Summer Student
Lecture Program 2022

Electronics, Trigger and Data Acquisition. 1/3

E. Meschi – CERN EP Dept. – CMS Experiment

- Trigger and DAQ system concepts
- From signal to physics through examples
- Timing
- Data transport, links, buses
- Queues and Event building
- On-line data processing

Credits:
Past SSLP ETD lecture series
EM: lectures on DAQ/Trigger at U.Padua 2018-2020
ISOTDAQ: International School of Trigger and DAQ
https://indico.cern.ch/event/928767/
Material from various papers and books (bibliography at the end)

A quick tour – menu

- Introduction:
  - Sensors, detectors, experiments, historical perspective
  - Acquiring data from sensors
- Basics of analog signal processing
- From analog to digital
- Measuring time
- Trigger
- Role of CPU and data buses
- Event building

Examples taken from nuclear and (mostly) particle physics

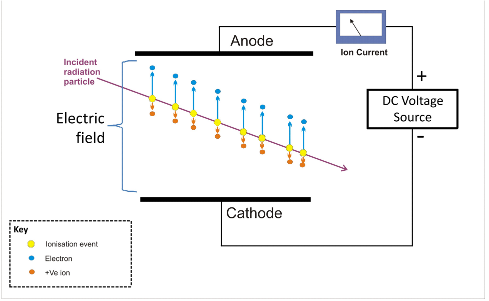
Sensors

- In modern parlance, “a device, module, machine, or subsystem whose purpose is to detect events or changes in its environment and send the information to other electronics”
- In practice, any device that **detects or measures a physical event or quantity** and transforms this event or quantity into another that is **“easier” to perceive and/or measure** – sensors and transducers can be often exchanged/confused
- In most cases today, the final quantity is an electrical signal (either steady or transient)
- In what follows, we will be dealing mostly with transducers that produce an electrical signal, in most cases a **pulse**

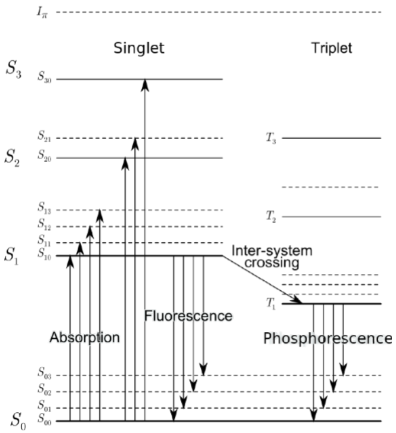
Detectors

In real life, we often deal with a complex of (one or more) sensors or transducers, not necessarily homogeneous. We refer to this complex as a “detector” and this often includes the electronics used to read out and process the information about the physical quantity or event. In NP and HEP, when we say “detector”, we almost always mean “ionizing particle detector”

Sometimes we mean a whole experiment (“the CMS detector”)



$$-\left\langle \frac{dE}{dx} \right\rangle = \frac{4\pi}{m_e c^2}\cdot\frac{nz^2}{\beta^2}\cdot\left(\frac{e^2}{4\pi\varepsilon_0}\right)^2\cdot\left[\ln\left(\frac{2m_e c^2\beta^2}{I\cdot(1-\beta^2)}\right)-\beta^2\right]$$



Detectors and Signals

- Sometimes, somewhere, something happens → in some short time, several **particles interact within our detector**
- Even a single particle interaction is composed of many different **probabilistic (quantum-mechanics) processes** → fluctuations are built-in
- Practically all modern detectors, at the end, generate “electrical” signals at their output terminals. This signals:

  a) have different characteristics (**size, arrival time, duration**, …)

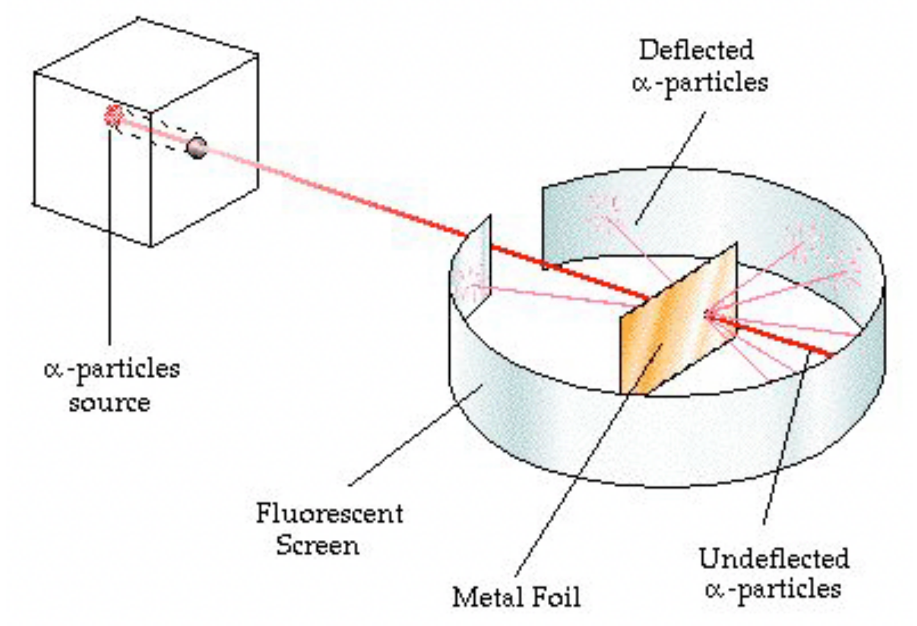
  b) carry different (normally independent) **information**

  c) require some **electronics** in order to become “usable“ to measure a physical quantity

A bit of history…

- Experiments of the past often used analog (or sensorial) means to measure/register a phenomenon
  - Visual or aural observation
    - Often involving counting the occurrences of some phenomenon
    - …and taking note (recording)
  - (analog) photography was often used for more complex observations (emulsion experiments, bubble chambers)
- Counting and recording information are all things a computer does better **once the information is in digital form**

Rutherford Scattering



A bit of history…

- Experiments of the past often used analog (or sensorial) means to measure/register a phenomenon
  - Visual or aural observation
    - Often involving counting the occurrences of some phenomenon
    - …and taking note (recording)
  - (analog) photography was often used for more complex observations (emulsion experiments, bubble chambers)
- Counting and recording information are all things a computer does better **once the information is in digital form**

Role of Electronics, Trigger and DAQ

Process the signals generated in a detector and save (only) the interesting information on a permanent storage medium

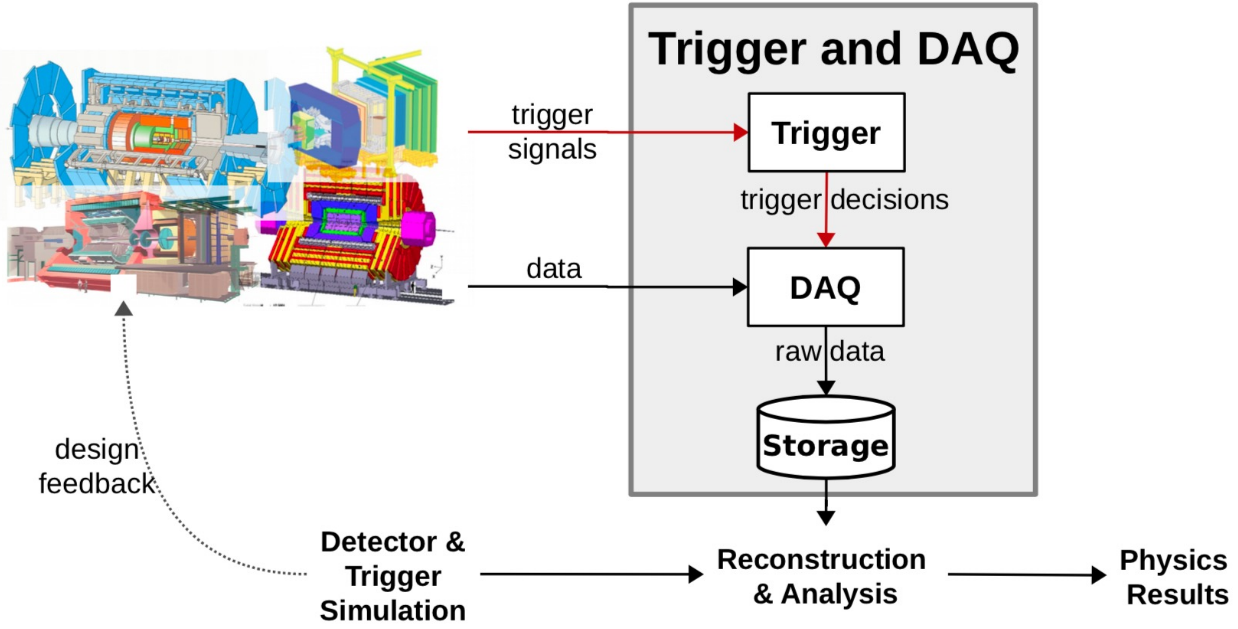
Modern DAQ is all about digital information

However, physics is not digital…sensors produce analog signals that **must be treated and interpreted** before being **digitized**

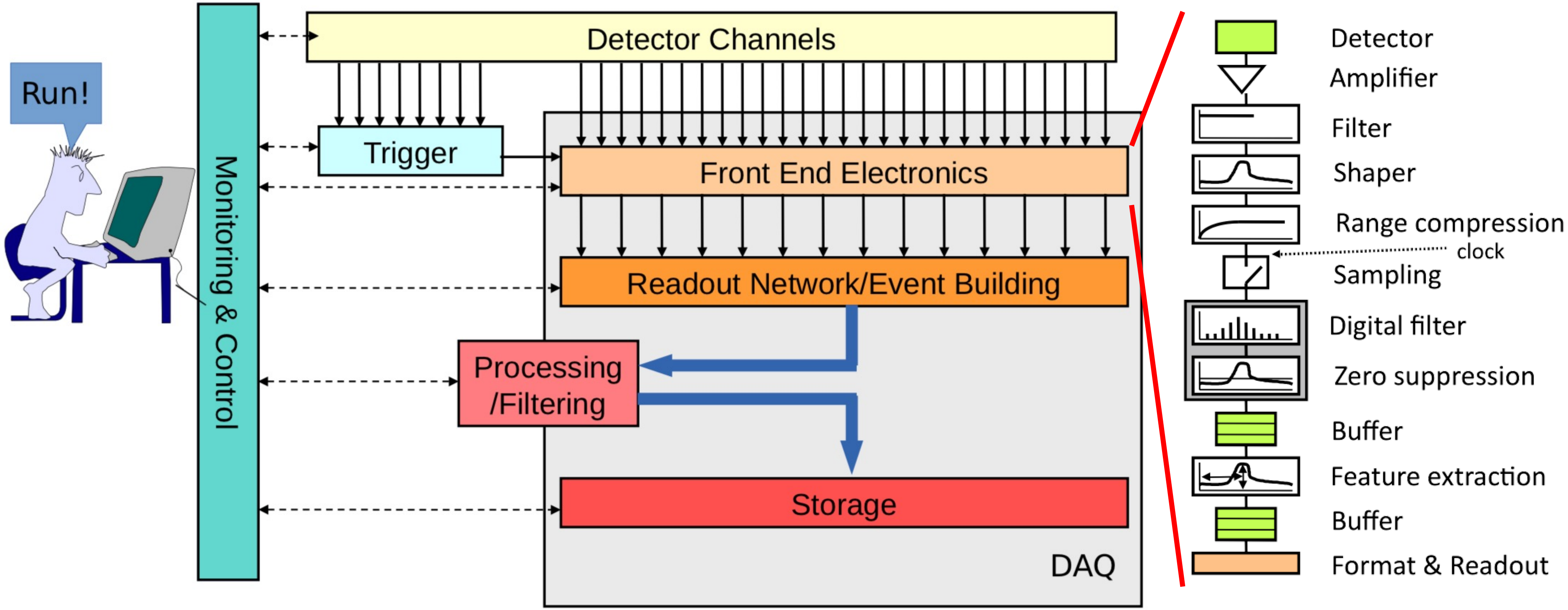
Most of the “real” physics analysis (the one that gets you the Nobel prize) happens ”offline”

(what “offline” means has changed over time) however:

→There is a lot of physics (and math, and technology) that you only learn in DAQ and trigger←



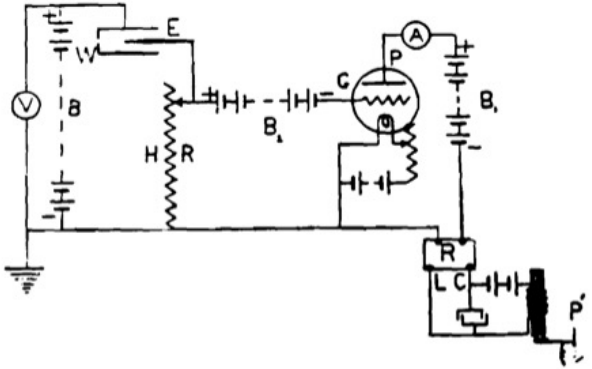
A modern Trigger/DAQ looks like this

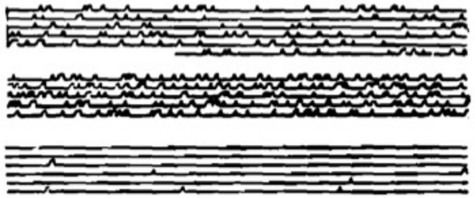


(Probably) the first DAQ

All the main components constituting an experiment, its acquisition **and trigger** are there.
Can you recognize them ?
The detector (E) is connected to a programmable trigger (HR) and analog front-end electronics (G) is acquired (R) and recorded on “digital” media by means of (P’)

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





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

Coincidence (trigger)

- Bruno Rossi (Nature, 1930):
- "Method of Registering Multiple Simultaneous Impulses of Several Geiger Counters"
- → online coincidence of 3 signals!

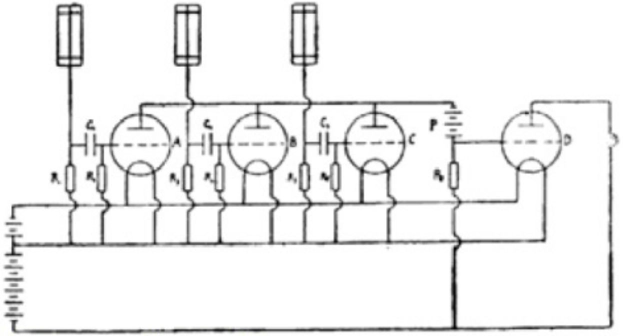


Fig. 17 – Il circuito di Rossi per rivelare coincidenze di raggi cosmici che arrivano sui contatori Geiger (i rettangoli in alto dello schema)[19].

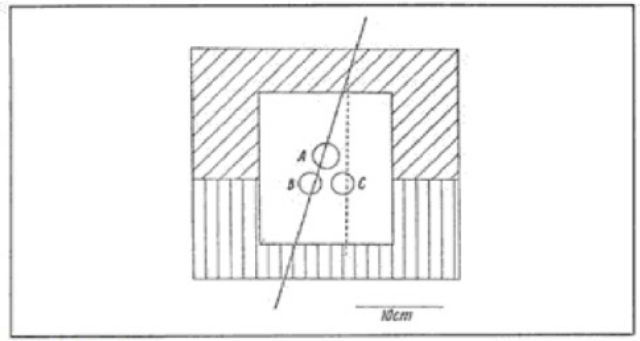


Fig. 18 – L'uso del circuito di Rossi per rivelare una coincidenza tripla che, nella disposizione in figura dei tre contatori, mostra la produzione di una radiazione secondaria (linea tratteggiata) da parte della radiazione primaria (linea continua)[20].

