Architectures for new pixel chip/cells

A. Rivetti INFN, Sezione di Torino

Step1: what is on the market

FE-I4 chip for ATLAS

Mohsine Menouni, Marseille group

on behalf of the ATLAS PIXEL Upgrade FE-I4 collaboration

8th International Meeting on Front-End Electronics

24-27 May 2011, University of Bergamo

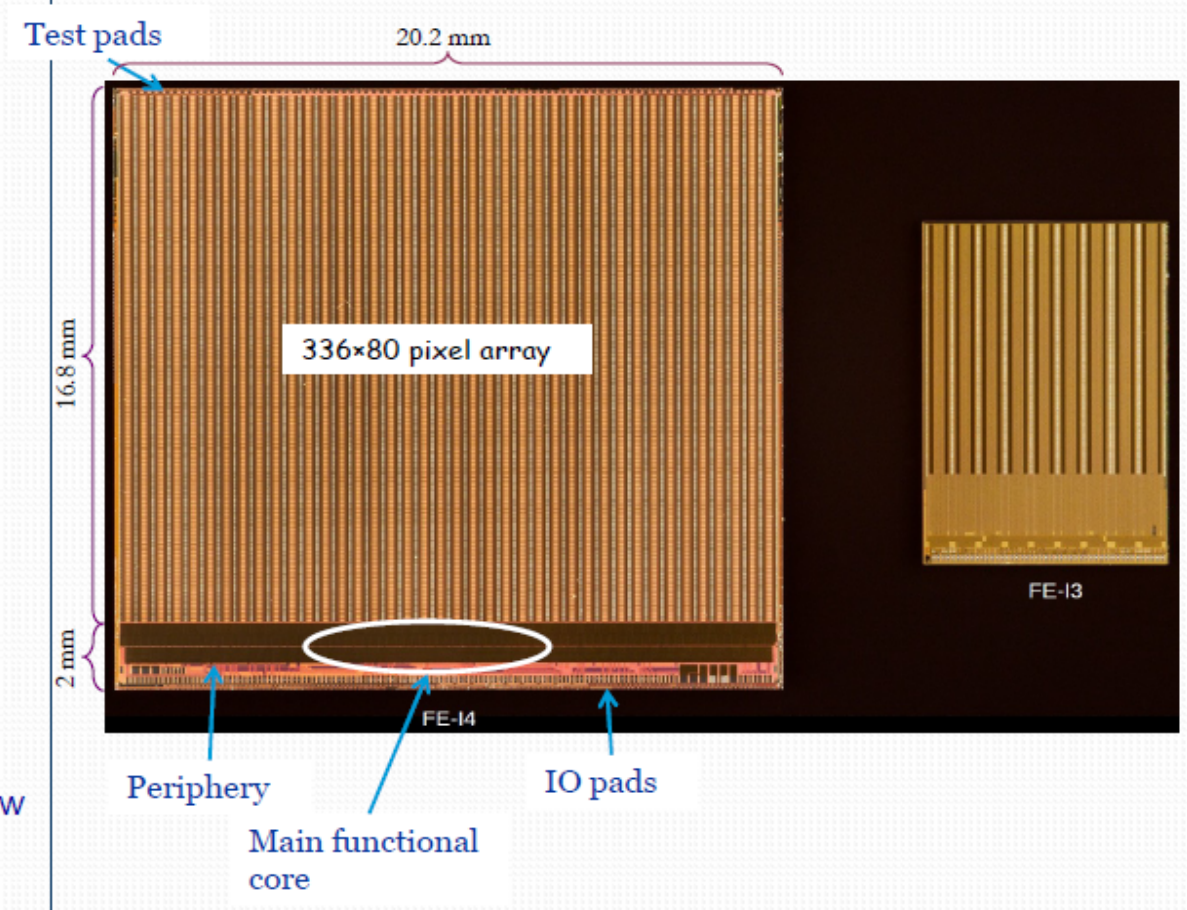
FE-I4 Specs

Pixel size	50 x 250	µm²
Pixel array size	80 x 336	Col x Row
Normal pixel input capacitance range	100-500	fF
Edge pixels input capacitance range	150-700	fF
In-time threshold with 20ns gate (400 fF)	4000	e-
Single channel ENC sigma (400 fF)	<300	e-
Tuned threshold dispersion	<100	e-
Charge resolution (ToT method)	4	bits
Operating temperature range	-40 to +60	°C
Radiation tolerance	300	MRad
DC leakage current tolerance	100	nA
Single serial command input (nominal)	40	Mb/s
Single serial data output (nominal)	160	Mb/s
Output data encoding	8b/10b	

