Timing Capture

Comprehensive Delay Calculation Flow for Programmable Devices

XILINX.

Authors

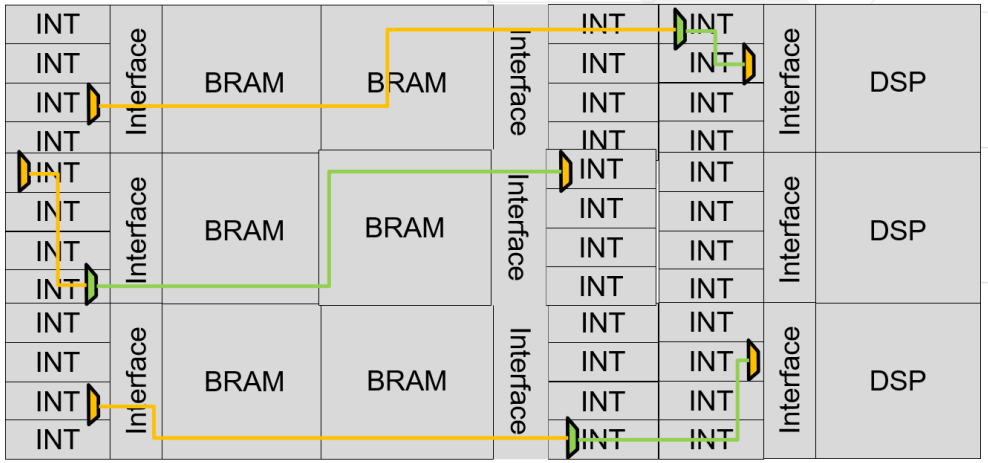
- **Nitin Navale**
  CAD Manager, Xilinx
- **Sumukh Nanjangud**
  CAD Engineer, Xilinx

Motivation

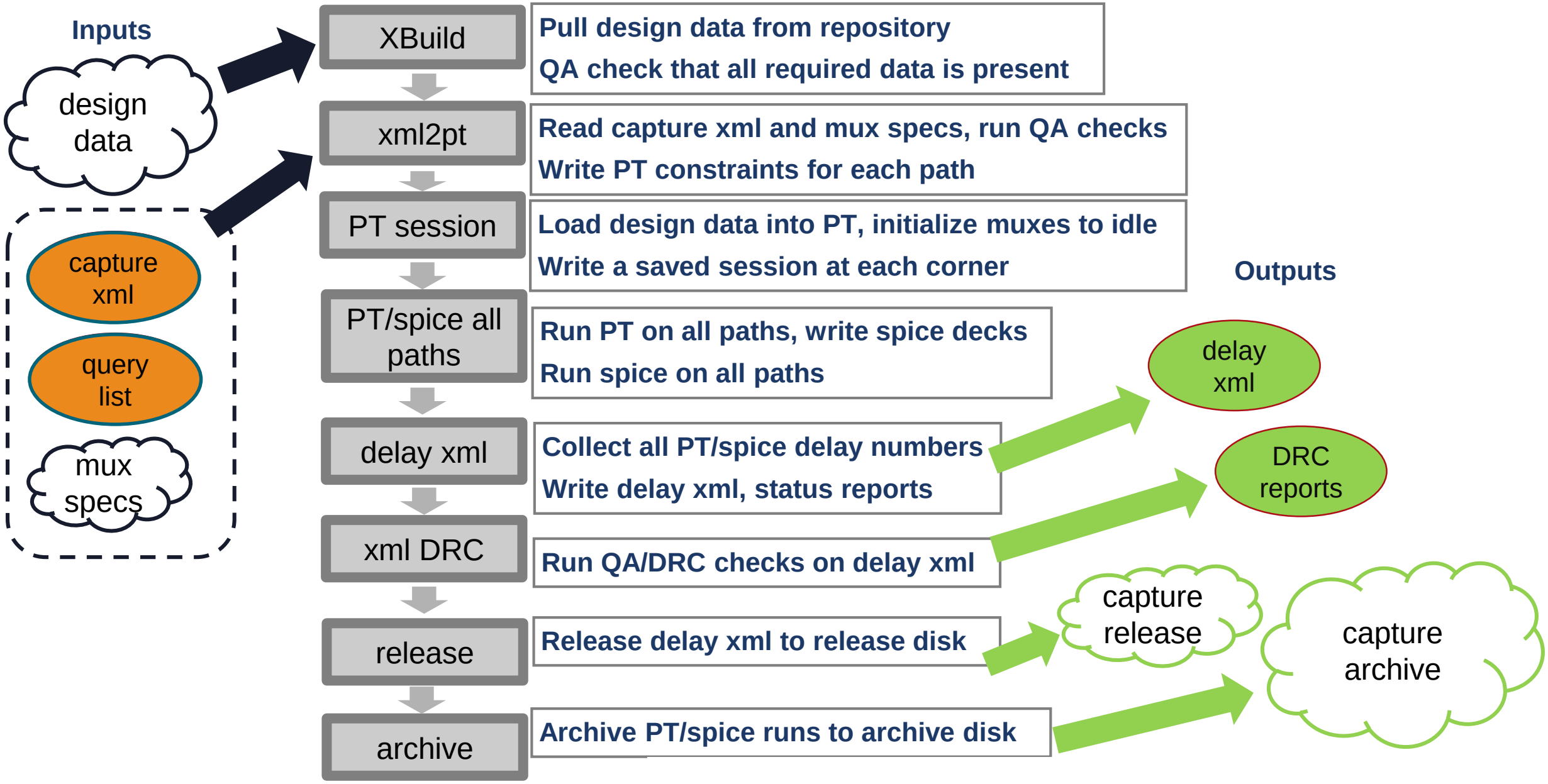
- **Xilinx provides all-programmable solutions to customers as a combined hardware/software package**
- **The software (Vivado) is required to program the chip as per the customer requirements**
  - To program effectively, Vivado needs to be aware of the physical layout of the chip, as well as the point-to-point delays between PIPs (**P**rogrammable **I**nterconnect **P**oints)
- **Need a chip-level flow that calculates these delays with SPICE accuracy**
  - Must handle complexity of scale for a full chip
  - Should be able to run during all phases of design process from early schematic through final tapeout

