

Chapter 3 :: Sequential Logic Design (2)

Digital Design and Computer Architecture, 2nd Edition

David Money Harris and Sarah L. Harris

참고도서 : 개정판 논리회로 설계(김종현저) , 홍릉과학출판사.

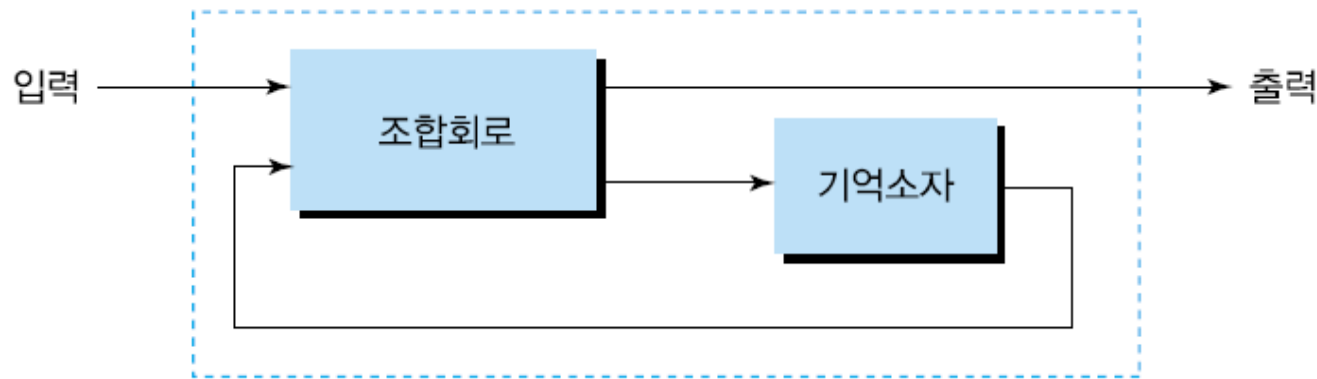
- 제7장 순차회로의 분석과 설계

Chapter 3 :: Topics

- Introduction
- Latches and Flip-Flops
- Synchronous Logic Design
- Finite State Machines
- Shifter
- Timing of Sequential Logic
- Parallelism

3.3 Sequential Logic

- 현재의 입력 뿐 아니라 과거의 입력 혹은 출력 값들도 함께 고려하여 현재의 출력 값을 결정하는 논리회로
- 조합회로와 기억 소자(memory element)들로 구성



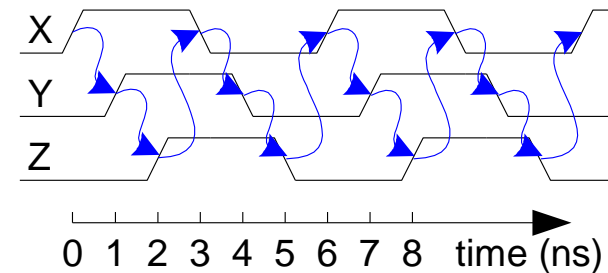
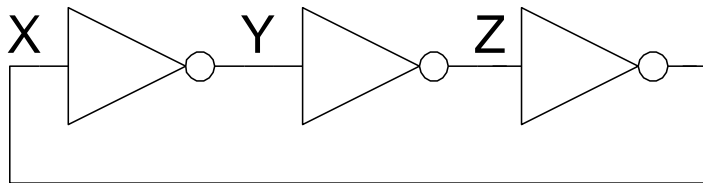
■ 그림 7-1 순차회로의 블록도

3.3 Sequential Logic

- 순차회로에 사용되는 기억 소자의 분류
 - 두 개의 안정된(bi-stable) 기억소자로, 메모리의 기본적인 형태 상태 중 하나를 가지는 1비트
 - 래치(latch) : 기본적인 게이트 회로로 구성되는 기억 소자
 - 플립-플롭(flip-flop) : 래치에 별도의 회로를 추가하여 구성되는 기억 소자
 - 저장된 값(즉, 상태 값)을 변경시키는 방법에서 차이

3.3 Sequential Logic

- Sequential circuits: all circuits that aren't combinational
- A **problematic circuit**:



- This circuit has no inputs and 1-3 outputs.
- It is an astable circuit that oscillates.
- Its period depends on the delay of the inverters – which depends on the manufacturing process, temperature, etc.
- The circuit has a **cyclic path**(순환경로): output fed back to input