8th International Meeting on Front-End Electronics FEE 2011, Bergamo, Italy

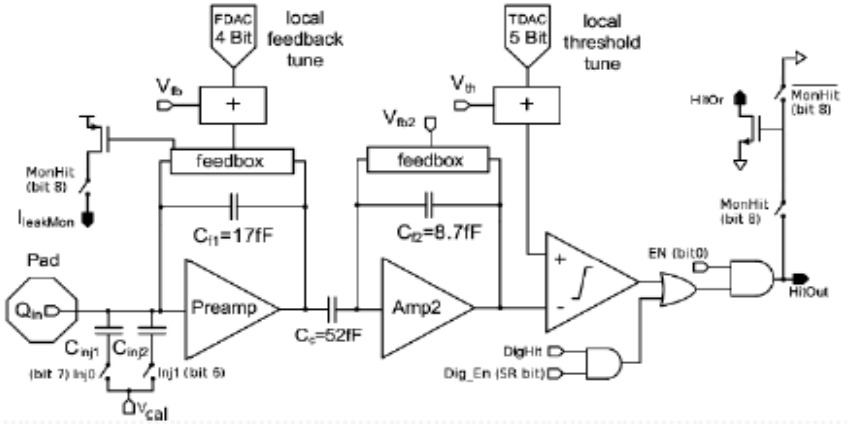
FE-I4A chip overview

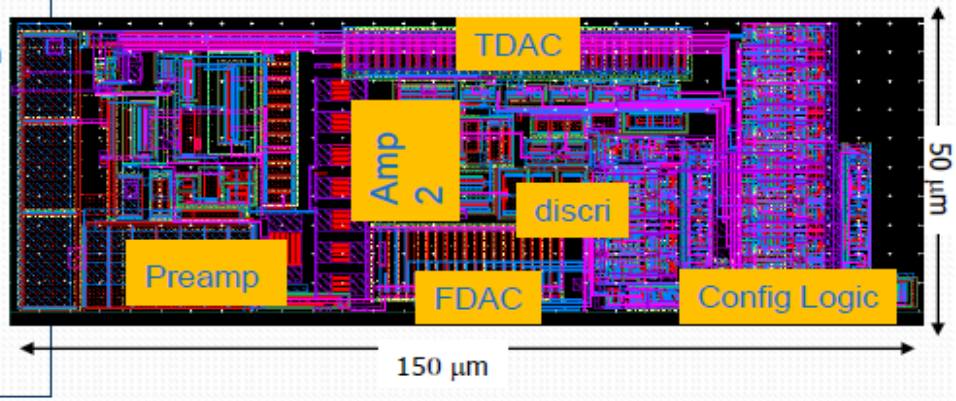
- The FE-I4A chip is a full scale prototype and is designed in a 130 nm CMOS process
- All transistors are linear, but not all conventional :
  - Extra guard rings are used
- There are wire bonding pads along the bottom and the top of the chip
  - Pads at the top are just for characterization
- Submitted July 1, 2010
- Wafer ship date September 14th
- The chip and hybrid modules still now under test and characterization