Timing Capture Methodology

- **Timing Capture is the methodology for measuring chip-level path delays across PIPs**
  - Delays are calculated using a mix of PrimeTime and HSPICE
  - Use real IP data (Verilog, spef, liberty, etc.), so delay accuracy matures as the IPs mature
  - Use chip partitioning and LSF distribution to handle scalability
  - Vivado consumes these delays and references them as it programs the chip
- **An automated flow for running Timing Capture on chip-level PIPs (multiplexers)**
  - Pulls IP data from release repository
  - Imports path information in xml format from Vivado team
  - Creates PT constraints per path with “set_case_analysis” commands
  - Runs PT on a subset of signoff corners to calculate path delays
  - Writes SPICE deck using PT and calculates SPICE delays
  - Packages up path delays per corner for Vivado consumption

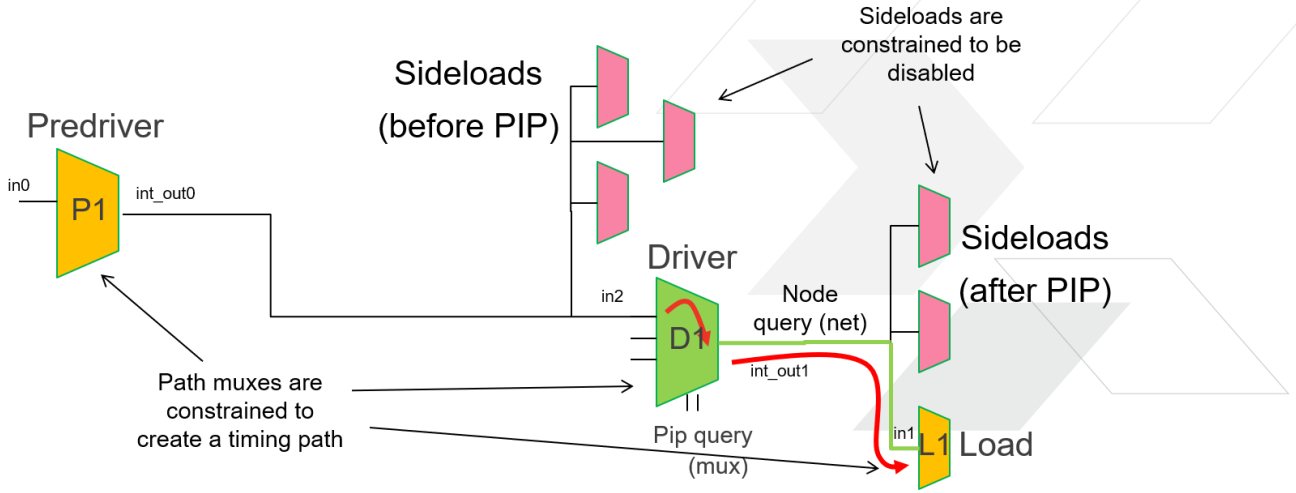


Timing Capture flow chart



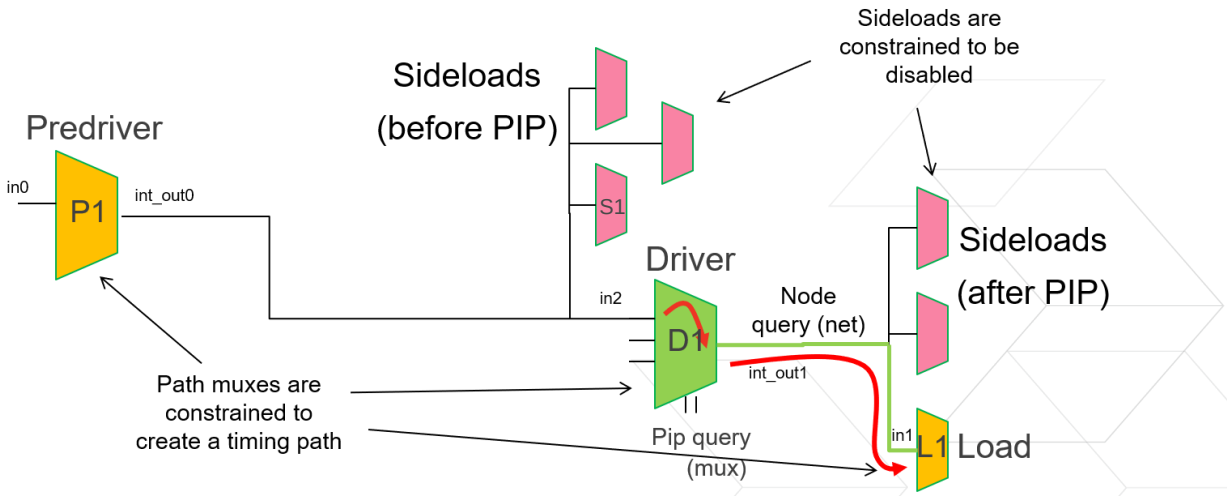
Not a “vanilla” STA run

- **No sequential elements (flops, latches etc.) on query paths**
  - No timing against clocks
  - Simply calculating point-to-point delays
- **Each path query measures the delay across at least one PIP (mux) and Node (net)**
  - A single query could potentially contain a series of Pip/Node combinations
