

학과		학번		반		이름	
----	--	----	--	---	--	----	--

1. 아래의 문장들에 대하여 O/X로 답하라(각 2점). 단 틀리면 1점 감점이 있음.
- (1) NAND(Not AND) gate의 한쪽 입력 값을 1로 고정하면 출력은 항상 다른 쪽 입력 값의 Complement가 된다.
- (2) 하나의 Truth table에 대응되는 Boolean expression은 하나이다.
- (3) ROM(Read-Only Memory)는 메모리 내의 특정 word에 접근하는데 소요되는 시간이 항상 동일하다.
- (4) RAM(Random Access Memory)는 메모리 내의 특정 word를 access하기 위한 Decoder를 포함하는 Combinational Circuit이다.
- (5) Fixed-point 표현에서 2의 보수인 경우가 가장 넓은 영역을 표현할 수 있다.
- (6) Floating point 표현의 경우 Mantissa field의 크기를 늘리면 두 정수 사이에 표현 할 수 있는 수의 가지 수가 늘어나게 된다.
- (7) 2의 보수를 shift left 연산할 때 overflow는 serial output으로 알아낼 수 있다.
- (8) Arithmetic shift left에서 serial input은 sign bit, serial output은 2로 나눈 나머지 이다.
- (9) Gray-code는 각 비트의 위치가 값을 지닌 weighted code이다.
- (10) 12×4096 디코더를 4×16 의 디코더를 이용하여 설계하려면, 4×16 의 디코더는 최소한 256개 필요하다.
- (11) Demultiplexer(DeMUX)를 따로 설계할 필요 없이 Multiplexer(MUX)의 입력과 출력을 바꾸어서 사용하면 된다.
- (12) Excitation table은 flip-flop의 상태 변화를 위한 입력 값을 정의한다.
- (13) 레지스터들을 serial로 연결하려면 레지스터들이 shift 레지스터여야 한다.
- (14) 버스를 3-state buffer gate를 이용하여 설계할 때, 레지스터의 입력과 출력 단에 3-state buffer gate를 달고, 2개의 Decoder 출력이 3-state buffer gate를 제어 한다.
- (15) 버스를 MUX와 DEC를 이용하여 설계할 때, MUX의 개수는 레지스터의 크기에 따라 달라진다.
- (16) A 레지스터에 4, B 레지스터의 모든 비트들은 1이 저장된 상태에서 두 레지스터를 더한 결과는 3이 된다.
- (17) PLA(Programmable Logic Array)를 이용하여 모든 Circuit을 설계할 수 있다.
- (18) RTL(Register Transfer Language)로 작성된 프로그램을 수행한 결과는 하드웨어 회로도이다.
- (19) 전송 중 한 비트가 변할 확률을 $\frac{1}{2}$ 이라고 할 때, Parity Bit를 포함하여 4비트를 전송하는 경우 에러를 검출하지 못할 확률은 $\left(\frac{1}{2}\right)^4 = \frac{1}{16}$ 이다.
- (20) A 레지스터의 특정 비트 값을 1로 만들려면, B 레지스터의 해당 위치에 1을 주고 나머지 비트들은 0을 저장한 후에 두 레지스터를 XOR한다.

1	2	3	4	5	6	7	8	9	10
0	X	0	X	0	0	X	X	X	X
11	12	13	14	15	16	17	18	19	20
X	0	0	0	0	0	X	0	X	X

2. 10진수 $X=56$ 와 $Y=50$ 가 있다. 컴퓨터를 이용하여 $X-Y$ 의 뺄셈 연산을 수행하려고 한다. 컴퓨터는 2의 보수 연산과 덧셈연산 밖에 수행할 수 없다. 2의 보수와 덧셈을 이용하여 $X-Y$ 연산을 수행하고 그 결과 값을 나타내어라. (2점)

$$X + (Y' + 1) = 0111000 + 1001110 = 0000110 = 110_{(2)}$$

3. K-map을 이용하여 함수 F에 대한 간소화(simplify)된 식을 유도하라.

- (1) $F(A, B, C, D, E) = \sum(0, 2, 5, 7, 8, 10, 13, 15, 16, 18, 21, 23, 24, 26, 29, 31)$ (2점)

