UNIVERSITÄT BONN

SI LAB
Silizium Labor Bonn

High bandwidth data transfer on and off-chip for HEP: modeling, design and verification

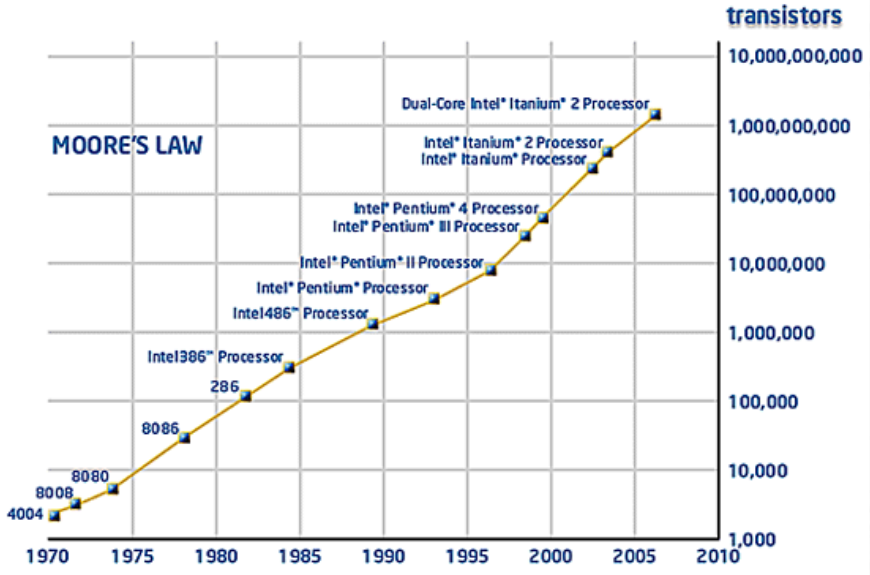
Tomasz Hemperek, Hans Krüger

Outline

UNIVERSITÄT BONN

- Introduction
- Modeling
- Design
- Verification
- Ouput links
- Examples
  - DHP (Belle 2)
  - RD53A (ATLAS/CMS)
  - VeloPix (LHCb)

Moore's law in HEP (pixel detectors)



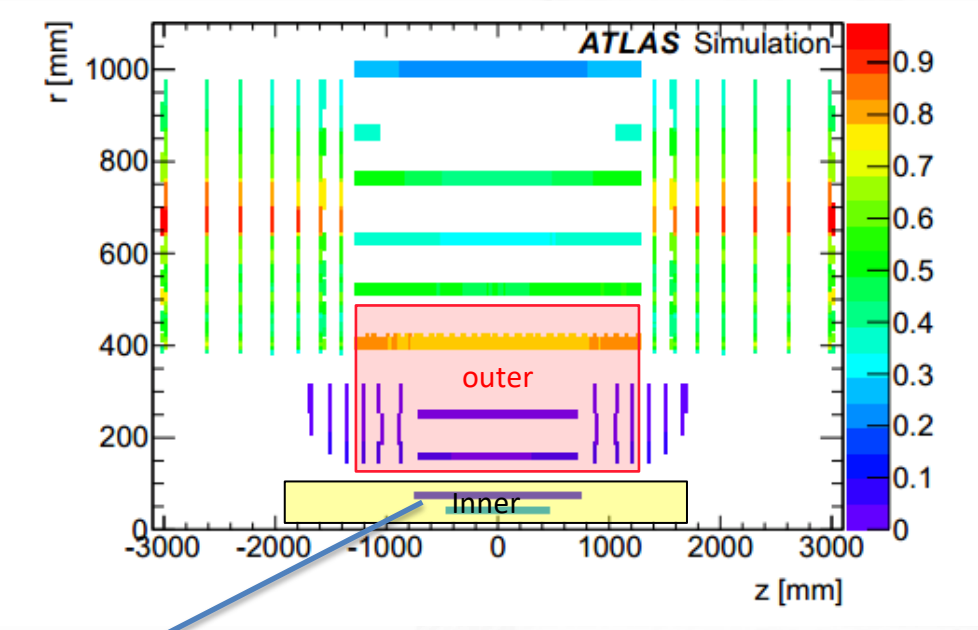
Name	D-OMEGA Ion	LHC1	FE-I3	FE-I4	RD53
Year	1991	~1996	~2005	~2011	2017/2019
Technology Node	3 µm	1µ	250 nm	130 nm	65 nm
Chip size	8.3x6.6 mm^2	8x6.35 mm^2	10.8x7.6 mm^2	10.2x19 mm^2	20x22mm^2
Pixel size	75x500 μm^2	50x500 μm^2	50x400 μm^2	50x250 μm^2	50x50 μm^2
Pixel array	16x63	16x127	18x160	80x336	~400x400
Transistor count	???	800k	3.5M	80M	>500M

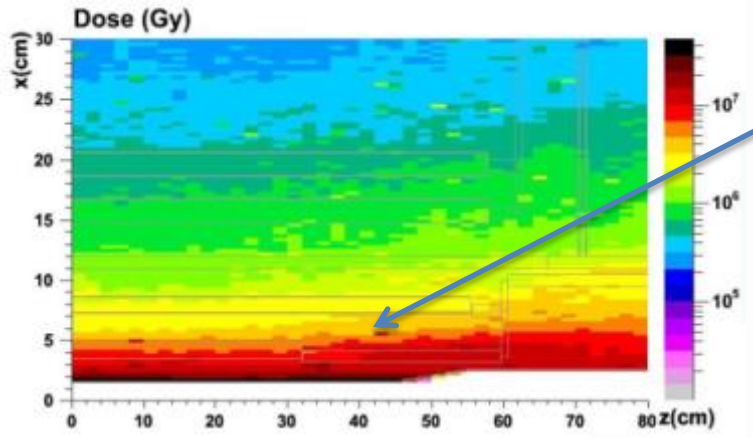
Memory Density

Technology node	130nm	65nm	28nm
6T SRAM cell (um^2)	2.4	0.52	0.127
bit size in memory (um^2)	~ 3.2	~ 0.7	~ 0.16
10bit words in 100x100um	~ 310	~ 1430	~ 6250
NRI/MPW* (mm^2)	$1000-$2000	$3000-$4000	$8000-$10000

Issues in pixel detector readout in HEP

- Time stamping (40MHz)
- Hit rate (<3GHz/cm^2)
- Trigger rate (up to 4MHz)
- Wait time for trigger (up to 35us)
- Cooling (+40 to -30C)
- Cables (up to 6m to DAQ)
- Power delivery/support mass
- Data Rates (>2Gbits/s/cm^2)
- Resolution (<15um)
- Radiation > 5MGry
- SEU/SET

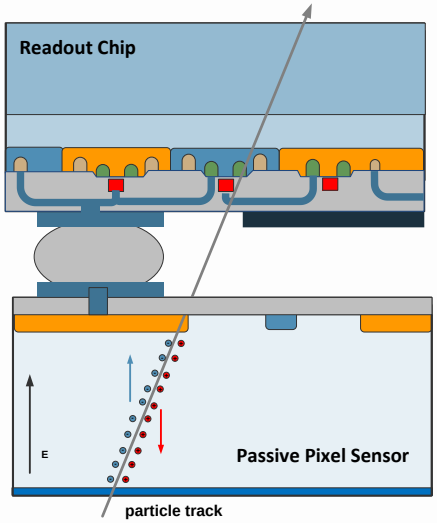


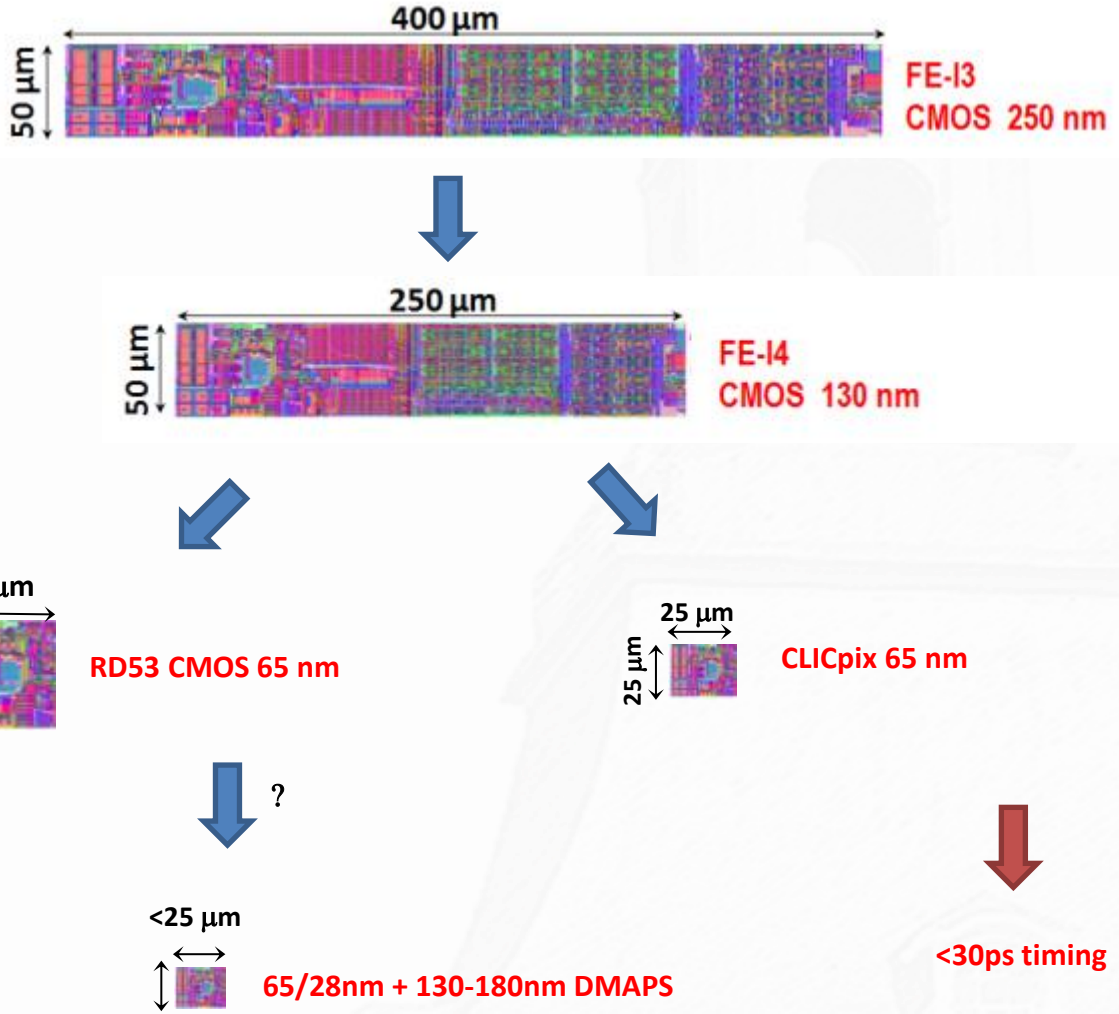


- **Radiation levels:**
  - at 5 cm : ~15 MGy (2•10^{16} n_{eq}/cm^2)
  - at 25cm : ~1 MGy (10^{15} n_{eq}/cm^2)

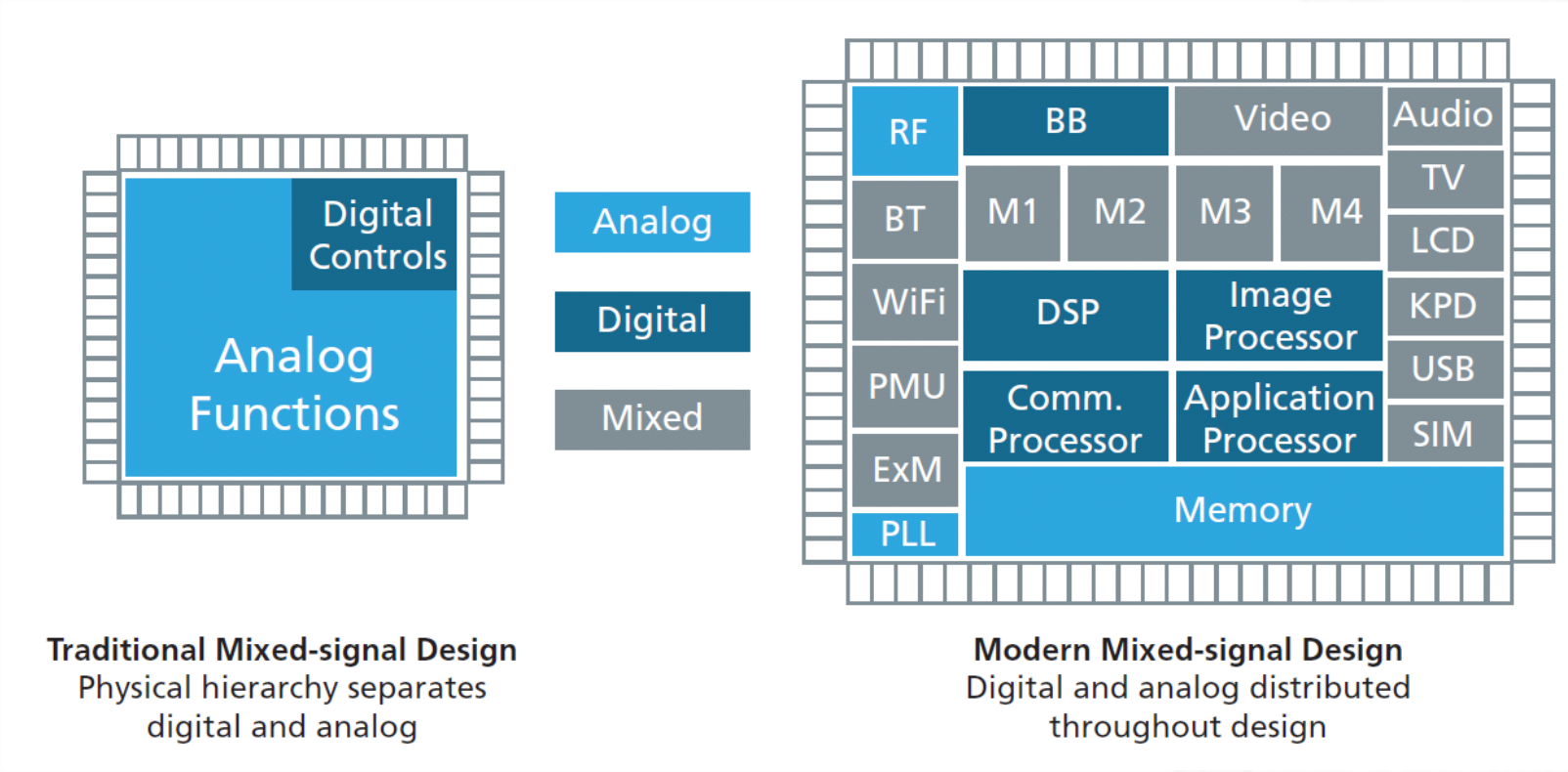
** estimates for 10years of operations*

Hybrid Pixel evolution in HEP



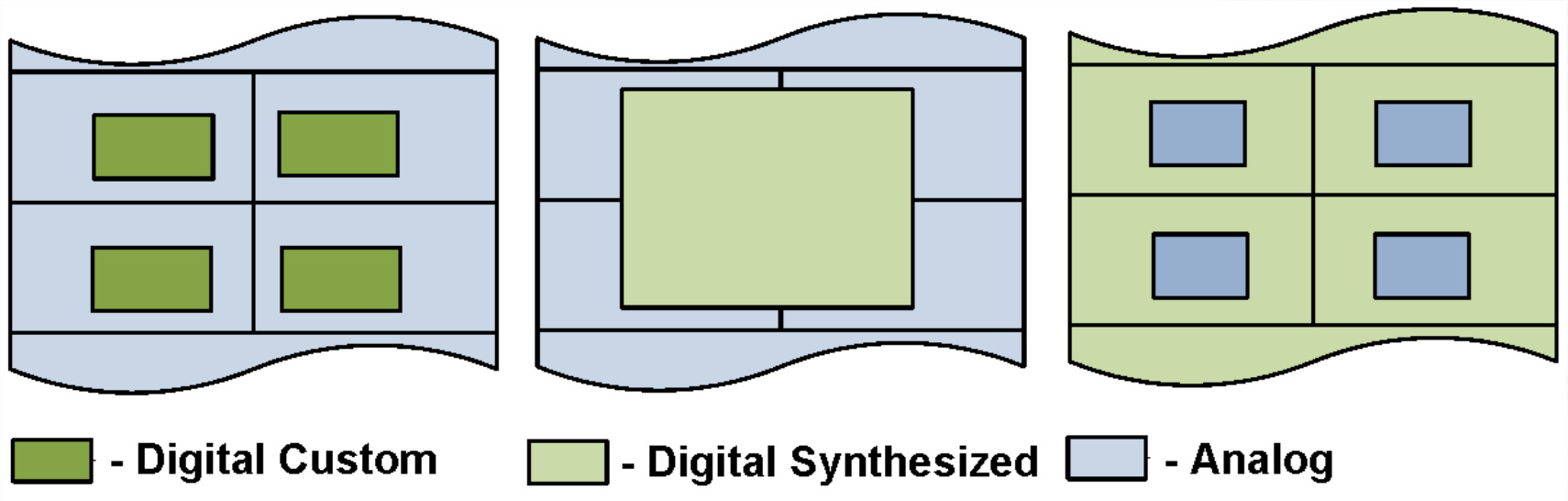


Switch to big “D”, little “A”



Same pattern for HEP

Evolution of array organization



Traditional Design:
- design 1 pixel
- step and repeat identical copies
- custom made digital

ex. FE-I3 (250nm)

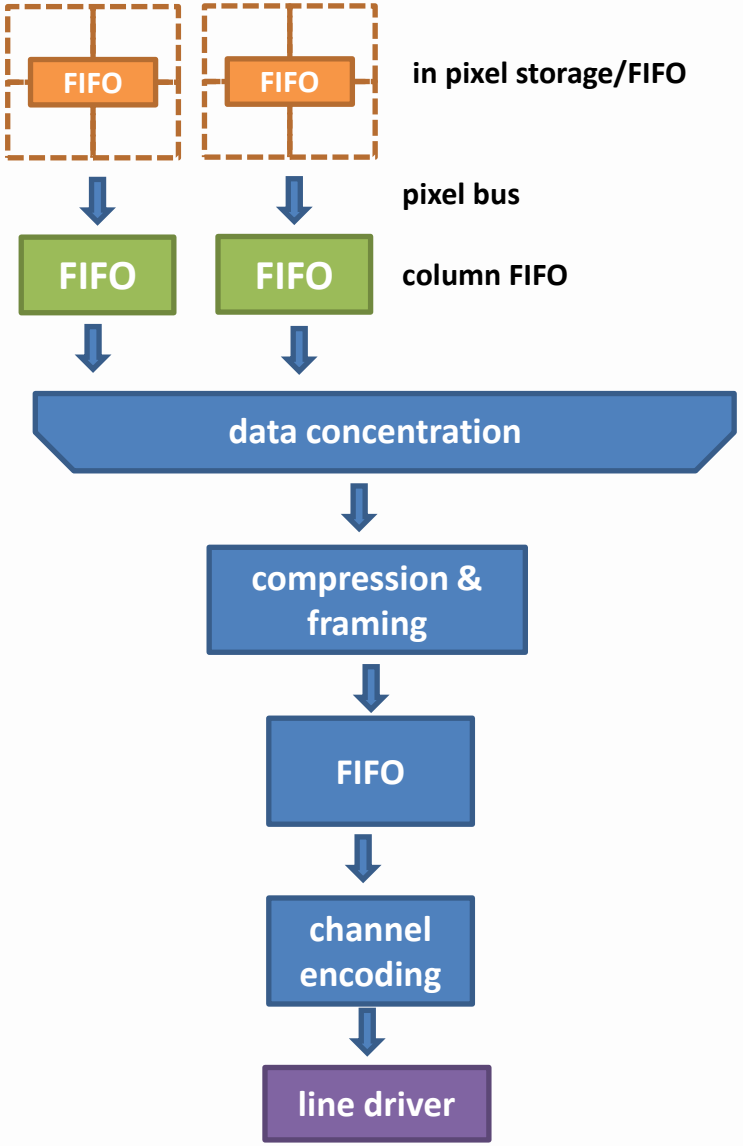
More Recent:
- design few-pixel region
- step and repeat identical copies
- synthesized digital

ex. FE-I4 (130nm)

Recent:
- synthesized entire design with analog IP in a hierarchical way

ex. RD53A (65nm)

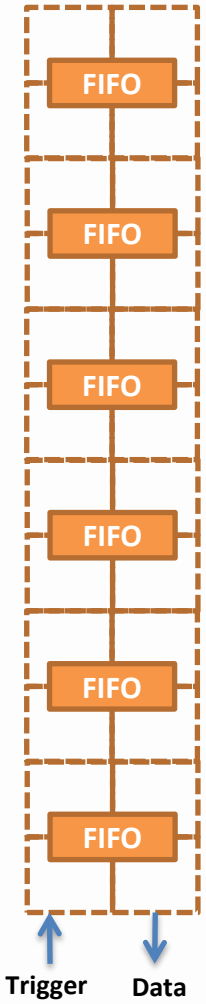
Modeling - Introduction



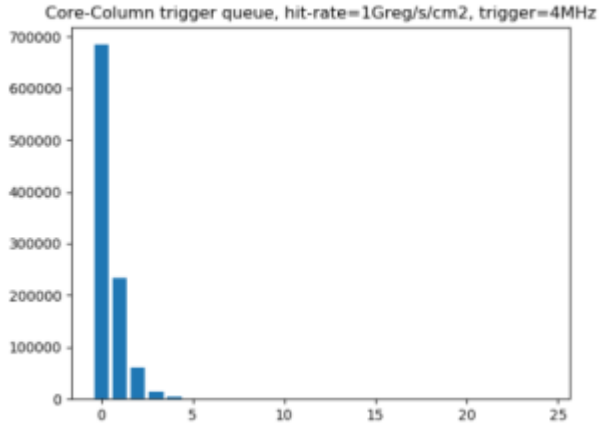
Questions:

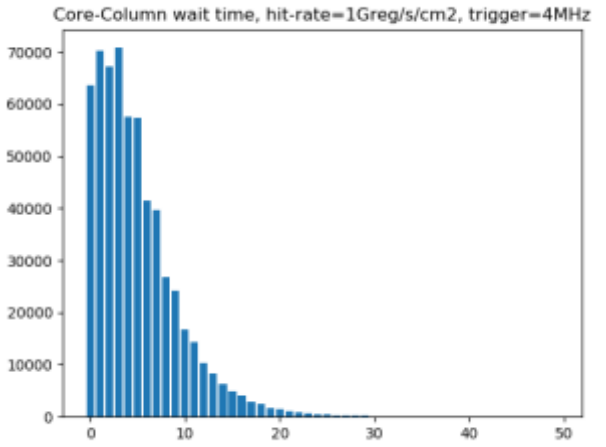
- How to portion pixel array (buffering and readout)?
- How wide buses?
- How fast clocks (domain crossing)?
- What data format/encoding?
- How to compress?
- How big the FIFOs and how many?
- What data processing?
- How much time to send and trigger?

