Trigger and Data Acquisition

at colliders

G. De Lentdecker

Université Libre de Bruxelles

Overview

- Introduction
  - *History*
  - *A basic T/DAQ system*
- Front end electronics
  - *Amplifier, shaper, pipeline,…*
- Digital electronics
  - *The basics (ADC, TDC), digital processing*
- Trigger
- Event builder
- Processor farm
- Future collider T/DAQ

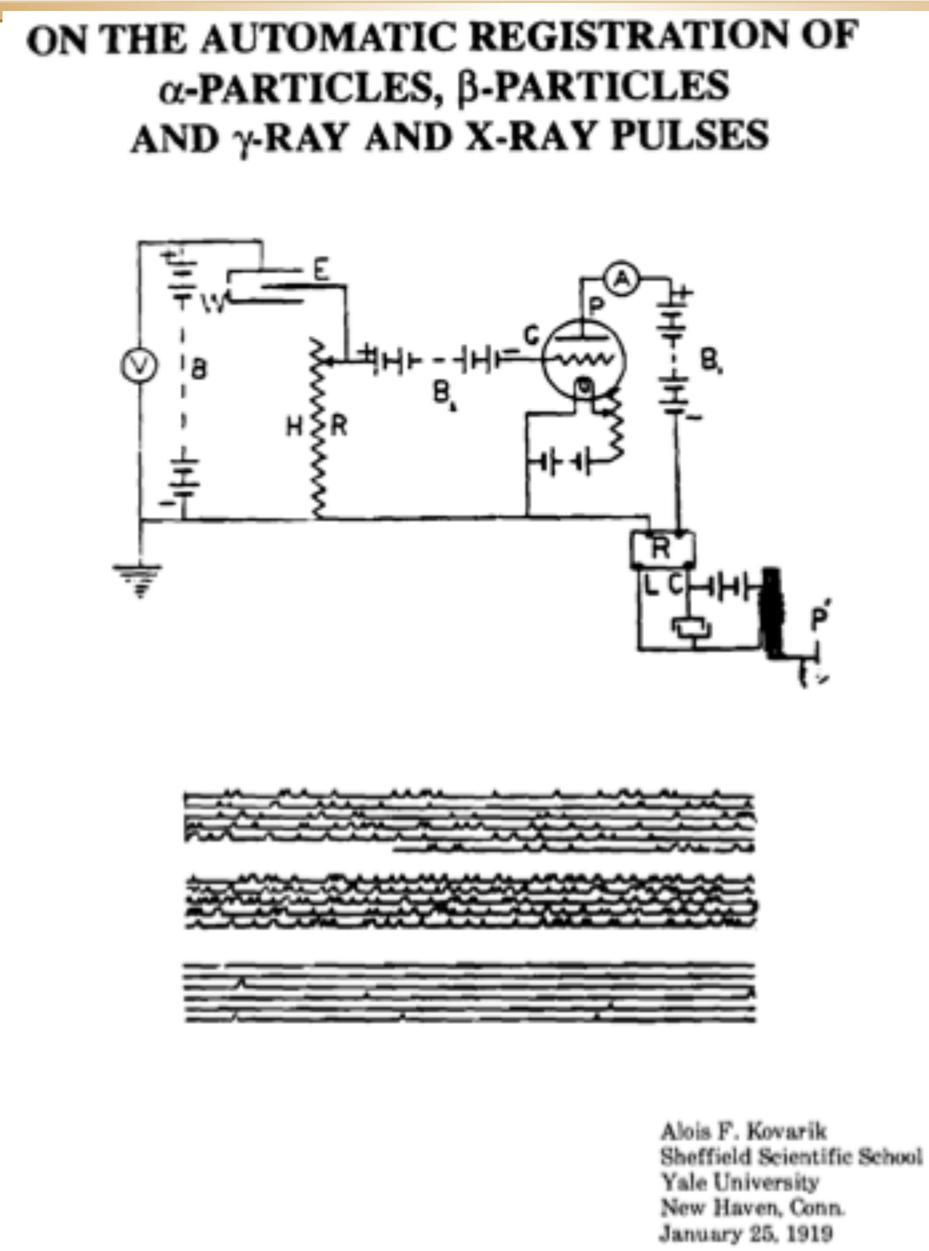
Introduction

- This series of lectures covers two subjects completely interlacing: the Trigger and the Data Acquisition (DAQ)
- What are they ?
  - *The DAQ is responsible to acquire the data, from the detector up to the recording of this data on a material support allowing offline analysis*
  - *The Trigger system is responsible to trigger the data acquisition*
- This looks trivial but in reality it is very complex because of many constraints from the physics and from the technology
- The goal of these lectures is to give you an overview of Trigger and DAQ systems by following step by step the data path from the detector up to the final storage for offline analysis

Some history

- Before electronic detectors:
  - *Detection system : the eyes (which was also the trigger) looking at flashes on a phosphorescent screen*
  - *Data acquisition : manual recording in a logbook*
- It worked BUT
  - *Can’t make long runs (unless you have a lot of students)*
  - *« manually recording the passage of particles […] was hard on the experimenter because of the random rate at which the information came in »*
- First one-channel electronic recording device in 1919:
  - *« On the Automatic Registration of Alpha Particles, Beta Particles, and Gamma Ray Pulses.” A. F. Kovarik*

First DAQ



ON THE AUTOMATIC REGISTRATION OF
α-PARTICLES, β-PARTICLES
AND γ-RAY AND X-RAY PULSES

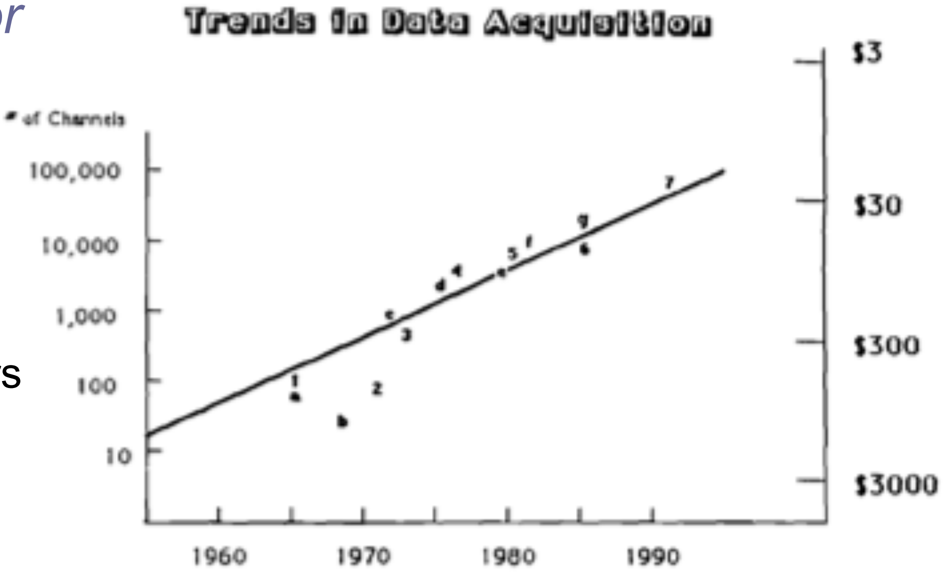
E
W
V
B
H
R
B_1
C
P
A
B_1
R
L
C
P'

Alois F. Kovarik
Sheffield Scientific School
Yale University
New Haven, Conn.
January 25, 1919

First electronic detectors

- With the introduction of electronics readout systems that DAQ and trigger history really started:
  - *1947 : introduction of the transistor*
  - *60's, < 100 channels experiments*
  - *68: MWPC (Charpak)*
    - Before tracking detectors were almost all optical:
      - – emulsions, cloud/bubble chambers
  - *Today (2010):*
    - ATLAS/CMS
      - – O(10^7) channels
      - – 40 MHz

Trends in Data Acquisition

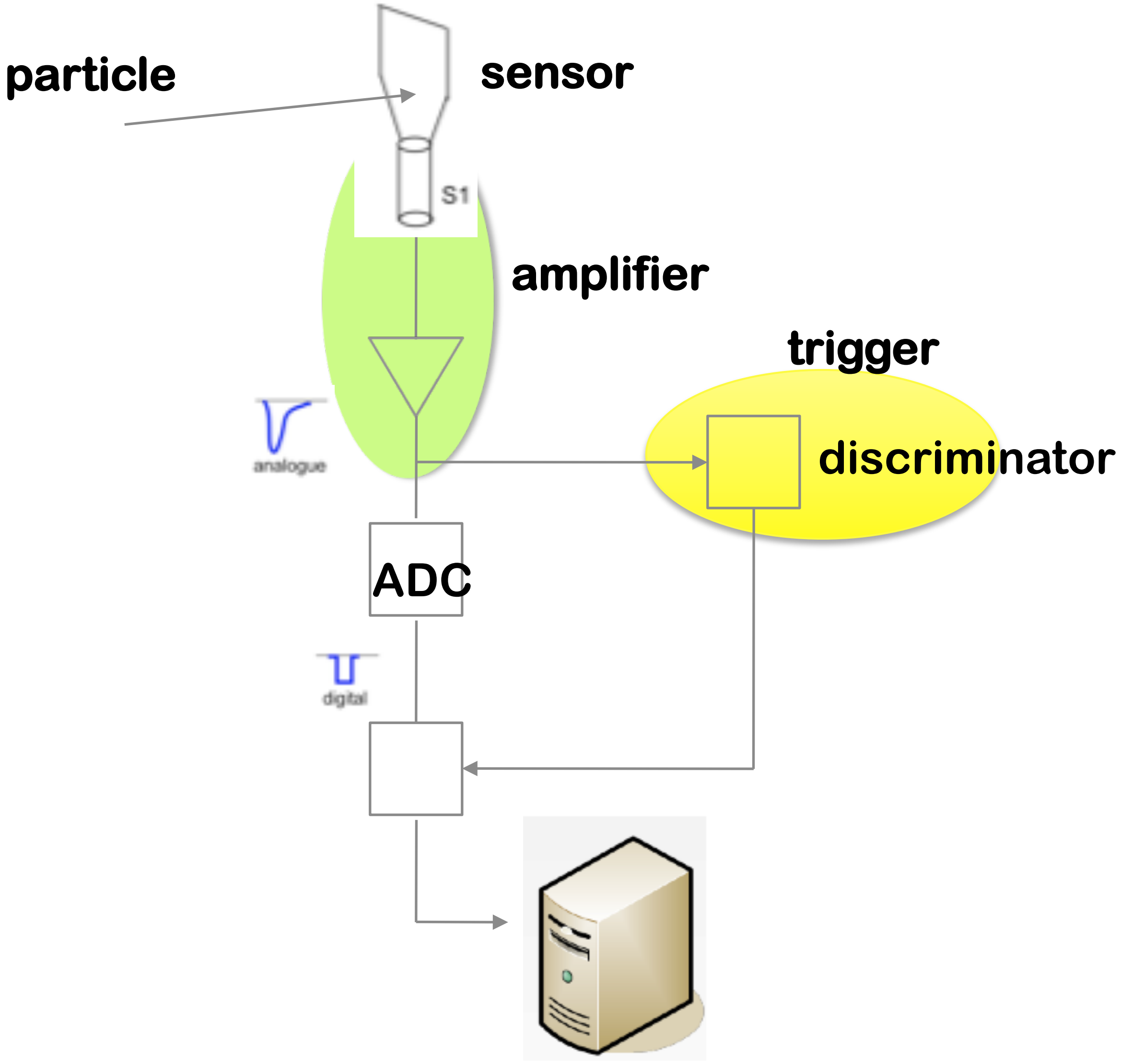


Number of Channels/Instrument/Experiment vs. Time
and
Cost/Channel of ADC or TDC vs. Time

- There is not only the « technical » issues to record so many events in a such a small time, there is also the issue of the event selection:
  - *We are now looking to very rare new phenomena:*
    - For instance at LHC (14 TeV, L = 10^{34}cm^{-2}s$^{-1)}$, SM Higgs rate < 0.01 Hz!
  - *=> the millions other collisions are uninteresting !!*
    - These have to be rejected while still recording the Higgs…

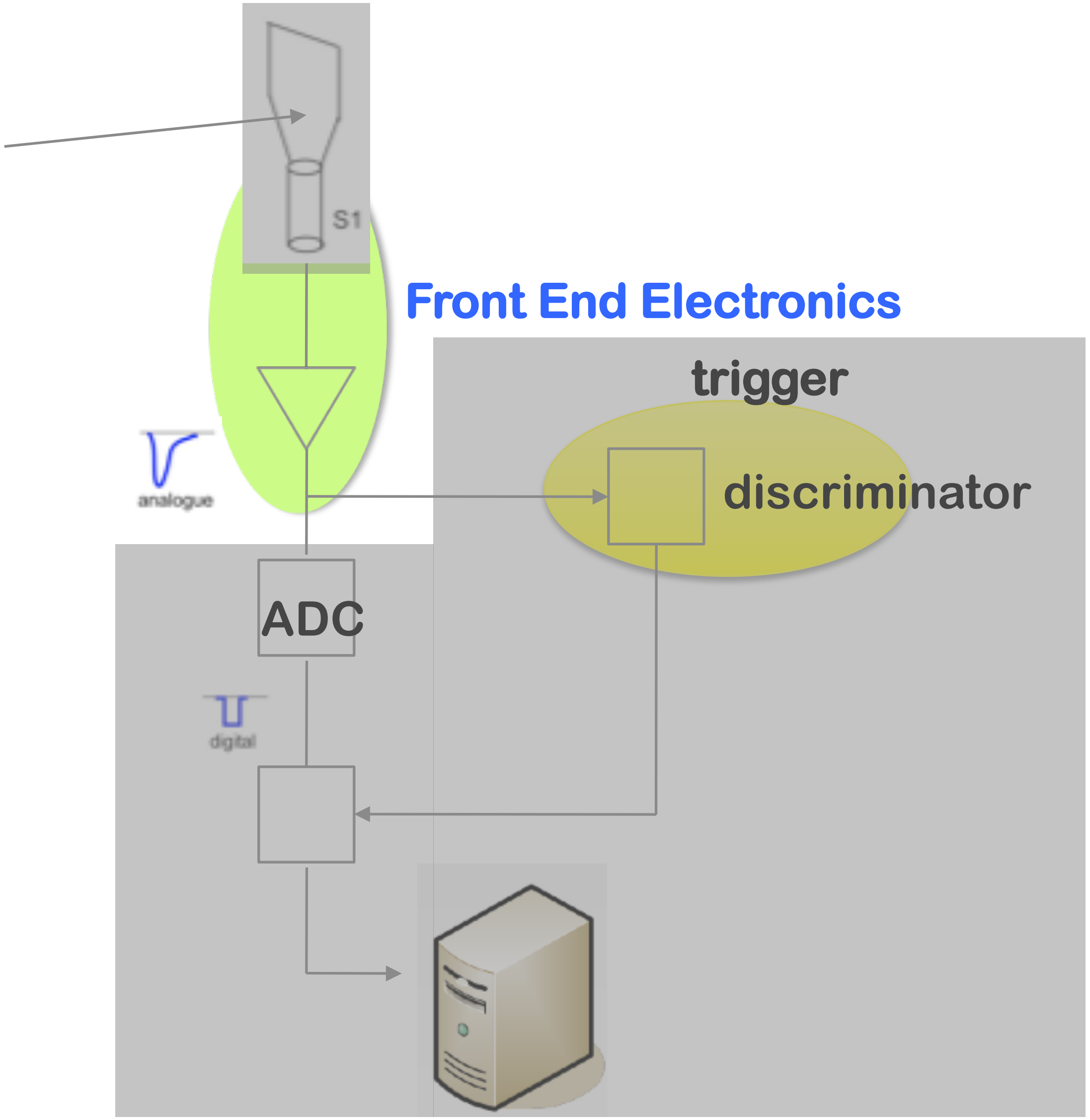
The basics

- Our basic picture of a DAQ/Trigger system:



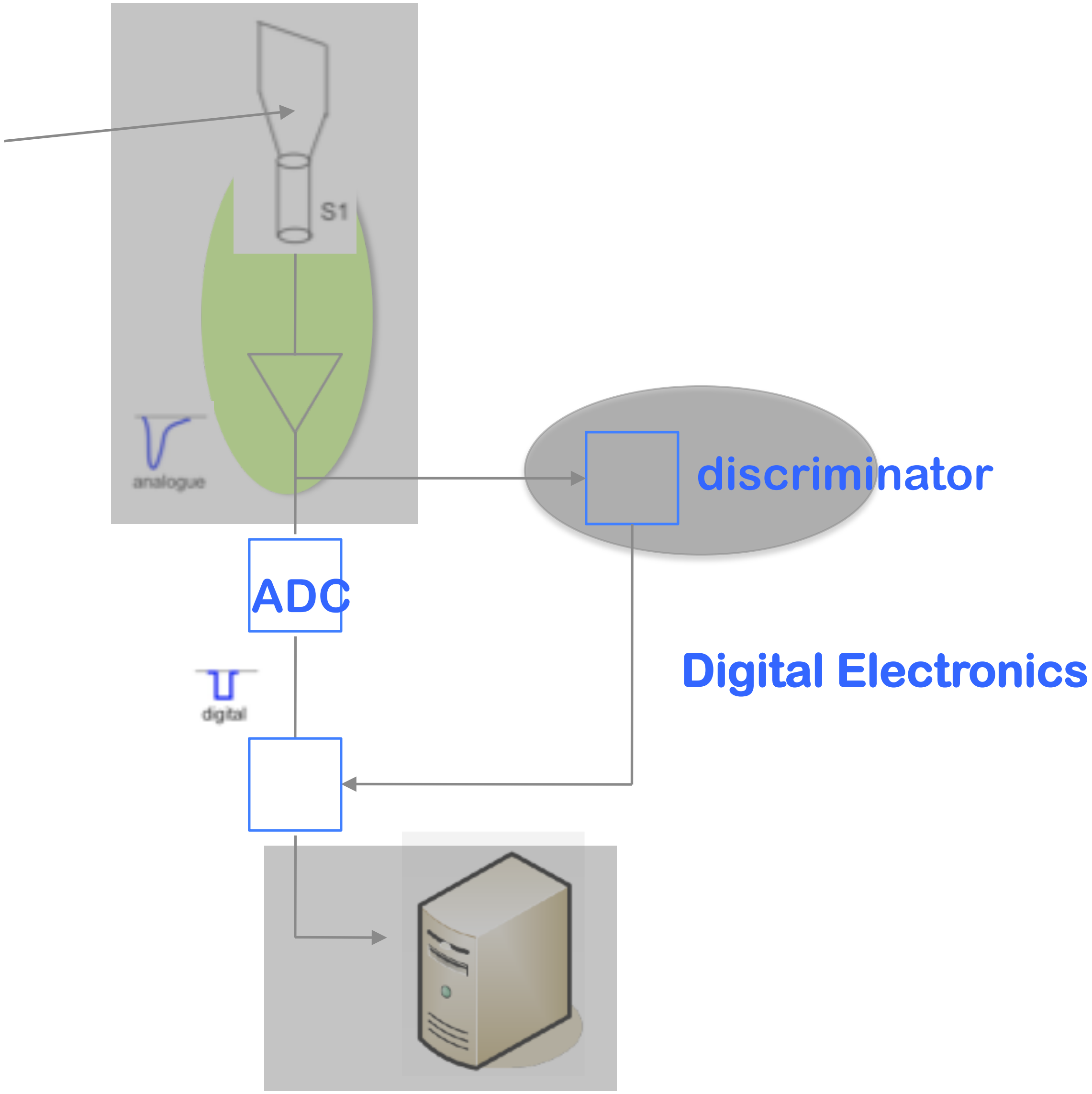
Part I: Front End Electronics

- I will talk about:



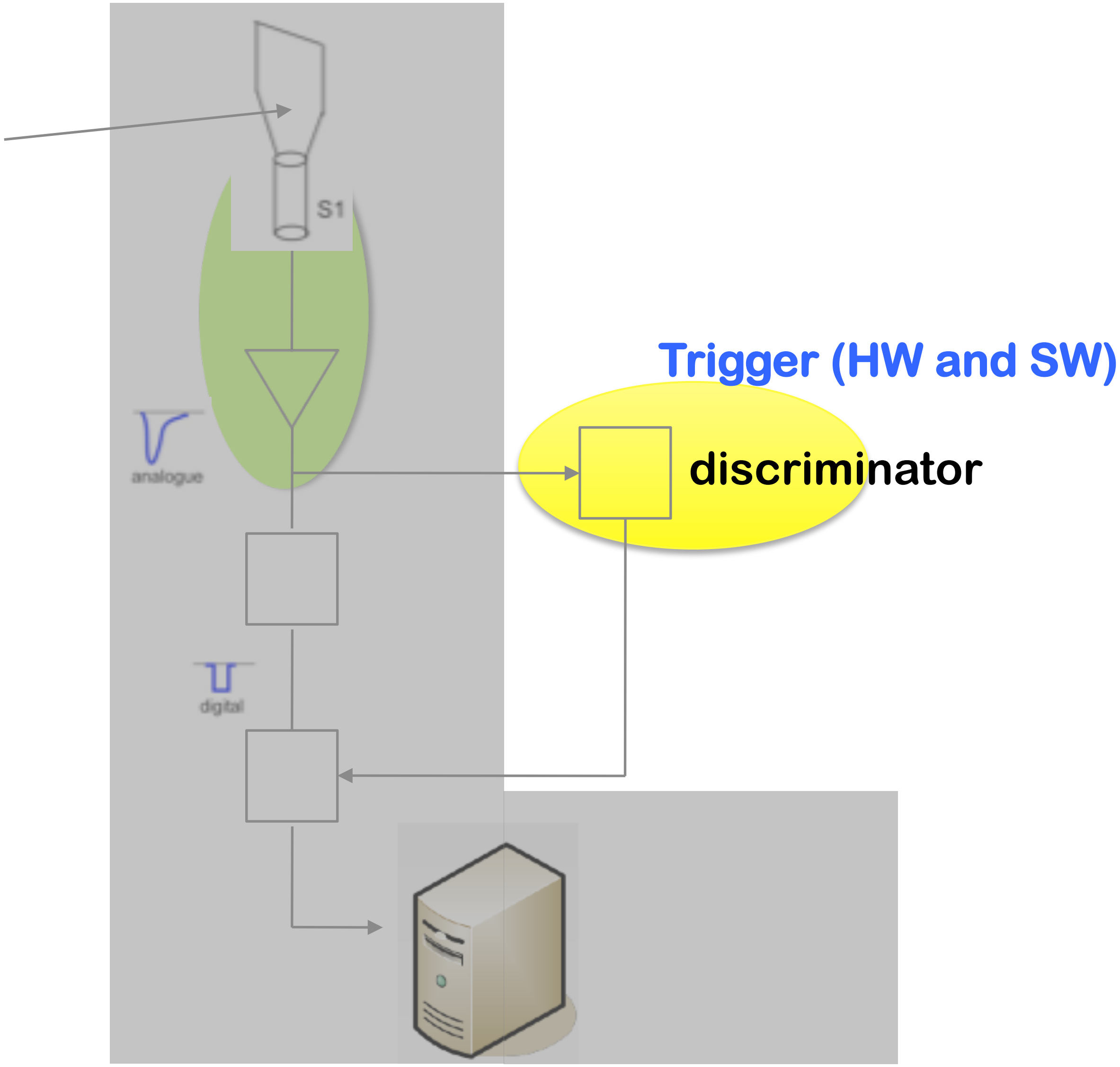
Part II: Digital Electronics

- I will talk about:



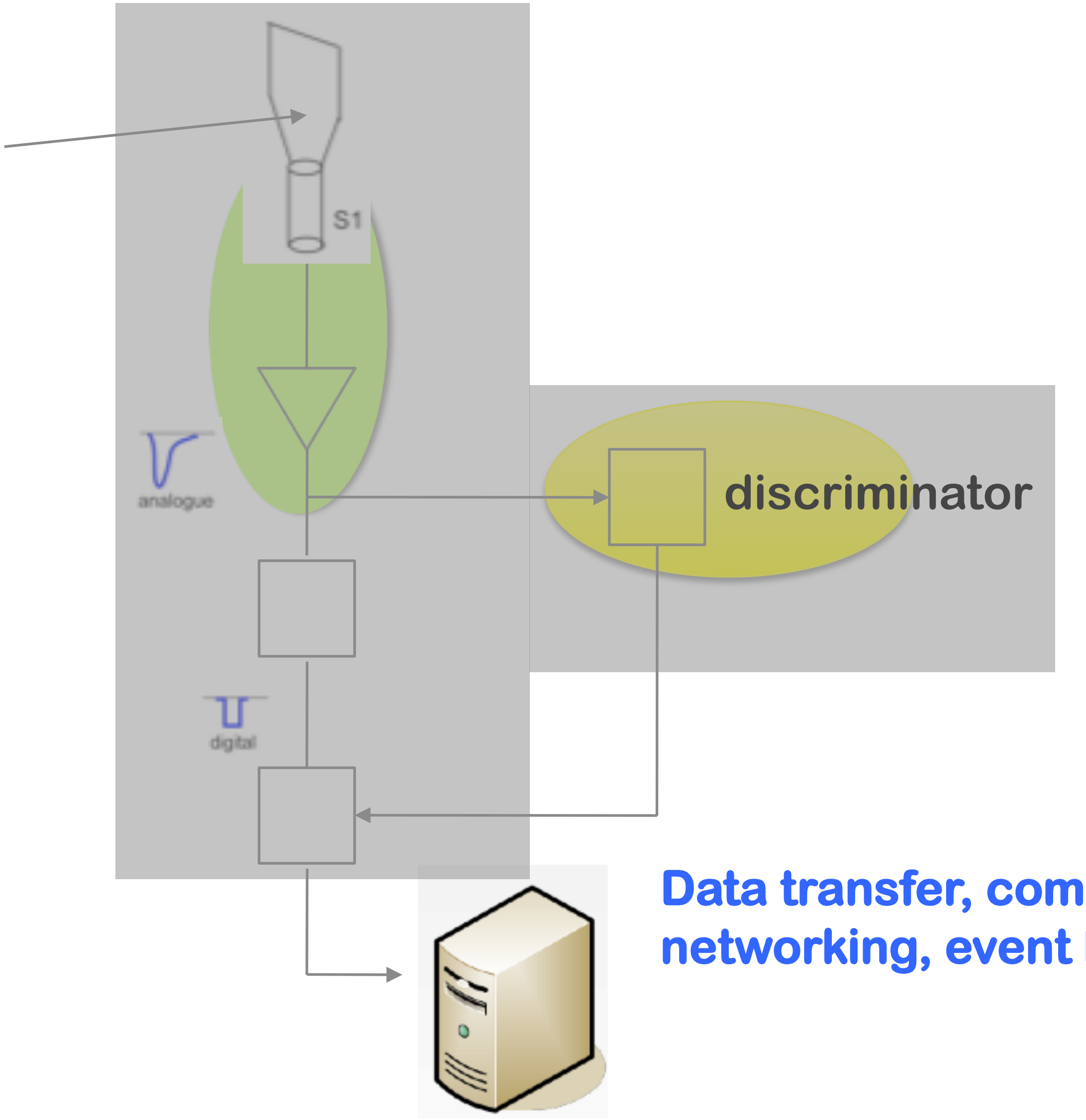
Part III: Trigger

- I will talk about:



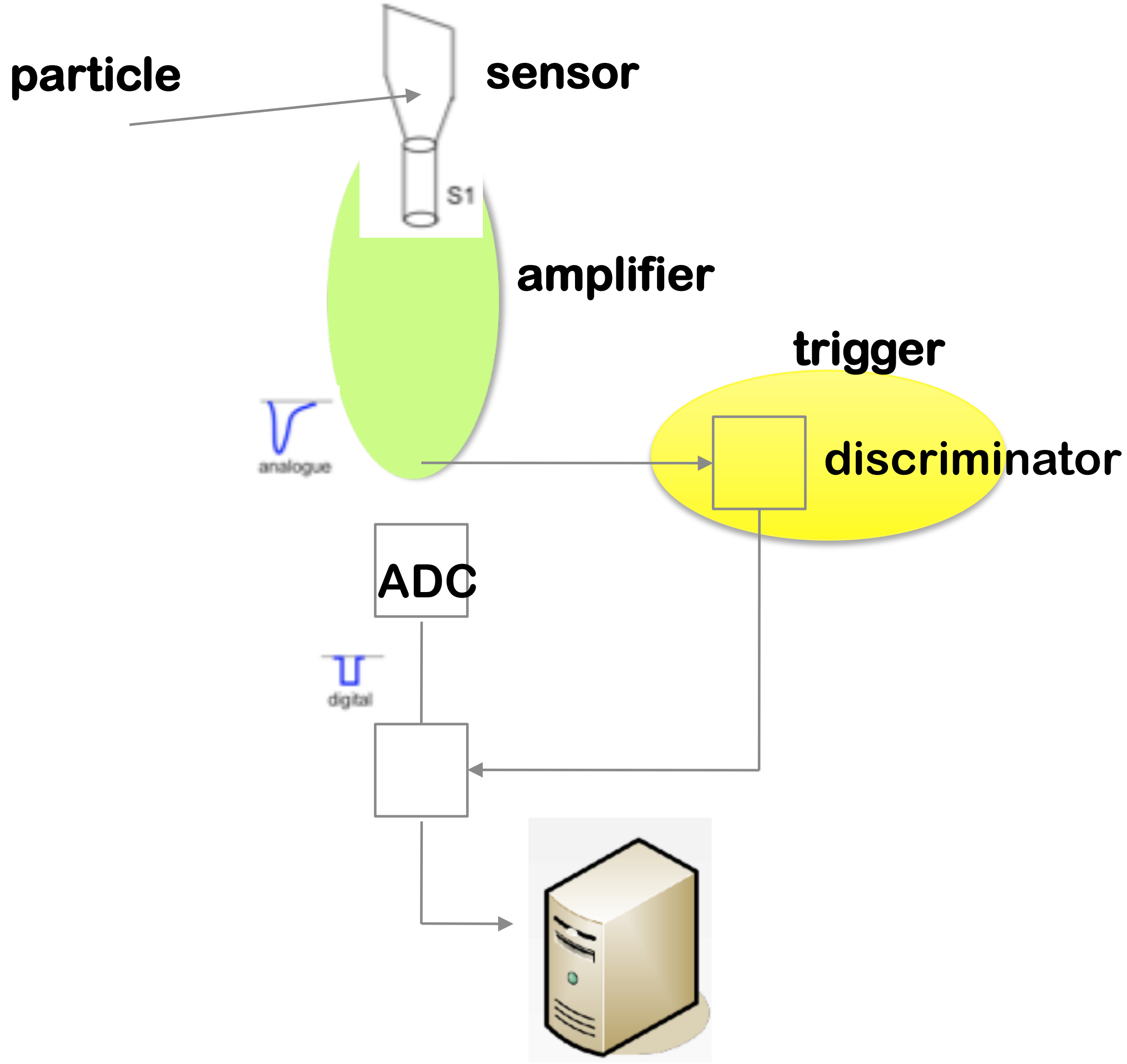
Part IV: All the rest...

- I will talk about:



Data transfer, computing, networking, event building,...

From here…


particle
sensor
S1
amplifier
trigger
analogue
discriminator
ADC
digital

… to there

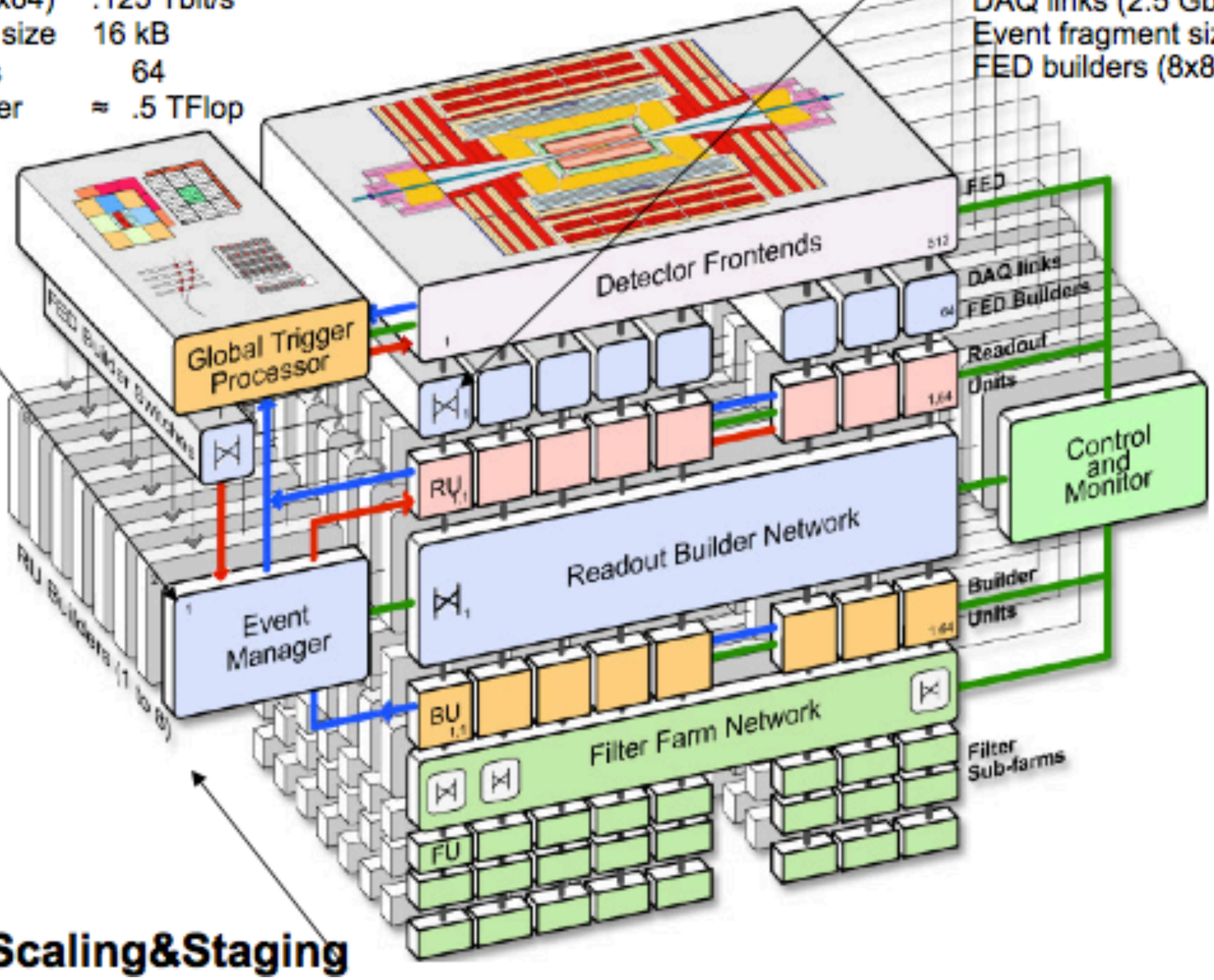
- CMS DAQ

DAQ unit (1/8th full system):

Lv-1 max. trigger rate	12.5 kHz
RU Builder (64x64)	.125 Tbit/s
Event fragment size	16 kB
RU/BU systems	64
Event filter power	≈ .5 TFlop

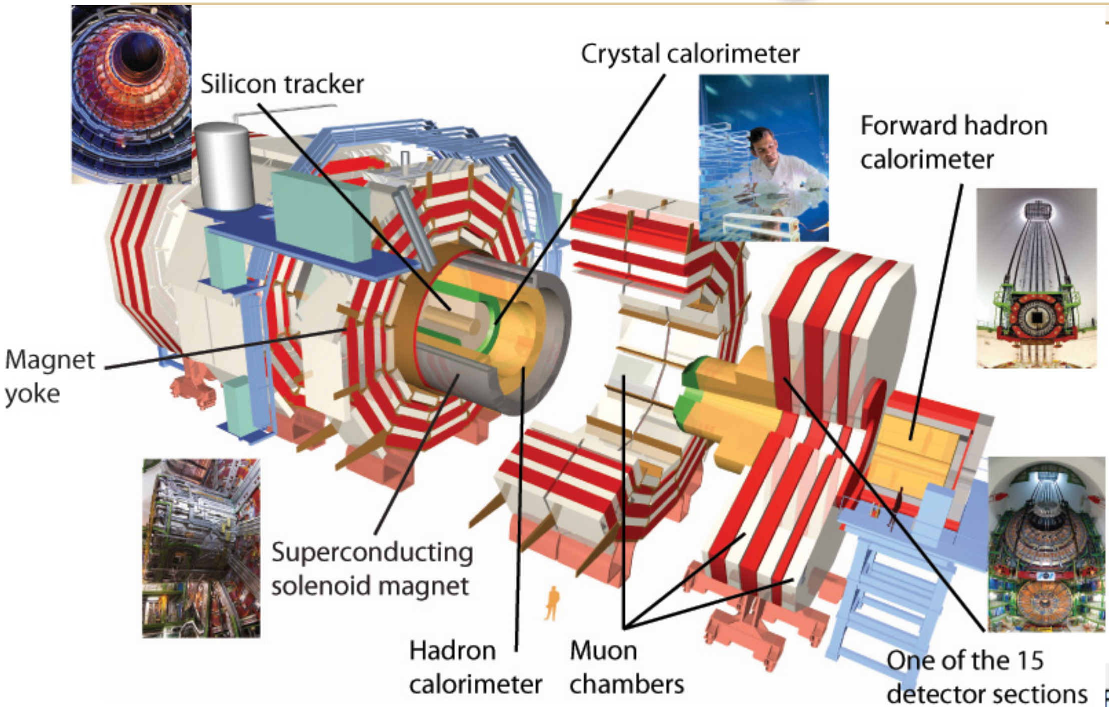
DATA to surface

Average event size	1 Mbyte
No. FED s-link64 ports	> 512
DAQ links (2.5 Gb/s)	512+512
Event fragment size	2 kB
FED builders (8x8)	≈ 64+64



