

# System I Assignment

1-1

\*Convert the following numbers from the given base to the other three bases listed in the table:

Decimal	Binary	Octal	Hexadecimal
369.3125	? 101110001.010	? 561.24	? 171.5
? 189.625	10111101.101	? 275.5	? BD.A
? 214.625	? 1101010.101	326.5	? D6.A
? 62407.625	? 1110011100011.101	? 171707.5	F3C7.A

1-2 In each of the following cases, determine the radix r:

$$(a) (BEE)r = (2699)_{10}$$

$$(b) (365)r = (194)_{10}$$

$$(a) 11r^2 + 14r + 14 = 2699 \Rightarrow r = 15.$$

$$(b) 3r^2 + 6r + 5 = 194 \Rightarrow r = 7.$$

For the Boolean functions  $E$  and  $F$ , as given in the following truth table:

X	Y	Z	E	F
0	0	0	0	1
0	0	1	1	0
0	1	0	1	1
0	1	1	0	0
1	0	0	1	1
1	0	1	0	0
1	1	0	1	0
1	1	1	0	1

- (a) List the minterms and maxterms of each function.
- $$E = \sum m(1, 2, 4, 6) = \overline{M}(0, 3, 5, 7)$$
- (b) List the minterms of  $\bar{E}$  and  $\bar{F}$
- $$\bar{F} = \sum m(0, 2, 4, 7) = \overline{M}(1, 3, 5, 6)$$
- (c) List the minterms of  $E + F$  and  $E \cdot F$ .
- $$(b) \bar{E} = \sum m(0, 3, 5, 7), \bar{F} = \sum m(1, 3, 5, 6)$$
- (d) Express  $E$  and  $F$  in sum-of-minterms algebraic form.
- (e) Simplify  $E$  and  $F$  to expressions with a minimum of literals.

(f)  $E + F = \sum m(0, 1, 2, 4, 6, 7)$ ,  $E \cdot F = \sum m(2, 4)$ .

(g)  $E = \bar{X}\bar{Y}Z + \bar{X}Y\bar{Z} + X\bar{Y}\bar{Z} + XY\bar{Z}$ ,  $F = \bar{X}\bar{Y}\bar{Z} + \bar{X}Y\bar{Z} + X\bar{Y}\bar{Z} + XY\bar{Z}$

2-2 Question: (e)  $E = \bar{X}\bar{Z} + Y\bar{Z} + \bar{X}\bar{Y}Z$ ,  $F = \bar{X}\bar{Z} + \bar{Y}\bar{Z} + XY\bar{Z}$

\*Find all the prime implicants for the following Boolean functions, and determine which are essential:

(a)  $F(W, X, Y, Z) = \sum m(0, 2, 5, 7, 8, 10, 12, 13, 14, 15)$   
Prime implicants:  $W\bar{Z}, \bar{X}\bar{Z}, W\bar{X}, X\bar{Z}$

2-3 Question: essential:  $\bar{X}\bar{Z}, \bar{X}\bar{Z}$

Prove the identity of the following Boolean equation (a), using algebraic manipulation

(只做 a):

(a)  $A\bar{B}\bar{C} + B\bar{C}\bar{D} + BC + \bar{C}D = B + \bar{C}D$

(b)  $WY + \bar{W}YZ + WXZ + \bar{W}XY = WY + \bar{W}X\bar{Z} + \bar{X}Y\bar{Z} + X\bar{Y}Z$

(c)  $\bar{A}\bar{D} + \bar{A}B + \bar{C}D + \bar{B}C = (\bar{A} + \bar{B} + \bar{C} + \bar{D})(A + B + C + D)$

(d)  $= AB\bar{C} + B\bar{C}\bar{D} + BCD + BC\bar{D} + B\bar{C}D + \bar{C}D$

$= AB\bar{C} + \underbrace{B\bar{C}\bar{D} + BD}_{= B\bar{D}} + \bar{C}D = AB\bar{C} + B + \bar{C}D = B + \bar{C}D$ .

2-4 Question

\*Optimize the following Boolean functions  $F$  together with the don't-care conditions  $d$ . Find all prime implicants and essential prime implicants, and apply the selection rule.