3.3.2 Synchronous Sequential Logic Design

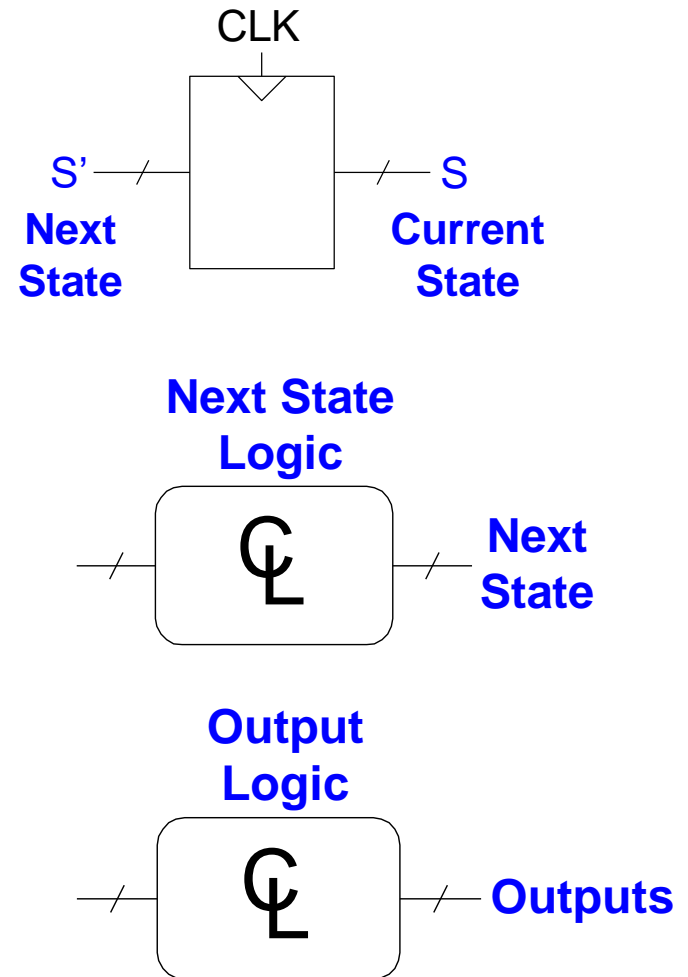
- Breaks cyclic paths by **inserting registers**
(순환 경로를 없애기 위해 레지스터를 삽입)
- These registers contain the **state** of the system
 - 상태(state)
 - 어떤 한 시점에서 순차회로의 기억 소자에 저장되어 있는 2진 정보
 - 한 비트 혹은 여러 비트들로 표현
 - 순차회로는 외부로부터 입력 신호(들)을 받으며, 그 입력들이 기억 소자에 저장된 현재 상태 값과 함께 조합회로를 통과하여 출력 값을 결정하며, 그 과정에서 기억 소자에는 새로운 상태 값이 저장됨.
- The state changes at the clock edge, so we say the system is **synchronized** to the clock. (상태는 클럭의 에지에서만 변화, 시스템은 클럭에 동기화 된다)

