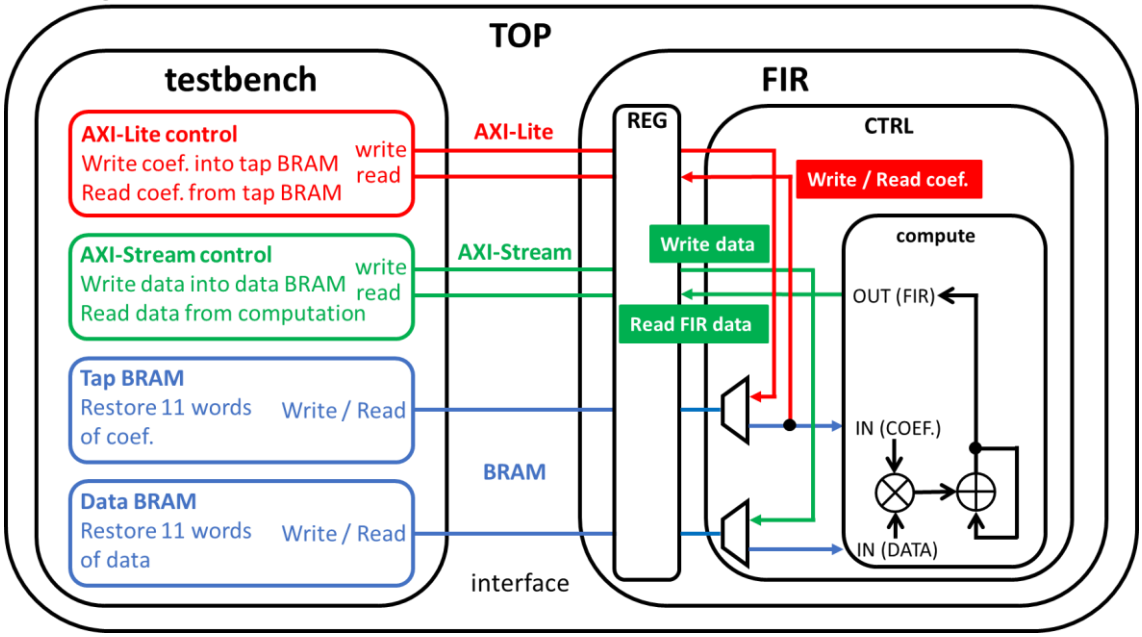
Lab 3: FIR design with AXI-lite, AXI-stream interface and BRAM

I. Abstract / Introduction

We implement FIR design in Verilog with different protocol interface to transfer control signal, coefficient, and data. Then check the function correction with testbench and run synthesis in Vivado. Divide the protocol into three parts: AXI-lite, AXI-stream, and BRAM interface. AXI-lite protocol transfers control signal and the coefficient, AXI-stream protocol transfers the data input and output, two BRAM memory restore 11 pair coefficient and data which generates one output value after FIR computation.

II. Block Diagram



III. Operations description

1. Finite state machine

The total computation flow can divide into four stages as FSM shown below.

A. IDLE stage

In this stage, we load coefficient from AXI-lite protocol and restore them into tap BRAM. After that, we read coefficient from tap BRAM to testbench for checking whether writing coefficient into correct address. The last, we would receive ap_start signal and then jump to LOAD stage.

B. LOAD stage

In each FIR computation, we need to load one new data and remove the oldest one. Thus, we receive the latest data from AXI-stream protocol and write it into data BRAM to replace the oldest one. After this operation, we directly jump to COMPUTE stage.

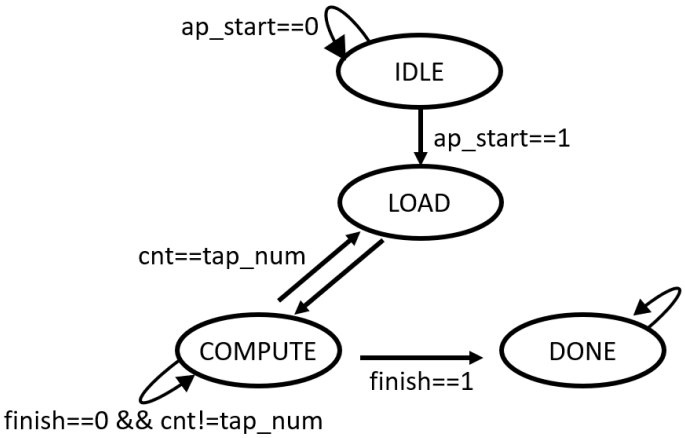
C. COMPUTE stage

In this stage, we spend 11 cycles to compute one FIR output. In each cycle, we load one coefficient from tap BRAM and one data from data BRAM. The order of data from data BRAM might be different for each FIR