		E=0				E=1				
		CD AB	00	01	11	10	00	01	11	10
00	00	1	1					1	1	
01	01	1	1					1	1	
11	11	1	1					1	1	
10	10	1	1					1	1	

녹색 부분 simplify: $C'E'$
 붉은색 부분 simplify: CE
 그러므로 $C'E' + CE$

(2) 위 (1)의 결과 값을 드모르간 법칙(DeMorgan's Theorem)을 이용하여 XNOR(Exclusive NOR)로 유도될 수 있다. 위의 $F(A,B,C,D,E)$ 결과 값을 XNOR로 유도하는 과정을 기술하라. (2점)

$$\begin{aligned}
 CE + C'E' &= (C+E')' + (C+E)' \\
 &= ((C+E')(C+E))' \\
 &= (CD' + EE' + C'E + E'C)' \\
 &= (C'E + CE')' \\
 &= (C \oplus E)'
 \end{aligned}$$

(3) 위 (1)의 F 에서 다음 don't-care condition을 추가하였을 때 간소화된 식을 유도하라. $d(A, B, C, D, E) = \sum(4, 6, 20, 22)$ (2점)

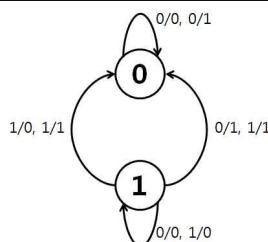
		E=0				E=1			
		00	01	11	10	00	01	11	10
CD AB	00	1	1	x	x			1	1
	01	1	1					1	1
	11	1	1					1	1
	10	1	1	x	x			1	1

변화 없음

$$C'E' + CE$$

4. SR flip-flop을 이용하여 JK Flip-flop을 아래 과정을 통해 설계하라.

(1) JK Flip-flop을 위한 state diagram을 그려라. (2점)



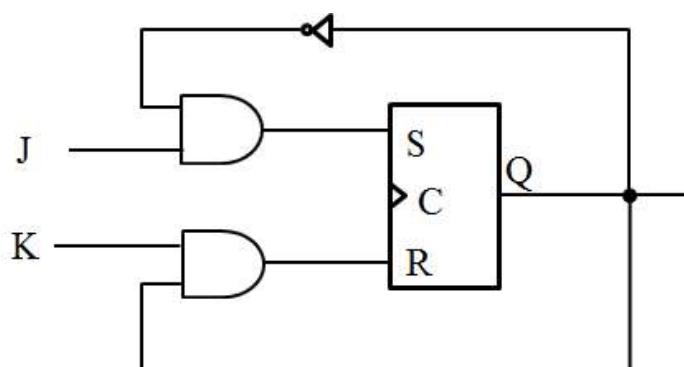
(2) State Transition table을 SR flip-flop의 Excitation table(여기표)를 이용하여 나타내라. (2점)

Present state	Input		Next state	Flip-flop inputs	
	J	K		S	R
Q			Q(t+1)		
0	0	0	0	0	X
0	0	1	0	0	X
0	1	0	1	1	0
0	1	1	1	1	0
1	0	0	1	X	0
1	0	1	0	0	1
1	1	0	1	X	0
1	1	1	0	0	1

