Homework 3 Report

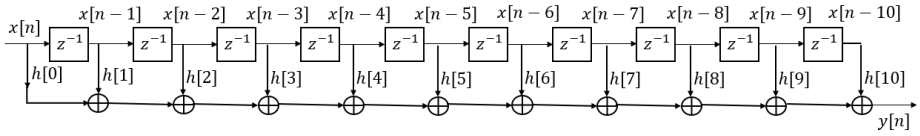
Student ID: 111064537 Name: 林亭君

Github link: github

1. Block Diagram
   - Algorithm – FIR

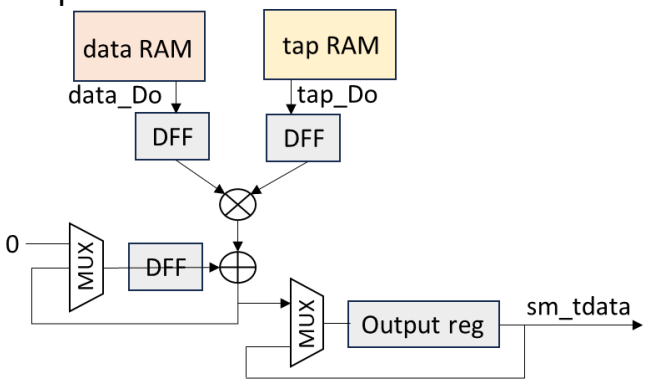
     11-tap FIR filter, but we can only use one adder and one multiplexer to implement

