Trigger and Data Acquisition
at colliders

G. De Lentdecker
Université Libre de Bruxelles

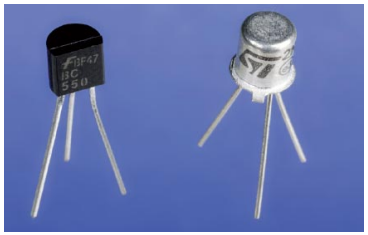
DIGITAL ELECTRONICS

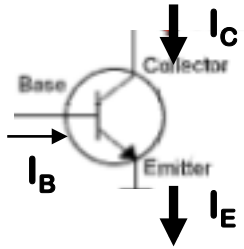
Reminder

- 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, analogue signals have to be converted to digital signals*
- We will start with some basic logical circuits and move towards much more complex systems

The basic logical gates

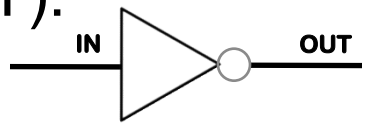
- Elementary piece: the transistor





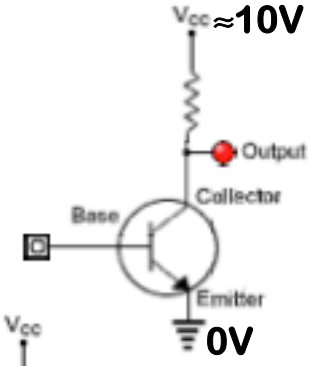
$I_C = I_B \cdot \beta$
$\beta \approx 100$
$I_C \approx I_E$

- Inversion (NOT):



IN	0	1
OUT	1	0

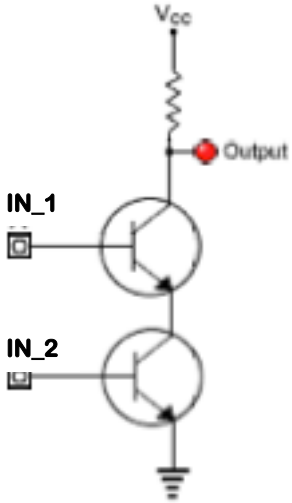
NOT circuit:



- NAND:

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

NAND circuit :

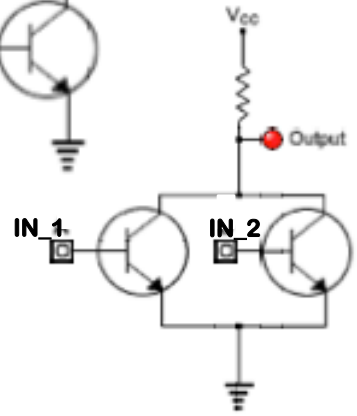


- AND = NAND + NOT

- NOR:

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

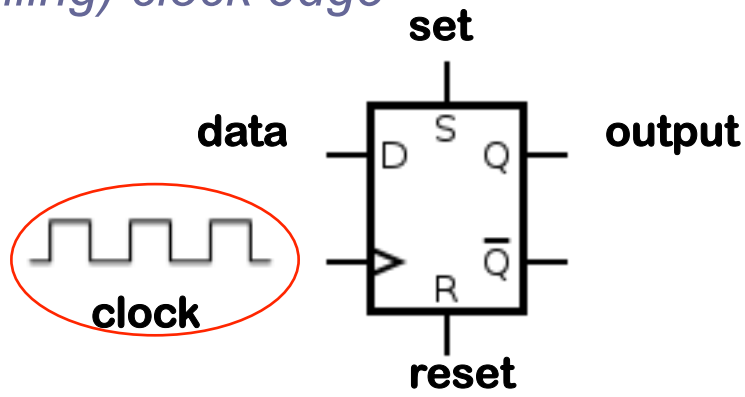
NOR circuit:

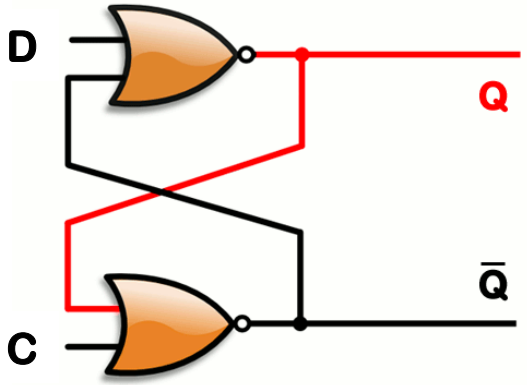


- OR = NOR + NOT
- From those you can build adders, multipliers, etc.