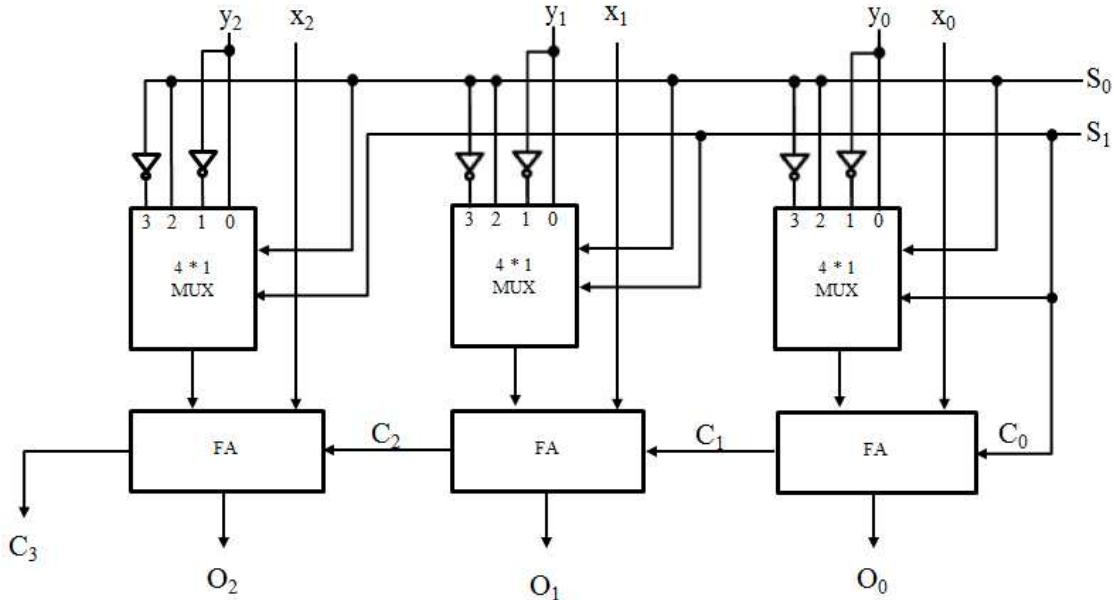
(3) JK flip-flop 설계를 위한 K-map을 그리고 식을 유도하여라. (각 2점)

S		R		
Q \ JK	00	01	11	10
0			1	1
1	x			x
$= Q'J$		$= QK$		

(4) 최종 설계를 diagram으로 표현하라. (2점)



5. 아래 회로는 3 비트 크기인 두 2의 보수($x_2x_1x_0$, $y_2y_1y_0$)에 대해 4가지 산술연산을 수행하는 회로이다.



(1) 다음의 각 경우 위 회로의 기능을 설명하라. (각 2점)

S_0 / S_1	기능
0 0	$X+Y$
0 1	$X-Y$
1 0	$X-1$
1 1	$X+1$

(2) 위 회로의 전체 Propagation delay 값을 줄이기 위해서, 왼쪽 비트들의 연산에 오른쪽 비트들의 연산에서 발생하는 carry 값을 사용하지 않아야한다. 세 번째 비트의 연산에 필요한 Carry(C_2)를 C_1 이 없는 식으로 표현하라.(4점)

$C_2 =$	$x_1y_1 + (x_1 \oplus y_1)(x_0y_0 + y_0C_0 + x_0C_0)$ $(x_1 \oplus y_1)$ 을 $(x_1 + y_1)$ 로 해도 정답
---------	---

6. Hamming이 고안한 Error Correction Code를 설계하고자 한다. 데이터의 크기는 4비트(x_0-x_3)이다.

(1) 이때 필요한 추가 코드는 최소 3비트(c_0-c_2)이다. 그 이유를 쓰라(2점).

$$D + P + 1 \leq 2^P \text{ (D: data bits, P: parity bits)}$$

$$4 + 2 + 1 \leq 2^2, 4 + 3 + 1 \leq 2^3$$

(2) 3비트의 추가 Code 생성을 위한 Boolean식을 완성하라(각2점).

$c_0 =$	$x_0 \oplus x_1 \oplus x_3$
$c_1 =$	$x_0 \oplus x_2 \oplus x_3$
$c_2 =$	$x_1 \oplus x_2 \oplus x_3$

(3) Receiver가 받은 c_0-c_2 와 전달된 x_0-x_3 를 이용하여 다시 계산한 $c_0^* - c_2^*$ 값을 비교하였을 때, 아래의 결과는 어떤 의미인가?(각2점)

c_0-c_2	$c_0^*-c_2^*$	의미
1,1,0	1,1,0	에러가 발생하지 않았다.
0,0,0	0,1,1	6번째 위치(011)의 데이터비트에 오류
1,1,1	0,0,1	3번째 위치(110)의 데이터비트에 오류

