ISOTDAQ 2023

Trigger/DAQ design:
from test beam to medium size experiments

触发器 /DAQ 设计 :
从测试光束到中型实验

Roberto Ferrari
INFN - Pavia

Hefei, June 22nd, 2024

Uh, not sure about this:

触发器 /DAQ 设计 :
从测试光束到中型实验

Uh, not sure about this:

触发器 /DAQ 设计 :
从测试光束到中型实验

Progettazione infradito(“flip-flop”)/DAQ:
Dalle travi di prova agli esperimenti su media scala

Roberto Ferrari
Istituto Nazionale di Fisica Nucleare

Roberto Ferrari
Istituto Nazionale di Fisica Nucleare

Oh my!

Roberto Ferrari
Istituto Nazionale di Fisica Nucleare

Oh my!

Yet another f…[(1)] *Italian !*[(2)]

Roberto Ferrari
Istituto Nazionale di Fisica Nucleare

Oh my!

Yet another f…[(1)] Italian ![(2)]

(1) fanatic ... fantastic … ?

Roberto Ferrari
Istituto Nazionale di Fisica Nucleare

Oh my!

Yet another f…(1) *Italian !*(2)

(1) fanatic ... fantastic … ?

(2) about 12 lectures over 30 covered by (half/full) Italians

ISOTDAQ:

Italian School Of Trigger and Data AcQuisition

(Credit: Markus)

But …

But …

1) How do Italians communicate?

But …

1) How do Italians communicate?

2) How can you really understand them?

In Appendix A, you may find

introductory slides for

such demanding environment

In Appendix A, you may find

introductory slides for

such demanding environment

(students' homework & brief introduction to Italians' body language)

regardless of that ...

(1)

→ hope to give you something sensible ←

(2)

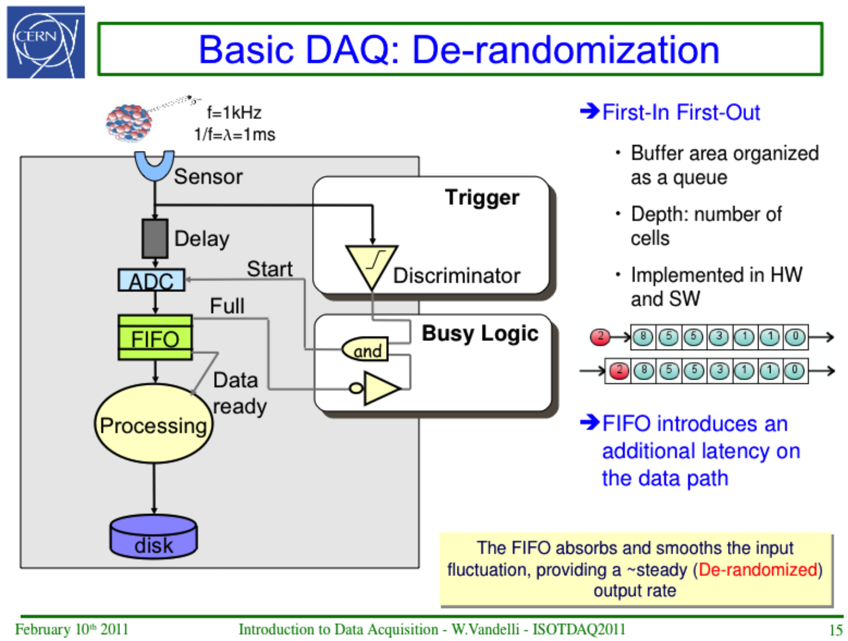
→ but, please, don’t take anything at face value ←

just aiming at enlightening some critical issues

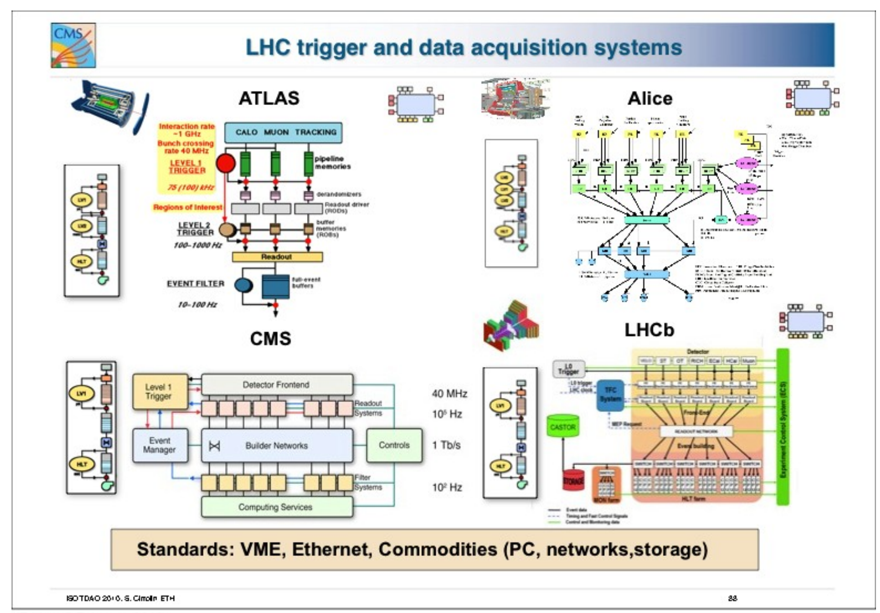
not meant to be exhaustive (no way!)

Trying to move …

from here:

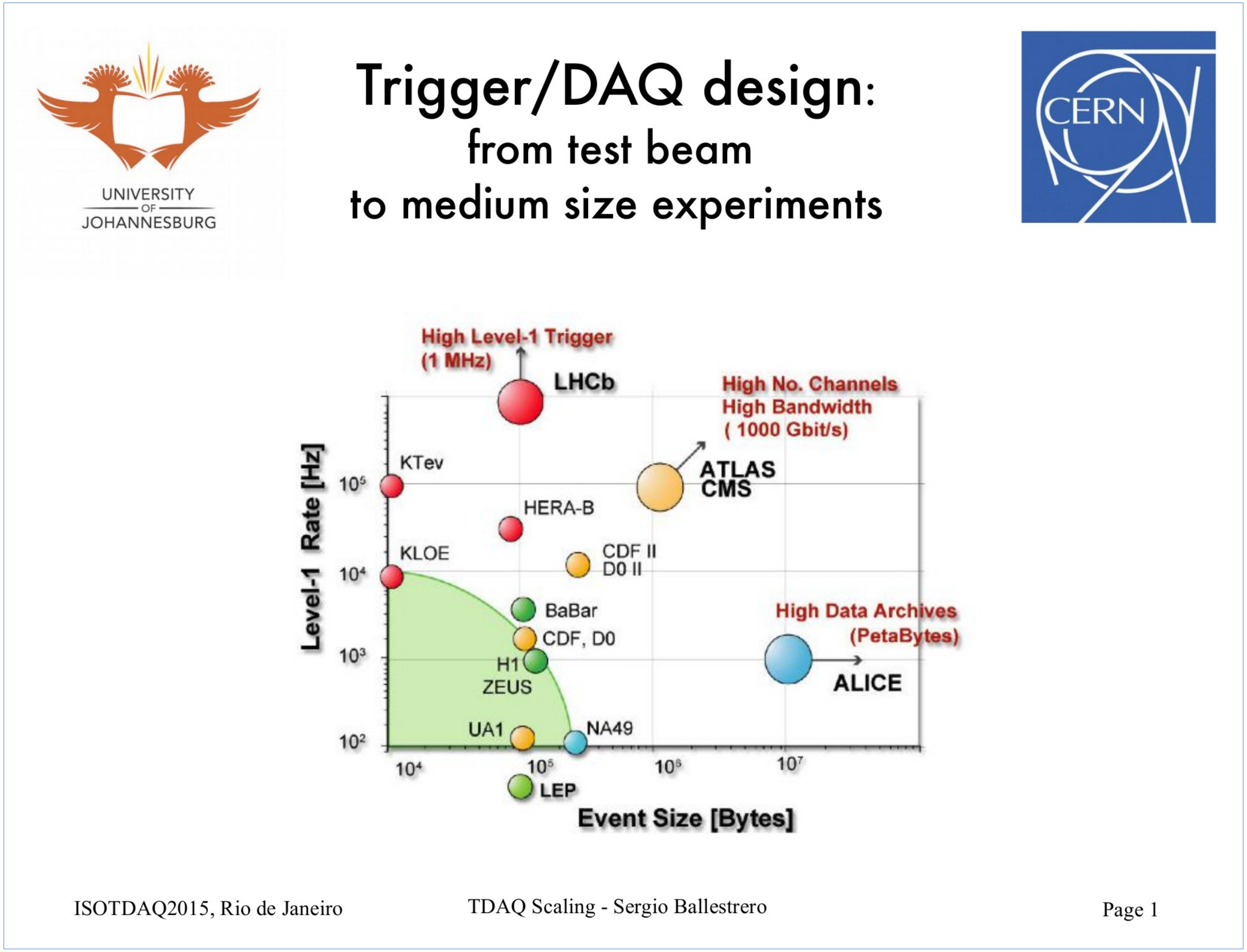


to here:

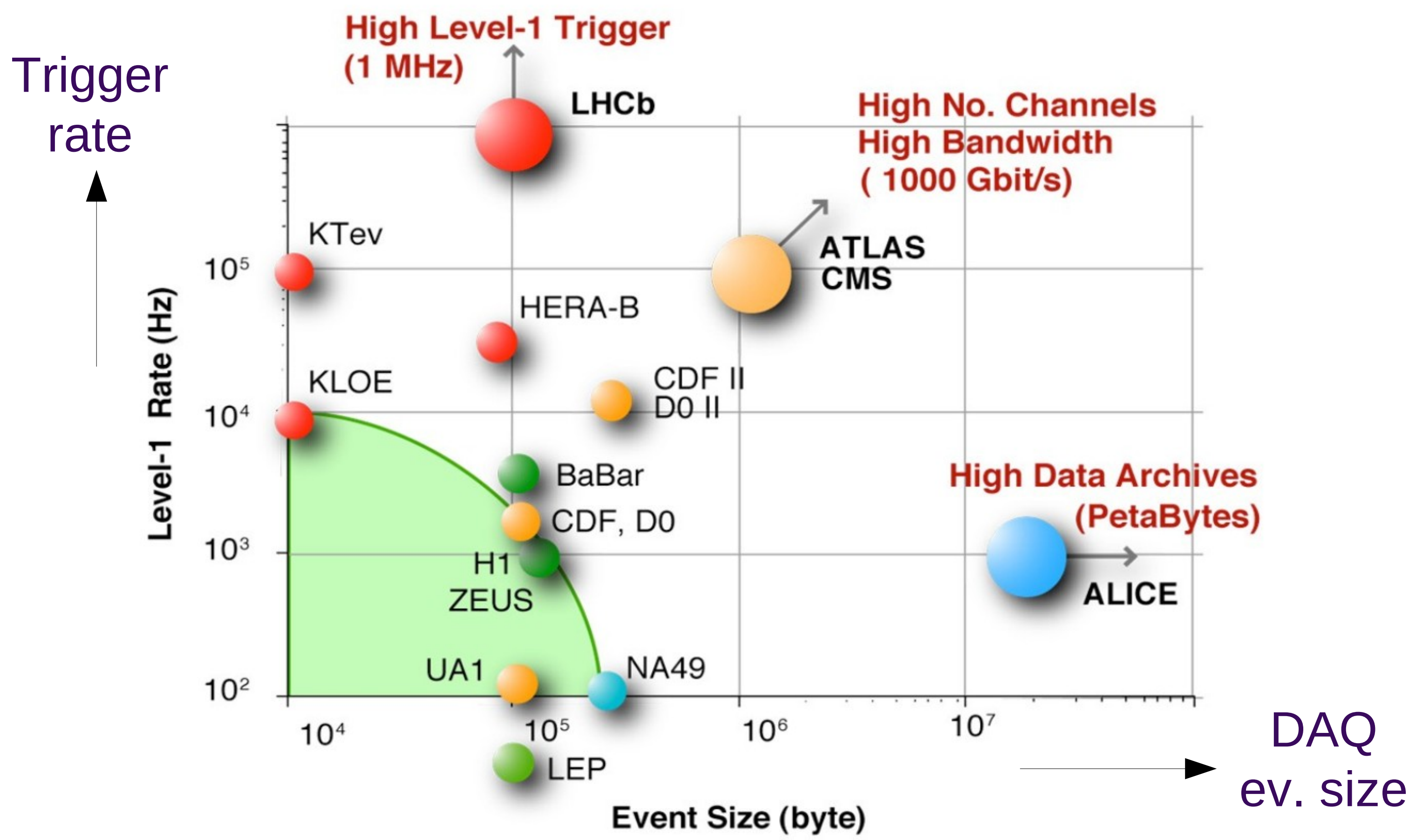


Friday, 5 February 2010

credit to Sergio Ballestrero
most material from his talk at ISOTDAQ 2015



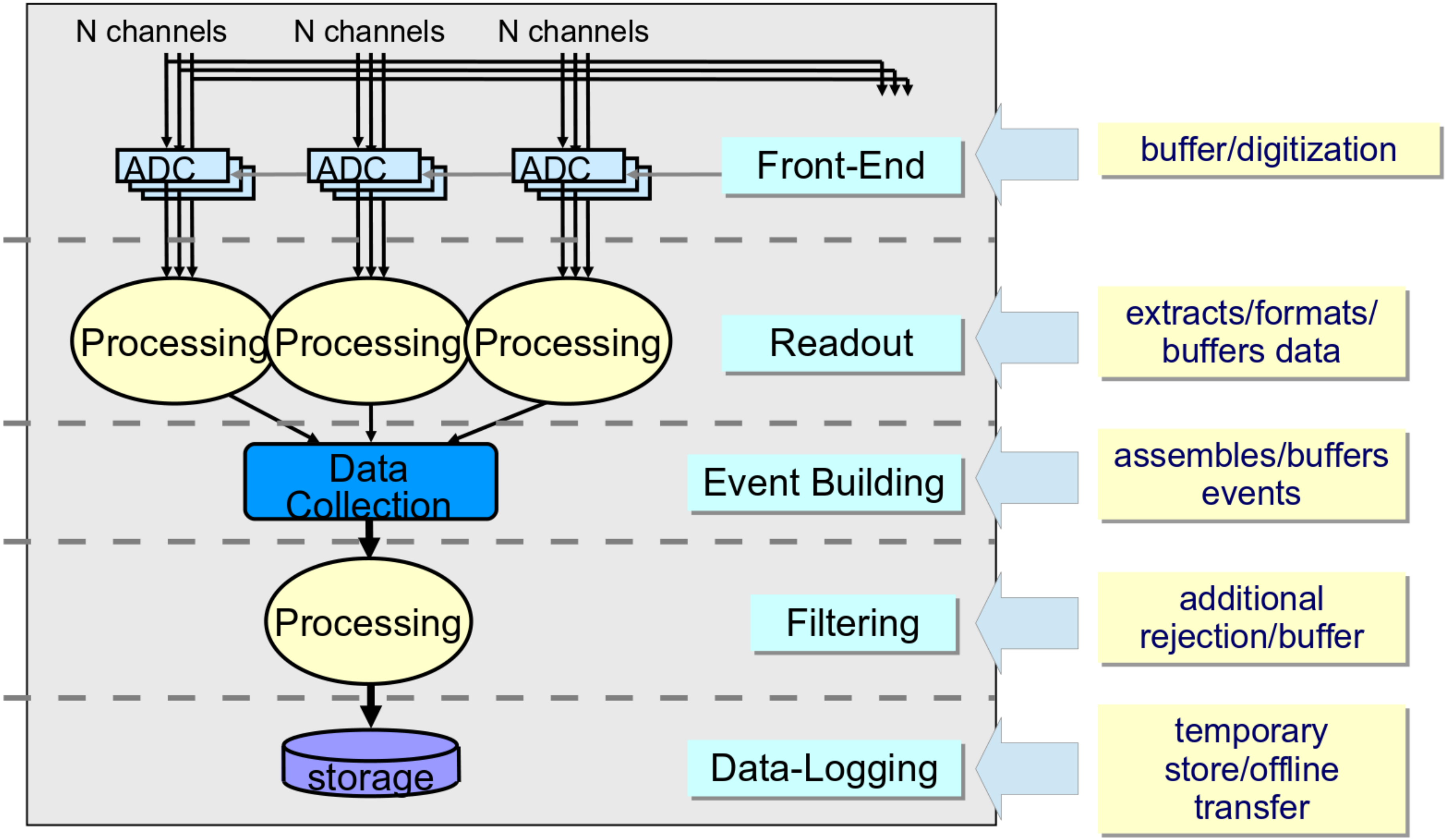
Trigger & DAQ in HEP



different issues → different solutions

no magic, unique solution for all cases

medium/large DAQ: constituents



breakdown into 5 steps …

- Step 1: Increasing rate
- Step 2: Increasing sensors
- Step 3: Multiple front-ends
- Step 4: Multi-level trigger
- Step 5: Data-flow control

back to square one

A minimal system: what do we need ?

back to square one

Do we really need a trigger ?

back to square one

Do we really need a trigger ?

not obvious … triggerless DAQ systems do exist

back to square one

Do we really need a trigger ?

not obvious … triggerless DAQ systems do exist

even in HEP experiments

back to square one

Do we really need a trigger ?

not obvious … triggerless DAQ systems do exist

even in HEP experiments

e.g.:

a) LHCb upgrade: 40 MHz readout

b) DUNE: LAr TPC 2 MHz readout

however, in most cases,

triggering is crucial !

how trigger is born

how trigger is born

Walther Bothe (1924-1929):

Offline (then online) coincidence (logic **AND**) of two independent conditions

Bruno Rossi (Nature, 1930):

"Method of Registering Multiple Simultaneous Impulses of Several Geiger Counters"

→ online coincidence of 3 signals

→ scalable!

→ one order of magnitude faster!

first “modern” trigger

Coincidence circuit [wikipedia]:

“Rossi coincidence circuit was rapidly adopted by experimenters around the world. It was the first practical AND circuit, precursor of the AND logic circuits of electronic computers”

Geiger-Muller counters

Rossi's circuit: coincidence of signals of 3 Geiger-Muller counters

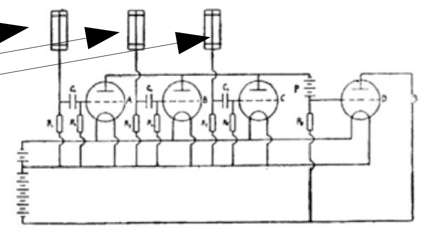


Fig. 17 – Il circuito di Rossi per rivelare coincidenze di raggi cosmici che arrivano sui contatori Geiger (i rettangoli in alto dello schema)[19].

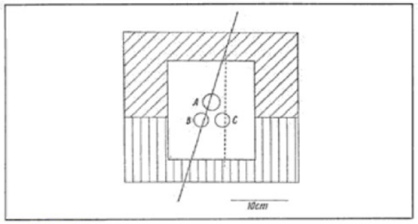


Fig. 18 – L’uso del circuito di Rossi per rivelare una coincidenza tripla che, nella disposizione in figura dei tre contatori, mostra la produzione di una radiazione secondaria (linea tratteggiata) da parte della radiazione primaria (linea continua)[20].

Why triple coincidence ?

To strongly suppress random coincidences

Data storage:

record acoustic signals

This instrument works from A.C. mains and operates a six-figure counting train, one unit corresponding to one microvolt-hour. The integrator is arranged to handle electromotive forces ranging from − 150 to + 300 microvolts. A description of the apparatus is being prepared for publication.

A. F. Dufton.

Building Research Station,
Garston, Herts, Mar. 24.

Method of Registering Multiple Simultaneous Impulses of Several Geiger's Counters.

Prof. W. Bothe in the *Zeitschrift für Physik* (vol. 59, p. 1) describes a method for registering simultaneous impulses of two Geiger's counters, which depends principally on the working of a two-grid thermionic valve. Lately, I have had the opportunity of experimenting with a circuit which perhaps is simpler and at the same time has the advantage that it can be extended also to the registering of triple simultaneous impulses or even more. The circuit adopted (for triple coinciding impulses) is shown in the accompanying diagram (Fig. 1).

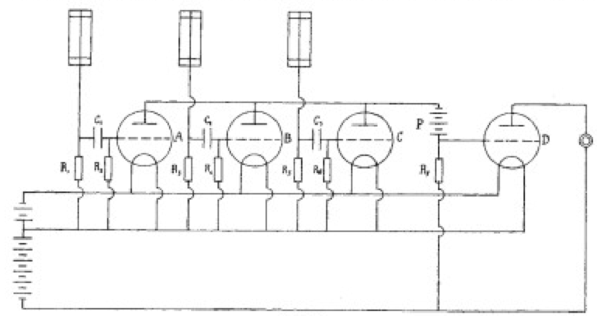


Fig. 1.

$R_1, R_3, R_5 = 5 \times 10^9$ ohms.
$R_2, R_4, R_6, R_7 = 8 \times 10^6$ ohms.
$C_1, C_2, C_3 = 10^{-4}\ \mu F$.

The positive electrodes of the three counters (in my experiments I have used Geiger's wire counters) are electrostatically coupled to the grids of the three valves *A*, *B*, *C*. In normal conditions these grids have a zero potential; whenever a discharge occurs they become negative, thus interrupting the current flow.

As the resistance R_7 is very great compared with the internal resistances of the valves *A*, *B*, *C*, their anodes are at a potential near to zero. The grid of the valve *D* (for the introduction of the auxiliary battery *P*) is at a slight negative potential. This potential varies very little when only one or two counter tubes are working, while it undergoes a sudden rise when, for the simultaneous working of the three counter tubes, the current is interrupted in all the three valves.

The consequent variation of the anode current (eventually amplified by a fifth valve) is acoustically detected by a telephone.

The circuit arrangement, in regard to the counter tubes, is perfectly symmetrical, a condition which is not fulfilled in the circuit of Prof. Bothe, because the grids of the two-grid valve have rather different characteristics.

It appears that the triple coincidences method is the only one available for studying the form of the paths of cosmic rays, and I mean to employ it in experiments on the magnetic deviation of these radiations.

Bruno Rossi.

Physical Institute of
the University of Florence,
Arcetri, Italy, Feb. 7.

The Conversion of α Benzilmonoxime into the β Oxime by Animal Charcoal.

During the course of an investigation into the properties of the isomeric monoximes of benzil, we have made the following somewhat startling observation.

We have been able to devise a method for estimating mixtures of the α and β oximes and have shown that the α oxime shows no appreciable change into its isomer (which is the more stable of the two) in solution in alcohol or benzene at 50° in a period of thirty-six hours, and that the change is not accelerated by acids or alkalis when present in small concentration. On the other hand, if a benzene solution of the α oxime is boiled with animal charcoal for a few seconds, the change is complete and no α oxime can be detected in the solution.

Finely powdered soft-wood charcoal and powdered silica gel showed no such effect, the α oxime being recovered unchanged. Finely divided calcium phosphate is also without action. That the conversion does not arise from the action of catalysts dissolved from the charcoal by the benzene is shown by boiling some benzene with animal charcoal, filtering off the charcoal, and using the filtrate as a solvent for the α oxime; there is no conversion into the isomer.

This observation suggests that care should be exercised in the use of animal charcoal as a decolorising agent in the purification of isomers of the type of this α oxime. A full account of our work on this subject will be published later elsewhere.

T. W. J. Taylor.
Sally Marks.

The Dyson Perrins Laboratory,
Oxford, Mar. 17.

Fluorescent and Phosphorescent Substances.

Substances which fluoresce strongly under the influence of X-rays are barium and magnesium platinocyanides and cadmium tungstate. The formulæ of these compounds, as given by Werner, are as follows:

$[Pt(CN)_4][Ba(H_2O)_4]$,
$[Pt(CN)_4][Mg(H_2O)_7]$,
$[WO_4]Cd$.

An atom of high stopping power with four light atoms or radicals arranged about it, perhaps tetrahedrally, and a bivalent positive ion, are present in all.

