FIR Workbook (lab_3)

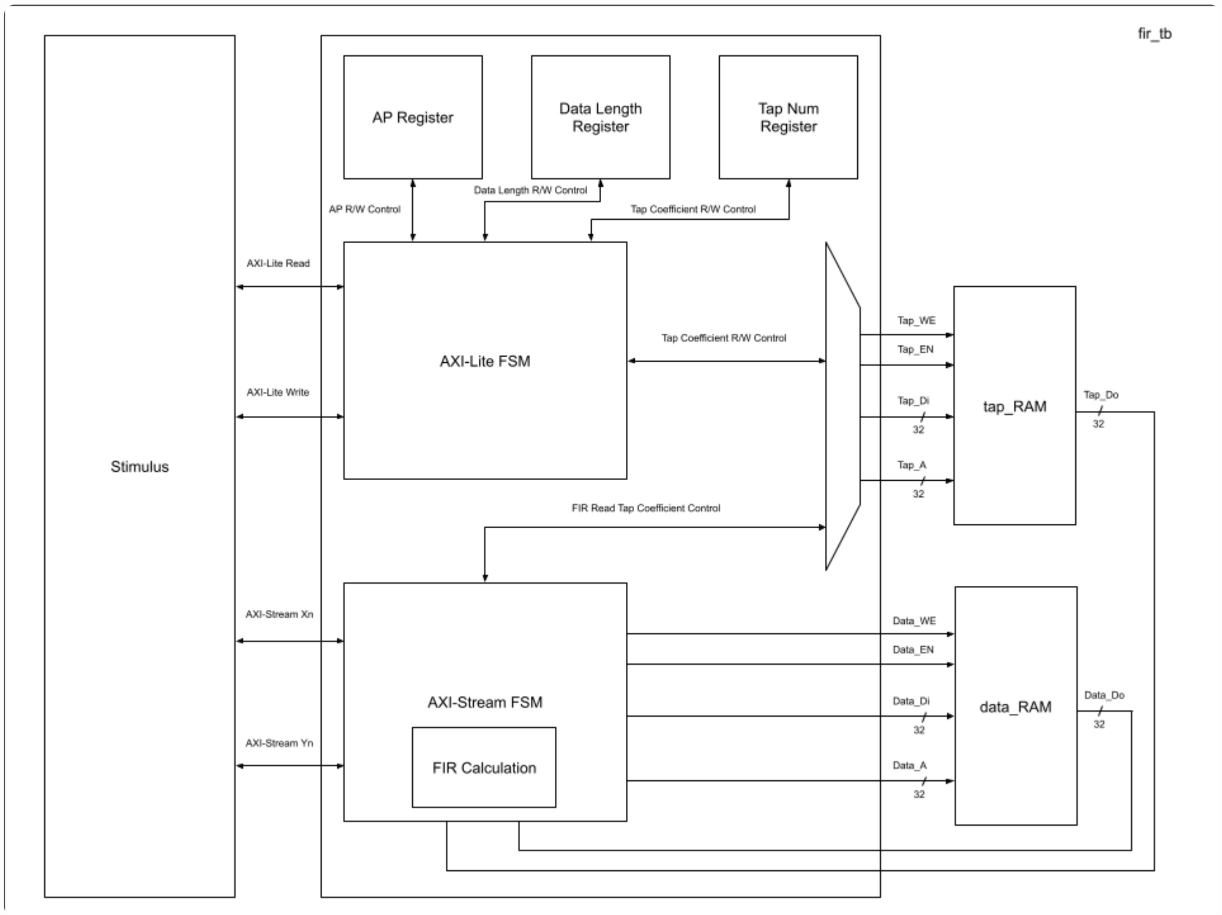
Student ID: 20726557

This lab involves designing a FIR filter which communicates through the AXI-Lite and AXI-Stream protocols. AXI-Lite controls the configuration registers and read/write of the tap coefficients, while AXI-Stream controls the dataflow of the FIR filter. The tap coefficients and data for the FIR filter are stored inside BRAM modules.

At the beginning of operation, ap_idle is set. After initial configuration of the data length, number of taps, and the tap coefficients, the FIR filter receives ap_start through AXI-Lite protocol and deasserts ap_idle.

When ap_start is asserted, FIR filter operation retrieves and outputs the computed FIR data through the AXI-Stream protocol. When the last data is retrieved and outputted, ap_done and ap_idle will be asserted to await for next FIR filter operation.

Block Diagram



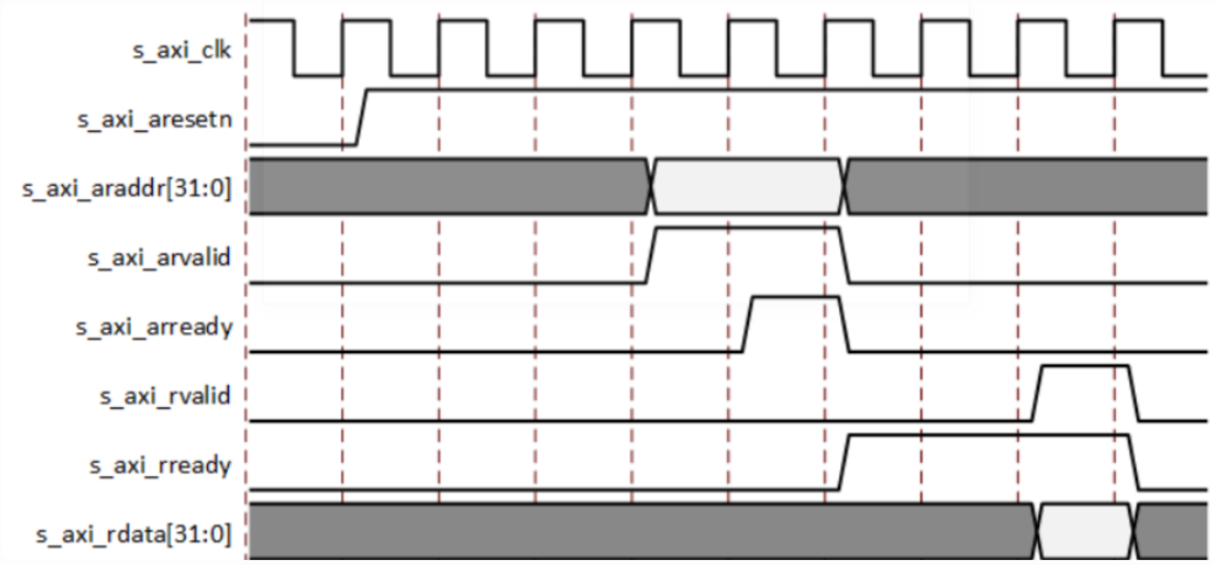
Operation

AXI Protocols

For this lab, the AXI-Lite and the AXI-Stream Protocols are used to read/write data to the configuration registers and the BRAM.

AXI-Lite Read

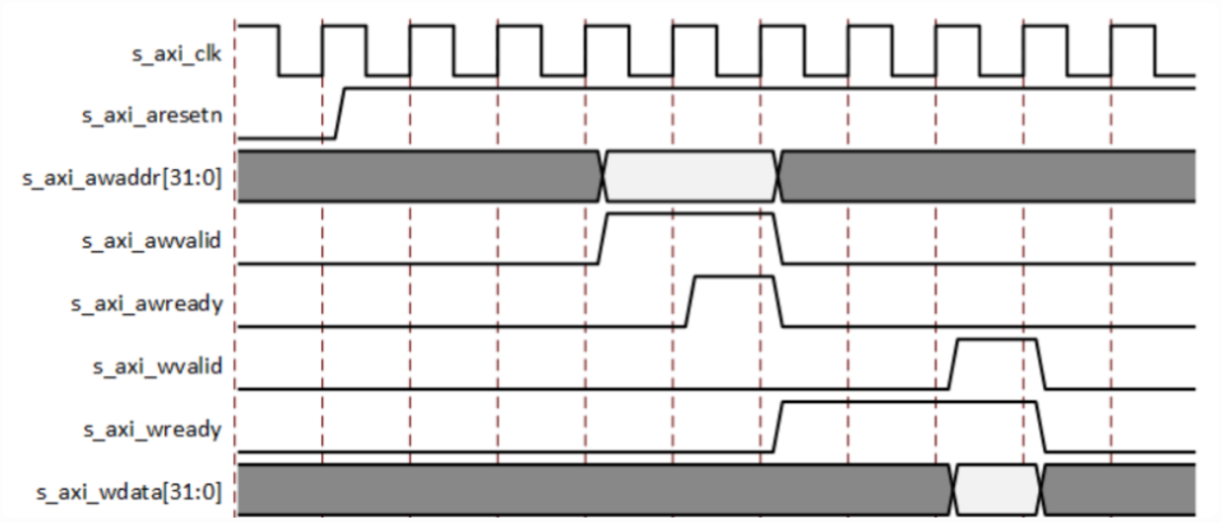
The AXI-Lite Read protocol contains two processes: address and data.



To read the data, the receiver sends an "arvalid" along with the address to be read "araddr". The transmitter samples the address and asserts "arready" to notify the receiver to invalidate "araddr". After sending a "arready" as a response to the "arvalid", the transmitter will assert "rvalid" along with the data to be read "rdata". When the receiver is ready to retrieve the data, it sets "rready". When both "rvalid" and "rready" is asserted, the receiver reads the data from the transmitter and both signals deasserts in the next cycle.

AXI-Lite Write

Similar to the AXI-Lite Read protocol, AXI-Lite Write protocol contains both the address and data processes.



