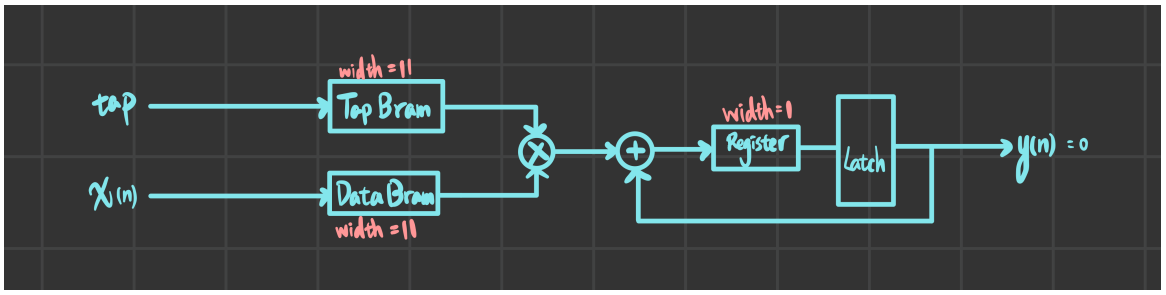
lab3_report

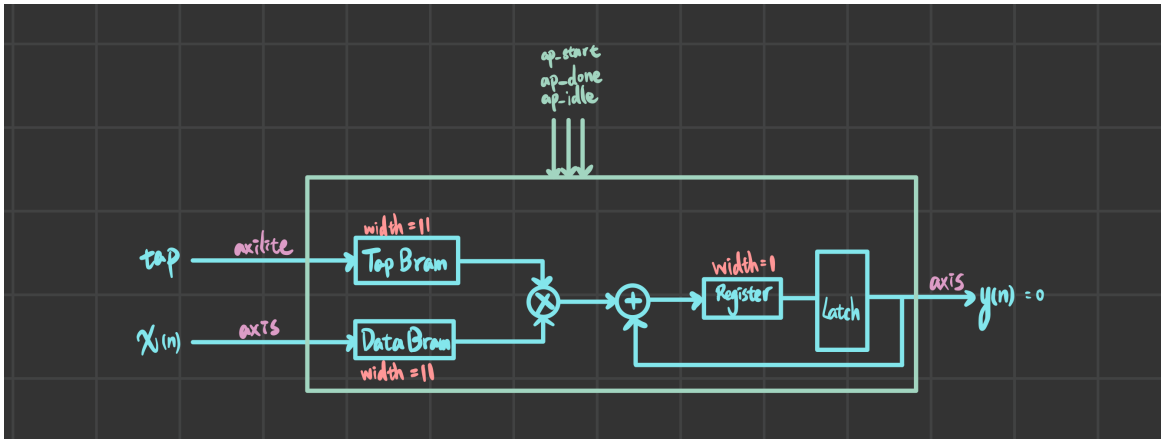
https://github.com/Josephood7/SoC-lab3.git

▼ Block Diagram

- Data path - dataflow



- Control Signals



▼ Operation Description

1. Load tap coefficient into tap block ram through AXI4-Lite
2. load input data into data block ram through AXI4-Stream
3. When both tap and data ram are full, start calculation using 1 multiplier and 1 adder
4. When sm_tready && sm_tvalid, output the result, FIR operation is done
5. FIR enter idle mode

- How to receive data-in and tap parameters and place into SRAM

  Output arready then wait for arvalid to tell where to put the upcoming tap coefficients. Output the rvalid to wait for rready to handshake for tap.

  When the FIR engine output the ss_tready signal and the data source input a ss_tvalid signal, a handshake is formed. The datas steam into the data RAM.

- How to access shiftram and tapRAM to do computation

  While ap_start is 1.

  If the state of FIR is BUSY, we can put the output of the shiftram and the tapRAM directly into our FIR kernel.

