Finite Impulse Response Filter VLSI Design

Group: YYYY
Members: Yixiao Du, Yuxiang Long, Yibang Xiao, Yifan Yang

ECE 4740 Course Project

5/20/2020

CONTENTS

I. Literature Review

II. Architecture

III. Implementation

IV. Verification and Evaluation

CONTENTS

I. Literature Review

Adder: CSA, BEC (Binary Excess Converter)
Multiplier: Wallace Tree, CSM (Computation Sharing Memory)

CONTENTS

II. Architecture

Figure of Merit

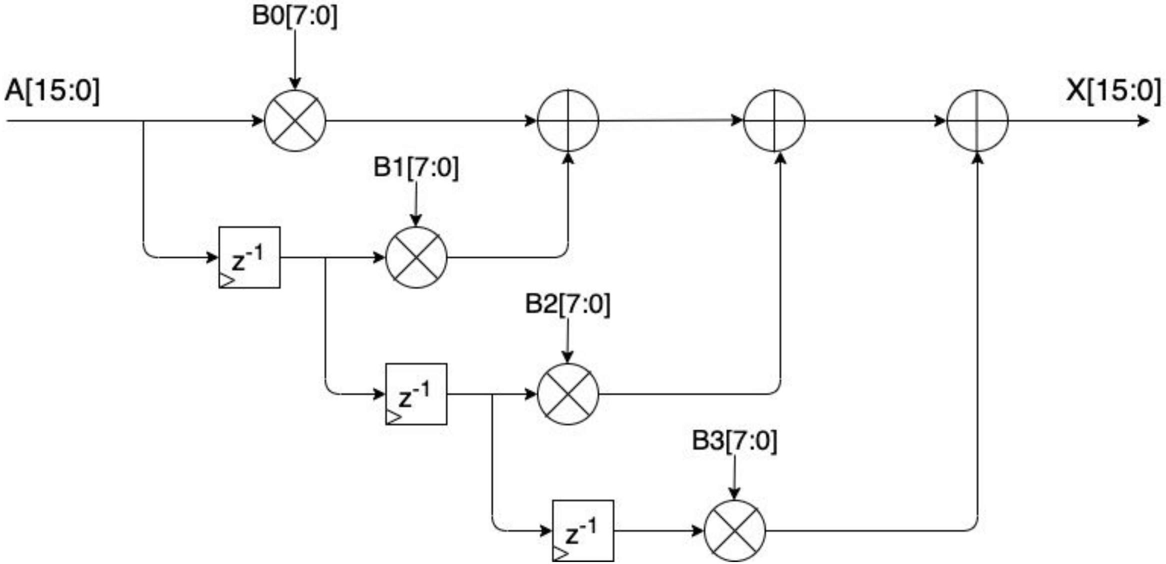
1. Sampling frequency
2. Precision
3. Power and Area

Third-order FIR System Diagram

X[n] = b0 * A[n] + b1 * A[n-1] + b2 * A[n-2] + b3 * A[n-3]

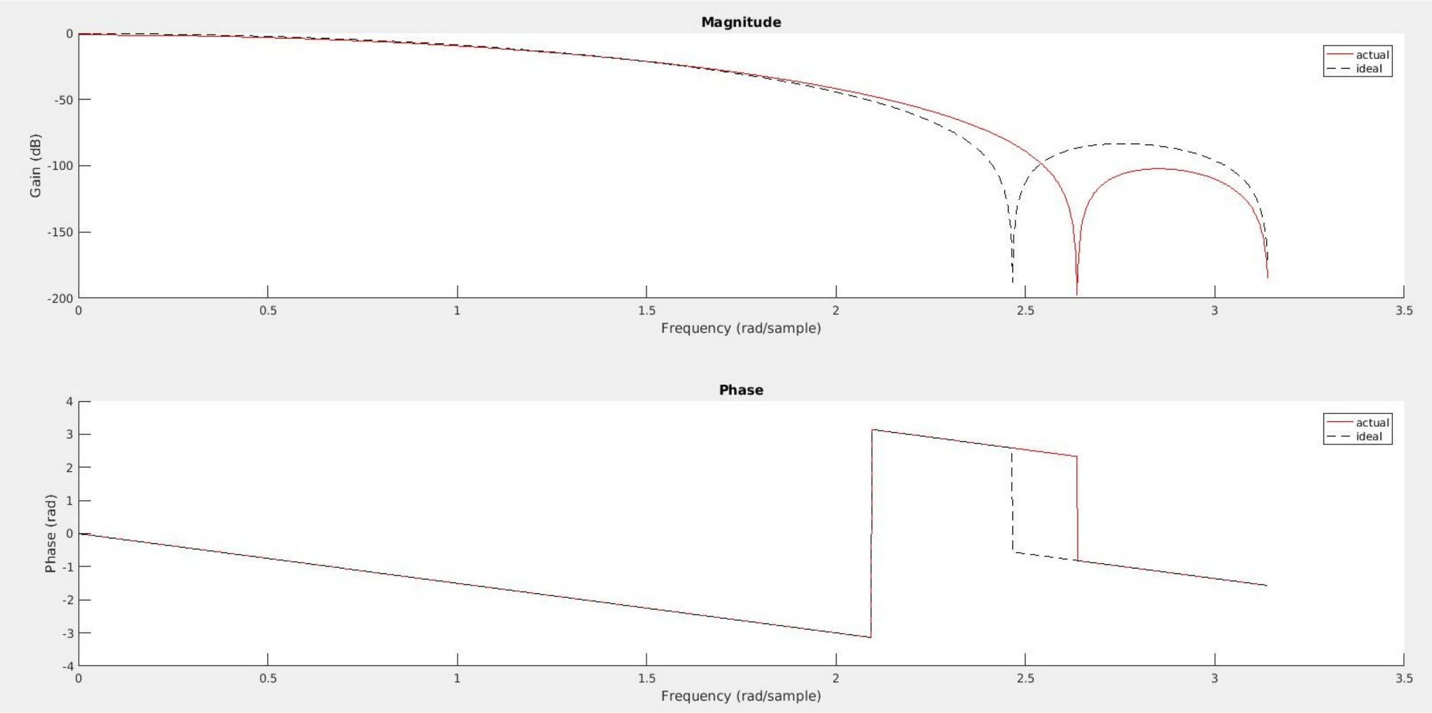
$H(z) = b0 + b1z^{-1} + b2z^{-2} + b3z^{-3}$

Sample Rate : 100kHz



FIR System Response

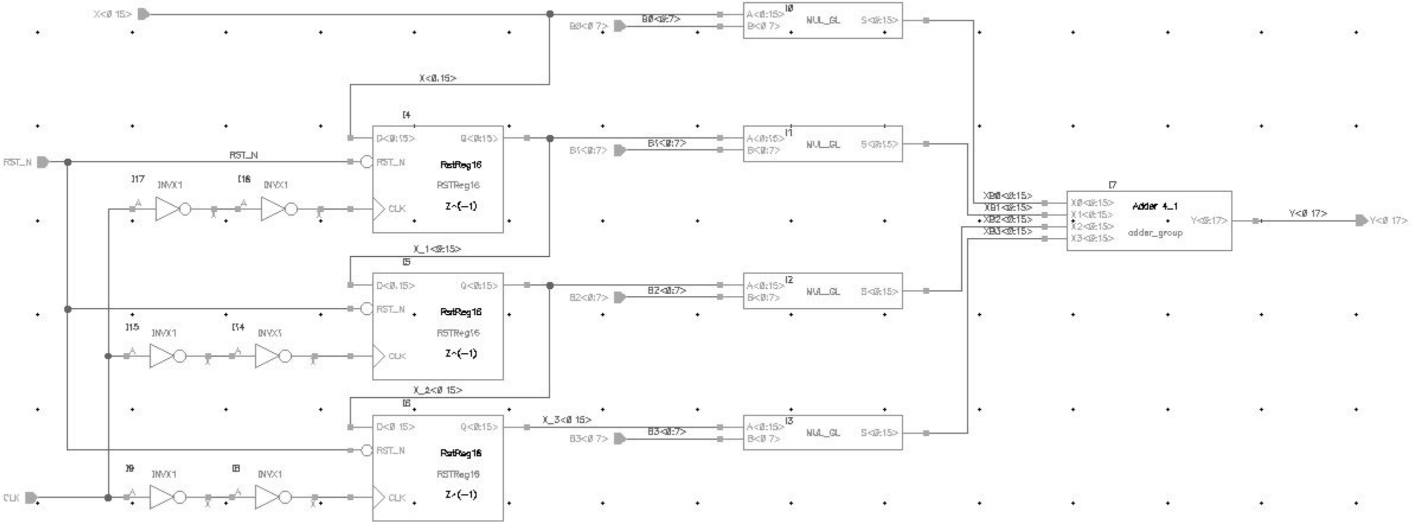
b0 =
0.138336181640625
b1 =
0.354217529296875
b2 =
0.354217529296875
b3 =
0.138336181640625