Modeling – python (C++…)




```
import numpy as np

PIXEL_NO = 8*400
PIXEL_AREA = 50*50
HIT_RATE_CM = 0.5*(10**9) # this is hit rate should be region rate for this
TRIGGER_RATE = 4.0/40

pixel_hit_rate_bx = ((float(PIXEL_AREA)/(10000*10000)) * HIT_RATE_CM )/ (40*(10**6))
trig_mem = np.full((100000) ,-1, dtype=np.int)
stat_trig = np.full((10000) ,0, dtype=np.int)
wait_time = np.full((10000) ,0, dtype=np.int)
hits_list = {}
reg_wait_time = 0
hits_count = 0
trig_id = 0
trig_req_id = 0

for bx in range(1000000): #(1000000):

    trig = np.random.rand(1) < TRIGGER_RATE #if trigger

    if trig:
        hits =  np.count_nonzero(np.random.rand(PIXEL_NO) < pixel_hit_rate_bx) #lottery
        empty = np.where(trig_mem==-1)[0] #check in queue empty position
        trig_mem[empty[:hits]] = trig_id #insert
        hits_list[trig_id] = [bx,hits]
        trig_id += 1

    #every 2nd bx remove one hit or move with trigger
    if bx % 2 == 0:
        reg_hits = np.where(trig_mem==trig_req_id)[0]
        if len(reg_hits): #if hits with this trigger id
            waited = bx - hits_list[trig_req_id][0]
            ht = hits_list[trig_req_id][1]
            hits_count += ht
            reg_wait_time += (waited*ht)
            wait_time[waited] += ht

            trig_mem[reg_hits[0]] = -1

        reg_hits = np.where(trig_mem==trig_req_id)[0]
        if len(reg_hits) == 0:
            if trig_id > trig_req_id:
                trig_req_id += 1

    trig_in_fifo = np.count_nonzero(np.unique(trig_mem))-1
    stat_trig[trig_in_fifo] += 1
    if bx % 1000 == 999:
        print 'bx:', bx , 'trig_queue:', trig_in_fifo
```

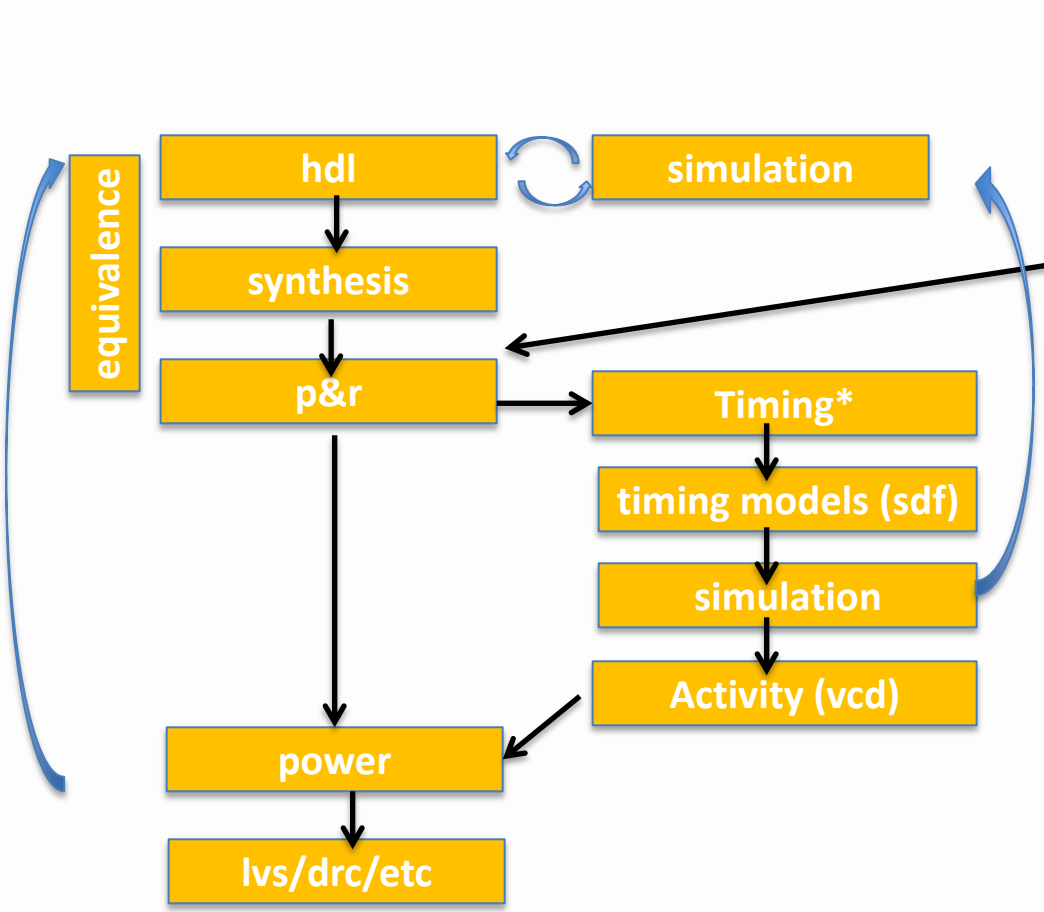




Simple and good for exploration.
Use UVM?

https://gist.github.com/themperek/31720b7a186618b17f489a3ad504638c

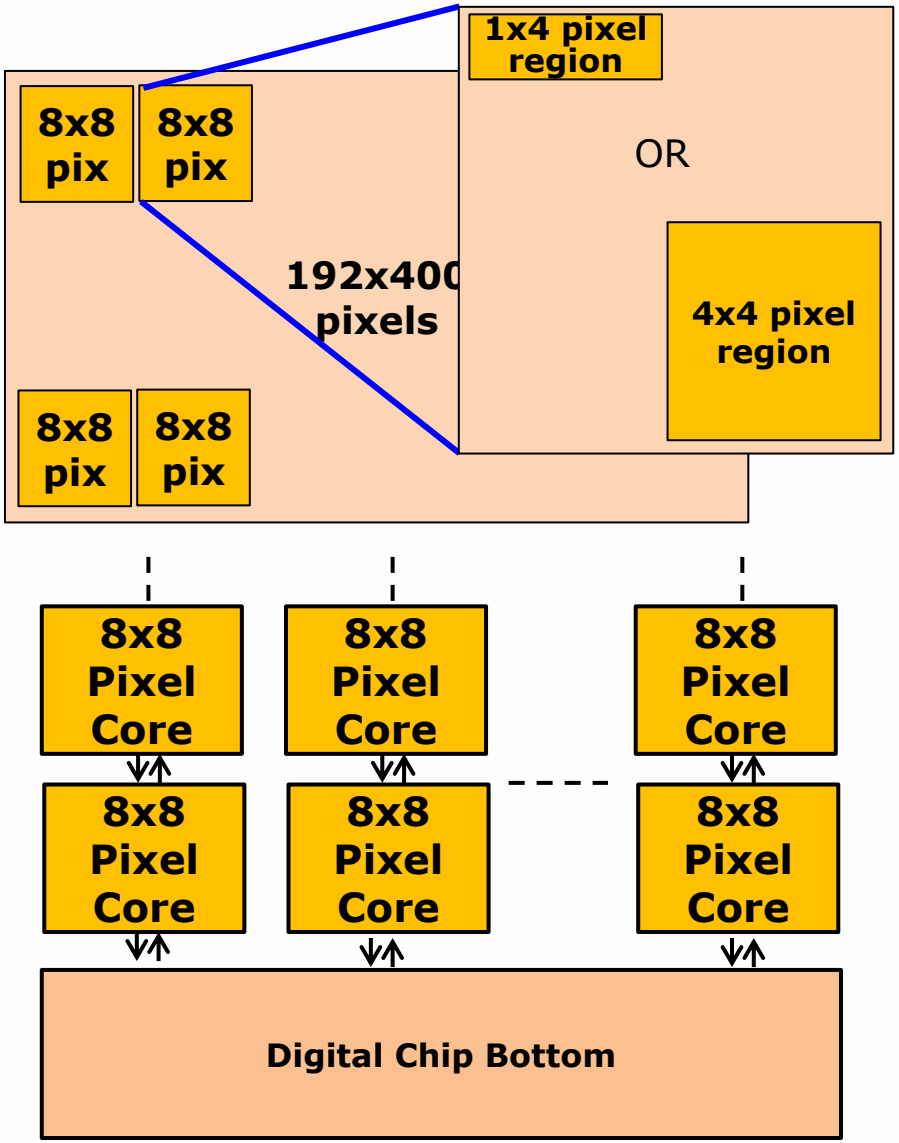
Digital Flow – from digital perspective

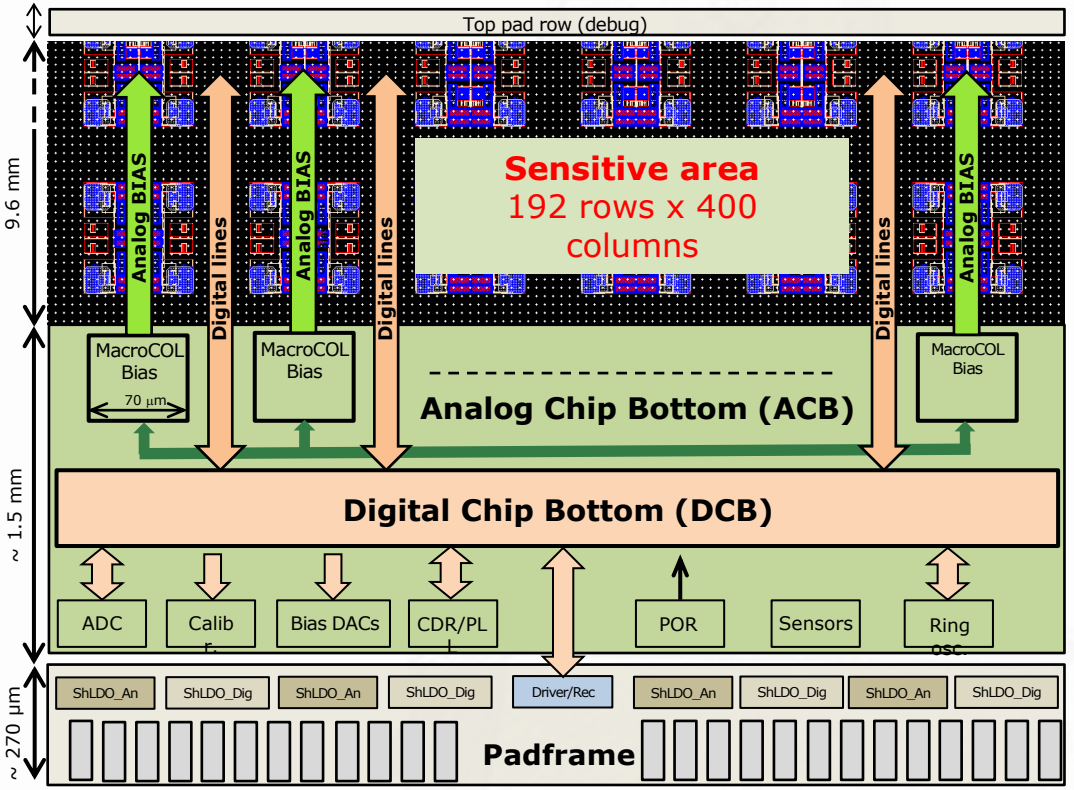


The software zoo:

- Synthesis
- Place & Route
- Timing & Si
- Power Verification
- DFT / ATPG
- Equivalence
- Formal
- Simulation (many tools)

The design (RD53A)

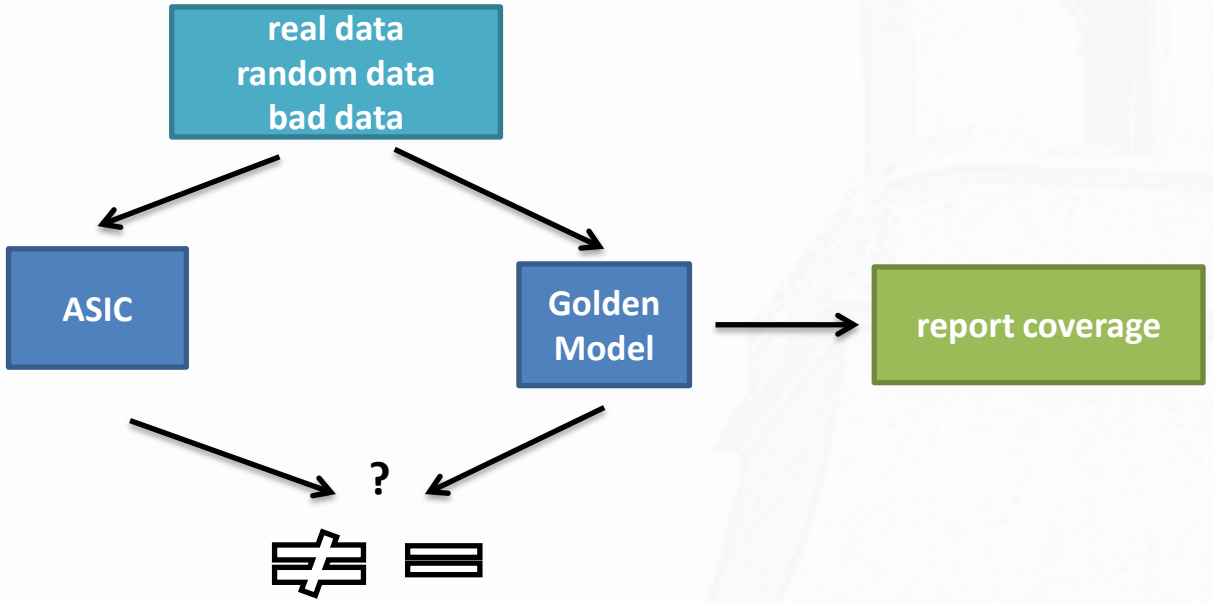




- Fully digital with analog IP
- Hierarchical
- Fully automated
- 1 day to resin and verify the whole chip

Verification Introduction

- Verification takes more time then design.
- It has to start before/together with the design.
- Failing a $1M chip (65nm) is not a good idea.
- No way out for complex digital chips.

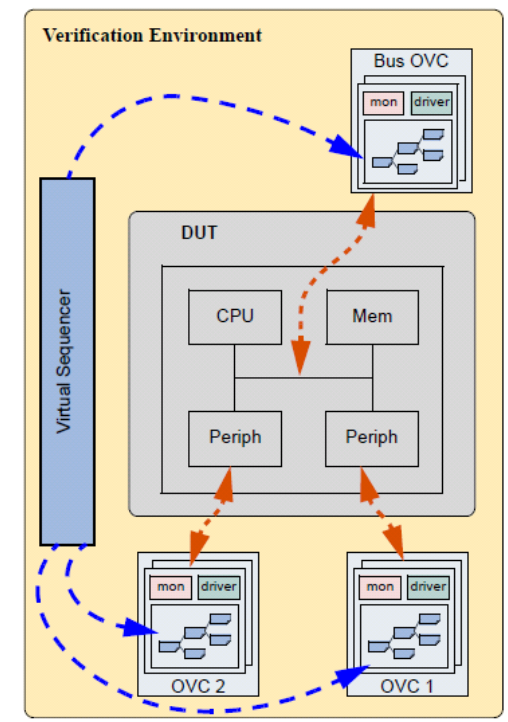


Verification Plan -> Most important part (trash in trash out)

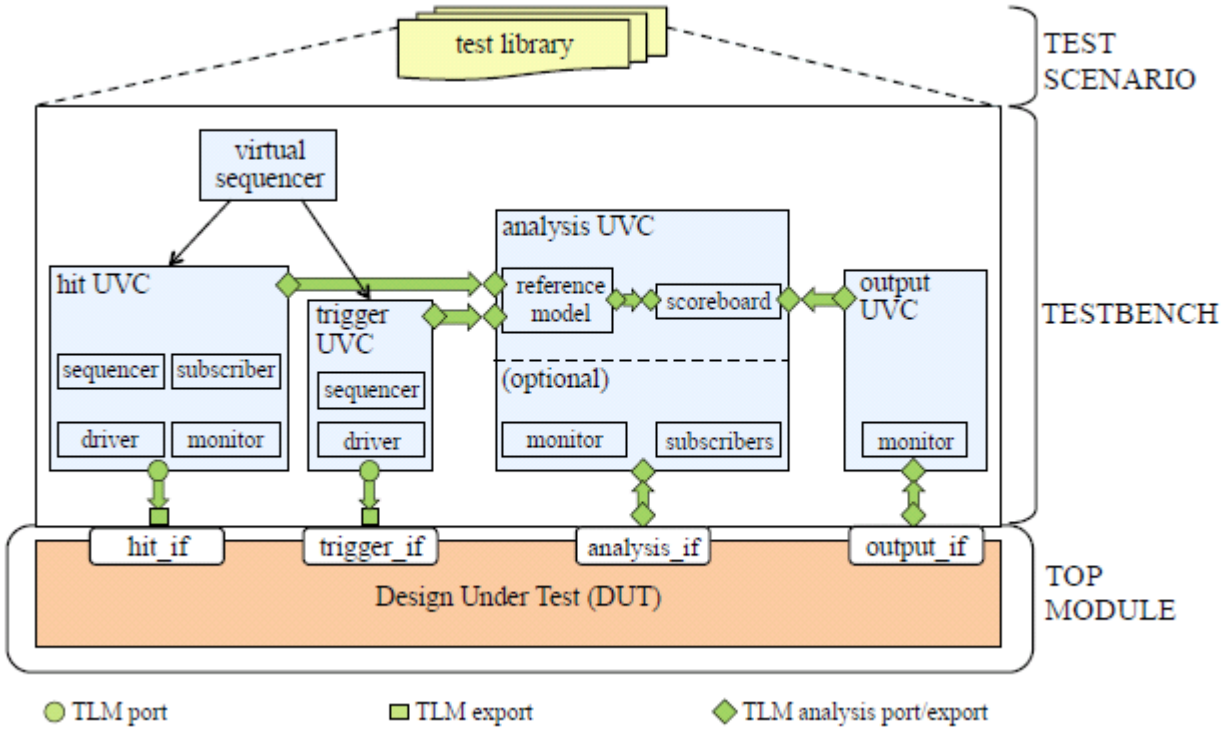
Universal Verification Methodology (UVM)

UNIVERSITÄT BONN

UVM



Example for RD53A:

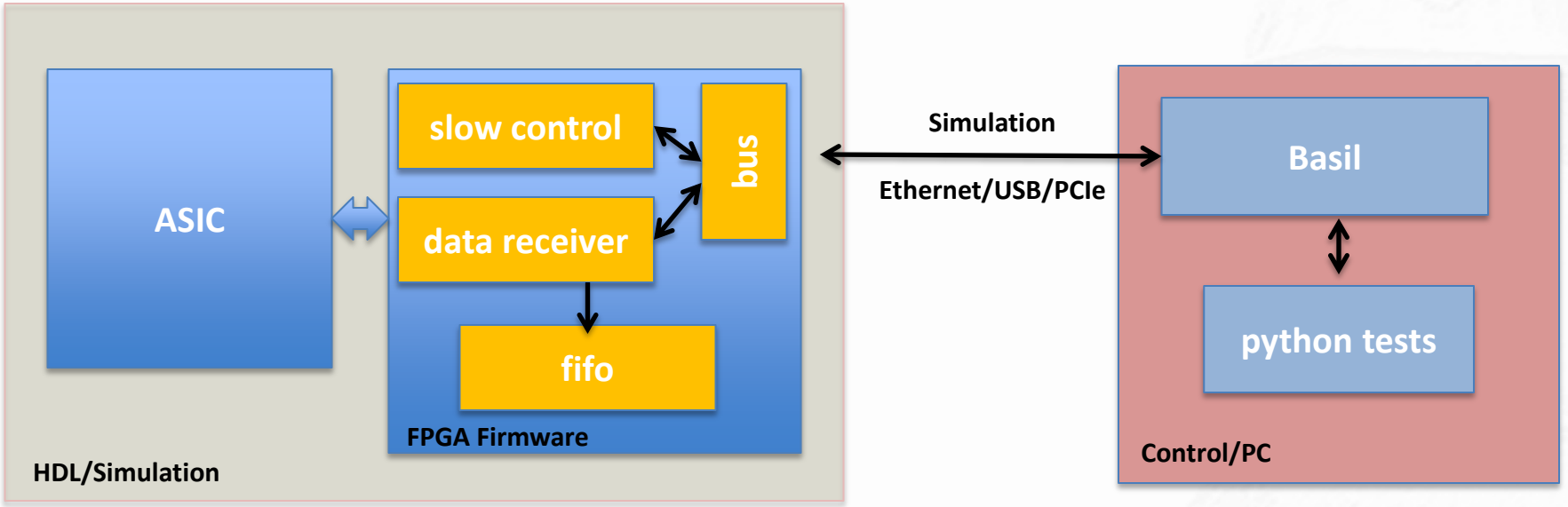


Very complex industrial standard

Formall verification ...

https://arxiv.org/pdf/1408.3232.pdf

Verification – DAQ Integration




```
self.chip['global_conf']['InjEnLd'] = 0
self.chip['global_conf']['TDacLd'] = 0
self.chip['global_conf']['PixConfLd'] = 0
self.chip['global_conf'].set_wait(200)
self.chip.write_global() #send some test hit

while not self.chip['trigger'].is_done():
    pass

print 'fifo_size', self.chip['fifo'].get_fifo_size()
data = self.chip['fifo'].get_data()
```

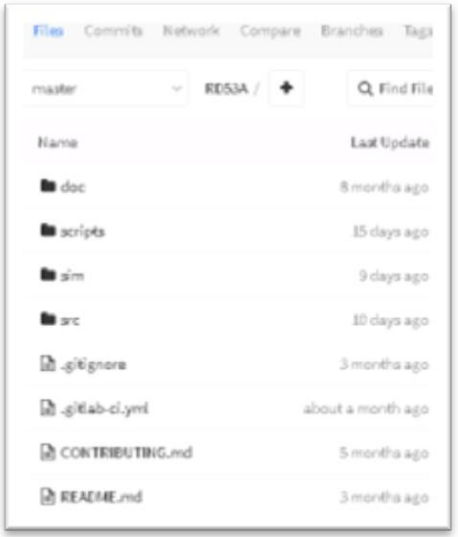
- Exactly the same python code can be used for testing physical chip.
- Same firmware for FPGA (ready to use by DAQ). DAQ specification.
- Can get first data within minutes from chip delivery.

