EE5393HW4
LI DAOQI 5101728 daoqi001@umn.edu

Problem 1

I do question b. See uploaded code. This is a program that generate the truth table corresponding to the conditional permutation written in CKT.txt.

In uploaded code CKT is the circuit corresponding to function

```
F(x, y, z) = (x z)' (x'y')'
```

```
varList =

xzy


TT =

     0     1     0     1     0     1     0     1
     0     0     1     1     0     0     1     1
     0     0     0     0     1     1     1     1


ans =

     0     1     0     0     1     1     1     0


TTr =

     0     1     0     0     1     1     1     0
```

TTr is the truth table compute from my program, ans is the truth table compute directly from the function, they are the same, this verify my program works correctly.

Note that the order of x, y, z in TT is not conventional order x, y, z but rather x, z, y, showing in varList. This is the order that variables appear in the circuit.

Problem 2. Counting Networks:

(a) choose $t = \log_2(n)$.

$$Merge[2^t] = Merge[2^{t-1}] + 1$$

$$Merge[1] = Merge[2^0] = 0$$

$$Merge[2^t] = t$$

So $Merge[n] = \log_2(n)$

(b) $Batcher[2^t] = Batcher[2^{t-1}] + Merge[2^t]$

$$B[2^t] = B[2^{t-1}] + t$$

$$B[2^{t-1}] = B[2^{t-2}] + t-1$$

$$\vdots$$

$$B[2^1] = B[2^0] + 1 = Batcher[1] + 1 = 0 + 1 = 1$$

$$Batcher[2^t] = \frac{(1+t)t}{2} = \frac{(1+\log_2(n))\log_2(n)}{2}$$

Thus $Batcher[n] = ((\log_2(n))^2 + \log_2(n)) \times \frac{1}{2}$

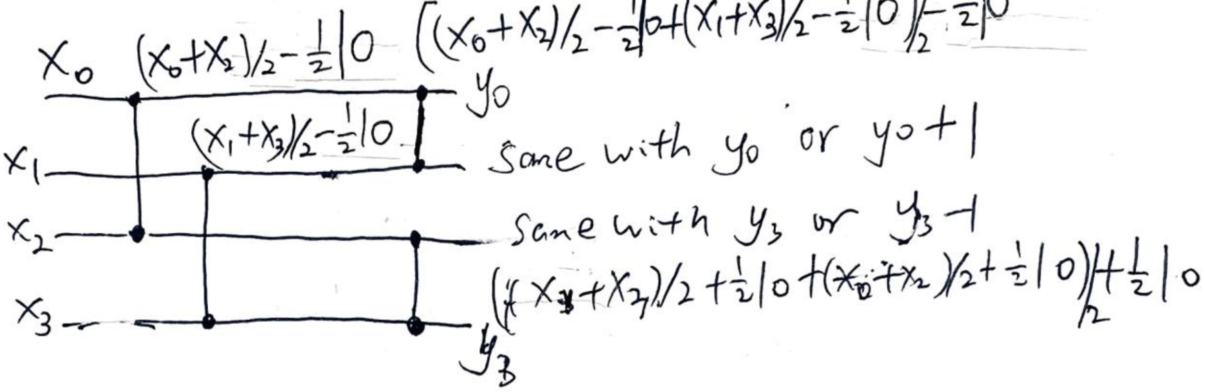
is $O((\log_2(n))^2)$

Problem 3.

(a) if the input are 7,3,2,6 output

y_0 y_1 y_2 y_3 are 4, 4, 5, 5

(b)



$-\frac{1}{2}|0$ this represent $-\frac{1}{2}$ or 0.

the smallest number y_0 can get is.

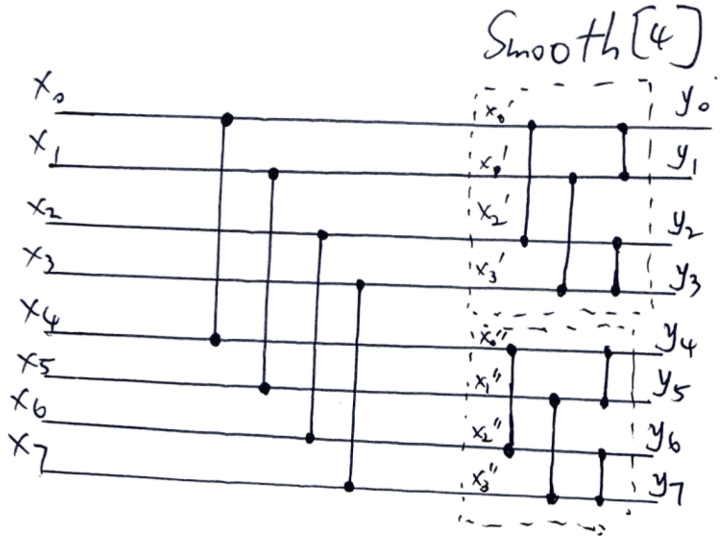
$$((x_0+x_2)/2-\frac{1}{2}+(x_1+x_3)/2-\frac{1}{2})/2-\frac{1}{2}=\frac{1}{4}(x_1+x_2+x_3+x_4)-1$$

the largest number y_3 can get is

$$((x_0+x_2)/2+\frac{1}{2}+(x_1+x_3)/2+\frac{1}{2})/2+\frac{1}{2}=\frac{1}{4}(x_1+x_2+x_3+x_4)+1$$