     $y[n] = \sum_{k=0}^{N-1} h[k]x[n-k], where\ N = 11$

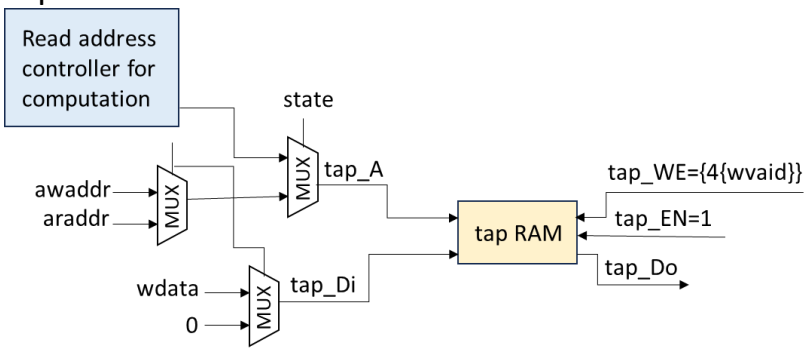


   - Datapath – dataflow

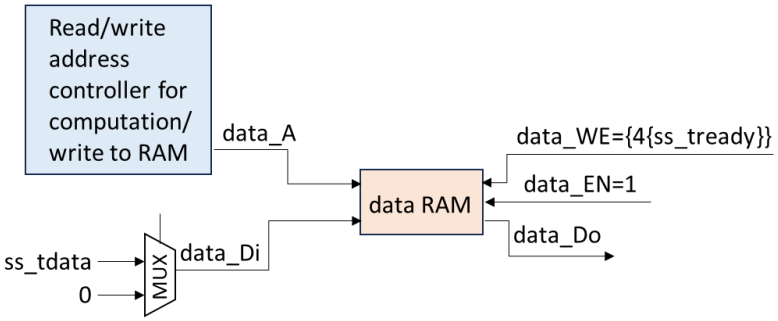
     Compute



     Tap RAM

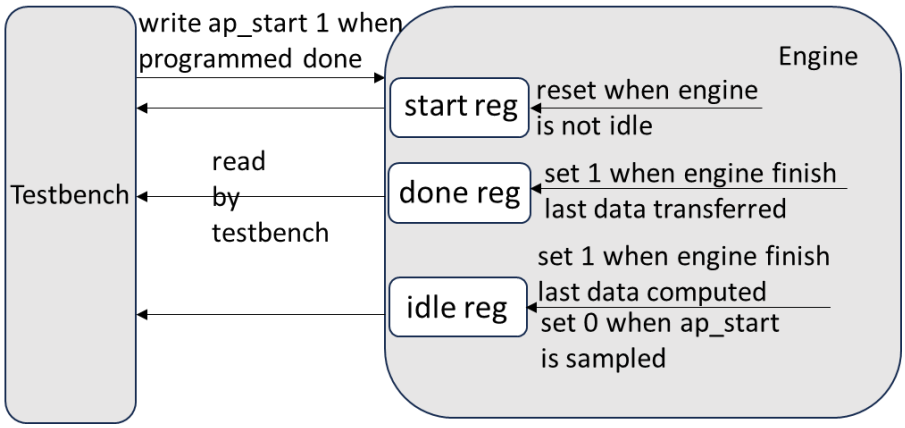


     Data RAM

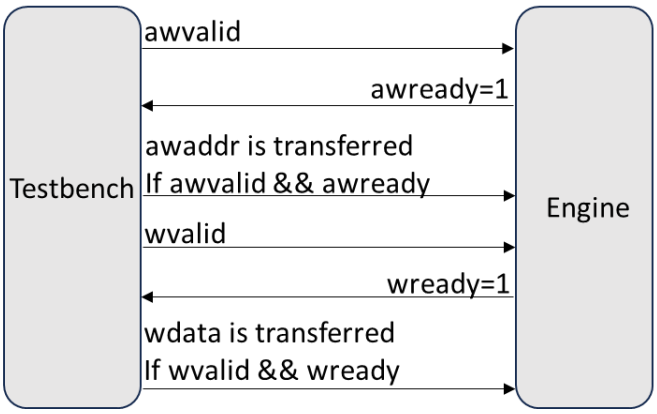