DAQ Scaling&Staging

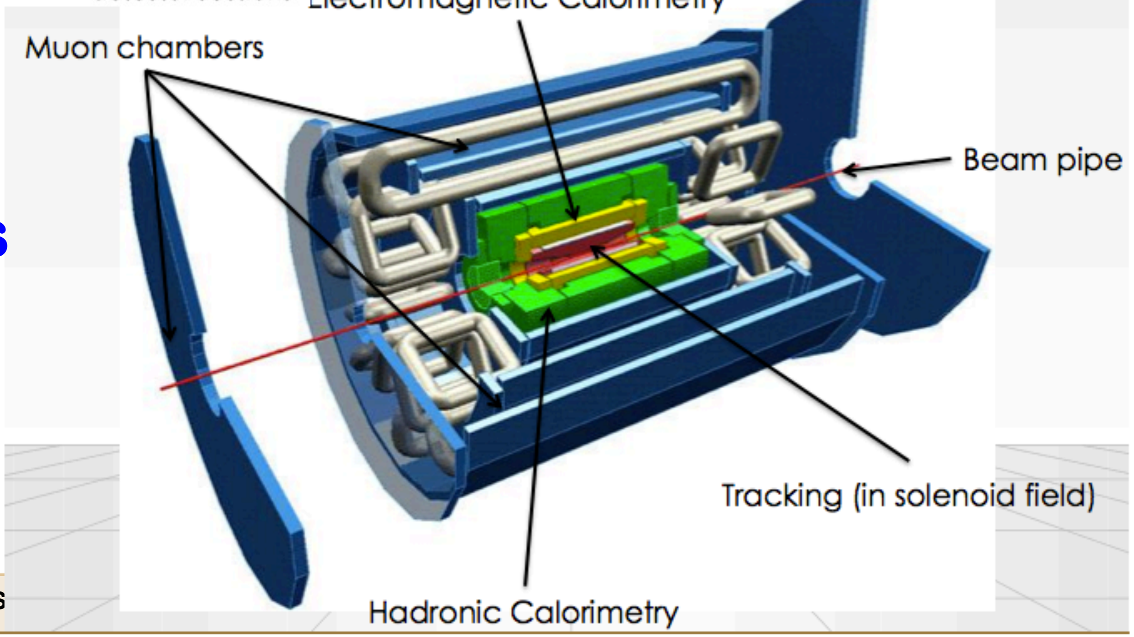
Before starting: a word on detectors


Silicon tracker
Crystal calorimeter
Forward hadron calorimeter
Magnet yoke
Superconducting solenoid magnet
Hadron calorimeter
Muon chambers
One of the 15 detector sections

Most of the talk take CMS and ATLAS as examples

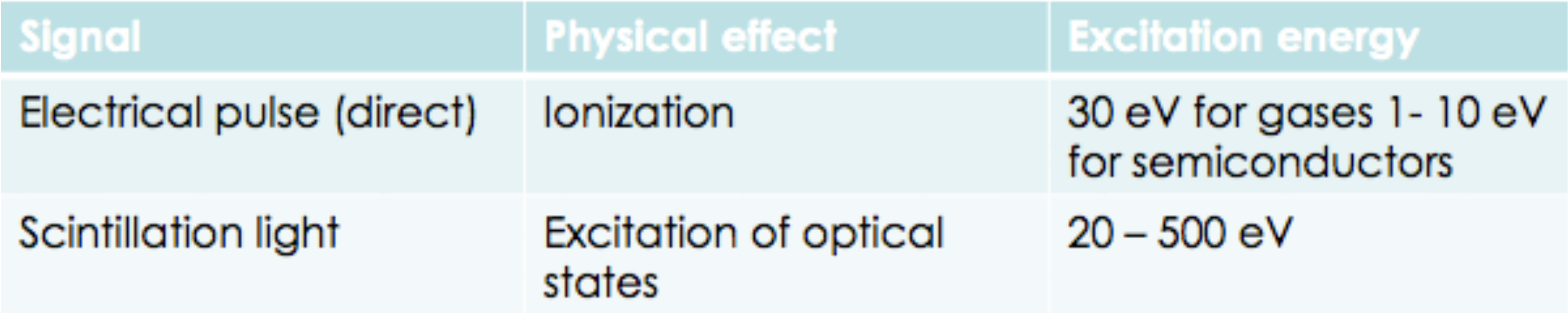
CMS

ATLAS



Detector signals

- The signal is usually a small current pulse varying in duration
  - *from ~ 100 ps for a Si sensor to O(10) μs for inorganic scintillators*
- There are many sources of signals.
  - *Magnitude of signal depends on deposited signal (energy / charge) and excitation energy*



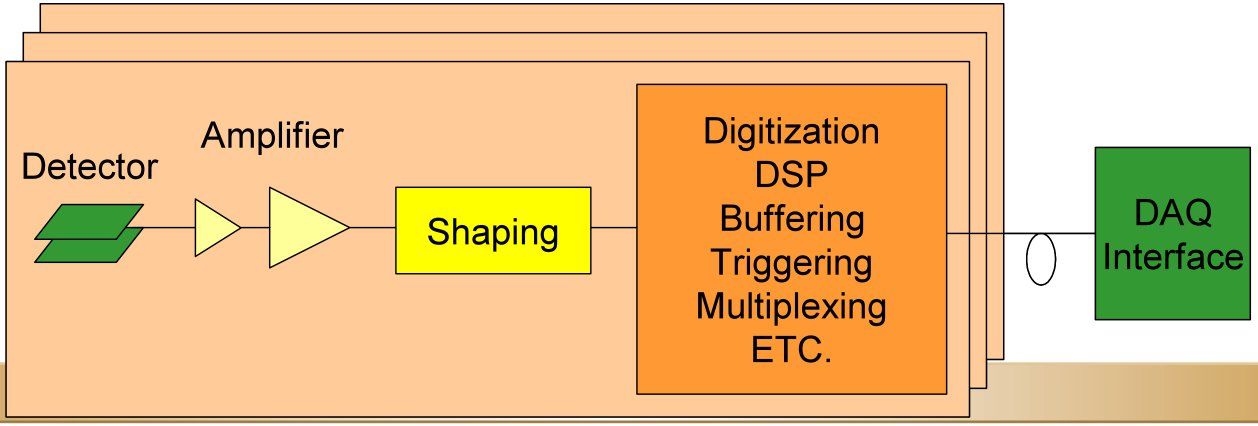
Signal	Physical effect	Excitation energy
Electrical pulse (direct)	Ionization	30 eV for gases 1- 10 eV for semiconductors
Scintillation light	Excitation of optical states	20 – 500 eV

- Signals must be amplified, digitized to allow storage and processing

PART I:
FRONT END ELECTRONICS

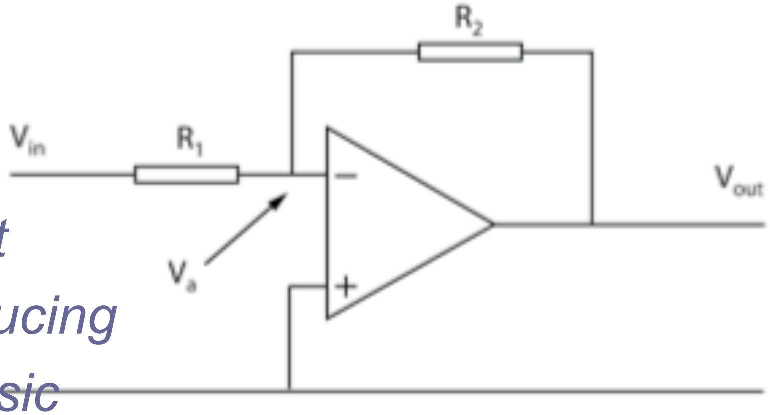
Signal from the detectors

- The Front End Electronics (FEE) is the electronics which is “on” the detector, that is directly connected to the detector.
- The FEE is detector dependent but the basics are the same:
  - *Detectors generate signals (charge) from ionization in a medium (gas, liquid, semiconductor), sometimes with a detector gain (gas, PM)*
  - *These signals are generally small*
    - ~22k electrons (4fC) in a 300 um thick Si sensor
  - *These signals need to be:*
    - Amplified
    - Shaped, to optimize signal to noise ratio, event rate, energy/time measur.
    - Digitized (not always at FEE stage)
    - Stored for further treatment (for instance waiting for the trigger decision)



FEE basics: the signal amplification

- From electronic point of view, detectors behaves like a current source with a capacitance (C_d) and an internal resistor
- There are 3 types of amplifier:
  - *Current sensitive (works w/. very low impedance devices → not for HEP)*
  - *Voltage sensitive*
  - *Charge sensitive*
- Voltage sensitive amplifier:
  - *Amplifies any voltage appearing at input*
  - *Since detectors are mainly charge producing devices, voltage appears through the intrinsic capacitance of the detector + any stray capacitance at the input*



⇒ *$V = Q/C_{tot}$, where $C_{tot} = C_d + C_{other}$*

⇒ *detector capacitance must be very stable*

⇒ *Not advisable for semiconductor devices*

Another reason why this amplifier is not optimal is the presence of a resistor connected at the input (see noise section)

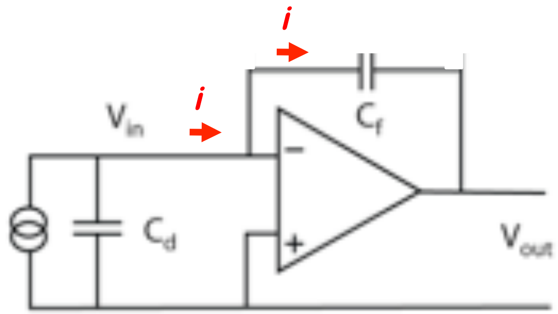
Charge integrator

- Charge sensitive preamp (90% of HEP pre-amp):
  - *Integrates the charge carried by the incoming pulse on the feedback capacitance C_f*

    ⇒ No dependence on detector capacitance:

    $$\Rightarrow i = C_f \frac{d}{dt}(v_{in} - v_{out}) \approx -C_f \frac{dv_{out}}{dt}$$

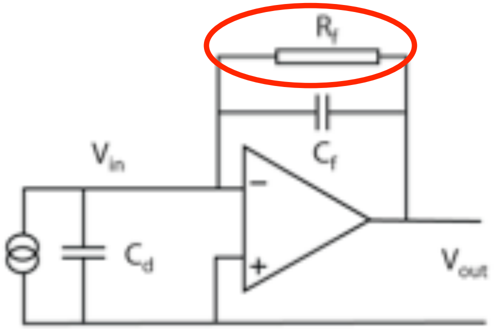
    $$\Rightarrow v_{out} = -\frac{1}{C_f}\int i dt = -\frac{Q}{C_f}$$



  - *Gain = $1/C_f$: 1 pF -> 1 mV/fC*
  - *The needed gain determines C_f*
  - *Gain only depends on charge integration capacitor C_f and not on detector capacitance*
    - Gain is independent of detector and parasitic capacitance variations
    - However detector capacitance has direct effect on noise
  - *Problem: output voltage continues to integrates detector current*

    ⇒ need to reset the capacitor

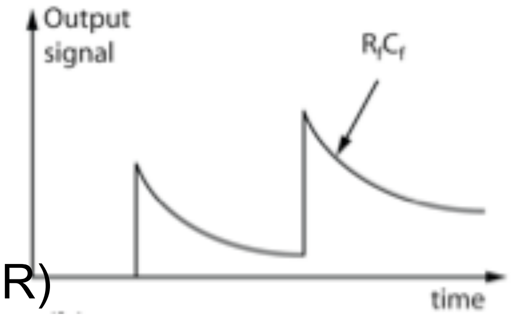
    ⇒ continuous discharge by resistor:

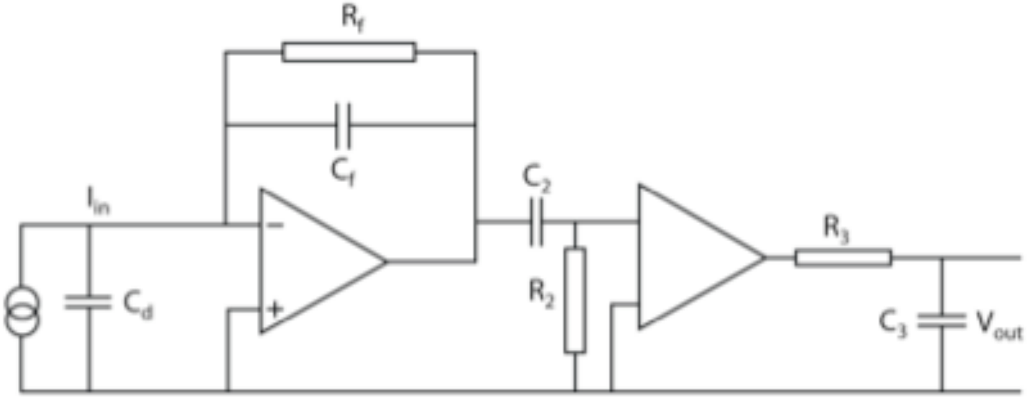


Shaping

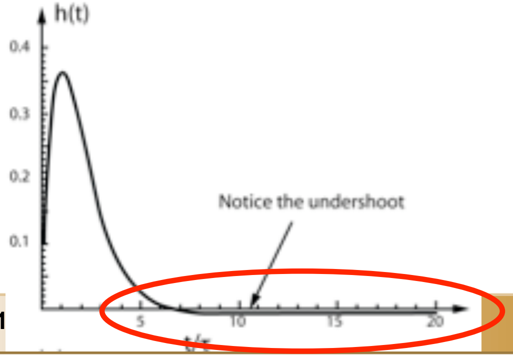
- The response of a charge integrating amplifier to a current pulse will be a sharp rising edge at the moment the pulse arrives, followed by an exponential decay with a time constant $R_f C_f$, of the order of (typical.) 40-50 μsec.
  - *This severely limits the counting rate*
  - *Need a way to make output pulses shorter*

    ⇒ Shaping
    - Typically CR-RC stages: Pulse is filtered at low (CR) and high (RC) frequencies.
    - Usually $C_2R_2=R_3C_3 << R_fC_f$





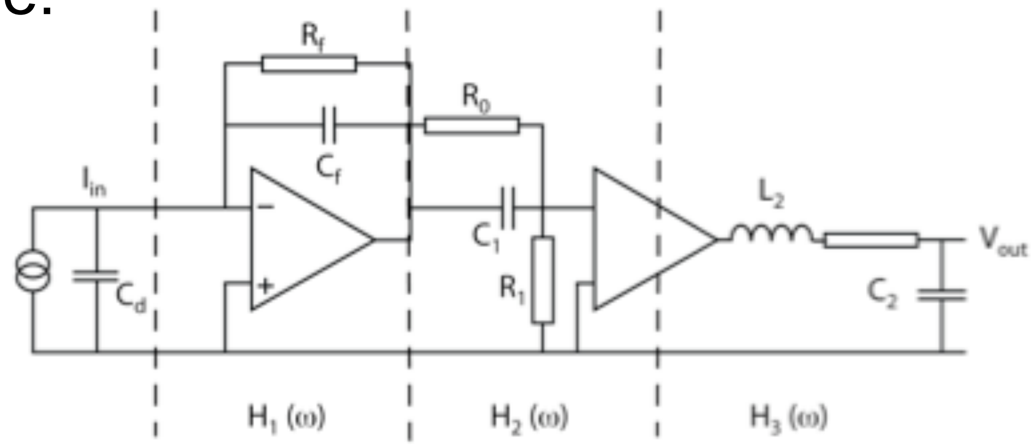
- Drawbacks: presence of undershoot:



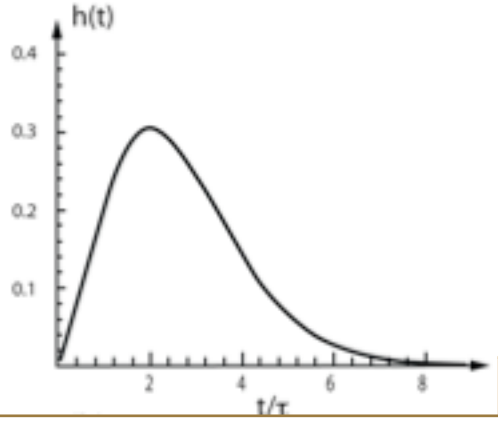
Pole-zero cancellation and baseline restoration

- The long negative tail after the main pulse (called undershoot) is due to the differentiation (CR) of finite length pulses. Theoretically no undershoot would occur if these tail pulses were infinitely long.

- Without entering into details there are several ways to cancel the undershoot. These are called ‘pole zero cancellation’. One way is illustrated in the picture:



- This circuit gives the following output pulse:

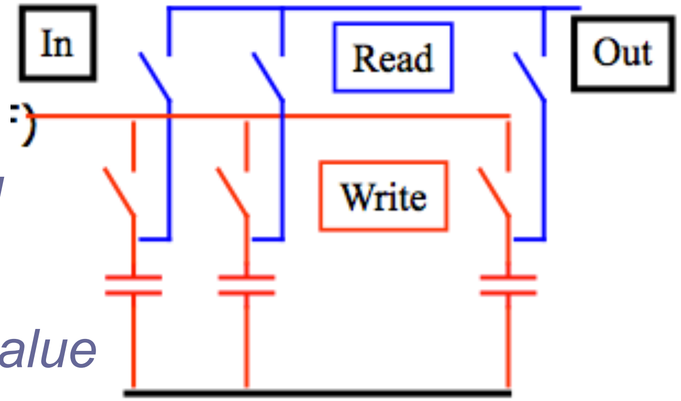


Electronic noise

- Would need a 4 hours lecture for itself
- As a staring point, remember that:
  - *Any piece of matter is made up of electrons and nuclei. These charges are constantly in motion due to thermal agitation. It is obvious that these motions will induce small voltage and current fluctuations in any piece of material, and in particular in any resistor. This noise is called thermal noise or Johnson noise.*
  - *In order to minimise this noise at the pre-amp stage. it is essential that the feedback resistance is as large as possible. On the other hand, the feedback capacitance cannot be made arbitrarily small. It should always be sufficiently large compared to any stray capacitance that are unavoidably present in the system. In practice it is difficult to have this capacitance less than about 1pF. The result is that R_fC_f must be large if we want to have low noise.*

Analog memories

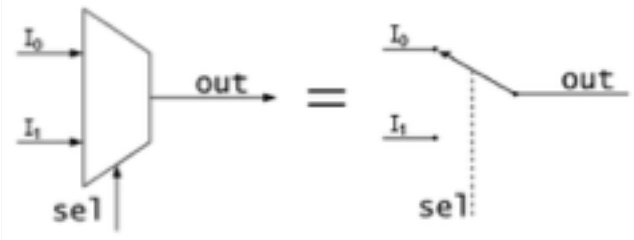
- It is often convenient to store analog signals while the trigger makes its decision.
- => use of Switching Capacitor Array (SCA), often called pipeline
  - *The signal is sampled and stored on capacitors (pF)*
  - *Fast write (~GHz)*
  - *# of cells 100 – 1000*
  - *The capacitor cells can be written to or read from in a sequential or random manner*
  - *No dependence on the absolute capacitor value*
  - *Low power consumption*
  - *BUT possible loss in signal integrity (signal drop, leakage current)*



- Example CMS Tracker APV chip:
  - *192 cell pipeline / ch, at 40 MHz sampling => 4.8 us*

Mux

- In electronics a mux (abbreviation of multiplexer) is a device which perform multiplexing that is it selects one out of many analog or digital input signals and forward the selected input into a single line.
- Multiplexers are used as one method of reducing the number of integrated circuit packages required by a particular circuit design. This in turn reduces the cost of the system.
- Typical representation:



Let put all these components together:

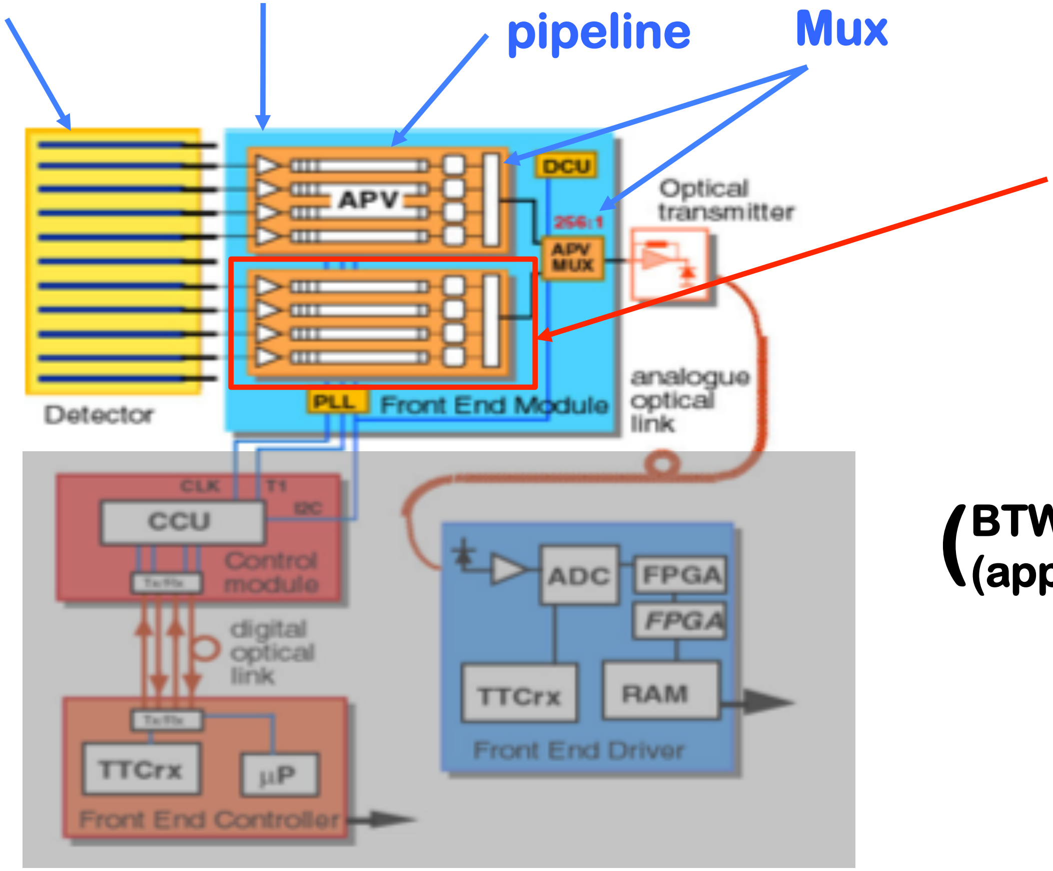
- Example : the CMS Tracker FEE (APV):

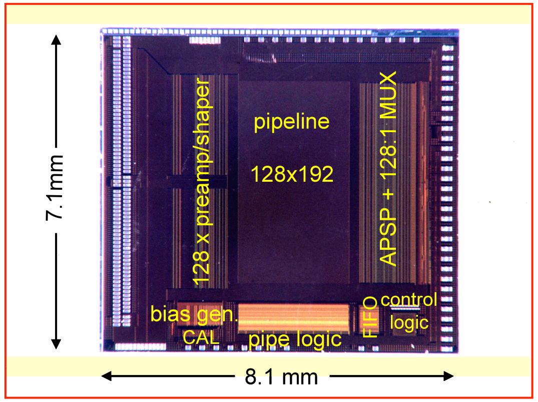
Silicon sensor

Preamplifier, shaper

Analog pipeline

Mux





128 channels APV chip

(**BTW: this is what we call an ASIC (application-specific integrated circuit)**)

Next steps

- At this stage in the CMS Si strip tracker the data are sent out of the detector towards crates in the underground area
- In the case of the CMS Si strip tracker, analog data are sent through optical fibers towards modules were further processing will be performed (like conversion to digital signals, etc.)
- Other detector electronics go from analog to digital at a earlier stage (see ALICE TPC readout later)
- So let’s move to the digital world

Trigger and Data Acquisition
at colliders

G. De Lentdecker

Université Libre de Bruxelles

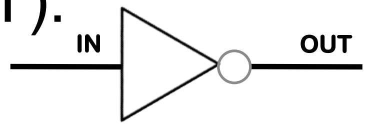
DIGITAL ELECTRONICS

Reminder

- Computing power of a processor depends on logical circuits
  - *These circuits have a rudimentary behaviour essentially described by the binary algebra (Boolean algebra)*
  - *→ at some point, in the data path, analog signals will be converted to digital signals*
- We will start with some basic logical circuits and move towards much more complex systems

The basic logical gates

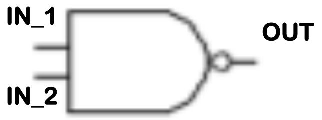
- Inversion (NOT):



IN	0	1
OUT	1	0

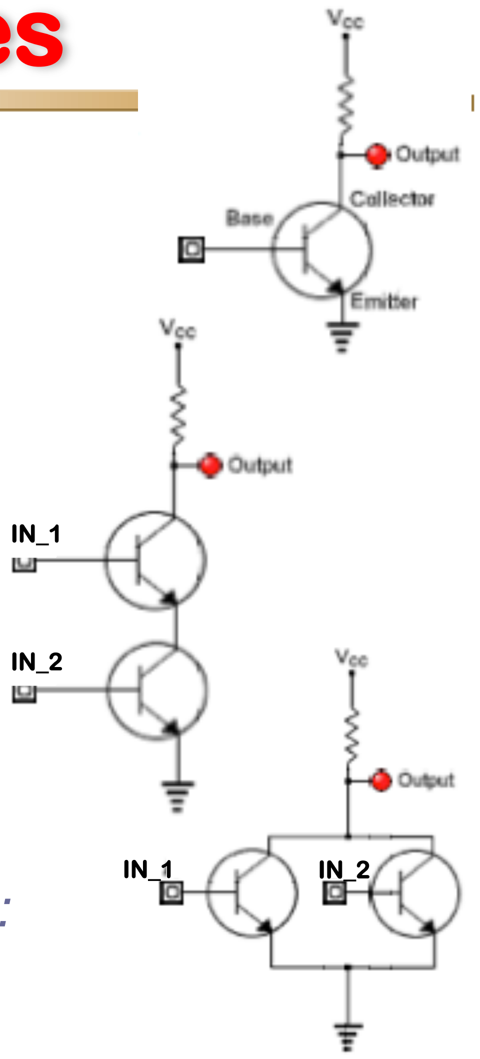
NOT circuit:

- NAND:



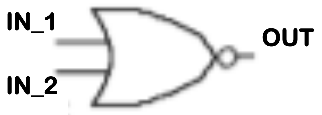
IN_1	0	0	1	1
IN_2	0	1	0	1
OUT	1	1	1	0

NAND circu



- AND = NAND + NOT

- NOR:



IN_1	0	0	1	1
IN_2	0	1	0	1
OUT	1	0	0	0

NOR circuit:

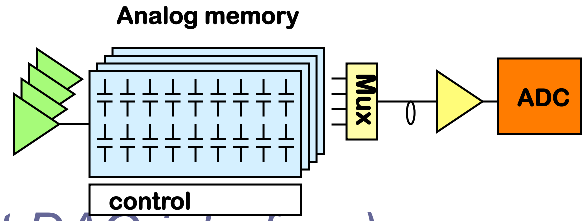
- OR = NOR + NOT
- From those you can build addition, subtraction, multiplication, etc.

Memories

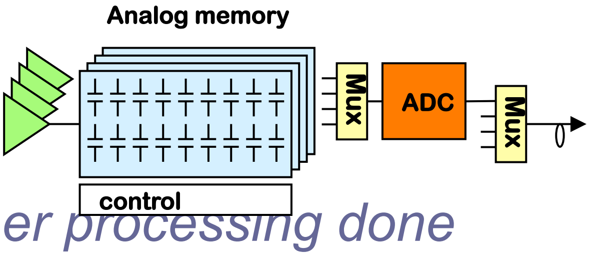
- ROM, DRAM, SRAM, etc.
- Buffers:
  - *Memory used to temporarily hold data while it is being moved from one place to another*
- FIFO (First In - First Out)
  - *primarily consists of a set of read and write pointers, storage and control logic*
  - *This expression describes the principle of a queue processing technique by ordering processes by first-come, first-served*

Analog to Digital/Binary Conversion

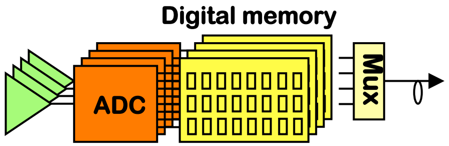
- After amplification the analog signal will have at some point to be digitized to allow for further processing (by computers, for instance)
- Analog readout
  - *Analog buffering with digitization done after buffer or after analog transmission off detector (at DAQ interface)*



- Analog buffer with digital readout



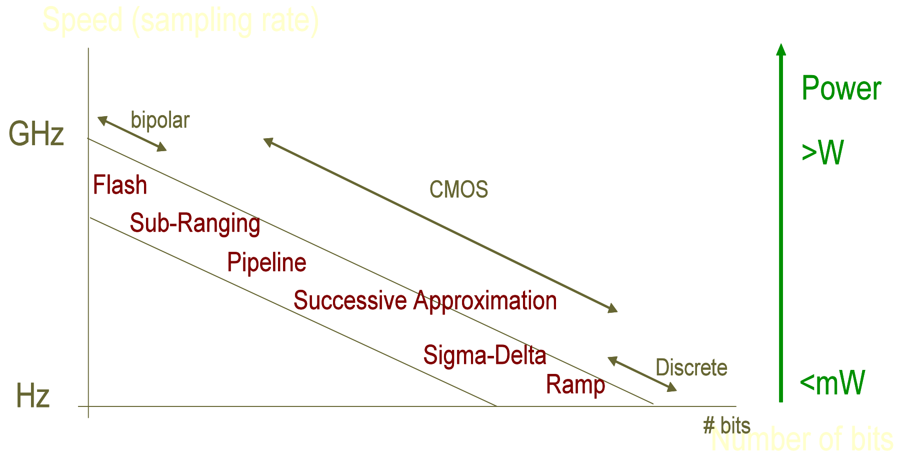
- Digital readout
  - *Information digitized after shaping and all further processing done digital*



- We will now review ADC, TDC, ...

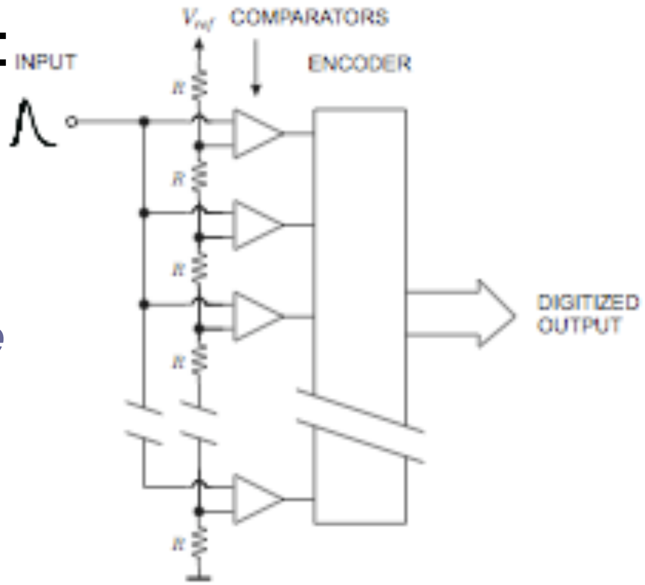
Analog to Digital Converter (ADC)

- The ADC is a device which converts the information contained in an analog signal to an equivalent digital form
- Important parameters for an ADC are:
  - The resolution (the number of bits)
  - The linearity (is the digital output proportional to the analog input ?)
  - Conversion time
  - Stability
- ADCs exist in many flavours:
  - *The "cost" of the ADC determines which architecture is chosen*
  - *Strongly depends on speed and resolution*
    - Cost is here:
      - –Power consumption
      - –Silicon area (technology)
      - –Availability of radiation hard ADC



FADC

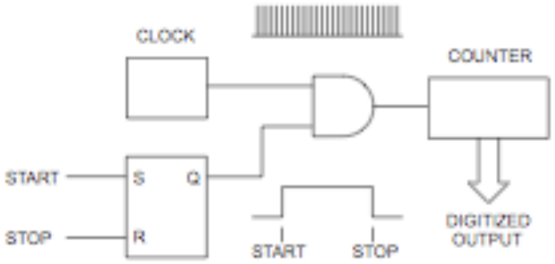
- Working principle of a Flash-ADC (FADC):
  - *Conceptually the simplest technique of ADC.*
  - *The signal is fed in parallel to a bank of threshold comparators.*
  - *The individual thresholds are set by a resistive divider.*
  - *The comparator outputs are encoded such that the output of the highest level comparators that fires yields the correct bit pattern.*
    - Pro's: fast (conversion is one step), conversion time < 10ns achievable
    - Con's: number of components (therefore power consumption): the larger the number of bits the larger the bank required. Limited to ~8 bits



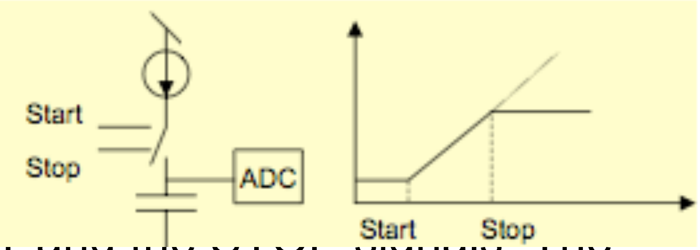
Time To Digital Converter (TDC)

- Time measurements are important in many HEP applications
  - *Identification of bunch crossing (LHC: 25ns)*
  - *Distinguishing among individual collisions (events) in continuous beam like experiments (or very short bunch interval like CLIC: ~250ps)*
  - *Drift time*
    - Position in drift tubes (binary detectors with limited time resolution: ~1ns)
    - Time projection chamber (both good time and amplitude)
    - Time Of Flight (TOF) detectors (very high time resolution: 10-100ps)

TDCs



- *Counter:*
  - The clock pulses are counted between the START and direct readout in real time.
  - Large dynamic range
  - Good and cheap time references available as crystal oscillators
  - Synchronous to system clock (like accelerator clock), so good for time tagging
  - The resolution is limited to the speed of the counter (1GHz -> 1ns resolution)



- *Charge integration*
  - A capacitor is charged between the START and the STOP signals. The voltage on the capacitor is then digitized by an ADC
  - Limited dynamic range
  - High resolution: ~1-100 ps
  - Sensitive analog circuit needing ADC for final conversion.
  - Sensitive to temperature, etc. so often needs in-system calibration
  - Can be combined with time counter for large dynamic range

Some others logical functions

- Discriminator:
  - *device which responds only to input signals with a pulse height greater than a certain threshold value. If this condition is satisfied, the discriminator responds by issuing a standard logic signal; if not, no response is made.*

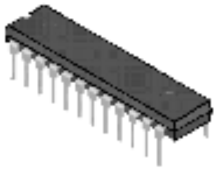
- But sometimes you want to perform more sophisticated tasks than buffering, +, -, ×, ÷, …
- For those you need digital processors : DSP, FPGAs, etc.