With the first part of the formulæ may be compared the structure of zinc sulphide and diamond, which phosphoresce in X-rays; phosphorus and yellow arsenic exhibit phosphorescence on oxidation, and arsenious oxide is luminous on crystallisation from acid solution.

J. R. Partington.

East London College,
University of London, E.1.

reasons for triggering

1) identify “random” (unpredictable), “rare” events → efficiency

2) suppress background → rejection

a simple trigger system

Gokhan's talk:

$$N1 = s1 \cdot s2 \cdot \overline{s3}$$

Veto (anti-coincidence)

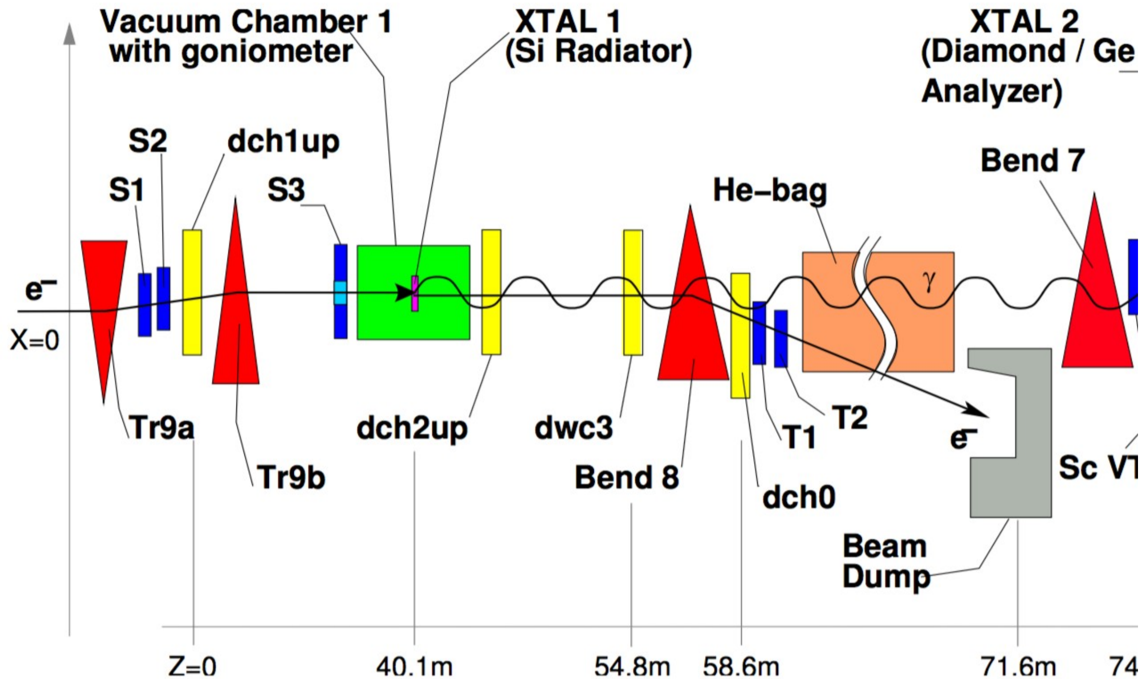
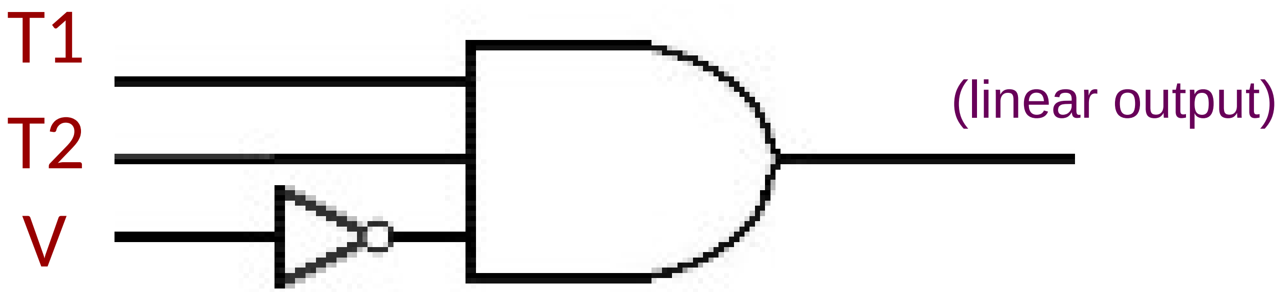


Fig. 1. Setup of the Na59 Experiment

any issue ?

any issue ?

T1, T2, V : logic pulses (“0/1” values)

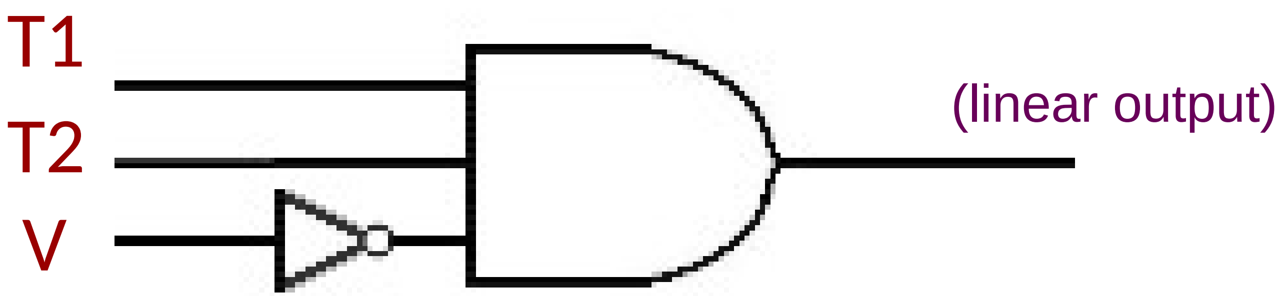


(anti-)coincidence with veto

→ easy !

any issue ?

T1, T2, V : logic pulses ("0/1" values)



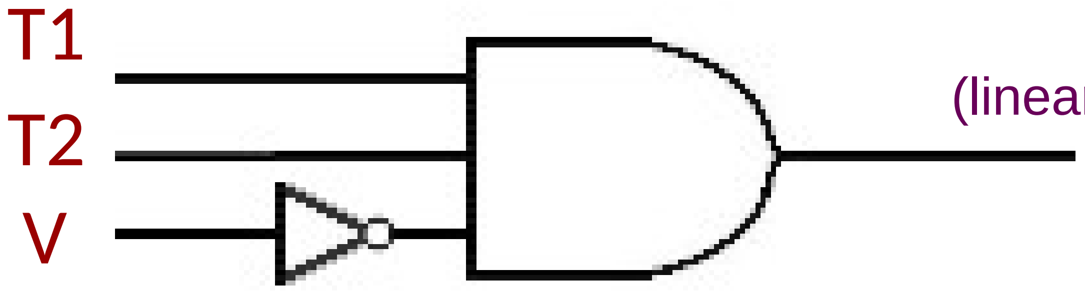
(anti-)coincidence with veto

→ easy !

sure it works ?

uh !

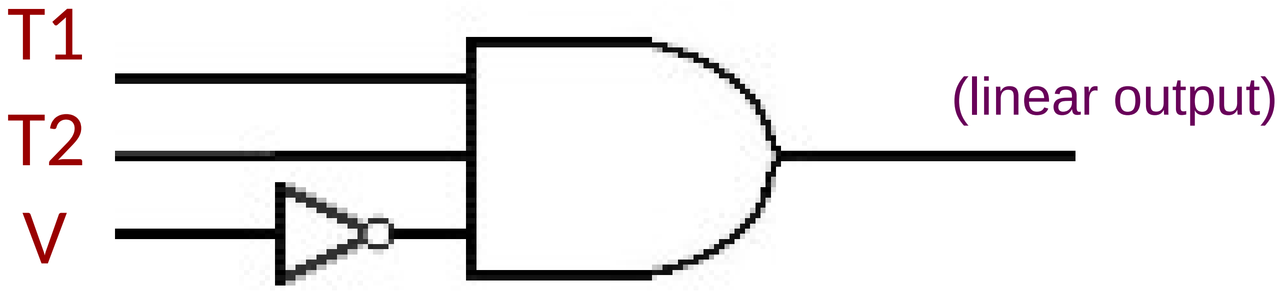
(anti-)coincidence with veto



(linear output)

faulty !

(anti-)coincidence with veto



faulty !

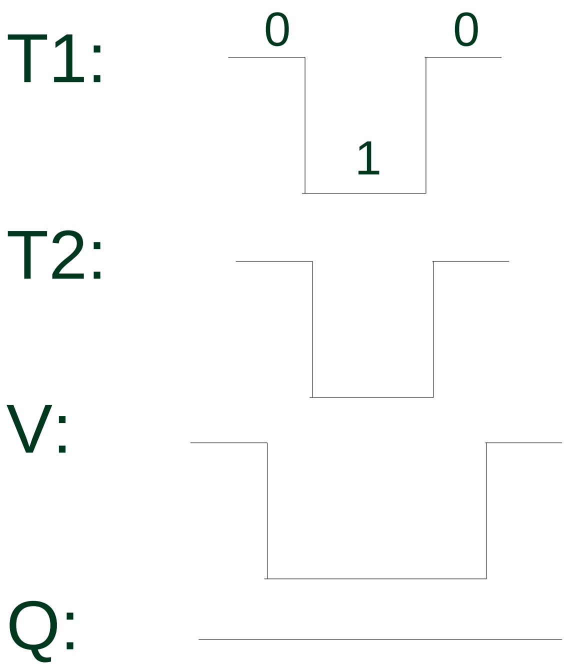
output signal does:

a) jitter

b) fluctuate in duration

why ?

(good) in-time signals



NIM signals:

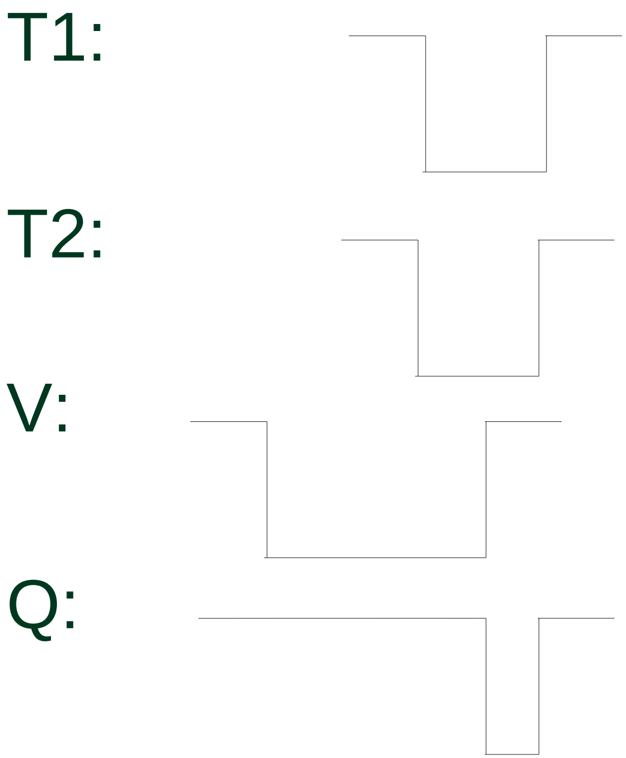
0 ⇔ 0 V

1 ⇔ - 800 mV

1) T1 and T2 (almost) perfectly in time

2) In case it is present, V totally covers in time T1 and T2

(bad) out-of-time veto signals

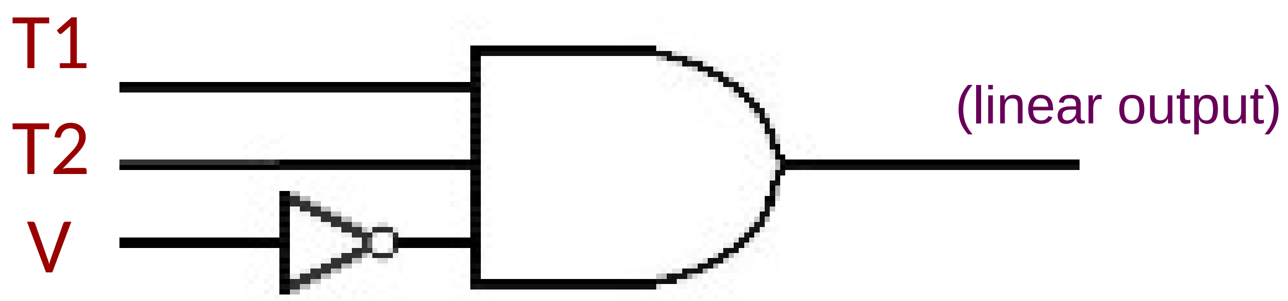


in some cases:

wrong transition time or wrong duration or both

(anti-)coincidence with veto

combinatorial logic

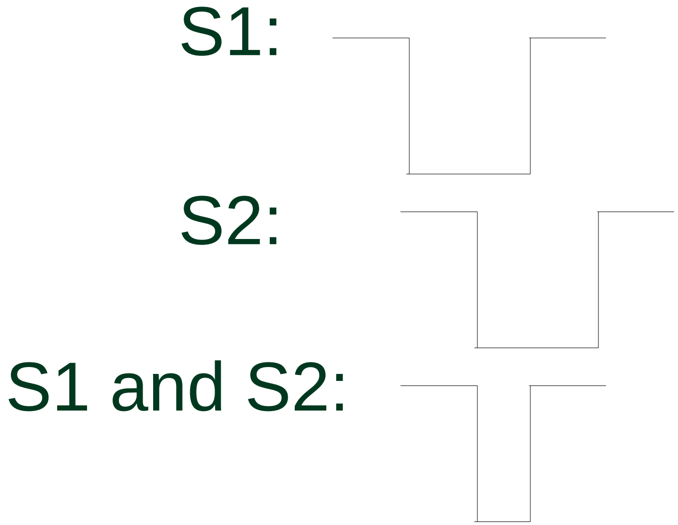


faulty !

output signal will:

a) jitter
b) fluctuate in duration

because of independent signals from T1, T2, V



independent random (i.e. uncorrelated) signals

→ even without veto, trigger signals needs to be formed (“shaped”)

Ah!

Ah!

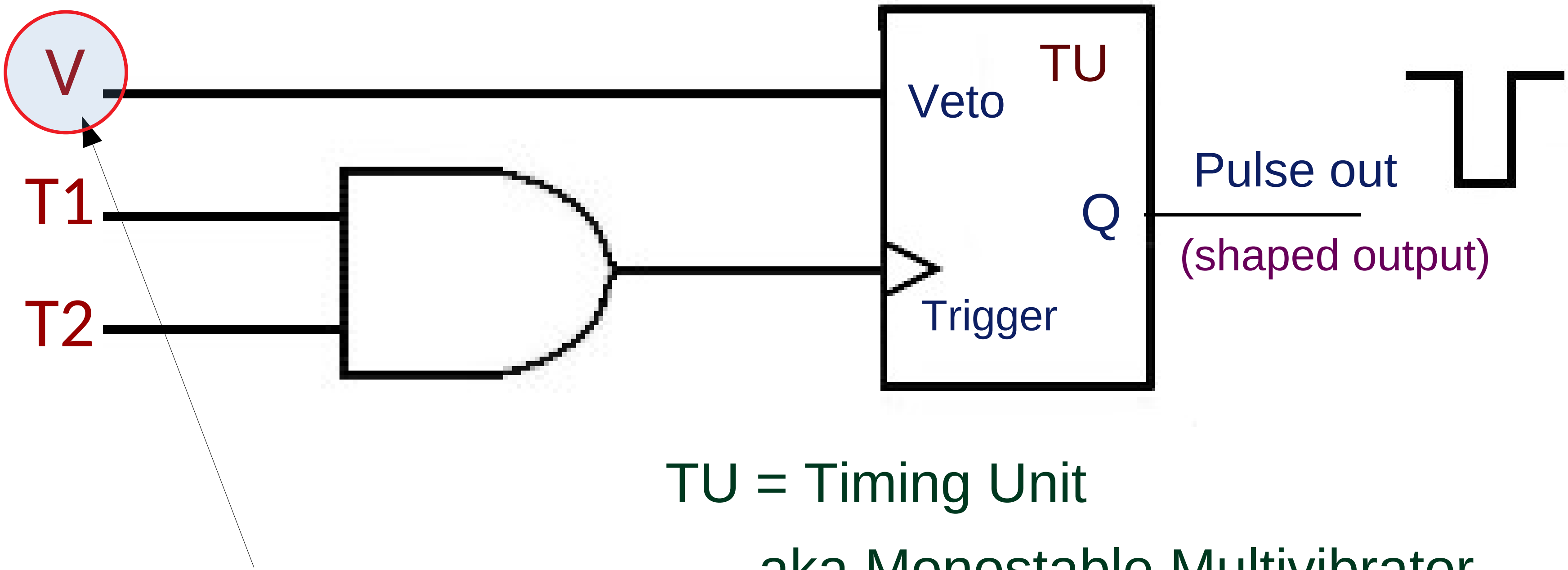
innocent random coincidences are enough to break/violate your perfect trigger setup

can't even blame high rate, high occupancy, …

(anti-)coincidence with veto

combinatorial logic

sequential logic



TU = Timing Unit

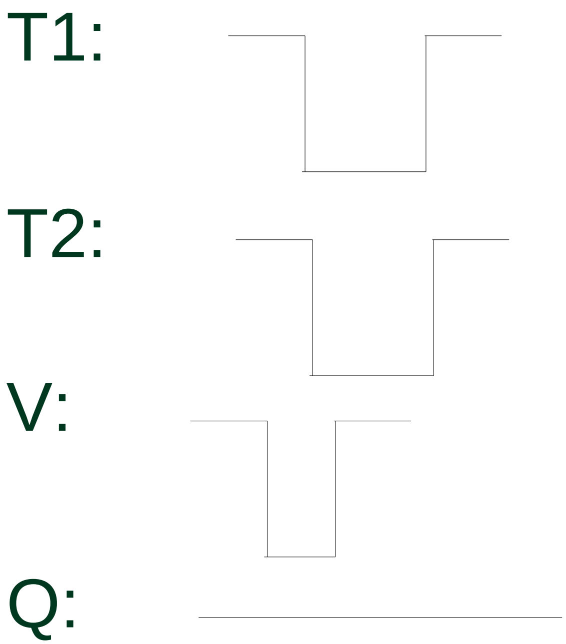
aka Monostable Multivibrator

aka One-Shot Pulse Generator

can also be a busy signal → busy logic

much better !

now



Veto only covers transition region (0→1) of T1 and T2
→ lower dead time → higher efficiency

T1 and T2 → transition time

TU → duration time

T1 and T2 → transition time

TU → duration time

Q: What the relevant information?

first lesson(s)

trigger signal:

1) should be formed!

→ pulse with predefined duration

2) veto/busy should block pulse generation

3) need both combinatorial (AND, OR, NOT) and sequential logic (TU, FF)

step one: increase rate

Many issues:

→ trigger latency

→ readout latency

→ throughput

→ rate fluctuations (trigger bursts)

→ throughput fluctuations
(correlated noise, …)

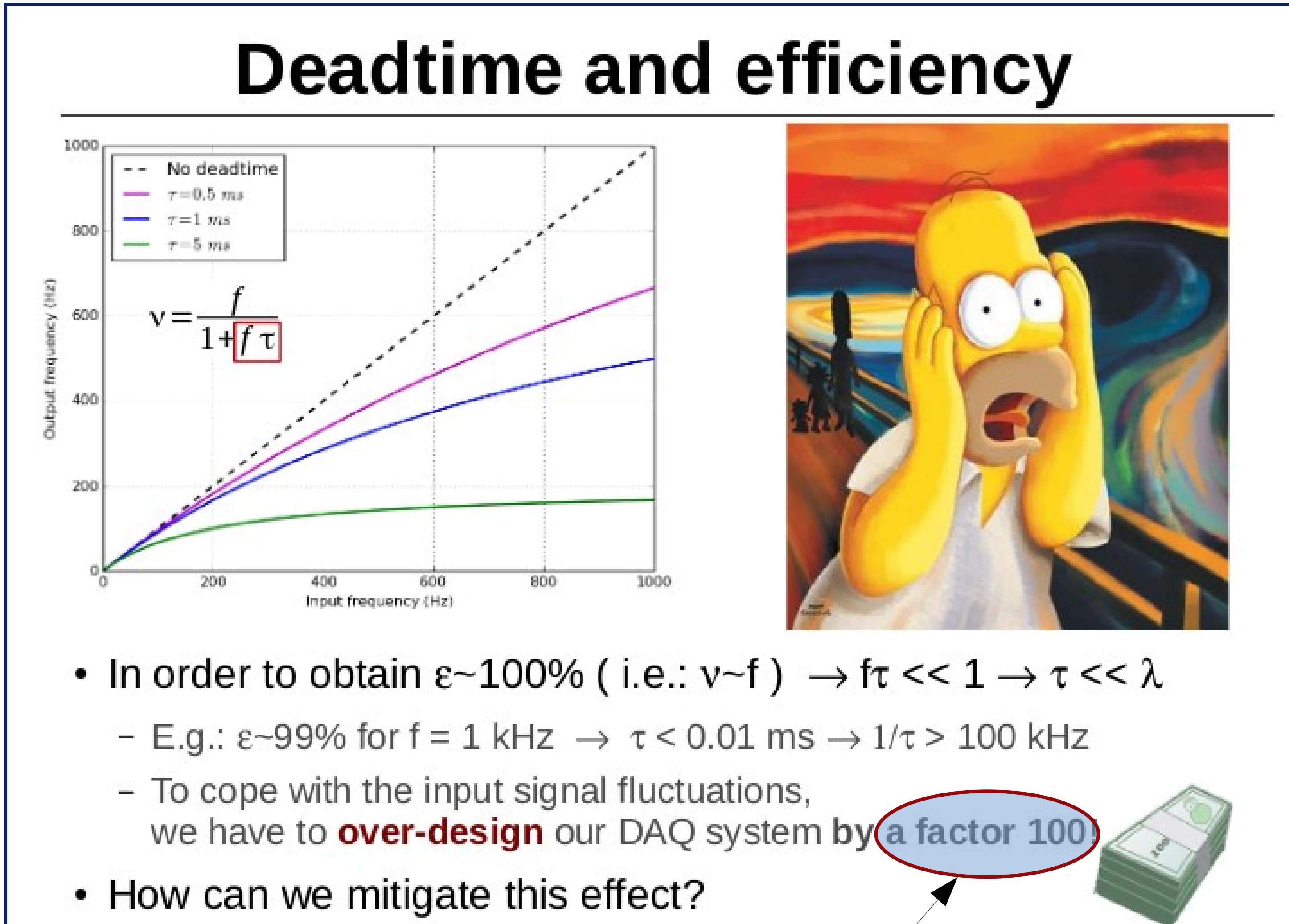
step one: increase rate

Many issues:

→ trigger latency
→ readout latency
→ throughput
→ rate fluctuations (trigger bursts)
→ throughput fluctuations (correlated noise, …)

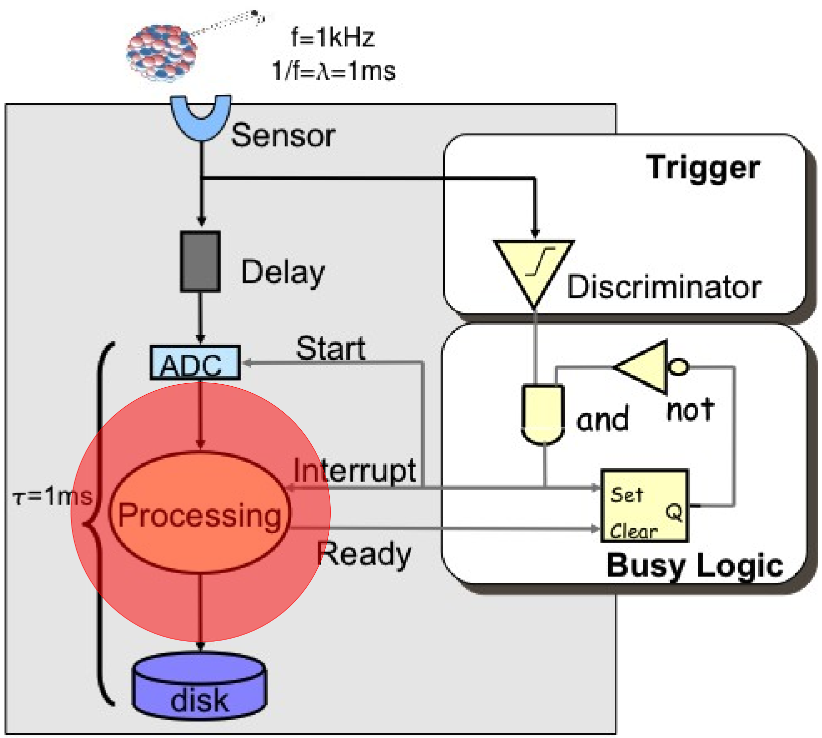
→ dead-time

deadtime (from Andrea's introduction)



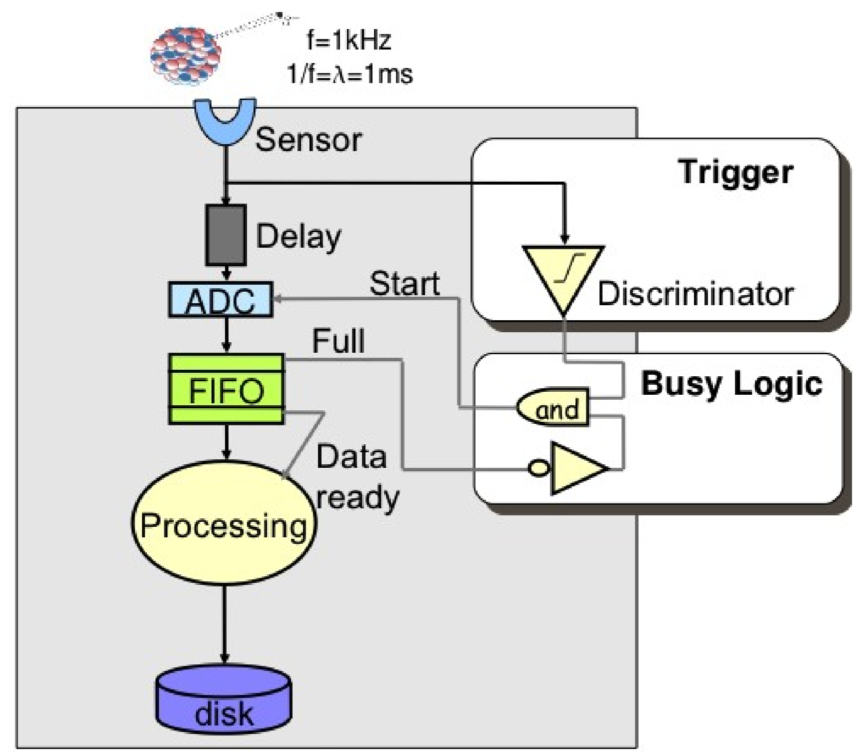
deadtime → de-randomise

- Processing → bottleneck ?



