Characterization of Digital Cells

X. Llopart

Outline

- Introduction to cell characterization
- Liberate tool (Cadence)
- Characterization example using a combinatorial cell (AND2)
- Applications in HEP

Disclaimer…

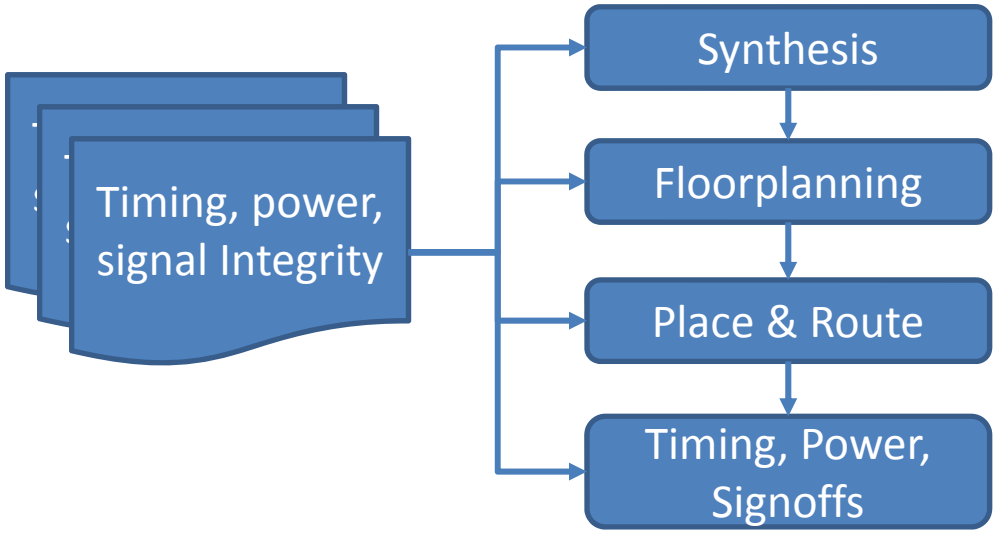
- My origins are full custom analog designer
- This presentation is based in my acquired knowledge using ELC and Liberate tools… but other tools are available
- My objectives are:

  1) Transfer the advantages of using a Library Characterizer tool

  2) Change the way we can design chips → towards more digital top designs

Objective of Cell Characterization

- Create a set of high quality models of a standard cell library that accurately and efficiently model cell behaviour
- This set of models are used by several different digital design tools for different purposes



Digital flow in deep submicron technologies

- Nanometre geometries (<65nm) require an increase number of library views:
  - Issues related to leakage power and process variation

Nanometer Requirements	130 nm	65 nm	28 nm
PVT	3 (fast,typ,slow)	3 (fast,typ,slow)	3 (fast,typ,slow)
Leakage	n/a	3 Vths	3 Vths
Voltage Scaling	n/a	2 Voltages	2 Voltages
Temperature	n/a	n/a	2 Temperatures
Yield	n/a	n/a	2 Yield
Total Views	**3**	**18**	**72**

Liberate reference manual

What is a Library Characterizer?

- Creates electrical views (timing, power and signal integrity) in industry standard formats such as Synopsys Liberty (.lib) format

- Normally it requires only the foundry device models and the extracted cell netlists from which it will create all the required electrical views

- By automating the process for generating views, it ensures that the library's functional, timing, power and signal integrity values are both accurate and complete thus avoiding potential chip failures caused by missing or bad library data

Library Characterizer packages

- The tools needed are:
  - Analog simulator (Hspice, Spectre,…)
  - Netlist of the cells (parasitics extracted)
  - Device models from the technology vendor
  - Timing arcs
  - … and a lot of careful simulations → CERN 250nm IBM rad-hard library

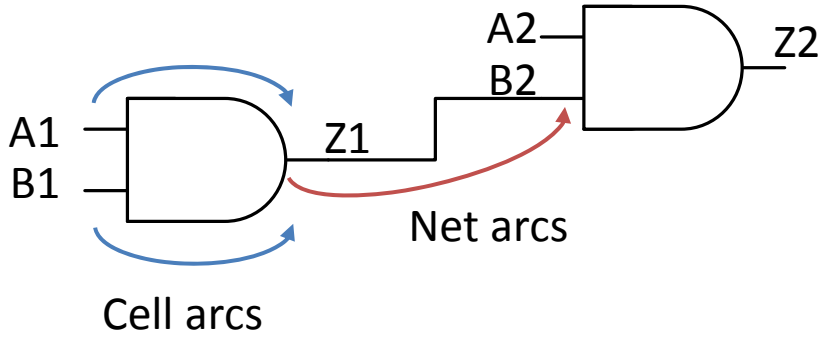
- Many vendors provide today packages (bunch of scripts) which automatizes the process:
  - Kronos (Mentor Graphics)
  - Synopsys SiliconSmart
  - Silvaco
  - ELC
  - ALTOS
  - Liberate (Cadence)

NLDM, CCS and ECSM formats

- NLDM: non-linear delay model
  - Input transition vs capacitive load time table
  - Valid if technology >=130nm
- New physical effects (<130nm) mean that the NLDM is no longer accurate enough:
  - Input capacitances have become more complex
  - Interconnect net impedance dominates cell delay
  - Voltage drop across the device can have a significant effect on delays
- CCS → Composite current source (ccsn, ccsp)
- ECSM → Effective current source model (ecsmn, ecsmp)

Timing Arcs

- Timing arc is a component of a timing path
- Types of arcs:
  - Cell Arcs: Between an input pin and output pin of a cell
    - Defined in Liberty files (.lib)
  - Net Arcs: Between driver pin of a net and load pin of a net
    - Calculated inside P&R tool → requires parasitic RC technology information
- Cell Arcs:
  - Combinatorial cells (INV, NAND…)→ IO arcs
  - Sequential cells (DFF, LAT…)→ Edge sensitive (clk↑, clk↓), Reset,…

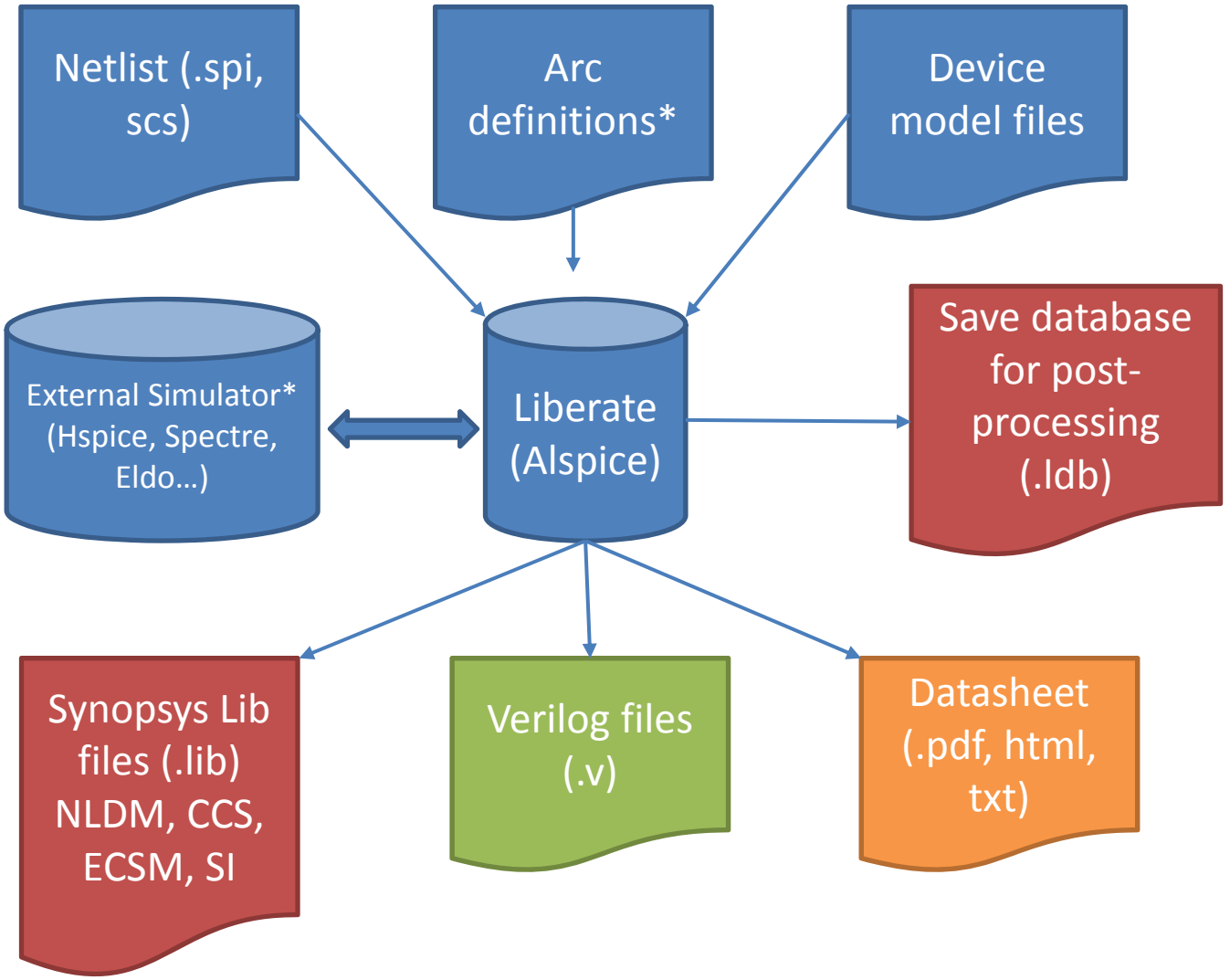


Timing Cell Arc Concepts

- Timing arcs can be delay arcs or constraint arcs
- Each timing arc has a startpoint and an endpoint:
  - The startpoint can be an input, output, or inout pin
  - The endpoint is always an output pin or an inout pin
- The only exception is a constraint timing arc, such as a setup, hold, recovery or removal constraint between two input pins
- related_pin: This attribute defines the pin or pins representing the startpoint of a timing arc

- Timing arc generation and definition must be understood in case cell recognition is not achieved

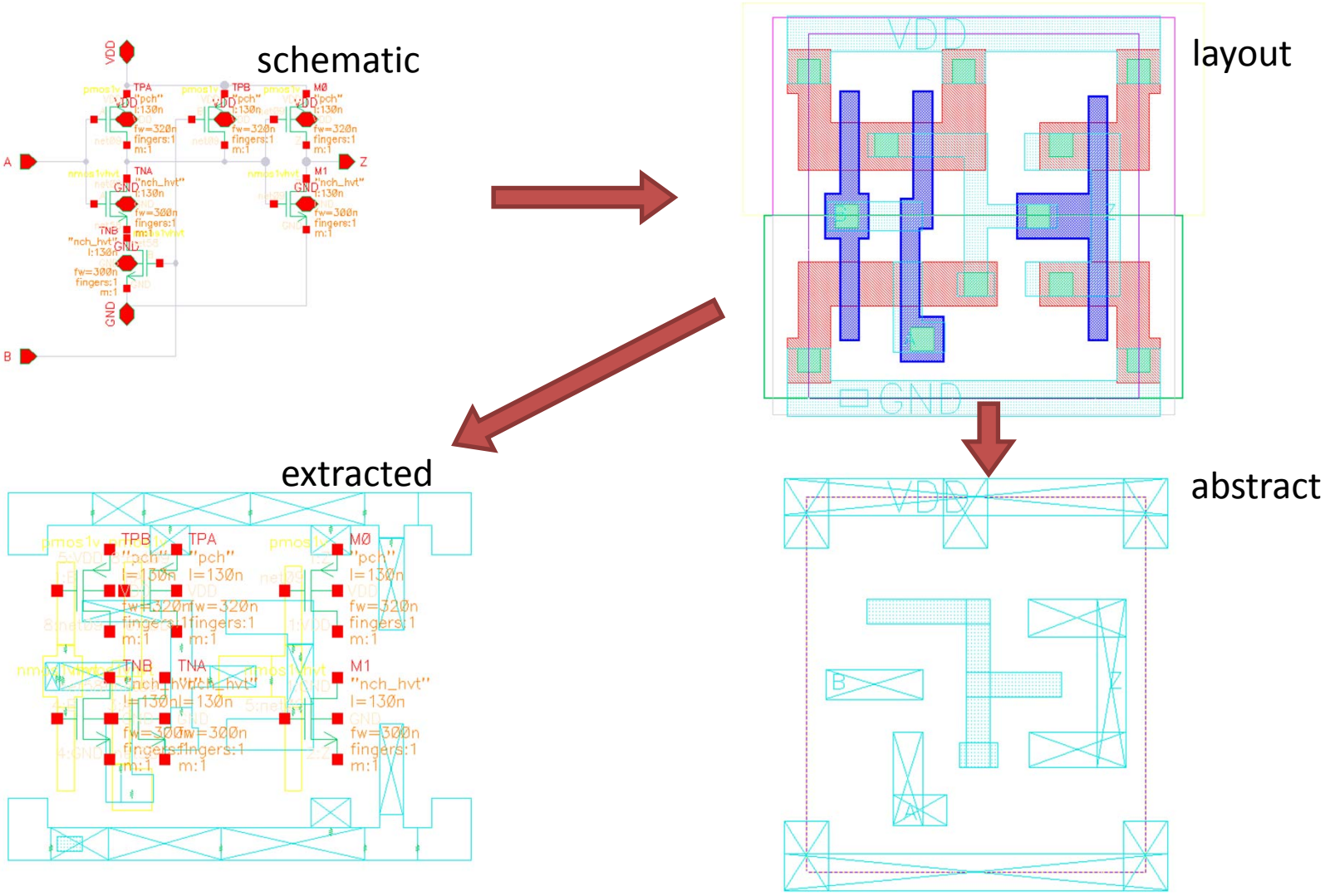
Liberate


Netlist (.spi, scs)
Arc definitions*
Device model files
External Simulator* (Hspice, Spectre, Eldo…)
Liberate (Alspice)
Save database for post-processing (.ldb)
Synopsys Lib files (.lib) NLDM, CCS, ECSM, SI
Verilog files (.v)
Datasheet (.pdf, html, txt)

Liberate (Cadence)

- ELC (Encounter Library Characterizer) not available since January 2015
- Altos renamed to Liberate on 2014
- Europractice includes the Liberate package (Altos until last year):
  - Liberate: Standard cell and complex I/O characterization
  - Liberate MX: Memory characterization
  - Liberate LV: Library verification package
  - Liberate AMS: Mixed-Signal characterization (Oct 2014)

Example AND2



AND2 netlist

```
// Library name: cern_cmos8rf_hd_tsmc_hvtnW
// Cell name: AND2_B_XL_TSMC_HVTN
// View name: av_extracted
subckt AND2_B_XL_TSMC_HVTN A B GND VDD Z
   c1 (VDD GND) capacitor c=2.00781e-16
   c2 (A GND) capacitor c=1.72585e-16
   c3 (B GND) capacitor c=1.40005e-16
   c4 (Z GND) capacitor c=5.9908e-17
   c5 (net58 GND) capacitor c=2.56284e-17
   c6 (net09 GND) capacitor c=9.28723e-17
   c7 (\1\:A GND) capacitor c=1.16705e-16
   c8 (\1\:B GND) capacitor c=6.54007e-17
   c9 (\5\:net09 GND) capacitor c=6.22271e-17
   ....
   rj10 (net09 \3\:net09) resistor r=31.2646 c=0
   rj12 (\3\:net09 \5\:net09) resistor r=22.4521 c=0
   rj11 (\3\:net09 \4\:net09) resistor r=19.2893 c=0
   TPB (\8\:net09 \1\:B \5\:VDD VDD) pch l=1.3e-07 w=3.2e-07 ad=0.0608p \
      as=0.22828p pd=0.7u ps=2.37u nrd=0.59375 nrs=2.22925 sa=2.95e-07  sb=8.8e-07 m=1
   TPA (\6\:VDD \1\:A \8\:net09 VDD) pch l=1.3e-07 w=3.2e-07 ad=0.2669p \
      as=0.0608p pd=2.37u ps=0.7u nrd=2.60645 nrs=0.59375 sa=8.05e-07  sb=3.7e-07 m=1
   M0 (\1\:VDD net09 \2\:Z VDD) pch l=1.3e-07 w=6.5e-07 ad=0.3715p \
      as=0.28275p pd=2.57u ps=2.17u nrd=0.87929 nrs=0.669231 sa=4.35e-07  sb=1.22989e-06 m=1
   TNB (net58 \4\:B \4\:GND GND) nch_hvt l=1.3e-07 w=3e-07 ad=0.057p \
      as=0.22687p pd=0.68u ps=2.37u nrd=0.633333 nrs=2.52083 sa=2.95e-07  sb=9.85e-07 m=1
   TNA (\9\:net09 \3\:A net58 GND) nch_hvt l=1.3e-07 w=3e-07 ad=0.1425p \
      as=0.057p pd=1.55u ps=0.68u nrd=1.58333 nrs=0.633333 sa=8.05e-07   sb=4.75e-07 m=1
   M1 (\1\:GND \5\:net09 \1\:Z GND) nch_hvt l=1.3e-07 w=3e-07 ad=0.347p \
      as=0.1215p pd=2.57u ps=1.41u nrd=3.85556 nrs=1.35 sa=4.05e-07  sb=1.05854e-06 m=1
ends AND2_B_XL_TSMC_HVTN
// End of subcircuit definition.
```

```
// Library name: cern_cmos8rf_hd_tsmc_hvtnW
// Cell name: AND2_B_XL_TSMC_HVTN
// View name: schematic
subckt AND2_B_XL_TSMC_HVTN A B GND VDD Z
   M0 (Z net09 VDD VDD) pch l=130n w=650n ad=221f as=221f pd=1.98u \
      ps=1.98u nrd=292.308m nrs=292.308m sa=340n sb=340n m=1
   TPA (net09 A VDD VDD) pch l=130n w=320n ad=108.8f as=108.8f pd=1.32u \
      ps=1.32u nrd=593.75m nrs=593.75m sa=340n sb=340n m=1
   TPB (net09 B VDD VDD) pch l=130n w=320n ad=108.8f as=108.8f pd=1.32u \
      ps=1.32u nrd=593.75m nrs=593.75m sa=340n sb=340n m=1
   M1 (Z net09 GND GND) nch_hvt l=130n w=300n ad=102f as=102f pd=1.28u \
      ps=1.28u nrd=633.333m nrs=633.333m sa=340n sb=340n m=1
   TNB (net58 B GND GND) nch_hvt l=130n w=300n ad=102f as=102f pd=1.28u \
      ps=1.28u nrd=633.333m nrs=633.333m sa=340n sb=340n m=1
   TNA (net09 A net58 GND) nch_hvt l=130n w=300n ad=102f as=102f pd=1.28u \
      ps=1.28u nrd=633.333m nrs=633.333m sa=340n sb=340n m=1
ends AND2_B_XL_TSMC_HVTN
// End of subcircuit definition.
```

Define_cell and define_arc

```
if {[ALAPI_active_cell "AND2_A_XL_TSMC_HVTN"]} {
define_cell \
      -input { A B } \
      -output { Z } \
      -pinlist { A B Z } \
      -delay delay_template_7x7 \
      -power power_template_7x7 \
      -si_immunity si_immunity_template_7x7 \
      AND2_A_XL_TSMC_HVTN
```

Cell Definition

```
define_leakage -when "!A&!B" AND2_A_XL_TSMC_HVTN
define_leakage -when "A&!B" AND2_A_XL_TSMC_HVTN
define_leakage -when "!A&B" AND2_A_XL_TSMC_HVTN
define_leakage -when "A&B" AND2_A_XL_TSMC_HVTN
```

