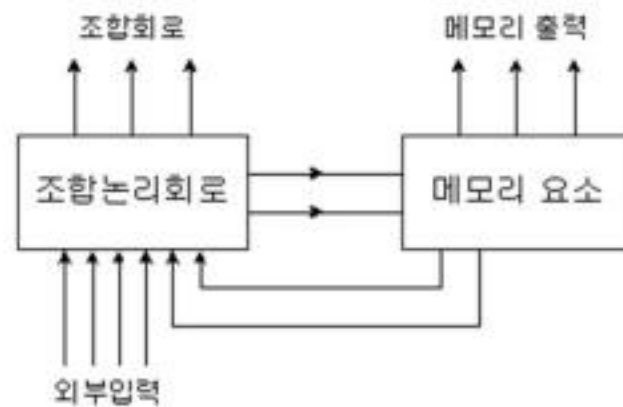


Sequential Circuits Design p.205



1

1

Sequential Circuits Design p.205

■ Design Procedure

1. specification
2. Formulation : 상태도 또는 상태표
3. State Assignment : 상태에 2진 코드 할당
4. FF input Equation Optimization
Output Equation Optimization
5. Technology Mapping
6. Verification

● 설계시 고려사항 :

- 적절한 FF의 선택,
- 요구되는 상태수에 따라 FF수의 결정 :
상태수 $2^n \rightarrow n$ 개의 FF필요

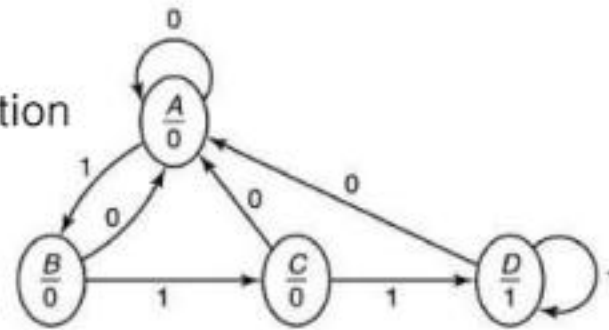
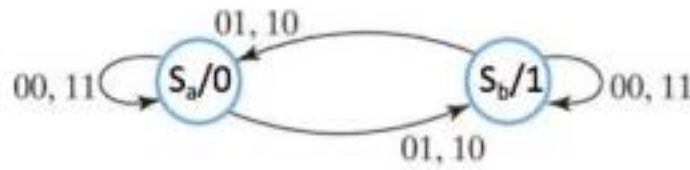
2

2

비교 참고) 조합회로 설계 과정

■ 순차회로 설계 Steps

1. specification : 기능
2. Formulation
: **상태도 또는 상태표**
3. State Assignment
: 상태에 2진 코드 할당
4. FF input Equation,
Output Equation Optimization
(2단계 최적화)



3

3

Example) Sequential Circuits Design

■ 문제에) 입력 x의 값이 1로 입력될 때마다

11→00→01→10→11→00→01...을 반복하는 회로를 작성하시오.

출력 : 상태가 10→11로 변환할 때만 1을 출력한다.

그외는 0를 출력한다.

단) 초기값은 11으로 둔다.

-이러한 설계는 조합회로로 구성할 수 없다.

-현재상태가 무엇인가에 따라 입력 X에 따라 다음값이 결정되기 때문이다.



5

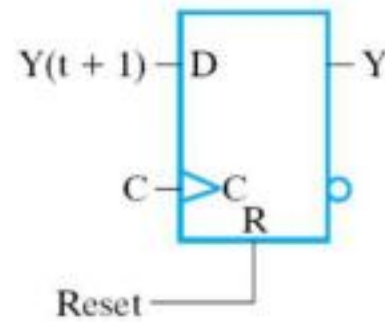
5

Master reset을 가진 FF

p.207

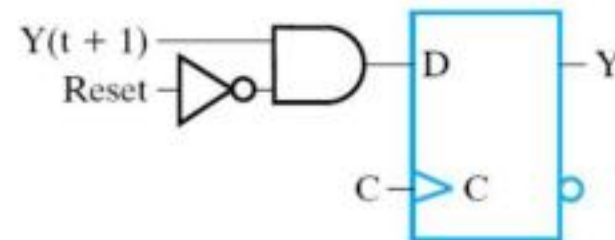
■ FF의 저장값을 초기값으로 set

- 전원이 On 되었을때
- 리셋 버튼에 의해



(a) Asynchronous Reset

direct input으로 reset



(b) Synchronous Reset

조합회로로 reset

6

6

State Diagram (1)

p.208

■ 예) "1101" sequence Recognizer

- 기능: 입력sequence에서 "1101"이 발생할 경우, 1출력
-001011010001110110100100....
 1출력 1출력

★이전의 입력에 대한 기억이 요구되므로 조합회로로는 설계할 수 없다.