- How ap_done is generated

  If (sm_tvalid && sm_tready && sm_tlast) → ap_done = 1

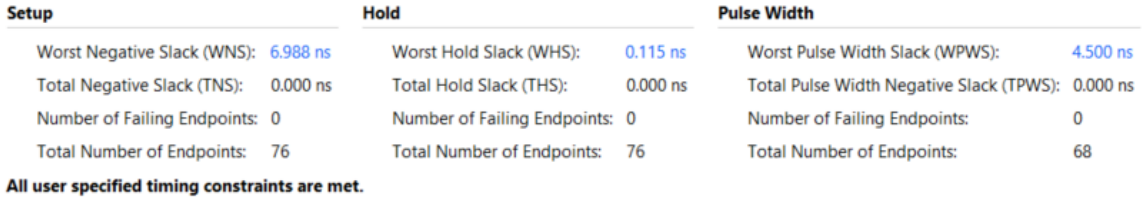
  Which means that while the FIR engine is in an output mode, the ap_done is generated once the last data is transmitted.

▼ Resource Usage

- Flip-flop → 71
- Lookup Table → 186
- Bram → 2
- Adder → 1 for FIR, 2 for tap& data counters
- Multiplier → 1

▼ Timing Report

Timing on longest path, slack

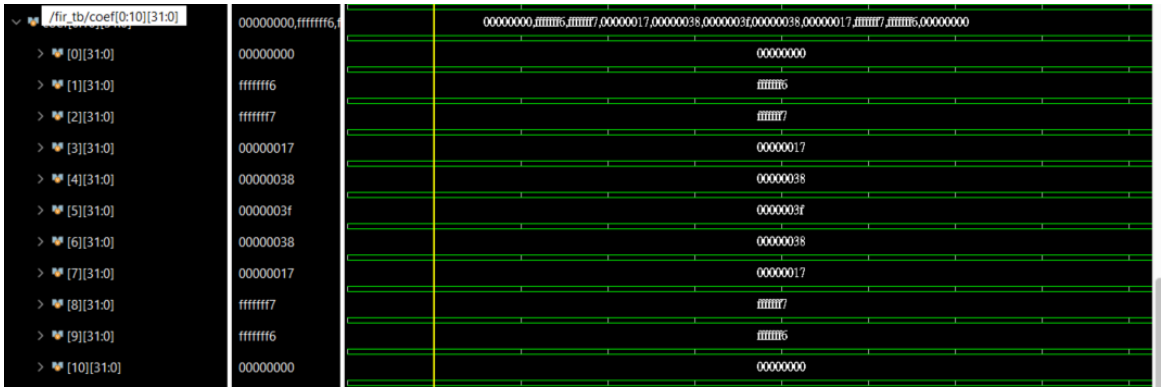


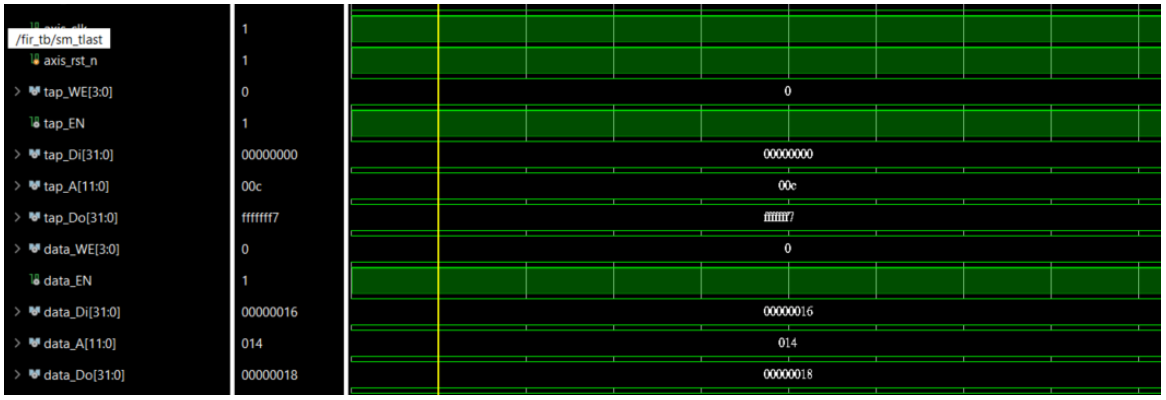
Setup		Hold		Pulse Width	
Worst Negative Slack (WNS):	6.988 ns	Worst Hold Slack (WHS):	0.115 ns	Worst Pulse Width Slack (WPWS):	4.500 ns
Total Negative Slack (TNS):	0.000 ns	Total Hold Slack (THS):	0.000 ns	Total Pulse Width Negative Slack (TPWS):	0.000 ns
Number of Failing Endpoints:	0	Number of Failing Endpoints:	0	Number of Failing Endpoints:	0
Total Number of Endpoints:	76	Total Number of Endpoints:	76	Total Number of Endpoints:	68

