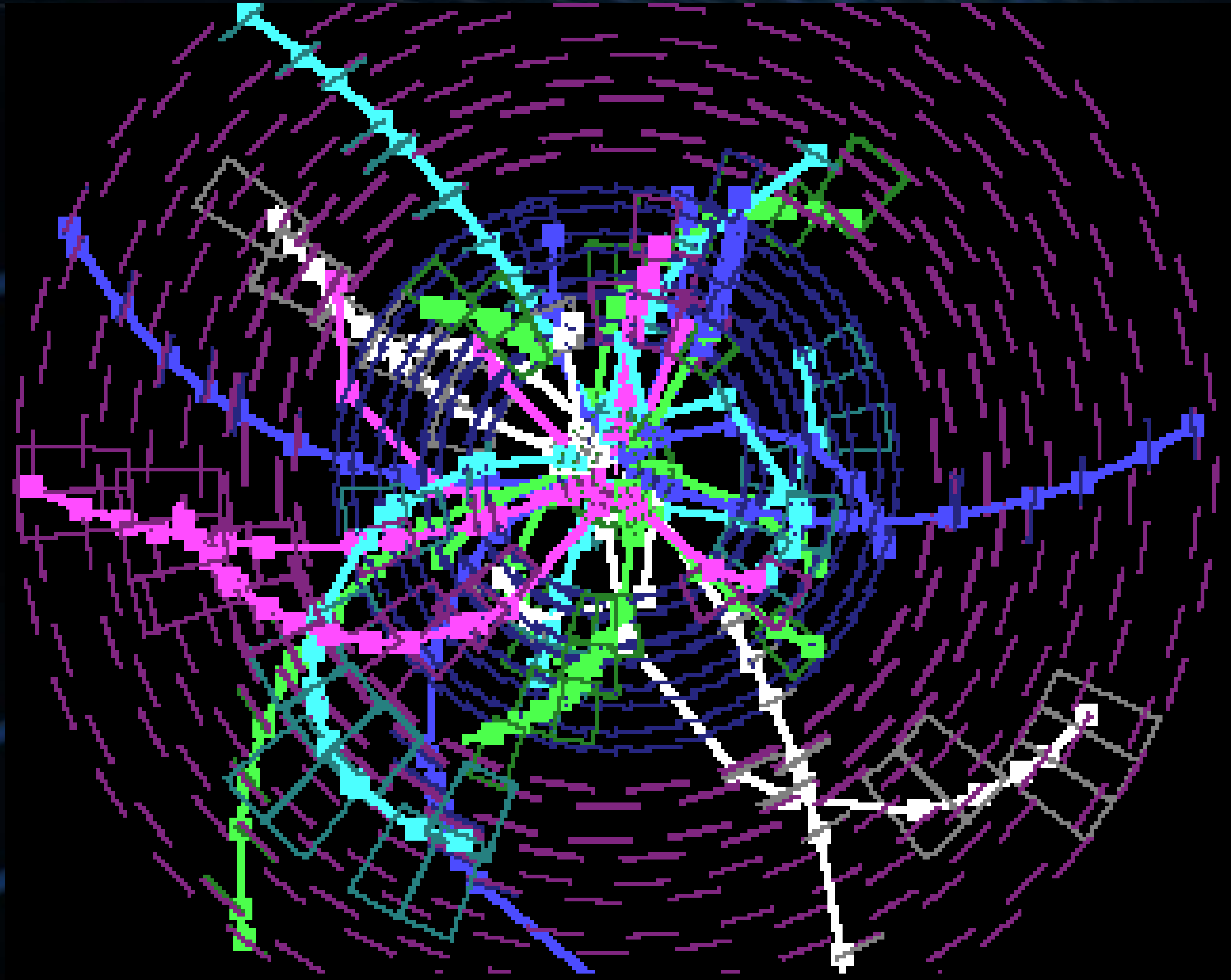
# Minimal disruption

- Lightweight tracker that records the position of charged particles as accurately as possible, while affecting the particle as little as possible



# Minimal disruption

- Lightweight tracker that records the position of charged particles as accurately as possible, while affecting the particle as little as possible
- Join-the-dots
- Apply a magnetic field to determine the charge and momentum

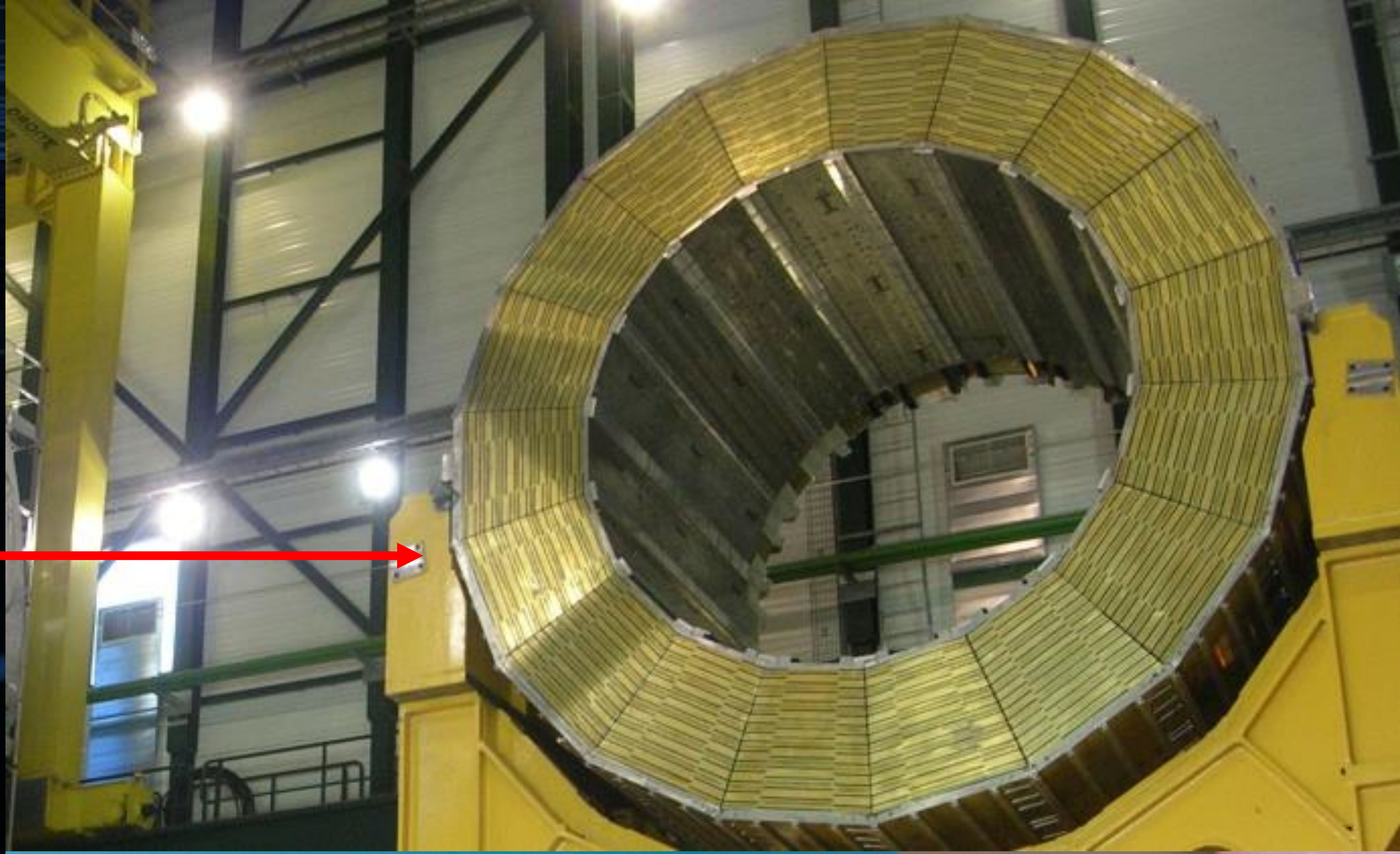


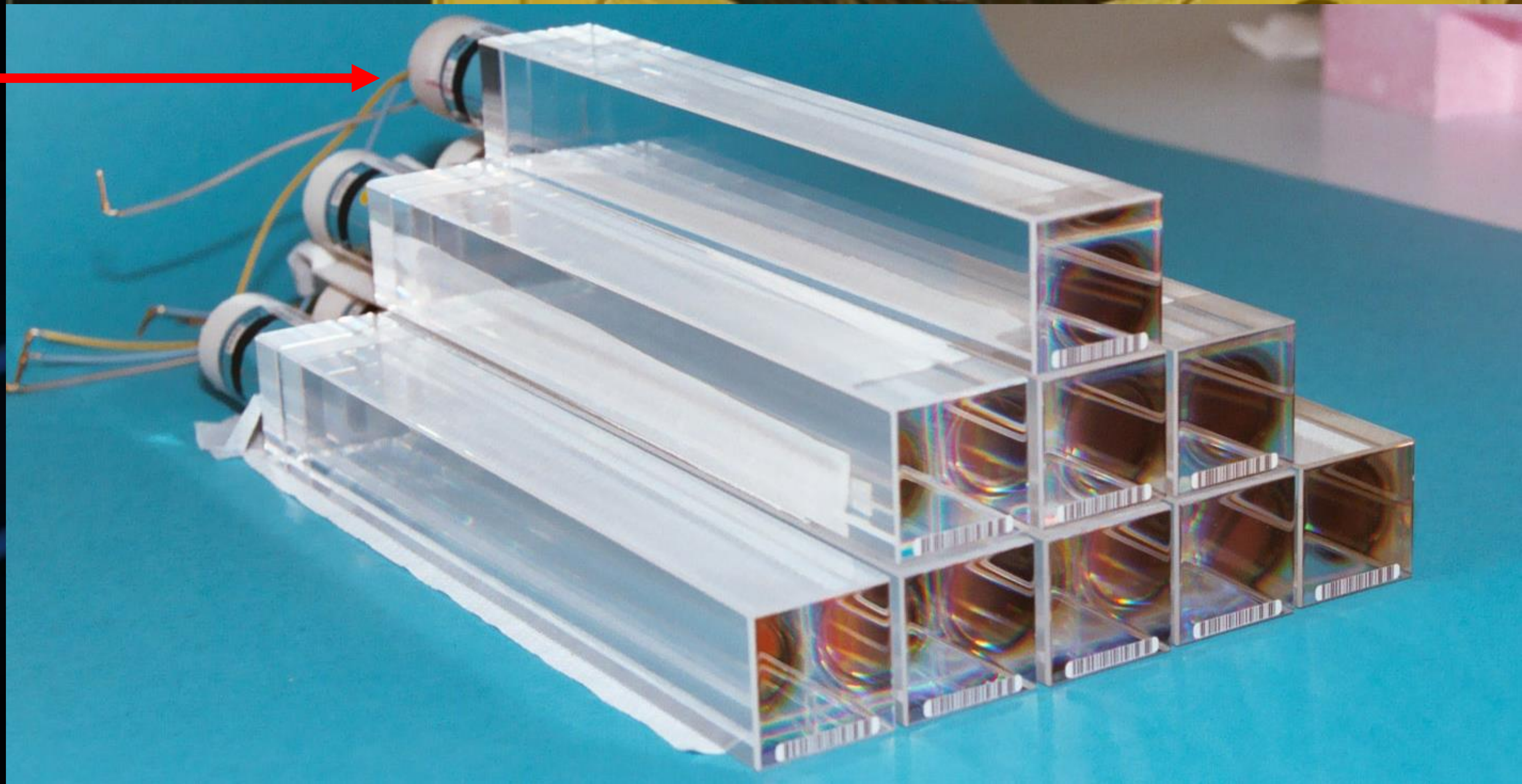
# Maximal disruption

- Place something very heavy in the way
- Collect all the bits; measure how far the pieces get thrown
- Certainly does affect the speed/momentum/ energy of the car

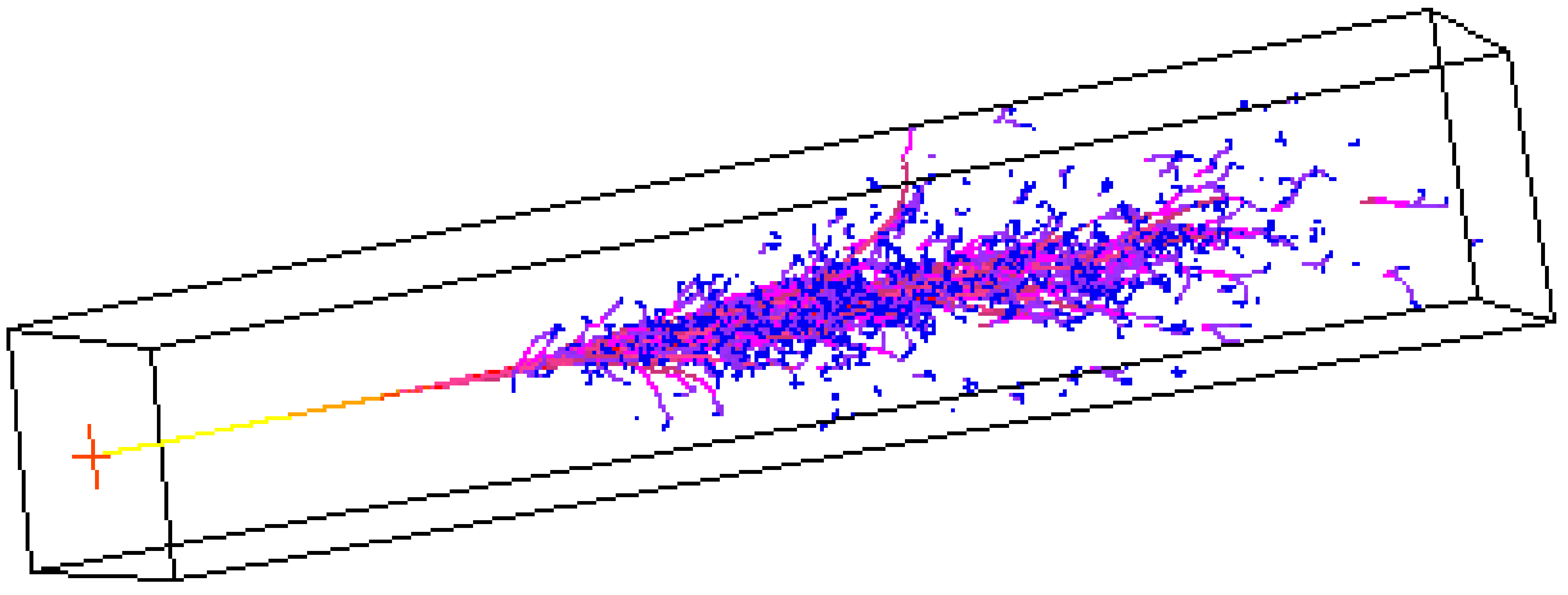
# Maximal disruption

- Put something very dense in the way
  - Brass
  - Steel
  - Tungsten
  - Depleted uranium
  - Lead tungstate crystals
- Catch the light in some transparent medium using photomultipliers

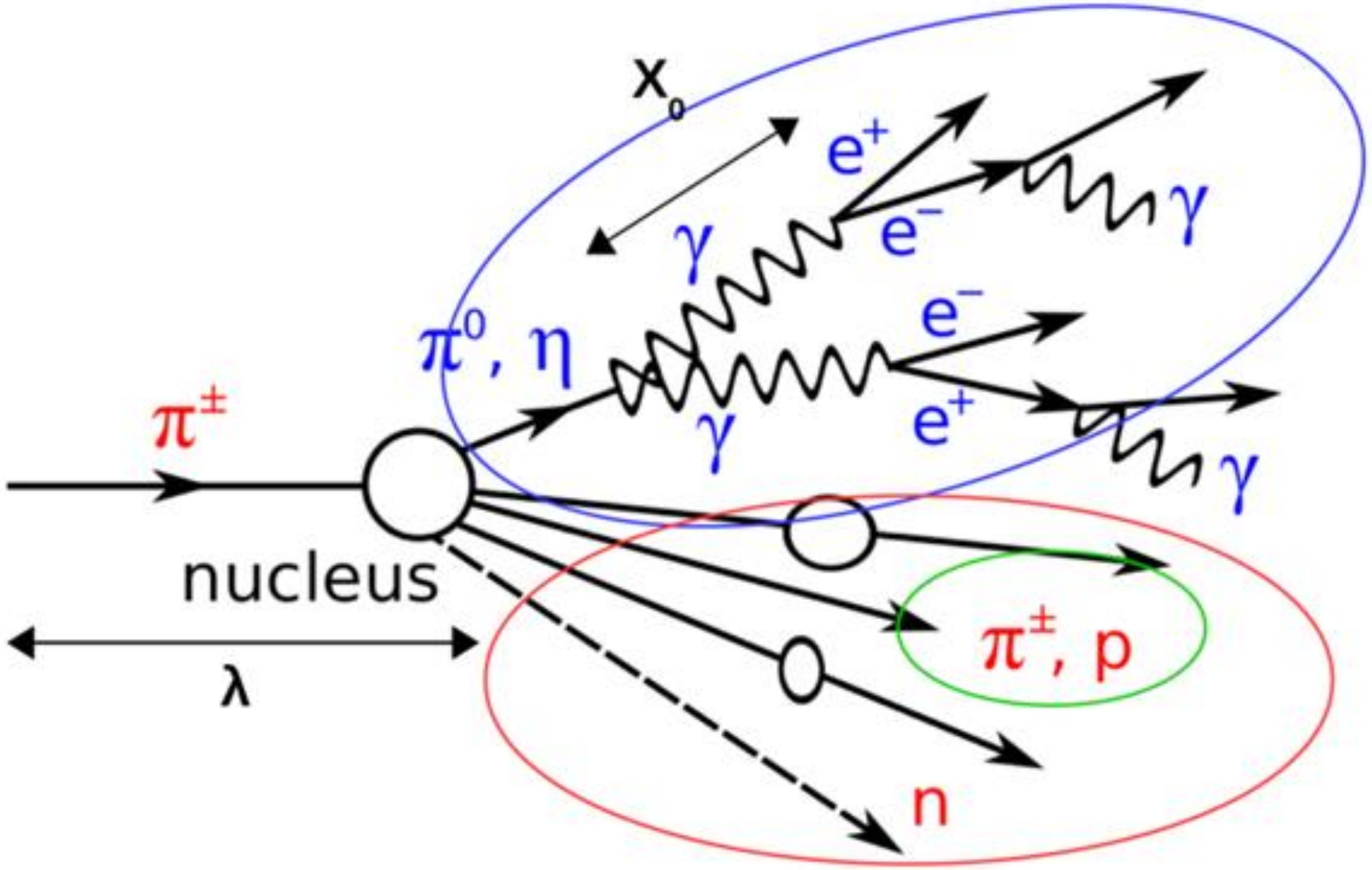




# Maximal disruption



- Energetic leptons particles emit photons
  - Energetic photons pair-produce electrons and positrons
    - Which emit photons
      - Which pair-produce electrons and positrons
        - Which emit photons
          - Which pair-produce electrons and positrons
            - Which emit photons…
- Energetic hadrons break up into lighter hadrons
  - Which break up into lighter hadrons
    - Which break up into lighter hadrons
      - Which break up into lighter hadrons…
  - Pions decay to photons
    - Which pair-produce electrons and positrons…
      - Which emit photons…



# The CMS detector

# The CMS detector

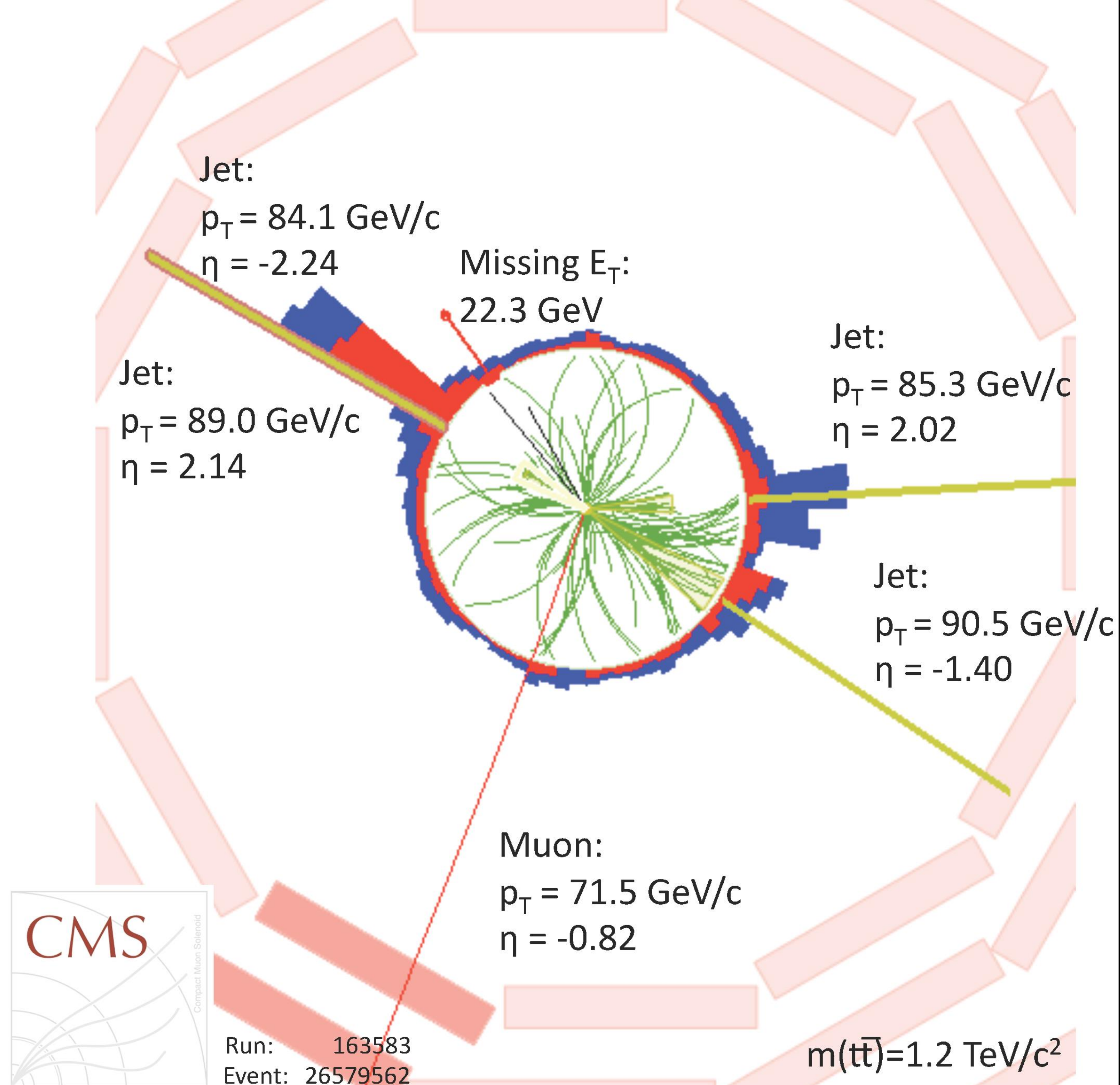
# The CMS detector

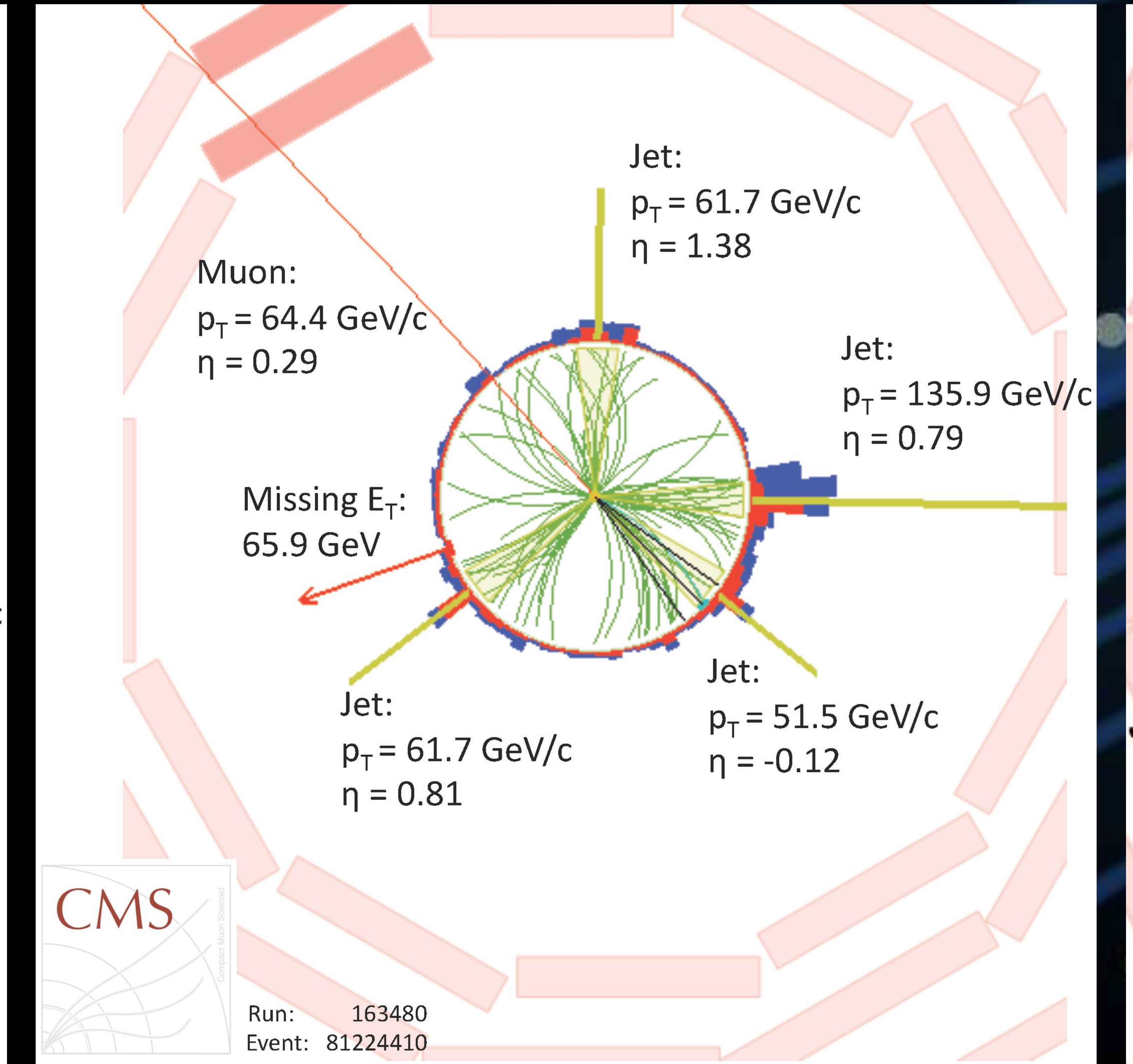
# The CMS detector

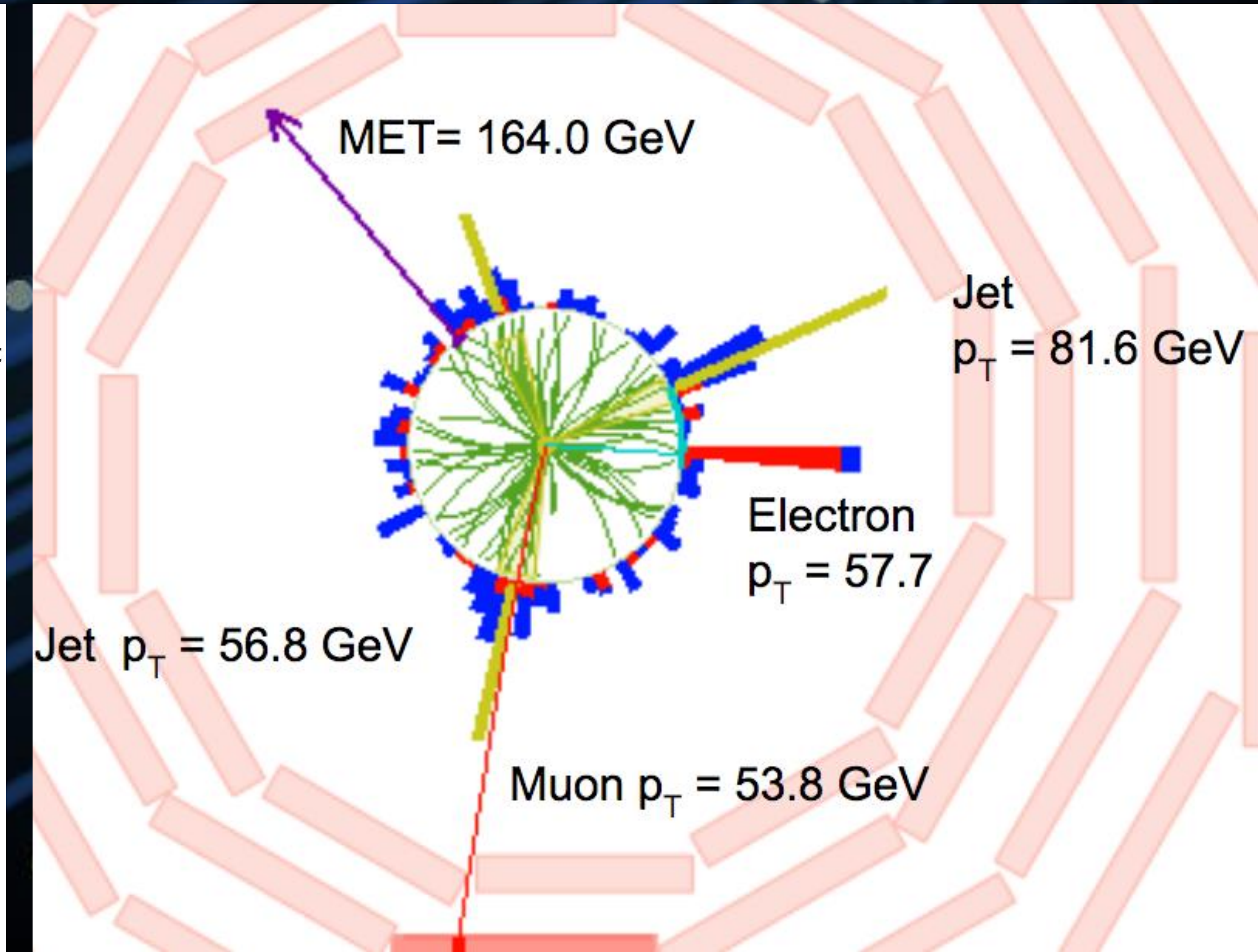
# The CMS detector

# Event reconstruction

- Join the tracks with the energy deposits
- Apply energy and momentum conservation to reconstruct everything all the way down to the interaction point