Leakage

```
# power arcs from  => A  hidden
define_arc \
      -type hidden \
      -vector {Rxx} \
      -pin A \
      AND2_A_XL_TSMC_HVTN

# power arcs from  => A  hidden
define_arc \
      -type hidden \
      -vector {Fxx} \
      -pin A \
      AND2_A_XL_TSMC_HVTN

# power arcs from  => B  hidden
define_arc \
      -type hidden \
      -vector {xRx} \
      -pin B \
      AND2_A_XL_TSMC_HVTN

# power arcs from  => B  hidden
define_arc \
      -type hidden \
      -vector {xFx} \
      -pin B \
      AND2_A_XL_TSMC_HVTN
```

Dynamic power arcs

```
# delay arcs from A => Z positive_unate combinational
define_arc \
      -vector {RxR} \
      -related_pin A \
      -pin Z \
      AND2_A_XL_TSMC_HVTN

# delay arcs from A => Z positive_unate combinational
define_arc \
      -vector {FxF} \
      -related_pin A \
      -pin Z \
      AND2_A_XL_TSMC_HVTN

# delay arcs from B => Z positive_unate combinational
define_arc \
      -vector {xRR} \
      -related_pin B \
      -pin Z \
      AND2_A_XL_TSMC_HVTN

# delay arcs from B => Z positive_unate combinational
define_arc \
      -vector {xFF} \
      -related_pin B \
      -pin Z \
      AND2_A_XL_TSMC_HVTN

}
```

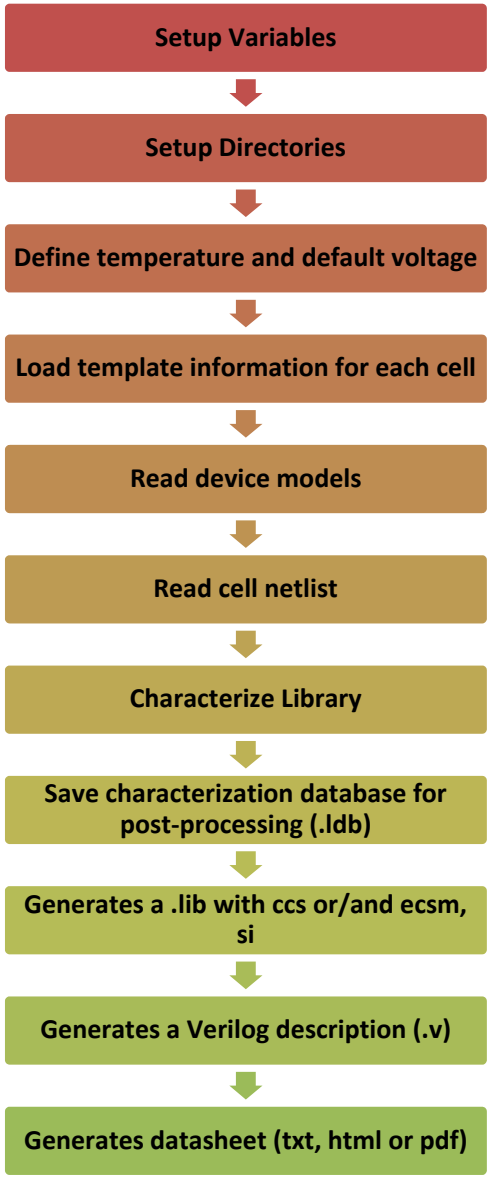
Delay arcs

```
define_template -type delay \
       -index_1 {0.0224 0.0608 0.12 0.32 0.72 1.6 3.0 } \
       -index_2 {0.0014 0.003 0.0062 0.0125 0.0251 0.0504 0.101 } \
       delay_template_7x7

define_template -type power \
       -index_1 {0.0224 0.0608 0.12 0.32 0.72 1.6 3.0 } \
       -index_2 {0.0014 0.003 0.0062 0.0125 0.0251 0.0504 0.101 } \
       power_template_7x7

define_template -type si_immunity \
       -index_1 {0.224 0.608 1.2 3.2 7.2 16.0 30.0 } \
       -index_2 {0.0014 0.003 0.0062 0.0125 0.0251 0.0504 0.101 } \
       si_immunity_template_7x7
```

Templates

Liberate Example Tcl File

```
set rundir $env(PWD)
set CORNER tt
set VOLTAGE 1.2
set TEMPERATURE 25
set cell AND2_B_XL_TSMC_HVTN

# Create the directories Liberate will write to.
exec mkdir -p ${rundir}/LDB
exec mkdir -p ${rundir}/LIBRARY
exec mkdir -p ${rundir}/DATASHEET
exec mkdir -p ${rundir}/VERILOG

set_operating_condition -voltage $VOLTAGE -temp $TEMPERATURE

## Load template information for each cell ##
source ${rundir}/TEMPLATE/cern_cmos8rf_hd_tsmc_hvtnW_v120_tt_25C_hvtn.tcl

read_spice $rundir/MODELS/hspice/include_$CORNER\.sp

read_spice -format spectre ${rundir}/CELLS/cern_cmos8rf_hd_tsmc_hvtnW_nwsx.sch.scs

char_library -ecsmn -ecsmp -si -cells ${cells} -thread 8

write_ldb ${rundir}/LDB/$SOURCE_LIBRARY\_$VOLTAGE\V\_$TEMPERATURE\C_$CORNER.ldb

write_library -overwrite -ecsmn -si ${rundir}/LIBRARY/$SOURCE_LIBRARY\_$VOLTAGE\V\_$TEMPERATURE\C_$CORNER\_ecsm_si.lib

write_verilog  ${rundir}/VERILOG/$SOURCE_LIBRARY\_$VOLTAGE\V\_$TEMPERATURE\C_$CORNER\.v

write_datasheet -format html -dir ${rundir}/DATASHEET/$SOURCE_LIBRARY\_$VOLTAGE\V\_$TEMPERATURE\C_$CORNER test
```



Output files AND2 .lib / .v

```
cell (AND2_A_XL_TSMC_HVTN) {
  area : 0;
  cell_leakage_power : 0.14561;
  leakage_power () {
   value : 0.0891823;
   when : "!A&!B";
  }
  leakage_power () {
   value : 0.129436;
   when : "A&!B";
  }
  leakage_power () {
   value : 0.146063;
   when : "!A&B";
  }
  leakage_power () {
   value : 0.217761;
   when : "A&B";
  }
  pin (Z) {
   direction : output;
   ecsm_gndres : 2.84782;
   ecsm_noise_cap : 0.000606332;
   ecsm_vddres : 7.98722;
   function : "(A * B)";
   max_capacitance : 0.101;
   output_voltage : "DC_0";
   timing () {
    related_pin : "A";
    timing_sense : positive_unate;
    timing_type : combinational;
    cell_rise (delay_template_7x7) {
     index_1 ("0.0224, 0.0608, 0.12, 0.32, 0.72, 1.6, 3");
     index_2 ("0.0014, 0.003, 0.0062, 0.0125, 0.0251, 0.0504, 0.101");
     values ( \
      "0.0778087, 0.0943178, 0.126192, 0.188057, 0.311119, 0.557708, 1.05057", \
      "0.0864114, 0.102866, 0.134701, 0.196585, 0.319701, 0.566341, 1.05924", \
      "0.0989374, 0.115341, 0.147097, 0.208953, 0.332114, 0.578826, 1.07177", \
      "0.124335, 0.141053, 0.172989, 0.234982, 0.358065, 0.604813, 1.09784", \
      "0.150635, 0.168129, 0.200259, 0.261964, 0.385234, 0.63214, 1.12516", \
      "0.175095, 0.19526, 0.229331, 0.291785, 0.4152, 0.66214, 1.15554", \
      "0.181677, 0.205527, 0.244012, 0.30901, 0.434542, 0.683591, 1.17743" \
     );
    }
.....
```

.lib

```
`timescale 1ns/10ps
`celldefine
module AND2_A_XL_TSMC_HVTN (Z, A, B);
        output Z;
        input A, B;

        // Function
        and (Z, A, B);

        // Timing
        specify
                specparam tpd_A_Z = 0;
                specparam tpd_B_Z = 0;

                (A => Z) = tpd_A_Z;
                (B => Z) = tpd_B_Z;
        endspecify
endmodule
`endcelldefine
```

.v

Datasheet output AND2

AND2_A_XL_TSMC_HVTN

test Cell Library: Process , Voltage 1.20, Temp 25.00

Truth Table

INPUT		OUTPUT
A	B	Z
0	x	0
1	0	0
1	1	1

Pin Capacitance Information

Cell Name	Pin Cap(pf)		Max Cap(pf)
	A	B	Z
AND2_A_XL_TSMC_HVTN	0.00153	0.00148	0.10100

Leakage Information

Cell Name	Leakage(nW)		
	Min.	Avg	Max.
AND2_A_XL_TSMC_HVTN	0.08918	0.14561	0.21776

Delay Information

Delay(ns) to Z rising :

Cell Name	Timing Arc(Dir)	Delay(ns)		
		Min	Mid	Max
AND2_A_XL_TSMC_HVTN	A->Z (RR)	0.07781	0.23498	1.17743
	B->Z (RR)	0.08329	0.23490	1.16232

Delay(ns) to Z falling :

Cell Name	Timing Arc(Dir)	Delay(ns)		
		Min	Mid	Max
AND2_A_XL_TSMC_HVTN	A->Z (FF)	0.07199	0.19680	0.85873
	B->Z (FF)	0.08133	0.21189	0.89763

Power Information

Internal switching power(pJ) to Z rising :

Cell Name	Input	Power(pJ)		
		min	mid	max
AND2_A_XL_TSMC_HVTN	A	0.00384	0.00415	0.01111
	B	0.00405	0.00417	0.01060

Internal switching power(pJ) to Z falling :

Cell Name	Input	Power(pJ)		
		min	mid	max
AND2_A_XL_TSMC_HVTN	A	0.00384	0.00418	0.01101
	B	0.00492	0.00527	0.01198

Passive power(pJ) for A rising :

Cell Name	Power(pJ)		
	min	mid	max
AND2_A_XL_TSMC_HVTN	0.00002	0.00002	0.00000

Passive power(pJ) for A falling :

Cell Name	Power(pJ)		
	min	mid	max
AND2_A_XL_TSMC_HVTN	0.00001	0.00001	0.00001

Passive power(pJ) for B rising :

Cell Name	Power(pJ)		
	min	mid	max
AND2_A_XL_TSMC_HVTN	0.00025	0.00025	0.00025

Passive power(pJ) for B falling :

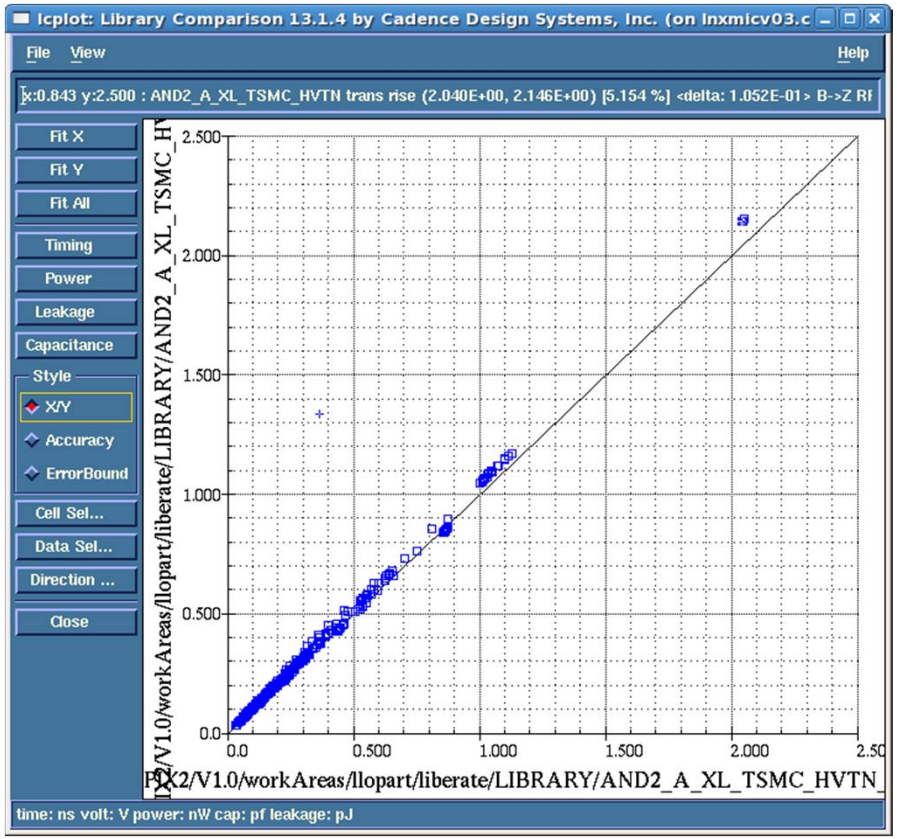
Cell Name	Power(pJ)		
	min	mid	max
AND2_A_XL_TSMC_HVTN	-0.00021	-0.00024	-0.00025

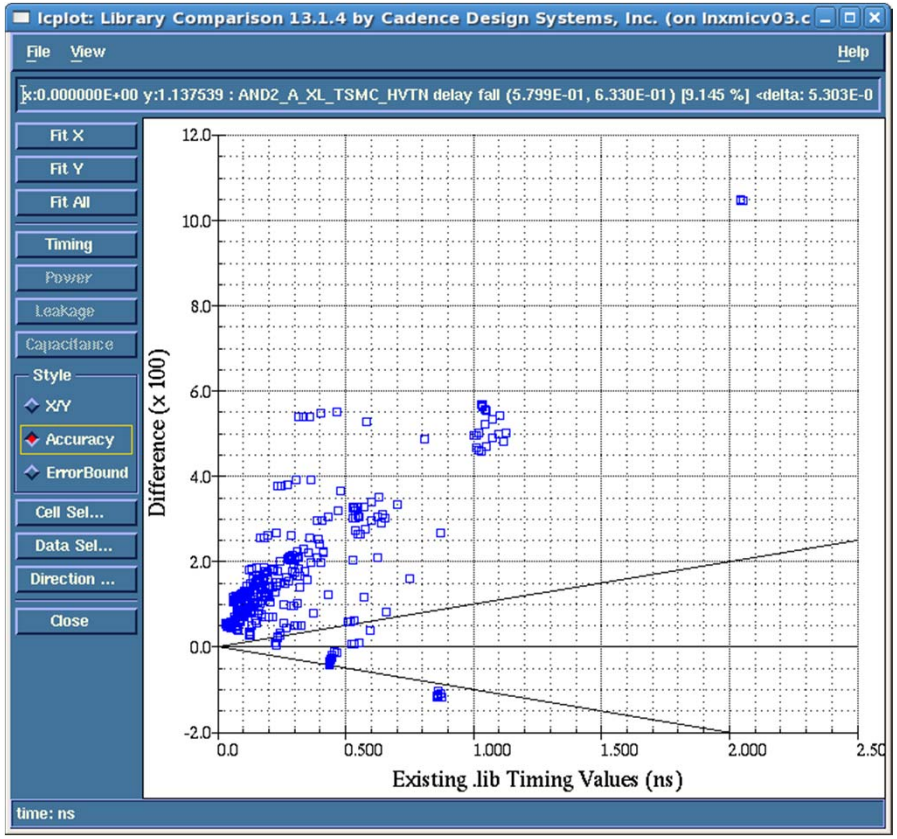
Liberate LV (Library Verification)

- Data consistency library checking
- compare_library:
  - Compares in two libraries textually and graphically
  - CCS or ECSM delay vs NLDM
  - Supports functional overlap
- validate_data_range
  - Check data is within range, not zero, negative etc
- validate_monotonicity
  - Ensures that library tables are monotonically increasing
- validate_sdf
  - Check SDF back-annotation to Verilog/VHDL simulation data consistency

compare library

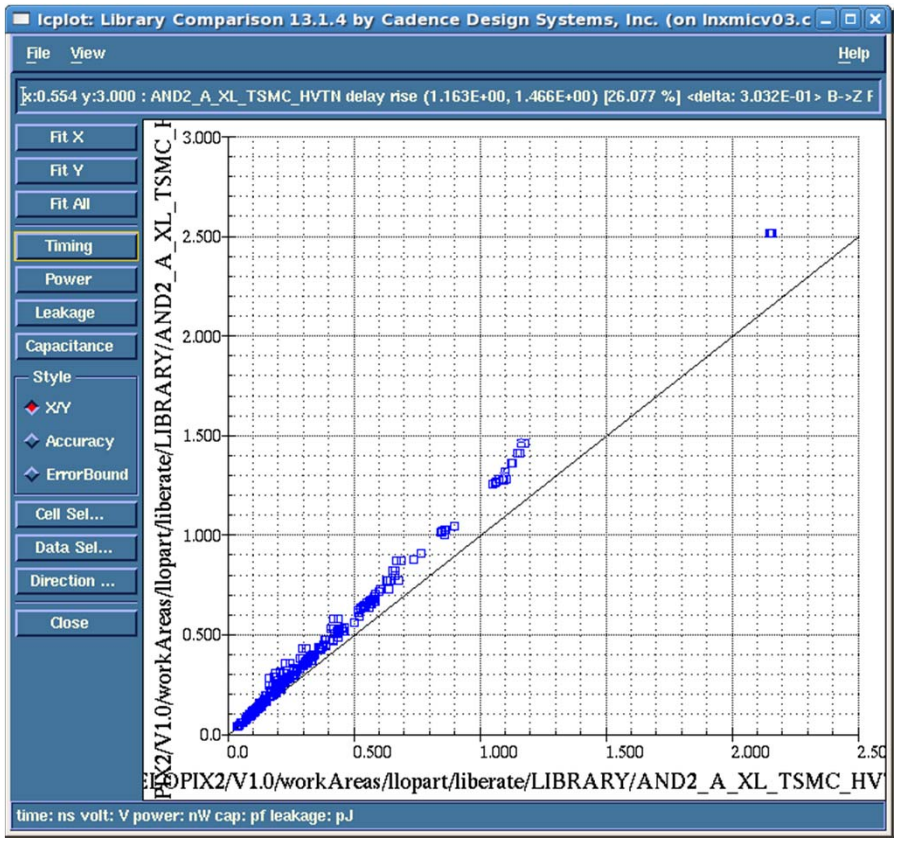
- Comparison of AND2 cell schematic vs extracted view