Synchronous Sequential Logic Design

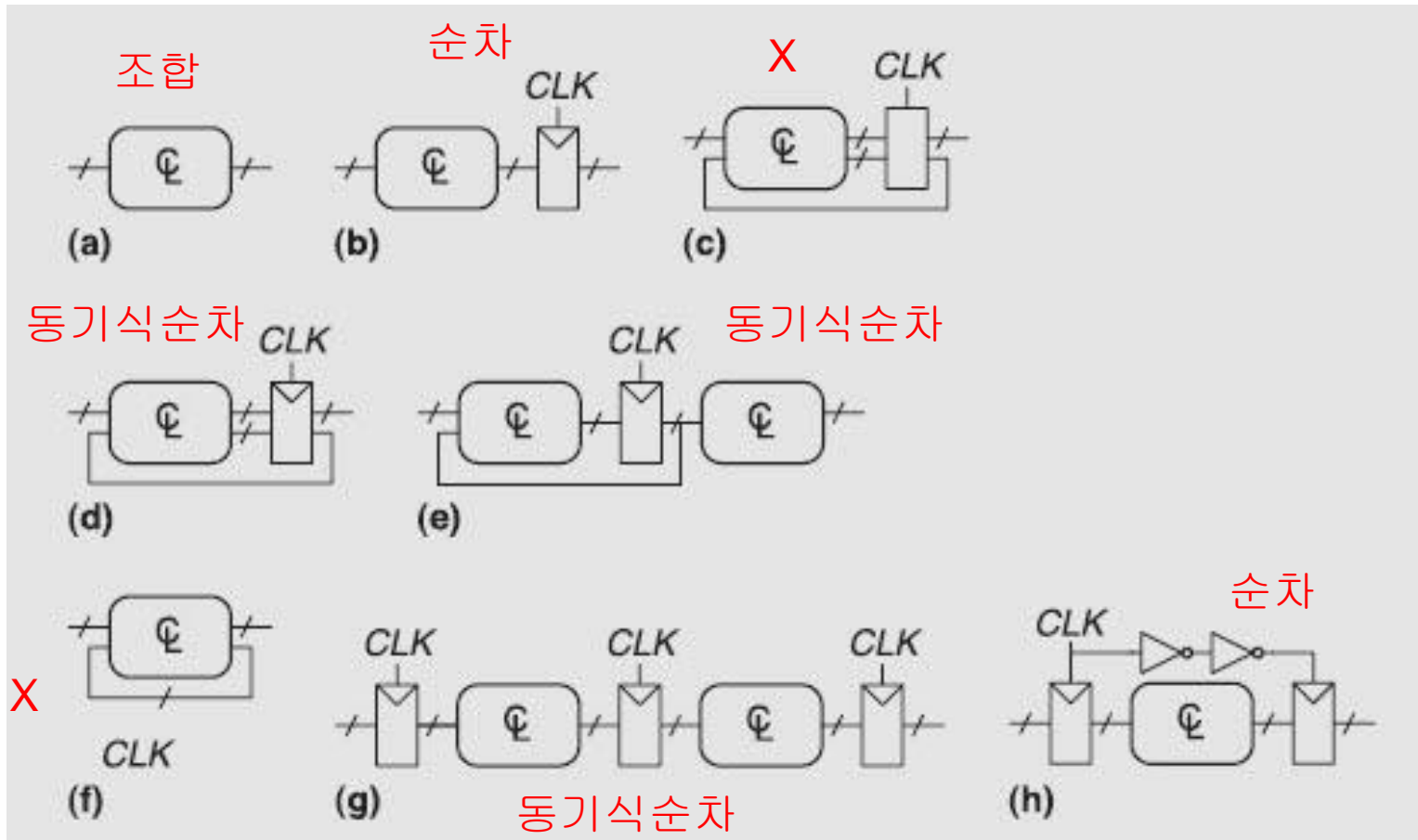
- Rules of synchronous sequential circuit composition:
 - Every circuit element is either a register or a combinational circuit
(모든 회로 소자가 레지스터이거나 조합회로이다.)
 - At least one circuit element is a register.
(적어도 하나의 회로 소자는 레지스터이다.)
 - All registers receive the same clock signal.
(모든 레지스터는 똑 같은 클럭 신호를 받는다.)
 - Every cyclic path contains at least one register.
(모든 순환경로는 적어도 하나의 레지스터를 포함한다.)
- Two common synchronous sequential circuits
 - Finite state machines (FSMs)
 - Pipelines

3.4 Finite State Machine (FSM)

- Consists of:
 - **State register** that
 - Store the current state
 - Load the next state at the clock edge
 - **Combinational logic** that
 - Computes the next state
 - Computes the outputs



동기식 순차회로는?



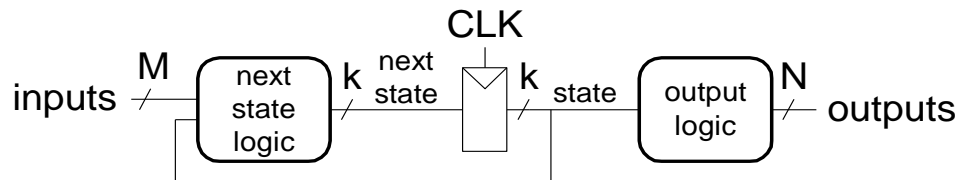
Sequential Logic

- 동기식 순차회로(Synchronous Sequential Logic)
 - 속도가 느리다 .(동작이 미리 종료되더라도 클록의 트리거 에지 때까지 기다려야 하므로)
 - 설계가 용이하다.
 - 출력 신호가 안정된다.
- 비동기식 순차회로(Asynchronous Sequential Logic)
 - 속도가 빠르다.
 - 설계가 어렵다.
 - 출력 값이 입력 신호에 따라 계속 변하여 불안정하다.

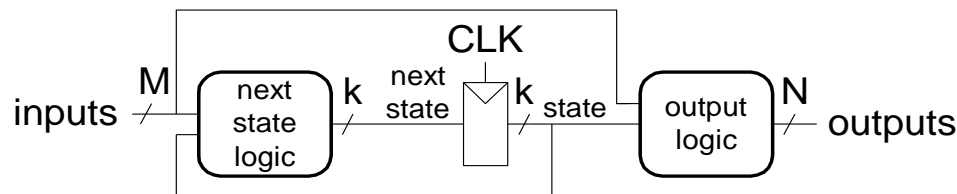
3.4 Finite State Machines (FSMs)

- Next state(다음상태) is determined by
 - the current state(현재상태) and the inputs(입력)
- Two types of finite state machines differ in the output logic:
 - **Moore FSM**: outputs depend only on the current state
 - **Mealy FSM**: outputs depend on the current state *and* the inputs