The largest difference of y_0 y_3 is 2.

(c)



Smooth [4] produce the average of 4 input.

y_0 can be 1 less than the average.

y_3 can be 1 larger than the average.

the input to the upper smooth [4] is

$(x_0+x_4)/2 - \frac{1}{2}|0$, $(x_1+x_5)/2 - \frac{1}{2}|0$, ... $(x_3+x_7)/2-\frac{1}{2}|0$

the input to the lowwer smooth [4] is.

$(x_0+x_4)/2+\frac{1}{2}|0$, $(x_1+x_5)/2+\frac{1}{2}|0$, ... $(x_3+x_7)/2+\frac{1}{2}|0$

Suppose all $-\frac{1}{2}$ take for these 4 term

then all $+\frac{1}{2}$ take as well.

the average of upper 4 is 1 less then lower 4.

And y_0 can be 1 less then the average of upper 4 and y_7 can be 1 greater than the average of lower 4. Thus maximum $y_7 - y_0$ is 3

(d) follow the prove in c
adding one more level at front
the maximum difference increase by 1.

$$\text{Smooth}[2^t] = \text{Smooth}[2^{t-1}] + 1.$$

$$\text{Smooth}[2^{t-1}] = \text{Smooth}[2^{t-2}] + 1.$$

$$\cdots$$

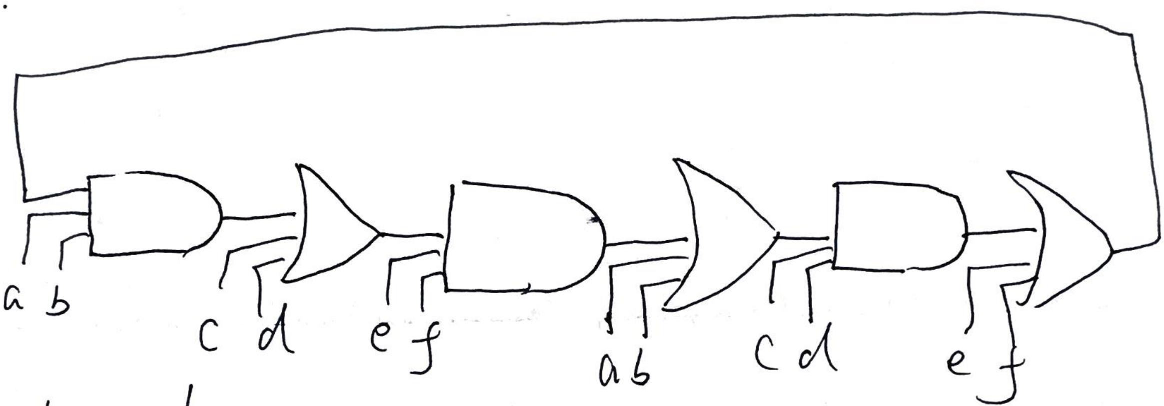
$$\text{Smooth}[2] = 1$$

Thus $\text{Smooth}[2^t] = t.$

$$\text{Smooth}[n] = \log_2 n$$

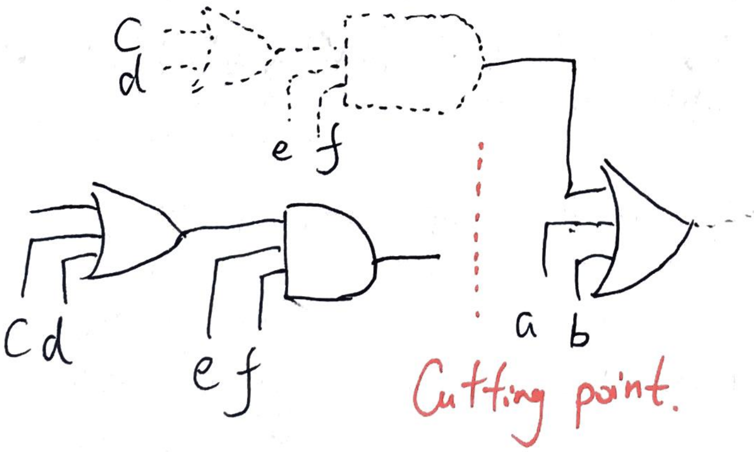
Problem 4.