output, and thus we control the order directly by control address. The details about controller would be shown later. After computation, if this output is the last one, we jump to DONE stage and finish the total FIR computation. Otherwise, we jump to LOAD stage to load the next new data and then compute again.

D. DONE stage

This stage means the design finish the total FIR computation and we determine whether finishing the total FIR computation or not depending on the signal ss_tlast which means this data input is the last one. And then, we assert ap_done and ap_idle after entering this stage.



2. Data flow operation

A. Receive tap parameter and write into tap BRAM

After AXI-lite protocol write handshake successfully, we would receive one tap parameter and then write into tap BRAM by control the address and assert tap_EN and tap_WE. Since the order of coefficient should be fixed and won't change when we do computation, thus we write directly into tap BRAM following the received order.

B. Receive data parameter and write into data BRAM

After AXI-Stream protocol write handshake successfully, we would receive one data parameter and then write into data BRAM by control address and assert data_EN and data_WE. However, in data BRAM, we should restore the 11 data which need in the next FIR computation. Each loading time, we should load the next data into data BRAM and remove the oldest one if the RAM is full. Otherwise, we write directly into data BRAM following the received order. Hence, we have flag signal to record which address should be written next.

C. Access tap BRAM and data BRAM to do computation

We need 11 cycles to finish one FIR computation, and then we design a counter to know whether we finish computation this time or not. For coefficient, the order won't be change and then we read directly following the counter value as tap BRAM address. But this way wouldn't be work for data BRAM. We should read the oldest data in data BRAM as the first data and so on, so we also use flag signal to know which data we should read out

in the first cycle and the counter would be used in the next cycles. The flow is as shown below.

flag	0	0	0	0	0	0	0	0	0	0	0	1	2	3	4	5	6	7	…
cycle iteration	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	
ram addr																			
0	d0	d0	d0	d0	d0	d0	d0	d0	d0	d0	d0	d11	d11	d11	d11	d11	d11	d11	
1	0	d1	d1	d1	d1	d1	d1	d1	d1	d1	d1	d1	d12	d12	d12	d12	d12	d12	
2	0	0	d2	d2	d2	d2	d2	d2	d2	d2	d2	d2	d2	d13	d13	d13	d13	d13	
3	0	0	0	d3	d3	d3	d3	d3	d3	d3	d3	d3	d3	d3	d14	d14	d14	d14	
4	0	0	0	0	d4	d4	d4	d4	d4	d4	d4	d4	d4	d4	d4	d15	d15	d15	
5	0	0	0	0	0	d5	d5	d5	d5	d5	d5	d5	d5	d5	d5	d5	d16	d16	
6	0	0	0	0	0	0	d6	d6	d6	d6	d6	d6	d6	d6	d6	d6	d6	d17	
7	0	0	0	0	0	0	0	d7	d7	d7	d7	d7	d7	d7	d7	d7	d7	d7	
8	0	0	0	0	0	0	0	0	d8	d8	d8	d8	d8	d8	d8	d8	d8	d8	
9	0	0	0	0	0	0	0	0	0	d9	d9	d9	d9	d9	d9	d9	d9	d9	
10	0	0	0	0	0	0	0	0	0	0	d10	d10	d10	d10	d10	d10	d10	d10	
	*cycle iteration: finish one FIR computing means one time iteration, in this case should be 11 cycle																		
	red mark means the newest data																		

3. Control signal operation

The ap_signal can be divided into three parts, ap_start, ap_done, ap_idle.

A. ap_start

After reset, ap_idle would be low. When AXI-lite write handshake successfully, write address is 0x00, and write data is 1, the ap_start would have one cycle pulse to start the FIR engine.

B. ap_done

After reset, ap_done would be low. When the current state is DONE stage which means finishing the total FIR computation, ap_done would be assert to high and keep on.

