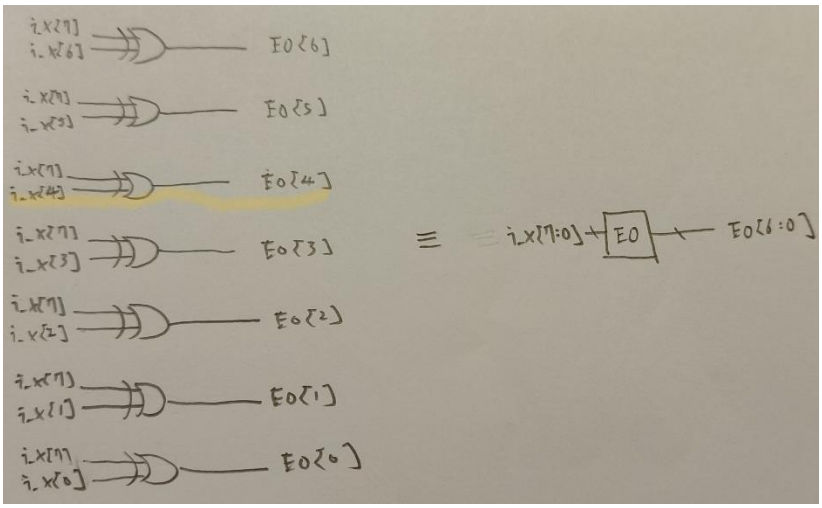
(a) Simulation

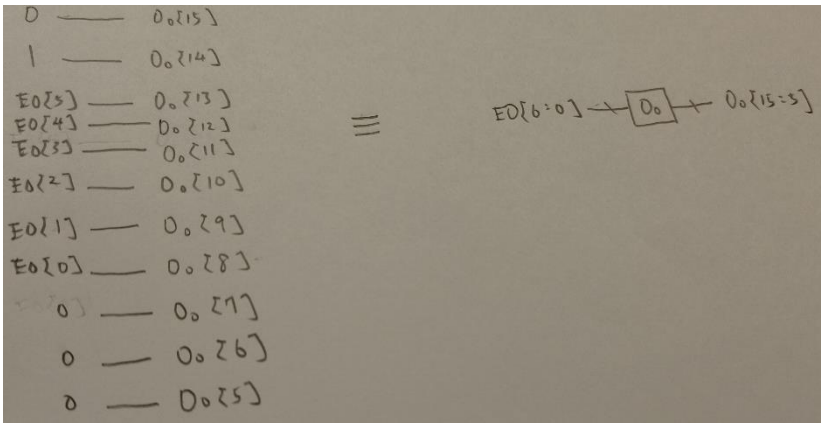
minimum cycle time is 2.8

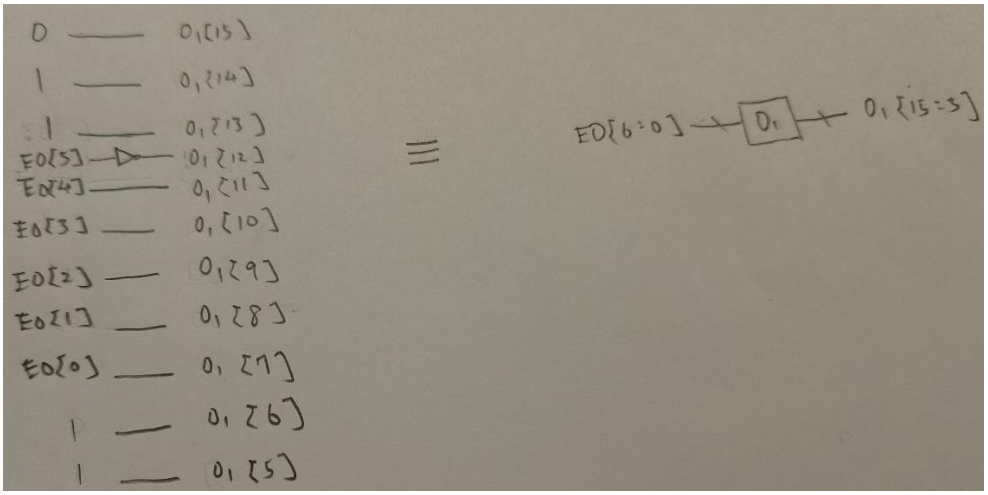
```
================================================
              Simulation finished
================================================

================================================
                    Summary
================================================
  Clock cycle:                2.8 ns
  Number of transistors:      863
  Total excution cycle:       256
  Approximation Error Score: 30365.0
  Performance Score:          618598.4
================================================
```