- Control signals

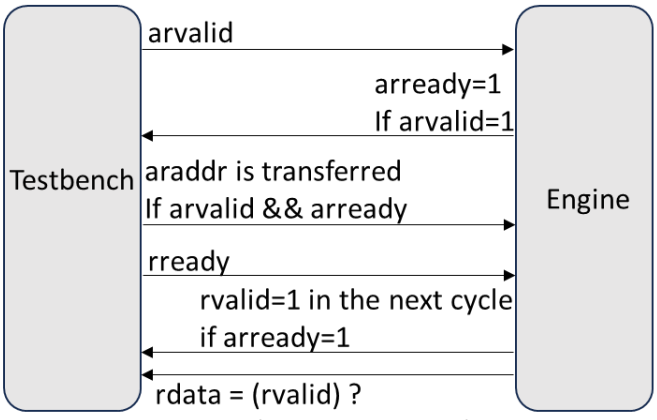
ap control



axi lite write

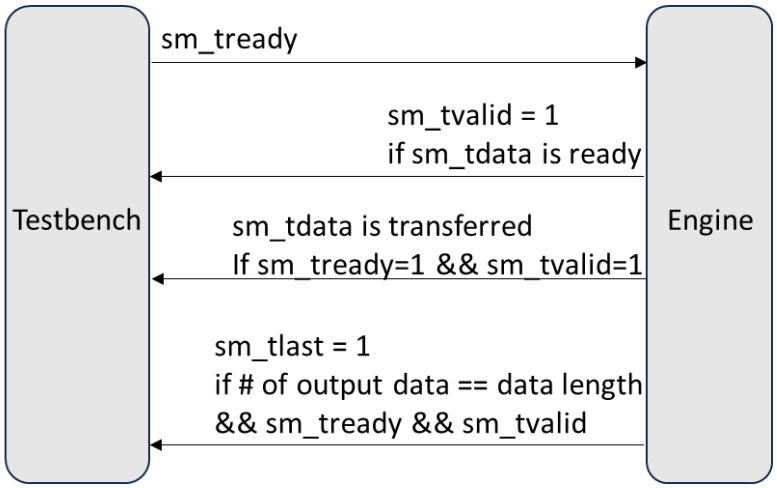


axi lite read

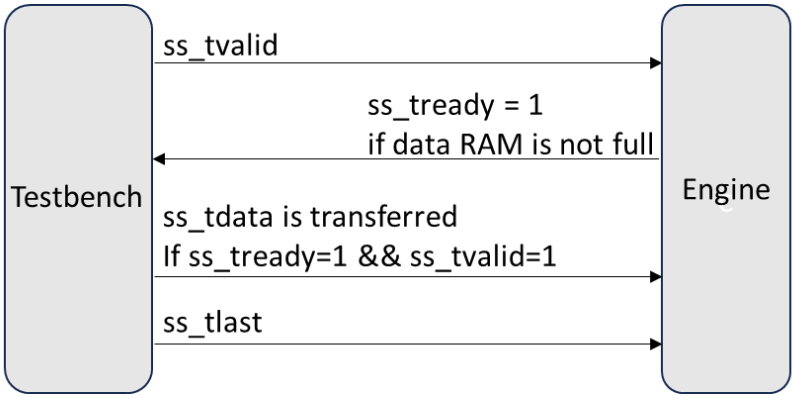


rdata = (rvalid) ?
(araddr==12'h00) ? ap reg :
(araddr>=12'h20) ? tap_Do : 0 : 0
And it is transferred if rvalid && rready