C. ap_idle

After reset, ap_idle would be high. If ap_start assert to high, then ap_idle should be low. When the current state is DONE stage which means finishing the total FIR computation, ap_idle would be assert to high and keep on.

IV. Resource usage

Site Type	Used	Fixed	Prohibited	Available	Util%
Slice LUTs*	249	0	0	53200	0.47
LUT as Logic	249	0	0	53200	0.47
LUT as Memory	0	0	0	17400	0.00
Slice Registers	380	0	0	106400	0.36
Register as Flip Flop	380	0	0	106400	0.36
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

V. Timing report

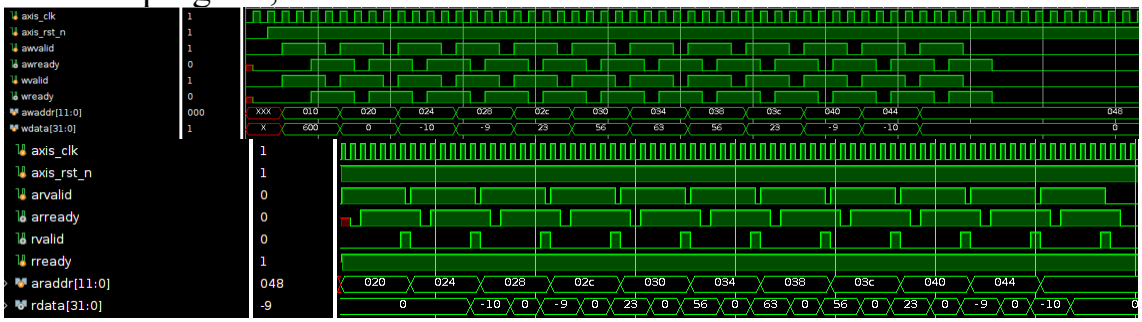
Synthesis clock period: 10ns (100MHz)

