Constructive Computer Architecture

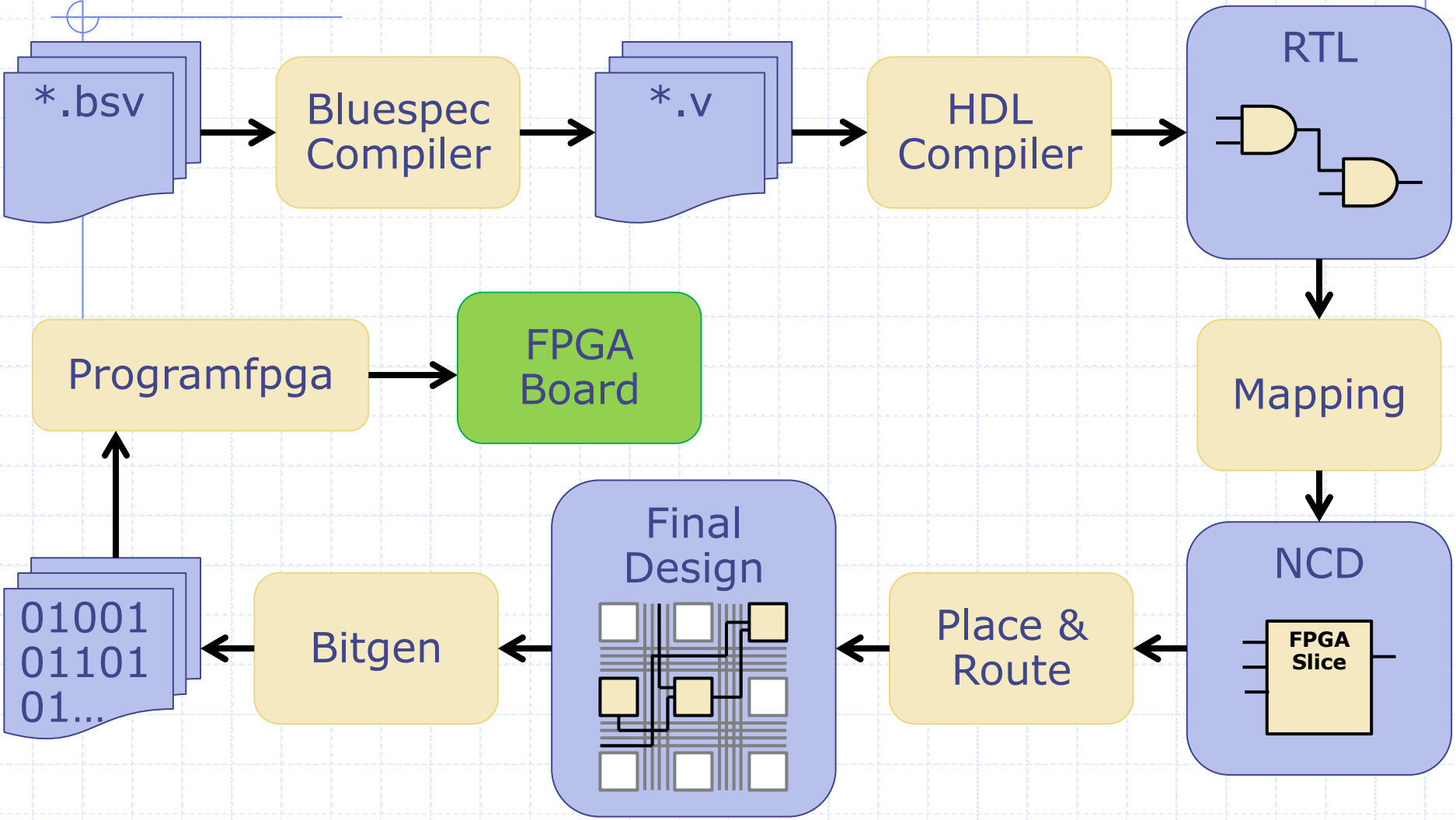
Tutorial 8: FPGA Synthesis

Andy Wright
6.S195 TA

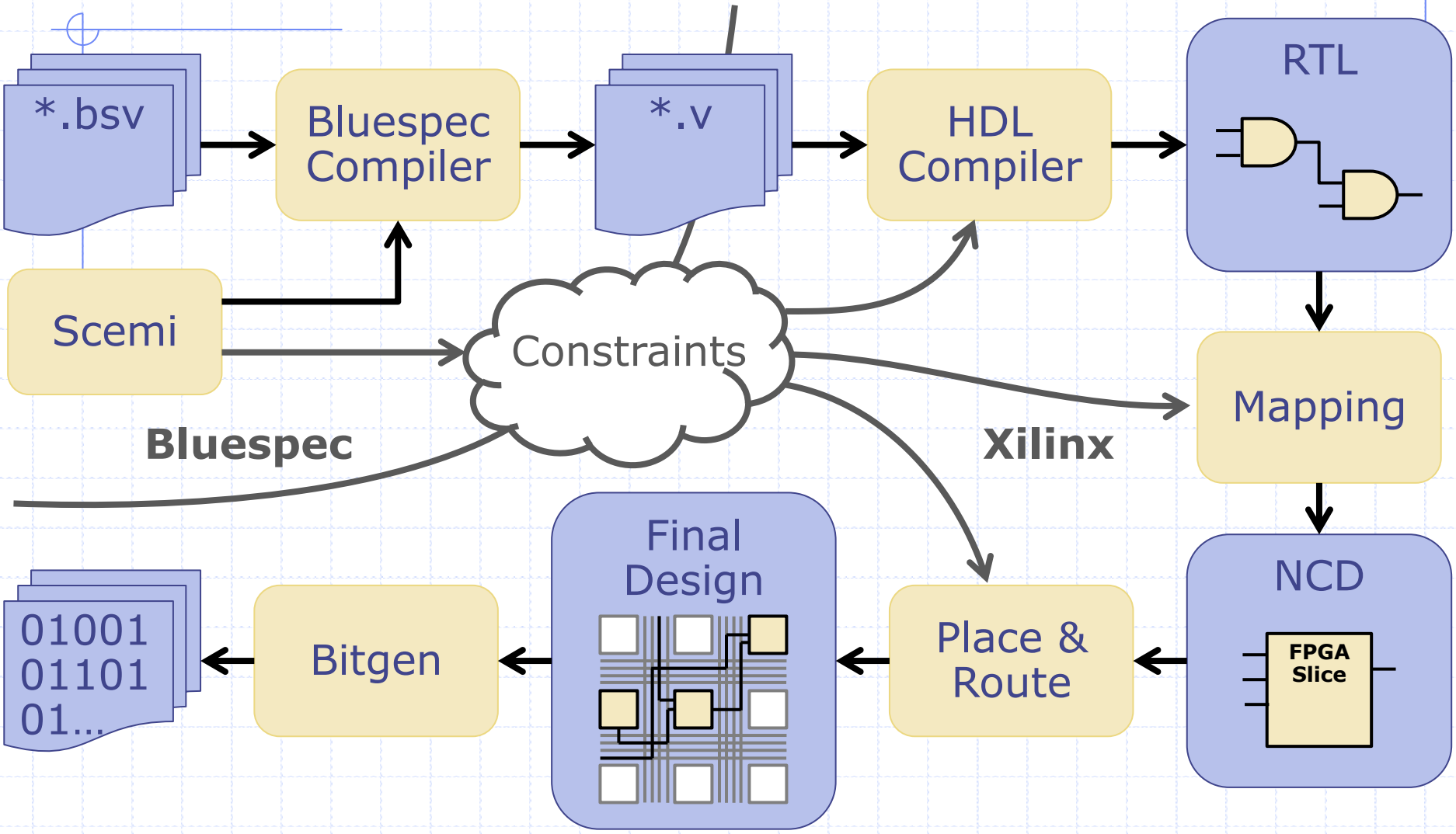
Lab Update

- FPGAs are working again
  - If you have any problems with them, let me know

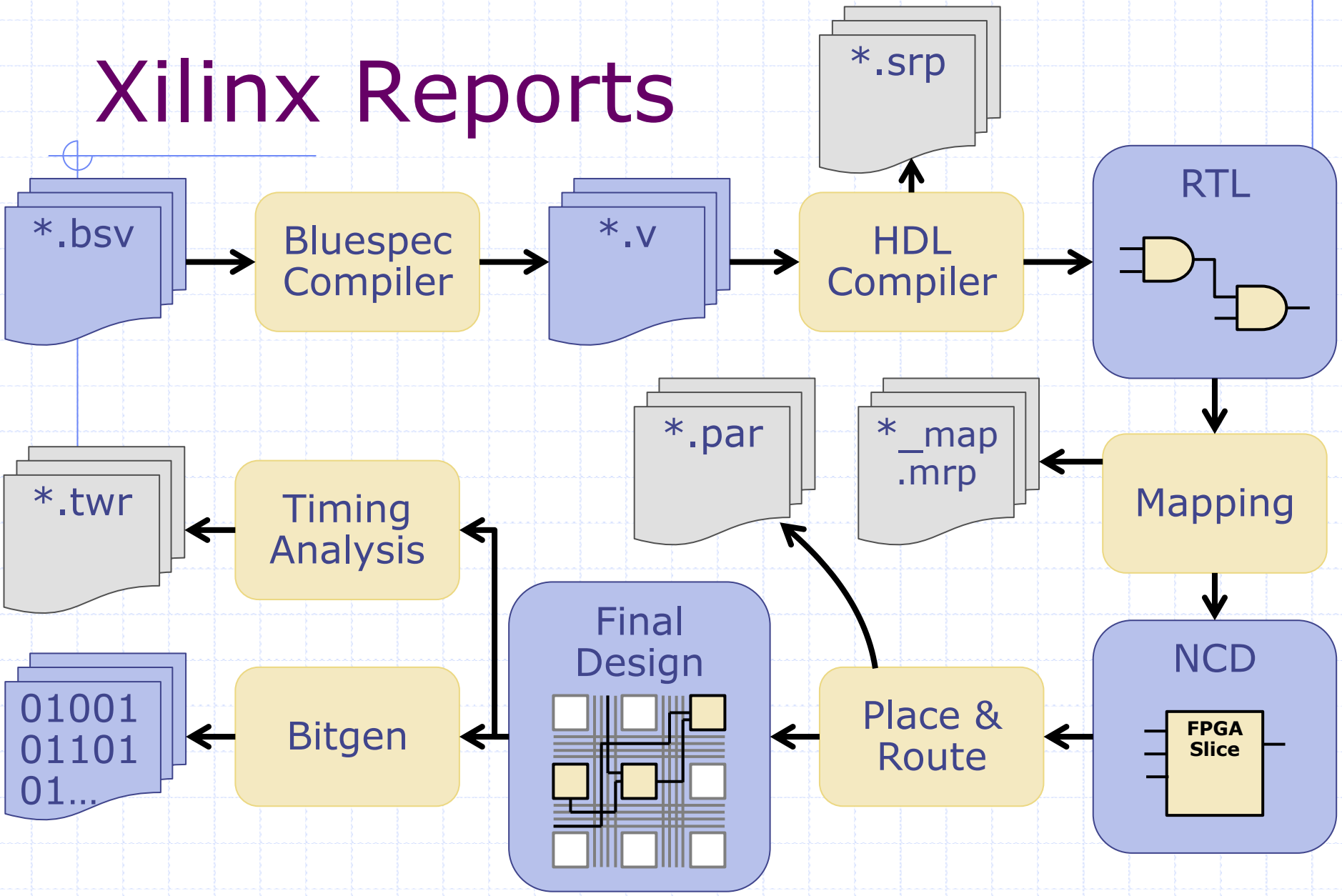
Xilinx Tool Flow



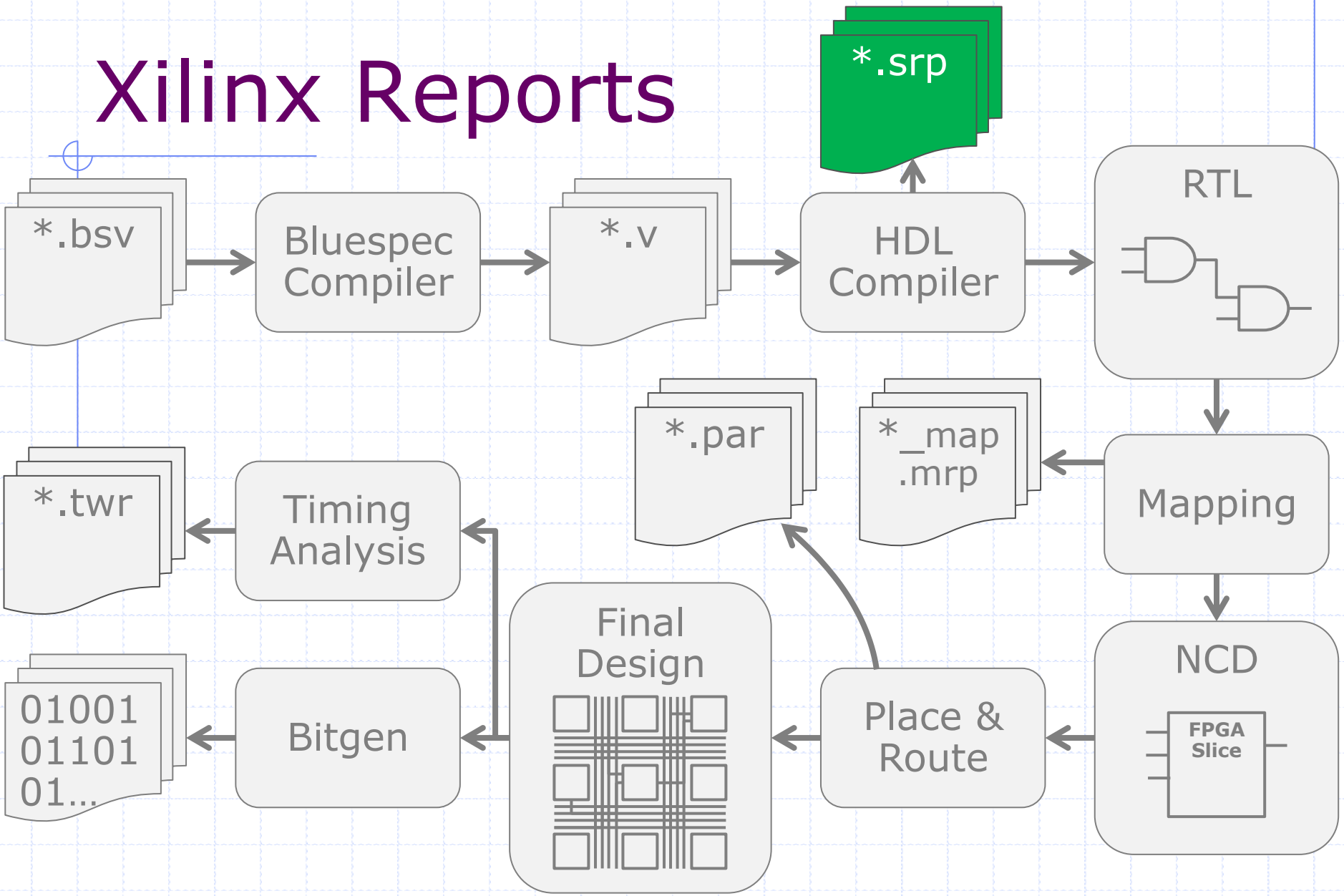