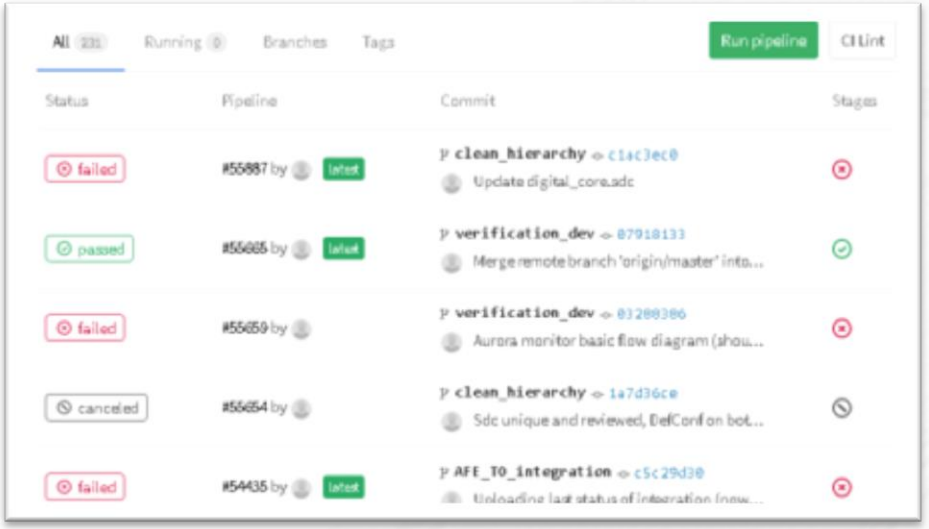
Solution in C++ and other languages exist,

Work organization/procedures

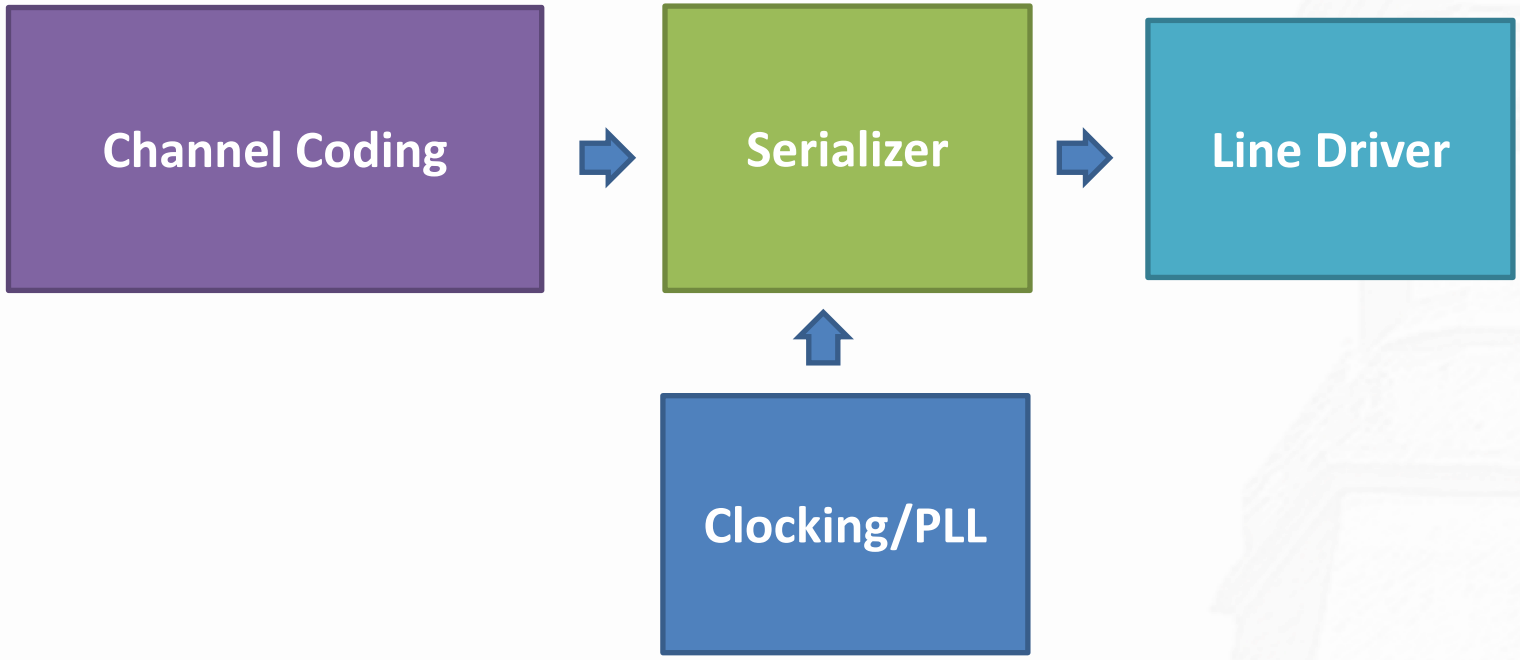
UNIVERSITÄT BONN

- We use version control (for everything)
  - Branching for new features
  - Poll request review on request
  - Features and bugs discussions via issues system (not email)
- Continuous integration
- Agile development vs waterfall





Output Links

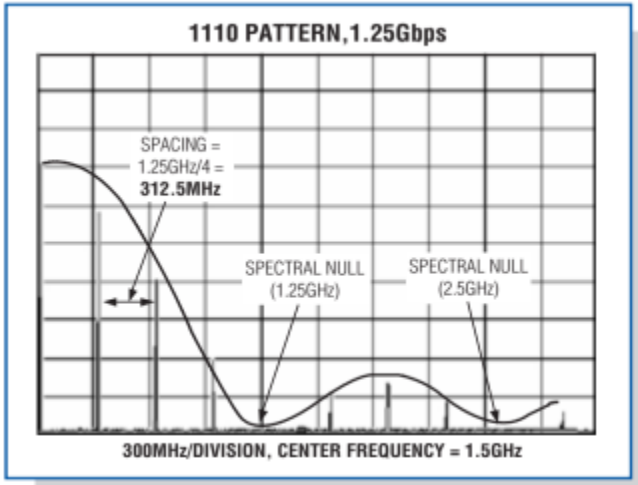


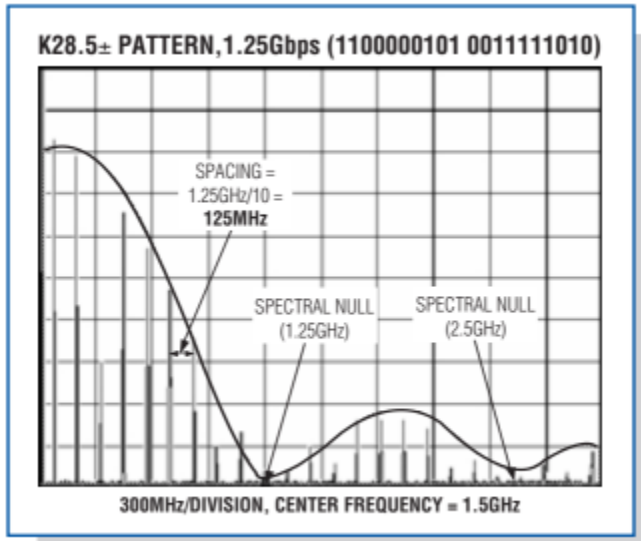
Limitation is often cables no the transmitter.

Output Links – Line encoding

Considerations:

- Run length
- DC balance
- Hamming distance
- Support by DAQ
- Framing/Streaming
- **Cables**





Examples:

8b/10b : Ethernet, Fibre Channel, high-speed video applications

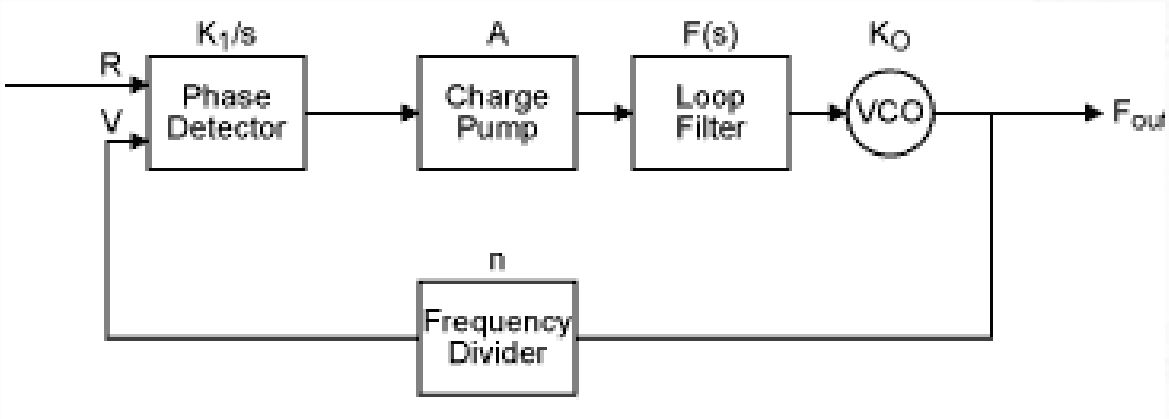
Ex. implemetation: Aurora 8b10b from Xilinx

64b66b: SONET and SDH telecommunication

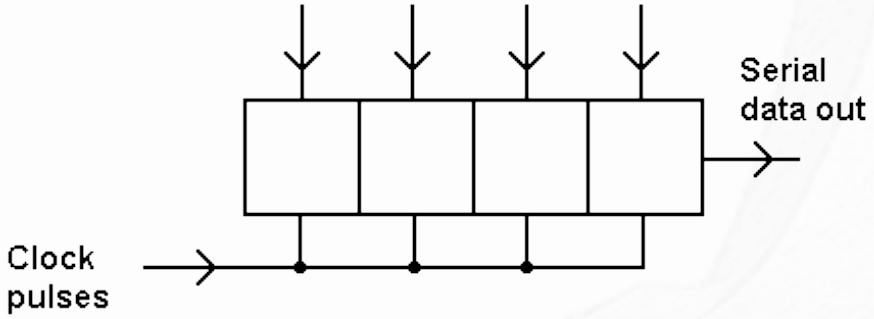
Ex. implemetation: Aurora 64b66b from Xilinx

Output Links - Clocking

Phase Locked Loop (PLL)

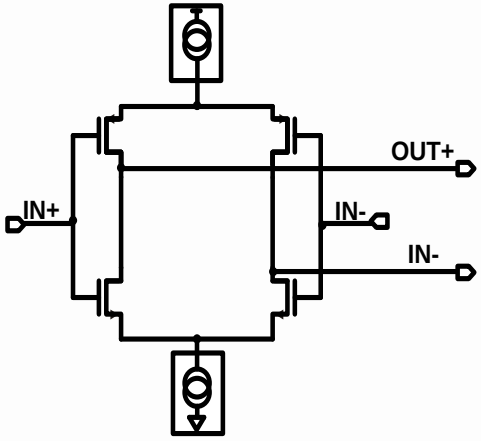


Serializer



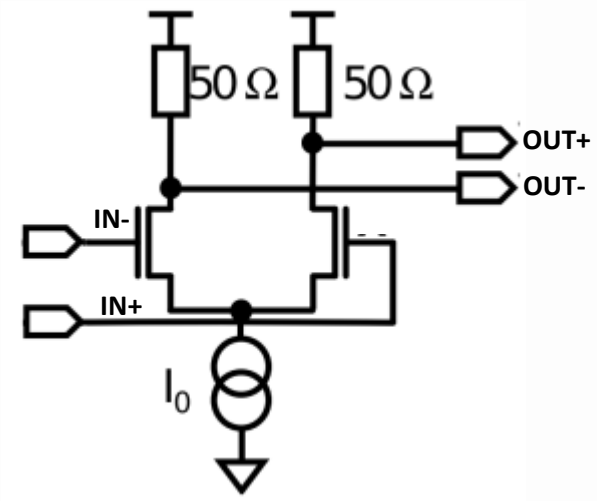
Output Links – Line Driver

LVDS



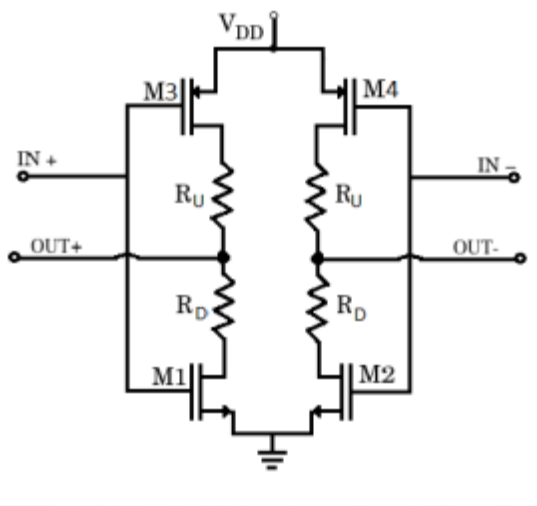
Power: ~3-4mA
Speed: ~1Gbit/s

CML



Power: ~20-30mA
Speed: >10Gbit/s

SST

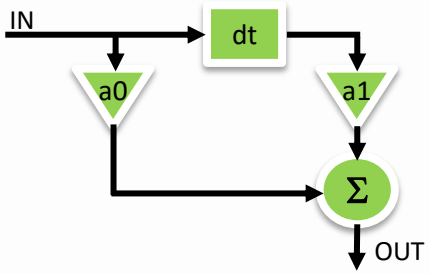


Power: lower then CML
Speed: >10Gbit/s

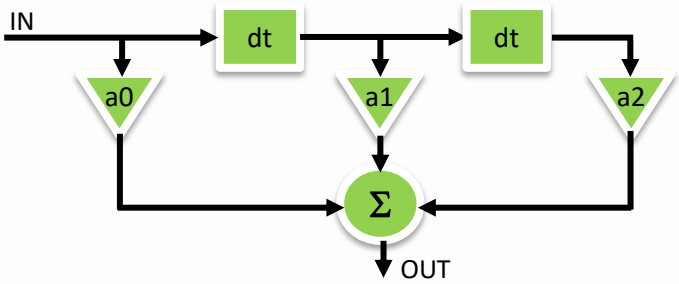
Pre-emphasis

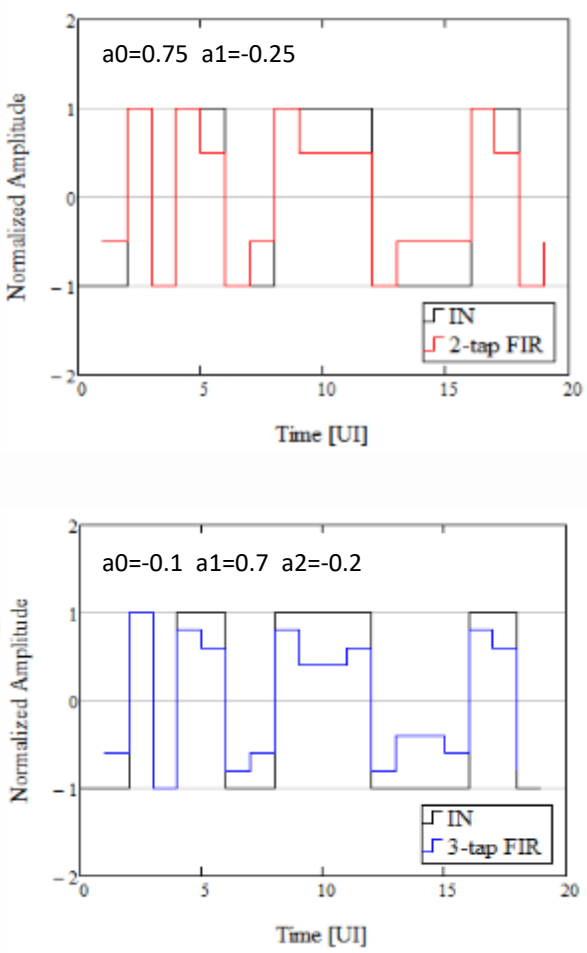
Implementation with a digital filter (FIR filter)