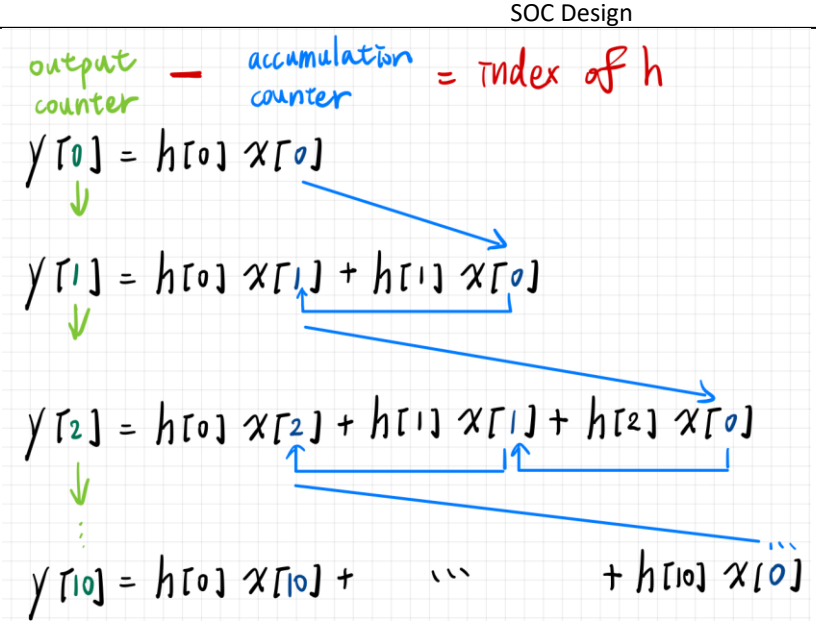
Data stream out

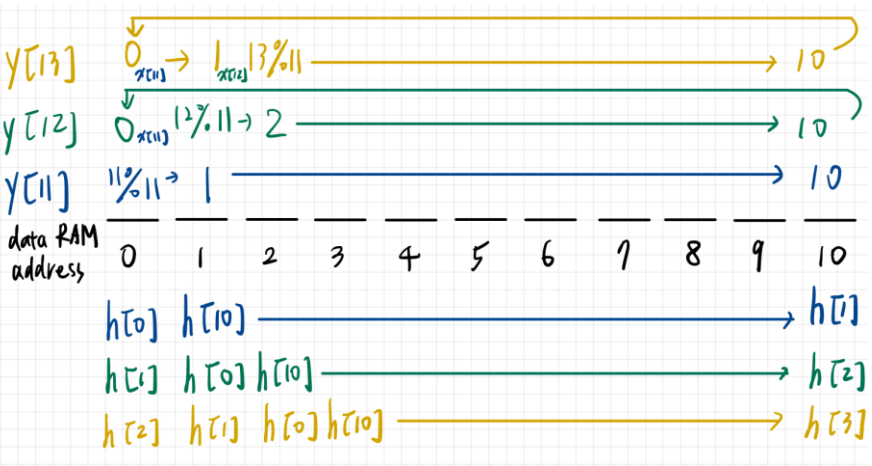


Data stream in



2. Describe operation
   - 對於 tap parameters & data-in 的接收，只要看到 testbench 準備好傳輸資料的訊號後，可以透過 FIR engine 控制信號 awready, wready & ss_ready 來決定是否要接收這些 data，一旦決定接收就代表 RAM 裡面有位置可以存放，所以當信號 awready, wready & ss_read 拉為一時，便將送過來的 address & parameter or data 直接接上對應的 RAM 的 input signal 即可。
   - 在計算上面，主要是需要對 tap & data RAM 去做 controll，對於 tap RAM 裡面的 11 個 coefficients，控制 address 來讀取正確的值即可，但對 data RAM 來說，因為最多只能放置 11 筆資料，而我們需要處理 600 筆 data，所以除了讀值之外，計算過程中還需要跟 testbech 做 communication，使用 axi-stream 接收新的數值並寫入 data RAM。將算式展開後，分別對 y[0]~y[10] & y[11]~y[599]得到以下兩張圖的規律，首先 accumulation counter 的作用為每筆 output 需要相加幾組數值才會完成計算，並順便將其當作前面十一筆 data 從 data RAM 取值的 index 變化。對於 y[11]之後每次讀進來的 data 都將其寫入 RAM 的同時也送進乘法器裡面做運算，所以 coefficient 都是從 idex (address) 0 先取值，接著便從 10 開始往下遞減一，而 data 的 idex 單純把取餘數後的結果每次遞增一





- data_length 會存在 FIR engine 內的 register，ap_start, ap_done, ap_idle 則分別存在 FIR ap control register 的第 0,1,2 個 bit。當 ap_start 要從 testbench 寫入 1 時，確定 ap_idle is 1 後，會在下一個 clock 存入 ap control register 的第 0 個 bit，接著便進入 EXEUTION state，由於 ap_start is sampled，所以將 ap_idle 拉為 0，接著將 ap_start reset 為 0。ap_idle 下一次的變化出現在運算完最後一筆 output 後就可以拉為 1，而 ap_done 則要確認最後一筆 data stream-out 給 testbench 後 (亦即 sm_tready==1 && sm_tvalid==1 && output counter == data_length) 才可拉為 1。

3. Resource usage

LUT	FF	BRAM	URAM	DSP
301	111	0	0	3

4. Timing report
   Maximum frequency

```
Clock     Waveform(ns)       Period(ns)      Frequency(MHz)
-----     ------------       ----------      --------------
axis_clk  {0.000 5.000}      10.400          96.154
```

Setup		Hold		Pulse Width	
Worst Negative Slack (WNS):	0.334 ns	Worst Hold Slack (WHS):	0.072 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:	207	Total Number of Endpoints:	205	Total Number of Endpoints:	112

All user specified timing constraints are met.

Longest path & its slack

```
Slack (MET) :             0.334ns  (required time - arrival time)