Xilinx Tool Flow



Xilinx Reports



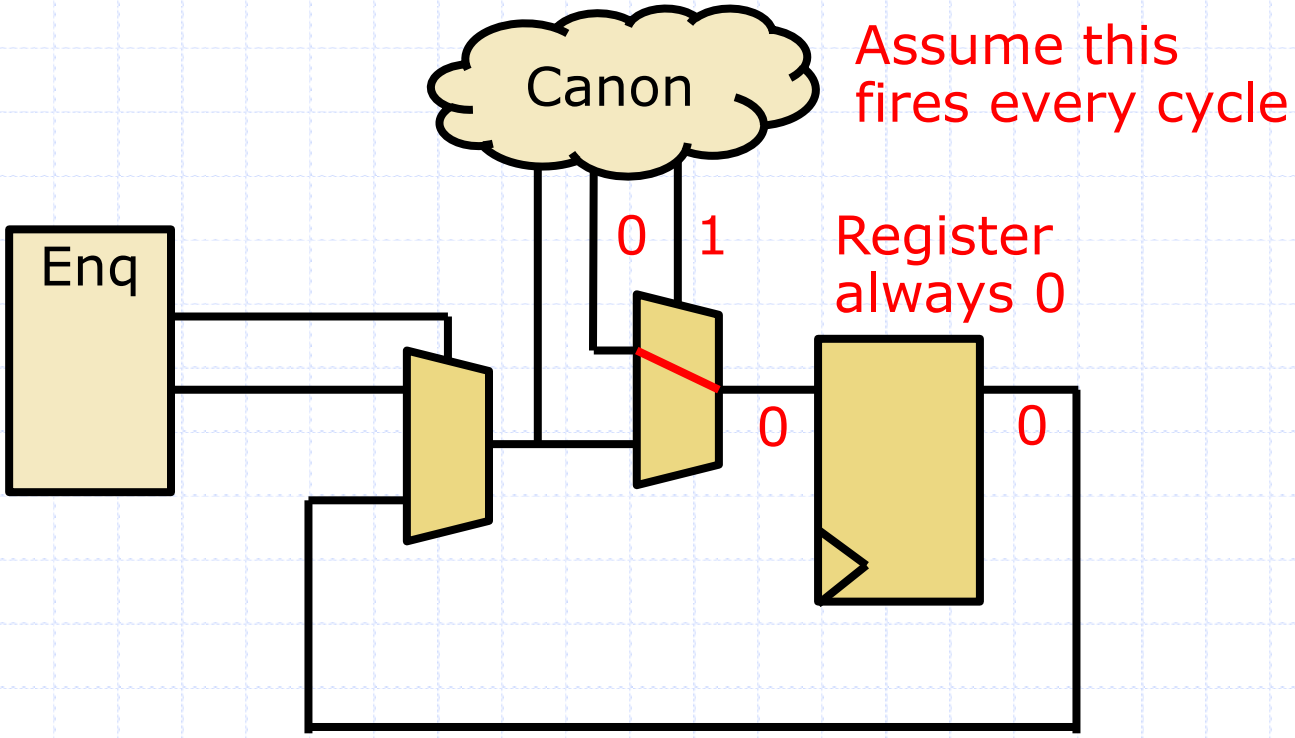
Xilinx Reports



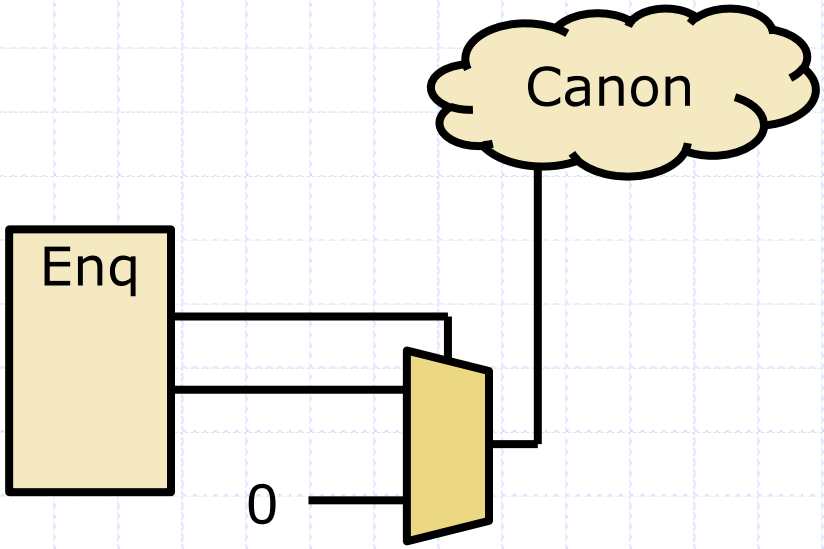
mkBridge.srp

- Search for: “Low Level Synthesis”
  - You’ll see some optimizations performed such as removing constant value registers.
  - This portion removes unwanted overhead of EHRs