Moore FSM



Mealy FSM



3.4 Finite State Machines (FSMs)

- **Moore FSM:** outputs depend only on the current state
 - 회로의 출력 신호가 **현재 상태들** 만**에 의해 결정**되는 형태의 순차회로
 - 공통 클록의 트리거 에지에서 상태들이 바뀌면, 출력 신호도 그와 동시에 갱신
 - 이 모델의 회로를 **무어 기계(Moore machine)**라 부름
- **Mealy FSM:** outputs depend on the current state *and* the inputs
 - 회로의 출력 신호가 **현재 상태들과 입력 신호에 의해 결정**되는 형태의 순차회로. 즉, 외부로부터 들어오는 입력 신호도 출력 신호에 영향을 줌
 - 입력 신호 값이 바뀌면 출력 신호도 (클록 신호에 상관 없이) 바뀜
 - 입력 신호를 클록 신호와 동기 시키지 않는 경우에는 출력이 수시로 바뀌는 불안정한 회로가 됨
 - 이 모델의 회로를 **밀리 기계(Mealy machine)**라 부름

FSM 설계 절차

- ① 설계할 회로의 기능 및 동작 특성에 대하여 상세하게 기술하며, 필요 시 상태 다이어그램을 그린다.
- ② 기억 소자로 사용할 **Flip-Flop**의 종류를 선택한다.
- ③ 상태표를 작성한다.
- ④ 입력 함수 및 출력 함수를 구하고, 간략화 한다.
- ⑤ 회로를 구성한다.

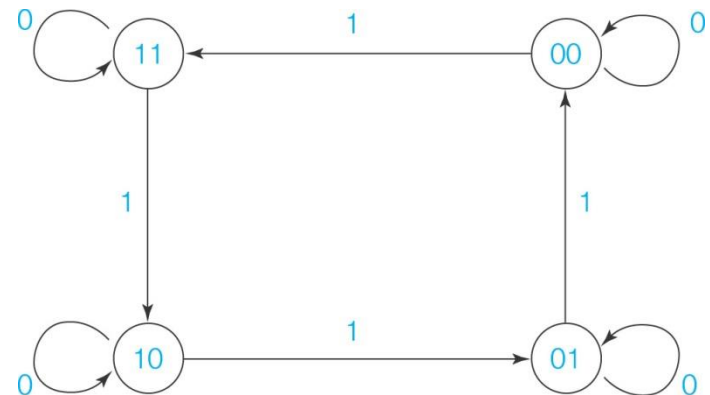
D Flip-Flop을 이용한 순차회로의 설계 1

- 문제 기술(problem description)

- 2비트 다운-카운터(down-counter)를 설계하고자 한다.
- 이 순차 회로의 초기값은 $3(11_{(2)})$ 에서 시작하며,
- 입력 펄스가 들어올 때마다 1씩 감소하여 0까지 된 다음, 다시 3으로 반복되는 2비트 상태 조합들을 발생한다.

- 요구사항 분석

- 상태 수 = 2 (A, B)
- $x = 1 \rightarrow AB = 11 \rightarrow 10 \rightarrow 01 \rightarrow 00$, & $\rightarrow 11$ 으로 되돌아가서 반복
- $x = 0 \rightarrow$ 상태 불변
- 상태 다이어그램:



D Flip-Flop을 이용한 순차회로의 설계 2

[표 7-8] 다운 카운터에 대한 상태표

현재 상태		입력	다음 상태	
$A(t)$	$B(t)$	$x(t)$	$A(t+1)$	$B(t+1)$
0	0	0	0	0
0	0	1	1	1
0	1	0	0	1
0	1	1	0	0
1	0	0	1	0
1	0	1	0	1
1	1	0	1	1
1	1	1	1	0

D Flip-Flop을 이용한 순차회로의 설계 3

• 플립-플롭의 종류

- 설계과정이 가장 간단한 D 플립-플롭을 선택

• 입력 함수의 유도

- $D_A = A(t+1) = A'B'x + AB'x' + ABx' + ABx$
 $= A'B'x + Ax' + AB$
- $D_B = B(t+1) = A'B'x + A'Bx' + AB'x + ABx'$
 $= B'x + Bx' = B \oplus x$

[표 7-8] 다운 카운터에 대한 상태표

현재 상태		입력	다음 상태	
$A(t)$	$B(t)$	$x(t)$	$A(t+1)$	$B(t+1)$
0	0	0	0	0
0	0	1	1	1
0	1	0	0	1
0	1	1	0	0
1	0	0	1	0
1	0	1	0	1
1	1	0	1	1
1	1	1	1	0