(b)  $F(W, X, Y, Z) = \sum m(0, 2, 4, 5, 8, 14, 15), d(W, X, Y, Z) = \sum m(7, 10, 13)$

Prime implicants:  $\bar{X}\bar{Z}, W\bar{X}Y, WY\bar{Z}, \bar{X}\bar{Z}, \bar{W}X\bar{Y}$ ,  
 $\bar{W}\bar{Y}\bar{Z}$ .

essential:  $\bar{X}\bar{Z}$ .

$F = \cancel{\bar{X}\bar{Z}} + \bar{X}\bar{Z} + \bar{W}X\bar{Y} + WXY$ .

## 4-1

**3.20** [5] <§3.5> What decimal number does the bit pattern  $0 \times 0C000000$  represent if it is a two's complement integer? An unsigned integer?

**3.21** [10] <§3.5> If the bit pattern  $0 \times 0C000000$  is placed into the Instruction Register, what MIPS instruction will be executed?

**3.22** [10] <§3.5> What decimal number does the bit pattern  $0 \times 0C000000$  represent if it is a floating point number? Use the IEEE 754 standard.

**3.23** [10] <§3.5> Write down the binary representation of the decimal number 63.25 assuming the IEEE 754 single precision format.

**3.24** [10] <§3.5> Write down the binary representation of the decimal number 63.25 assuming the IEEE 754 double precision format.

## 4-2

**3.27** [20] <§3.5> IEEE 754-2008 contains a half precision that is only 16 bits wide. The leftmost bit is still the sign bit, the exponent is 5 bits wide and has a bias of 15, and the mantissa is 10 bits long. A hidden 1 is assumed. Write down the bit pattern to represent  $-1.5625 \times 10^{-1}$  assuming a version of this format, which uses an excess-16 format to store the exponent. Comment on how the range and accuracy of this 16-bit floating point format compares to the single precision IEEE 754 standard.

4-1 3.20 均为  $12 \times 16^6 = 201326592$ .

3.21 jal 0x00000000

3.22  $0 \times 0CD00000 = \underline{0000} \underline{1100} \underline{0000} \underline{0000} \underline{0000} \underline{0000} \underline{0000} \underline{0000}$

$\exp = 24 - 12 = -103$

$1.0 \times 2^{-103}$

3.23.  $63.25 = 1.111101 \times 2^5$

0 10000100 1111010000 00000000 0000

3.24

0 10000000100 1111010000 00000000 00000000 00000000 00000000

4-2 3.27  $-0.15625 = -0.00101 \times 2^0 = -1.01 \times 2^{-3}$

1 0110001000000000

精度不够高,  $2^{-14} \cdot 2^{-10} = 2^{-24}$

6-1

**2.12** [5] <§§2.2, 2.5> Provide the instruction type and assembly language instruction for the following binary value:

0000 0000 0001 / 0000 1000 / 0000 1011 0011<sub>two</sub>  
 R-type add x1, x1, x1  
 6-2

**2.13** [5] <§§2.2, 2.5> Provide the instruction type and hexadecimal representation of the following instruction:

sd x5, 32(x7) 1111 0011 0000 001000011  
 S-type 0x025F3023  
 6-3

**2.31** [20] <§2.8> Translate function f into RISC-V assembly language. Assume the function declaration for g is int g(int a, int b). The code for function f is as follows:

```
int f(int a, int b, int c, int d){  

    return g(g(a,b), c+d);  

}
```

6-4

**2.35** Consider the following code:

```
lb x6, 0(x7)  

sd x6, 8(x7)
```

f:  
 addi x2, x2, -16  
 sd x1, 0(x2)  
 add x5, x12, x13  
 sd x5, 8(x2)  
 jal x1, g  
 ld x11, 8(x2)  
 jal x1, g  
 ld x1, 0(x2)  
 addi x2, x2, 16  
 jalr x0, 0(x1)

Assume that the register x7 contains the address 0x10000000 and the data at address is 0x1122334455667788.