- 입력 1개, 출력 1개
- 상태도 표현

이진수	"0"	"1"	"1"	"0"	"1"
상태명	0% A	25% B	50% C	75% D	100%

초기상태

25%

50%

75%

7

7

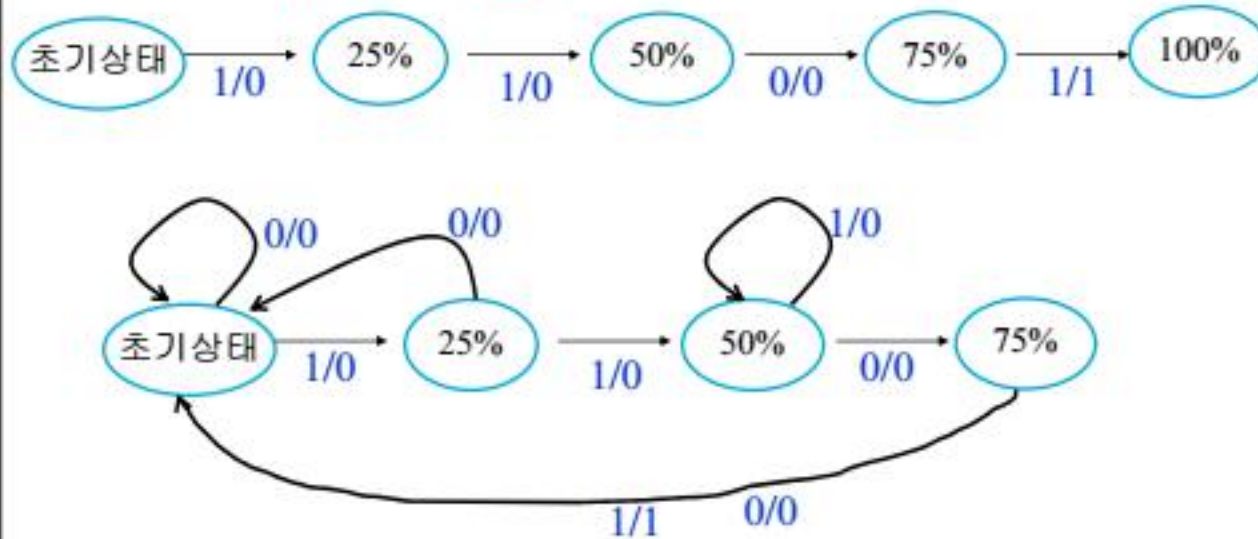
State Diagram (1)

p.208

예) "1101" sequence Recognizer

기능: 입력sequence에서 "1101"이 발생할 경우, 1출력

....001011010001110110100100....
 1출력 1출력



8

8

State Diagram (1)

p.208

예) "1101" sequence Recognizer

기능: 입력sequence에서 "1101"이 발생할 경우, 1출력(중복인식) ★★

....0010110100011101110100100....
 1출력 1 1출력

★이전의 입력에 대한 기억이 요구되므로 조합회로는 설계할 수 없다.

입력 1개, 출력 1개

상태도 표현

이진수	"0"	"1"	"1"	"0"	"1"
상태명	0%	25%	50%	75%	100%
	A	B	C	D	



9

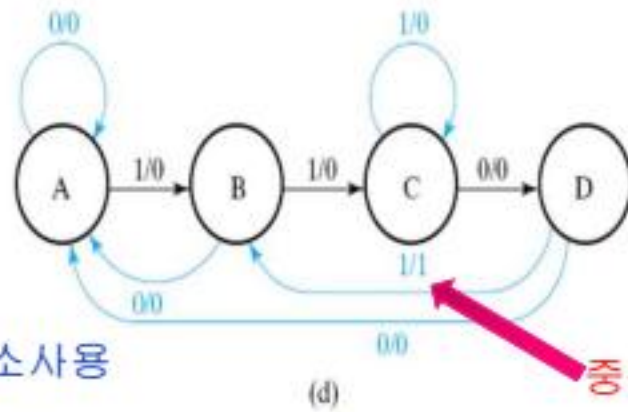
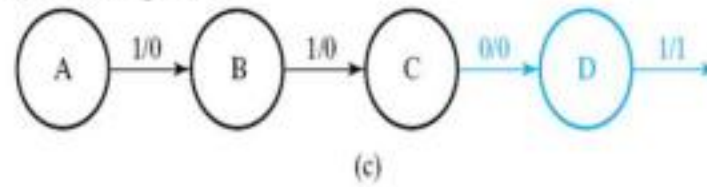
9

State Diagram (2)

p.208

예) "1101" sequence Recognizer

상태도 표현



상태의 최소화: FF의 최소사용

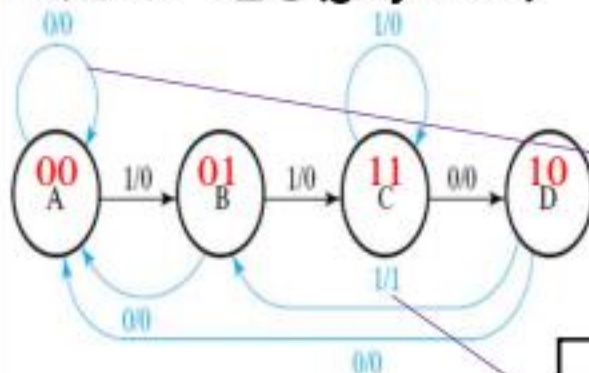
상태에 이진코드 할당 (00, 01, 10, 11)
또는 (00, 01, 11, 10)

10

10

상태도 작성(p.214)

이진 코드 할당(gray code)



상태 4개 → FF 몇 개 필요?