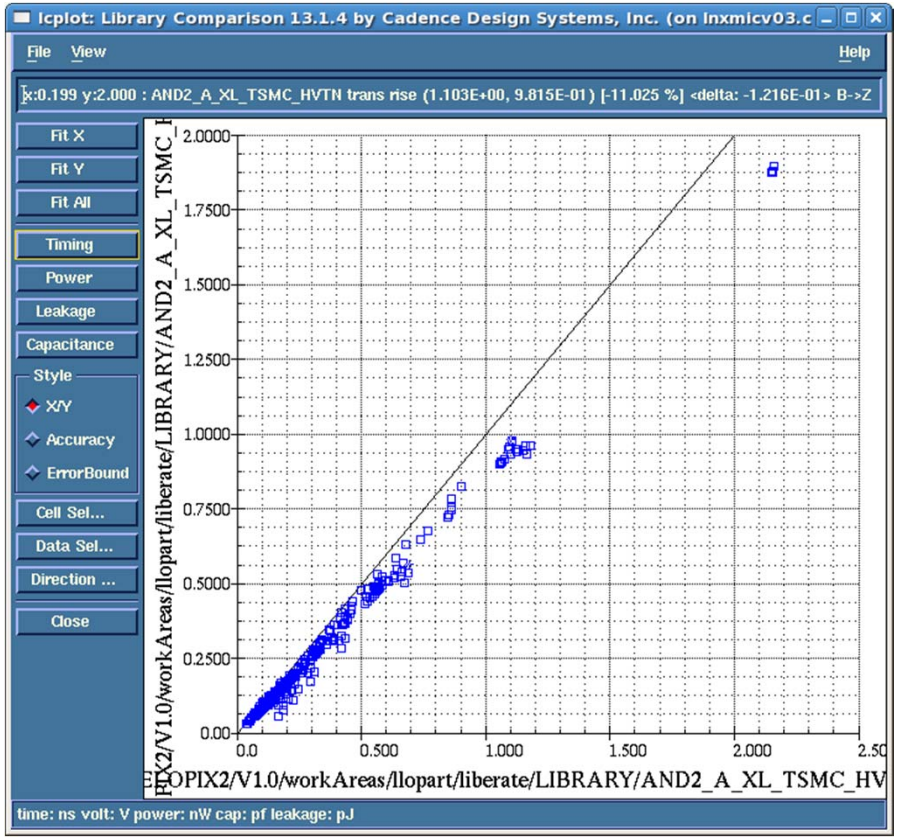


PVT: Process

- Comparison of AND2 gate



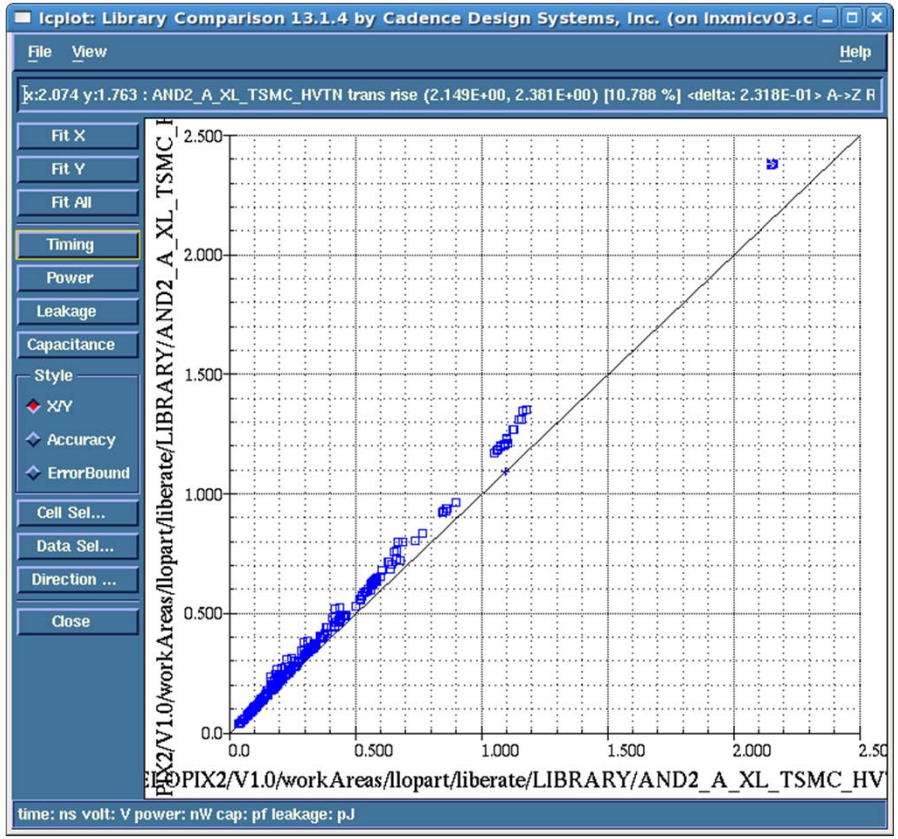
Comparison 1.2V ss/tt 25C



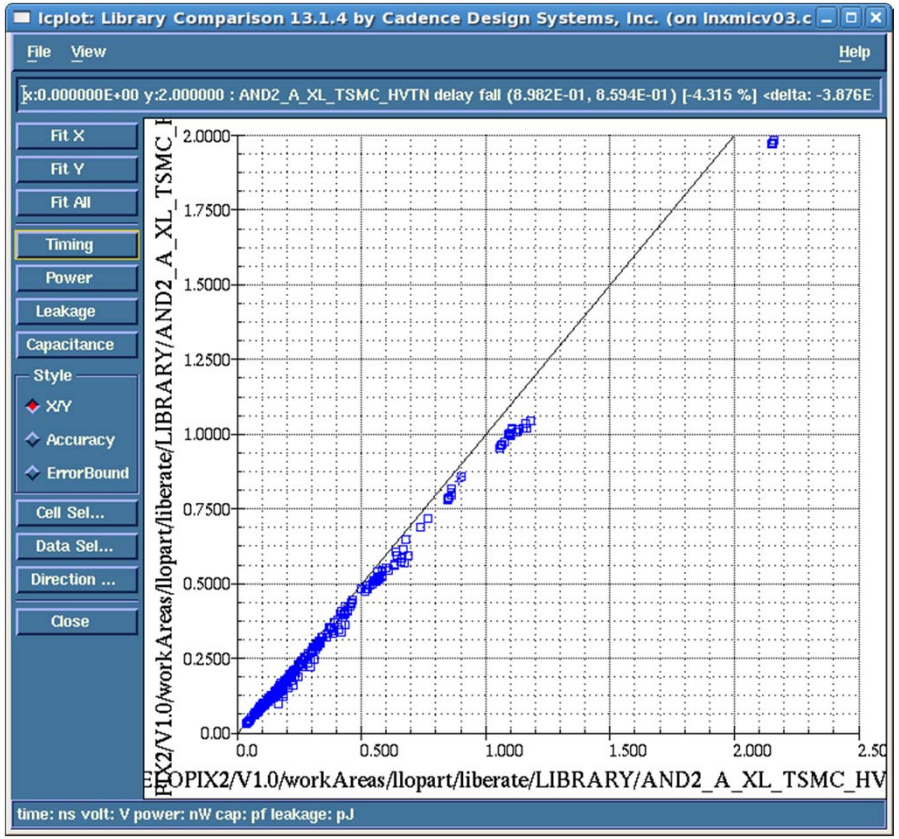
Comparison 1.2V ff/tt 25C

PVT: Voltage

- Comparison of AND2 gate



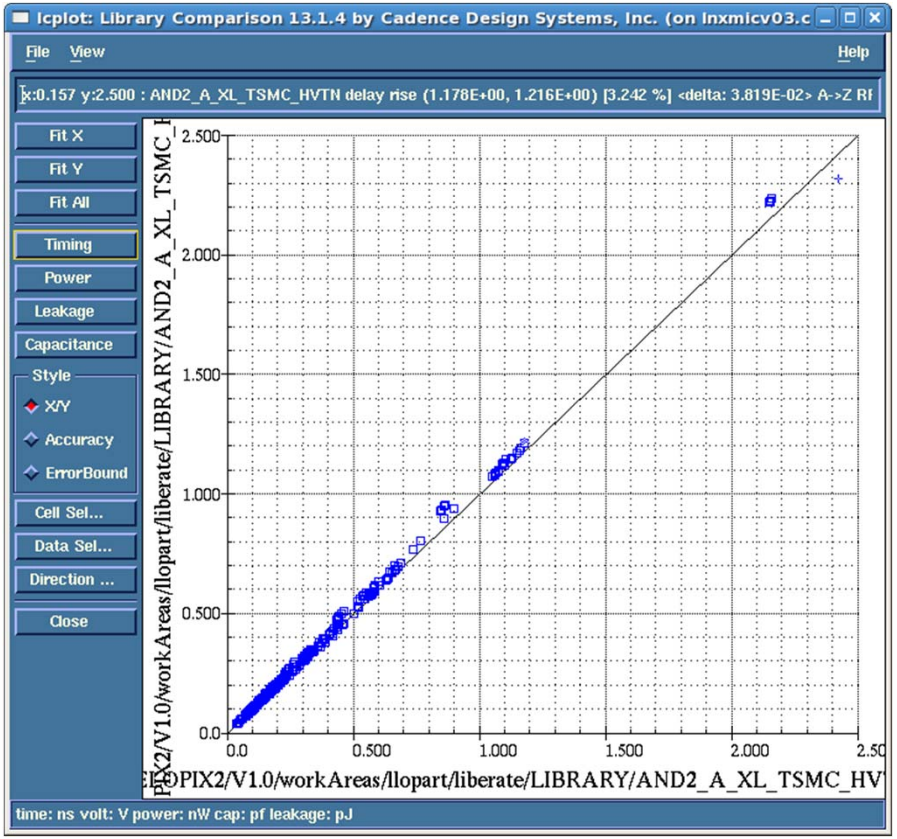
Comparison 1.1V vs 1.2V tt 25C



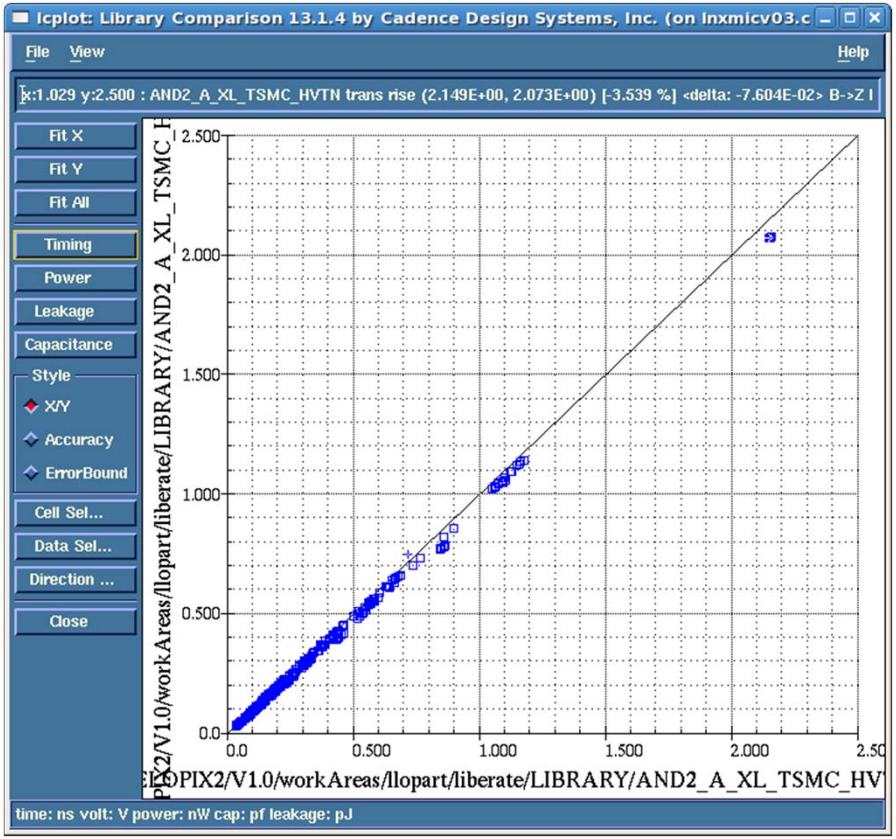
Comparison 1.3V vs 1.2V tt 25C

PVT: Temperature

- Comparison of AND2 gate



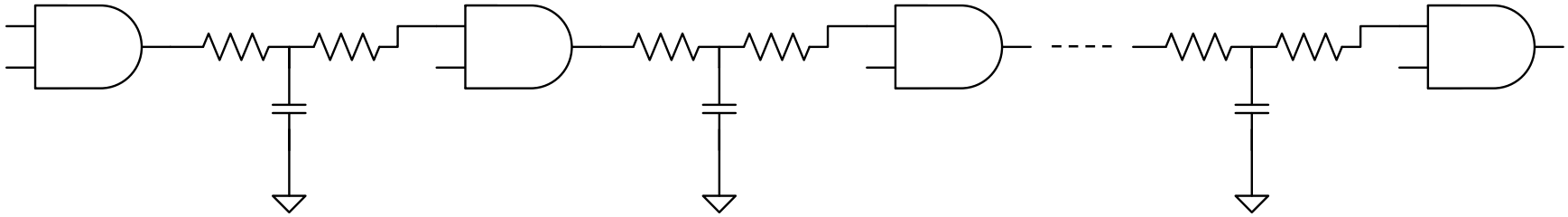
Comparison 1.2V tt 80C/25C



Comparison 1.2V tt -20C/25C

Liberate LV Validation - Timing and Power

- Compares Spice vs STA (ETS, Tempus, PrimeTime)
  - Text, Excel or Html reporting and Graphical comparison display (lcplot)
  - Arbitrary chain length, user controllable interconnect RC and fanout
  - All slews/loads or defined slews/loads or interpolated or extrapolated slew/loads
  - Also supports SSTA Vs Monte Carlo simulation
- Power Correlation for NLPM & ECSMP via EPS



validate_library example

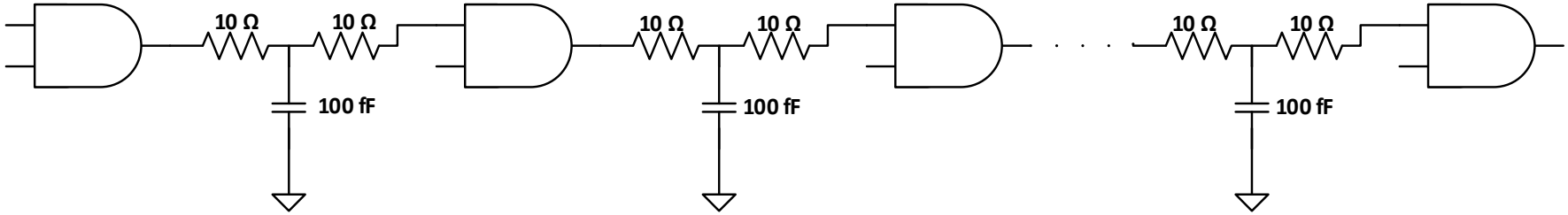
```
set rundir $env(PWD)
set CORNER tt
set SPECTRE_MODELS_FILE "${rundir}/MODELS/tsmc13rf_$CORNER\.scs"
set SPECTRE_CMD "/eda/cadence/2014-15/RHELx86/MMSIM_13.11.252/bin/spectre"

set_operating_condition -voltage 1.2 -temp 25

set cells {AND2_A_XL_TSMC_HVTN}
set modelspectre $SPECTRE_MODELS_FILE
set subckts ${rundir}/CELLS/lat.scs

set_var extsim_cmd "$SPECTRE_CMD -64"
set_var extsim_cmd_option " +aps +spice +mt=2 +lqt 0 +errpreset=conservative -format psfbin"
set_var extsim_deck_dir "${rundir}/temp/Spectre_decks_$CORNER"

# ETS based flow
validate_library                                                \
                    -dir            TEMPUS                      \
                    -timer          tempus                      \
                    -subckts        $subckts                    \
                    -model          $modelspectre               \
                    -extsim         spectre                     \
                    -extsim_format  spectre                     \
                    -chain_length   10                          \
                    -format         htm                         \
                    -timing_models  ecsm                        \
                    -wire_res       100                         \
                    -wire_cap       0.010                       \
                    -cells          ${cells}                    \
                    LIBRARY/cern_cmos8rf_hd_tsmc_hvtnW_liberate_1.2V_25C_tt_ecsm_si.lib
```

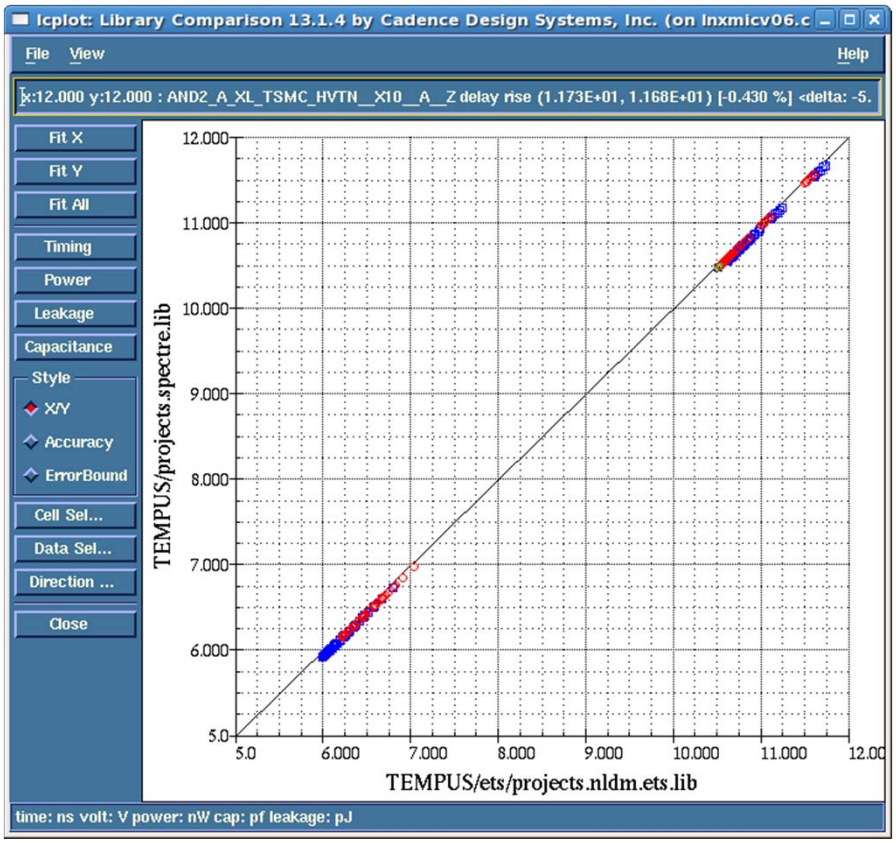
Number of combinatorial elements in the chain