- **2-tap FIR**

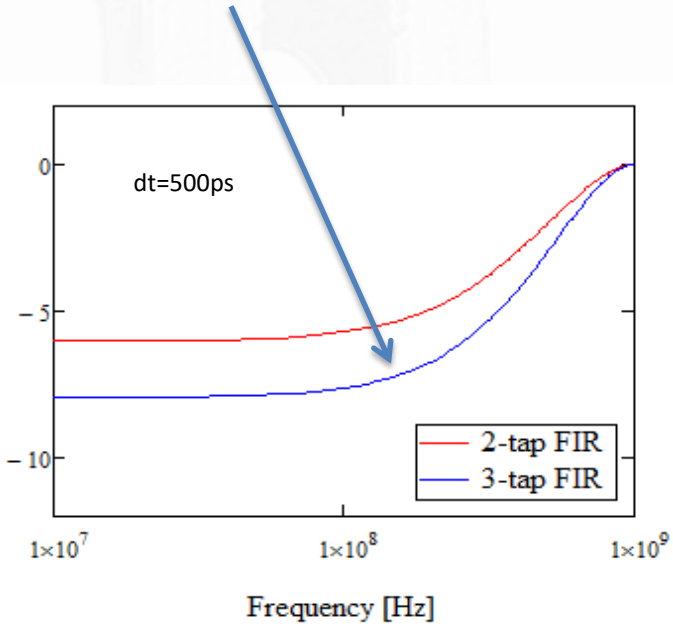


- **3-tap FIR**



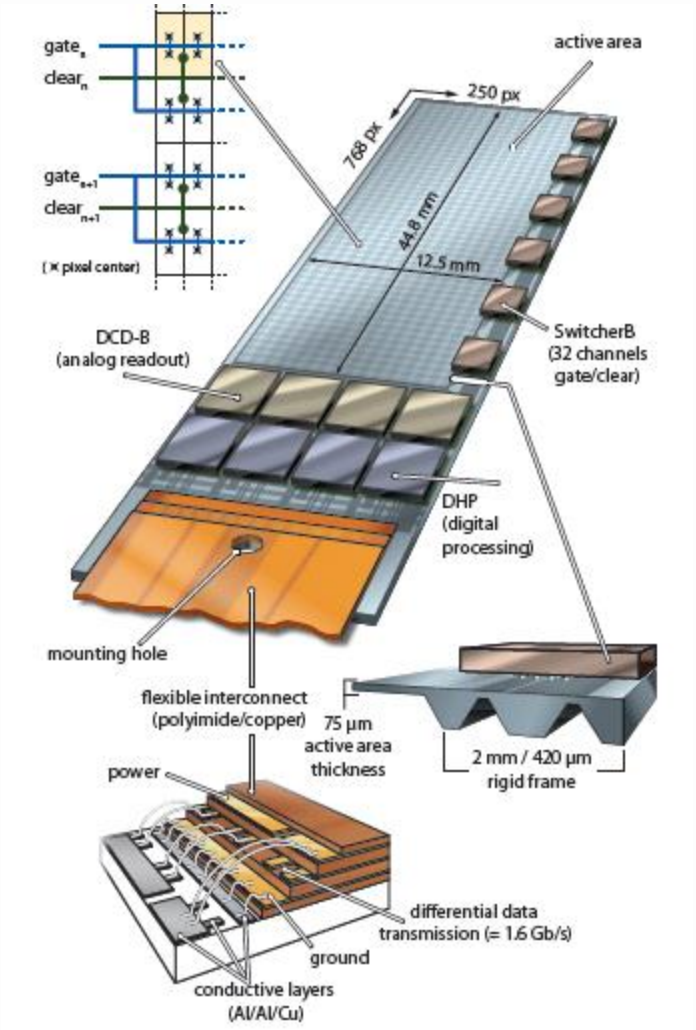


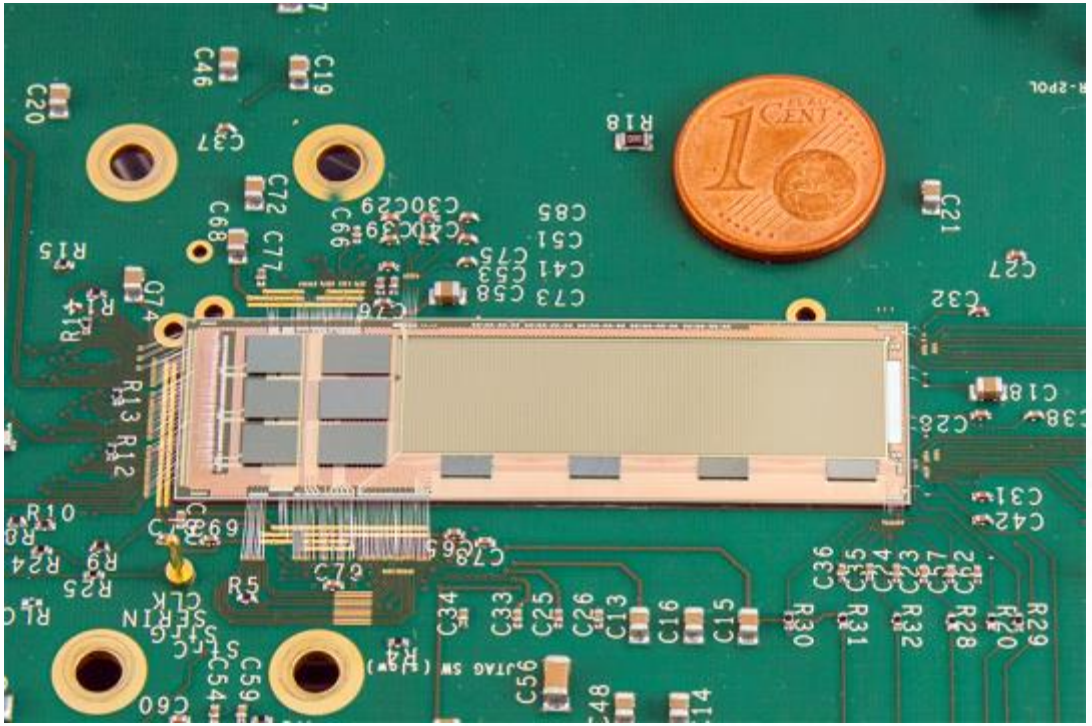
3 taps allow steeper transfer function compared to 2 tap



EXAMPLES

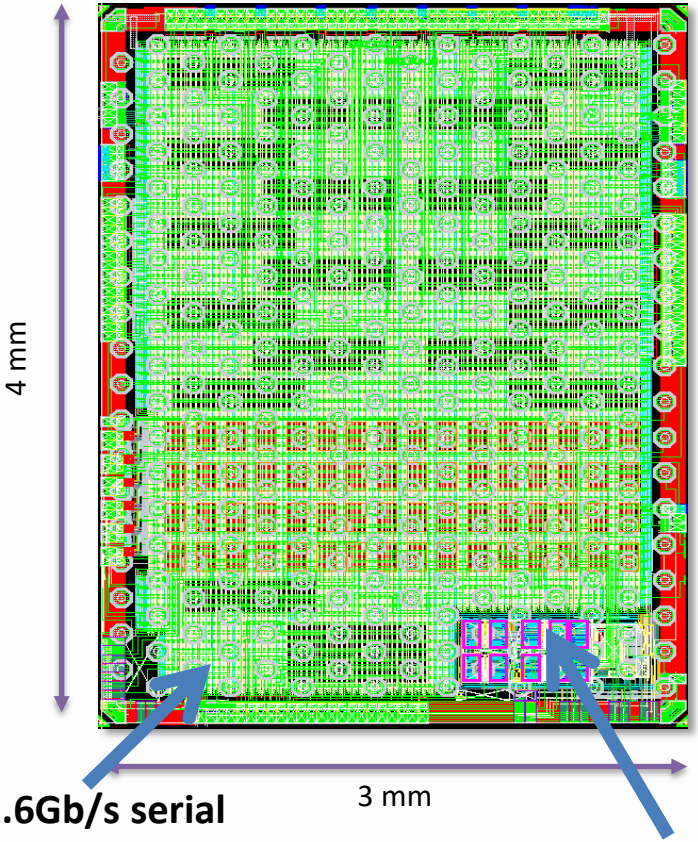
DHP - BELLE 2 Pixel Module





Examples – DHP

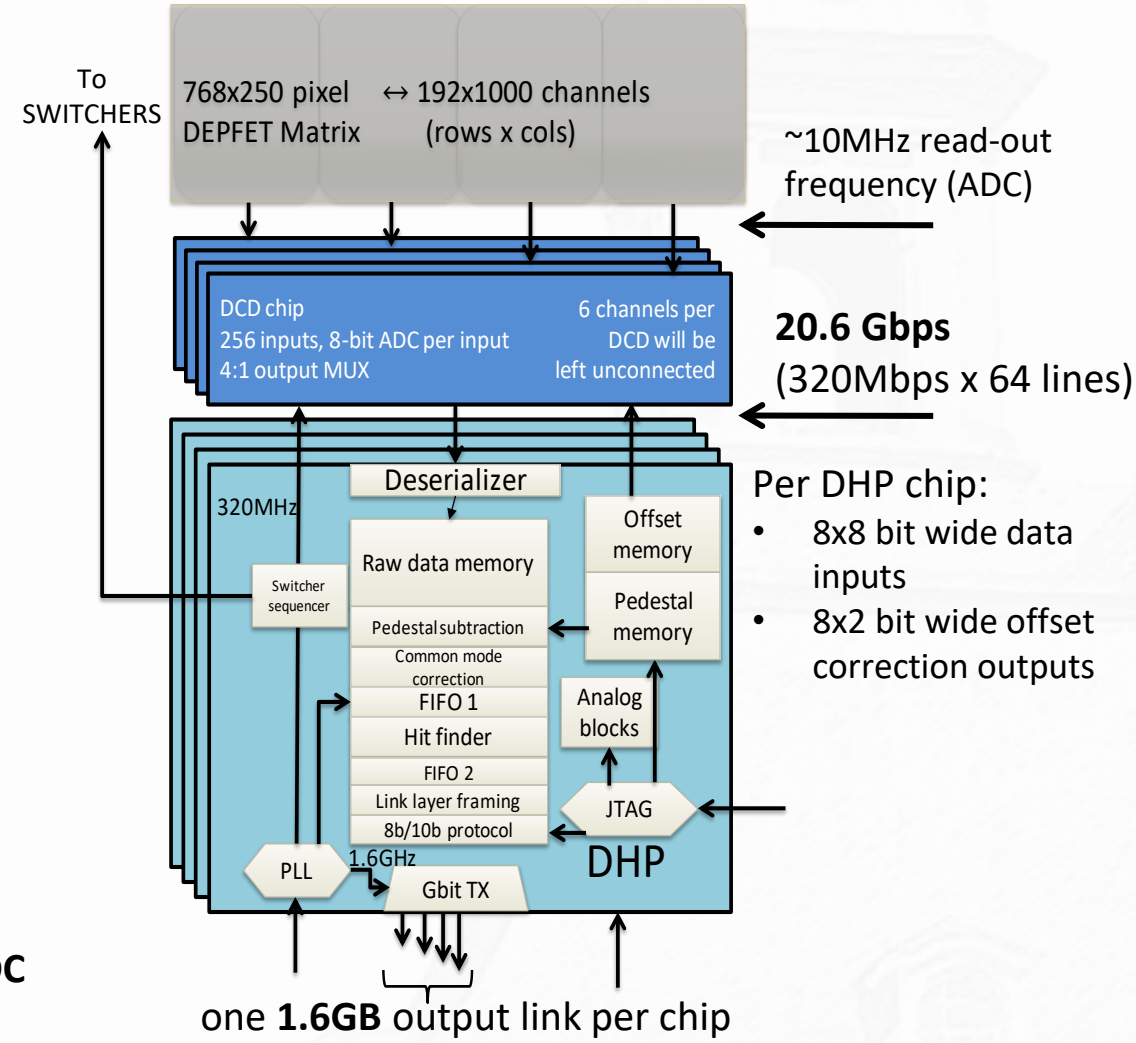
300k Gates, >3MB SRAM(~100 Blocks)
~100 x LVDS and HSTL IO



PLL, 1.6Gb/s serial link, CML preamhasis

12x DACs, ADC

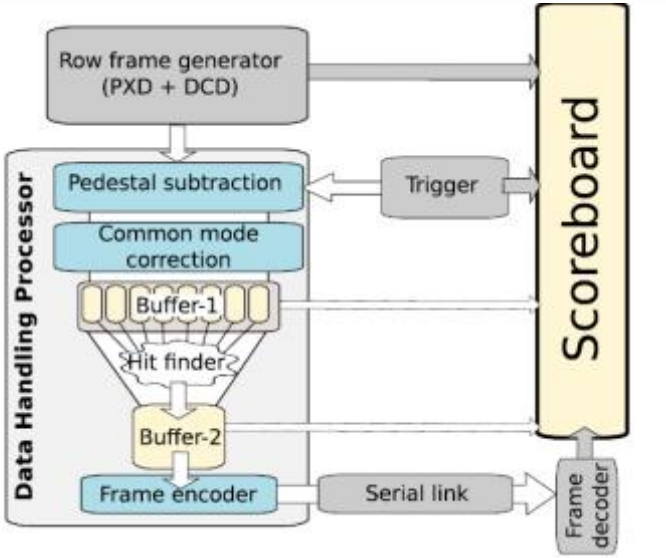
4 Power Domains, 2 Clock Domains, Lot of FIFOs/memory and processing.