Low Level Optimizations

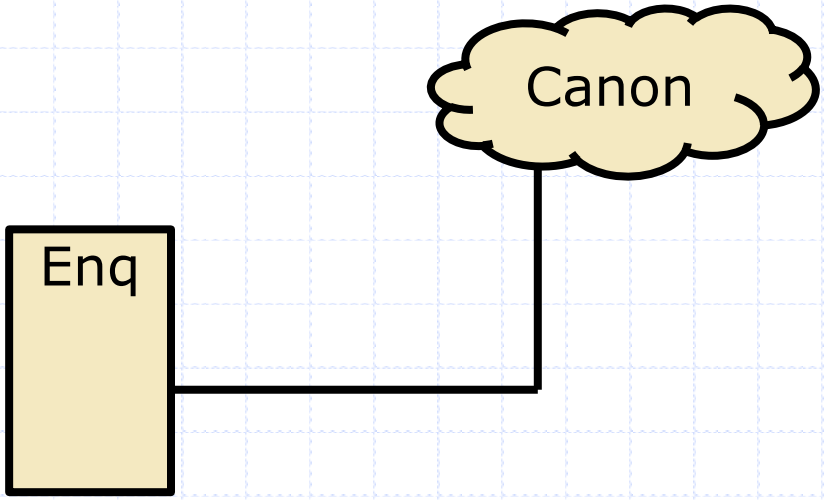


Low Level Optimizations



This can still be optimized

Low Level Optimizations



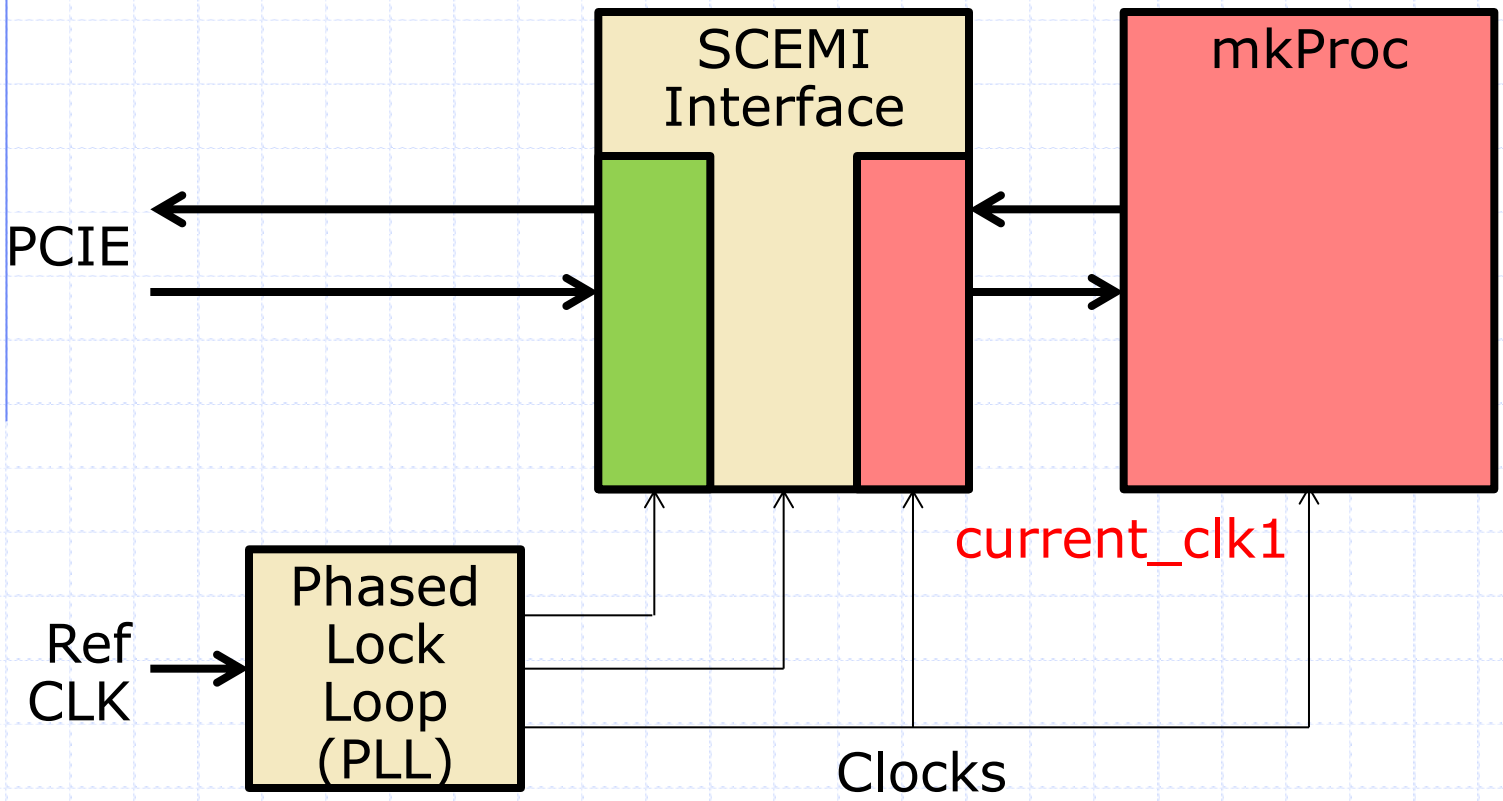
No overhead from using an EHR

mkBridge.srp

- Search for: “current_clk1”
  - This will show up in a few places, but the interesting one is in a line starting with “Timing constraint”
  - You’ll find the max clock period and the critical path for the clock.
  - You will also find information about other clocks.

Why is there more than 1 clock?

Different Clocks

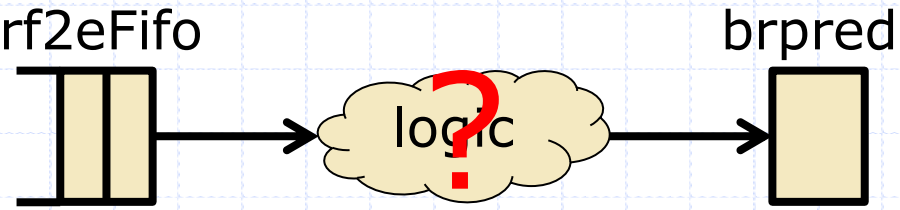


Critical Path

- Critical path example:

m is mkProc

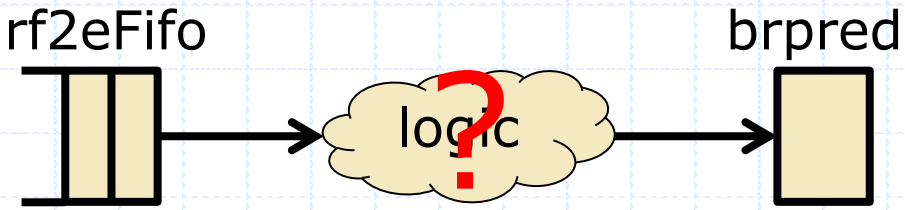
```
==========================================================================
Timing constraint: Default period analysis for Clock
'scemi_clk_port_clkgen/current_clk1'
  Clock period: 9.874ns (frequency: 101.277MHz)
  Total number of paths / destination ports: 114672315 / 13117
--------------------------------------------------------------------------
Delay:               9.874ns (Levels of Logic = 17)
  Source:            scemi_dut_dut_dutIfc_m_dut/m/rf2eFifo_data_1_96 (FF)
  Destination:       scemi_dut_dut_dutIfc_m_dut/m/brpred/arr/Mram_arr1 (RAM)
  Source Clock:      scemi_clk_port_clkgen/current_clk1 rising
  Destination Clock: scemi_clk_port_clkgen/current_clk1 rising

  Data Path: scemi_dut_dut_dutIfc_m_dut/m/rf2eFifo_data_1_96 to
scemi_dut_dut_dutIfc_m_dut/m/brpred/arr/Mram_arr1

...
```



Critical Path

```
                    Gate     Net
Cell:in->out       fanout   Delay   Delay  Logical Name (Net Name)
----------------------------------------  -----------
 FDE:C->Q              1    0.471          a_1_96)
 LUT3:I0->O           55    0.094          702<31>)
 begin scope: 'instance_exec_1'
 begin scope: 'instance_aluBr_0'
 INV:I->O              2    0.238   0.581  Mmux_aluBr_not00011_INV_0 (...)
 LUT2:I0->O            1    0.094   0.000  Mcompar_aluBr_a_SLE_0___d6_lut<6> (...)
 MUXCY:S->O            1    0.600   0.576  Mcompar_aluBr_a_SLE_0___d6_cy<6> (...)
 LUT6:I4->O           28    0.094   0.607  Mmux_aluBr61 (aluBr)
 end scope: 'instance_aluBr_0'
 begin scope: 'instance_brAddrCalc_2'
 LUT5:I4->O            6    0.094   0.737  brAddrCalc<0>11 (N01)
 LUT5:I2->O            2    0.094   0.715  brAddrCalc<27> (brAddrCalc<27>)
 end scope: 'instance_brAddrCalc_2'
...
```

Branch Target Calculation

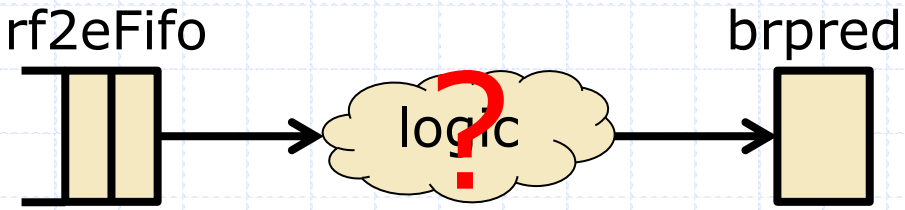


Critical Path

```
                    Gate     Net
Cell:in->out       fanout   Delay   Delay  Logical Name (Net Name)
----------------------------------------  ------------
 LUT6:I3->O             1   0.094   0.000  Mcompar_IF_..._d32_cmp_ne0000_lut<9> (...)
 MUXCY:S->O             1   0.372   0.000  Mco                    0000_cy<9> (...)
 MUXCY:CI->O            3   0.254   0.491  Mco                    0000_cy<10> (...)
 end scope: 'instance_exec_1'
 LUT6:I5->O           196   0.094   0.638  redirectFifo_data_0_lat_0_whas11 (...)
 LUT6:I5->O            65   0.094   0.613  CASE_y5239_0_IF_redirectFifo_data_... (...)
 begin scope: 'brpred'
 LUT2:I1->O            56   0.094   0.468  arr_WE1 (tagArr_WE)
 begin scope: 'arr'
 RAM64M:WE                  0.490          Mram_arr1
----------------------------------------
Total                       9.874ns (3.271ns logic, 6.603ns route)
                                    (33.1% logic, 66.9% route)
```