Analog Pixel

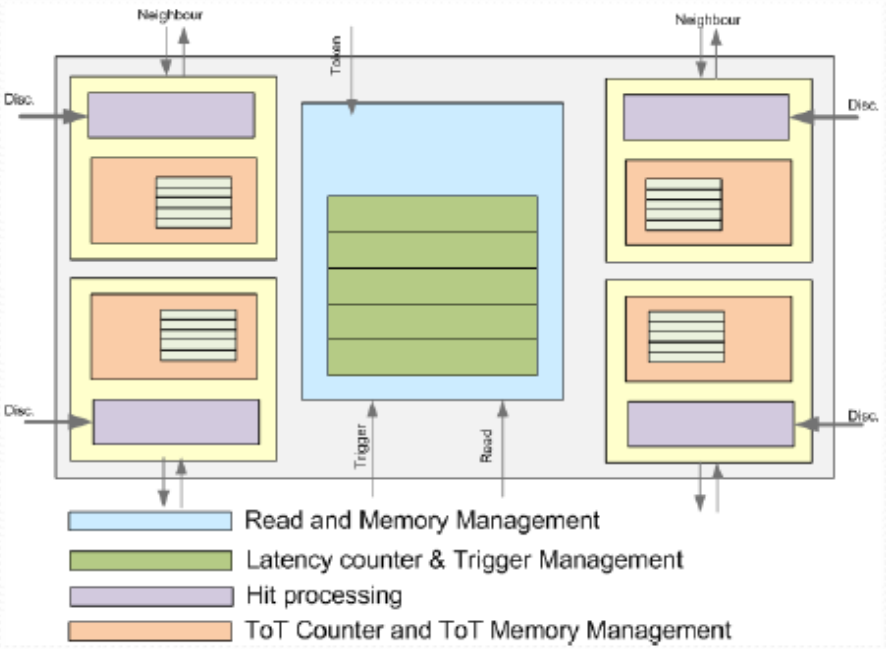
- Two-stage Amplifier configuration
  - Optimized for low power, low noise and fast rise time
- The second stage Amp2 is AC coupled to the preamplifier
  - Additional gain Cc/Cf2~ 6
  - decoupled from preamplifier DC potential shift caused by leakage
- The main motivation on the 2 stage structure is to provide a High gain
  - More flexibility on the choice of Cf1
  - The charge collection less dependent on the detector capacitor
- Local DACs for tuning feedback current and global threshold
- Charge injection circuitry for testing and characterization
- 13 bits for pixel configuration:
  - 4 FDAC: tuning feedback current
  - 5 TDAC: tuning of discriminator threshold
  - 2 Local charge injection circuitry
  - 1 Hit Enable
  - 1 HitBus / IleakMonitor