A recent one

L1-Trigger/HLT/DAQ

hTracks in L1-Trigger at 40 MHz

- PFlow-like selection 750 kHz output
- HLT output 7.5 kHz

Barrel Calorimeters

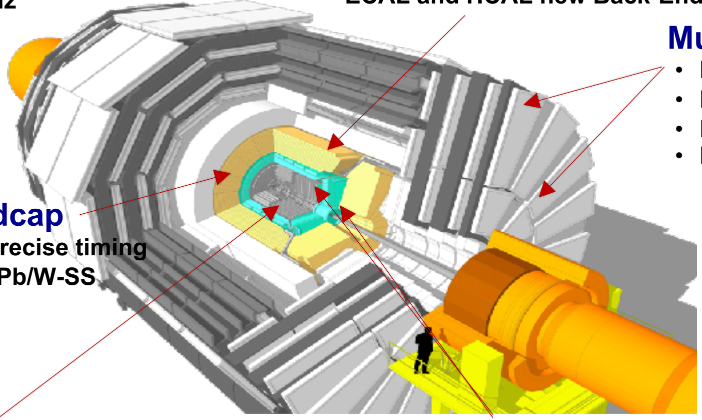
- ECAL crystal granularity readout at 40 MHz with precise timing for e/γ at 30 GeV
- ECAL and HCAL new Back-End boards

Muon systems

- DT & CSC new FE/BE readout
- RPC back-end electronics
- New GEM/RPC 1.6 < η < 2.4
- Extended coverage to η ≃ 3

Calorimeter Endcap

- 3D showers and precise timing
- Si, Scint+SiPM in Pb/W-SS



Tracker

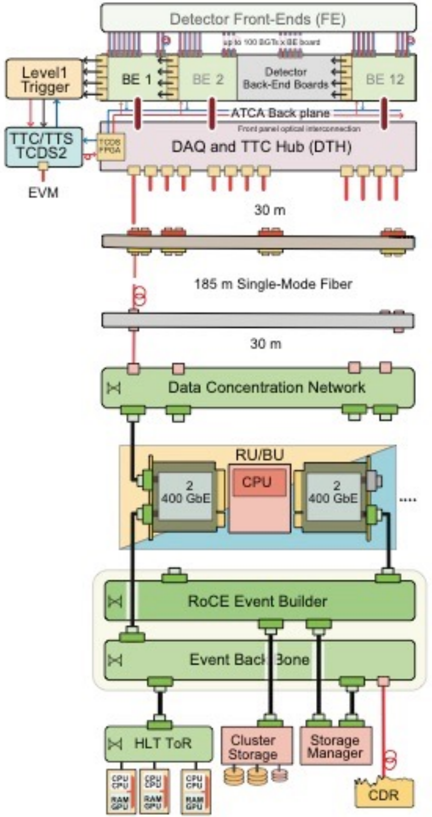
- Si-Strip and Pixels increased granularity
- Design for tracking in L1-Trigger
- Extended coverage to η ≃ 3.8

MIP Timing Detector

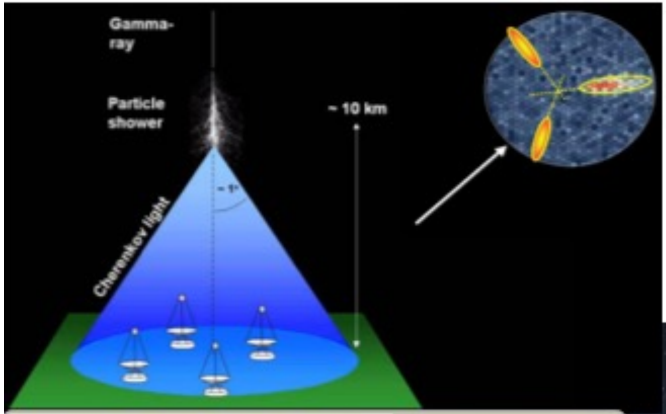
Precision timing with:

- Barrel layer: Crystals + SiPMs
- Endcap layer: Low Gain Avalanche Diod

Beam Radiation Instr. and Luminosity, and Common Systems and Infrastructure



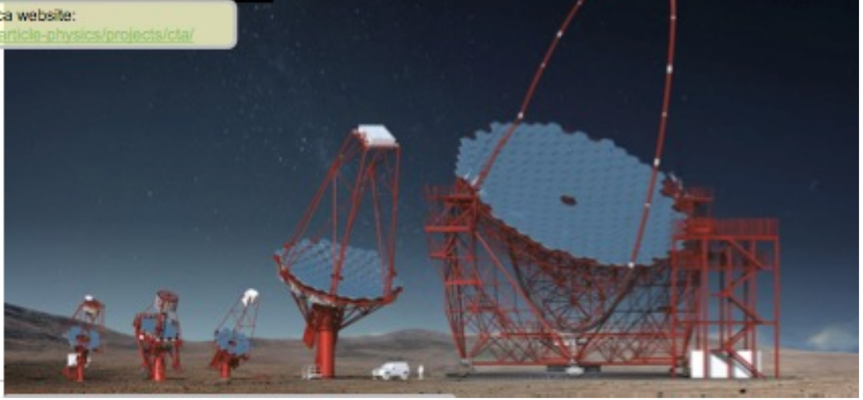
Not only for colliders…



Picture University of Nova Gorica website:
http://www.ung.si/en/research/laboratory-for-astroparticle-physics/projects/cta/

cta
cherenkov telescope array

- Very high energy γ-ray observatory
- Two arrays of 100 and 20 telescopes



H.E.S.S.

...

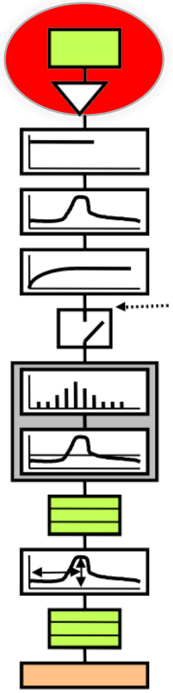
Picture SKA website: https://www.skatelescope.org/

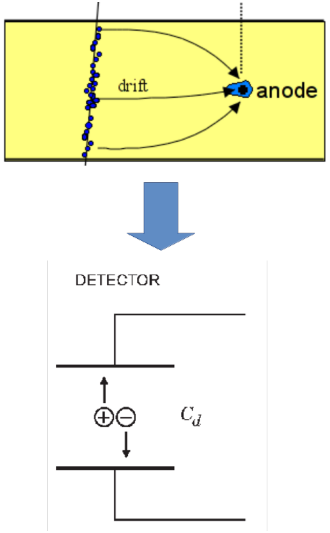
SKA

SQUARE KILOMETRE ARRAY

A quick tour of the DAQ/Trigger Chain

Signals from a detector: amplification





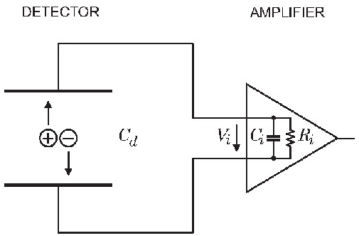
Detectors may be electrically represented as a capacitor C_d (more realistic schemes will include other contributions)

Interactions of passing particle ➔ energy release E ➔ short current pulse i_s

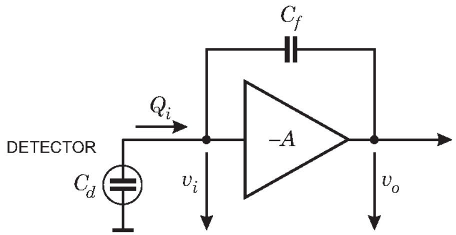
Weak signals require amplification:

- adapts it to next stages
- avoids Signal-to-Noise-Ratio (SNR) degradation

$$E \propto Q_s = \int i_s(t)dt$$



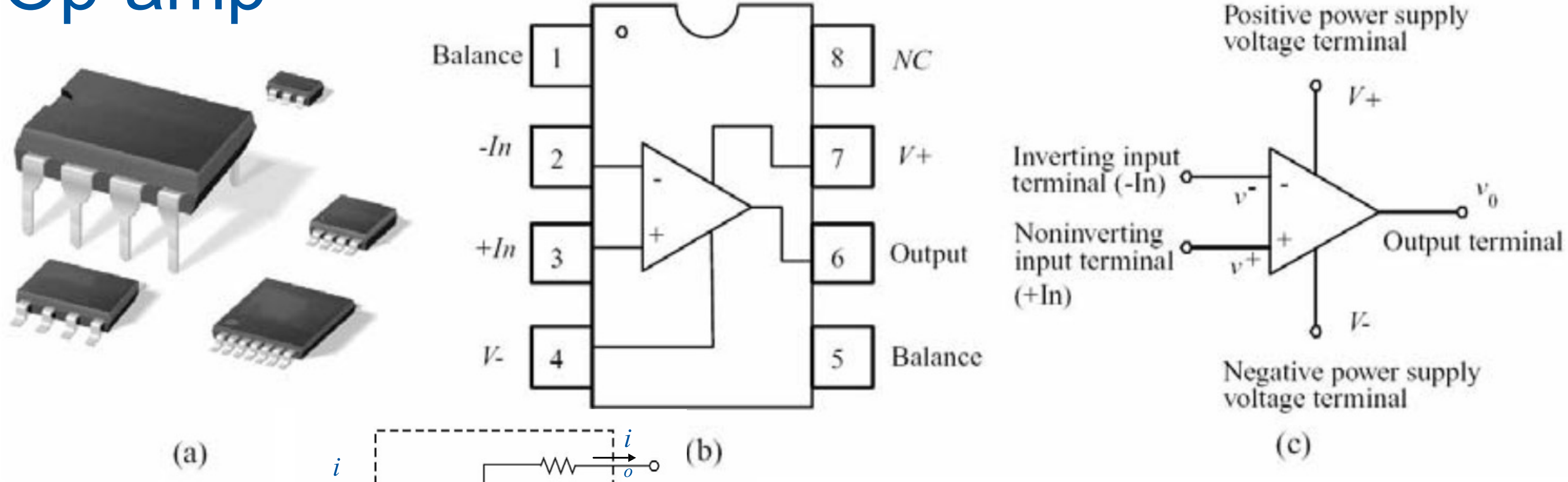
voltage



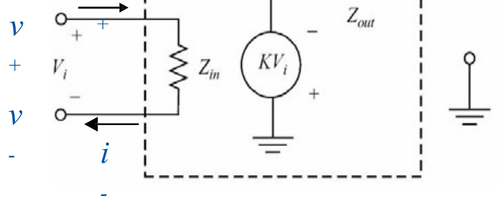
current

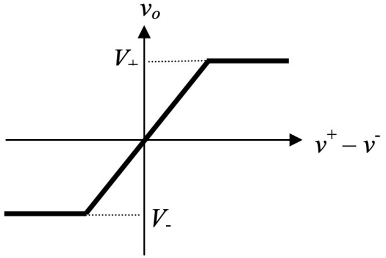
A current-sensitive amplifier provides a signal That **does not depend on** **C_d** – more on this later if time

Op-amp



Balance
1
-In
2
+In
3
V-
4
8
NC
7
V+
6
Output
5
Balance
-
+
Positive power supply voltage terminal
V+
Inverting input terminal (-In)
v-
Noninverting input terminal (+In)
v+
-
+
v_0
Output terminal
V-
Negative power supply voltage terminal
(a)
(b)
(c)

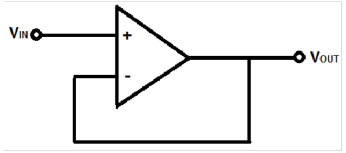


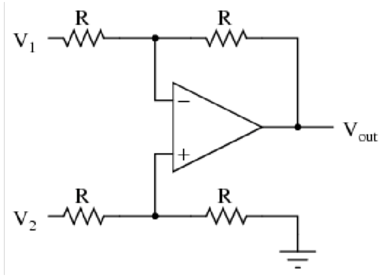


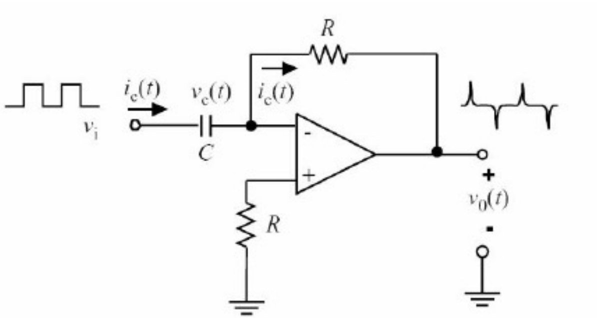
The gain of an op-amp (K) ranges from 10^4 to 10^7 with a typical value of 10^5. To amplify the difference between the input signals, the op-amp draws power from an electrical power supply. If V_+ and V_- denote the positive and negative voltages provided by the power supply, the output of the op-amp cannot exceed these limits and therefore saturates at these levels, as shown in the figure on the left.