(synchronous) Sequential logic

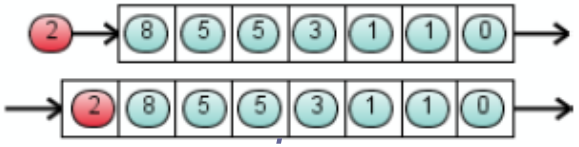
- Output are determined by the inputs and their history (sequence)
- Example: D Flip-flop
  - *Samples the data at the rising (or falling) edge of the clock*
  - *The output will equal the last sampled input until the next rising (or falling) clock edge*





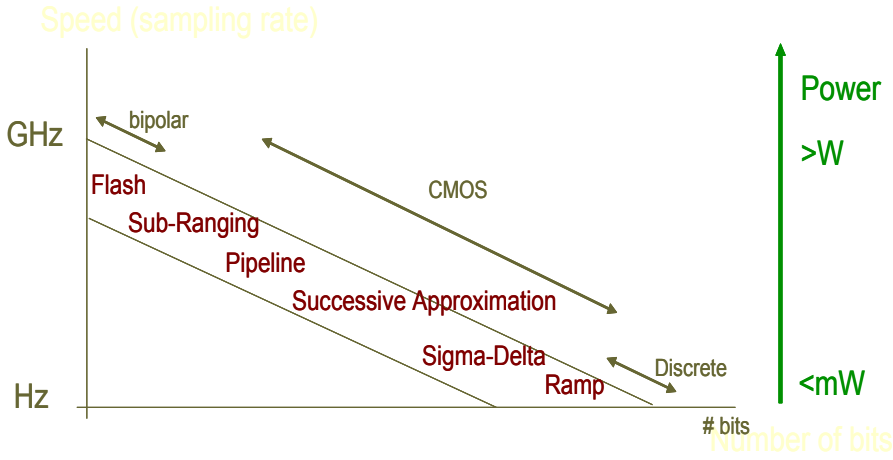
Memories (some definitions)

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



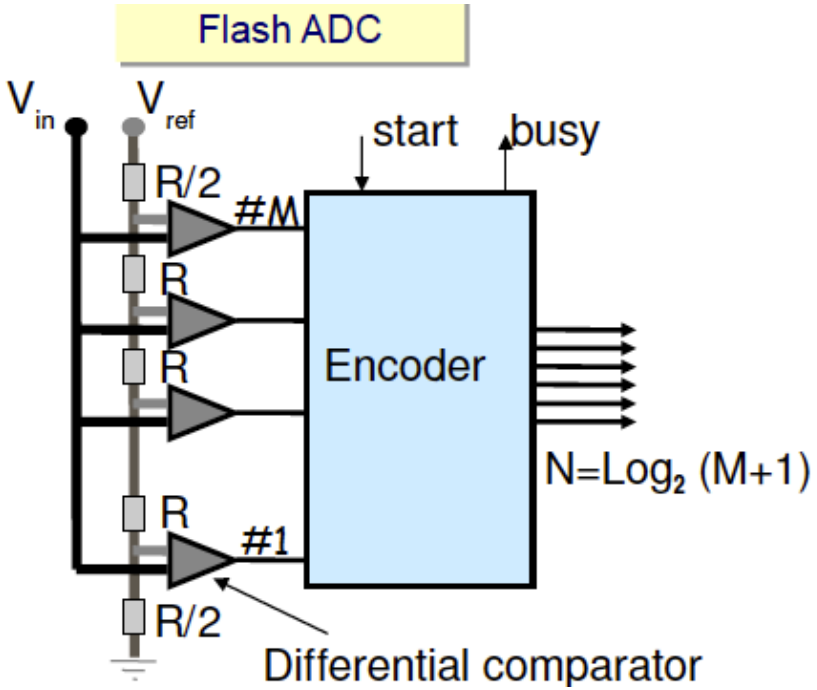
Analog to Digital Converter (ADC)

- The ADC is a device which converts the information contained in an analogue 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 analogue 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.*