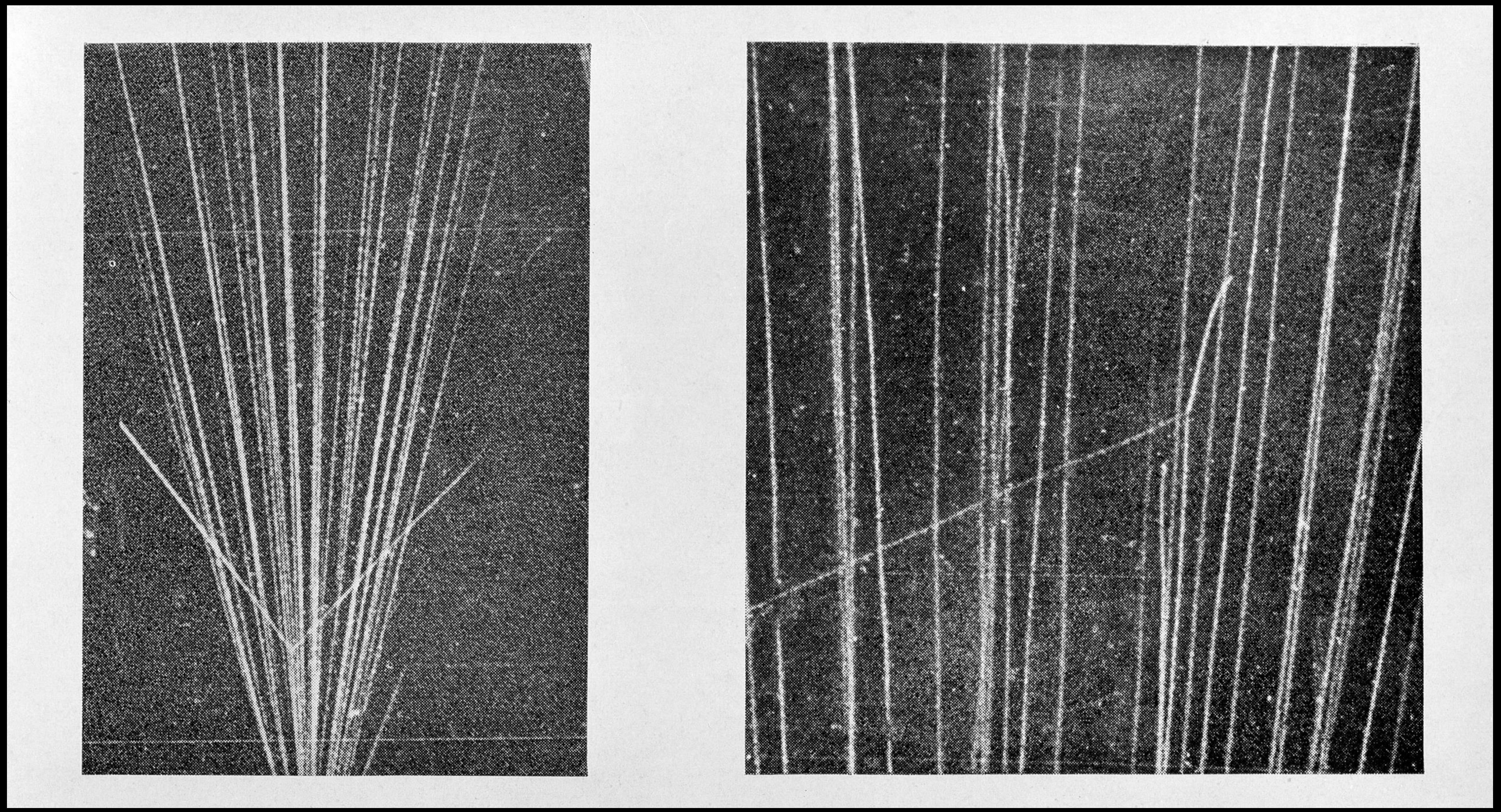
# Spares: Introduction to triggers

# REMINDER

- Trigger basic requirements
  - Need high efficiency for selecting processes for physics analysis
  - Need large reduction of rate from unwanted high-rate processes
  - Robustness is essential
  - Highly flexible,  to react to changing conditions
  - System must be affordable

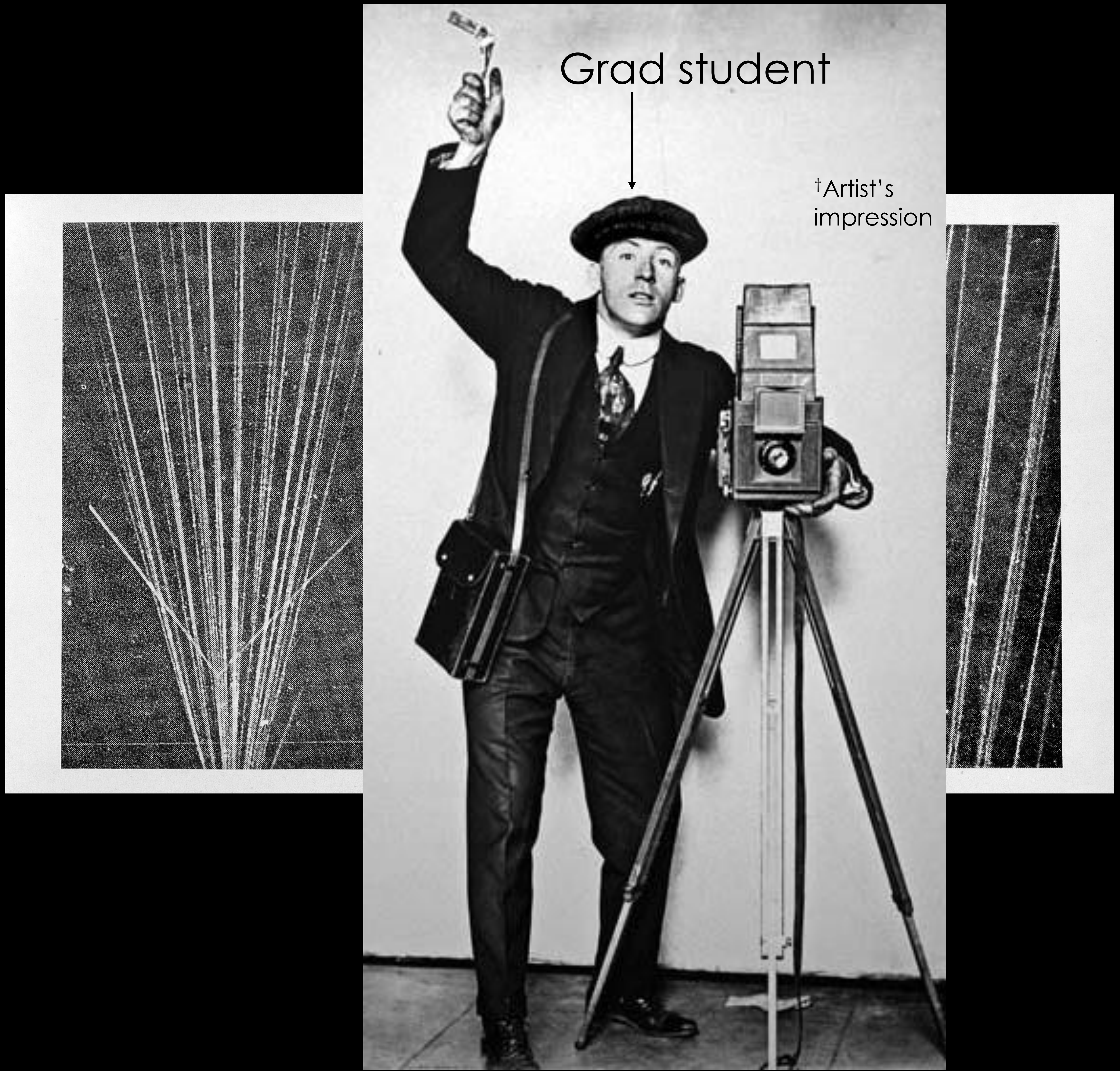
# THE EARLIEST TRIGGER

- Cloud-chamber images recorded on film
- Need some way to trigger the camera



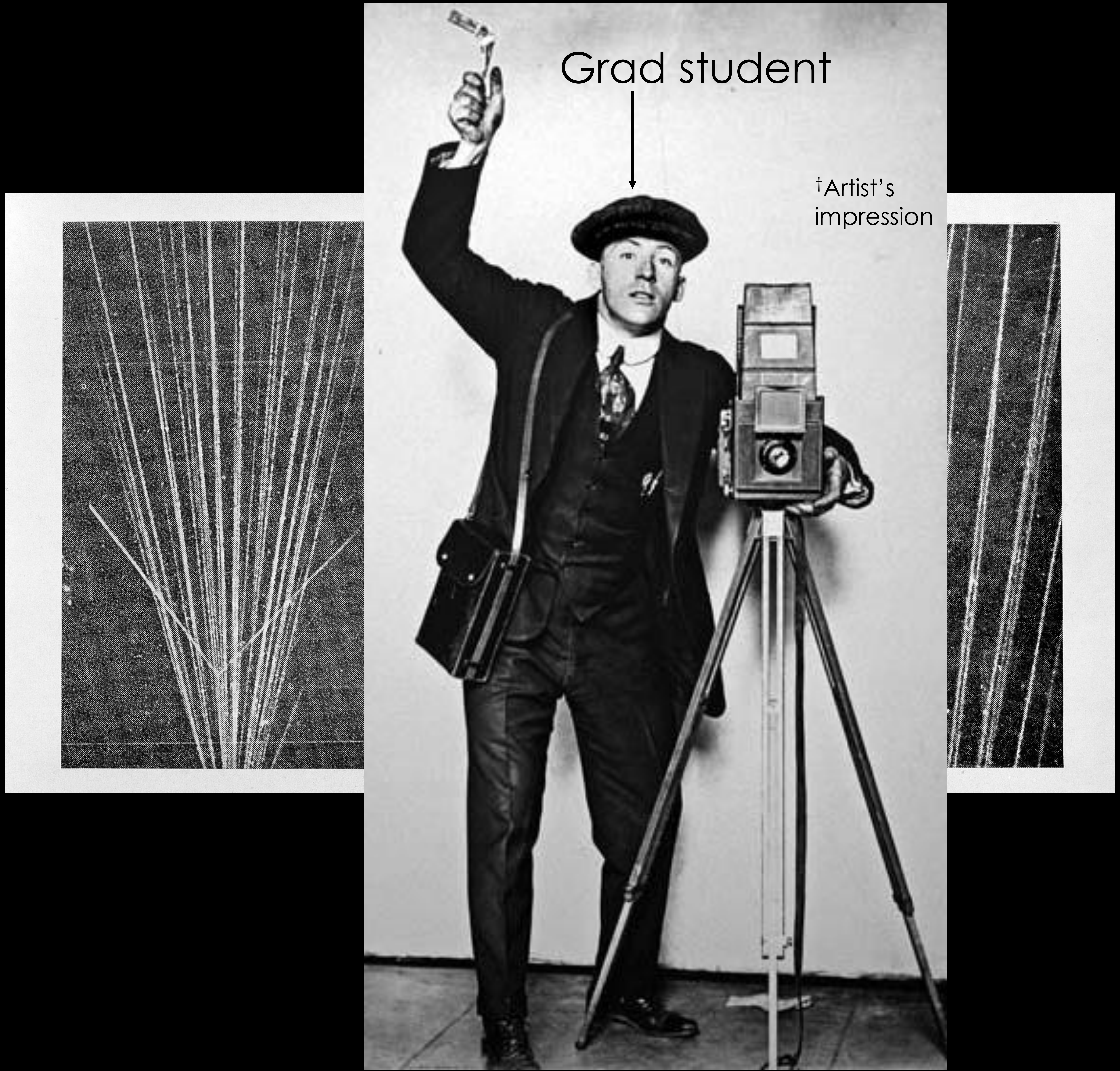
# THE EARLIEST TRIGGER

- Cloud-chamber images recorded on film
- Need some way to trigger the camera



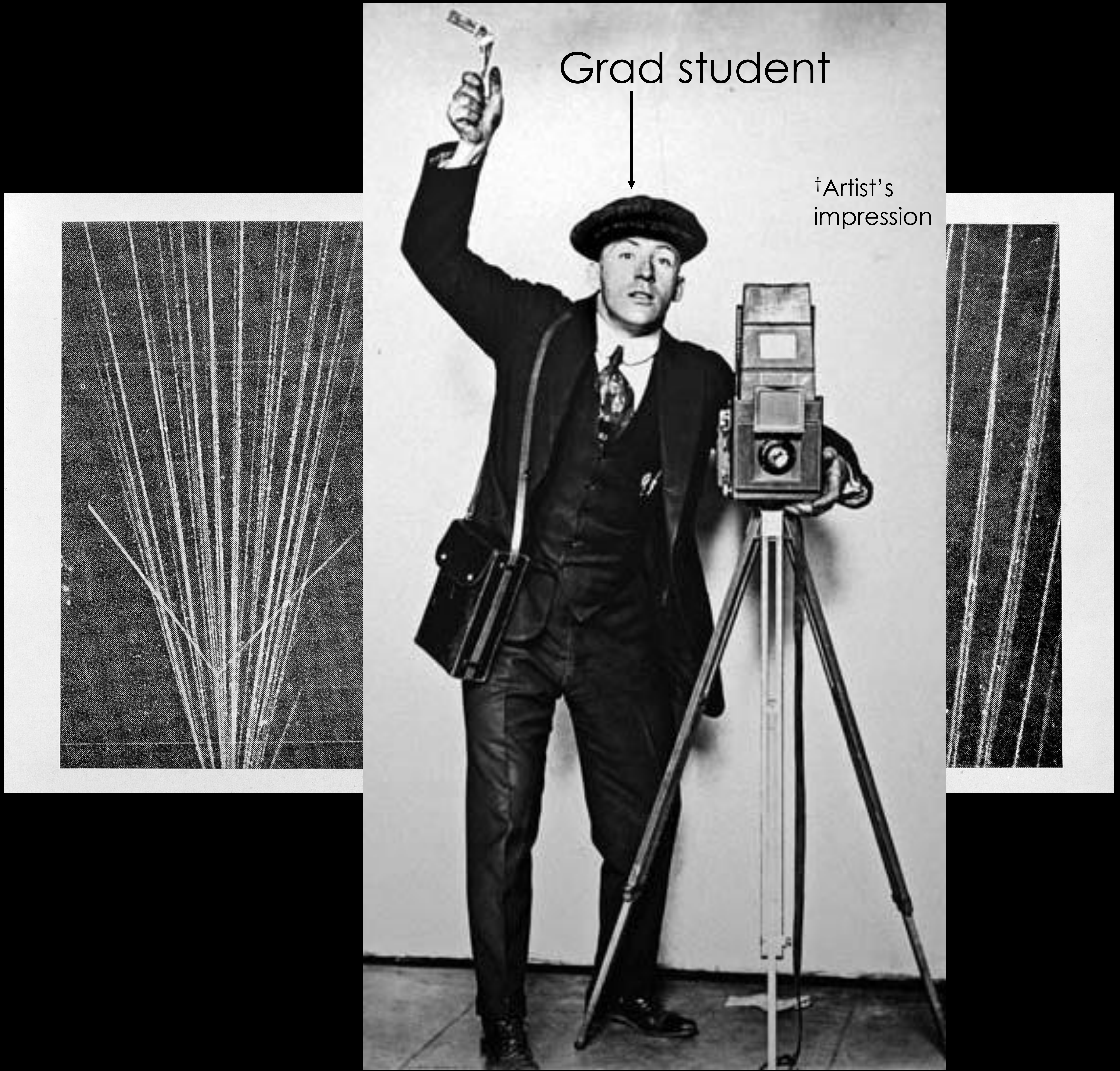
# THE EARLIEST TRIGGER

- High efficiency? Nope – reflexes too slow
- Large rate reduction? Better than nothing
- Robustness? No – keep wanting sleep, coffee, toilet breaks, etc.
- Highly flexible? Depends on the student



# THE EARLIEST TRIGGER

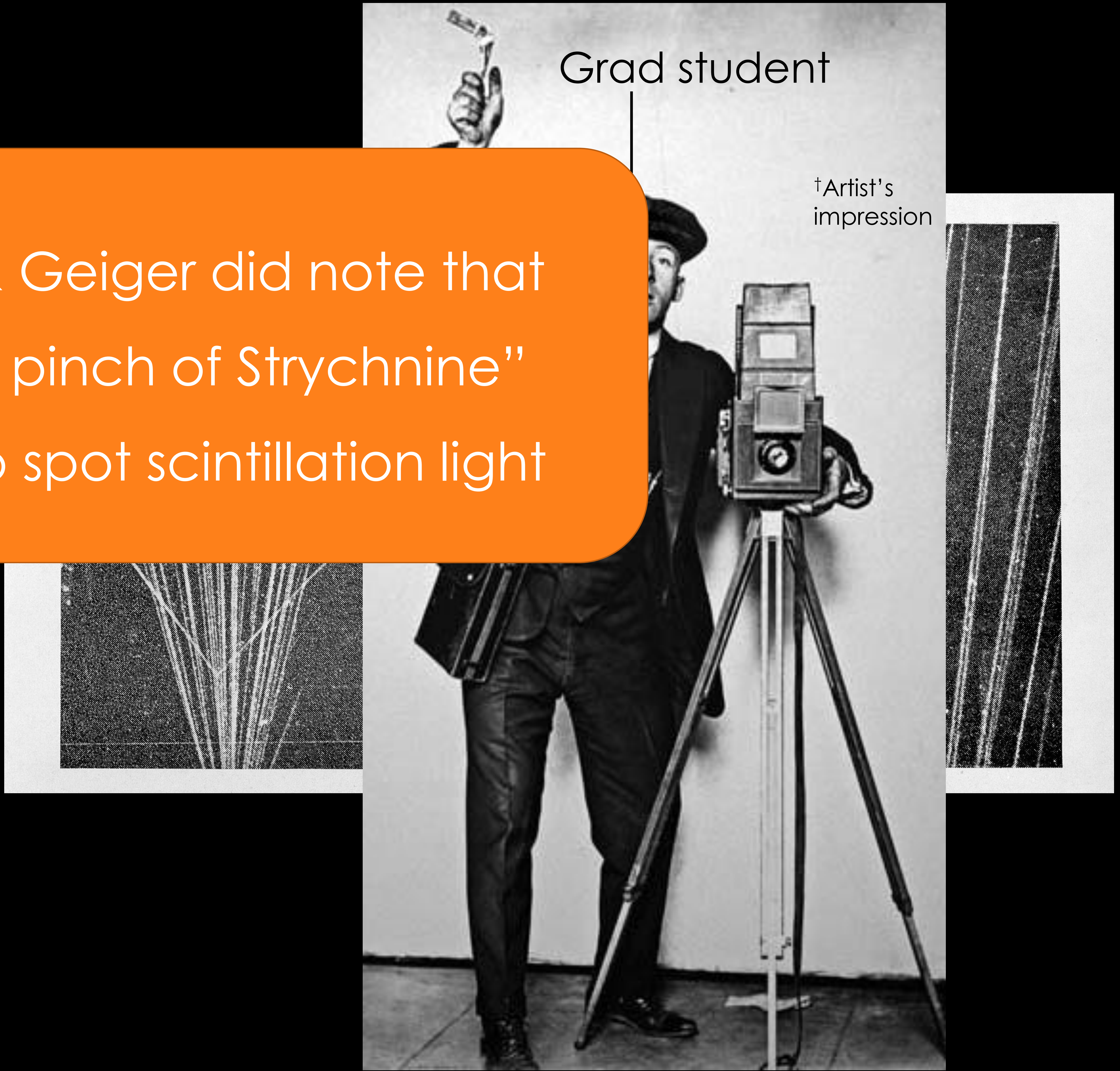
- High efficiency? Nope – reflexes too slow
- Large rate reduction? Better than nothing
- Robustness? No – keep wanting sleep, coffee, toilet breaks, etc.
- Highly flexible? Depends on the student
- Affordable? Well that's one thing in your favour, I suppose



# THE EARLIEST TRIGGER

- High efficiency? N
- Large rate reduc
- Robustness? No
  coffee, toilet bre
- Highly flexible? Depends on the student
- Affordable? Well that's one thing in your favour, I suppose

Although Rutherford & Geiger did note that "Strong coffee with a pinch of Strychnine" improved the ability to spot scintillation light



# THE EARLIEST TRIGGER

- Blackett pioneered a technique to trigger the camera of cloud chambers (and got the Nobel prize for this and other work)
- Just missed out on discovering the positron in 1932
- Stevenson and Street used this to confirm the discovery of the muon in 1937

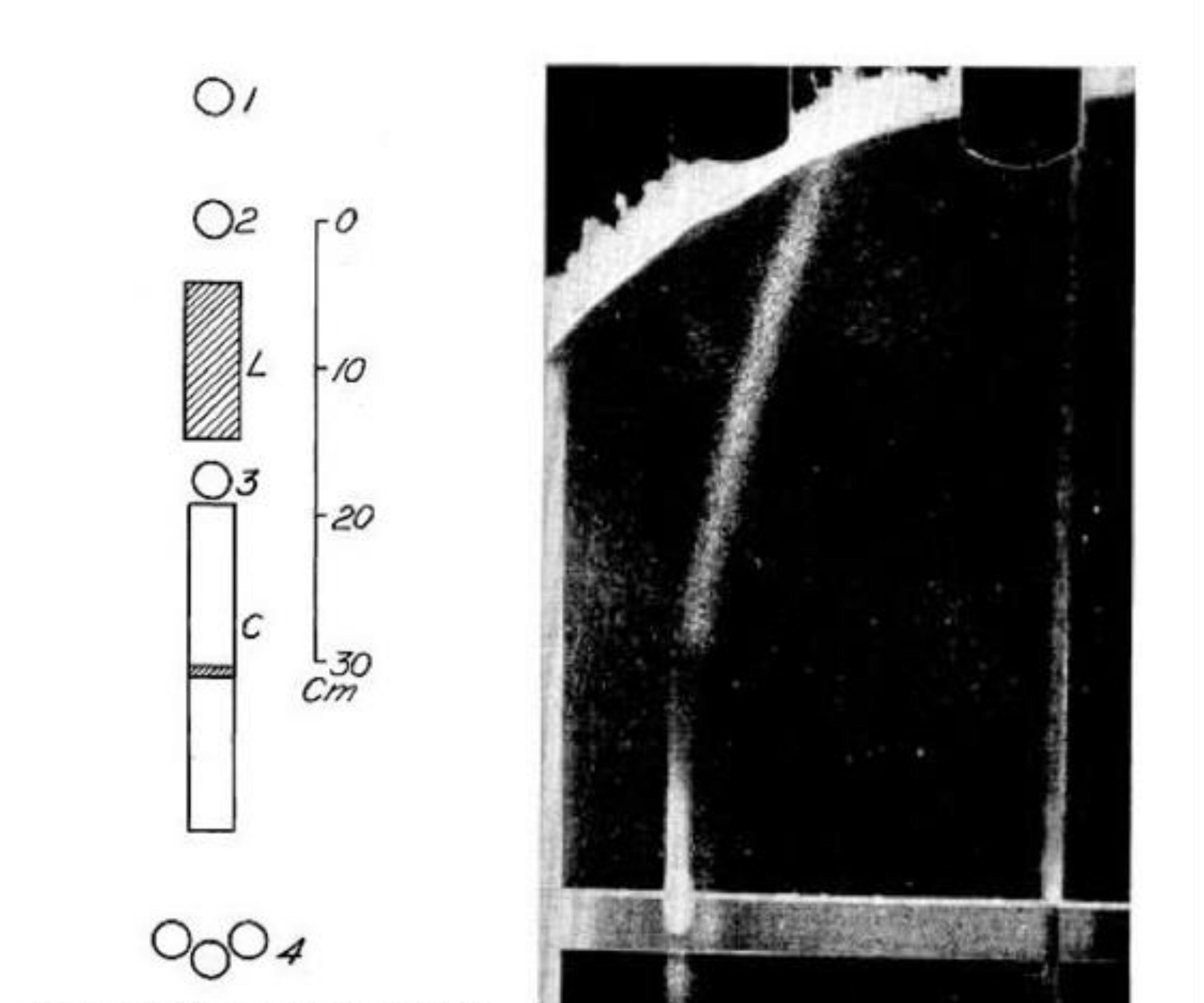
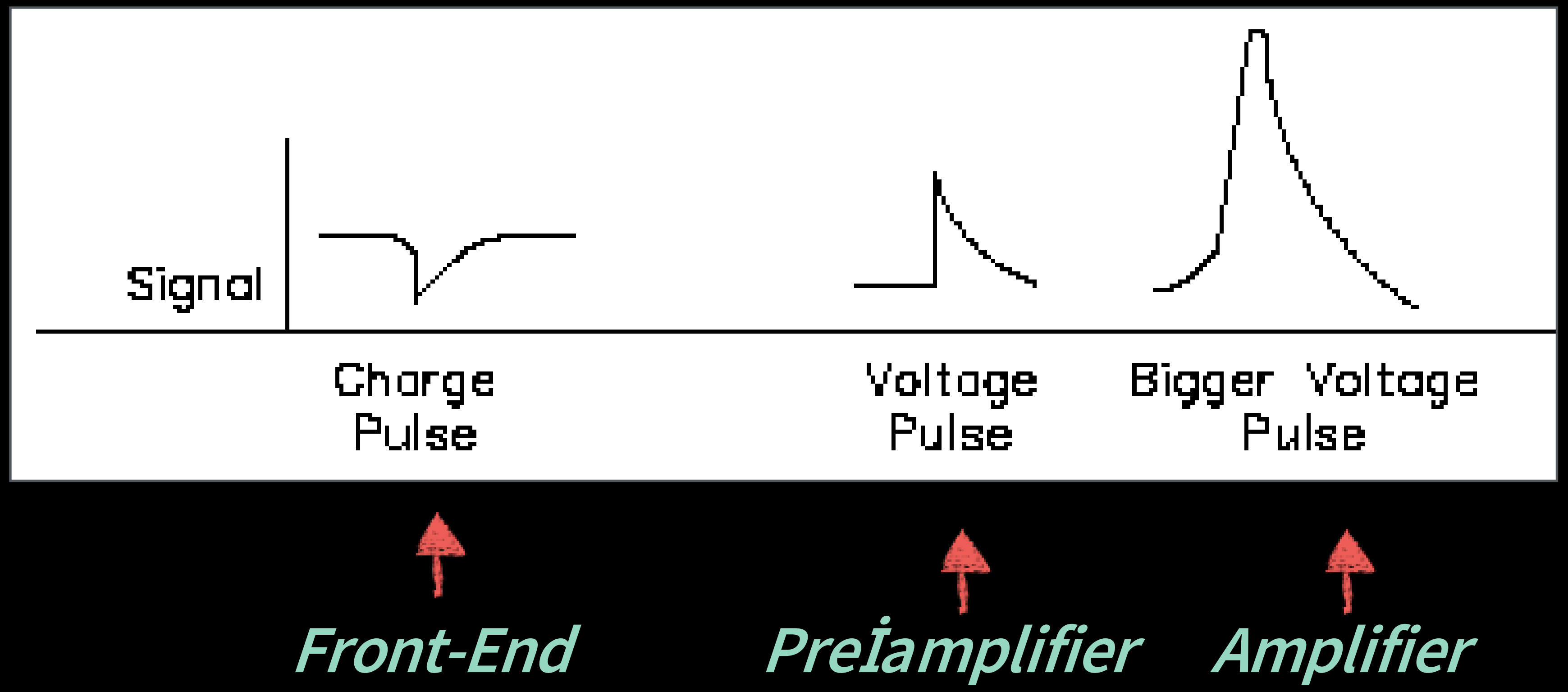


FIG. 1. Geometrical arrangement of apparatus.