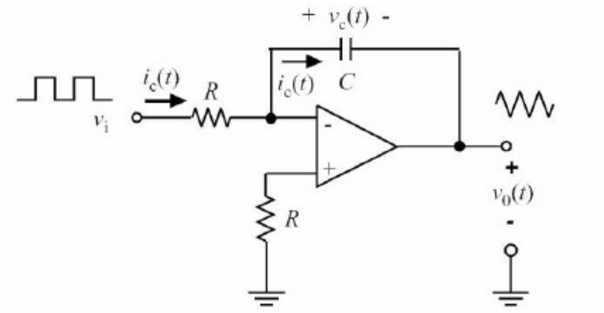
Voltage amplifiers with: a) 2 (differential) inputs b) high gain (A+ = A- = ∞) c) high input impedance (Zin = ∞) d) low output impedance (Zout = 0)

Op-amp circuit examples

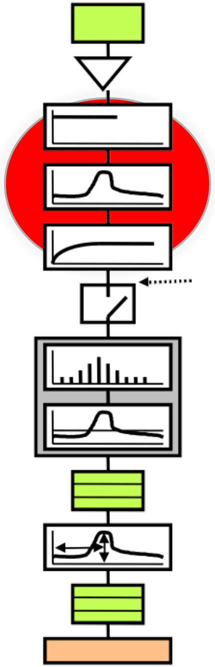








Signals from a detector: shaping



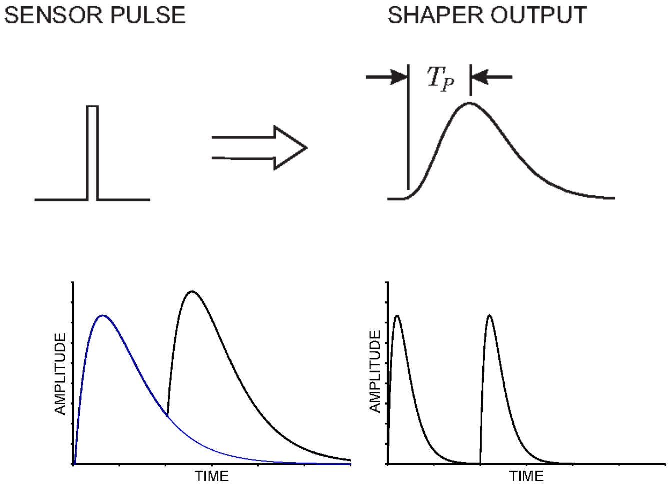
Reduce signal bandwidth (low-pass filter) ➔ improve SNR

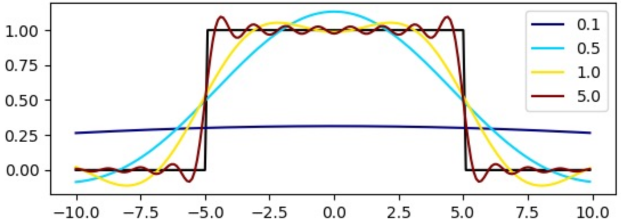
fast rising signals have large bandwidth

shaper broadens signals

Limit pulse width (high-pass filter) ➔ avoid overlap of successive pulses

increase maximum signal rate **at the cost of more noise**

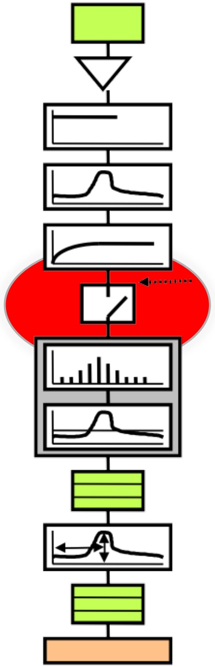




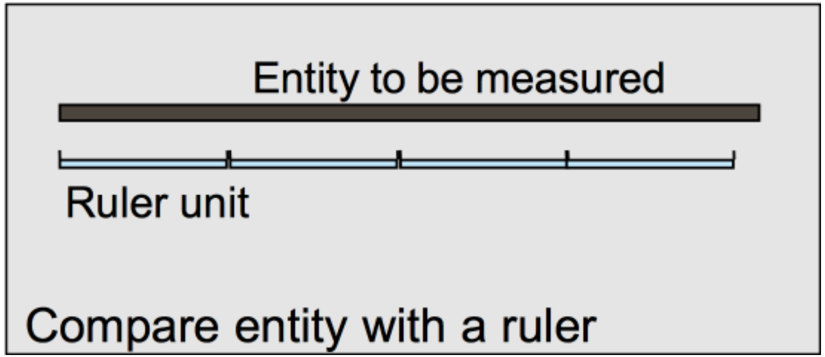
Analog signal treatment

- Many other aspects
  - Real-life amplifiers
  - Charge-sensitive amplifiers, integration
  - Gate generation, delay
  - Signal transmission: reflection, impedance matching
  - Response function of an apparatus
- Many good textbooks to go in depth
- We will now move on to digital
  - We will start with analog-to-digital conversion

Analog to digital conversion (sampling)



Reminder: we need an Analog to Digital Converter (**ADC**) to turn our voltage pulse into a binary number for processing and storage



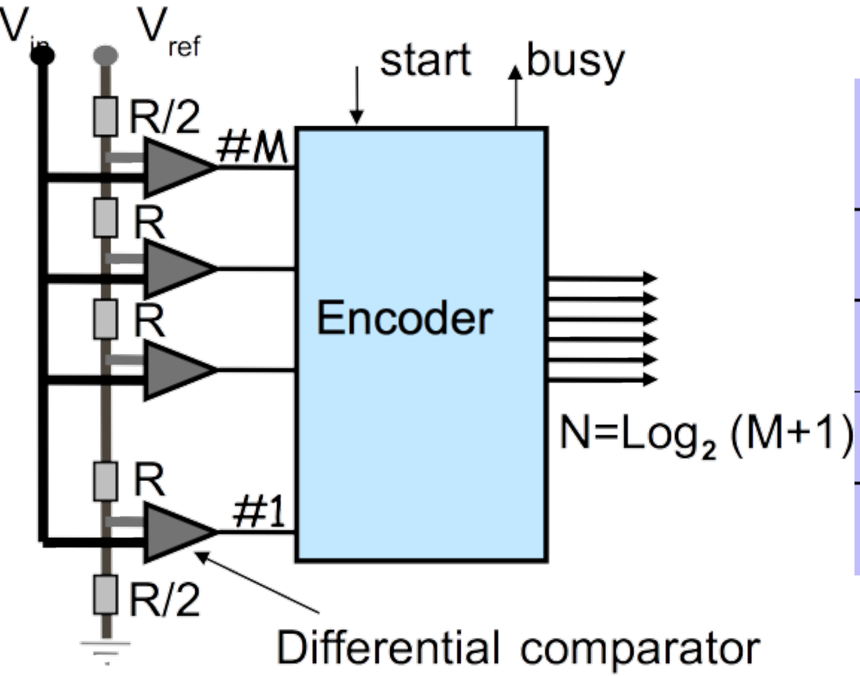
Entity to be measured

Ruler unit

Compare entity with a ruler

In its simplest form, an ADC compares the signal with M fractions of a reference voltage (the unit ticks on the ruler)
In a nutshell, this is the working principle of the **FlashADC**

Digitizing a voltage pulse: flash ADC



binary code

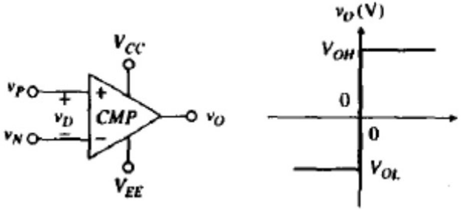
V_{in}/V_{ref}	Comparison results	Encoded form
<1/6	000	00
1/6≤ <3/6	001	01
3/6≤ <5/6	011	10
5/6≤	111	11

Thermometer code

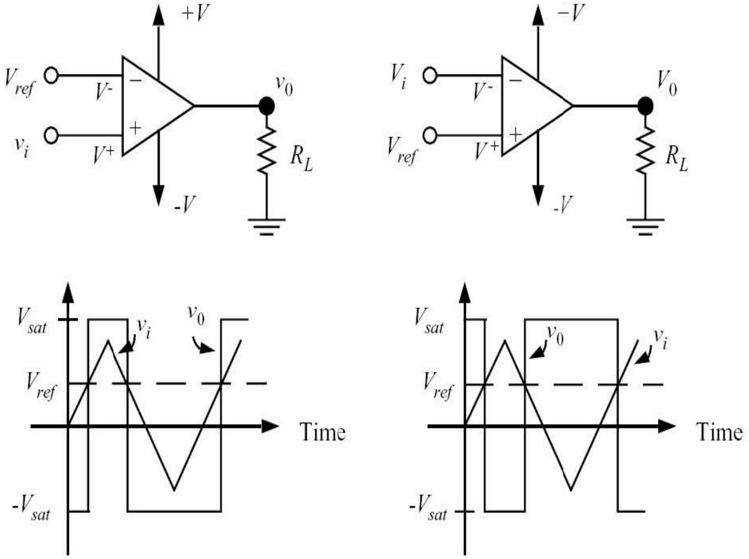
Voltage comparator, op amps

$v_O = V_{OL}$ for $v_P < v_N$

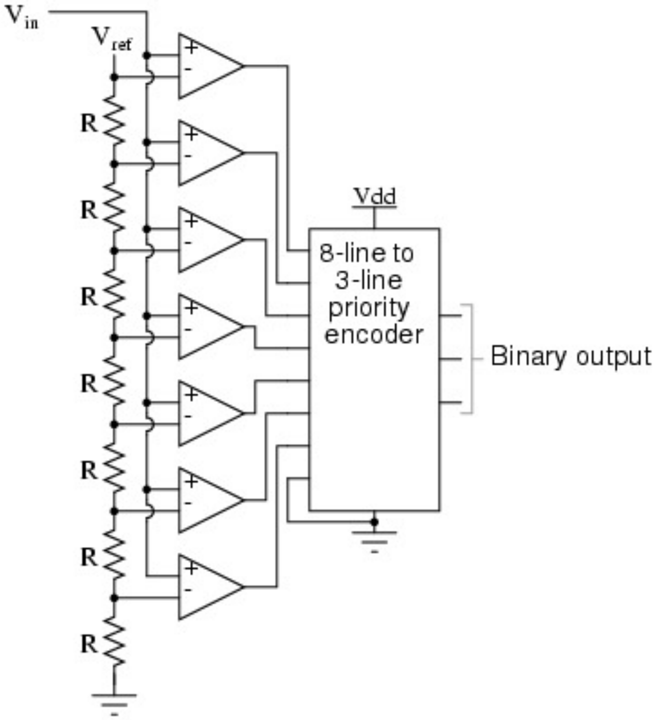
$v_O = V_{OH}$ for $v_P > v_N$

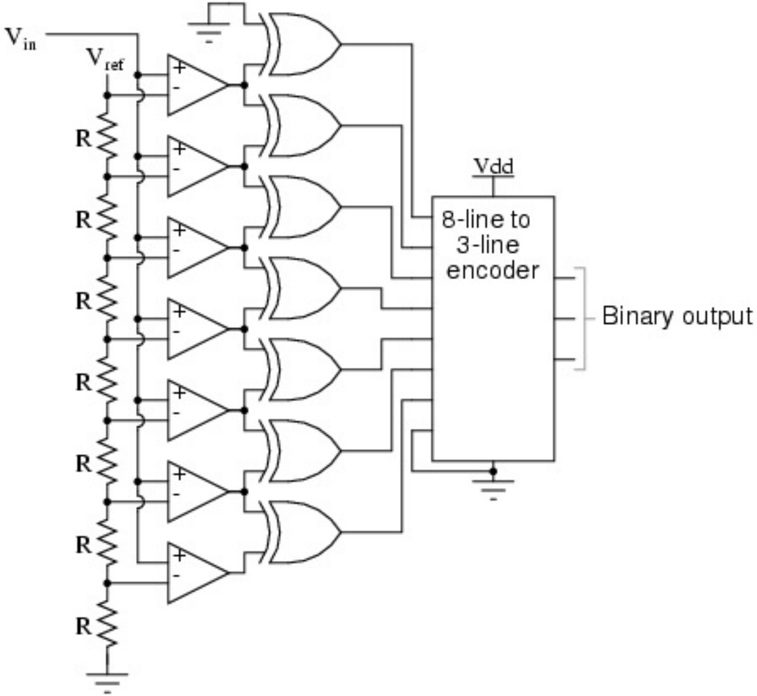


voltage transfer curve

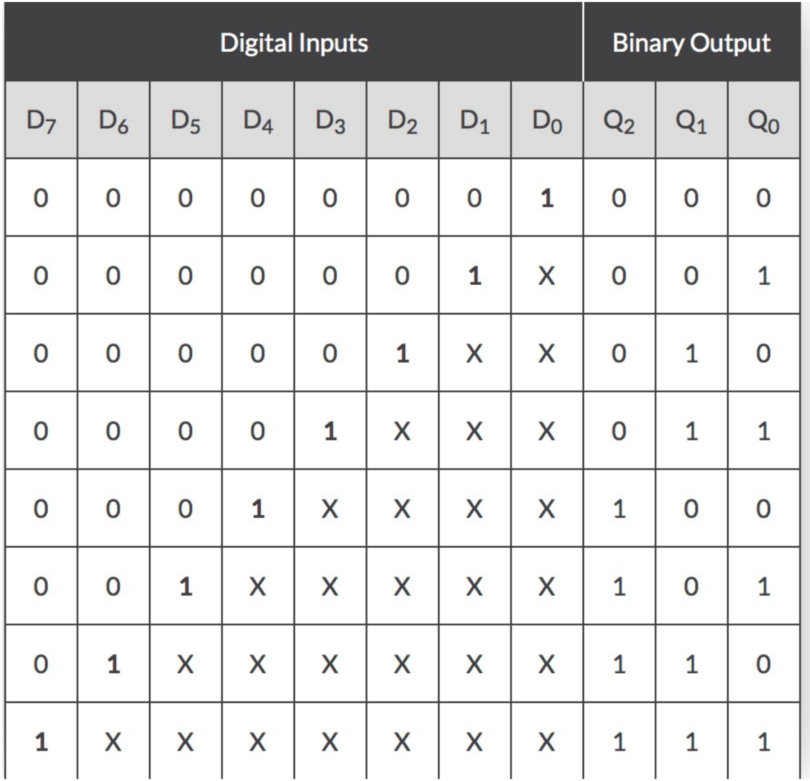


Thermometer or one-hot





Thermometer to binary: priority encoder



Digital Inputs								Binary Output		
D_7	D_6	D_5	D_4	D_3	D_2	D_1	D_0	Q_2	Q_1	Q_0
0	0	0	0	0	0	0	1	0	0	0
0	0	0	0	0	0	1	X	0	0	1
0	0	0	0	0	1	X	X	0	1	0
0	0	0	0	1	X	X	X	0	1	1
0	0	0	1	X	X	X	X	1	0	0
0	0	1	X	X	X	X	X	1	0	1
0	1	X	X	X	X	X	X	1	1	0
1	X	X	X	X	X	X	X	1	1	1

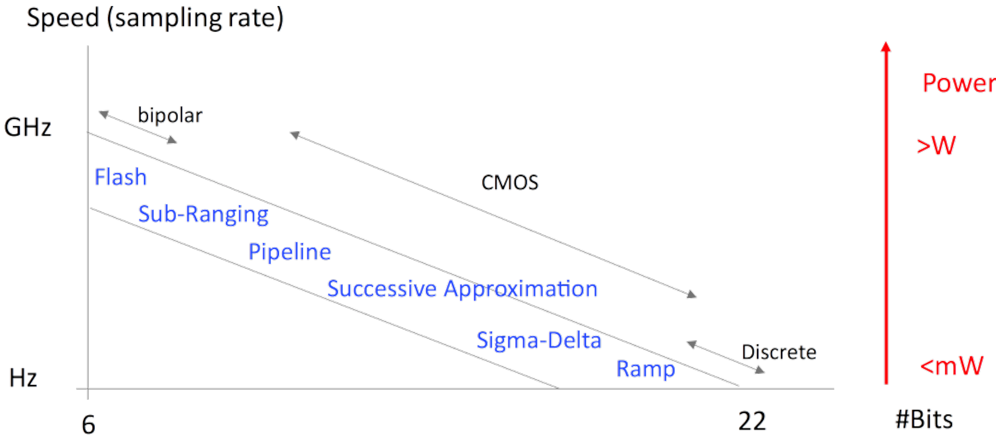
Exercise: use the truth table to design the 8 to 3 priority encoder

How do things change for the one-hot case ?

Do you see any advantages/disadvantages in one or the other ?

