FIR Filter Design

2023-fall W4823 Project Report

Tong Cao
Department of Electrical Engineering
Columbia University
New York, U.S.

Yuqi Lin
Department of Electrical Engineering,
Columbia University
New York, U.S.

***Abstract*—This is a 64-tap 16-bit FIR filter.**

***Keywords*—FIR, Verilog**

I. Introduction (*Heading 1*)

We know FIR computation is actually discrete convolution：

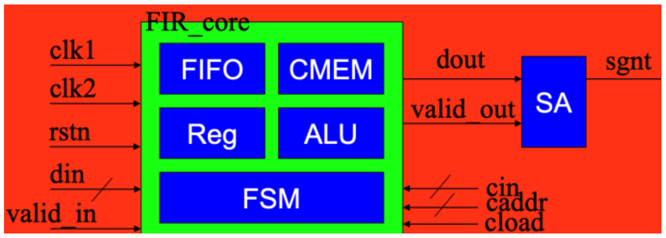
$$y[n]=\sum_{i=0}^{N} bi \cdot x[n-i].$$

x[n] is the input signal, y[n] is the output signal, N is the filter order (number of taps), bi is the filter coefficient. b and x are both 16bit fixed-point, y is 16bit half-decision floating point.

II. Signals Defination

A. Clock

clk1 is set as 10kHz, and the clk2 we choose is 10 times faster. Because we have asynchronous FIFO, clk2 can be set much higher than clk1.



B. Control

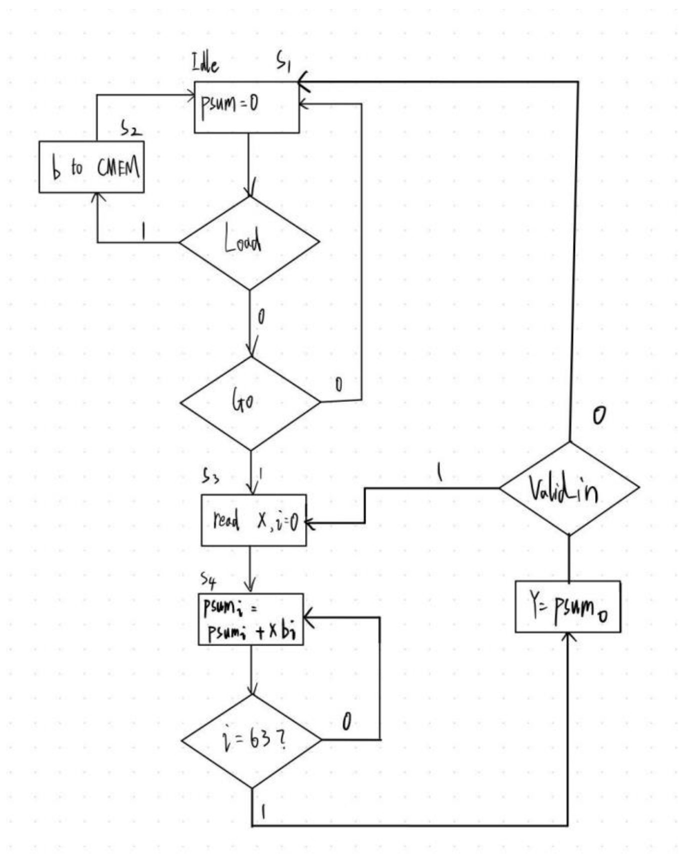
When rstn=0, reset the chip. cin, caddr, cload controls the process of preload, 64 coefficient (b) is loaded into CMEM consecutively following the address when cload is 1. n is input data (x) from outside, and when valid_in=1, the chip starts to compute. dout is output data (y).

III. Architecture

Our chip is working on clk2 frequency, while input of FIFO is sent within clk1. That means the FIFO is asynchronous. First, preload b into CMEM, and pass these coefficient to on-chip register. Every very cycle, we receive a x from FIFO, grab every b store in b register, and then multiply with x, we called these product psum[i] and restore them in shift-register.