FF 종류 선택

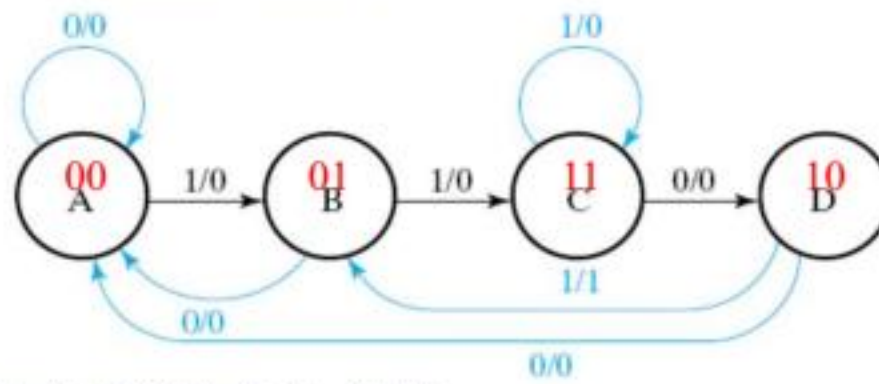
D FF을 이용하여 설계하자.

State(t)		Input X	State(t+1)		output Z
FFA	FFB		FFA	FFB	
0	0	0	0	0	0
0	0	1	0	1	0
0	1	0	0	0	0
0	1	1	1	1	0
1	0	0	0	0	0
1	0	1	0	1	1
1	1	0	1	0	0
1	1	1	1	1	0

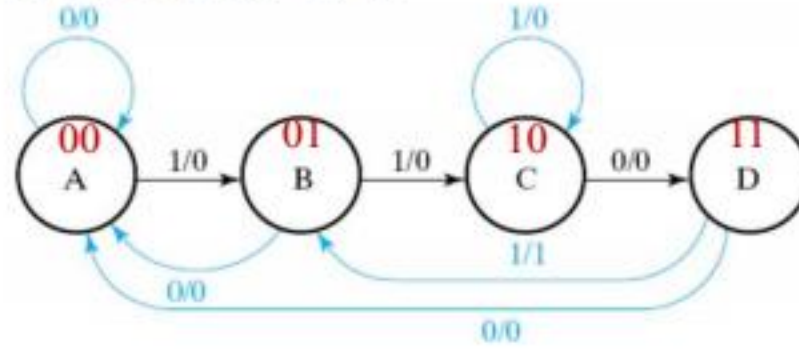
11

11

이진코드 : gray code 선택



이진코드 : BCD 순서 선택



12

12

State Table (p.210, p.214)

State(t)		input	State(t+1)		output
FFA	FFB		FFA	FFB	
0	0	0	0	0	0
0	0	1	0	1	0
0	1	0	0	0	0
0	1	1	1	1	0
1	0	0	0	0	0
1	0	1	0	1	1
1	1	0	1	0	0
1	1	1	1	1	0

Present State	Next State		Output Z	
	X = 0	X = 1	X = 0	X = 1
A	A	B	0	0
B	A	C	0	0
C	D	C	0	0
D	A	B	0	1

Present State	Next State		Output Z	
	X = 0	X = 1	X = 0	X = 1
00	00	01	0	0
01	00	11	0	0
11	10	11	0	0
10	00	01	0	1

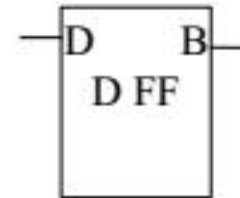
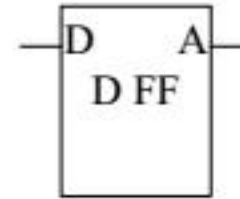
14

14

설계 입출력 기호 지정

- 입력 X
- 출력 Z
- D FF 2개 필요 : 각 저장값을 A,B라고 하자

X —————



———— Z

FFA 대신에 A라 하자
FFB 대신에 B라고 하자.

15

15

D FF을 이용한 설계(p.215)

- $D_A = A_{(t+1)} = \sum m(3,6,7)$
- $D_B = B_{(t+1)} = \sum m(1,3,5,7)$
- $Z = \sum m(5)$

- Map 간략화

■ $D_A = AB + BX$

■ $D_B = X$

■ $Z = AB'X$

D_A :

			B	B
	0	0	1	0
A	0	0	1	1
			X	X

D_B :

			B	B
	0	1	1	0
A	0	1	1	0
			X	X

State(t)		Input	State(t+1)		output
A	B	X	A	B	Z
0	0	0	0	0	0
0	0	1	0	1	0
0	1	0	0	0	0
0	1	1	1	1	0
1	0	0	0	0	0
1	0	1	0	1	1
1	1	0	1	0	0
1	1	1	1	1	0

16

16

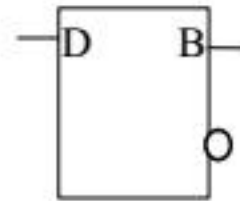
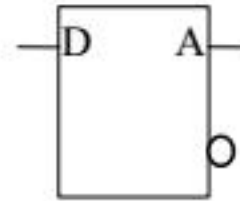
논리회로도를 그리시오.