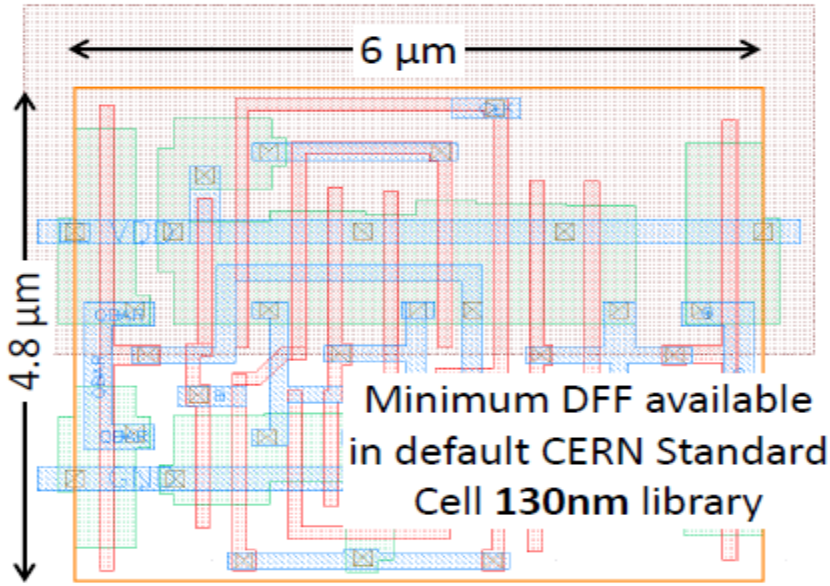
Pixel Digital Region

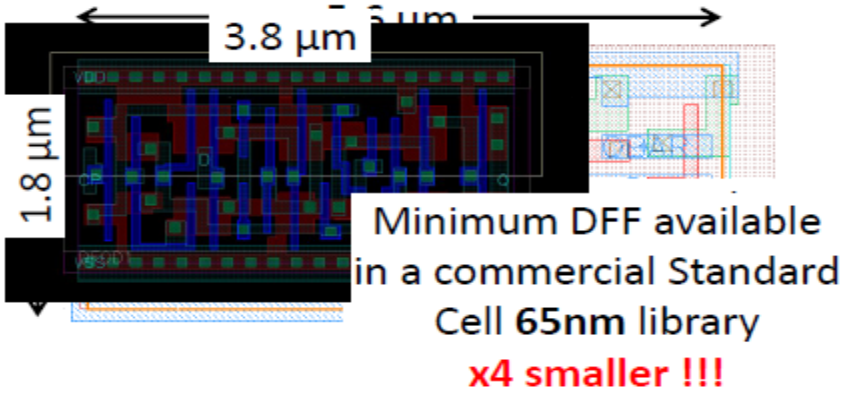
- In the present chip (FEI3) :
  - Each pixel is logically independent inside the DC
  - All hit pixels are shipped to End of Column buffer
  - A hit pixel need to transfer its data to EoC before accepting new hit : Congestion Problem for a high hit rate
- For the FEI4 chip
  - Basic idea is to store the hit locally until L1T
  - Implementation of local buffers is possible because of the smaller feature size (130 nm)
  - Organized on PDR : Pixel Digital Region
- A PDR processes the data from 4 pixel discriminator
  - Store locally up to 5 events/hits (different BC)
  - Small/big hit discrimination (3 programmable modes)
  - 2 BC association for small hit
  - 4 bit ToT (small hit, no hit, long hit, 13 x value)
  - Neighbor logic (1 bit)
  - Records up to 16 consecutive triggers (4 bit)
  - Programmable latency max. 257 BC
  - Token type readout



SC_130nm_XL Library

- Row Height is fixed to 2.4 µm
- Well Tap library
- Maximum frequency <~350 MHz
- Low power transistors
- ~45 cells are available in the library (growing fast…)
  - No SEU registers (so far)
- Encounter Library Characterizer (ELC) has been used to fully characterize the library in different corners
  - Full Synopsis library
  - Verilog library
  - LEF files





Step 2: what can we do?

Basic considerations:

- If we come to the conclusion that the best is to do like ATLAS, just use FE-I4 (maybe with some adaptation...)

- FEI-4 designed by a large community: many different architectures considered in the preliminary simulations: not trivial to do better.

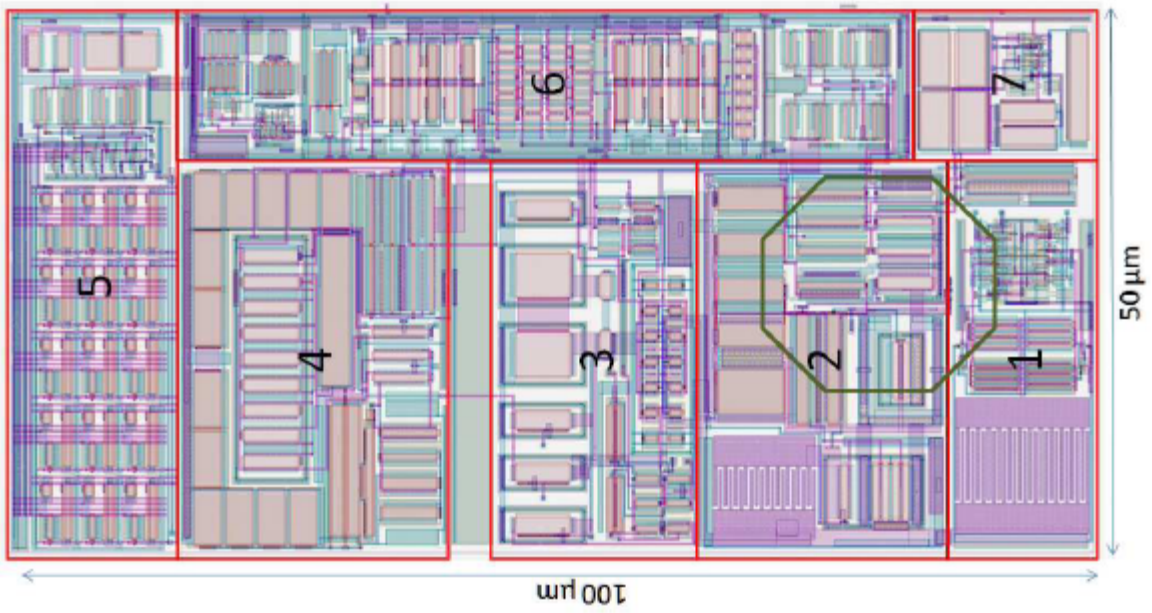
- A complete new chip only motivated if looks reasonably different and already targets phase 2 specs.