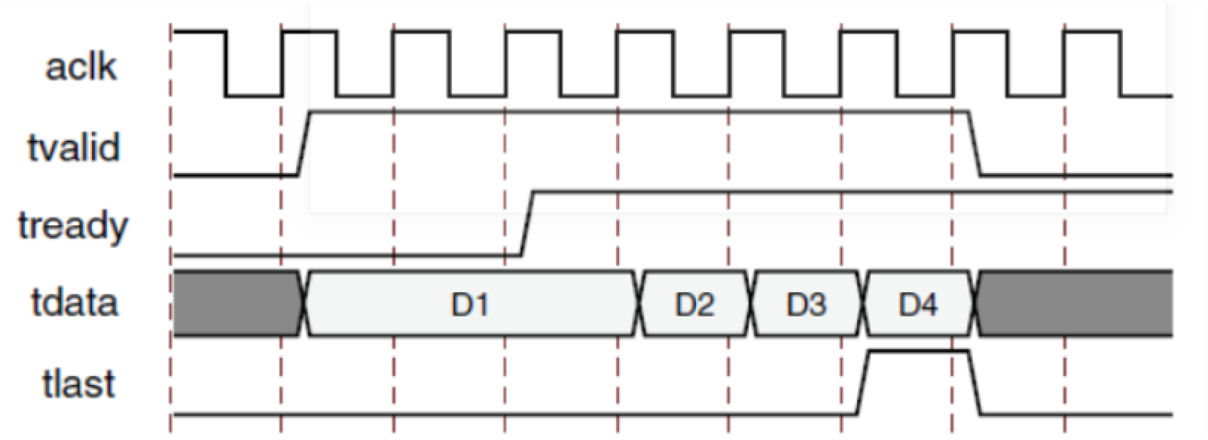
To write data, the transimitter asserts "awvalid" along with the receiver address "awaddr". When the receiver detects the "awvalid", the receiver sends a response "awready" to notify the transimitter that it has sampled the address. The transmitter will deassert "awvalid" and invalidate "awaddr" upon detecting "awready" from the receiver.

When the data is ready to be transmitted from the transmitter to the receiver, the transmitter asserts "wvalid" and sets the data to "wdata". The receiver samples the data from "wdata" and sends a "wready" to transmitter to deassert "wvalid" and invalidate "wdata". When the receiver have both the address and data, the data is written to the given address.

There is no dependency between transimitter asserting "awvalid" and "wvalid". The transmitter can send both send both at the same time, or one before the other.

AXI-Stream

AXI-Stream is used to stream-in and stream-out a series of data.



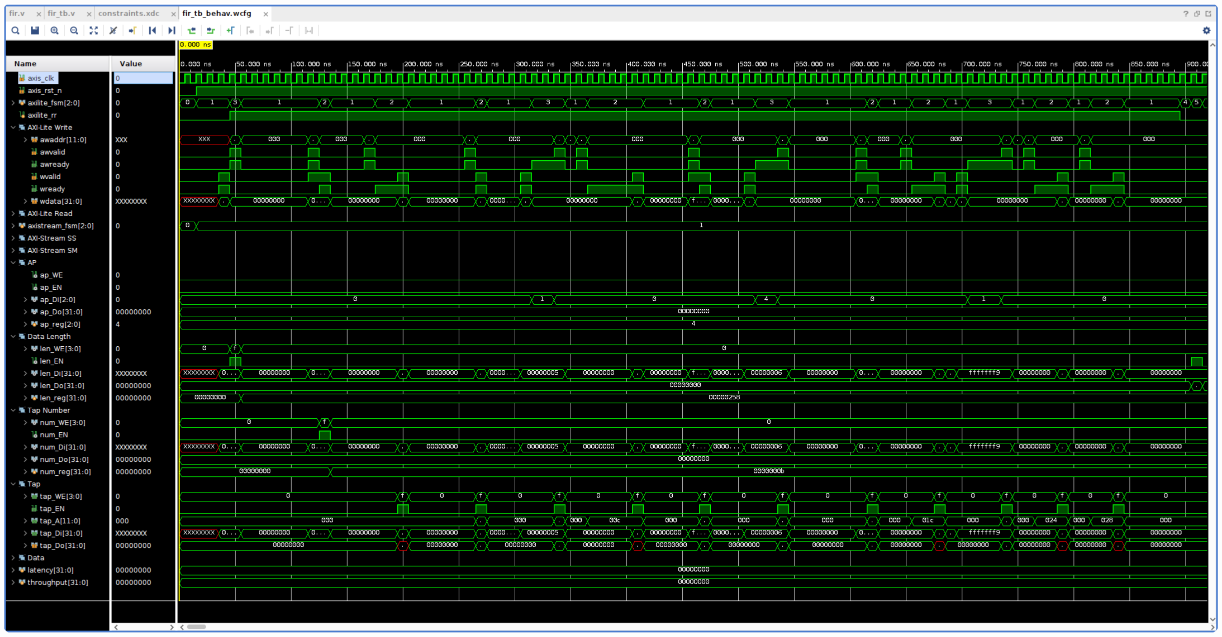
To transmit data, the transmitter asserts "tvalid" to indicate the data is valid to be read by the receiver. When the receiver is ready to retrieve the data through "tdata", it asserts "tready" to indicate to the transmitter to send the next set of data in the stream. The cycle continues until the transmitter sets "tlast" along with the "tvalid" to indicate the last set of data to be sent. After receiving "tlast" and receiving the "tready" from the receiver, the transmitter will halt the stream and set "tvalid" to logic 0.

Configuration Registers and Tap Coefficient

When the system first turns on, the system is in "reset" state due to the "axis_rst_n" set to logic low (active-low). The system becomes "idle" state once "axis_rst_n" is set to logic high. The ap_idle in the AP Configuration Register is set to logic 1, while ap_start and ap_done is set to logic 0.

Before the start of the operation, the configuration of the data length, number of taps, and the tap coefficient must be written to the system. The data length and the number of taps are saved in their respective Configuration Registers, while the tap coefficients are saved into a BRAM. The AXI-Lite protocol is used to write the data into these configuration registers and the BRAM. The system contains a AXI-Lite Finite State Machine (FSM) to control the read/writes based on the AXI-Lite protocol mentioned above. The following are the correspond addresses for the AXI-Lite protocol.

- 0x00 AP Configuration register
- 0x10-13 Data Length Configuration Register
- 0x14-18 Tap Number Configuration Register
- 0x40-7F Tap Coefficient BRAM

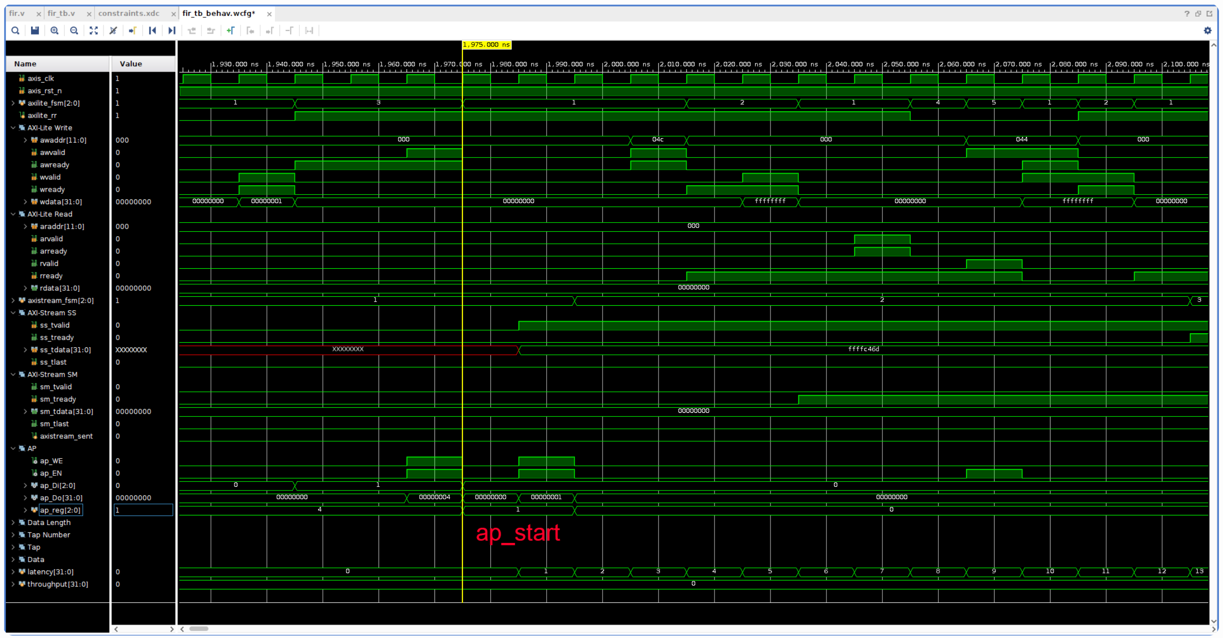


Each AXI-Lite operation only writes to one address at a time. Therefore, multiple AXI-Lite operations are needed to fully program the data into the system. When the data length, number of taps, and the tap coefficient are written to their respective locations, the system can start operation.

FIR Computation

To start the operation, the ap_start is generated by writing logic 1 through AXI-Lite Write. If ap_idle is logic 0, the write is ignored by the configuration register as the system is already running. Writing ap_start as logic 1 will also set ap_idle to logic 0.

Setting ap_start allows the FIR computation to begin and enables the use of the AXI-Stream FSM to stream-in and stream-out the data through the AXI-Stream Protocol. In this lab, the name of the "Xn" stream-in data is "ss", while the "Yn" output stream is "sm". When the system detects "ss-tready" is asserted, the axi-stream protocol starts streaming in data for the FIR computation. In addition, the ap configuration register performs a write to set ap_start to logic 0.



When the FIR computation is started, the data BRAM is first initated by writing 0 to all addresses such that there is no "x" unknown state during computation.