$(f\cdot\tau) \sim 1 \rightarrow$ deadtime $\sim$ 50%

- Buffering → decouple problems



What the impact ?

$(f\cdot\tau) \sim 1 \rightarrow$ deadtime?

FIFO

First-In First-Out memory:

1) independent read/write (sequential) access

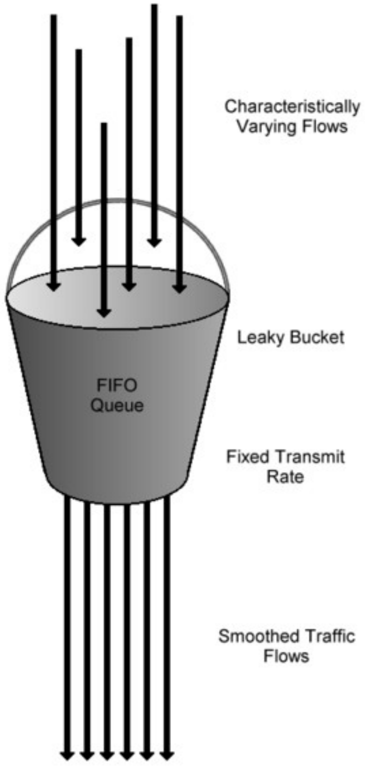
2) may be hardware or over RAM

if RAM better Dual-Port RAM

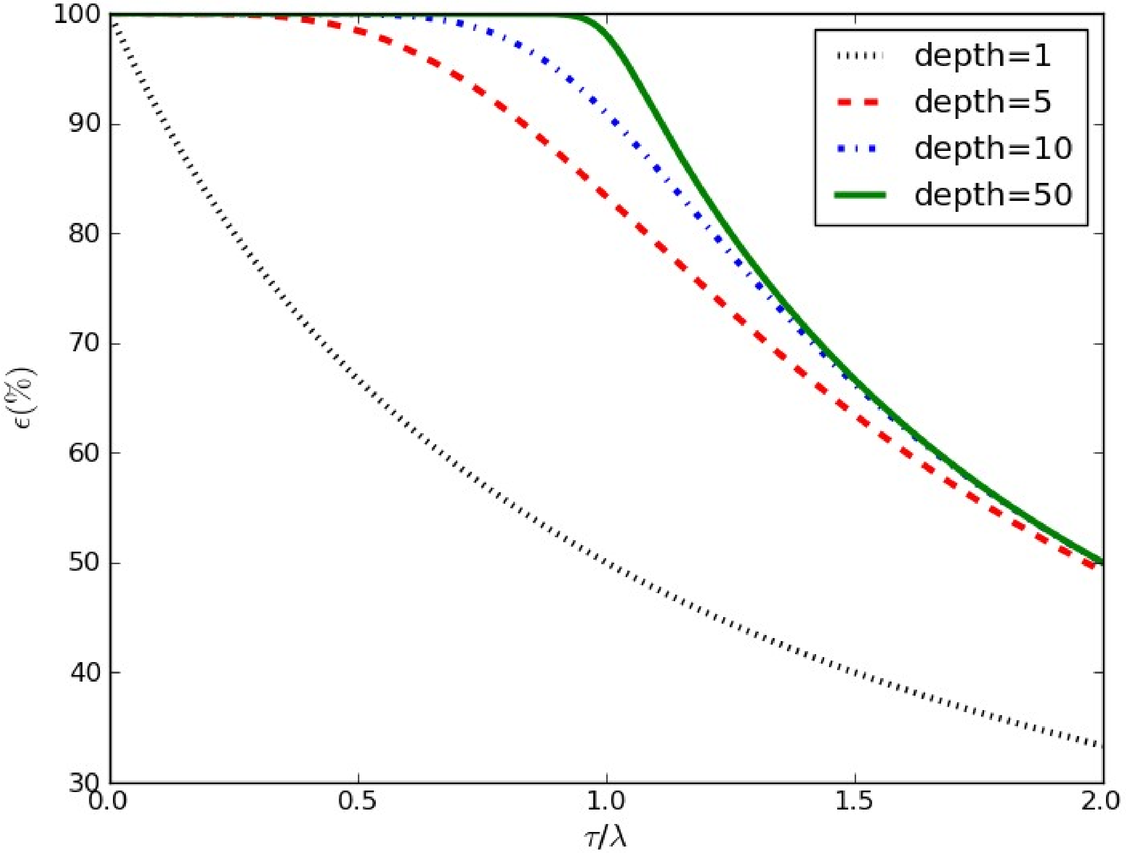
buffering solve all problems ?

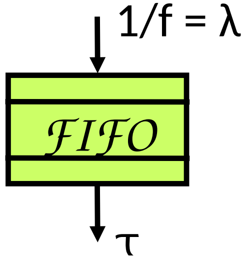
- FIFO (front-end buffers)
  1) filling at very variable input flow
  2) emptying at smoothed output flow
  → the Leaky-Bucket problem

  Q: how often may overflow?



de-randomisation





- DAQ ε ~100% with:
  - τ ~ 1/f
  - “moderate” buffer size
- Two degrees of freedom to play with
- This deadtime often managed by trigger system itself (“complex deadtime”)

deadtime in trigger system

1) Simple deadtime: avoid overlapping (conflicting) readout window

2) Complex deadtime: avoid overflow in front-end buffers (protection against trigger bursts)

→ different subdetector & different front-end elx

→ different algorithm/parameters

ATLAS deadtime @ end of run 2

1) Simple deadtime: 4 LHC BC [i.e. 100 ns] after any LVL-1 trigger

2) Complex deadtime:

2.a) four leaky-bucket algorithms

[two params: bucket size S (in number of events), readout time R (in BC units)]

1) 15 / 370 for “LVL-1 Calorimeter” and CSC readout

2) 42 / 384 for TRT readout

3) 9 / 351 for LAr readout

4) 14 / 260 for “LVL-1 Topo” readout

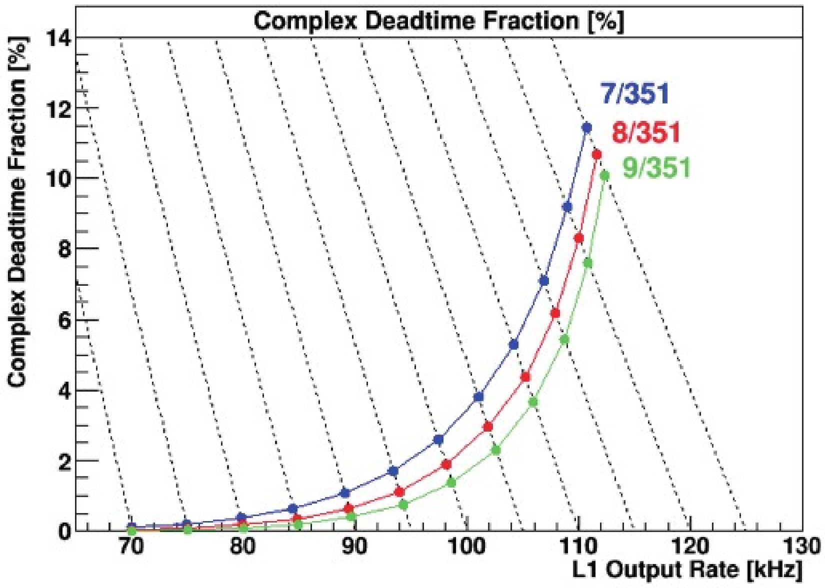
2.b) one sliding-window algorithm

< 16 LVL-1 signals in any 3600 BC sliding window

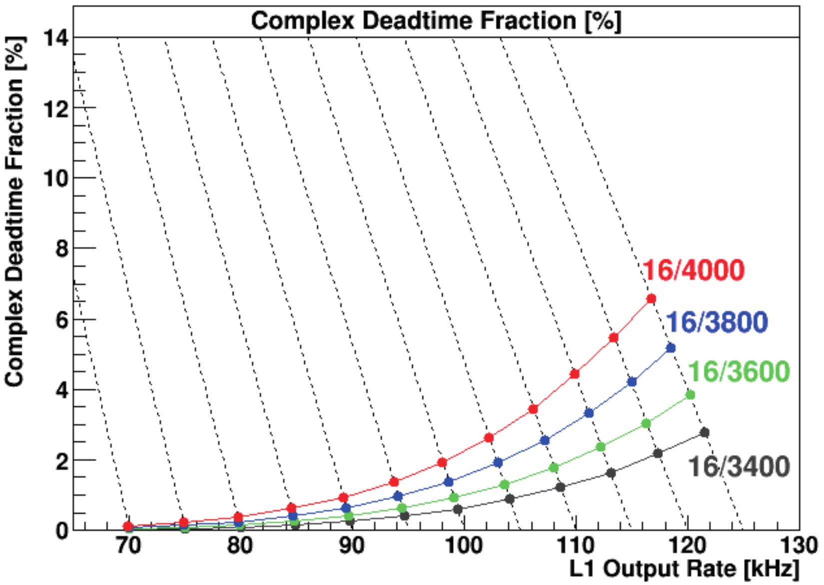
ATLAS deadtime @ end of run 2

Total deadtime @ 90 kHz trigger rate < 2%

Leaky bucket (LAr readout)

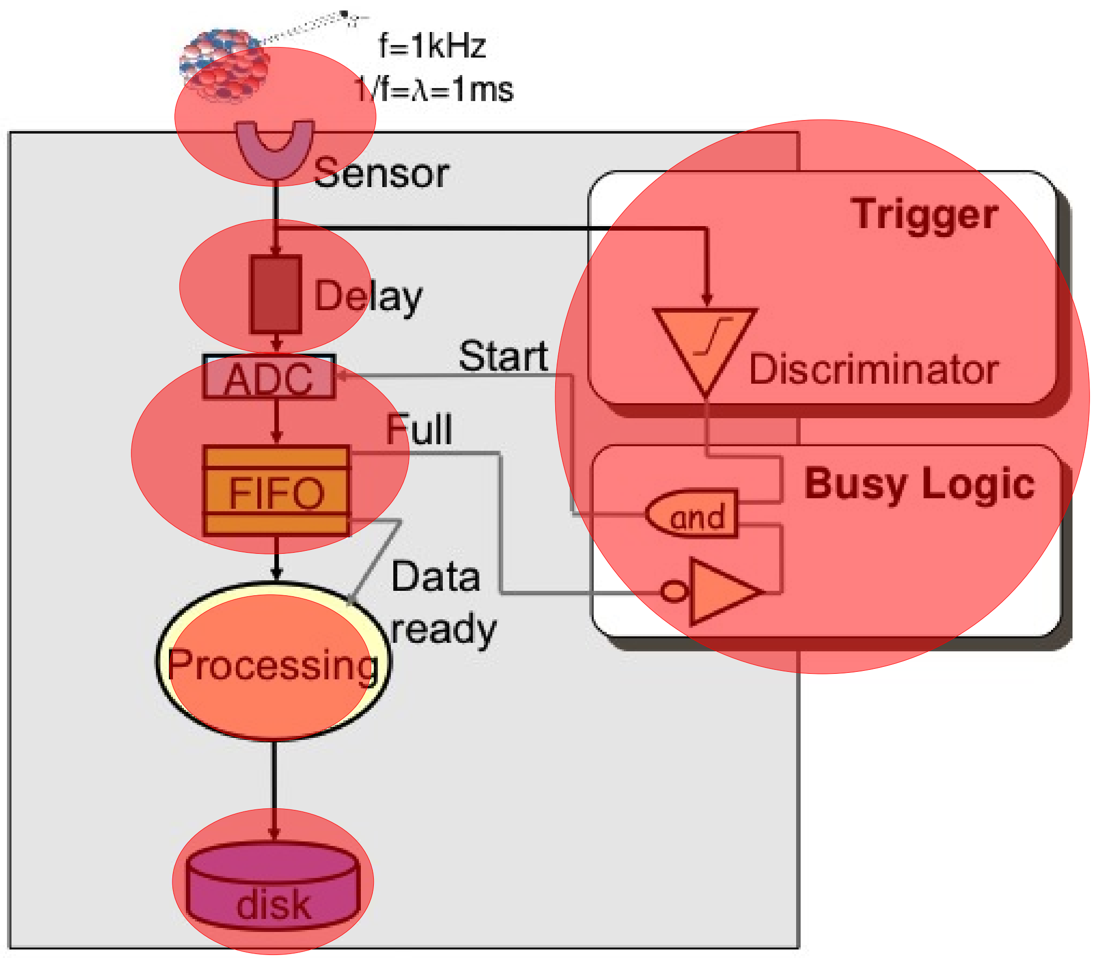


Sliding window (SCT readout)



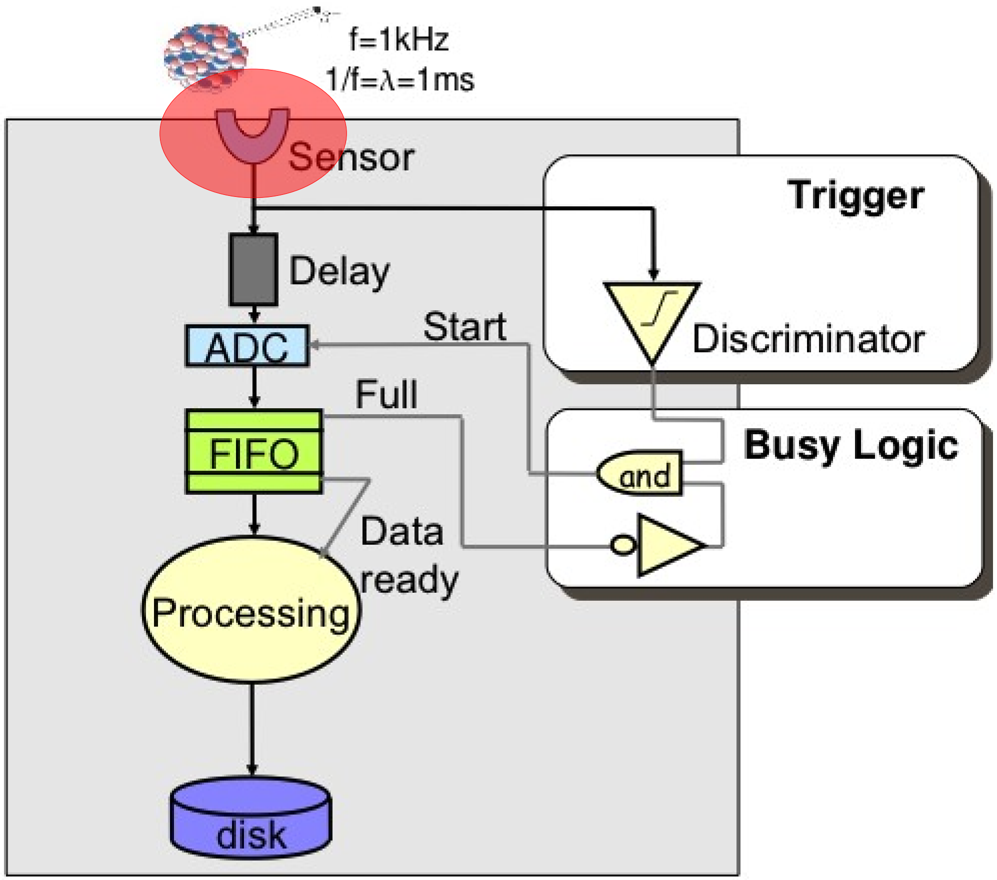
game over ?

many other possible limits even in a simple DAQ



→ sensor

- Sensors limited by physical processes such as:
  - drift times in gases
  - charge collection in Si
- (possibly) choose fast processes
- analog FE imposes limits as well
- split sensors, each gets less rate: "increase granularity"

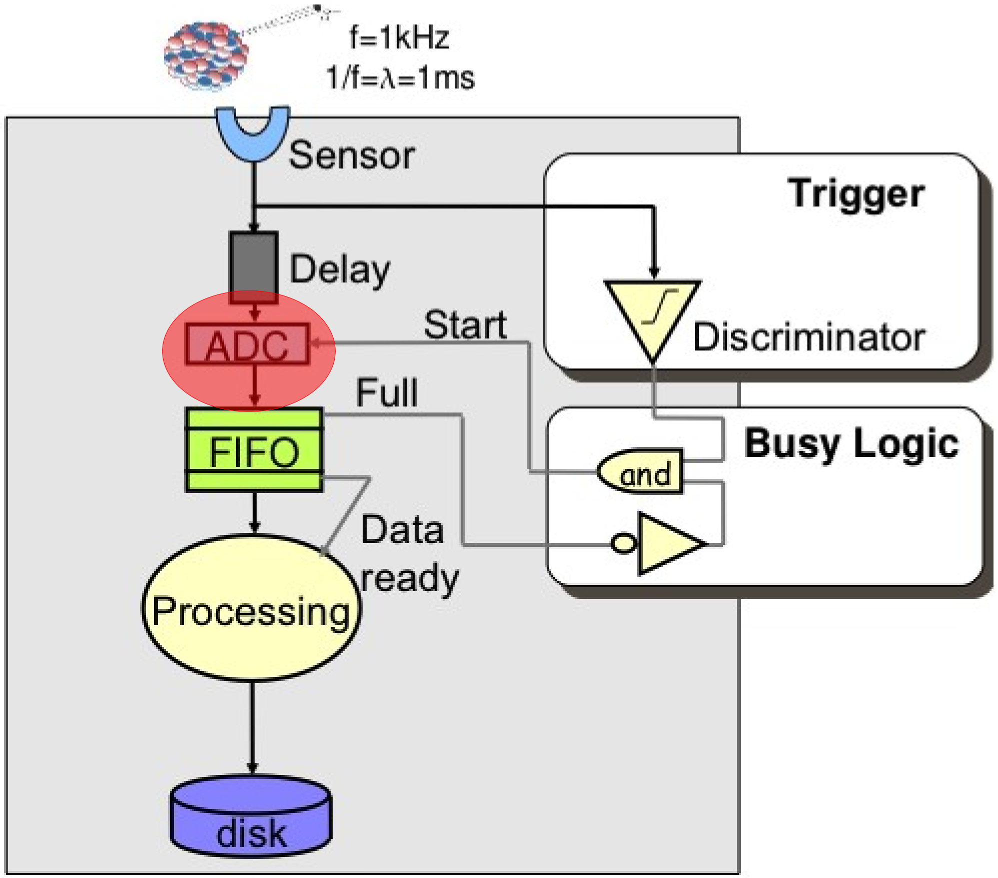


→ ADC

- A/D conversion also limited
- Fast ADC
  - → # of bits (resolution)
  - → power consumption
- Alternatives:

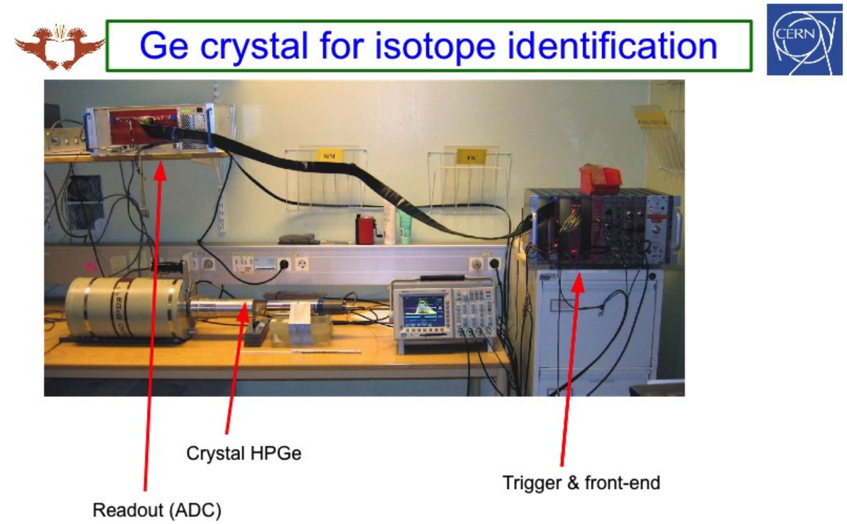
  analog buffers

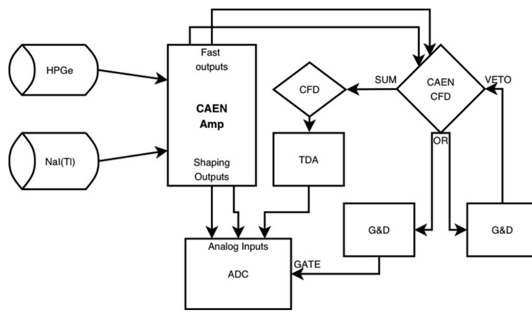
  (e.g. switched capacitor arrays)
- You may need integration (or sampling) over quite some time