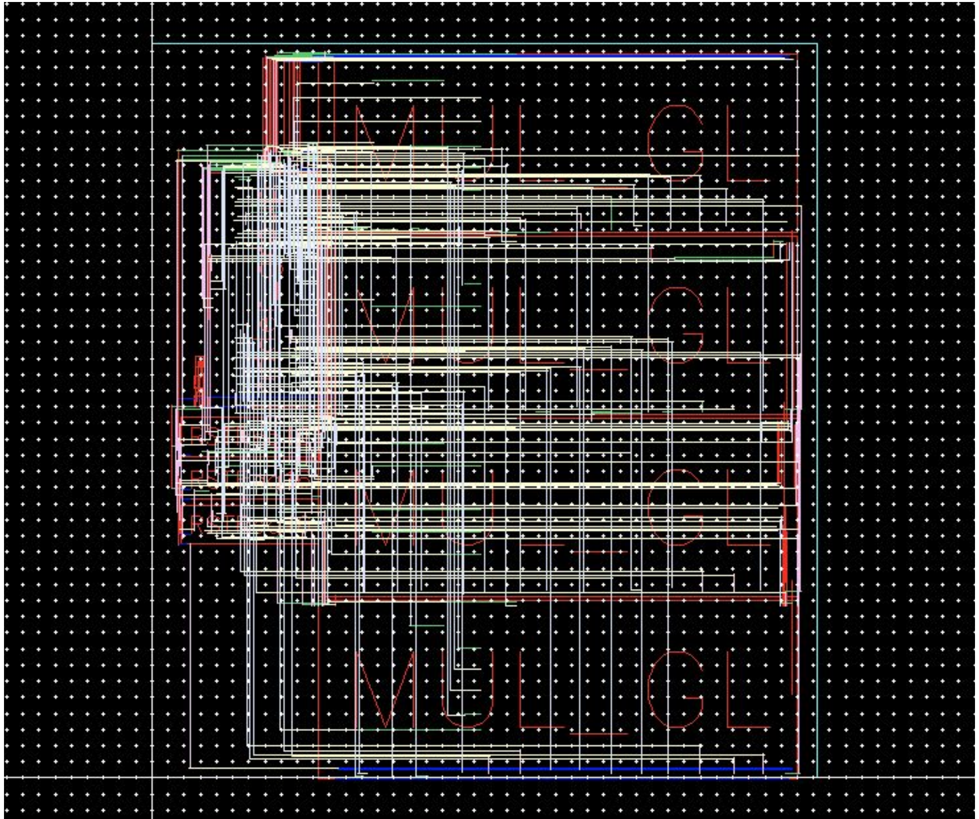
CONTENTS

III. Implementation

Top level


X<Ø 15>
B0<Ø 7>
BØ<Ø:7>
A<Ø:15>
B<Ø 7>
IØ
MUL_GL
S<Ø:15>
X<Ø.15>
I4
D<Ø:15>
Q<Ø:15>
RST_N
RST_N
RstReg16
RSTReg16
CLK
Z^(−1)
I17
INVX1
I16
INVX1
A
Y
B1<Ø:7>
B1<Ø:7>
I1
X_1<Ø:15>
I5
D<Ø.15>
Q<Ø:15>
I15
INVX1
I14
INVX1
B2<Ø:7>
B2<Ø:7>
I2
X_2<Ø 15>
I6
D<Ø 15>
Q<Ø:15>
X_3<Ø 15>
B3<Ø 7>
B3<Ø:7>
I3
I9
INVX1
I8
INVX1
CLK
I7
XBØ<Ø:15>
XB1<Ø:15>
XB2<Ø:15>
XB3<Ø:15>
XØ<Ø:15>
X1<Ø:15>
X2<Ø:15>
X3<Ø:15>
Adder 4_1
adder_group
Y<Ø:17>
Y<Ø 17>
Y<Ø 17>

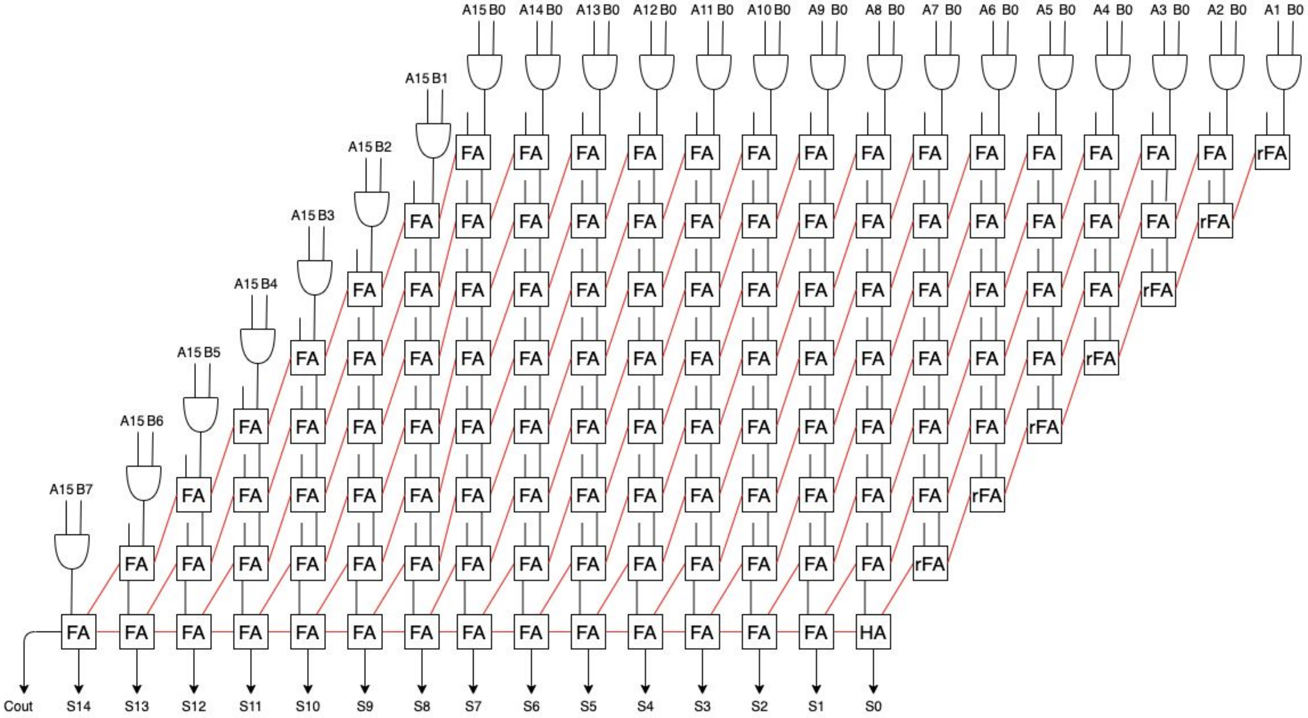
Top level



AREA: 206.16*227.57 um^2

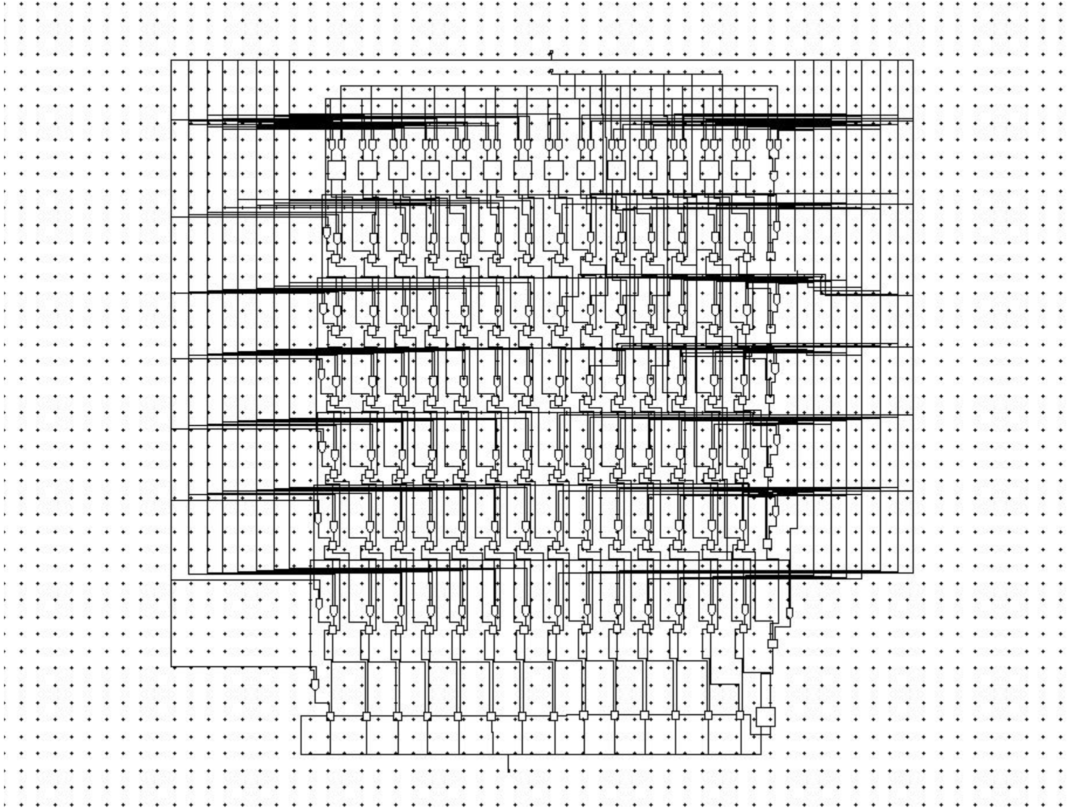
We passed all the DRC and LVS!