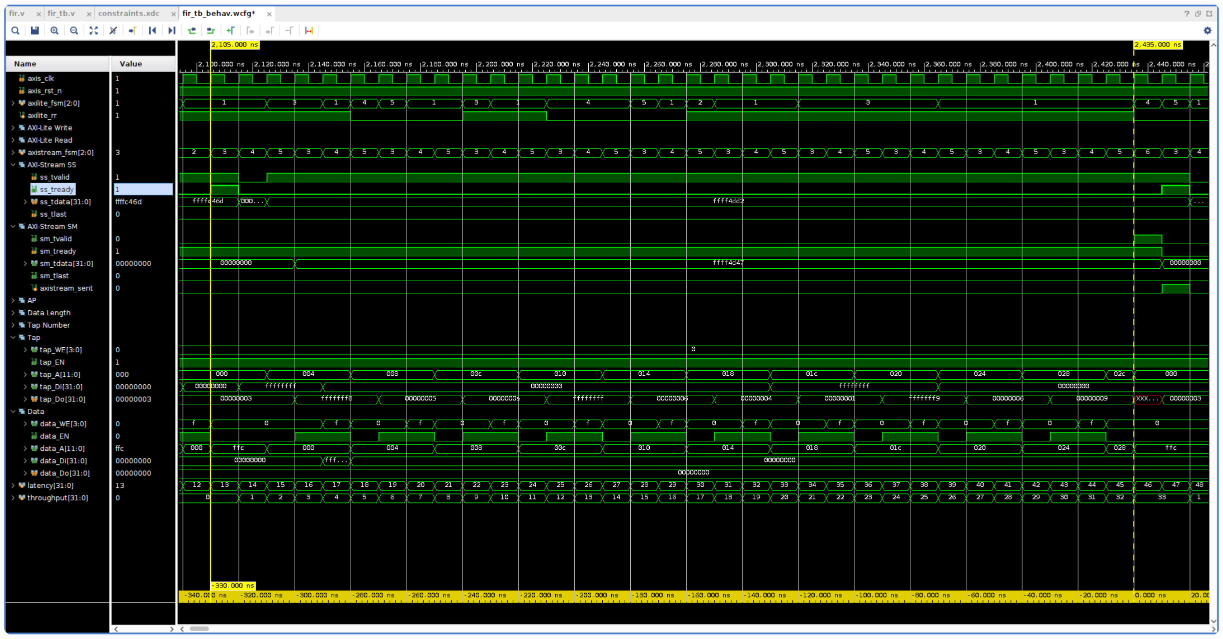
The FIR computation follows the following formula:

> y[t] = Σ (h[i] * x[t - i])

To perform the FIR computation, the "Xn" data is streamed in through AXI-Stream "ss-tdata". "ss-tready" is set such that the stream-in data can prepare the next set of data. At the same time, the tap coefficient 0 is read from at address[0] (0x40-43) of the Tap BRAM. The "Xn" stream-in data is multiplied by with the tap coefficient that was read. The mutiplied data is then stored and saved in a register.

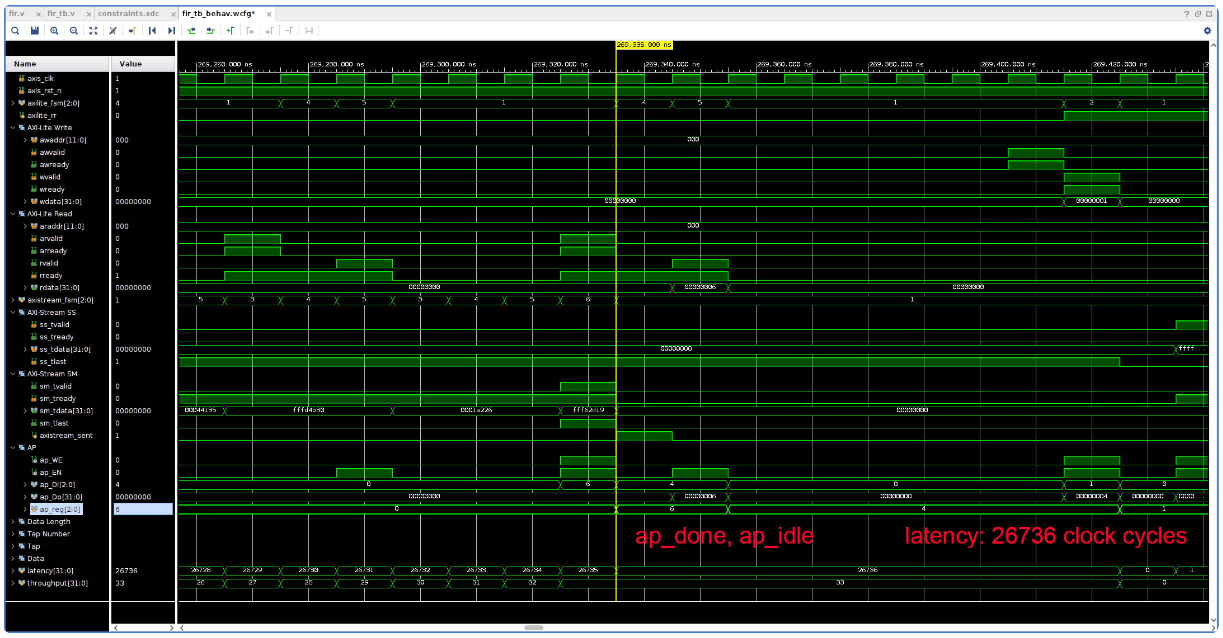
After multiplying, the Data from address[0] is read from the data BRAM, and the Tap coefficient 1 is read from address[1] (0x44-47). The data from stream-in is written to data BRAM address[0] in the next cycle, and the original data from data BRAM address[0] is multiplied with the Tap coefficient 1. This is then added with the previous multiply result saved in a register. In the subsequent cycle, a new data and tap coefficient is read from the next address and the previous data is written to the next address, causing a "shift" in data. The data is multiplied with the tap coefficient and added to the sum of all the multiples.

When the address reaches the end (depends on the number of taps), the last tap coefficient from the tap BRAM and the data from data BRAM is read out. The computed result of the FIR is then stream-out as "Yn" through "sm-tdata" in the AXI-Stream protocol. The data that was read out of the RAM is shifted out and removed from the data BRAM. The system repeats the steps above to compute the next set of FIR computation by streaming-in the next "Xn".



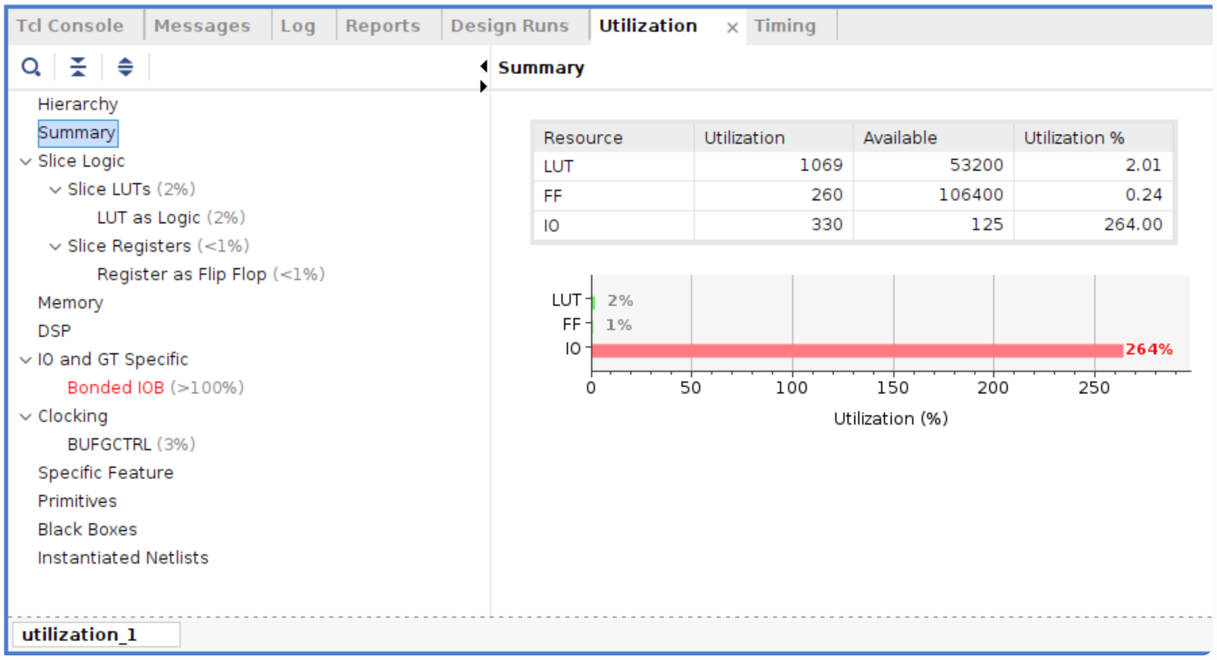
When the system stream-in the last data (determined by the data length), "ss-tlast" is asserted. The final round of FIR computation is performed. When the last FIR computation is completed, "Yn" is streamed out along with "sm-tlast" to indicate the last stream-out operation. The AXI-Stream FSM sets "ap_idle" and "ap_done" to logic 1.

To check the status on the FIR engine, the system can sample the AP configuration register for the current status. The system samples "ap_done" to check for the completion of the FIR engine. "ap_done" will be cleared and reset to logic 0 once it is read. The FIR engine can then be reconfigured and perform the FIR computation after setting "ap_start" to logic 1 again.