■ $D_A = AB + BX$

■ $D_B = X$

■ $Z = AB'X$

X ———



—— Z

17

17

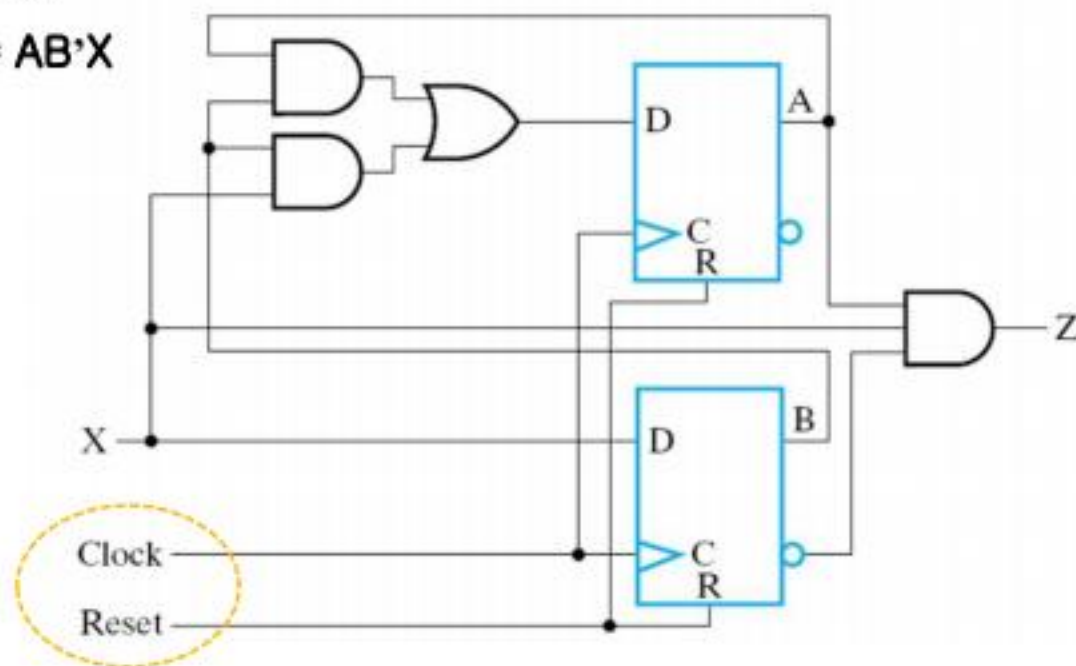
완성된 논리도

p.215

■ $D_A = AB + BX$

■ $D_B = X$

■ $Z = AB'X$



18

18

혼자 해보자

■ 예) sequence Recognizer

- 기능: 입력 sequence에서 "1011"이 발생할 경우, 1을 출력한다.

입력 시퀀스의 중복 인식을 허용한다.

....001010110001101101100100....

- 상태도

★조건1 : 상태이진비트를 **이진code**로 할당하자.

00 → 01 → 11 → 10

- 상태표

- 플립플롭 입력식

★조건2 : D FF를 사용하여 설계하자.

- 논리회로 다이어그램

19

19

State Diagram

■ 예) "1011" sequence Recognizer

- 기능: 입력 sequence에서 "1011"이 발생할 경우, 1출력(중복인식)

....001010110001101101100100....

★입력 1개, 출력 1개

- 상태도 표현



20

20

Example) Sequential Circuits Design

■문제예) 입력 x의 값이 1로 입력될 때마다

11→00→01→10→11→00→01...을 반복하는 회로를 작성하시오.

출력 : 상태가 10→11로 변환할 때만 1을 출력한다.

그외는 0을 출력한다.

단) 초기값은 11으로 둔다.

1. 상태도



2. 상태표

3. 플립플롭입력식

4. 논리회로도

21

21

A	B	X	A	B	Y
0	0	0	0	0	0
0	0	1	0	1	0
0	1	0	0	1	0
0	1	1	1	0	0
1	0	0	1	0	0
1	0	1	1	1	1
1	1	0	1	1	0
1	1	1	0	0	0

DA =

DB =

Y

DA:

		B	B
	0	0	1
A	1	1	0
		X	X

DB:

		B	B
	0	1	0
A	0	1	0
		X	X

22

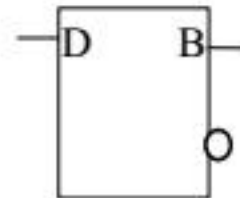
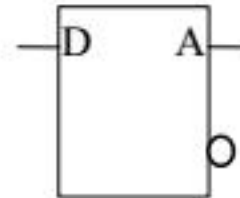
22

- $D_A = AB' + AX' + A'BX$

- $D_B = B'X + BX'$

- $Y = AB'X$

X ———



———— Y

23

23

sequence Recognizer Design 2 p.212

■ 경제적인 순차 회로 설계

1. 적은 수의 FF 을 이용한다.

2. 조합회로의 비용을 고려한다.

• 적은 수의 FF을 사용하였으나 조합회로가 더 복잡할 수 있다.