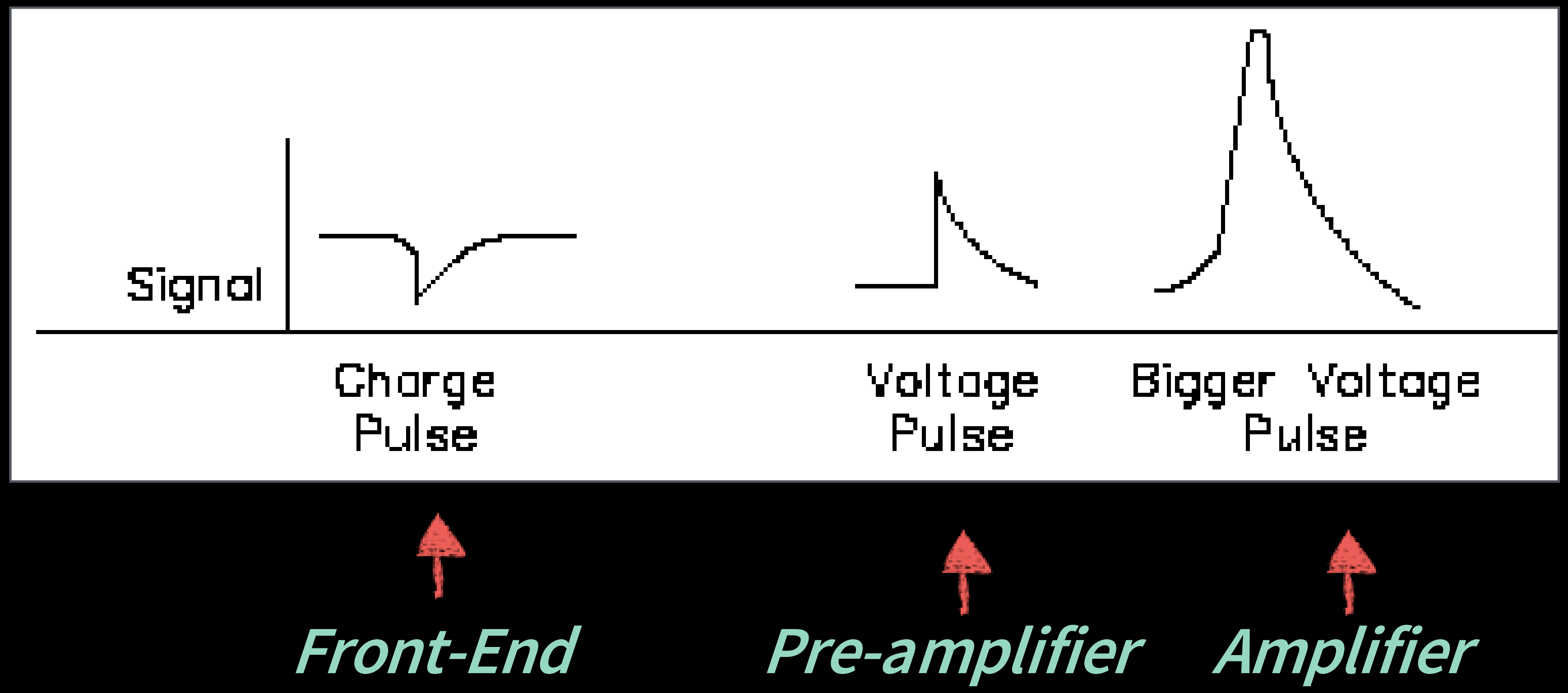
# THE SIMPLEST TRIGGER SYSTEMS

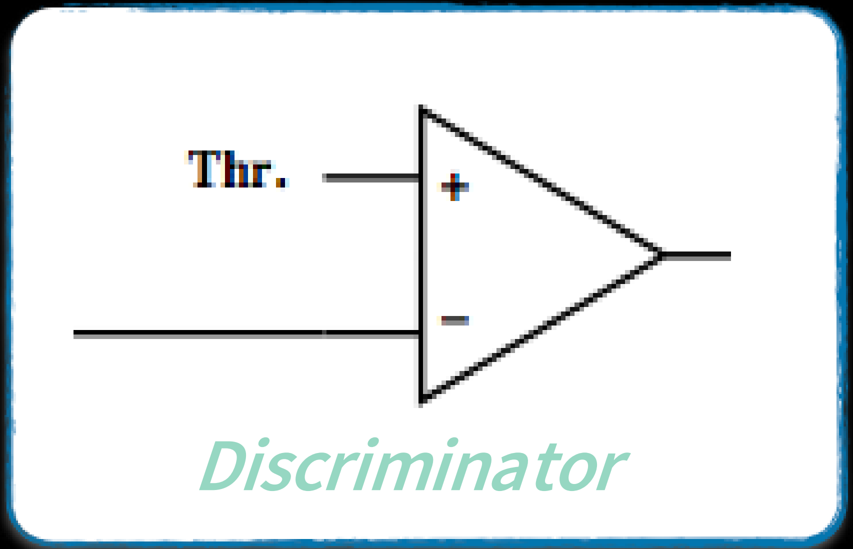
- Source: Use the signals from the Front-End of the detectors themselves
  - Binary: tracking detectors (pixels, strips)
  - Analog: tracking detectors, time of flight detectors, calorimeters, …



# THE SIMPLEST TRIGGER SYSTEMS

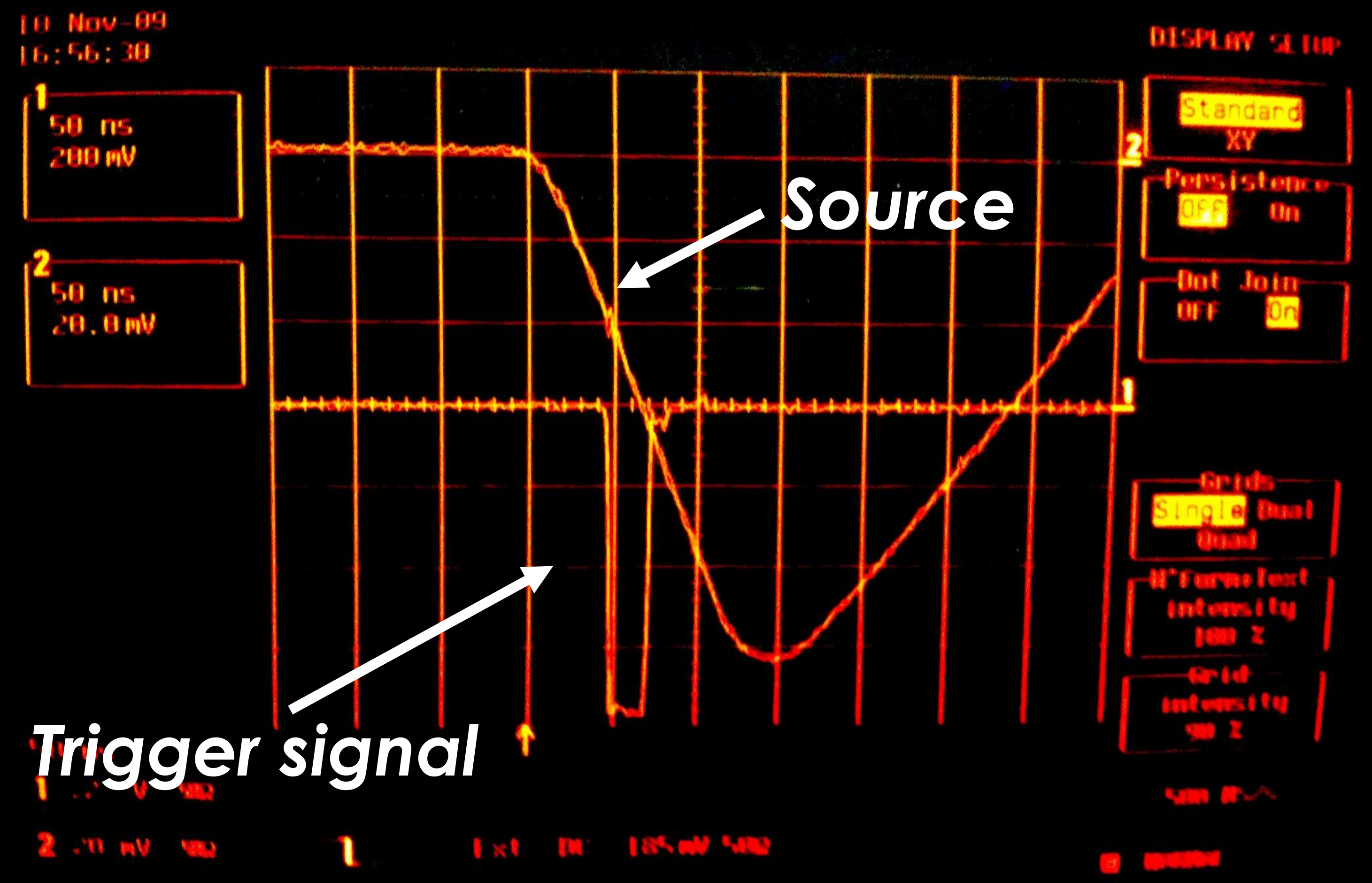
- Source: Use the signals from the Front-End of the detectors themselves
  - Binary: tracking detectors (pixels, strips)
  - Analog: tracking detectors, time of flight detectors, calorimeters, …




Accept

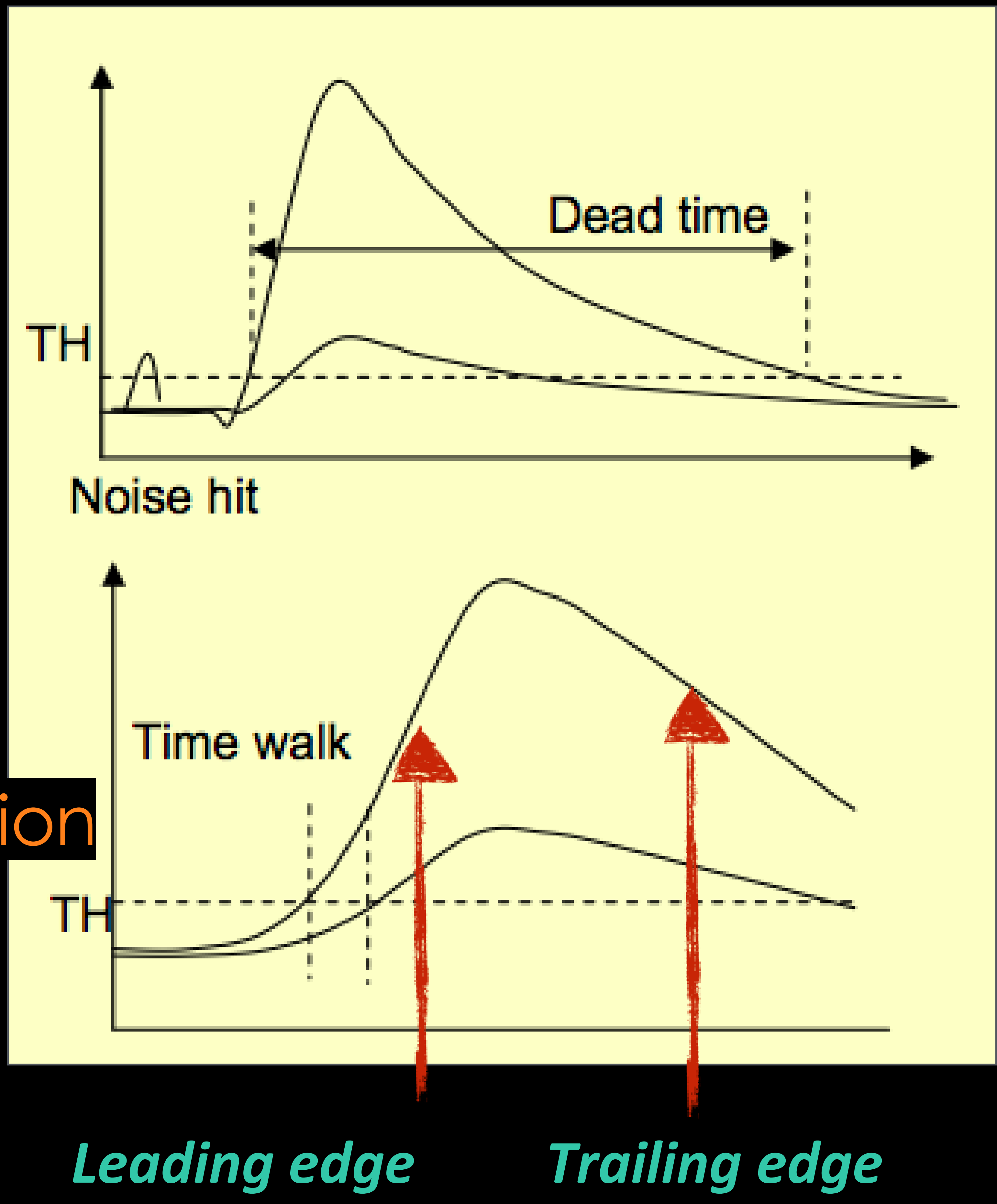
Reject



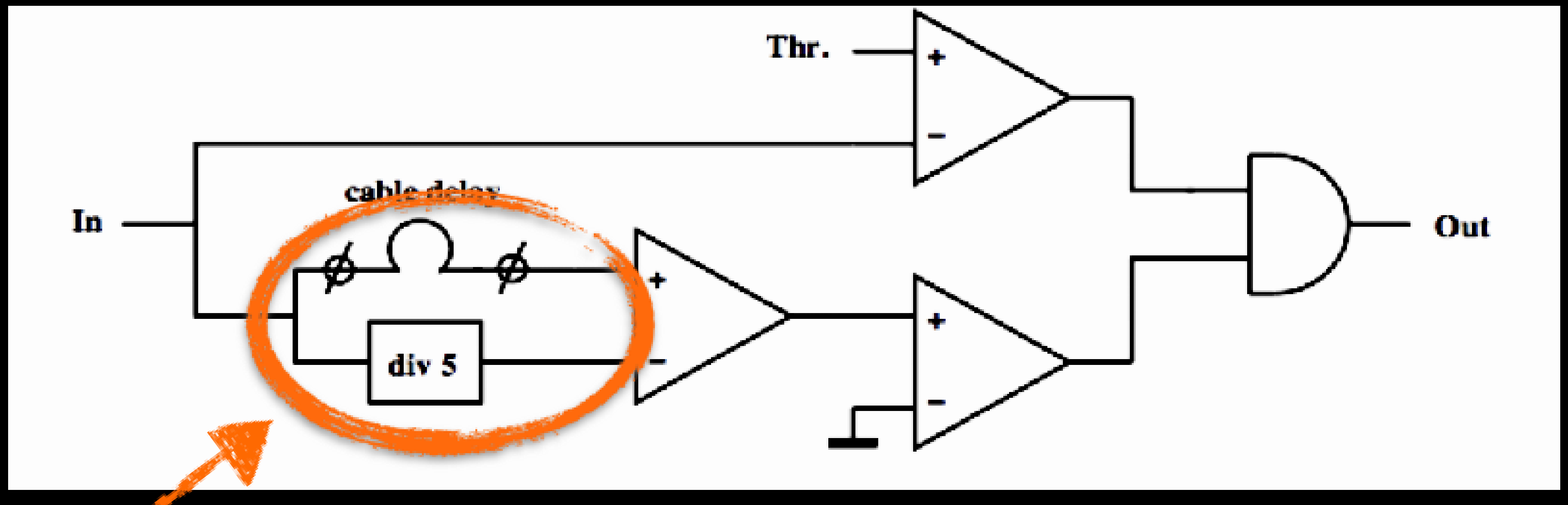

- The most trivial trigger algorithm: ***Signal > Threshold***
  - Apply the lowest possible threshold
  - Identify best compromise between hit efficiency and noise rate

# DETECTOR SIGNALS CHARACTERISTICS

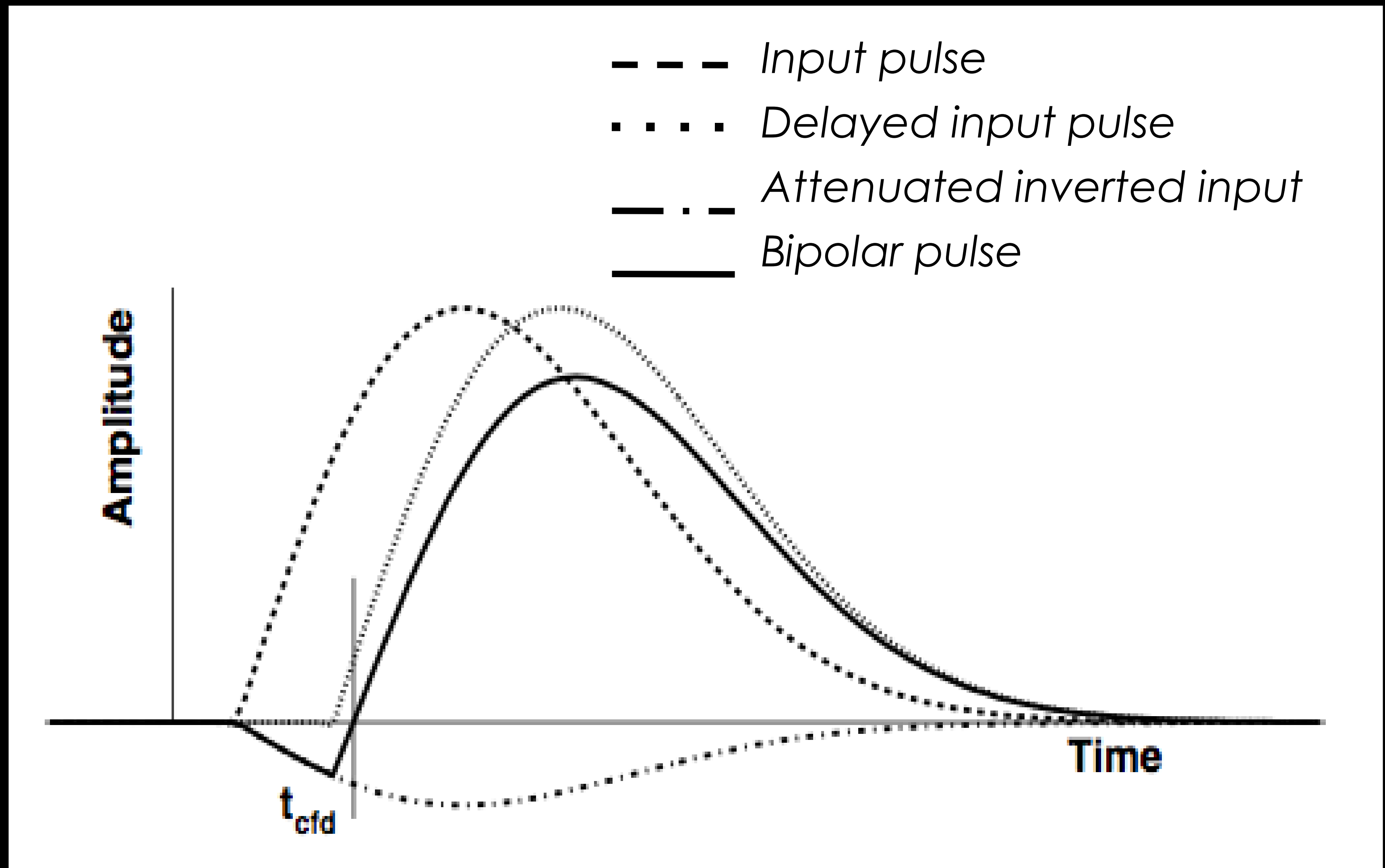
- Pulse width
  - Limits the effective hit rate
  - Must be adapted to the desired trigger rate
- Time walk
  - The threshold-crossing time depends on the signal amplitude
  - Must be minimal good trigger systems
- Time walk can be suppressed by triggering on total signal fraction
  - Applicable on same-shape input signals with different amplitude
  - Scintillator detectors and photomultipliers



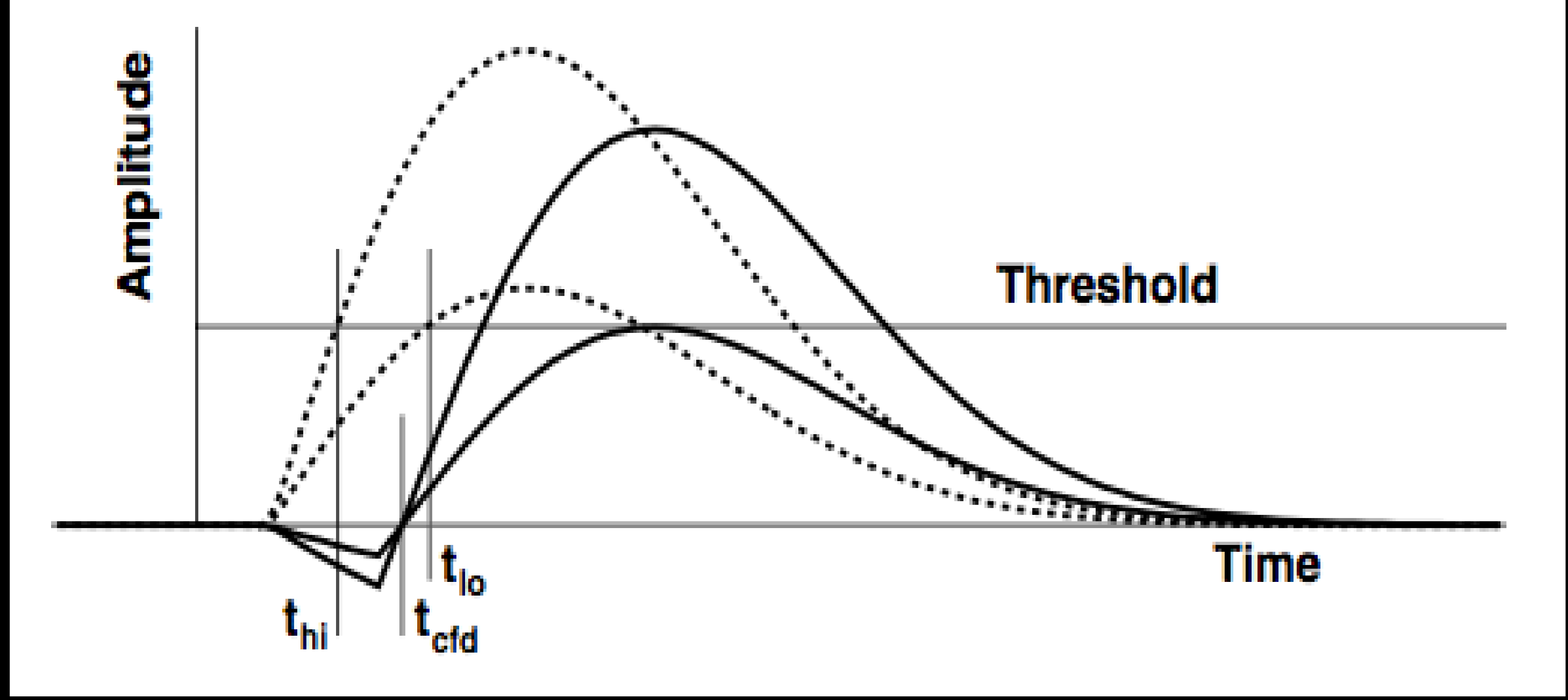
# THE CONSTANT FRACTION DISCRIMINATOR




- Attenuation + configurable delay applied before the discrimination determines $t_{CFD}$
- If delay too short, the unit works as a normal discriminator since the output of the normal discriminator fires later than the CFD part




*The output of the CFD fires when the bipolar pulse changes polarity*



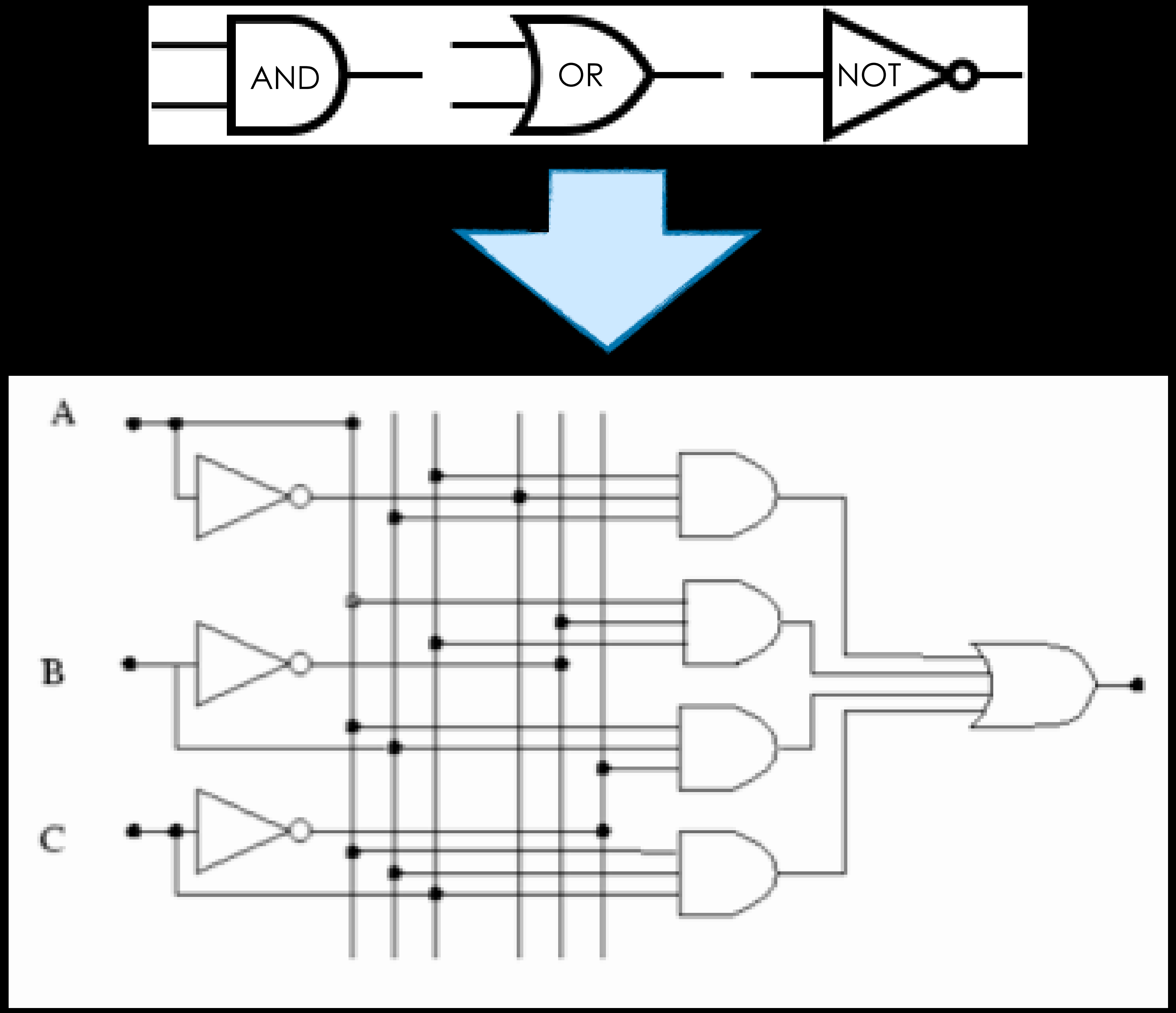

Signals with the same rising time, at a fraction f

$$\Delta t_f = t(f \cdot A_0) - t(A_0) = \text{const.}$$

$$A(t)/f - \cdot A(t - \Delta t) = 0 \quad \text{at } t = t_{cfd}$$

# TRIGGER LOGIC IMPLEMENTATION

- Once we are in the digital domain, all manipulations can be broken down to a Boolean operations
- Combinatorial
  - Summing, Decoders, Multiplexers,…
- Sequential
  - Flip-flops, Registers, Counters,…

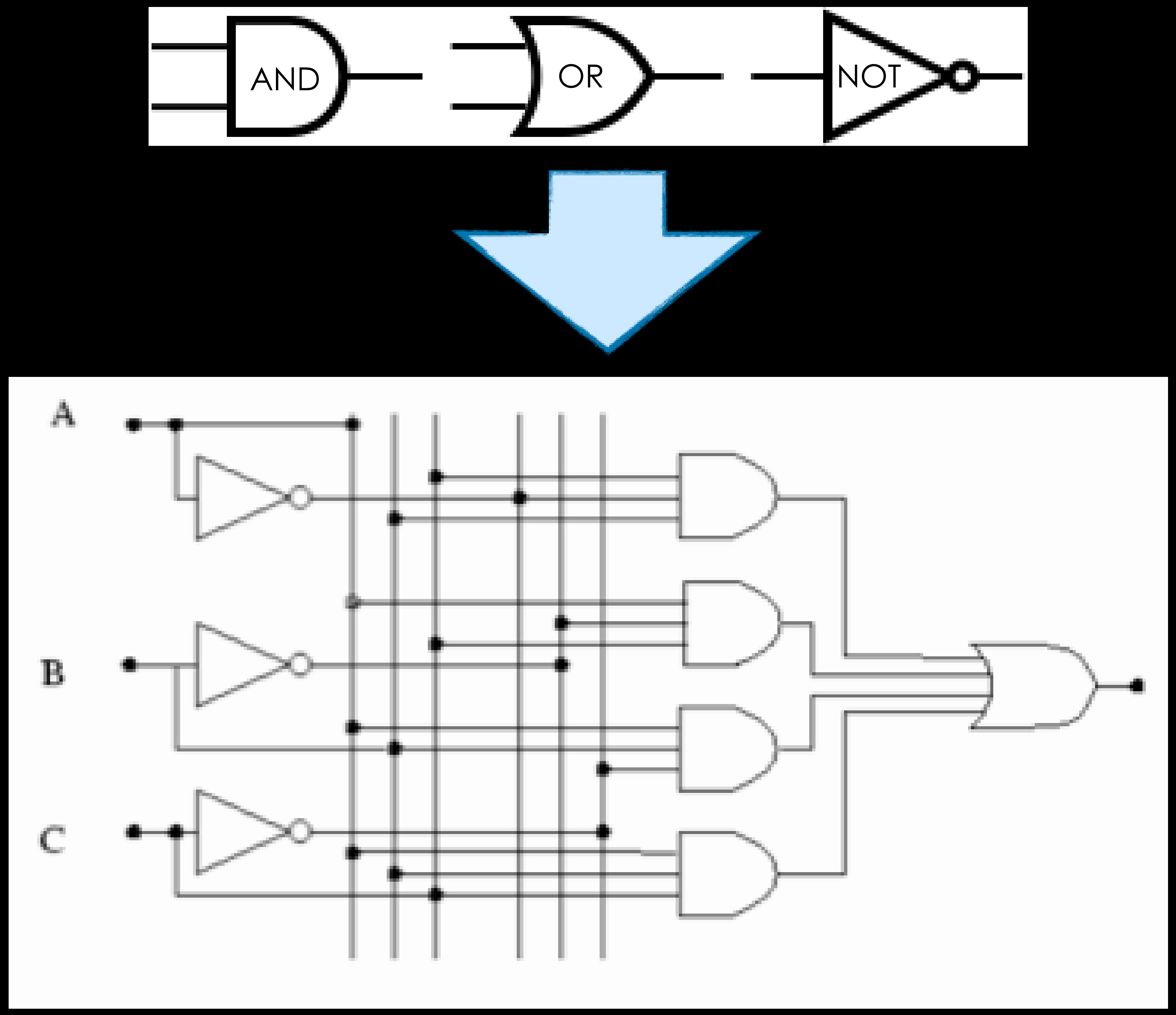


# TRIGGER LOGIC IMPLEMENTATION

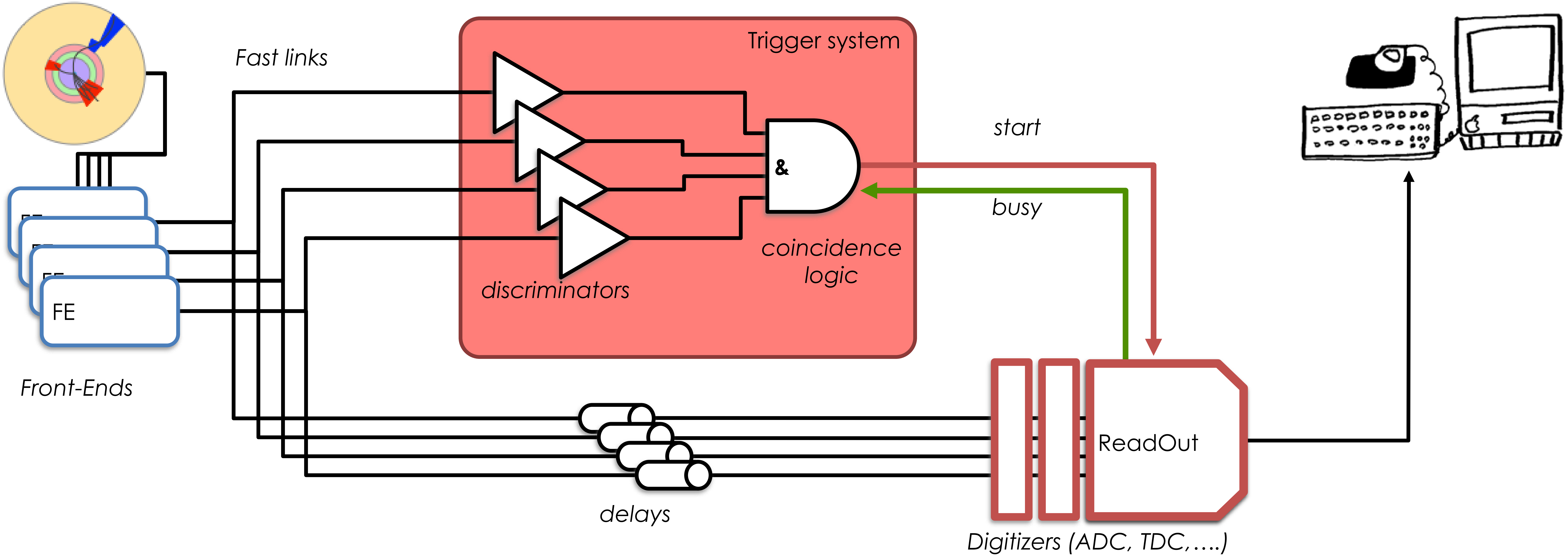
- Once we are in the digital domain, all manipulations can be broken down to a Boolean operations
- Combinatorial
  - Summing, Decoders, Multiplexers,…
- Sequential
  - Flip-flops, Registers, Counters,…