Redirect Fifo

Branch Predictor

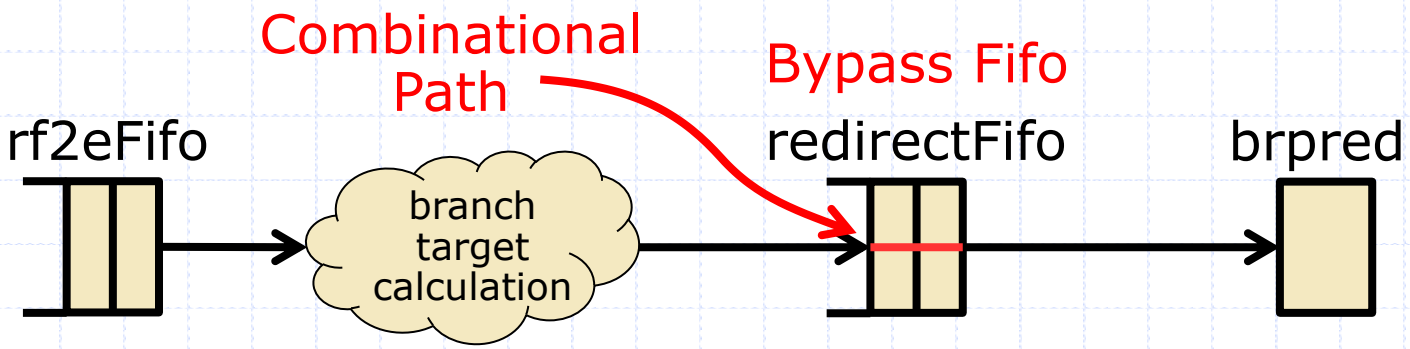


Critical Path

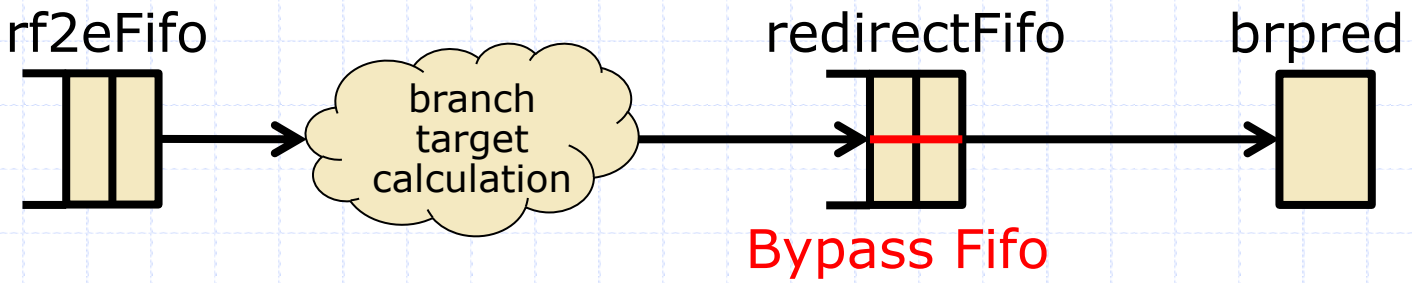
```
                        Gate     Net
Cell:in->out           fanout   Delay   Delay  Logical Name (Net Name)
----------------------------------------  ------------
 LUT6:I3->O                1   0.094   0.000  Mcompar_IF_..._d32_cmp_ne0000_lut<9> (...)
 MUXCY:S->O                1   0.372   0.000  Mco                    0000_cy<9> (...)
 MUXCY:CI->O               3   0.254   0.491  Mco                    0000_cy<10> (...)
 end scope: 'instance_exec_1'
 LUT6:I5->O              196   0.094   0.638  redirectFifo_data_0_lat_0_whas11 (...)
 LUT6:I5->O               65   0.094   0.613  CASE_y5239_0_IF_redirectFifo_data_... (...)
 begin scope: 'brpred'
 LUT2:I1->O               56   0.094   0.468  arr_WE1 (tagArr_WE)
 begin scope: 'arr'
 RAM64M:WE                     0.490          Mram_arr1
----------------------------------------
Total                          9.874ns (3.271ns logic, 6.603ns route)
                                       (33.1% logic, 66.9% route)
```

Redirect Fifo

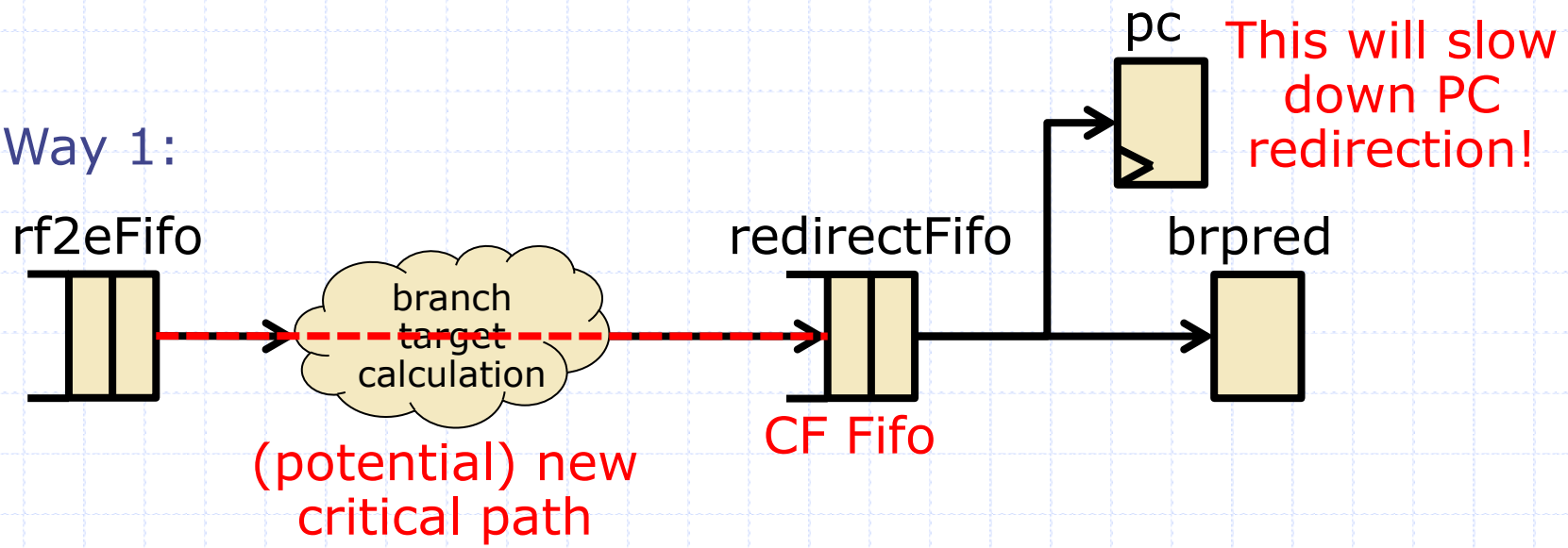
Branch Predictor



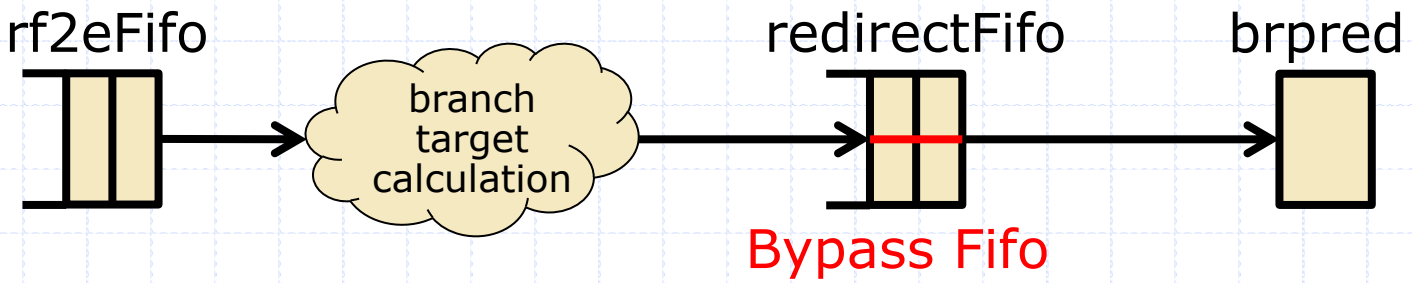
Splitting Critical Paths



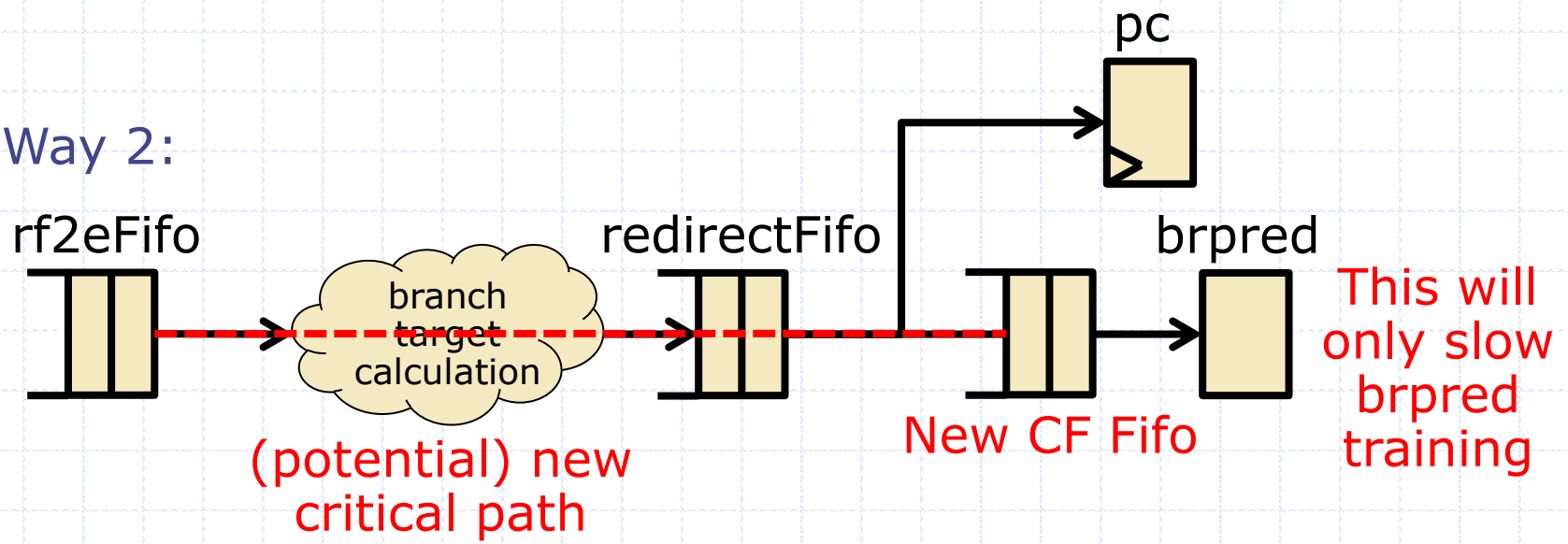
Way 1:



Splitting Critical Paths



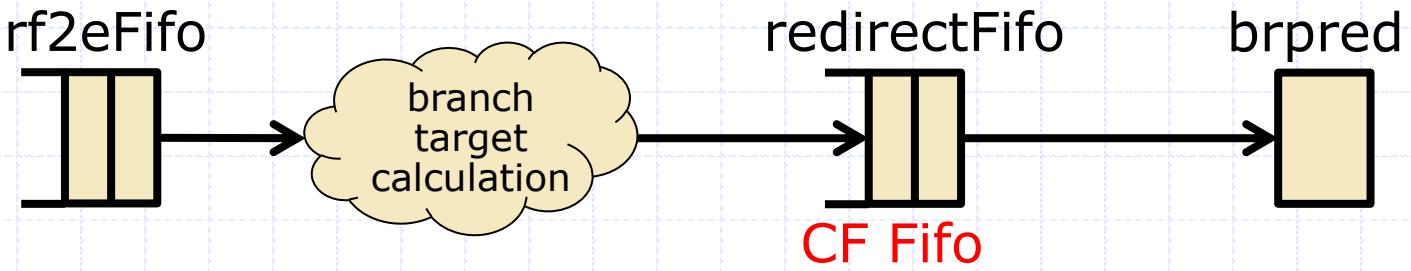
Way 2:



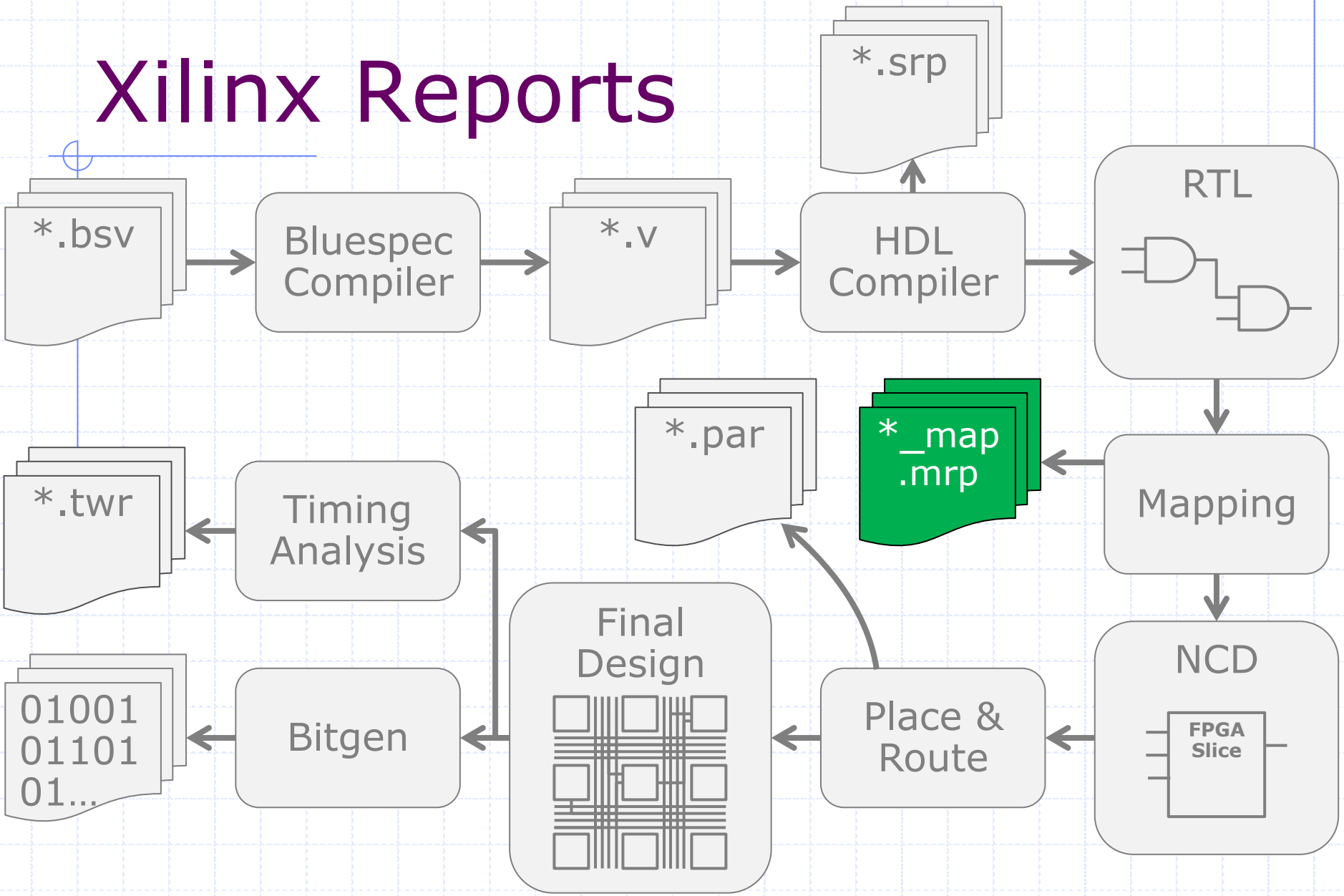
Critical Path

- After splitting the critical path with way 1:

```
================================================================================
Timing constraint: Default period analysis for Clock
'scemi_clk_port_clkgen/current_clk1'
  Clock period: 8.749ns (frequency: 114.294MHz)
  Total number of paths / destination ports: 38299383 / 13320
--------------------------------------------------------------------------------
Delay:               8.749ns (Levels of Logic = 19)
  Source:            scemi_dut_dut_dutIfc_m_dut/m/rf2eFifo_data_1_96 (FF)
  Destination:       scemi_dut_dut_dutIfc_m_dut/m/rf2eFifo_enqEn_rl (FF)
  Source Clock:      scemi_clk_port_clkgen/current_clk1 rising
  Destination Clock: scemi_clk_port_clkgen/current_clk1 rising

  Data Path: scemi_dut_dut_dutIfc_m_dut/m/rf2eFifo_data_1_96 to
scemi_dut_dut_dutIfc_m_dut/m/rf2eFifo+enqEn+rl
```



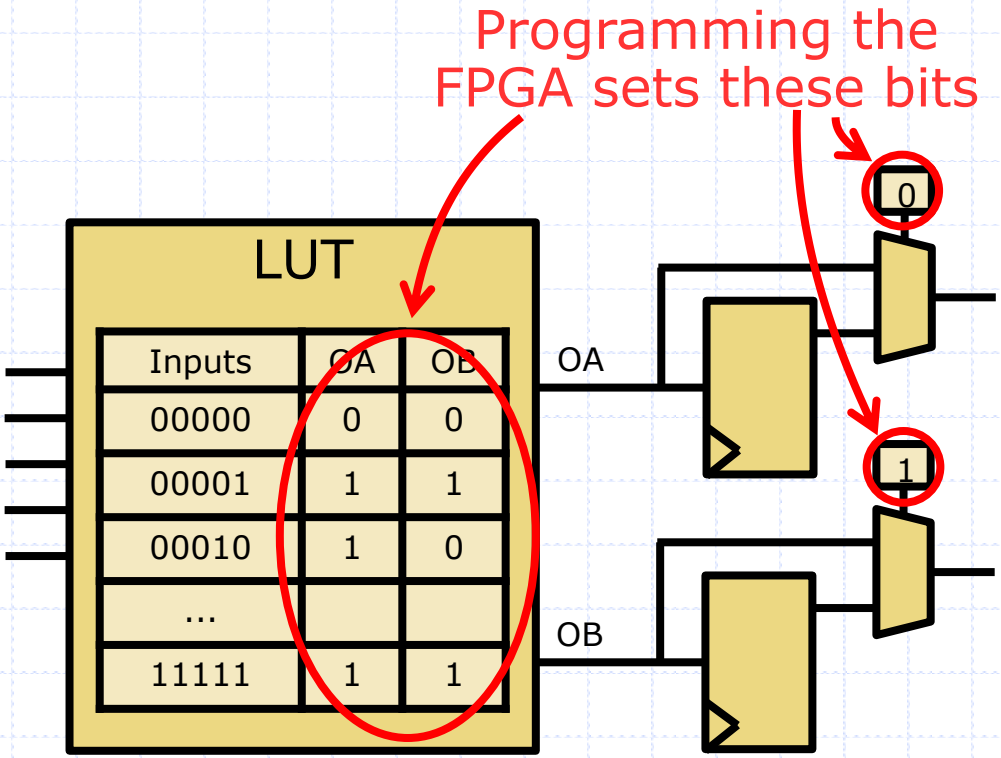
Xilinx Reports



mkBridge_map.mrp

- Search for: “Design Summary”
  - You’ll see how much of the FPGAs resources are being used by your designs.
  - This information reveals how big your design is and how much routing congestion to expect.

LUT-FF Pair



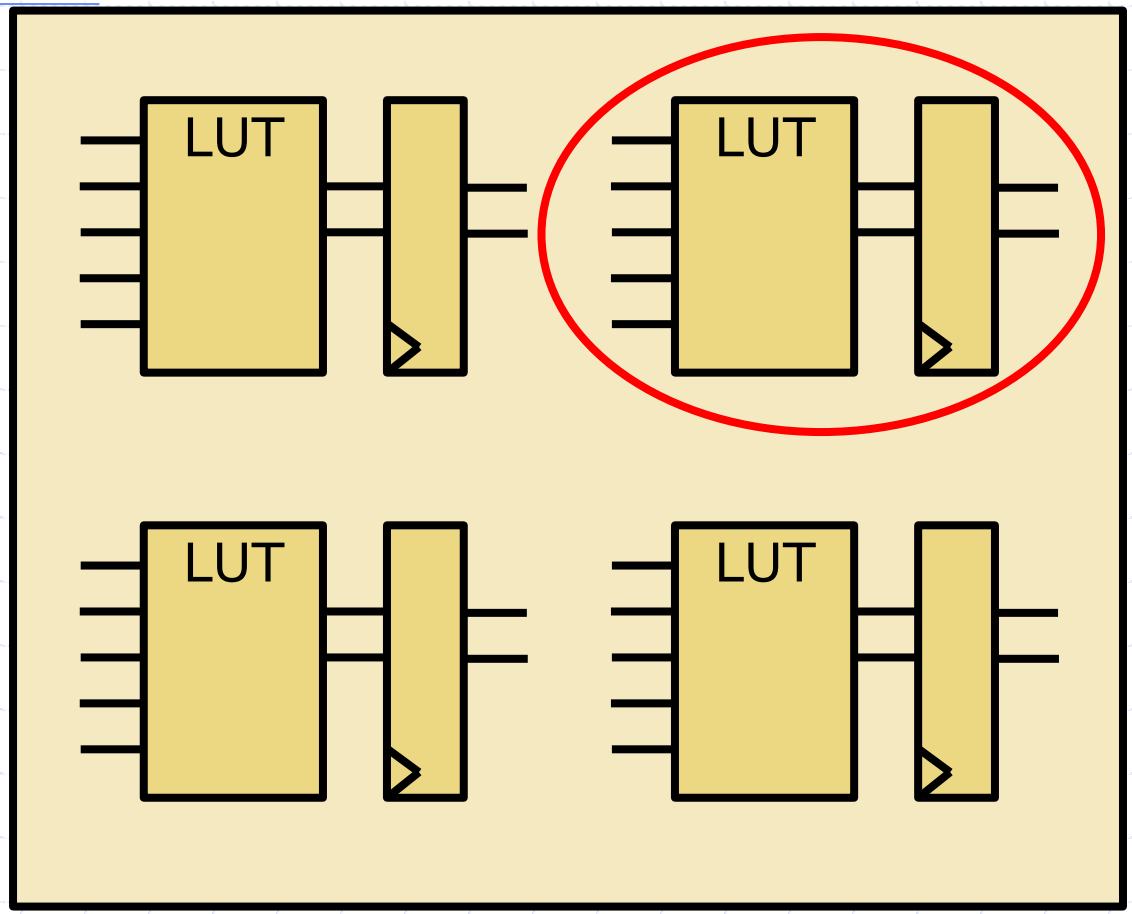
This is a simplified version of LUT-FF Pairs

Design Summary

Slice Logic Utilization:	Total number used		Total number on FPGA	
Number of Slice Registers:	11,697	out of	69,120	16%
Number used as Flip Flops:	11,693			
Number used as Latches:	1			
Number used as Latch-thrus:	3			
Number of Slice LUTs:	17,958	out of	69,120	25%
Number used as logic:	17,392	out of	69,120	25%
Number using O6 output only:	16,372			
Number using O5 output only:	613			
Number using O5 and O6:	407			
Number used as Memory:	520	out of	17,920	2%
Number used as Dual Port RAM:	376			
Number using O6 output only:	136			
Number using O5 output only:	3			
Number using O5 and O6:	237			
Number used as Shift Register:	144			
Number using O6 output only:	144			
Number used as exclusive route-thru:	46			
Number of route-thrus:	715			
Number using O6 output only:	653			
Number using O5 output only:	57			
Number using O5 and O6:	5			