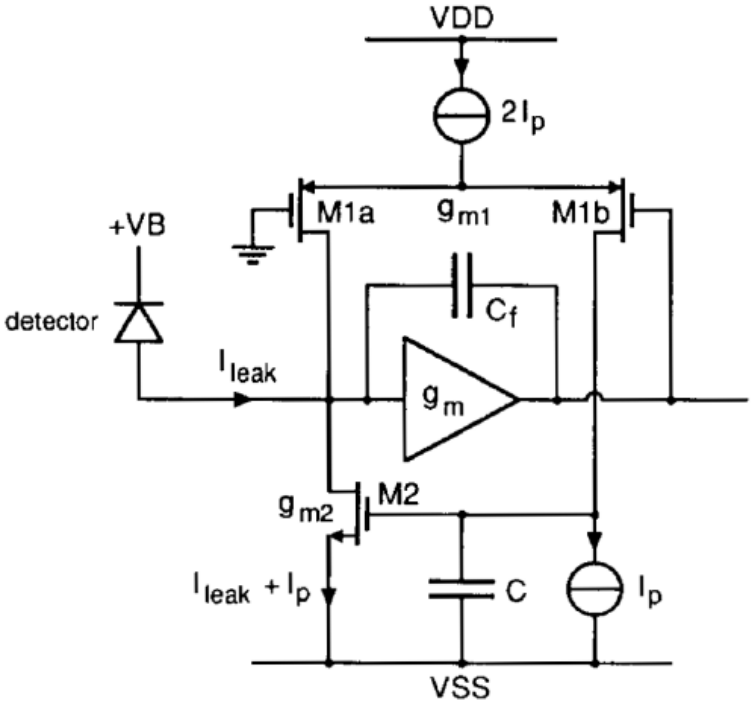
ToT: is still suitable?

- The front-end dead-time defines the ultimate efficiency.
- With a ToT scheme longer hits can be tolerated if they are less probable.
- 100 um x 100 um and 250 MHz (1GHz) /cm^2 : 25 kHz/100 kHz pixel.
- Average allowable dead-time for 0.5 % pile up probability: 200 ns (50 ns).
- 160 MHz clock=6.25 ns time bin=8 counts for the average value.
- ToT still usable up to the highest rates.
- Faster digitization techniques could also be considered (space?!)
- More than 4 bits impractica in pixel, unless one used analog sampling and out of pixel digitization.

ToPix evolution

- Straightforward approach:
- Re-optimize the front-end to cope with the new ToT time: try to benefit from the shorter ToT to simplify and spare some area.
- In the present chip, significant filtering capacitance is used to have a very linear ToT also for long ToT values and high leakage current