	x	c0	c1	x0	c2	x1	x2	x3
c2	0	0	0	0	1	1	1	1
c1	0	0	1	1	0	0	1	1
c0	0	1	0	1	0	1	0	1

$$c2 = x_1 \oplus x_2 \oplus x_3$$

$$c1 = x_0 \oplus x_2 \oplus x_3$$

$$c0 = x_0 \oplus x_1 \oplus x_3$$

011 => 6번째 위치 데이터비트에 오류 : x2

110 => 3번째 위치 데이터비트에 오류 : x0

	x	c2	c1	x0	c0	x1	x2	x3
c0	0	0	0	0	1	1	1	1
c1	0	0	1	1	0	0	1	1
c2	0	1	0	1	0	1	0	1

$$c0 = x_1 \oplus x_2 \oplus x_3$$

$$c1 = x_0 \oplus x_2 \oplus x_3$$

$$c2 = x_0 \oplus x_1 \oplus x_3$$

011 => 3번째 위치 데이터비트에 오류 : x0

110 => 6번째 위치 데이터비트에 오류 : x2

	x	c2	c1	x3	c0	x2	x1	x0
c0	0	0	0	0	1	1	1	1
c1	0	0	1	1	0	0	1	1
c2	0	1	0	1	0	1	0	1

$$c0 = x_0 \oplus x_1 \oplus x_2$$

$$c1 = x_0 \oplus x_1 \oplus x_3$$

$$c2 = x_0 \oplus x_2 \oplus x_3$$

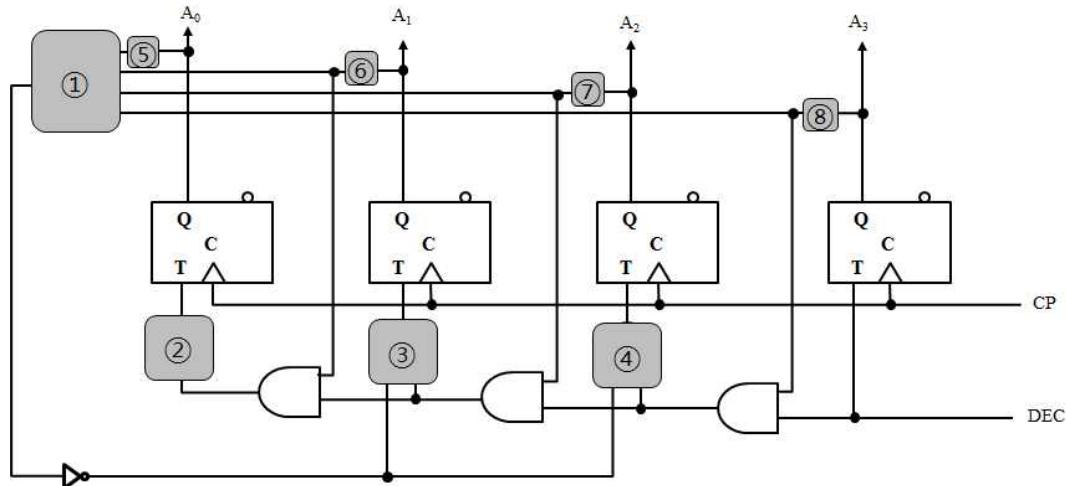
011 => 3번째 위치 데이터비트에 오류 : x3

110 => 6번째 위치 데이터비트에 오류 : x1

7. Decimal Counter를 설계하고자 한다.

(1) 아래의 Decimal Down Counter 회로의 빈칸에 들어갈 게이트 종류를 기입하라. ※ 게이트가 들어갈 필요가 없는 경우 X 기입 (각 1점).

①	AND	②	X	③	AND	④	AND
⑤	NOT	⑥	NOT	⑦	NOT	⑧	NOT



(2) 아래의 Decimal Up Counter 회로의 빈칸에 들어갈 게이트 종류를 기입하고
 (①~④), A₀~A₃까지 ①에 연결될 필요 있는 것은 O, 아닌 것은 X기입 하시오(⑤~⑧)
 ※ 게이트가 들어갈 필요가 없는 경우 X 기입 (각 1점).

①	AND	②	OR	③	NOT	④	AND
⑤	O	⑥	X	⑦	X	⑧	O