Digital Signal Processor (DSP)

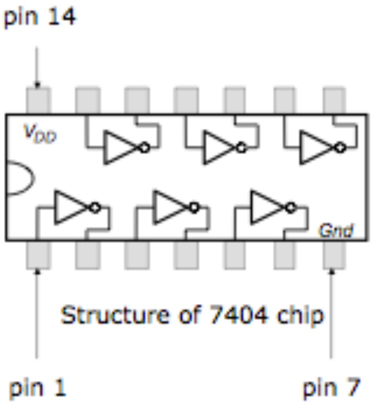
- A digital signal processor is a digital device for which both hardware and software are fully optimized for digital signal processing applications (FFT, filtering).
- ≠ from a general purpose personal computer processor, which has to support a variety of applications
  - *a PC can easily be programmed, but its major obstacle for real-time or online signal processing is that it is developed for general applications*
  - *PC performance can be increased, mainly, by increasing the clock speed, which results in increasing the power consumption (limiting the use of this processor for portable devices).*
- a DSP, as long as it will be used exclusively for digital signal processing applications, can explore some inherent characteristics of this type of applications, like:
  - *Large number of multiply and accumulate (inner product) operations.*
  - *Strong algorithm iteration.*
  - *Circular buffer utilization.*

Field Programmable Gate Arrays (FPGA)

- *Commodity integrated circuits with basic logic blocks are readily available (for instance four AND gates in one package, etc.).*



Dual-inline package

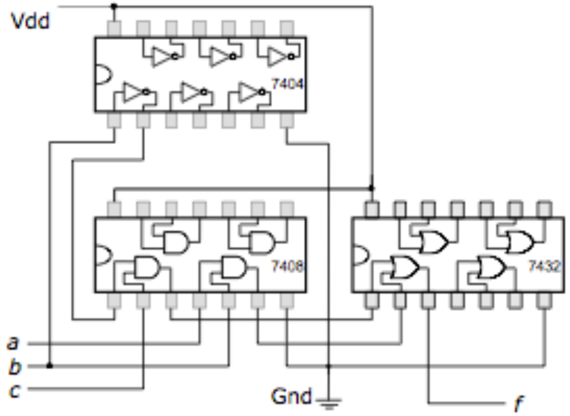


Structure of 7404 chip

- *These circuits can be combined to form simple digital systems.*

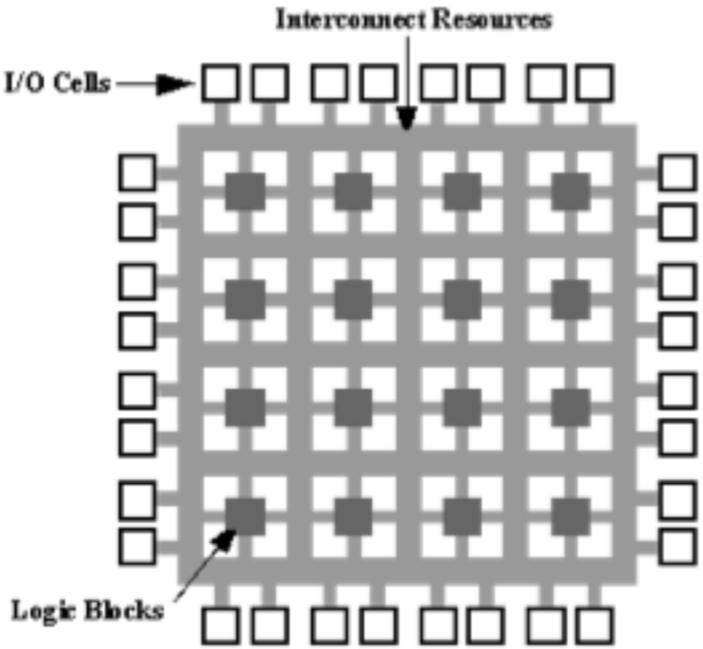
- *However, complex logic systems are no longer designed using individual gates. Instead, logic functions are described in a high-level language (for instance VHDL), synthesized using design libraries, and implemented as custom Integrated Circuit (IC) or programmable gate arrays (FPGA).*
- *The advantage of FPGA over the IC, is that the logic can be re-programmed at any time, while the IC is made once and for good.*

Implementation of f=ab+b'c



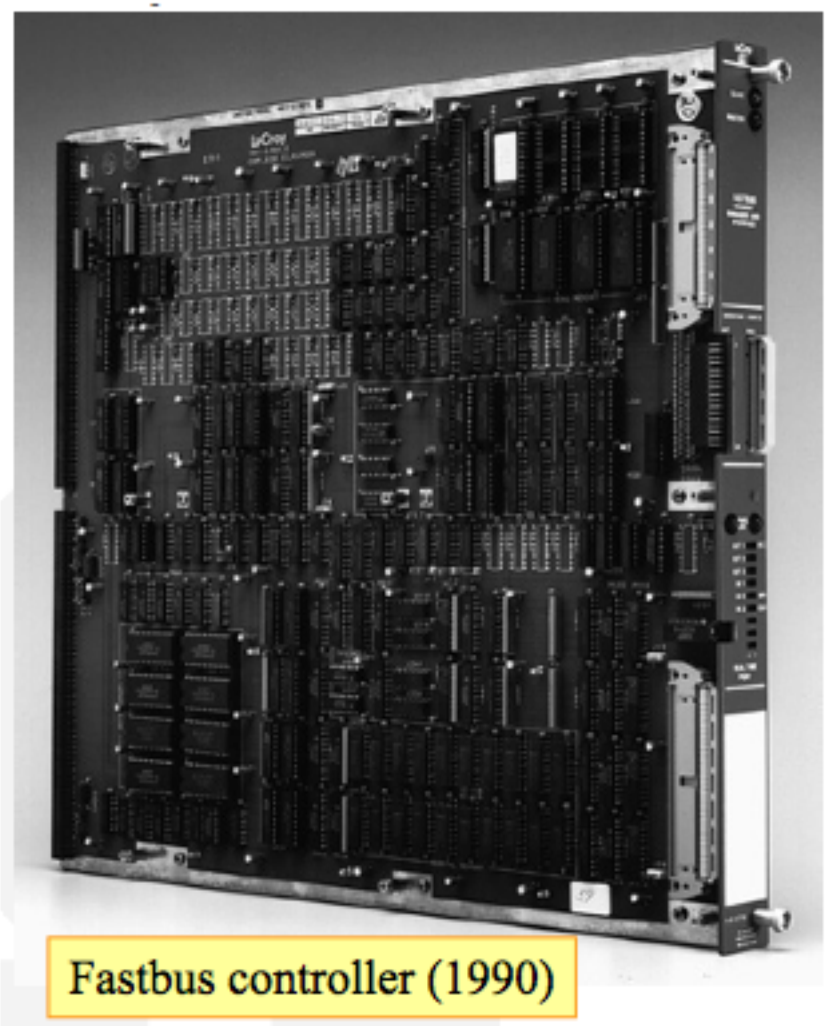
Field Programmable Gate Arrays (FPGA)

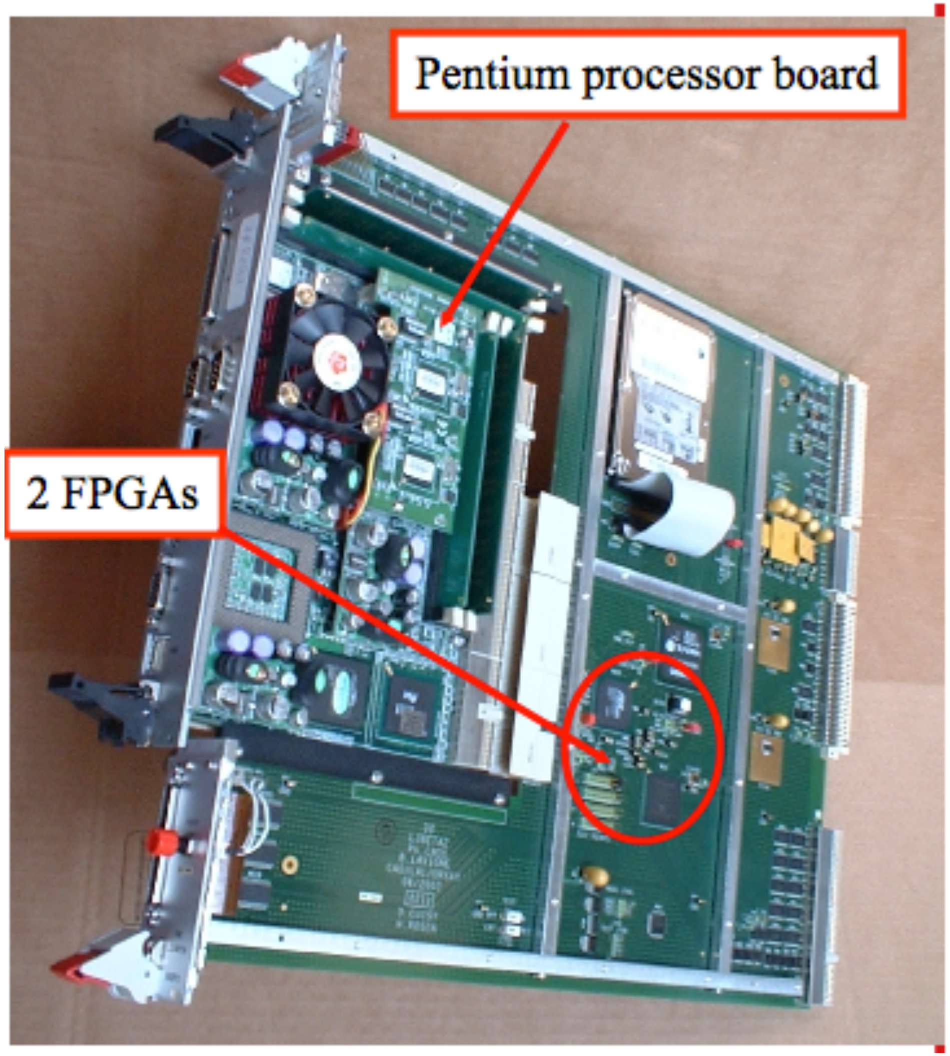
- *FPGAs are built from one basic "logic-cell", duplicated ~O(10^5-10^6) of time.*
- *Each logic-cell can be connected to other logic-cells through interconnect resources*
- *Each cell can do little, but with lots of them connected together, complex logic functions can be created.*
- *Typical clock freq: 500 MHz*
- *Lot of memory available: several 10's of Mb*
- *Even soft core processors can be implemented*



FPGA = Digital (Re)volution

- From stack of circuits to FPGAs:


Fastbus controller (1990)


Pentium processor board

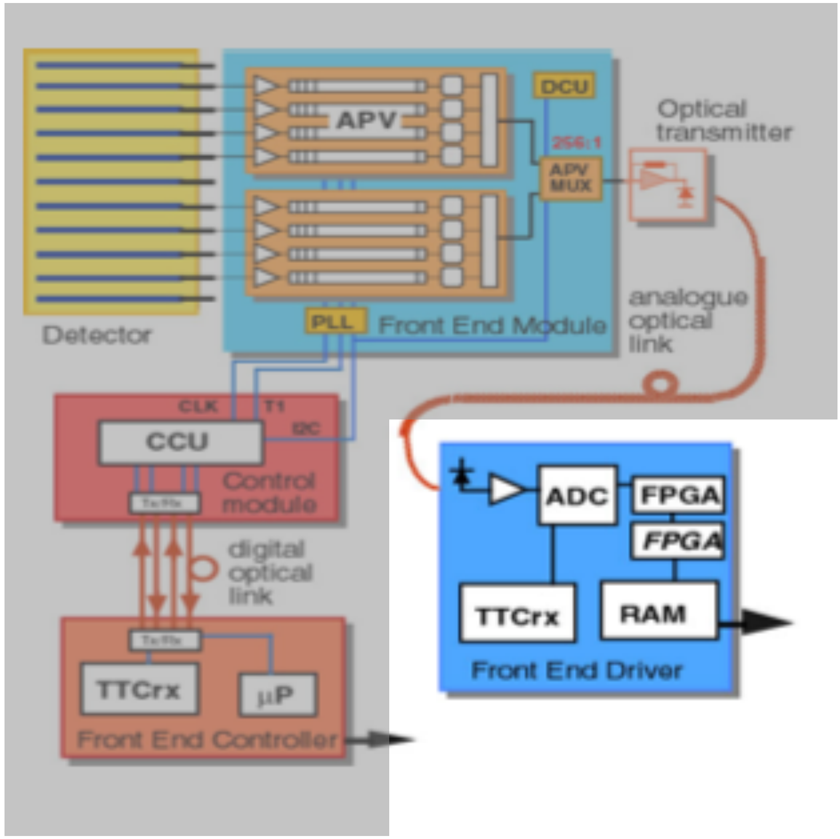
2 FPGAs

Other Digital processors

- Microcontrollers Units (MCU)
  - *can be programmed in Assembly language, but its inner structure is developed for simple applications such as control decisions after taking the comparison of the input signal with some kind of reference value. A more complex application, like a digital filter, would not be matched for this device.*
- CPLD (Complex Programmable Logic Device)
  - *Older than FPGA*
  - *generally used for smaller designs than FPGAs*

Examples of Readout architecture

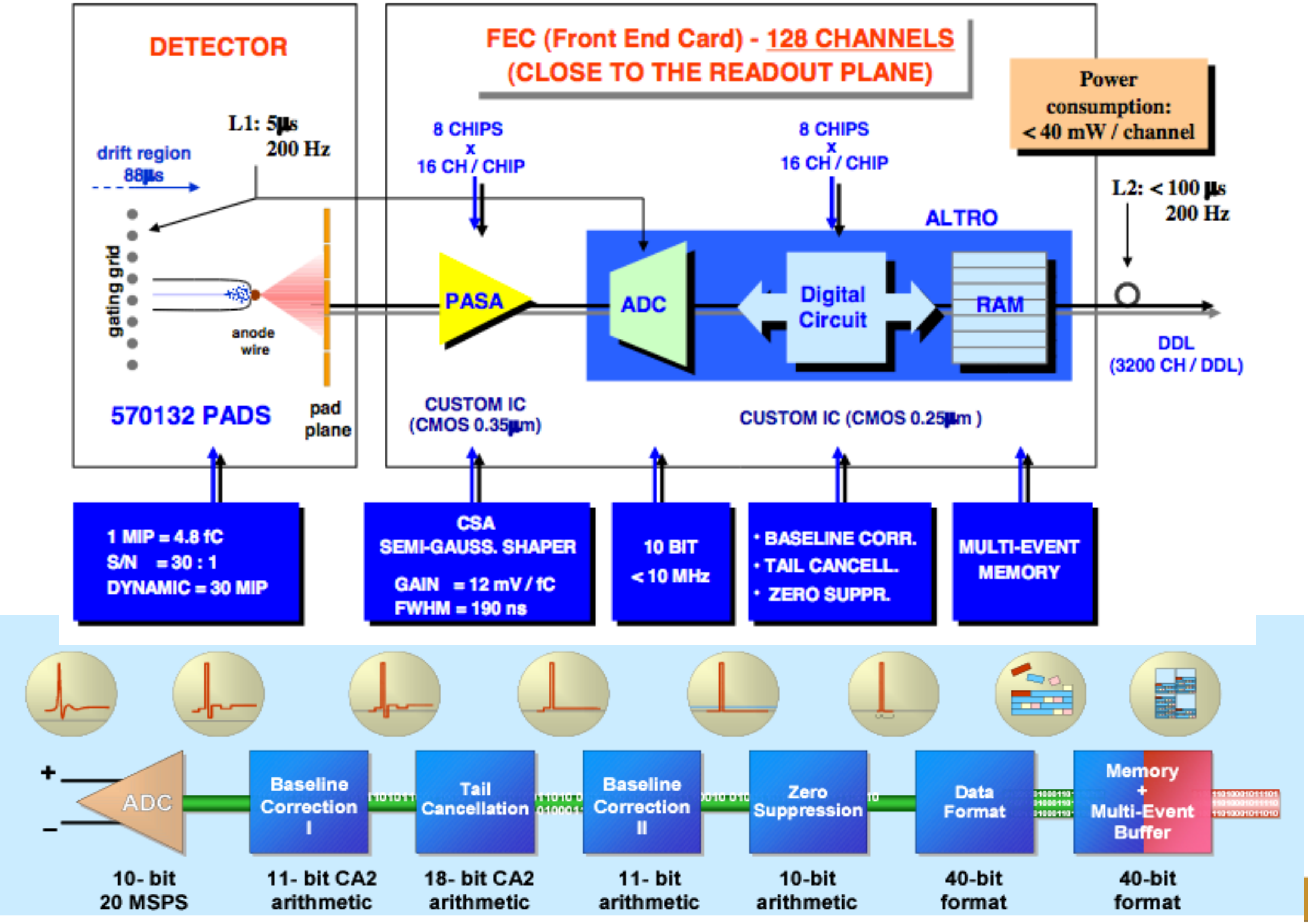
- CMS Si Strip Tracker FE:



- Front End Driver (FED) module housed in the underground counting room outside the cavern.
- Pulse height data from the front-end chips are converted back to electrical levels matched to the range of a 10bit ADC.
- The FED digitises the data, performs some signal processing (FPGA), including reordering and pedestal subtraction, and stores the results in a local memory until required by the CMS data acquisition.
- In high luminosity conditions at maximum trigger rate, cluster finding will reduce the data volume to be transmitted.

Examples of Readout architecture

- ALICE TPC Readout



DETECTOR

L1: 5μs
200 Hz

drift region
88μs

gating grid

anode wire

570132 PADS

pad plane

FEC (Front End Card) - 128 CHANNELS
(CLOSE TO THE READOUT PLANE)

8 CHIPS
x
16 CH / CHIP

8 CHIPS
x
16 CH / CHIP

Power consumption:
< 40 mW / channel

ALTRO

PASA

ADC

Digital Circuit

RAM

L2: < 100 μs
200 Hz

DDL
(3200 CH / DDL)

CUSTOM IC
(CMOS 0.35μm)

CUSTOM IC (CMOS 0.25μm)

1 MIP = 4.8 fC
S/N = 30 : 1
DYNAMIC = 30 MIP

CSA
SEMI-GAUSS. SHAPER
GAIN = 12 mV / fC
FWHM = 190 ns

10 BIT
< 10 MHz

- BASELINE CORR.
- TAIL CANCELL.
- ZERO SUPPR.

MULTI-EVENT MEMORY

ADC — 10- bit, 20 MSPS

Baseline Correction I — 11- bit CA2 arithmetic

Tail Cancellation — 18- bit CA2 arithmetic

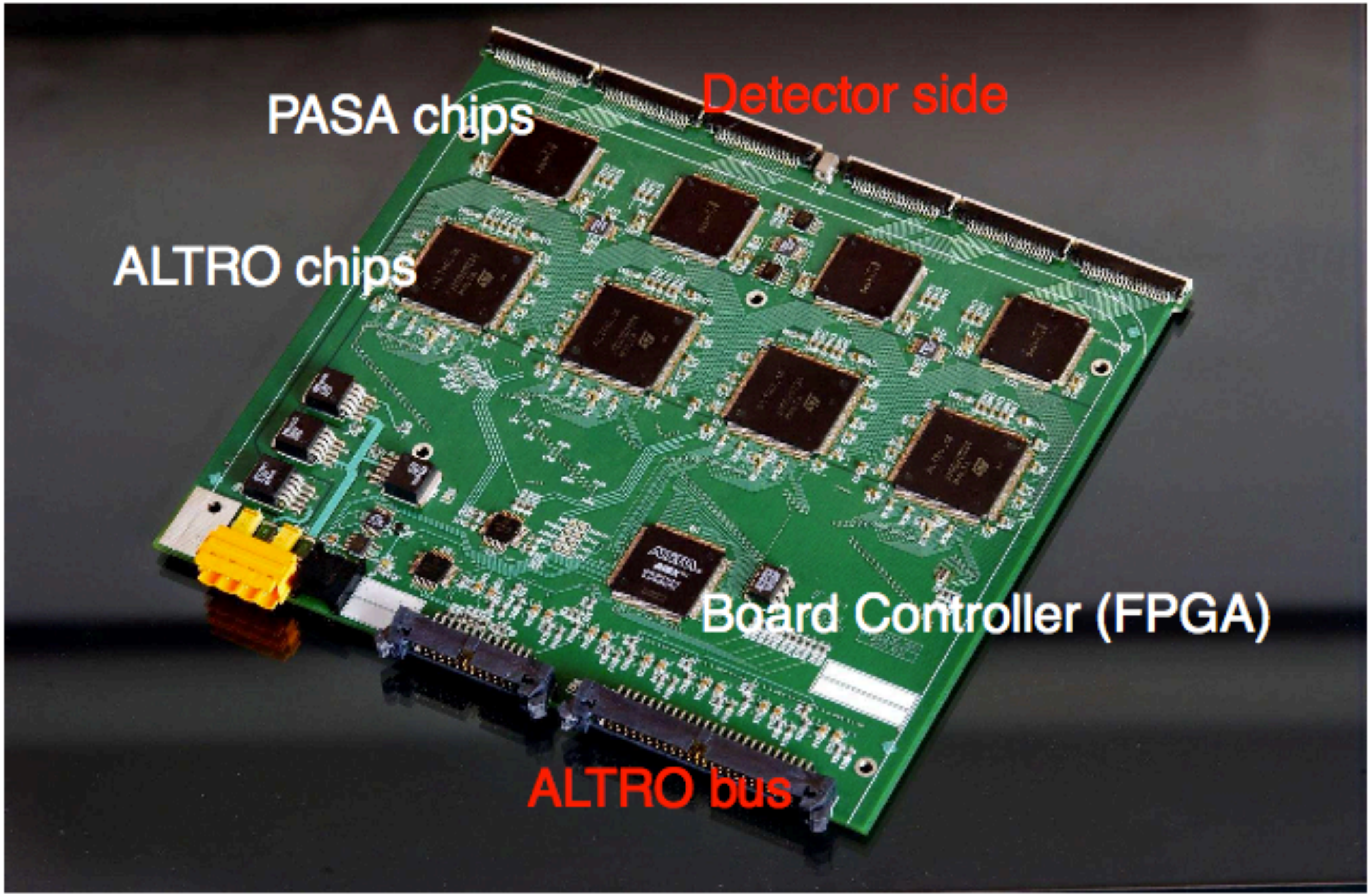
Baseline Correction II — 11- bit arithmetic

Zero Suppression — 10-bit arithmetic

Data Format — 40-bit format

Memory + Multi-Event Buffer — 40-bit format

ALICE TPC Readout



+ 4 PASA chips on rear side

Trigger and Data Acquisition
at colliders

G. De Lentdecker

Université Libre de Bruxelles

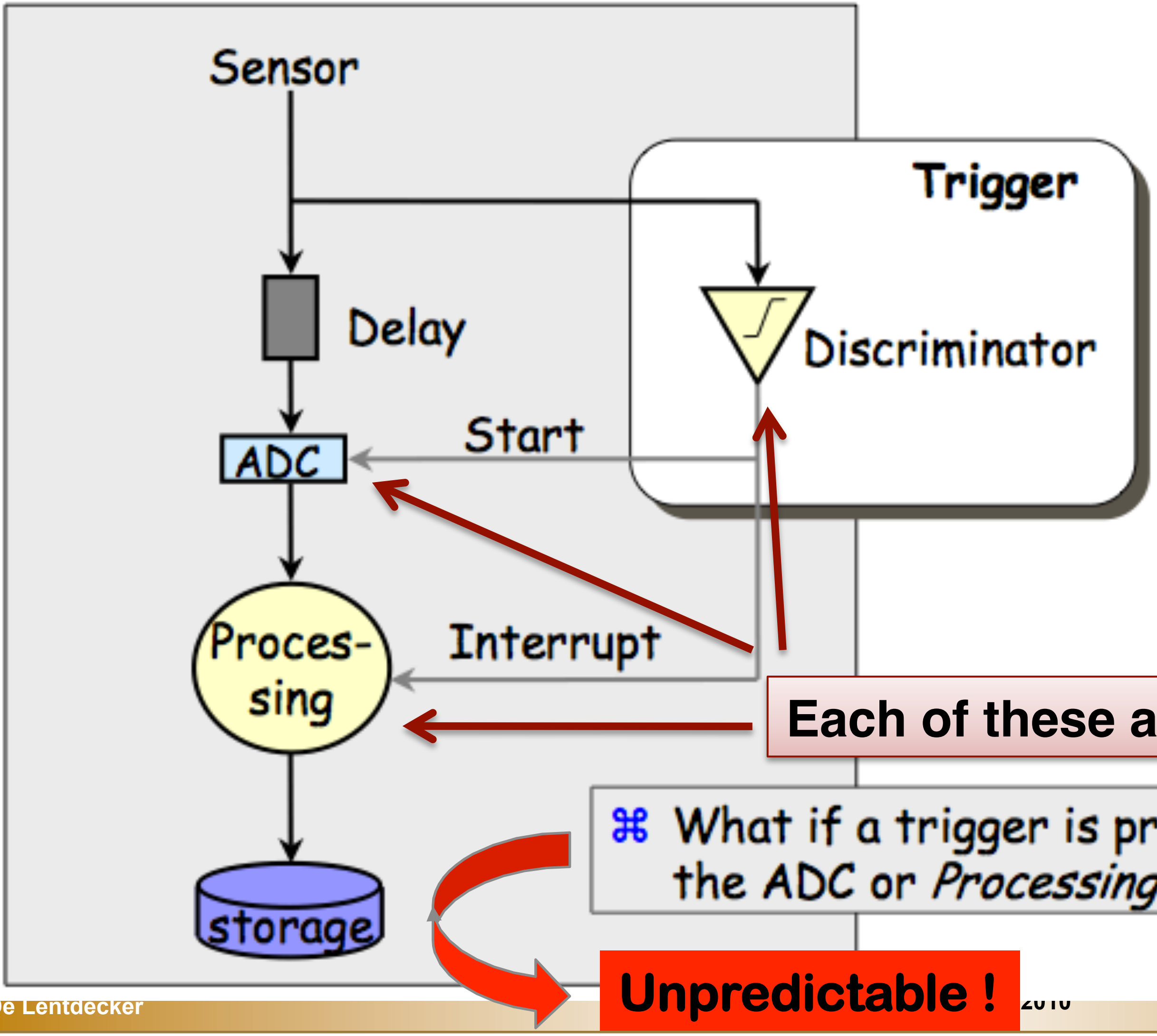
DAQ ARCHITECTURE

Break

- Now we will make a break in the data flow:
  - *We have to introduce a couple of concepts about DAQ architecture and trigger before resuming the data flow*

A Simple DAQ

- Although we haven’t talked about trigger yet, you need a “trigger” to start the data acquisition, for instance in this example the trigger is : “the signal amplitude is above a fixed threshold”:



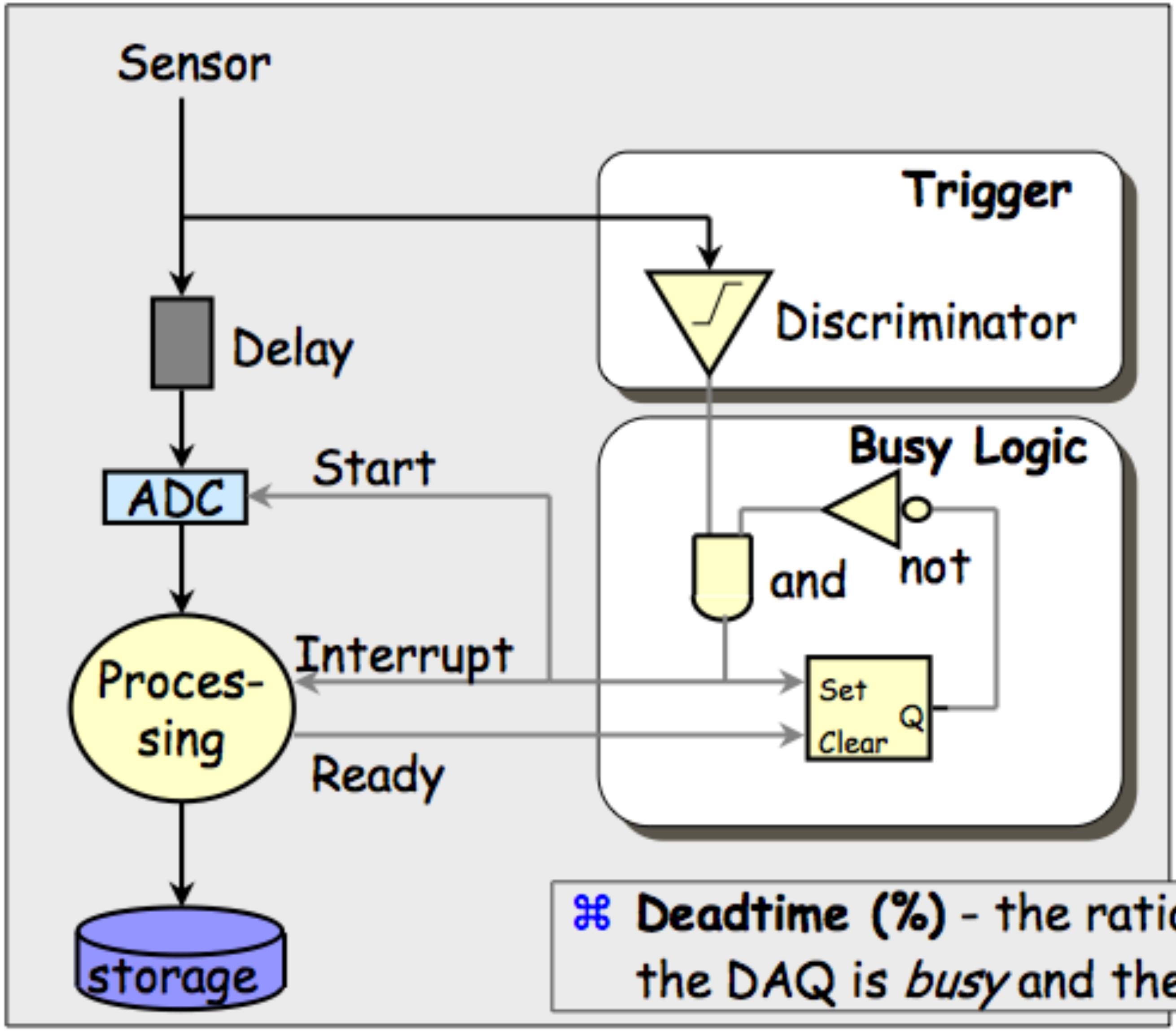
Each of these actions takes some time

⌘ What if a trigger is produced when the ADC or *Processing* is busy ?

Unpredictable !

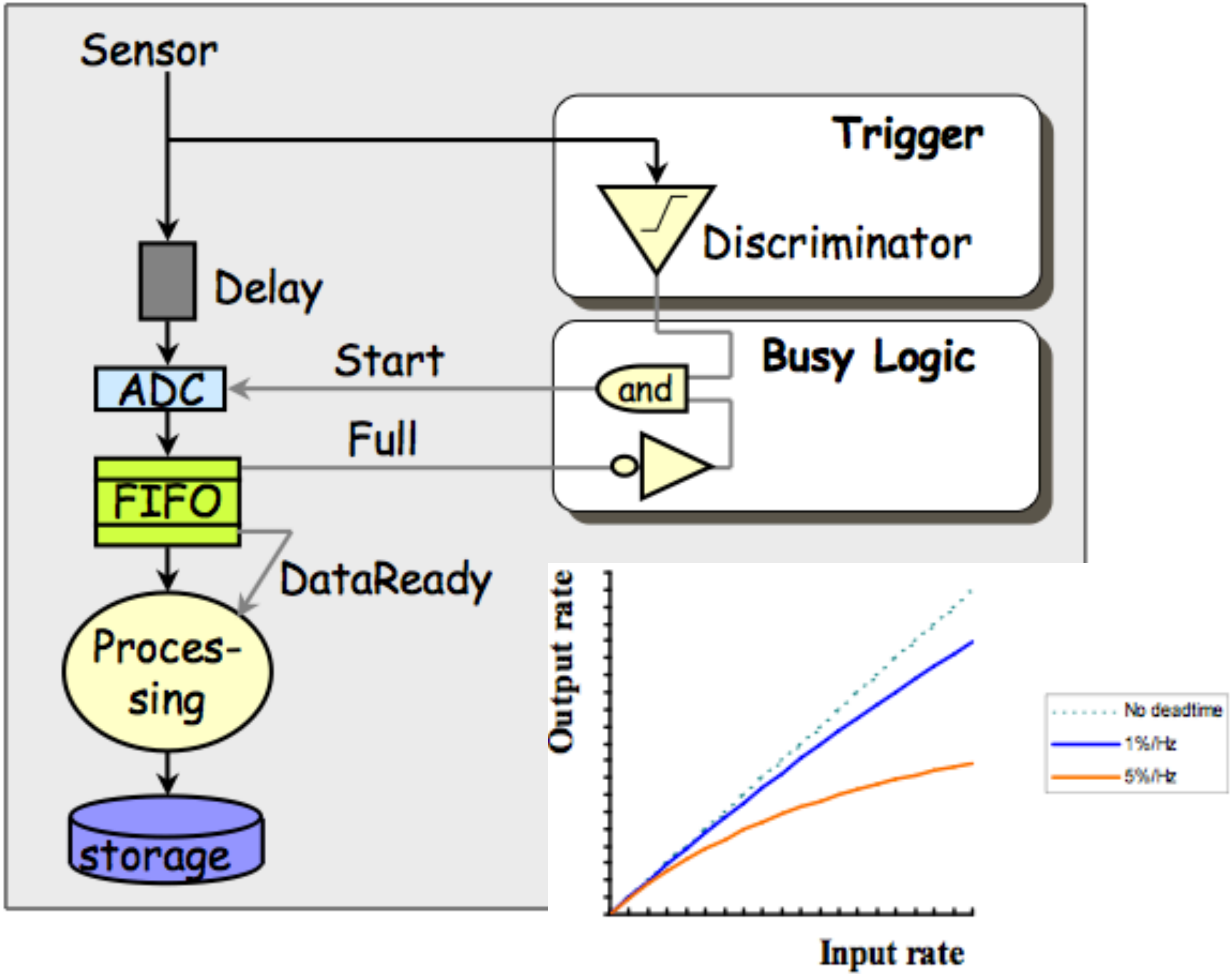
deadtime

- To avoid this unpredictable state, a busy logic is introduced:

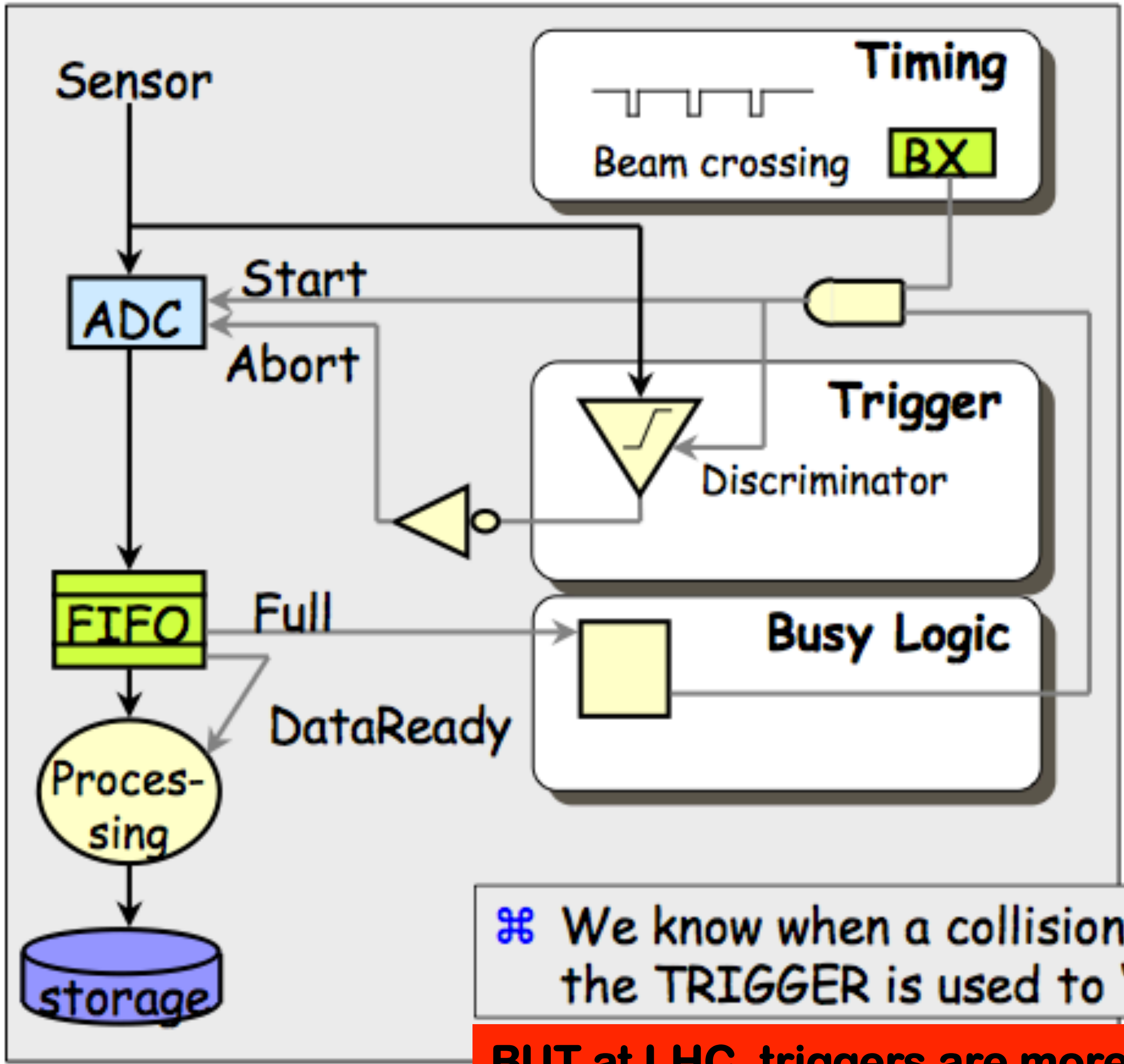


⌘ **Deadtime (%)** - the ratio between the time the DAQ is *busy* and the total time

Reducing the deadtime



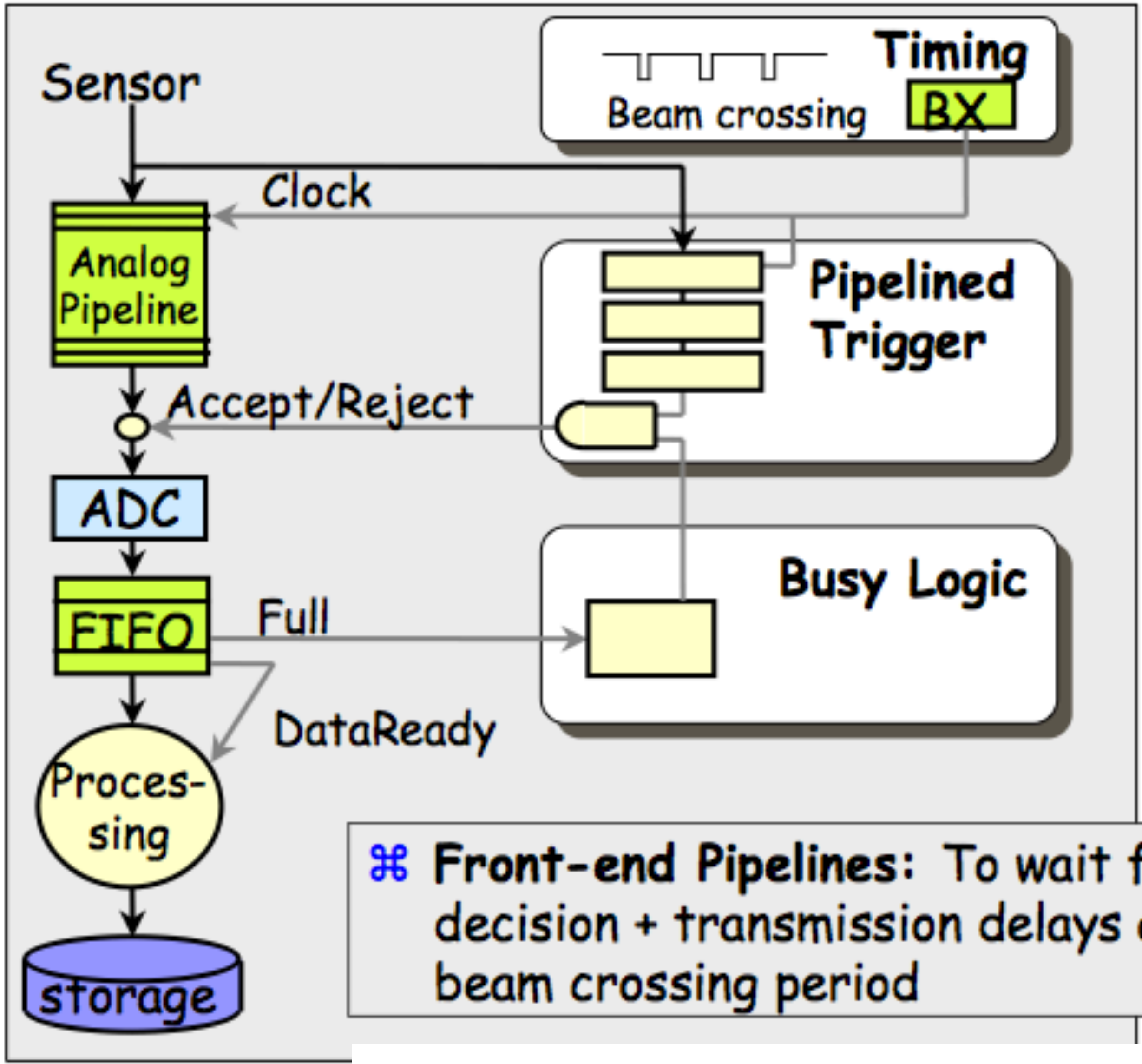
Simple DAQ in collider mode



⌘ We know when a collision happens
the TRIGGER is used to “reject” the data

BUT at LHC, triggers are more sophisticated than just a discriminator; and typically takes ~a few us while LHC bunch crossing is 25 ns ?!?