Data propagates as a wave through the logic

Operations happen at well defined times and in a well defined order



# A SIMPLE TRIGGER SYSTEM



# DEADTIME

- The key parameter in high speed trigger systems design
  - The fraction of the acquisition time when no events can be recorded.
  - Typically of the order of **few %**
  - Reduces the overall system efficiency
- Arises when a given processing  step takes a finite amount of time
  - Readout dead-time
  - Trigger dead-time
  - Operational dead-time

# DEADTIME EXAMPLE

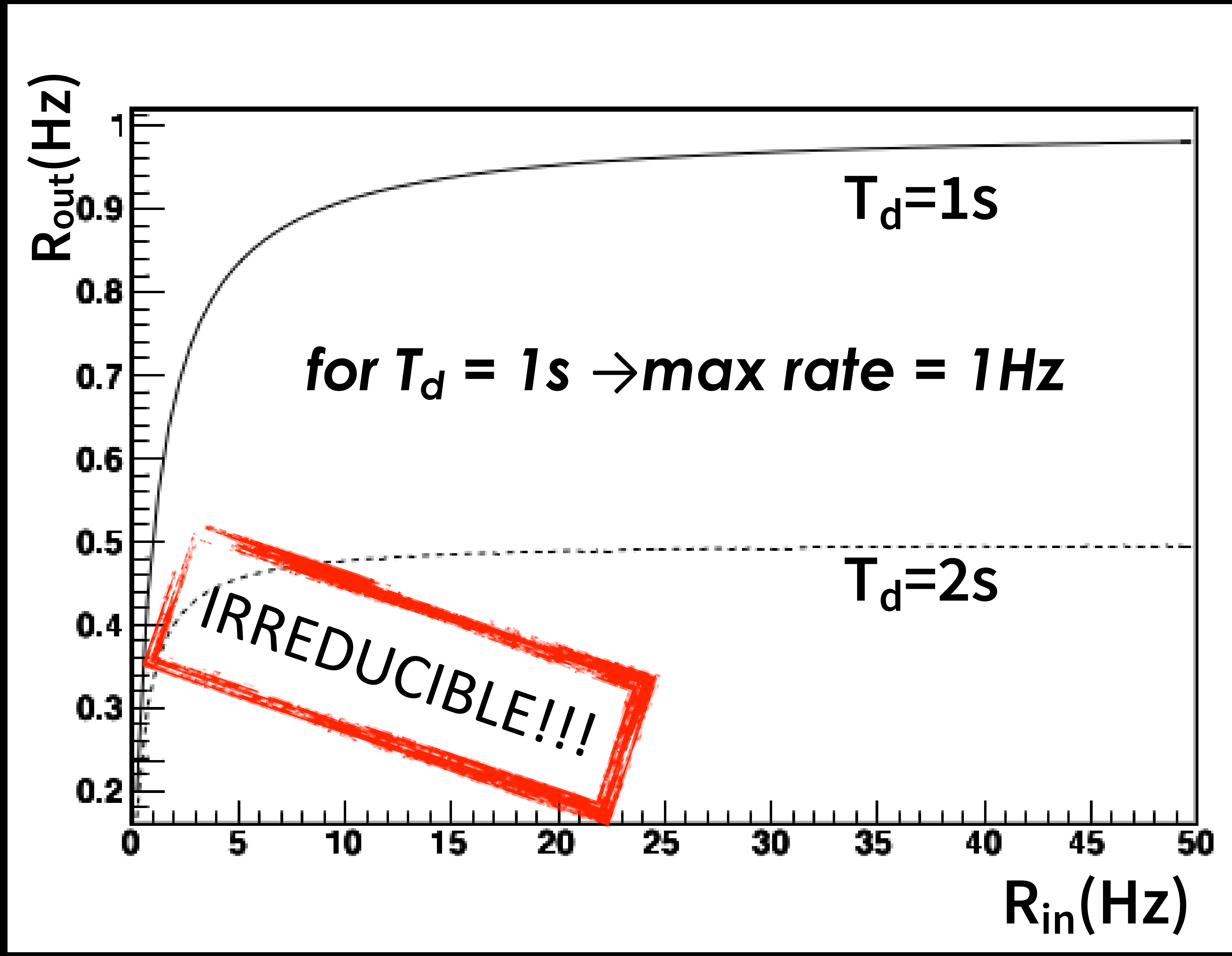
- Writing to disk or tape is much slow than accepting data into RAM
- If you select an event and start writing it to disk, you cannot accept  any more events until you finish writing, even if they are interesting

# DEADTIME EXAMPLE

- For input rate, "$R_{in}$", Readout rate, "$R_{out}$", and time taken to write to disk, "$T_d$"
- Fraction of lost events = $R_{out} \bullet T_d$
- Event output rate $R_{out} = (1 - R_{out} \bullet T_d) \bullet R_{in}$

***Fraction of surviving events***

$$\frac{R_{out}}{R_{in}} = \frac{1}{1 + R_{in} T_d}$$

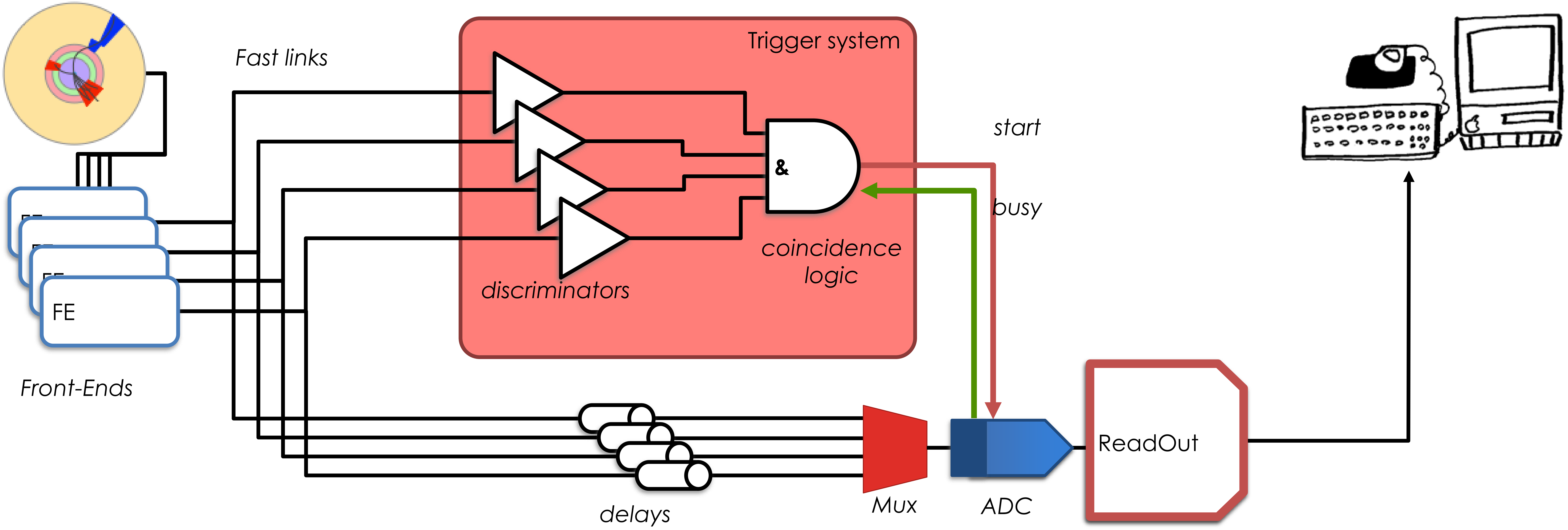



To achieve high efficiency $\Rightarrow R_{in} \cdot T_d \ll 1$

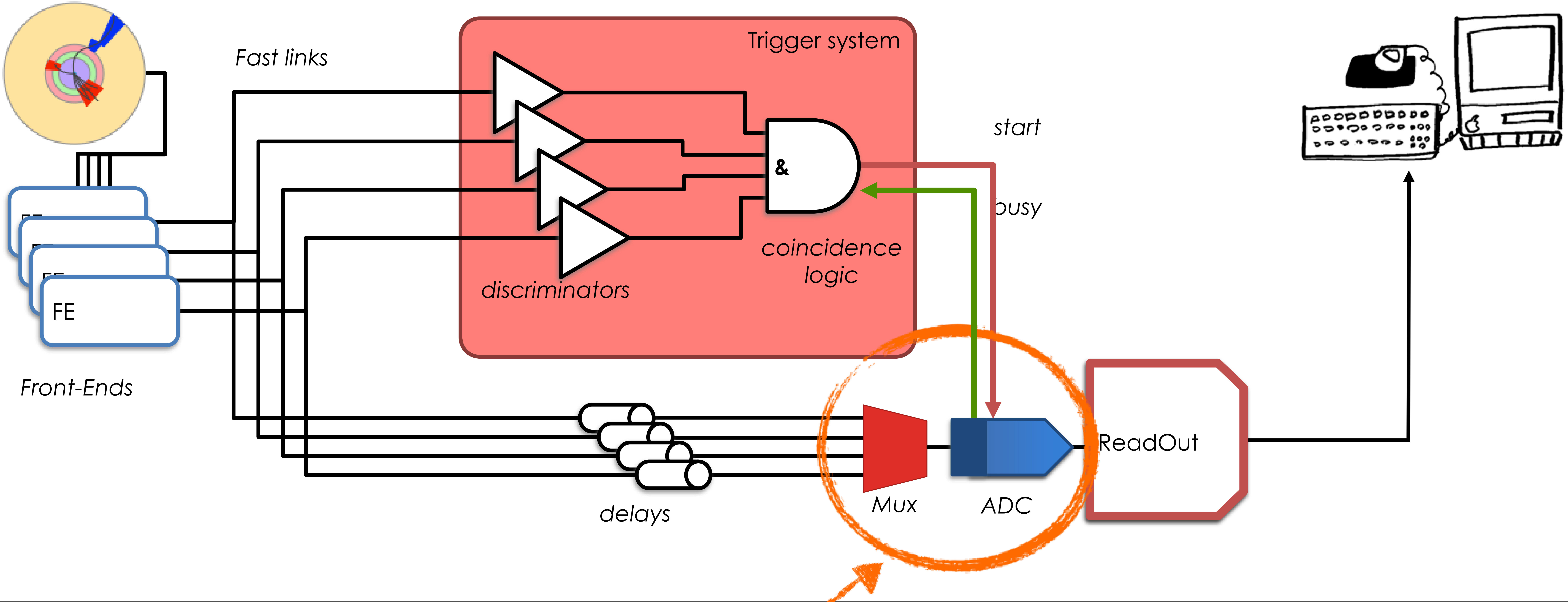
# DEADTIME

- Writing to disk or tape is much slow than accepting data into RAM
- If you select an event and start writing it to disk, you cannot accept any more events until you finish writing, even if they are interesting
- Same principle applies to processing time
  - For example, ADCs

# A SIMPLE TRIGGER SYSTEM: DEADTIME



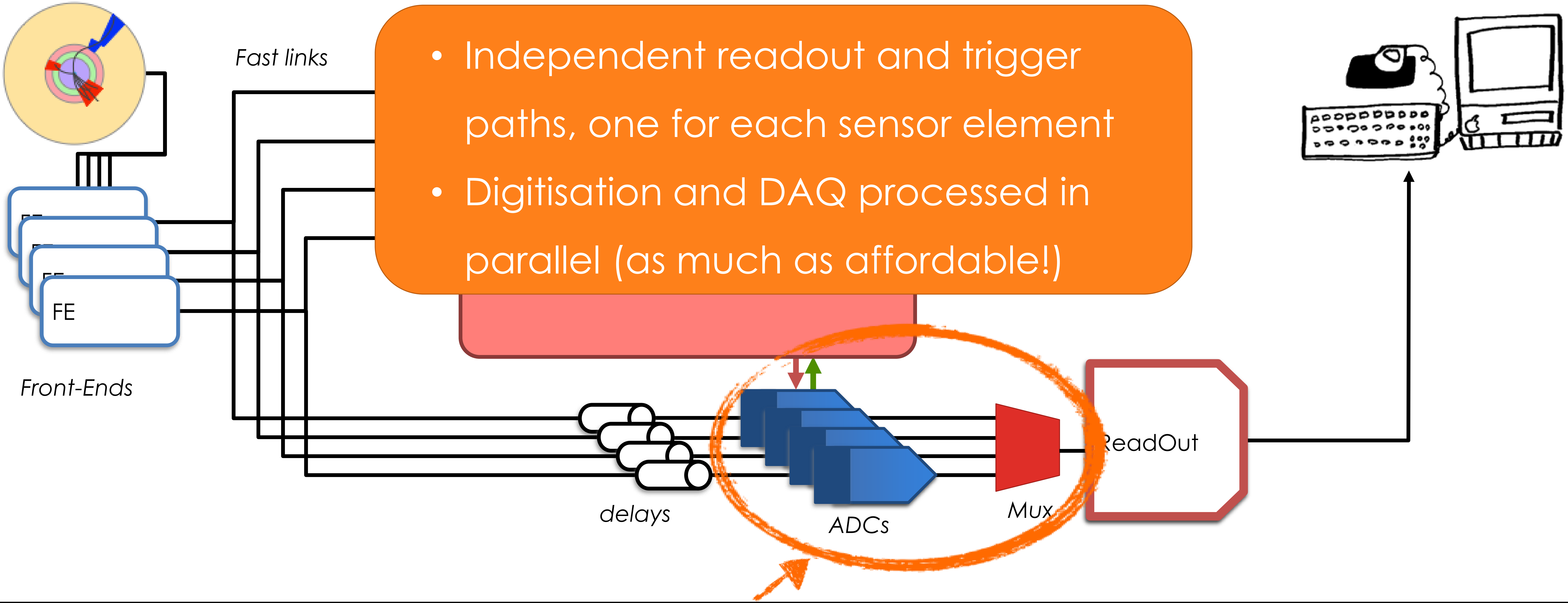
# A SIMPLE TRIGGER SYSTEM: DEADTIME




If ADC is the critical step for deadtime,
this is clearly a really bad plan

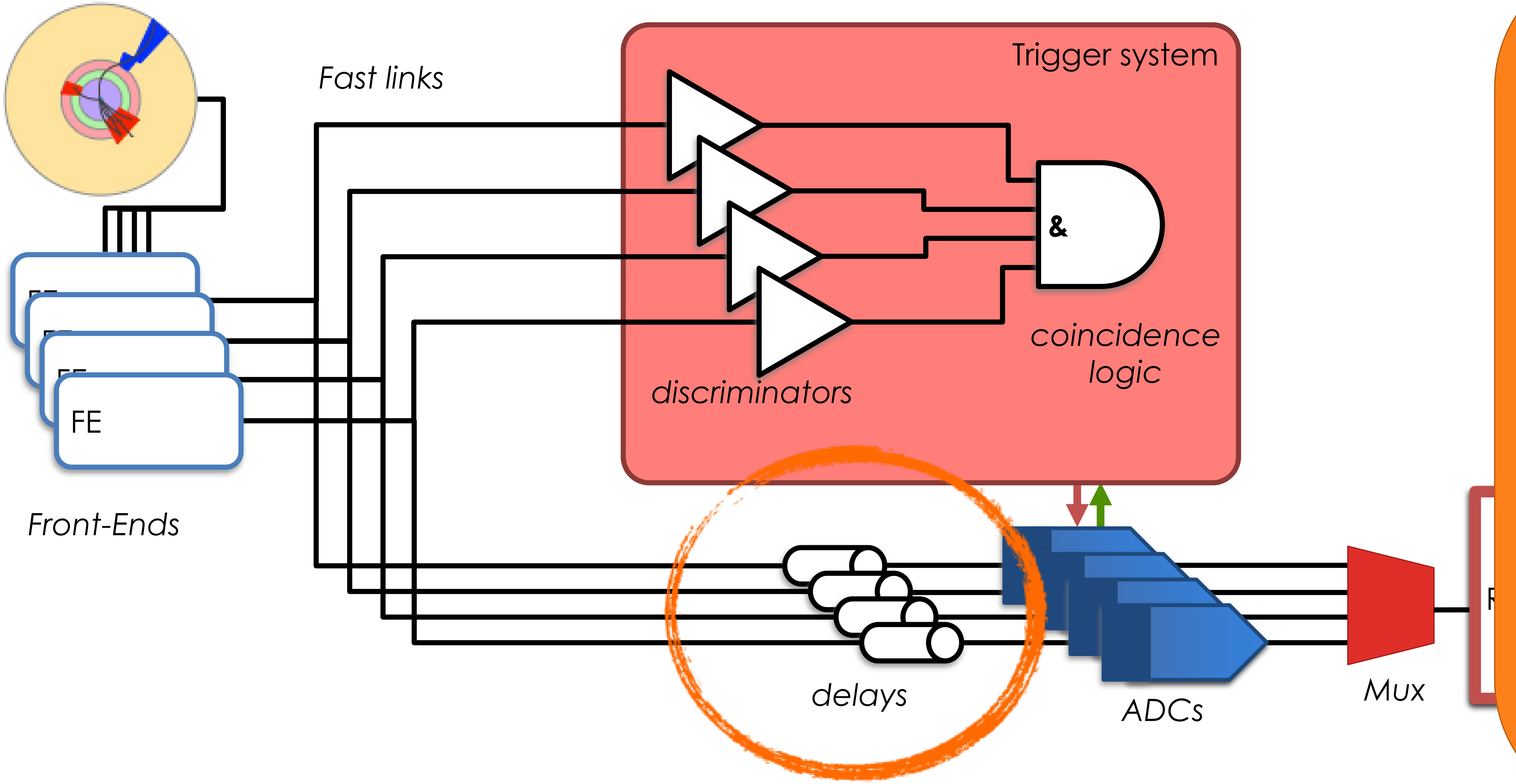
# A SIMPLE TRIGGER SYSTEM: PARALLELISM

- Independent readout and trigger paths, one for each sensor element
- Digitisation and DAQ processed in parallel (as much as affordable!)



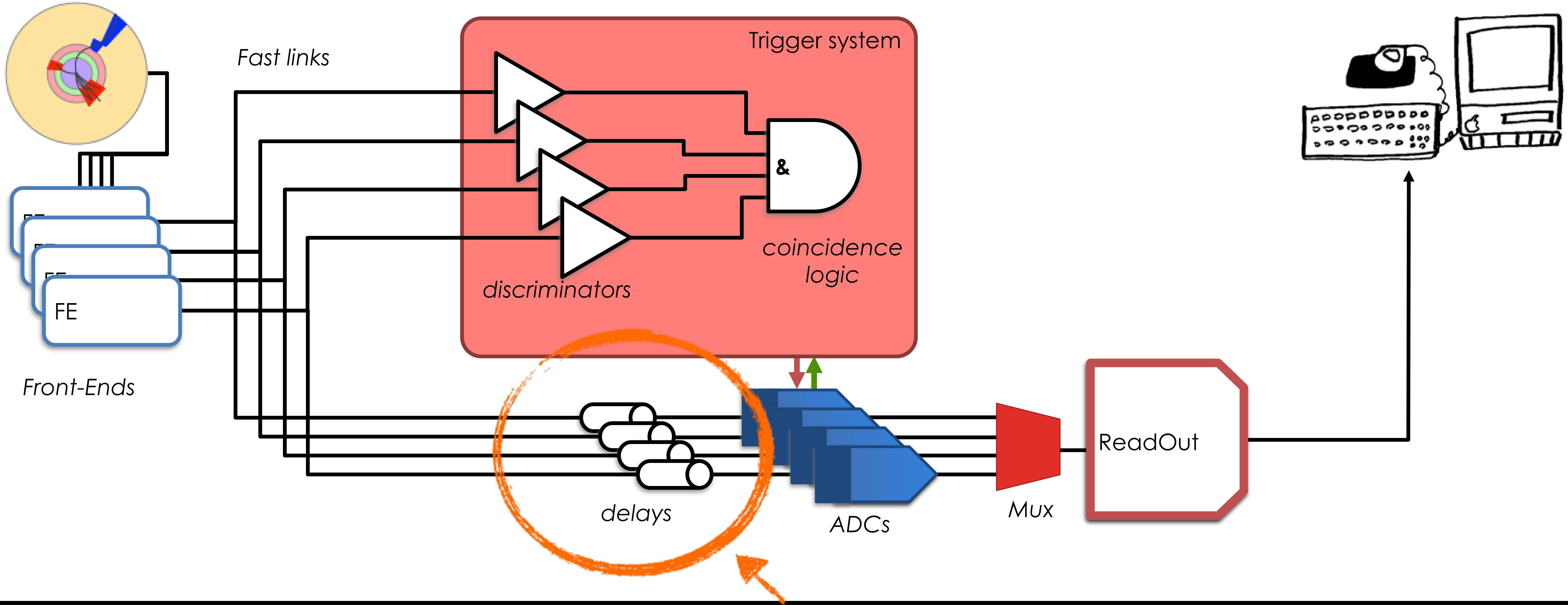

Much more sensible!
Potentially much more expensive!

# A SIMPLE TRIGGER SYSTEM: LATENCY



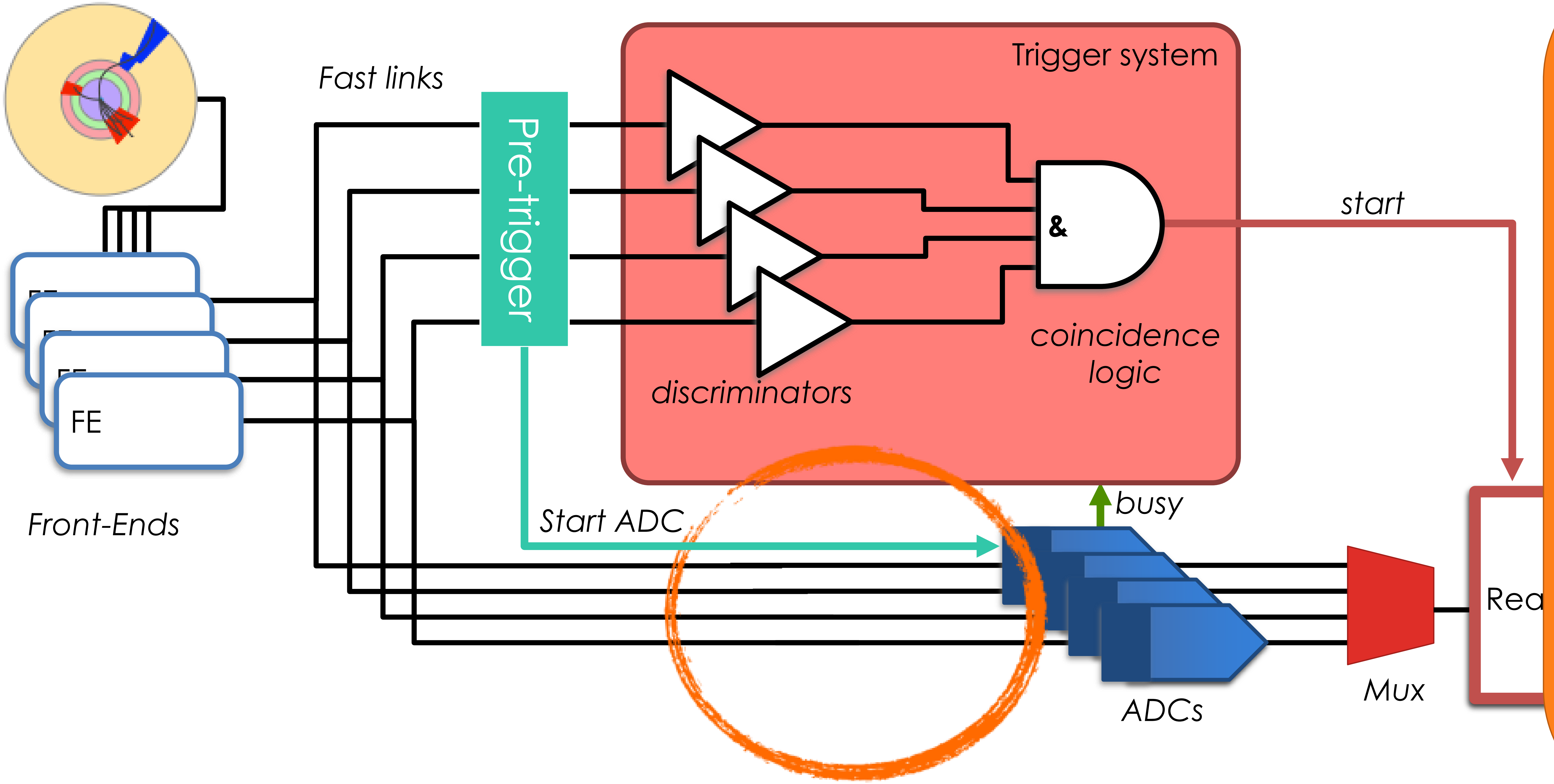

- Latency: Time to form the trigger decision and distribute to the digitisers
- Signals must be delayed until the trigger decision is available
- The more complex is the selection, the longer is the latency

# A SIMPLE TRIGGER SYSTEM: LATENCY



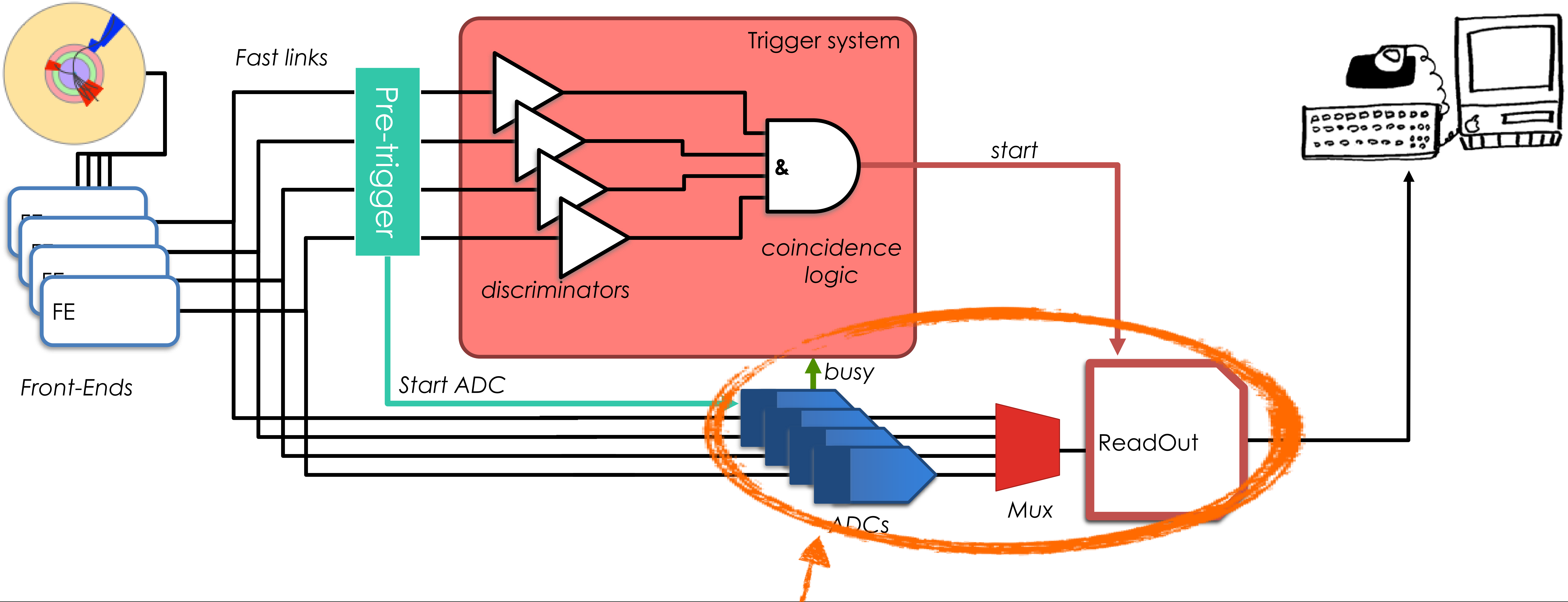

Analogue delay-lines are a bit risky, don't you think?
Especially for more than one channel

# A SIMPLE TRIGGER SYSTEM: PRE-TRIGGER



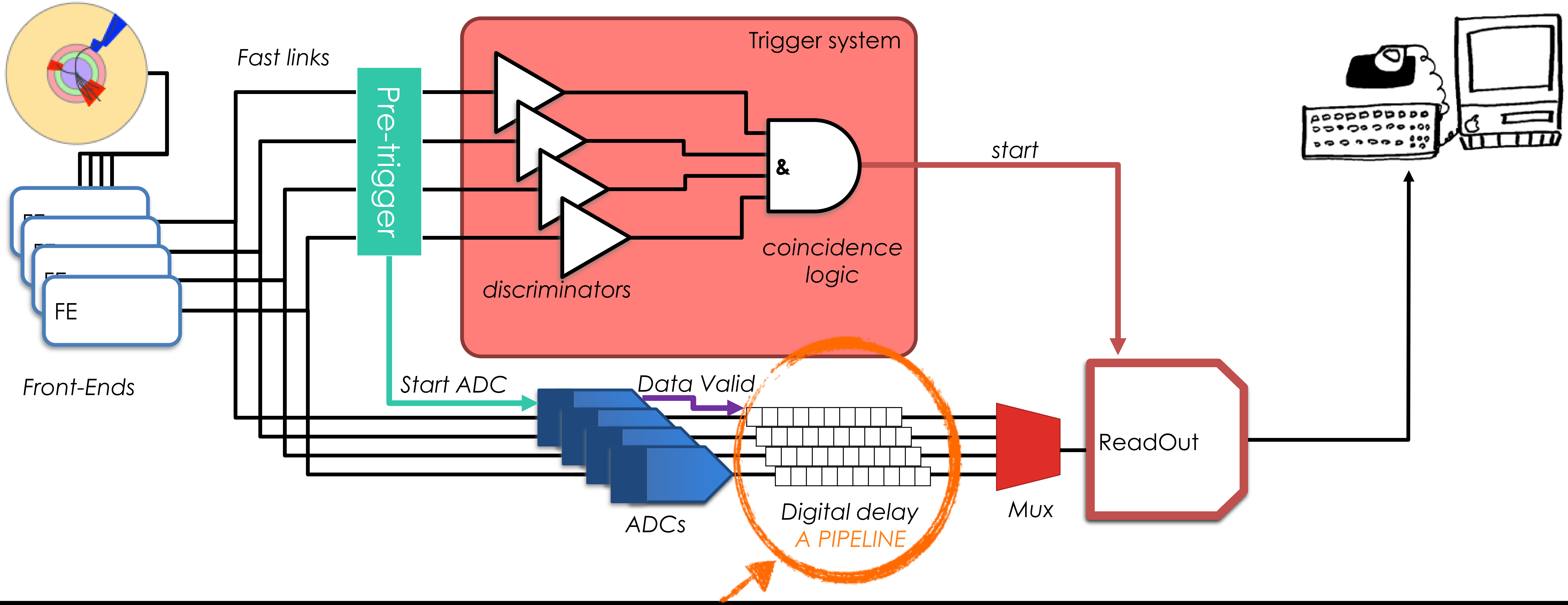

- Pre-Trigger stage: very fast indicator of some minimal activity in the detector
- Used to START the digitisers, with no delay
- The complex trigger decision comes later

# A SIMPLE TRIGGER SYSTEM: PRE-TRIGGER



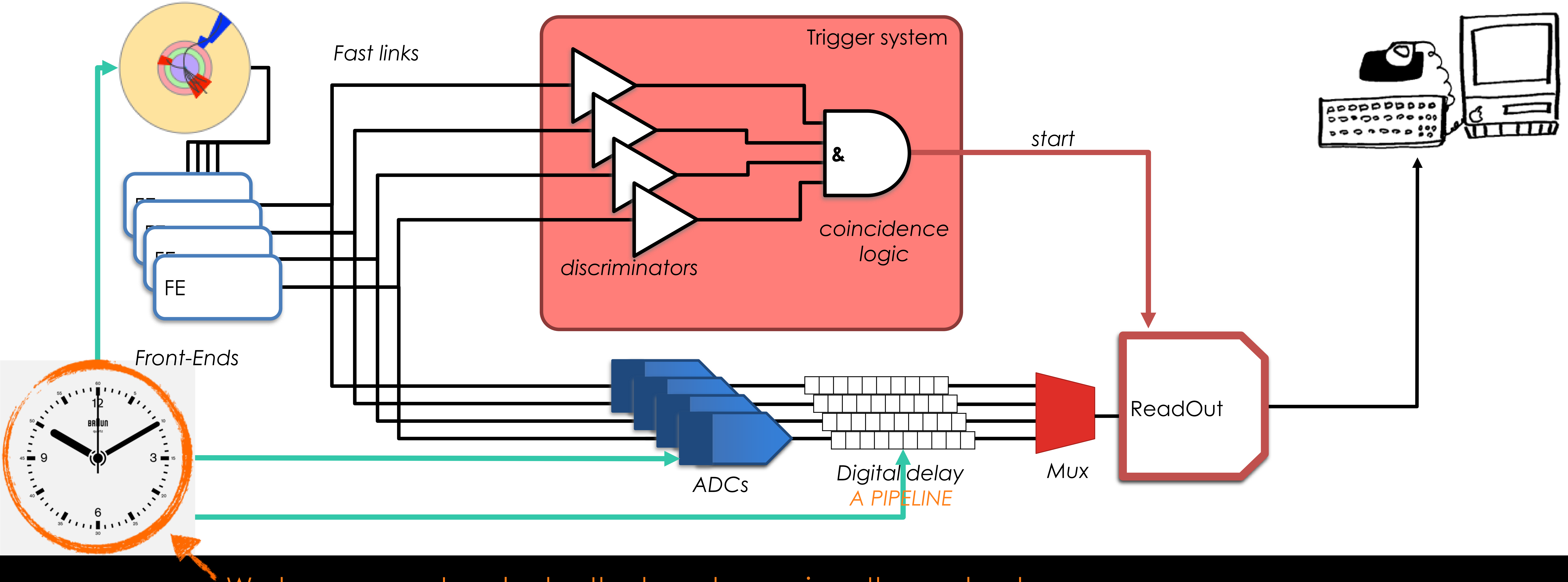

Assumes the digitization time is longer than the latency of the trigger system!
What if that is not true?