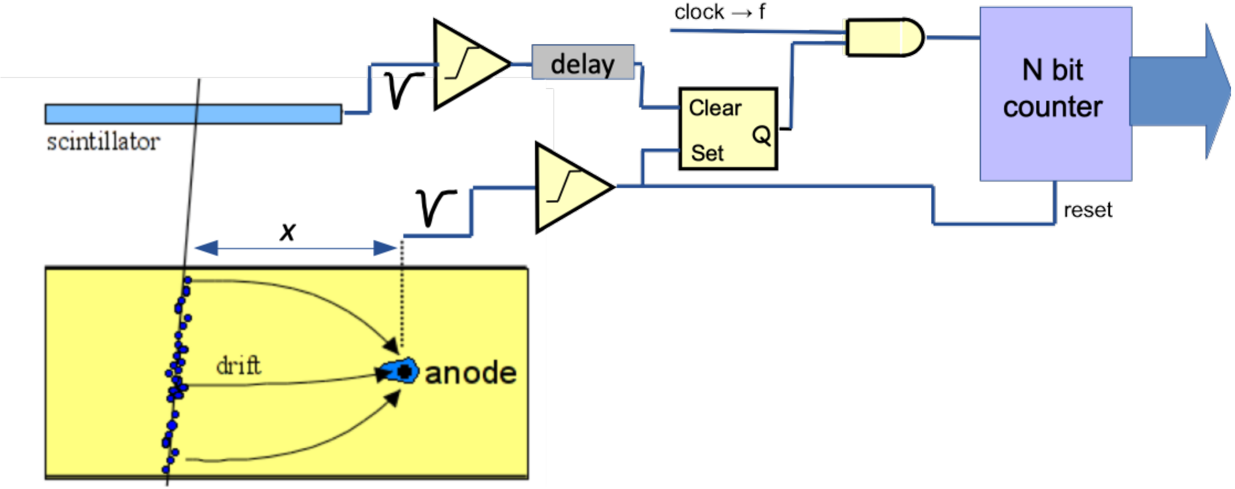
Flash ADC: the simplest (and fastest)

- **Resolution** (Least Significant Bit), the ruler unit:
  - V_{max} /2N
  - E.g. 8bit, 1V → LSB=3.9mV
- Quantization error
  - because of finite size of the ruler unit: ± LSB/2
- Dynamic range: Vmax /LSB
  - N for linear (flash) ADC
  - >N for non-linear ADC
- Flash ADC has constant relative resolution on the valid input range



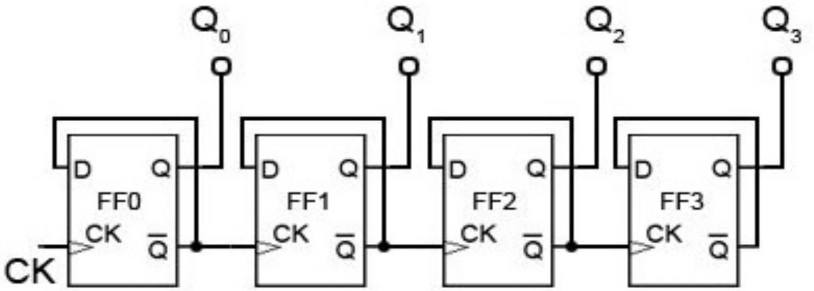
Measuring Time

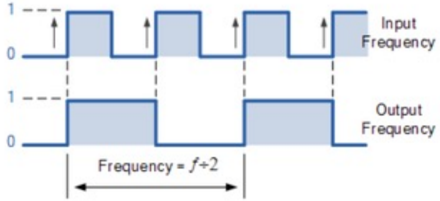
Time-to-digital Converter

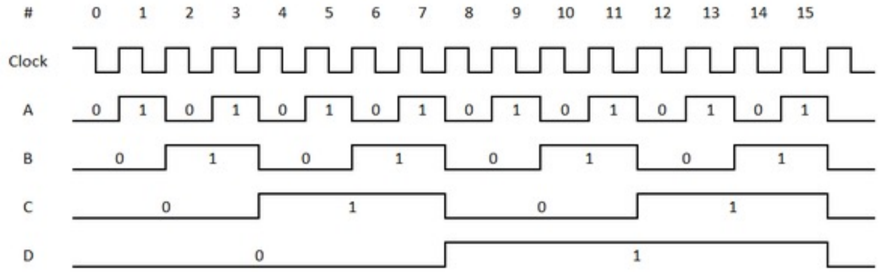


- TDC principle is quite simple: count regular pulses from a start to a stop signal
- Resolution: 1/*f*
- Dynamic range: N
- Single hit TDC
  - e.g. a noise spike comes just before the signal → measure is lost

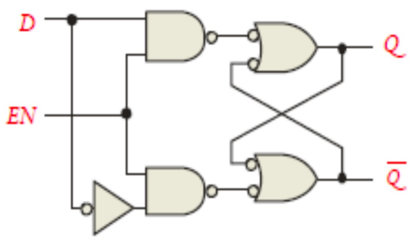
Counter

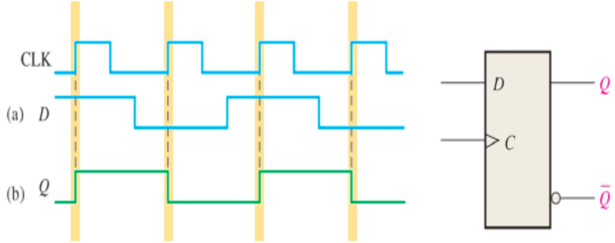






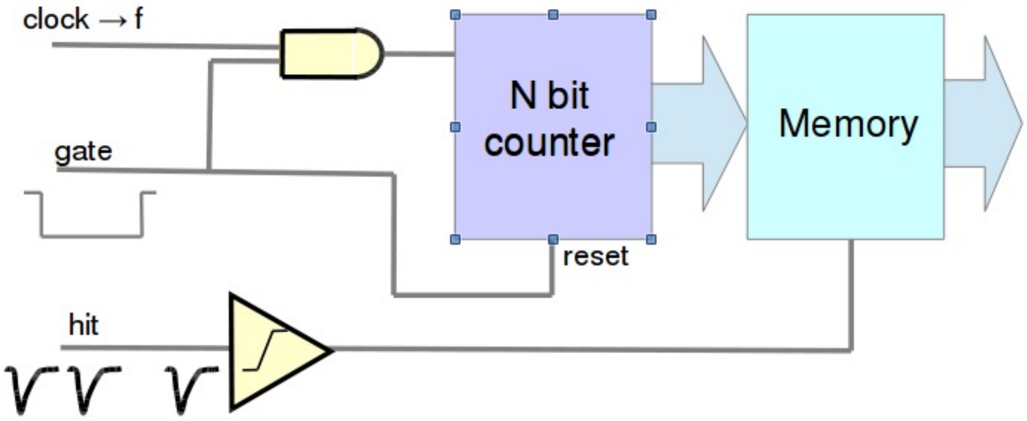
Q follows D on the rising edge of the clock





D-latch and D-type FF

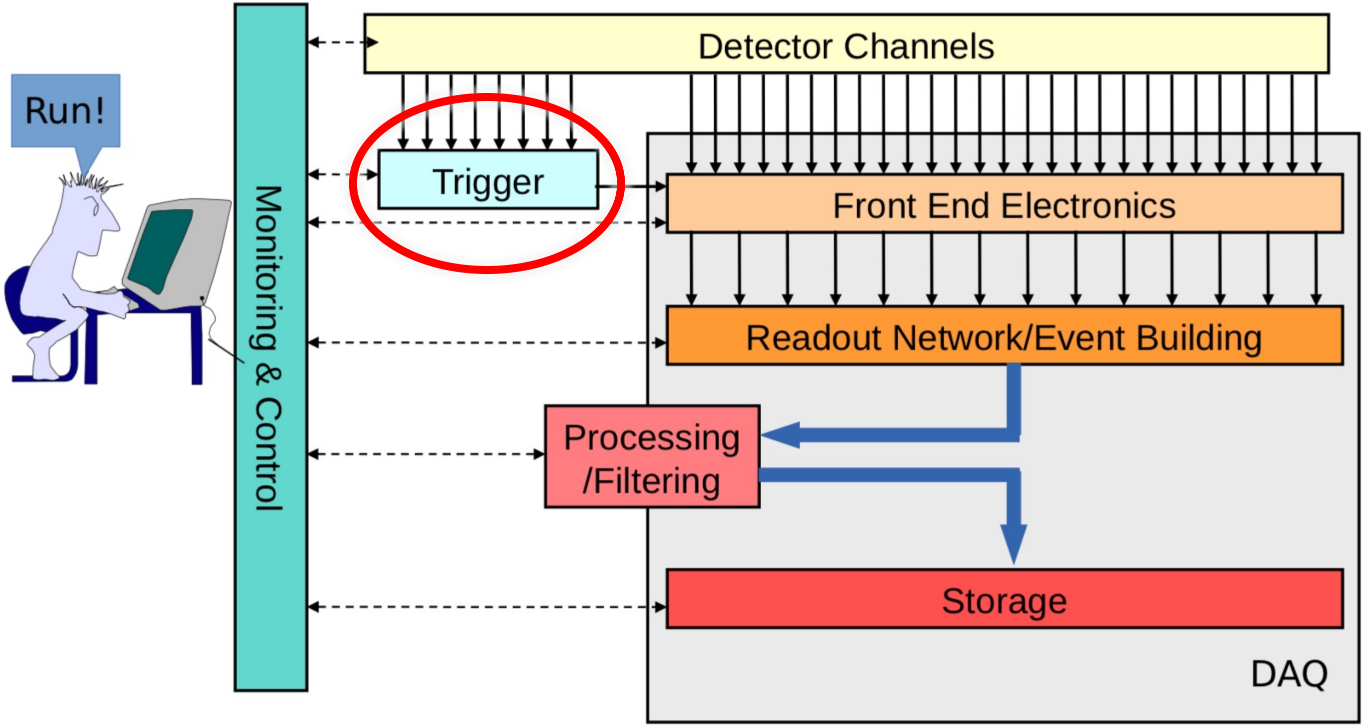
Multi-hit TDC



- Gate resets and starts the counter. It also provides the measurement period. It must be smaller than $2^N/f$
- Each “hit” (i.e. signal) forces the FIFO to load the current value of the counter, that is the delay after the gate start
    - In order to distinguish between hits belonging to different gates, some additional logic is needed to tag the data
- Common-start configuration
- This is e.g. a typical configuration FOR A COLLIDER EXPERIMENT:

Trigger

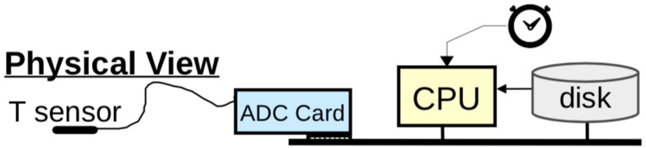
Trigger

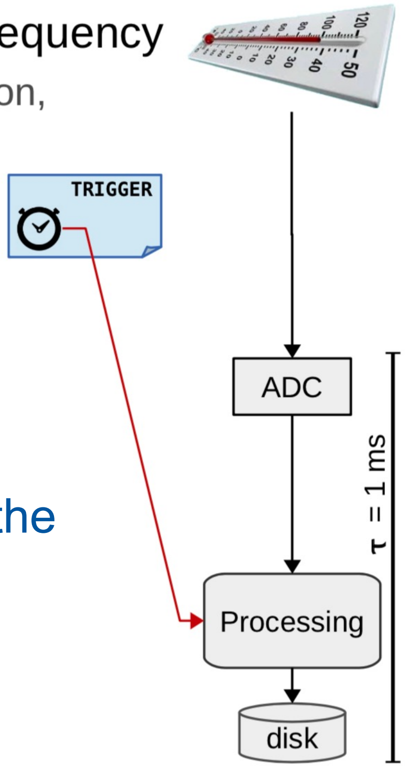

Run!
Monitoring & Control
Detector Channels
Trigger
Front End Electronics
Readout Network/Event Building
Processing /Filtering
Storage
DAQ

Sampling with a periodic clock

- Es: measure temperature at a fixed frequency
  - ADC performs analog to digital conversion, digitization (our front-end electronics)
  - CPU does readout and processing
- System clearly limited by the time τ to process an "event"

Questions for later:
How are data stored ?
What does it mean that "CPU does the readout" ?



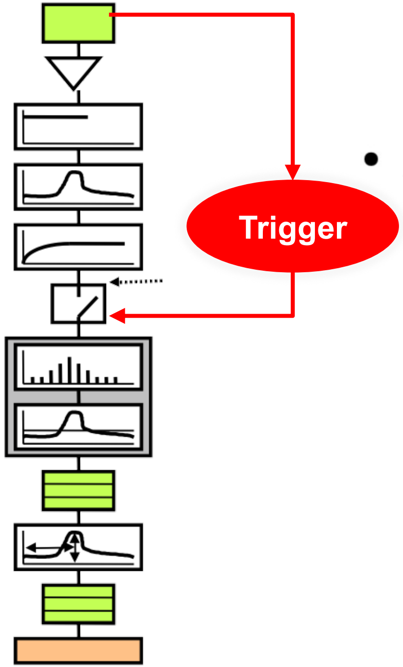


Fully sequential system

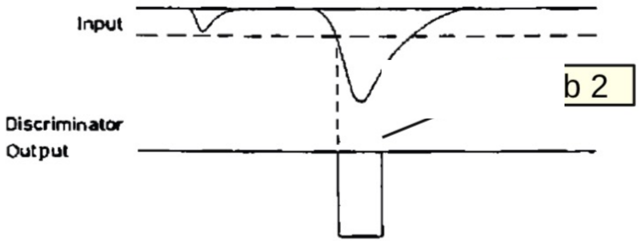
System limited by single-event processing time

If τ ~ 1 ms for ADC conversion +CPU processing +storage → can sustain up to ~ 1 kHz of ***periodic (synchronous) trigger*** rate

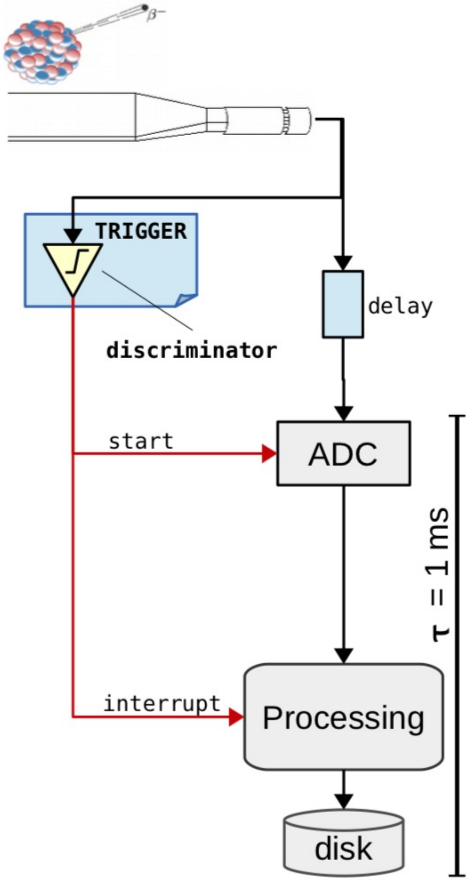
What if…



- Events asynchronous and unpredictable
  - E.g.: beta decay studies
- A physics trigger is needed
  - **Discriminator**: generates an output signal only if amplitude of input pulse is greater than a given threshold

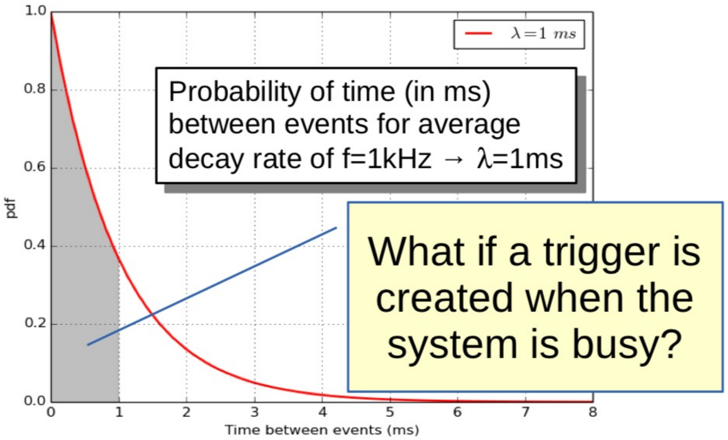


  - delay introduced to compensate for the **trigger latency**



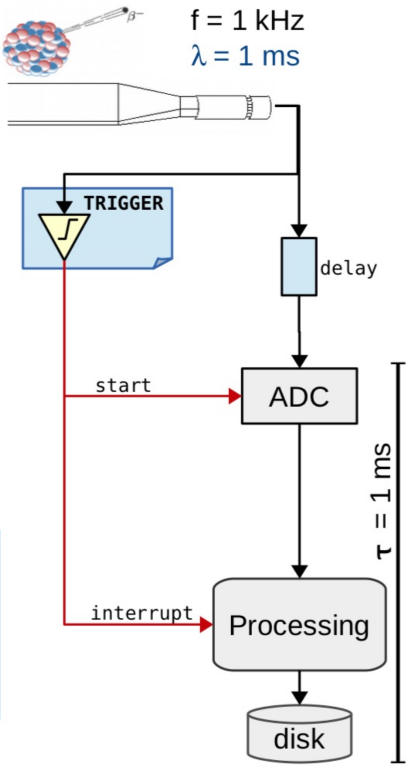
Sampling a physics process

- Stochastic process
  - Fluctuations in time between events
- Let's assume for example
  - A physics rate f = 1 kHz, i.e. λ = 1 ms
  - and, as before, τ = 1 ms



Probability of time (in ms) between events for average decay rate of f=1kHz → λ=1ms

What if a trigger is created when the system is busy?