$\underline{D_A}$

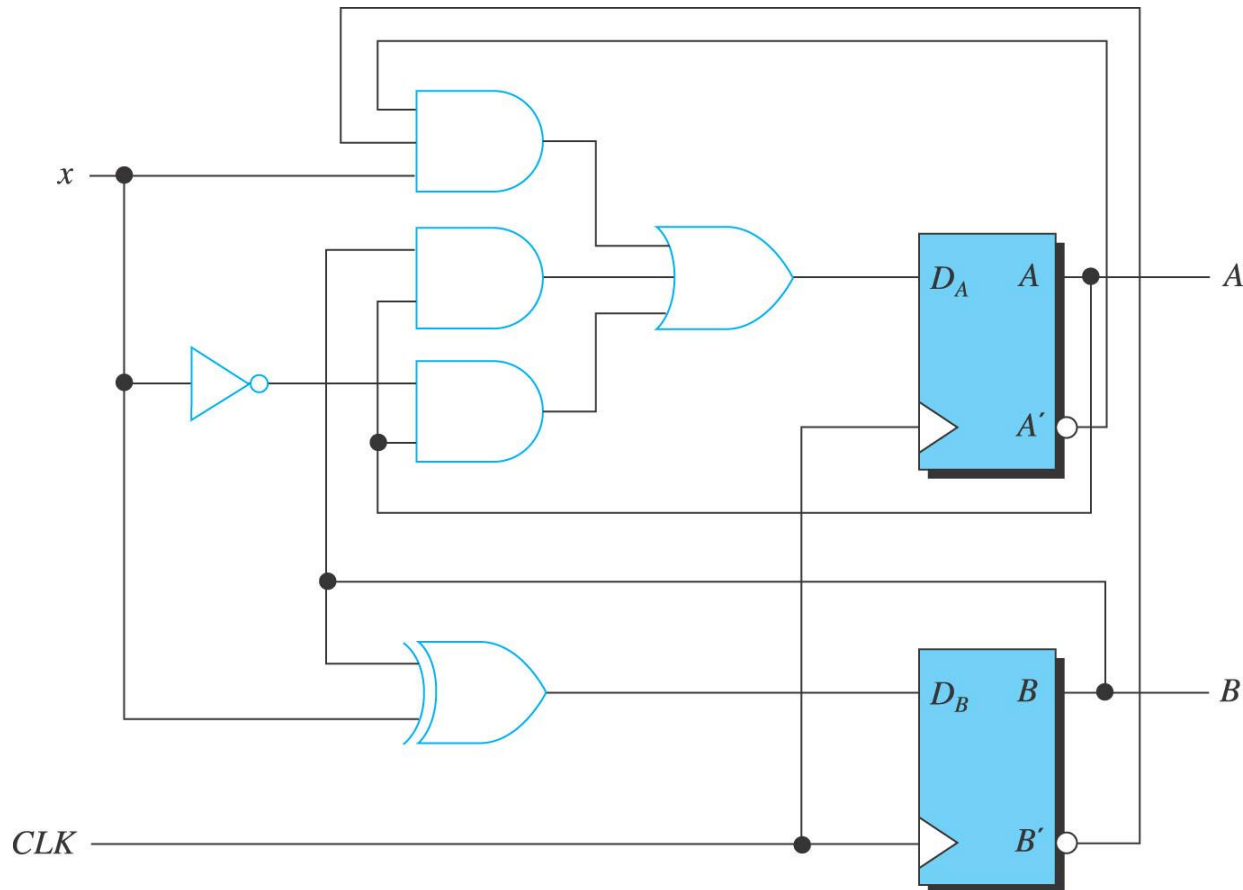
A	Bx			
	00	01	11	10
0	0	1	0	0
1	1	0	1	1

$\underline{D_B}$

A	Bx			
	00	01	11	10
0	0	1	0	1
1	0	1	0	1

D Flip-Flop을 이용한 순차회로의 설계 4

- 회로 구성



JK Flip-Flop을 이용한 순차회로의 설계 1

- 문제 기술(problem description)
 - D 플립-플롭을 이용한 2비트 **다운-카운터(down-counter)** 설계와 동일한 예를 사용하고, 회로 복잡도를 비교 한다.
- JK 플립-플롭의 여기표(excitation table)

$Q(t)$	$Q(t+1)$	J	K
0	0	0	X
0	1	1	X
1	0	X	1
1	1	X	0

JK Flip-Flop을 이용한 순차회로의 설계 2

- 상태표 (excitation table 이용)

$Q(t)$	$Q(t+1)$	J	K
0	0	0	X
0	1	1	X
1	0	X	1
1	1	X	0

[표 7-10] JK 플립-플롭을 이용한 다운-카운터의 상태표

현재 상태		입력	다음 상태		플립-플롭 입력들			
A	B	x	A	B	J_A	K_A	J_B	K_B
0	0	0	0	0	0	X	0	X
0	0	1	1	1	1	X	1	X
0	1	0	0	1	0	X	X	0
0	1	1	0	0	0	X	X	1
1	0	0	1	0	X	0	0	X
1	0	1	0	1	X	1	1	X
1	1	0	1	1	X	0	X	0
1	1	1	1	0	X	0	X	1