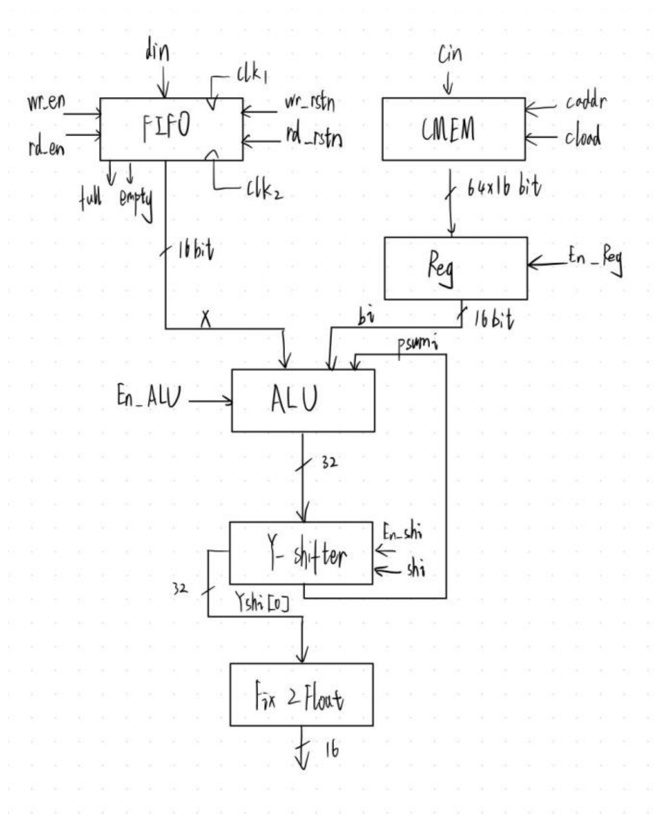
Every results of ALU, plus the former correspond result in shifter is the new psum[i]. Meanwhile, left-shift the psum register, and the output is connected to the psum[0] which is pushed out.

ASM:



Datapath:

Identify applicable funding agency here. If none, delete this text box.



IV. Blocks

A. FIFO

Because of the duel FIFO clocks, we used two pointer to indicate the position of read and write operation. The FIFO's input is controlled by the clk1 and the output is controlled by the clk2, which makes it the asynchronous FIFO. The read and write are controlled by the read_ptr and write_ptr. Using the pointer to indicate the place is easy for us to insert the data or read the data. One of the beneficiaries in using pointer is that they can help us to judge if the FIFO is empty or full. The code is in fifo.v. If rd_ptr==wr_ptr, then it's empty. If wr_ptr's MSB is different with rd_ptr, while the other bits are the same, then it's full. Consider of metastable state, convert the pointer to gray code and use two stage comparison synchronizer.

B. CMEM

Generated from Memory Compiler. Sized 64x16bits. we mainly need the cmem to store the coefficients. Imem can work as the memory for the input data. In this project, we mainly use the memory compiler to generate the 64-words, 16 bits sram verilog and library file. The first problem in this project is that we need to write a RTL code (sram.v) to initialize the memory we generated using the memory compiler. Then, we write the testbench to read the coefficients file can store them in the sram array in the write mode. When writing the testbench, we need to be aware of the setup time and hold time. And then we read the data in the sram and compare them with the original data. The error stores the comparison result.

C. ALU

Our ALU has three inputs: x_local, b_local, y_former (from psum shifter), it computes x_local*b_local+y_former and send to shift register of psum.

That means, every clock cycle, we can get an output because the number of psum is different in first 64 outputs. The psum which is needed for later output is restored and fetched by ALU every cycle too.

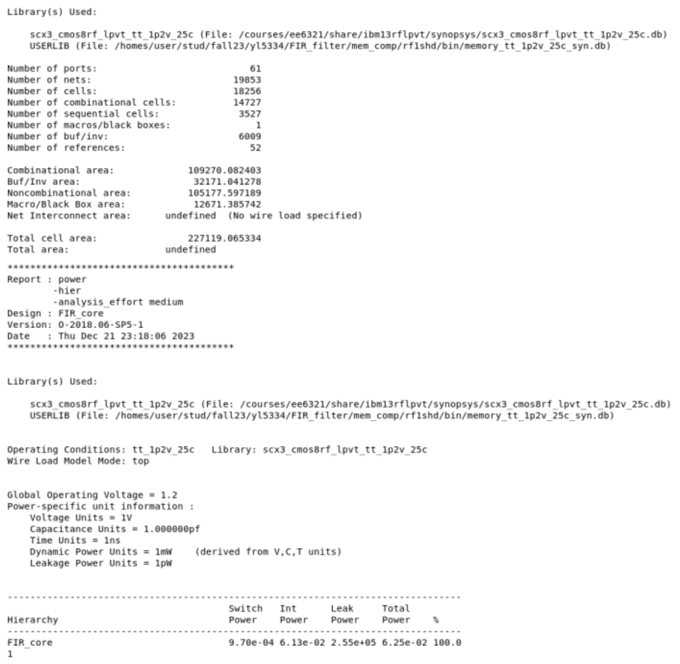
D. FSM

We have 5 states in this design. S1 stands for idle, S2 is loading, S3 is reading coefficient from FIFO, S4 is ALU computing, S5 is shifting and output.