**2.35.1** [5] <§2.3, 2.9> What value is stored in 0x10000008 on a big-endian machine? 0x11

**2.35.2** [5] <§2.3, 2.9> What value is stored in 0x10000008 on a little-endian machine? 0x88

6-6

**2.36** [5] <§2.10> Write the RISC-V assembly code that creates the 64-bit constant 0x1122334455667788<sub>two</sub> and stores that value to register x10.

```
lui x10, 0x11223  

addi x10, x10, 0x344  

lui x5, 0x55667  

addi x5, x5, 0x788  

add x10, x10, x5
```

slli x10, x10, 32

**4.13** Examine the difficulty of adding a proposed ss rs1, rs2, imm (Store Sum) instruction to RISC-V.

Interpretation:  $\text{Mem}[\text{Reg}[rs1]] = \text{Reg}[rs2] + \text{immediate}$

**4.13.1** [10] <§4.4> Which new functional blocks (if any) do we need for this instruction?

**4.13.2** [10] <§4.4> Which existing functional blocks (if any) require modification?

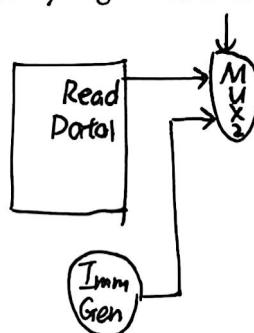
**4.13.3** [5] <§4.4> What new data paths do we need (if any) to support this instruction?

**4.13.4** [5] <§4.4> What new signals do we need (if any) from the control unit to support this instruction?

**4.13.5** [5] <§4.4> Modify Figure 4.21 to demonstrate an implementation of this new instruction.

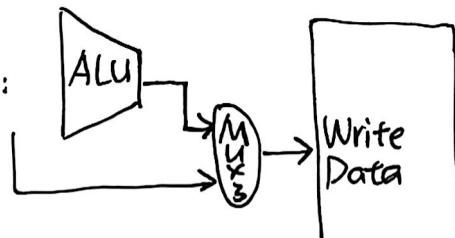
4.13.1

新增 MUX2:

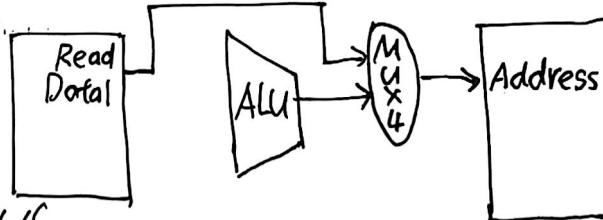


4.13.3  
& 4.13.5

新增 ALU 到 Write Data 的连线:  
并新增 MUX3.



新增 MUX4.



4.13.2 没有需要修改的。

4.13.4 新增的这些 MUX 需要来自 CONTROL 的控制信号。

# System I Assignment

## 3-1

3-1

A hierarchical component with the function is to be used along with inverters to implement the following equation:

$$H = \bar{X}Y + XZ$$
$$G = \bar{A}\bar{B}C + \bar{A}BD + A\bar{B}\bar{C} + AB\bar{D}$$

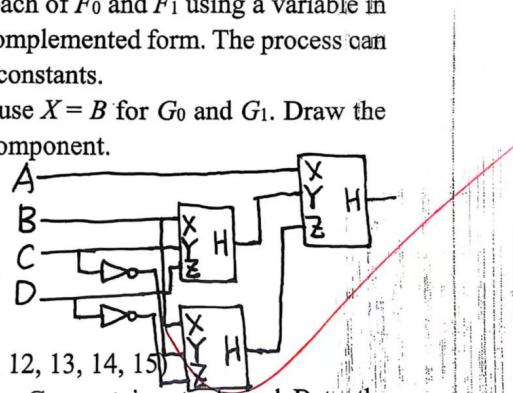
The overall circuit can be obtained by using **Shannon's expansion** theorem,

$$F = \bar{X}F_0(X) + XF_1(X)$$

where  $F_0(X)$  is  $F$  evaluated with variable  $X = 0$  and  $F_1(X)$  is  $F$  evaluated with variable  $X = 1$ . This expansion  $F$  can be implemented with function  $H$  by letting  $Y = F_0$  and  $Z = F_1$ . The expansion theorem can then be applied to each of  $F_0$  and  $F_1$  using a variable in each, preferably one that appears in both true and complemented form. The process can then be repeated until all  $F_i$ 's are single literals or constants.