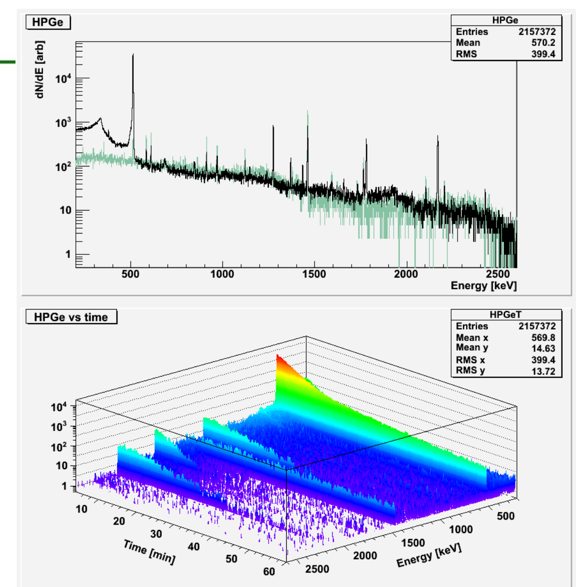


an example

- HPGe + NaI Scintillator
  High res spectroscopy and beta+ decay identification
- minimal trigger with busy logic
- Peak ADC with buffering, zero suppression
- VME SBC with local storage
- Root for monitor & storage
- Rate limit ~14 kHz
  - HPGe signal shaping for charge collection
  - PADC conversion time

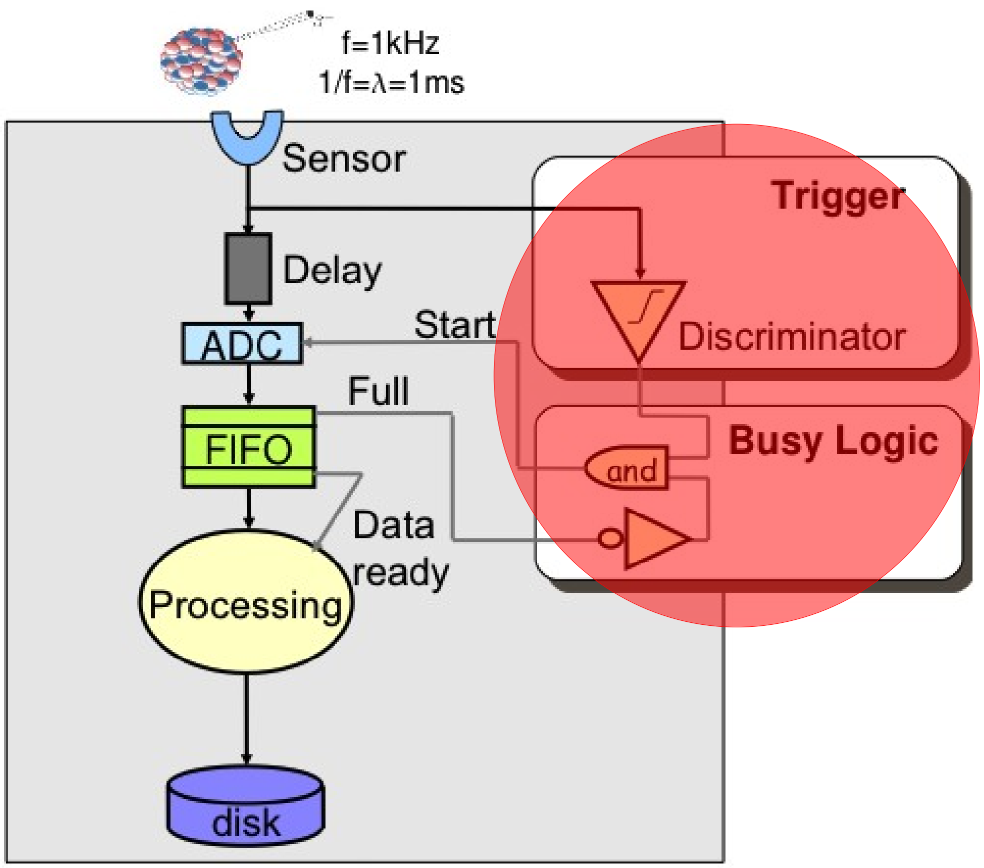






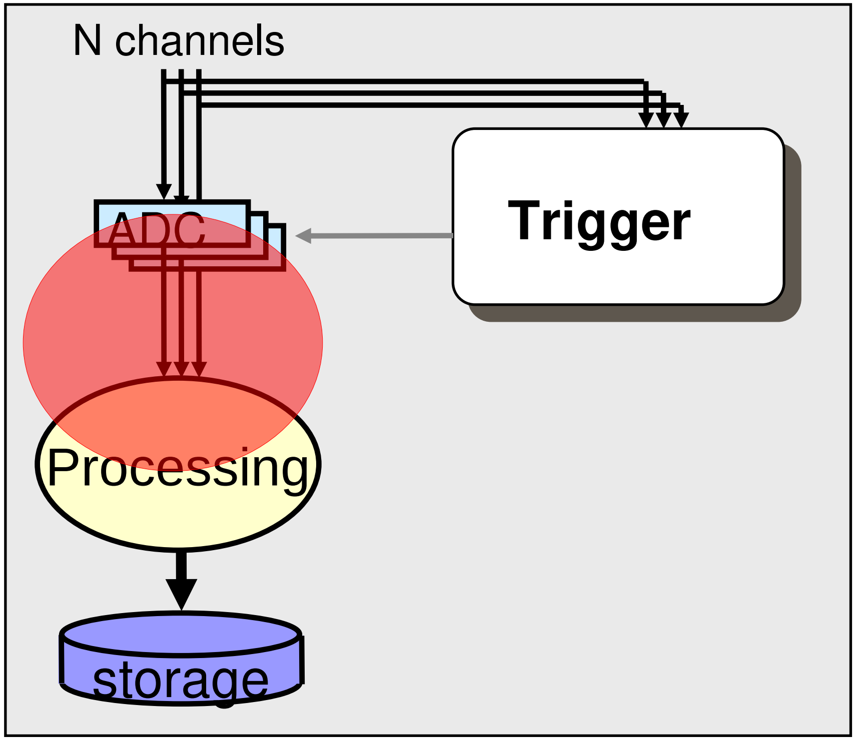
→ trigger latency

- simple trigger: ~fast
- complex trigger logic: not obvious [even when all in hw]
- some trigger detectors may be far away / slow → latency
- trigger signal is one: all information at single point
  - in one step: too many cables
  - in many steps: delays



→ discrete modules: ~ 5-10 ns delay → tot. latency ≥ 20-30 ns ←

bottleneck: readout



- single-event readout can be too slow even w/ DMA block transfer

- exploit:
  - multievent buffering on front-end elx
  - multievent readout
  - concurrent access to front-end buffers (ADC writing and processor reading at same time)

DREAM/RD52 (2006→): a testbeam case

R&D on dual-readout calorimetry, setup:

- Calorimetes prototypes (mainly lead/copper fibre-sampling matrices)
- Scintillator arrays as shower leakage counters
- Trigger/veto/muon counters
- Precision chamber hodoscope → Si beam telescope

… always evolving

acquiring: waveforms, total charge, time information

DREAM/RD52 (2006→): a testbeam case

R&D on dual-readout calorimetry, setup:

- Calorimetes prototypes (mainly lead/copper fibre-sampling matrices)
- Scintillator arrays as shower leakage counters
- Trigger/veto/muon counters
- Precision chamber hodoscope → Si beam telescope

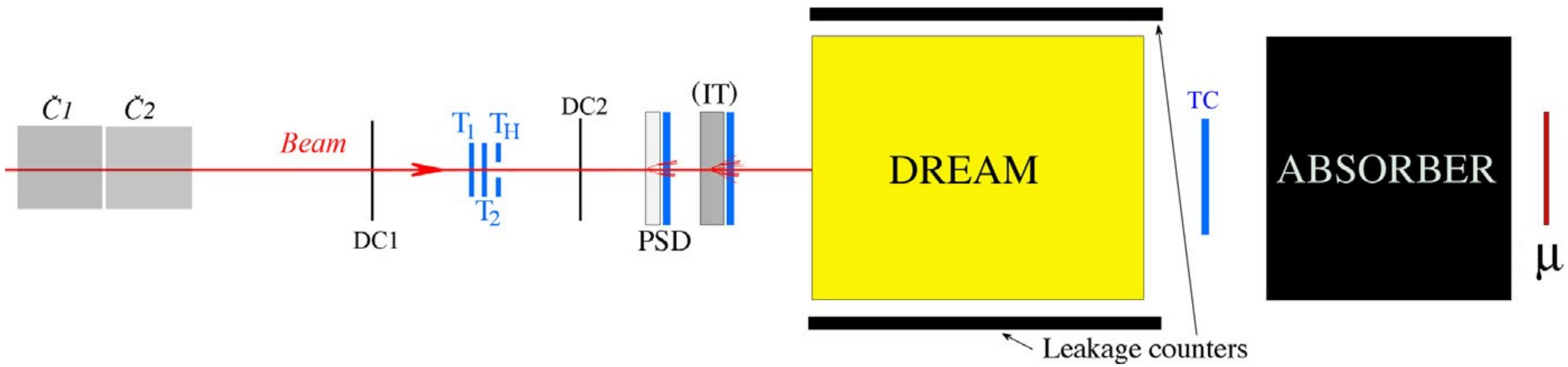
… always evolving

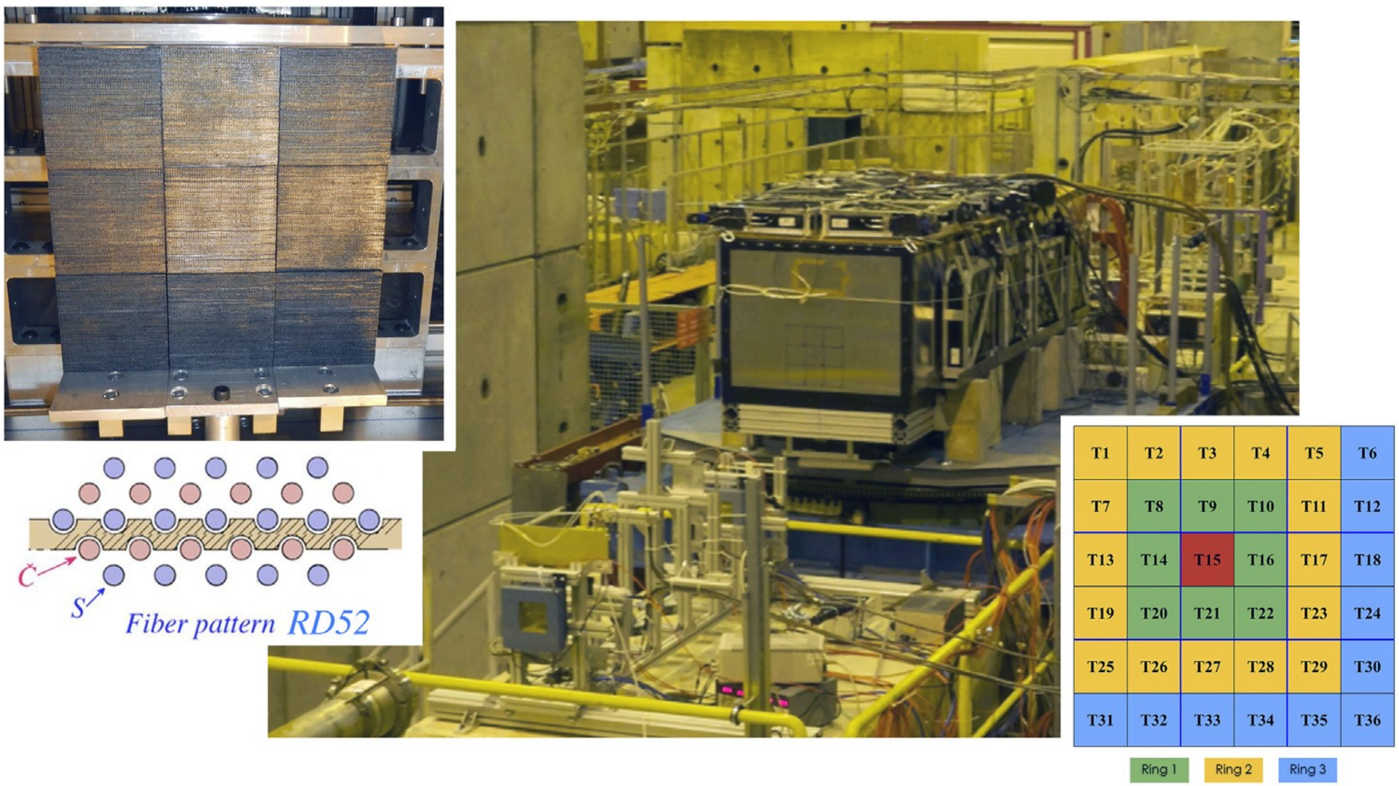
sometime running with 2 or even 3 independent DAQ systems

→ trigger and busy signals used for DAQs' synchronisation

→ offline event building

DREAM/RD52: fibre-sampling prototype





DREAM/RD52 (2006→): a testbeam case

typical SPS cycle:

∼2 s "continuous" beam (SPS flat top) arriving every 14.4 s

Trigger = $\overline{V} \times T_1 \times T_2$ **| ped** → easy !

readout system

1 PC $\rightarrow$ readout of 2 VME crates (via CAEN optical interfaces)
1 PC $\rightarrow$ storage

6 × 32 ch QDCs + TDCs $\rightarrow$ CAEN V792, V862, V775

1 × 34 ch (5 Gs/s) digitiser $\rightarrow$ CAEN V1742
(single event: $\sim$34×1024×12bit)

1 × 4 ch (20 Gs/s) oscilloscope $\rightarrow$ Tektronix TDS 7254B

… few VME I/O & discriminator boards

… all in the control room

readout system

Digitisation ~ 10 μs/event

Data readout (VME → PC) ~ 200-300 μs/event

→ VME readout latency limiting factor

dataflow

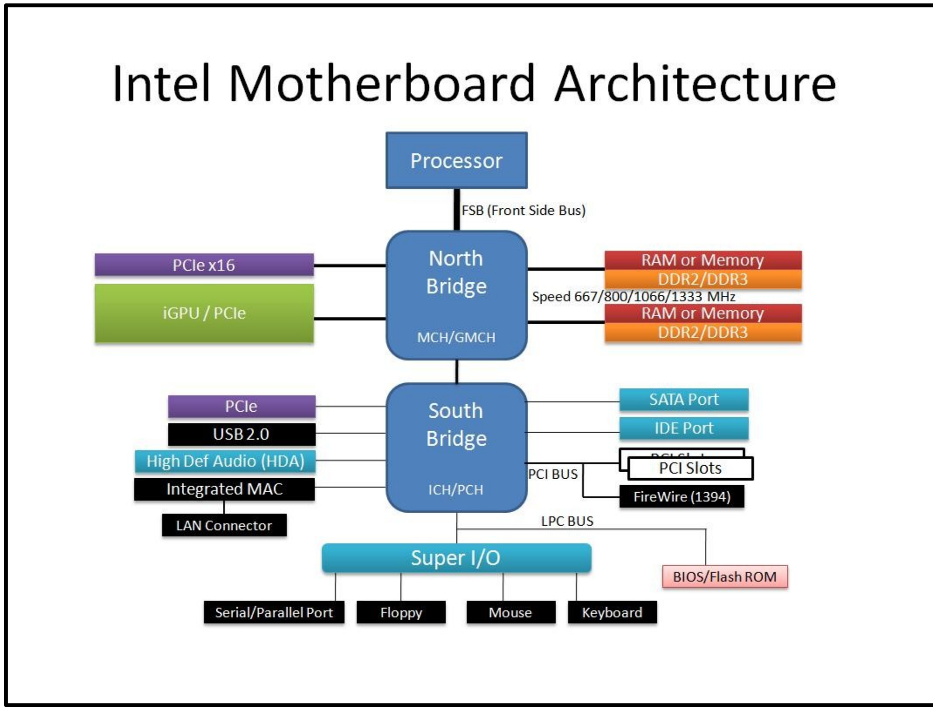
1) Pull mode → FE electronics waiting for PC readout
(self-blocking trigger, re-enabled after readout)

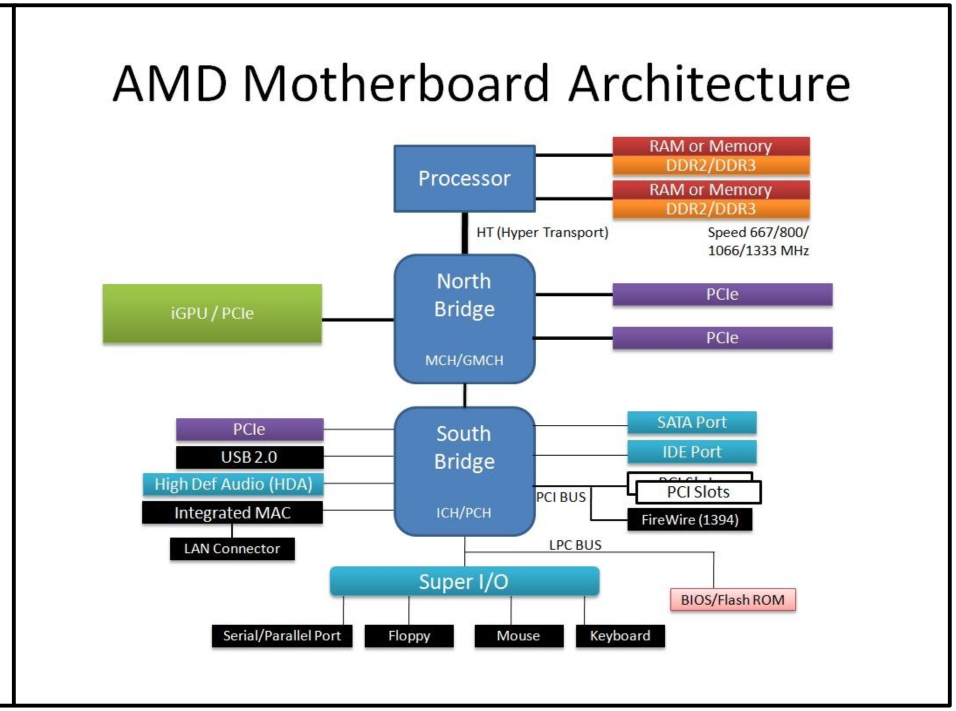
2) Block data transfer → DMA (Direct Memory Access)
data moved by specialised hw (not by CPU)

[Push mode → FE electronics sending data as soon as available]

off-topic: computer architecture

main actual implementations

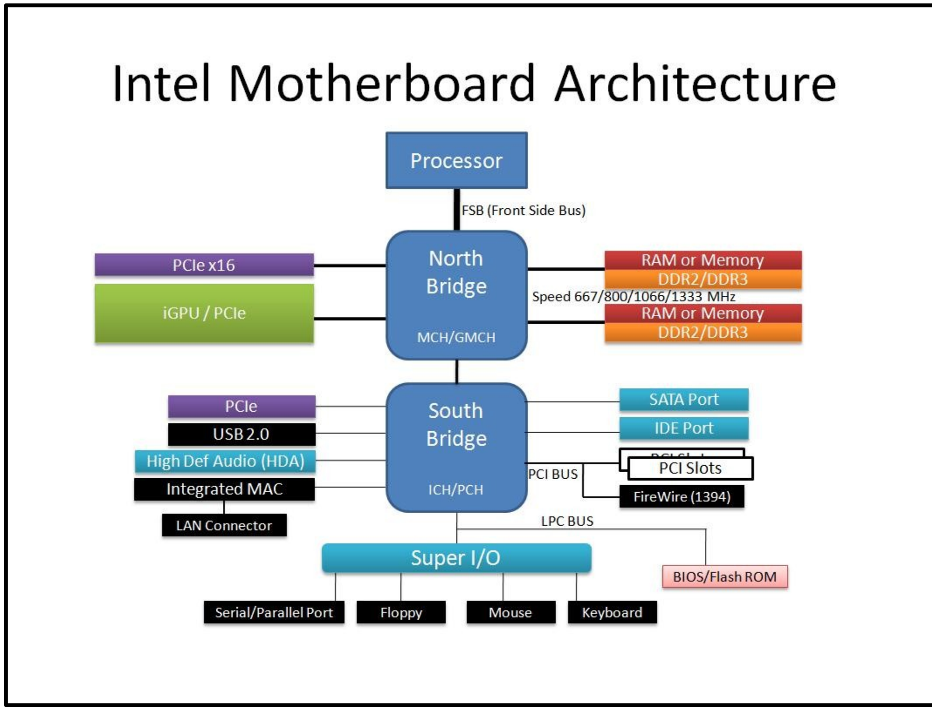


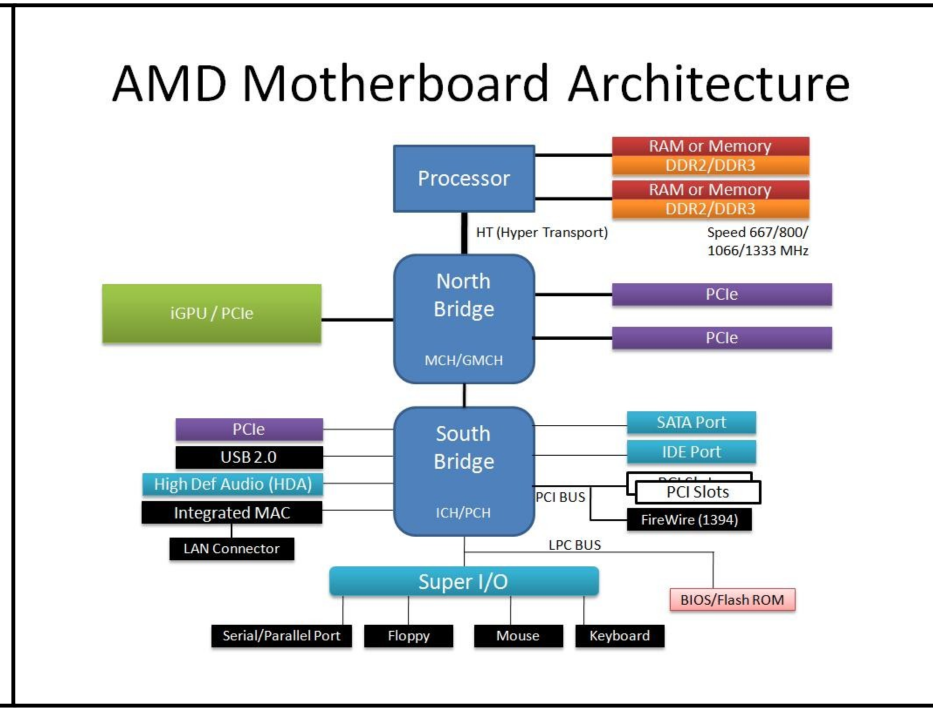


North Bridge: graphics and memory controller hub
South Bridge: I/O controller hub

off-topic: computer architecture

main actual implementations





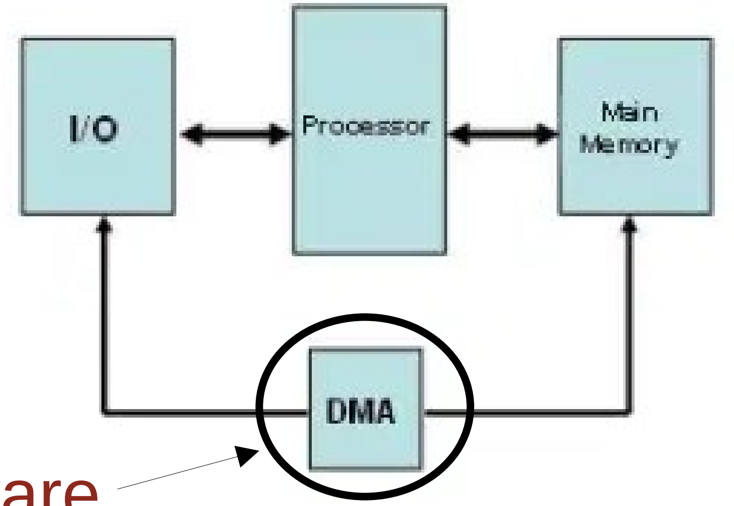
→ is really tuned for data acquisition ?