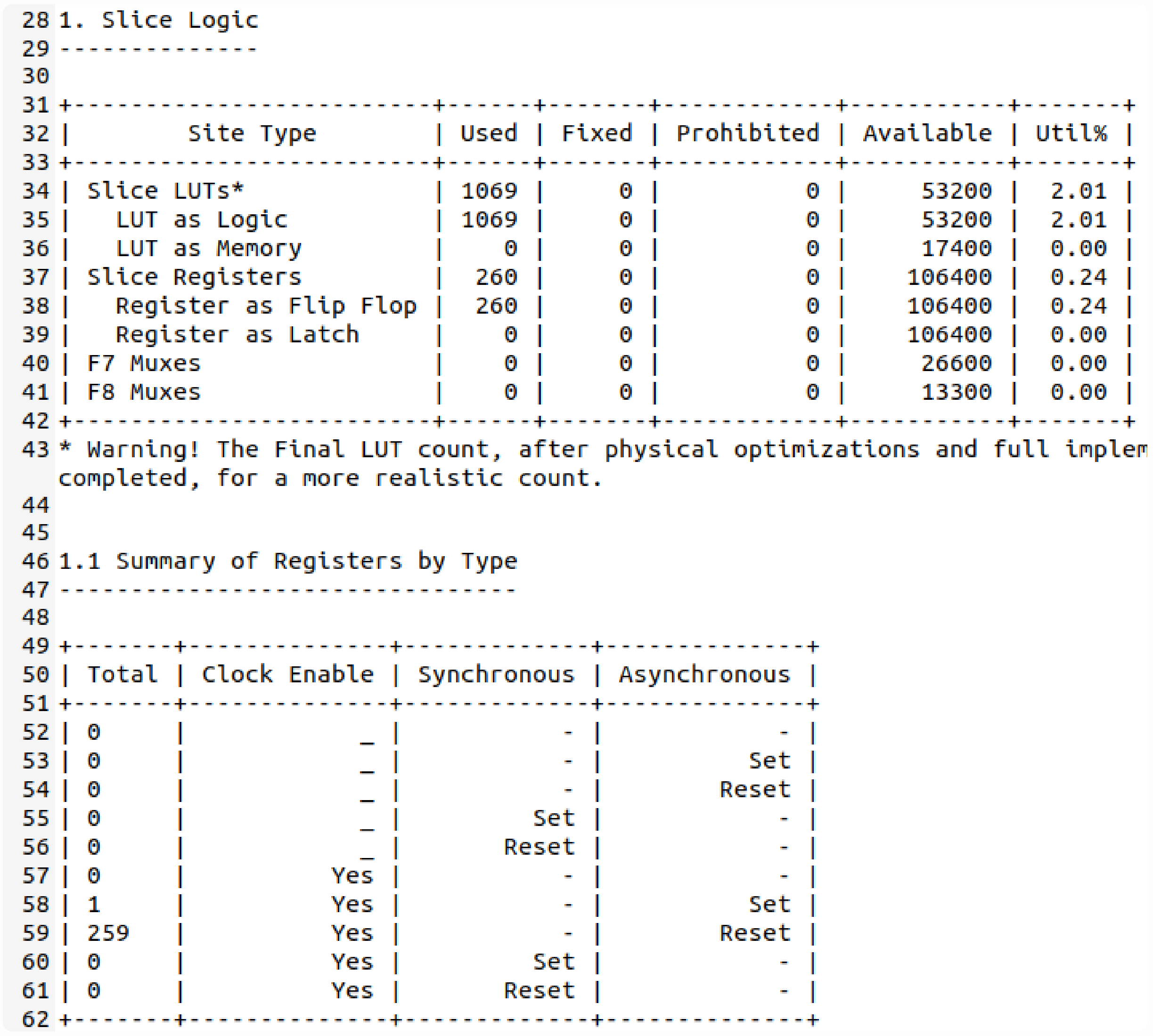


Resource Usage

FF, LUT



Resource	Utilization	Available	Utilization %
LUT	1069	53200	2.01
FF	260	106400	0.24
IO	330	125	264.00



```
1. Slice Logic
--------------

+-------------------------+------+-------+------------+-----------+-------+
|        Site Type        | Used | Fixed | Prohibited | Available | Util% |
+-------------------------+------+-------+------------+-----------+-------+
| Slice LUTs*             | 1069 |     0 |          0 |     53200 |  2.01 |
|   LUT as Logic          | 1069 |     0 |          0 |     53200 |  2.01 |
|   LUT as Memory         |    0 |     0 |          0 |     17400 |  0.00 |
| Slice Registers         |  260 |     0 |          0 |    106400 |  0.24 |
|   Register as Flip Flop |  260 |     0 |          0 |    106400 |  0.24 |
|   Register as Latch     |    0 |     0 |          0 |    106400 |  0.00 |
| F7 Muxes                |    0 |     0 |          0 |     26600 |  0.00 |
| F8 Muxes                |    0 |     0 |          0 |     13300 |  0.00 |
+-------------------------+------+-------+------------+-----------+-------+
* Warning! The Final LUT count, after physical optimizations and full implem
completed, for a more realistic count.


1.1 Summary of Registers by Type
--------------------------------

+-------+--------------+-------------+--------------+
| Total | Clock Enable | Synchronous | Asynchronous |
+-------+--------------+-------------+--------------+
| 0     |            _ |           - |            - |
| 0     |            _ |           - |          Set |
| 0     |            _ |           - |        Reset |
| 0     |            _ |         Set |            - |
| 0     |            _ |       Reset |            - |
| 0     |          Yes |           - |            - |
| 1     |          Yes |           - |          Set |
| 259   |          Yes |           - |        Reset |
| 0     |          Yes |         Set |            - |
| 0     |          Yes |       Reset |            - |
+-------+--------------+-------------+--------------+
```

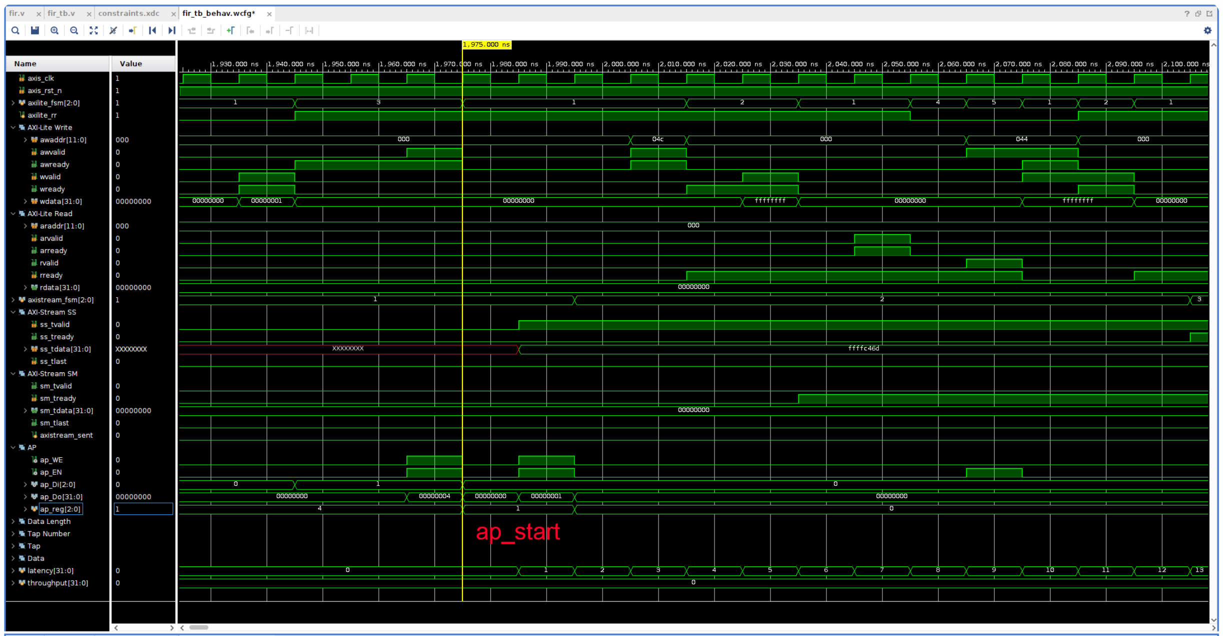
BRAM

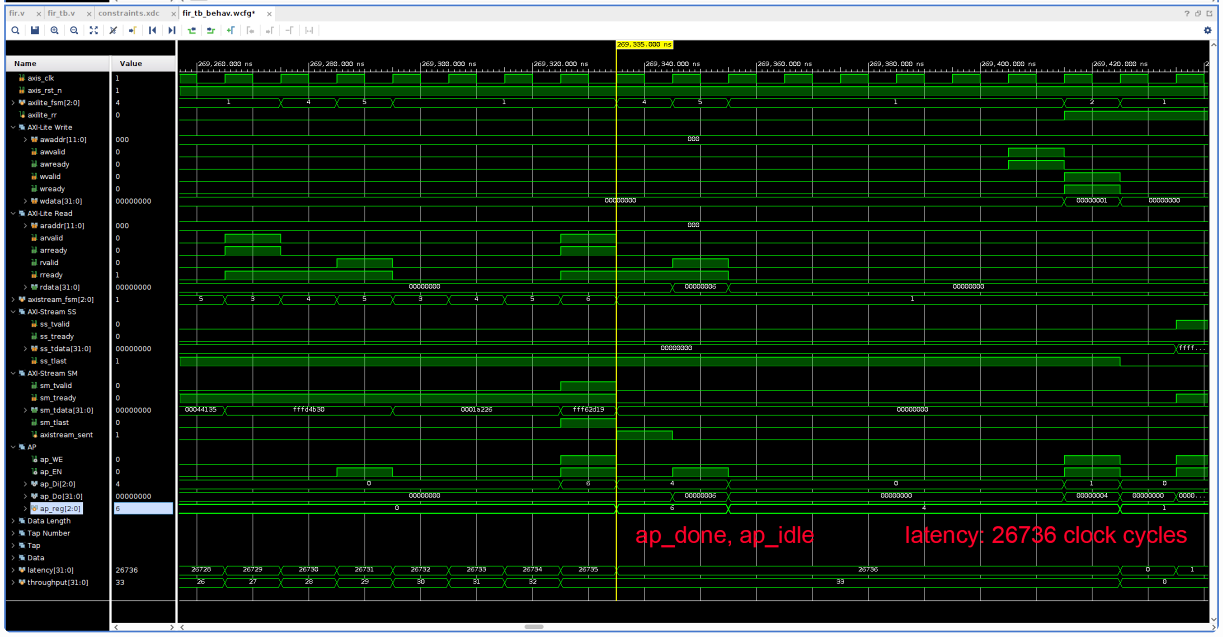
```
2. Memory
---------

+----------------+------+-------+------------+-----------+-------+
|    Site Type   | Used | Fixed | Prohibited | Available | Util% |
+----------------+------+-------+------------+-----------+-------+
| Block RAM Tile |    0 |     0 |          0 |       140 |  0.00 |
|   RAMB36/FIFO* |    0 |     0 |          0 |       140 |  0.00 |
|   RAMB18       |    0 |     0 |          0 |       280 |  0.00 |
+----------------+------+-------+------------+-----------+-------+
* Note: Each Block RAM Tile only has one FIFO logic available and t
Block RAM Tile, that tile can still accommodate a RAMB18E1
```