Using about the quarter of the chip's resources

FPGA Slice



Design Summary

Slice Logic Distribution:				
Number of occupied Slices:	7,385	out of	17,280	42%
Number of LUT Flip Flop pairs used:	21,432			
Number with an unused Flip Flop:	9,735	out of	21,432	45%
Number with an unused LUT:	3,474	out of	21,432	16%
Number of fully used LUT-FF pairs:	8,223	out of	21,432	38%
Number of unique control sets:	881			
Number of slice register sites lost to control set restrictions:	1,953	out of	69,120	2%
IO Utilization:				
Number of bonded IOBs:	11	out of	640	1%
Number of LOCed IOBs:	11	out of	11	100%
Number of bonded IPADs:	4			
Number of LOCed IPADs:	2	out of	4	50%
Number of bonded OPADs:	2			

Using about half of the chip's area

Design Summary

Specific Feature Utilization:

Number of BlockRAM/FIFO:	140	out of	148	94%
Number using BlockRAM only:	140			
Total primitives used:				
Number of 36k BlockRAM used:	140			
Total Memory used (KB):	5,040	out of	5,328	94%
Number of BUFG/BUFGCTRLs:	8	out of	32	25%
Number used as BUFGs:	8			
Number of BUFDSs:	1	out of	8	12%
Number of LOCed BUFDSs:	1	out of	1	100%
Number of GTP_DUALs:	1	out of	8	12%
Number of LOCed GTP_DUALs:	1	out of	1	100%
Number of PCIEs:	1	out of	1	100%
Number of LOCed PCIEs:	1	out of	1	100%
Number of PLL_ADVs:	2	out of	6	33%

Using almost all of the chip's BRAM

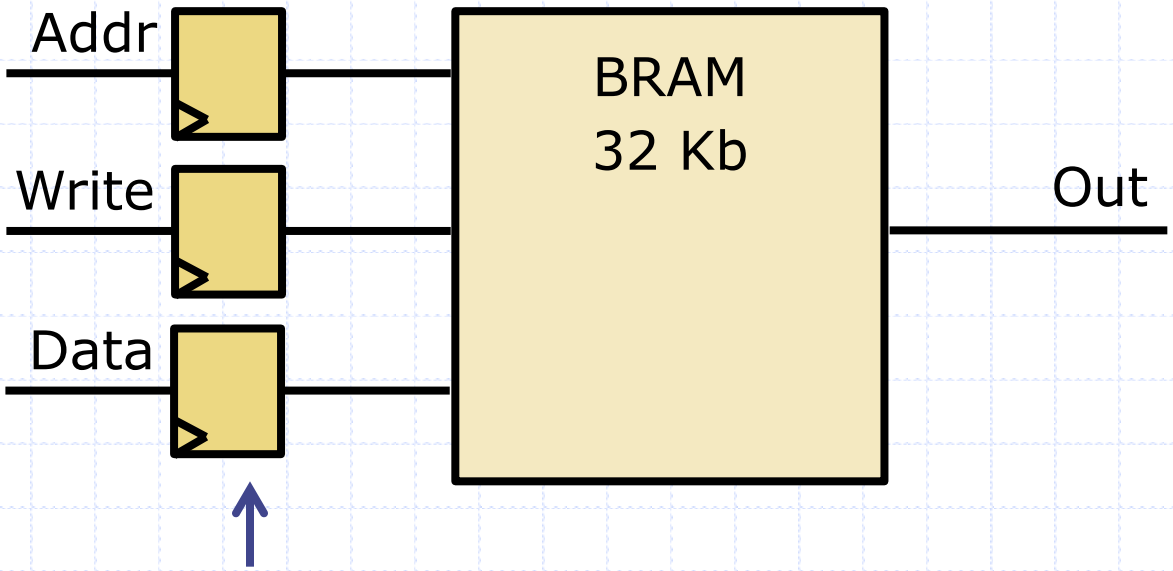
This could be a problem

Block Ram

- Dedicated memory slices on FPGA
- Can contain 32Kb of data per BRAM
- Evenly distributed across FPGA fabric

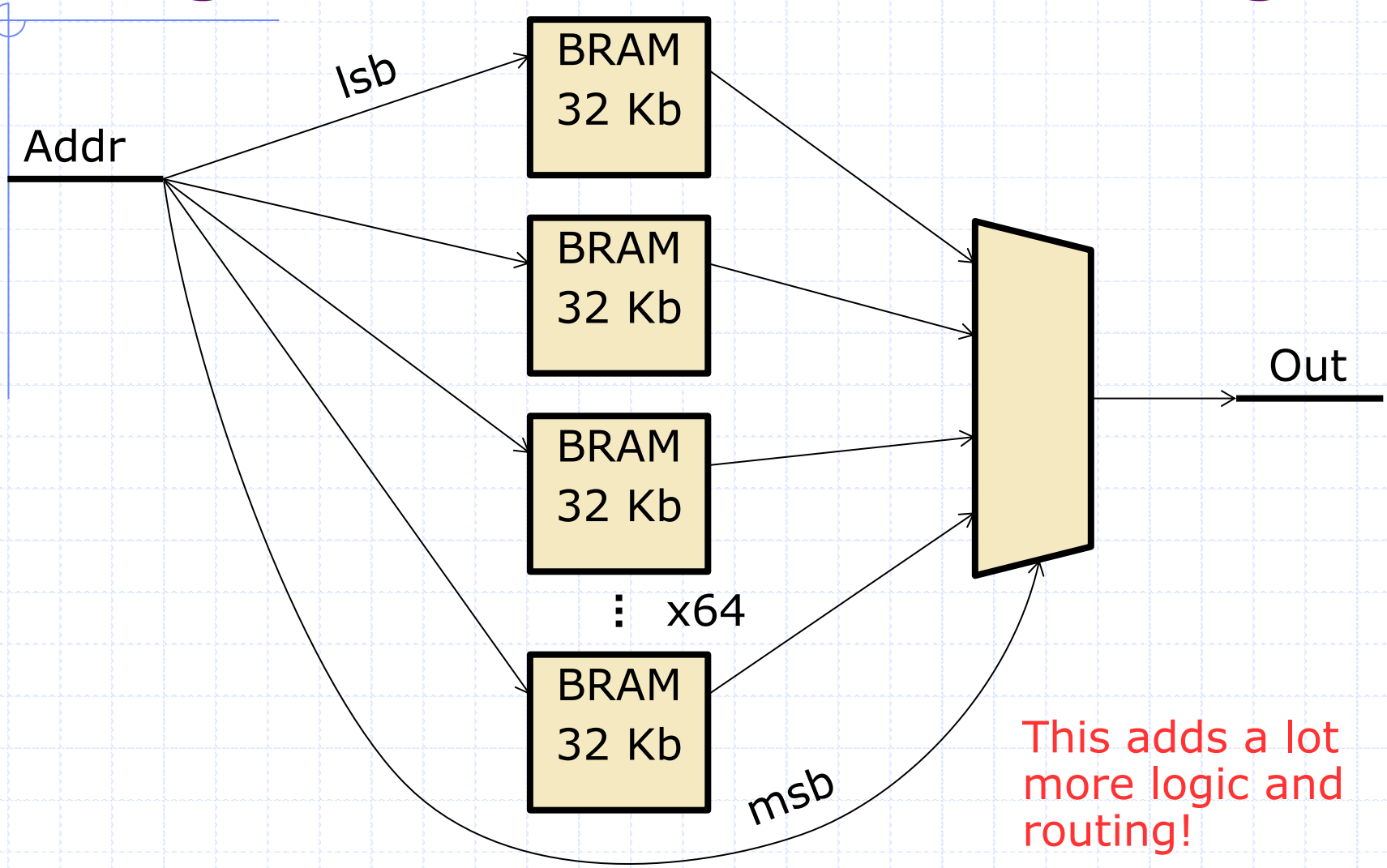
We have 2048 Kb of instruction and data memory.
How does this fit on 32 Kb blocks?