For  $G$ , use  $X = A$  to find  $G_0$  and  $G_1$  and then use  $X = B$  for  $G_0$  and  $G_1$ . Draw the top-level diagram for  $G$  using  $H$  as a hierarchical component.

$$G = \bar{A}(\bar{B}C + BD) + A(\bar{B}\bar{C} + B\bar{D}).$$



3-2

Implement the Boolean function

$$F(A, B, C, D) = \sum m(1, 3, 4, 11, 12, 13, 14, 15)$$

## 3-2

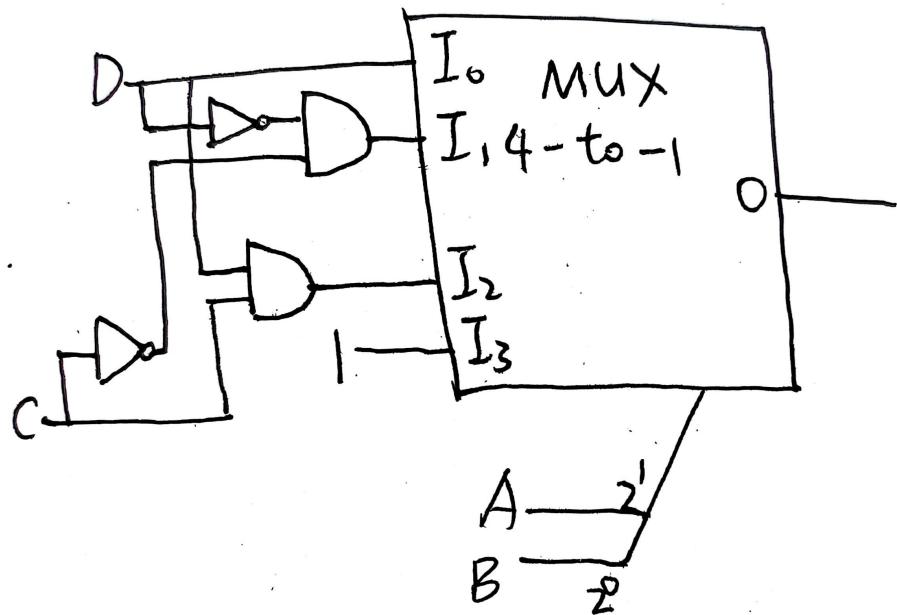
A 和 B 作为 selection lines, C 和 D 作为 MUX 4-to-1 的输入。

当  $A = 0, B = 0$  时,  $I_0 = \sum m(1, 3) = D$ ;

当  $A = 0, B = 1$  时,  $I_1 = \sum m(4) = \bar{C}\bar{D}$ ;

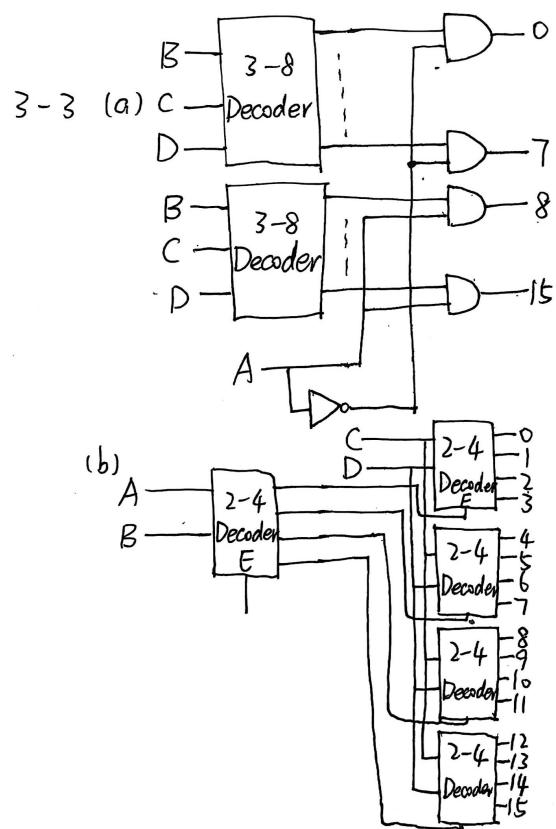
当  $A = 1, B = 0$  时,  $I_2 = \sum m(11) = CD$ ;

当  $A = 1, B = 1$  时,  $I_3 = \sum m(12, 13, 14, 15) = 1$ .