  - Each is driven by one or more Predriver muxes (for context) and will feed into one or more Load muxes
- **Need to re-constrain PT for each path**
  - Chip contains over 1M+ paths
  - Paths must be timed independently and may intersect with each other
    - One path’s Driver may be another path’s Sideload
  - Unique constraints required for each path to ensure clean delay calculation
  - Some path grouping done to improve throughput



Single Path Analysis

- **Delay calculation from D1/in2 → L1/in1**
- **Treat Predriver P1 data input as timing start point**
  - `set_input_delay 0 P1/in0`
  - `set_annotated_transition 0.040 P1/in0`
- **Enable path through Driver D1 for in2 → int_out1**
  - `set_case_analysis 0 D1/mc_sel[0]`
  - `set_case_analysis 0 D1/mc_sel[1]`
  - `set_case_analysis 1 D1/mc_sel[2]`
  - `set_case_analysis 0 D1/mc_sel[3]`
- **Disable Sideloads by turning OFF their select lines**
  - `set_case_analysis 0 S1/mc_sel*`
- **Capture delays from PrimeTime and HSPICE**
  - `write_spice_deck` command generates SPICE deck for HSPICE simulation




```
## All delays reported in picoseconds (ps)
#<corner>                       <edge>   <spice delay>  <PT delay>     <status_flags>
ssgnp_650_m40_rct_cap_a         RR       98.6           93.379         SPICE_PASS,PTSPICE_WITHIN_10%
                                FF       99.32          95.288         SPICE_PASS,PTSPICE_WITHIN_10%
```