E. Fix to Floating Point Coverting

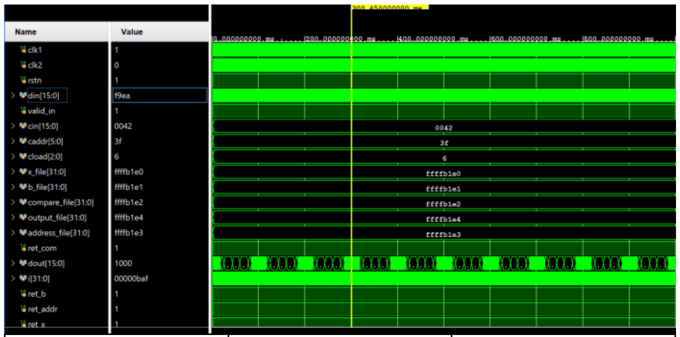
We finally need to generate half precision floating point result from fixpoint out of shifter. And in the testbench, the test data file we read, which is first generated by FIR definition function in Matlab and converted to fixed or floating point we need by Python code.

V. Results



```
Library(s) Used:

    scx3_cmos8rf_lpvt_tt_1p2v_25c (File: /courses/ee6321/share/ibm13rflpvt/synopsys/scx3_cmos8rf_lpvt_tt_1p2v_25c.db)
    USERLIB (File: /homes/user/stud/fall23/yl5334/FIR_filter/mem_comp/rf1shd/bin/memory_tt_1p2v_25c_syn.db)

Number of ports:                           61
Number of nets:                         19853
Number of cells:                        18256
Number of combinational cells:          14727
Number of sequential cells:              3527
Number of macros/black boxes:               1
Number of buf/inv:                       6009
Number of references:                      52

Combinational area:             109270.082403
Buf/Inv area:                    32171.041278
Noncombinational area:          105177.597189
Macro/Black Box area:            12671.385742
Net Interconnect area:      undefined  (No wire load specified)

Total cell area:                227119.065334
Total area:                 undefined
****************************************
Report : power
        -hier
        -analysis_effort medium
Design : FIR_core
Version: O-2018.06-SP5-1
Date   : Thu Dec 21 23:18:06 2023
****************************************


Library(s) Used:

    scx3_cmos8rf_lpvt_tt_1p2v_25c (File: /courses/ee6321/share/ibm13rflpvt/synopsys/scx3_cmos8rf_lpvt_tt_1p2v_25c.db)
    USERLIB (File: /homes/user/stud/fall23/yl5334/FIR_filter/mem_comp/rf1shd/bin/memory_tt_1p2v_25c_syn.db)


Operating Conditions: tt_1p2v_25c   Library: scx3_cmos8rf_lpvt_tt_1p2v_25c
Wire Load Model Mode: top


Global Operating Voltage = 1.2
Power-specific unit information :
    Voltage Units = 1V
    Capacitance Units = 1.000000pf
    Time Units = 1ns
    Dynamic Power Units = 1mW    (derived from V,C,T units)
    Leakage Power Units = 1pW


--------------------------------------------------------------------------------
                                       Switch   Int      Leak     Total
Hierarchy                              Power    Power    Power    Power    %
--------------------------------------------------------------------------------
FIR_core                               9.70e-04 6.13e-02 2.55e+05 6.25e-02 100.0
1
```

```
Startpoint: b_local_reg_2_
            (rising edge-triggered flip-flop clocked by clk2)
Endpoint: adder_multipe_ALU_out_t_reg_31_
          (rising edge-triggered flip-flop clocked by clk2)
Path Group: clk2
Path Type: max

Point                                                   Incr       Path
---------------------------------------------------------------------------
clock clk2 (rise edge)                                  0.00       0.00
clock network delay (ideal)                             0.00       0.00
b_local_reg_2_/CK (DFFRXLTS)                            0.00 #     0.00 r
b_local_reg_2_/QN (DFFRXLTS)                            1.00       1.00 r
U6790/Y (INVX2TS)                                       0.20       1.19 f
U14432/Y (OAI22X1TS)                                    0.20       1.40 r
U14433/Y (AOI21X1TS)                                    0.13       1.53 f
U14434/Y (NOR2XLTS)                                     0.32       1.85 r
U14670/Y (AOI21X1TS)                                    0.20       2.05 f