Well, nobody's perfect

reminder (Markus's talk): block transfer

DMA (direct memory access):

1) load source address (can be FIFO)
2) load destination address (can be FIFO)
3) load size (or until "data-available")
4) run



needs specialised hardware

DREAM DAQ

DAQ logic spill-driven (no “real time”, SLC desktops)

in-spill (slow extraction)

poll trigger signal … if trigger present:

a) (block) read all VME boards

b) format & store on large buffers (FIFO over RAM)

c) re-enable trigger

out-of-spill

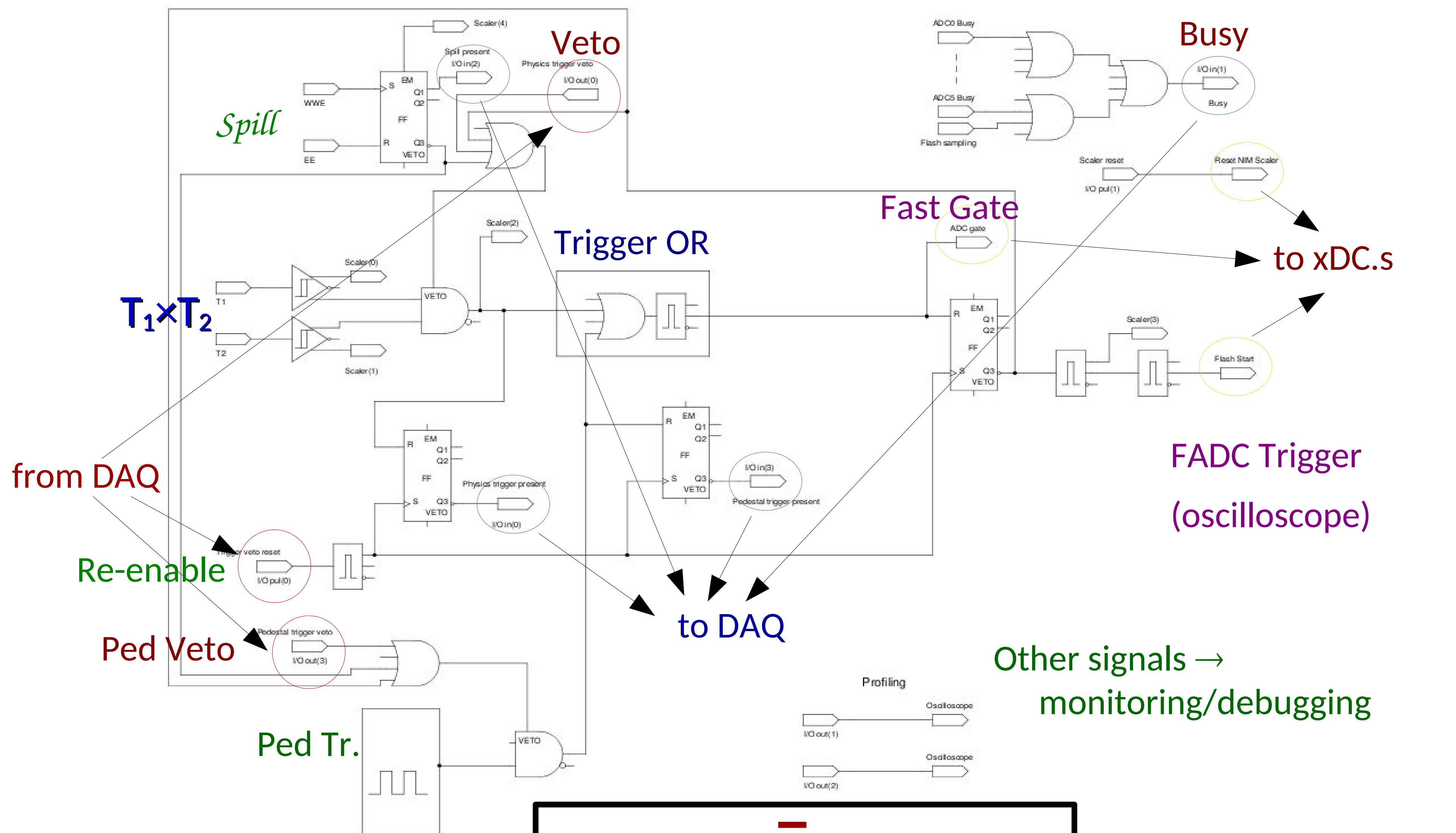
a) read scope (in case) → event size fixed at run start

b.1) flush buffers to disk (beam and pedestal files) over network

b.2) monitor data (produce root files)

rate ~ O(1 kHz) limited by DAQ readout

spill-driven (asynchronous) trigger



Trigger = $\overline{V} \times T_1 \times T_2$ | ped

showstopper: trigger system

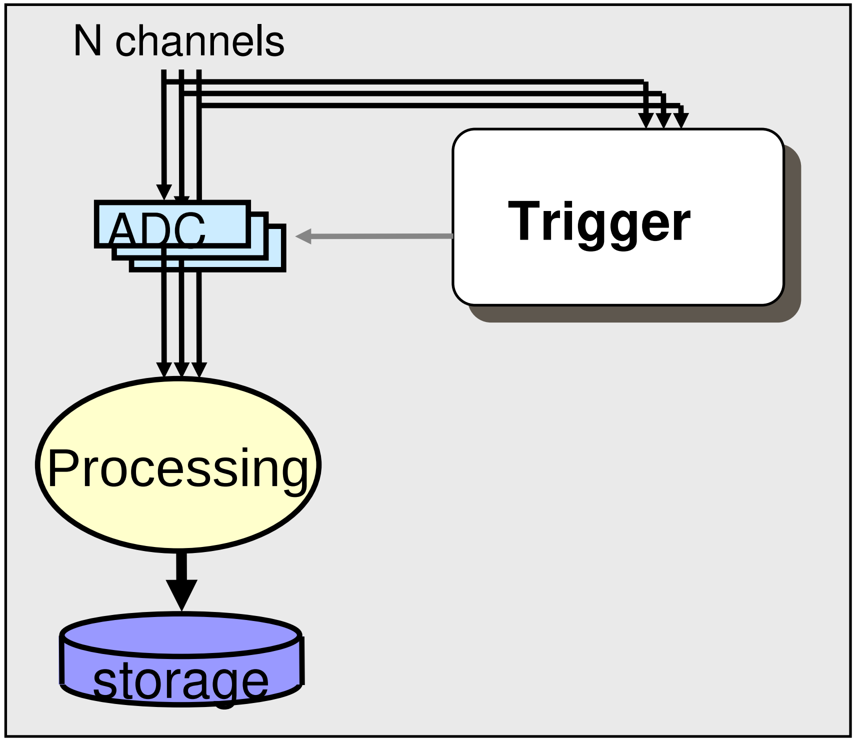
a) crystals w/ fast PMT.s
b) no analog buffering

$\rightarrow$ low-latency trigger

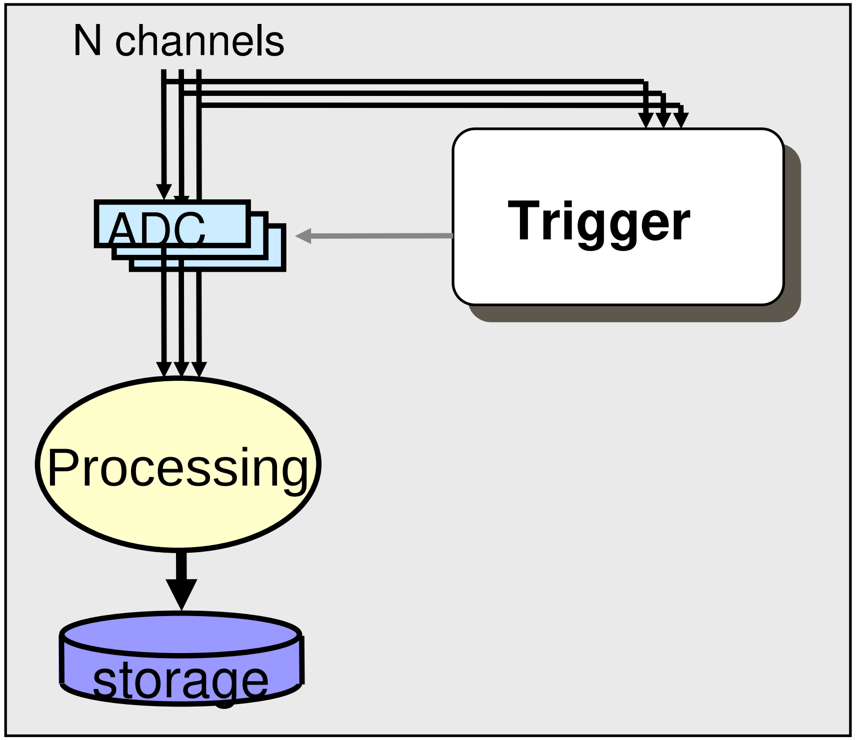
first discrete, then FPGA
(Xilinx Spartan 3AN evaluation board)

step two: increase # of sensors

- More granularity at the physical level
- Multiple channels (usually with FIFOs)
- Single, all-HW trigger
- Single processing unit
- Single I/O

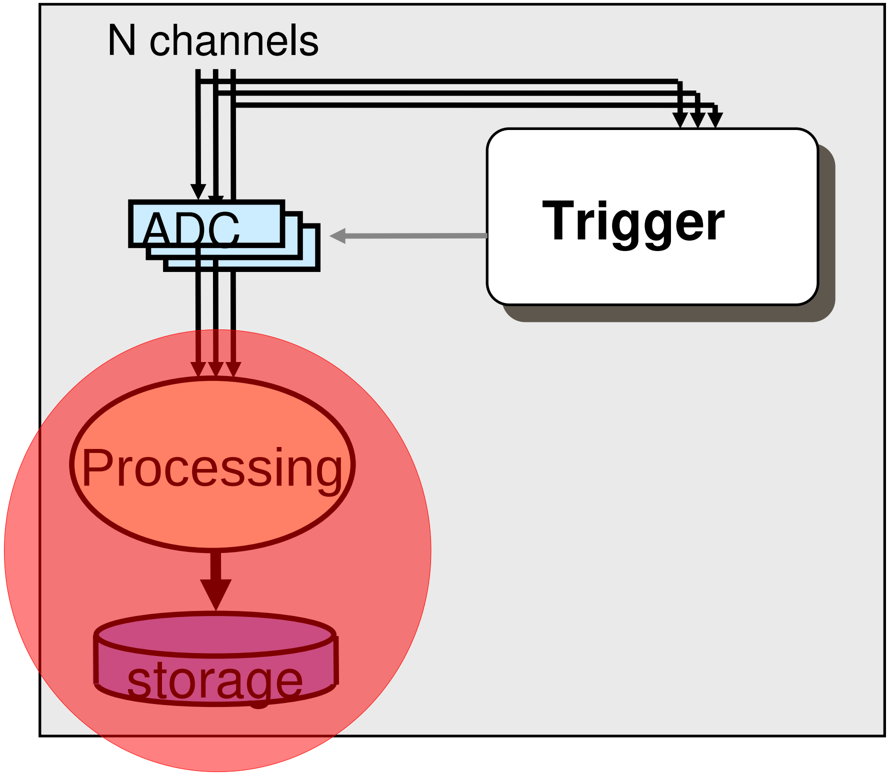


multi-channel, single-PU system



- common architecture in test beams and small experiments
- often rate limited by (interesting) physics itself, not TDAQ system
- or by the sensors

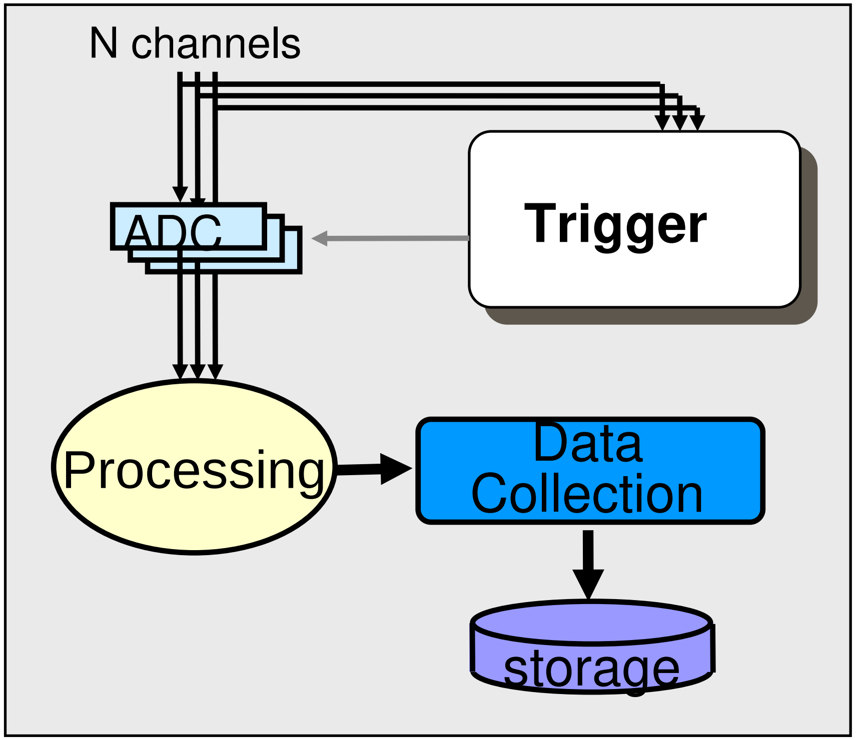
bottlenecks: PU and storage



- a single PU can be a limit
  - collect / reformat / compress data can be heavy
  - simultaneously writing storage
- final storage too:
  - VME up to 50MB/s
    → 1TB in 6h
    too many disks in a week!

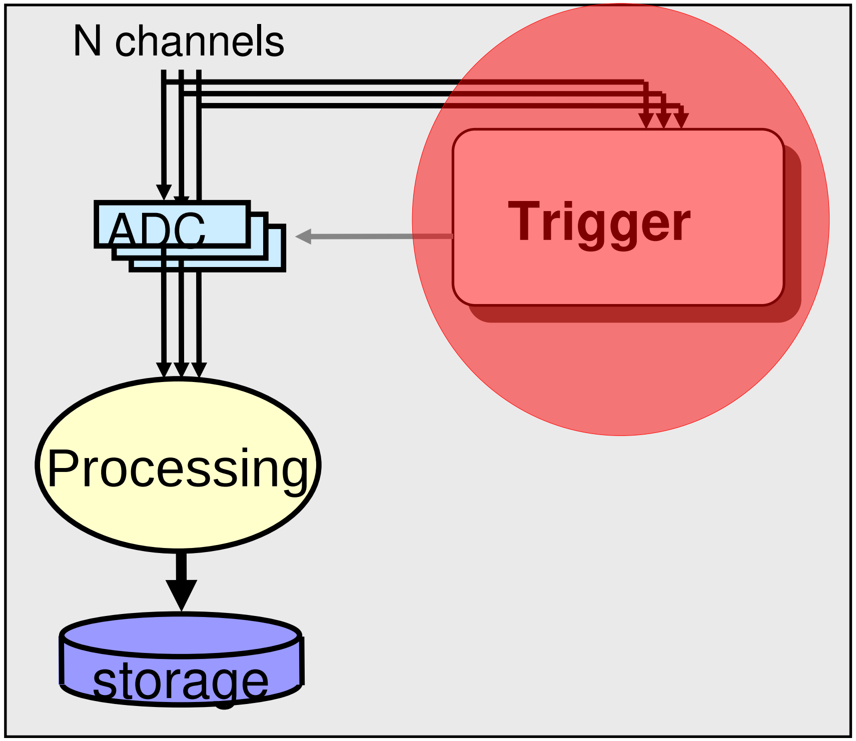
Laptop SATA disk: ~100 MB/s
USB2: ~60 MB/s

→ decouple storage from PU



- data transfer data → dedicated “Data Collection” unit to format, compress and store
- more room for smarter processing or decreased deadtime on non-buffered ADCs

bottleneck: trigger



- to reduce data rates (to avoid storage issues)

  → non-trivial trigger

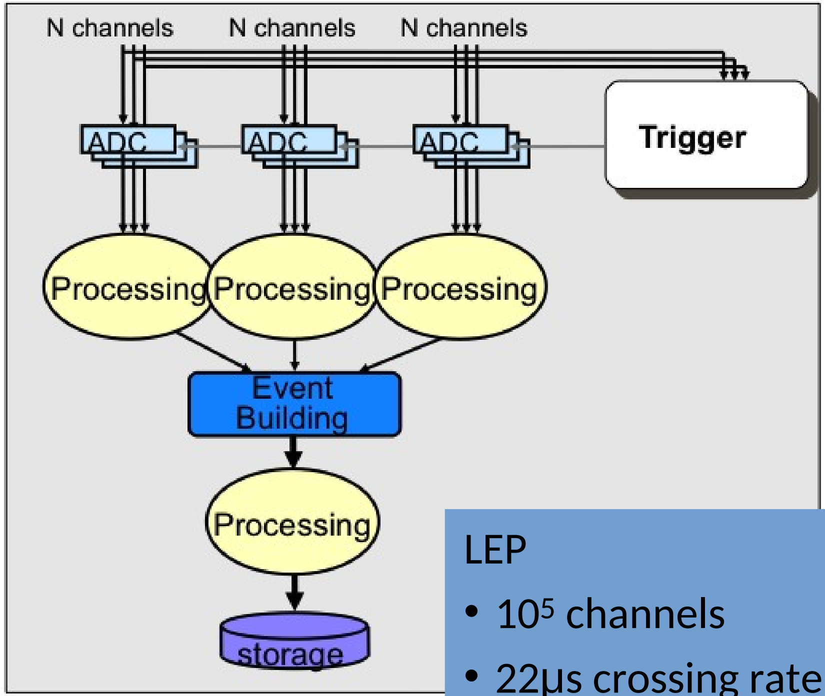
- complexity may already hit manageability limits for discrete logic (latency!)

- integrated, programmable logic came to rescue (FPGA)

  → latency may go down to O(few ns)

step three: multiple PUs (SBC)

- e.g.: CERN LEP experiments
- complex detectors, moderate trigger rate, very little background
- little pileup, limited channel occupancy
- simpler, slow gas-based main trackers

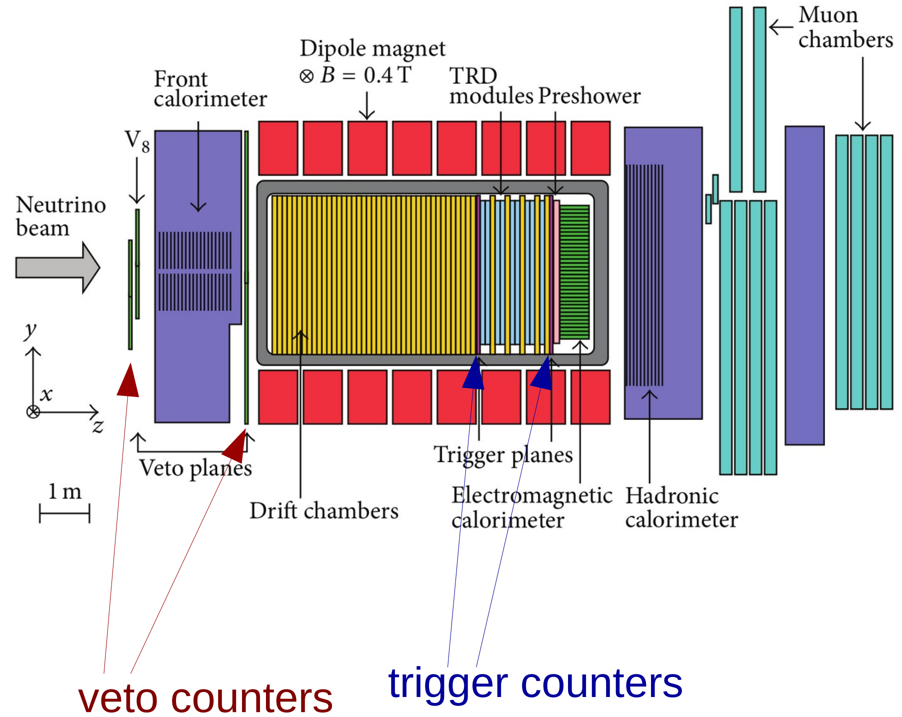


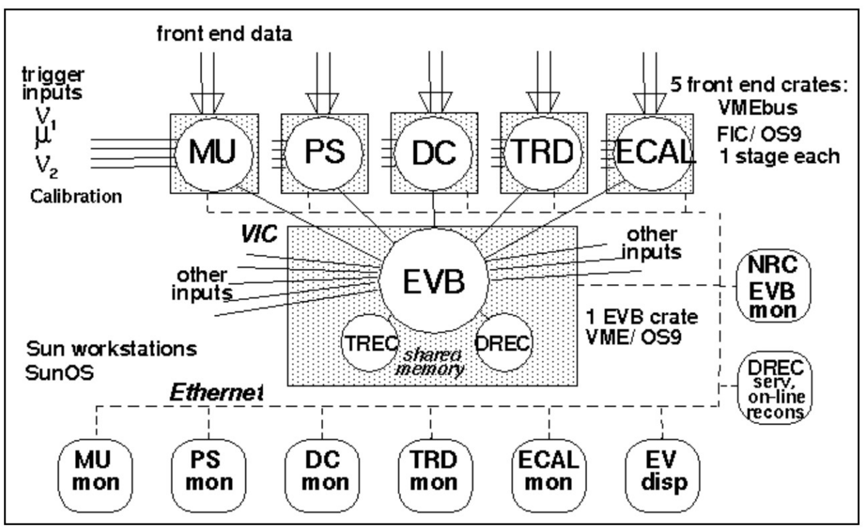
LEP

- 10^5 channels
- 22µs crossing rate
  - –no event overlap
- single interaction

NOMAD (1995-1998)

- Search for $\nu_\mu \rightarrow \nu_\tau$ oscillations at the CERN West-Area Neutrino Facility (WANF)
- 2.4×2.4 m^2 fiducial (beam) area
- Two 4 ms spills with 1.8×10^{13} P.o.T. each (ν spills)
- One (2s) slow-extraction spill (μ spill)
- 14.4s cycle duration





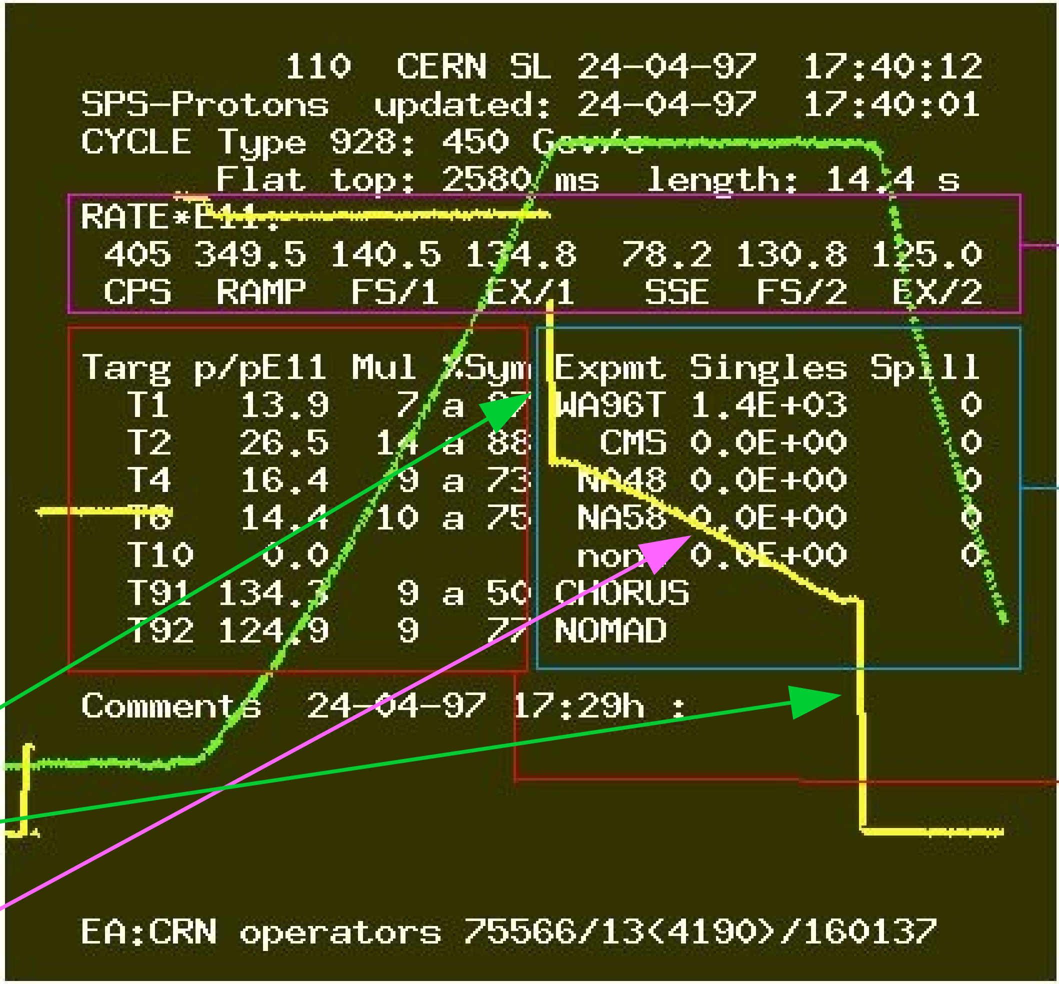
→ DAQ layout

WANF - SPS SuperCycle

14.4 s cycle length

2 × 4 ms neutrino spills (f/s extractions)