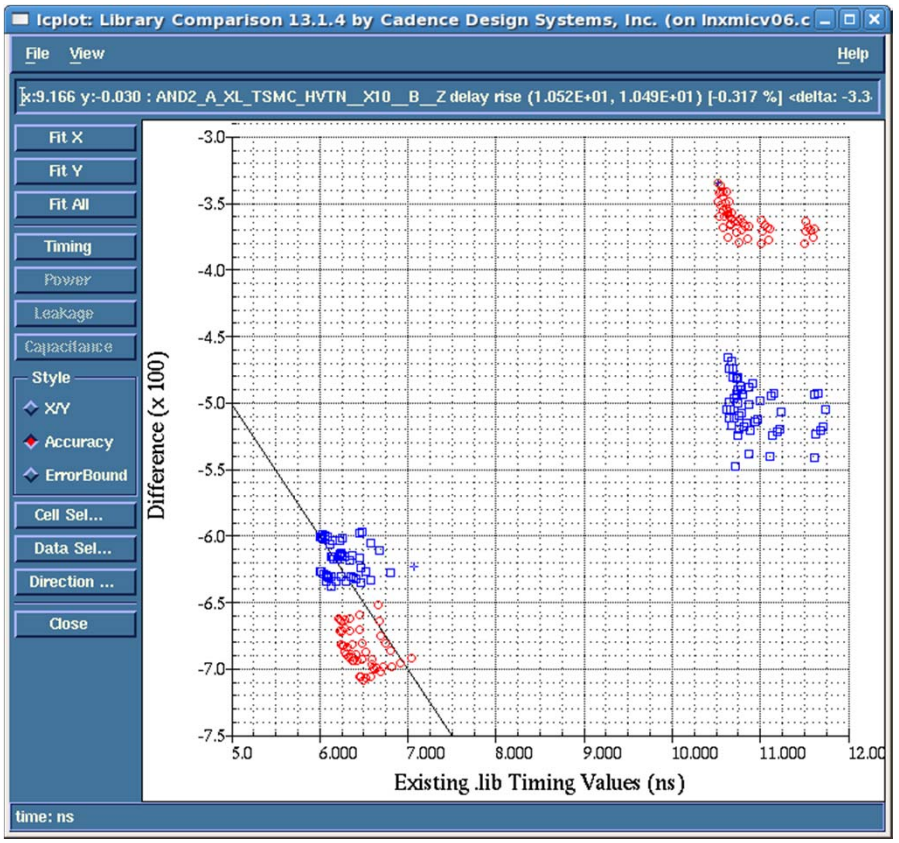
Resistance in Ohms

Capacitance in pF

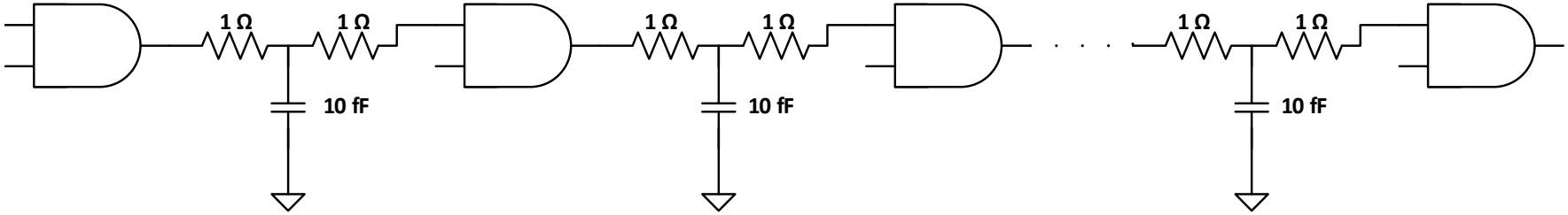
10 AND2 cells R=10Ω and C=100fF

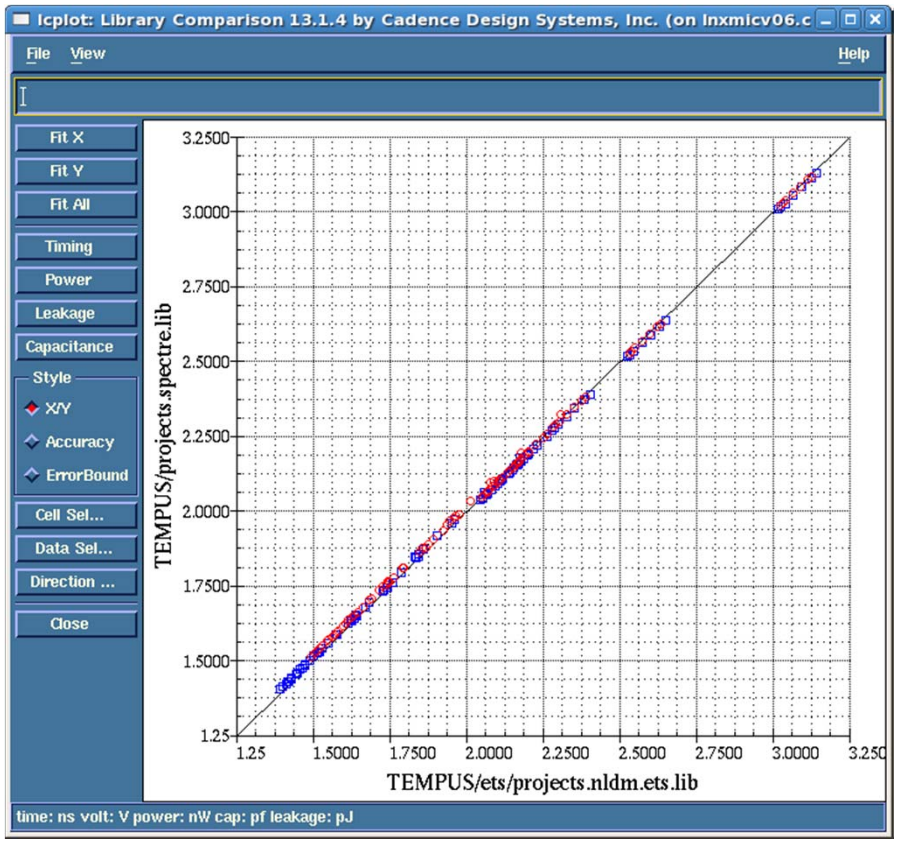


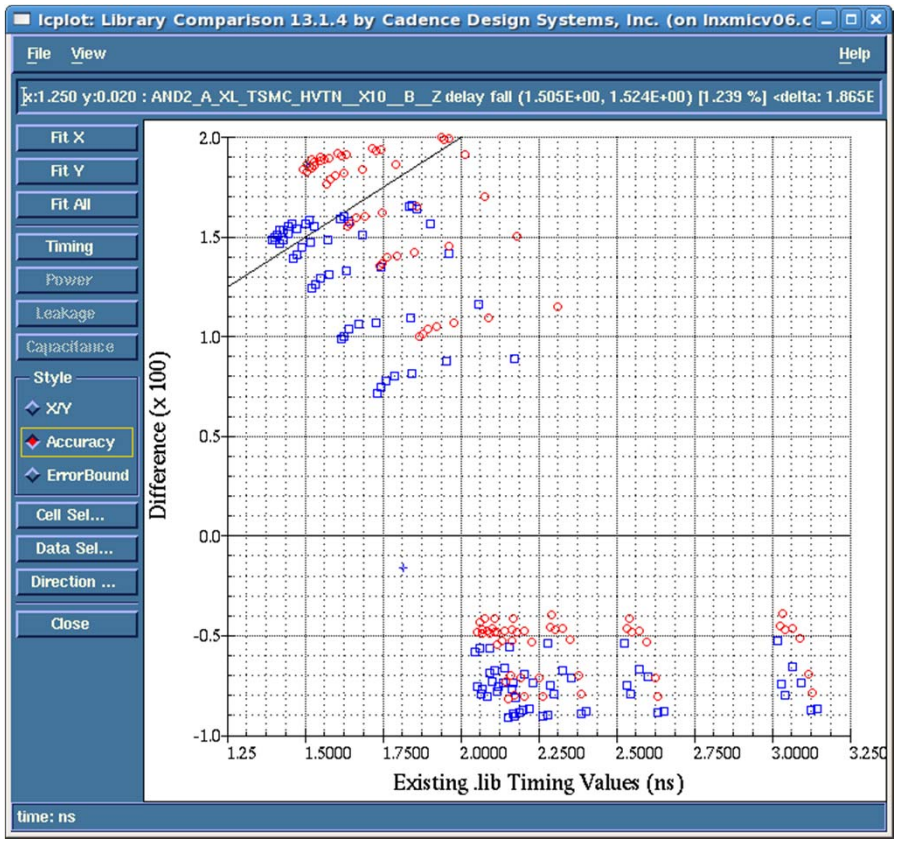




10 AND2 cells R=1Ω and C=10fF







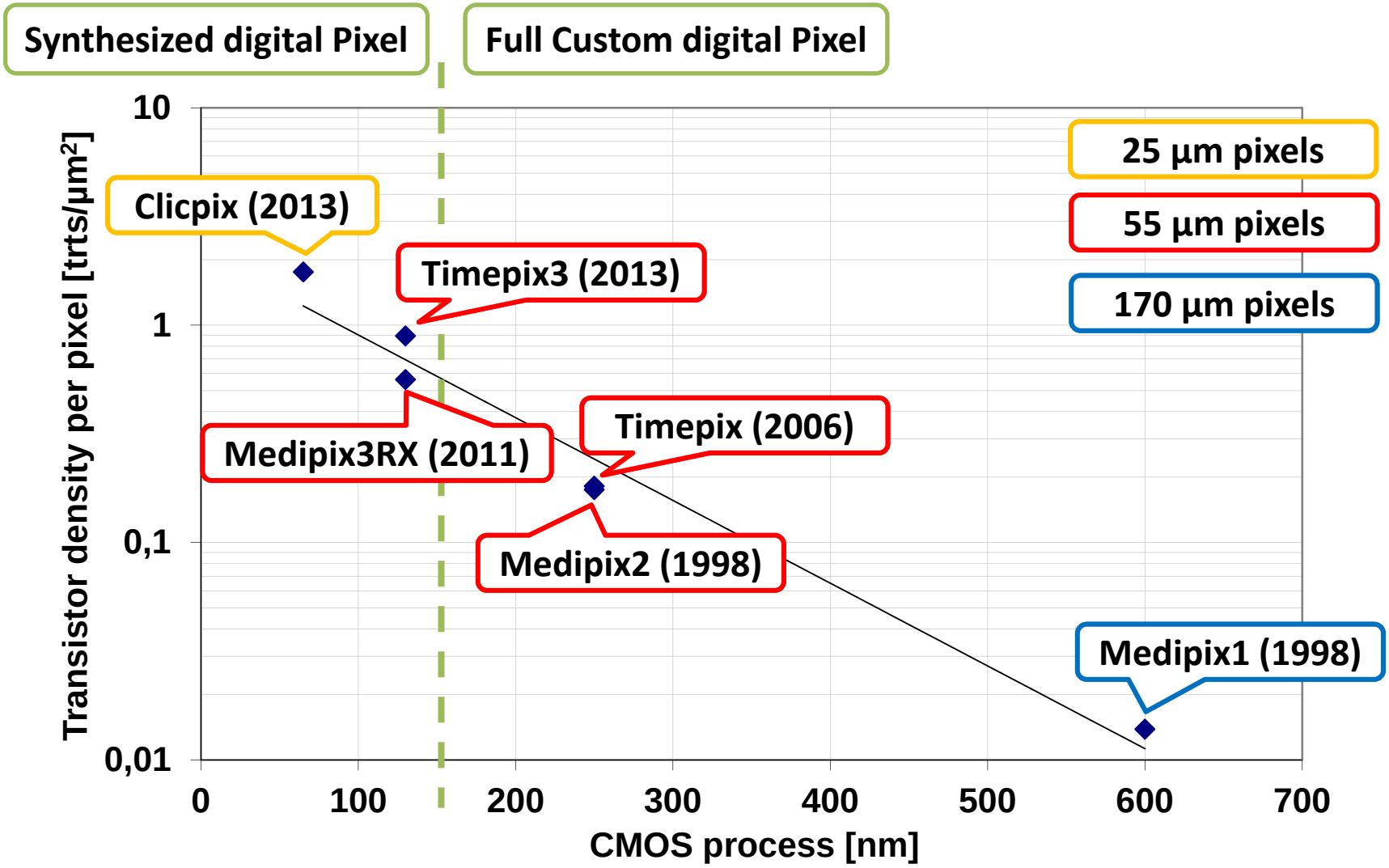
Applications for HEP

- Re-characterize cells:
  - New PVT corners (temperature, voltage, corner)
  - New Models (model with radiation variation)
  - Modify input slew or output load templates
- Characterize new cells:
  - New Library: CERN 130nm HD library,…
  - Complex combinatorial gates: SEU voters,…
  - Synchronous cells: Dynamic flip-flops, SEU DFF,…
  - To improve digital verification accuracy (GBT/ALPIDE)
- Characterize new mixed-mode blocks

Examples on HEP applications

- Medipix3
- Timepix3
- Velopix
- GBT serializer
- ALPIDE

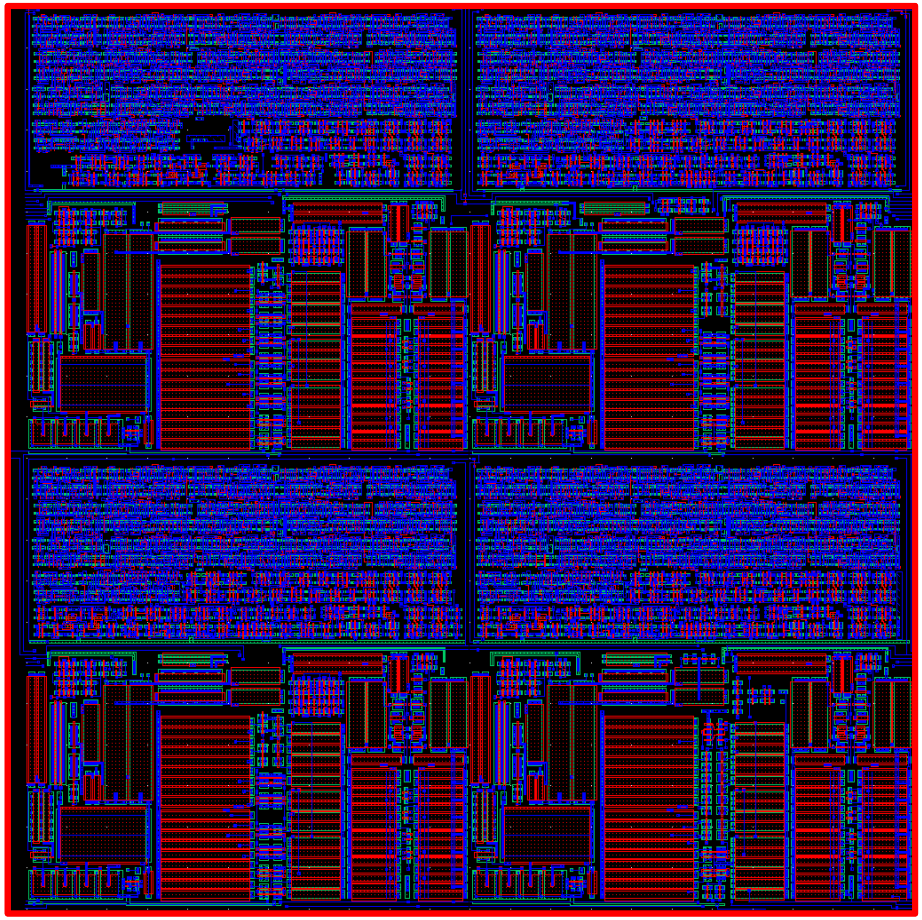
Medipix chip family



First steps towards made library

Medipix3.0 (2009)

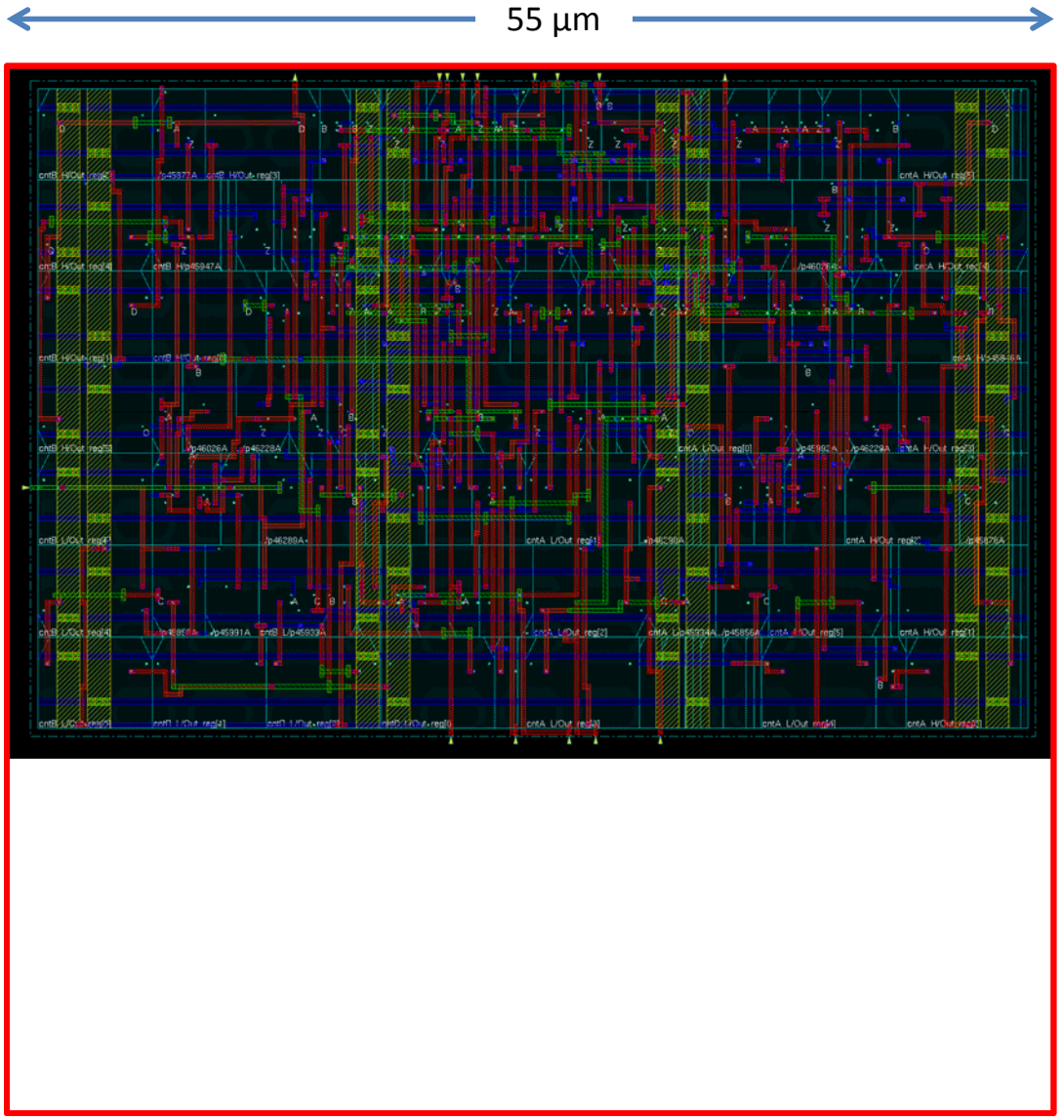
- Medipix3.0 (130nm) was designed in Full custom mode
  - Pixel architecture required that the basic cell was formed by 4 pixels
  - >1 year to layout the pixel
  - Digital verification done by gate level simulation only
    - Gates characterized “by hand”
- Chip debugging showed failure in some operating modes due to poor characterization of cells



Medipix3.0 pixel counter (24 bits) + control logic

- Synthesized using the CERN IBM 130nm Standard Cell Library (CMOS8RF)
- Area is too big (52 x 33.6) → ~60% pixel area
- High static (leakage) power consumption:

VDD	Temp	Leakage in cell	leakage In Chip
1.4 V	125 C	22.5 µW	**1.5 W !!!**
1.5 V	25 C	223 nW	**14.6 mW**
1.6 V	-55 C	470 pW	**30 µW**



A high density low power 130 nm digital library

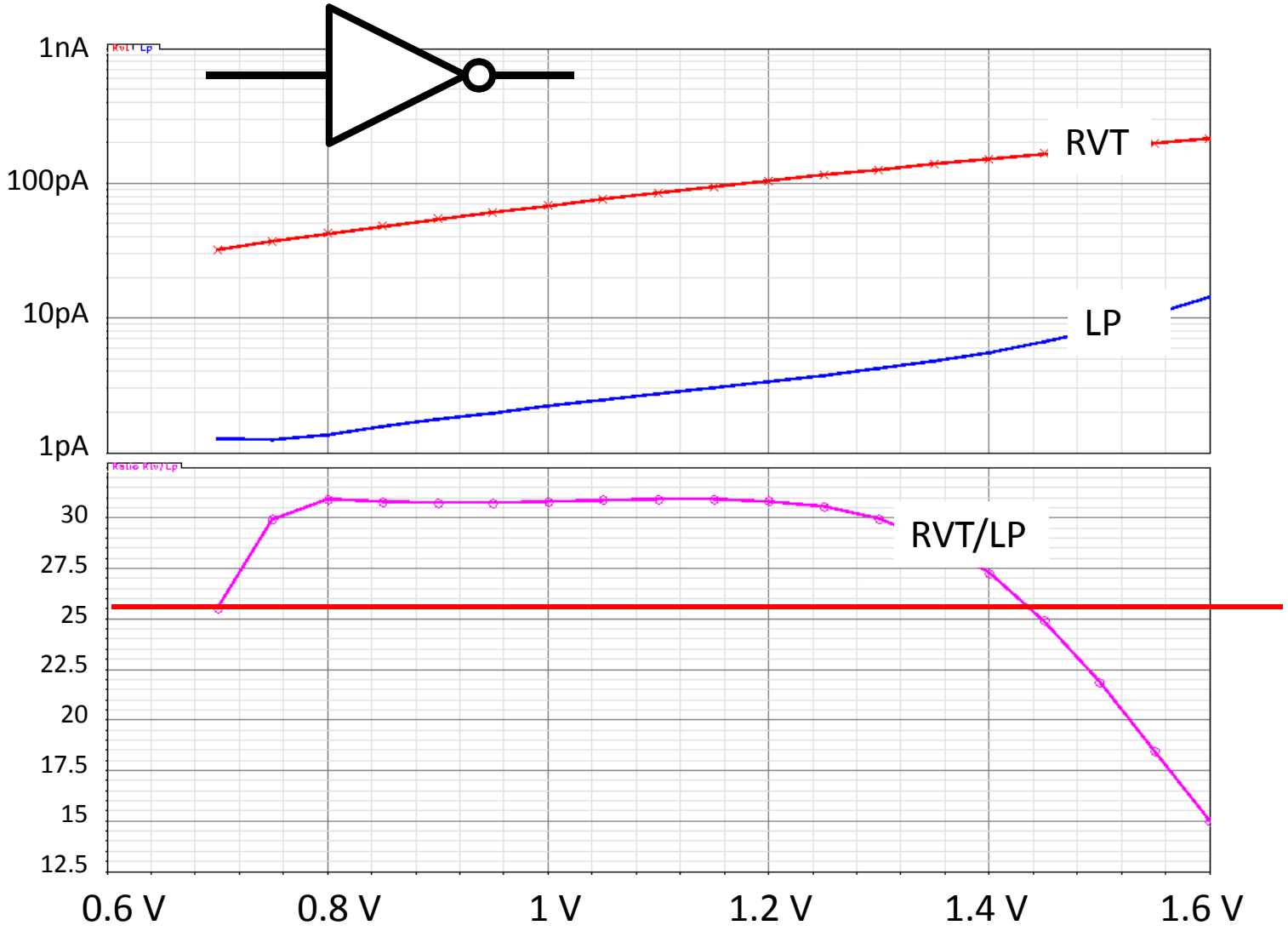
- The Medipix3RX redesign required significant changes in the digital part of the pixel (designed as a full custom layout)