Critical Path: multiplication and addition when computing temp FIR result

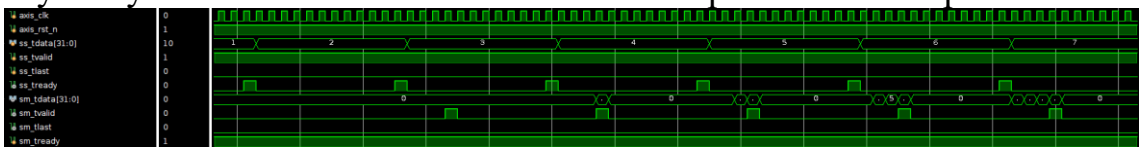
```
Max Delay Paths
--------------------------------------------------------------------------------------
Slack (MET) :             0.565ns  (required time - arrival time)
  Source:                 temp_sum__0/CLK
                            (rising edge-triggered cell DSP48E1 clocked by axis_clk  {rise@0.000ns fall@5.000ns period=10.000ns})
  Destination:            prev_sum_reg[29]/D
                            (rising edge-triggered cell FDRE 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:        9.331ns  (logic 7.849ns (84.120%)  route 1.482ns (15.880%))
  Logic Levels:           8  (CARRY4=5 DSP48E1=1 LUT2=2)
  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=383, unplaced)       0.584     2.456    axis_clk_IBUF_BUFG
                         DSP48E1                                     r  temp_sum__0/CLK
  -------------------------------------------------------------------    -------------------
                         DSP48E1 (Prop_dsp48e1_CLK_PCOUT[47])
                                                      4.206     6.662 r  temp_sum__0/PCOUT[47]
                         net (fo=1, unplaced)         0.055     6.717    temp_sum__0_n_106
                                                                     r  temp_sum__1/PCIN[47]
                         DSP48E1 (Prop_dsp48e1_PCIN[47]_P[0])
                                                      1.518     8.235 r  temp_sum__1/P[0]
                         net (fo=2, unplaced)         0.800     9.035    temp_sum__1_n_105
                                                                     r  prev_sum[19]_i_9/I0
                         LUT2 (Prop_lut2_I0_O)        0.124     9.159 r  prev_sum[19]_i_9/O
                         net (fo=1, unplaced)         0.000     9.159    prev_sum[19]_i_9_n_0
                                                                     r  prev_sum_reg[19]_i_6/S[1]
                         CARRY4 (Prop_carry4_S[1]_CO[3])
                                                      0.533     9.692 r  prev_sum_reg[19]_i_6/CO[3]
                         net (fo=1, unplaced)         0.009     9.701    prev_sum_reg[19]_i_6_n_0
                                                                     r  prev_sum_reg[23]_i_6/CI
                         CARRY4 (Prop_carry4_CI_CO[3])
                                                      0.117     9.818 r  prev_sum_reg[23]_i_6/CO[3]
                         net (fo=1, unplaced)         0.000     9.818    prev_sum_reg[23]_i_6_n_0
                                                                     r  prev_sum_reg[27]_i_6/CI
                         CARRY4 (Prop_carry4_CI_O[3])
                                                      0.331    10.149 r  prev_sum_reg[27]_i_6/O[3]
                         net (fo=1, unplaced)         0.618    10.767    temp_sum__2[27]
                                                                     r  prev_sum[27]_i_2/I1
                         LUT2 (Prop_lut2_I1_O)        0.307    11.074 r  prev_sum[27]_i_2/O
                         net (fo=1, unplaced)         0.000    11.074    prev_sum[27]_i_2_n_0
                                                                     r  prev_sum_reg[27]_i_1/S[3]
                         CARRY4 (Prop_carry4_S[3]_CO[3])
                                                      0.376    11.450 r  prev_sum_reg[27]_i_1/CO[3]
                         net (fo=1, unplaced)         0.000    11.450    prev_sum_reg[27]_i_1_n_0
                                                                     r  prev_sum_reg[31]_i_2/CI
                         CARRY4 (Prop_carry4_CI_O[1])
                                                       0.337    11.787 r  prev_sum_reg[31]_i_2/O[1]
                         net (fo=1, unplaced)          0.000    11.787    cur_sum[29]
                         FDRE                                        r  prev_sum_reg[29]/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=383, unplaced)       0.439    12.128    axis_clk_IBUF_BUFG
                         FDRE                                        r  prev_sum_reg[29]/C
                         clock pessimism              0.184    12.311
                         clock uncertainty           -0.035    12.276
                         FDRE (Setup_fdre_C_D)        0.076    12.352    prev_sum_reg[29]
  -------------------------------------------------------------------
                         required time                         12.352
                         arrival time                         -11.787
  -------------------------------------------------------------------
                         slack                                  0.565
```

VI. Simulation waveform

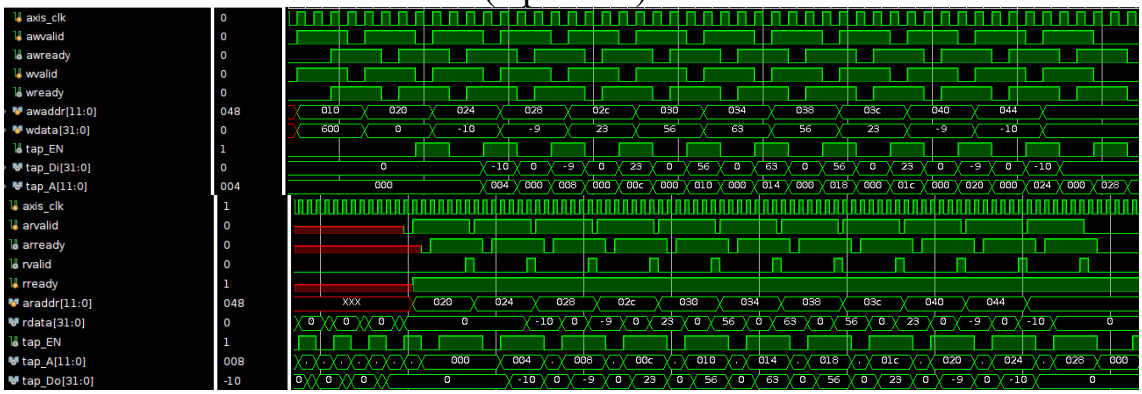
1. Coefficient program, and read back



2. Data-in stream-in / Data-out stream-out
   Every 11 cycles can receive one new data and output one FIR output value



3. RAM access control
   A. write coefficient and read back (tap RAM)



   B. Load data and compute (tap RAM and data RAM)