  Source:                 data_out_cnt_reg[4]/C
                            (rising edge-triggered cell FDRE clocked by axis_clk  {rise@0.000ns fall@5.000ns period=10.400ns})
  Destination:            tap_A[2]
                            (output port clocked by axis_clk  {rise@0.000ns fall@5.000ns period=10.400ns})
  Path Group:             axis_clk
  Path Type:              Max at Slow Process Corner
  Requirement:            10.400ns  (axis_clk rise@10.400ns - axis_clk rise@0.000ns)
  Data Path Delay:        7.074ns  (logic 3.771ns (53.313%)  route 3.303ns (46.687%))
  Logic Levels:           5  (LUT4=1 LUT5=1 LUT6=2 OBUF=1)
  Output Delay:           0.500ns
  Clock Path Skew:        -2.456ns (DCD - SCD + CPR)
    Destination Clock Delay (DCD):    0.000ns = ( 10.400 - 10.400 )
    Source Clock Delay      (SCD):    2.456ns
    Clock Pessimism Removal (CPR):    0.000ns
  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
```

```
Slack (MET) :             0.334ns  (required time - arrival time)
  Source:                 data_out_cnt_reg[4]/C
                            (rising edge-triggered cell FDRE clocked by axis_clk  {rise@0.000ns fall@5.000ns period=10.400ns})
  Destination:            tap_A[3]
                            (output port clocked by axis_clk  {rise@0.000ns fall@5.000ns period=10.400ns})
  Path Group:             axis_clk
  Path Type:              Max at Slow Process Corner
  Requirement:            10.400ns  (axis_clk rise@10.400ns - axis_clk rise@0.000ns)
  Data Path Delay:        7.074ns  (logic 3.771ns (53.313%)  route 3.303ns (46.687%))
  Logic Levels:           5  (LUT4=1 LUT5=1 LUT6=2 OBUF=1)
  Output Delay:           0.500ns
  Clock Path Skew:        -2.456ns (DCD - SCD + CPR)
    Destination Clock Delay (DCD):    0.000ns = ( 10.400 - 10.400 )
    Source Clock Delay      (SCD):    2.456ns
    Clock Pessimism Removal (CPR):    0.000ns
  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
```