➔Example M=3 → N=2

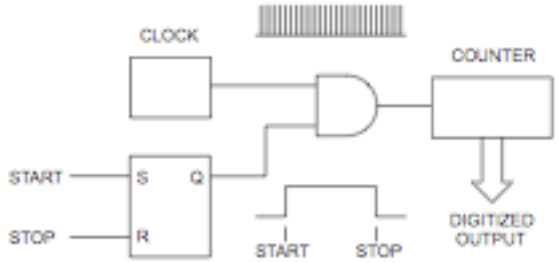
V_{in}/V_{ref}	Comparison results	Encoded form
<1/6	000	00
1/6≤ <3/6	001	01
3/6≤ <5/6	011	10
5/6≤	111	11

- 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)

- Example: Counter



  - The clock pulses are counted between the START and the STOP signals, which yields a 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)

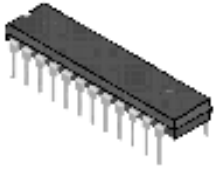
Some other logical functions

- 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.*

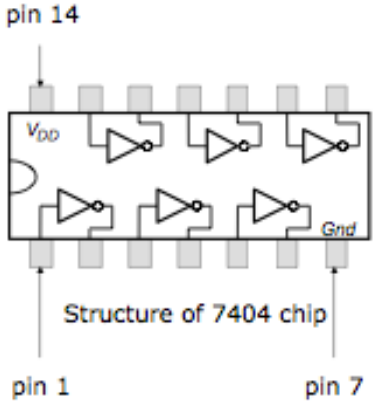
- But sometimes you want to perform more sophisticated tasks than buffering, +, -, ×, ÷, …
- For those you need digital processors : DSP, FPGAs, etc.

Field Programmable Gate Arrays (FPGA)

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



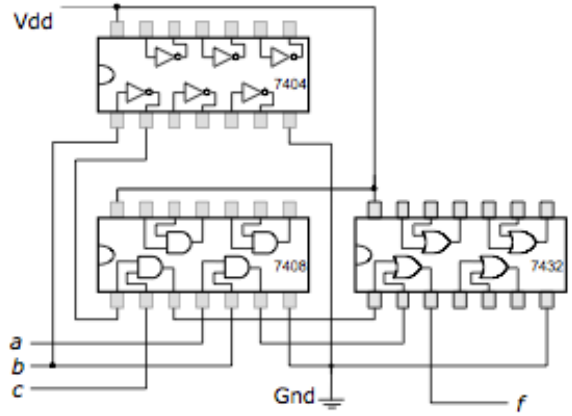
Dual-inline package



- *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, Verilog), synthesized using design libraries, and implemented as custom Integrated Circuit (ASIC) or programmable gate arrays (FPGA).*

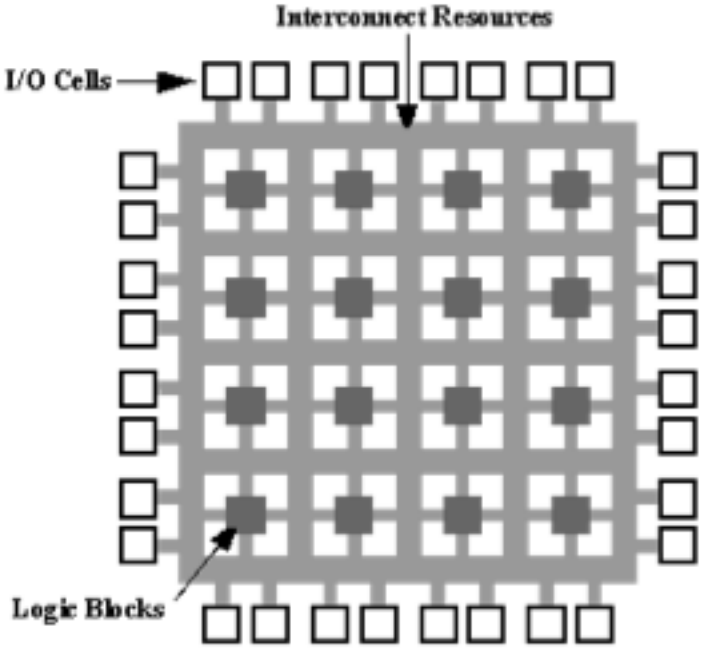
Implementation of f=ab+b'c



- *The advantage of FPGA over the ASIC, is that the logic can be re-programmed at any time, while the ASIC is made once and for good.*