What happens if
New trigger arrives while system is busy ?

a) Each new trigger is accepted and "restarts" the process

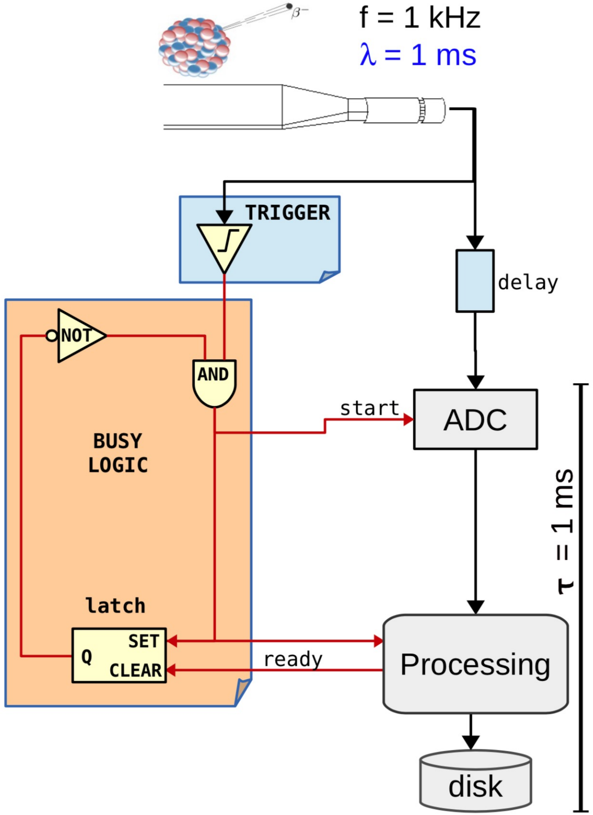
-> **paralysable**

b) No new trigger is accepted until the process is complete

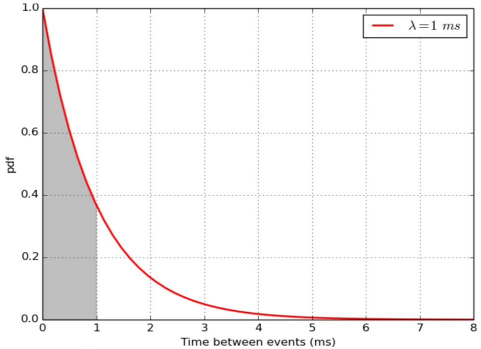
-> **non-paralysable**

DAQ and Trigger with busy logic

- **Busy logic** avoids triggers while the system is busy in processing
  - E.g.: AND port and a latch
- Latch (**flip-flop**):
  - a bistable circuit that changes state (Q) by signals applied to the control inputs (SET, CLEAR)



Deadtime



- Which (average) DAQ rate can we achieve now?
  - Reminder: w/ a clock trigger and τ = 1 ms the limit is 1 kHz
- Definitions
  - **f:** average rate of physics phenomenon (input)
  - **ν:** average rate of DAQ (output)
  - **τ: deadtime**, the time the system requires to process an event, without being able to handle other triggers
  - probabilities: P[busy] = ν τ;   P[free] = 1 - ν τ
- Therefore:

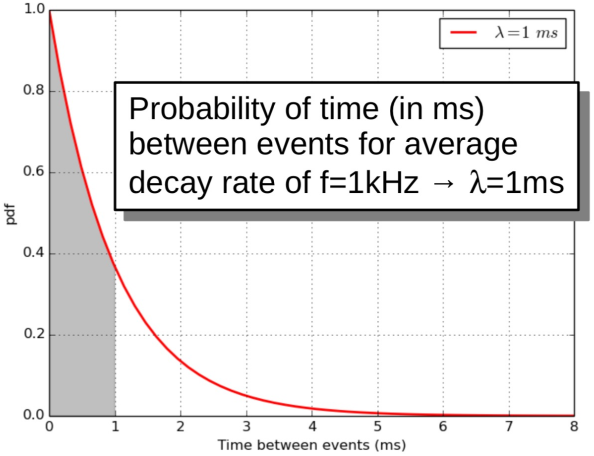
$$\nu = f\,P[free] \;\Rightarrow\; \nu = f(1-\nu\tau) \;\Rightarrow\; \nu = \frac{f}{1+f\tau}$$

Deadtime and Efficiency

- Due to stochastic fluctuations
  - DAQ rate always < physics rate $\nu = \dfrac{f}{1+f\,\tau} < f$
  - Efficiency always < 100% $\epsilon = \dfrac{N_{saved}}{N_{tot}} = \dfrac{1}{1+f\,\tau} < 100\,\%$

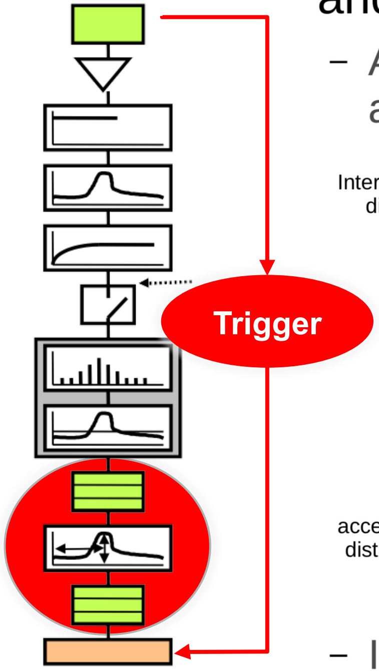
- So, in our specific example

$$\left|\begin{array}{l} f=1\,kHz \\ \tau=1\,ms \end{array}\right. \rightarrow \left|\begin{array}{l} \nu=500\,Hz \\ \epsilon=50\,\% \end{array}\right.$$

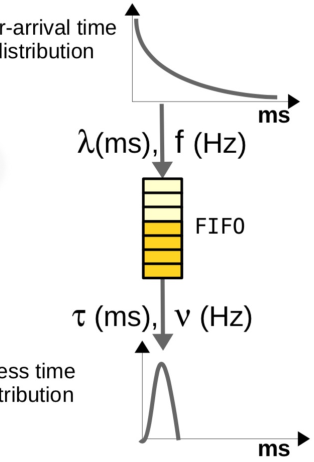


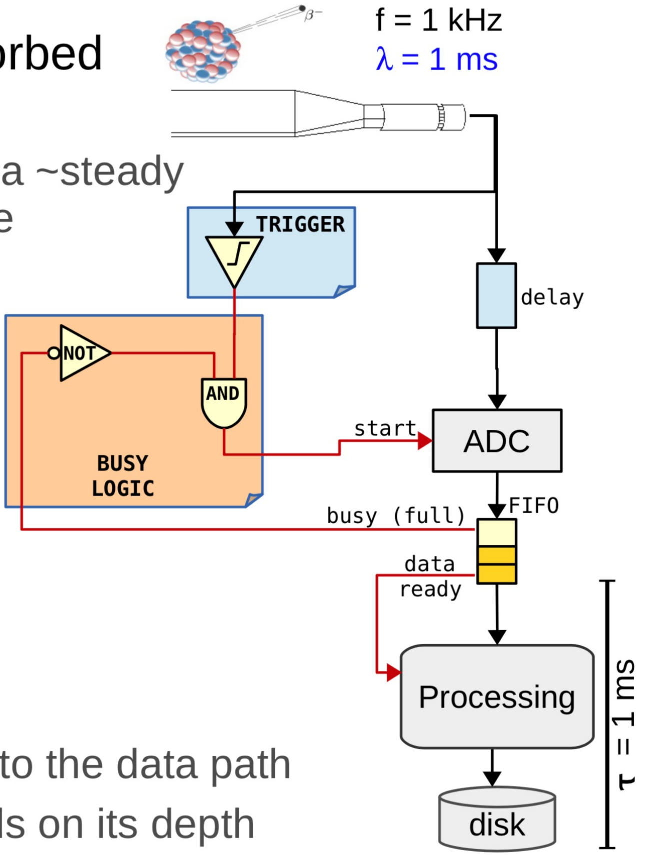
Probability of time (in ms) between events for average decay rate of f=1kHz → λ=1ms

Derandomization



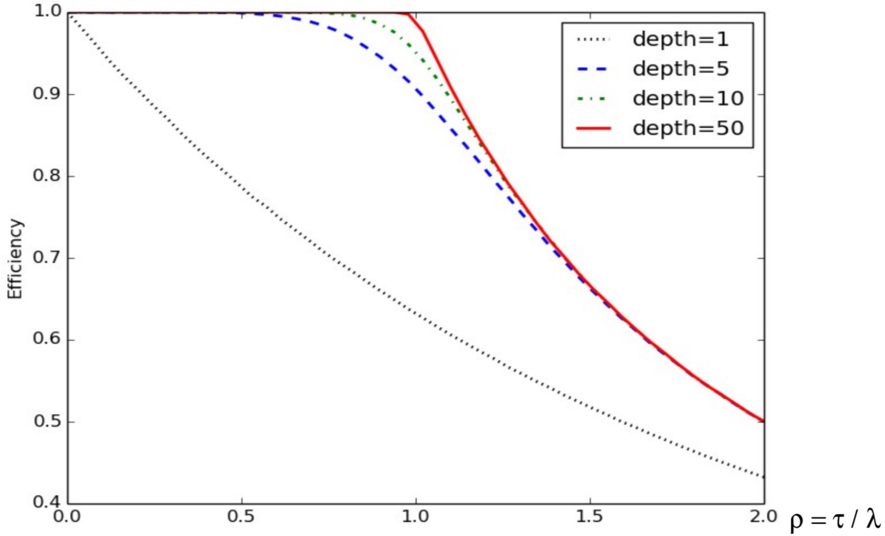
- Input fluctuations can be absorbed and smoothed by a queue
  - A First In First Out can provide a ~steady and **de-randomized** output rate

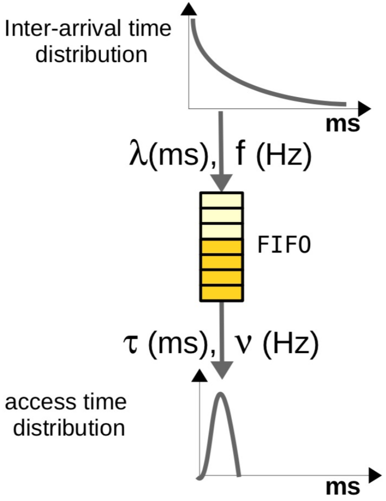




  - It introduces additional latency to the data path
  - The effect of the queue depends on its depth

A bit of queueing theory

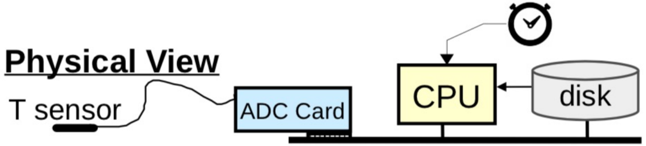




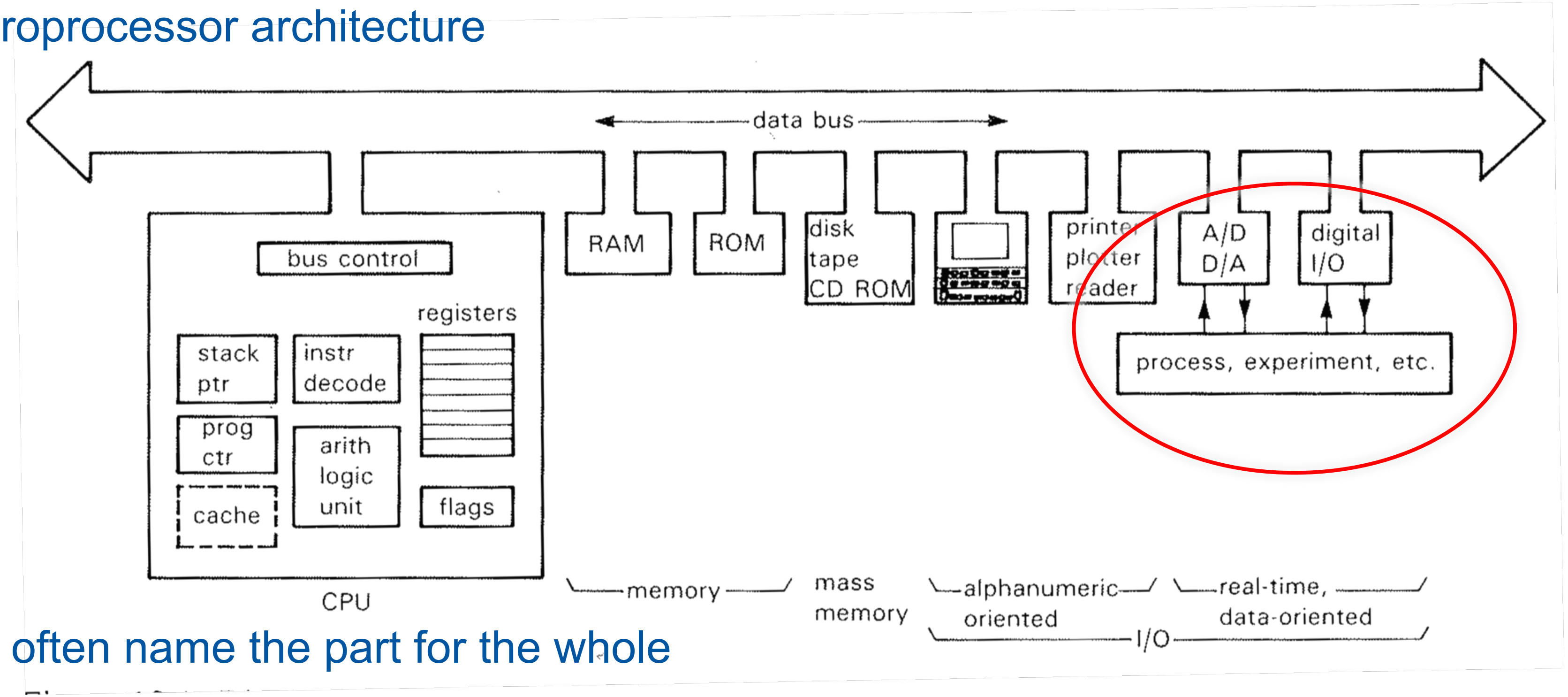
- Efficiency vs traffic intensity ($\rho = \tau / \lambda$) for different queue depths
  - **ρ > 1**: the system is overloaded ($\tau > \lambda$)
  - **ρ << 1**: the output is over-designed ($\tau << \lambda$)
  - **ρ ~ 1**: using a queue, high efficiency obtained even w/ moderate depth