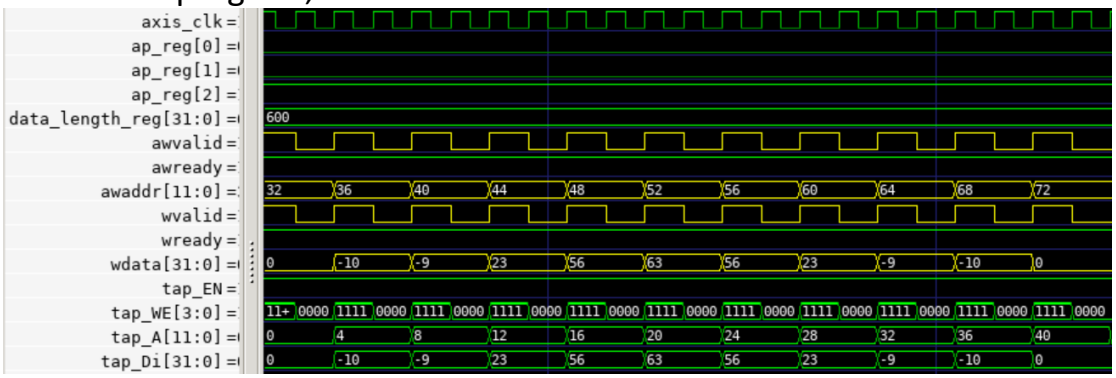
5. Simulation waveform

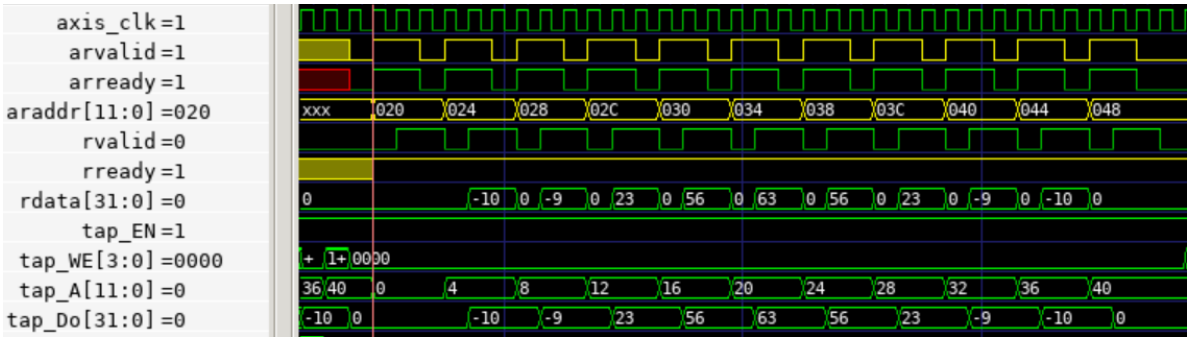
ap_reg[0] corresponds to ap_start

ap_reg[1] corresponds to ap_done

ap_reg[2] corresponds to ap_idle

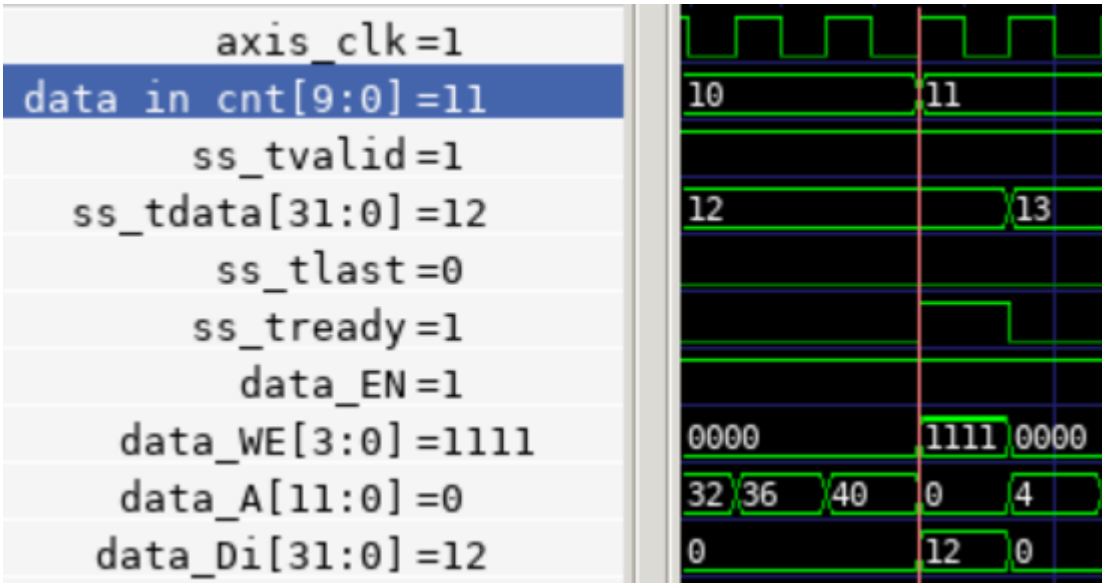
- Coefficient program, and read back



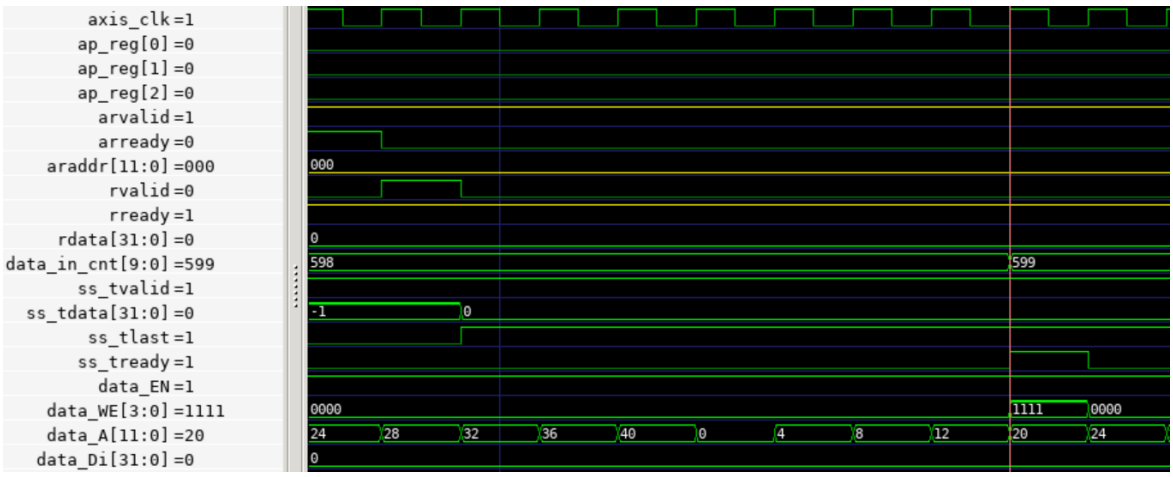


- Data-in stream-in

The 12th data stream-in

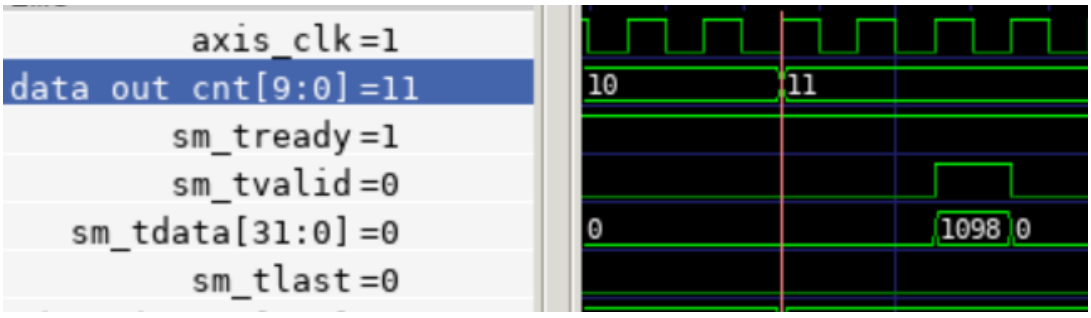