JK Flip-Flop을 이용한 순차회로의 설계 3

- 입력 함수의 유도

		Bx			
		00	01	11	10
A	0	0	1	0	0
	1	X	X	X	X

(a) $J_{A(next)} = B'x$

		Bx			
		00	01	11	10
A	0	X	X	X	X
	1	0	1	0	0

(b) $K_{A(next)} = B'x$

		Bx			
		00	01	11	10
A	0	0	1	X	X
	1	0	1	X	X

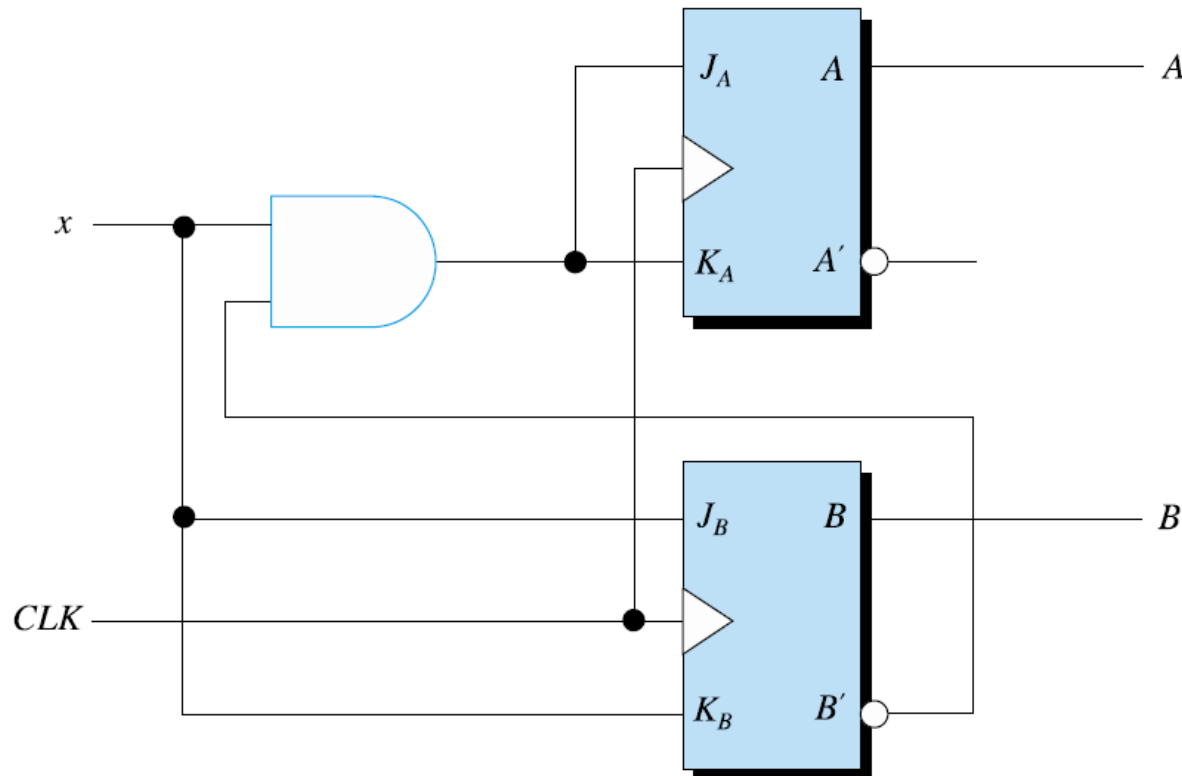
(c) $J_{B(next)} = x$

		Bx			
		00	01	11	10
A	0	X	X	1	0
	1	X	X	1	0

(d) $K_{B(next)} = x$

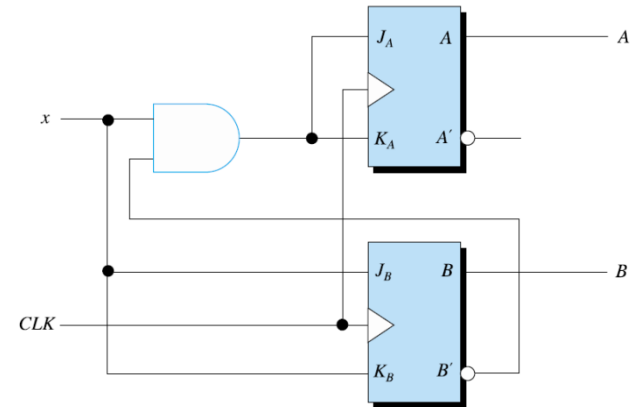
JK Flip-Flop을 이용한 순차회로의 설계 4