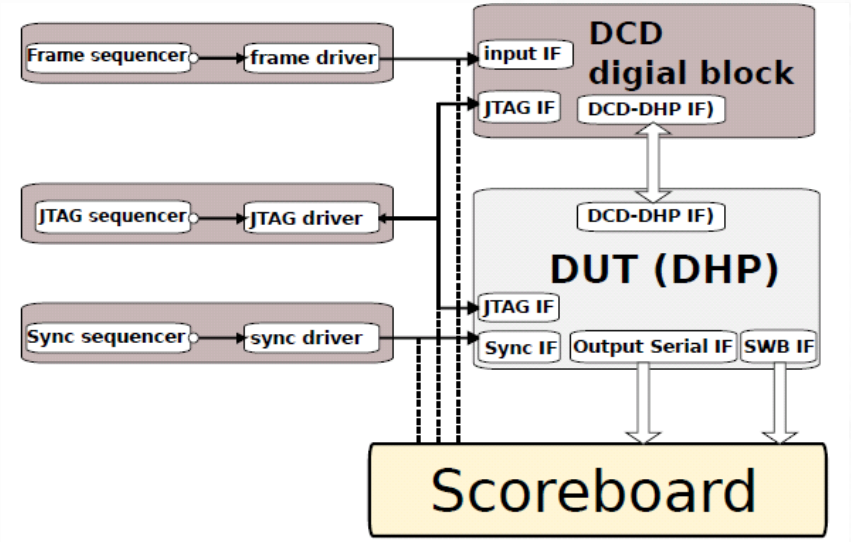
~ 150GBit/s per pixel detector

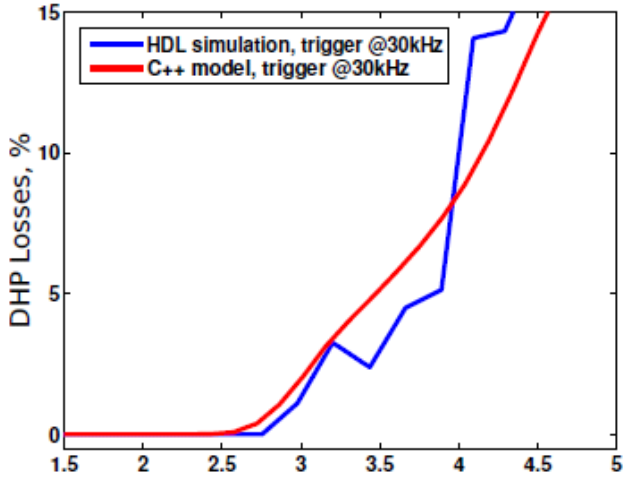
DHP - Modeling/Verification

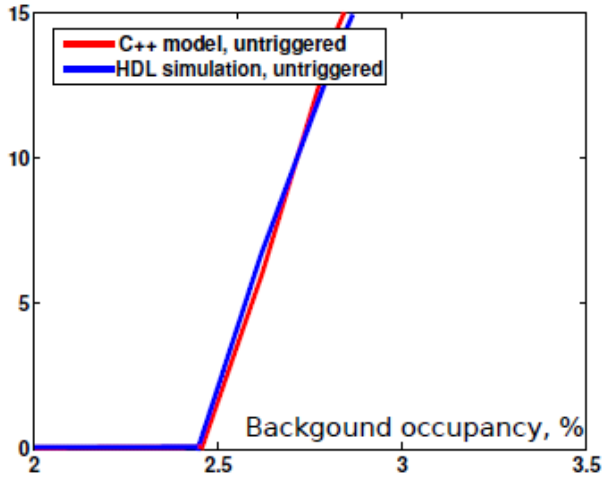
C++ model



UVM Test Enviromtn



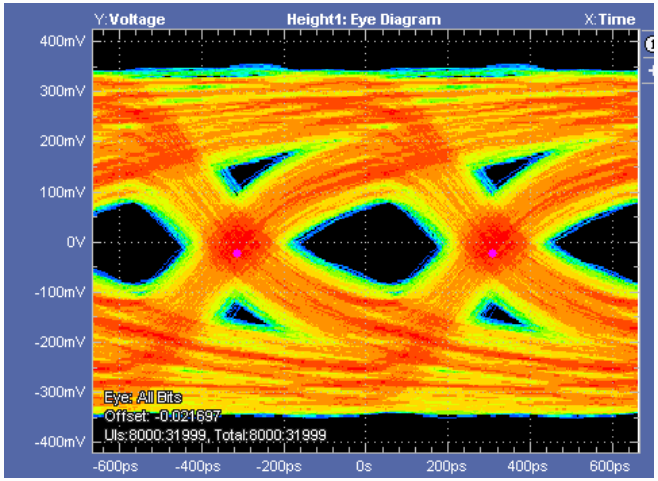




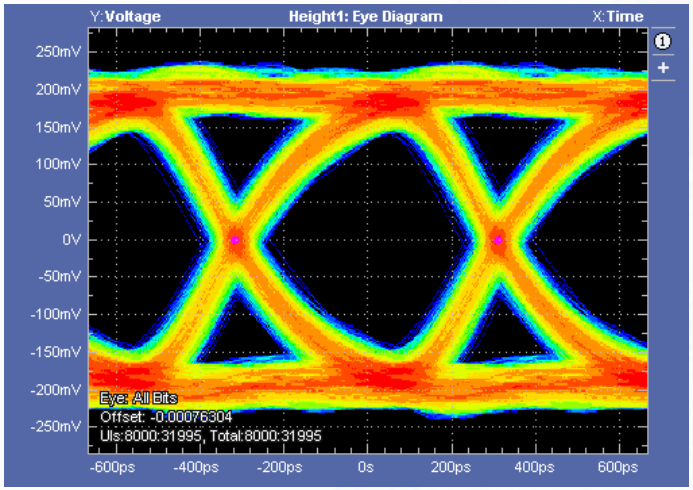
DHP – Output link – CML 1.6 Gbit/s

UNIVERSITÄT BONN

Preemphasis Off

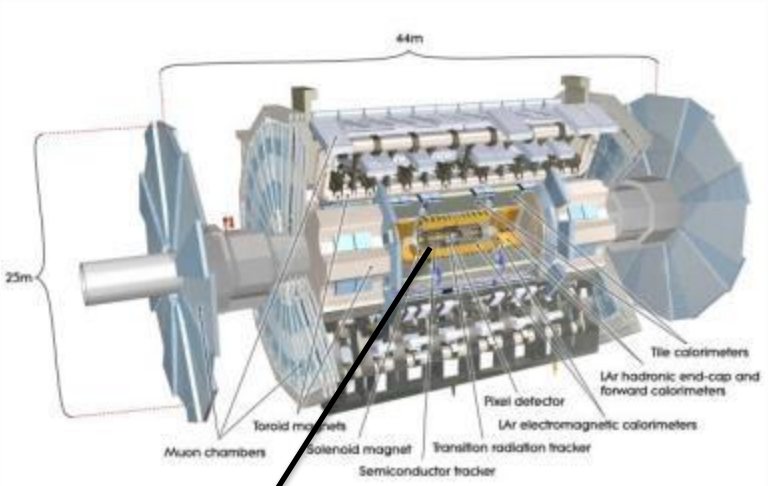


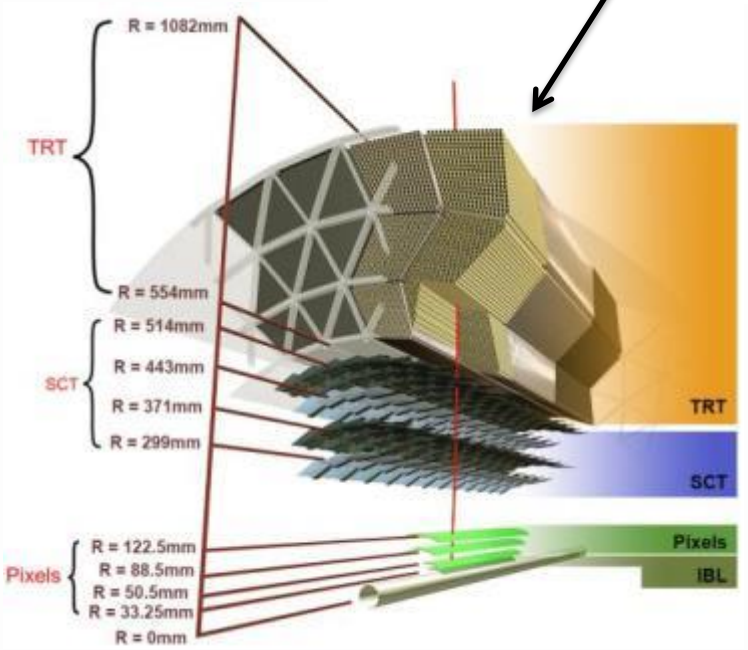
Preemphasis On

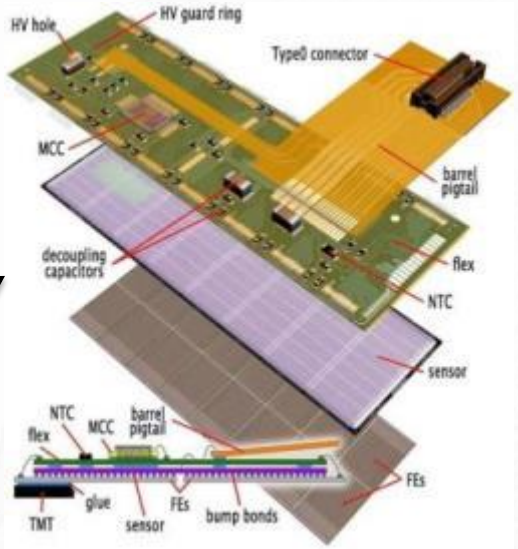


20m cables

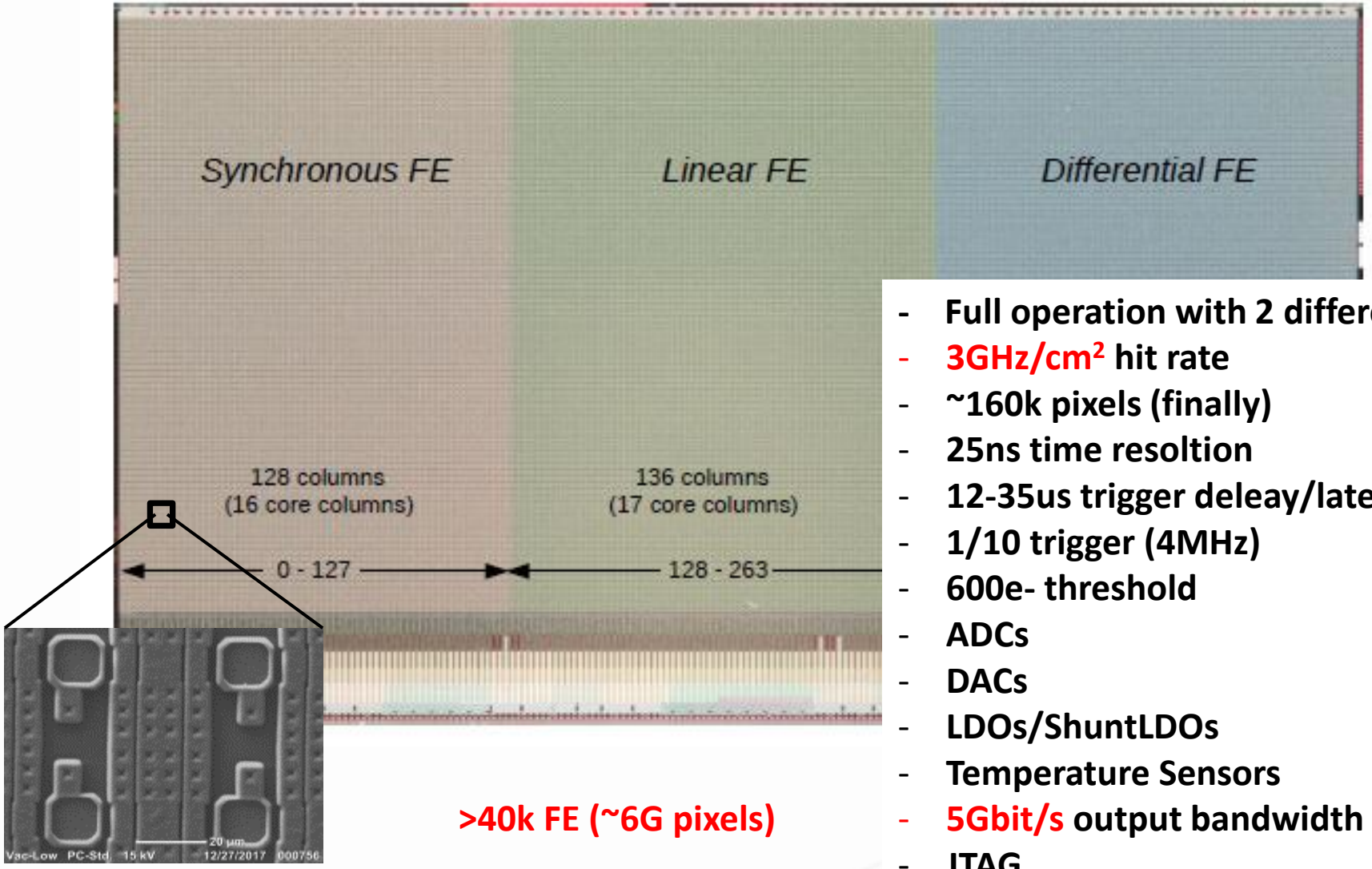
ATLAS @ LHC


44m
25m
Muon chambers
Toroid magnets
Solenoid magnet
Transition radiation tracker
Semiconductor tracker
Pixel detector
LAr electromagnetic calorimeters
LAr hadronic end-cap and forward calorimeters
Tile calorimeters


R = 1082mm
TRT
R = 554mm
R = 514mm
R = 443mm
SCT
R = 371mm
R = 299mm
TRT
SCT
Pixels
IBL
R = 122.5mm
R = 88.5mm
Pixels
R = 50.5mm
R = 33.25mm
R = 0mm


HV hole
HV guard ring
Type0 connector
MCC
barrel pigtail
decoupling capacitors
flex
NTC
sensor
NTC
flex
MCC
barrel pigtail
FEs
TMT
glue
sensor
FEs
bump bonds

RD53A – ATLAS/CMS Prototype

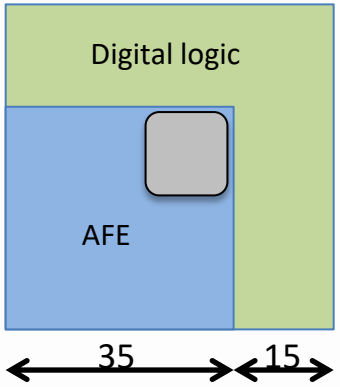


>40k FE (~6G pixels)

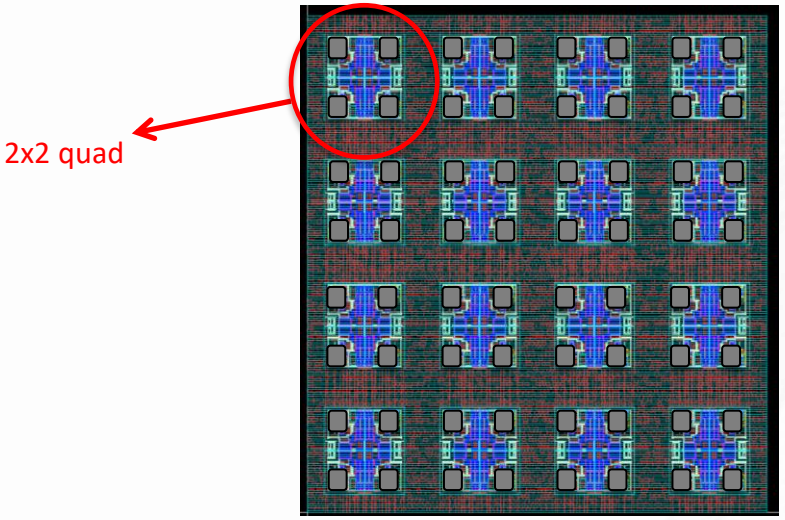
- Full operation with 2 differential lines
- 3GHz/cm^2 hit rate
- ~160k pixels (finally)
- 25ns time resoltion
- 12-35us trigger deleay/latency
- 1/10 trigger (4MHz)
- 600e- threshold
- ADCs
- DACs
- LDOs/ShuntLDOs
- Temperature Sensors
- 5Gbit/s output bandwidth
- JTAG
- …

50µm X 50µm Pixel floorplan

1) 50% Analog Front End (AFE)
   50% Digital cells



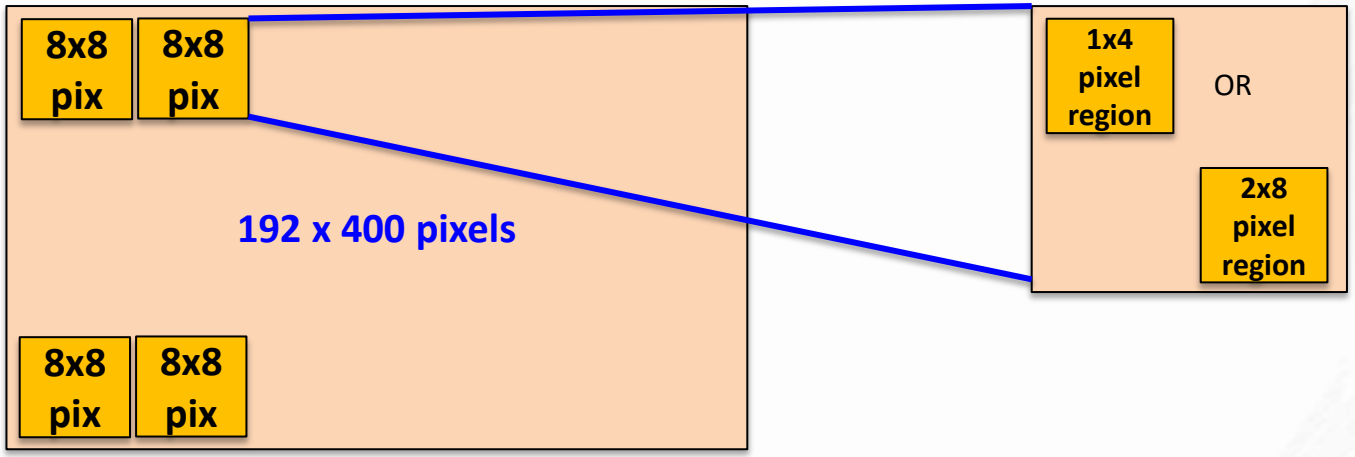
2) The pixel matrix is built up of 8 x 8 pixel cores → 16 analog islands (quads) embedded in a flat digital synthesized sea



3) A pixel core can be simulated at transistor level with analog simulator
4) All cores (for each FE flavour) are identical → Hierarchical verifications

Pixel array logic organization

basic layout unit: **8x8 digital Pixel Core → synthesized as one digital circuit**

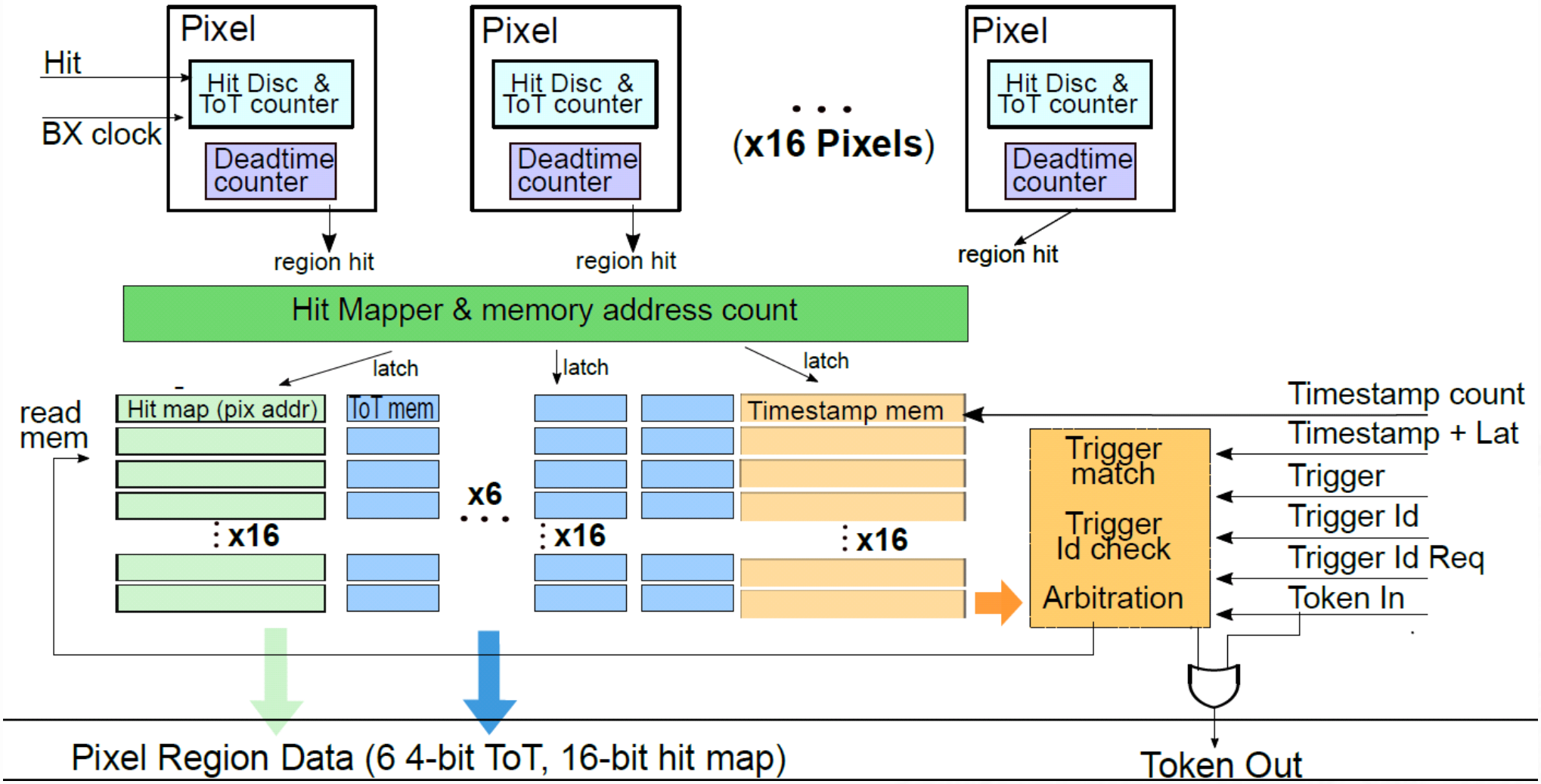


- **One Pixel Core** contains **multiple Pixel Regions (PR) and** some **additional arbitration and clock logic**
- **Pixel Regions share most of logic and trigger latency buffering**

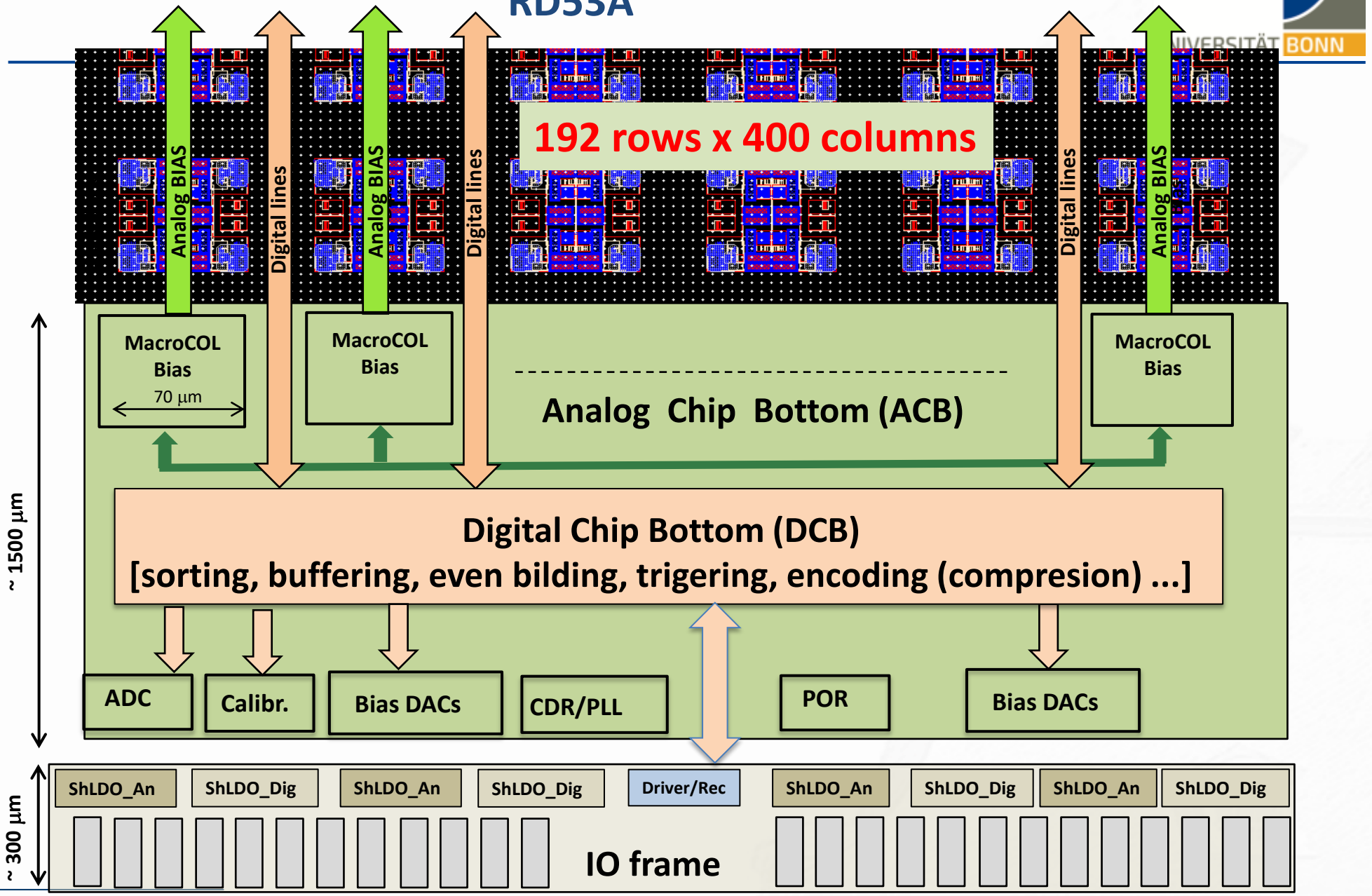
Distributed Buffering Architecture (DBA) :
- Distributed TOT storage (in pixel)
- Integrated with Lin and Diff FE

Centralized Buffering Architecture (CBA)
- Centralized TOT storage (in region)
- Integrated with Sync FE (Fast ToT)

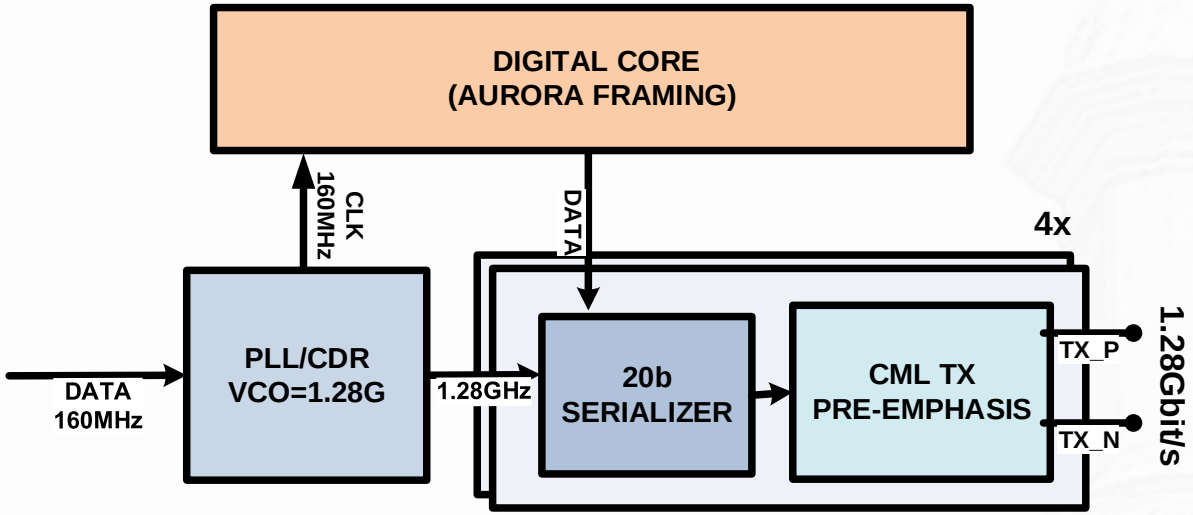
RD53A - Centralized Buffer Architecture - 2x8



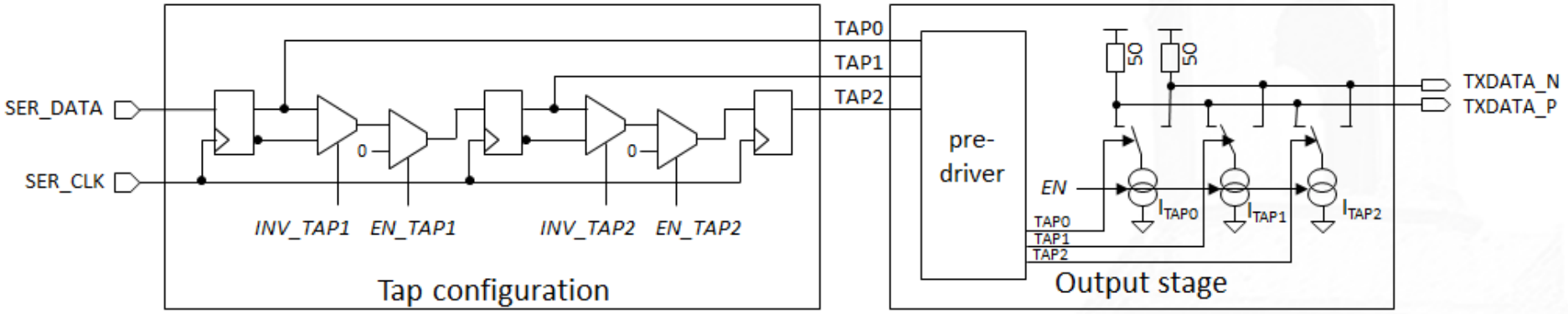
RD53A


192 rows x 400 columns
Analog BIAS
Digital lines
Analog BIAS
Digital lines
Digital lines
Analog BIAS
MacroCOL Bias
70 µm
MacroCOL Bias
MacroCOL Bias
Analog Chip Bottom (ACB)
~ 1500 µm
Digital Chip Bottom (DCB)
[sorting, buffering, even bilding, trigering, encoding (compresion) ...]
ADC
Calibr.
Bias DACs
CDR/PLL
POR
Bias DACs
~ 300 µm
ShLDO_An
ShLDO_Dig
ShLDO_An
ShLDO_Dig
Driver/Rec
ShLDO_An
ShLDO_Dig
ShLDO_An
ShLDO_Dig
IO frame