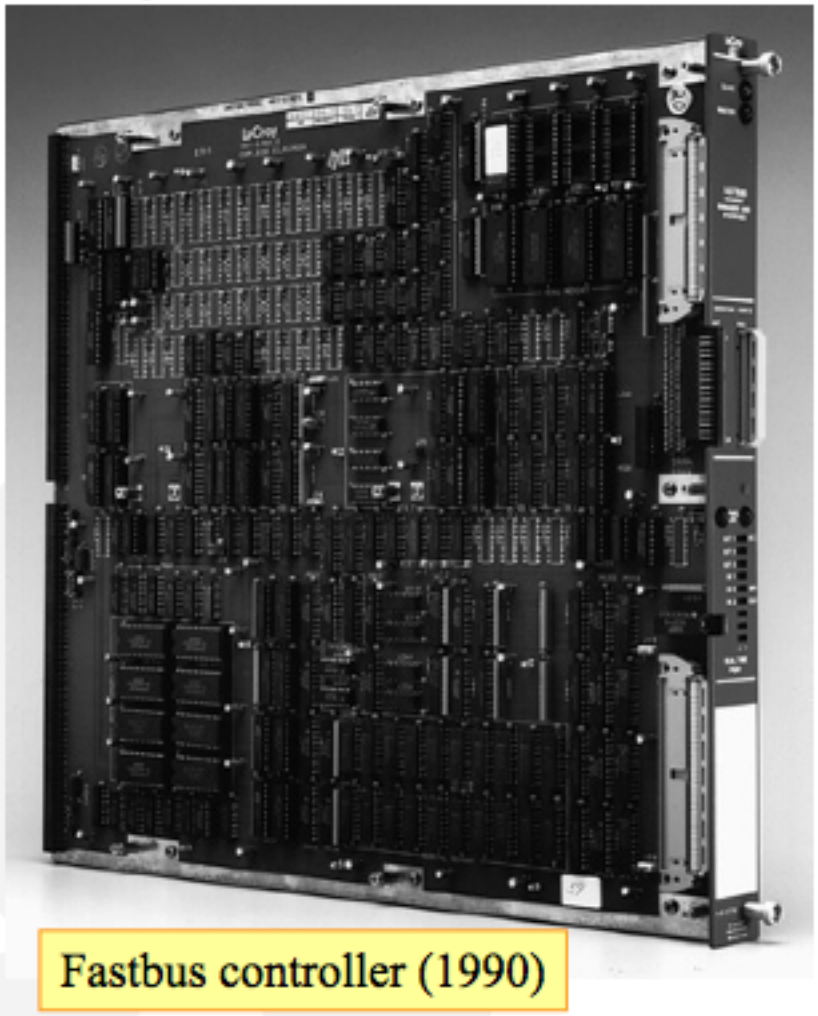
Field Programmable Gate Arrays (FPGA)