- 회로 구성
 - JK 플립-플롭을 이용하면, 회로가 단순해 짐

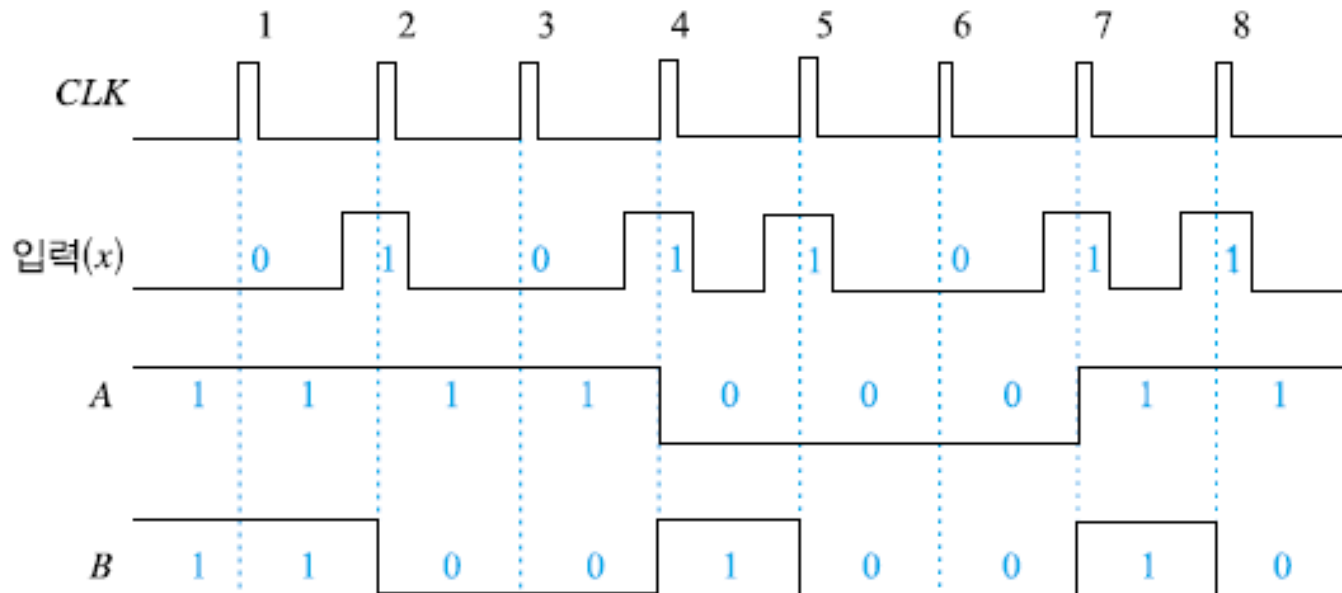


JK Flip-Flop을 이용한 순차회로의 설계 5

- 아래의 입력 신호 시퀀스에 대한
다운-카운터의 출력 파형
(A, B 의 초기값 = 1)



입력 시퀀스 : 0 1 0 1 1 0 1 1



FSM 분석

- 순차회로 분석의 목적

- : 외부 신호 x 와 현재 상태 $A(t)$ 및 $B(t)$ 에 의해

- : 다음 상태 $A(t+1)$ 과 $B(t+1)$ 이 클록의 트리거 에지에서 어떤 값으로 결정되는지를 파악하는 것

- 순차회로 분석 절차

- ① 모델을 분석한다.

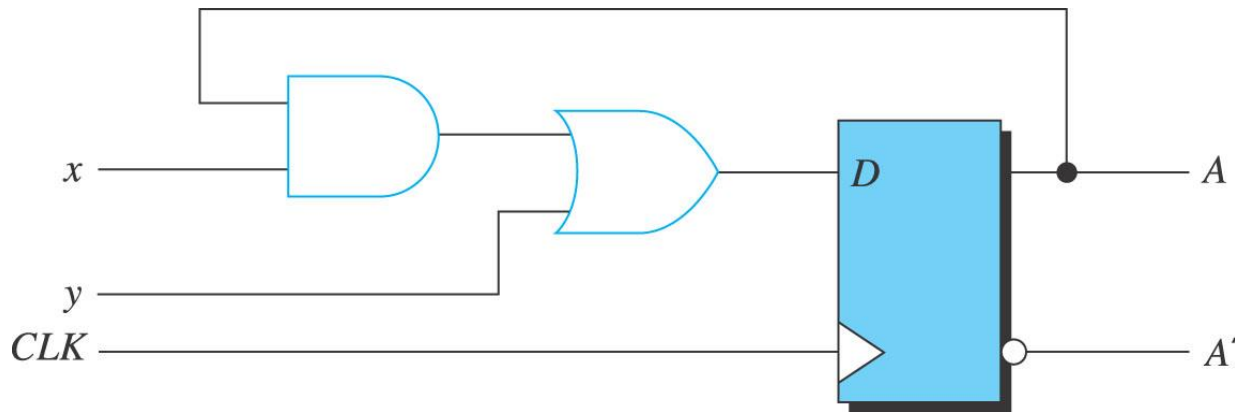
- ② 플립-플롭의 입력 함수들을 구한다.