Carry-save 16x8 Multiplier



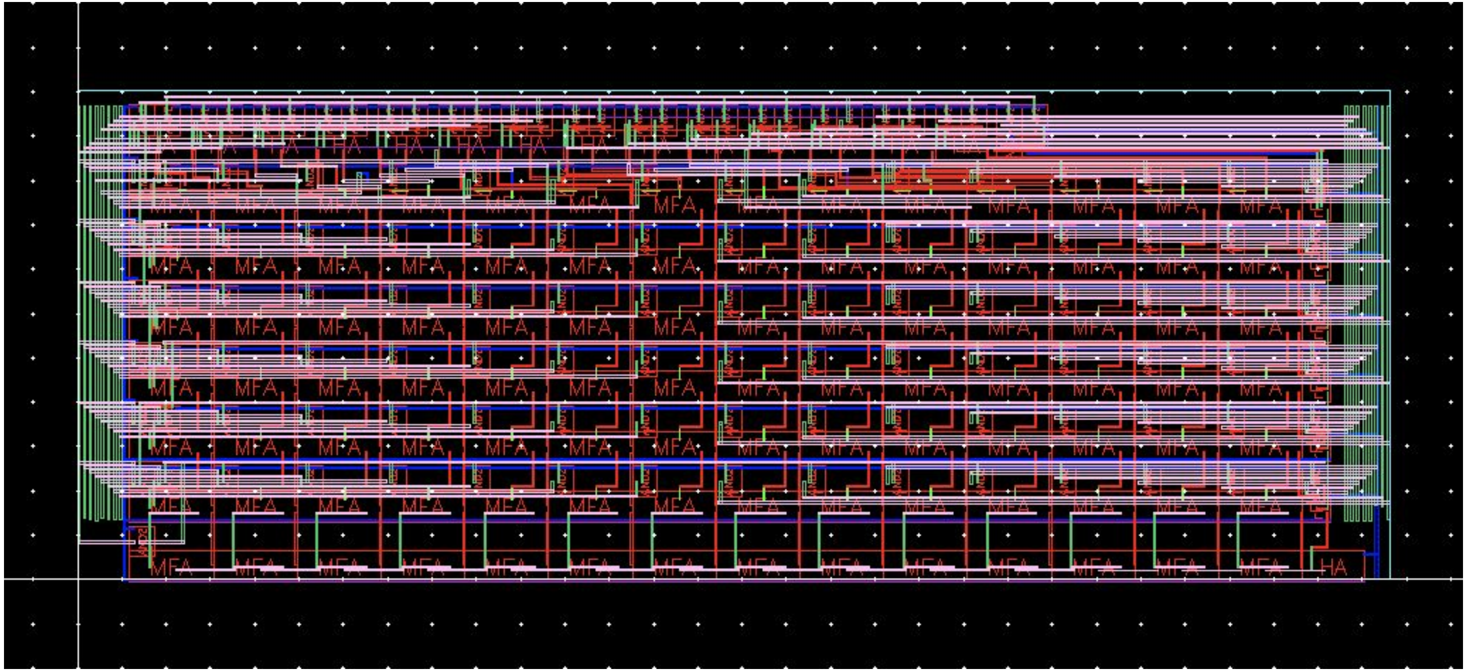
5/20/2020

Carry-save 16x8 Multiplier



5/20/2020

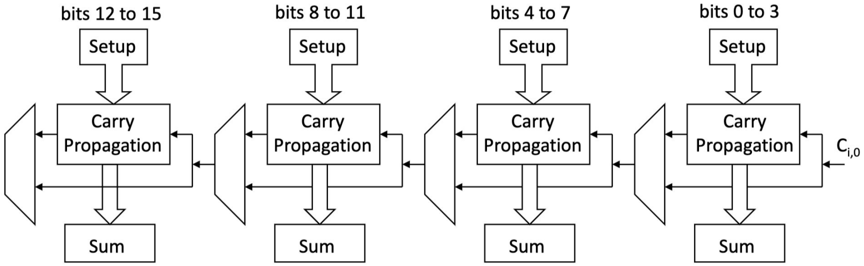
Carry-save 16x8 Multiplier



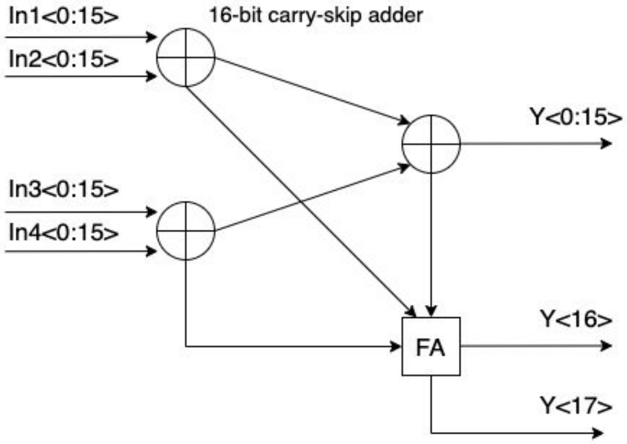
AREA: 148.19*55.135 um^2

Carry-skip Adder group

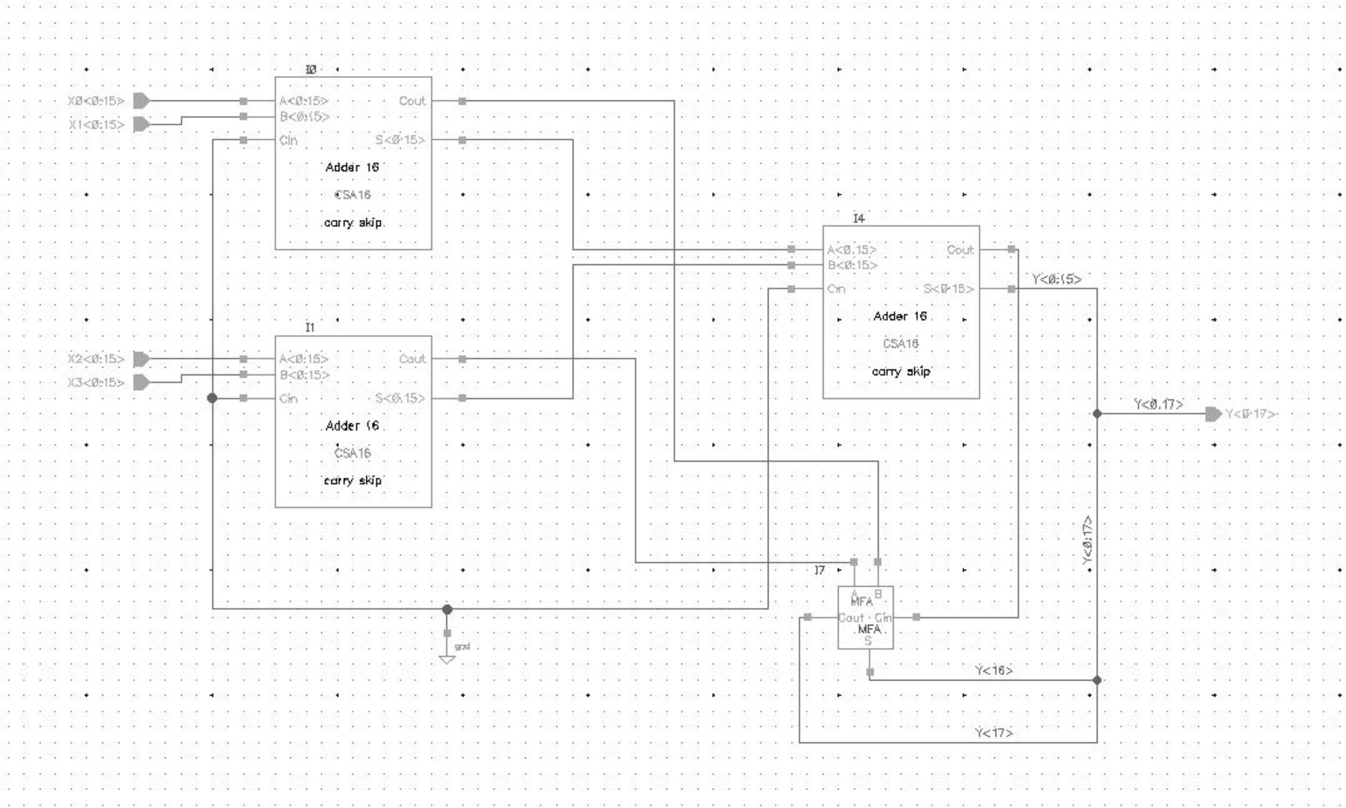
16-bit carry-skip adder (from ECE4740 lecture: adder)



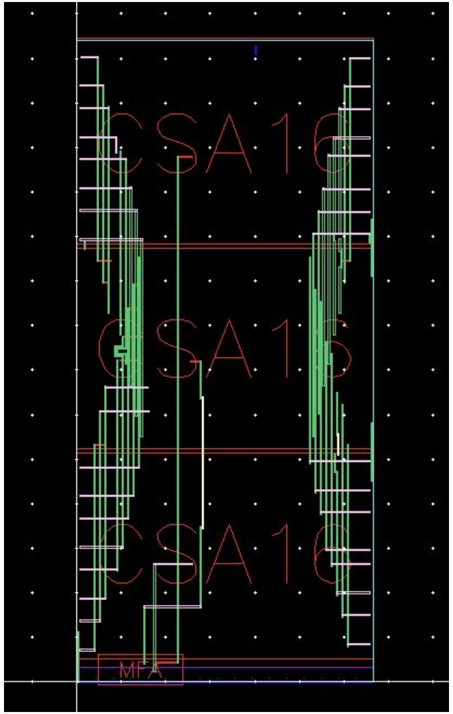
adder group



Adder group


IØ
XØ<Ø:15>
X1<Ø:15>
A<Ø:15>
B<Ø:15>
Cin
Cout
S<Ø:15>
Adder 16
CSA16
carry skip
I1
X2<Ø:15>
X3<Ø:15>
A<Ø:15>
B<Ø:15>
Cin
Cout
S<Ø:15>
Adder 16
CSA16
carry skip
I4
A<Ø:15>
B<Ø:15>
Cin
Cout
S<Ø:15>
Adder 16
CSA16
carry skip
Y<Ø:15>
Y<Ø:17>
Y<Ø:17>
Y<Ø:17>
I7
A
B
MFA
Cout
Cin
MFA
S
gnd
Y<16>
Y<17>