CPUs, data buses and event building

The CPU does the rest…



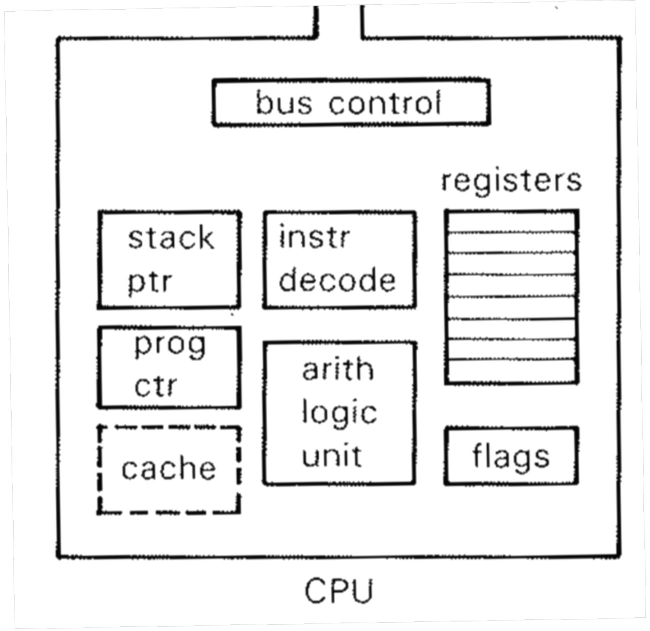
Microprocessor architecture



We often name the part for the whole

CPU (a simplified view)

- Does computation on **words**
  - Sequences of bits (32,64,128…)
- Fetches **instructions** from **memory** (the “program”)
- Instructions are bit codes corresponding to an operation (part of an “instruction set”)
  - They are first decoded and then fed to an **Arithmetic and Logical Unit** (ALU) that performs the operation on **data** contained in **registers** (e.g. add, complement, compare, shift, move…)
  - A **program counter** keeps track of the current location in the program being executed
- Data (as instructions) are fetched (usually) from **memory** over a **bus**
  - A **bus controller** handles the communication with memory and other I/O peripherals (such as a DAQ board, for example)

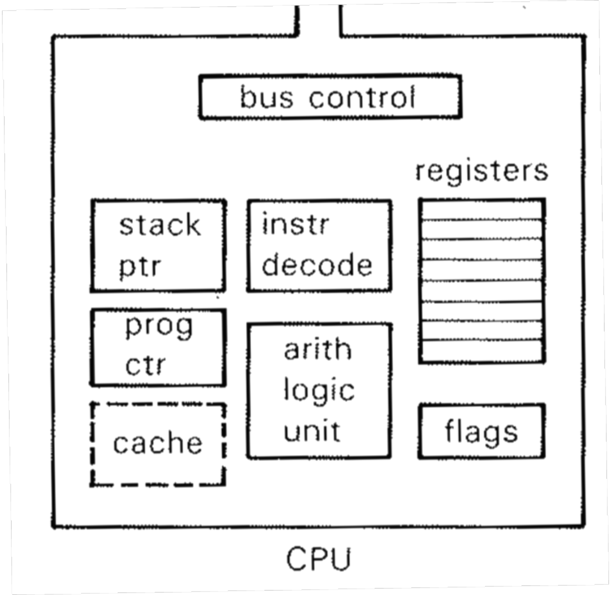


CPU (a simplified view)

- Modern CPUs have a more or less large “**cache**” – fast memory that contains recently or frequently accessed data for quick retrieval (not requiring access to the memory bus)

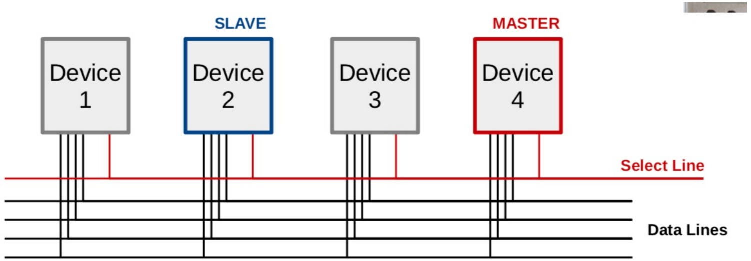
The **stack** is a portion of memory that works like a LIFO: there are two fundamental instructions PUSH and POP to move data to and from the stack, and a **stack pointer** always pointing at the top – more on this (maybe) later

This simplified view is common (give or take few parts) to many different architectures, **including those specific to DAQ** and trigger (the CPU could be just an “intelligent bus master”)

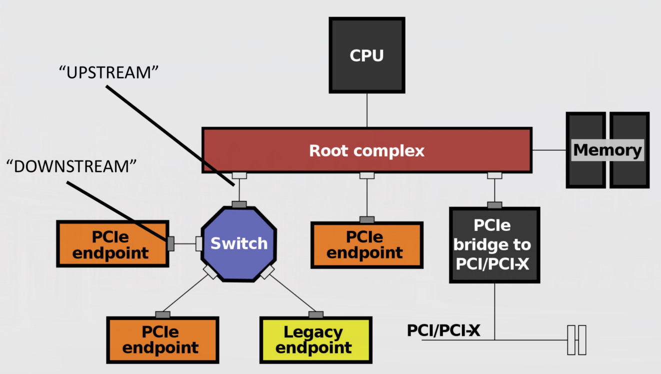


Back to DAQ

- In complex experiments, many channels are received by multiple electronic boards interconnected by a data **bus** – a computer may not be the most convenient form factor for this, we use **modular electronics**
  - In an architecture **similar to the one** discussed before – at least one particular element on the BUS is a CPU
  - It is common to **use the bus** (e.g. VME) to collect **data from multiple boards** in a single portion of memory



Parallel (e.g. VME)
Shared lines



Serial (e.g. PCIe)
Point-to-point connections

A note about parallel vs. serial

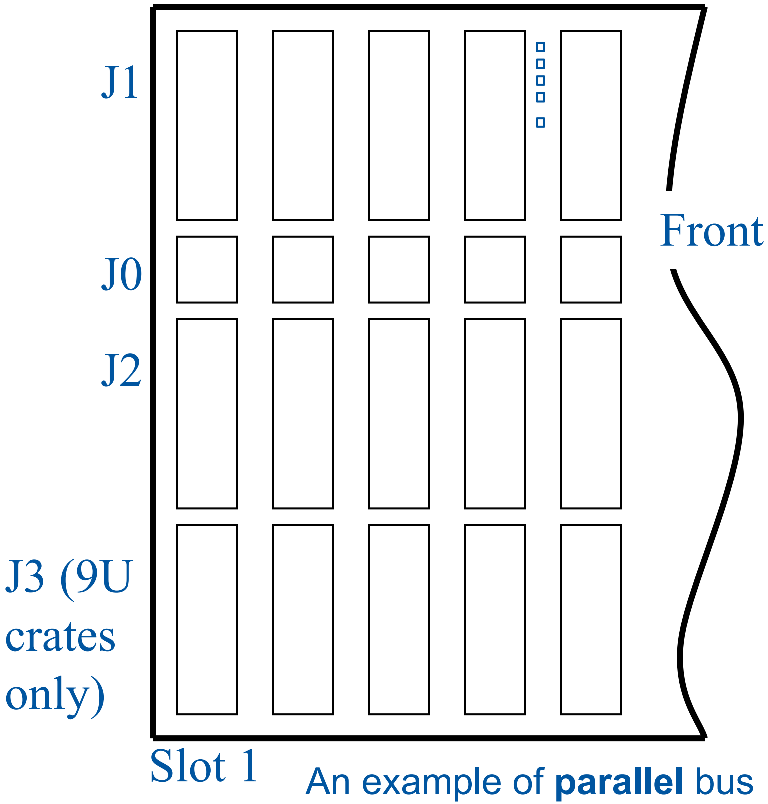
Parallel Buses Are Dead! (RT magazine, 2006)

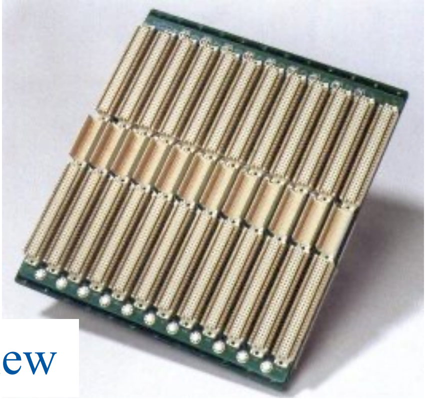
What is wrong about “parallel”?
- You need lots of pins on the chips and wires on the PCBs
- The skew between lines limits the maximum speed

What is wrong about “bus”?
- Speed is a function of the length (impedance) of the lines
- Communication is limited to one master/slave pair at a time (no scalability)
- The handshake may slow down the maximum speed
-

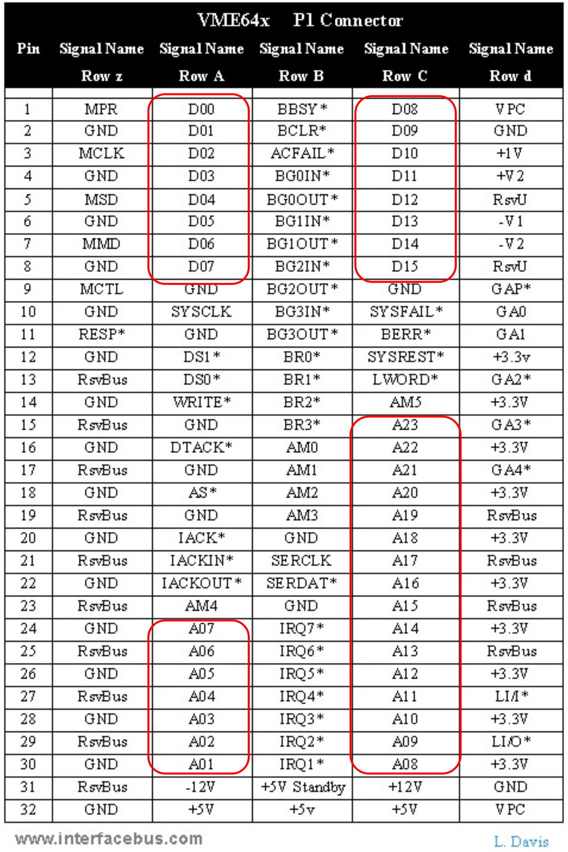
All parallel buses are dead. All? No!
- There is lots of legacy equipment
- VMEbus is still used heavily
  (military / research)

Parallel: VMEbus





Automatic 6U VME64x backplane



	VME64x	P1 Connector			
Pin	Signal Name Row z	Signal Name Row A	Signal Name Row B	Signal Name Row C	Signal Name Row d
1	MPR	D00	BBSY*	D08	VPC
2	GND	D01	BCLR*	D09	GND
3	MCLK	D02	ACFAIL*	D10	+1V
4	GND	D03	BG0IN*	D11	+V2
5	MSD	D04	BG0OUT*	D12	RsvU
6	GND	D05	BG1IN*	D13	-V1
7	MMD	D06	BG1OUT*	D14	-V2
8	GND	D07	BG2IN*	D15	RsvU
9	MCTL	GND	BG2OUT*	GND	GAP*
10	GND	SYSCLK	BG3IN*	SYSFAIL*	GA0
11	RESP*	GND	BG3OUT*	BERR*	GA1
12	GND	DS1*	BR0*	SYSREST*	+3.3v
13	RsvBus	DS0*	BR1*	LWORD*	GA2*
14	GND	WRITE*	BR2*	AM5	+3.3V
15	RsvBus	GND	BR3*	A23	GA3*
16	GND	DTACK*	AM0	A22	+3.3V
17	RsvBus	GND	AM1	A21	GA4*
18	GND	AS*	AM2	A20	+3.3V
19	RsvBus	GND	AM3	A19	RsvBus
20	GND	IACK*	GND	A18	+3.3V
21	RsvBus	IACKIN*	SERCLK	A17	RsvBus
22	GND	IACKOUT*	SERDAT*	A16	+3.3V
23	RsvBus	AM4	GND	A15	RsvBus
24	GND	A07	IRQ7*	A14	+3.3V
25	RsvBus	A06	IRQ6*	A13	RsvBus
26	GND	A05	IRQ5*	A12	+3.3V
27	RsvBus	A04	IRQ4*	A11	LI/I*
28	GND	A03	IRQ3*	A10	+3.3V
29	RsvBus	A02	IRQ2*	A09	LI/O*
30	GND	A01	IRQ1*	A08	+3.3V
31	RsvBus	-12V	+5V Standby	+12V	GND
32	GND	+5V	+5v	+5V	VPC

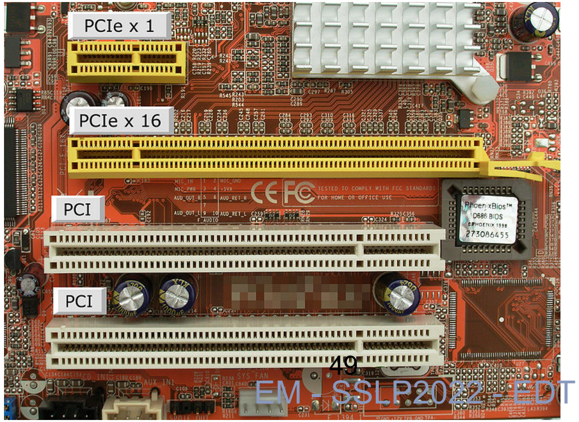
www.interfacebus.com L. Davis

An example of **parallel** bus

Besides modular electronics, it was the base of Sun-2 computer arch. in the 80s

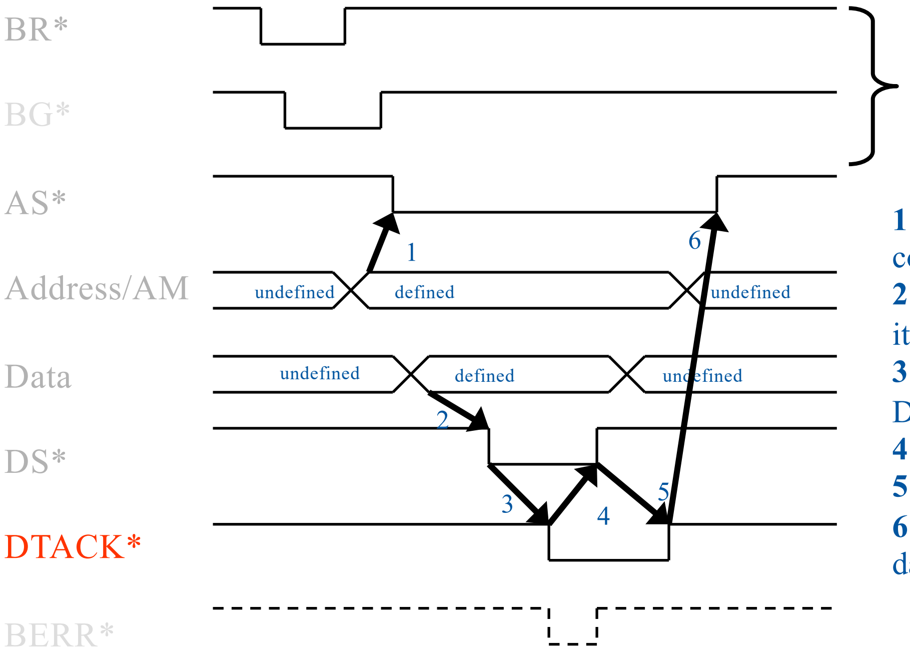
Serial: PCIe (aka PCI Express)