The last data stream-in

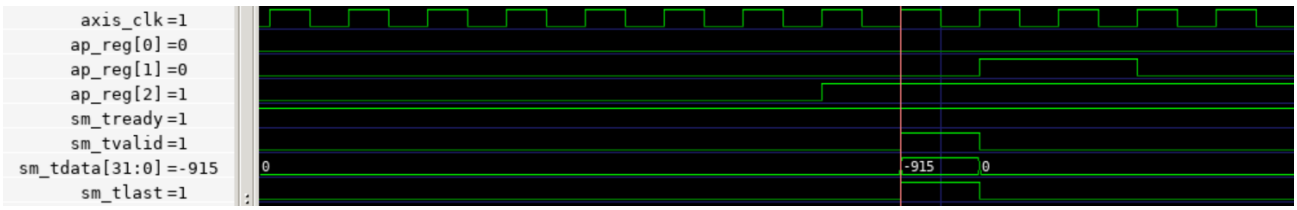


- Data-out stream-out

The 12th data stream

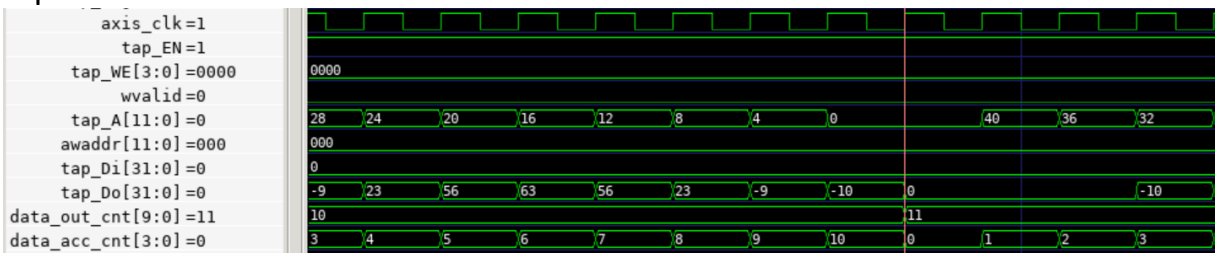


The last data stream-out

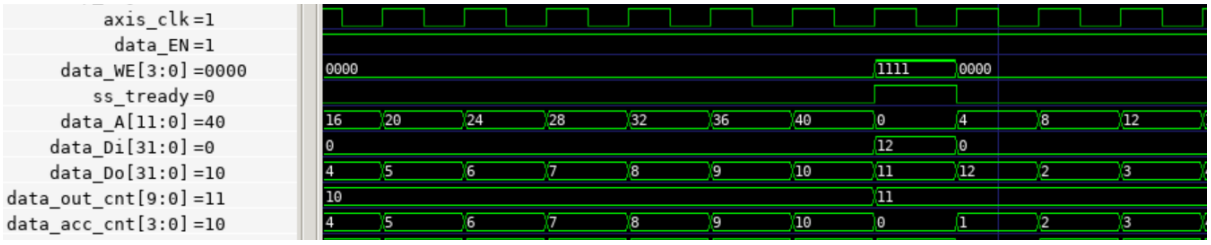


- RAM access control

  tap RAM:



  data RAM:



- FSM

  There are only two states: SETUP and EXECUTION. The state transitions occur when the FIR engine samples the ap_start signal