■ 상태에 대한 적절한 2진 코드 지정 → 순차회로의 설계 비용
(자주 적용되는 2진 코드 : BCD, Gray code, One-Hot code)

■ One-hot code (성능이 좋고 설계/증명이 쉽다는 특징)

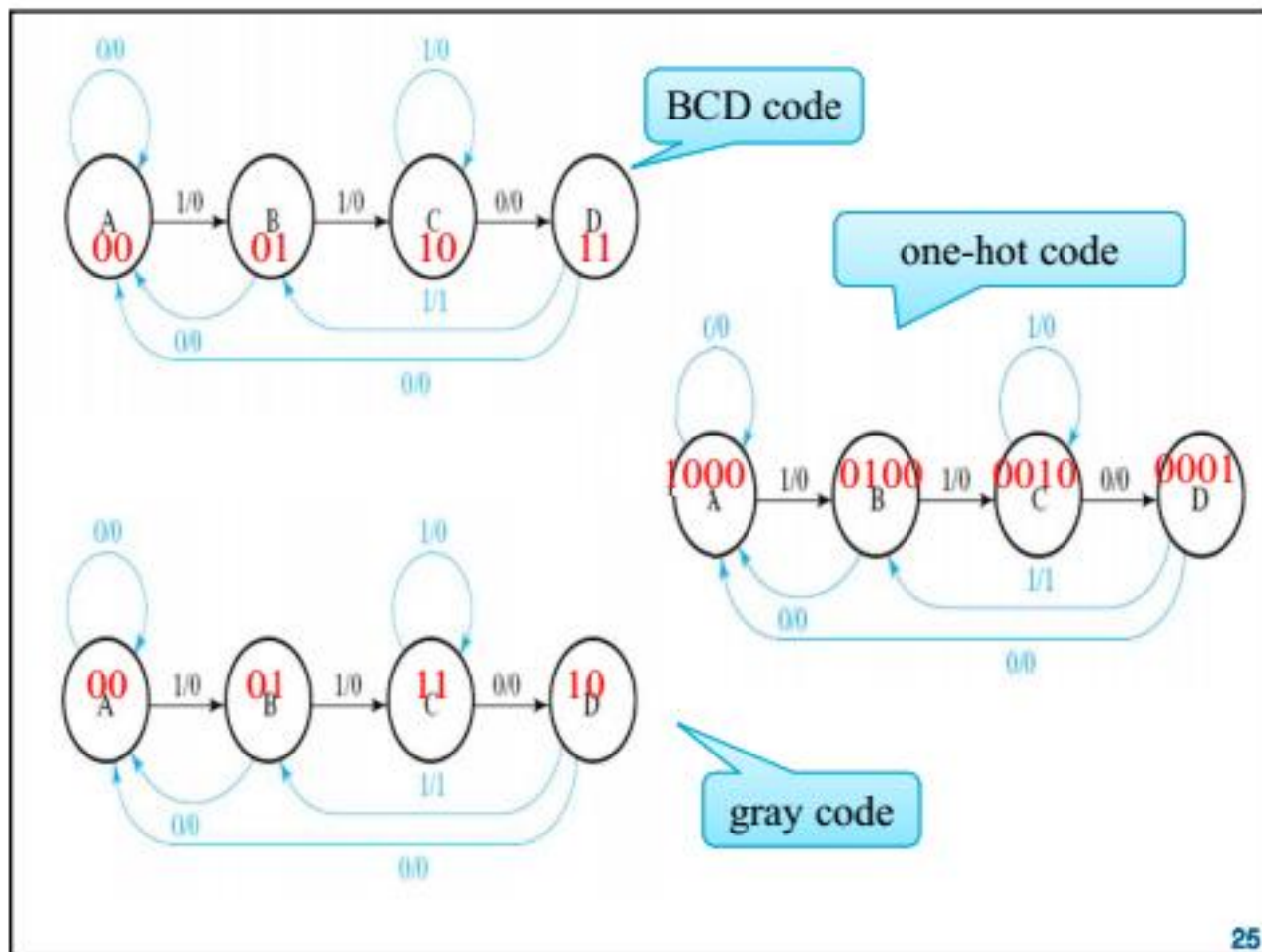
— 상태 각각에 1비트를 할당하고 한 비트만 1을 설정한다.

1을 다음 상태로 보내는 보내는 것과 의미가 같다.

정확성 > 비용

24

24



25

Sequential Circuits Design 2' p.214

Gray code를 적용

Present State	Next State		Output Z	
	X = 0	X = 1	X = 0	X = 1
AB				
00	00	01	0	0
01	00	11	0	0
11	10	11	0	0
10	00	01	0	1

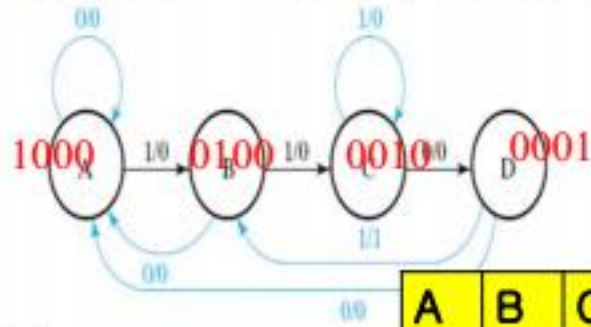
One-Hot code를 적용

Present State	Next State		Output Z	
	X = 0	X = 1	X = 0	X = 1
ABCD				
1000	1000	0100	0	0
0100	1000	0010	0	0
0010	0001	0010	0	0
0001	1000	0100	0	1