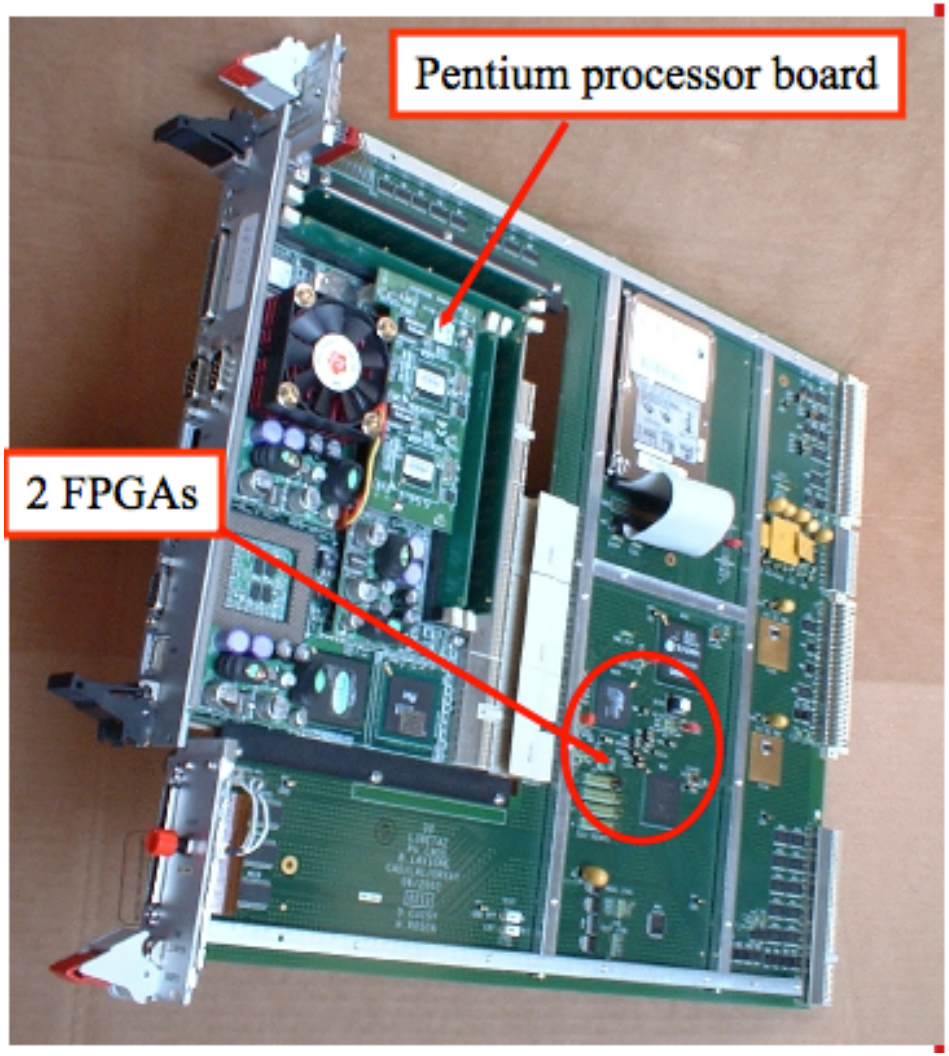
- *FPGAs are built from one basic "logic-cell", duplicated ~O(10^5-10^6) of time.*
- *Each logic-cell can be connected to other logic-cells through interconnect resources*
- *Each cell can do little, but with lots of them connected together, complex logic functions can be created.*
- *Typical clock freq: 50-100 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:



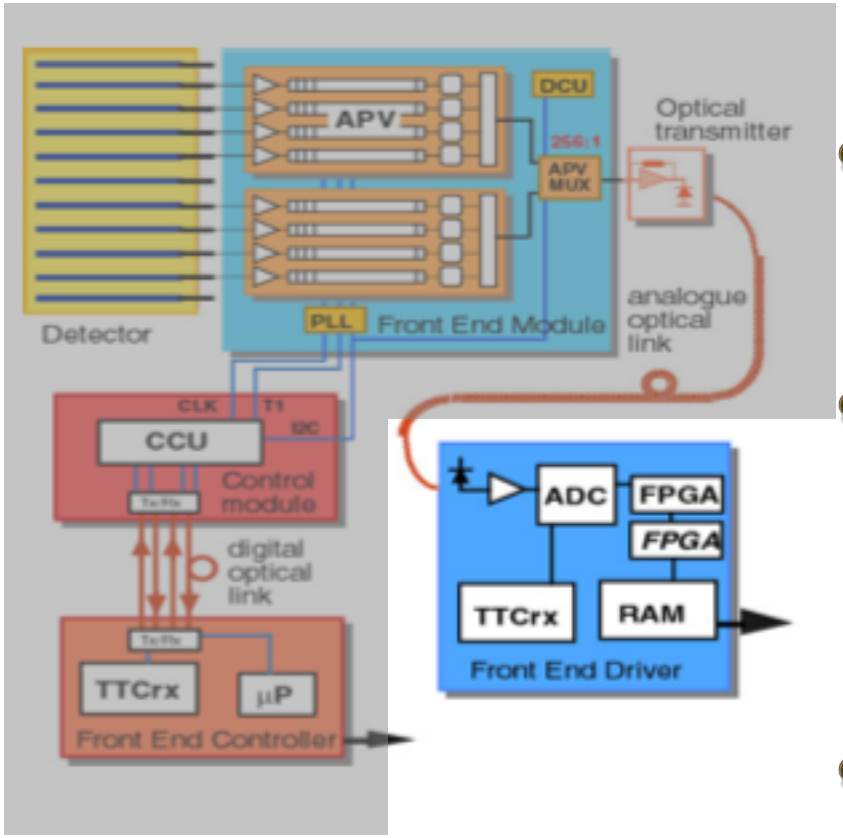


Other Digital processors

- 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

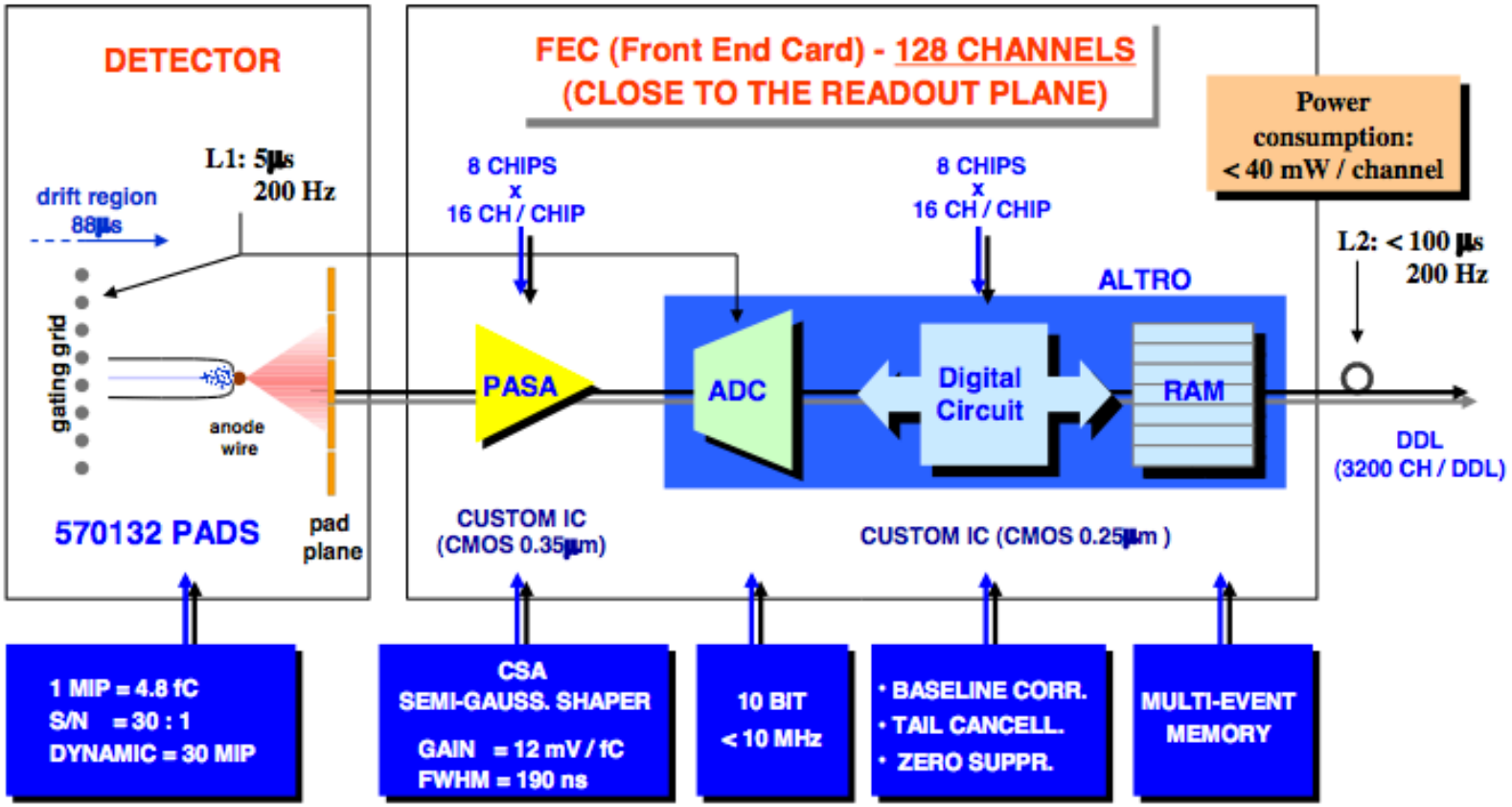
- CMS Si Strip Tracker FE:

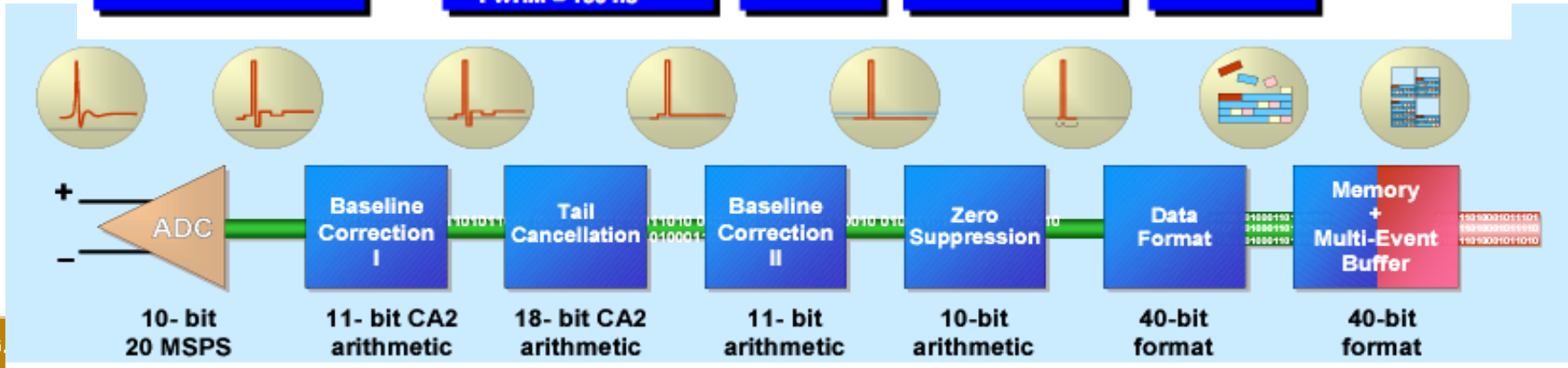


- Front End Driver (FED) module housed in the underground counting room outside the cavern.
- Pulse height data from the front-end chips are converted back to electrical levels matched to the range of a 10bit ADC.
- The FED digitizes the data, performs some signal processing (FPGA), including reordering 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.