All user specified timing constraints are met.

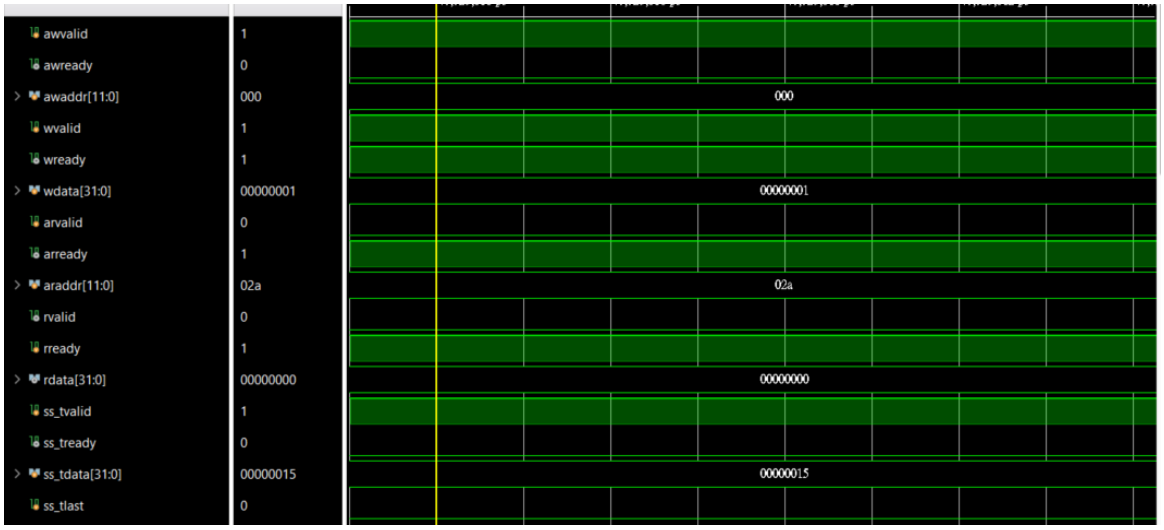
▼ Simulation Waveform

Coefficient program, and read back

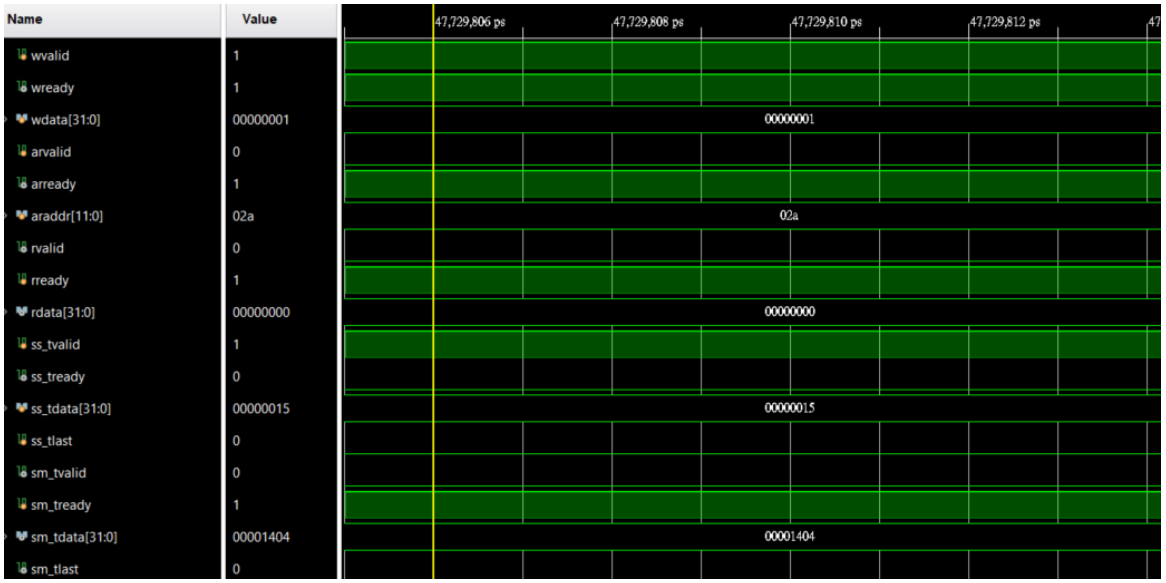