Performance Report

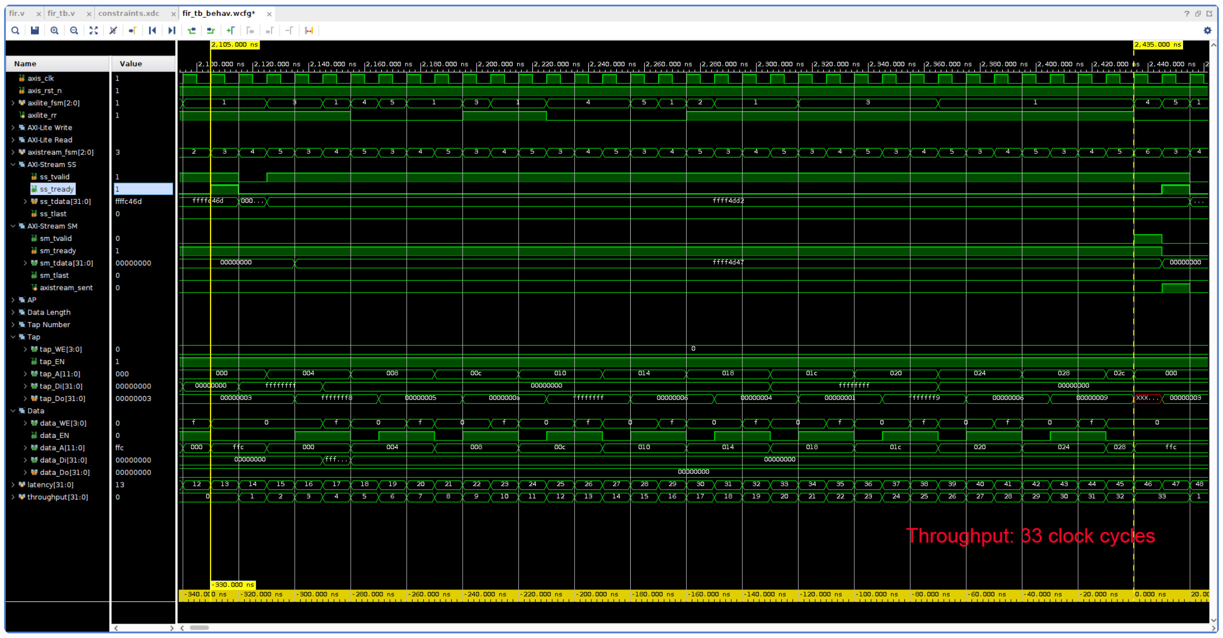
Latency





```
ap_done sampled!
------End the ap_done sampling(AXI-Lite)------
------End the data output(AXI-Stream)------
------End the data input(AXI-Stream)------
------End the illegal tap sampling(AXI-Lite)------
The latency is       26736 cycles
```

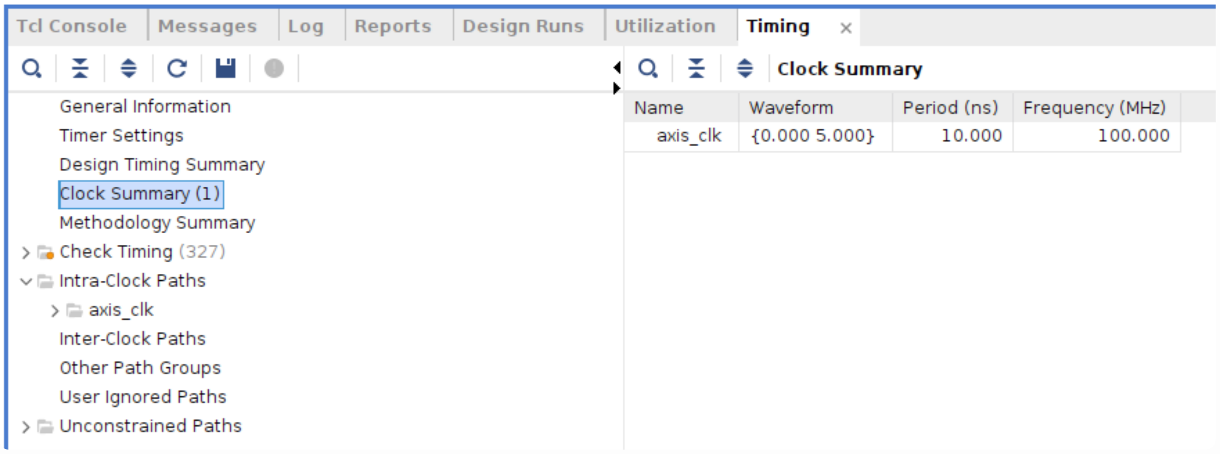
Throughput



```
-----------Start simulation----------
 Start FIR, Round          1
----Start the data input(AXI-Stream)----
----Start the data output(AXI-Stream)----
----Start the ap_done sampling(AXI-Lite)----
----Start the illegal tap sampling(AXI-Lite)----
AXI-Stream inputting data          0...
AXI-Stream outputting data          0...
The throughput is          33 cycles
AXI-Stream inputting data          1...
AXI-Stream outputting data          1...
The throughput is          33 cycles
AXI-Stream inputting data          2...
AXI-Stream outputting data          2...
The throughput is          33 cycles
```

Timing Report

Frequency

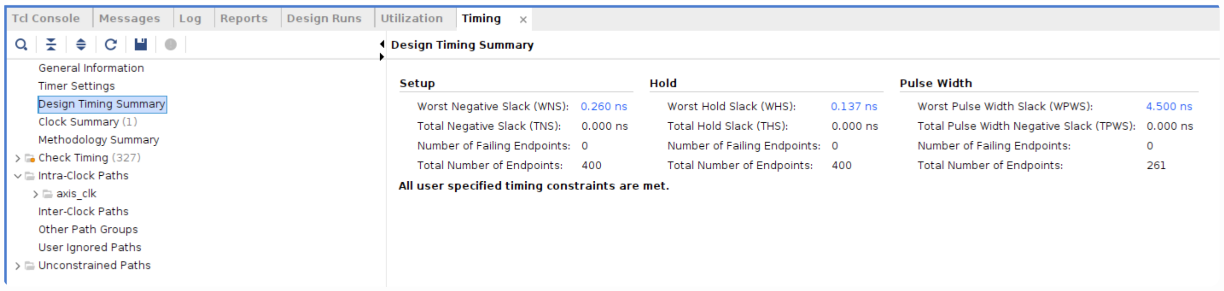


Clock Summary

- General Information
- Timer Settings
- Design Timing Summary
- Clock Summary (1)
- Methodology Summary
- Check Timing (327)
- Intra-Clock Paths
  - axis_clk
- Inter-Clock Paths
- Other Path Groups
- User Ignored Paths
- Unconstrained Paths

Name	Waveform	Period (ns)	Frequency (MHz)
axis_clk	{0.000 5.000}	10.000	100.000

Report Timing



Tcl Console | Messages | Log | Reports | Design Runs | Utilization | Timing

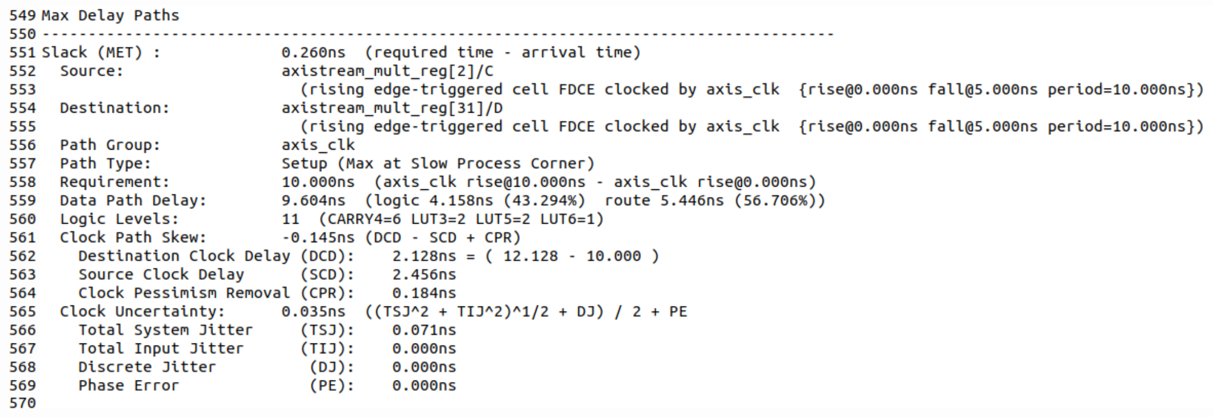
General Information
Timer Settings
Design Timing Summary
Clock Summary (1)
Methodology Summary
Check Timing (327)
Intra-Clock Paths
axis_clk
Inter-Clock Paths
Other Path Groups
User Ignored Paths
Unconstrained Paths

Design Timing Summary

Setup		Hold		Pulse Width	
Worst Negative Slack (WNS):	0.260 ns	Worst Hold Slack (WHS):	0.137 ns	Worst Pulse Width Slack (WPWS):	4.500 ns
Total Negative Slack (TNS):	0.000 ns	Total Hold Slack (THS):	0.000 ns	Total Pulse Width Negative Slack (TPWS):	0.000 ns
Number of Failing Endpoints:	0	Number of Failing Endpoints:	0	Number of Failing Endpoints:	0
Total Number of Endpoints:	400	Total Number of Endpoints:	400	Total Number of Endpoints:	261