- Not a bus any more but a point-to-point link
- Data not transferred on parallel lines but on one or several serial lanes
  - Lane: One pair of LVDS lines per direction
  - Clock rate: 2.5 GHz (PCIe2.0: 5 GHz, PCIe 3.0: 8 GHz, PCIe 4.0: 16 GHz)
  - 8b/10b encoding (from PCIe3.0: 128/130b encoding)
  - 250 MB/s (PCIe 1.0) raw transfer rate per lane
  - Devices can support up to 32 lanes



Example: VME write

Example: (Simplified) write cycle



1: Master drives address and AM code. Then it asserts AS

2: Master puts data on the bus. Then it asserts DS

3: Slave latches data and drives DTACK

4: Master removes DS

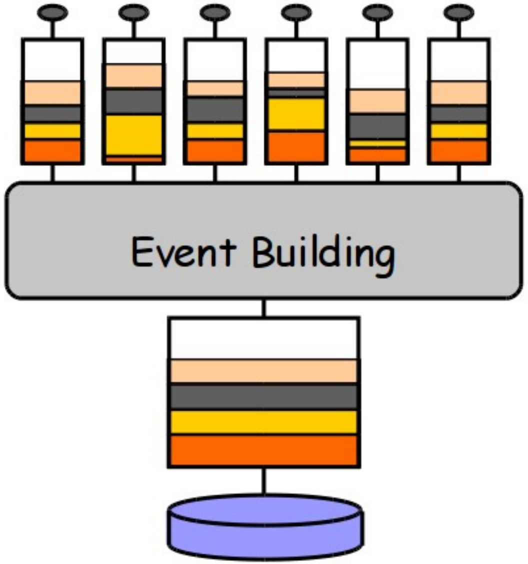
5: Slave removes DTACK

6: Master releases Address, AM and data lines. Then it releases AS

Color code: Master - Slave - Arbiter

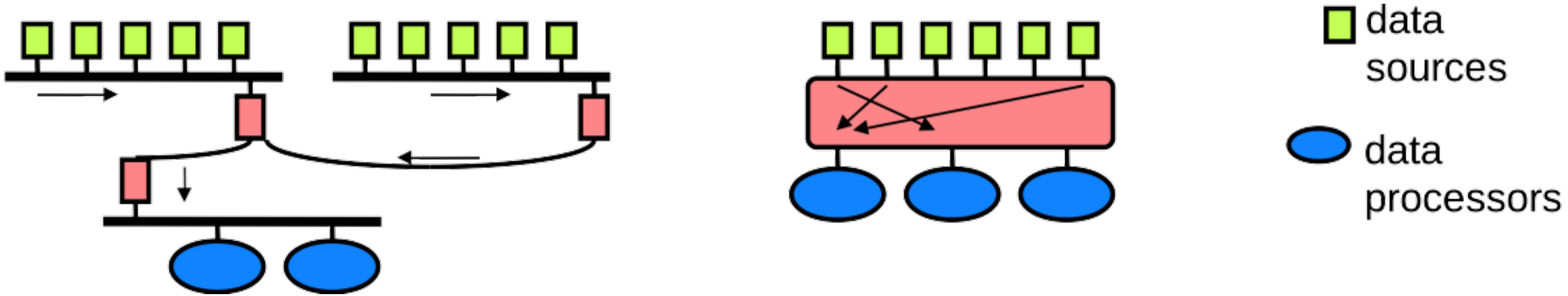
Event Building

- In large experiments consisting of many different sub-detectors, read out is performed by different boards and by multiple CPUs (for example in multiple VME crates)
  - We want to combine all the portions corresponding to the same “event” in the memory of a single CPU for processing and eventually storage
- Need a mechanism to associate all data corresponding to the same event (e.g. a bunch crossing in a collider, a gamma ray shower in a telescope array…)
  - This process is called **event building**



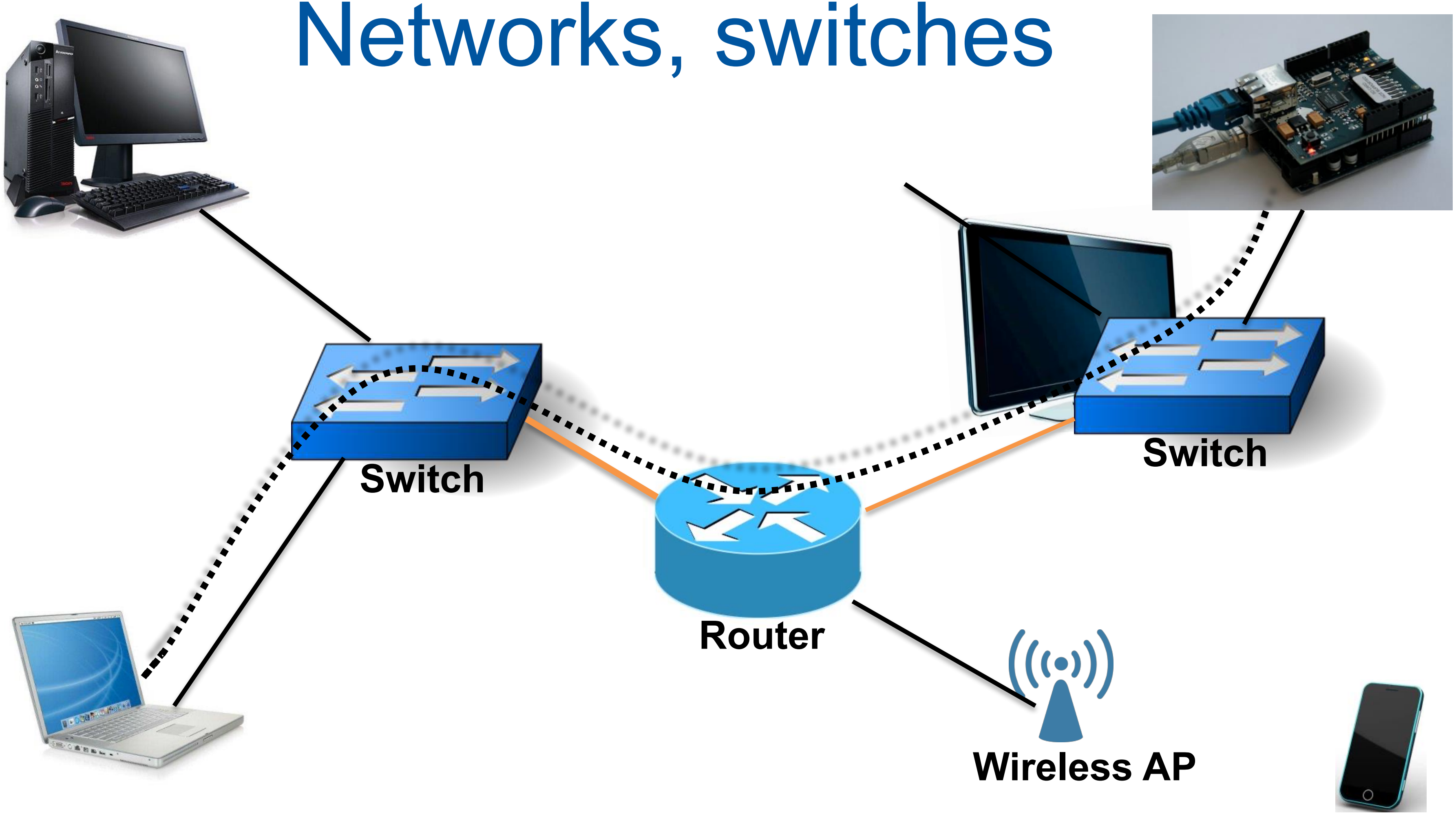
Event Building

- Event Building used to be performed on the bus itself



  - But the bus forces the process to be sequential (only one board can “speak” at a time)
  - It is also not infinitely extendible (does not “scale”)
- In all LHC experiments event building is performed by distributed processes through a **switched network**

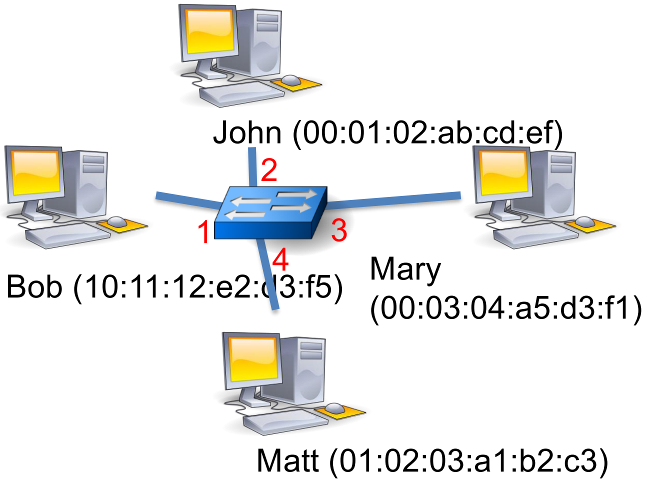
Networks, switches



Ethernet switch

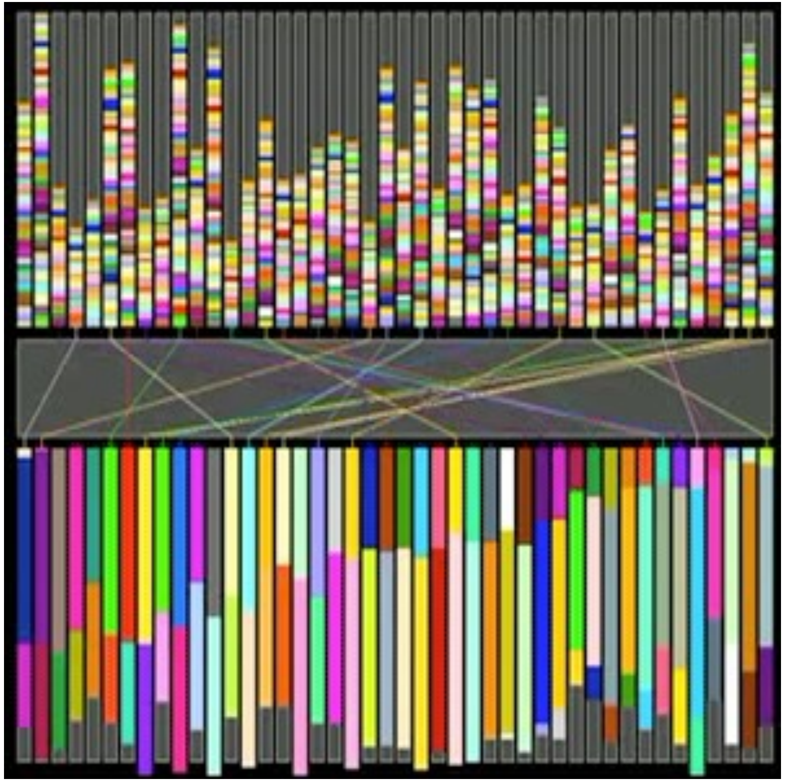
Layer-2 device

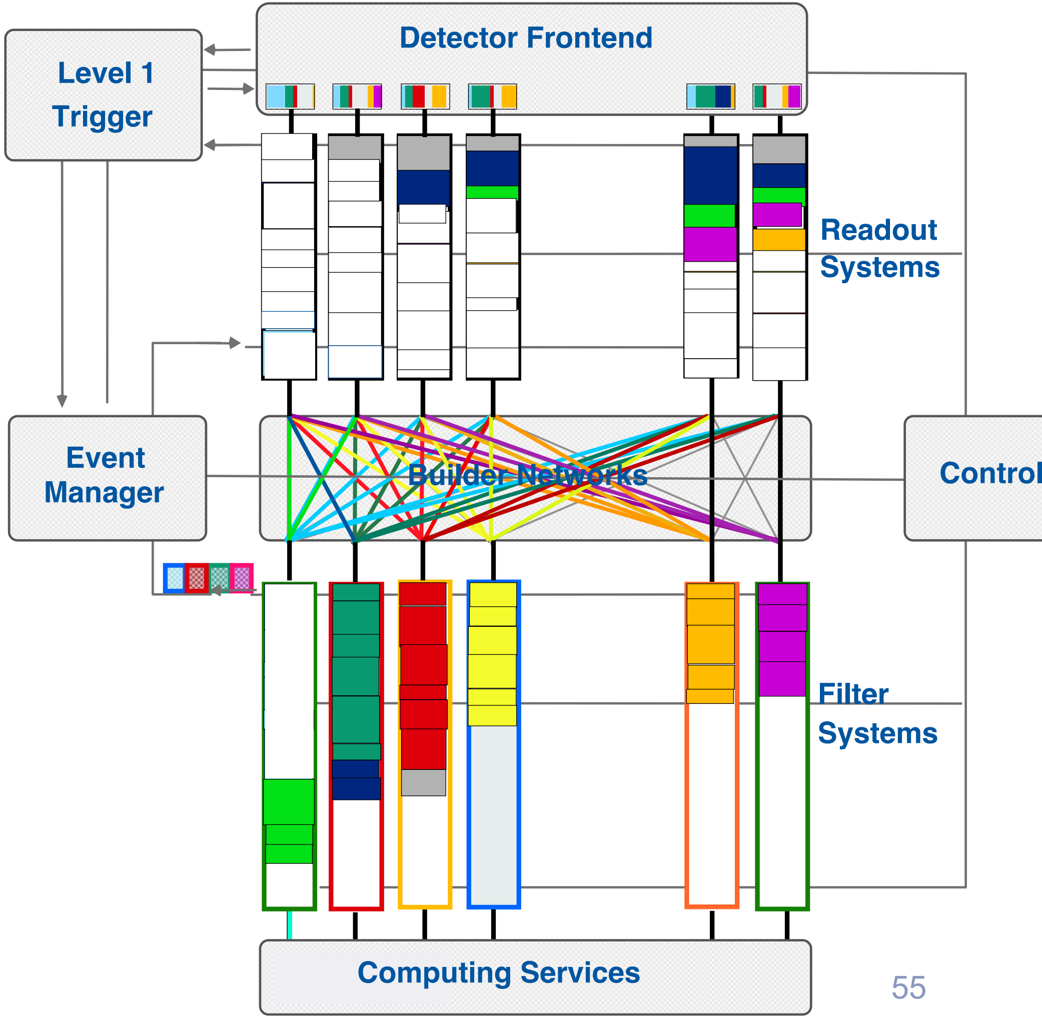
- Switches frames to their destination using the MAC address
- Learns the address associated to each port and stores it in a table



Port	MAC Address
2	00:01:02:ab:cd:ef(John)
4	01:02:03:a1:b2:c3(Matt)
….	…..