1 × 2 s muon spill (slow extraction)



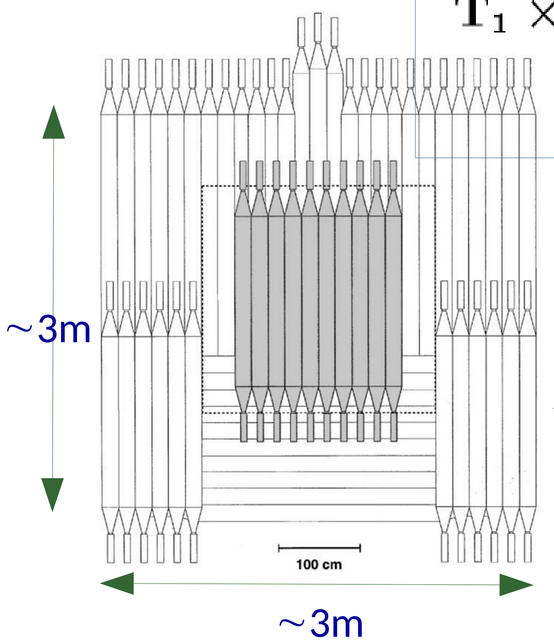
f/s extractions

slow extraction

triggering once more ...

menu for NOMADs:

ν-spill triggers	μ-spill triggers
$\overline{V} \times T_1 \times T_2$	$V \times T_1 \times T_2$
$\overline{V_8} \times FCAL$	$V_8 \times T_2$
$\overline{V_8} \times FCAL' \times T_1 \times T_2$	$V_8 \times T_1$
$\overline{T_1 \times T_2} \times ECAL, \overline{V_8} \times ECAL$	$V_8 \times T_1 \times T_2 \times FCAL'$
RANDOM	$V \times T_1 \times T_2 \times ECAL$

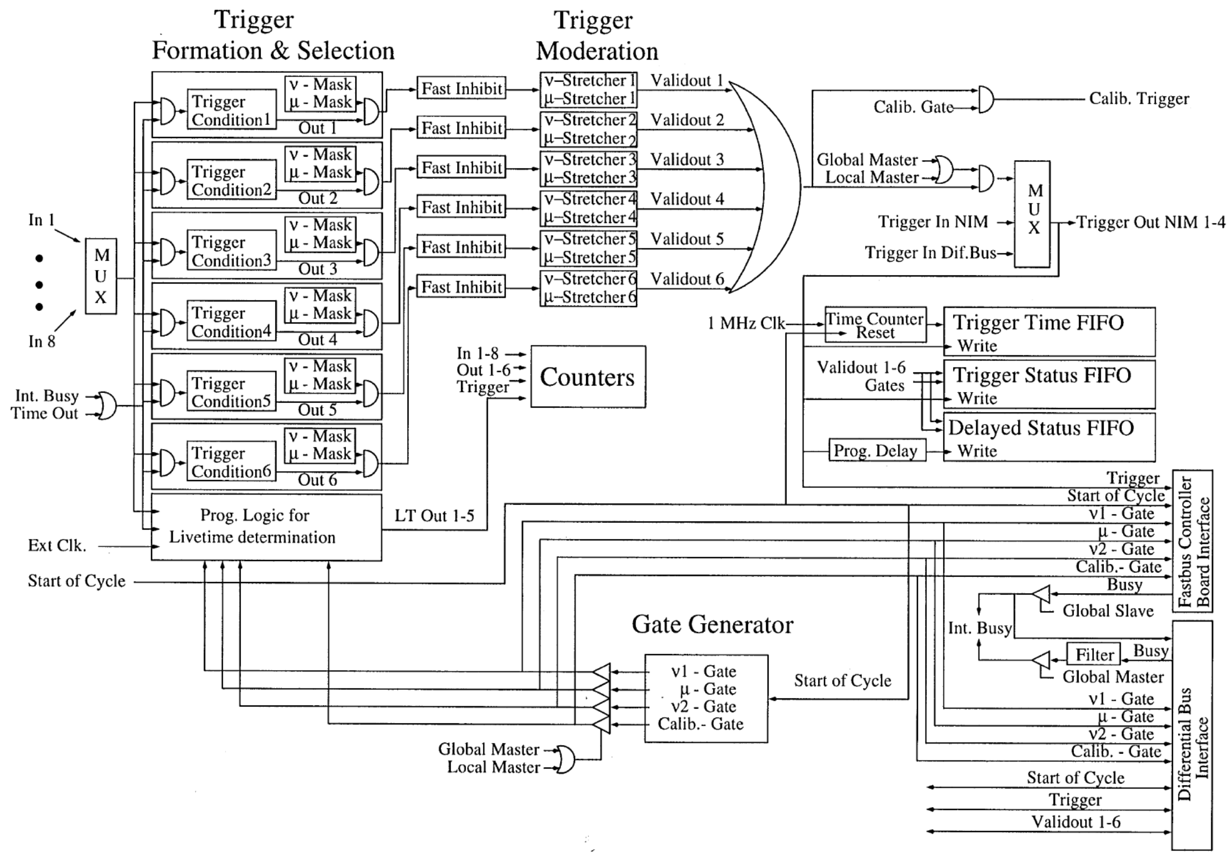


veto counters (central shaded area is V8)

triggering → FPGAs at work

MOdular TRIgger for NOmad (MOTRINO):

6 VME boards providing local and global trigger generation and propagation



DAQ

- FASTBUS digitisers:
  - ~200 (either 64 or 96 channel) xDC boards [x=Q,P,T]
  - O(≥ 2 us) conversion time, 256 event buffers

- VME readout and processing:
  - Motorola 68040 FIC8234 (OS9 real-time system) VME PUs
  - 5 for readout + 1 for event building

- Typically
  - ~4 kHz of neutrino triggers (~15 evts in each 4ms spill)
  - ~30 Hz of muon triggers (~60 evts in each 2s spill)
  - 256-events in off-spill calibration cycles (calibration triggers)

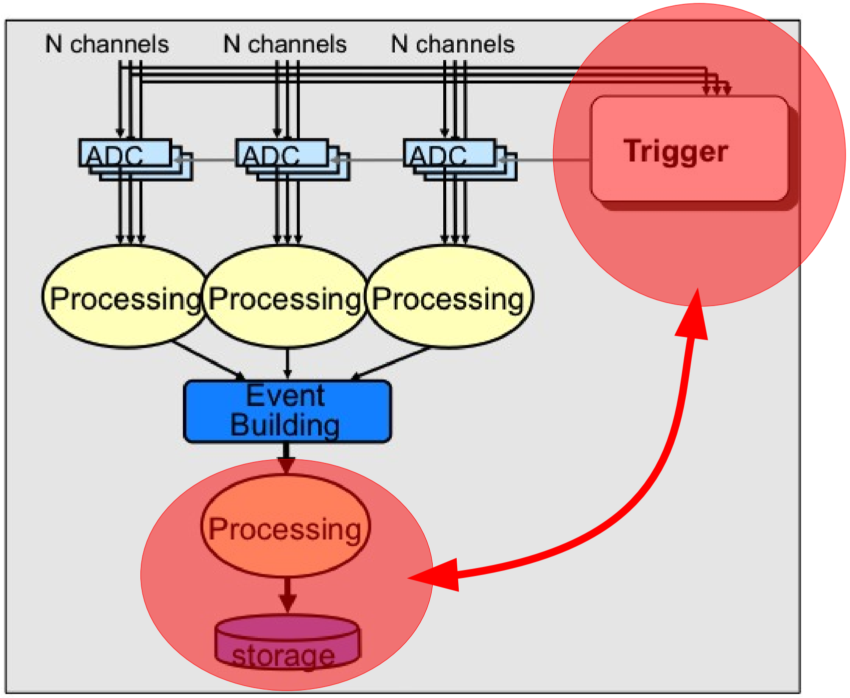
readout sequence

- On-spill on-board buffering
- Off-spill (i.e. off-beam) data transfer and processing
  - on spill (or calibration cycle): on-board event buffering (no way to read event by event)
  - end of spill (or calibration cycle): block transfer to VME
  - then event building + storage

- monitoring and control on SunOs and Solaris workstations

→ deadtime in ν spills: ∼10% due to ADC digitisation time

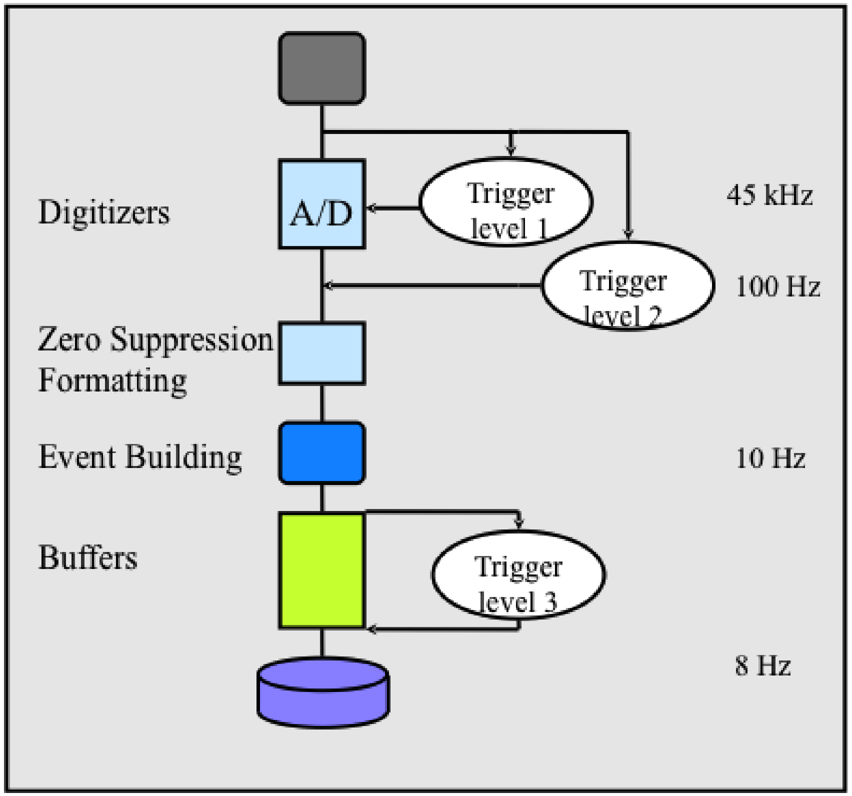
more bottlenecks ?



- trigger complexity ↔ storage
- single HW trigger not sufficient to reduce rate
- add L2 Trigger
- add HLT

step four: multi-level trigger

Typical Trigger / DAQ structure at LEP

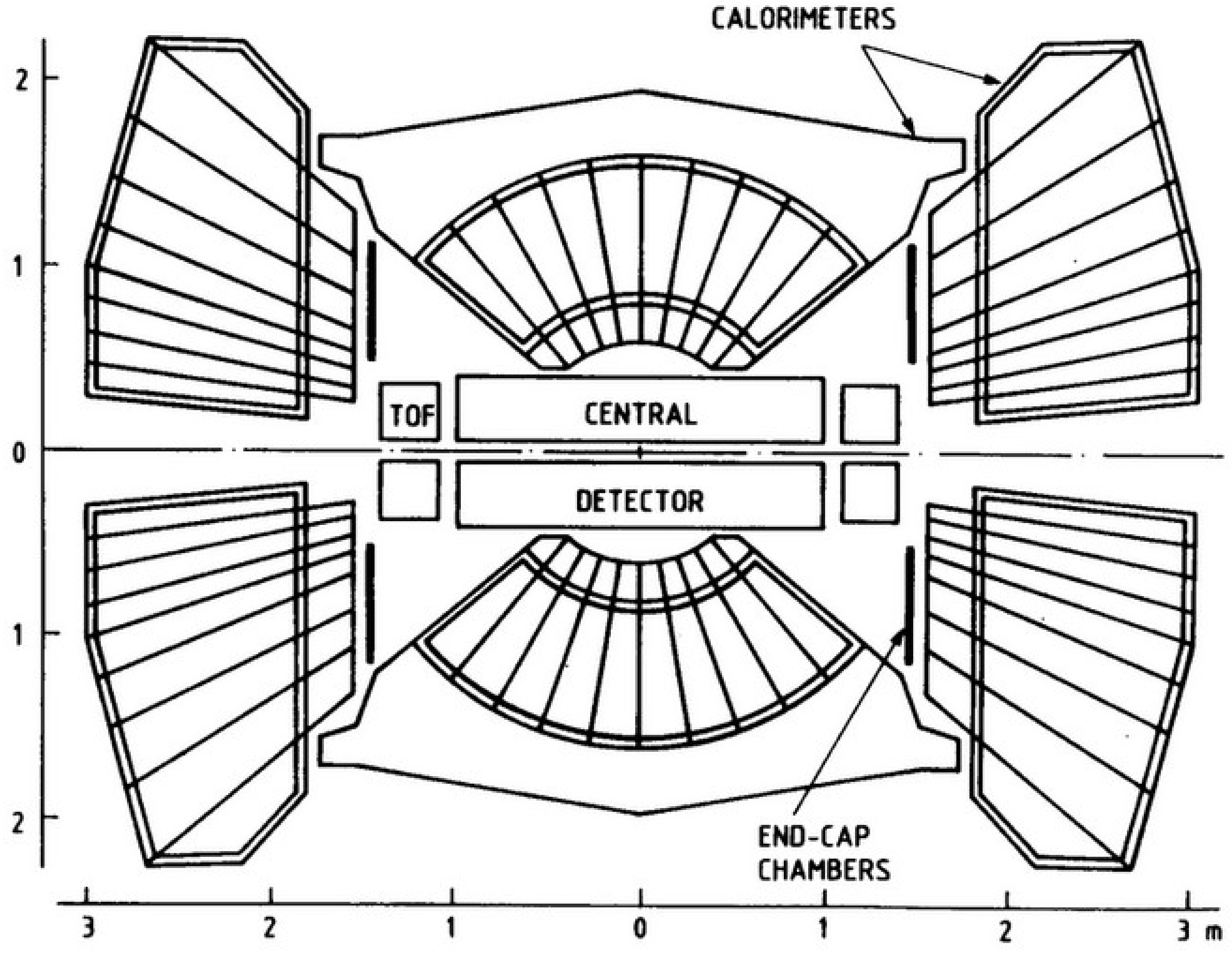


- more complex filters
  - → slower
  - → applied later in the chain

see Trigger lectures

LEP
- 10^5 channels
- 22μs crossing rate
  - –no event overlap
- single interaction
- L1 ~10^3 Hz
- L2 ~10^2 Hz
- L3 ~10^1 Hz
- 100kB/ev → 1MB/s

Upgraded UA2 experiment (1988-1991)



Upgraded UA2 experiment (1988-1991)

High-lumi $p\bar{p}$ collisions @ CERN $p\bar{p}$ collider:

$\sqrt{s}$ = 630 GeV

L = 5×10^{30} cm^{-2} s^{-1} (one order of magnitude increase)

Goal:

W/Z physics

QCD

!! top quark and SUSY discovery !!

Upgraded UA2 experiment (1988-1991)

High-lumi $p\bar{p}$ collisions @ CERN $p\bar{p}$ collider:

$\sqrt{s}$ = 630 GeV

L = 5×10^{30} cm^{-2} s^{-1} (one order of magnitude increase)

Goal:

W/Z physics

QCD

!! top quark and SUSY discovery !!

→ robust theoretical prediction for new physics

Upgraded UA2 experiment (1988-1991)

High-lumi $p\bar{p}$ collisions @ CERN $p\bar{p}$ collider:

$\sqrt{s}$ = 630 GeV

L = 5×10^{30} cm^{-2} s^{-1} (one order of magnitude increase)

Goal:

W/Z physics

QCD

!! top quark and SUSY discovery !!

→ robust theoretical prediction for new physics

unluckily we learnt nature doesn't know physics …

Upgraded UA2 experiment (1988-1991)

High-lumi $p\bar{p}$ collisions @ CERN $p\bar{p}$ collider:

$\sqrt{s}$ = 630 GeV

L = 5×10^{30} cm^{-2} s^{-1} (one order of magnitude increase)

Goal:

W/Z physics

QCD

!! top quark and SUSY discovery !!

→ robust theoretical prediction for new physics

unluckily we learnt nature doesn't know physics …

uh … "physics" means "nature" …

Upgraded UA2 experiment (1988-1991)

High-lumi $p\bar{p}$ collisions @ CERN $p\bar{p}$ collider:

$\sqrt{s}$ = 630 GeV

L = 5×10^{30} cm^{-2} s^{-1} (one order of magnitude increase)

Goal:

W/Z physics

QCD

!! top quark and SUSY discovery !!

Complex trigger signatures:

em and hadronic isolated showers, missing E_T (?)

Upgraded UA2 experiment (1988-1991)

High-lumi $p\bar{p}$ collisions @ CERN $p\bar{p}$ collider:

$\sqrt{s}$ = 630 GeV

L = 5×10^{30} cm^{-2} s^{-1} (one order of magnitude increase)

Goal:

W/Z physics

QCD

!! top quark and SUSY discovery !!

Complex trigger signatures:

em and hadronic isolated showers, missing E_T (?)

local trigger signals

can't be local !

Upgraded UA2 experiment (1988-1991)

Three-level trigger selection:

- L1 from on-detector hardware
- L2 over dedicated “hardware” processors
- L3 over Motorola 68020 processors (ALEPH event builder)

DAQ readout & monitoring:

- CAMAC & FASTBUS (HEP buses)
  - → VAX/VMS real-time platforms

Upgraded UA2 experiment (1988-1991)

Three-level trigger selection:

L1 from on-detector hardware

L2 over dedicated “hardware” processors

L3 over Motorola 68020 processors (ALEPH event builder)

DAQ readout & monitoring:

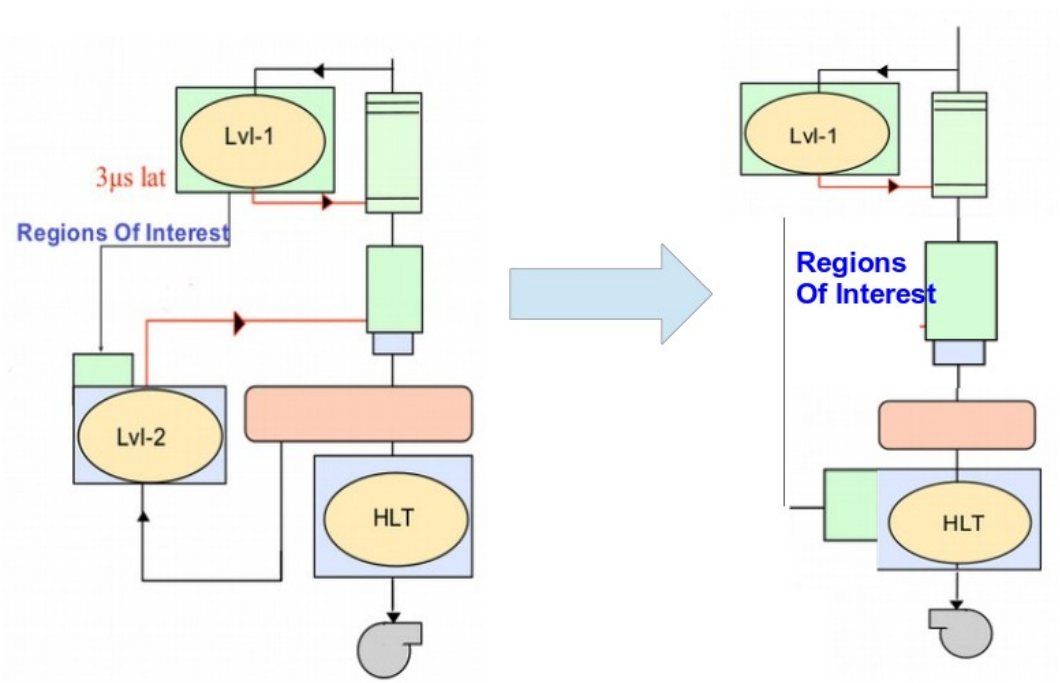
CAMAC & FASTBUS (HEP buses)

→ VAX/VMS real-time platforms

Just improved limits on new physics but many new/better SM measurements:

gluon structure function, α_S, W mass, ...