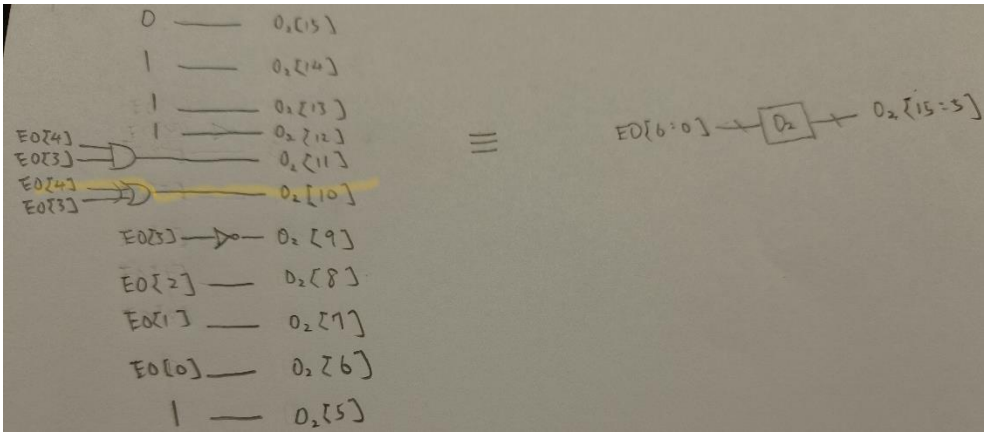
(b) Circuit diagram

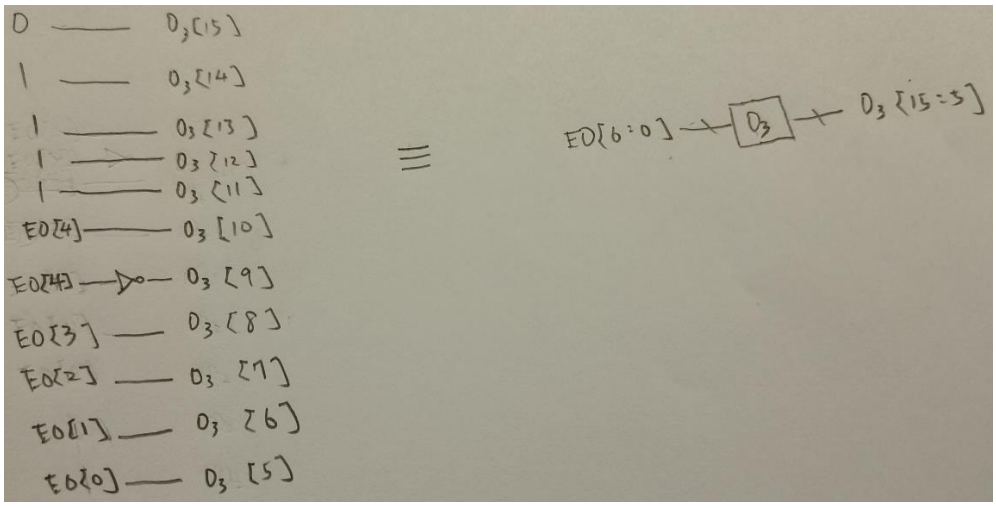
Critical path is marked in yellow

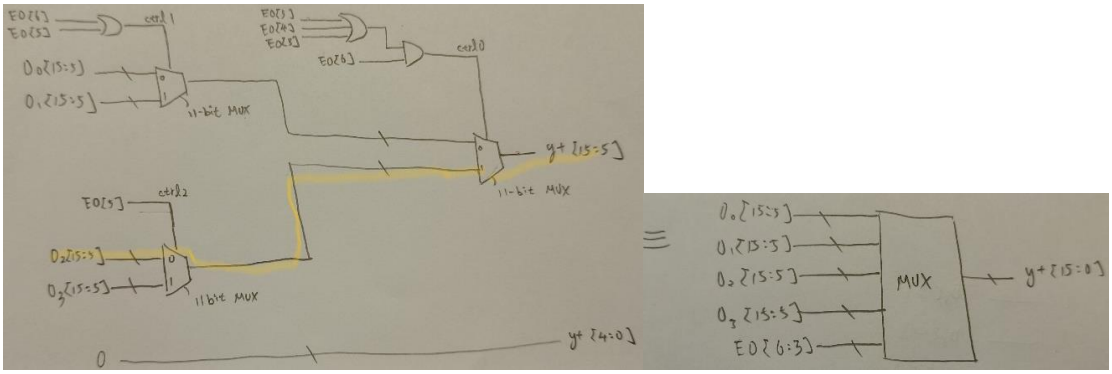


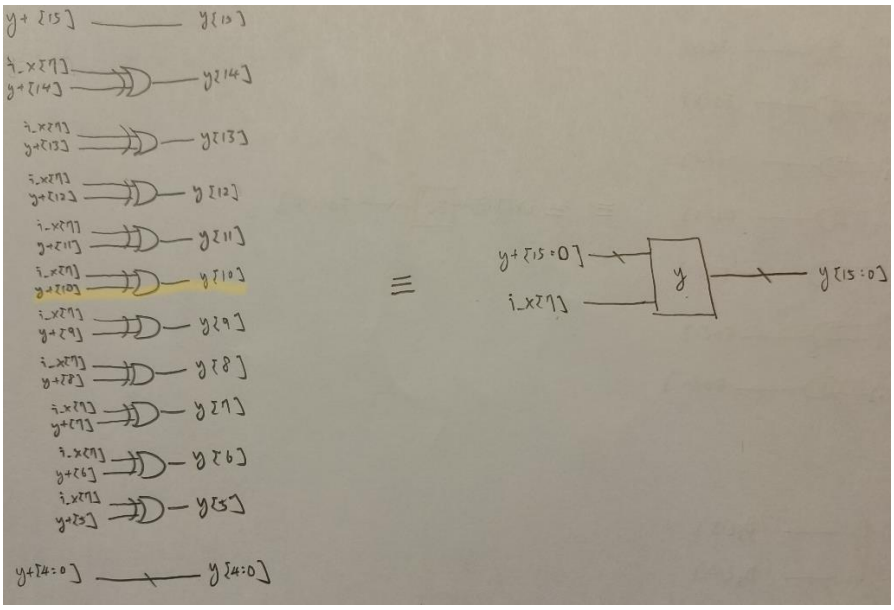


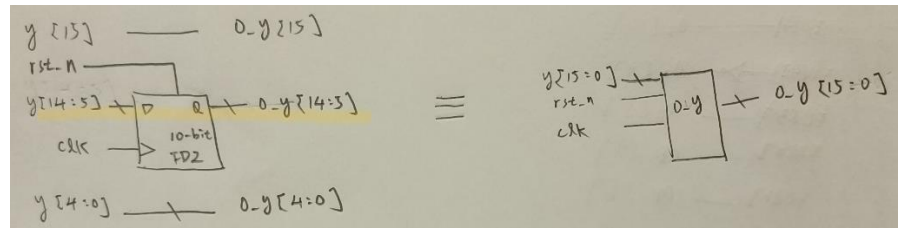

0 — O1[15]
1 — O1[14]
1 — O1[13]
EO[5] — O1[12]
EO[4] — O1[11]