(a).



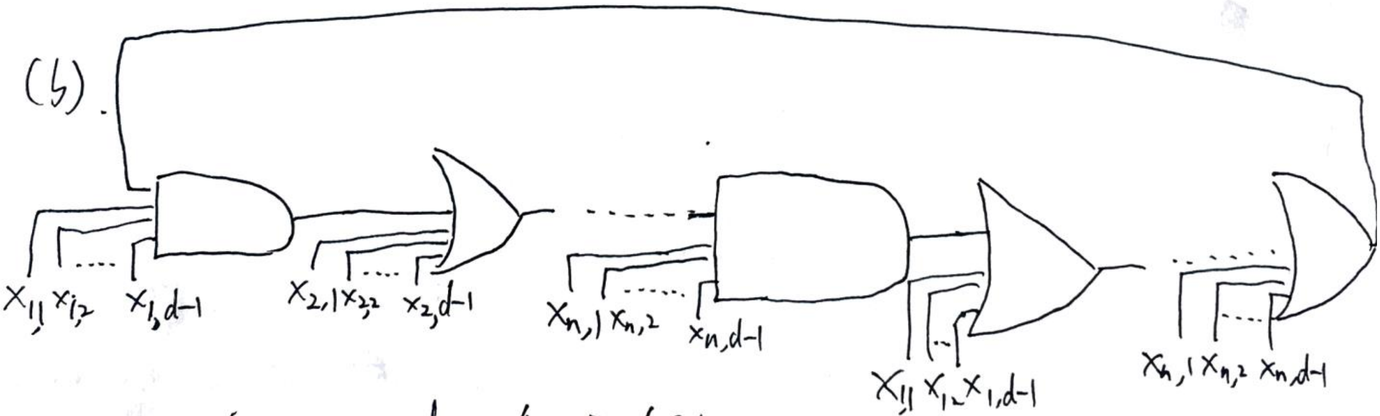
Acyclic circuit produce same function will require 8 fan-in 3 gates (or 7 fan-in 3 and 1 fan-in 2 gates).

Suppose we cut the cycle at any output of a gate.



Cutting at any point will need to duplicate last two gates in the cycles, the output of these two added gate only depend on 2/4 variables is not any of the 6 function. And we won't able to save any gates from adding dependency to these two added gates since once function already require one more gate.

(b).



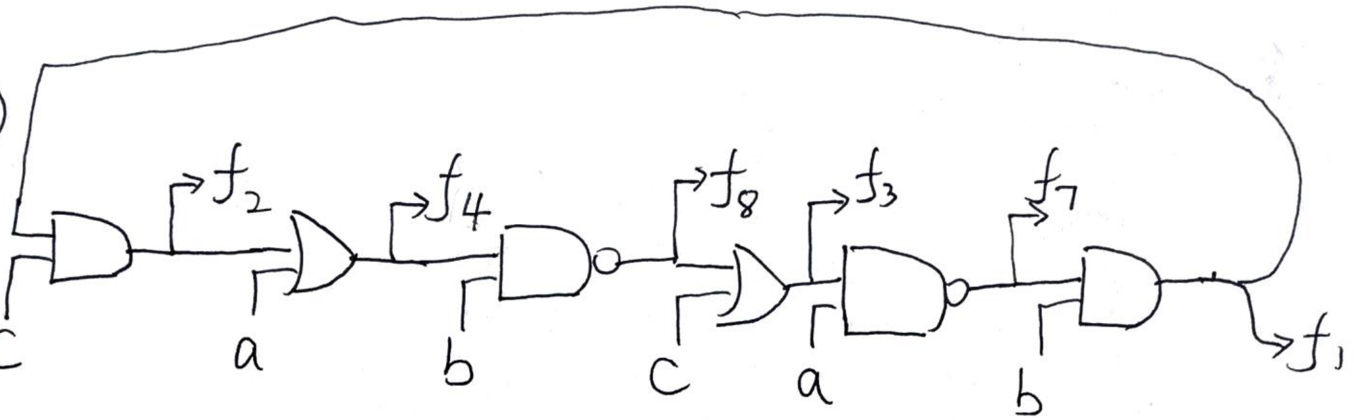
$2n$ fan-in d AND/OR gate produce $2n$ function depend on all $n(d-1)$ variables.

For acyclic implementation

For $d>2$: $3n-1$

For $d=2$: $3n-2$, saving one more because a fan-in 2 gate with only 1 input could be delete

(c)



(d) Prove this circuit is combinational

For $a=0$, 0 is controlling value to the NAND gate produce f_7

For $a=1$, 1 is controlling value to the OR gate produce f_4