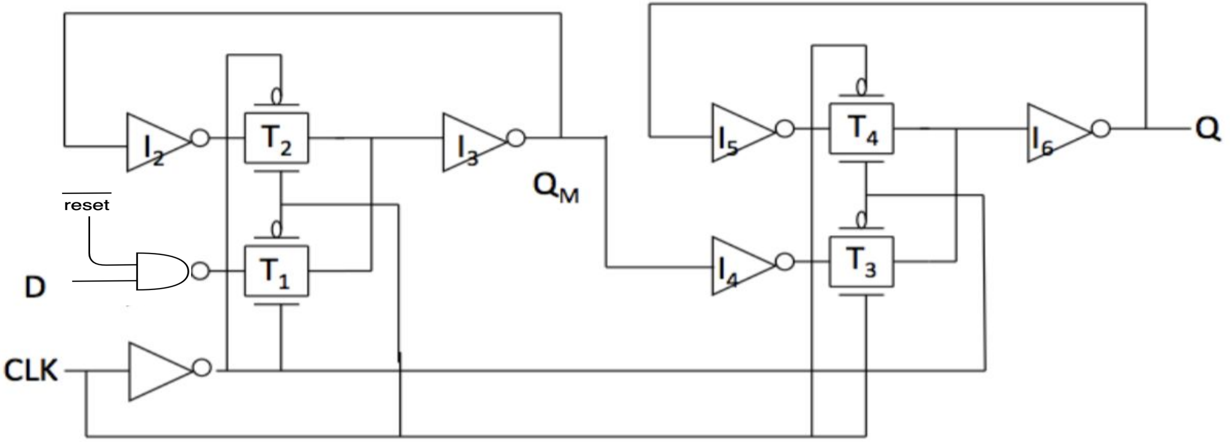
Adder group



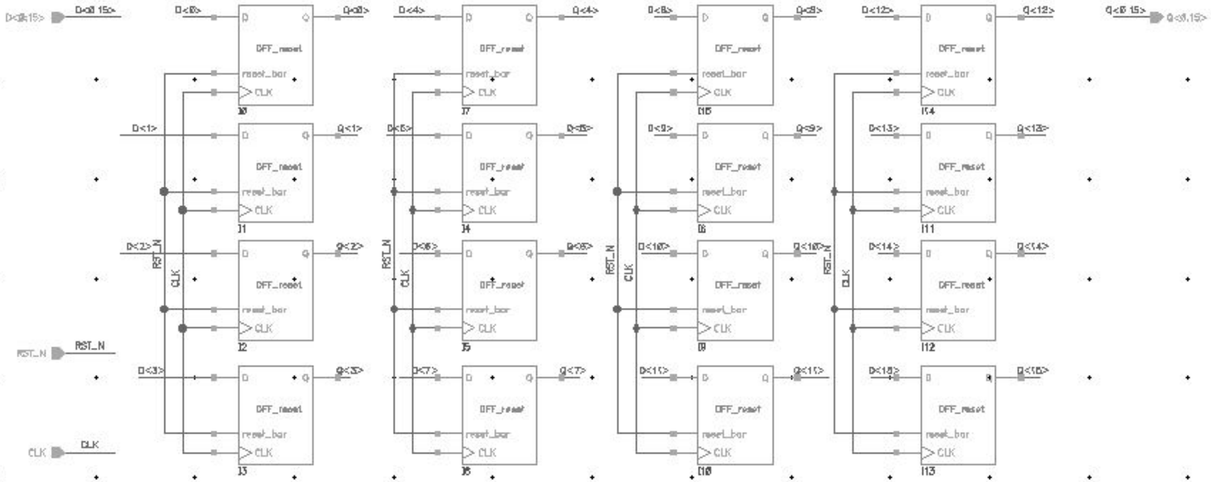
AREA: 33.28 * 72.54um^2

16-bit Register with reset

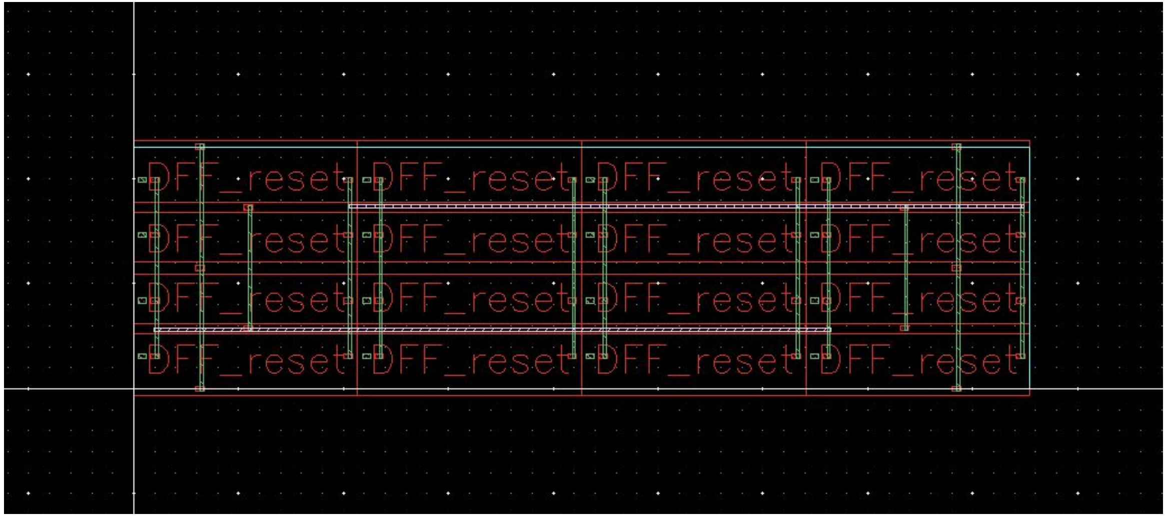
D Flip-flop with reset * 16



16-bit Register



16-bit Register



AREA: 42.72 * 12.12 um^2

CONTENTS

IV. Verification and Evaluation

Test and Evaluation Strategy

Use Python to auto-generate the VerilogA signal generator

Use VerilogA to apply stimulus to the design under test

Use MATLAB to verify results and post-process data

Unit test : Multiplier, Adder group, Register; directed and random test cases

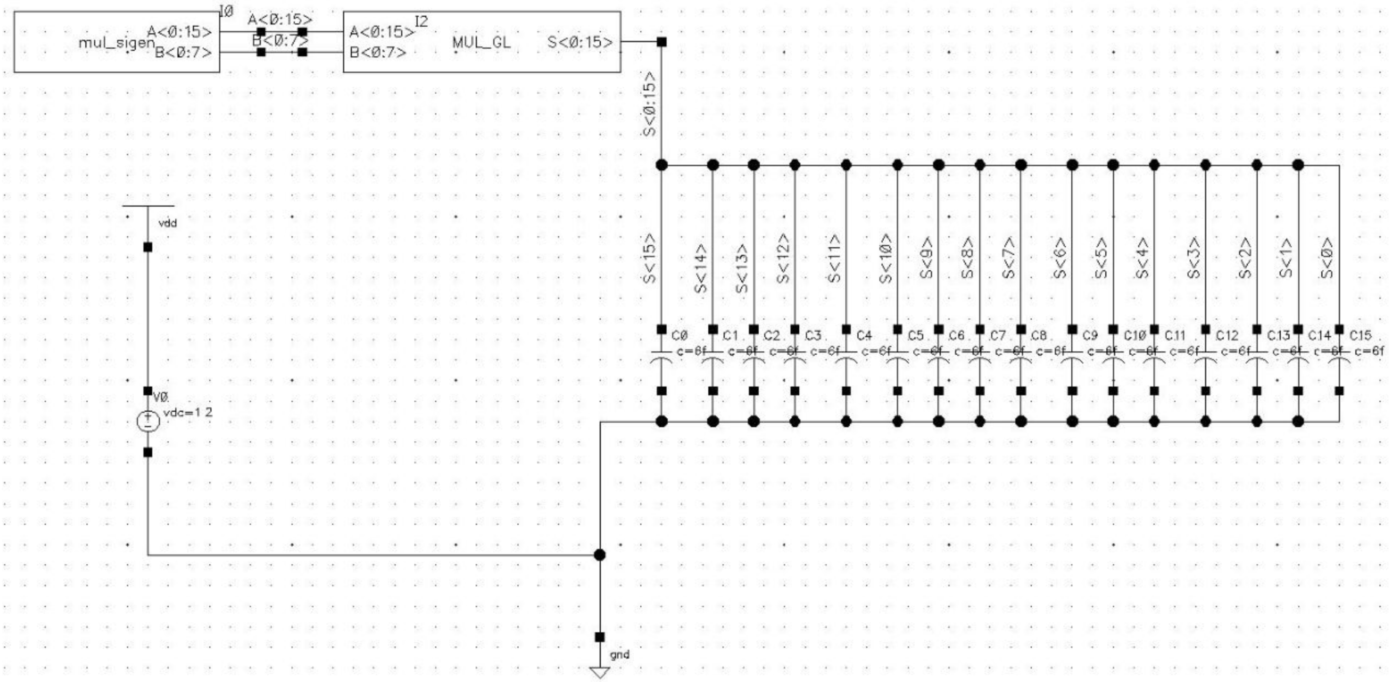
Static Timing Analysis (not finished)