Simple DAQ at LHC



⌘ **Front-end Pipelines:** To wait for the trigger decision + transmission delays are longer than beam crossing period

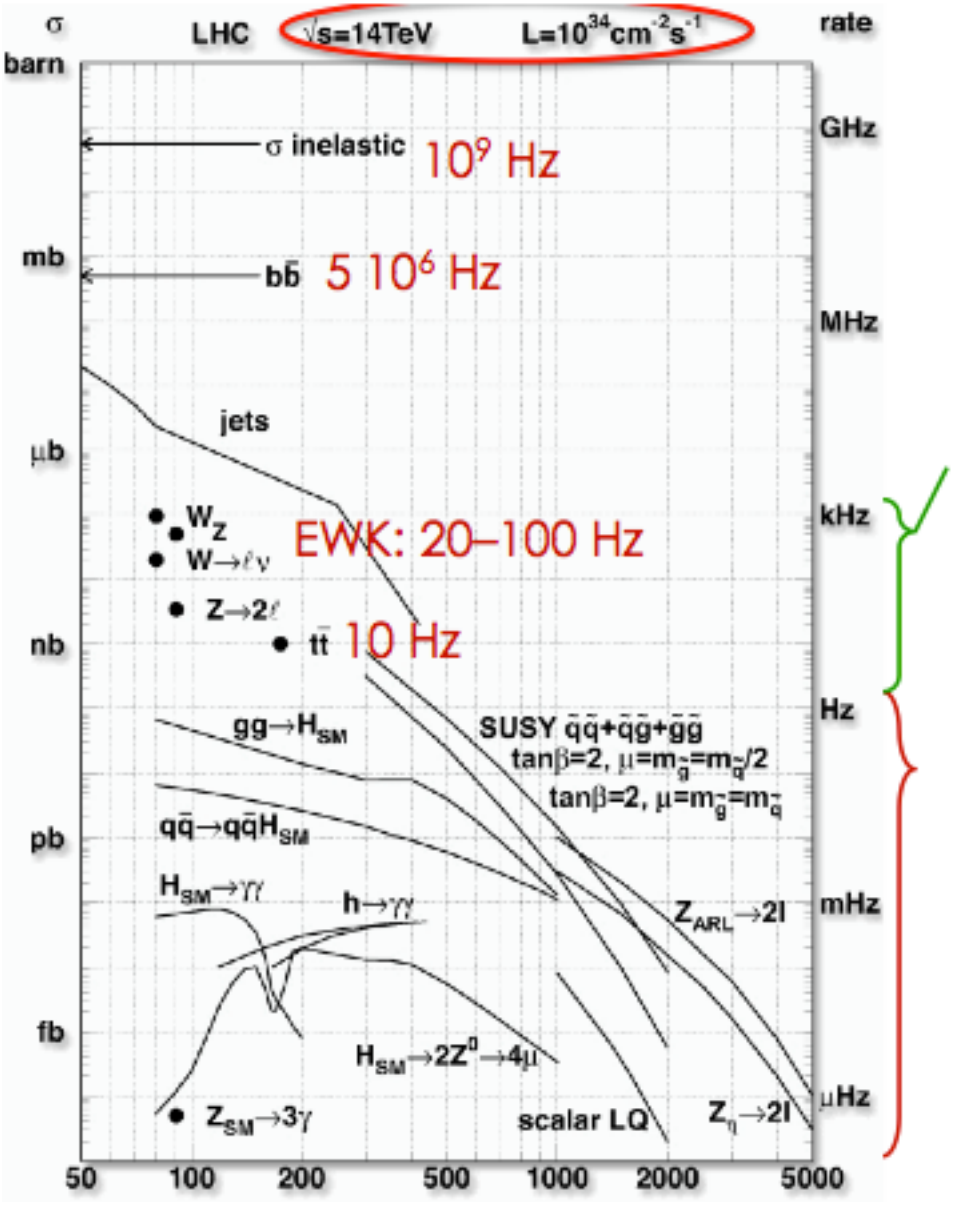
The Time the data are buffered waiting for the trigger is called trigger latency

TRIGGER

Why a trigger ?

- Today a typical PC runs at several GHz, so couldn't we just use the LHC Clock (40 MHz) to trigger the DAQ and use a large number of (relatively) cheap PC for data processing ?
- The answer is no, of course:
  - *No (affordable) DAQ system could read out O(10^7) channels at 40 MHz → 600 TBit/s to read out – even assuming binary channels!*
  - *In addition most of these millions of events per second are totally uninteresting: one Higgs event every O(100) seconds*
- → we need event selection, that is a "trigger"

- Hence new questions arise:
  - *What trigger rate can we afford ?*
  - *How do we select the data ?*
  - *How do we make sure that the values from the many different channels refer to the same original event (collision)?*



- Total collision rate ~ 1 GHz @ L = 10^{34}cm^{-2}s^{-1} and 14 TeV

- "Interesting" physics is about 6-8 orders of magnitude smaller

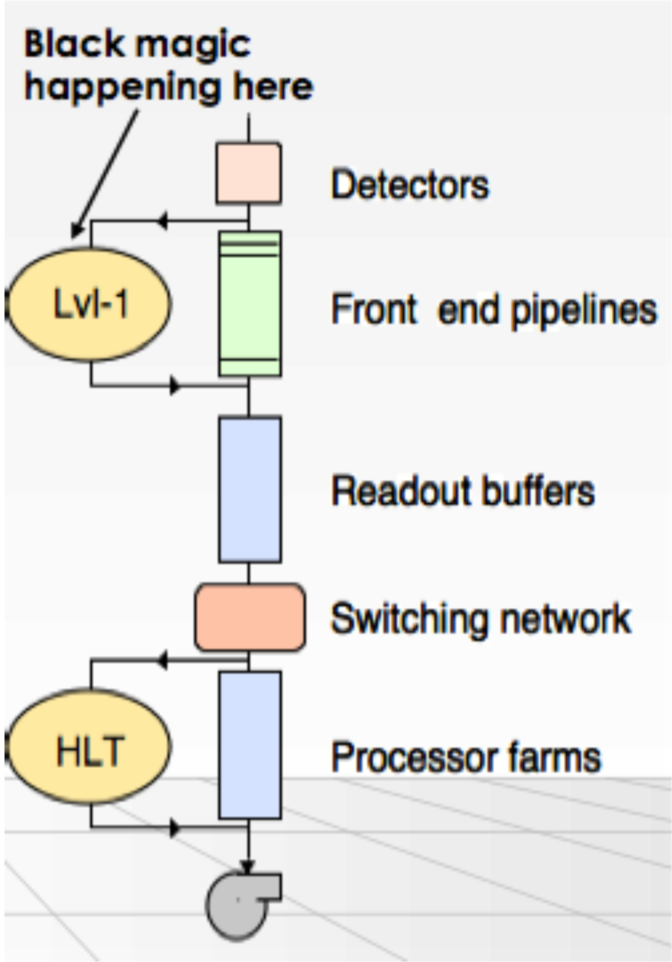
- New physics is 9 orders of magnitude smaller

- We "just" need to identify and select these rare processes from the overwhelming background before reading out and storing the whole event.

Trigger

- Usually the reduction of the data rate to a reasonable rate of O(100 Hz) that can be archived cannot be achieved in one step
- Therefore the trigger is split in several levels (usually called level-1, 2, 3,…). Each level performs a reconstruction of the data before applying a selection.
- Each trigger level reduces the data rate => subsequent trigger levels have more time to perform the data processing.
- Usually the first level (Lvl-1) is hardwired
  - *Short latency: a few μs.*
  - *In HEP Lvl-1 usually uses data from calorimeters and muon detectors*
- The second level (Lvl-2)
  - *Processor based (standard CPU's or dedicated custom/FPGA processing)*
  - *Latency : a few ms.*
- The higher levels are usually software.
  - *The reconstruction program is run in parallel on a large number of CPUs O(100-1000).*
  - *Having more time (~1s) the algorithms can be more complex and can use all the detector data, including the tracker data.*

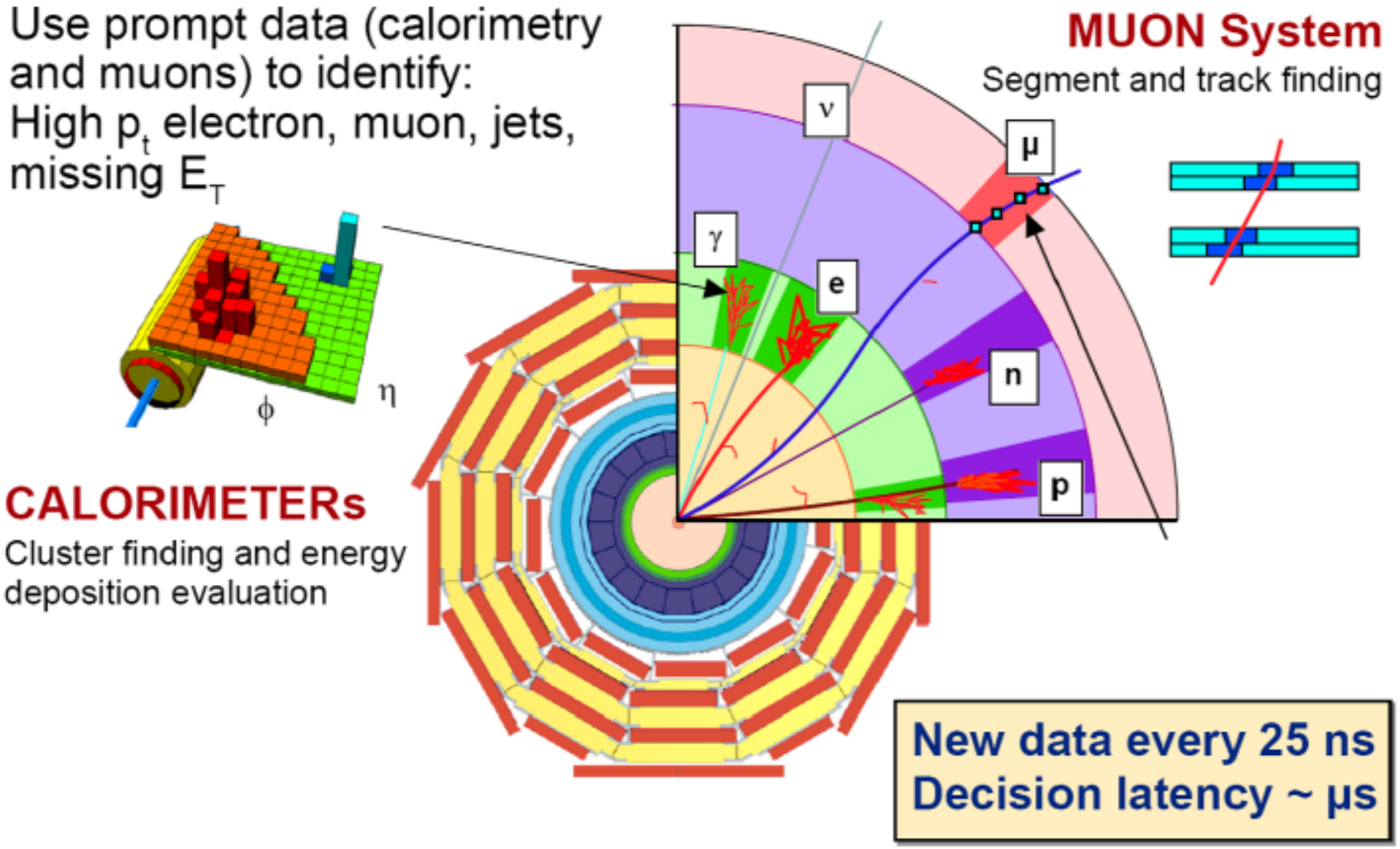
Multilevel trigger



- Level1 (the black magic box):
  - *latency ~ 1μs*
  - *O(10^7) channels*
- Work with local information to avoid many interconnections
- Must be fast and robust
  - *Look for simple signatures*
  - *implemented in hardware*
  - *Use Calo and muon data*

Fortunately Mother Nature is kind

- p-p collisions mostly produce low p_t particles ~1 GeV
- Physicists are interested in heavy particles (W, Z, Top, Higgs, Z', W', sparticles,…) often decaying in high pt leptons or jets (pt>25 GeV)
- Therefore Level1 triggers will often look for such high p_t leptons and jets (as well as large MET):



Use prompt data (calorimetry and muons) to identify:
High p_t electron, muon, jets, missing E_T

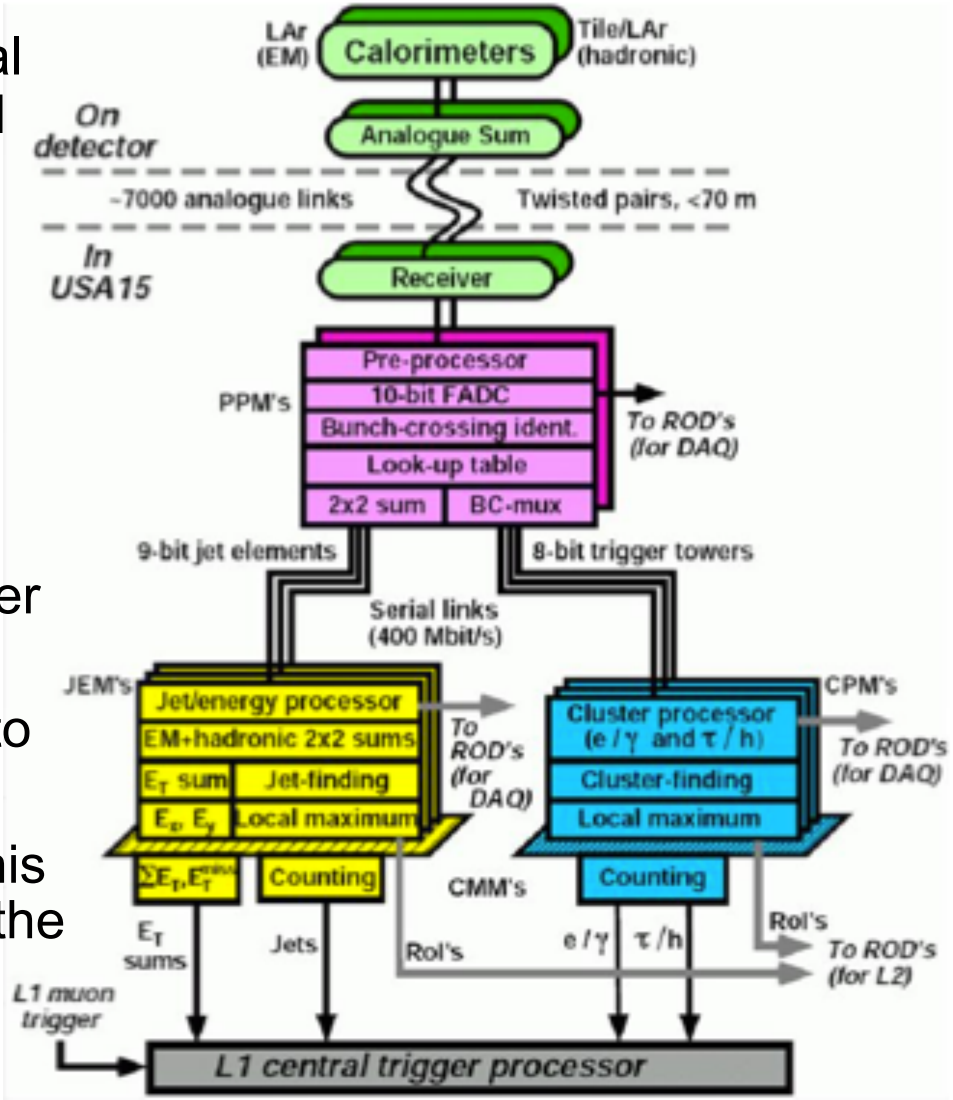
MUON System
Segment and track finding

CALORIMETERs
Cluster finding and energy deposition evaluation

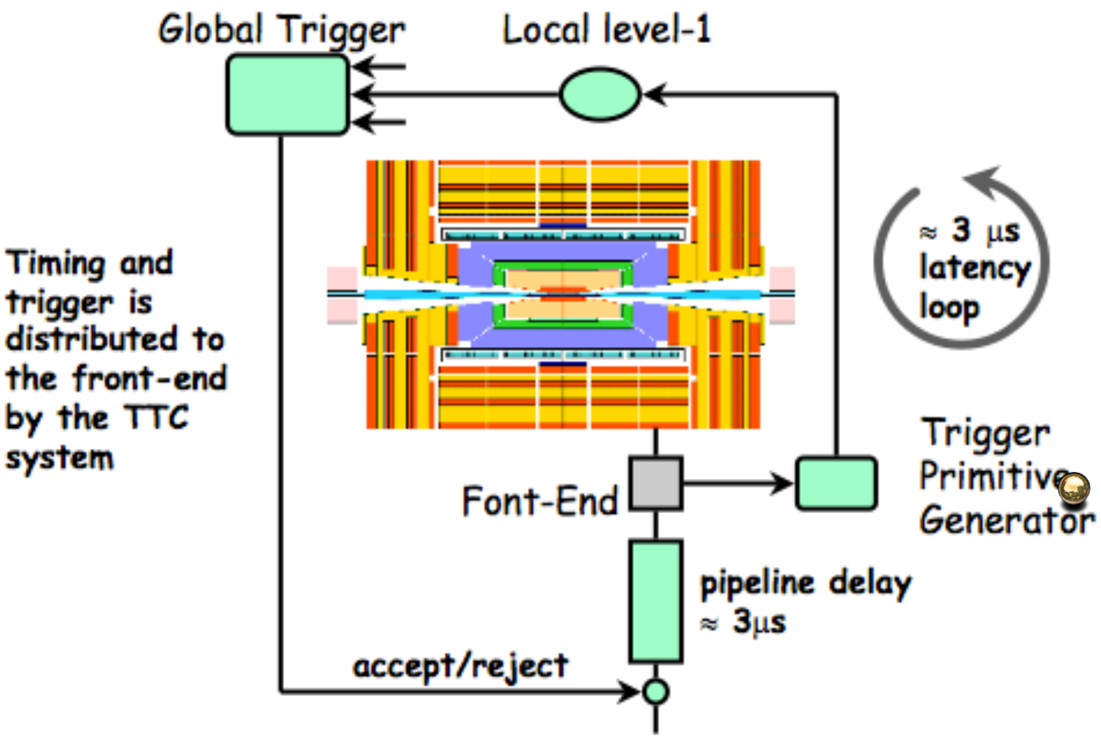
New data every 25 ns
Decision latency ~ µs

ATLAS Level1 Calo Trigger

- 7000 trigger towers (EM+HA) 0.1x0.1 in eta,phi space
- On FEE, analogue signals of individual cells are summed to form tower signal
- Signal transmitted to Pre-processor (PPM): digitization and E_t estimate at 40 MHz
- Tower data are transmitted to cluster processor (CPM); there are 4 crates
- e/γ processing on large FPGA
- e/γ candidates sent to L1 central trigger processor 1.5 μs after bunch crossing
- Note the remaining L1 latency is due to signal transmission !
- After Level1 has made his decision, this information has to be sent back to all the FEE



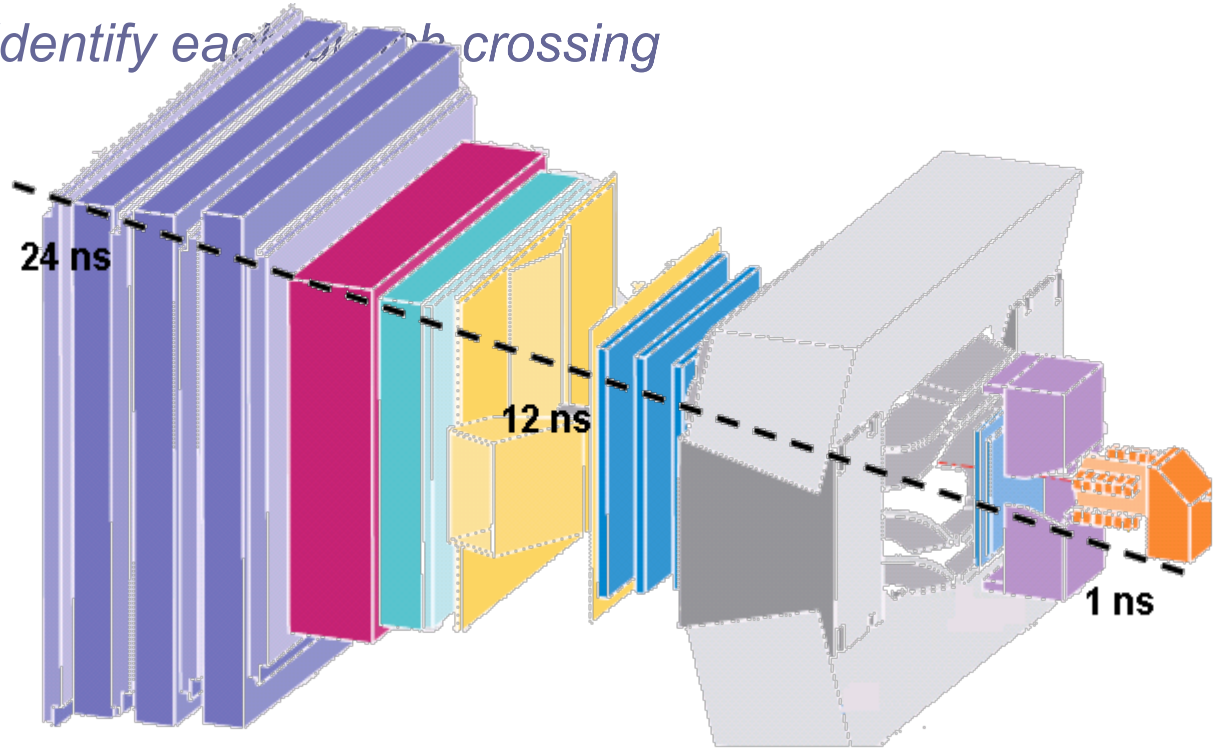
Level1 Trigger distribution


Global Trigger
Local level-1
Timing and trigger is distributed to the front-end by the TTC system
≈ 3 µs latency loop
Trigger Primitive Generator
Font-End
pipeline delay ≈ 3µs
accept/reject

- Now that the magic box tells for each Bunch Crossing if the event has to be kept or rejected, this information has to be sent back to the FEE so that they can send or discard their data.
- Synchronisation at the output of the pipeline is not trivial => needs timing calibration

Level1 Timing Issues

- Need to synchronize all detectors components to better than 25ns
  - *The system need a reference clock*
- But for some detectors, signal development/time of flight > 25 ns
  - *Integrates more than one bunch crossing information*
  - *Particles travel 7.5 m in 25 ns*
  - *Need to identify each bunch crossing*

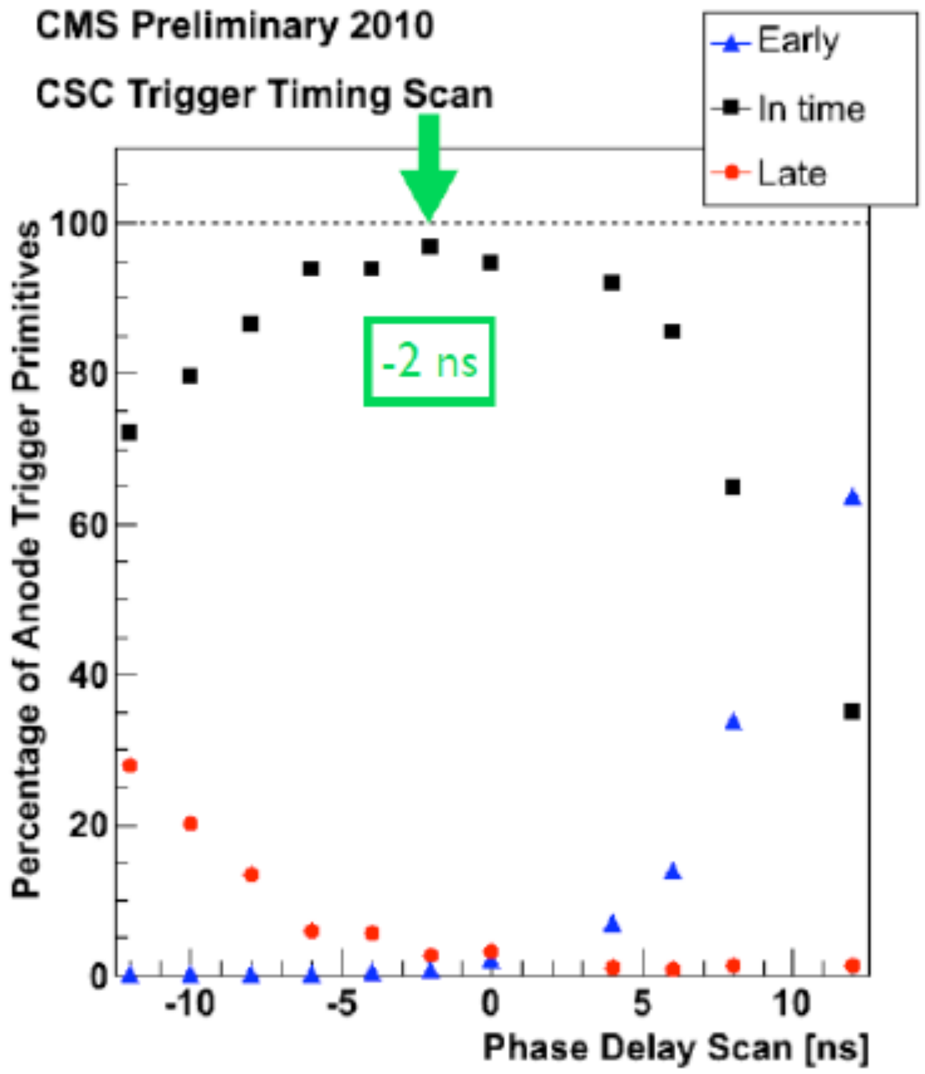


LHC Clock

- At LHC, there is common system for all LHC experiments:
  - *The Timing, Trigger and Control (TTC)*
- TTC is based on the LHC Clock
  - *The clock is extracted by monitoring the passage of the proton bunches*
- It is sent to millions of components in ALICE, ATLAS, CMS & LHCb (by optical fibers)
- TTC also sends the trigger signal to the FEE
- Before starting to data taking, all the components had to be aligned in time
- System has to be robust :
  - *A loss of synchronization may be very difficult to be detected…*

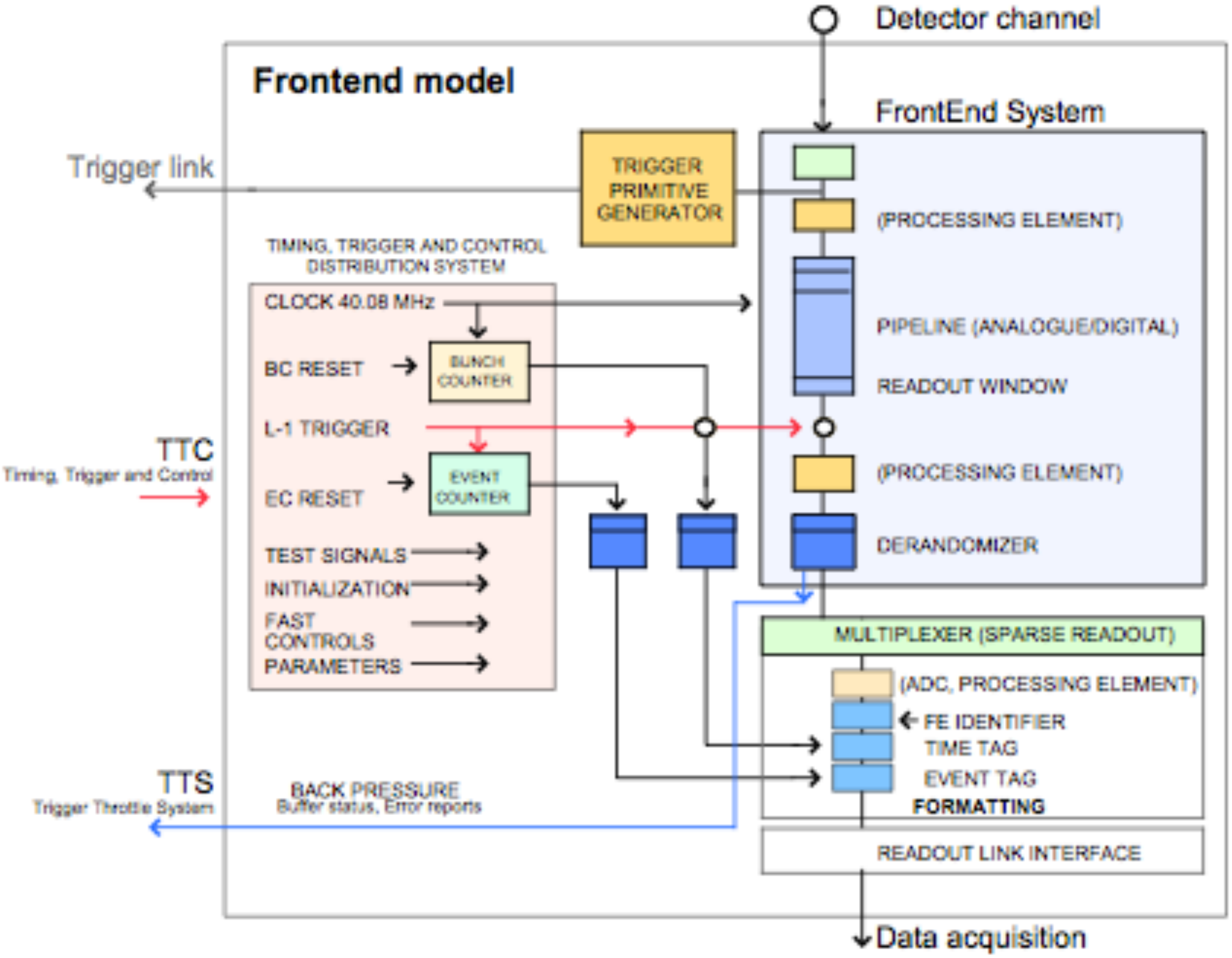
Timing synchronization

- Example : CMS CSC chambers
  - *Perform Timing scan:*
    - Scan a range of timing delays
    - Find best alignment:
      - – highest % of Anode Trigger Primitives



LHC Front-End examples:

- CMS FE model:

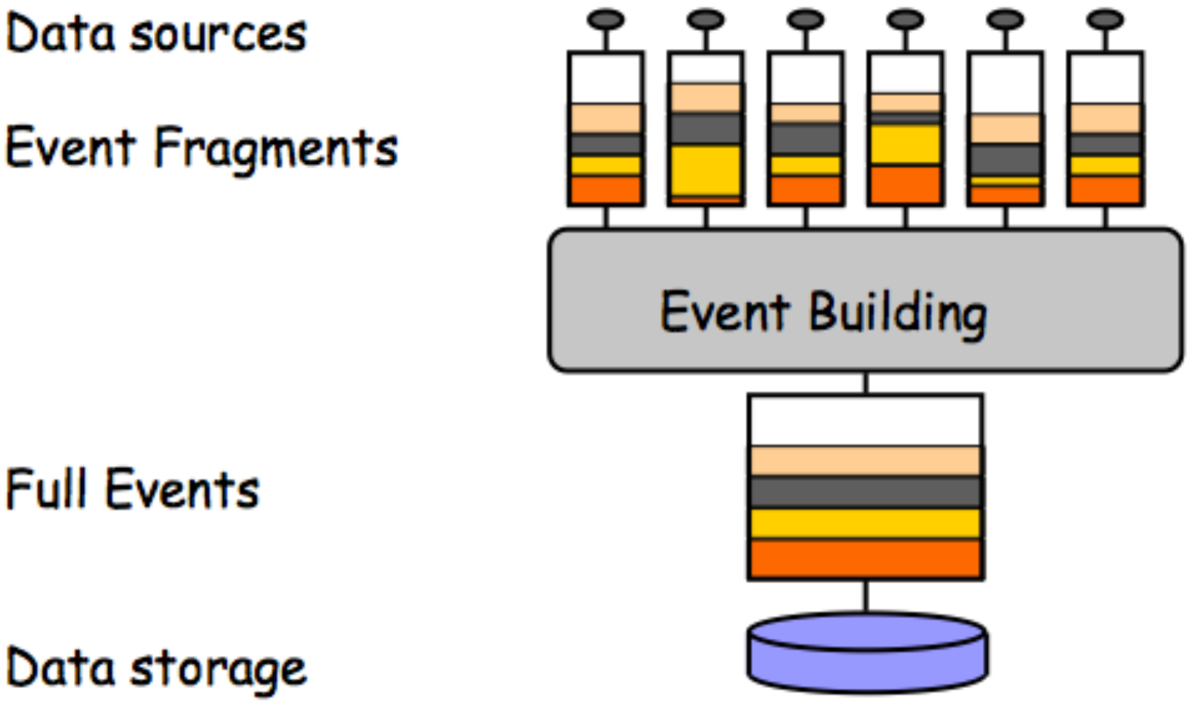


Where are we so far ?