# A SIMPLE TRIGGER SYSTEM: PRE-TRIGGER



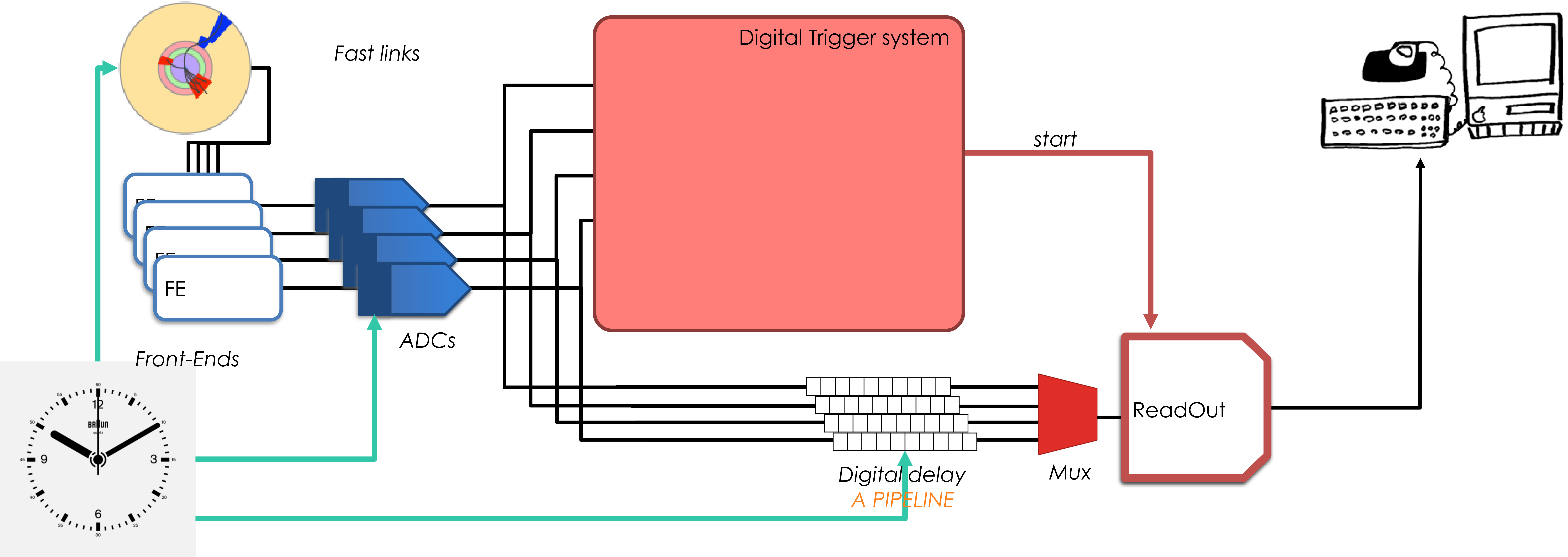

Since each digitization takes a finite time
Can store the result of each digitization in RAM until trigger decision is made

# SIMPLE TRIGGER SYSTEM: BUNCHED COLLIDERS

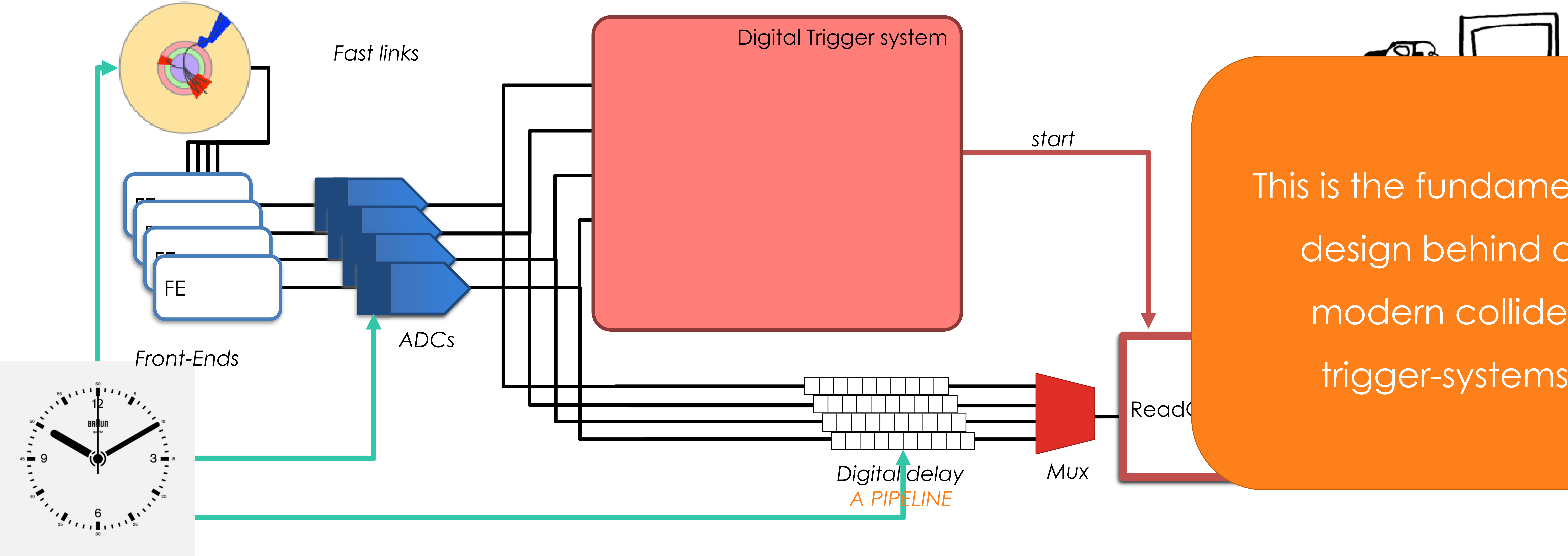



We have a master-clock – the bunch-crossings themselves!
No need for a pre-trigger

# A SIMPLE TRIGGER SYSTEM: DIGITAL TRIGGERS

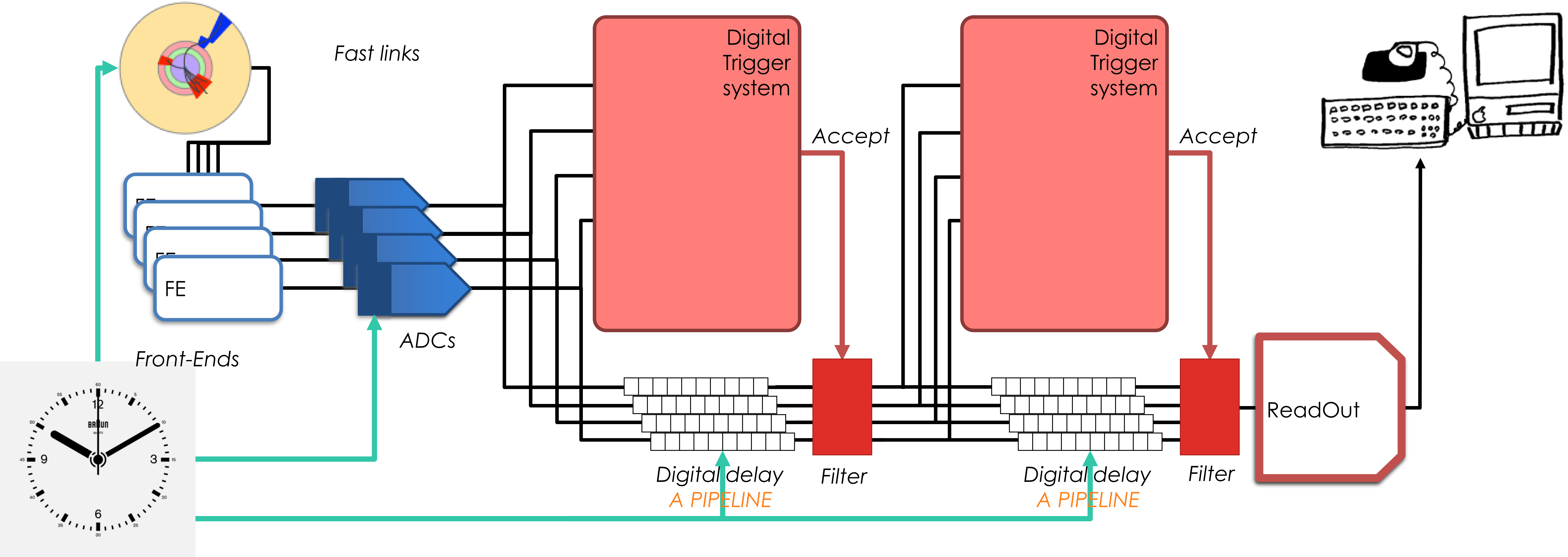


# A SIMPLE TRIGGER SYSTEM: DIGITAL TRIGGERS




This is the fundamental design behind all modern collider trigger-systems

# A TRIGGER SYSTEM: MULTILAYER TRIGGERS



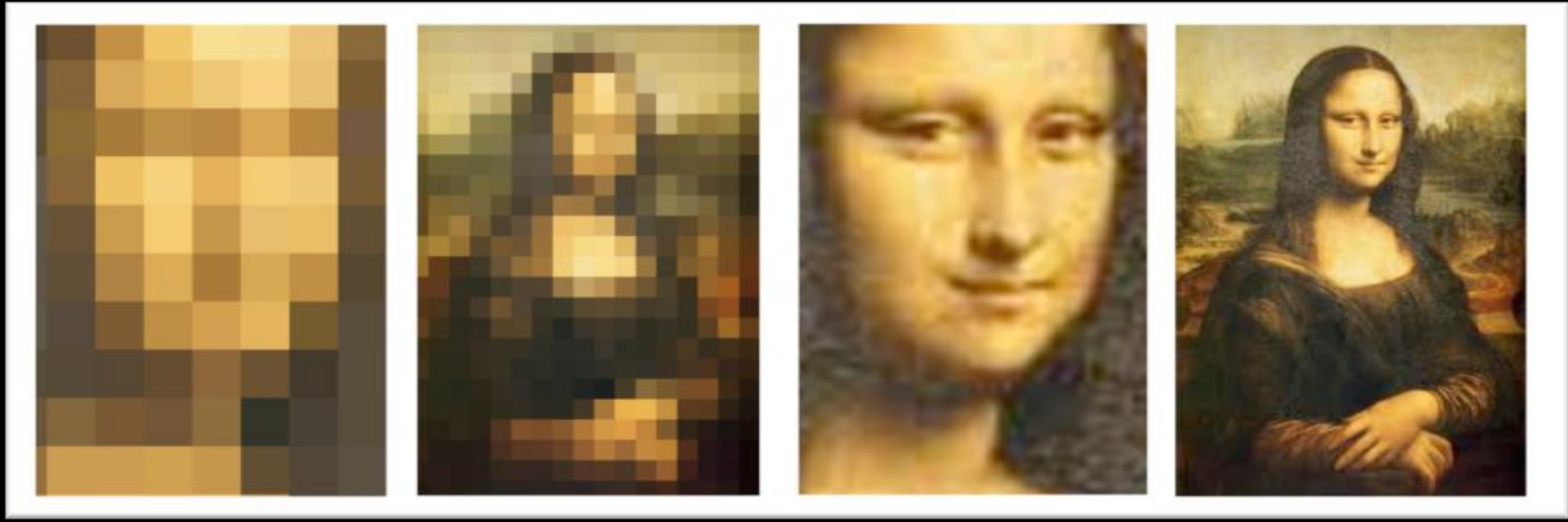
# MULTILAYER TRIGGERS

- Each stage reduces the rate, so later stages have longer latency
- Complexity of algorithms increases at each level
- Dead-time is the sum of the trigger dead-time, summed over the trigger levels, and the readout dead-time

# MULTILAYER TRIGGERS

- Adopted in large experiments
  - More and more complex algorithms are applied on lower and lower data rates
- Efficiency for the desired physics must be kept high **AT ALL LEVELS**, since rejected events are lost for ever

*Level-1* *Level-2* *Level-3* *Analysis*



- Low latency
- Full event rate
- Small event fragment size
- Lower algorithmic complexity
- Access to coarse granularity information

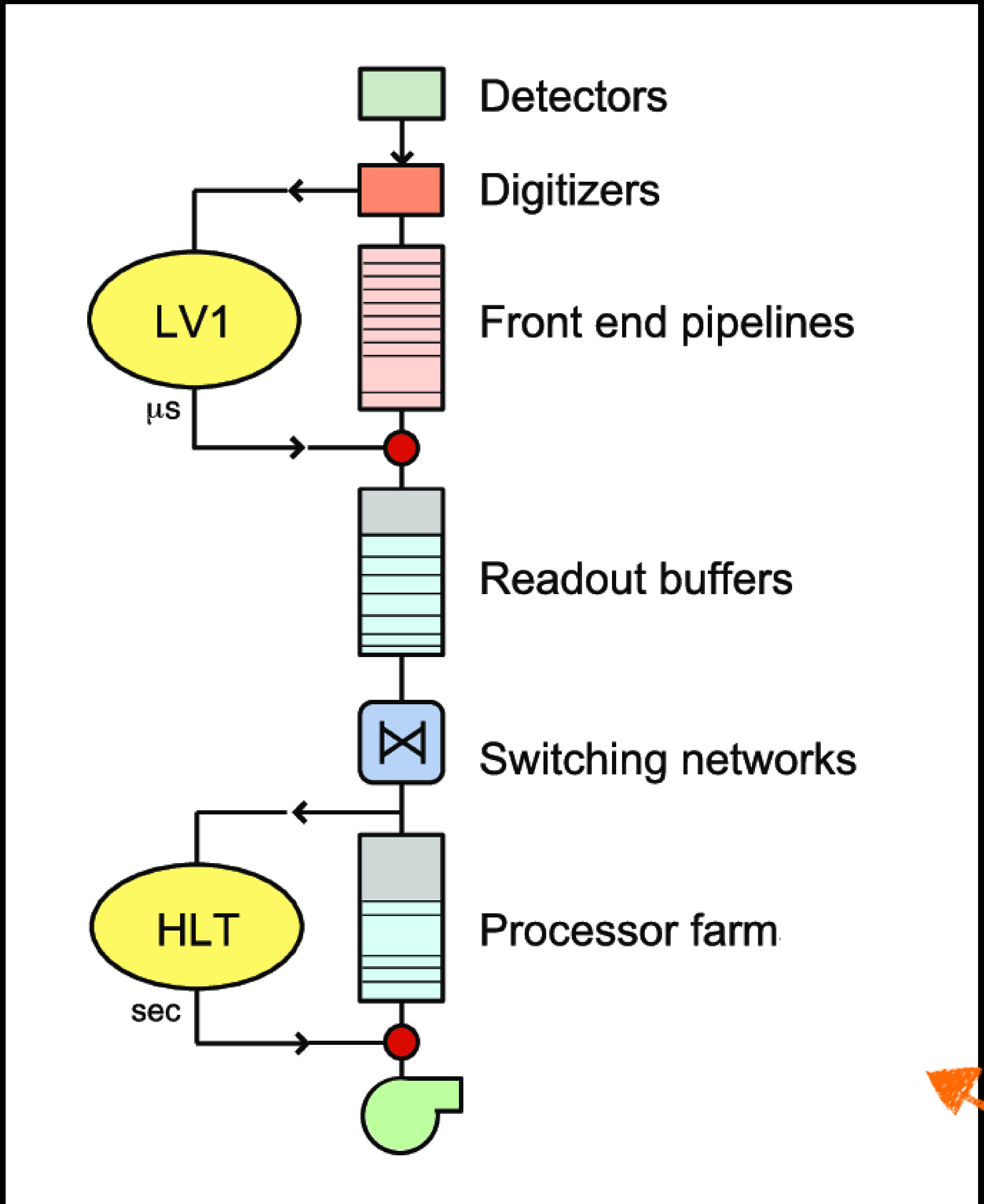
- Longer latency
- Lower event rate
- Larger event fragment size
- Higher algorithmic complexity
- Access to higher granularity information

| LHC experiments @ Run1 | |
|---|---|
| **Experiment** | **Number of Levels (excl. analysis)** |
| **ATLAS** | **3** |
| **CMS** | **2** |
| **LHCB** | **3** |
| **ALICE** | **4** |

# A TRIGGER SYSTEM: MULTILAYER TRIGGERS


If your input rate is low enough

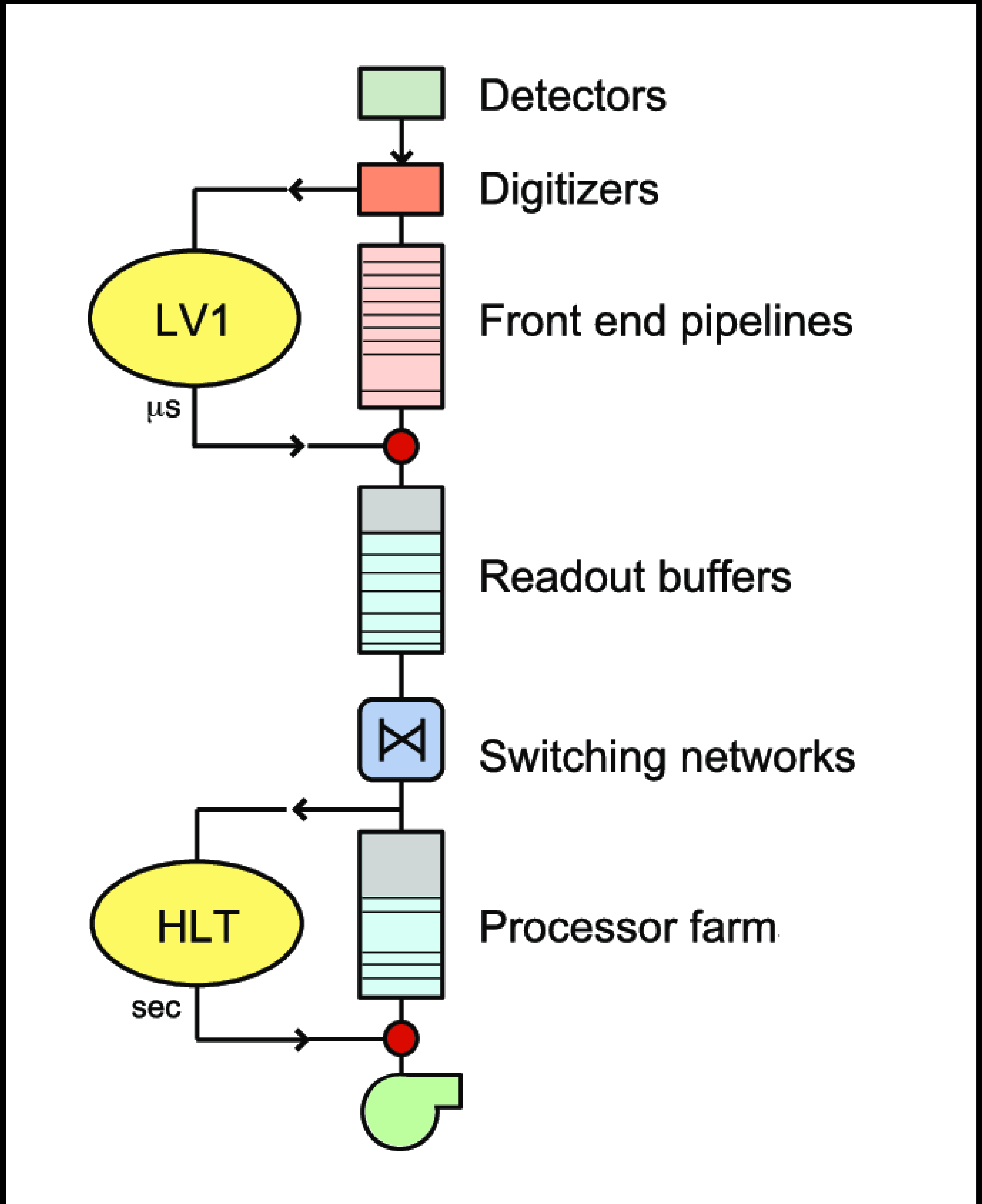
# A TRIGGER SYSTEM: MULTILAYER TRIGGERS

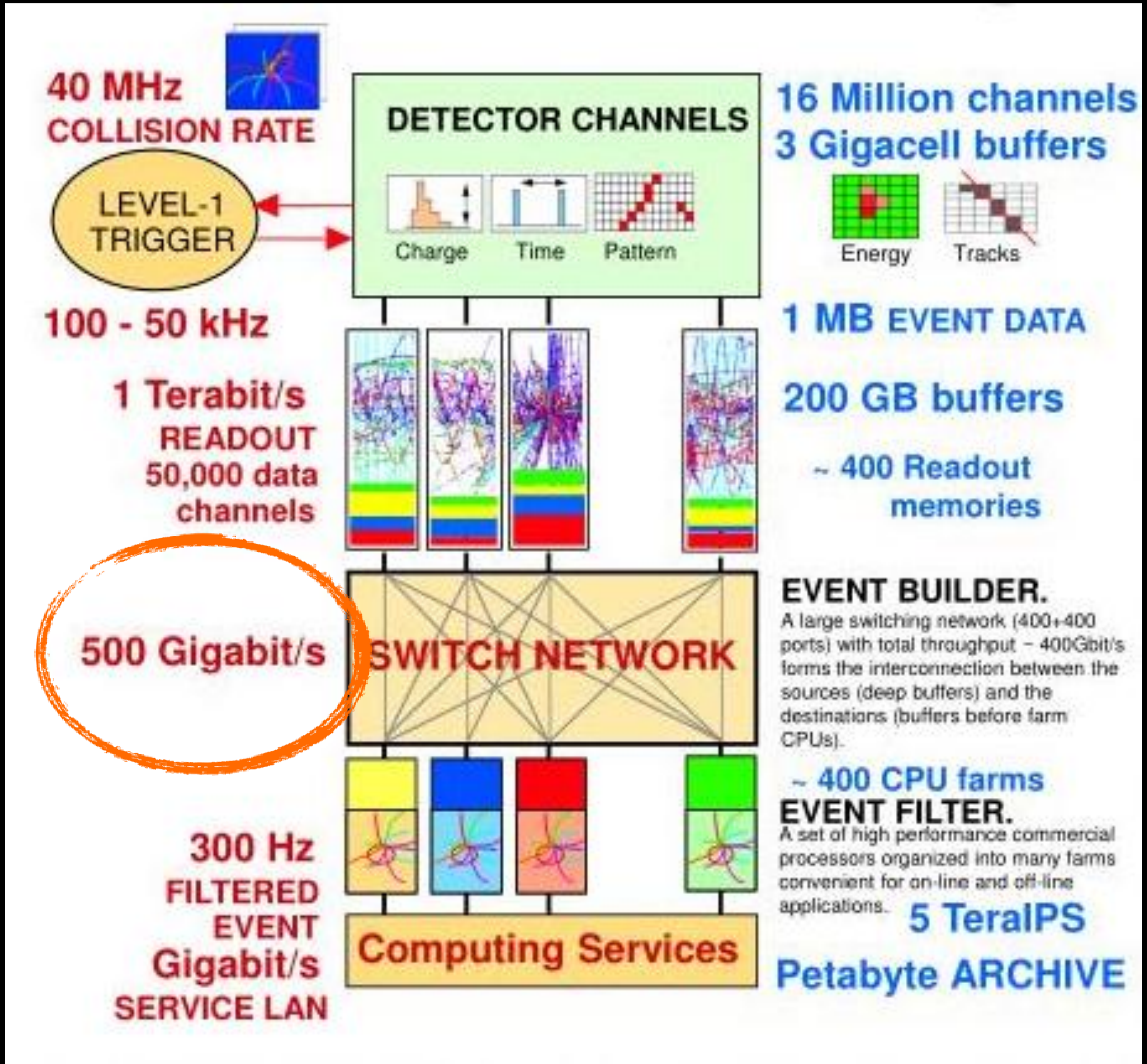



- And this is exactly what the CMS Trigger does

"Standard" figure for the CMS Trigger & DAQ

# OF COURSE, "LOW ENOUGH" IS RELATIVE…



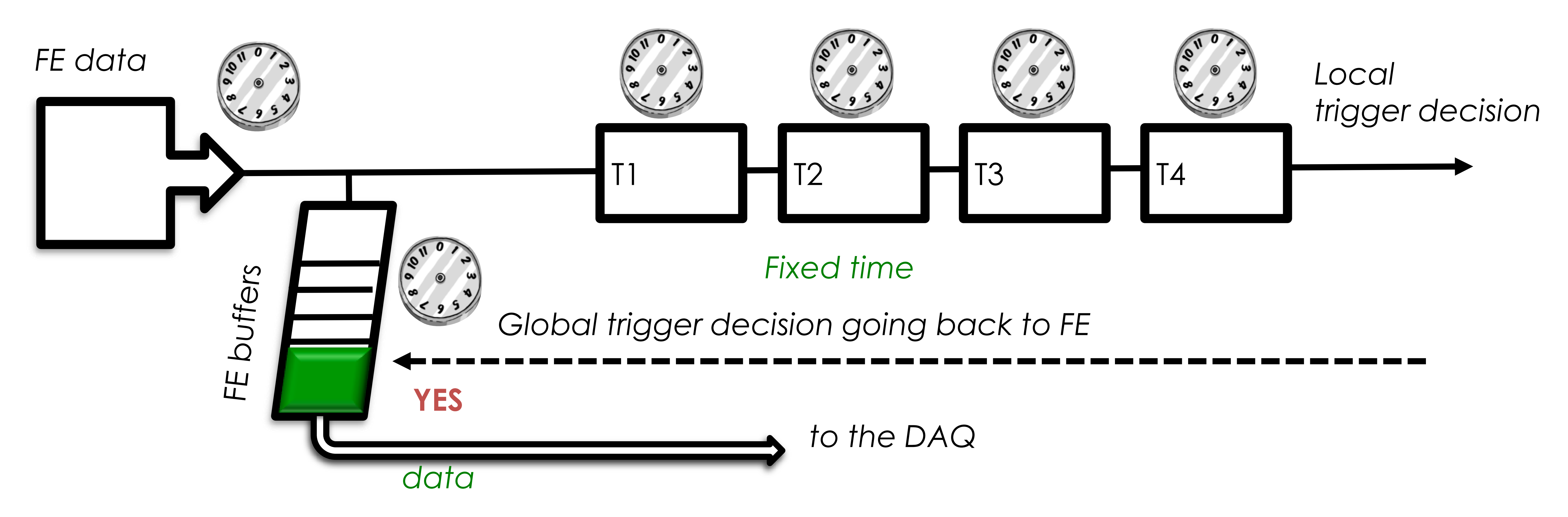



# SYNCHRONOUS OR ASYNCHRONOUS?

- Synchronous: operates phase-locked with master clock
  - Data move in lockstep with the clock through the trigger chain
  - Fixed latency
  - The data, held in storage pipelines, are either sent forward or discarded
  - Used for L1 triggers in collider experiments, exploiting the accelerator bunch crossing clock

✓ **Pro's:** dead-time free (just few clock cycles to protect buffers)

✗ **Con's:** cost (high frequency stable electronics, sometimes needs to be custom made); maintain synchronicity throughout the entire system, complicated alignment procedures if the system is large (software, hardware, human…)