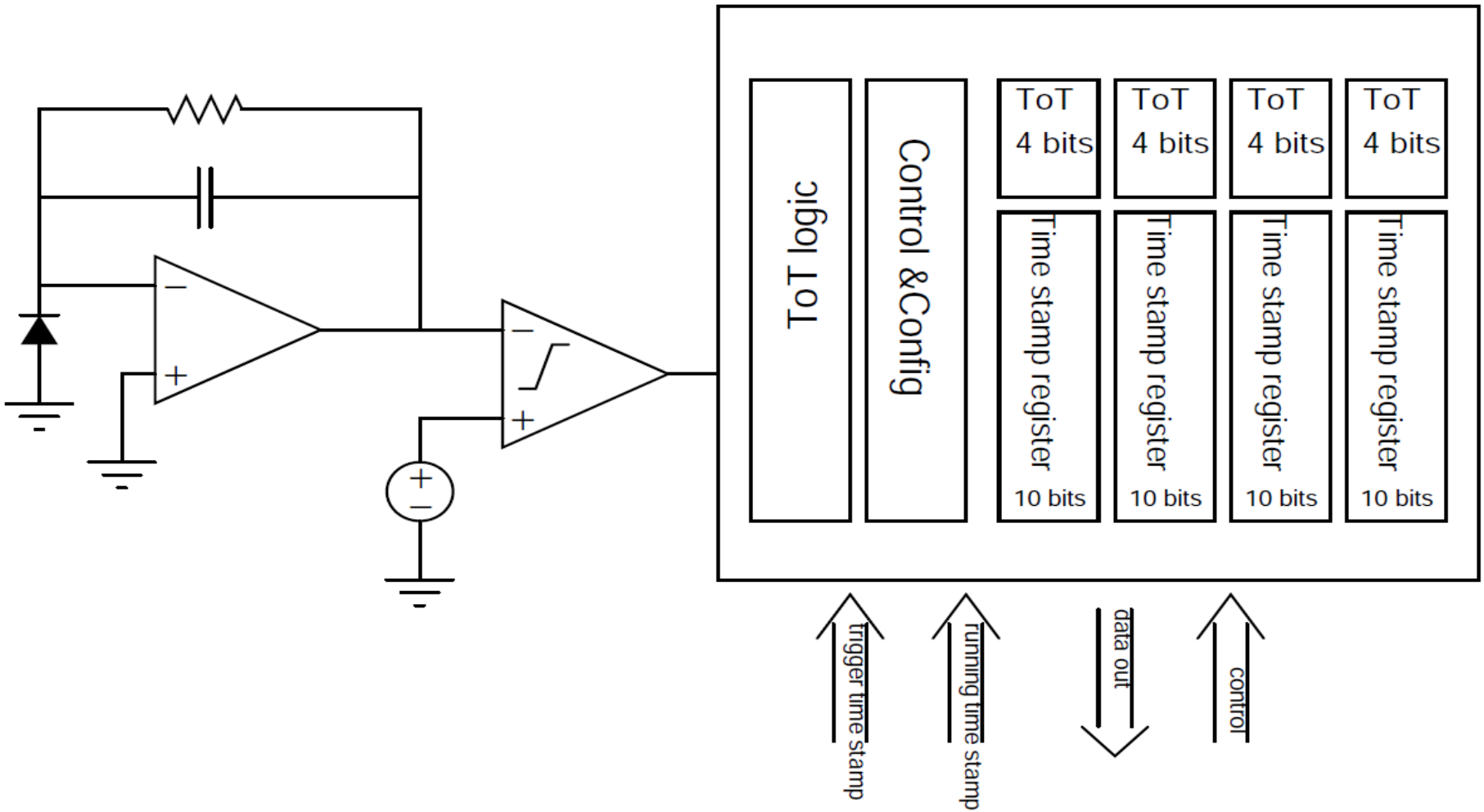
What for digital?

- Maintain the same approach: the pixel is read-out when hit, regardless if the hit is validated or not.
- Transfer all the hit to the periphery, where trigger matching is done.
- Present time for a pixel to transmit data (after it has been addressed): 56.25 ns (9 clock cycles at 160 MHz). “This time is very conservative (for PANDA was more than enough), but it could be significantly reduced” (Gianni dixit!).
- Pros': straightforward extension of the present design
- Con's: transferring to the periphery hits which then are discarded is power inefficient
- More buffering is needed at the periphery, more dead area.
- We are now evaluating this solution: it is not the most efficient one, but we know it very well so it serves as a useful benchmark. Also useful to set-up the simulation environment.