RD53A – Ouput Link



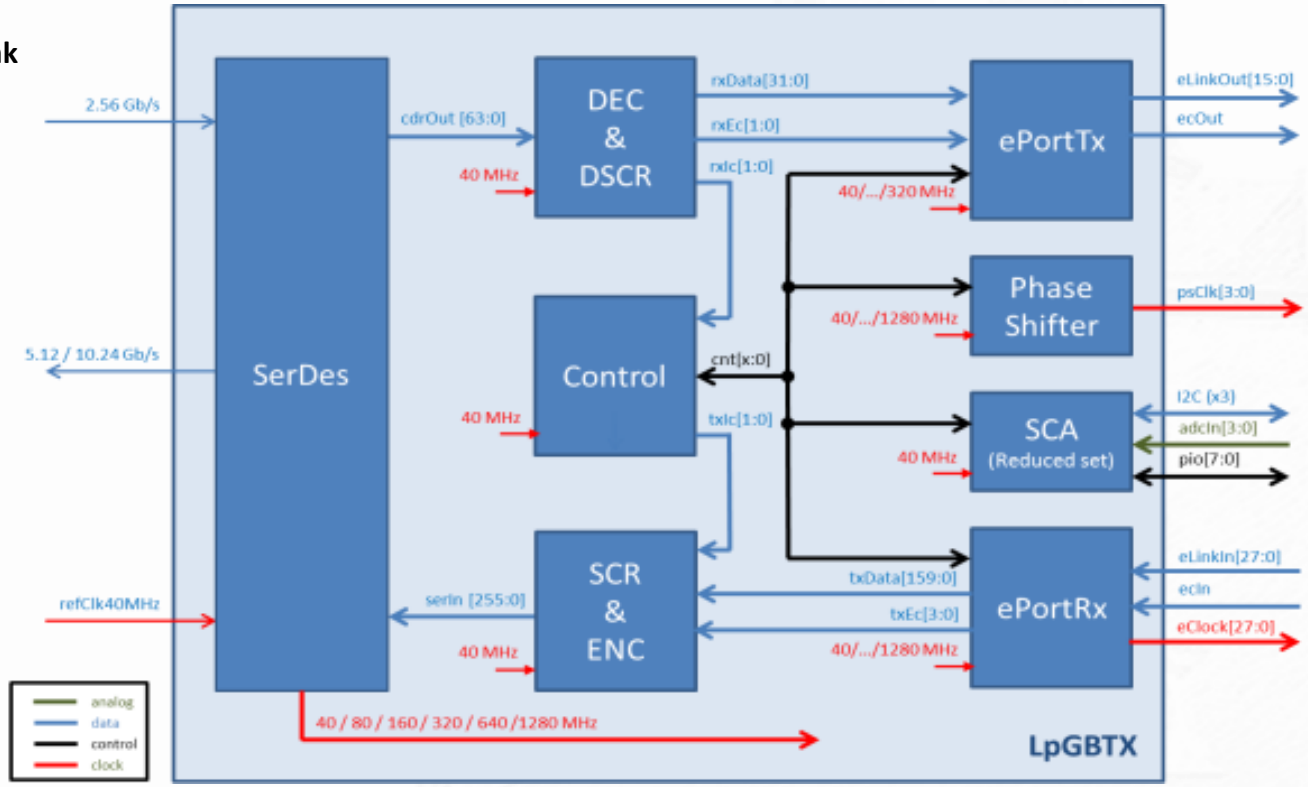
Configurable 3-tap pre-emphasis filter:



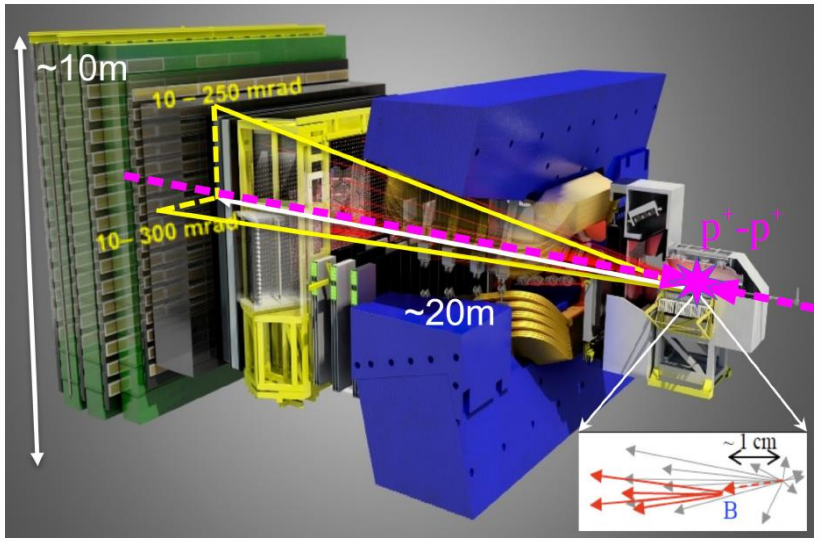
- **Cable limited**
- **Fiber -> radiation**

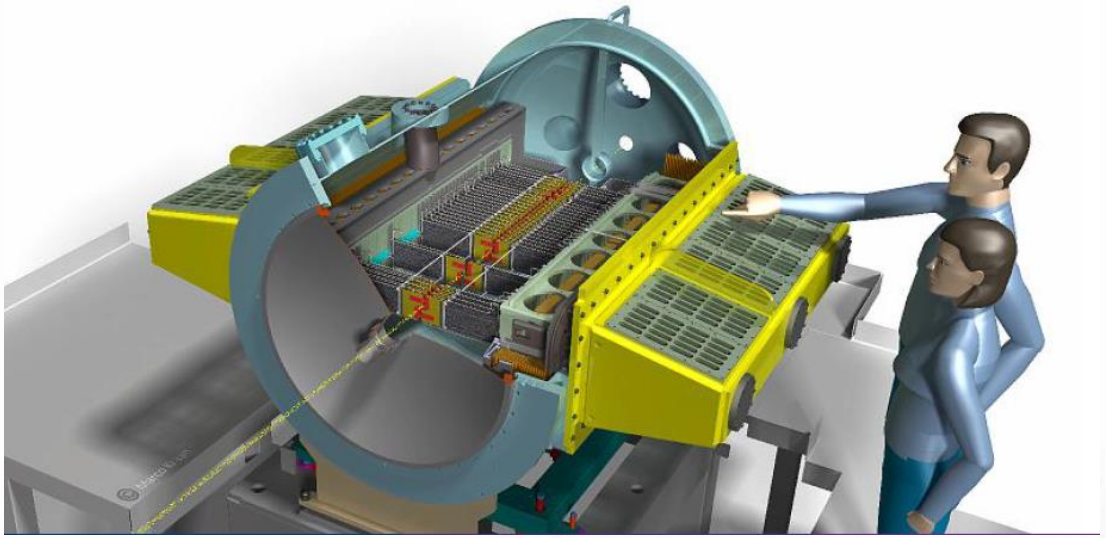
The Low Power GBTX (LpGBTX)

- Low Power Dissipation and Small Footprint:
  - **Target: 500 mW**
- Bandwidth:
  - **Low-Power mode**
    - **2.56 Gb/s for the optical down link**
    - **5.12 Gb/s for the optical up link**
  - **High-Speed mode:**
    - **2.56 Gb/s for the optical down link**
    - **10.24 Gb/s for the optical up link**

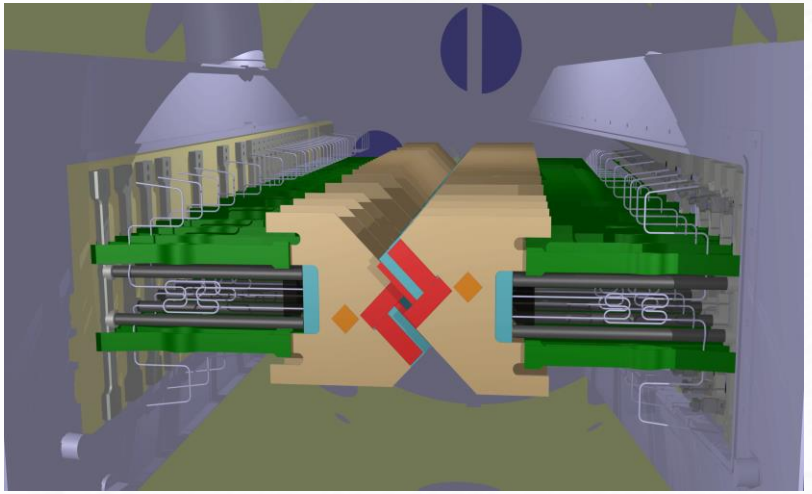


VeloPix - LHCb



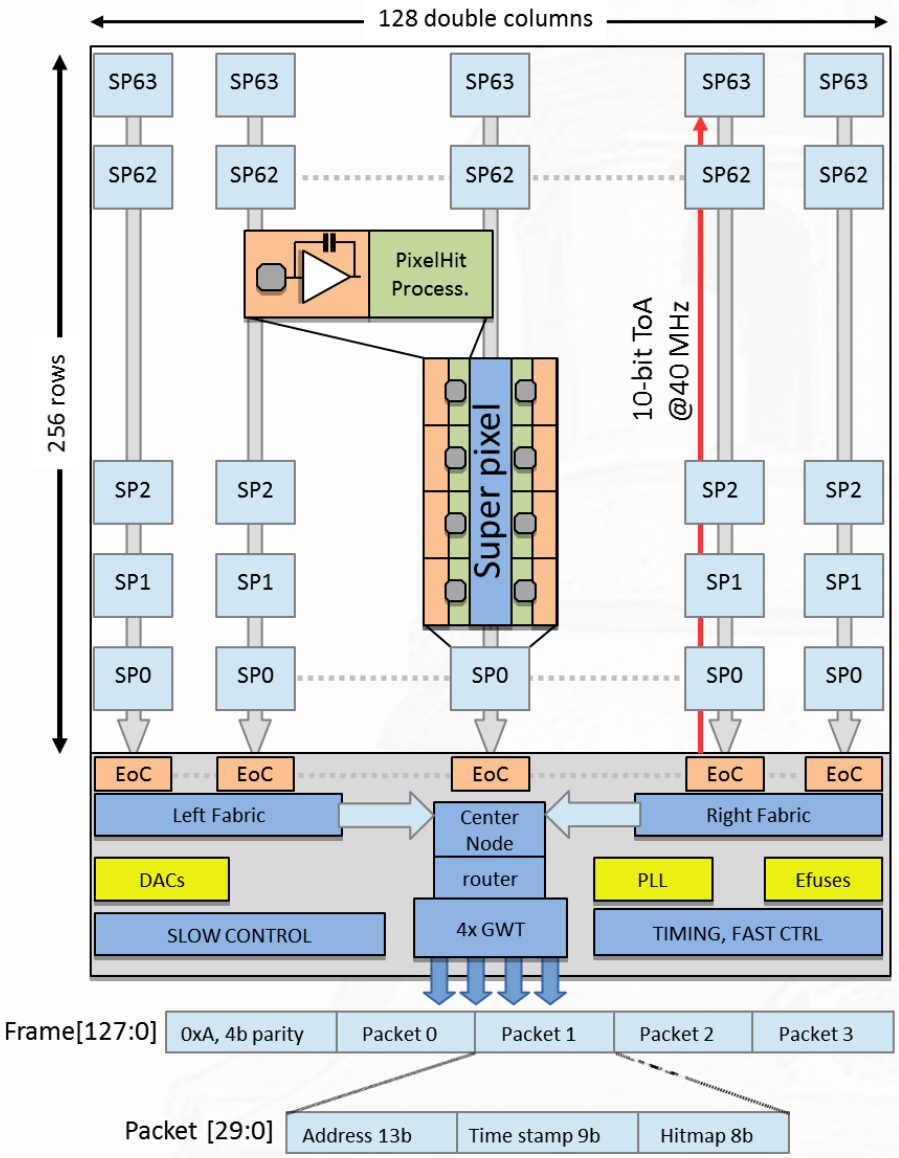


- Vertex detector surrounding collision region
  - In vacuum
  - Close to the beam: 5.1 mm
- From silicon strips to pixels
- New R/O chip VeloPix, derived from Timepix3
- In total 624 ASICs, ~41 Mpixels
- Trigger-less readout (~2.9 Tbits/s)

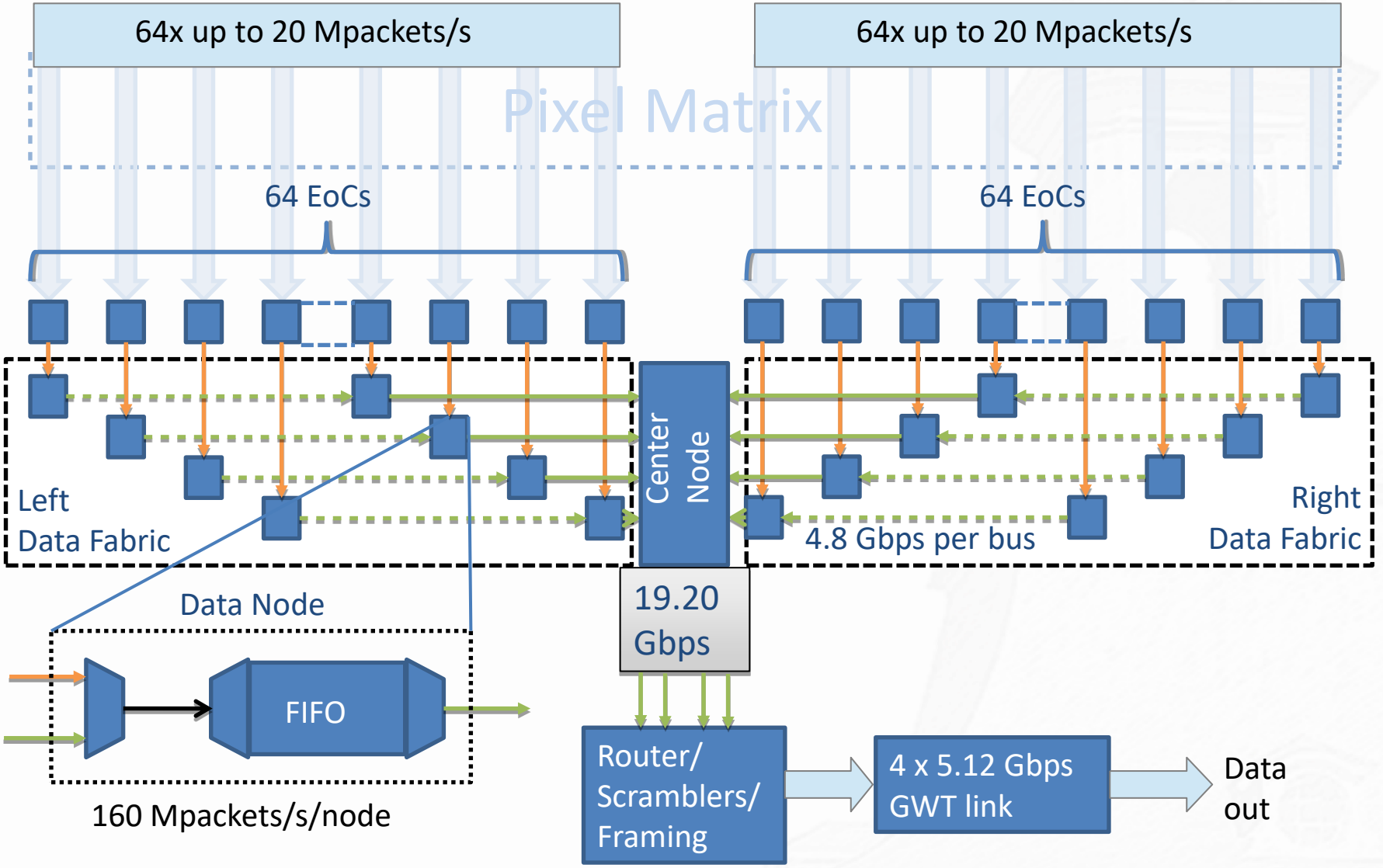


Examples – VeloPix

- Pixel matrix:
  - 256 x 256 pixels
  - 128 x 64 super pixels (2x4 pixels each)
  - @40MHz
- Packet-based architecture:
  - 8 pixels/packet + 9 bit time stamp → 30% reduction in data rate
- Data-driven readout:
  - 20 Mpackets/s/double column
- 40, 80, 160 and 320 MHz TMR clock domains in the periphery
- 1 to 4 configurable serializers (GWT)
- Similar to the GBT frame

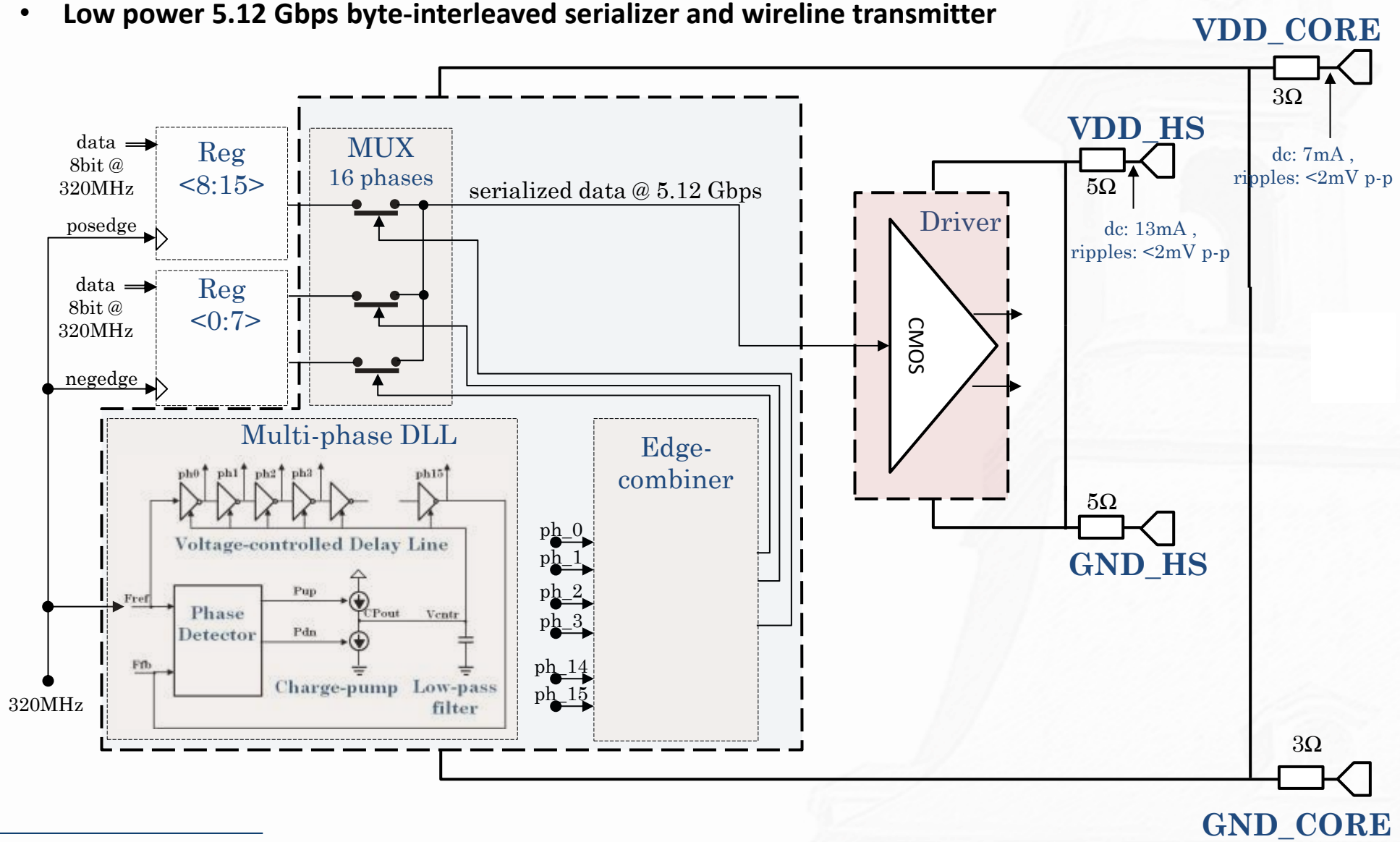


Periphery data path

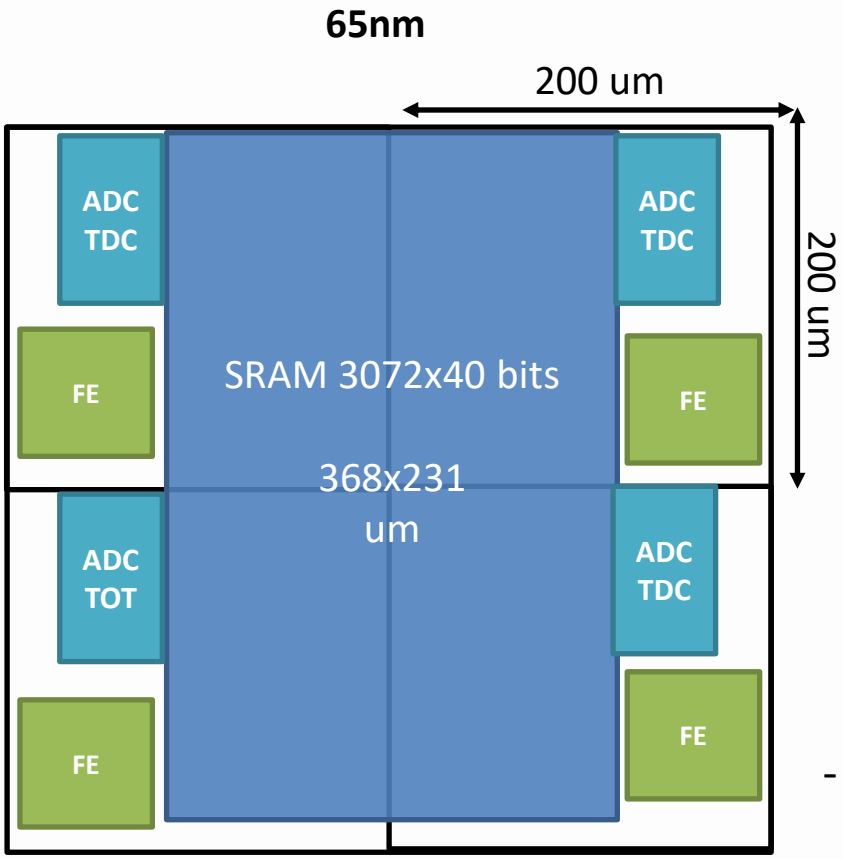


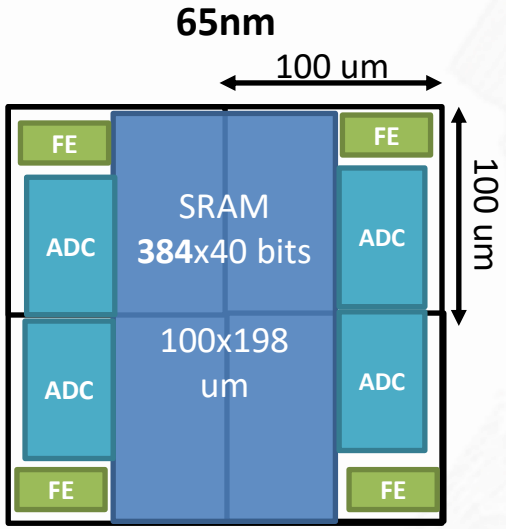
VeloPix - GWT

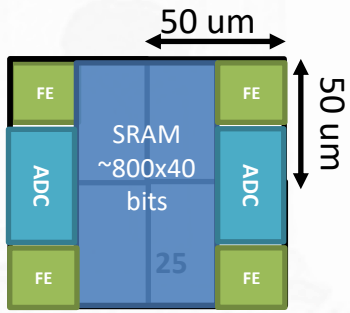
- **Low power 5.12 Gbps byte-interleaved serializer and wireline transmitter**