Unit impulse response, Unit step response, Output to a 10kHz sine wave

AC analysis

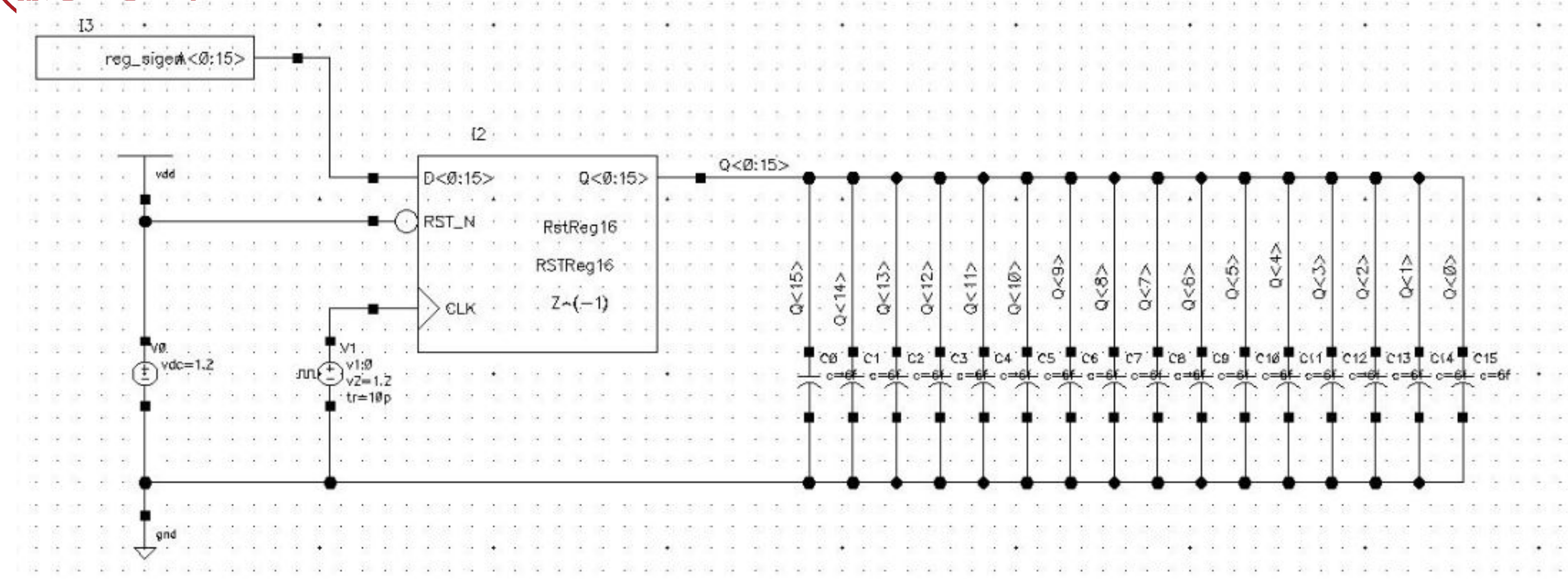
Power (not finished)

