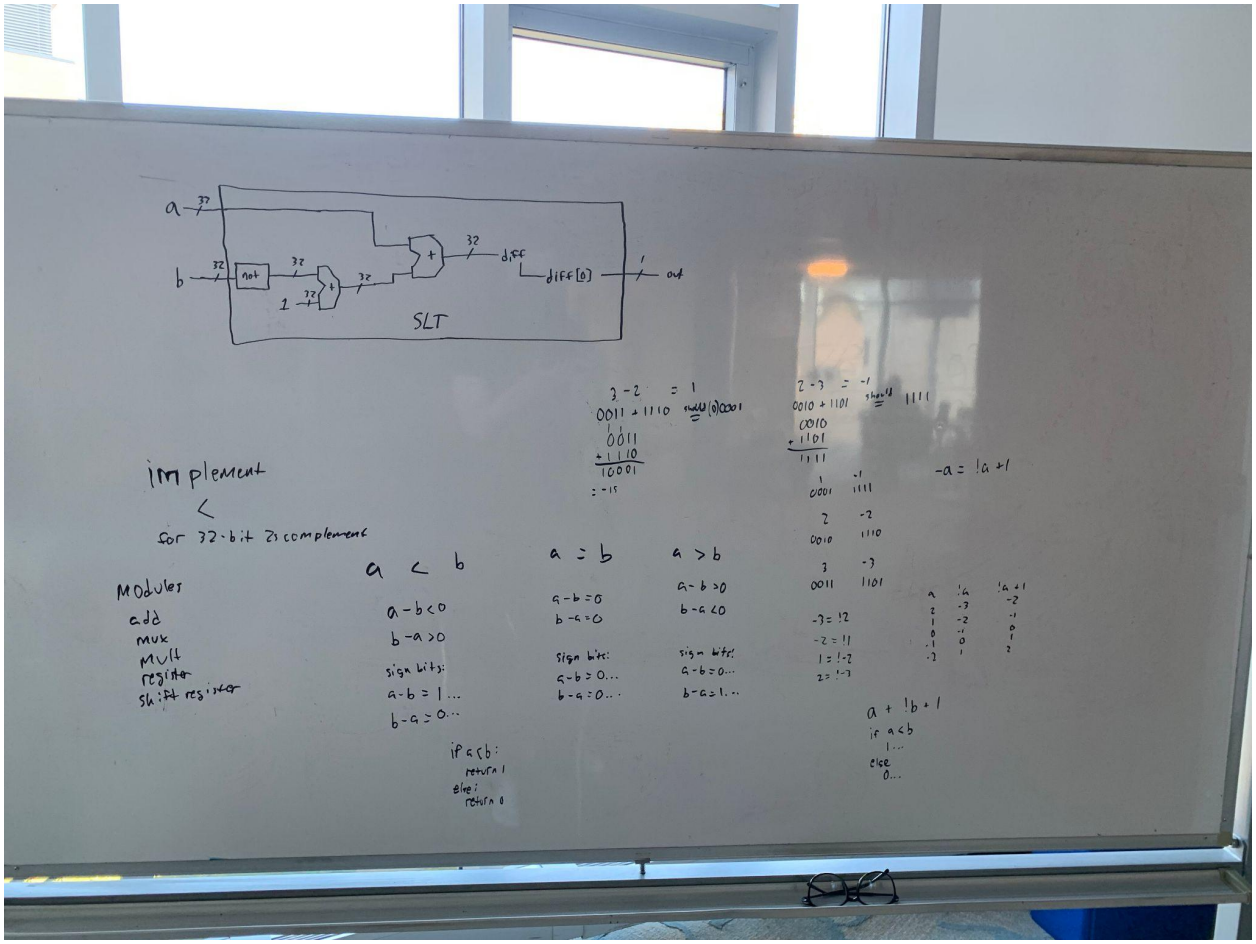