- Foundry standard digital cell 130nm library available
  - **Too big cells:** The foundry digital library has large cells since is targeted for a general purpose and high speed design ~800MHz !!
  - **Too much leakage power:** The foundry digital library used regular transistors

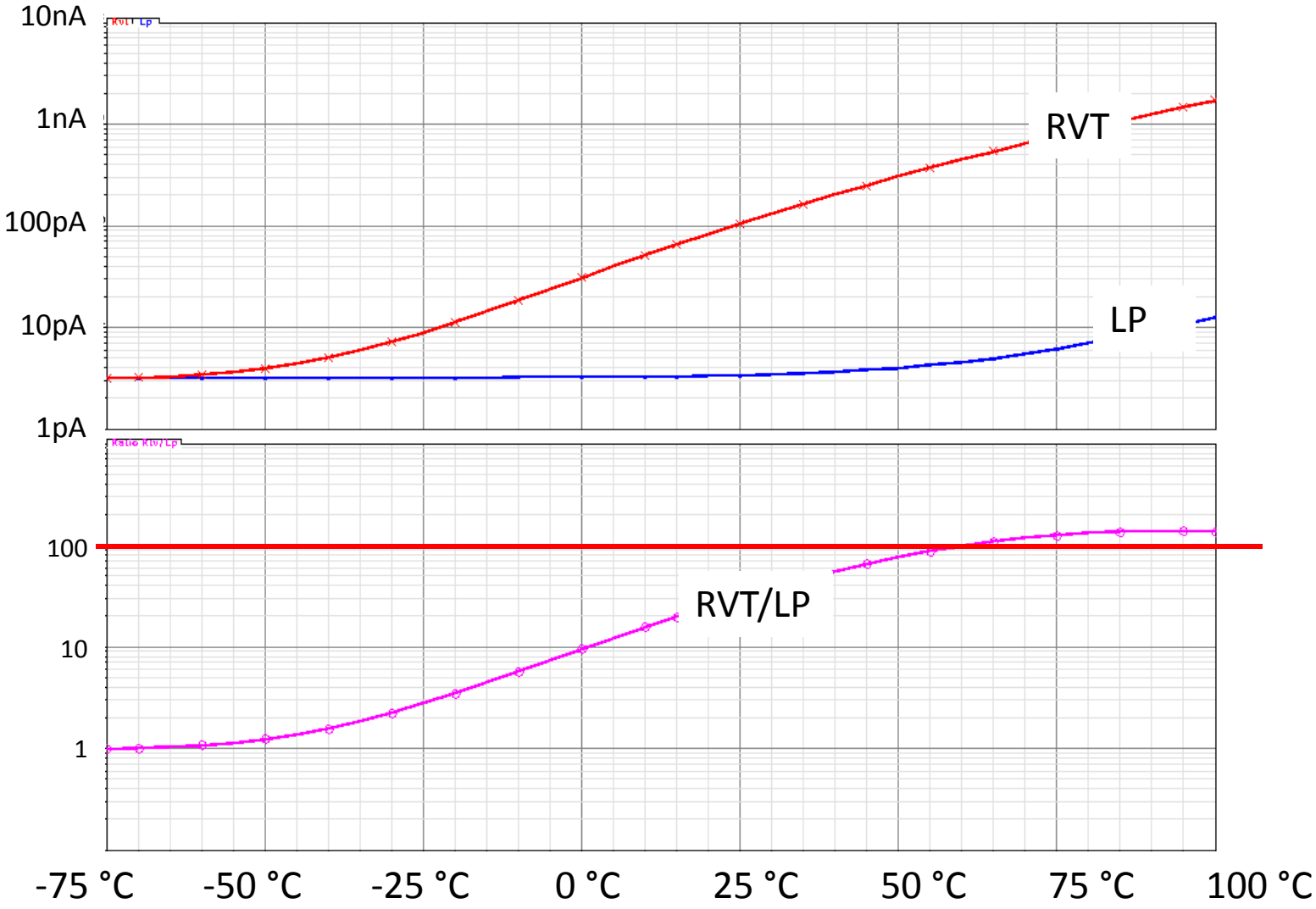
- A custom made high density library (SC_130nm_XL) was designed with the initial idea of using it in the Medipix3RX pixel but with a huge potential impact in later developments (Timepix3, Velopix…)

- Main actions:
  - Reduce cells height → No need for big buffers (speed)
  - Keep all transistor small or minimum size (W/L=0.28/0.12)
  - ADD NV and PV layers → Low power transistors (no area penalty)

RVT vs LP : VDD vs Power @25C

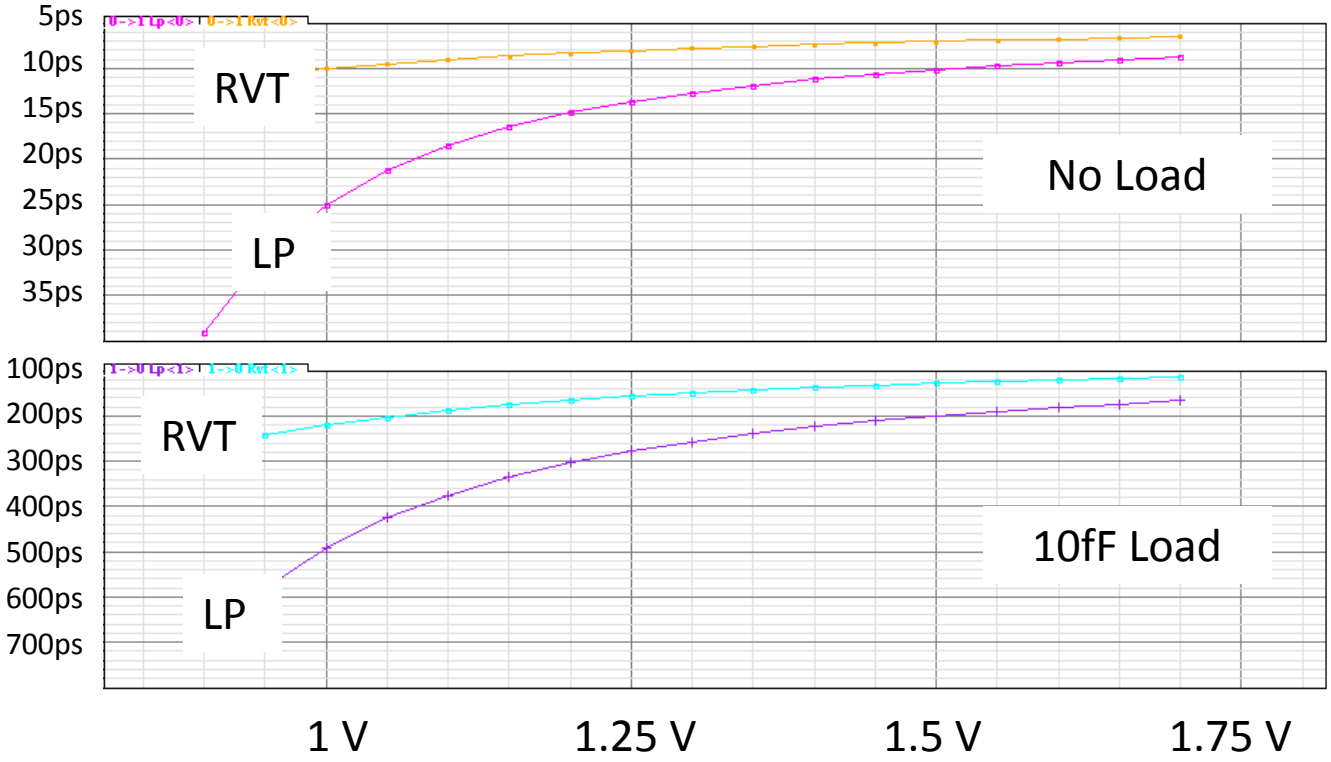


RVT vs LP : Temp vs Power @VDD=1.2V



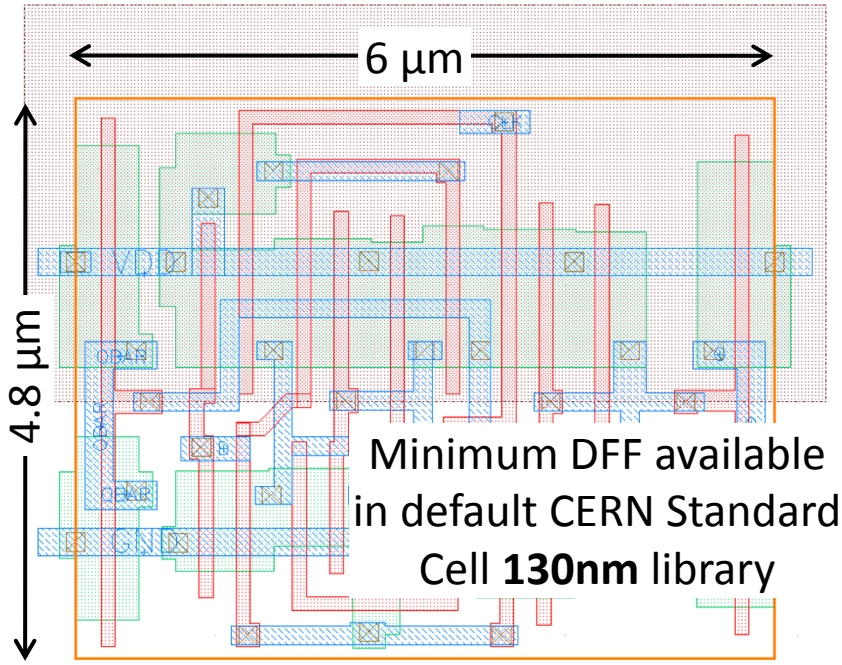
Simulated Delay

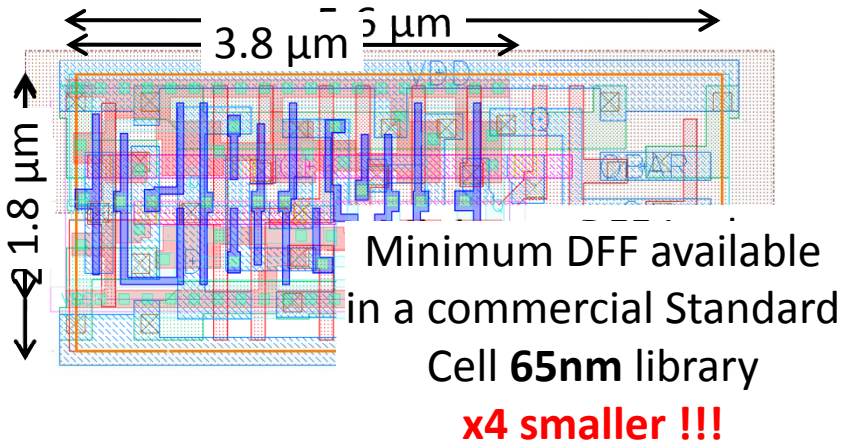
- @ VDD=1.5 the rise/fall times difference between Rvt and LP are:
  - ~3ps (0 → 1)
  - ~10ps (1 → 0)
- At lower VDDs the delays are more important



CERN 130nm HD Library

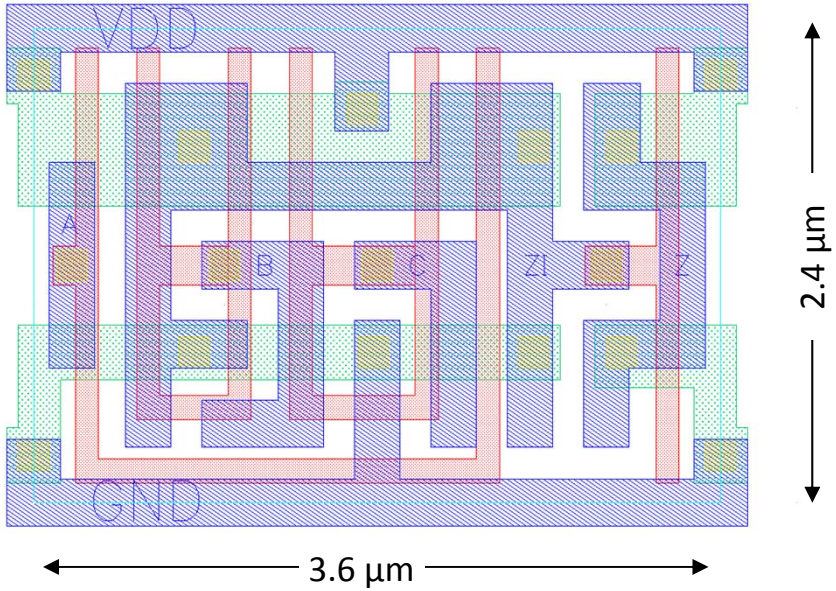
- Physical specs:
  - "Mainly" Low power transistors
  - Row Height is fixed to 2.4 µm
  - Well Tap library
- Maximum frequency < ~700 MHz (@1.5V)
- Encounter Library Characterizer (ELC) used:
  - Full Synopsis library:
    - lib, ecsm, ecsm_si and ccs
    - delays, static and dynamic power
    - Corners:
      - 1.2V : -55C FF, 25C TT and 125C SS
      - 1.5V : -55C FF, 25C TT and 125C SS
  - Verilog library
  - LEF files
  - HTML documentation
- ~50 cells available in the library



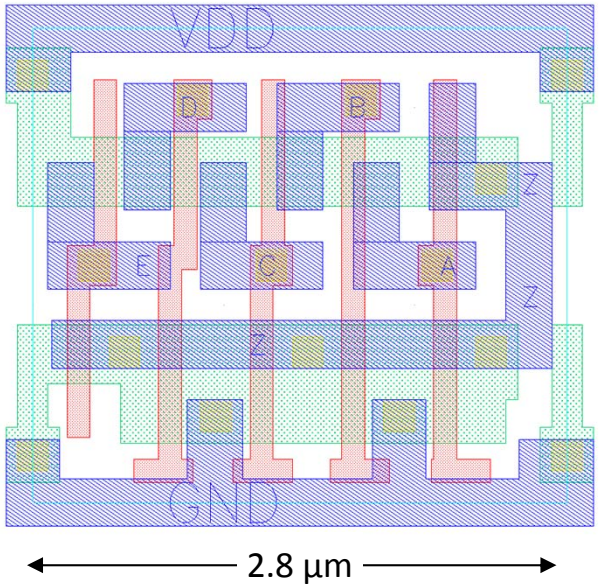


CERN HD Special cells (I)

- Cells with “non-standard” functional behaviour can be integrated



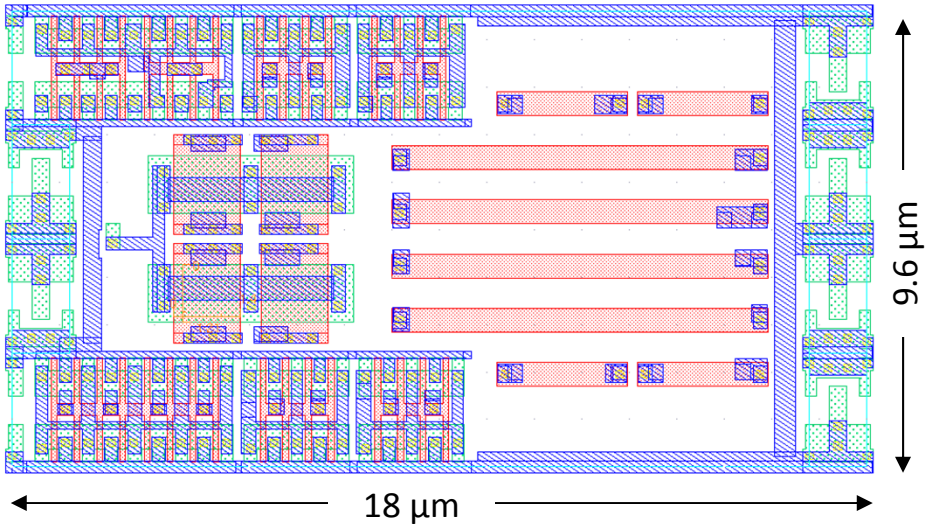
VOTERI_B_XL
(VeloPix)



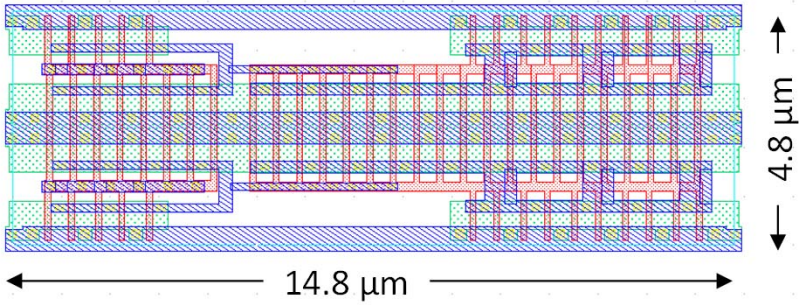
NOR5_A_XL
(Medipix3RX)

CERN HD Special cells (II)

- Cells with non-standard row height are also possible to integrate:
  - Row height multiple (2.4 µm)
  - Pitch length multiple (0.4 µm)