- Event-data are now digitized, pre- processed and tagged with a unique, monotonically increasing number
- The event data are distributed over many electronics boards ("sources")
- For the next stage of selection, or even simply to write it to tape we have to get the pieces together

  => Need and Event Builder (EVB)
- But don't forget:
  - *At LHC the Level1 will typically accept 1/10000 events*
    - Output L1 rate is still very high ~100kHz
  - *Amount of data is still very large:*
    - Typical event size ~1Mbyte (after zero suppression or data compression)
- => need a bandwidth of 100 Gbytes/s (=800 Gbits/s) to extract the data from the Front End.

Event Builder Basic's



Data sources

Event Fragments

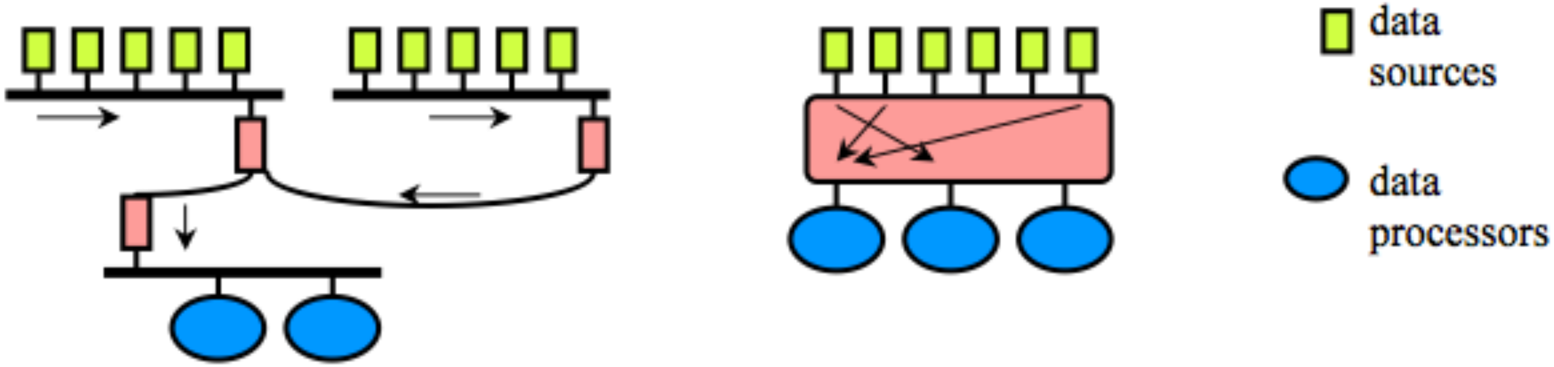
Event Building

Full Events

Data storage

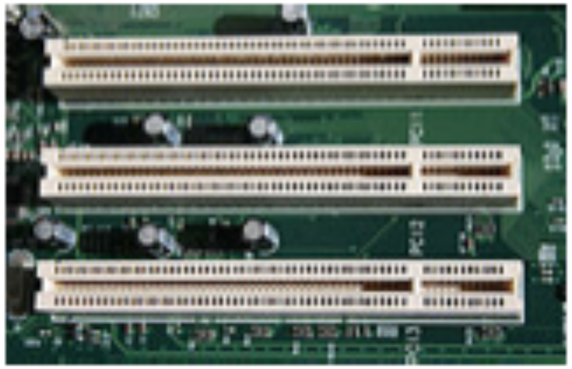
- But which technology to use?

Bus ***OR*** ***switch ??***

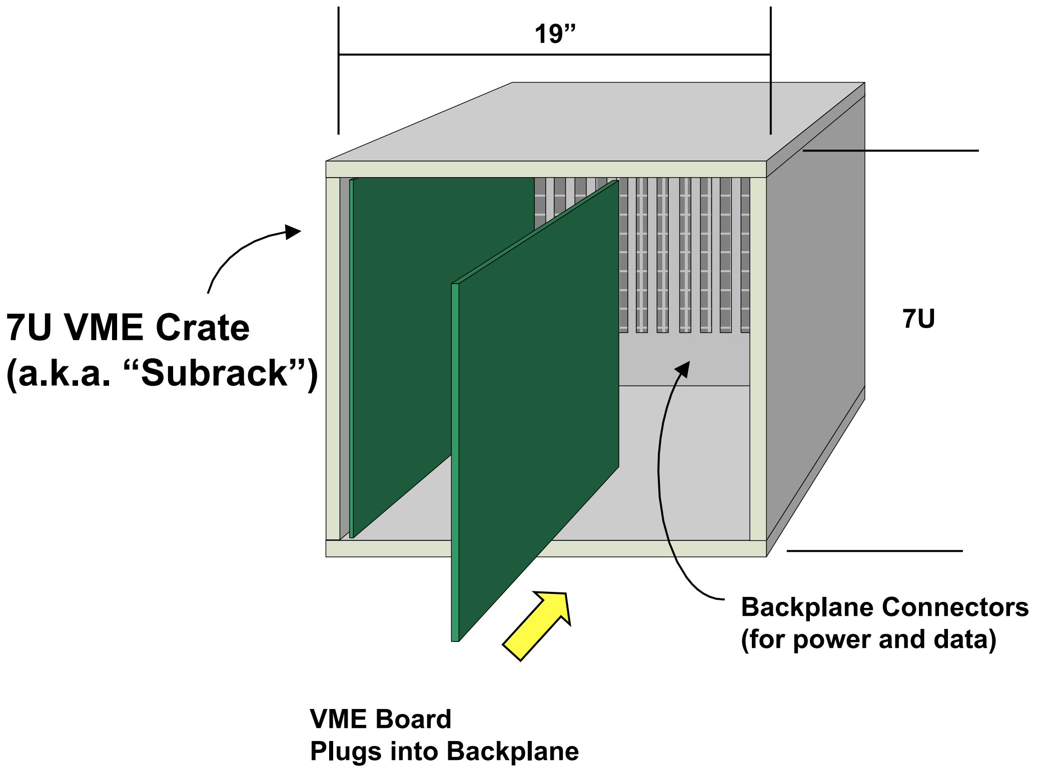


Buses

- Let’s do a parenthesis here on common buses used in HEP, because even old ones are still very popular in our labs…
- What is a bus ?
  - *A bus is a common electrical connection between several electrical devices*
  - *A bus can either allow signals to be transferred between devices, the summing (mixing) of output signals from the devices or the distribution of input signals or power amongst the devices.*
  - *A bus often takes the form of a wire or printed circuit conductor that terminate at multiple connectors which allows devices to be plugged into the bus*
  - *A common example is the PCI bus in PCs:*



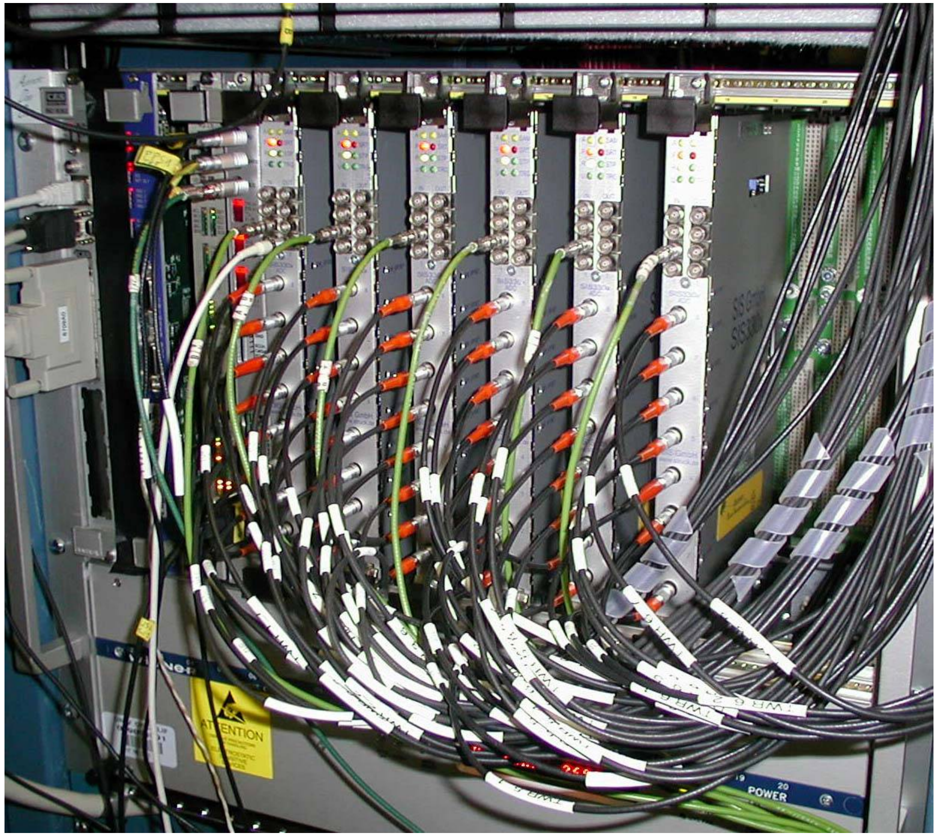
Buses in HEP (example VME)



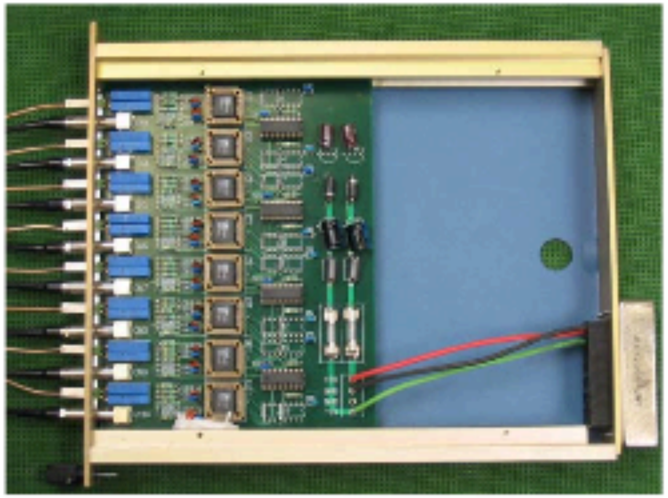
- Buy or build readout boards with some inputs
- Put many of these multi-port modules together in a common chassis or crate
- The modules need
  - *Mechanical support*
  - *Power*
  - *A standardized way to access their data (our measurement values)*
- All this is provided by standards for (readout) electronics such as VME (IEEE 1014)

VME Example:

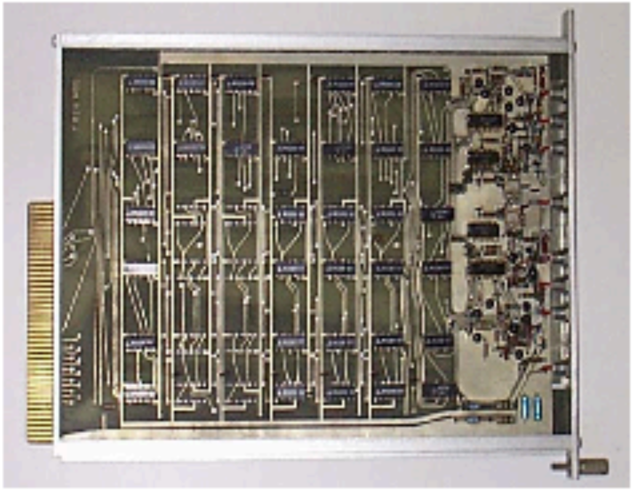
- The standard (here VME) defines:
  - *mechanical standard*
  - *electrical standard for power on the backplane*
  - *signal and protocol standard for communication on the bus*



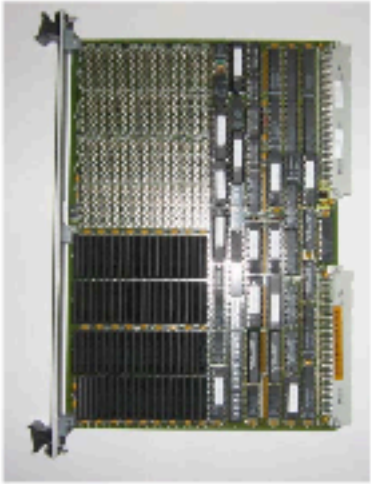
Some famous buses in HEP



NIM, 1964

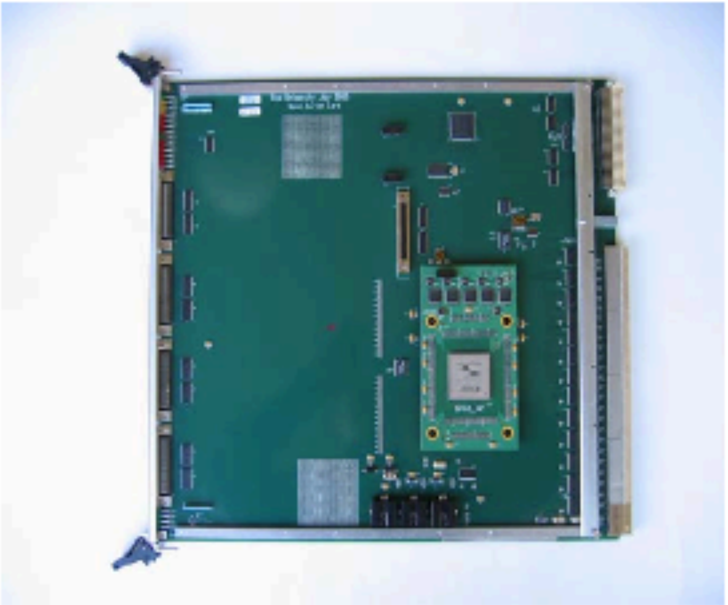


CAMAC, 1969



VME 6U, 1981

FASTBUS, 1986

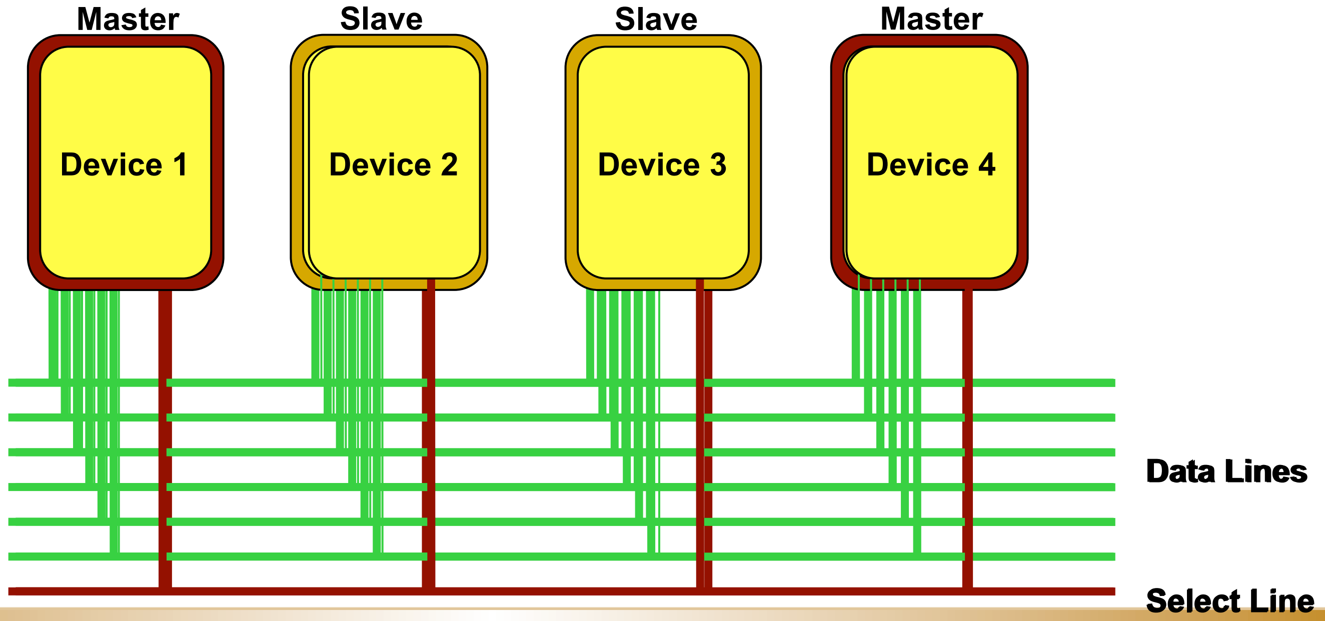


VME 9U, 1994

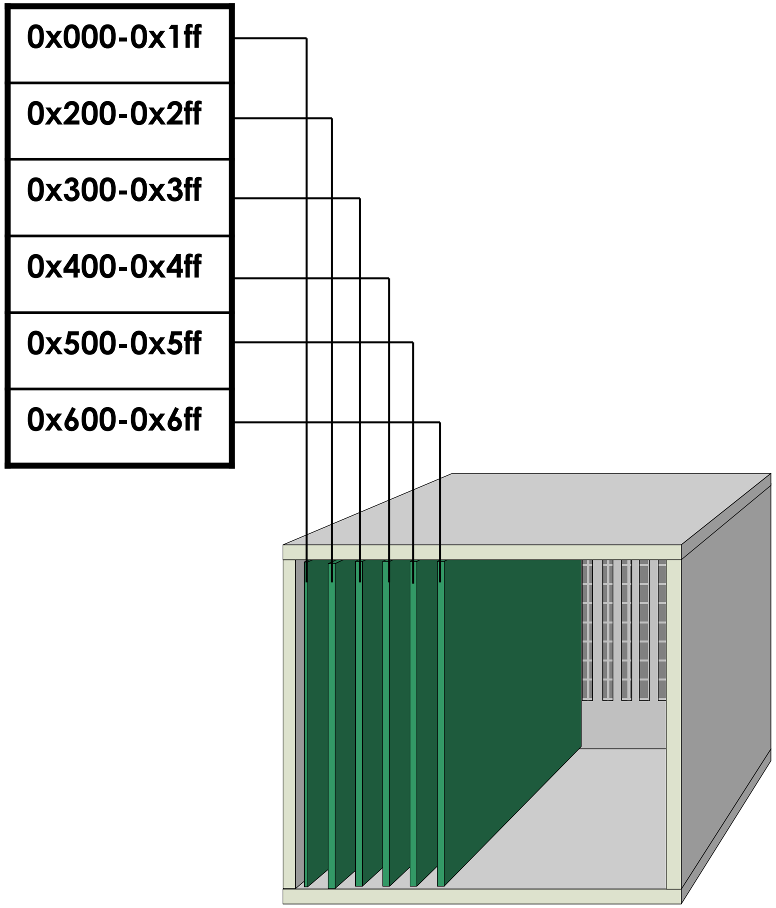
CompactPCI, 1995

Bus communication in a VME crate

- A bus connects two or more devices and allows them to communicate
- The bus is shared between all devices on the bus → arbitration is required
- Devices can be masters or slaves (some can be both)
- Devices can be uniquely identified ("addressed") on the bus

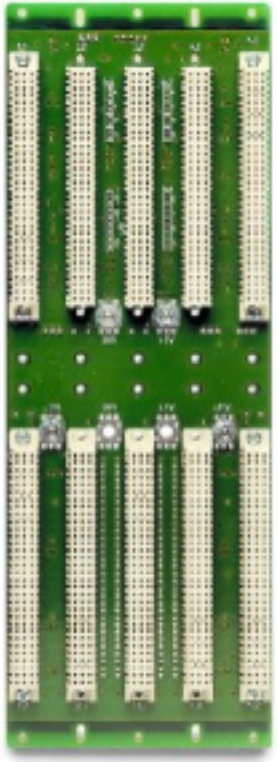


The VME bus



- In a VME crate we can find three main types of modules
  - *The controller which monitors and arbitrates the bus*
  - *Masters read data from and write data to slaves*
  - *Slaves send data to and receive data from masters*
- Addressing of modules
  - *In VME each module occupies a part of a (flat) range of addresses (24 bit to 32 bit)*
  - *Address range of modules is hardwired (conflicts!)*

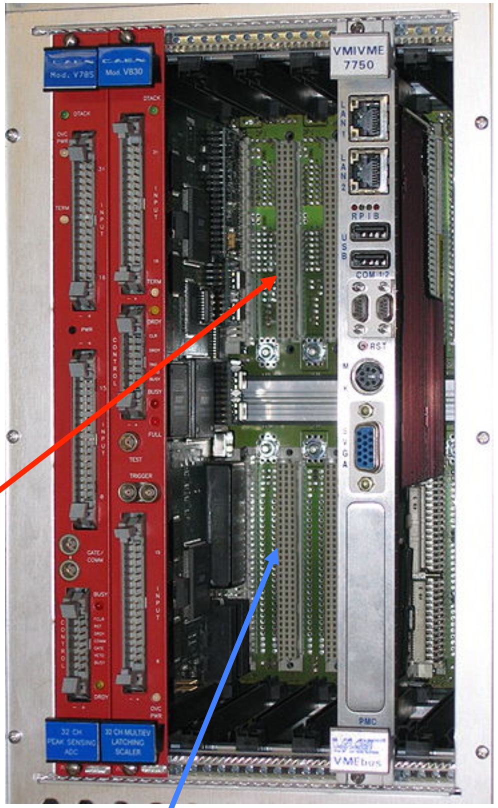
VME specifications



Mandatory bus

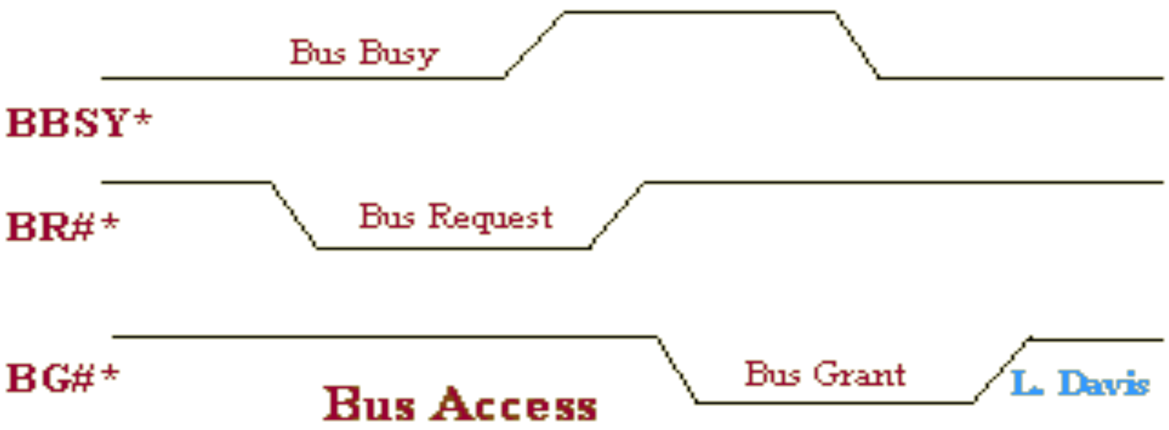
Extension bus

- <15:0> 16-bit Data Bus
- <23:0> 24-bit Address Bus
- 9 lines for Arbitration Bus
- Clock, Control, Status signals
- Power: +5V, +12 V, -12V
- GND: 5 lines



- <31:16> 16-bit Data Bus Extension
- <31:24> 8-bit Address Bus Extension
- Additional power lines and GND lines

VME Bus at Work

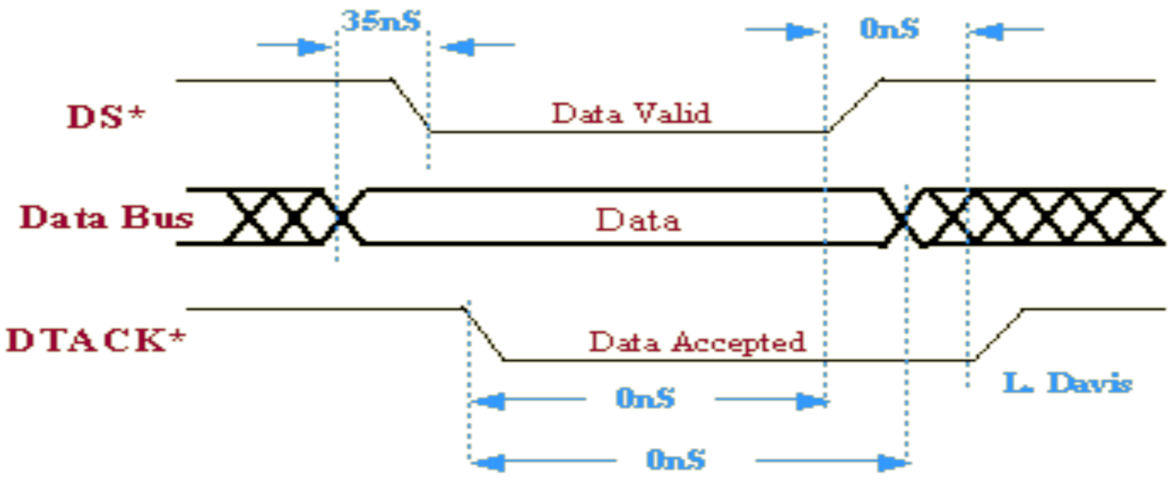


- Arbitration: Master asserts(*) BR# (Bus Request), Controller answers by asserting BG# (Bus Grant)
- If there are several masters requesting at the same time the one physically closest to the controller wins
- The winning master drives BBSY* high to indicate that the bus is now in use

(*) *assert means driving the line to logical level 0 (VME control lines are inverted or active-low)*

VME data transfer

- The Master writes data and address to the data / address bus respectively
- It asserts the lines DS* (DATA Strobe) and AS* (ADDRESS Strobe –not shown) to signal that the data and address are valid
- The slave reads and acknowledges by asserting DTACK* (DATA Transfer Acknowledge)
- The master releases DS*, AS* and BSBSY*, the cycle is complete
- Note: there is no clock! The slave can respond whenever it wants. VME is an asynchronous bus



Data Transfer

Speed

- Theoretically ~ 16 MB/s can be achieved
  - *assuming the databus to be full 32-bit wide*
- Better performance by using block-transfer
  - *After an address cycle several (up to 256) data cycles are performed*
  - *Performance goes up to 40 MB/s*
- Since 1994 VME64 (64bits)
  - *use the address bus also for data*
  - *Twice the bandwidth of VME (up to 80 MB/s)*
- More recently VME64x
  - *New 160-pin 5-row connector*
  - *Twice the bandwidth of VME64 (up to 160 MB/s)*

Disadvantages of parallel buses

- A bus is shared between all devices (each new active device slows everybody down)
  - *Bus-width can only be increased up to a certain point (128 bit for PC-system bus)*
  - *Bus-frequency (number of elementary operations per second) can be increased, but decreases the physical bus-length*
- Nbr of devices and physical bus-length is limited (scalability!)
  - *For synchronous high-speed buses, physical length is correlated with the number of devices (e.g. PCI)*
  - *Typical buses have a lot of control, data and address lines (look at a SCSI or ATA cable)*
- Buses are typically useful for systems < 1 GB/s

Network based DAQ

- Network technology solves the scalability issues of buses
  - *In a network all devices are equal ("peers")*
  - *In a network all devices communicate directly with each other*
    - no arbitration necessary
    - bandwidth guaranteed
  - *data and control use the same path*
    - much fewer lines (e.g. in traditional Ethernet only two)
  - *At the signaling level buses tend to use parallel copper lines. Network technologies can be also optical, wire-less and are typically (differential) serial*

Trigger and Data Acquisition
at colliders

G. De Lentdecker

Université Libre de Bruxelles

NETWORK BASED DAQ

Network technologies

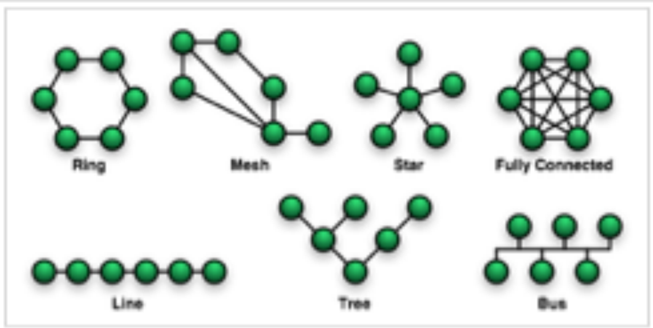
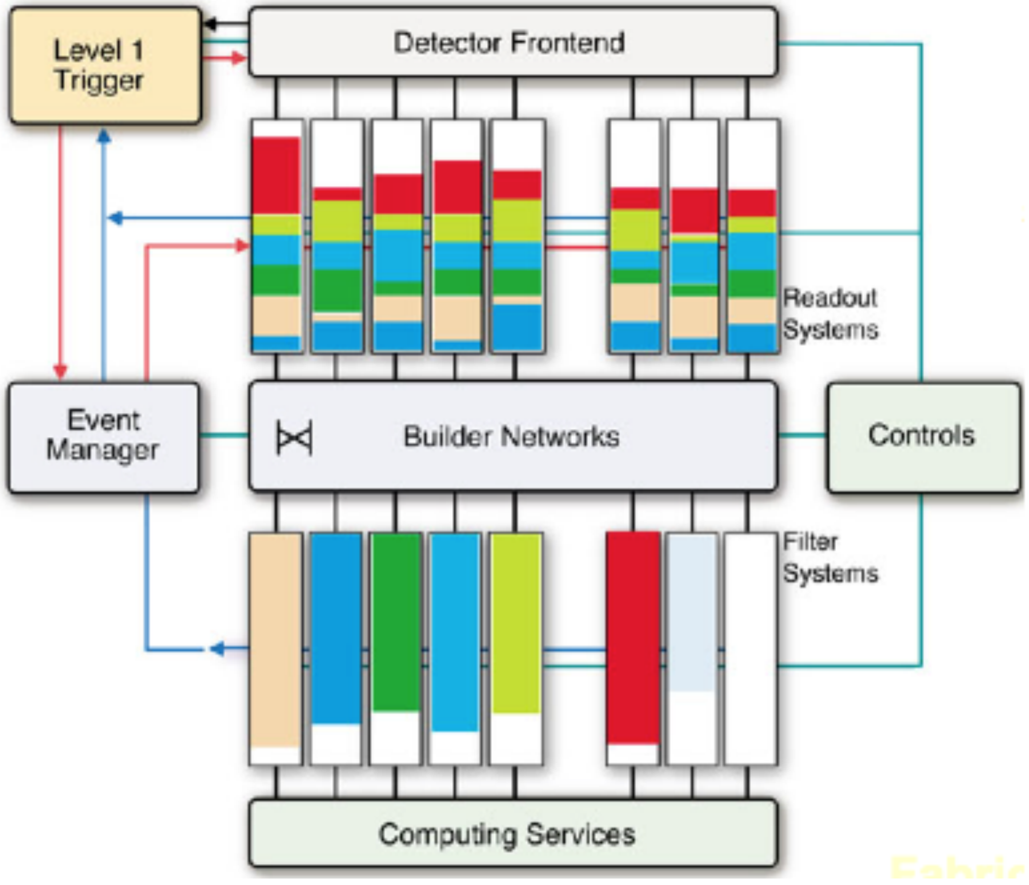


Diagram of different network topologies.