ATLAS (from Run 1 to Run 2)



→ Merge L2 and L3 into single HLT farm

– preserve Region of Interest but dilute farm separation and fragmentation

– increase flexibly, computing power efficiency

trigger/event-selection latencies

Possible (e.g. ATLAS Run 1) values:

- L1 : O(1 μs in real-time) → let say = 1.9 μs
- L2 : O(10 ms) → let say = 40 ms
- L3(HLT) : O(s) → let say = 1 s

Q: do the 3 numbers mean the same thing ?

latency and real-time

real time: system must respond within some fixed delay

→ *Latency = Max Latency*

→ over fluctuations bad, will create deadtime

non-real-time: system responds as soon as it's available

→ *Latency = Mean Latency*

→ over fluctuations fine, shouldn't create deadtime

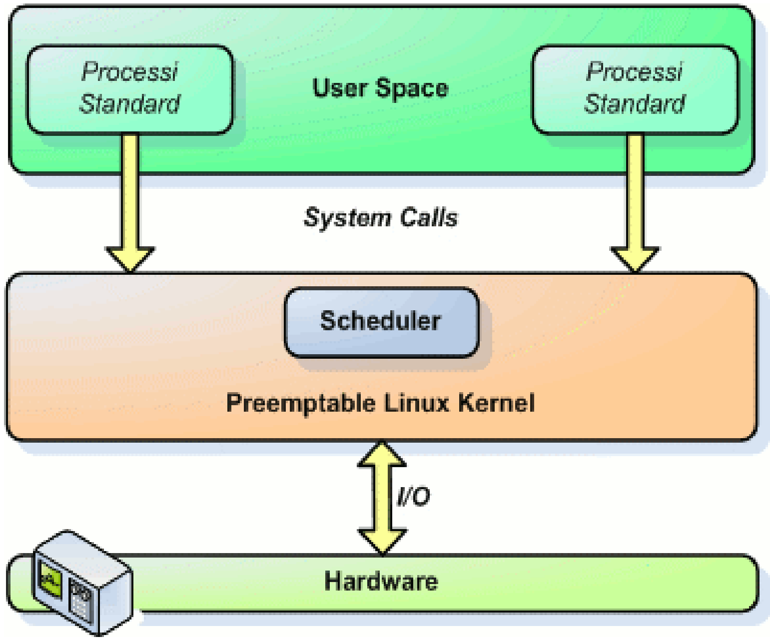
real time o.s. :

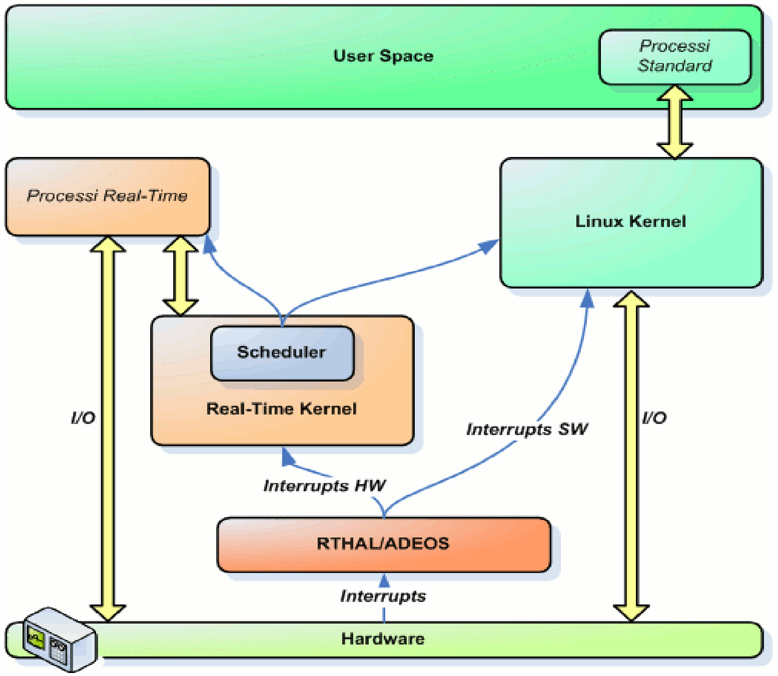
very stable time delay in responding to events

standard unix kernels are not real time:

a system call can in principle take any time

off-topic: real-time linux





Low-latency Ubuntu patch

(soft real time) :

Interruptible linux kernel

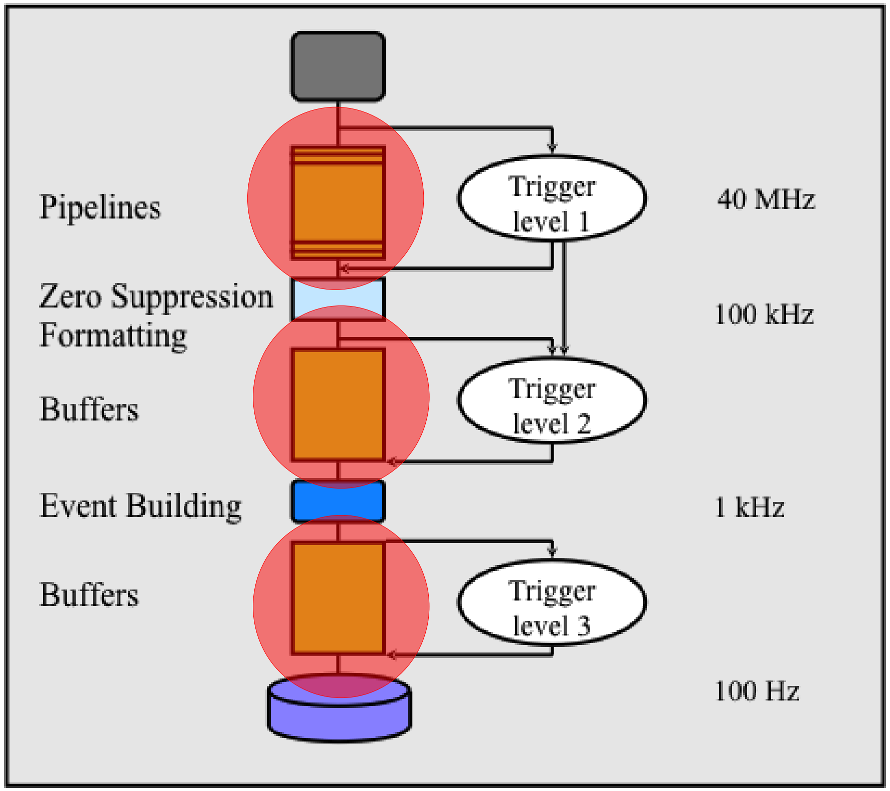
https://help.ubuntu.com/community/UbuntuStudio/RealTimeKernel

RTAI (hard real time) :

linux kernel as high-priority application

https://www.rtai.org/

step five: dataflow control



- Buffers:
  - not “final solution”
  - can overflow due to:
    - bursts
    - unusual event sizes
    - too large “processing” time

Who controls the flow?

FE (*push*) or EB/Readout (*pull*)

push vs. pull

- Push (bottom sends upstream):

  a) Data fragments sent to next level ("readout") as soon as possible

  b) If "readout" goes full/busy, it must exert backpressure to block dataflow (e.g., with "XON/XOFF" mechanism)

  c) Fragments/events may be lost by "readout"

- Pull (top asks downstream):

  a) Data fragments are read out by "readout" elements as soon as possible

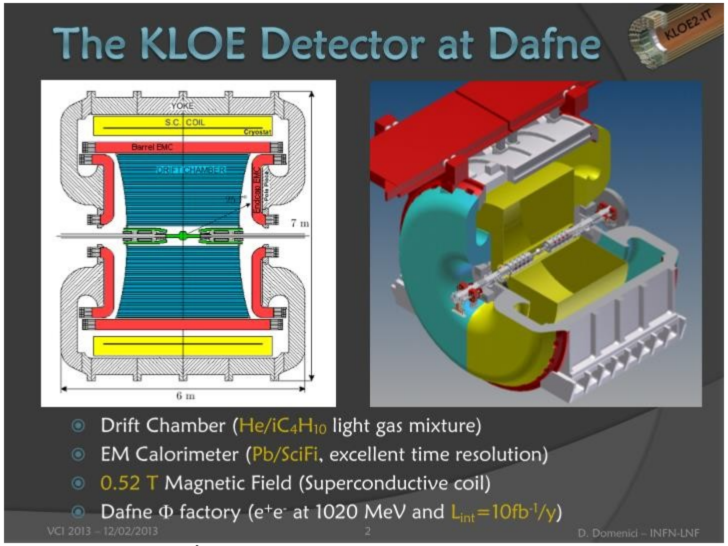
  b) If FE elx goes full/busy, it may/must block triggering (e.g. assert BUSY signal) --- it may just ignore new triggers/events

  c) In any case, new events will NOT be properly acquired

- At some point, new events will be either fully or partially lost (loss of DAQ efficiency)

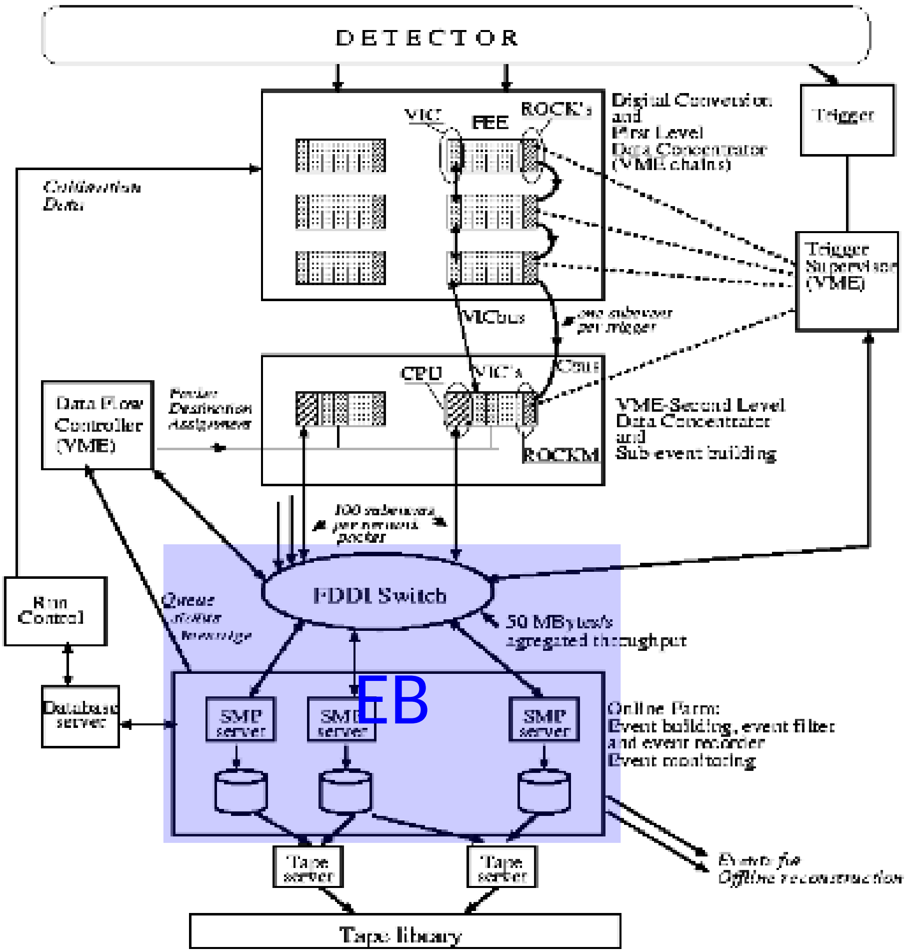
a *push* example: KLOE

- DAΦNE e^+e^- collider in Frascati
- CP violation parameters in the Kaon system
- “factory”: rare events in a high-rate beam



- 10^5 channels
- 2.7 ns crossing rate
  - rarely event overlap
  - “double hit” rejection
- high rate of small events
- L1 ~10^4 Hz
  - 2 μs fixed deadtime
- HLT ~10^4 Hz
  - ~COTS, cosmic rejection only
- 5 kB/ev → 50 MB/s [design]

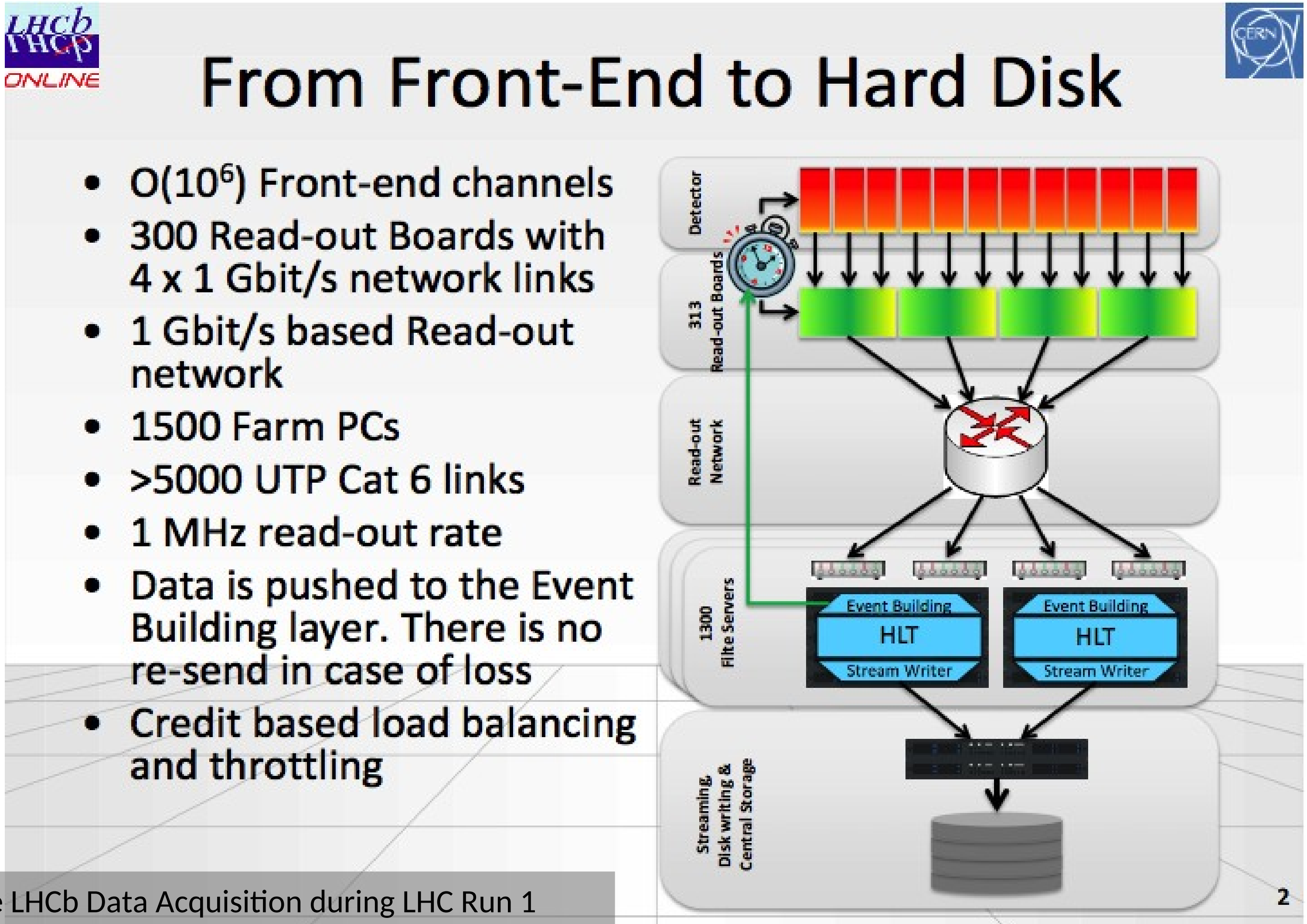
KLOE



- deterministic FDDI network
- buffering at all levels (from FE to EB)
- *push* architecture
  vs pull used in ATLAS
  see DAQ Software lecture
- try EB load redistribution before resorting to backpressure

Which LHC experiment has a somewhat similar dataflow architecture ?

LHCb: network is dataflow



The LHCb Data Acquisition during LHC Run 1
CHEP 2013

more info in "TDAQ for the LHC experiments"

ATLAS TDAQ in Run 2

~ 2 MB events, ~ 50 GB/s network bandwidth,
~ 1.5 GB/s recording throughput

Custom point-to-point links

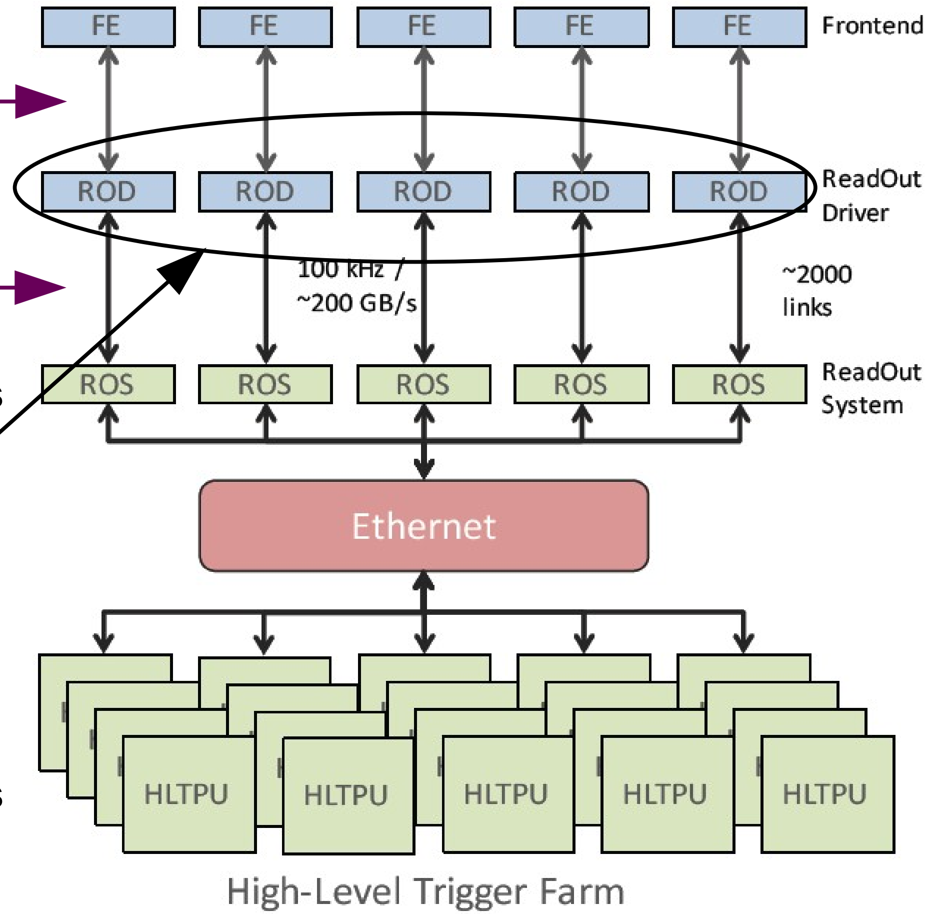
Point-to-point S-LINKs*

VME (detector-specific)

~1000 boards

~150 servers

~1500 servers



Custom elx components

PCs (COTS)

*S-LINK: CERN Simple Link

ATLAS dataflow

Push mode from front-end elx up to ROS/swROD system

→ data sent as soon as available

Pull mode from ROS to HLT

→ data requested by HLT as soon as HLT is free

⇒ ROS/swROD must handle all critical dataflow issues

looking forward to LS2 and beyond

On some long term, all experiments looking forward to significant increase in L1 trigger rate and bandwidth. ALICE and LHCb will pioneer this path during LS2

DAQ@LHC Workshop

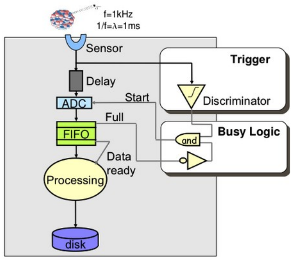
ALICE

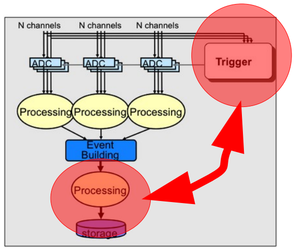
- First level trigger for Pb-Pb interactions **500 Hz** → **50 kHz**
- 22 MB/event
  - **1 TB/s readout** → **<u>500 PB/month</u>**
- Data volume reduction
  - on-line full reconstruction
  - discard raw-data
- Combined DAQ/HLT/offline farm
  - COTS, FPGA and GPGPU

LHCb

- **1 MHz** → **40 MHz** readout and event building → trigger-less
  - trigger support for staged computing power deployment
- 100 kB/event
  - on-detector zero suppression → rad-hard FPGA
  - 4 TB/s event-building

trends





- Integrate synchronous, low latency in front end
  - limitations do not disappear, but decouple (factorise)
  - all-HW implementation
  - isolated in replaceable(?) components
- Use networks as soon as possible

- Deal with dataflow instead of latency
- Use COTS network and processing
- Use "network" design already at small scale
  - easily get high performance with commercial components

take care, lot of issues not addressed:

Hw configuration

Sw configuration

Hw control & recovery

Sw control & recovery

Monitoring

…

Appendices:

A: Students’ homework

(with a brief introduction to Italians’ body language)

B: Cables and transmission lines

C: Backtrace

D: Profiling

E: Some crude queueing theory

F: Trigger qualification

Thank you for your patience …

Appendix A: students’ homework

(with a brief introduction to Italians’ body language)

Students' homework

Students' homework

About the number of Italians in Trigger/DAQ, debate one of the following hypotheses:

Students' homework

About the number of Italians in Trigger/DAQ, debate one of the following hypotheses:

1) baseline: statistical fluctuation or new physics ?
(to be submitted to Nature)

Students' homework

About the number of Italians in Trigger/DAQ, debate one of the following hypotheses:

1) baseline: statistical fluctuation or new physics ?
(to be submitted to Nature)

2) romantic: what about "Italians do it better" ?
(to be submitted to Vanity Fair)

Students' homework

About the number of Italians in Trigger/DAQ, debate one of the following hypotheses:

1) baseline: statistical fluctuation or new physics ?
(to be submitted to Nature)

2) romantic: what about "Italians do it better" ?
(to be submitted to Vanity Fair)

3) last but not least (the "paranoic/complottist" vision):

Students' homework

About the number of Italians in Trigger/DAQ, debate one of the following hypotheses:

1) baseline: statistical fluctuation or new physics ?
(to be submitted to Nature)

2) romantic: what about "Italians do it better" ?
(to be submitted to Vanity Fair)

3) last but not least (the "paranoic/complottist" vision):
what about the famous Mafia-Pizza-Spaghetti-TDAQ connection ?
(will go anonymous on the dark web)

On the other hand …

… please, take care !

you can't afford such a demanding environment *you can't afford such a demanding environment without specific training …*

you can't afford such a demanding environment without specific training …

… about the ***Italians' way***

Luckily

on the web

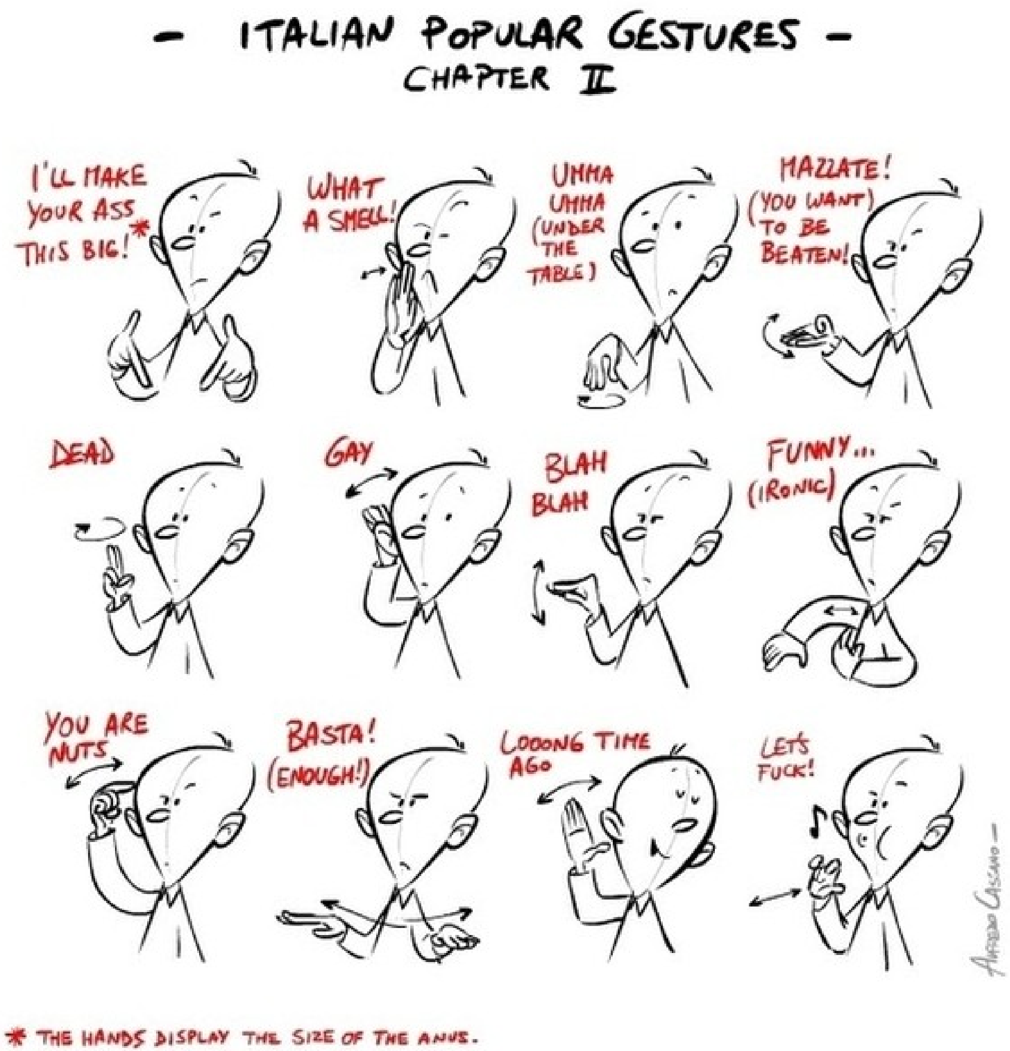
there are plenty of survival kits

Example 1: basic course (mild concepts)

Example 1: basic course (mild concepts)

Example 2: advanced course (includes sensitive concepts)

Example 2: advanced course (includes sensitive concepts)



Please take care:

Please take care:

be careful while doing practice!

(expecially for the advanced course)

Appendix B: Cables and transmission lines

Spoken about signals, amp.s, digitisers, … but ...

… almost nothing about how signals are transmitted over long distances. ***Is there any issue ?***

Q(1): what is a cable (for a single signal) ?

a couple of ideal conductors (R=C=L=0) ?