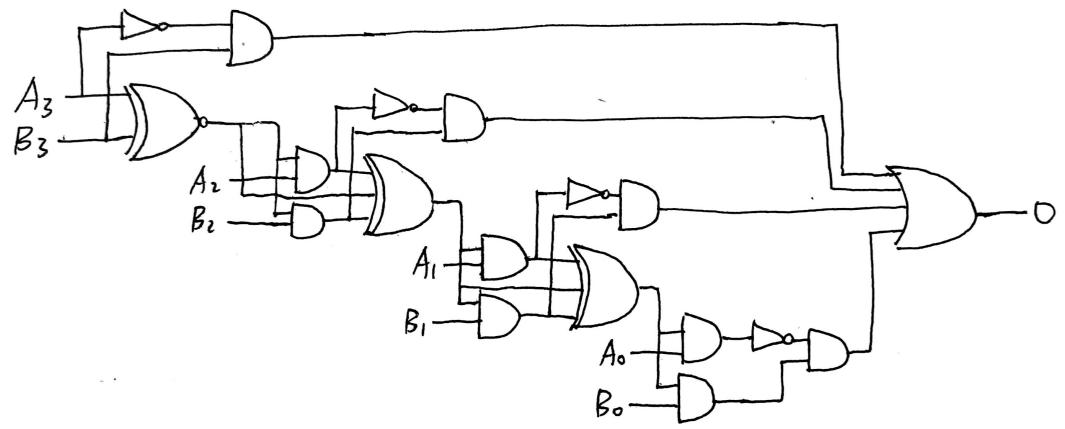


### 3-3

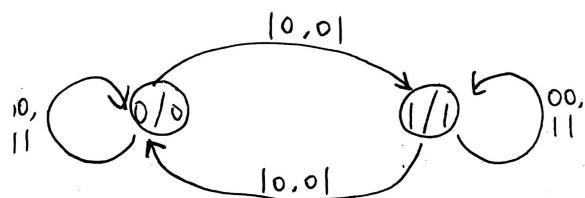
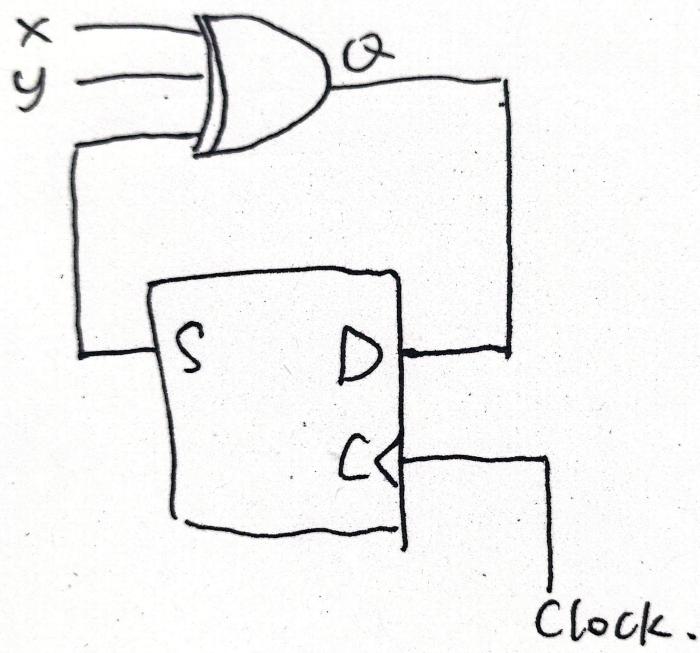
注: (a) 如果说不允许用非门, 那底下的设计可以将 A 的信号单独用一个 3-to-8 Decoder, 然后将其跟与门连接即可。