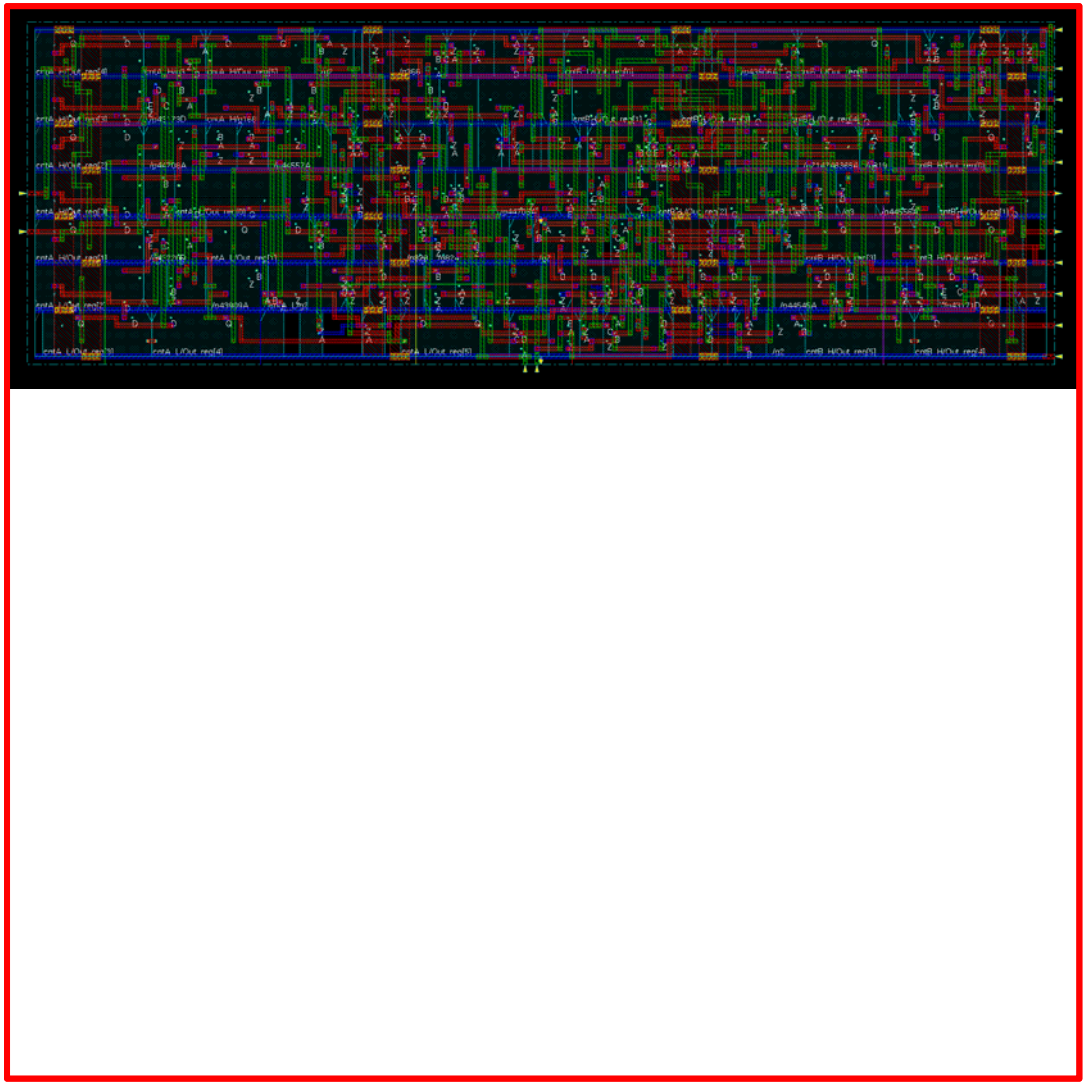
640MHz VCO
(Timepix3)



CLK_Q_RVT_XL
(Timepix3)

Medipix3 pixel counter (24 bits) + control logic

55 µm

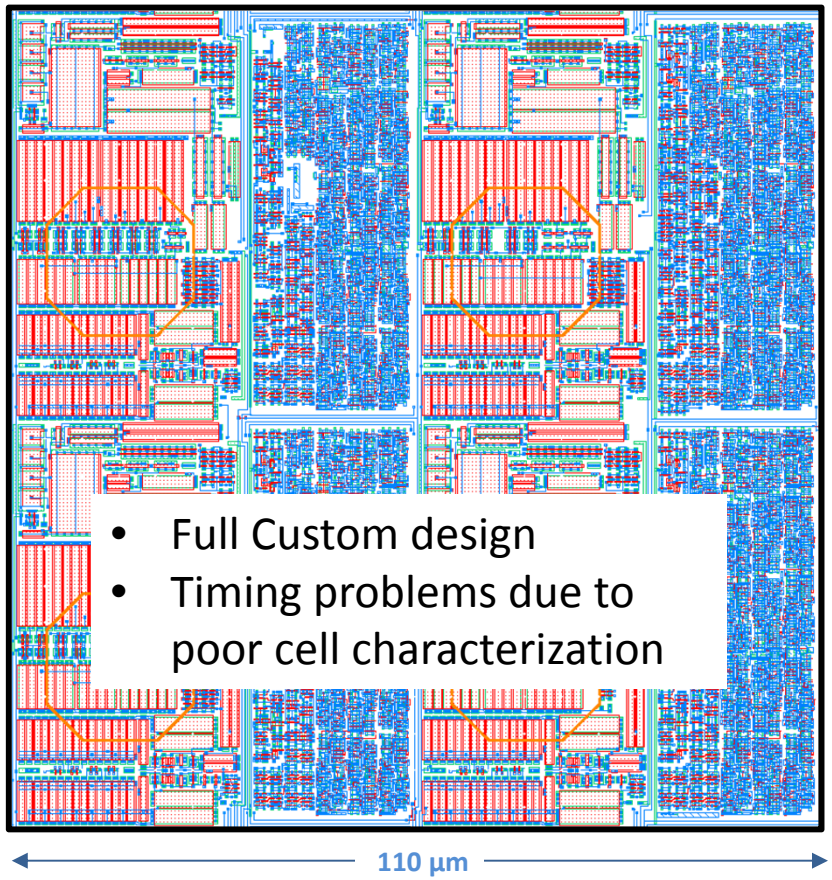


- Synthesized using the SC_130nm_XL library
- Area (52 x 16.8) → ~29% pixel area
- Low cell leakage power

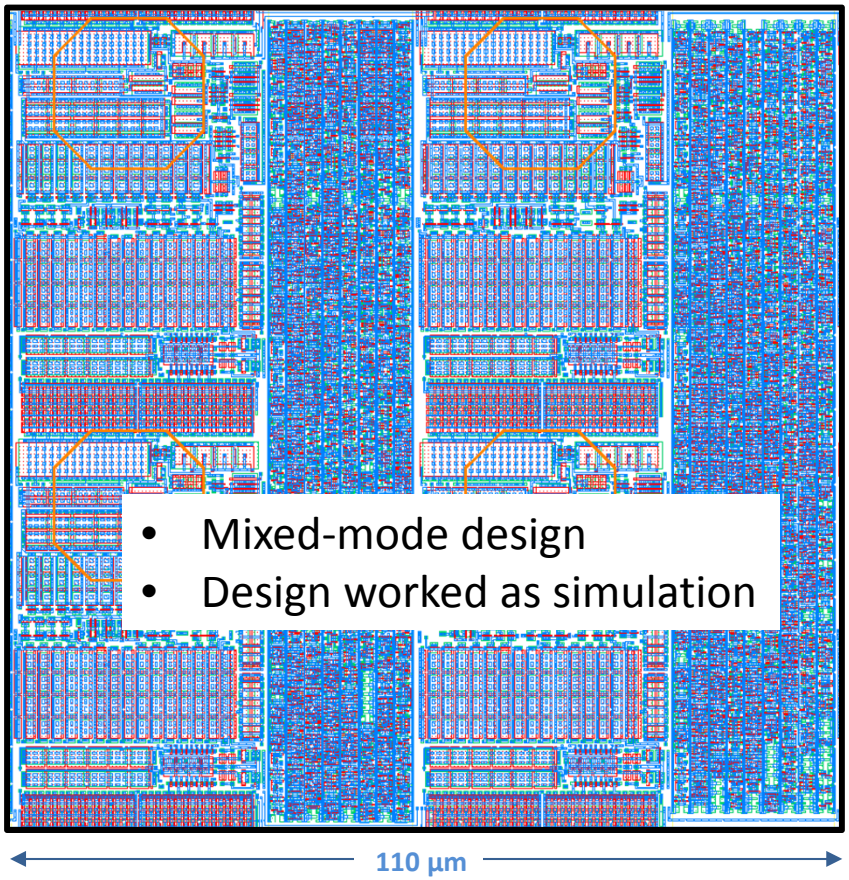
VDD	Temp	Leakage in cell	leakage In Chip
1.5 V	25 C	2.5 nW	**163 µW**
1.2 V	25 C	1.15 nW	**75.3 µW**

Medipix3.0 vs Medipix3RX

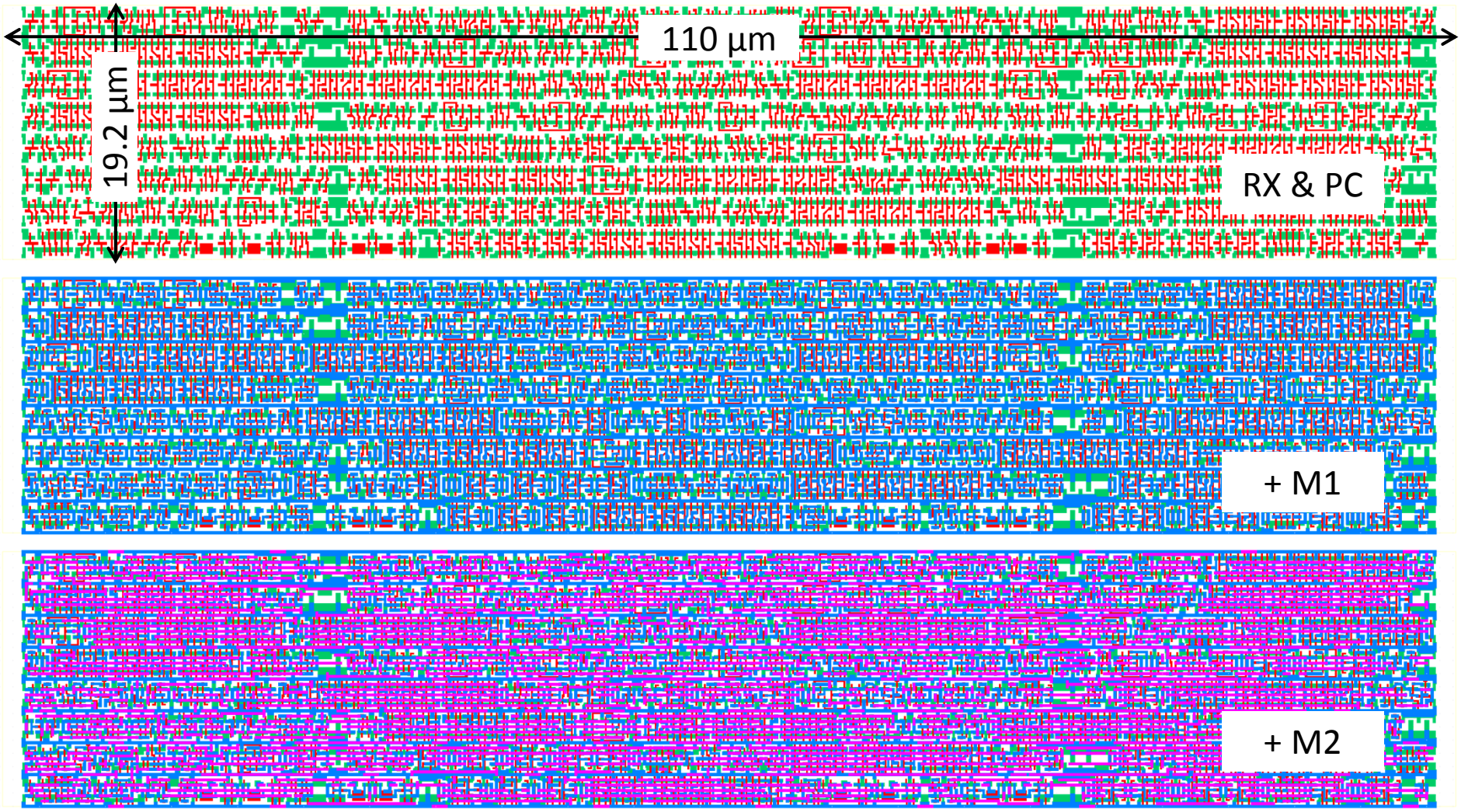
Medipix3.0 (2009)



Medipix3RX (2011) using SC_130nm_XL



Medipix3RX counter using SC_130nm_XL library



Medipix3 Counter Specs

	Medipix3.0	Medipix3RX
Counter Type	Configurable Binary Ripple Counter with full dynamic range coverage	Configurable Linear Feedback Shift Register (LFSR) with full dynamic range coverage
Binary output	YES	NO
Data and Clock multiplexing	YES	NO
Pixel Counter Depths	2x1bit (1) 2x4bit (15) 2x12bit (4095) 1x24bit (16777216)	2x1bit (1) 2x6bit (63) 2x12bit (4095) 1x24bit (16777216)
Overflow control	YES (1,4,12 and 24 bits)	YES (1,6,12 and 24 bits)
LFSR Decoding	No	**1 bit** **6 bit** [5:0] mode LUT[64] → [5] ⊕ [4] **12 bit** [11:0] mode LUT[4096] → [11] ⊕ [5] ⊕ [3] ⊕ [0] **24 bit** [23:0] mode → 2x12 bit LUT[4096]
Glitch Control	Narrow Disc pulses (Hits at Threshold) Multi-Hit Shutter ON/OFF	Narrow Disc pulses (Hits at Threshold) Multi-Hit Shutter ON/OFF
Layout	Full custom	Synthesized + automatic Place and Route
Routing	Local Routing: M1 to M4 Global Routing: M5 Power distribution: M6,M7,M8	Local Routing: M1 to M4 Global Routing: M4 Power distribution: M5,M6,M7
Layout size per pixel (mean)	~1100 μm^2 (~36.4% pixel area)	~1075 μm^2 (~35.6% pixel area)

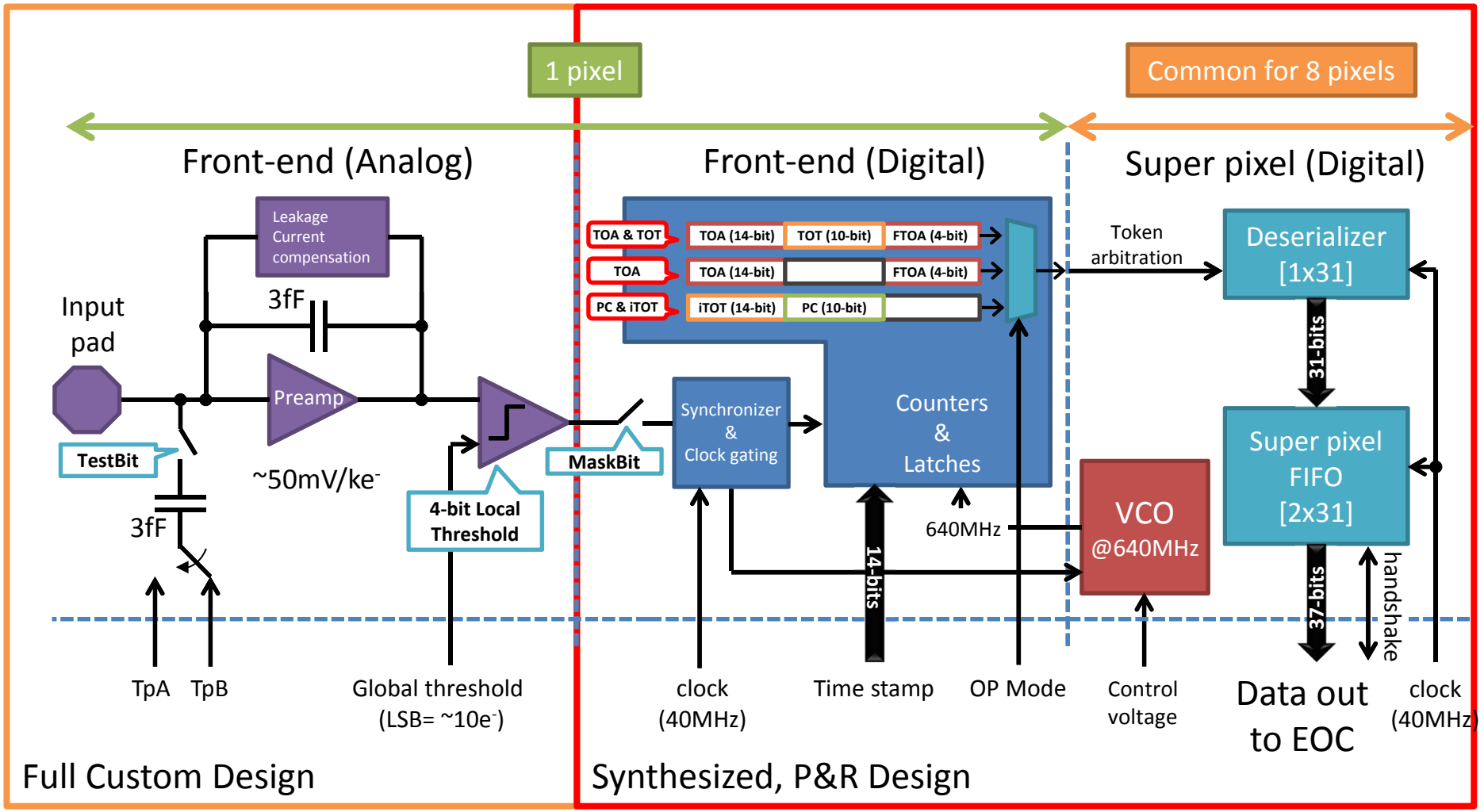
Timepix3 motivation

- Main driving requirements:
  1. Simultaneous TIME (TOA) and CHARGE (TOT) information per pixel
  2. Minimize dead time → Event-by-event readout and 0-supressed
  3. Monotonic TOT in both detection polarities
  4. Improve time measurements resolution

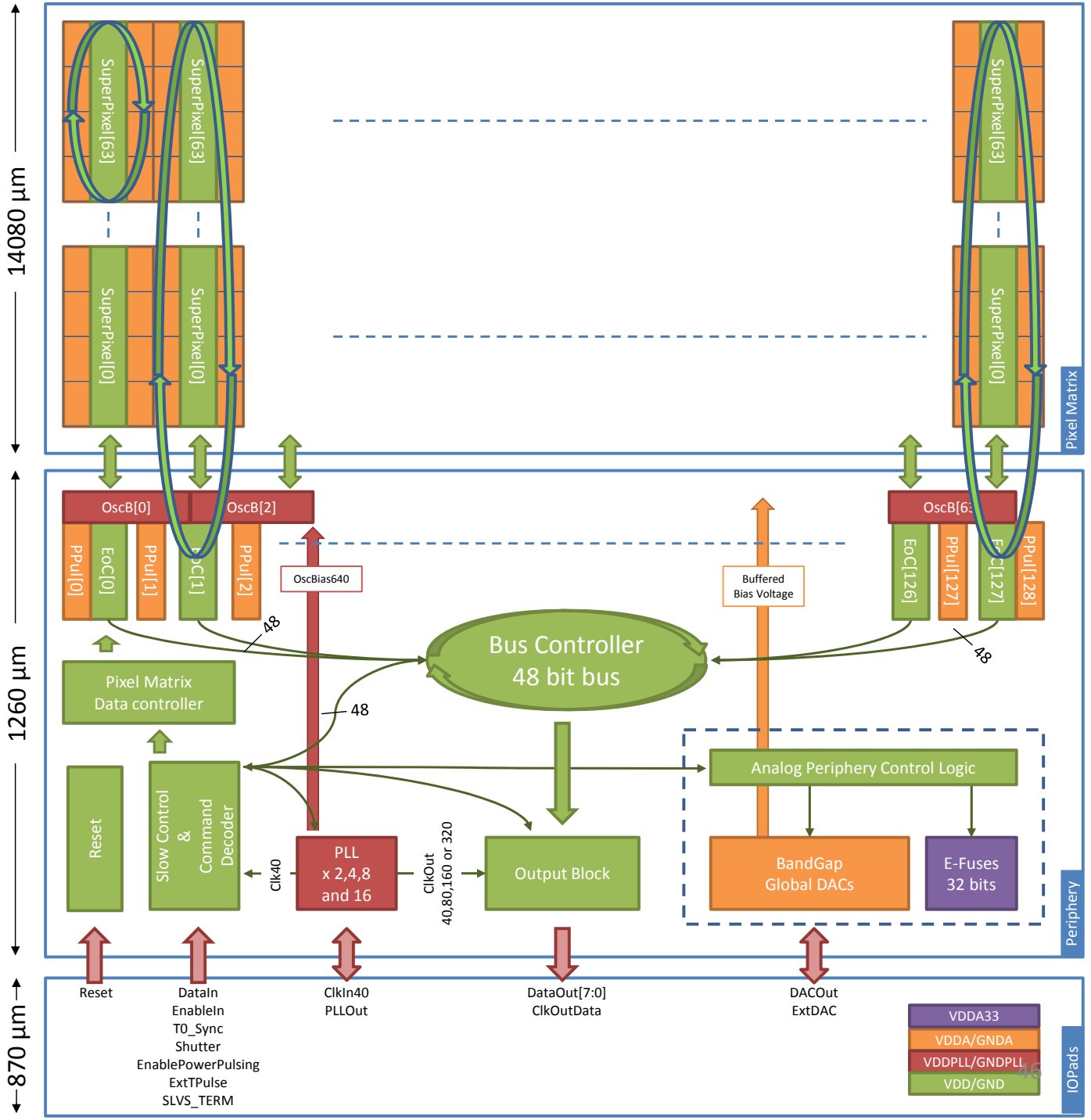
- Experience gained in the design of the Medipix3 chip (2009):
  - Technology (130nm CMOS)
  - Building blocks recycled (CERN's HD Standard Cell library, DACs, …)