26

26

상태표 : 순차 인식기 ← One-Hot code 적용 p.214



DA =

DB =

DC =

DD =

Z =

A	B	C	D	X	A	B	C	D	Z
1	0	0	0	0	1	0	0	0	0
1	0	0	0	1	0	1	0	0	0
0	1	0	0	0	1	0	0	0	0
0	1	0	0	1	0	0	1	0	0
0	0	1	0	0	0	0	0	1	0
0	0	1	0	1	0	0	1	0	0
0	0	0	1	0	1	0	0	0	0
0	0	0	1	1	0	1	0	0	1

27

27

상태표: 순차 인식기 ← One-Hot code 적용

p.214

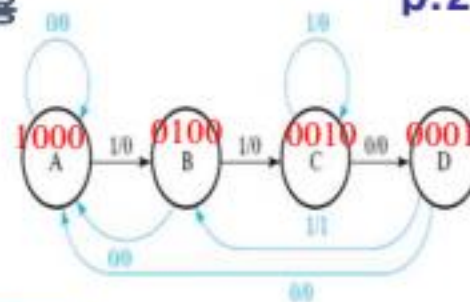
$$DA = AX' + BX' + DX'$$

$$DB = AX + DX$$

$$DC = BX + CX$$

$$DD = CX'$$

$$Z = DX$$



논리
다이아그램
그리기

A	B	C	D	X	A	B	C	D	Z
1	0	0	0	0	1	0	0	0	0
1	0	0	0	1	0	1	0	0	0
0	1	0	0	0	1	0	0	0	0
0	1	0	0	1	0	0	1	0	0
0	0	1	0	0	0	0	0	1	0
0	0	1	0	1	0	0	1	0	0
0	0	0	1	0	1	0	0	0	0
0	0	0	1	1	0	1	0	0	1

28

28

- $DA = AX' + BX' + DX'$

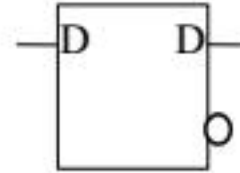
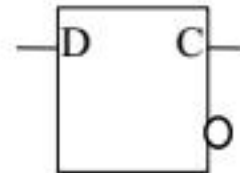
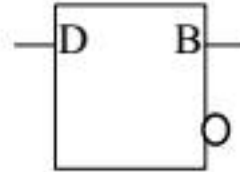
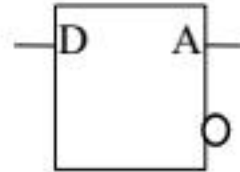
- $DB = AX + DX$