- Examples:
  - *The telephone network*
  - *Ethernet (IEEE 802.3)*
  - *ATM (the backbone for GSM cell-phones, small fixed sized packets)*
  - *Infiniband (point-to-point bidirectional serial links)*
  - *Myrinet (high-speed LAN designed by Myricom)*
  - *many, many more*
- Note: some of these have "bus"-features as well (Ethernet, Infiniband)
- Network technologies are sometimes functionally grouped
  - *Cluster interconnect (Myrinet, Infiniband) 15 m*
  - *Local area network (Ethernet), 100 m to 10 km*
  - *Wide area network (ATM, SONET) > 50 km*

Event Building

- Form full event data buffers from fragments in the readout
  => must interconnect data sources to destinations

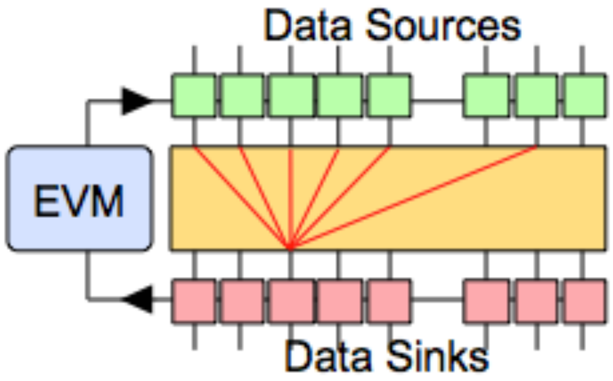


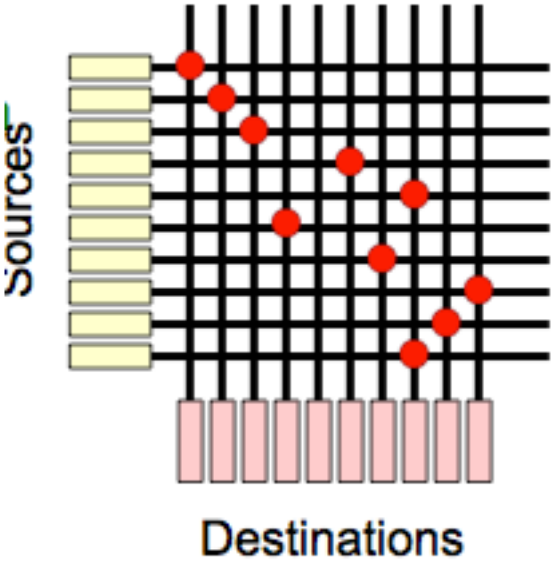
Data fragments are stored
In separated physical memory
systems

Full event data are stored
Into one physical memory system
Associated to one processor
unit

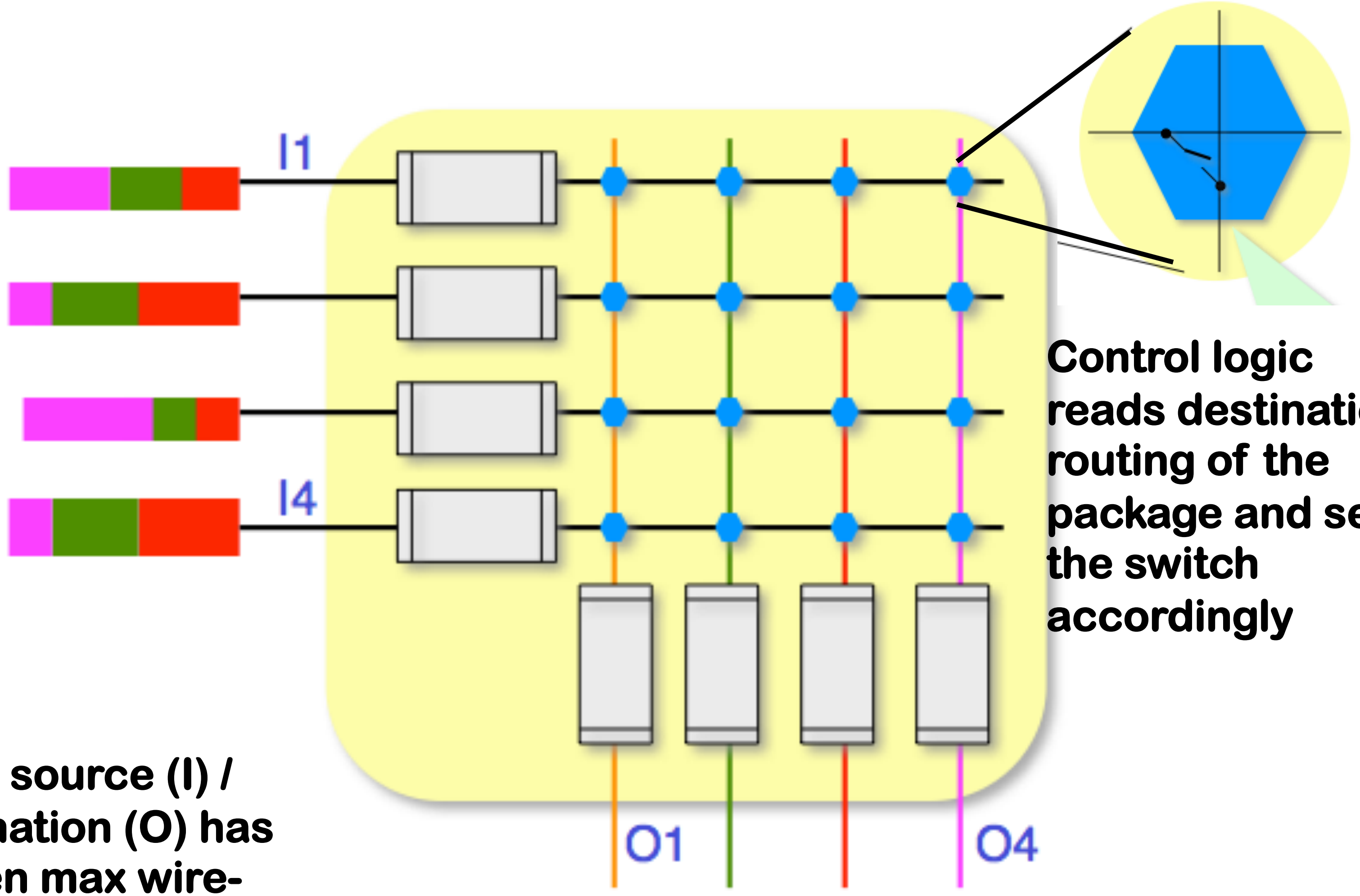
Event building with a switch

- 3 major issues
  - *Large number of ports*
  - *Possible bottleneck at the output*
  - *Link utilization*
- Crossbar implementation
  - *Simultaneous transfers between any arbitrary set of inputs and outputs*
  - *Can be both self-routing and arbiter-based (determine connectivity between S's and D's for each cycle);*
  - *the faster the fabric, the smaller the arbitration complexity*
  - *Does not solve Output Contention issue => Need Traffic Shaping*





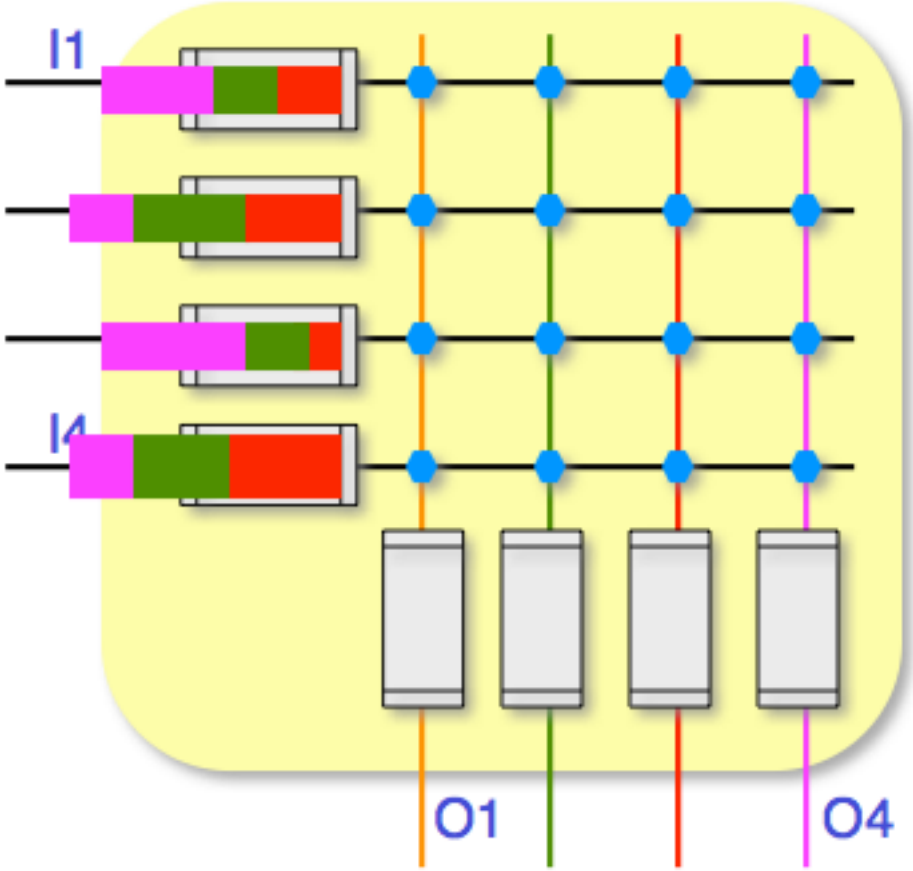
Example



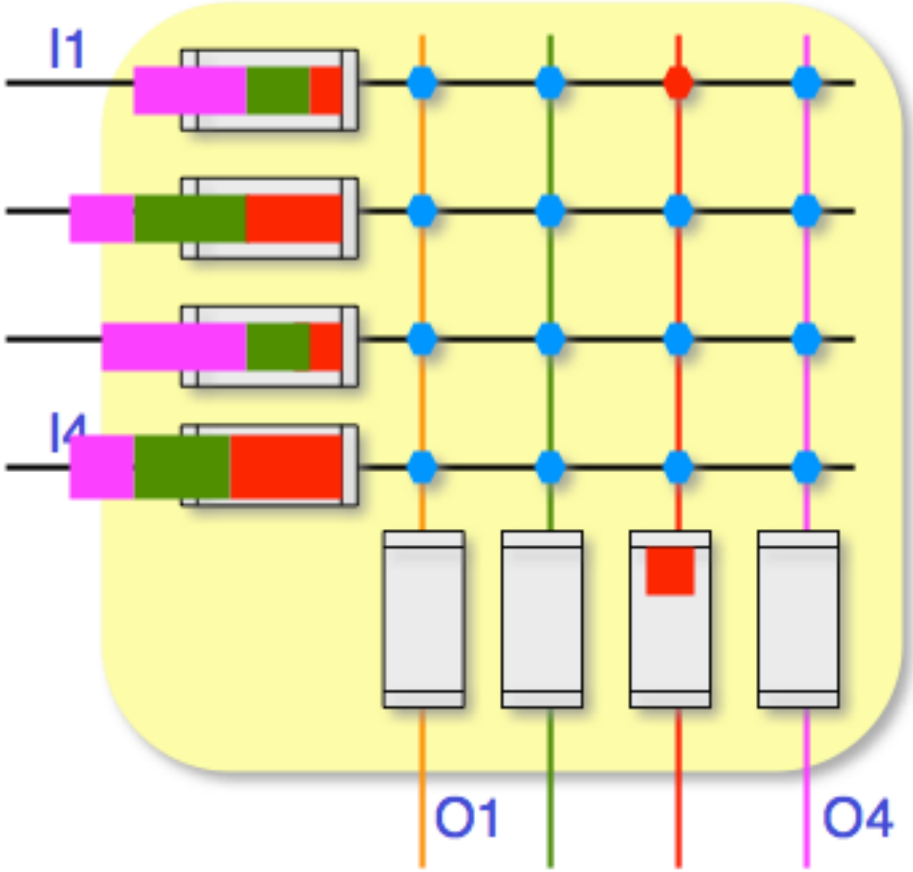
Control logic reads destination routing of the package and sets the switch accordingly

Every source (I) / destination (O) has a given max wire-speed (~2Gb/s)

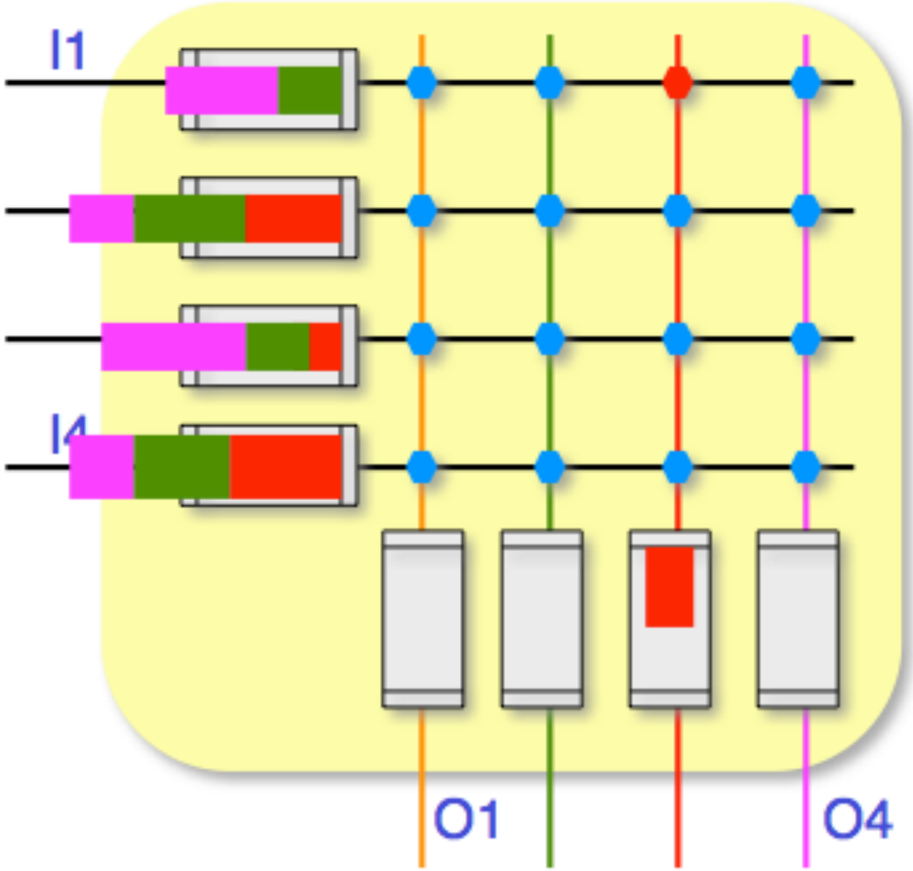
Example



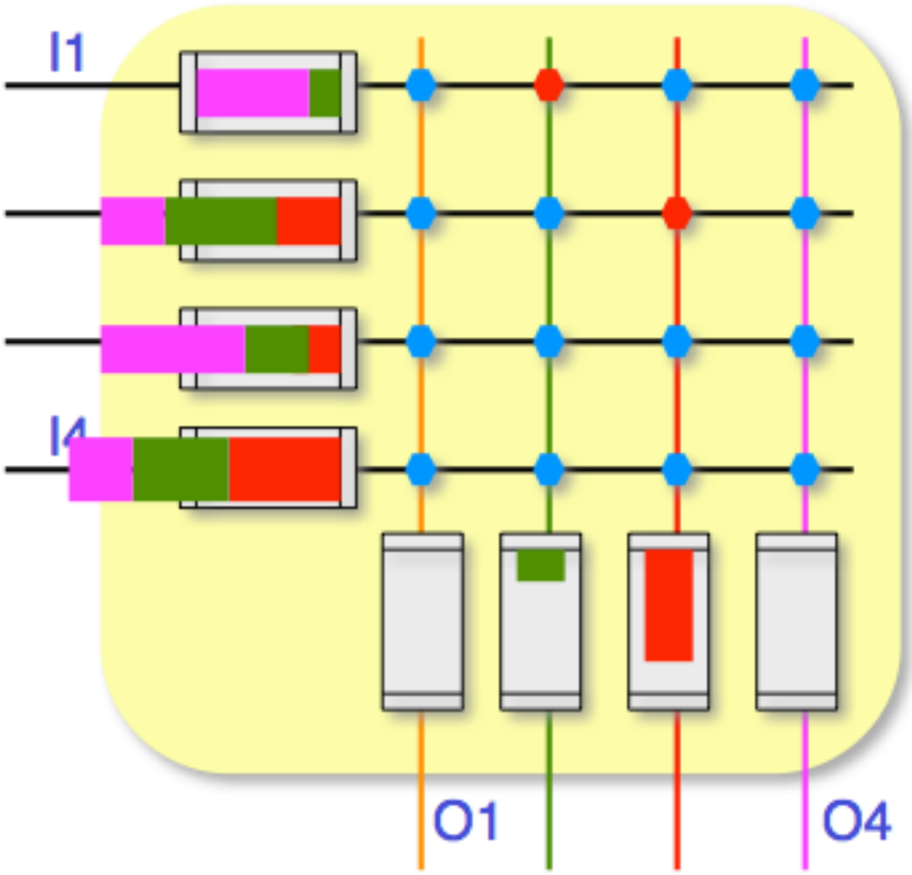
Example



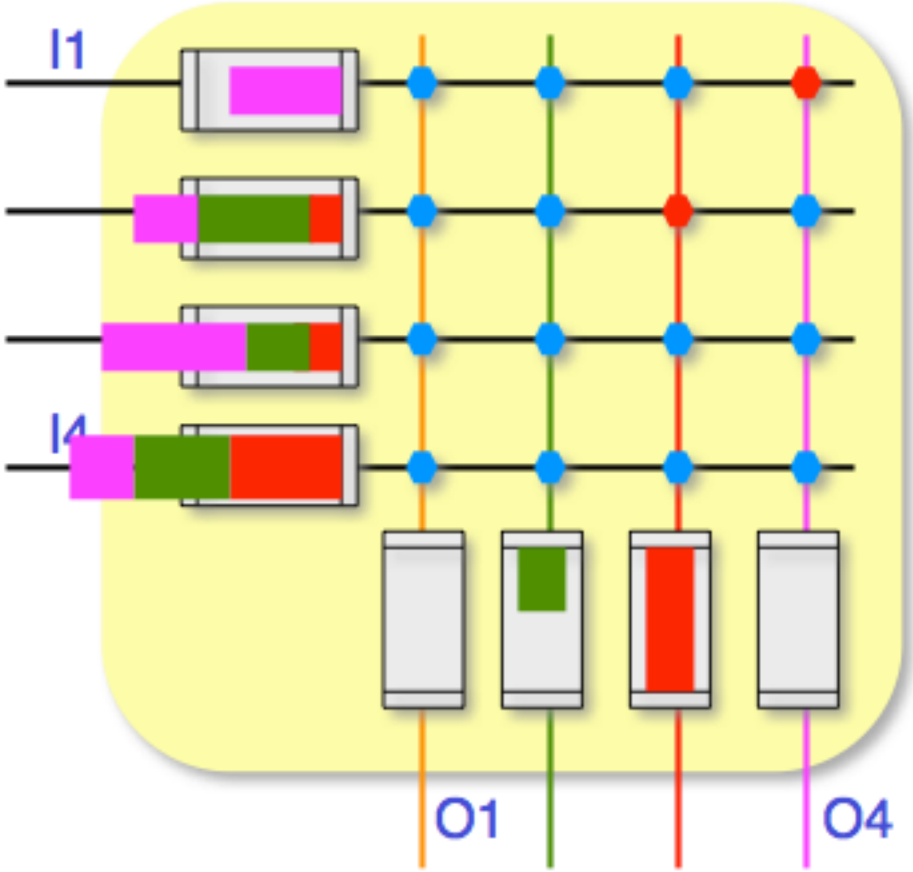
Example



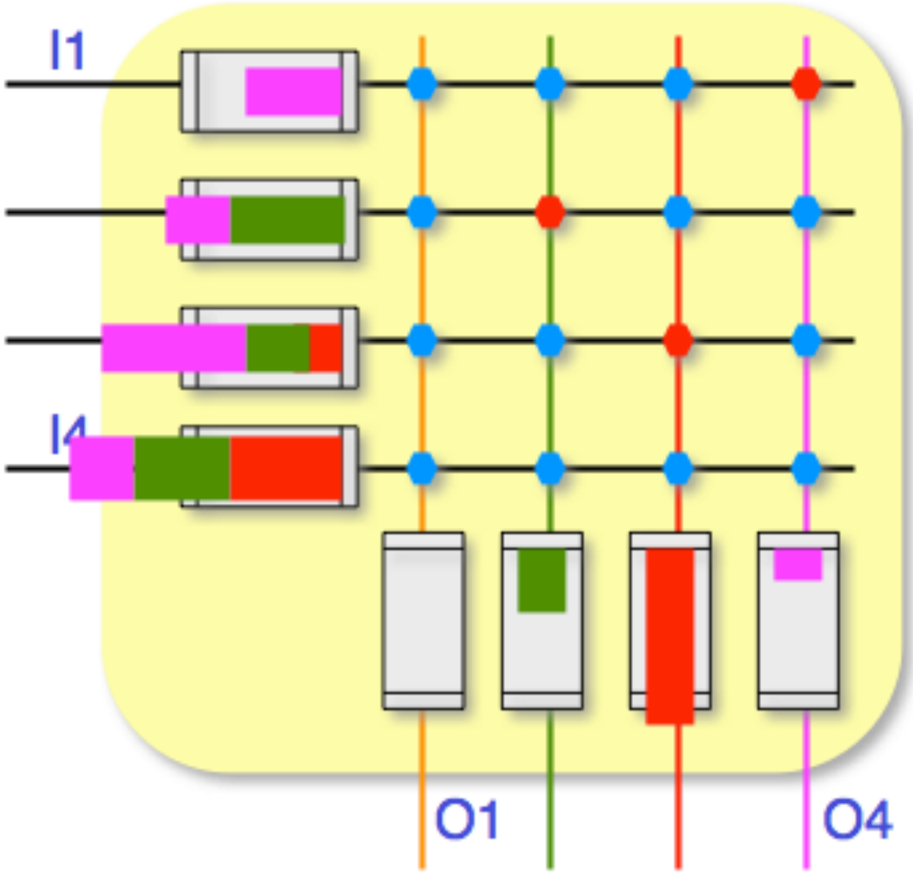
Example



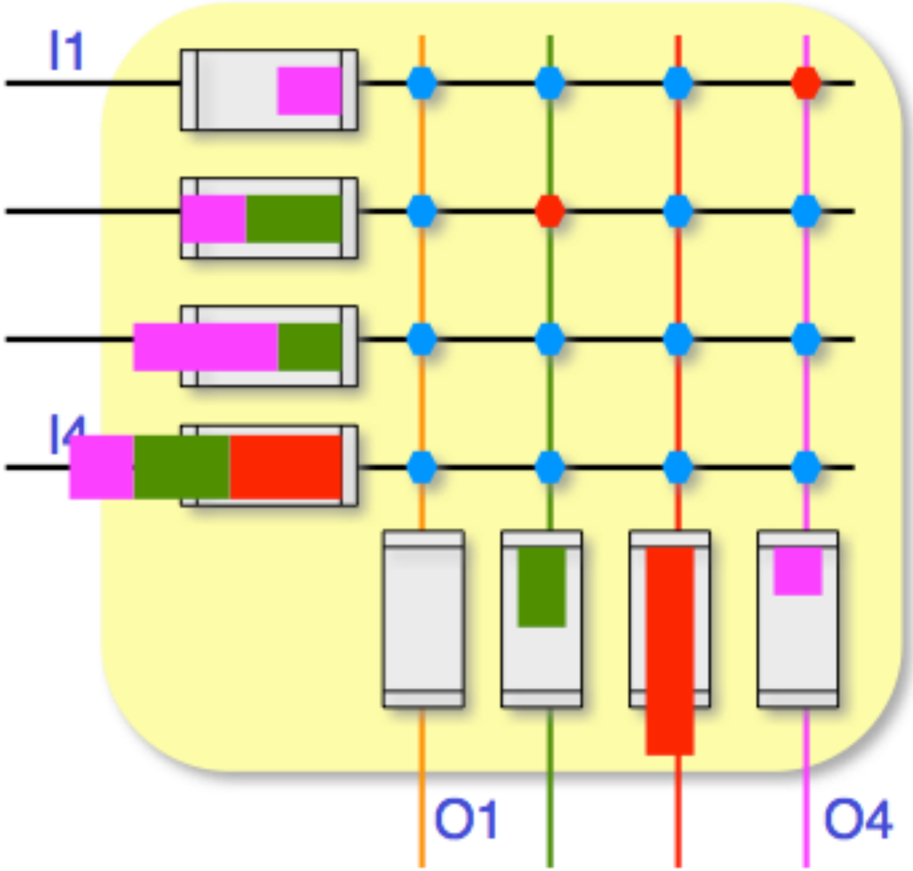
Example



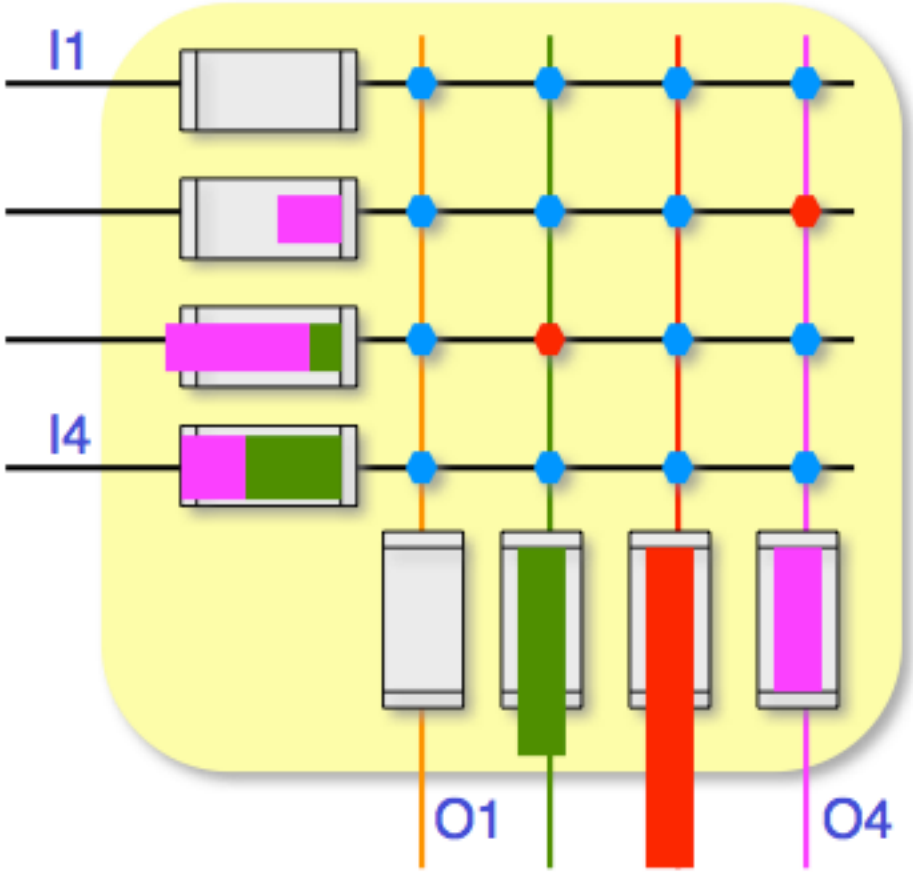
Example

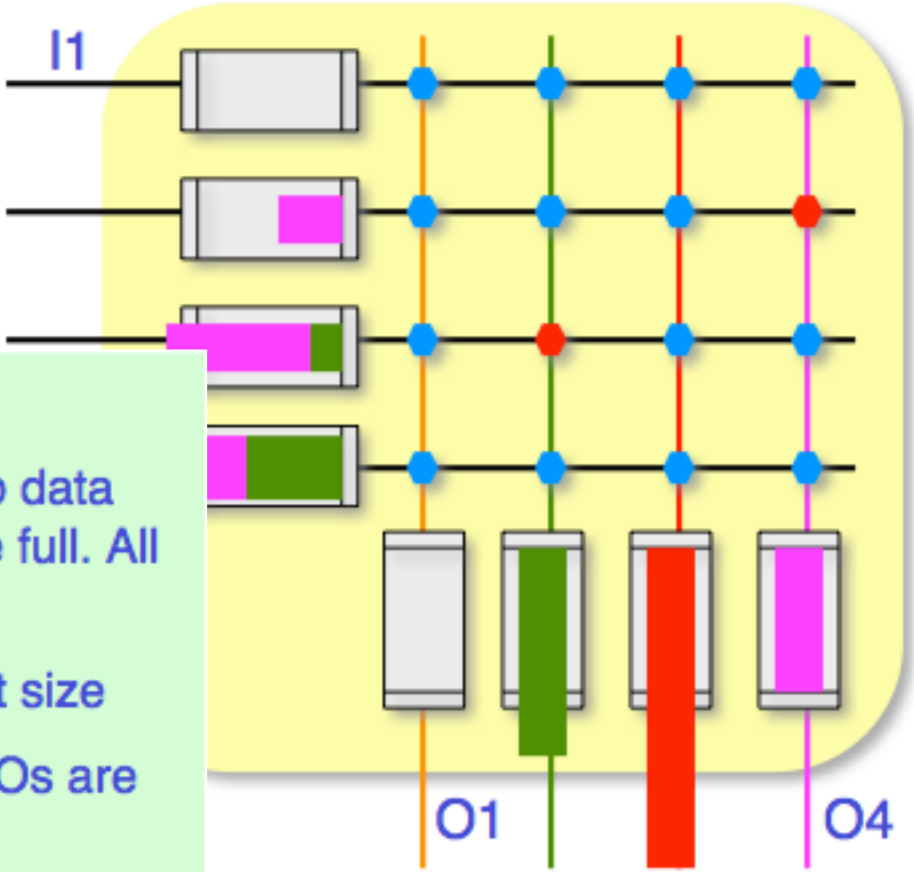


Example



Example





Problematic:

Input Fifios can absorb data fluctuations until they are full. All fine if:

Fifos capacity > event size

In practice: sizes of FIFOs are much smaller!

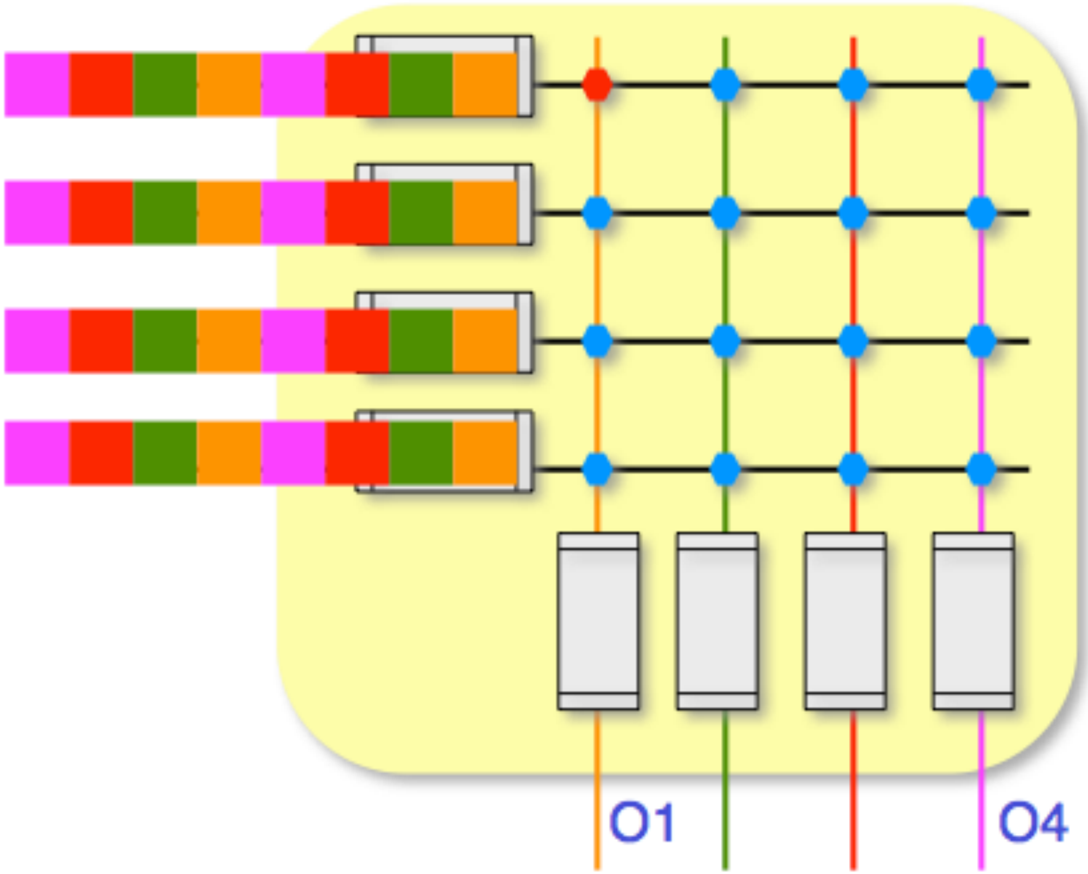
EVB traffic: switch will partially block

Avoid Congestion

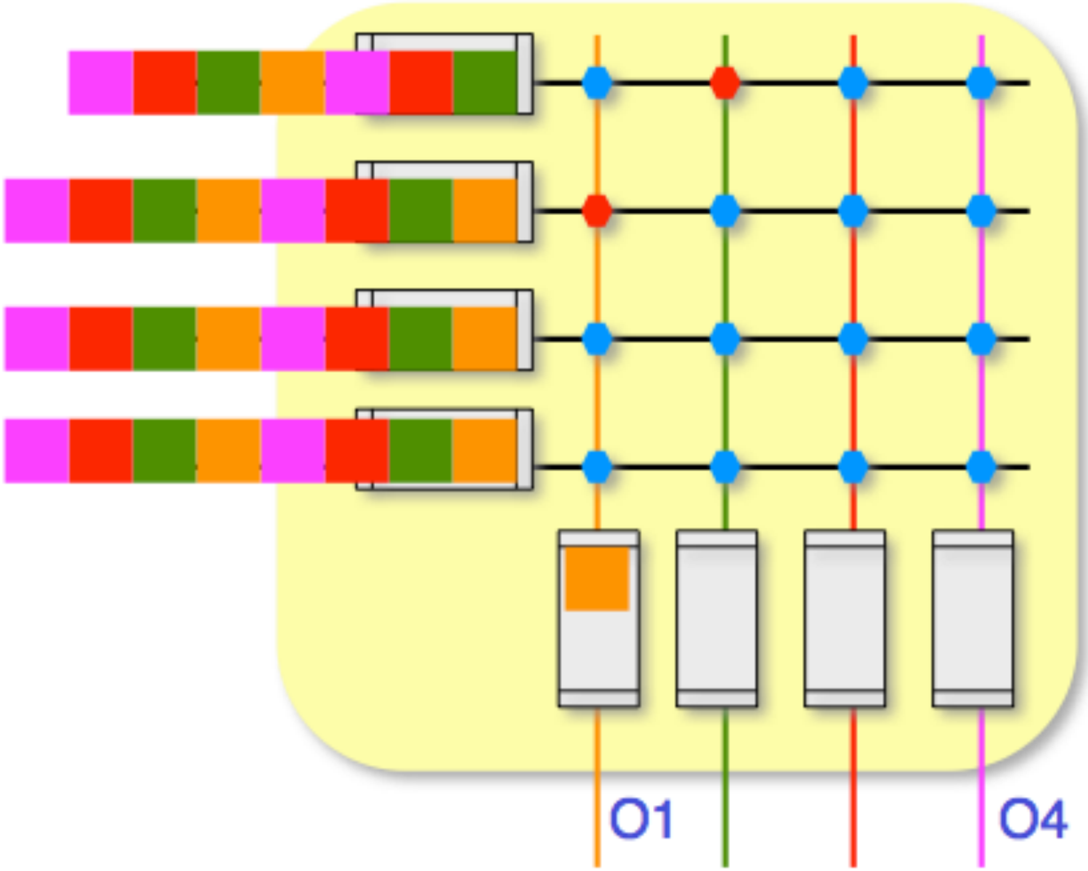
- => over size the system or do traffic shaping…