- **Reset path by disabling all select lines to put all PIPs in idle state, before constraining the next path**

PT constraints

Set input constraints at the input of Predriver

```
#INPUT CONSTRAINTS at the Predriver input
# PATH_INDEX=0: Instance=Imega_int_bot_X8Y0_R0/Iint_bot_ft_X0Y18_R0/Iint_core_X0Y0_R0/I1/Iint_long_column/I7/I3,
Pin=in0, Master=int_long_atom
set_input_delay 0 Imega_int_bot_X8Y0_R0/Iint_bot_ft_X0Y18_R0/Iint_core_X0Y0_R0/I1/Iint_long_column/I7/I3/in0
set_annotated_transition 0.040
Imega_int_bot_X8Y0_R0/Iint_bot_ft_X0Y18_R0/Iint_core_X0Y0_R0/I1/Iint_long_column/I7/I3/in0
```

Set all the constraints to disable side loads & to activate the query path

```
#Disable sideloads
## INSTANCE Imega_int_bot_X20Y0_R0/Iint_bot_ft_X0Y18_R0/Iint_core_X0Y0_R0/I1/Iint_long_column/I7/I3
## MASTER int_long_atom
# ARC in6 -> int_out0 (no_delay), SIDELOAD, pin_in {in2} disabled
set_case_analysis 0 Imega_int_bot_X20Y0_R0/Iint_bot_ft_X0Y18_R0/Iint_core_X0Y0_R0/I1/Iint_long_column/I7/I3/mc_sel[0]
set_case_analysis 0 Imega_int_bot_X20Y0_R0/Iint_bot_ft_X0Y18_R0/Iint_core_X0Y0_R0/I1/Iint_long_column/I7/I3/mc_sel[1]
set_case_analysis 0 Imega_int_bot_X20Y0_R0/Iint_bot_ft_X0Y18_R0/Iint_core_X0Y0_R0/I1/Iint_long_column/I7/I3/mc_sel[2]
set_case_analysis 0 ,,,,,,,,,,,,, ,,,,,,
,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,
# Activate query path mux
## INSTANCE Imega_int_bot_X20Y0_R0/Iint_bot_ft_X0Y20_R0/Iint_core_X0Y0_R0/I1/Iint_long_column/I0/I0
## MASTER int_long_atom
# ARC in2 -> int_out0 (no_delay), QUERY PATH, path_index=3, full arc defined
set_case_analysis 0 Imega_int_bot_X20Y0_R0/Iint_bot_ft_X0Y20_R0/Iint_core_X0Y0_R0/I1/Iint_long_column/I0/I0/mc_sel[0]
set_case_analysis 0 Imega_int_bot_X20Y0_R0/Iint_bot_ft_X0Y20_R0/Iint_core_X0Y0_R0/I1/Iint_long_column/I0/I0/mc_sel[1]
set_case_analysis 1 Imega_int_bot_X20Y0_R0/Iint_bot_ft_X0Y20_R0/Iint_core_X0Y0_R0/I1/Iint_long_column/I0/I0/mc_sel[2]
set_case_analysis 0 Imega_int_bot_X20Y0_R0/Iint_bot_ft_X0Y20_R0/Iint_core_X0Y0_R0/I1/Iint_long_column/I0/I0/mc_sel[3]
set_case_analysis 0 Imega_int_bot_X20Y0_R0/Iint_bot_ft_X0Y20_R0/Iint_core_X0Y0_R0/I1/Iint_long_column/I0/I0/mc_sel[4]
set_case_analysis 0 Imega_int_bot_X20Y0_R0/Iint_bot_ft_X0Y20_R0/Iint_core_X0Y0_R0/I1/Iint_long_column/I0/I0/mc_sel[5]
set_case_analysis 0 Imega_int_bot_X20Y0_R0/Iint_bot_ft_X0Y20_R0/Iint_core_X0Y0_R0/I1/Iint_long_column/I0/I0/mc_sel[6]
set_case_analysis 0 Imega_int_bot_X20Y0_R0/Iint_bot_ft_X0Y20_R0/Iint_co…………………………………………..
…………………………………………………………………..
```