Example possibilities







- Integrate storage and data processing it single pixels:
  - pattern recognition
  - histogramming (in pixel spectral analysis)
  - conversion to photons -> compression
  - clustering and subpixel counting (COG)
  - infinite* dynamic range

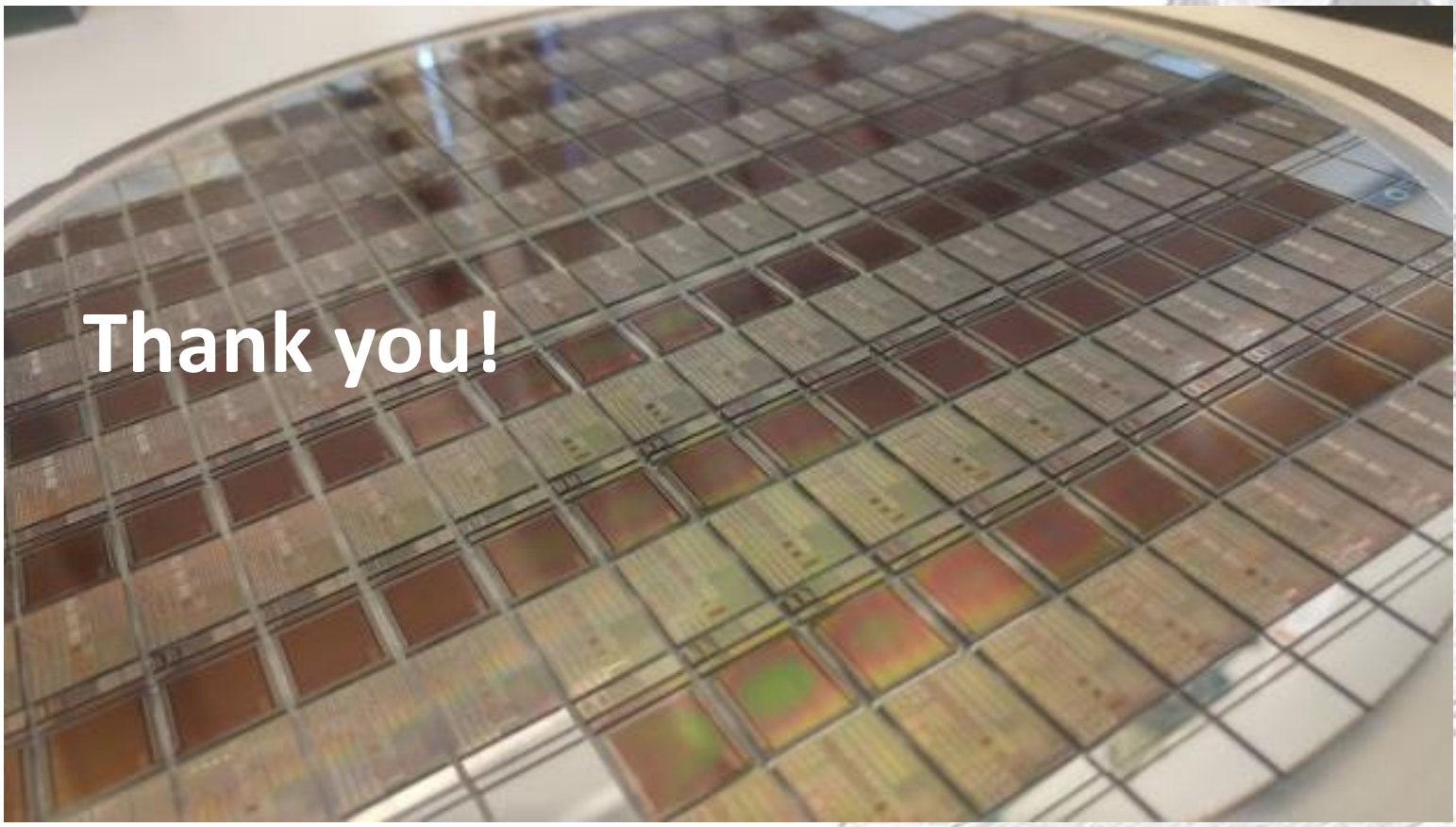
Conclusions

UNIVERSITÄT BONN

- Move from digitally assisted analog design to analog assisted digital
- Chips are more complex with lot of memory and data processing
- New tools for design
- Different type of verification (mistakes are very expensive)
- High speed serial communication
- Lot of opportunities in exploring small feature size

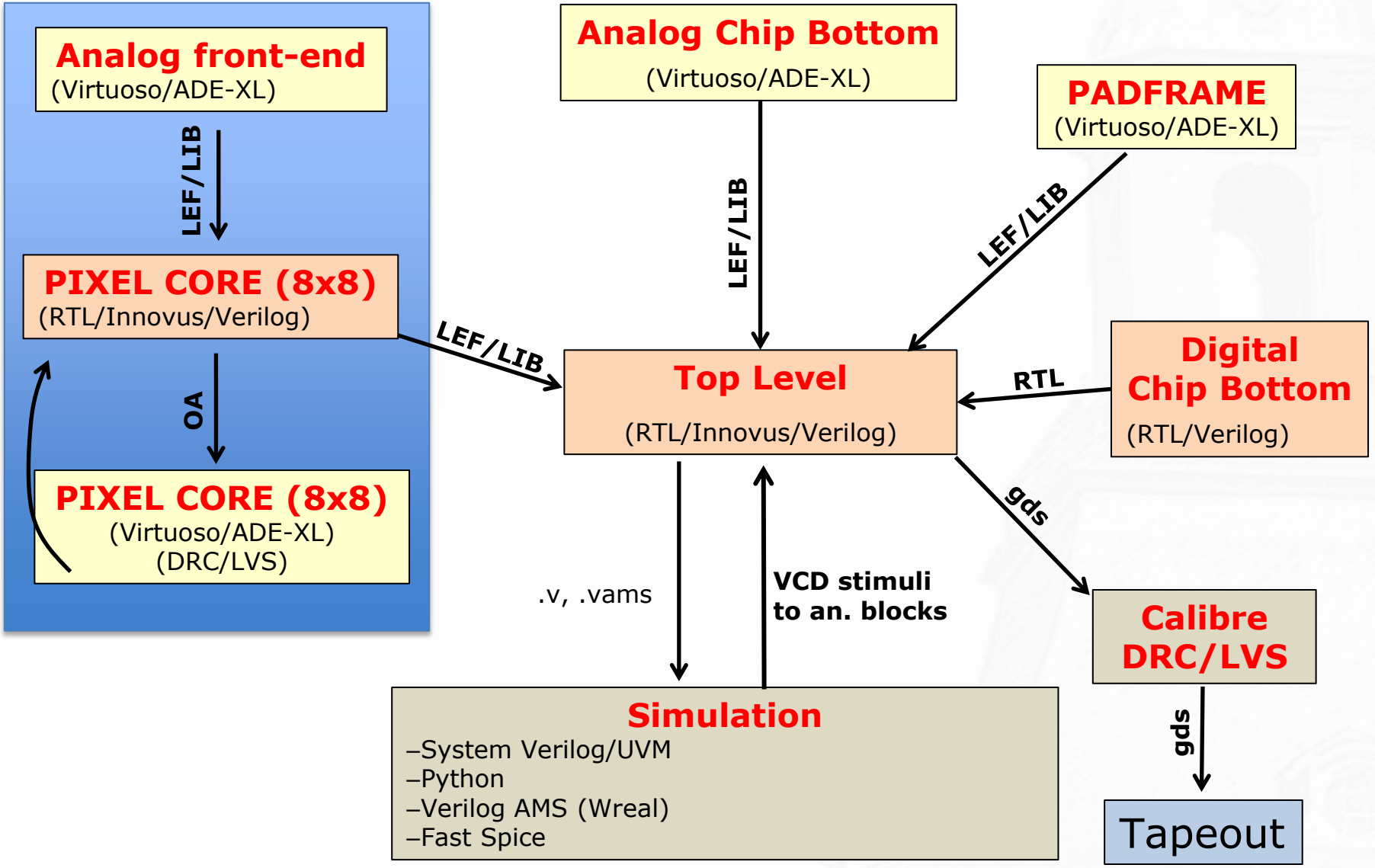
UNIVERSITÄT BONN

SI LAB
Silizium Labor Bonn

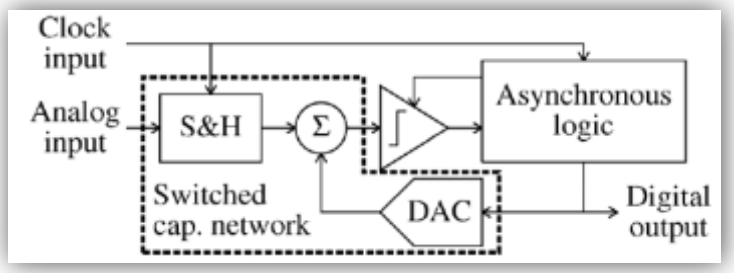


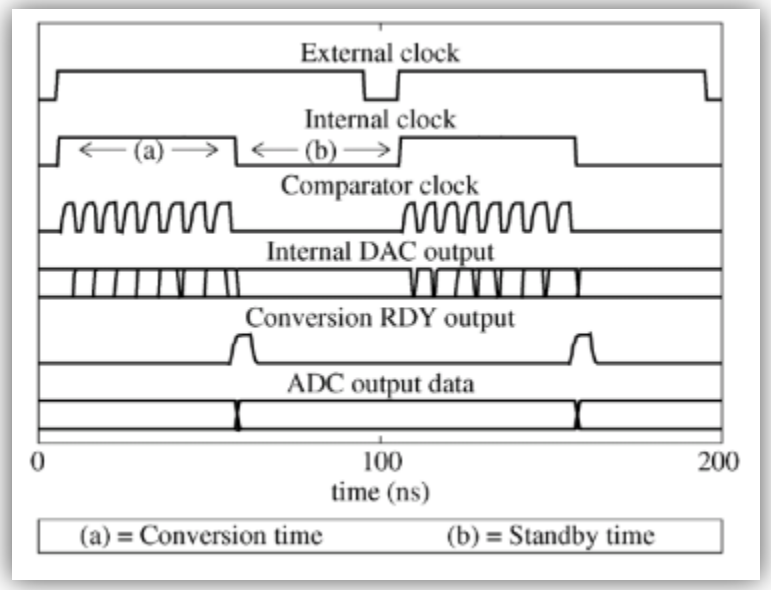
Thank you!

Design Flow – from analog perspective

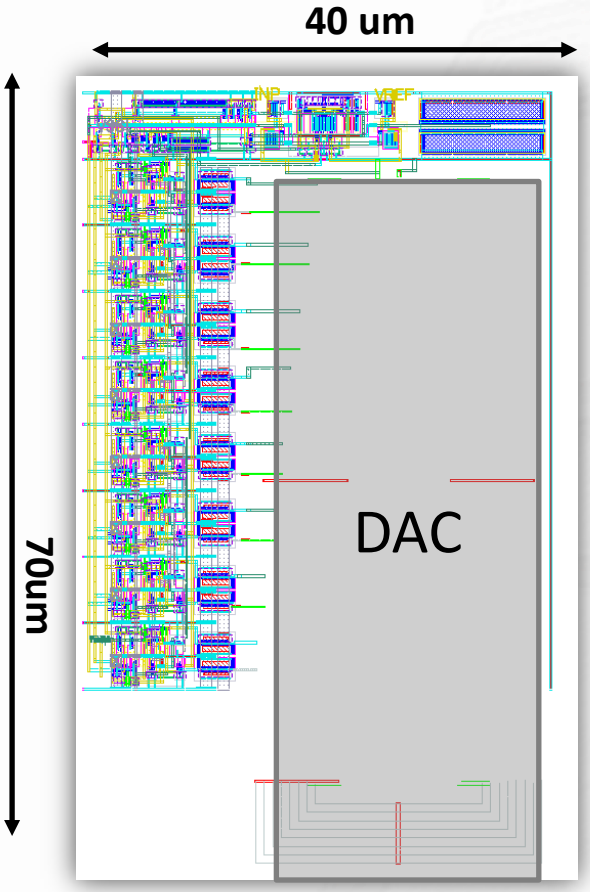


SAR ADC IN 65nm - Layout





Only external sample signal needed!

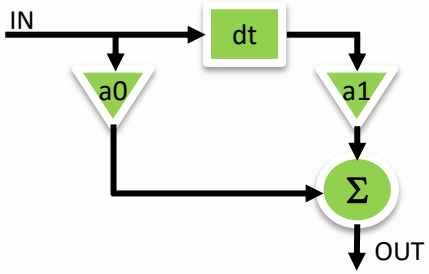


Layout is not area optimal
Possible de-cup under DAC?

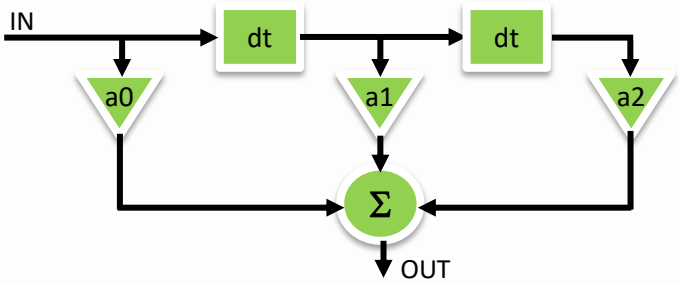
Pre-emphasis

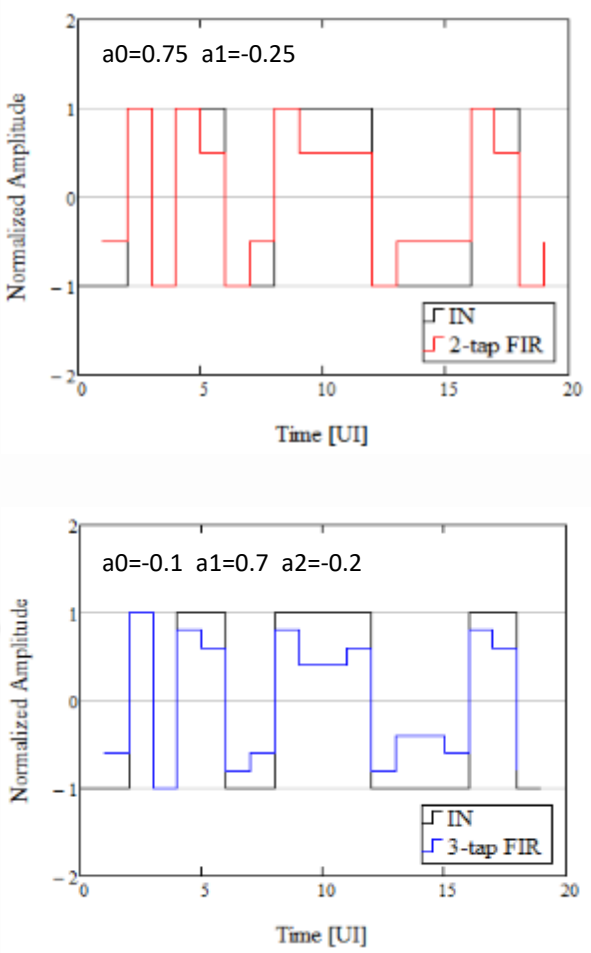
Implementation with a digital filter (FIR filter)

- **2-tap FIR**

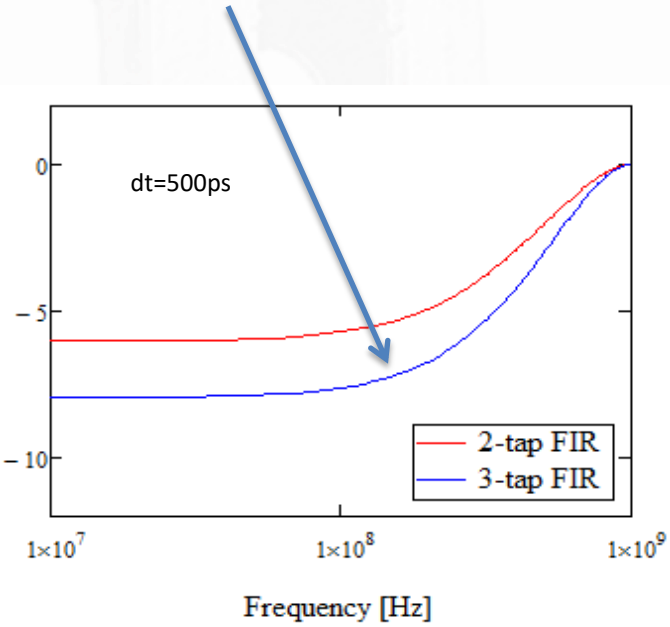


- **3-tap FIR**



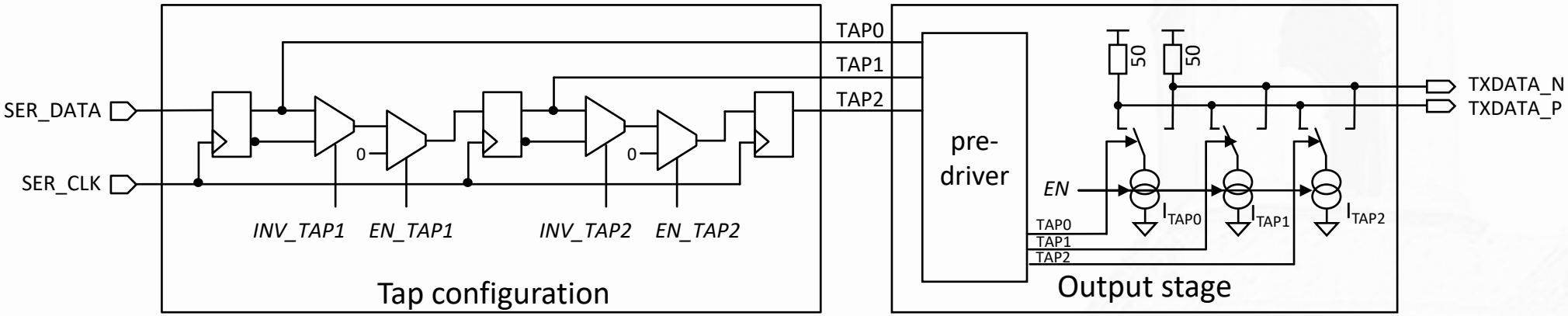


3 taps allow steeper transfer function compared to 2 tap



CML Cable Driver Implementation (RD53A)