Traffic shaping

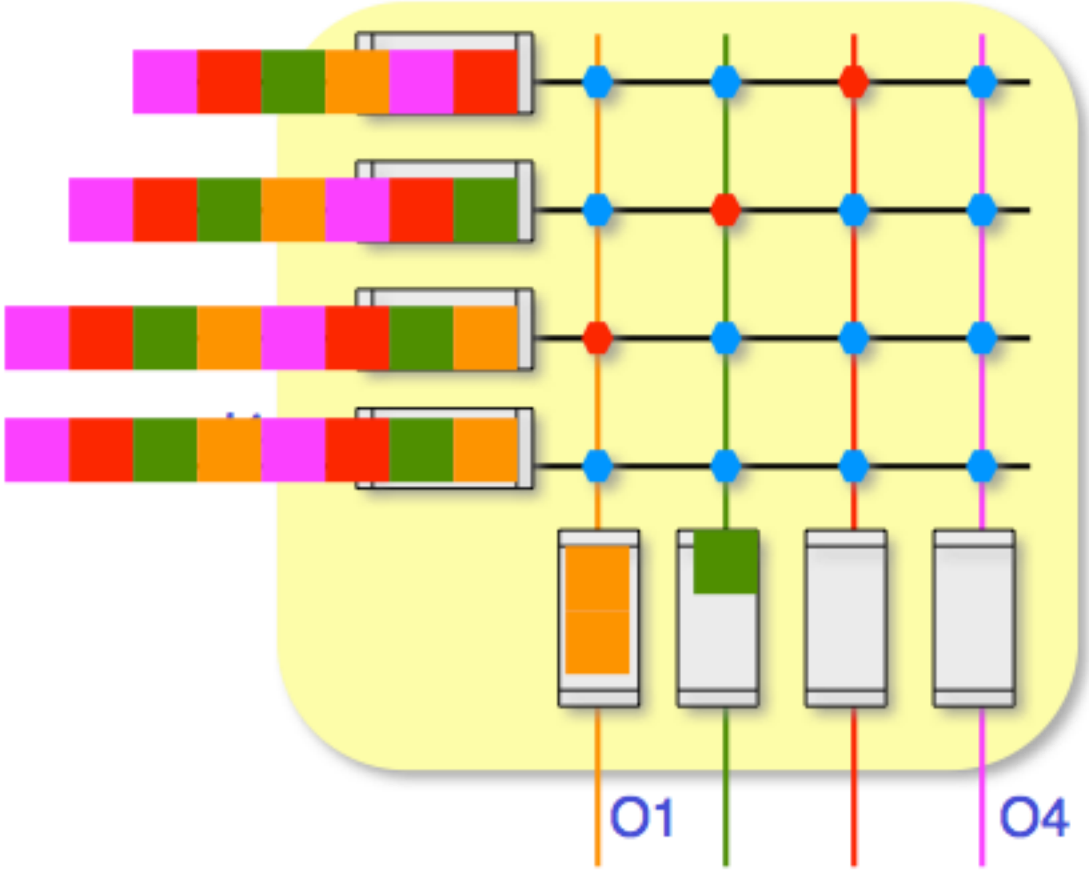
- Barrel shifter



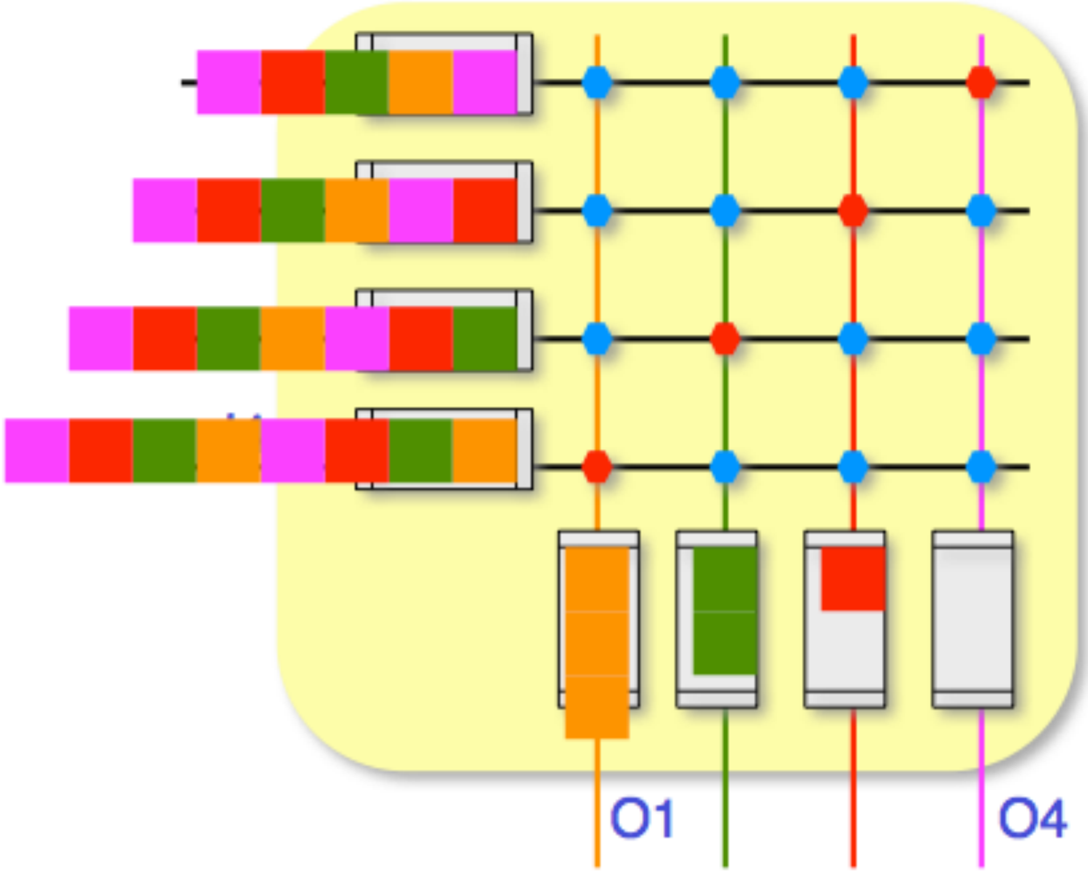
Traffic shaper



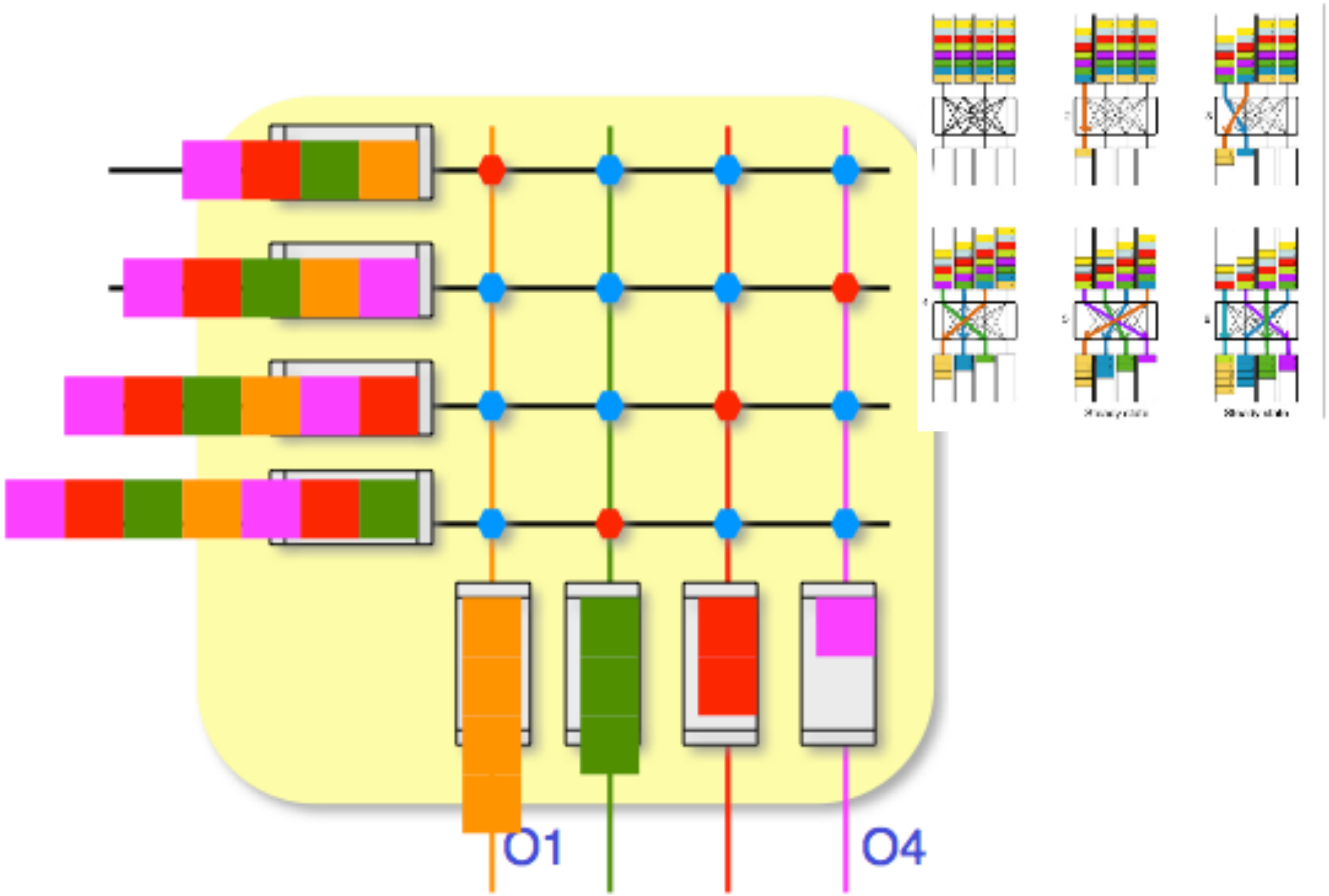
Traffic shaper



Traffic shaper

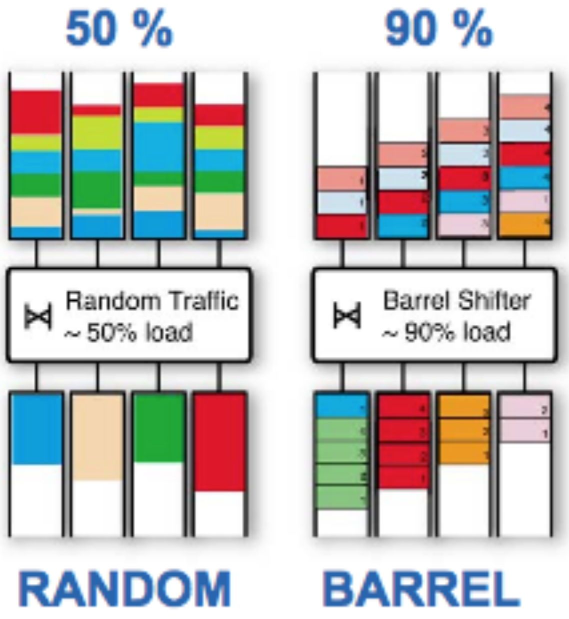


Traffic shaper

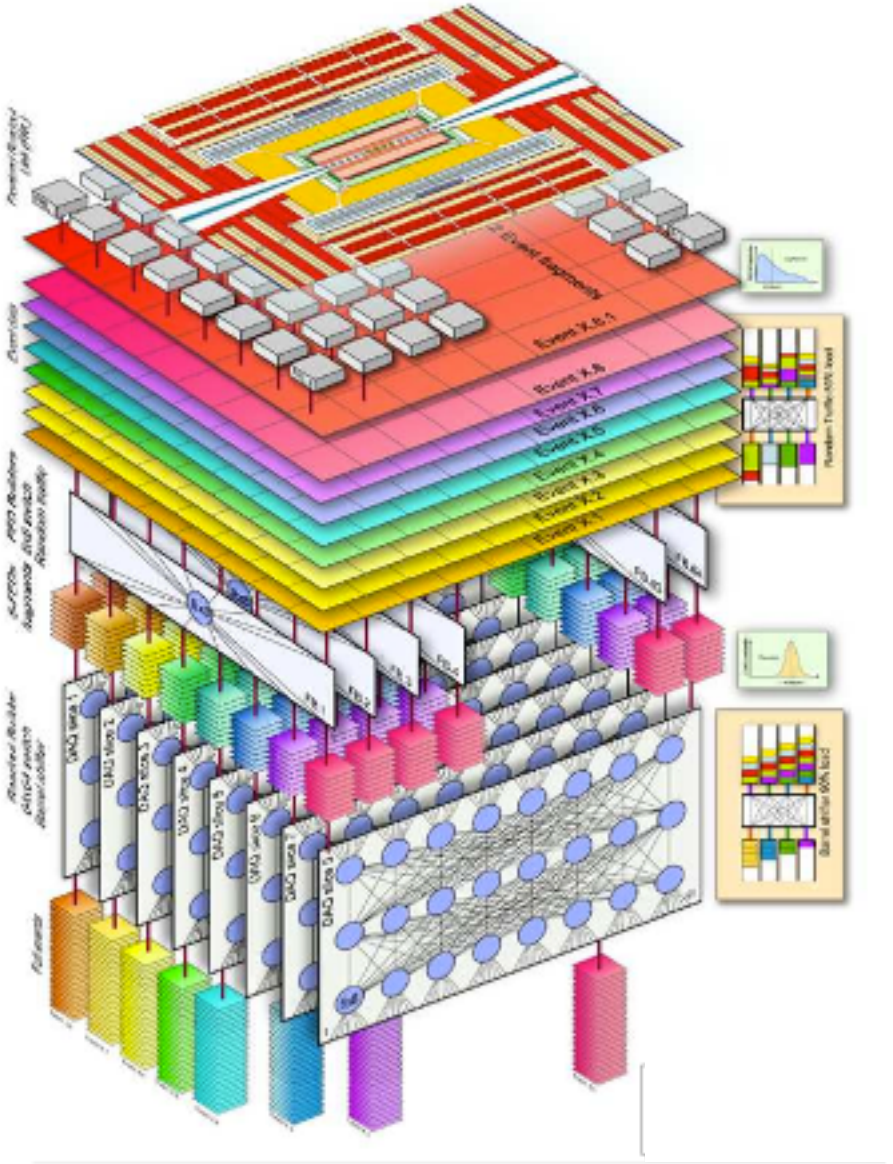


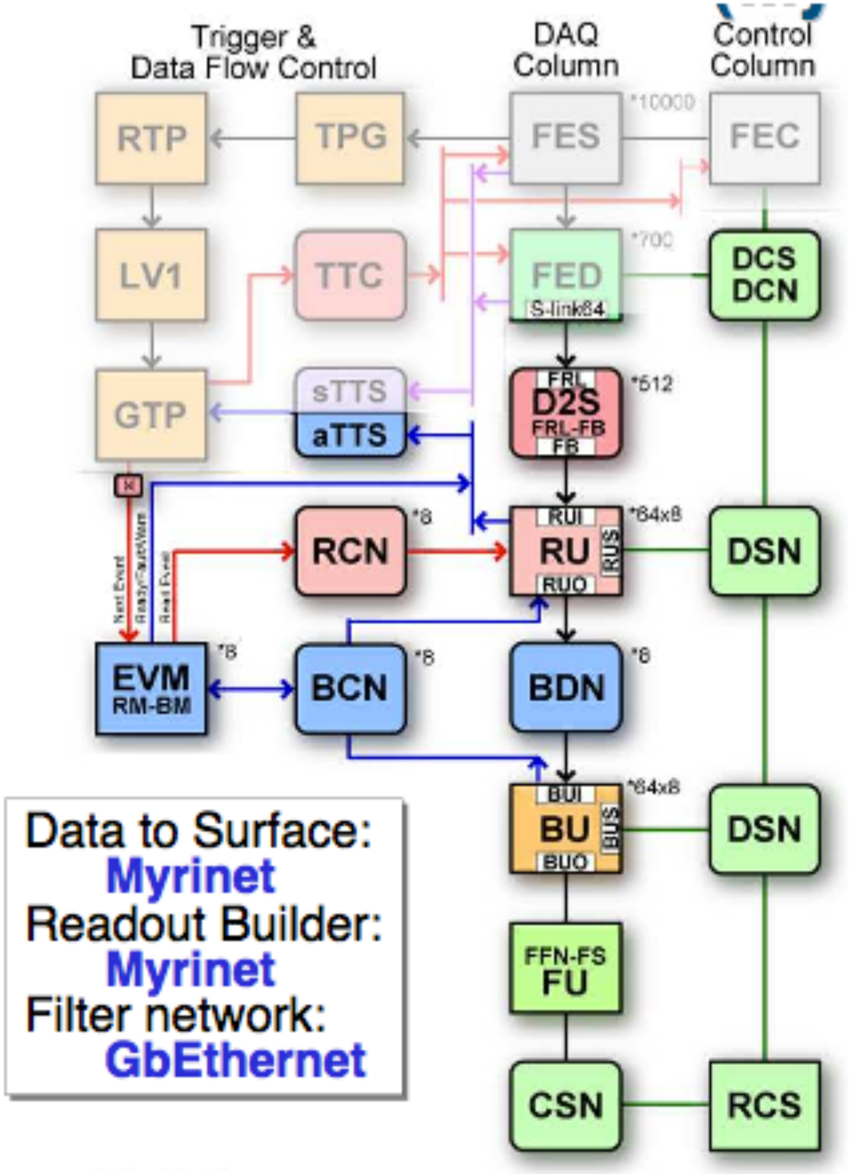
EVB Summary

- Random traffic: needs switch with factor 2 more bandwidth than throughput needed
- Barrel: can work with ~90% efficiency



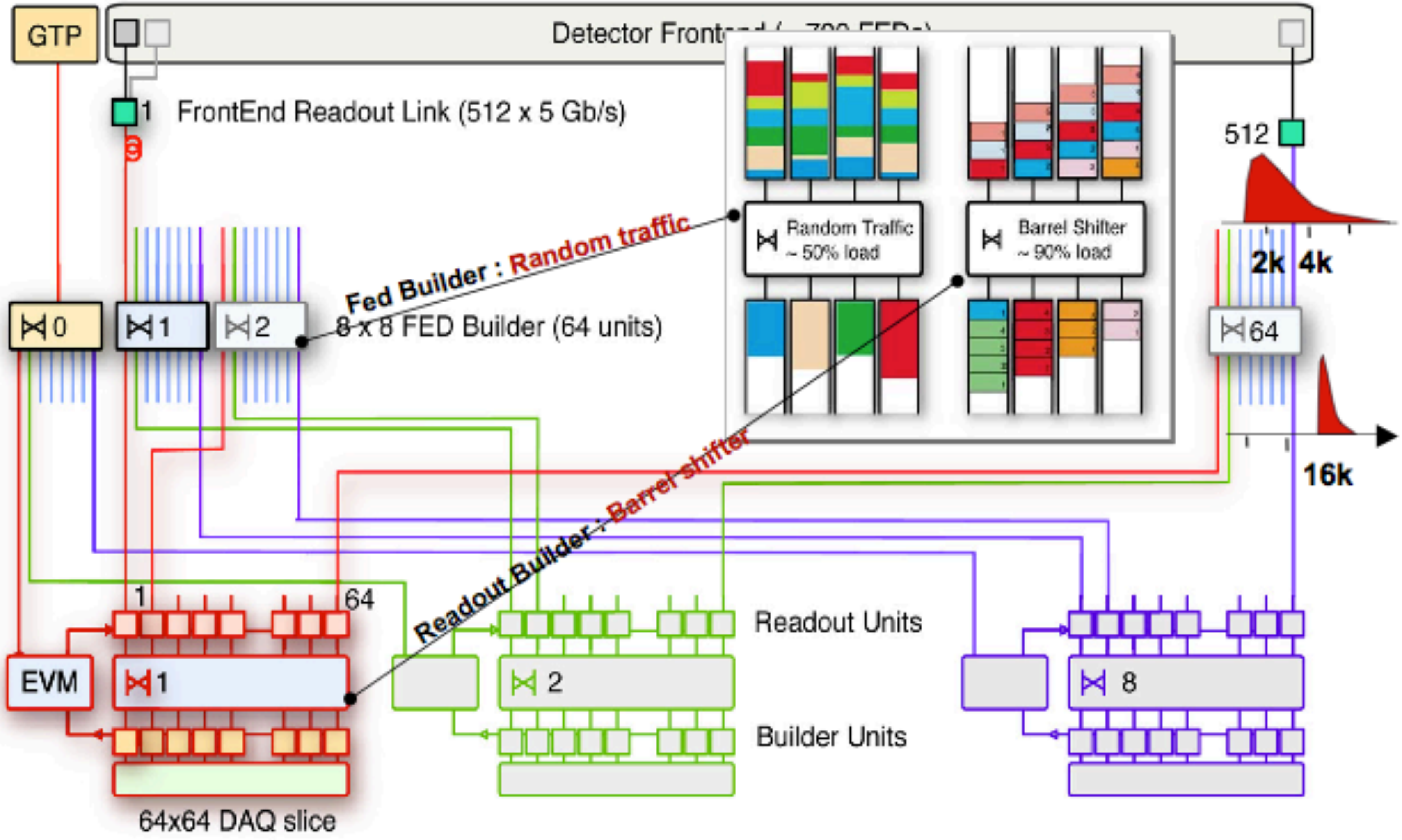
CMS EVB



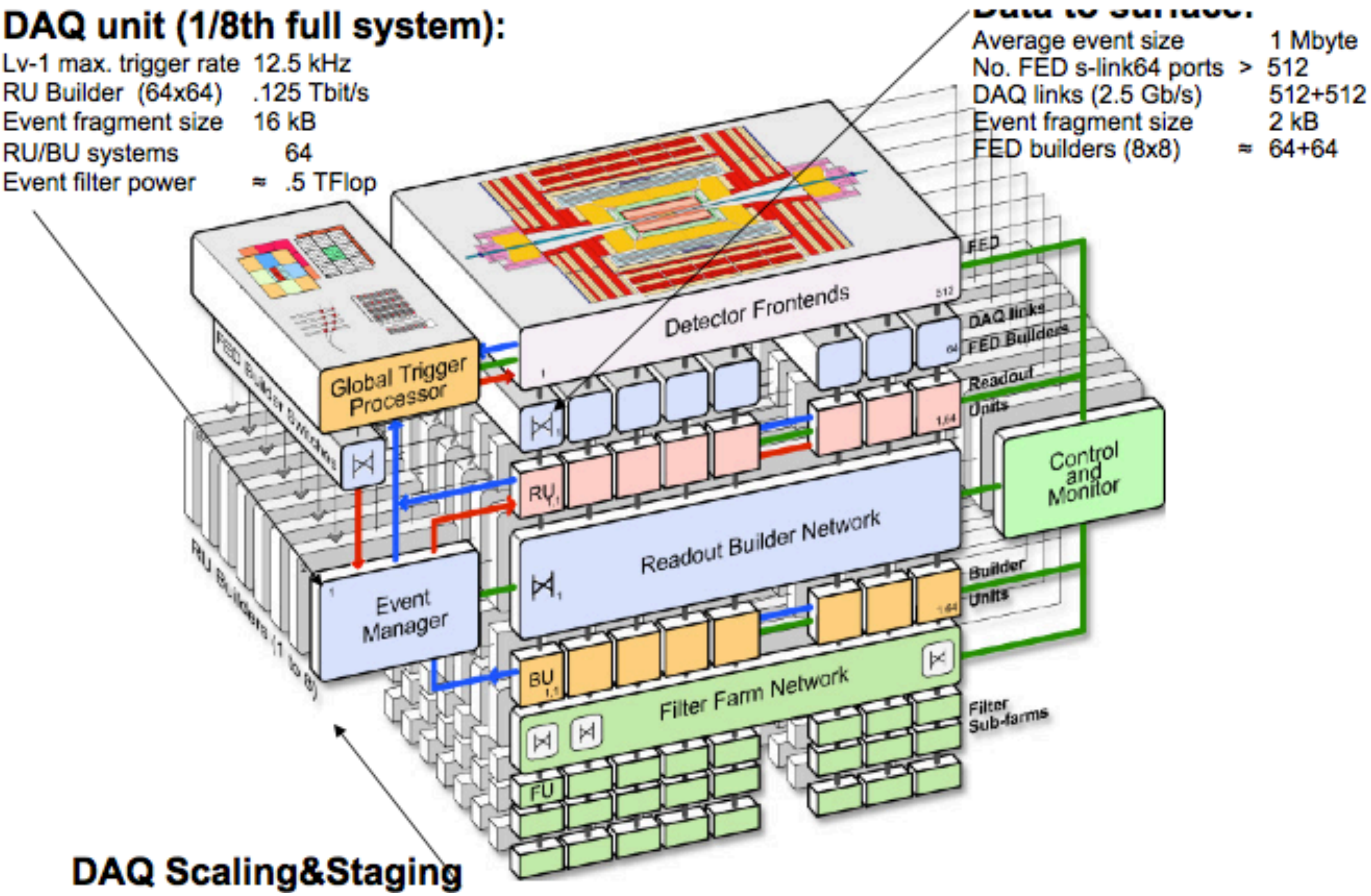


Data to Surface:
Myrinet
Readout Builder:
Myrinet
Filter network:
GbEthernet

Two stages CMS EVB



In 3D



DAQ unit (1/8th full system):

Lv-1 max. trigger rate	12.5 kHz
RU Builder (64x64)	.125 Tbit/s
Event fragment size	16 kB
RU/BU systems	64
Event filter power	≈ .5 TFlop

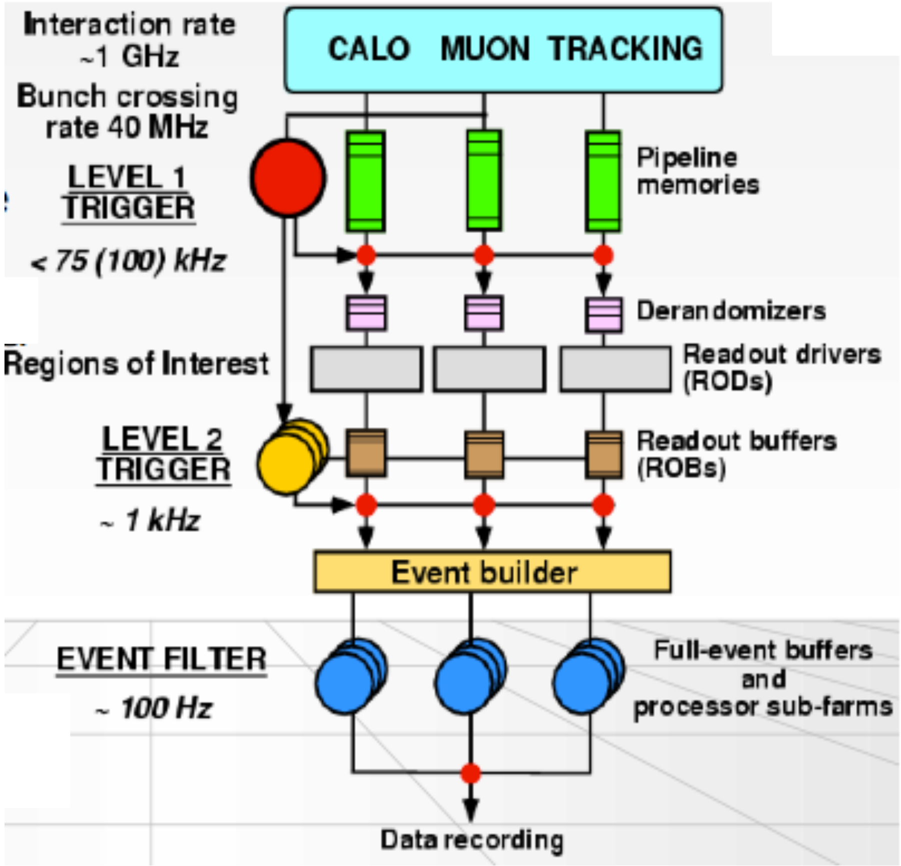
Data to surface:

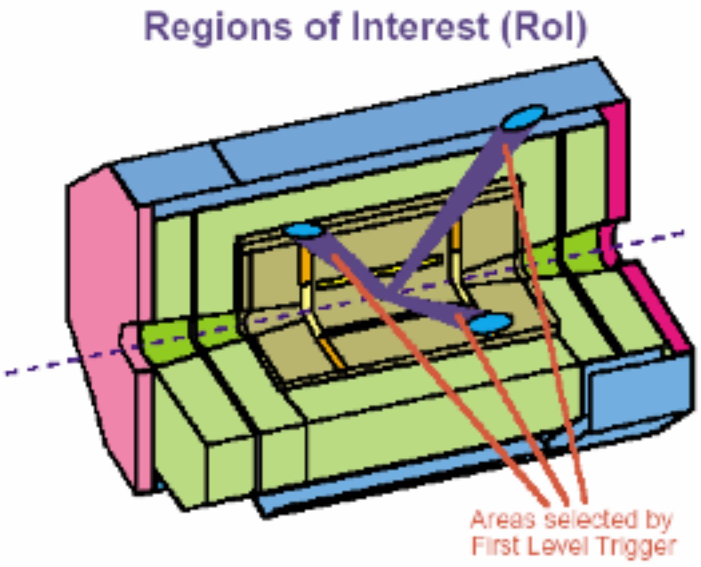
Average event size		1 Mbyte
No. FED s-link64 ports	>	512
DAQ links (2.5 Gb/s)		512+512
Event fragment size		2 kB
FED builders (8x8)	≈	64+64

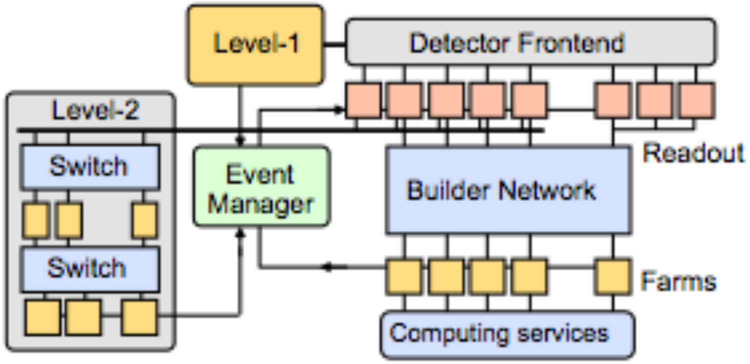
DAQ Scaling&Staging

ATLAS EVB

- After L1, L2 looks for Region of Interest (RoI)
- If L2 Accepts then all the event is sent to the next step




Regions of Interest (RoI)
Areas selected by First Level Trigger



The Filter Farm

- The final stage of the filtering process: almost an offline quality reconstruction & selection
  - *Very cost effective*
    - Linux is free
    - Interconnect : Ethernet (inexpensive & performant)
  - *Despite recent growth it is mature:*
    - The basic elements are mature: PC, Linux, Network

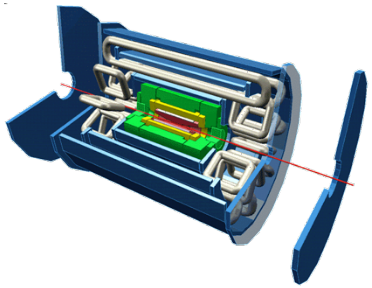
Algorithms & operation

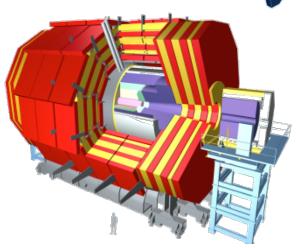
- Strategy/design guidelines
  - *Use offline software as much as possible*
    - Ease of maintenance, but also understanding of the detector
- Boundary conditions:
  - *Code runs in a single processor, which analyzes one event at a time*
  - *HLT (or Level-3) has access to full event data (full granularity and resolution)*
  - *Only limitations:*
    - CPU time
    - Output selection rate (~10^2 Hz)
    - Precision of calibration constants
- Main requirements:
  - *Satisfy physics program (see later): high efficiency*
  - *Selection must be inclusive (to discover the unpredicted as well)*
  - *Must not require precise knowledge of calibration/run conditions*
  - *Efficiency must be measurable from data alone*
  - *All algorithms/processors must be monitored closely*

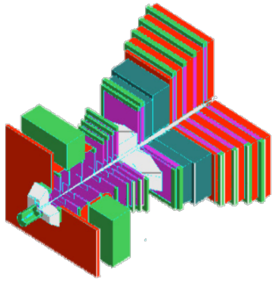
LHCb & ALICE DAQ

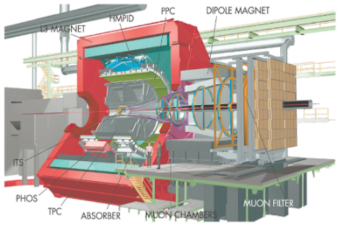
- LHCb
  - *Optimized to study B-hadron in p-p collisions*
  - *Level-0 output rate: 1MHz (L=$10^{32}cm^2s^{-1}$)*
  - *Event size = 40 kB*
- ALICE
  - *Optimized for heavy-ion collisions (Pb-Pb)*
    - Low collision rate (<10 kHz @ L=$10^{27}cm^2s^{-1}$)
    - But very high multiplicity (dN/deta ~ 8000)
      - Event size ~25 MB

LHC experiments DAQ



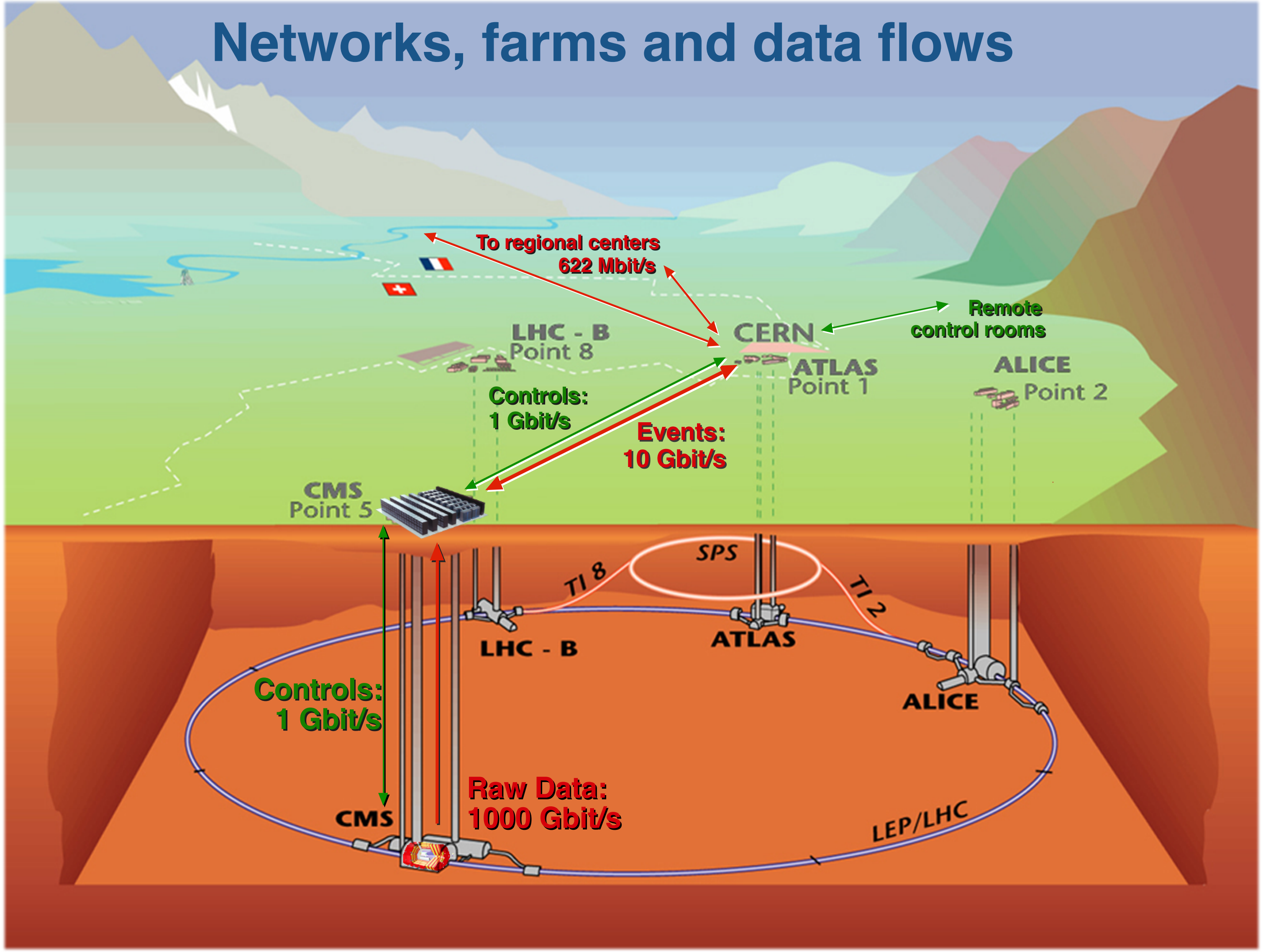






	Level-1 kHz	Event MByte	Storage MByte/s
ATLAS	100	1	100
CMS	100	1	100
LHCb	1000	0.04	80
ALICE	1(p-p) 0.2 (Pb-Pb)	25	1250

On to tape… and the GRID



Trigger and Data Acquisition

at colliders

G. De Lentdecker

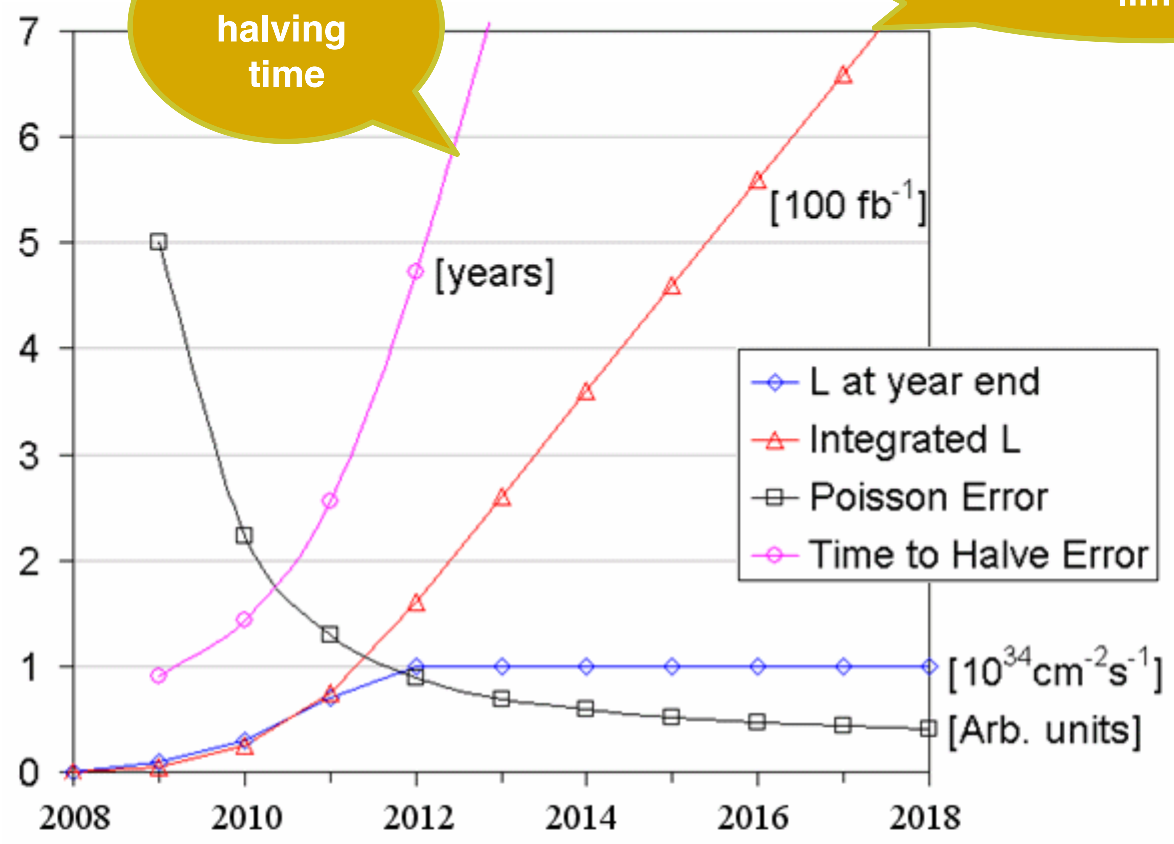
Université Libre de Bruxelles

FUTURE COLLIDERS DAQ

Introduction

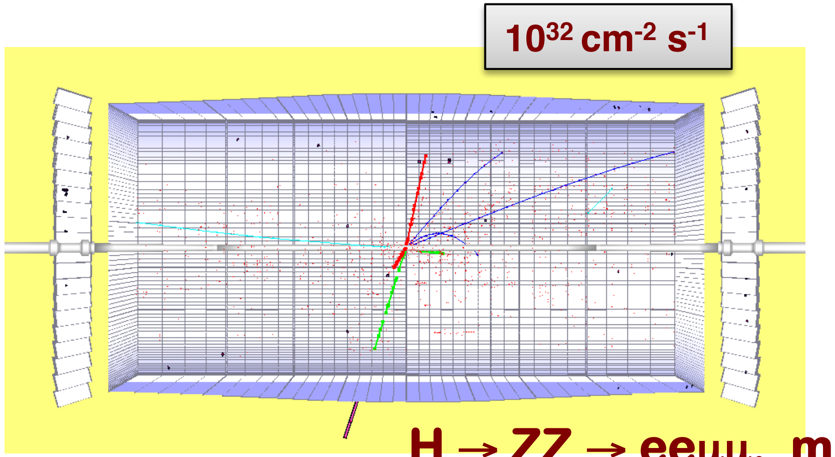
- LHC upgrade motivations
  - *1) Hardware will age*
  - *2) statistical error of measurements will reach a plateau*

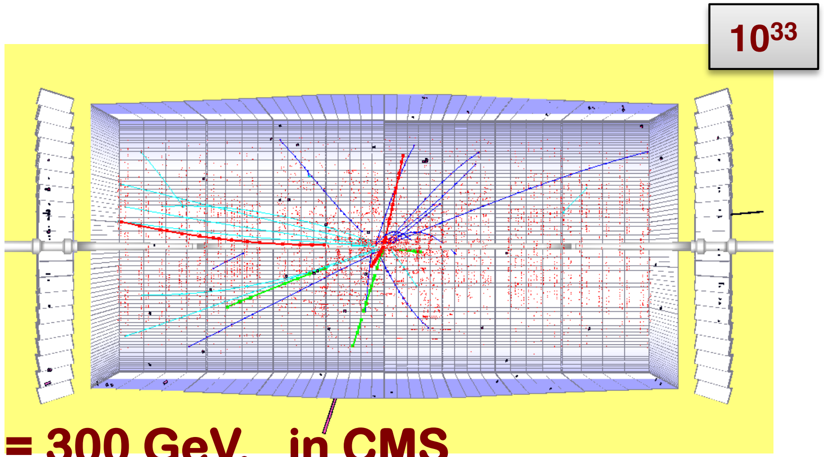
Need a
Major upgrade
~ 2018



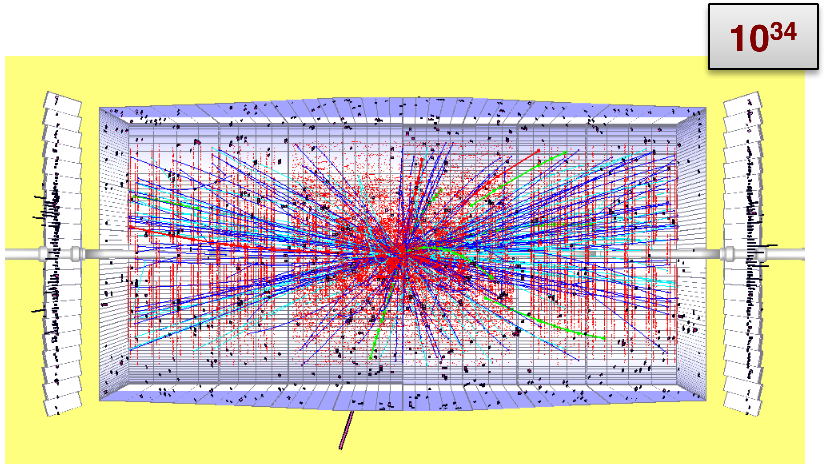
Towards a luminosity of ~$10^{35}cm^{-2}s^{-1}$

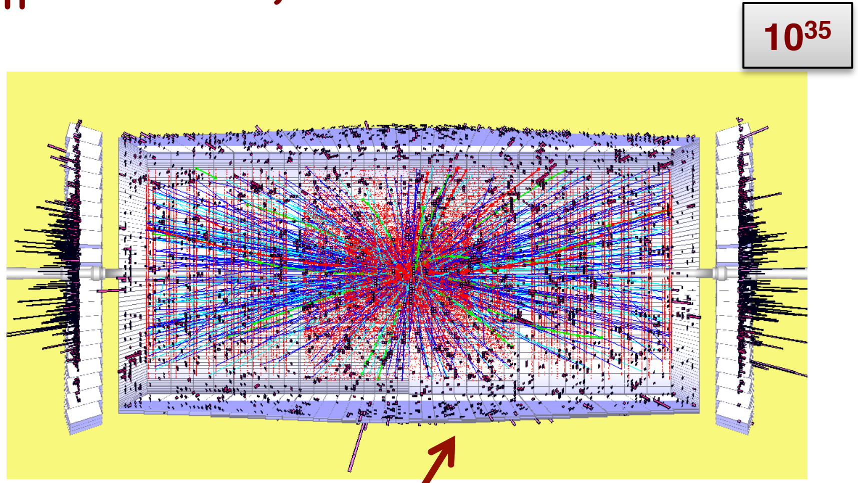
SLHC: Crossing every 25ns => 220 minimum bias events (10 x LHC)





H → ZZ → eeμμ, m_H = 300 GeV, in CMS





Generated tracks, $p_t > 1$ GeV/c cut, i.e. all soft tracks removed!