Testbench - Multiplier

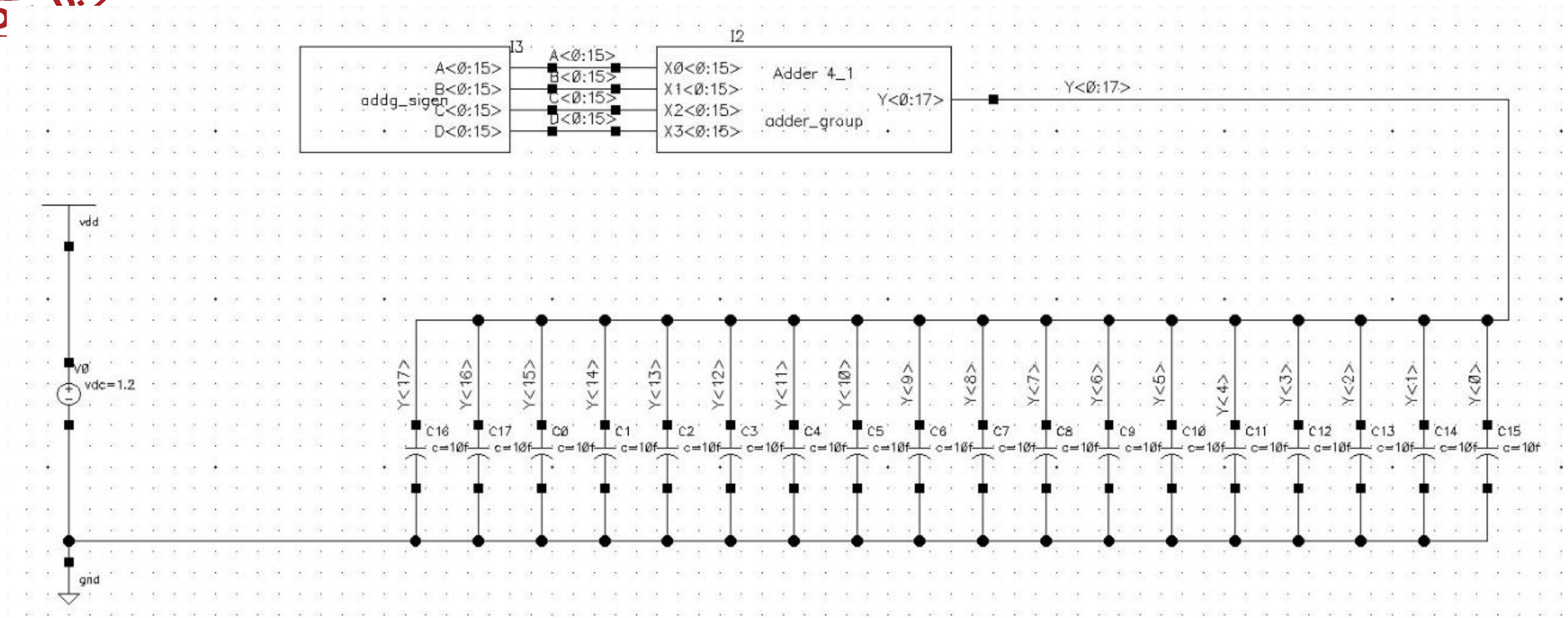


5/20/2020

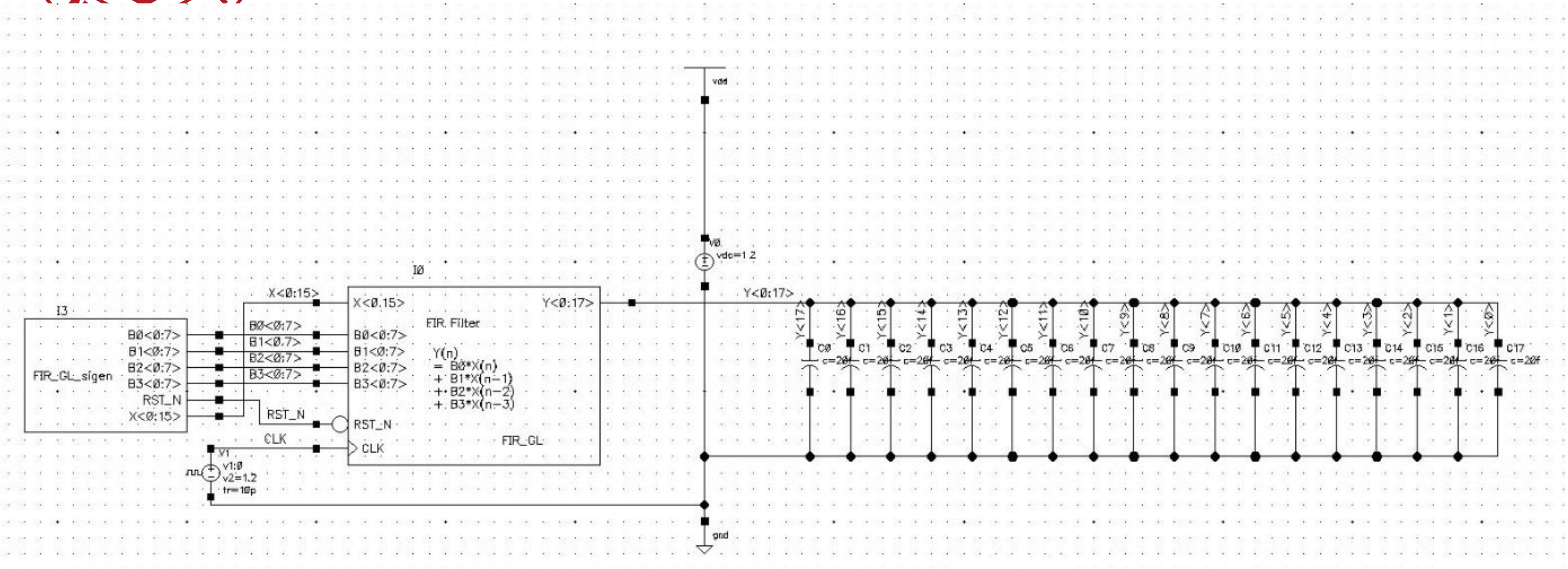
Testbench - Register



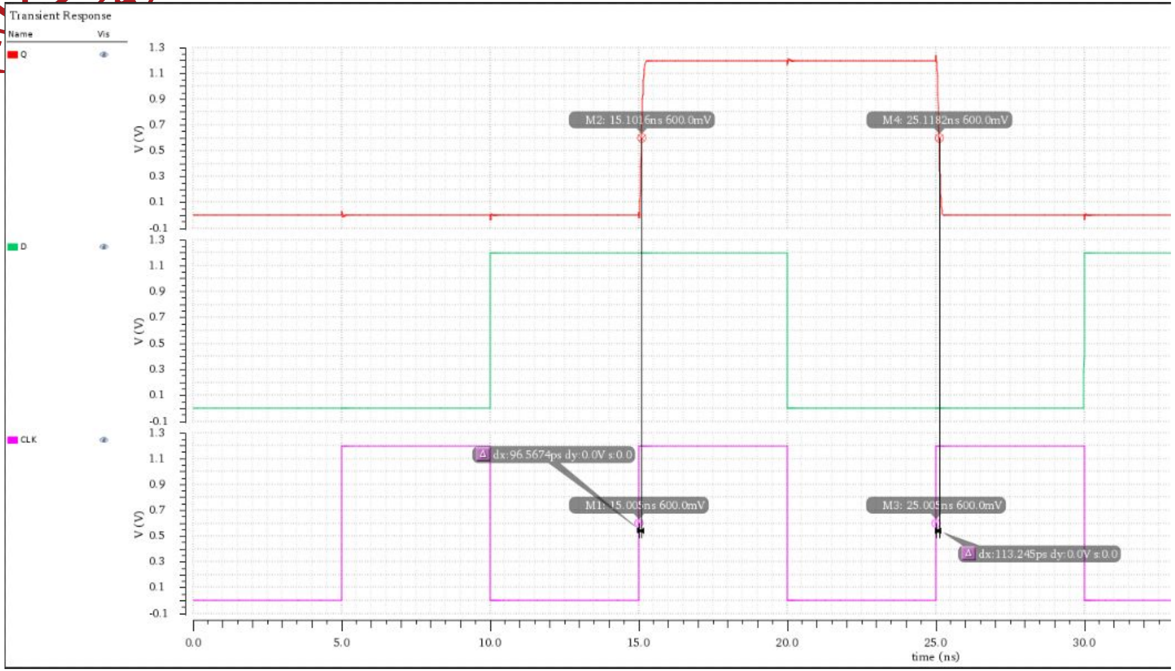
Testbench - Adder Group



Testbench - FIR Filter

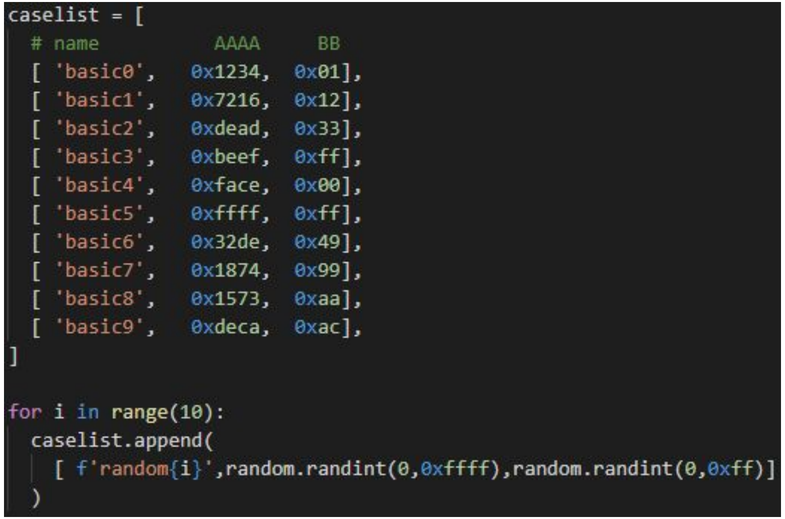


Unit tests - Register



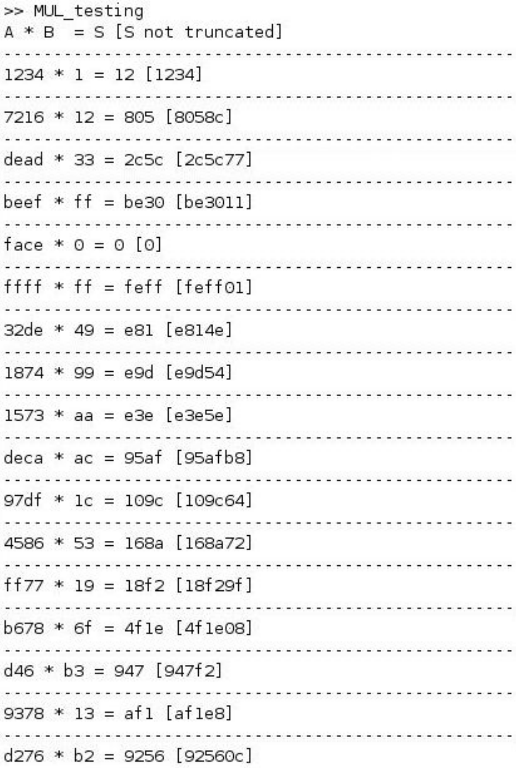
tpHL	191.37ps
tpLH	200.28ps
tsu	70ps
thd	-40ps

Unit tests - Multiplier



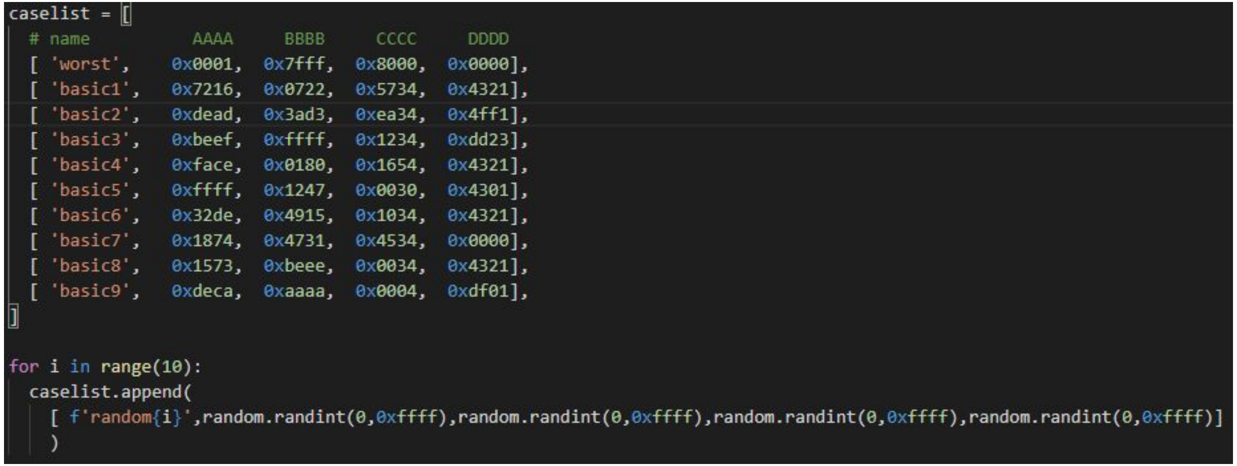
```python
caselist = [
  # name         AAAA     BB
  [ 'basic0',   0x1234,  0x01],
  [ 'basic1',   0x7216,  0x12],
  [ 'basic2',   0xdead,  0x33],
  [ 'basic3',   0xbeef,  0xff],
  [ 'basic4',   0xface,  0x00],
  [ 'basic5',   0xffff,  0xff],
  [ 'basic6',   0x32de,  0x49],
  [ 'basic7',   0x1874,  0x99],
  [ 'basic8',   0x1573,  0xaa],
  [ 'basic9',   0xdeca,  0xac],
]

for i in range(10):
  caselist.append(
    [ f'random{i}',random.randint(0,0xffff),random.randint(0,0xff)]
  )
```



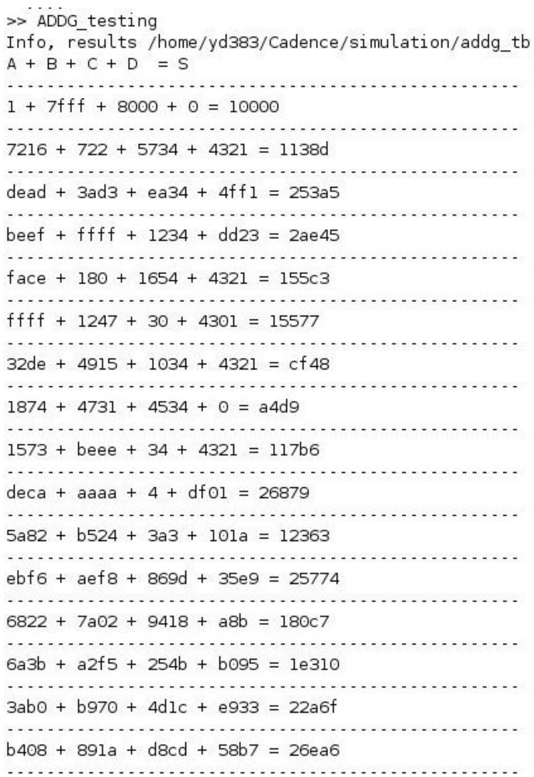
```
>> MUL_testing
A * B  = S [S not truncated]
--------------------------------------------------------------
1234 * 1 = 12 [1234]
--------------------------------------------------------------
7216 * 12 = 805 [8058c]
--------------------------------------------------------------
dead * 33 = 2c5c [2c5c77]
--------------------------------------------------------------
beef * ff = be30 [be3011]
--------------------------------------------------------------
face * 0 = 0 [0]
--------------------------------------------------------------
ffff * ff = feff [feff01]
--------------------------------------------------------------
32de * 49 = e81 [e814e]
--------------------------------------------------------------
1874 * 99 = e9d [e9d54]
--------------------------------------------------------------
1573 * aa = e3e [e3e5e]
--------------------------------------------------------------
deca * ac = 95af [95afb8]
--------------------------------------------------------------
97df * 1c = 109c [109c64]
--------------------------------------------------------------
4586 * 53 = 168a [168a72]
--------------------------------------------------------------
ff77 * 19 = 18f2 [18f29f]
--------------------------------------------------------------
b678 * 6f = 4f1e [4f1e08]
--------------------------------------------------------------
d46 * b3 = 947 [947f2]
--------------------------------------------------------------
9378 * 13 = af1 [af1e8]
--------------------------------------------------------------
d276 * b2 = 9256 [92560c]
```