write_spice_deck from PT

```
* Grounded coupling capacitors for nets 'Imega_int_bot_X20Y0_R0/if_vbus_y5x0.sbus[0]-if_hbus_outer__4__.wbus[0]'.
c03601  Imega_int_bot_X20Y0_R0/if_vbus_y5x0.sbus[0]:1   0       4.63889e-18
c03602  Imega_int_bot_X20Y0_R0/if_vbus_y5x0.sbus[0]:1   0       1.29857e-19
c03603  Imega_int_bot_X20Y0_R0/if_vbus_y5x0.sbus[0]:2   0       2.61546e-19
****************************************
:
:
* voltage source for Imega_int_bot_X20Y0_R0/Iint_bot_ft_X0Y3_R0/Iint_core_X0Y0_R0/I1/Iint_sdq_atom_column_e/I15/I3/in4
(victim)
* Set due to side pin of load cell.
vImega_int_bot_X20Y0_R0/Iint_bot_ft_X0Y3_R0/Iint_core_X0Y0_R0/I1/Iint_sdq_atom_column_e/I15/I3/in4
Imega_int_bot_X20Y0_R0/Iint_bot_ft_X0Y3_R0/Iint_core_X0Y0_R0/I1/Iint_sdq_atom_column_e/I15/I3/in4       0       0
* voltage source for Imega_int_bot_X20Y0_R0/Iint_bot_ft_X0Y3_R0/Iint_core_X0Y0_R0/I1/Iint_sdq_atom_column_e/I15/I3_C_SPC0/in0
(victim)
* Set due to side pin of load cell.
vImega_int_bot_X20Y0_R0/Iint_bot_ft_X0Y3_R0/Iint_core_X0Y0_R0/I1/Iint_sdq_atom_column_e/I15/I3_C_SPC0/in0
Imega_int_bot_X20Y0_R0/Iint_bot_ft_X0Y3_R0/Iint_core_X0Y0_R0/I1/Iint_sdq_atom_column_e/I15/I3_C_SPC0/in0        0       0
* voltage source for Imega_int_bot_X20Y0_R0/Iint_bot_ft_X0Y3_R0/Iint_core_X0Y0_R0/I1/Iint_sdq_atom_column_e/I15/I3_C_SPC0/in1
(victim)
* Set due to side pin of load cell.
vImega_int_bot_X20Y0_R0/Iint_bot_ft_X0Y3_R0/Iint_core_X0Y0_R0/I1/Iint_sdq_atom_column_e/I15/I3_C_SPC0/in1
Imega_int_bot_X20Y0_R0/Iint_bot_ft_X0Y3_R0/
:
```

Write_spice_deck generated from PT for HSPICE simulation

```
$DATA1 SOURCE='HSPICE' VERSION='O-2018.09-SP1 linux64' PARAM_COUNT=0
$OPTION MEASFORM=1
.TITLE ''
out_slew0 in_slew3 out_slew3 in_slew6 out_slew6 in_slew7 full_query temper alter#
 1.939e-11  2.572e-11  2.874e-11  4.403e-11  4.882e-11  4.968e-11  7.563e-11  -40.0000 1
```