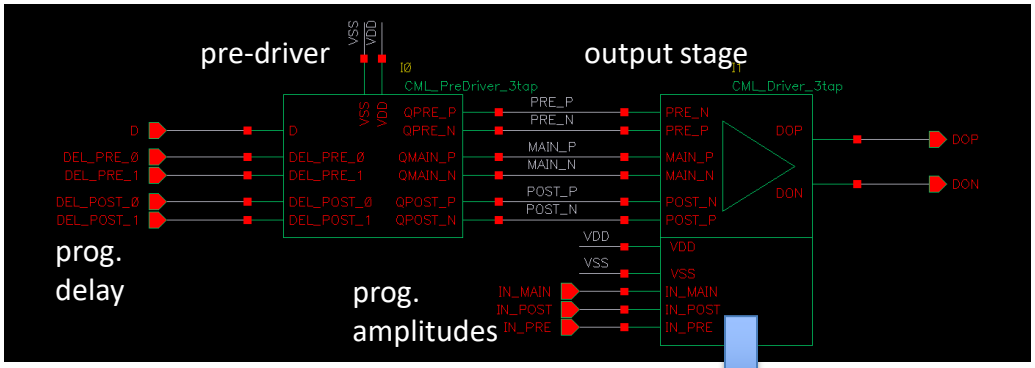
Configurable 3-tap pre-emphasis filter

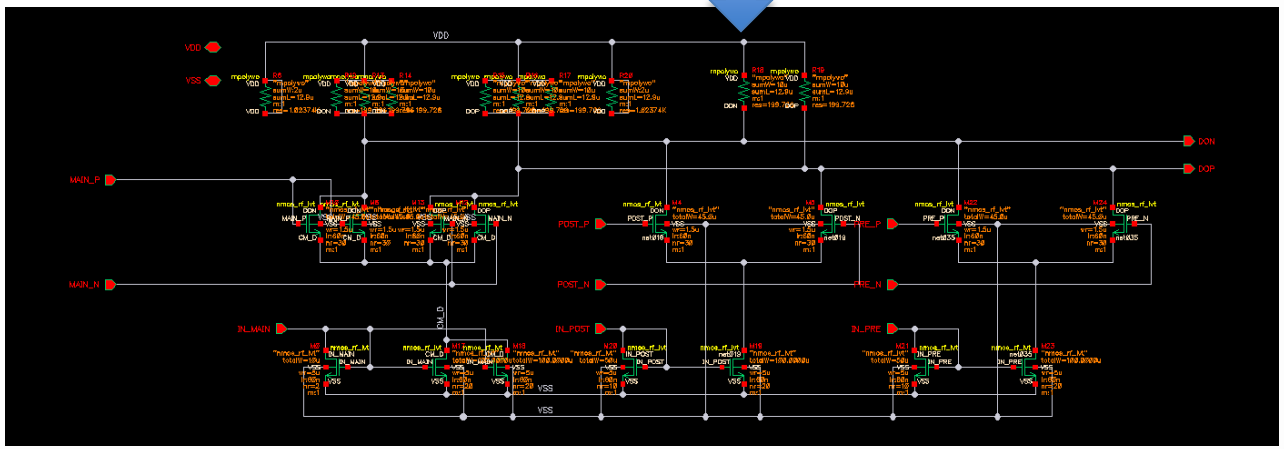


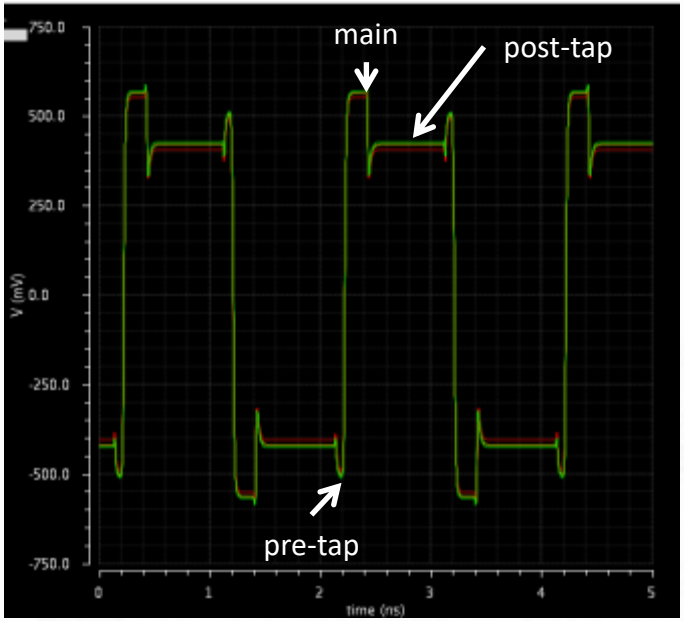
TAP configuration
- *INV_TAP[2:1]*
- *EN_TAP[2:1]*

CML output configuration
- *EN*
- *TAP0_BIAS[9:0]*
- *TAP1_BIAS[9:0]*
- *TAP2_BIAS[9:0]*

CML Cable Driver Implementation (RD53A)



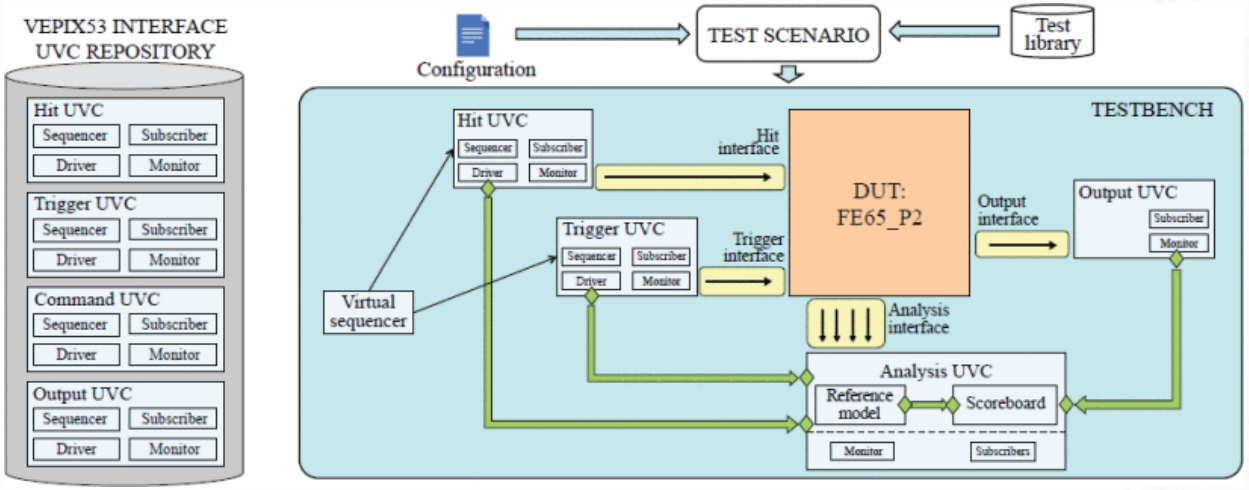


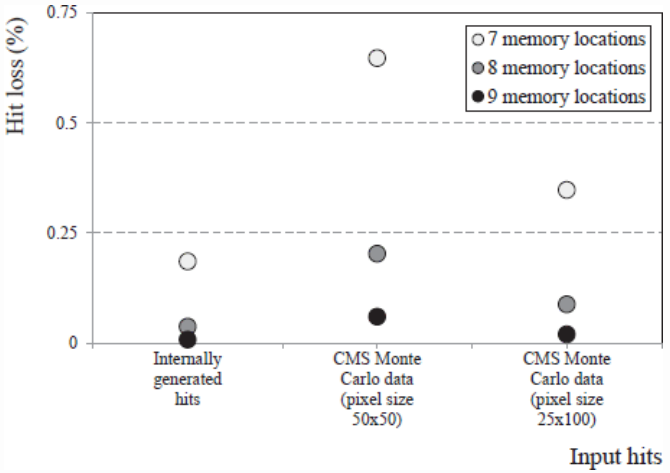


CML_TX simulation

- "no-load"
- pre- and post-tap active
- DEL_POST= 3, DEL_PRE = 0)
- IN_MAIN bias =[3, 4, 5 mA)

Modeling – UVM/Verilog

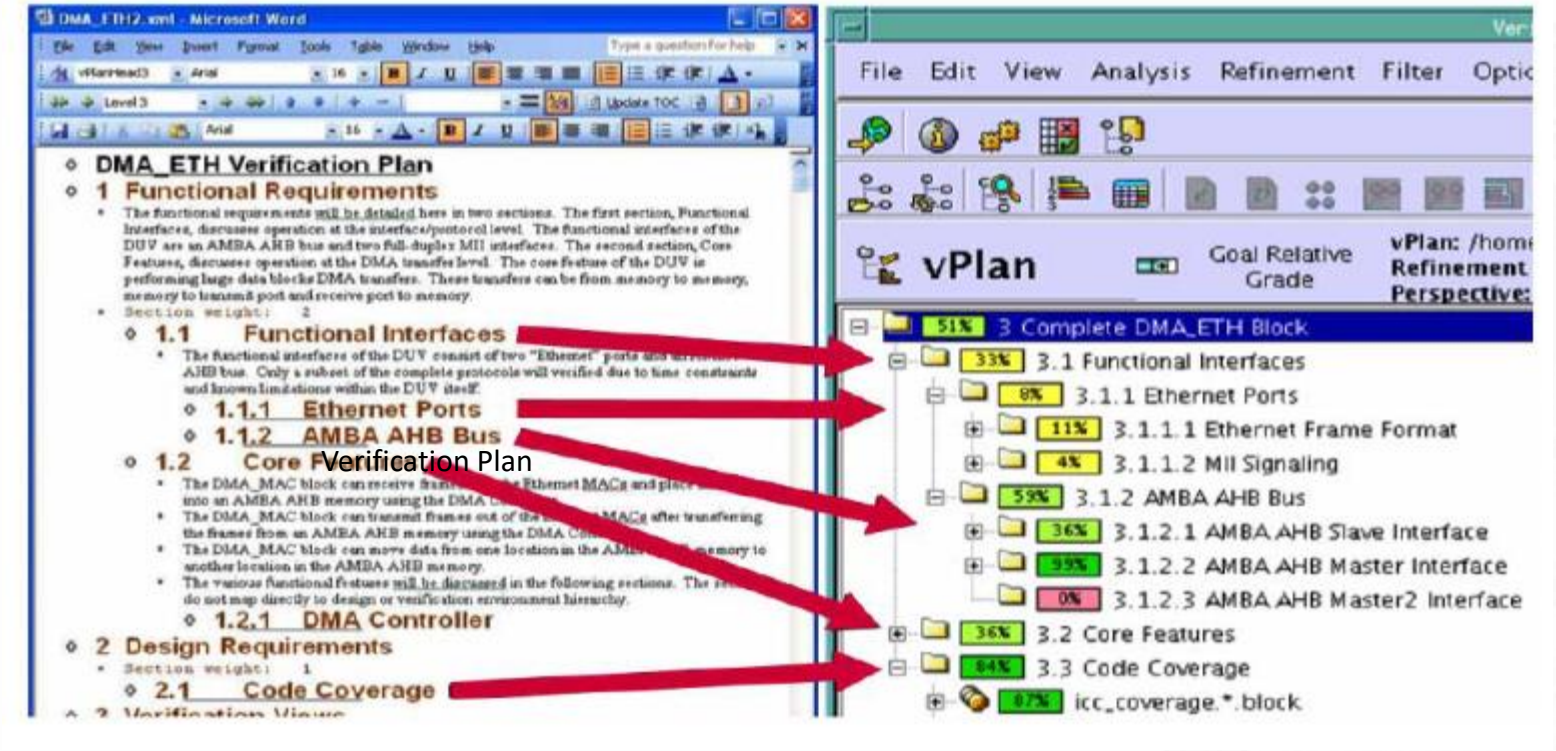




A lot of work. Can be reused for verification.

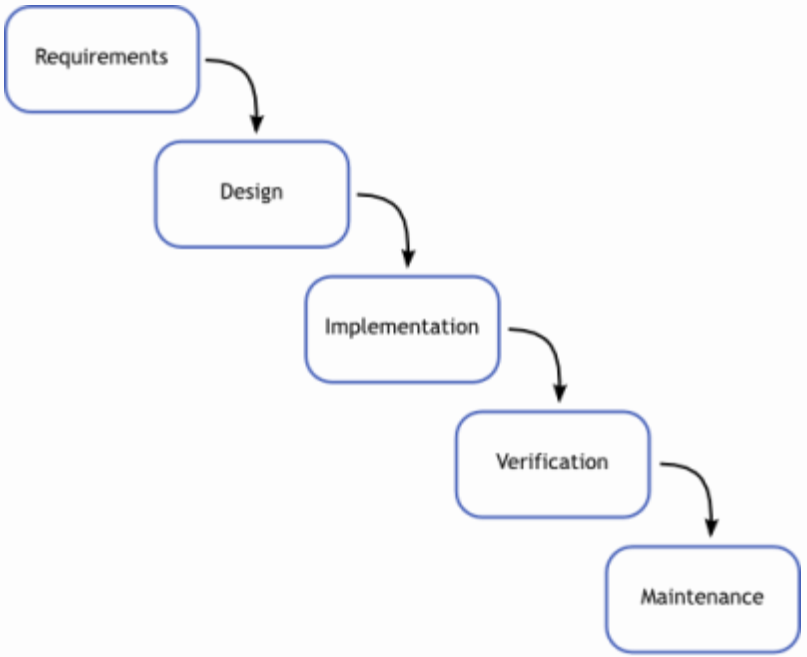
http://ieeexplore.ieee.org/document/8069646/

Verification Plan

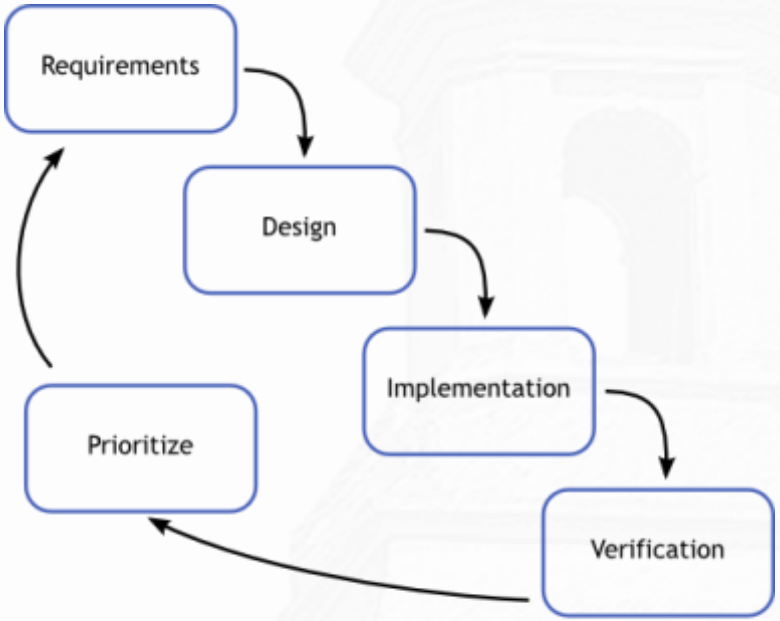


Most important part.

Waterfall vs Agile



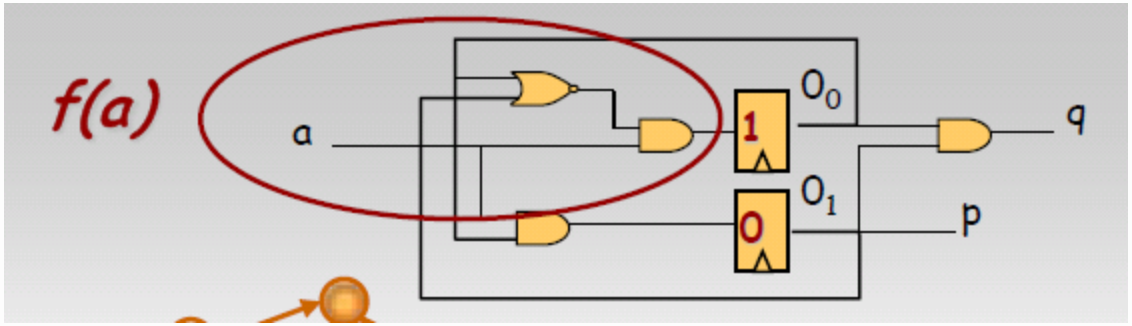
Waterfall model

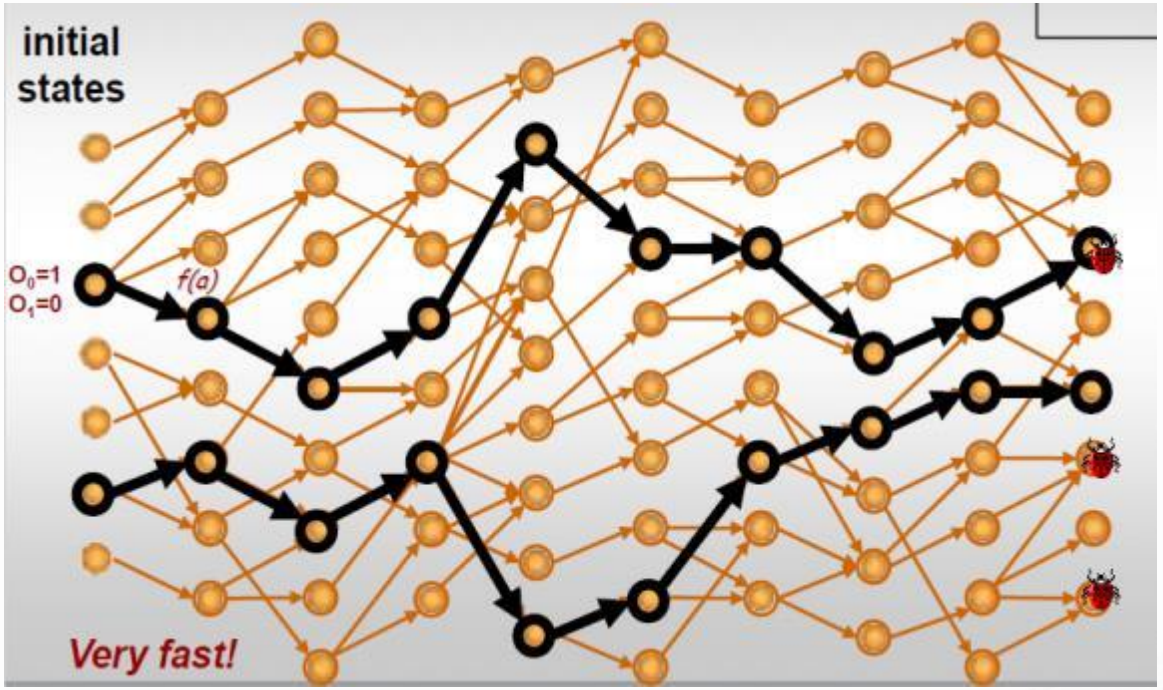


Agile development interpreted in the waterfall model

Changing and unclear specifications?

Verification - Formal



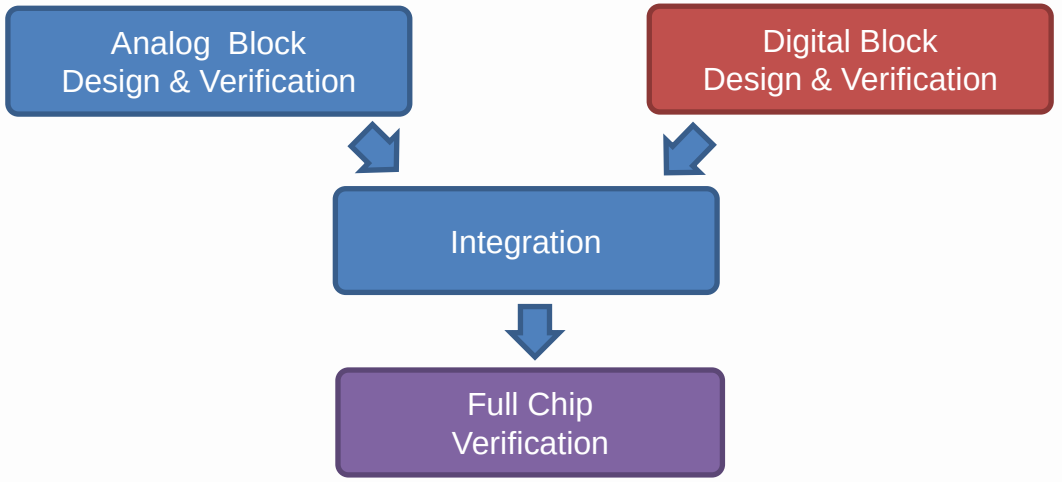



```
// SystemVerilog Assertion

property p_arb;
   @(posedge clk)
      req |=> ##[0:2] gnt;
endproperty
assert property (p_arb);
```

Single Event Upsets …

Design Flow

- Design in “big A small D” methodology
- Blocks designed and verified individually
- Full chip digital and mixed-signal verification
- Work synchronization with integrated Revision Control System
- Big chip = many difficulties with software and PDK!



System-Level Design	Functional Design and Verification	
	Chip Planning	
Block-Level Design	RTL Design and Verification	Design and Analysis
	Synthesis and Verification	Circuit Simulation
	Place and Route	Custom Layout
	DRC, LVS, RCX	
Chip Assembly	Chip Assembly	
Physical Verification	Full Chip Physical Verification, Extraction, and Analysis	
System Verification	Full Chip System-Level Verification Analog, Digital, RF	