EO[3] — O1[10]
EO[2] — O1[9]
EO[1] — O1[8]
EO[0] — O1[7]
1 — O1[6]
1 — O1[5]
≡
EO[6:0] → O1 → O1[15:5]

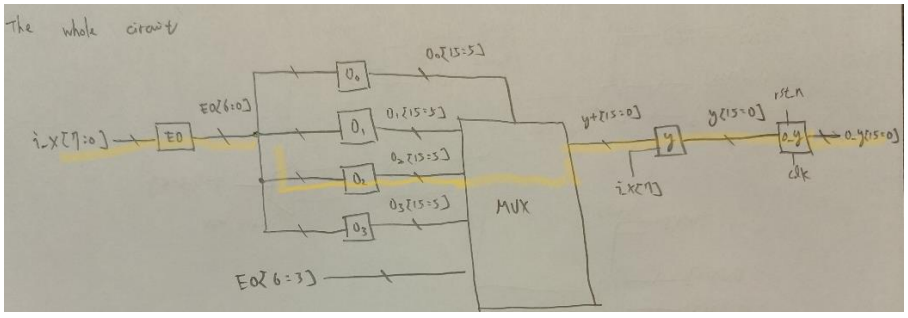

0 — O2[15]
1 — O2[14]
1 — O2[13]
1 — O2[12]
EO[4]
EO[3]
O2[11]
EO[4]
EO[3]
O2[10]
EO[3] — O2[9]
EO[2] — O2[8]
EO[1] — O2[7]
EO[0] — O2[6]
1 — O2[5]
≡
EO[6:0] → O2 → O2[15:5]