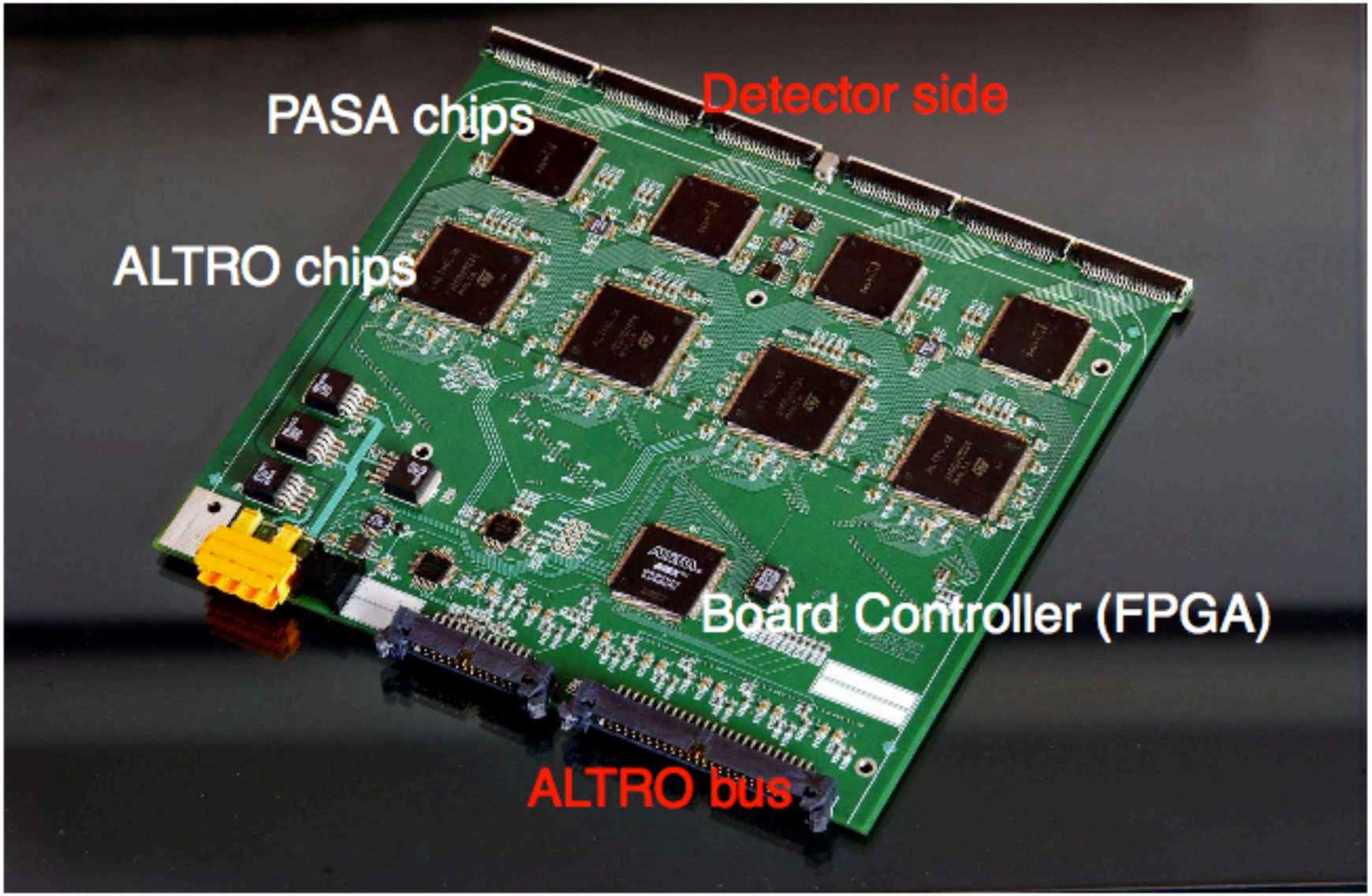
Examples of Readout architecture

- ALICE TPC Readout





ALICE TPC Readout

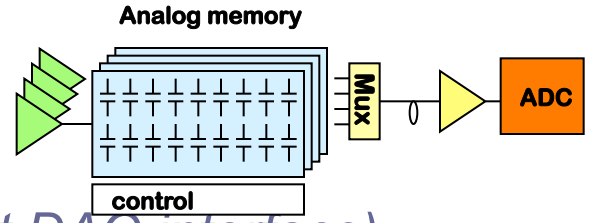


+ 4 PASA chips on rear side

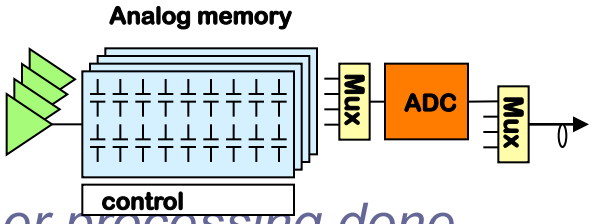
BACK-UP

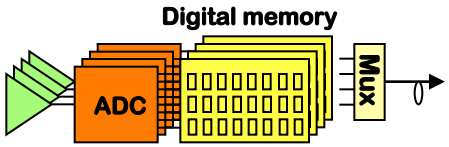
Analog to Digital/Binary Conversion

- After amplification the analogue 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*

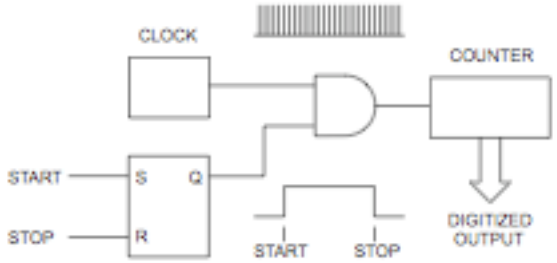




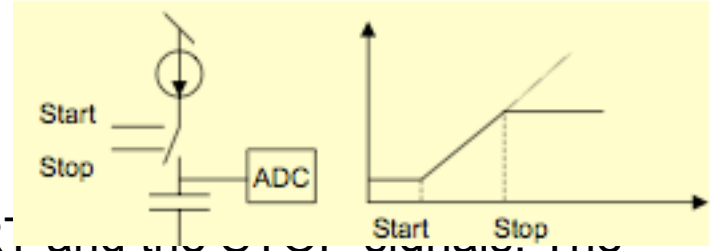
- We will now review ADC, TDC, ...

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



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.*