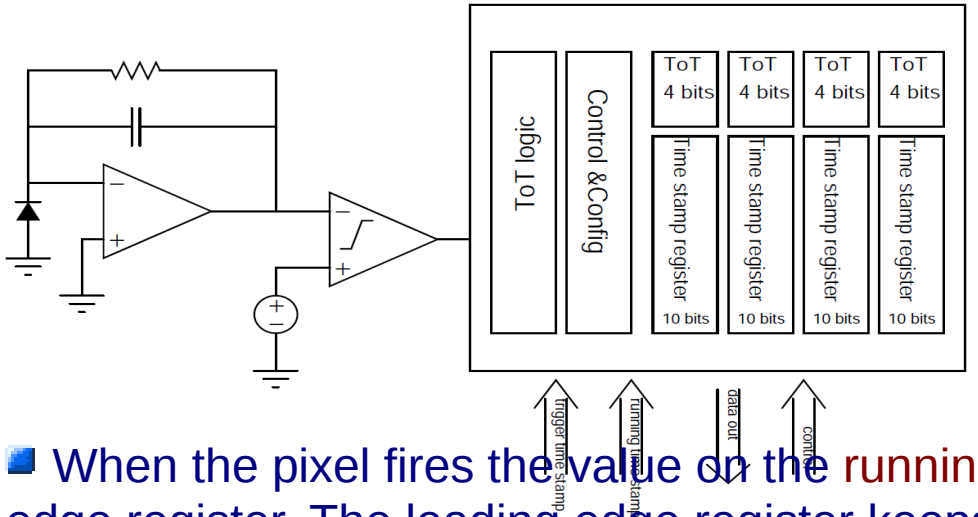
Trigger matching on pixel?

- At present, the time stamp is distributed with a pseudo-differential CMOS logic to minimize injection of noise: 24 lines to distributed 12 bits
- The technology has many insulation features,so this is probably not necessary.
- Singled ended time stamp distribution: 12 lines are freed to send a trigger time stamp.
- Actually, we do not need 12 bits: 10 are sufficient (10 bit counter at 160 MHz provides a maximum dynamic range of 6.4 us, more than sufficient to accommodate the trigger latency of 3.2 us).

Trigger matching on pixel?



Trigger matching on pixel?



- When the pixel fires the value on the running time stamp bus is stored into the leading edge register. The leading edge register keeps data for some time: SEU protected
- When the trailing edge arrives, the time stamp is stored into a temporary register: the ToT is calculated and stored into the ToT register. The temporary register is one per pixel and does not need triplication (short storage time). Alternative: use a counter for ToT measurement (requires distribution of clock).
- When a trigger arrives the trigger time stamp is calculated accounting for the latency an issue on the trigger bus (à la SuperB layer0).
- The buffers compare the stored values with the trigger time stamp and if a match is found the pixel is selected for read-out.
- The buffer compare also the stored value with the current time stamp and when the two matches the buffer is cleared (or after a read-out).
- “Four buffers needed to accommodate 3.2 us latency with negligible event loss “(Lino (and Poisson) dixerunt)

Trigger matching on pixel: realistic?

- Presently the leading edge register is 12 bits and is 10 um x 100 um with triplication
- The trailing edge register is equal to the leading edge
- In case of multi-buffering and short ToT storing both leading and trailing edges and calculating the ToT off-line in area inefficient.
- A 14 bits register to store both leading edge and ToT would take 12 um x 100 um (?).
- Four buffers would take 50 um x 100 um
- Then one must add the control logic and the triggering/time-out logic
- The configuration register in the present version is 14.4 um x 84 um
- However the new CERN cells allow a typical 2x saving in area compared with the standard cell used for the present ToPiX version
- Radiation policy?
- In ToPiX triple redundance and majority voting everywhere
- FE-I4: simple DICE cells for data storage, DICE cells AND triple redundancy for control registers. Ok for phase 1. What for phase 2?
- A pixel with local trigger matching and independent pixel not a priori impossible, but of course must be checked in reality.

What next?

- Perform a detailed study of the “straightforward adapation” (i.e. “read all hit pixels, read faster, match trigger in the EoC”). Simulations already started by Giulio Dellacasa.
- Study of the in-pixel logic for the local trigger matching to confirm is this is realistic.
- Define a clear prototyping strategy for the future.
- In pixel chips too small prototypes are useless unless one is trying really something fancy (i.e. 45 nm?)
- What is useful: prototype of reasonable size (à la ToPiX 3), bump bondable to a sensor.
- Complex building blocks which are not indispensable on the first prototype, but are of course fundamental in the final one (e.g PLL for clock multiplication, robust biasing generators, etc...).
- Also a more elaborate version of the EoC logic may come later (e.g. the EoC logic of ToPiX_v2 was minimal to allow to read-out the chip)