- $DC = BX + CX$

- $DD = CX'$

- $Z = DX$

X ———



——— Z

29

29

상태도 : 순차 인식기 ◀ One-Hot code 적용

설계가 수월하다 : 상태도만으로 설계가능하다. p.216

$$DA = AX' + BX' + DX' = (A+B+D)X'$$

$$DB = AX + DX = (A+D)X$$

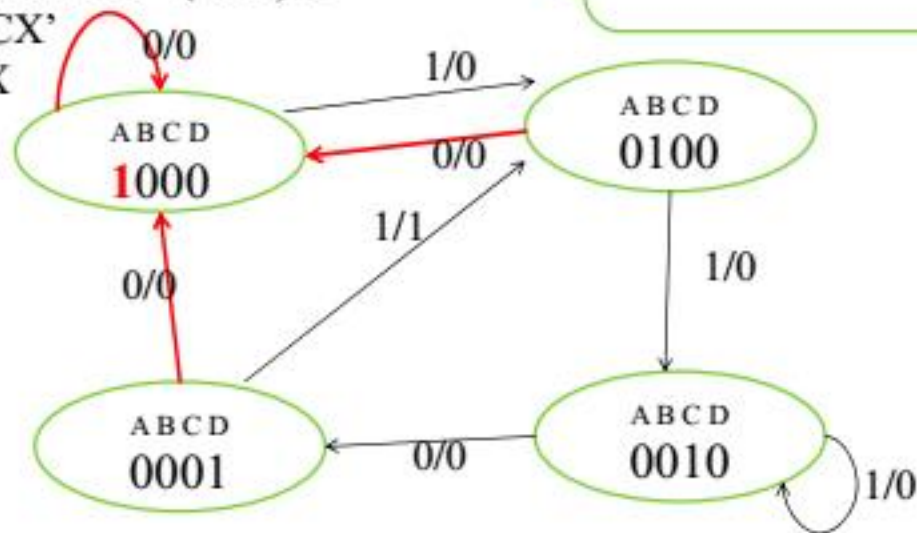
$$DC = BX + CX = (B+C)X$$

$$DD = CX'$$

$$Z = DX$$

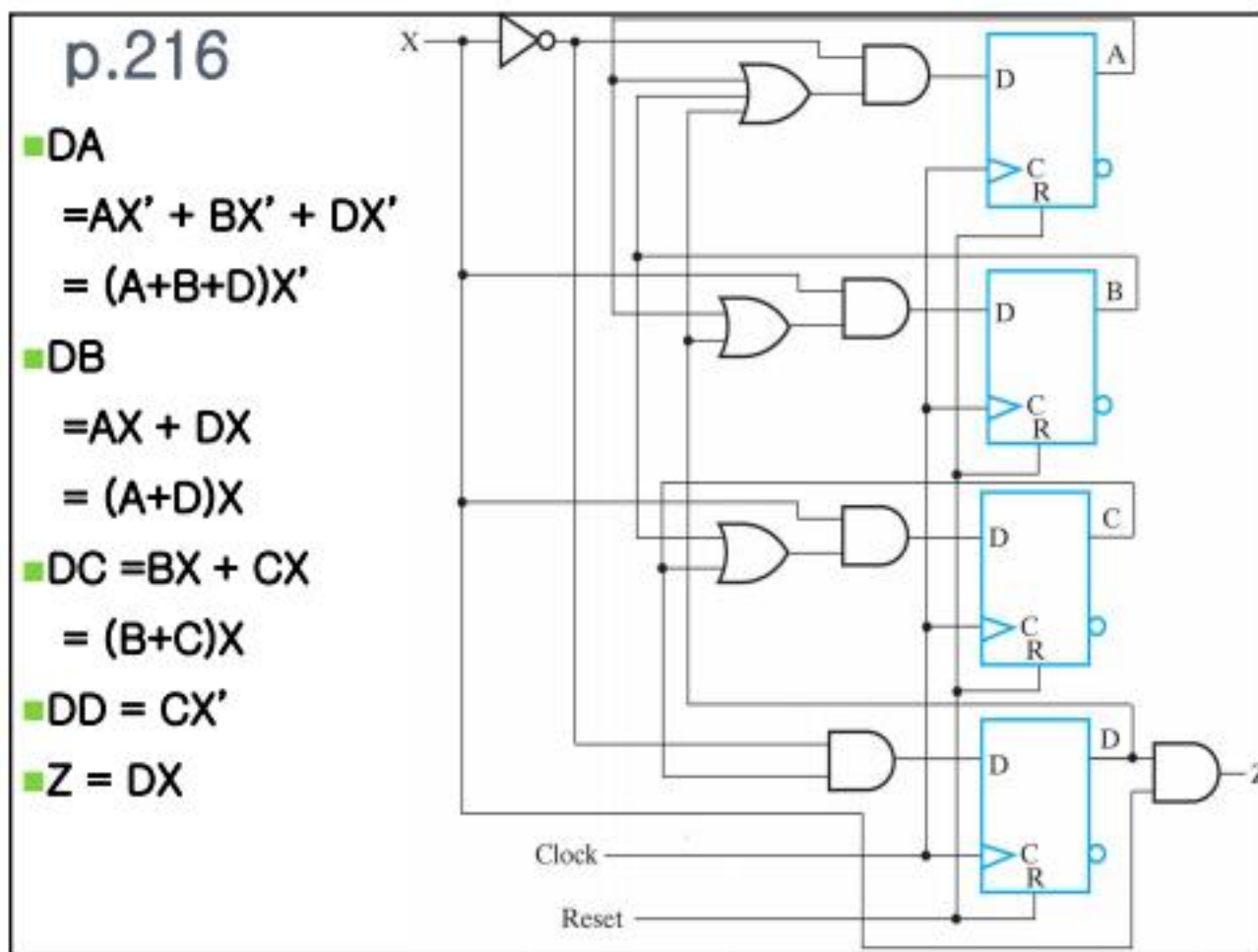
A가 1이 되는 경우:

A, B, D가 1이고
x가 0입력될 때



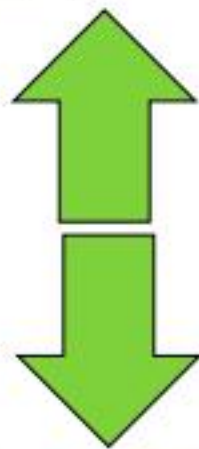
30

30



31

기본 순차회로 설계



상태 개수가 2의 n제곱이 아닐 경우
순차회로 설계

32

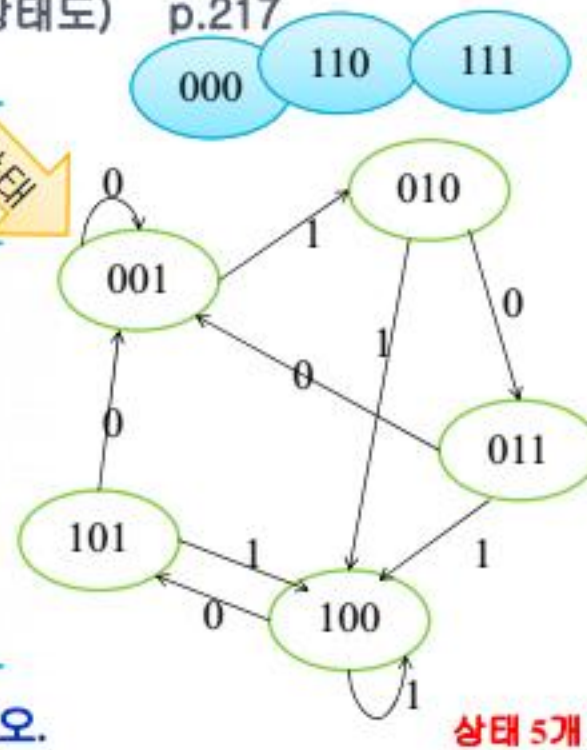
32

사용되지 않은 상태에 대한 고려 (상태도) p.217

Present State			Input	Next State		
A	B	C	X	A	B	C
0	0	1	0	0	0	1
0	0	1	1	0	1	0
0	1	0	0	0	1	1
0	1	0	1	1	0	0
0	1	1	0	0	0	1
0	1	1	1	1	0	0
1	0	0	0	1	0	1
1	0	0	1	1	0	0
1	0	1	0	0	0	1
1	0	1	1	1	0	0

Education Inc.