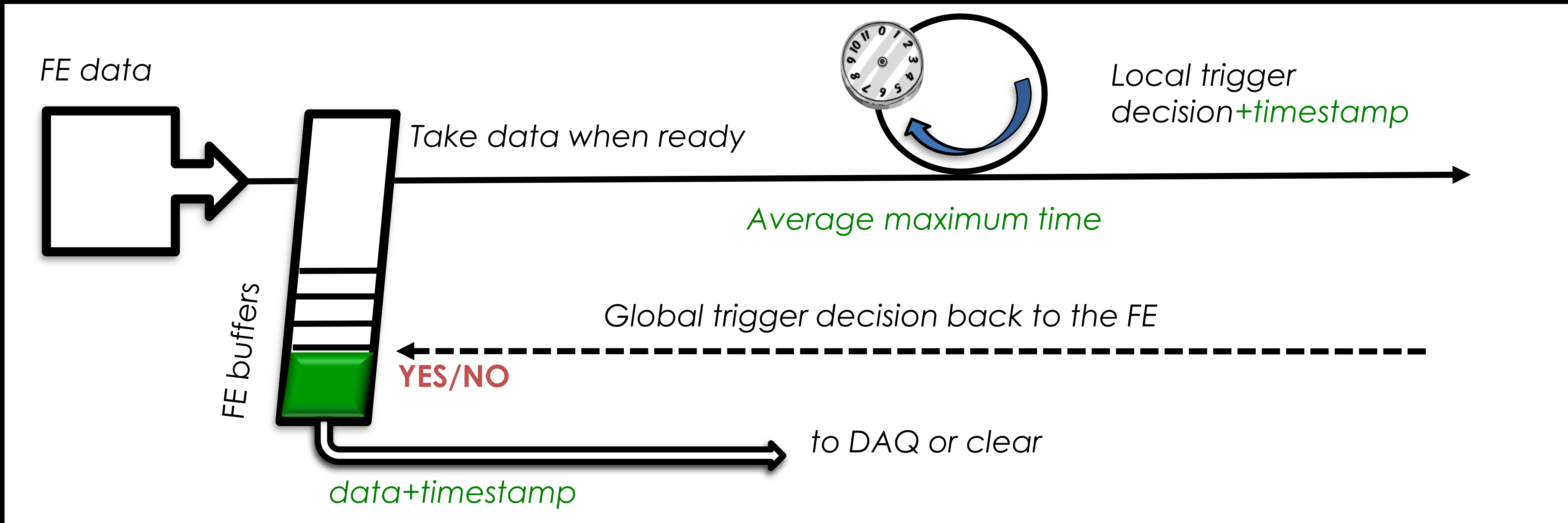


# SYNCHRONOUS OR ASYNCHRONOUS?

- Asynchronous: operations start at given conditions (when data ready or last processing is finished)
  - Used for larger time windows
  - Average latency (with large buffers to absorb fluctuations)
  - If buffer size ≠ dead-time → lost events
  - Used also for "software filters"

✓ **Pro's:** more resilient to data burst; running on conventional CPUs

✗ **Con's:** needs a timing signal synchronised to the FE to latch the data, needs time-marker stored in the data, data transfer protocol is more complex)

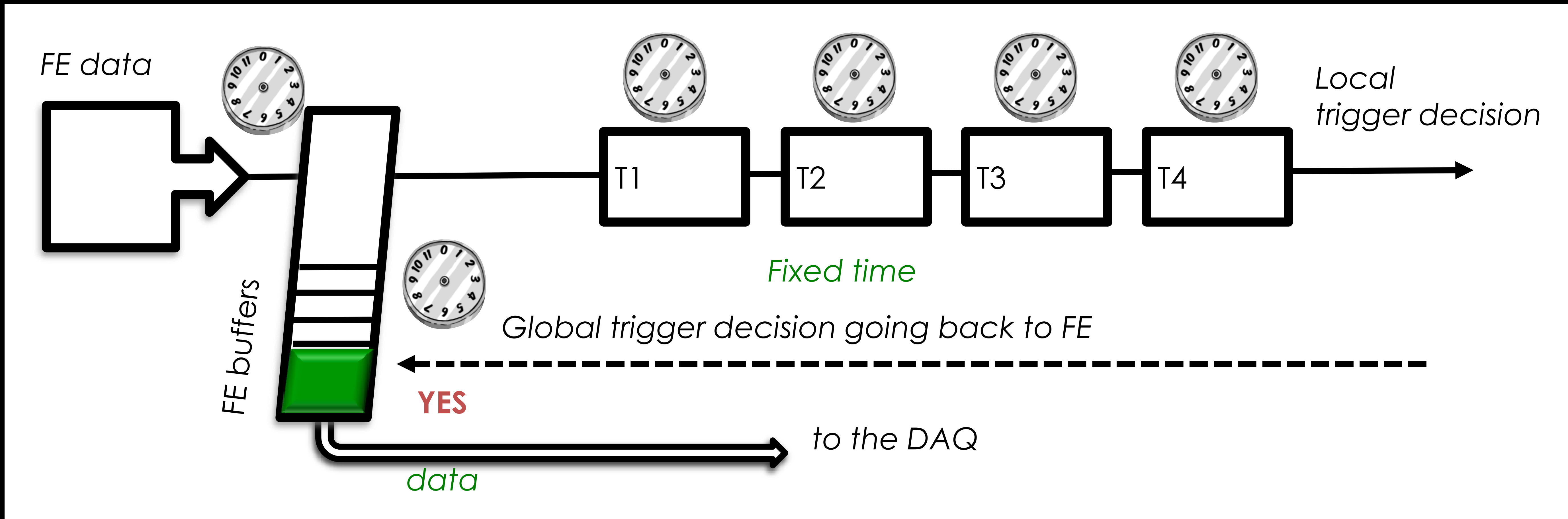


# SYNCHRONOUS OR ASYNCHRONOUS? WHY NOT BOTH?

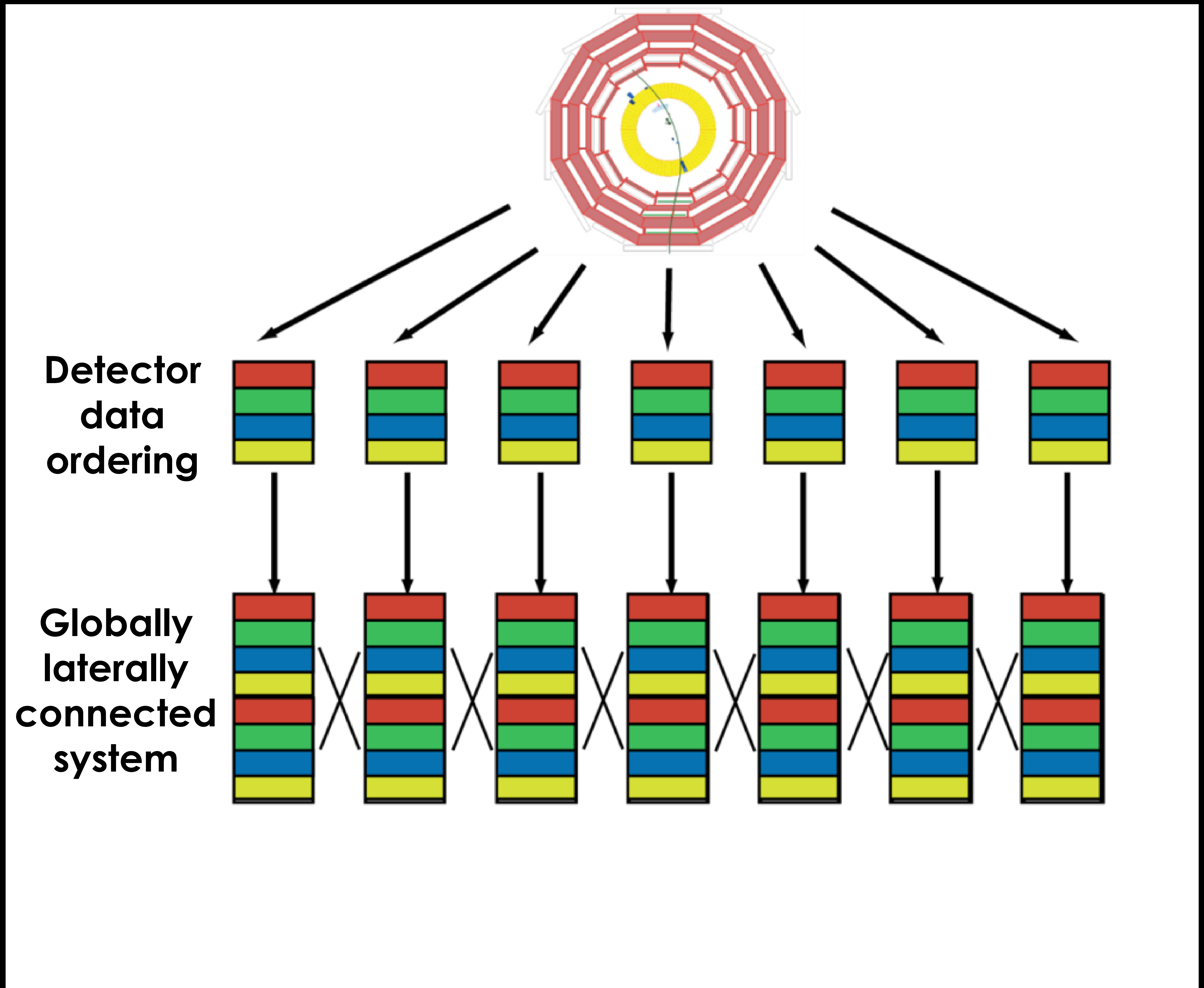
- Pseudo-synchronous: operates **locally** phase-locked
  - Data move in lockstep through the trigger chain from a set of local clocks
  - Buffering required whenever you move between clocks
  - Clocks run slightly faster than source data to prevent overflow
  - Realignment to global clock only after the final trigger stage
  - Fixed latency

✓ **Pro's:** dead-time free (just few clock cycles to protect buffers), no need for expensive globally-distributed clock, simpler alignment procedure

✗ **Con's:** must propagate timing info with data, buffering required to handle clock-domain change



# CONVENTIONAL ARCHITECTURE

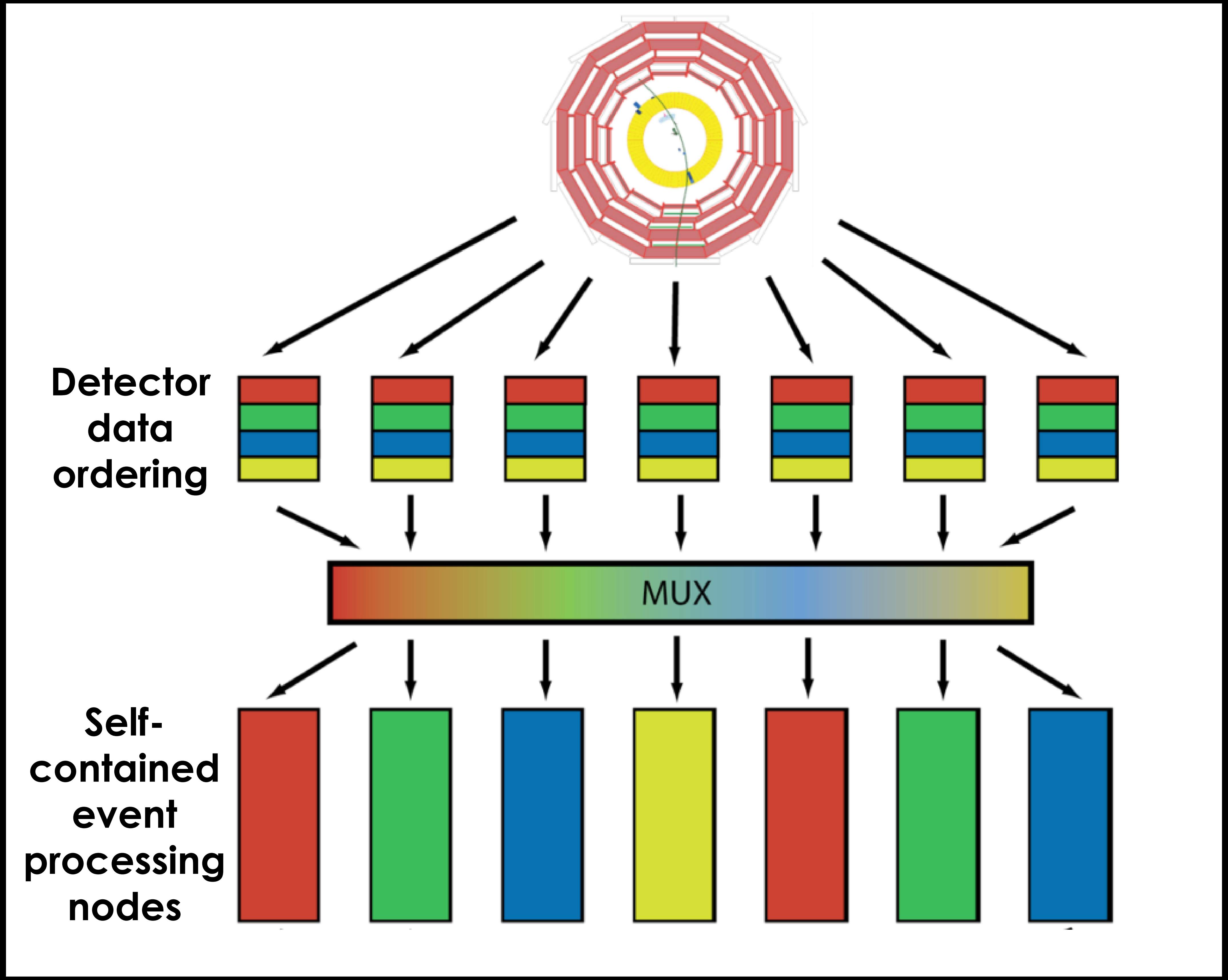



Many, many details on time-multiplexing and conventional architectures in sections 1-3 of https://cds.cern.ch/record/1421552/files/IN2011_022.pdf (although please note that the systems proposed in section 4-9 are very outdated and should be ignored)

- Each subsystem is regionally segmented
- Each region must talk to its neighbour
  - This is the root cause of requiring specialized boards for a given task!
- Each region of each processing layer compresses, suppresses, summarizes or otherwise reduces its data and passes it on to the next level which is less regionally segmented

# TIME-MULTIPLEXED ARCHITECTURE

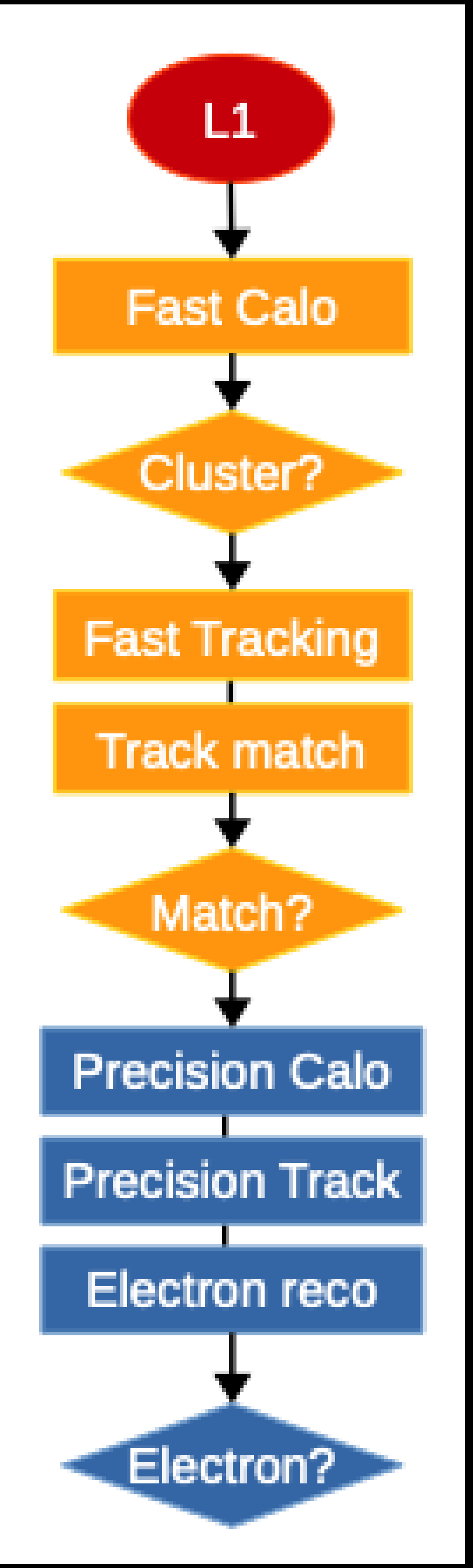
- Buffer data and stream it out optimized for processing
- Spread processing over time
  - Stream-processing rather than combinatorial-logic
  - Maximise reuse of logic resources
  - Easiest for FPGA design tools to route and meet timing
- Costs you latency, bought back by more efficient processing




Many, many details on time-multiplexing and conventional architectures in sections 1-3 of https://cds.cern.ch/record/1421552/files/IN2011_022.pdf (although please note that the systems proposed in section 4-9 are very outdated and should be ignored)

# HIGH LEVEL TRIGGER DESIGN PRINCIPLES

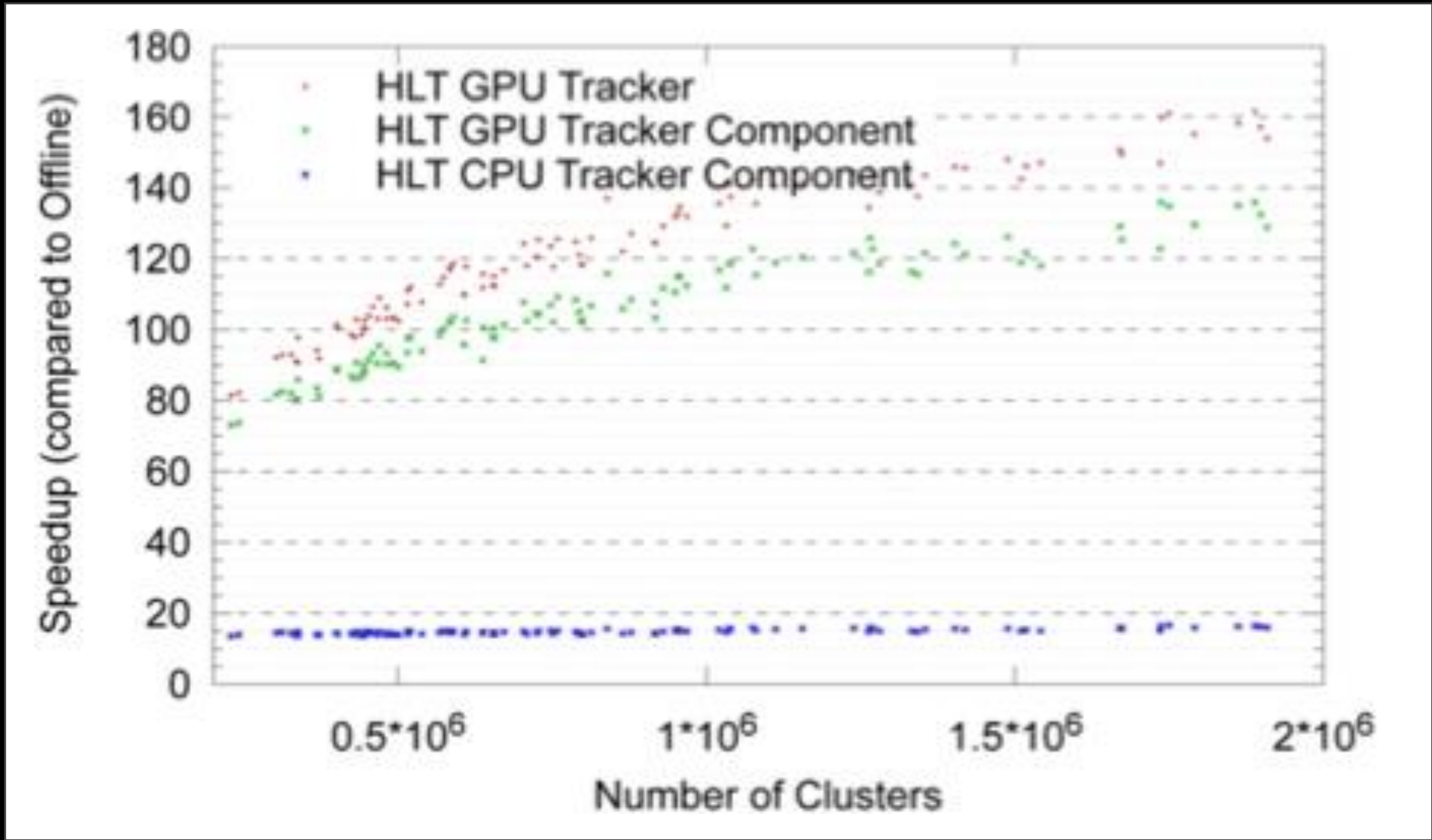
- Offline reconstruction too slow to be used directly
  - Takes >10s per event
  - HLT usually needs << 1s
- Instead, step-wise processing with early rejection
  - Stop processing as soon as one step fails
  - Event accepted if any of the trigger passes
  - Add a time-out to kill the Poisson tail!
- Fast reconstruction & L1-guided regional reconstruction first
- Precision reconstruction as full detector data becomes available



# HIGH LEVEL TRIGGER DESIGN PRINCIPLES

- Early rejection reduce data and resources (CPU, memory, etc.)
- Event-level parallelism
  - Process more events in parallel
  - Multi-processing or/and multi-threading
- Algorithm-level parallelism
  - **GPUs** effective whenever large amount of data can be processed concurrently (although bandwidth can be a limiting factor)

- Algorithms developed and optimized offline
- Common HLT-reconstruction software framework **reduces maintenance and increases reliability**



# EXAMPLE: CMS HLT

- Approximately 38,000 cores
  - An equal mix of Haswell, Broadwell and Skylake
- Multithreading allows the cores to share non-event data
  - Reduced memory footprint → can process more events: ~20% higher performance
  - ATLAS currently doesn't have this: a race to implement this in ATHENA for Run 3
- Upgrades to add a GPU in every filter farm node is ruled out by cost and power
  - More likely a dedicated server sub-farm which does heavy tasks on demand
  - FPGAs acceleration also a (possibly better) option

# CONCLUSION

- Triggers are not new
  - but they are constantly evolving as the accelerators and detectors do
- FPGAs are the weapon of choice for the Level-1 trigger
  - but they are not a magic bullet
- The design of how you structure the transfer of data around your system is the most important decision you will make
- Heterogeneous computing farms look likely to feature at HL-LHC
  - but it is a brave new world!

# CONCLUSION

- Triggers are not new
  - but they are constantly evolving as the accelerators and detectors do
- FPGAs are
  - but they

Oh, and be very suspicious if your supervisor plies you with strong coffee and gets you to look for scintillation light

- The design of how you structure the transfer of data around your system is the most important decision you will make
- Heterogeneous computing farms look likely to feature at HL-LHC
  - but it is a brave new world!

# Spares: Introduction to FPGAs

# And this is not a new point

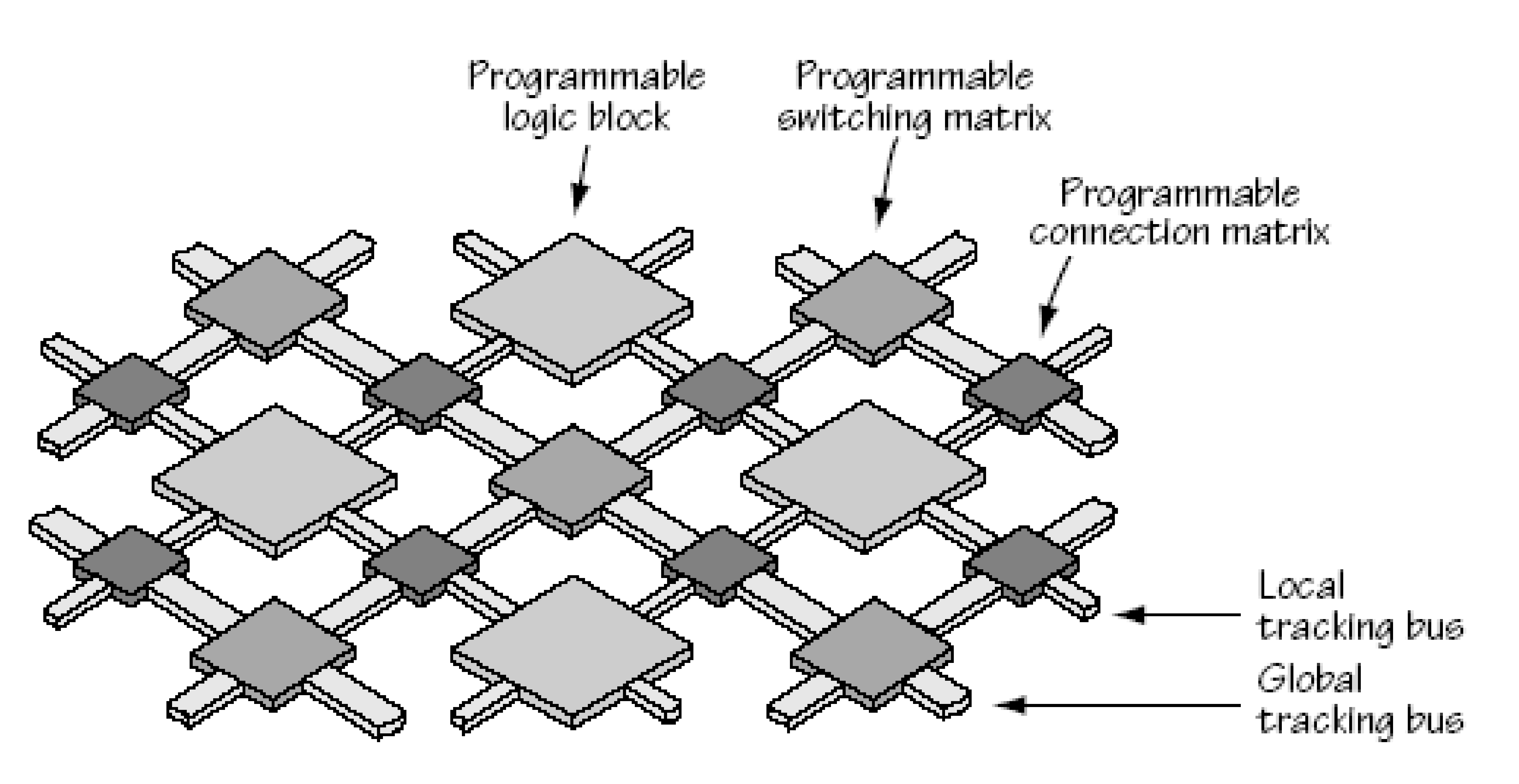
"The parallel approach to computing does require that some original thinking be done about numerical analysis and data management in order to secure efficient use.

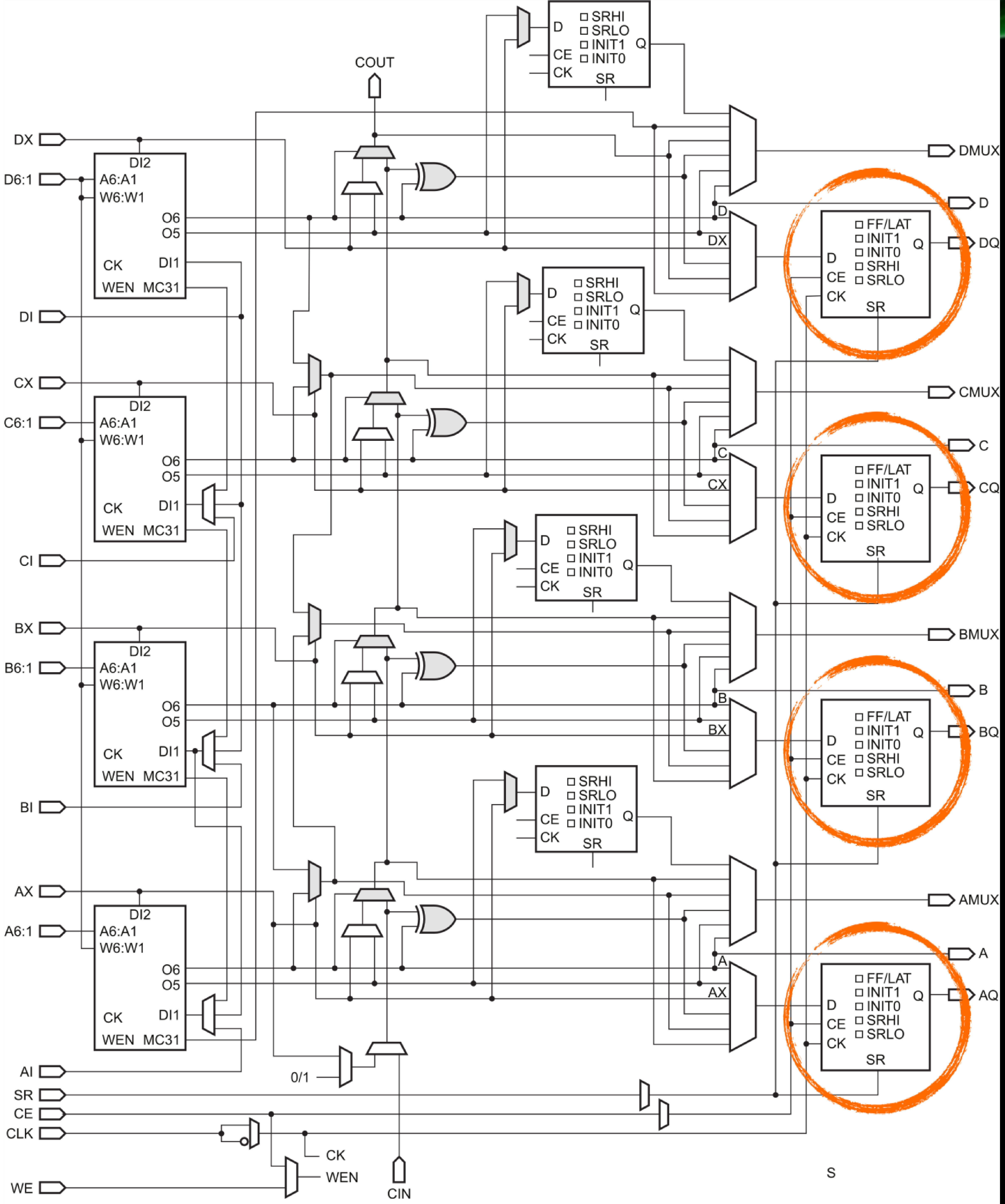
In an environment which has represented the absence of the need to think as the highest virtue, this is a decided disadvantage"