mt0 measurements from HSPICE

Consolidated Delay Report (Sample)

```
## Path count summary
## Number of paths using each type of delay, organized per corner per edge type
#<corner>                <edge>   <# spice>   <# PT>   <# dummy>   <# total>
ssgnp_650_m40_rct_cap_a   RR        1251         0         749         2000
                          FF        1251         0         749         2000
                          RF           0         0           0            0
                          FR           0         0           0            0

## Query 471568
## nodepip__N_187479_NODE_HDOUBLE_0_1__P_74918__N_187517_NODE_HDOUBLE_0_1__P_76220__N_690_NODE_INODE_0_0
## Results dir: ./paths_47/paths_4715/path_471568
## Path QA: 3 bad sideload pins, 0 bad query pins, 0 invalid net parasitics
## All delays reported in picoseconds (ps)
#<corner>                <edge>   <spice delay>   <PT delay>   <status_flags>
ssgnp_650_m40_rct_cap_a   RR          81.93          106.372     SPICE_PASS,BAD_SIDELOADS
                          FF          81.35          105.842     SPICE_PASS,BAD_SIDELOADS

## Query 471569
## nodepip__N_187479_NODE_HDOUBLE_0_1__P_74918__N_187517_NODE_HDOUBLE_0_1__P_76237__N_688_NODE_SDQNODE_0_0
## Results dir: ./paths_47/paths_4715/path_471569
## Path QA: 3 bad sideload pins, 0 bad query pins, 0 invalid net parasitics
## All delays reported in picoseconds (ps)
#<corner>                <edge>   <spice delay>   <PT delay>   <status_flags>
ssgnp_650_m40_rct_cap_a   RR          77.56          75.81       SPICE_PASS,BAD_SIDELOADS
                          FF          76.72          75.554      SPICE_PASS,BAD_SIDELOADS
```

Analysis corner

Summary across all paths:

- 1251 paths report spice delay
- 0 paths report PT delay (all paths that succeeded in PT also succeeded in spice)
- 749 paths failed in PT and fall back to a placeholder delay value (99999)

Rise/fall delays (ps) for this path in PT and spice

Status debug flags:

- This path expected certain sideloads in capture xml that couldn't be traced in the actual design (BAD_SIDELOADS)

Grouping Mechanism

- Grouping of paths is done to improve throughput
  - Must ensure we don't group paths with conflicting constraints
  - Will apply a 2-tier grouping mechanism
- 1. Coarse grouping
  - Divide the total paths across multiple PT runs
  - No specific sort mechanism
  - Goal is to maximize parallelism
- 2. Non-Collision grouping
  - Within a single PT run, divide the paths into multiple calls to update_timing based on collisions
  - If a particular path's query atoms (in-path mux cells) or sideload atoms are present in another path, they are considered to be colliding paths.

Consider 10 paths to be timed:

- Path1 is found to have collision with path2, path3 and path10.
- Path2 is found to have collision with path3
- All other paths are non-colliding

group0: path1, path4, path5, path6, path7, path8 and path9

```
#group0
set PT constraints for group0 paths
update_timing
delay calculation point-to-point(group0 paths)
reset PT constraints

#group1
```

group1: path2,path10

```
set PT constraints for group1 paths
update_timing
delay calculation point-to-point(group1 paths)
reset PT constraints.

#group2
```

group2: path3

```
set PT constraints for group2 paths
update_timing
delay calculation point-to-point(group2 paths)
reset PT constraints.
```