Single Block Ram

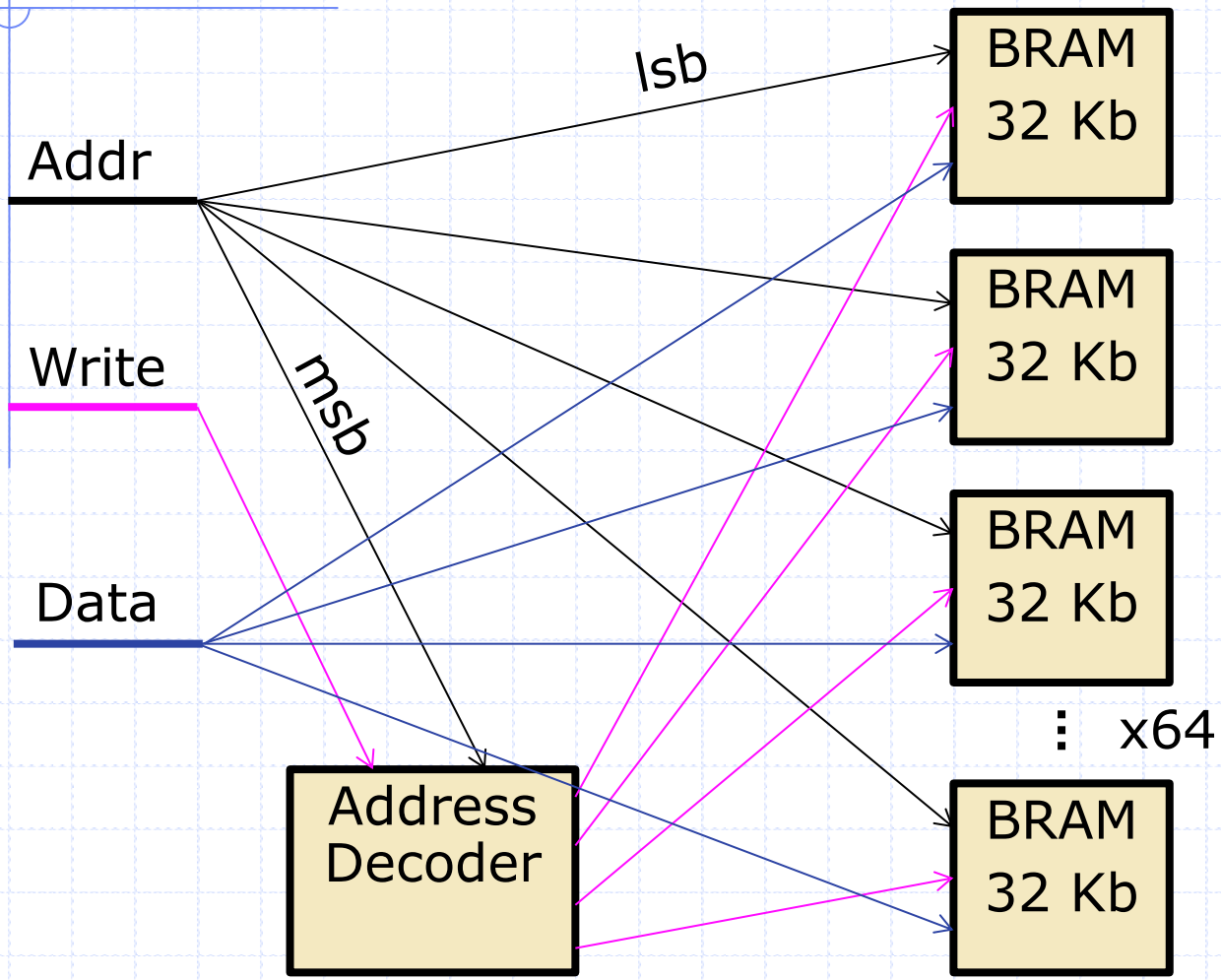


Registers on input so more combinational delay for output than input

Large Block Ram: Reading

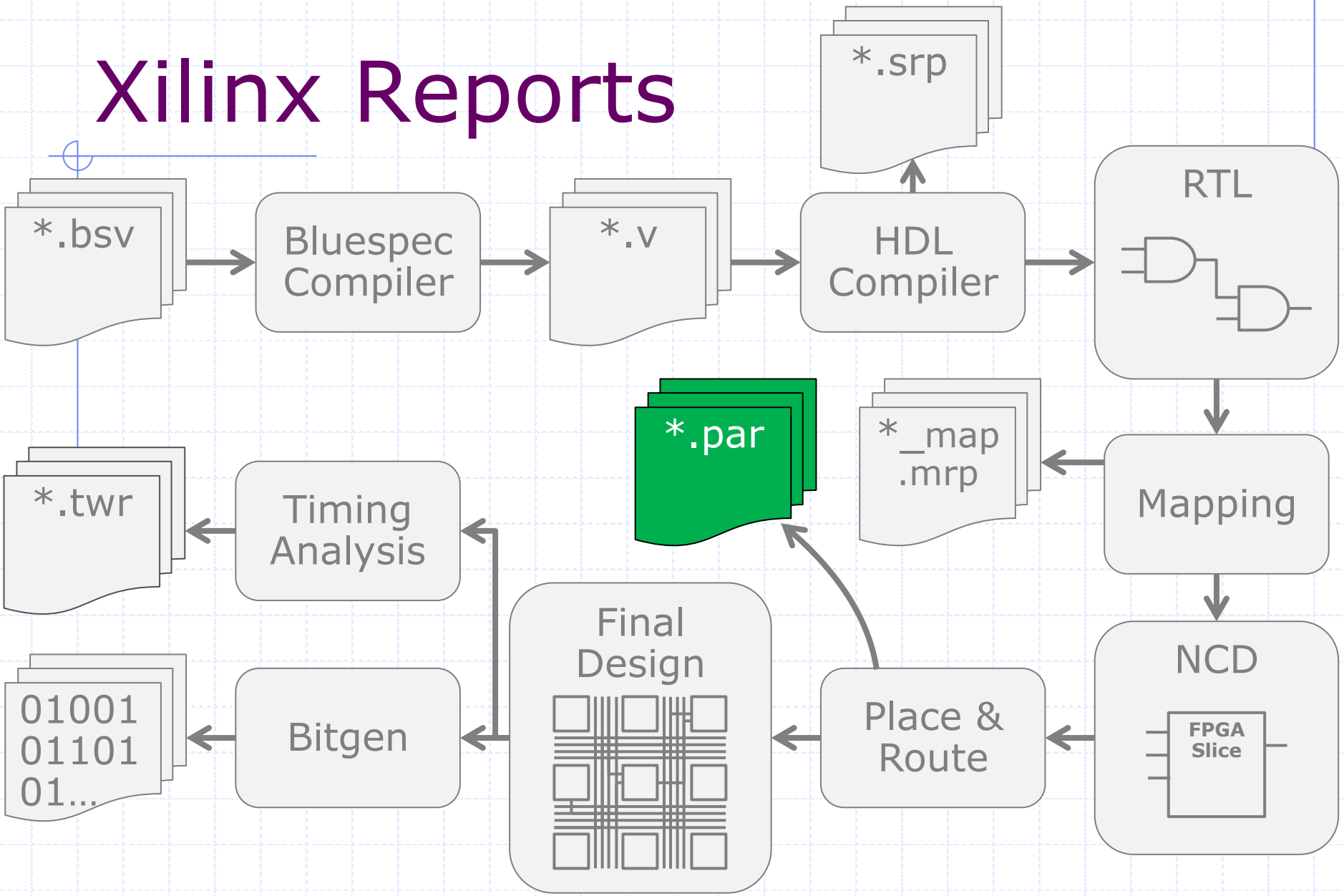


Large Block Ram: Writing



This also adds a lot more logic and routing!

Xilinx Reports



mkBridge.par

- Search for: “Deive Utilization Summary”
  - You’ll see a more accurate report of resource utilization

mkBridge_map.mrp

- Search for: “Generating Clock Report”
  - You’ll see some information about clock timing constraints
  - All of these constraints relate to internal SceMi clocks

Clock Report

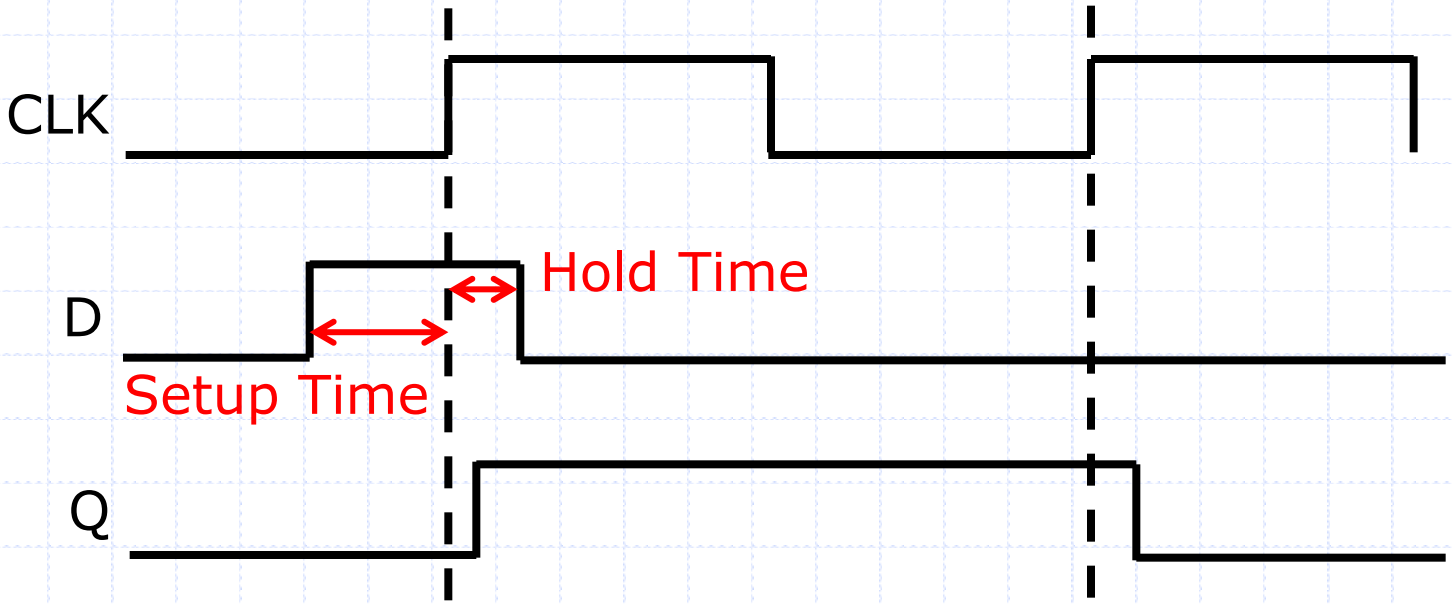
This report shows internal SceMi timing errors

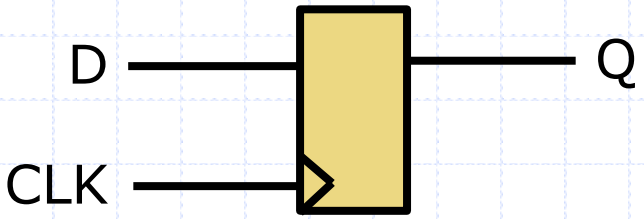
Constraint	Check	Worst Case Slack	Best Case Achievable	Timing Errors	Timing Score
* TS_scemi_pcie_ep_pcie_ep0_pcie_blk_clocking_i_clkout1_1 = PERIOD TIMEGRP " scemi_pcie_ep_pcie_ep0_pcie_blk_clocking_i_clkout1_1" TS_MGTCLK * 0.625 HIGH 50%	SETUP	-0.047ns	16.188ns	1	47
	HOLD	0.031ns		0	0
TS_scemi_pcie_ep_pcie_ep0_pcie_blk_clocking_i_clkout0_1 = PERIOD TIMEGRP " scemi_pcie_ep_pcie_ep0_pcie_blk_clocking_i_clkout0_1" TS_MGTCLK * 2.5 HIGH 50%	SETUP	0.045ns	3.955ns	0	0
	HOLD	0.418ns		0	0
	MINPERIOD	0.000ns	4.000ns	0	0
TS_scemi_pcie_ep_pcie_ep0_pcie_blk_clocking_i_clkout0_0 = PERIOD TIMEGRP " scemi_pcie_ep_pcie_ep0_pcie_blk_clocking_i_clkout0_0" TS_SYSCLK * 2.5 HIGH 50%	MINPERIOD	0.000ns	4.000ns	0	0

Asterisk (*)