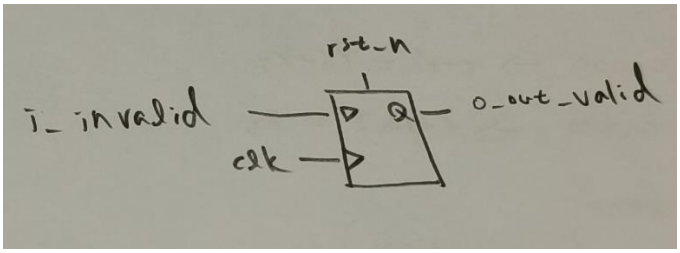

0 — O3[15]
1 — O3[14]
1 — O3[13]
1 — O3[12]
1 — O3[11]
EO[4] — O3[10]
EO[4] — O3[9]
EO[3] — O3[8]
EO[2] — O3[7]
EO[1] — O3[6]
EO[0] — O3[5]
≡
EO[6:0] → O3 → O3[15:5]


MUX


10-bit FD2


rst_n
clk
10-bit FD2


The whole circuit
MUX


rst_n
i_invalid
clk
o_out_valid

(c) Discussion

Introduce your design

Consider $x \geq 0 \Leftrightarrow x_7 = 0$, we approximate sigmoid by four functions

$$\Rightarrow y+[15:0] = \text{sigmoid}(x) = \begin{cases} \frac{1}{4}x + \frac{1}{2} = 0_0[15:5], & x = 00.00000 \sim 00.11111 \\ \frac{1}{8}x + \frac{1}{2} + \frac{1}{8} + \frac{1}{512} + \frac{1}{1024} = 0_1[15:5], & x = 01.00000 \sim 10.00111 \\ \frac{1}{16}x + \frac{1}{2} + \frac{1}{4} + \frac{1}{64} + \frac{1}{1024} = 0_2[15:5], & x = 10.01000 \sim 10.11111 \\ \frac{1}{32}x + \frac{1}{2} + \frac{1}{4} + \frac{1}{16} + \frac{1}{32} + \frac{1}{64} = 0_3[15:5], & x \geq 11.00000 \end{cases}$$

let x be $0\ x_6x_5.x_4x_3x_2x_1x_0$

(i) $O_0 = \frac{1}{4} \cdot x_6x_5.x_4x_3x_2x_1x_0 + \frac{1}{2} = 0.00x_4x_3x_2x_1x_0$

$$+ 0.1$$

$$= 0.10x_4x_3x_2x_1x_0000 = 0_0[15:5]$$

(ii) $O_1 = \frac{1}{8} \cdot x_6x_5.x_4x_3x_2x_1x_0 + \frac{1}{2} + \frac{1}{8} + \frac{1}{512} + \frac{1}{1024} = 0.0x_6x_5x_4x_3x_2x_1x_0$

$$+ 0.1010000011$$

$$= 0.11(x_5')x_4x_3x_2x_1x_011 = 0_1[15:5]$$

(iii) $O_2 = \frac{1}{16} \cdot x_6x_5.x_4x_3x_2x_1x_0 + \frac{1}{2} + \frac{1}{4} + \frac{1}{64} + \frac{1}{1024} = 0.0010x_4x_3x_2x_1x_0$

$$+ 0.1100010001$$

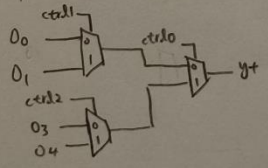
$$= 0.111(x_4x_3)(x_4 \oplus x_3)(x_3')x_2x_1x_01 = 0_2[15:5]$$