Distributed LSF runs

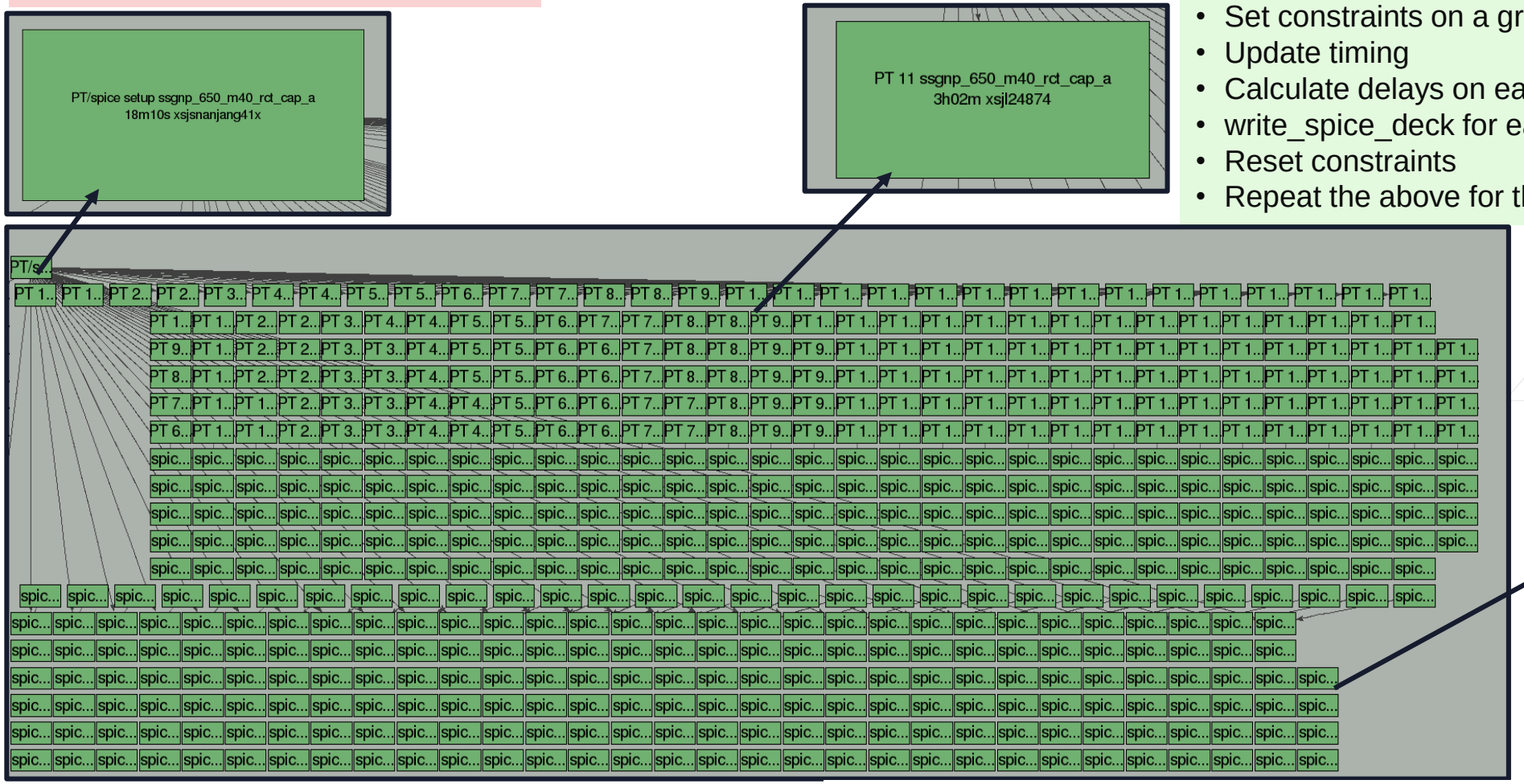
- **Each PT run contains several paths grouped into sets of non-colliding paths.**
- **PT writes out the SPICE deck using write_spice_deck**
- **Run HSPICE on each path**

Create PT session

- load design data (db, spef, verilog)
- update_timing
- save PT session
- exit

Calculate PT delays
Multiple groups of PT containing several paths.

- Restore previously saved session
- Multiple groups within the run arranged to avoid collision
- Set constraints on a group of paths
- Update timing
- Calculate delays on each path from a group
- write_spice_deck for each path
- Reset constraints
- Repeat the above for the next group



Calculate HSPICE delays

- Run HSPICE on PT generated spice deck

Future work

- **Full chip delay calculation needs faster turnaround time**
  - Test larger designs to push the limits of scalability and performance
  - Actively working with Synopsys to further improve PT capacity

- **Apply Timing Capture flow to other parts of the chip**
  - Current flow is applicable to full chip paths only
  - Test with lower level paths inside route blocks, which have messier PT constraints

Summary

> Issues encountered

- >> PT delays vary greatly based on the mux input pin capacitance
  - – Pin cap could vary as much as 10x-40x depending on ON/OFF state of the inputs
  - – NLDM libs fail to capture this effect; CCS libs needed (characterized using NanoTime)
- >> Tool bug in NanoTime to capture the OFF state pin cap
  - – Fixed by Vendor
- >> Tool bug in PrimeTime resetting mux pin case analysis after delay calculation
  - – Caused bugs in “write_spice_deck” netlist, preventing SPICE from toggling rail-to-rail

> Benefits of this flow

- >> Able to successfully time over 1M+ paths in the entire chip
- >> Scale and complexity issues handled smoothly by distributing the jobs over LSF
- >> Signoff accurate vs. our internal delay estimation flow
  - – Estimation flow correlated to be 9% pessimistic with 3-sigma range of ±16%