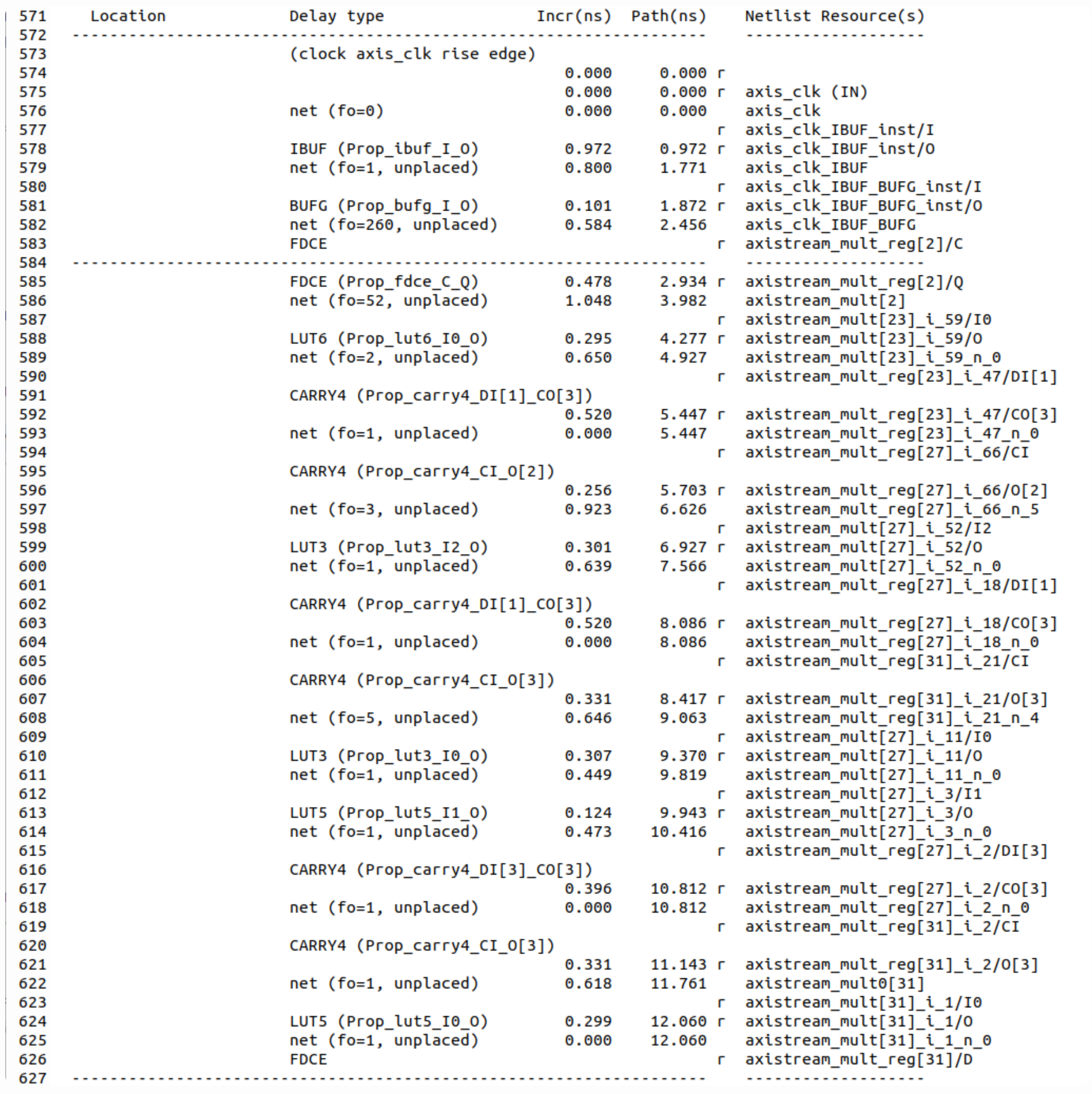
All user specified timing constraints are met.

Max Delay Path



```
Max Delay Paths
--------------------------------------------------------------------------------------
Slack (MET) :             0.260ns  (required time - arrival time)
  Source:                 axistream_mult_reg[2]/C
                            (rising edge-triggered cell FDCE clocked by axis_clk  {rise@0.000ns fall@5.000ns period=10.000ns})
  Destination:            axistream_mult_reg[31]/D
                            (rising edge-triggered cell FDCE clocked by axis_clk  {rise@0.000ns fall@5.000ns period=10.000ns})
  Path Group:             axis_clk
  Path Type:              Setup (Max at Slow Process Corner)
  Requirement:            10.000ns  (axis_clk rise@10.000ns - axis_clk rise@0.000ns)
  Data Path Delay:        9.604ns  (logic 4.158ns (43.294%)  route 5.446ns (56.706%))
  Logic Levels:           11  (CARRY4=6 LUT3=2 LUT5=2 LUT6=1)
  Clock Path Skew:        -0.145ns (DCD - SCD + CPR)
    Destination Clock Delay (DCD):    2.128ns = ( 12.128 - 10.000 )
    Source Clock Delay      (SCD):    2.456ns
    Clock Pessimism Removal (CPR):    0.184ns
  Clock Uncertainty:      0.035ns  ((TSJ^2 + TIJ^2)^1/2 + DJ) / 2 + PE
    Total System Jitter     (TSJ):    0.071ns
    Total Input Jitter      (TIJ):    0.000ns
    Discrete Jitter          (DJ):    0.000ns
    Phase Error              (PE):    0.000ns

```



```
    Location             Delay type                Incr(ns)  Path(ns)    Netlist Resource(s)
  -------------------------------------------------------------------    -------------------
                         (clock axis_clk rise edge)
                                                      0.000     0.000 r
                                                      0.000     0.000 r  axis_clk (IN)
                         net (fo=0)                   0.000     0.000    axis_clk
                                                                      r  axis_clk_IBUF_inst/I
                         IBUF (Prop_ibuf_I_O)         0.972     0.972 r  axis_clk_IBUF_inst/O
                         net (fo=1, unplaced)         0.800     1.771    axis_clk_IBUF
                                                                      r  axis_clk_IBUF_BUFG_inst/I
                         BUFG (Prop_bufg_I_O)         0.101     1.872 r  axis_clk_IBUF_BUFG_inst/O
                         net (fo=260, unplaced)       0.584     2.456    axis_clk_IBUF_BUFG
                         FDCE                                         r  axistream_mult_reg[2]/C
  -------------------------------------------------------------------    -------------------
                         FDCE (Prop_fdce_C_Q)         0.478     2.934 r  axistream_mult_reg[2]/Q
                         net (fo=52, unplaced)        1.048     3.982    axistream_mult[2]
                                                                      r  axistream_mult[23]_i_59/I0
                         LUT6 (Prop_lut6_I0_O)        0.295     4.277 r  axistream_mult[23]_i_59/O
                         net (fo=2, unplaced)         0.650     4.927    axistream_mult[23]_i_59_n_0
                                                                      r  axistream_mult_reg[23]_i_47/DI[1]
                         CARRY4 (Prop_carry4_DI[1]_CO[3])
                                                      0.520     5.447 r  axistream_mult_reg[23]_i_47/CO[3]
                         net (fo=1, unplaced)         0.000     5.447    axistream_mult_reg[23]_i_47_n_0
                                                                      r  axistream_mult_reg[27]_i_66/CI
                         CARRY4 (Prop_carry4_CI_O[2])
                                                      0.256     5.703 r  axistream_mult_reg[27]_i_66/O[2]
                         net (fo=3, unplaced)         0.923     6.626    axistream_mult_reg[27]_i_66_n_5
                                                                      r  axistream_mult[27]_i_52/I2
                         LUT3 (Prop_lut3_I2_O)        0.301     6.927 r  axistream_mult[27]_i_52/O
                         net (fo=1, unplaced)         0.639     7.566    axistream_mult[27]_i_52_n_0
                                                                      r  axistream_mult_reg[27]_i_18/DI[1]
                         CARRY4 (Prop_carry4_DI[1]_CO[3])
                                                      0.520     8.086 r  axistream_mult_reg[27]_i_18/CO[3]
                         net (fo=1, unplaced)         0.000     8.086    axistream_mult_reg[27]_i_18_n_0
                                                                      r  axistream_mult_reg[31]_i_21/CI
                         CARRY4 (Prop_carry4_CI_O[3])
                                                      0.331     8.417 r  axistream_mult_reg[31]_i_21/O[3]
                         net (fo=5, unplaced)         0.646     9.063    axistream_mult_reg[31]_i_21_n_4
                                                                      r  axistream_mult[27]_i_11/I0
                         LUT3 (Prop_lut3_I0_O)        0.307     9.370 r  axistream_mult[27]_i_11/O
                         net (fo=1, unplaced)         0.449     9.819    axistream_mult[27]_i_11_n_0
                                                                      r  axistream_mult[27]_i_3/I1
                         LUT5 (Prop_lut5_I1_O)        0.124     9.943 r  axistream_mult[27]_i_3/O
                         net (fo=1, unplaced)         0.473    10.416    axistream_mult[27]_i_3_n_0
                                                                      r  axistream_mult_reg[27]_i_2/DI[3]
                         CARRY4 (Prop_carry4_DI[3]_CO[3])
                                                      0.396    10.812 r  axistream_mult_reg[27]_i_2/CO[3]
                         net (fo=1, unplaced)         0.000    10.812    axistream_mult_reg[27]_i_2_n_0
                                                                      r  axistream_mult_reg[31]_i_2/CI
                         CARRY4 (Prop_carry4_CI_O[3])
                                                      0.331    11.143 r  axistream_mult_reg[31]_i_2/O[3]
                         net (fo=1, unplaced)         0.618    11.761    axistream_mult0[31]
                                                                      r  axistream_mult[31]_i_1/I0
                         LUT5 (Prop_lut5_I0_O)        0.299    12.060 r  axistream_mult[31]_i_1/O
                         net (fo=1, unplaced)         0.000    12.060    axistream_mult[31]_i_1_n_0
                         FDCE                                         r  axistream_mult_reg[31]/D
  -------------------------------------------------------------------    -------------------
```