### 3-4



5-1

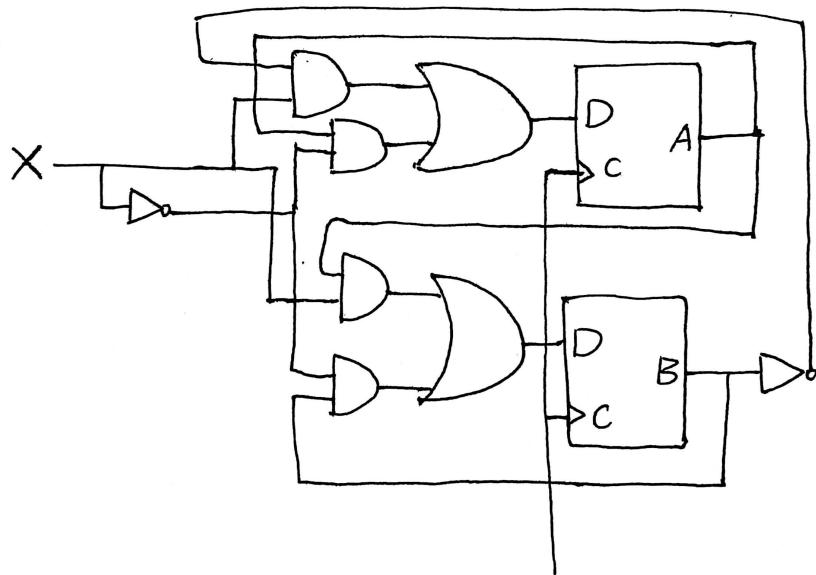


Present State	input $x$	input $y$	next state	Output
0	0	0	0	0
0	0	1	1	1
0	1	0	1	1
0	1	1	0	0
1	0	0	1	1
1	0	1	0	0
1	1	0	0	0
1	1	1	1	1

注：按照题意应该是 Moore 状态机，Output 为  $S = \text{Present State}$ ，因此 Output 那一栏之前写错了，应该分别为 00001111。

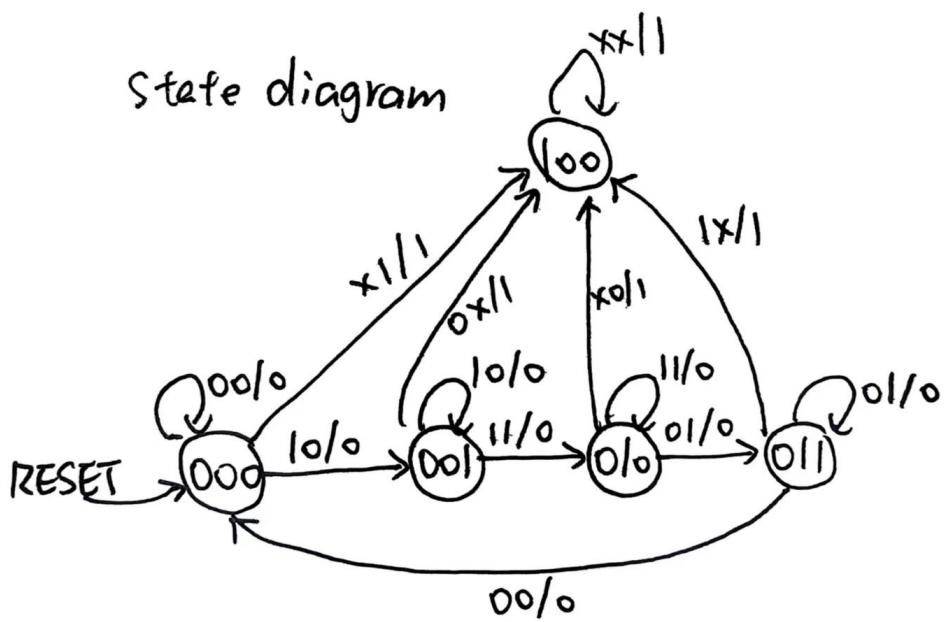
$$P_A^+ = X \bar{B} + \bar{X} A$$

$$D_B^+ = X A + \bar{X} B$$



### 5-3

状态转移表示 RS/E，状态分别表示四个阶段以及错误阶段。



当前状态	R	A	RESET	下一状态	E
<b>S0</b>	0	0	0	S0	0
S0	1	0	0	S1	0
S0	0	1	0	S_error	1
S0	1	1	0	S_error	1
<b>S1</b>	1	0	0	S1	0
S1	1	1	0	S2	0
S1	0	0	0	S_error	1
S1	0	1	0	S_error	1
<b>S2</b>	1	1	0	S2	0
S2	0	1	0	S3	0
S2	1	0	0	S_error	1
S2	0	0	0	S_error	1
<b>S3</b>	0	1	0	S3	0
S3	0	0	0	S0	0
S3	1	1	0	S_error	1
S3	1	0	0	S_error	1
<b>S_error</b>	X	X	0	S_error	1
任意状态	X	X	1	S0	0