Unit tests - Adder Group



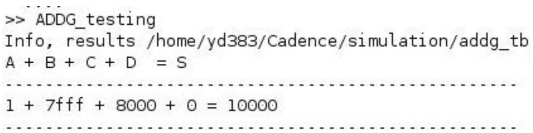
```python
caselist = [
  # name          AAAA     BBBB     CCCC     DDDD
  [ 'worst',    0x0001,  0x7fff,  0x8000,  0x0000],
  [ 'basic1',   0x7216,  0x0722,  0x5734,  0x4321],
  [ 'basic2',   0xdead,  0x3ad3,  0xea34,  0x4ff1],
  [ 'basic3',   0xbeef,  0xffff,  0x1234,  0xdd23],
  [ 'basic4',   0xface,  0x0180,  0x1654,  0x4321],
  [ 'basic5',   0xffff,  0x1247,  0x0030,  0x4301],
  [ 'basic6',   0x32de,  0x4915,  0x1034,  0x4321],
  [ 'basic7',   0x1874,  0x4731,  0x4534,  0x0000],
  [ 'basic8',   0x1573,  0xbeee,  0x0034,  0x4321],
  [ 'basic9',   0xdeca,  0xaaaa,  0x0004,  0xdf01],
]

for i in range(10):
  caselist.append(
    [ f'random{i}',random.randint(0,0xffff),random.randint(0,0xffff),random.randint(0,0xffff),random.randint(0,0xffff)]
    )
```

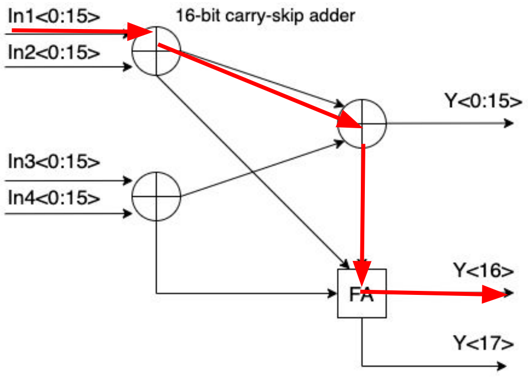


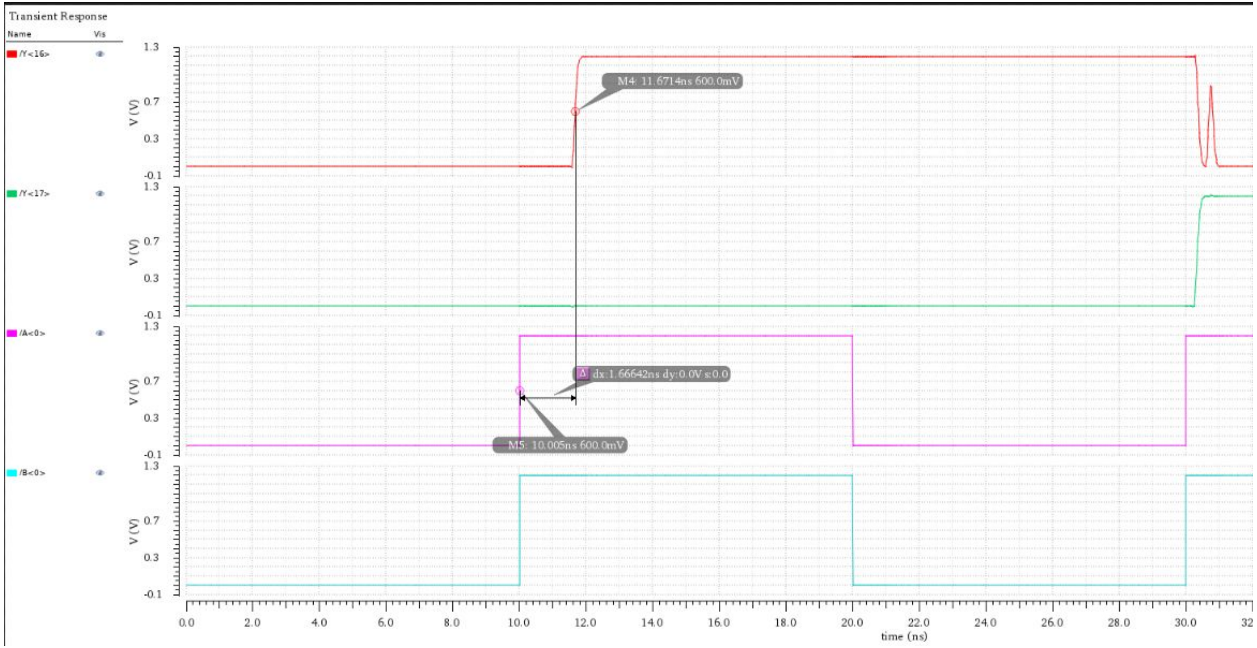
```
>> ADDG_testing
Info, results /home/yd383/Cadence/simulation/addg_tb
A + B + C + D  = S
--------------------------------------------------
1 + 7fff + 8000 + 0 = 10000
--------------------------------------------------
7216 + 722 + 5734 + 4321 = 1138d
--------------------------------------------------
dead + 3ad3 + ea34 + 4ff1 = 253a5
--------------------------------------------------
beef + ffff + 1234 + dd23 = 2ae45
--------------------------------------------------
face + 180 + 1654 + 4321 = 155c3
--------------------------------------------------
ffff + 1247 + 30 + 4301 = 15577
--------------------------------------------------
32de + 4915 + 1034 + 4321 = cf48
--------------------------------------------------
1874 + 4731 + 4534 + 0 = a4d9
--------------------------------------------------
1573 + beee + 34 + 4321 = 117b6
--------------------------------------------------
deca + aaaa + 4 + df01 = 26879
--------------------------------------------------
5a82 + b524 + 3a3 + 101a = 12363
--------------------------------------------------
ebf6 + aef8 + 869d + 35e9 = 25774
--------------------------------------------------
6822 + 7a02 + 9418 + a8b = 180c7
--------------------------------------------------
6a3b + a2f5 + 254b + b095 = 1e310
--------------------------------------------------
3ab0 + b970 + 4d1c + e933 = 22a6f
--------------------------------------------------
b408 + 891a + d8cd + 58b7 = 26ea6
--------------------------------------------------
```

Unit tests - Adder Group - Worst Case



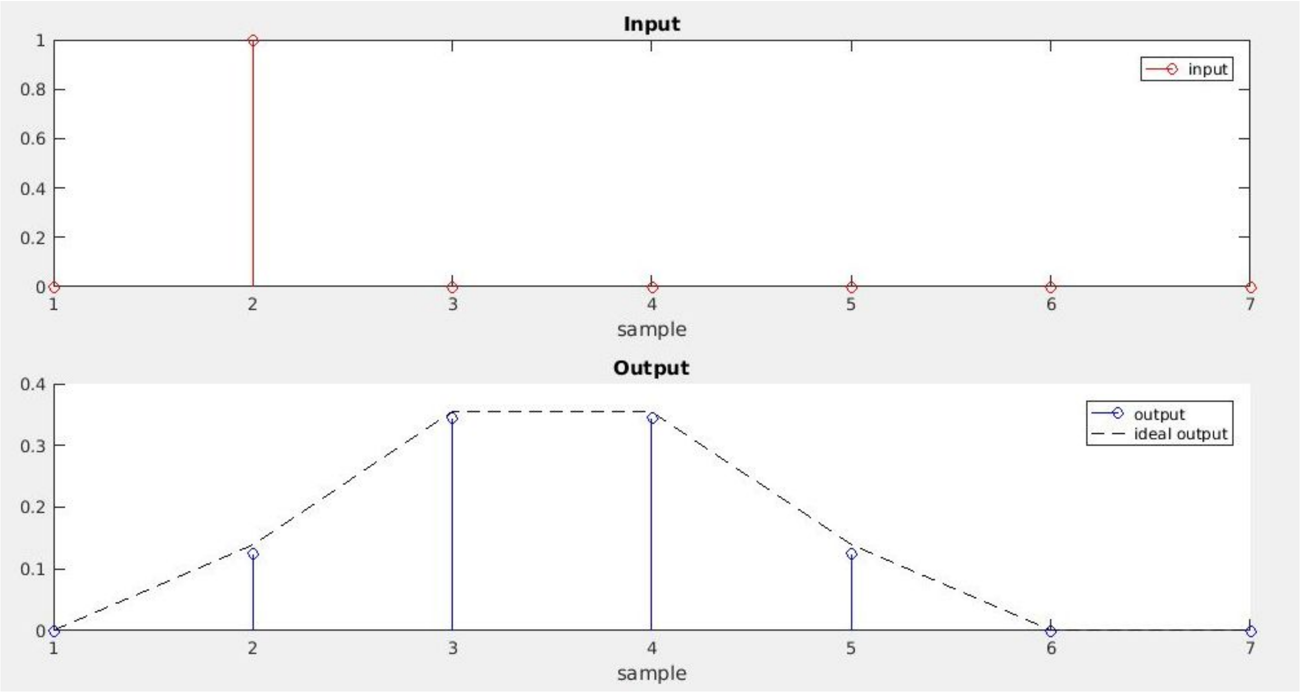
```
>> ADDG_testing
Info, results /home/yd383/Cadence/simulation/addg_tb
A + B + C + D  = S
------------------------------------------------------
1 + 7fff + 8000 + 0 = 10000
------------------------------------------------------
```