설계하라) D FF 입력식을 구하시오.



33

사용되지 않은 상태를 갖는 설계 p.217

■ N개의 FF이 주어졌을 때 사용되지 않는 상태에 대해 고려하자.

■ 3개의 FF → 최대 가능 상태수?

■ $D_A = A_{(t+1)} = \sum m(5, 7, 8, 9, 11)$
don't care = $\sum m(0, 1, 12 \sim 15)$

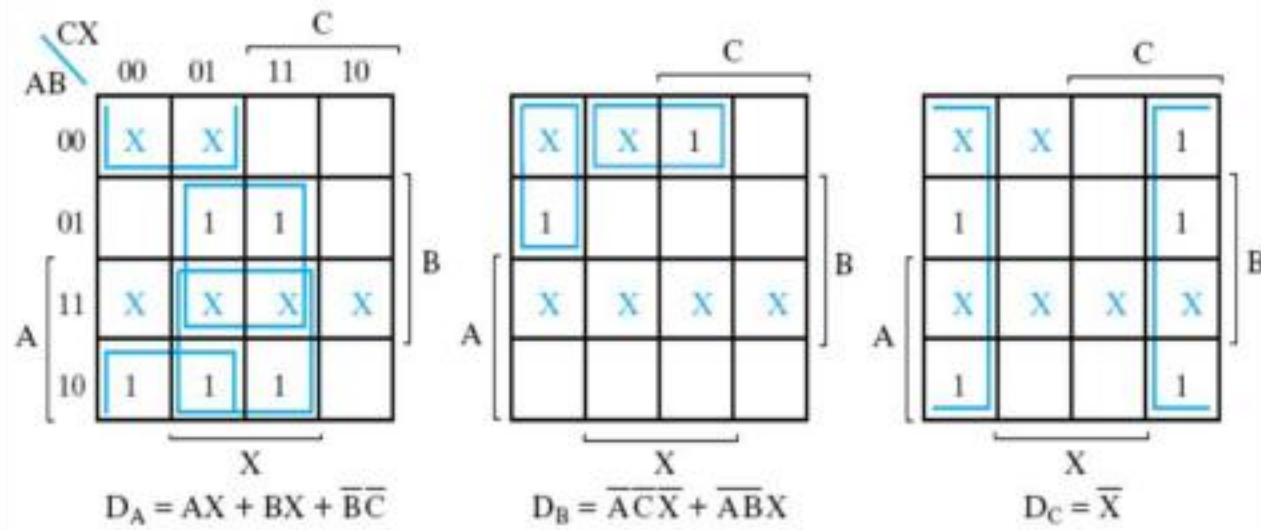
■ $D_B = B_{(t+1)} = \sum m(3, 4)$
don't care = $\sum m(0, 1, 12 \sim 15)$

■ $D_C = C_{(t+1)} = \sum m(2, 4, 6, 8, 10)$
don't care = $\sum m(0, 1, 12 \sim 15)$

Present State			Input	Next State		
A	B	C	X	A	B	C
0	0	0	?			
0	0	1	0	0	0	1
0	0	1	1	0	1	0
0	1	0	0	0	1	1
0	1	0	1	1	0	0
0	1	1	0	0	0	1
0	1	1	1	1	0	0
1	0	0	0	1	0	1
1	0	0	1	1	0	0
1	0	1	0	0	0	1
1	0	1	1	1	0	0
1	1	0	?			
1	1	1	?			

34

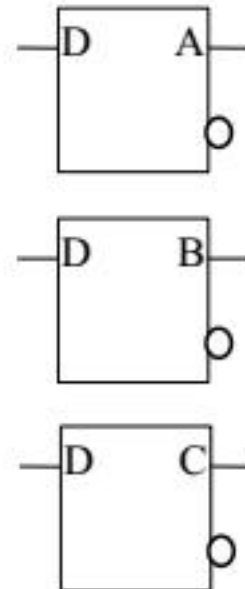
맵간략화 => 논리도 p.217



논리 회로도를 그리자.

- $D_A = AX + BX + B'C'$
- $D_B = A'C'X' + A'B'X$
- $D_C = X'$

X ———



순차회로 설계 정리.

- 기능 명세
- 상태도 구성
- 상태표 구성
- 플립 플롭의 개수 결정
- 맵 간략화
 - 맵 간략화를 위한 상태표 재구성
 - one hot code, gray code 선택
- 논리도 구성

- 사용하지 않는 상태가 있는 경우, 설계

37