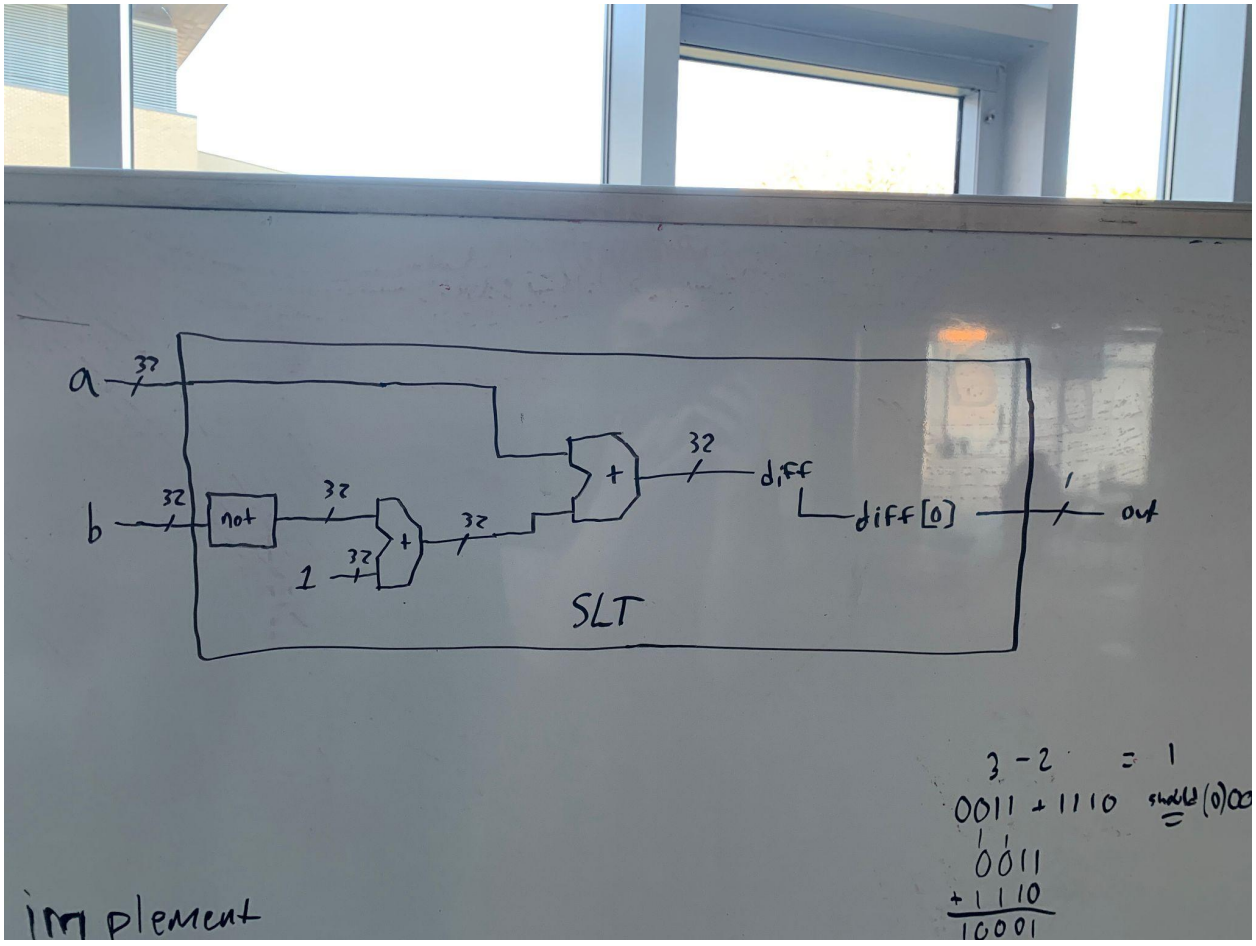
Mira