Daniel Slotnick, 1967

# FIELD PROGRAMMABLE GATE ARRAYS

- Programmable Logic Blocks
- Massive Fabric of Programmable Interconnects

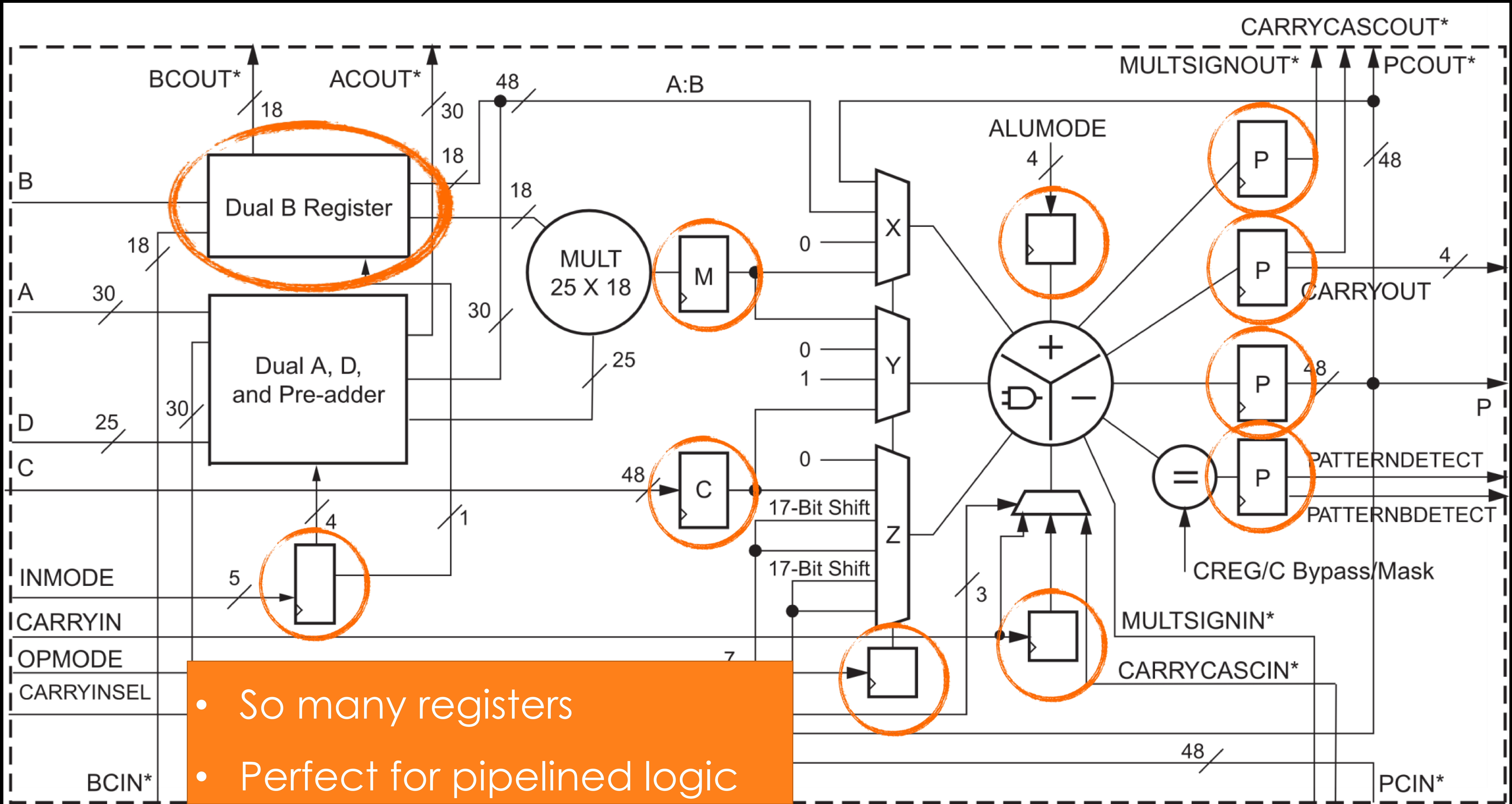





# COMBINATORIAL LOGIC BLOCK

- Registers on the output of every cell
- Perfect for pipelined logic

# INTEGRATED DIGITAL SIGNAL PROCESSING




- So many registers
- Perfect for pipelined logic

# BIGGEST XILINX "ULTRASCALE+" DEVICES

- Upwards of 9million logic cells
  - All clocked at up to 500MHz
  - Up to $O(10^{15})$ operations per second
- Upwards of 12000 DSPs
- All pipelined
- Fully programmable
- And we have a winner!

| Device Name | VU9P | VU11P | VU13P |
|---|---|---|---|
| Effective LEs[1] (K) | 2,485 | 2,575 | 3,435 |
| Logic Cells (K) | 2,069 | 2,147 | 2,863 |
| CLB Flip-Flops (K) | 2,364 | 2,454 | 3,272 |
| CLB LUTs (K) | 1,182 | 1,227 | 1,636 |
| Max. Distributed RAM (Mb) | 36.1 | 34.8 | 46.4 |
| Total Block RAM (Mb) | 75.9 | 70.9 | 94.5 |
| UltraRAM (Mb) | 270.0 | 324.0 | 432.0 |
| DSP Slices | 6,840 | 8,928 | 11,904 |
| PCIe® Gen3 x16 / Gen4 x8 | 6 | 3 | 4 |
| 150G Interlaken | 9 | 9 | 12 |
| 100G Ethernet w/ RS-FEC | 9 | 6 | 8 |
| Max. Single-Ended HP I/Os | 832 | 624 | 832 |
| GTY 32.75Gb/s Transceivers | 120 | 96 | 128 |

# BIGGEST XILINX "ULTRASCALE+" DEVICES

- Upwards of 9million logic cells
  - All clocked at up to 500MHz
  - Up to $O(10^{15})$ operations per second
- Upwards of 12000 DSPs
- All pipelined
- Fully programmable
- And we have a winner!
- So what is the catch?

| Device Name | VU9P | VU11P | VU13P |
|---|---|---|---|
| Effective LEs[(1)] (K) | 2,485 | 2,575 | 3,435 |
| Logic Cells (K) | 2,069 | 2,147 | 2,863 |
| CLB Flip-Flops (K) | 2,364 | 2,454 | 3,272 |
| CLB LUTs (K) | 1,182 | 1,227 | 1,636 |
| Max. Distributed RAM (Mb) | 36.1 | 34.8 | 46.4 |
| Total Block RAM (Mb) | 75.9 | 70.9 | 94.5 |
| UltraRAM (Mb) | 270.0 | 324.0 | 432.0 |
| DSP Slices | 6,840 | 8,928 | 11,904 |
| PCIe® Gen3 x16 / Gen4 x8 | 6 | 3 | 4 |
| 150G Interlaken | 9 | 9 | 12 |
| 100G Ethernet w/ RS-FEC | 9 | 6 | 8 |
| Max. Single-Ended HP I/Os | 832 | 624 | 832 |
| GTY 32.75Gb/s Transceivers | 120 | 96 | 128 |

# BIGGEST XILINX "ULTRASCALE+" DEVICES

- Upwards of 9m
  - All clocked
  - Up to O(10
- Upwards of 12
- All pipelined
- Fully programm
- And we have a winner!
- So what is the catch?

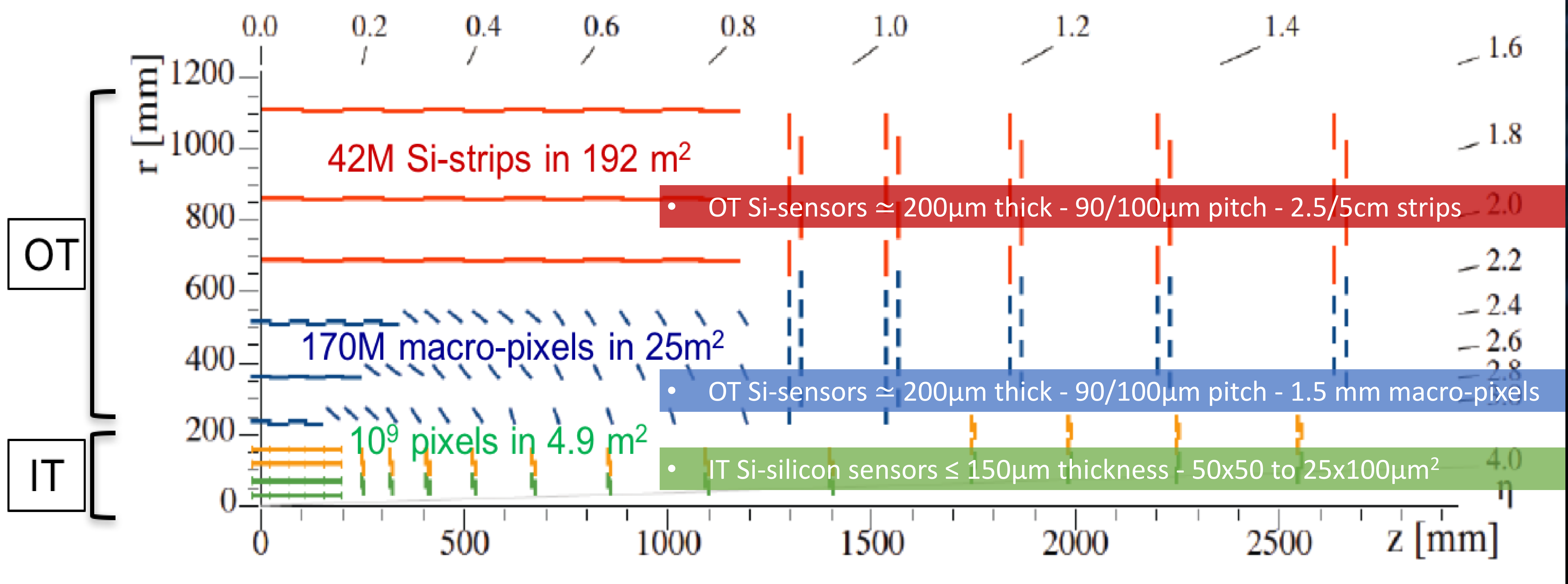
- Incredibly hard to program efficiently
  - Thinking in a parallel, pipelined-fashion is exceptionally difficult
  - A handful of real experts in CMS
- Efficient use depends on efficiently structured data
- The chip is just the start – needs to be attached to something
- You are also responsible for the infrastructure

| VU11P | VU13P |
|---|---|
| 2,575 | 3,435 |
| 2,147 | 2,863 |
| 2,454 | 3,272 |
| 1,227 | 1,636 |
| 34.8 | 46.4 |
| 70.9 | 94.5 |
| 324.0 | 432.0 |
| 8,928 | 11,904 |
| 3 | 4 |
| 9 | 12 |
| 6 | 8 |
| 624 | 832 |
| 96 | 128 |

# Spares: Phase-2

# Tracker

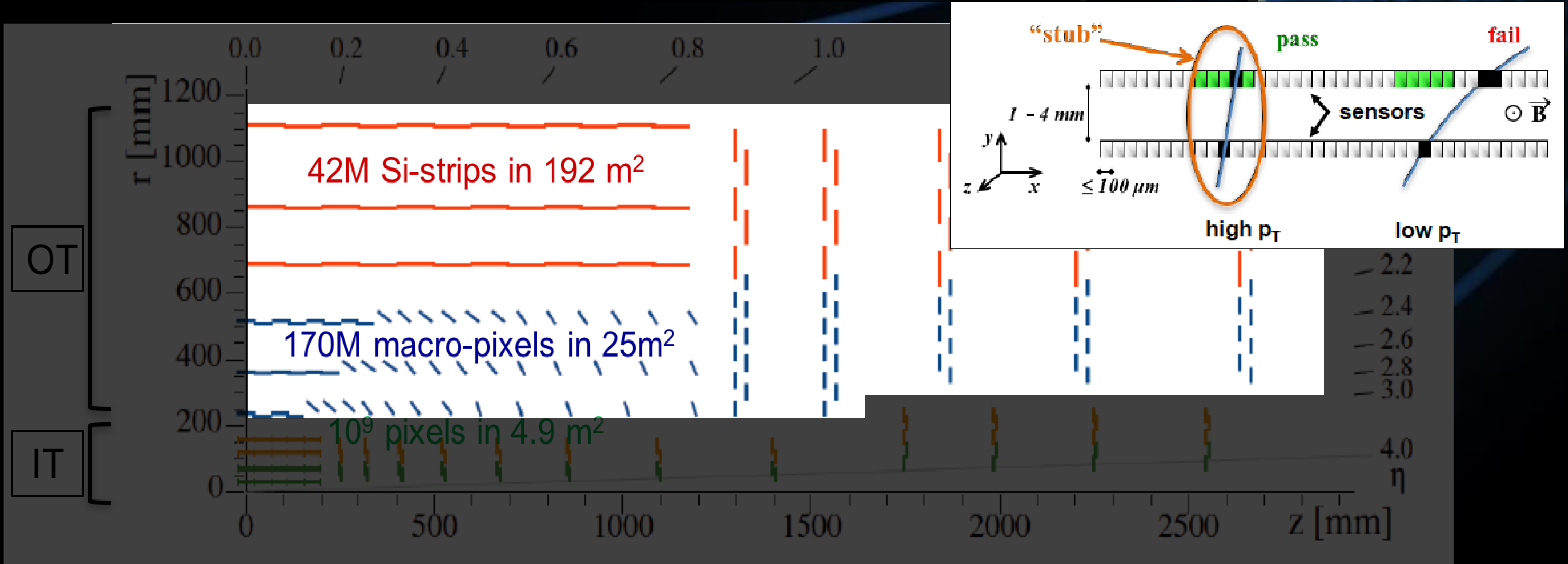
- Inner Tracker (pixel) design to extend coverage to η ≃ 3.8



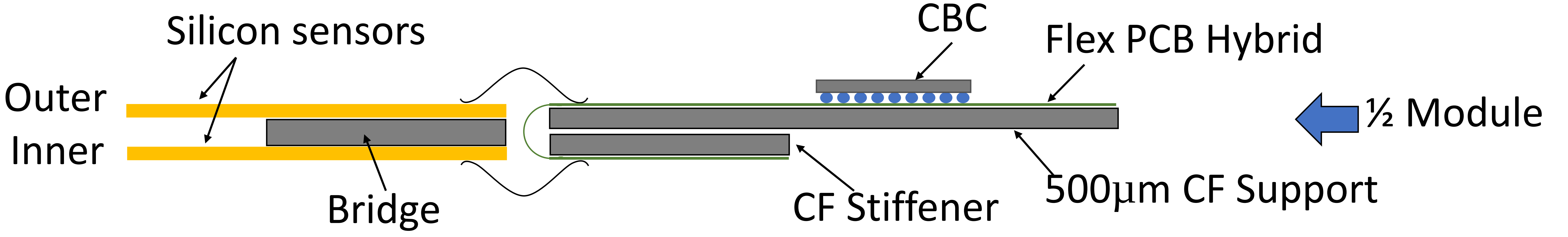

- OT Si-sensors ≃ 200μm thick - 90/100μm pitch - 2.5/5cm strips
- OT Si-sensors ≃ 200μm thick - 90/100μm pitch - 1.5 mm macro-pixels
- IT Si-silicon sensors ≤ 150μm thickness - 50x50 to 25x100μm²

# Tracker

- Inner Tracker (pixel) design to extend coverage to η ≃ 3.8
- Outer Tracker design driven by ability to provide tracks at 40 MHz to L1-trigger



# Outer tracker 2S modules

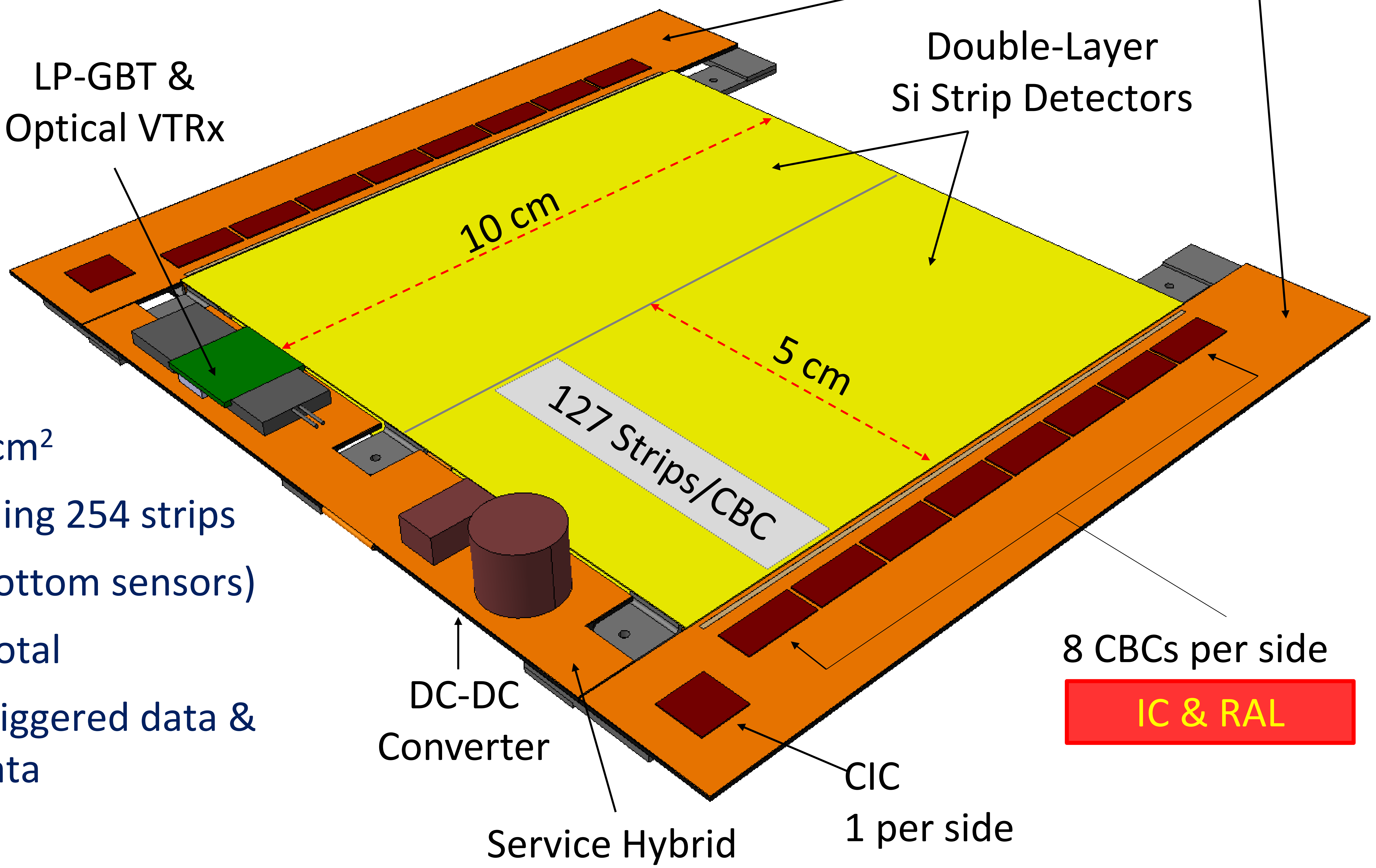



- 2S Modules: Two-strip double-layers
- ~10k modules
- 42M channels

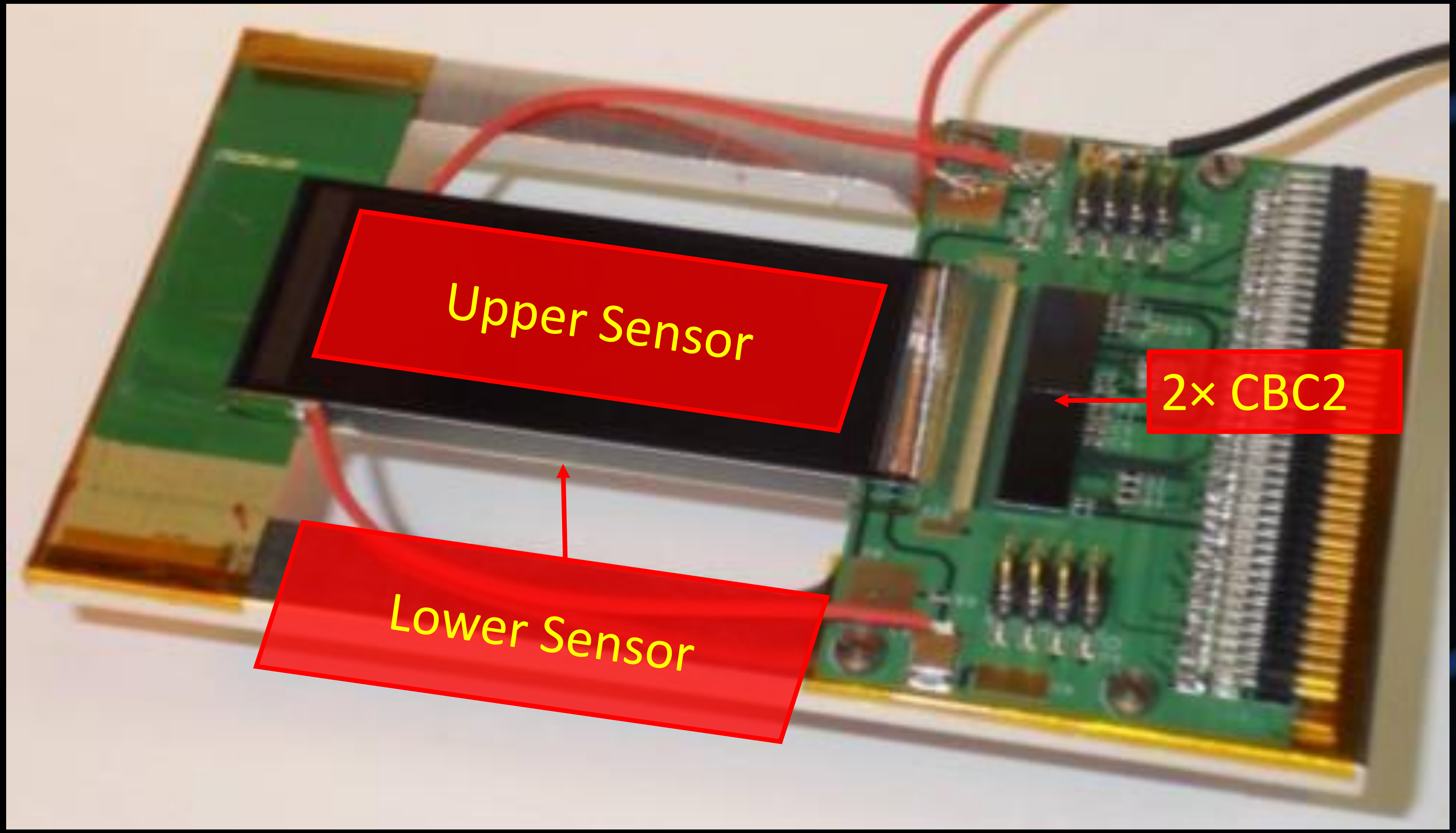
~5,000 modules @ 10Gb/s
+ ~10,000 modules @ 5Gb/s
= 100Tb/s = 395 EB/yr

**Each 2S Module:**

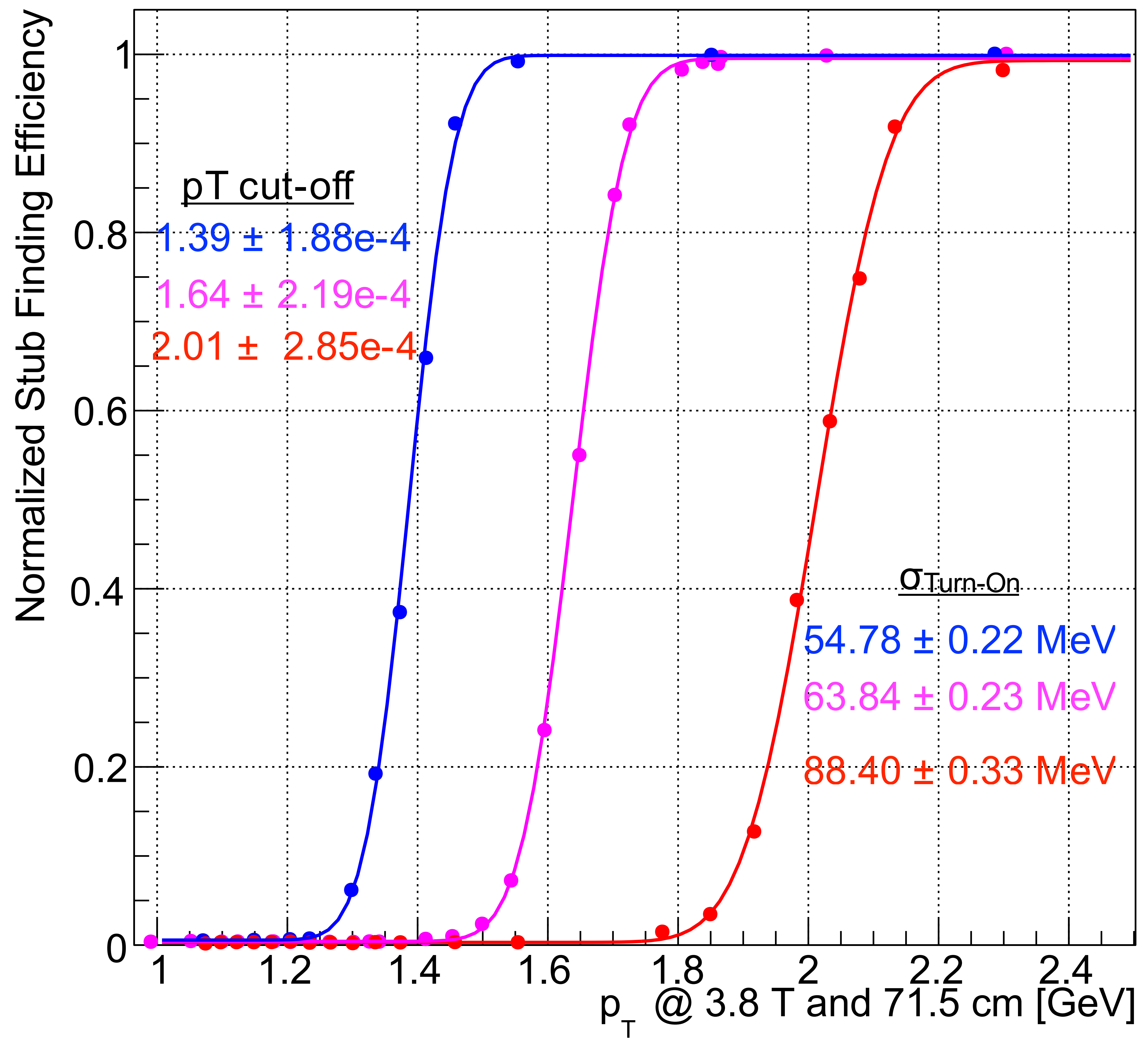
- Sensor Area ~100 cm$^2$
- 16 CBCs, each reading 254 strips (127 from top & bottom sensors)
- 4064 Channels in total
- Readout both L1 triggered data & Primitive trigger data