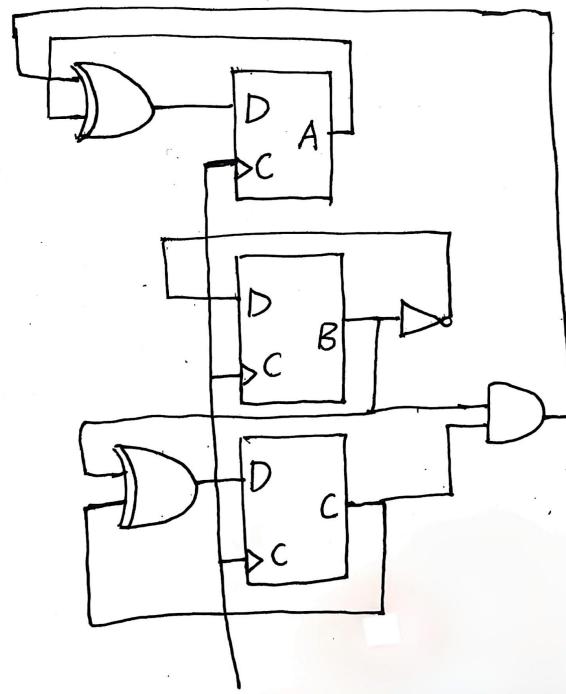
## 5-4

记状态为 ABC (即 A 为最高位, B 为次高位, C 为最低位) ,

$$A' = A \oplus (B \& C)$$

$$B' = \overline{B}$$

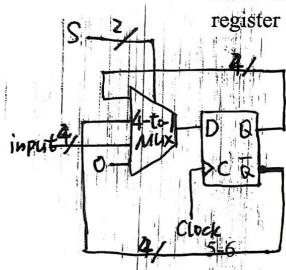
$$C' = B \oplus C$$



## 5-5, 5-6

5-5

Draw the logic diagram of a 4-bit register with mode selection inputs  $S_1$  and  $S_0$ . The register is to be operated according to the function table below.



$S_1$	$S_0$	Register Operation
0	0	No change
0	1	Complement output
1	0	Load parallel data
1	1	Clear register to 0

A register cell is to be designed for an 8-bit register  $A$  that has the following register transfer functions:

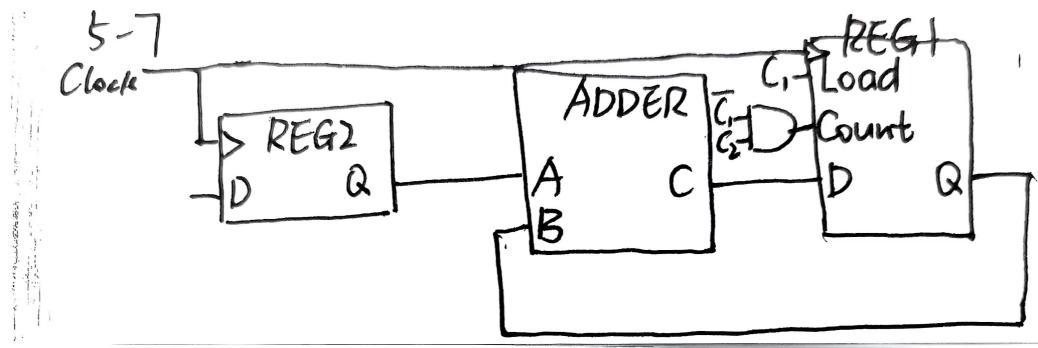
$$C_0: A \leftarrow A \cap B, C_1: A \leftarrow A \cup \overline{B}$$

Find optimum logic using AND, OR, and NOT gates for the  $D$  input to the  $D$  flip-flop in the cell.

$$D_A = AB + \overline{C}_0 C_1 \overline{B} + C_1 A + \overline{C}_0 A$$



## 5-7



## 5-8

- (a)  $t_1 = 0.04 + 0.04 = 0.08ns$
- (b)  $t_2 = 0.04 + 0.01 + 0.02 = 0.07ns$
- (c)  $t_3 = 0.08 + 2 * 0.04 = 0.16ns$
- (d)  $t_4 = 0.08 + 0.04 + 0.01 + 0.02 = 0.15ns$
- (e)  $f = \frac{1}{t_4} = 6358MHz$