For the SLT, I missed a pretty basic issue, which is that this diagram assumes a and b have the same sign. I realized that issue while coding, but I wasn't at the whiteboard anymore to fix the diagram.


a
32
b
32
not
32
1
32
+
32
+
32
diff
diff[0]
1
out
SLT
implement
<
for 32-bit 2s complement
Modules
add
mux
Mult
register
Shift register
a < b
a - b < 0
b - a > 0
sign bits:
a - b = 1...
b - a = 0...
if a<b:
return 1
else:
return 0
a = b
a - b = 0
b - a = 0
sign bits:
a - b = 0...
b - a = 0...
a > b
a - b > 0
b - a < 0
sign bits:
a - b = 0...
b - a = 1...
2 - 2 = 1
0011 + 1110
0011
+1110
10001
= -15
2 - 3 = -1
0010 + 1101
0010
+1101
1111
-a = !a + 1
a + !b + 1
if a<b
1...
else
0...

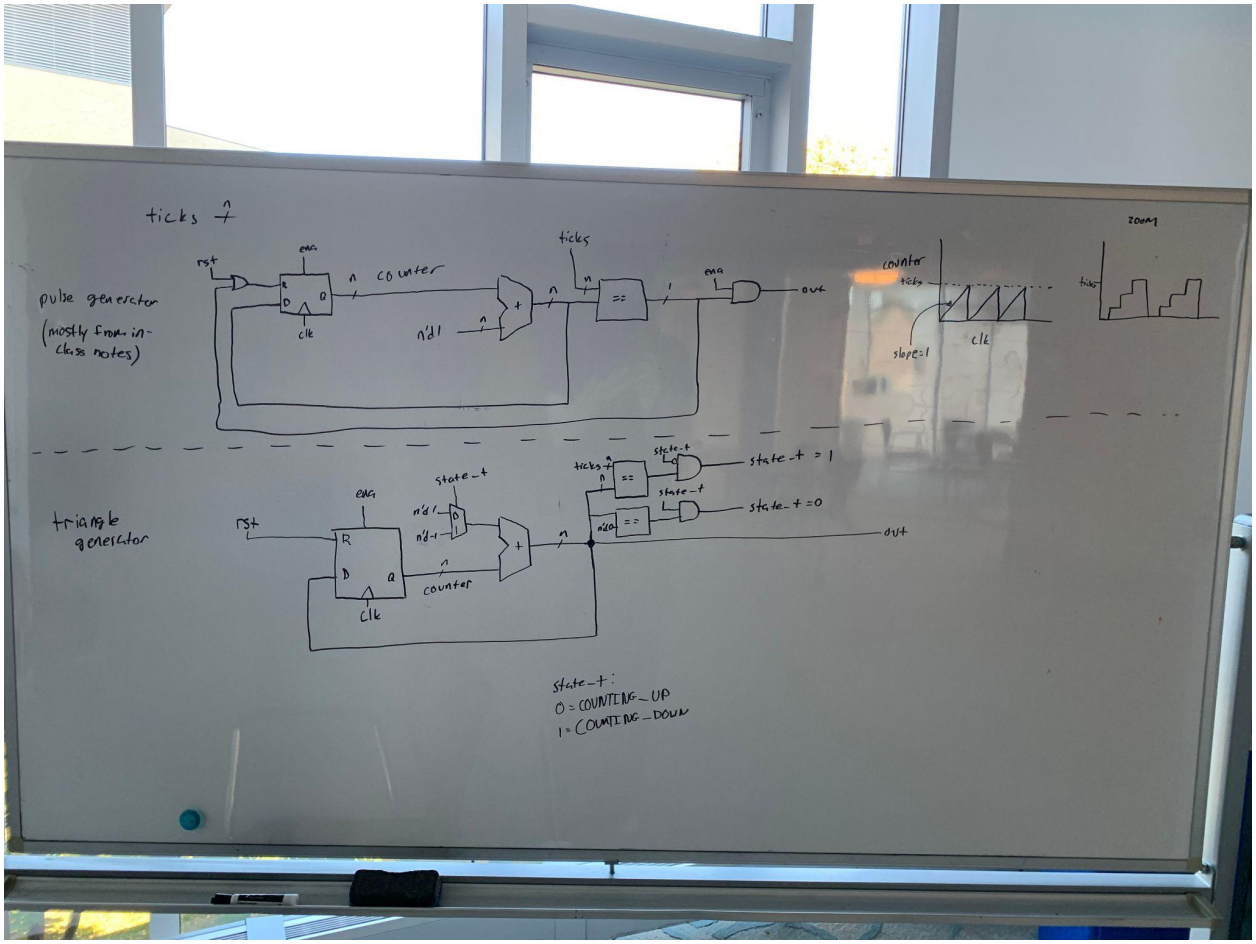


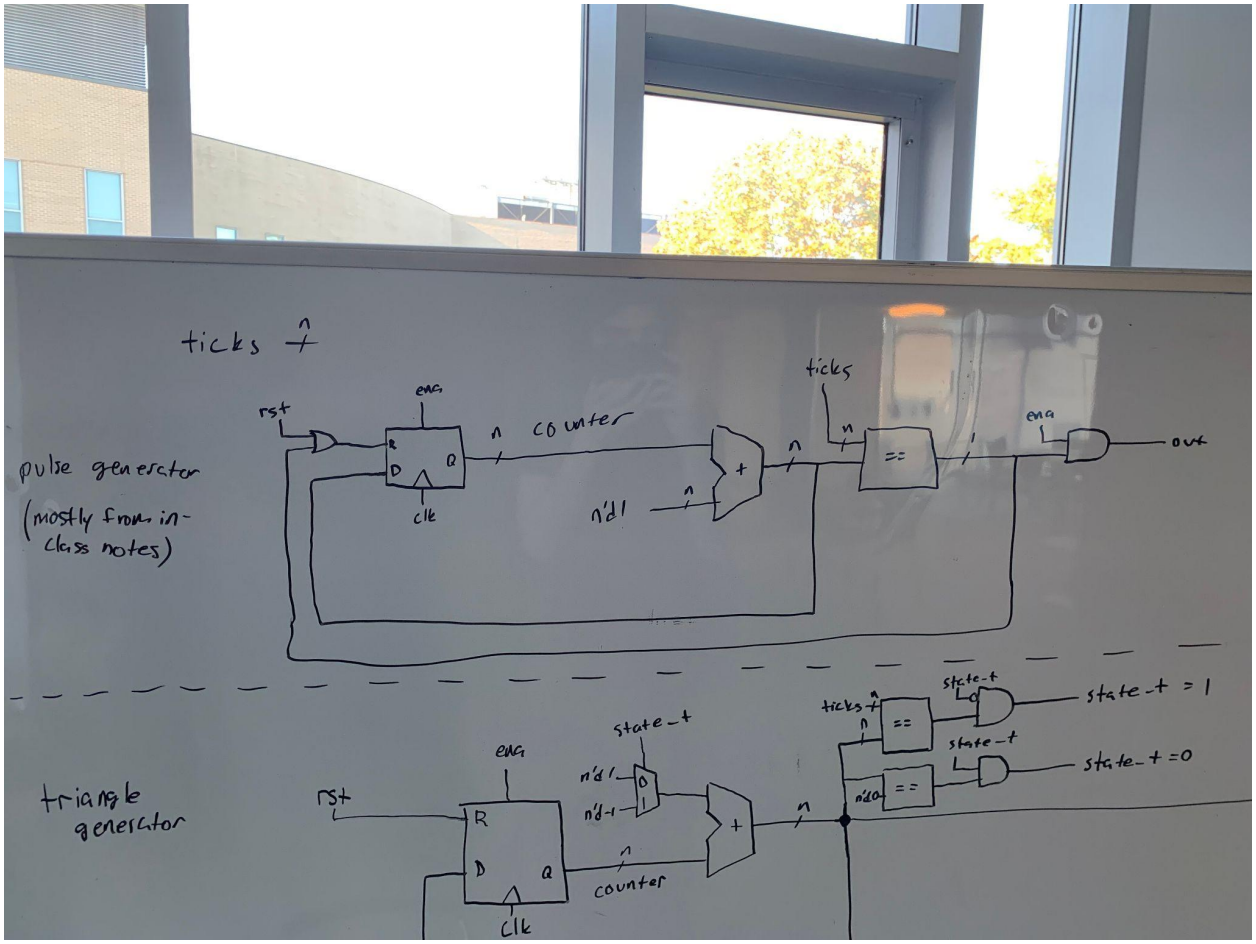
3 - 2 = 1

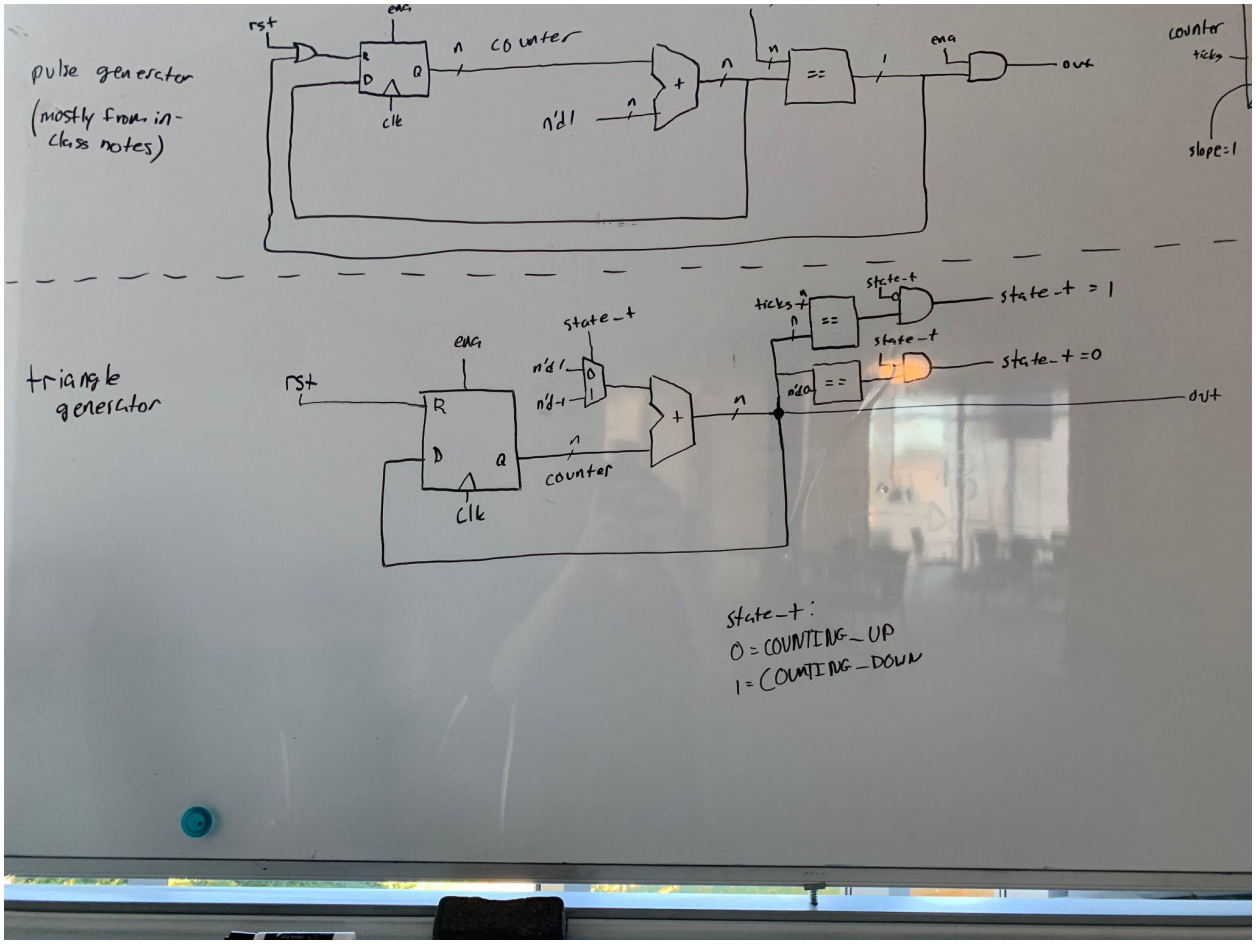
0011 + 1110 should (0)00

0011
+1110
10001

implement


ticks ↑
pulse generator
(mostly from in-class notes)
rst
en
counter
clk
n'd1
ticks
==
out
counter
ticks
slope=1
clk
zoom
ticks
triangle generator
state_t
n'd1
n'd-1
n'd0
state_t = 1
state_t = 0
dut
R
D
Q
counter
clk
state_t:
0 = COUNTING_UP
1 = COUNTING_DOWN


ticks ÷
pulse generator
(mostly from in-class notes)
rst
ena
R
D
Q
clk
n
counter
n'd1
+
ticks
==
ena
out
triangle generator
state_t
n'd1
n'd-1
counter
ticks÷
state_t = 1
state_t = 0
n'd0


pulse generator
(mostly from in-class notes)
rst
ena
counter
clk
n'd1
out
counter ticks
slope=1
triangle generator
state_t
ticks
state_t = 1
state_t = 0
counter
state_t:
0 = COUNTING_UP
1 = COUNTING_DOWN