- Designed by CERN, Nikhef and Bonn University with the support of the Medipix3 Collaboration

Timepix3 Pixel Schematic



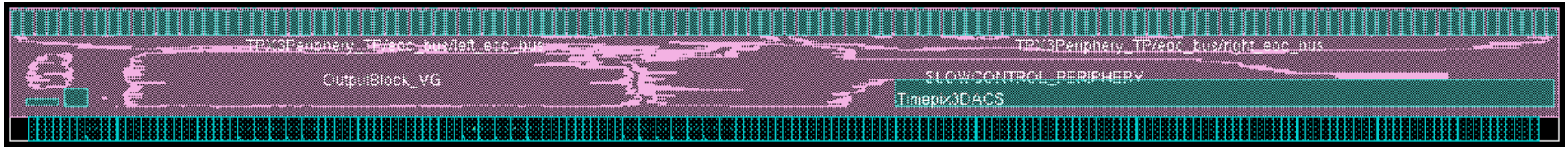
Timepix3 Floorplan



20th May 2015

Timepix3 Periphery Integration

- Digital on top design
- All analog full custom blocks require previous instantiation in Encounter:
  - Abstract view → LEF, DEF
  - Liberty Library (.lib)
  - Liberate AMS not used (it didn't exist) but it would have help if available…
- The full periphery can be then integrated together inside the digital flow
- Full digital data path verification (with SDF) from discriminator output (pixel) to output IO pads



Timepix3 Layout

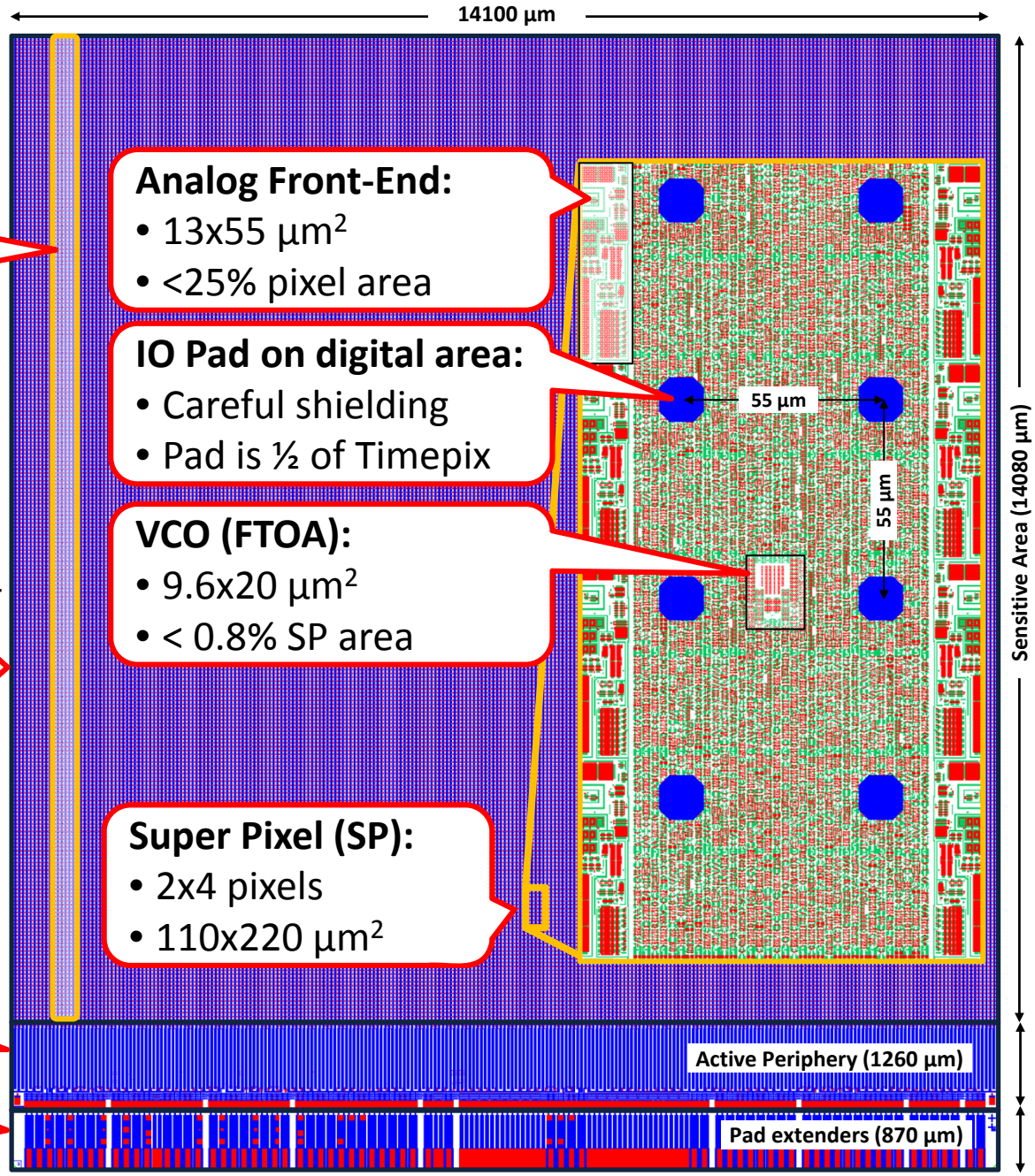
Double column:
2x256pixels
64 super pixels

Full Pixel Matrix:
256x256 pixels
128 double columns

8192 VCOs
(640MHz)
177 Mtransistors

Active Periphery

Pad Extenders:
Removed if TSV

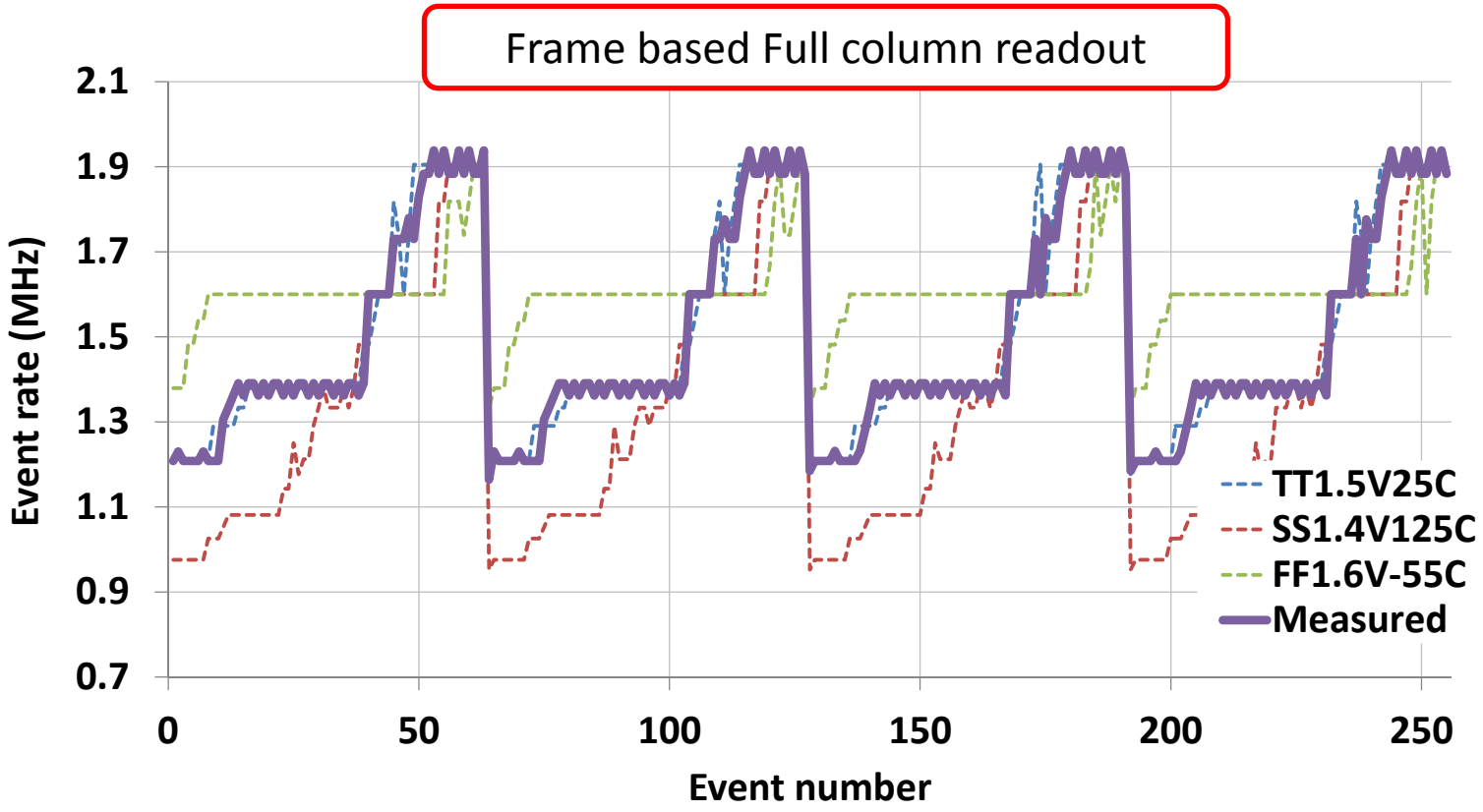


Timepix3 Pixel Logic design strategy

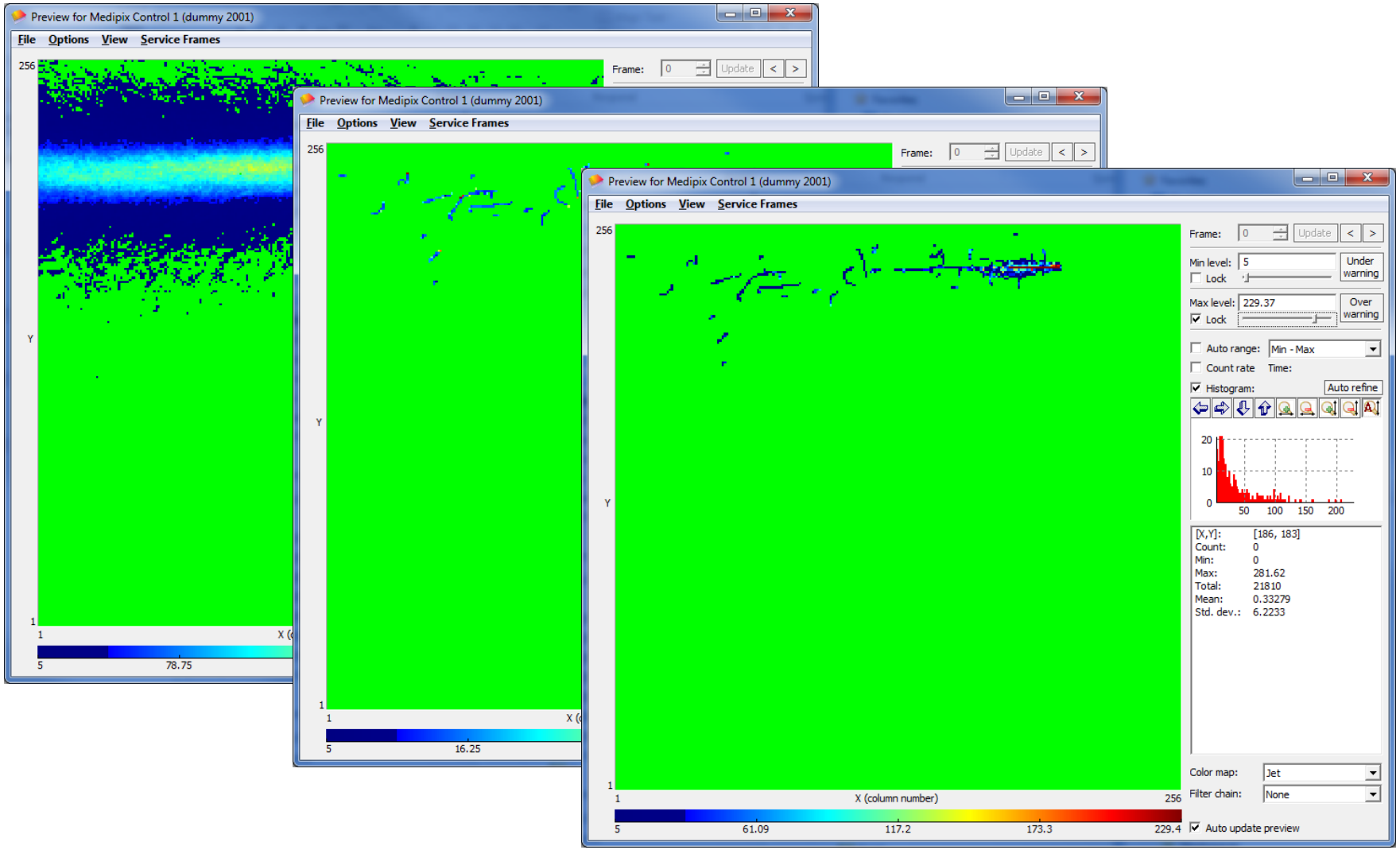
- Use CERN HD Library
- Special cells:
  - CLK_Q_RVT_XL: Fully integrated inside HD Library
  - VCO:
    - Abstract view
    - Capacitive IO definitions inside dedicated .lib file
- Full digital chip verification
  - SDF signoff back annotated delays
  - 3 corners:
    - 1.5V tt 25C
    - 1.6V ff -55C
    - 1.4V ss 125C

Pixel Data readout

- Data readout in Data Driven and Frame Based readout works as predicted in post-layout digital simulations



Timepix3 testbeam data



Swiss technology comparison

Timepix (2006)

Timepix3 (2013)

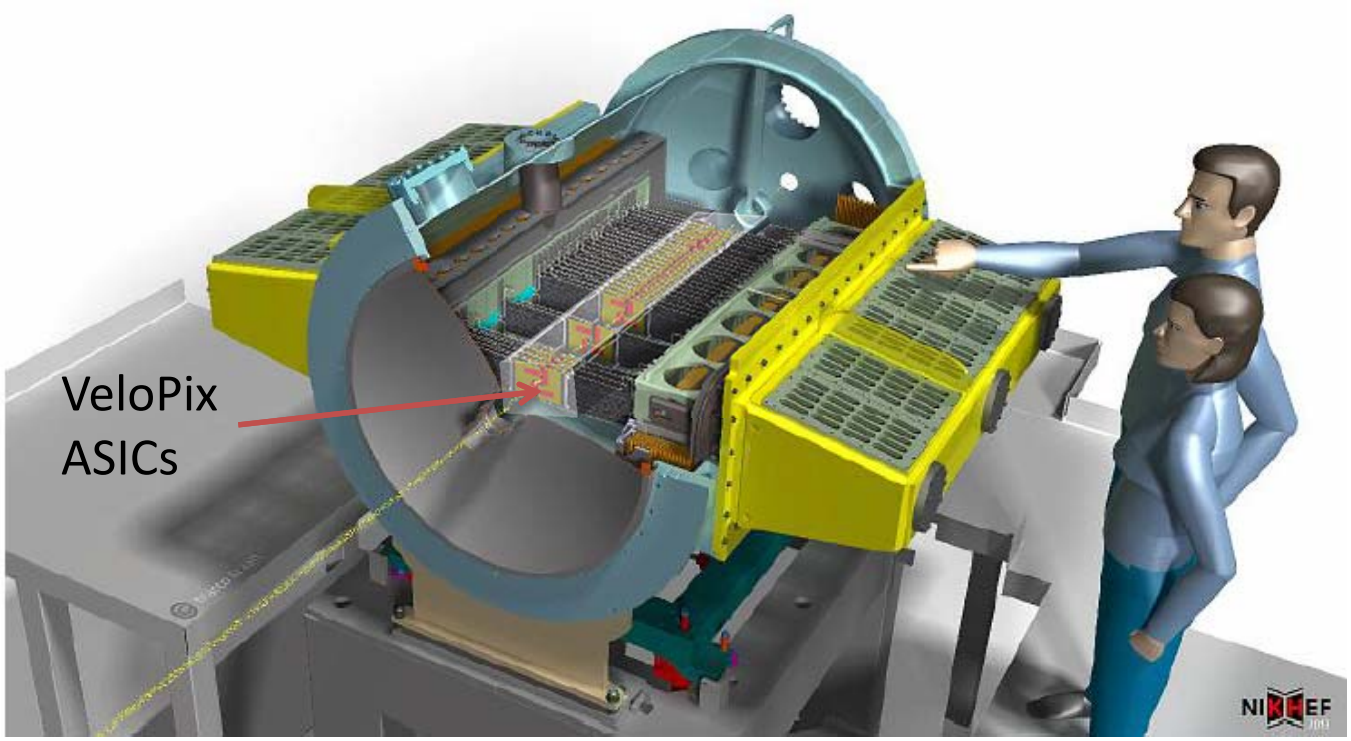
Introduction

T.Poikela

- VeloPix: Hybrid pixel detector (HPD) Readout ASIC for the LHCb VELO upgrade
- The ASIC reads out all bunch crossings at 40 MHz
- Installation planned for LS2

<u>The VELO upgrade:</u>

- Approx. 2.85 Tbit/s
- 26 module pairs
- 624 ASICs
- 41 Mpixels*



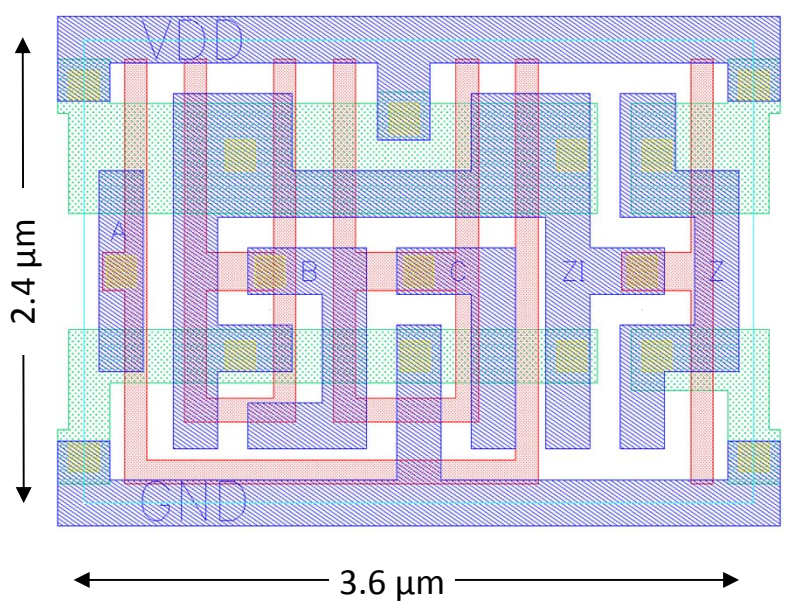
*Nokia Lumia 1020 (2013) also has 41 MP camera