Q(2): which speed can it reach ?

Q(3): what's its impedance ?

Q(4): what does it to your signal ?

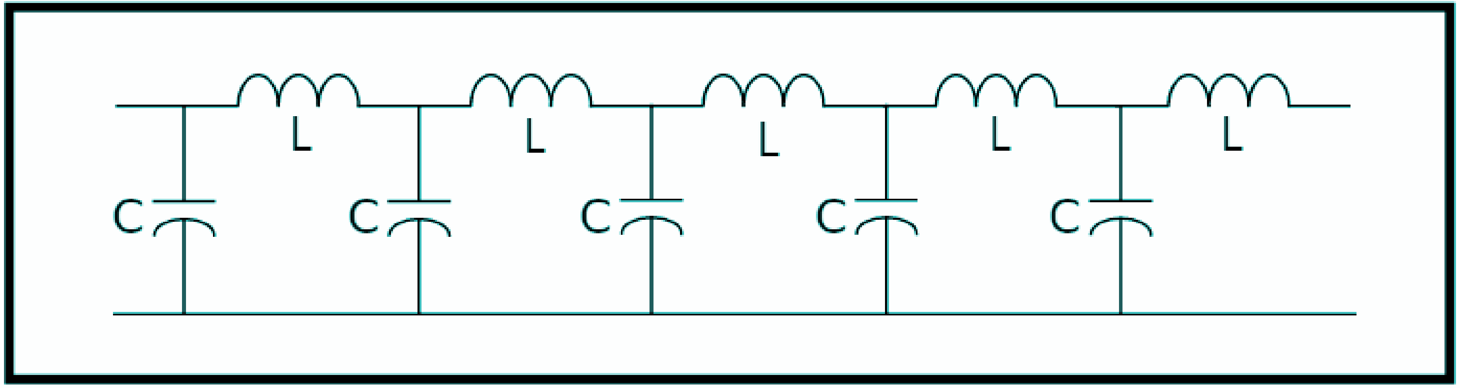
Ok the full line must be properly matched:

Z(out) = Z(cable) = Z(in)

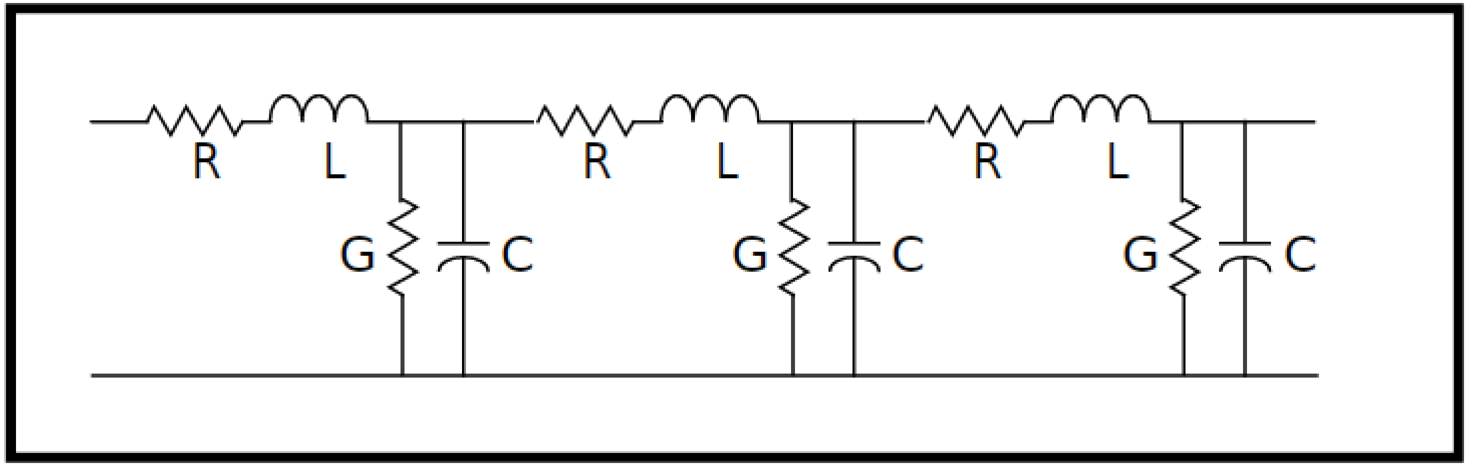
That's all ?

Cables and transmission lines

Lossless transmission line:



Lossy transmission line:

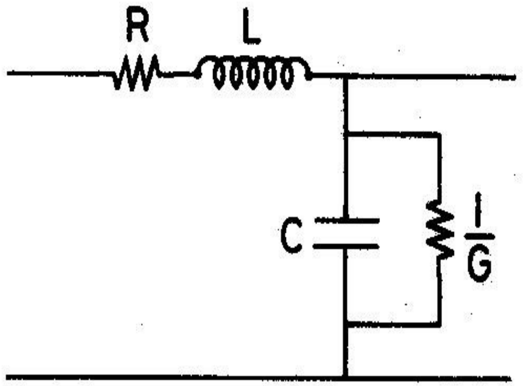


Cables

Cable element (dz):

$$L \approx \frac{\mu}{2\pi} \ln\left(\frac{b}{a}\right) \quad [\mathrm{H/m}]$$

$$C \approx \frac{2\pi\varepsilon}{\ln(b/a)} \quad [\mathrm{F/m}]$$



R depends on the frequency (skin effect)

G should be negligible

$$Z = (L/C)^{1/2}$$

$$v_p = (LC)^{-1/2} = (\mu\varepsilon)^{-1/2}$$

Cables

Equation for standing waves:

$$\frac{\partial^2 V}{\partial z^2} = LC\frac{\partial^2 V}{\partial t^2} + (LG + RC)\frac{\partial V}{\partial t} + RGV$$

solution:

$$\frac{d^2 V}{dz^2} = (R + i\omega L)(G + i\omega C)V = \gamma^2 V$$

$$\gamma = \alpha + ik = \sqrt{(R + i\omega L)(G + i\omega C)}$$

R usually dominated by the skin effect:

R(ω)= r*D/(4*δ)

r = resistance per unit length
D = diameter internal conductor
δ = skin depth ~ 1/√ω

Cable losses

Neglecting the transconductance G:

$$\alpha = R(\omega)/(2Z_0) \sim c\sqrt{\omega}$$

$$k = \omega\sqrt{RC} = \omega/(\beta c)$$

$$V(z,t) = V_1 \exp(-\alpha z)\exp[i(\omega t - kz)]$$

50-Ohm fast (v = 4 ns/m) CERN-store cables:

04.61.11.F - COAXIAL CABLE 50 OHM - TYPE C-50-6-1
04.61.11.H - COAXIAL CABLE 50 OHM - LOW LOSS - TYPE C-50-11-1

f(-3db, 40 m, cable C-50-6-1) ~ 120 MHz
f(-3dB, 40 m, low loss cable) ~ 640 MHz

Signal distortions

Time parameter:

$$\alpha \sim \mu \sqrt{f}$$

$$\tau_0 = (\mu z)^2 / \pi$$

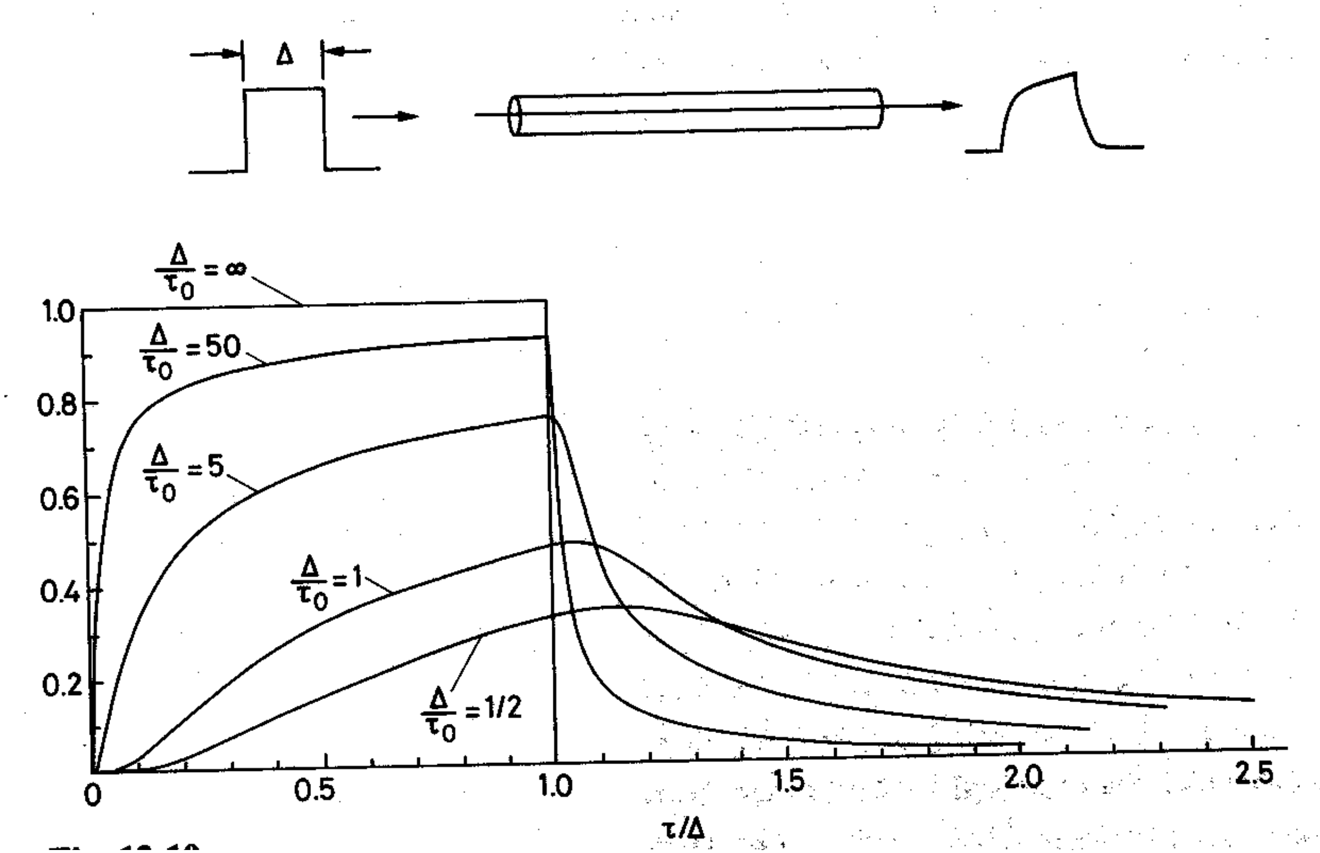
*μz ~ 32*E-6 (C-50-6-1), 14E-6 (low loss cables)*

$$\tau_0 \sim 320\, ns\, (C-50-6-1)$$

$$\tau_0 \sim 60\, ns\, (low\, loss\, cables)$$

*** *Take care: would like* $\tau_0 << \tau$ *(signal)*

Digital pulse distortions



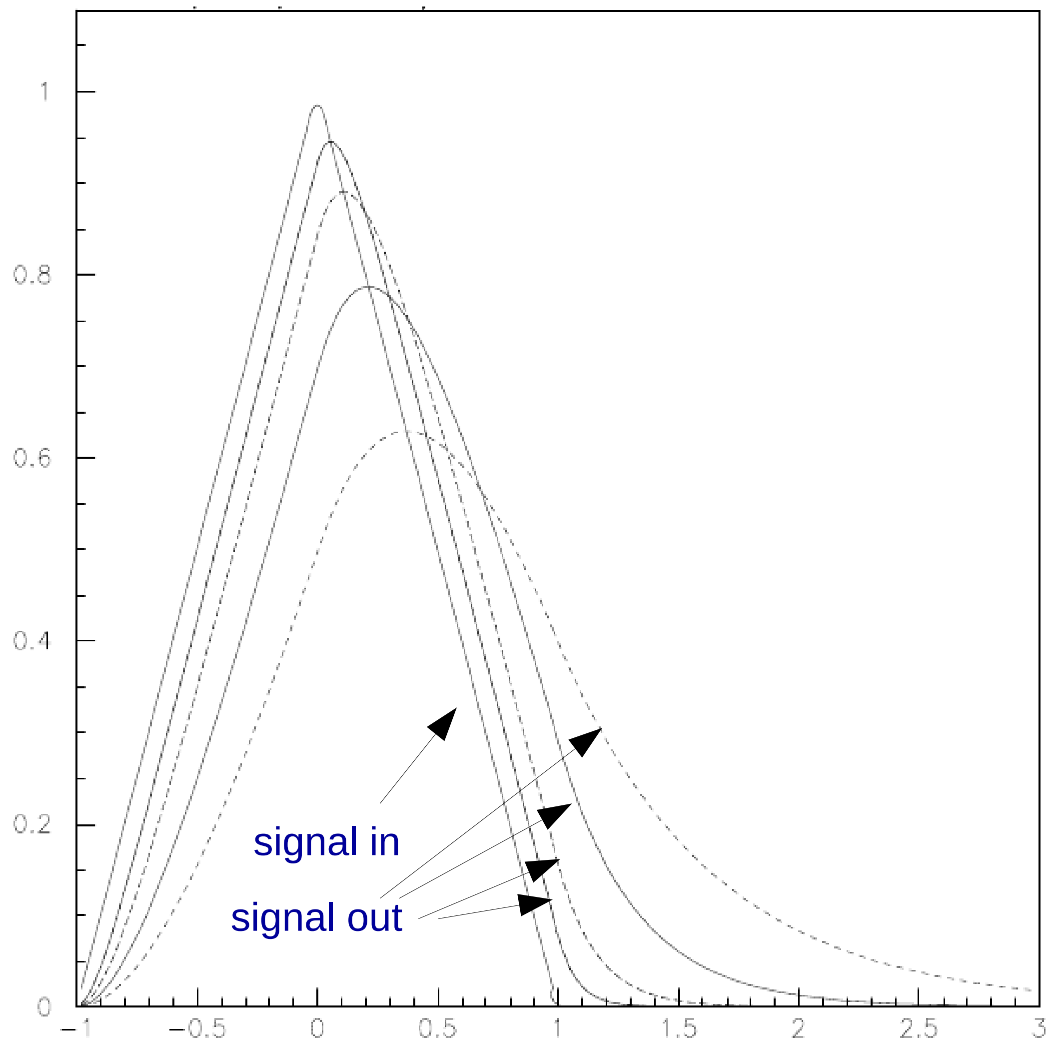
$$\text{for } \alpha \sim a\sqrt{f}$$

$$\tau_0 = (a\, z)^2 / \pi$$

Bandwidth effects – analog signals

~1ns analog-signal response for

BW ~ 300, 150, 75, ... MHz



Appendix C: Backtrace

Segfaulting ? Have a look at backtrace:

https://www.gnu.org/software/libc/manual/html_node/Backtraces.html

BACKTRACE(3) Linux Programmer's Manual BACKTRACE(3)

NAME

backtrace, backtrace_symbols, backtrace_symbols_fd - support for application self-debugging

SYNOPSIS

```
#include <execinfo.h>

int backtrace(void **buffer, int size);

char **backtrace_symbols(void *const *buffer, int size);

void backtrace_symbols_fd(void *const *buffer, int size, int fd);
```

HowTo

1) file “my_segf.cxx” : install a signal handler to print the backtrace

```
#include <stdio.h>
#include <execinfo.h>
#include <signal.h>
#include <stdlib.h>
#include <unistd.h>

void handler(int sig) {
  void *array[10];
  size_t size;

  // get void*'s for all entries on the stack
  size = backtrace(array, 10);

  // print out all the frames to stderr
  fprintf(stderr, "Error: signal %d:\n", sig);
  backtrace_symbols_fd(array, size, STDERR_FILENO);
  exit(1);
}

void baz() {
 int *foo = (int*)-1; // make a bad pointer
 printf("%d\n", *foo);       // causes segfault
}

void bar() { baz(); }
void foo() { bar(); }

int main(int argc, char **argv) {
  signal(SIGSEGV, handler);   // install our handler
  foo(); // this will call foo, bar, and baz.  Baz segfaults.
}
```

2) compile with -g debug flag on:

g++ -g -rdynamic my_segf.cxx -o my_segf

3) get the crash:

```
[roberto@bob-laptop ~]$ ./my_segf
Error: signal 11:
./my_segf(_Z7handleri+0x1c)[0x400a52]        <- handler
/lib64/libc.so.6(+0x347e0)[0x7fa55f1c07e0]   <- libc
./my_segf(_Z3bazv+0x14)[0x400aab]            <- my crash
./my_segf(_Z3barv+0x9)[0x400aca]
./my_segf(_Z3foov+0x9)[0x400ad6]
./my_segf(main+0x23)[0x400afc]
/lib64/libc.so.6(__libc_start_main+0xf1)
[0x7fa55f1ac731]
./my_segf(_start+0x29)[0x400969]
```

4) crash is at (_Z3bazv+0x14) ... the function name is “_Z3bazv” (c++ function name mangling). How to get it ?

5) Demangle it thanks to: http://demangler.com/

6) Take the Answer: baz() → crash is at (baz+0x14)

7) crash is at (baz+0x14) ... open the debugger: gdb my_segf

(gdb) info address baz
Symbol "baz()" is a function at address 0x400a55.

8) so crash is at address (0x499a55+0x14) … then:

(gdb) info line *(0x400a55+0x14)
Line 24 of "my_segf.cxx" starts at address 0x400a65 <baz()+16>
and ends at 0x400a7c <baz()+39>.

9) got it ! That’s not yet the reason but ...

Appendix D: Profiling

Take care: optimize your code – first of all - where it really needs. To get it, you may use of profiling.

for C/C++ code, look (for example) at this gprof tutorial:

http://www.thegeekstuff.com/2012/08/gprof-tutorial/

Very simple, at least for standalone code …

Appendix E: Some crude queueing theory

N-event buffer ... single queue size N:

P_k : % time with k events in ; P_N = no space available → deadtime

$\sum P_k = 1$ [k=0..N]

rate [j → j+1] = $\lambda \cdot P_j$ (fill at rate λ)

rate [j+1 → j] = $\mu \cdot P_{j+1}$ (empty at rate $\mu > \lambda$)

steady state: $\mu \cdot P_{j+1} = \lambda \cdot P_j \Rightarrow P_{j+1} = \rho \cdot P_j = \rho^{j+1} \cdot P_0$ [$\rho = (\lambda/\mu) <\sim 1$]

for $\rho \sim 1 \Rightarrow P_j \sim P_{j+1} \Rightarrow \sum P_k \sim (N+1) \cdot P_0 = 1 \Rightarrow P_0 \sim P_N \sim 1/(N+1)$

$\Rightarrow$ deadtime ~ 1/(N+1)

want ~ 1% ⇒ N ~ 100

Appendix E: Some crude queueing theory

N-event buffer ... single queue size N:

P_k : % time with k events in ; P_N = no space available → deadtim

$\sum P_k = 1$ [k=0..N]

rate [$j \rightarrow j+1$] = $\lambda \cdot P_j$ (fill at rate λ)

rate [$j+1 \rightarrow j$] = $\mu \cdot P_{j+1}$ (em ate $\mu > \lambda$)

steady state: $\mu \cdot P_{j+1} = \lambda$ $P_{j+1} = \rho \cdot P_j = \rho^{j+1} \cdot P_0$ [$\rho = (\lambda/\mu)$ <~1]

for c $P_j \sim P_{j+1}$ ⇒ $\sum P_k \sim (N+1) \cdot P_0 = 1$ ⇒ $P_0 \sim P_N \sim 1/(N+1)$

⇒ deadtime ~ 1/(N+1)

want ~ 1% ⇒ N ~ 100

Take care: analytic calculation possible for pretty simple systems only

Appendix F: Trigger qualification

trigger parameters:

1) (high) efficiency → can't be improved at HLT
2) (high) purity → can be improved at HLT
3) (low) latency → can be compensated for
4) (very low) jitter → can't be compensated for
5) synch/asynch → synch "easier"

Trigger qualification

Take care:

1) higher efficiency ⇔ lower purity

2) can compensate for (some) latency

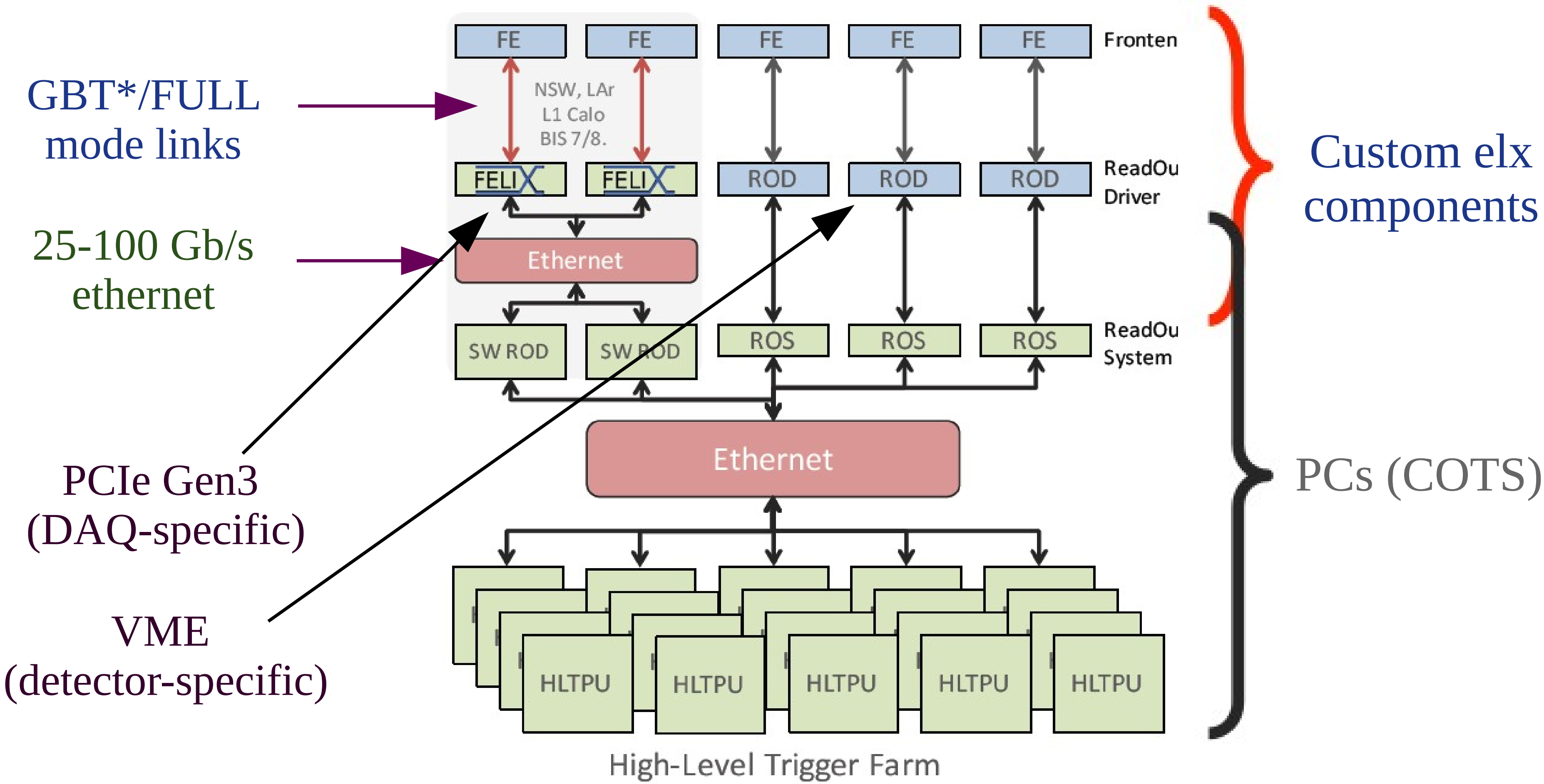
3) can NOT compensate for jitter

4) asynch trigger synch'ed will get jitter

Backup

ATLAS – TDAQ upgrade for Run 3

Same requirements as Run 2 but reduced custom components



*GBT: GigaBit Transceiver with Versatile Link