(iv) $O_3 = \frac{1}{32} \cdot x_6x_5.x_4x_3x_2x_1x_0 + \frac{1}{2} + \frac{1}{4} + \frac{1}{16} + \frac{1}{32} + \frac{1}{64} = 0.00011x_4x_3x_2x_1x_0$

$$+ 0.110111$$

$$= 0.1111(x_4)(x_4')x_3x_2x_1x_0 = 0_3[15:5]$$

Then, we use two level of MUX to get $y+[15:5]$



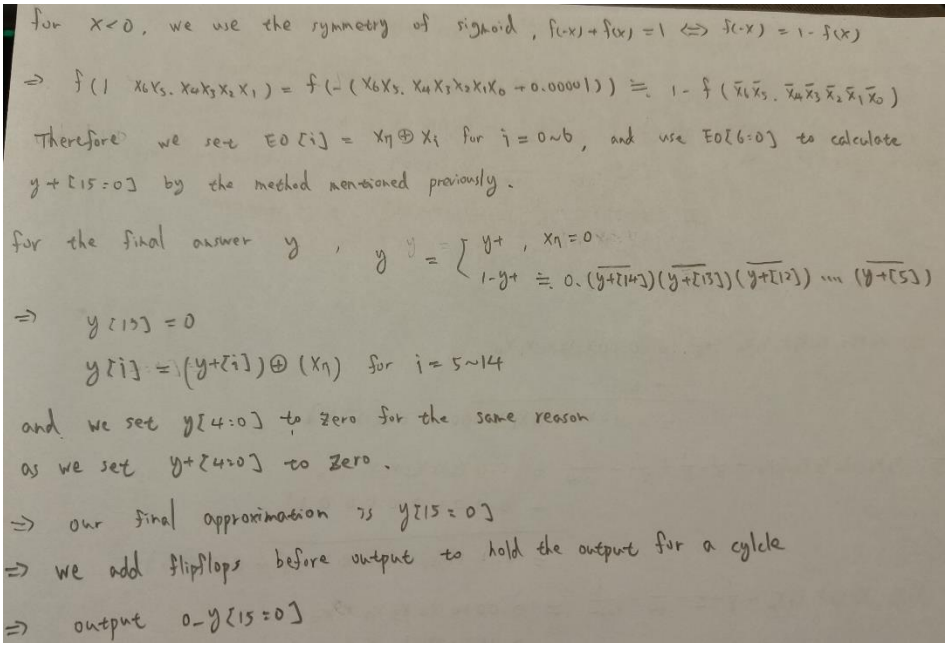
(i) ctrl0 = 1 when $x \geq 10.01000 \Rightarrow$ ctrl0 = $x_6(x_5+x_4+x_3)$

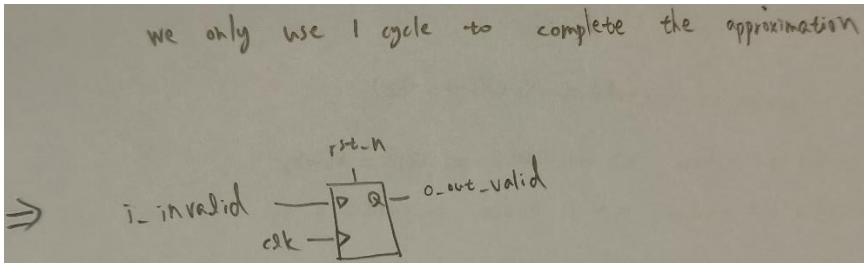
(ii) given ctrl = 0, ctrl1 = 1 when $x \geq 01.00000 \Rightarrow$ ctrl = x_6+x_5

(iii) given ctrl = 1, ctrl2 = 1 when $x \geq 11.00000 \Rightarrow$ ctrl2 = x_5

As for $y+[4:0]$, we just set them to zero because accuracy is meaningless here due to approximation

$\Rightarrow$ we get $y+[15:0] = \text{sigmoid}(x)$ for $x \geq 0$





How do you improve your critical path and the number of transistors

(i)

We use NDs, NRs instead of Ans and ORs, because the former requires fewer transistors and have less delay than the latter.

(ii)

o_y[15] and o_y[4:0] are always zero, so we can save 6 flipflops to reduce the usage of transistors.

(iii)

For EO, the delay from A to Z is higher than that of B to Z, and for MUX21H, the delay from A to Z is lower than that of B to Z, so we make the critical path go the one with smaller delay, we improved our critical path by 0.1 using this method.

How do you trade-off between area and speed?

I considered using pipeline in my design; however, because flipflops requires a lot of transistors, the reduction in the cycle time is less than the increase in the usage transistors. Hence, I did not use pipeline. Other than that, I found little trade-off between time and area, the main trade-off is between accuracy and A*T.

Compare with other architectures you have designed

Besides my original design, I also implemented another circuit using LUT based design. The simulation results are shown below. Though it has longer clock cycle, it has better error score and better performance score.