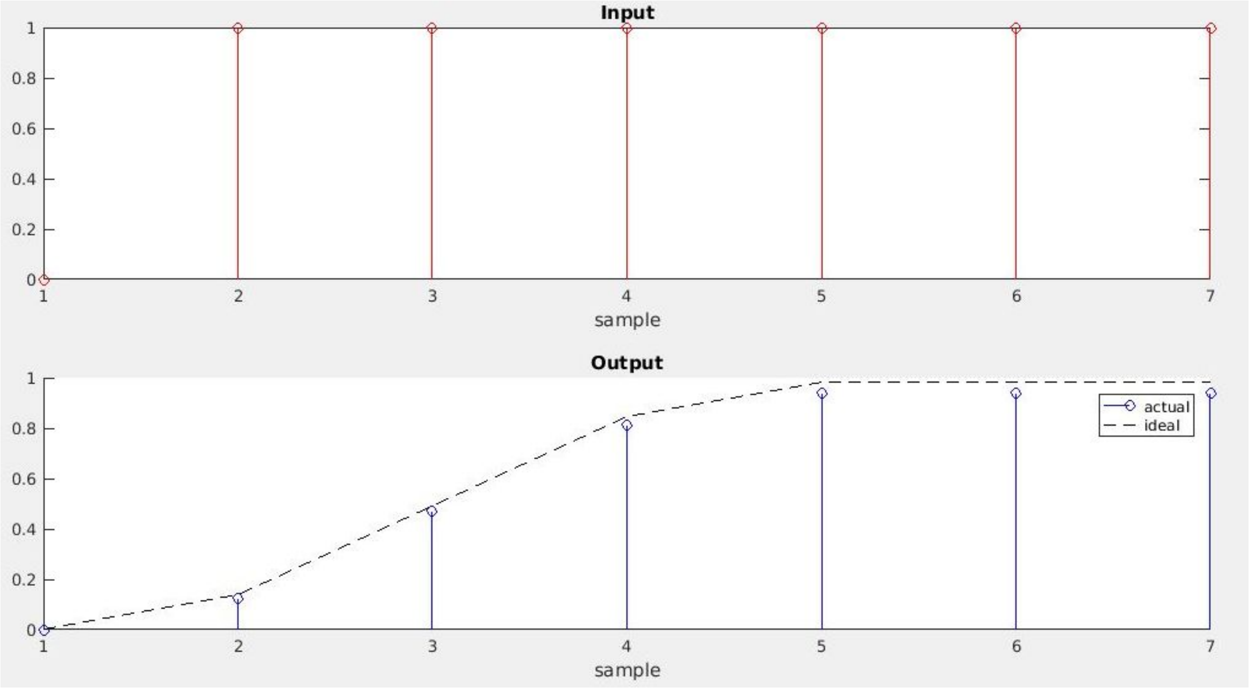


5/20/2020

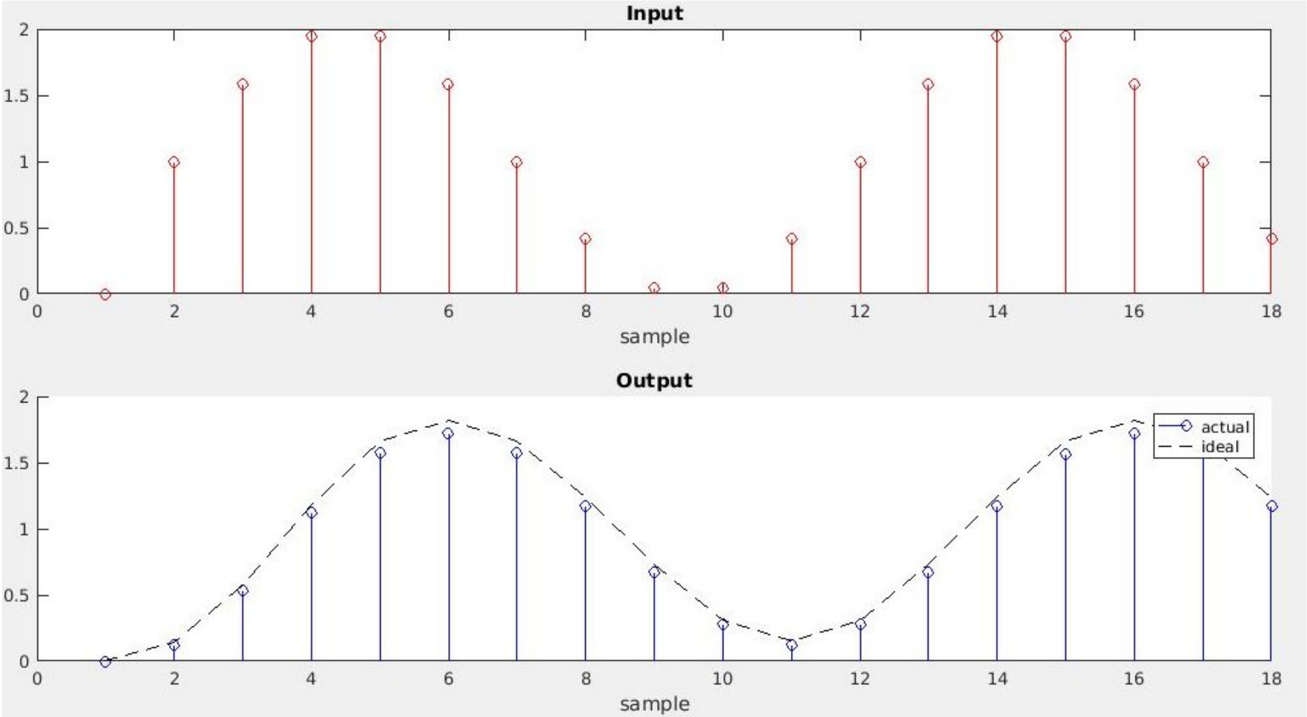
Unit Impulse Response



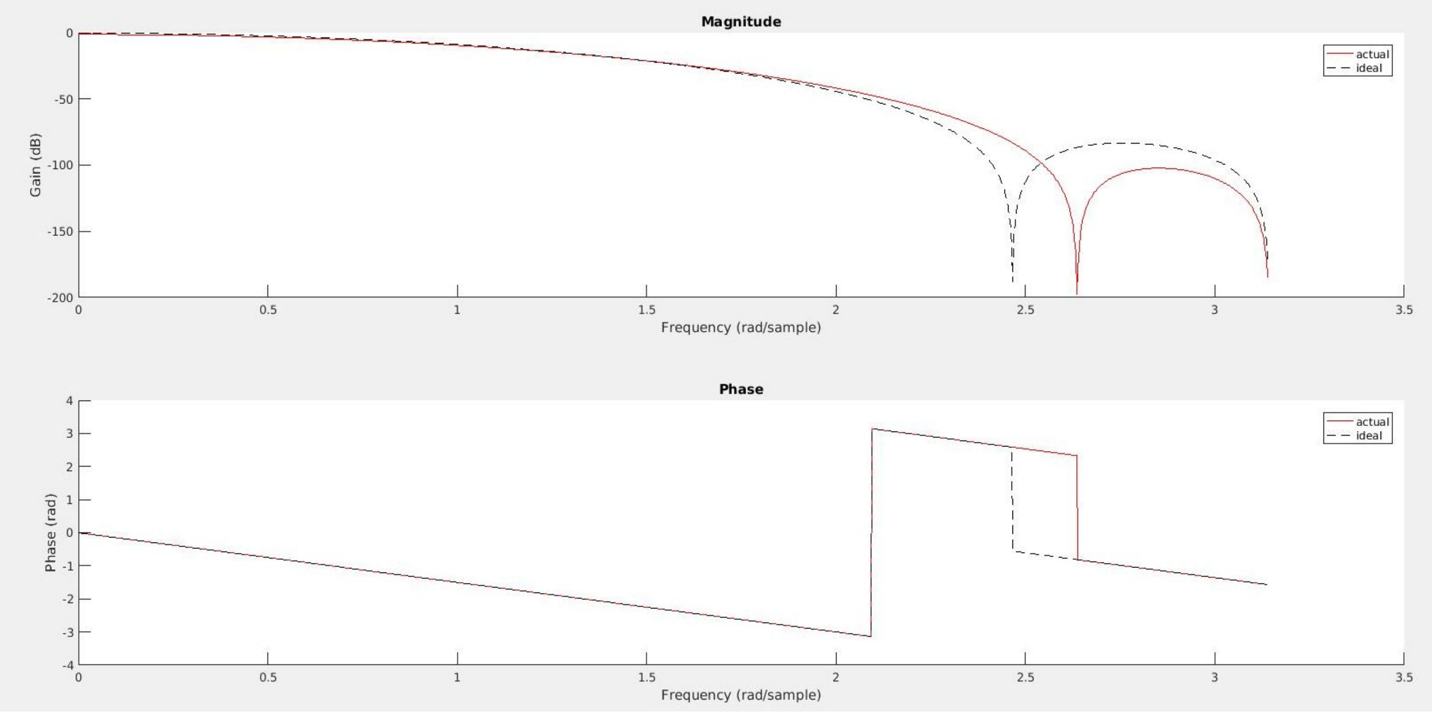
Unit Step Response



Sine Wave Input - Output



Frequency Response



Members' contribution

Yixiao Du (yd383)	Architecture, Adder group, Register, Test and verification
Yibang Xiao (yx455)	Multiplier Architecture and layout design, reduced HA, final FIR layout
Yuxiang Long (yl3377)	Multiplier design, reduced FA, FIR layout
Yifan Yang (yy887)	D Flip-flop, Buffer tree, participated in Multiplier layout design