VELOpix

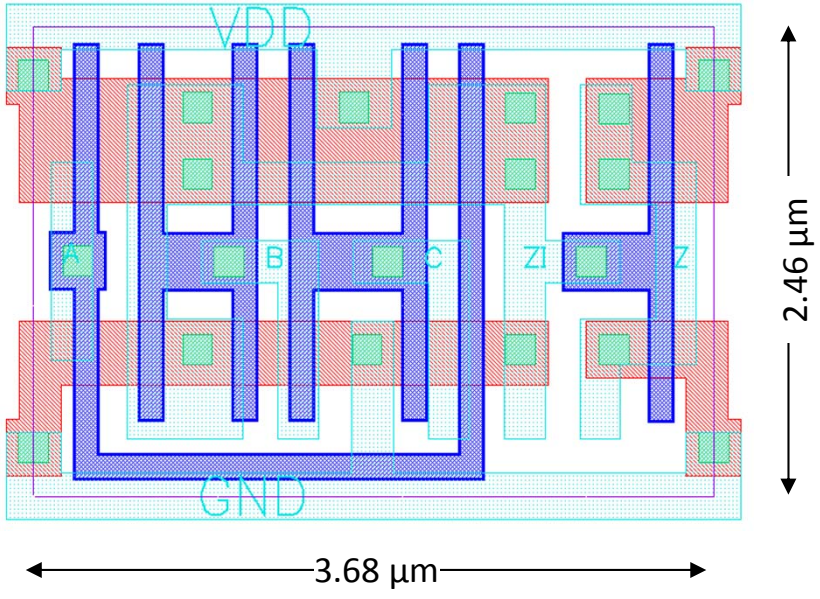
- Foreseen for pixel upgrade to LHCb VELO (Installation during LS2)
- Based heavily on experience with Timepix3 but with some important differences
  - Up to 800Mhits/s/chip (cf 80Mhits/s/chip)
  - 4 serialisers at 5Gbps each → binary hit information only
- Change of technology decided 1 year ago: IBM 130nm → TSMC 130nm
  - New HD library developed (IBM → TSMC):
    - Power supply: 1.5 V → 1.2 V
    - Devices used PMOSHVT/NMOSHVT → PMOS/NMOSHVT improves radiation robustness and balances driving strength
    - Track pitch: 0.4 x 0.4 µm → 0.41 x 0.46 µm due to different min pitch rules and via sizes
- Status:
  - Submission of full chip foreseen in Q2 2015

IBM → TSMC

- >60 cells in CERN TSMC HD library
- Characterized and verified with Liberate tool



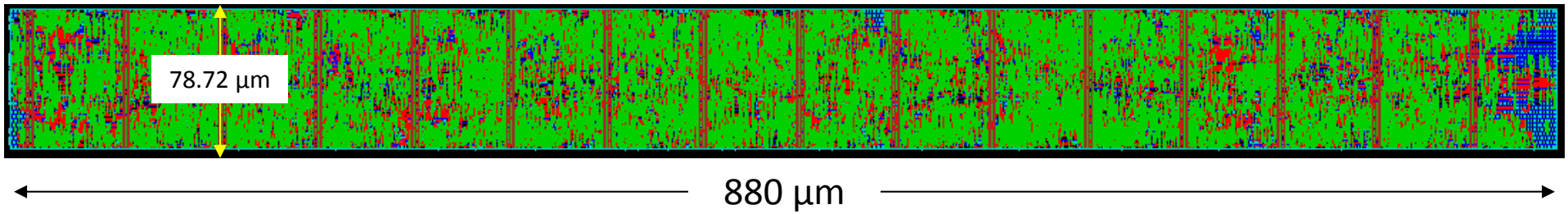
VOTERI_B_IBM_XL
(VeloPix)



VOTERI_B_TSMC_XL
(VeloPix)

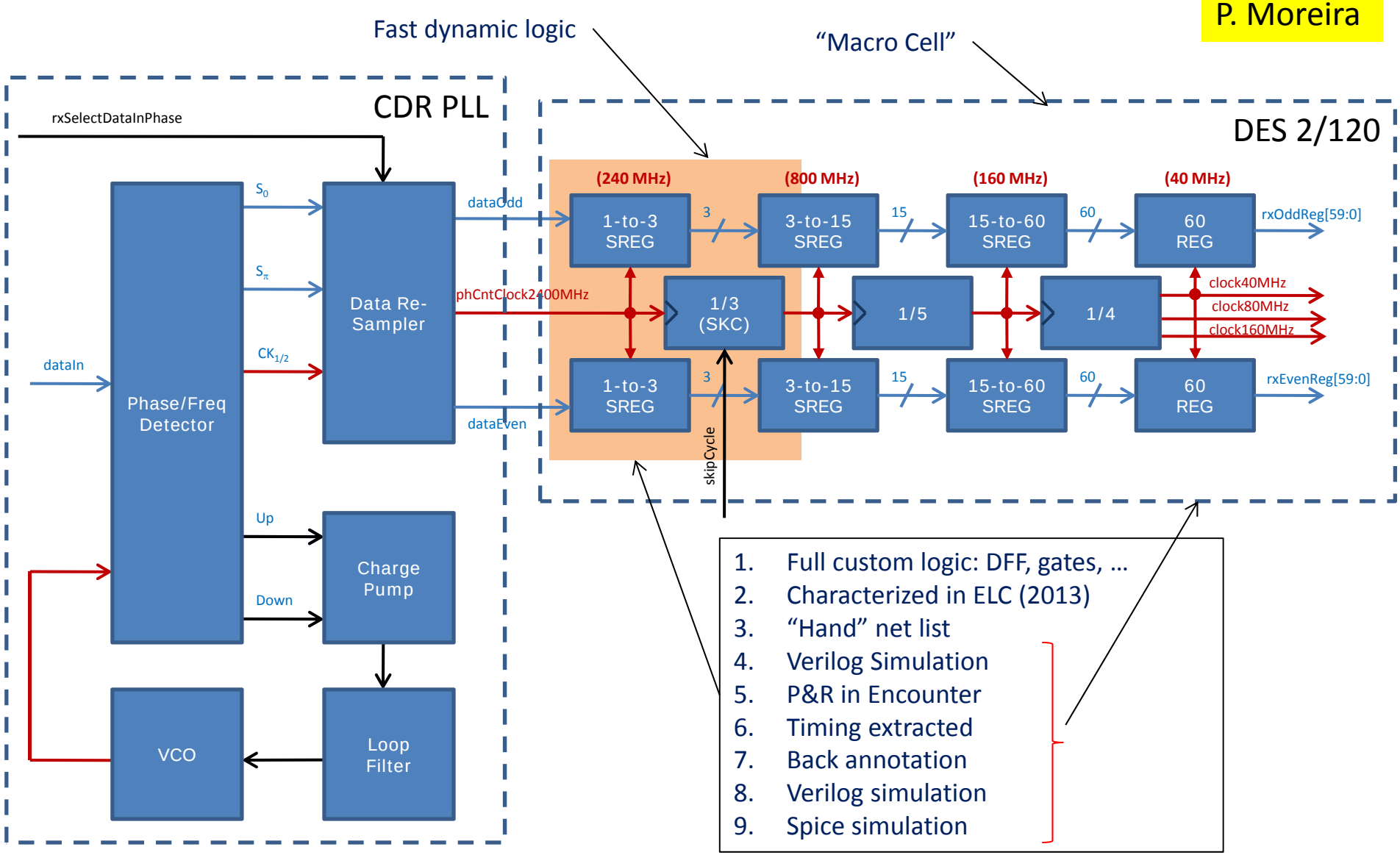
VELOpix MultiSuperPixel 2x16 pixels

- TMR in pixel configuration bits and in all control logic
- M1-M4 local and global routing and M5-M8 dedicated to power distribution
- ~5000 gates
  - 35% memory (flip-flops or latches)
  - 55% logic
  - 311 voters
- 4% smaller that Timepix3 pixel digital block → 2.88µm extra for the analog FE block.



DESerializer Architecture in the GBTx

P. Moreira



ALPIDE matrix read-out circuit

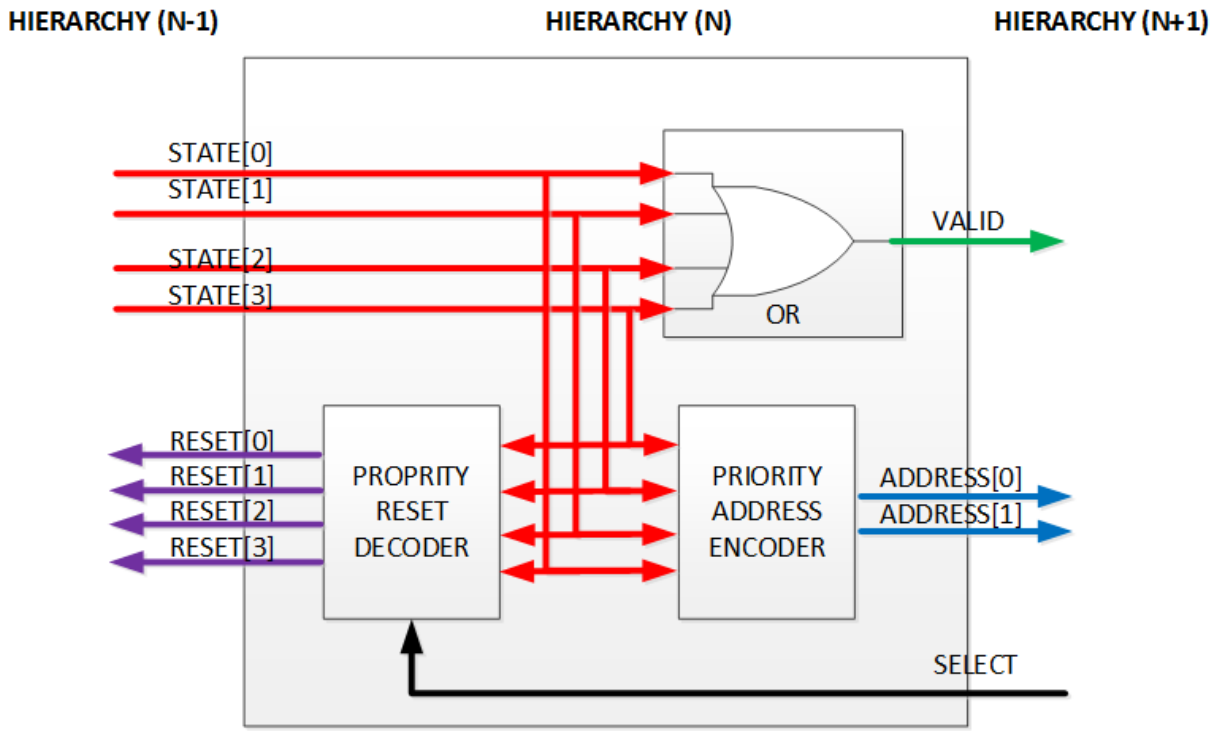
C. Marin

- ALPIDE is the new ultra low-power, CSA-based MAPS for the ALICE Upgrade
- ALPIDE matrix read-out circuit → Motivation
  - Zero Suppressed Read-out
    - Read-out only the fired pixels→ Encoded Address
  - Read-out on the same pixel area
  - Fast Read-out
  - Low Power

C. Marin

PRIORITY ENCODER READ-OUT

Basic block



COMPARISONS & DIFFERENCES BETWEEN

Implementation in pALPIDE V1 & V2

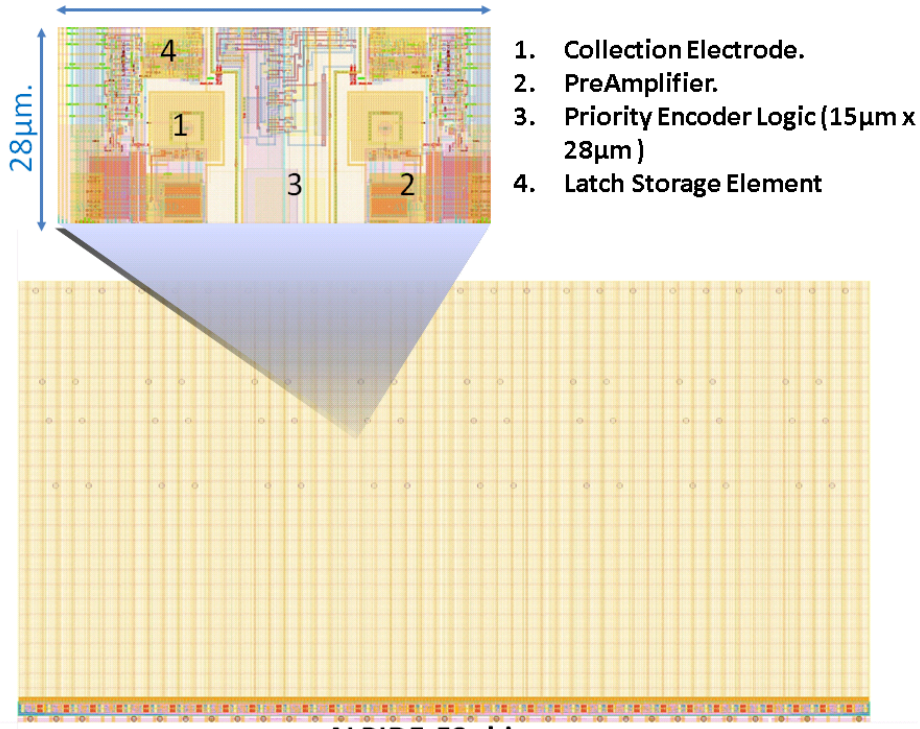
- Pixel size: 28 x 28 µm^2
- 1024 columns x 512 rows

- Full custom → **ELC and ALTOS**
  - Custom latch as Storage Element.
  - Priority encoder circuit per double column

- Fired pixels address encoded of 10 bits every clock cycle
- Max. Freq 10 Mhz

- Manual placement & Routing on 4 Metal Layers.

- Characterization used only for digital simulations
- Net capacitance extracted using liberty files and generating SDF for post layout simulations



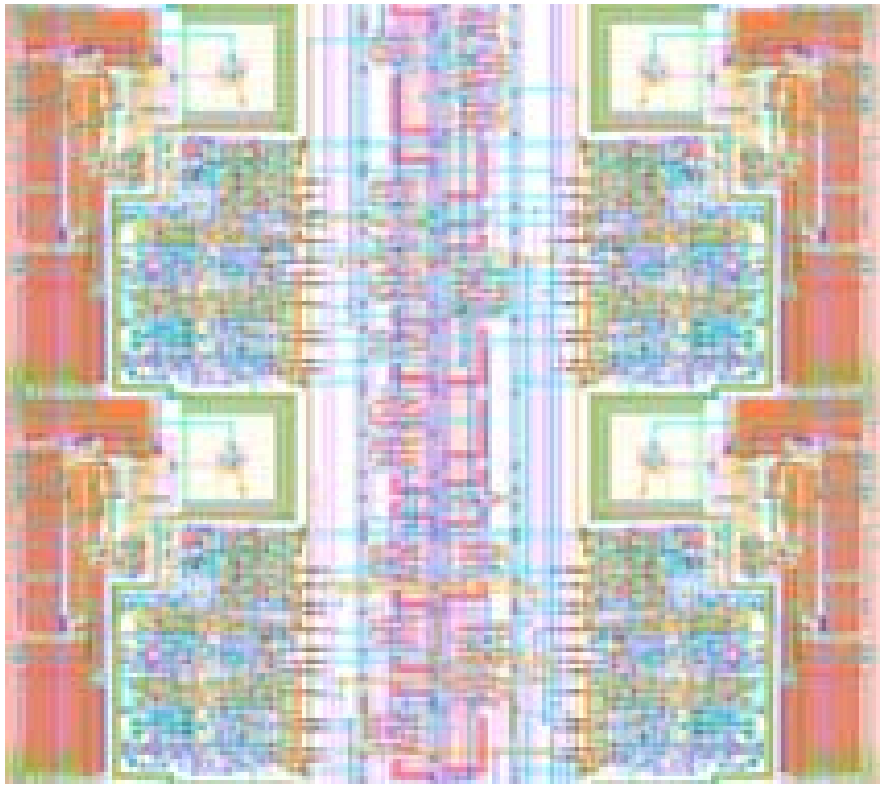
C. Marin

ALICE

COMPARISONS & DIFFERENCES BETWEEN

Implementation in pALPIDE V3

- Pixel size: 29.24 x 26.88 μm^2
- 1024 columns x 512 rows
- 3 Full custom FF as Storage Element → ALTOS
- Priority encoder circuit per double implemented with TowerJazz STD cells library
- Fired pixels address encoded of 10 bits every clock cycle
- Max. Freq 50 Mhz
- Seimi-automatic Placed & Routed on 4 Metal Layers
  - Analog → Full Custom
  - Digital →P&R flow
- The custom FFs has been characterized modeled and simulated on digital environment
- Digital pixel logic implemented using standard cells from ToweJazz



Conclusions

- What library characterization enables?
  - Mixed-mode designs → true digital verification
  - Allows modification or addition of new cells in the digital library
  - Complete std cell provider library files → add dynamic power, noise…
  - Add specific PVT corner to match application operating condition (radiation environment)
  - Integrate analog blocks inside digital flow (Liberate AMS)

- For analog designers (my case) helps to understand/trust the digital flow…

- All these tools are based in trusted device models and parasitic circuit extractions:
  - Silicon verification is required if any of above is not trusted

Acknowledgments

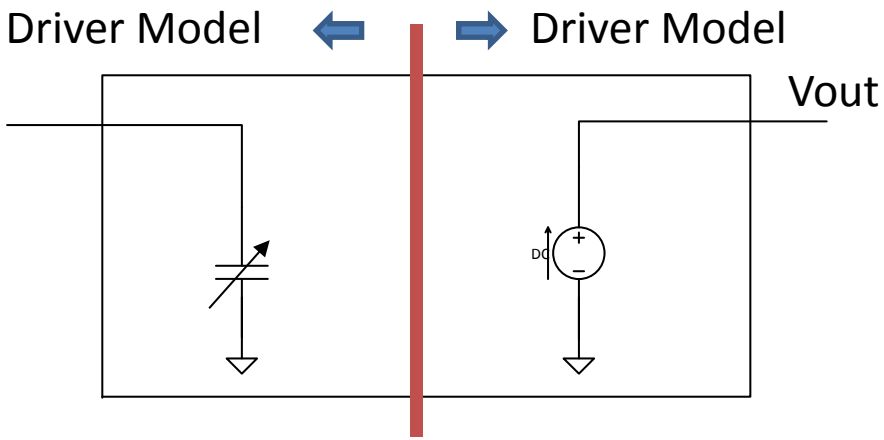
- T. Hemperek (Bonn University) transfer me his ELC knowledge used in FEI4 chip
- S. Bonacini (CERN) help me through out digital flow understanding
- P. Moreira and C. Marin

- Cadence for allowing to use some of their original material

Spare slides

ECSM model

- ECSM: effective current source model:
  - Uses characterized measurements of current and voltage (I/V curves) over multiple time intervals, with different combinations of input slew and output loading capacitance.
  - These I/V curves are used to create a more accurate output driver model, where each driver is represented as a voltage-controlled current source.
- Models the effects of timing, noise, power, and variation



Timing Cell Arc Concepts

- All delay information in a library refers to an input-to-output pin pair or an output-to-output pin pair defined as:
  - intrinsic delay: The fixed delay from input to output pins
  - transition delay: The time it takes the driving pin to change state. Transition delay attributes represent the resistance encountered in making logic transitions
  - slope sensitivity: The incremental time delay due to slow change of input signals