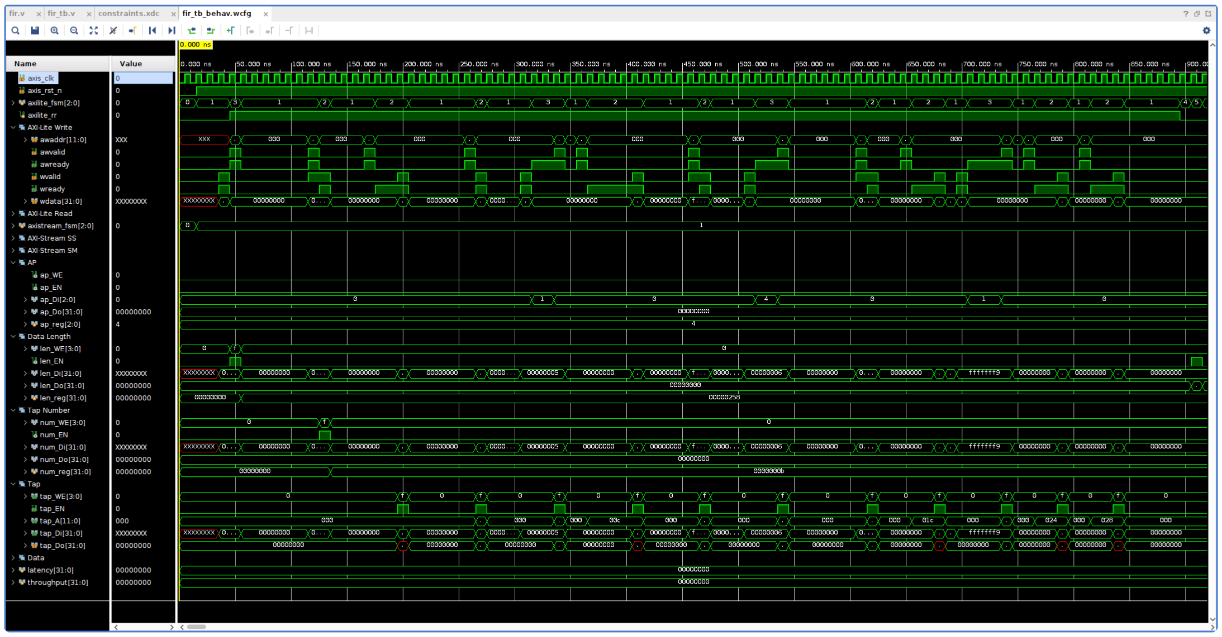
```

                       (clock axis_clk rise edge)
                                                   10.000    10.000 r
                                                    0.000    10.000 r  axis_clk (IN)
                       net (fo=0)                   0.000    10.000    axis_clk
                                                                    r  axis_clk_IBUF_inst/I
                       IBUF (Prop_ibuf_I_O)         0.838    10.838 r  axis_clk_IBUF_inst/O
                       net (fo=1, unplaced)         0.760    11.598    axis_clk_IBUF
                                                                    r  axis_clk_IBUF_BUFG_inst/I
                       BUFG (Prop_bufg_I_O)         0.091    11.689 r  axis_clk_IBUF_BUFG_inst/O
                       net (fo=260, unplaced)       0.439    12.128    axis_clk_IBUF_BUFG
                       FDCE                                         r  axistream_mult_reg[31]/C
                       clock pessimism              0.184    12.311
                       clock uncertainty           -0.035    12.276
                       FDCE (Setup_fdce_C_D)        0.044    12.320    axistream_mult_reg[31]
     -------------------------------------------------------------------
                       required time                         12.320
                       arrival time                         -12.060
     -------------------------------------------------------------------
                       slack                                  0.260
```