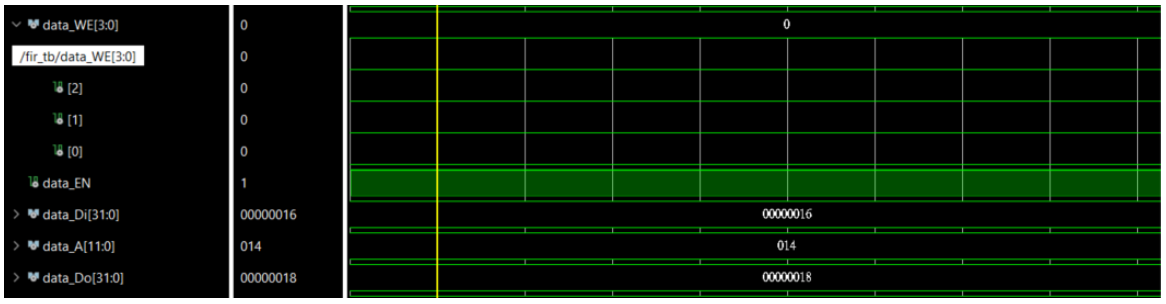
Data-in stream-in

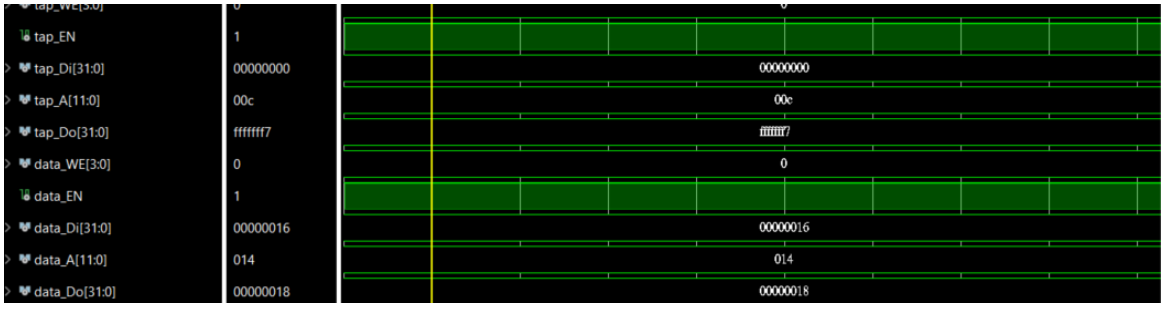


Data-out stream-out



RAM access control





FSM