U14675/S (CMPR32X2TS)                                   0.70       2.75 f
intadd_0_U29/CO (CMPR32X2TS)                            0.68       3.43 f
intadd_0_U28/CO (CMPR32X2TS)                            0.47       3.91 f
intadd_0_U27/CO (CMPR32X2TS)                            0.47       4.38 f
intadd_0_U26/CO (CMPR32X2TS)                            0.47       4.85 f
intadd_0_U25/CO (CMPR32X2TS)                            0.47       5.33 f
intadd_0_U24/CO (CMPR32X2TS)                            0.47       5.80 f
intadd_0_U23/CO (CMPR32X2TS)                            0.47       6.27 f
intadd_0_U22/CO (CMPR32X2TS)                            0.47       6.75 f
intadd_0_U21/CO (CMPR32X2TS)                            0.47       7.22 f
intadd_0_U20/CO (CMPR32X2TS)                            0.47       7.69 f
intadd_0_U19/CO (CMPR32X2TS)                            0.47       8.16 f
intadd_0_U18/CO (CMPR32X2TS)                            0.47       8.64 f
intadd_0_U17/CO (CMPR32X2TS)                            0.47       9.11 f
intadd_0_U16/CO (CMPR32X2TS)                            0.47       9.58 f
intadd_0_U15/CO (CMPR32X2TS)                            0.47      10.06 f
intadd_0_U14/CO (CMPR32X2TS)                            0.47      10.53 f
intadd_0_U13/CO (CMPR32X2TS)                            0.47      11.00 f
intadd_0_U12/CO (CMPR32X2TS)                            0.47      11.48 f
intadd_0_U11/CO (CMPR32X2TS)                            0.47      11.95 f
intadd_0_U10/CO (CMPR32X2TS)                            0.47      12.42 f
intadd_0_U9/CO (CMPR32X2TS)                             0.47      12.89 f
intadd_0_U8/CO (CMPR32X2TS)                             0.47      13.36 f
intadd_0_U7/CO (CMPR32X2TS)                             0.47      13.84 f
intadd_0_U6/CO (CMPR32X2TS)                             0.47      14.31 f
intadd_0_U5/CO (CMPR32X2TS)                             0.47      14.78 f
intadd_0_U4/CO (CMPR32X2TS)                             0.47      15.26 f
intadd_0_U3/CO (CMPR32X2TS)                             0.47      15.73 f
intadd_0_U2/CO (CMPR32X2TS)                             0.46      16.19 f
U14834/Y (XOR2XLTS)                                     0.21      16.40 f
adder_multipe_ALU_out_t_reg_31_/D (EDFFX1TS)            0.00      16.40 f
data arrival time                                                 16.40

clock clk2 (rise edge)                               1000.00    1000.00
clock network delay (ideal)                             0.00    1000.00
adder_multipe_ALU_out_t_reg_31_/CK (EDFFX1TS)           0.00    1000.00 r
library setup time                                     -0.77     999.23
data required time                                               999.23
---------------------------------------------------------------------------
data required time                                               999.23
data arrival time                                                -16.40
---------------------------------------------------------------------------
slack (MET)                                                      982.83
```



Name	Value
clk1	1
clk2	0
rstn	1
din[15:0]	f9ea
valid_in	1
cin[15:0]	0042
caddr[5:0]	3f
cload[2:0]	6
x_file[31:0]	ffffb1e0
b_file[31:0]	ffffb1e1
compare_file[31:0]	ffffb1e2
output_file[31:0]	ffffb1e4
address_file[31:0]	ffffb1e3
ret_com	1
dout[15:0]	1000
i[31:0]	00000baf
ret_b	1
ret_addr	1
ret_x	1

Throughput:	10	[kS/s]
Maximum clock frequency	0.1	[MHz]
Power	5.017e-9	pJ
Area:	0.227119	[mm2]
Accuracy	TBD	%