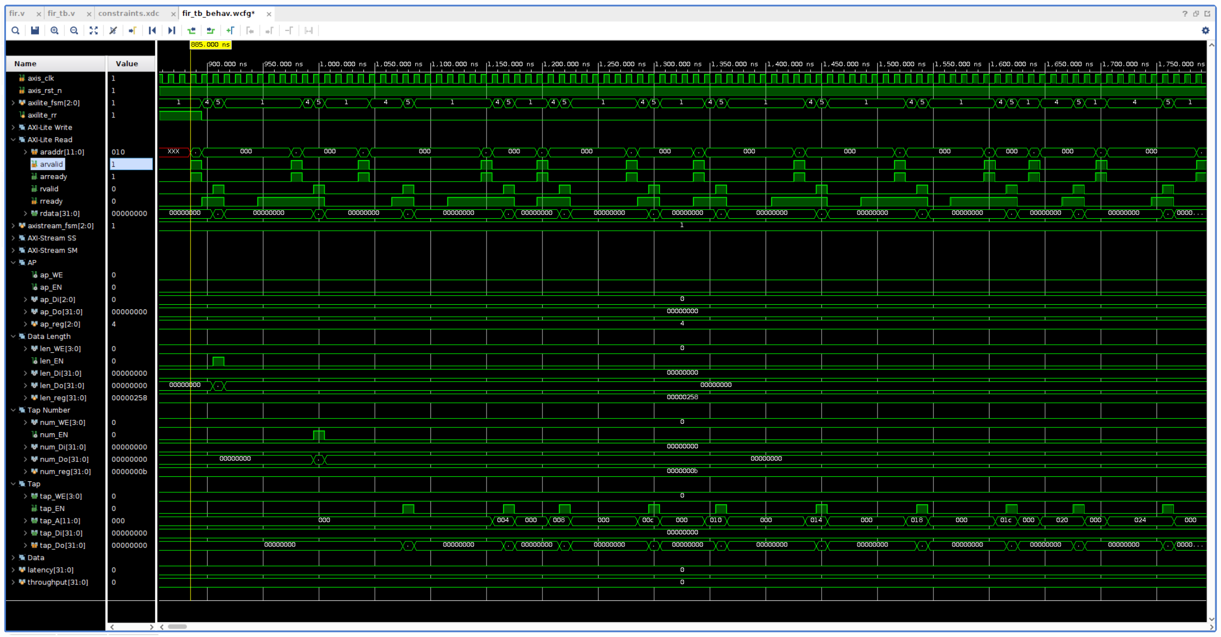
Simulation Waveform

Coefficient Program and Read Back

Coefficient Program

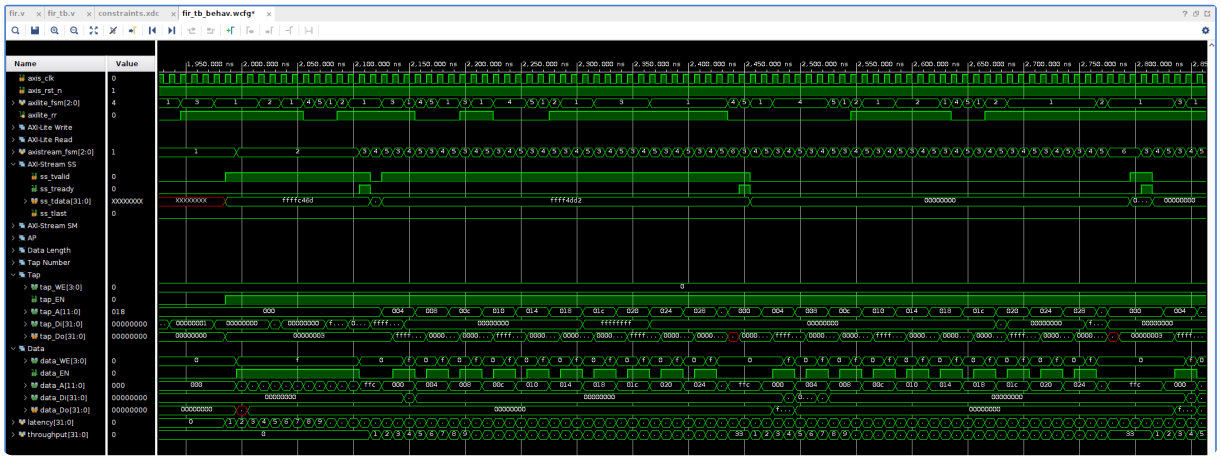


Read Back

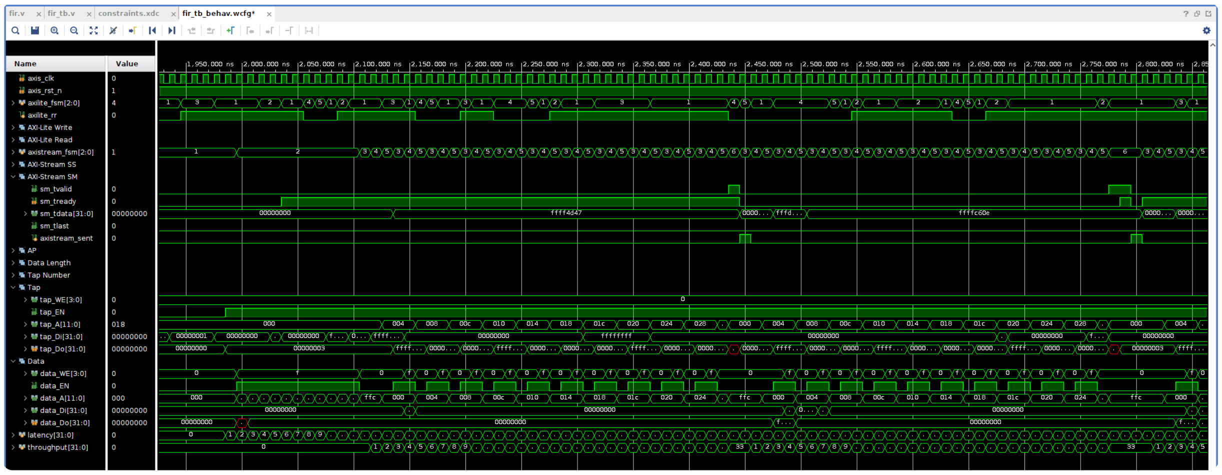


Data Stream

Data-in Stream-in

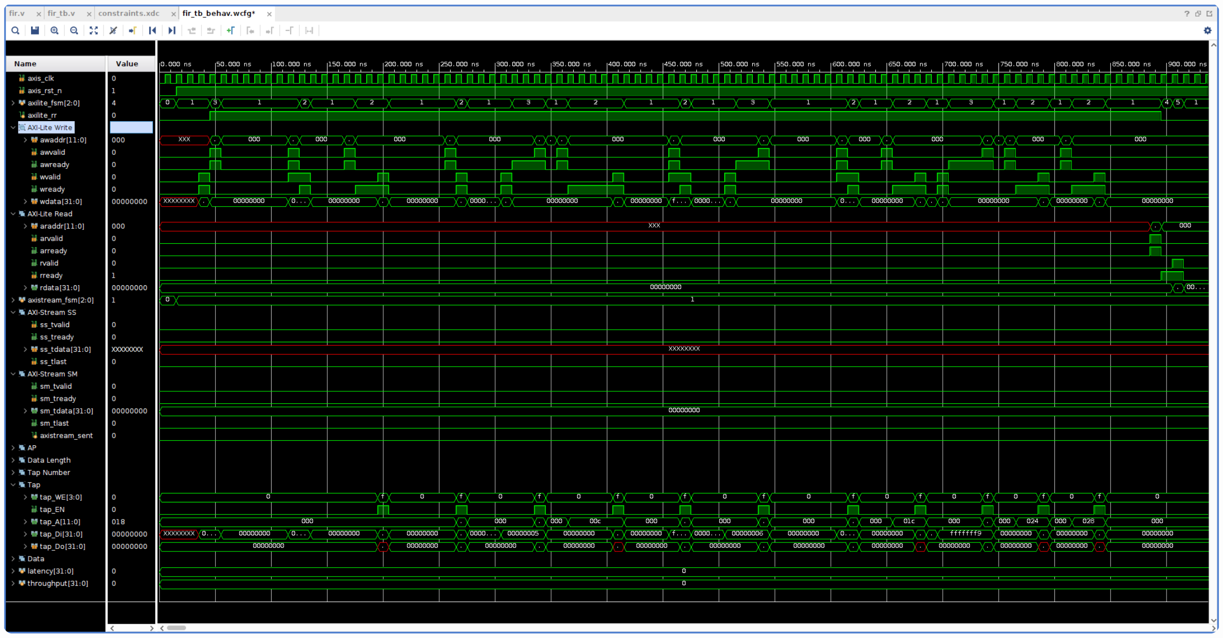


Data-out Stream-out

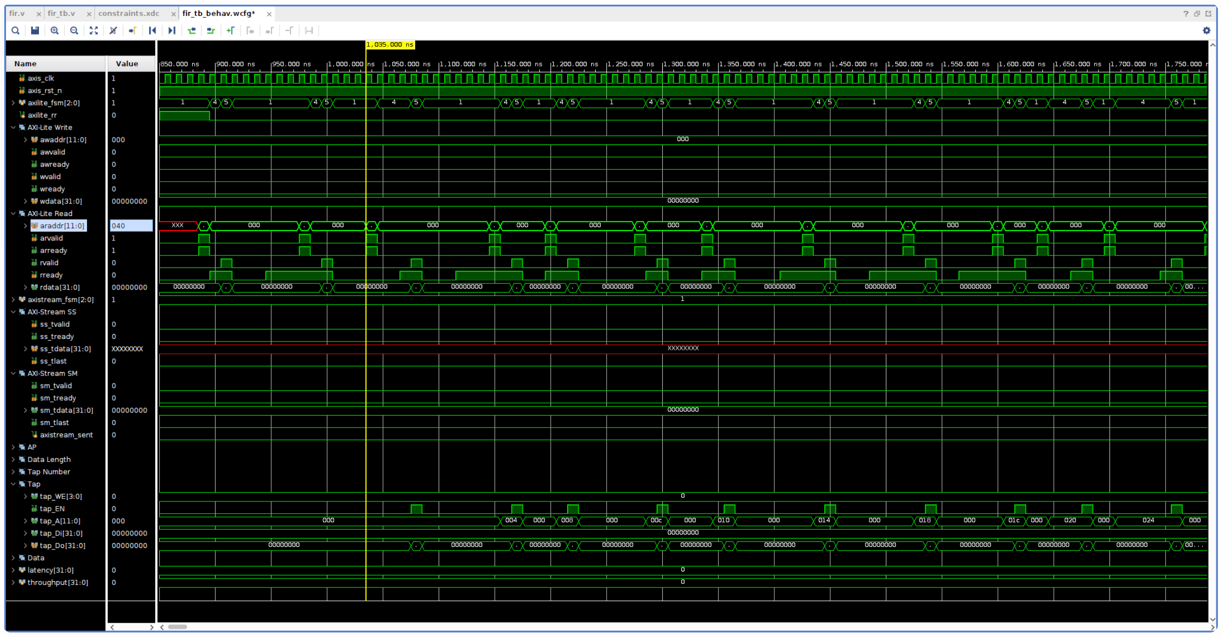


RAM Access Control

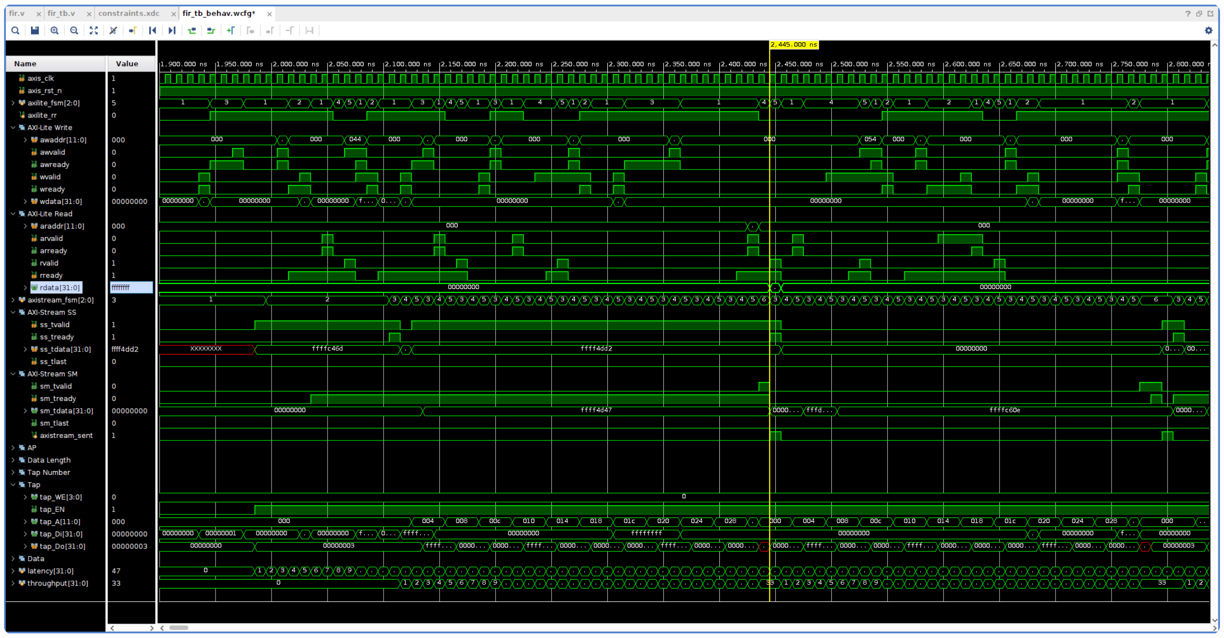
Tap BRAM (AXI-Lite Write during ap_idle)



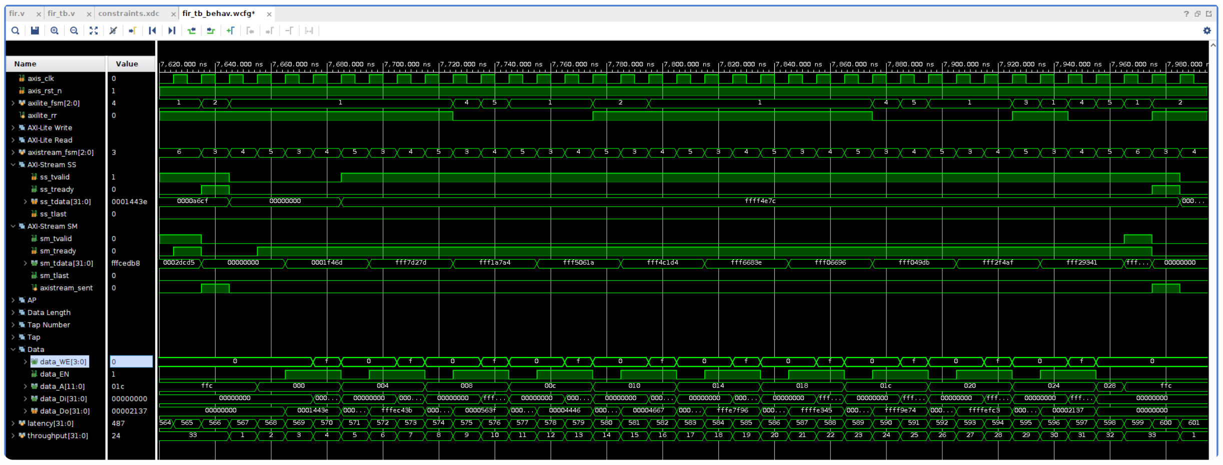
Tap BRAM (AXI-Lite Read during ap_idle)



Tap BRAM (Not ap_idle, drop write transaction, read invalid value 32'hffffffff)



Data BRAM (AXI-Stream in, Shift Data)



Github

https://github.com/AnthonyGithub/EESM6000C-Lab-3