Event Building

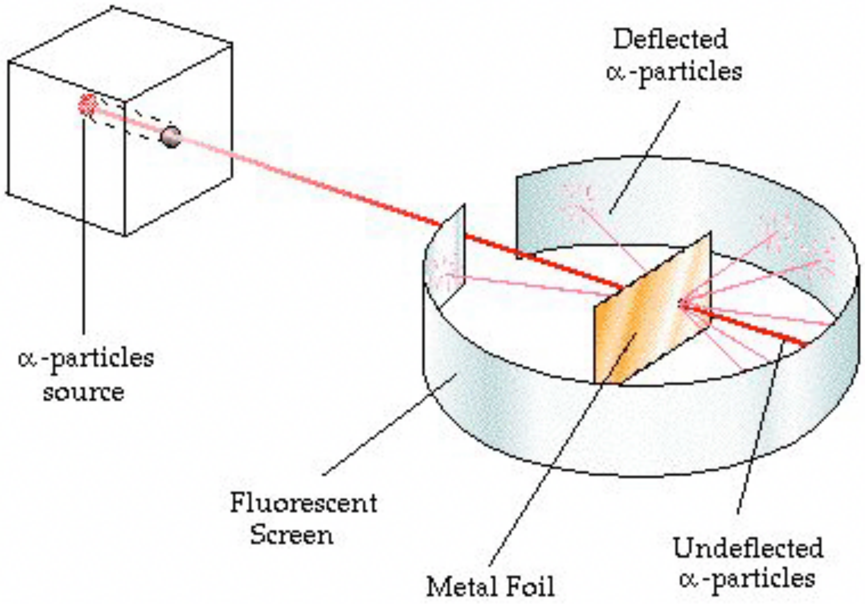




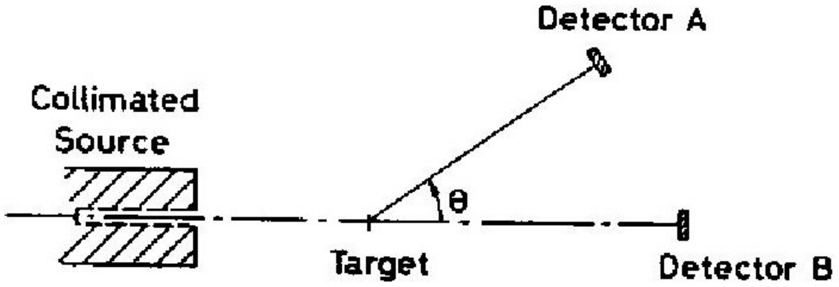
Questions?

Additional topics

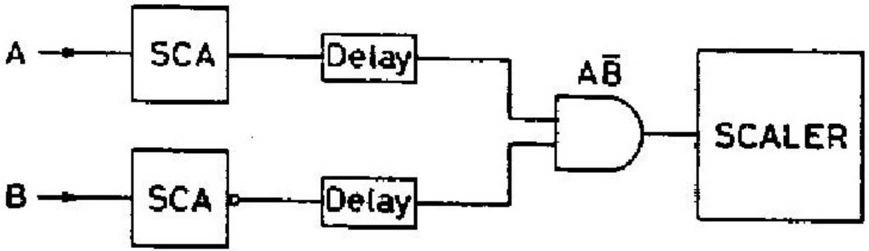
Modernized Geiger-Marsden Experiment

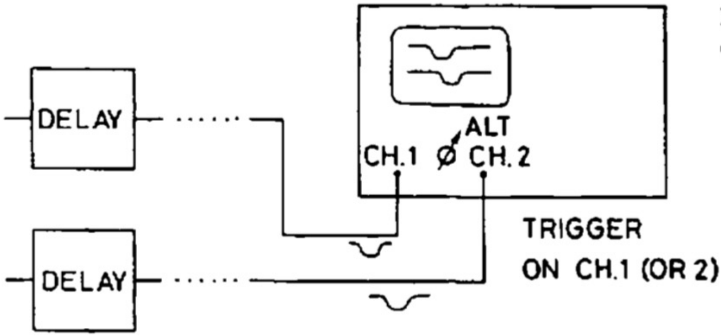


Modernized Geiger-Marsden Experiment

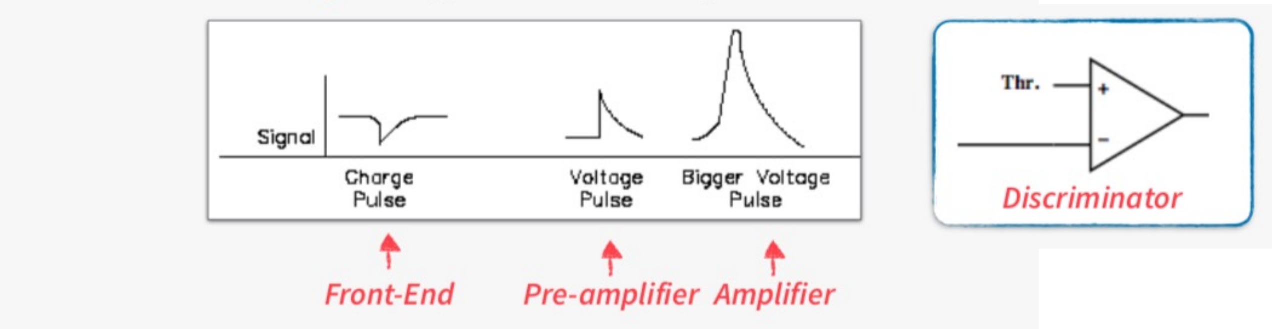


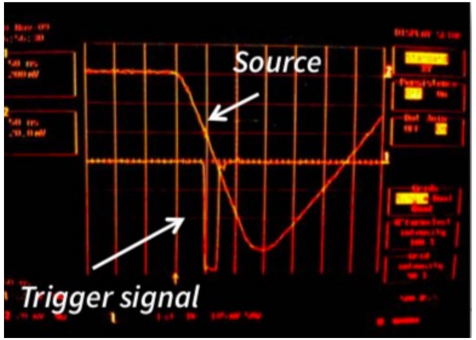
- Can you identify the elements in the diagram ?
- Scaler in nuclear instrumentation is another word for “counter”
- Counting pulses over some threshold (or in a certain window of amplitude) is a common need
- We will discuss counters in the context of digital logics
- The anticoincidence A not B is a form of trigger





Counting pulses: discriminator





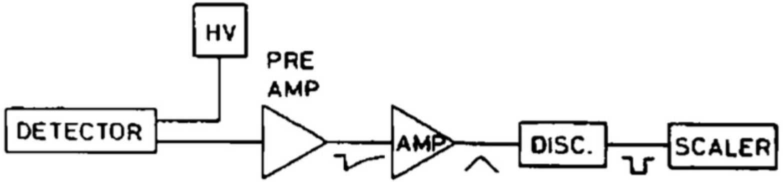
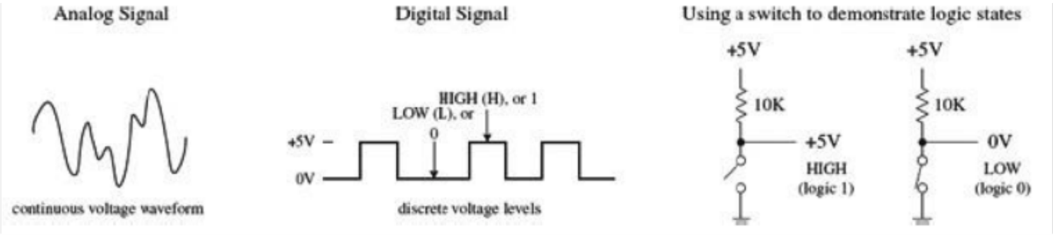


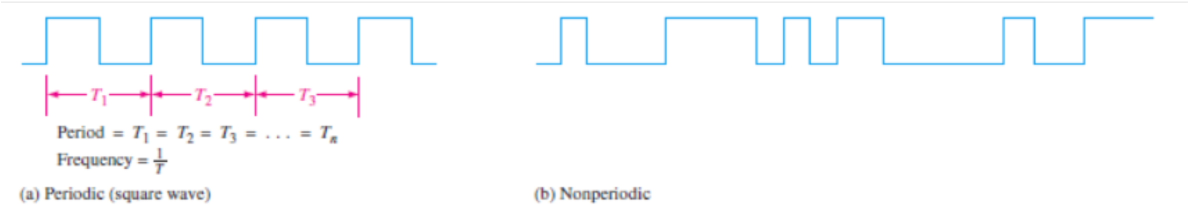
Fig. 15.1. A simple counting system

Combining signals from multiple discriminators may require compensating for the different length of the signal paths

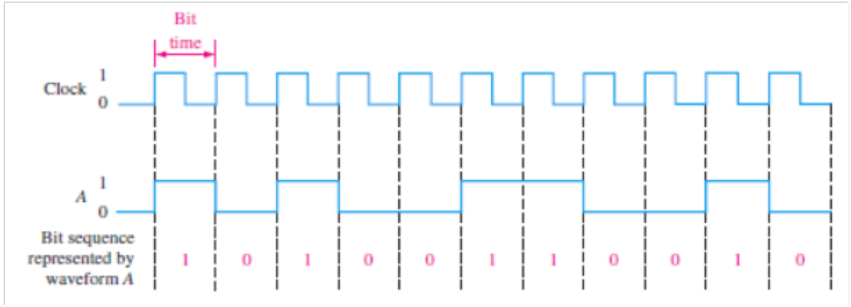
Possibly useful information

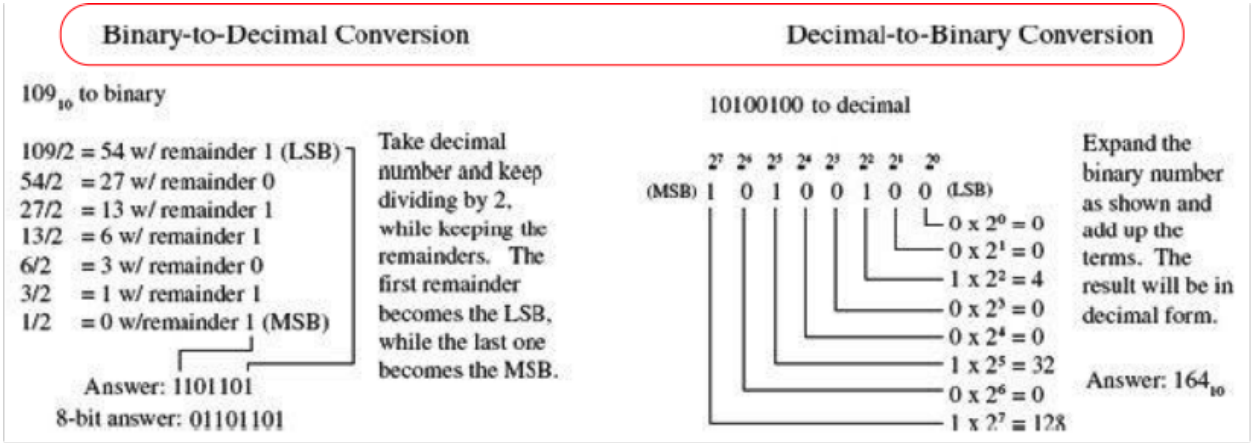
Analog and digital: basic concepts





T_1
T_2
T_3
Period = $T_1 = T_2 = T_3 = \ldots = T_n$
Frequency = $\frac{1}{T}$
(a) Periodic (square wave)
(b) Nonperiodic





Binary-to-Decimal Conversion

109_{10} to binary

109/2 = 54 w/ remainder 1 (LSB)
54/2 = 27 w/ remainder 0
27/2 = 13 w/ remainder 1
13/2 = 6 w/ remainder 1
6/2 = 3 w/ remainder 0
3/2 = 1 w/ remainder 1
1/2 = 0 w/remainder 1 (MSB)

Answer: 1101101
8-bit answer: 01101101

Take decimal number and keep dividing by 2, while keeping the remainders. The first remainder becomes the LSB, while the last one becomes the MSB.

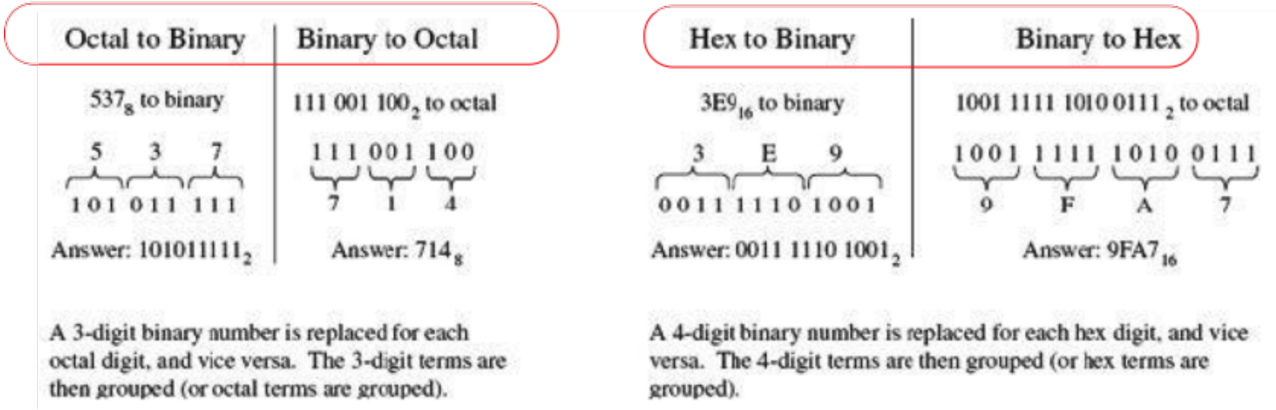
Decimal-to-Binary Conversion

10100100 to decimal

$0 \times 2^0 = 0$
$0 \times 2^1 = 0$
$1 \times 2^2 = 4$
$0 \times 2^3 = 0$
$0 \times 2^4 = 0$
$1 \times 2^5 = 32$
$0 \times 2^6 = 0$
$1 \times 2^7 = 128$

Expand the binary number as shown and add up the terms. The result will be in decimal form.

Answer: 164_{10}



Octal to Binary

537_8 to binary

Answer: 101011111_2

Binary to Octal

$111\ 001\ 100_2$ to octal

Answer: 714_8

A 3-digit binary number is replaced for each octal digit, and vice versa. The 3-digit terms are then grouped (or octal terms are grouped).

Hex to Binary

$3E9_{16}$ to binary

Answer: $0011\ 1110\ 1001_2$

Binary to Hex

$1001\ 1111\ 1010\ 0111_2$ to octal

Answer: $9FA7_{16}$

A 4-digit binary number is replaced for each hex digit, and vice versa. The 4-digit terms are then grouped (or hex terms are grouped).

Bibliography (partial)

Manuals:

W.R. Leo - Techniques for Nuclear and Particle Physics Experiments: A How-to Approach – Springer

P. Horowitz and W. Hill – The Art of Electronics – Cambridge University Press

Papers:

Kovaric A 1919 Physical Review **13** 272
Rossi, B. Method of Registering Multiple Simultaneous Impulses of Several Geiger's Counters. *Nature* **125,** 636 (1930)

Monographs:

V.P.Heuring and H.F.Jordan - Computer Systems Design and Architecture – Pearson

ADC: https://www.maximintegrated.com/en/design/technical-documents/tutorials/6/641.html