# Outer tracker 2S modules: Do they work?

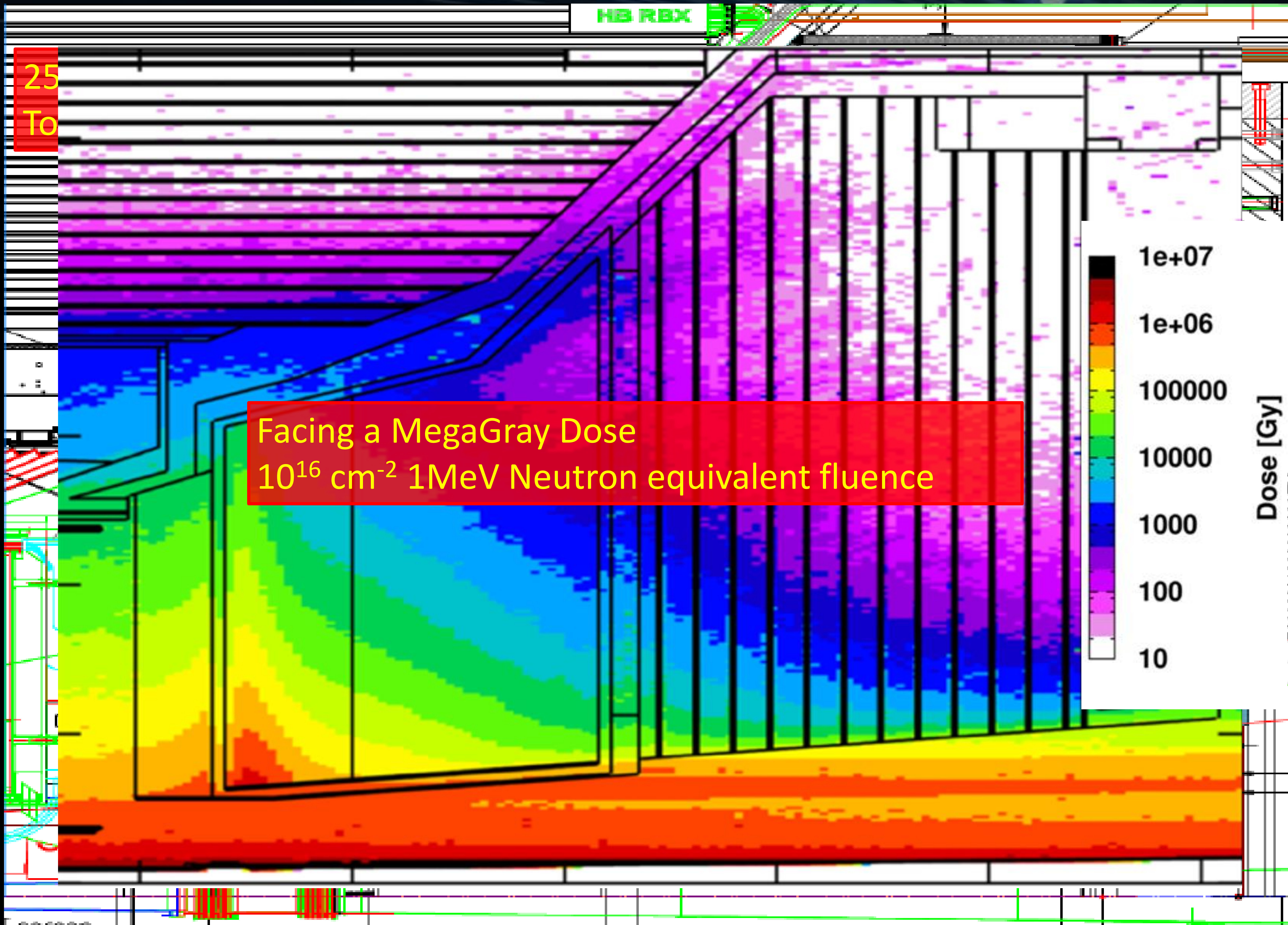



Stub turn-on curve for 2CBC mini-module at FNAL test-beam



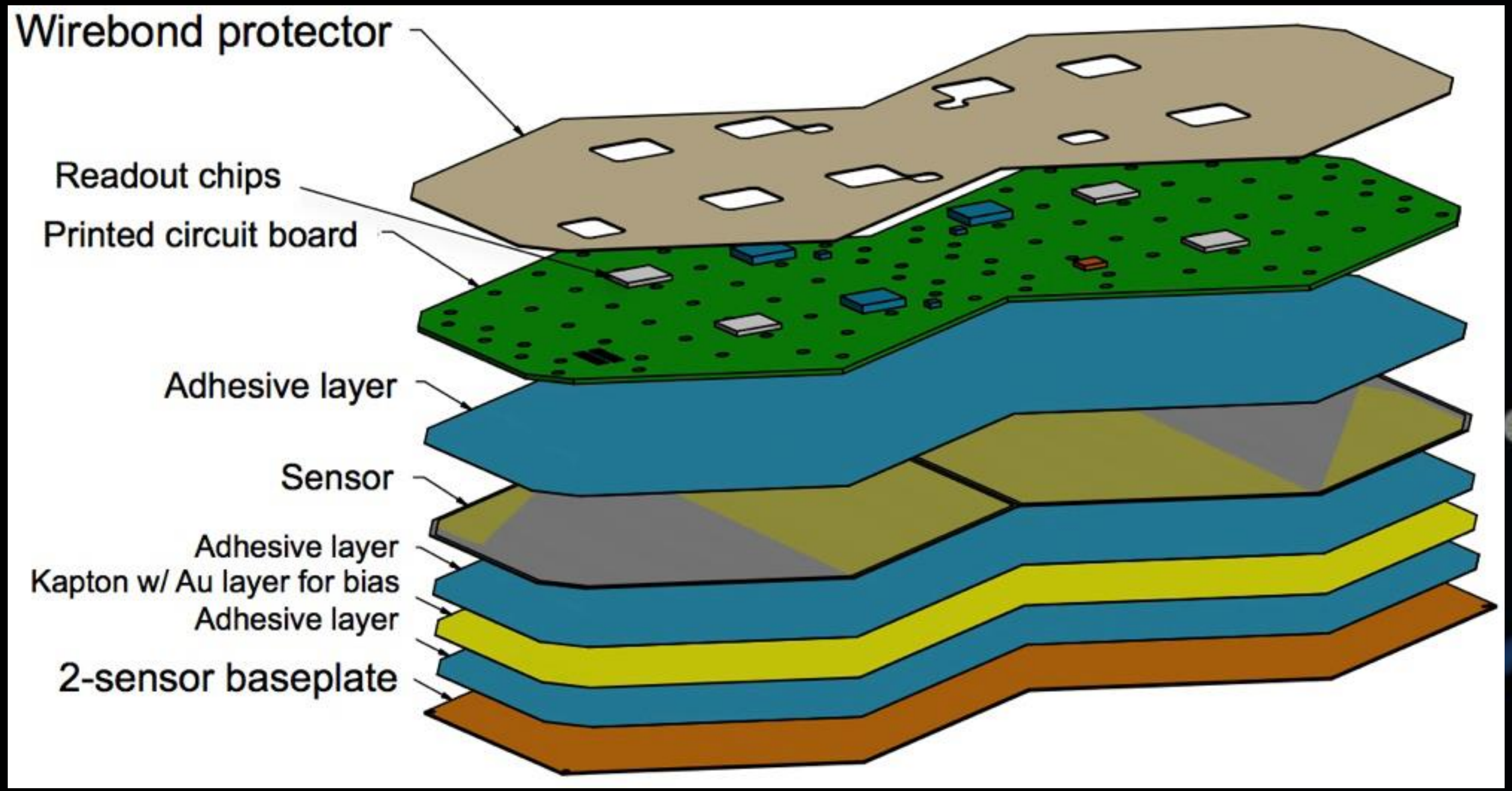
# Calorimeter Endcap design

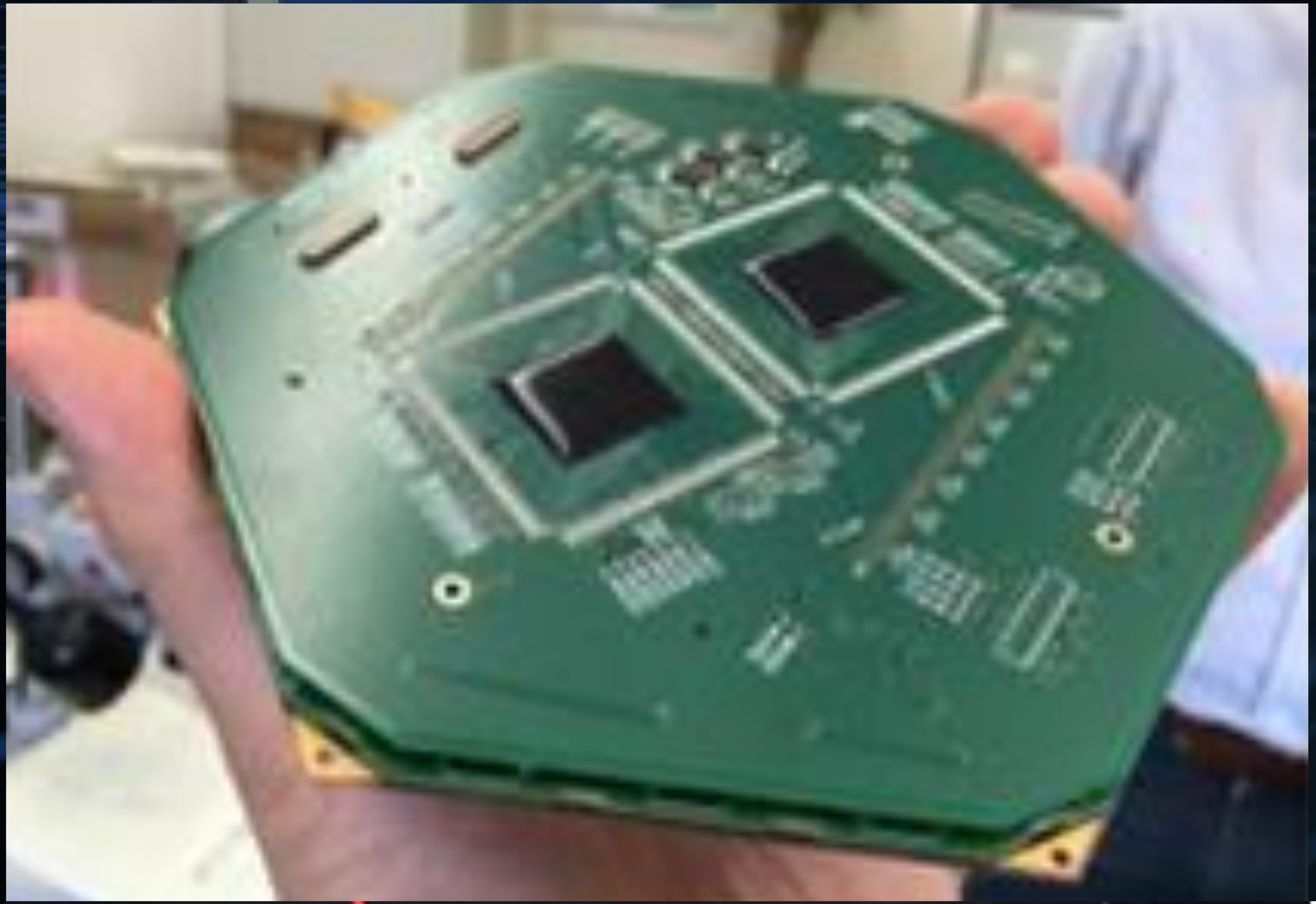
- 3D shower topology and time resolution of ~30ps
- Electromagnetic Endcap (EE)
  - 28 layers of Silicon sensors in W/Pb absorber (25 $X_0$, 1.7λ)
- Hadronic Endcap (EH)
  - 24 layers: 8 silicon + 16 silicon/scint. tiles at high/low η in stainless steel absorber (9λ)

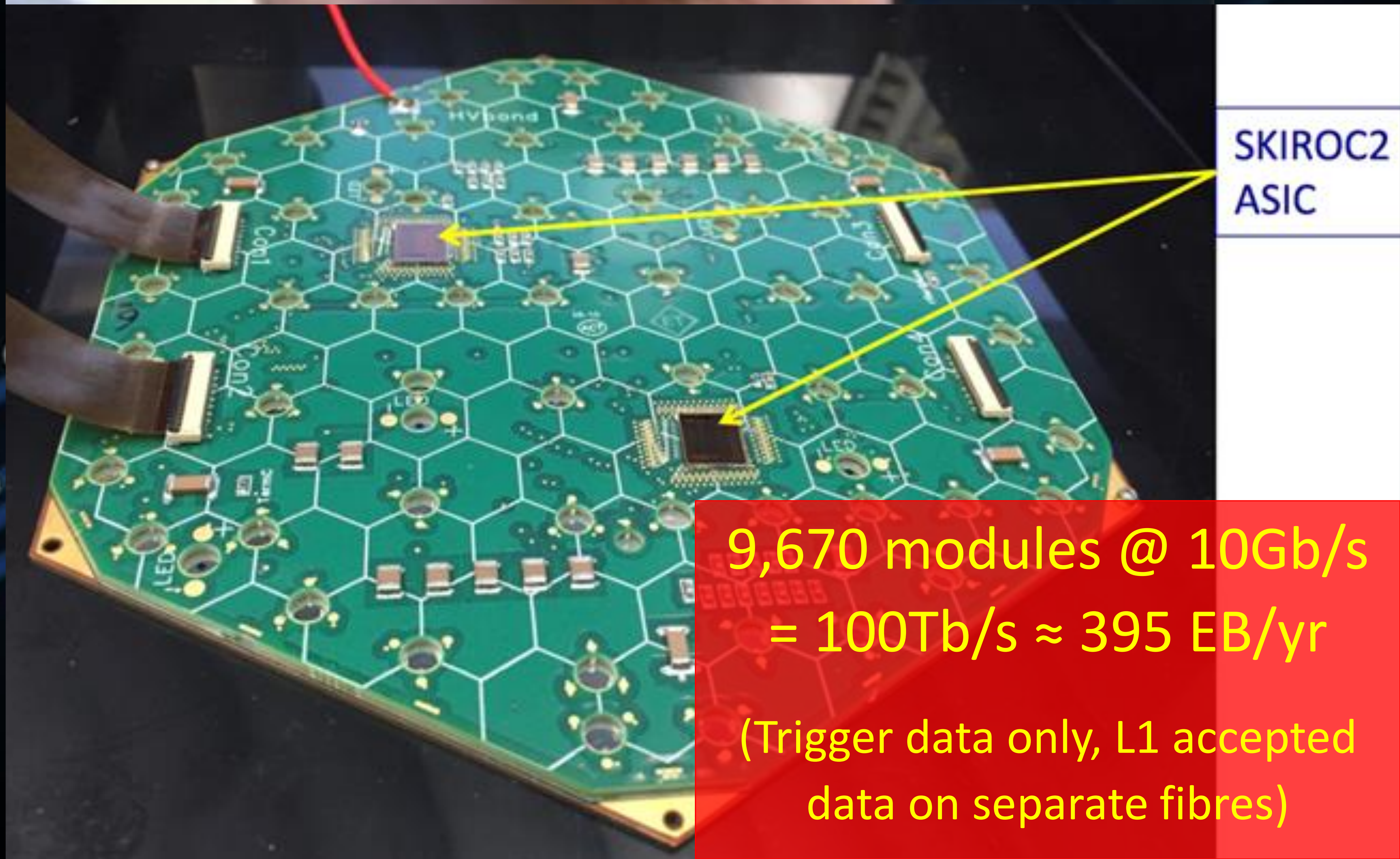


# Calorimeter Endcap modules

- 593 m2 of silicon
- 6M ch, 0.5 or 1 cm2 cell-size
- 21,660 modules (8” or 2x6” sensors)
- 92,000 front-end ASICS





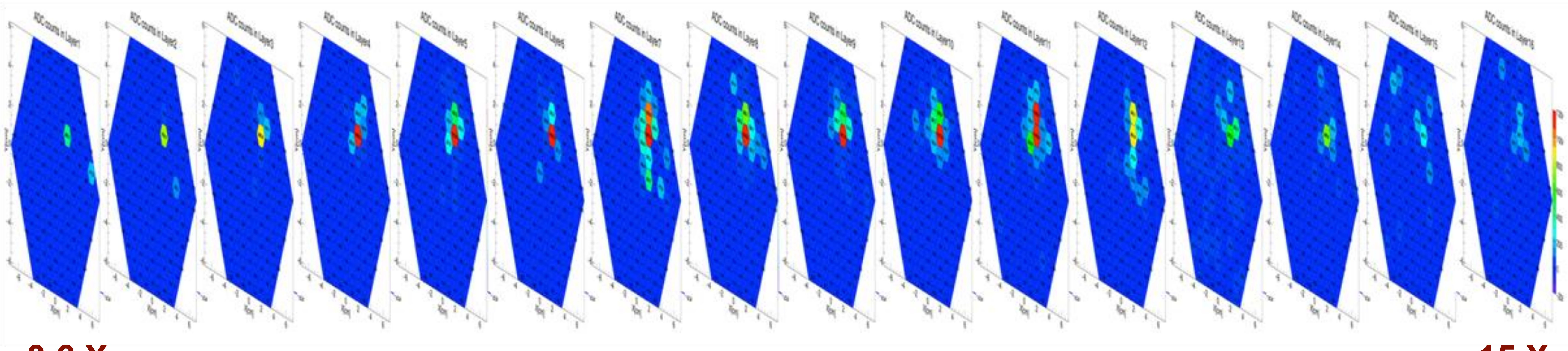

9,670 modules @ 10Gb/s = 100Tb/s ≈ 395 EB/yr

(Trigger data only, L1 accepted data on separate fibres)

# Calorimeter Endcap modules: Do they work?

**Fermilab**: **32 GeV electrons** passing through **15 $X_0$**.

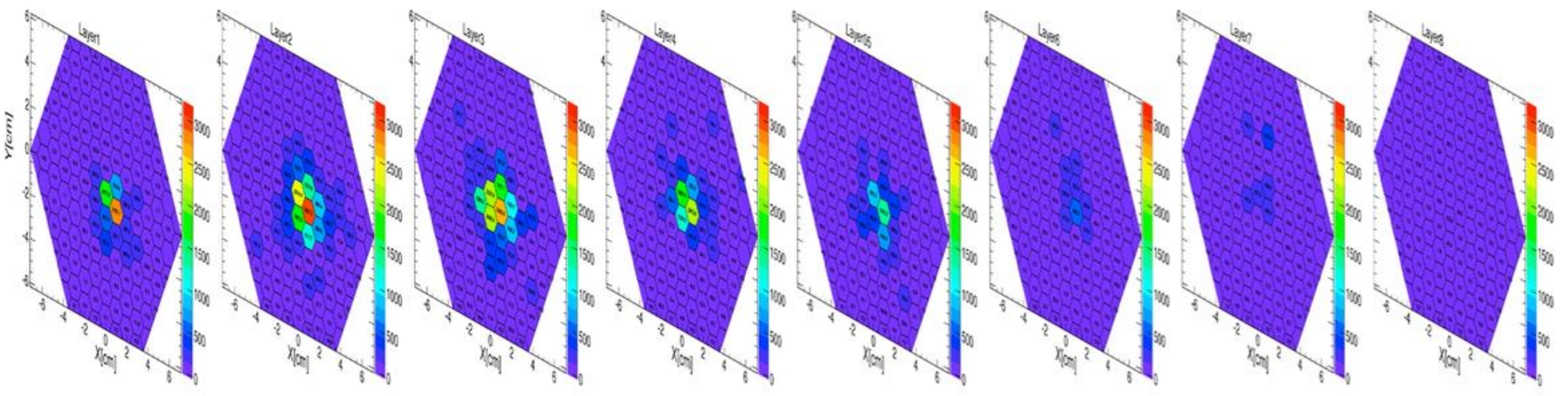
$e^-$
**32 GeV**
→



**0.6 $X_0$** **15 $X_0$**

**CERN**: **250 GeV electrons** passing through **27 $X_0$**.

$e^-$
**250 GeV**
→



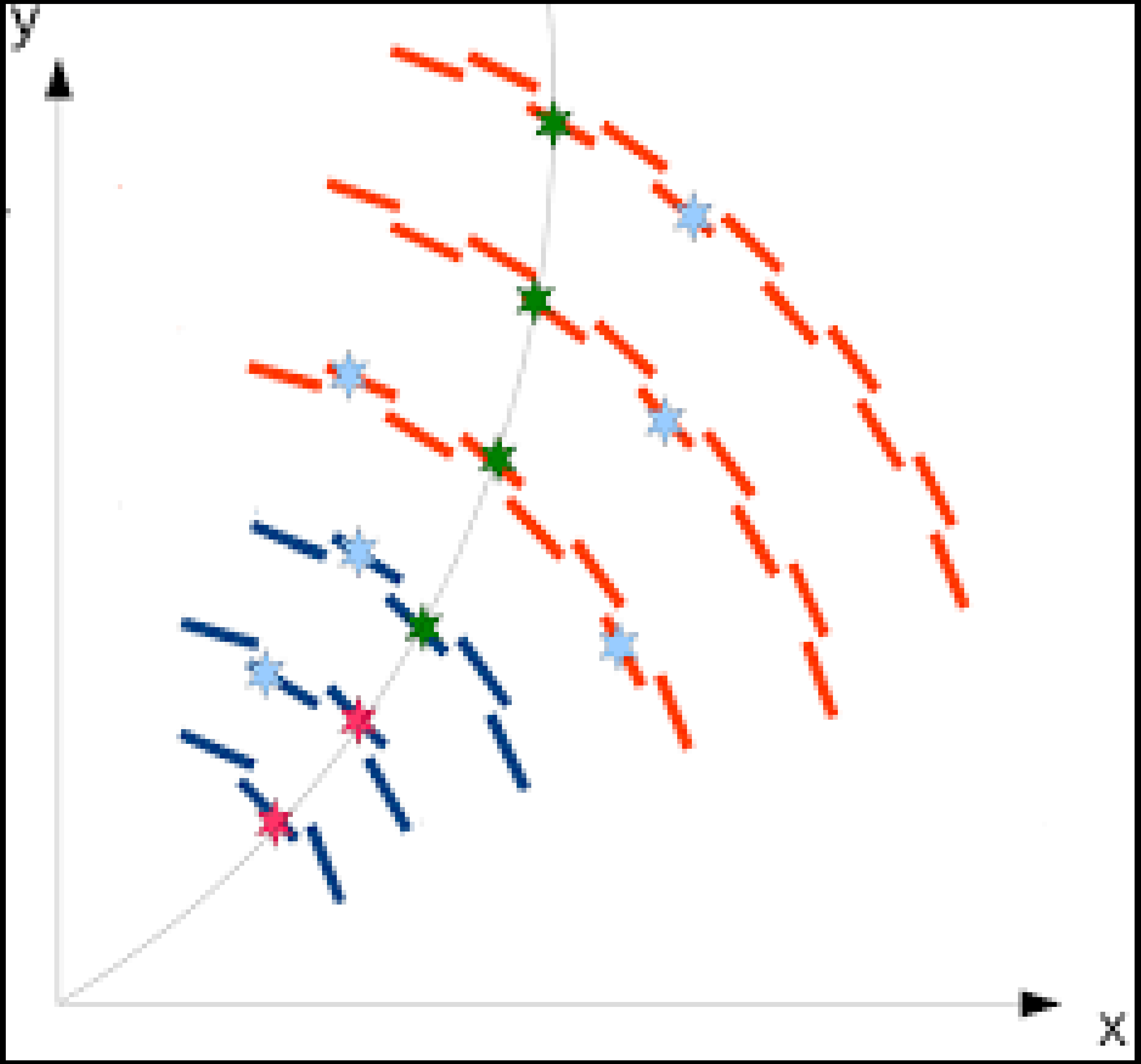
**5 $X_0$** **8.5 $X_0$** **12 $X_0$** **15 $X_0$** **17 $X_0$** **19 $X_0$** **21 $X_0$** **27 $X_0$**

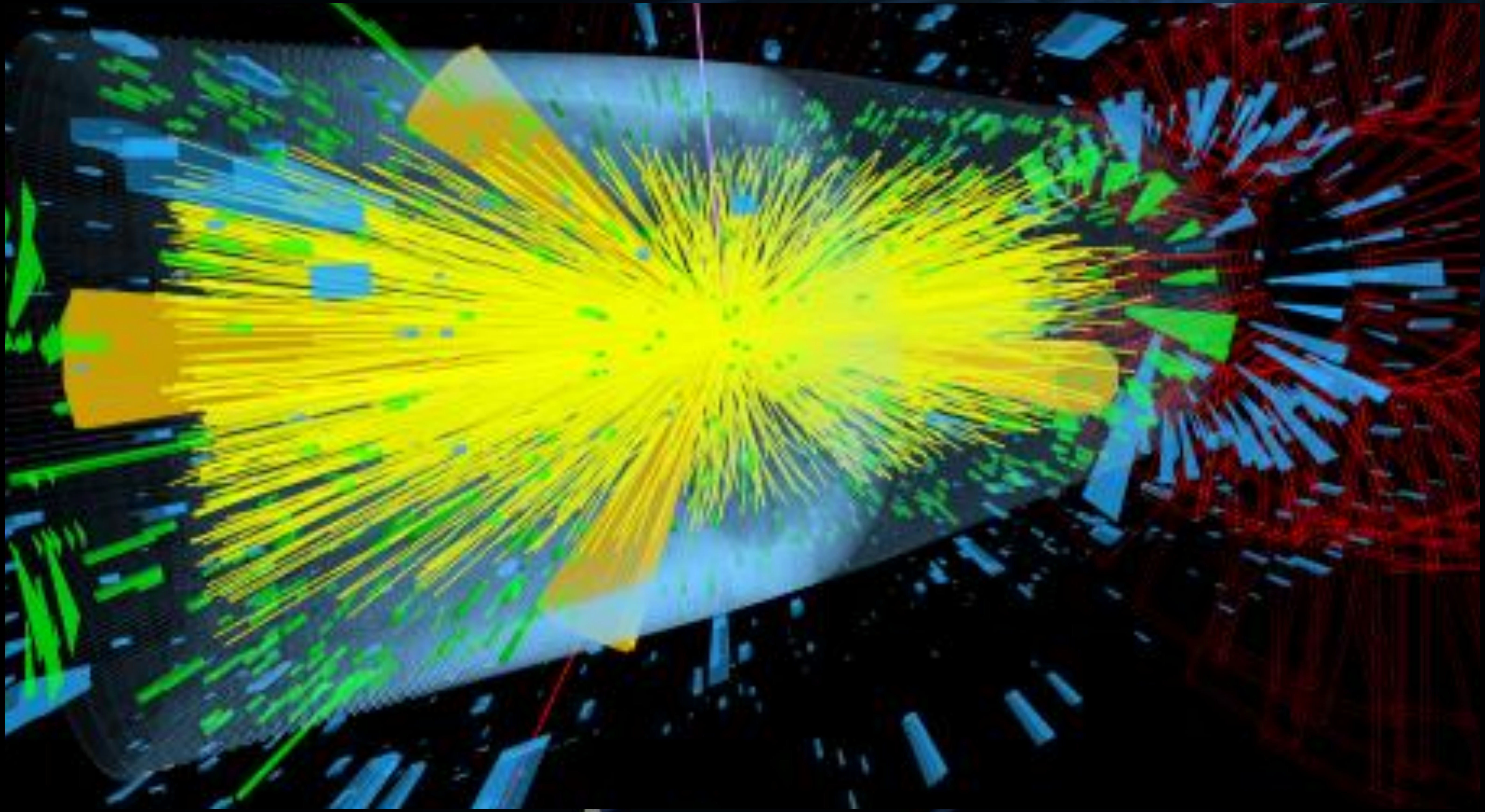
# What are the firmware challenges at Phase-II?

- You mean, apart from the small matter of 300Tb/s of data?
- So much data it has to be zero-suppressed
  - No (or, at least, limited) geometric timing which can be utilized
  - Variable data-volume
    - Do you handle the worst case? Very inefficient
    - Do you handle the average? How do you handle overflows?
- We did such a good job at Phase-I, people have very high expectations…

# What are the firmware challenges at Phase-II?

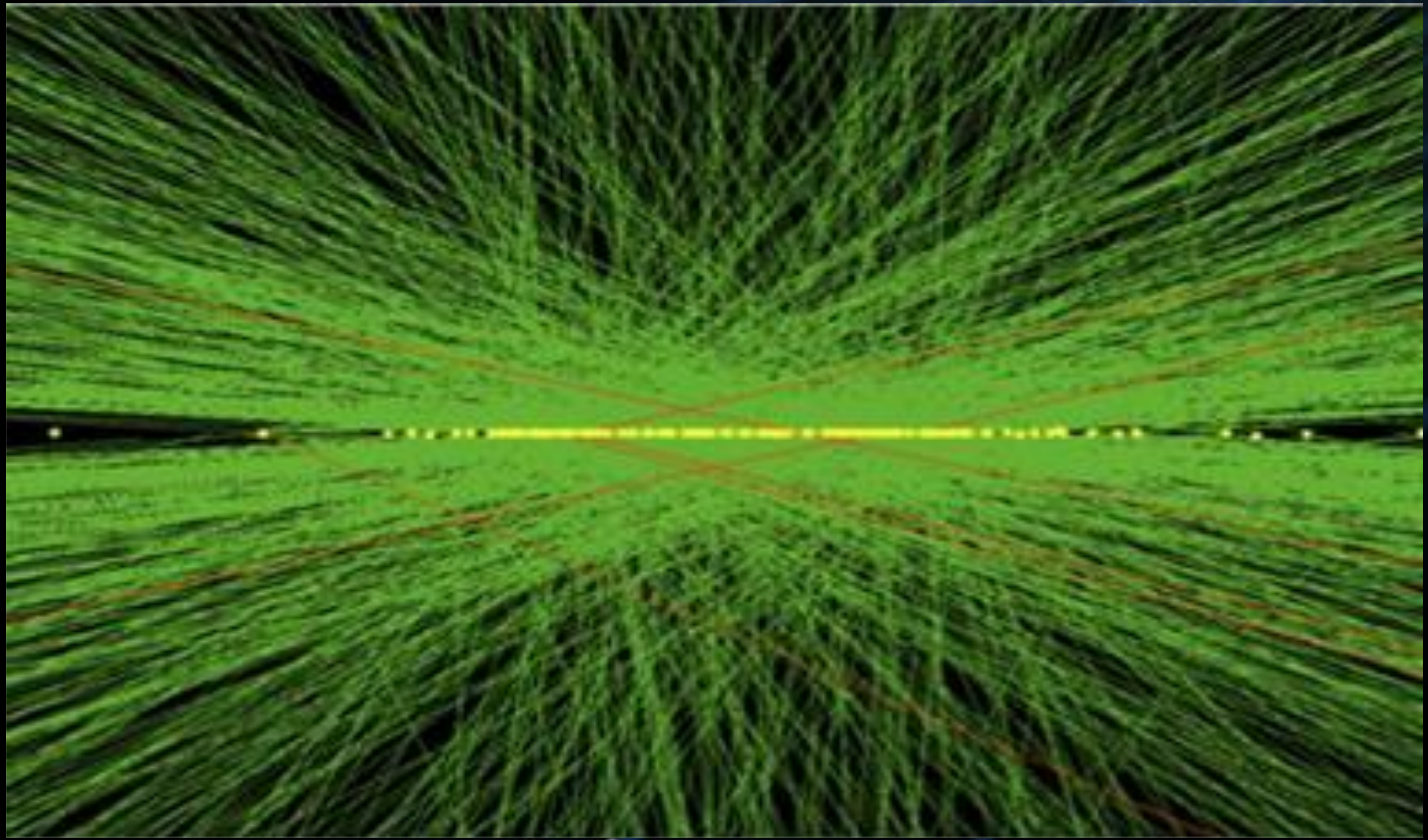
- Real-time track-finding and fitting



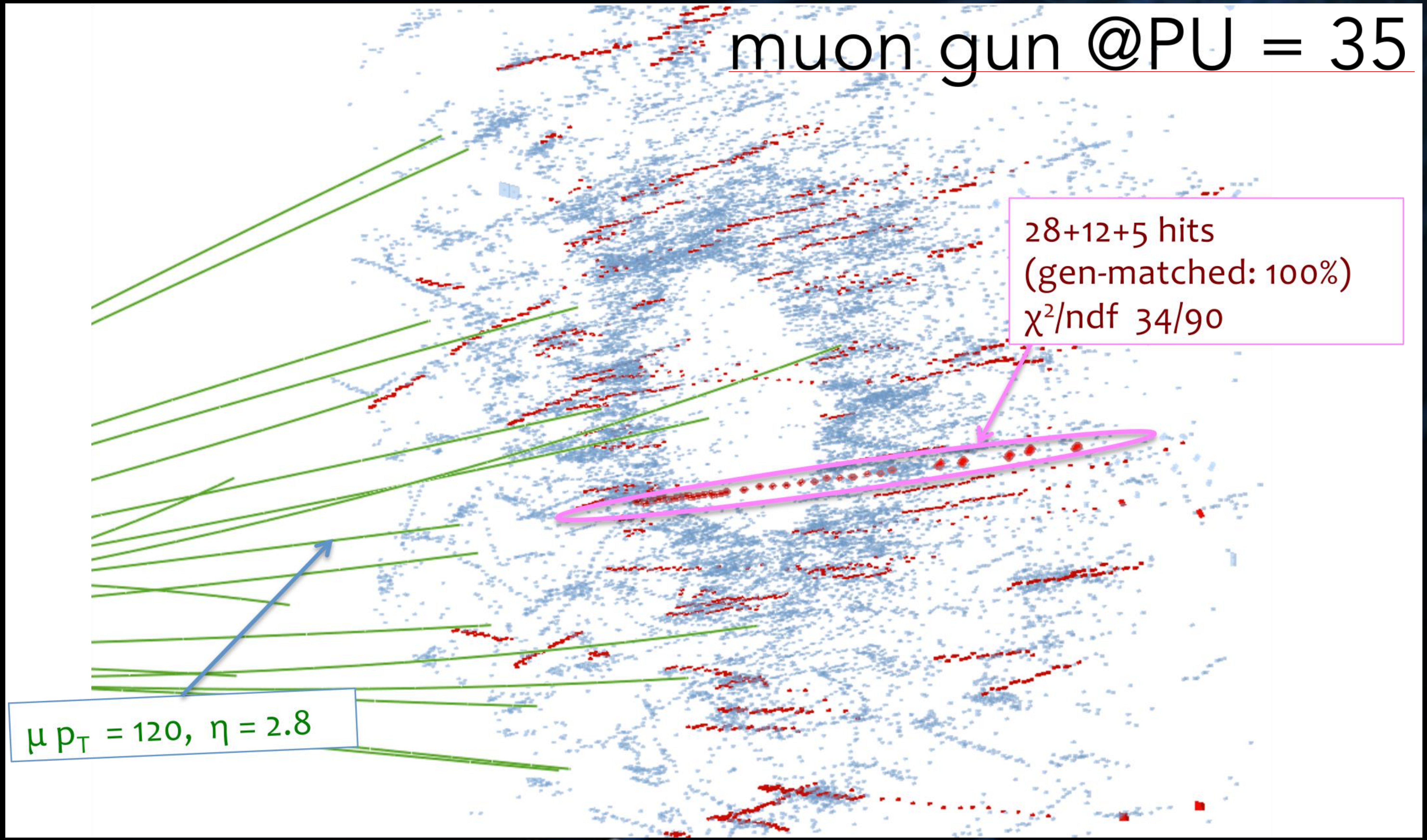
# What are the firmware challenges at Phase-II?

- Real-time track-finding and fitting
- Real-time vertex-finding



# What are the firmware challenges at Phase-II?

- Real-time track-finding and fitting
- Real-time vertex-finding
- 3D cluster-finding in endcap



# What are the firmware challenges at Phase-II?

- Real-time track-finding and fitting
- Real-time vertex-finding
- 3D cluster-finding in endcap
- Particle-flow