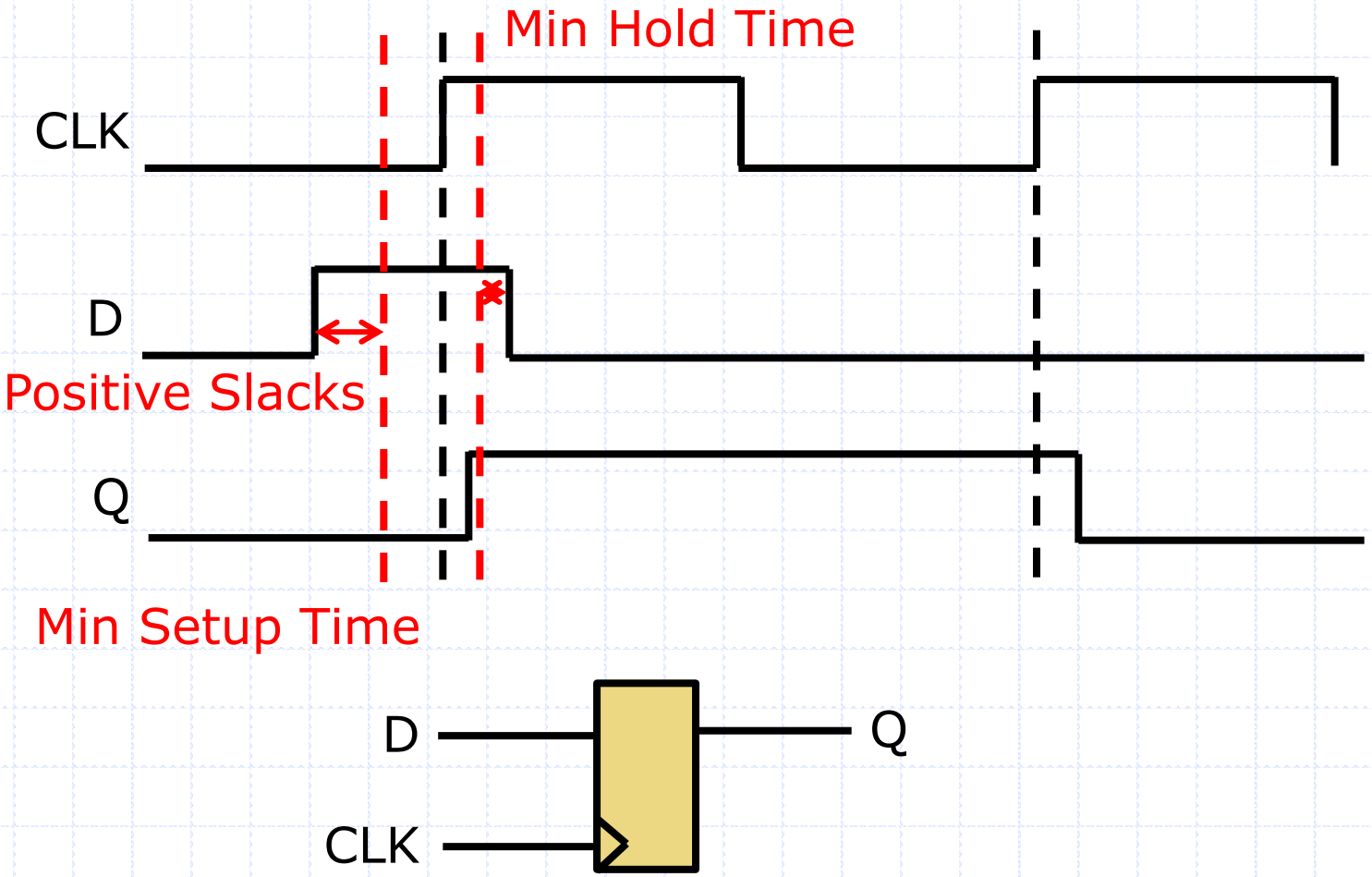
Negative slack

Setup and Hold

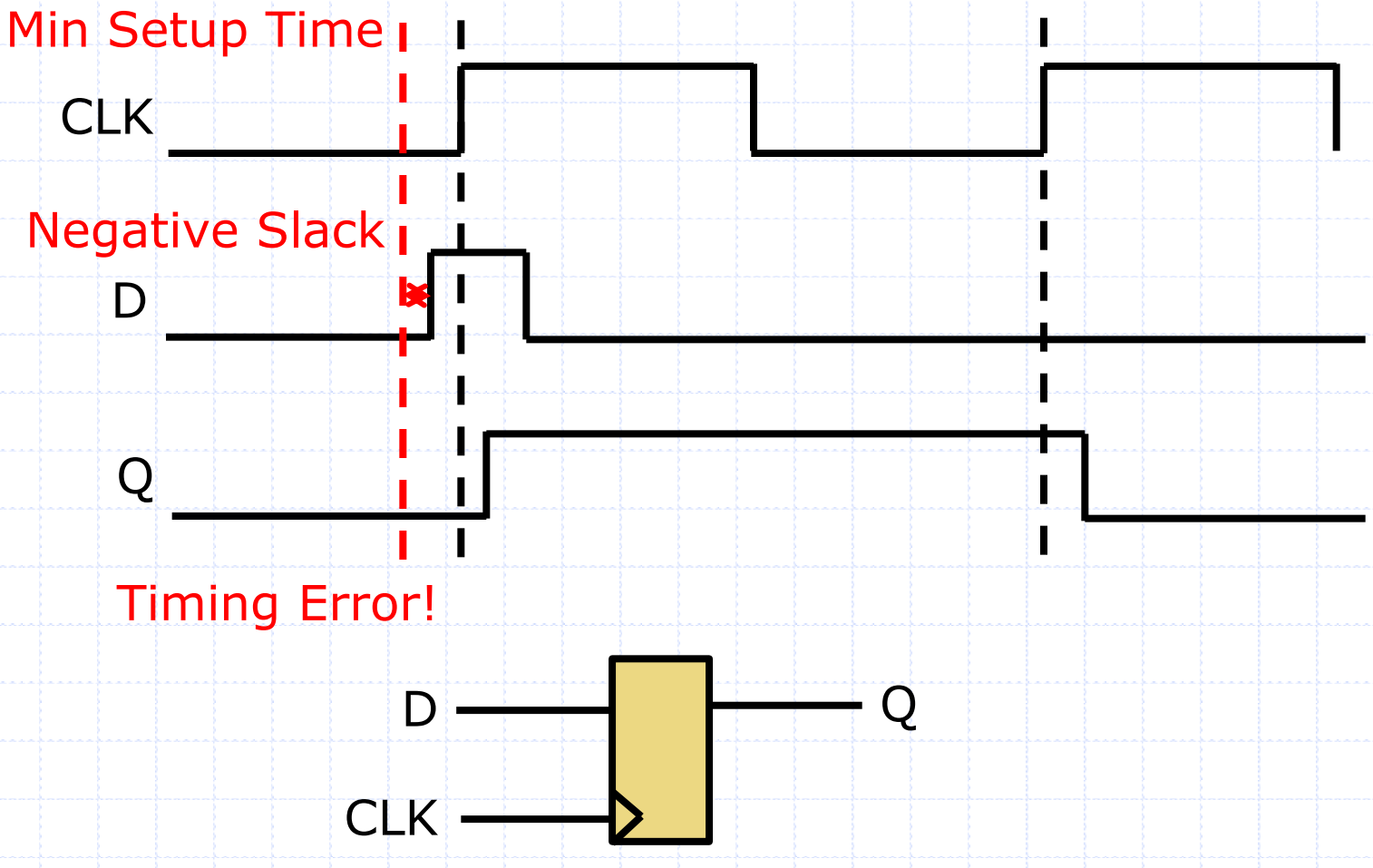




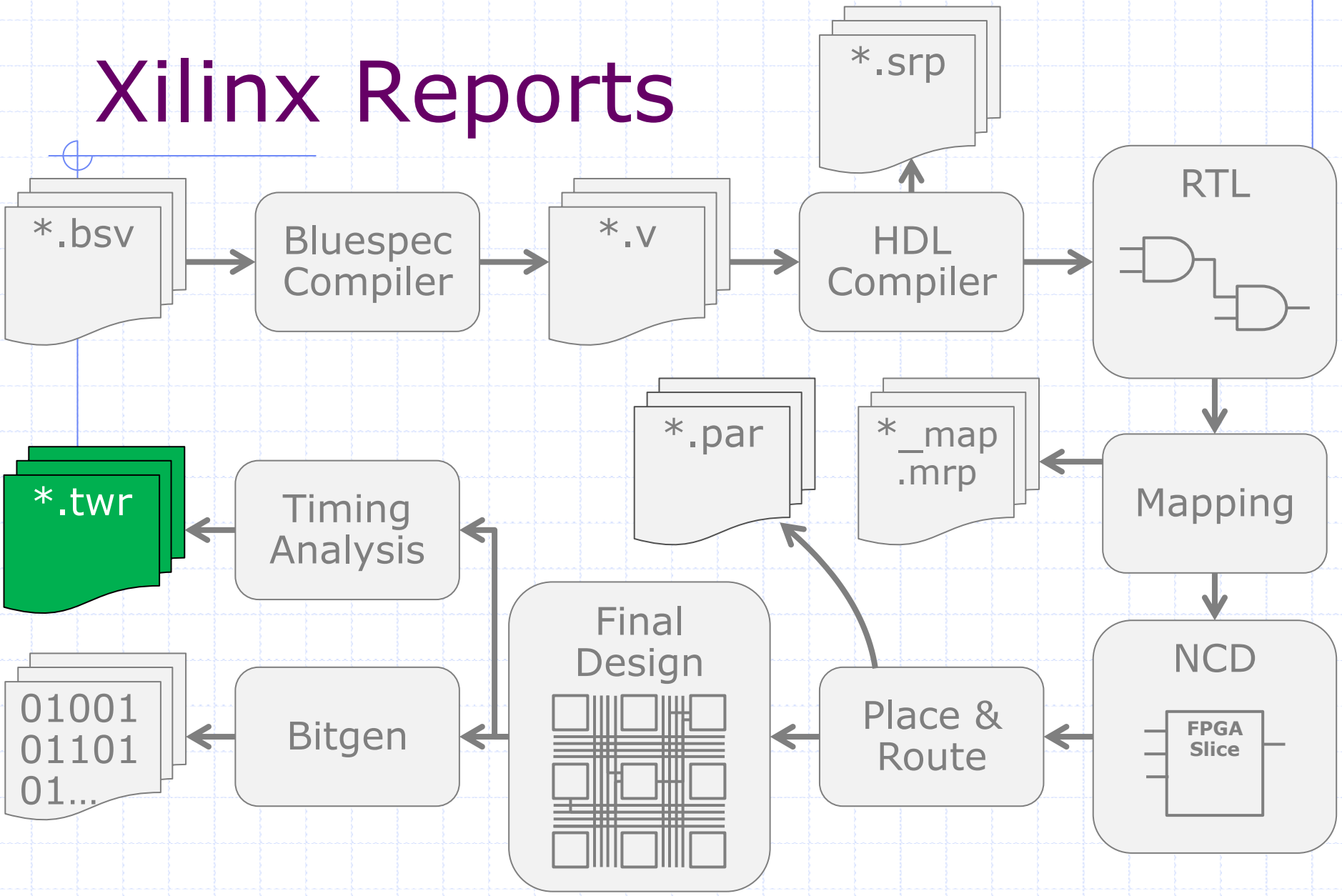
Setup and Hold



Setup and Hold



Xilinx Reports



mkBridge.twr

- More timing information
  - All about internal SceMi Clocks
  - No information about current_clk1

Conclusion

- Any Questions?

‘build’ utility

- Automates the Xilinx tool flow
- See `build --doc` for more information

‘build’ utility

- Performs the following stages in order:
  1. delete_build_dirs
  2. make_build_dirs
  3. compile_for_verilog (bsc –verilog)
  4. generate_scemi_parameters
  5. xilinx_cleanup
  6. make_xilinx_directory
  7. create_ucf_file
  8. create_xcf_file
  9. create_scr_file
  10. prepare_project_files
  11. xst_compile
  12. translate_and_build
  13. map_to_device
  14. place_and_route
  15. timing_analysis
  16. gen_bit_file
  17. timing_check
  18. gen_ace_file

‘build’ utility

- Major stages
  - compile_for_verilog
  - xst_compile
  - translate_and_build
  - map_to_device
  - place_and_route
  - timing_analysis
  - gen_bit_file
  - timing_check