I. Osborne

Can we trigger on that ?

The CMS Tracker

- Tracker has to have at least the same performance **AND**:
  - *Has to cope with 10x track density per BX*
    - Increase the granularity (by reducing the strip length)
  - *Has to cope with 10x irradiation levels*
    - Mostly a sensor issue
  - *Reduce the material budget*
  - *Maintain CMS trigger performance*
    - 10x the luminosity while keeping the LHC L1 output rate (100kHz)

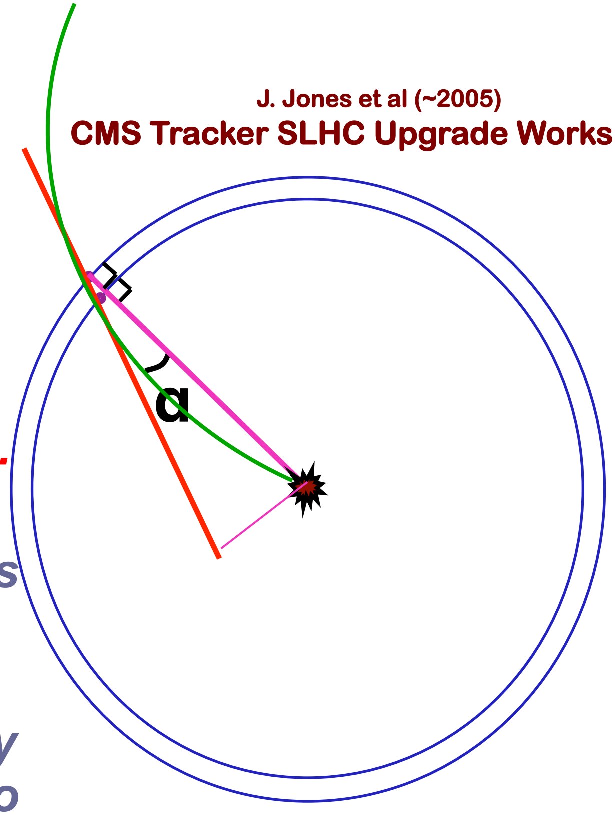
- The design of the new tracker will be defined from the track trigger algorithms
- Indeed the tracker will have to be part of Level1 trigger…

Tracker @ L1 => Track Trigger

- *Consider pixel detector which consists of two layers of silicon placed ~ mm apart in radius.*
- *On detector electronics form hit coincidences between pixels hits in the two layers.*
- ***High P_t tracks will point to the origin***
- ***If only 'high-angle-coincidence' data are to be transmitted out of the detector this would amount to a Pt cut.***
- ***Transmitting only hits that correspond to tracks with momentum above 5 GeV would result to an enormous data reduction (10-100).***
- ***Advances in FPGA and data transmission technologies** make the off-detector electronics a less challenging issue.*
- ***There are considerable issues with power consumption and inactive material particularly since one needs to place this detector close to the interaction point.***

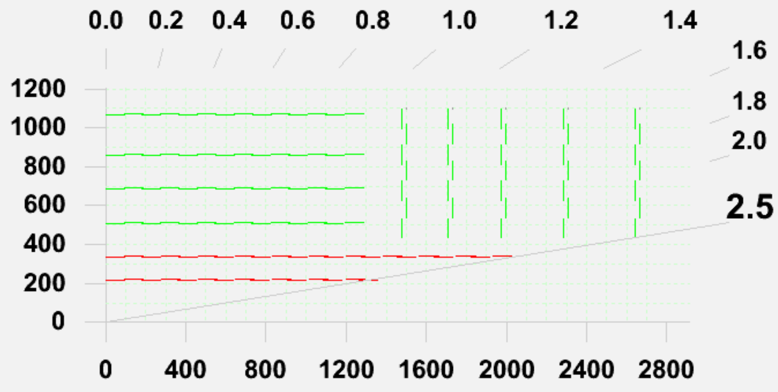
J. Jones et al (~2005)

CMS Tracker SLHC Upgrade Workshops



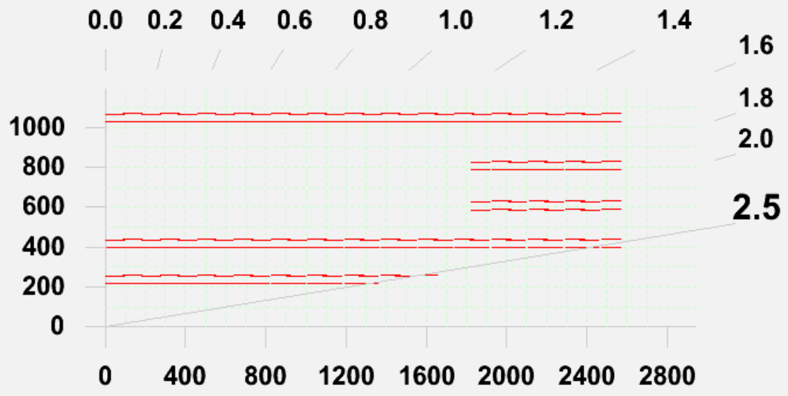
New Tracker Geometry ?

Strawman A:
"Conventional Tracker"
+ minimal Track Trigger layers



Present data packing : 120,000 Gbps
Improved data packing: 1,400 Gbps

Strawman B:
A fully integrated Tracking Trigger Tracker

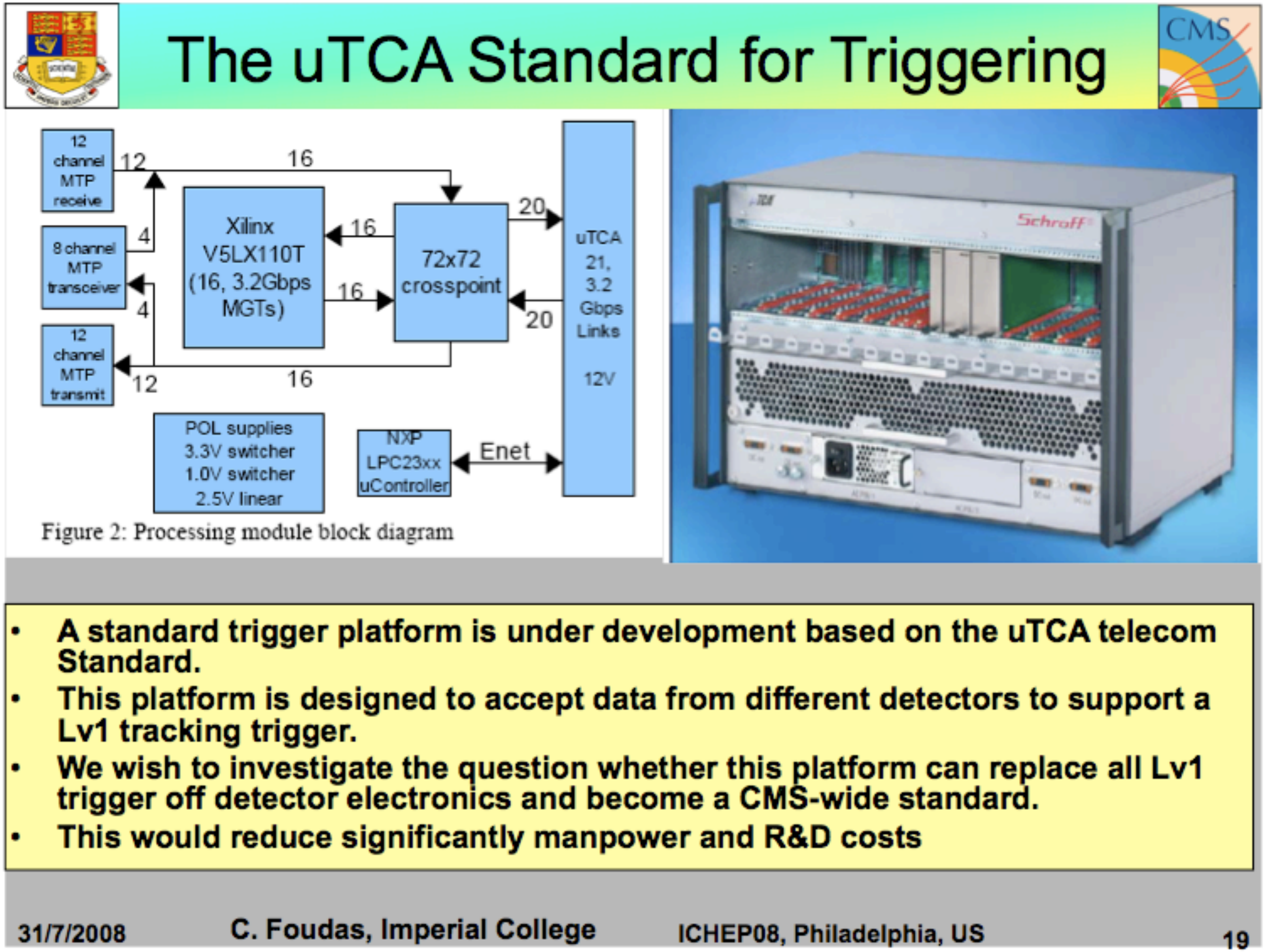


Present data packing : 1,000,000 Gbps
Improved data packing: 2,600 Gbps

CMS Event Builder bandwidth : 1000 Gbps

The future DAQ system

- ATCA: backplane does not use a bus, but supports GbE.



The uTCA Standard for Triggering

CMS
12 channel MTP receive
8 channel MTP transceiver
12 channel MTP transmit
12
16
4
4
12
16
Xilinx V5LX110T (16, 3.2Gbps MGTs)
16
16
72x72 crosspoint
20
20
uTCA 21, 3.2 Gbps Links
12V
POL supplies
3.3V switcher
1.0V switcher
2.5V linear
NXP LPC23xx uController
Enet
Schroff

Figure 2: Processing module block diagram

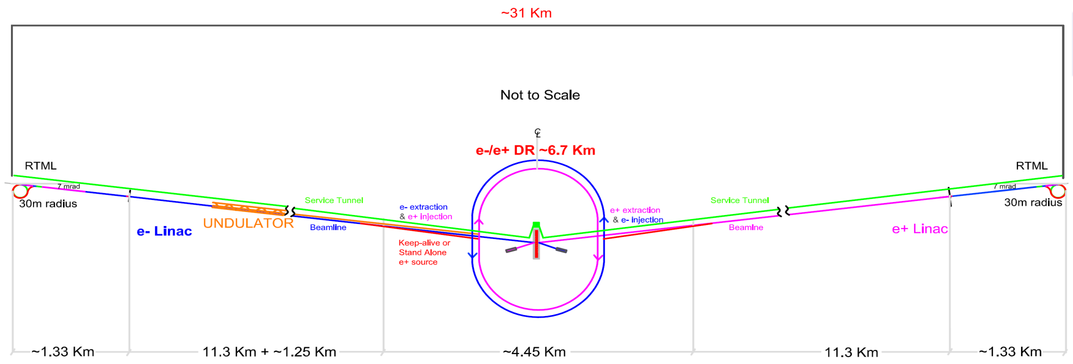
- A standard trigger platform is under development based on the uTCA telecom Standard.
- This platform is designed to accept data from different detectors to support a Lv1 tracking trigger.
- We wish to investigate the question whether this platform can replace all Lv1 trigger off detector electronics and become a CMS-wide standard.
- This would reduce significantly manpower and R&D costs

31/7/2008 C. Foudas, Imperial College ICHEP08, Philadelphia, US 19

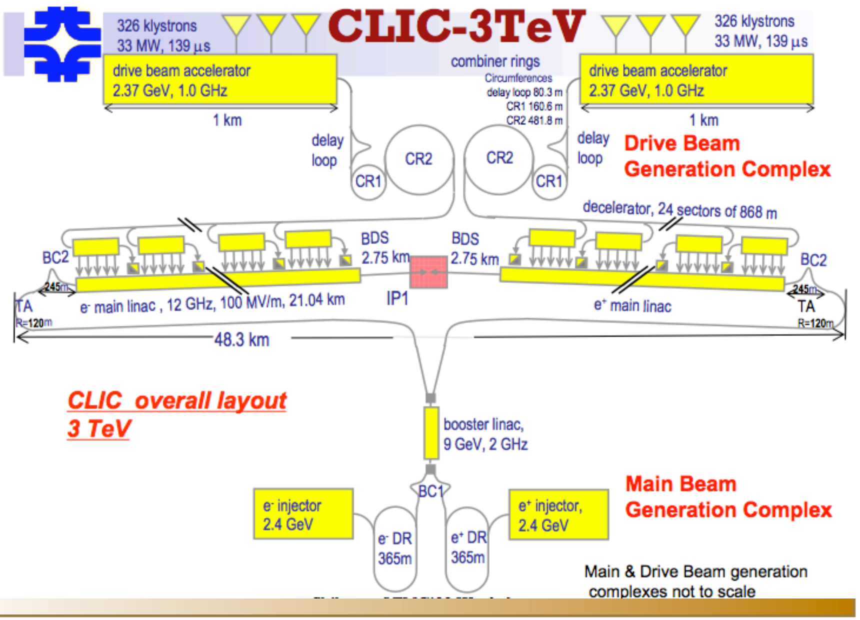
ILC / CLIC

- International Linear Collider (ILC) :
  - *E_{CM}=0.5 - 1.0 TeV*
  - *L=2x10^{34} cm^{-2} s^{-1}*
  - *Cold technology (RF = 1.3 GHz, Gradient = 35 MV/m)*
  - *Collision rate : 3 MHz (3000 bunches/1ms train at 5 Hz)*
  - *Expect EDR phase ends 2012, construction starts after 2015*

- Compact Linear Collider (CLIC):
  - *E_{CM}=1.0 - 3.0 TeV*
  - *L~10^{35} cm^{-2} s^{-1}*
  - *2-Beam accelerator scheme*
    - Normal (warm) conducting accelerating cavities
      - –High accelerating gradients (100MV/m) and freq. (12 GHz)
    - 0,7 ns bunch spacing !
  - *Expect construction after '17*



Schematic Layout of the 500 GeV Machine



CLIC overall layout
3 TeV

ILC detector (ILD)

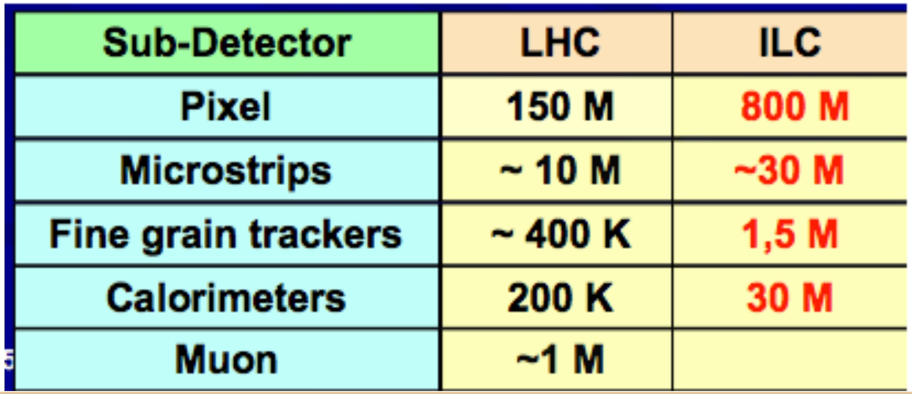
- Concept based on particle flow

  -> large tracker and high granularity calorimeters inside high (3.5T) magnetic field:

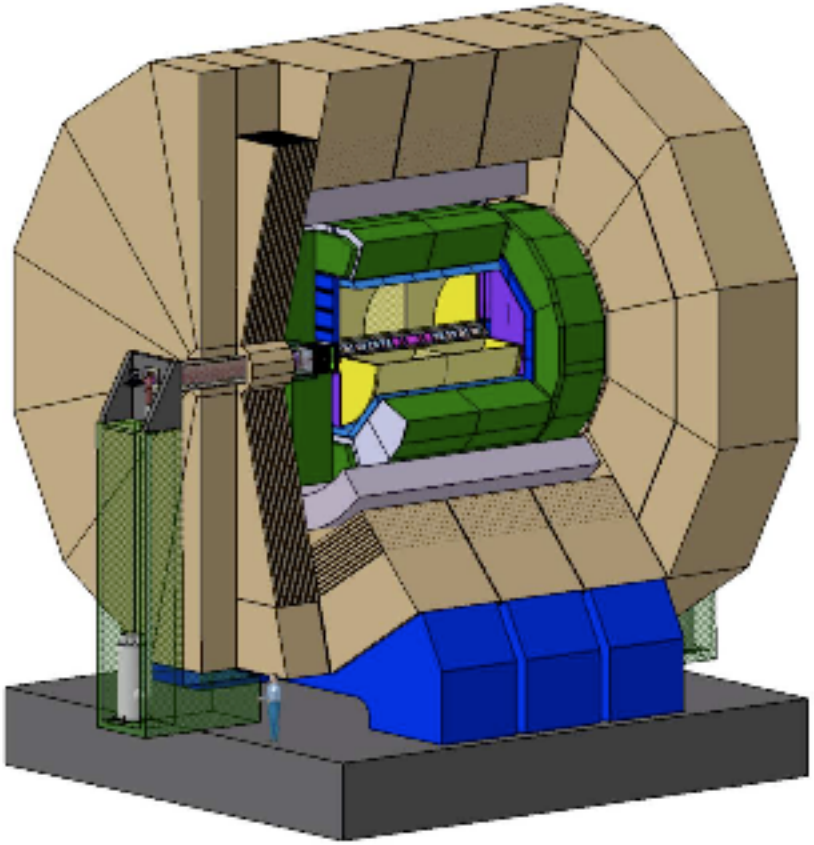
  - *Si vertex detector*
  - *Large TPC (L=4.3m, Ø=3.6m)*
  - *Si envelope (in- and outside)*
    - Precise calorimeter impact
  - *Calorimeter*

$$\sigma_{E_{\mathrm{em}}}/E_{\mathrm{em}} \simeq 15\%/\sqrt{E(\mathrm{GeV})} \oplus 1\%$$

$$\sigma_{E_{\mathrm{jet}}}/E_{\mathrm{jet}} \simeq 30\%/\sqrt{E(\mathrm{GeV})}$$

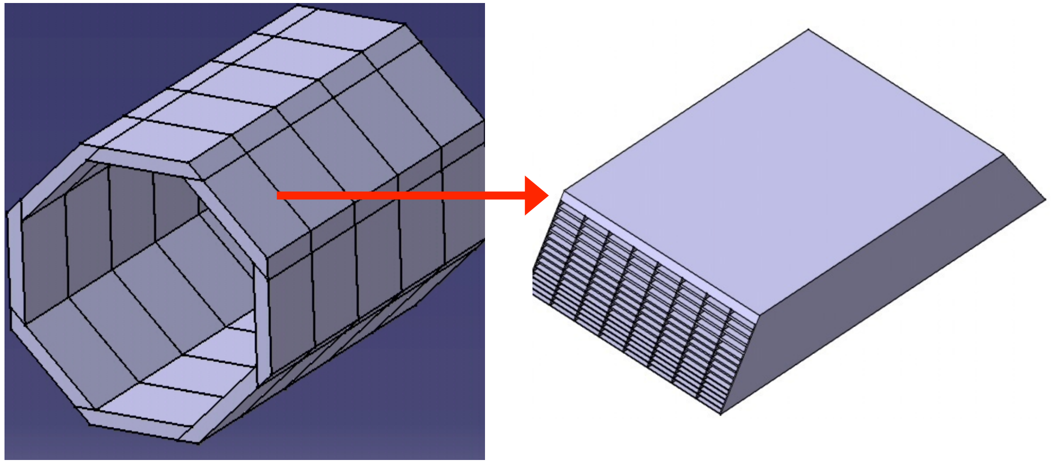


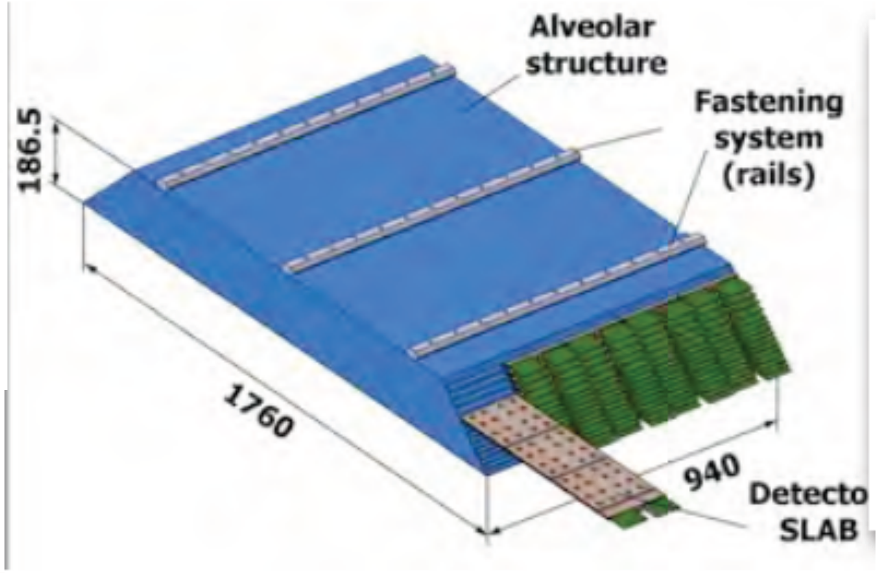
Sub-Detector	LHC	ILC
Pixel	150 M	800 M
Microstrips	~ 10 M	~30 M
Fine grain trackers	~ 400 K	1,5 M
Calorimeters	200 K	30 M
Muon	~1 M	



Example of subdetector: ECAL

- The ECAL
  - *High granularity, particle separation, compactness*
  - *=> Sampling calorimeter (~30 layers)*
    - Silicon diodes or scintillator separated by tungstene
  - *Lateral cell size : 5 x 5 mm^2*
  - *100 Million of channels*
  - *FE electronics embedded in detection layer*



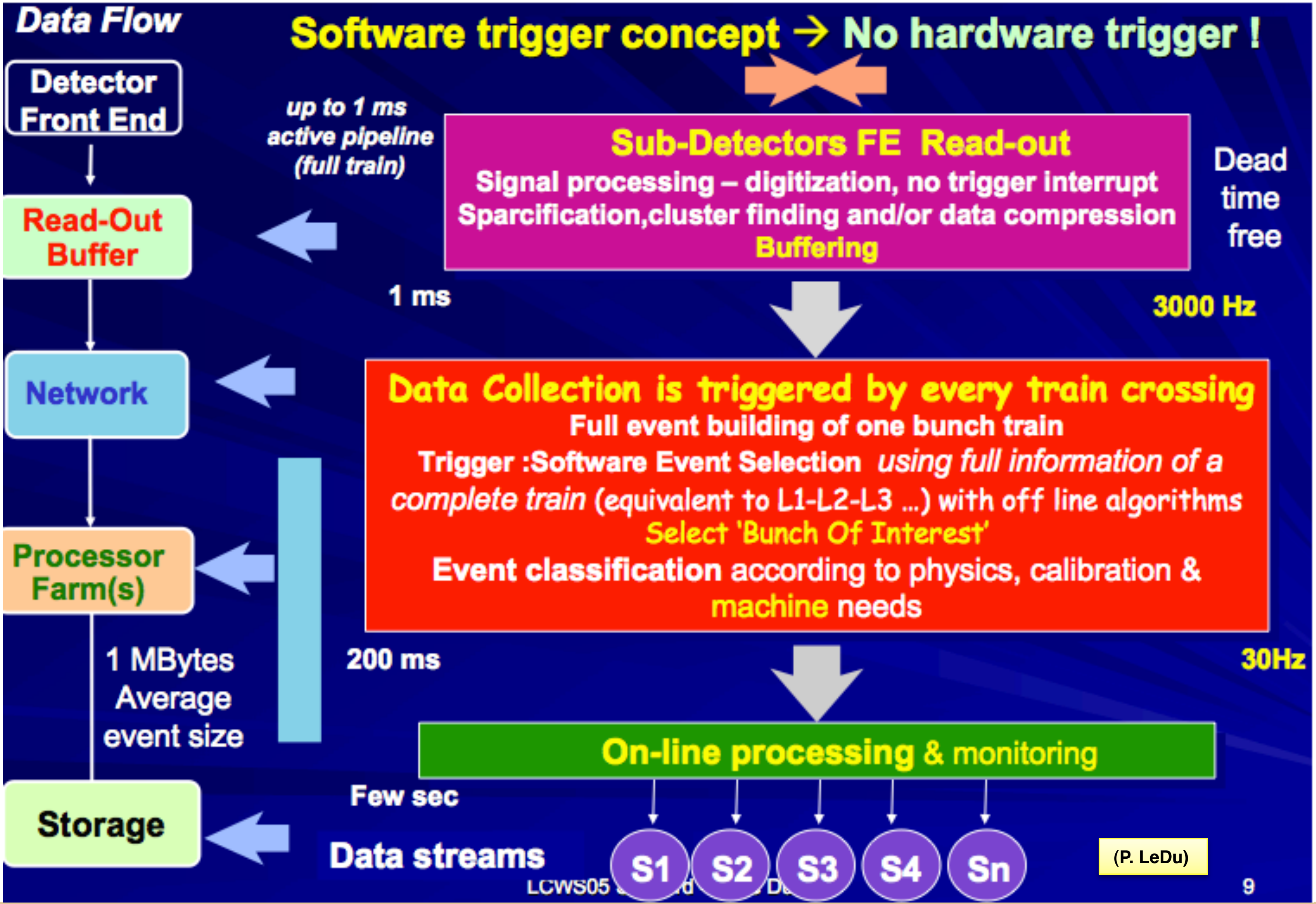


- ➢ Small dead zone area
- ➢ All modules are identical (Tungsten wrapped by Cfi)
- ➢ The detector slabs would be tested before assembling

ILC DAQ

- Basic scheme:
  - *The FEE integrates everything*
    - from signal processing & digitizer to the RO BUFFER
  - *Very large number of channels to manage (Trakers & EM)*
  - *should exploit power pulsing to cut power usage during interburs*
  - *Interface between detector and machine is fundamental to optimize the luminosity*
    - -> consequences on the DAQ
  - *Burst mode allows a fully software trigger !*
    - Looks like the Ultimate Trigger: Take EVERYTHING & sort later !

ILC DAQ concept



Data Flow

Software trigger concept → No hardware trigger !

up to 1 ms active pipeline (full train)

Sub-Detectors FE Read-out

Signal processing – digitization, no trigger interrupt

Sparcification,cluster finding and/or data compression

Buffering

Dead time free

1 ms

3000 Hz

Data Collection is triggered by every train crossing

Full event building of one bunch train

Trigger :Software Event Selection *using full information of a complete train* (equivalent to L1-L2-L3 ...) with off line algorithms

Select 'Bunch Of Interest'

Event classification according to physics, calibration & machine needs

1 MBytes Average event size

200 ms

30Hz

On-line processing & monitoring

Few sec

Data streams

(P. LeDu)

LCWS05

9

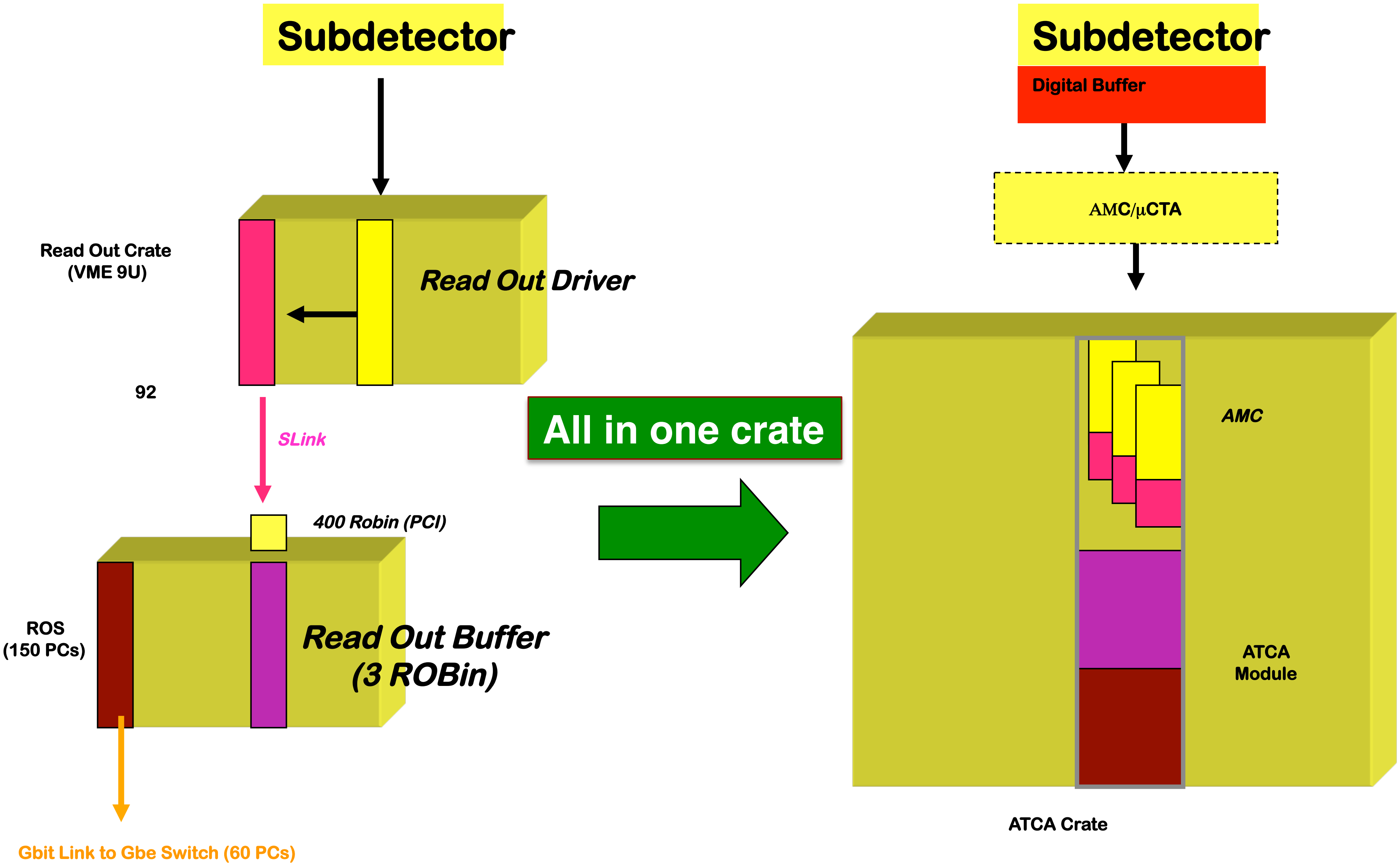
Advantages

- Flexible
  - *Fully programmable: unforeseen backgrounds and physics rates easily accomodated*
- Easy maintenance and cost effective
  - *Commodity products:Off The Shelf products (Links, memory, switches, processors)*
  - *Commonly OS and high level languages*
  - *on-line computing ressources usable for «off-line»*
- Scalable :
  - *Modular system*
- Looks like the ‘ ultimate trigger ’
  - *satisfy everybody : no loss and fully programmable*

Technology

- FPGA for signal processing and buffering
  - *Integrates receiver links, PPC, DSPs and memory …*
- Processors and memories
  - Continuous increasing of the computing power
  - Memory size quasi illimited !
    - Today : 256 MB 2010 : > 1 GB ... then ?
- Links & Networks:
  - Commercial telecom/computer standard
  - 10 -30- 100 GBEthernet !
- Systematic use of COTS (Comm. Off The Shell) products
  - => can wait 2-3 years before start to make a decision

Readout evolution LHC -> ILC



Conclusions

- With this series of lectures we covered the entire path the data are following from the detector up to the storage for offline analysis.
- DAQ and triggers are very dependent on the experiment and the physics
- Therefore I realize that this only gives you a general picture on how DAQ and Triggers are working
- but I really hope that you learned something

My sources:

- Unfortunately there is no recent book on T/DAQ but:
  - *Lectures of N. Neufeld, CERN Summer Student progr. 2010*
  - *Lectures of J. Christiansen, CERN Summer Student progr. 2009*
    - And their predecessors
  - *The Technical Design Reports of CMS, ATLAS, LHCb & ALICE*
  - *Electronics lecture from Ch. de La Taille*