- ③ 입력 함수들을 이용하여, 현재 상태들과 입력 신호 값에 따른 다음 상태 값들을 구하여 상태표를 작성한다.

- ④ 상태 다이어그램을 그려서 상태 전이를 파악하고, 회로의 동작 특성을 이해한다.

D Flip-Flop이 한 개 포함된 FSM 분석 1-1

[1] 모델분석



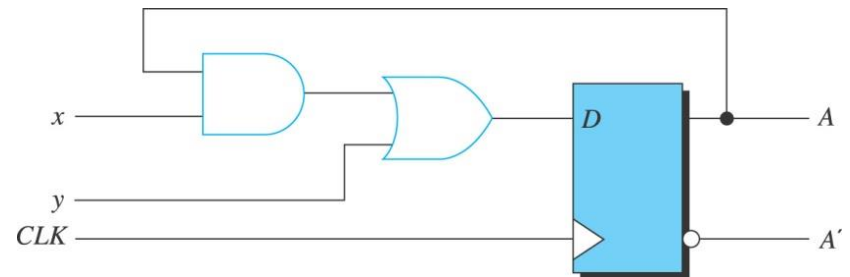
- <무어 모델> 출력 값(A)이 입력 신호에 직접 영향을 받지 않음

한 개의 **D Flip-Flop**이 포함된 **FSM** 분석 1-2

[2] 상태 함수(state equation)의 유도

- 다음 상태(next state)를 현재 상태(present state)와 입력에 대한 함수로 표현

- 입력 함수: $D_{(next)} = Ax + y$



- 상태 함수: $A(t+1)_{(next)} = D(t) = A(t)x(t) + y(t)$ 혹은
 $A(next) = Ax + y$

한 개의 **D Flip-Flop**이 포함된 **FSM** 분석 1-3

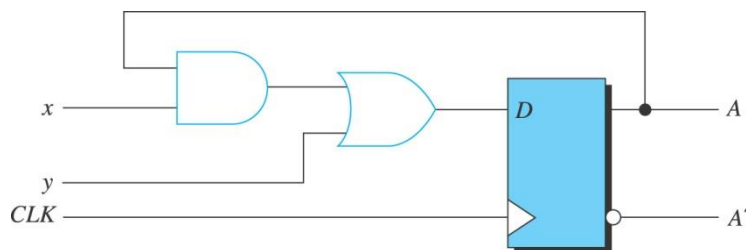
[3] 상태표(state table) 작성

- 상태 함수

$[A_{(next)} = Ax + y]$ 에 근거하여

- 입력 값들에 따른 상태 변화를 표시한 표

- 전이표(transition table)라고도 부름



[표 7-2] 그림 7-29의 회로에 대한 상태표

현재 상태 A	입력 $x \quad y$		다음 상태 $A(t+1)$
0	0	0	0
0	0	1	1
0	1	0	0
0	1	1	1
1	0	0	0
1	0	1	1
1	1	0	1
1	1	1	1

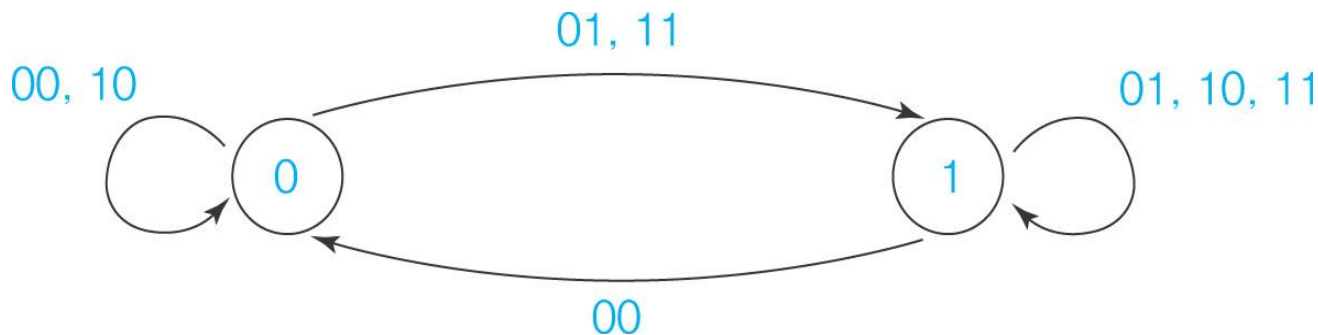
한 개의 **D Flip-Flop**이 포함된 **FSM** 분석 1-4

[4] 상태 다이어그램(state diagram)의 작성

- 상태 : 원으로 표시
- 상태 전이
 - 상태 간을 연결하는 선과 화살표 및
 - 해당 상태 전이를 야기하는 입력 조합(xy) 표시

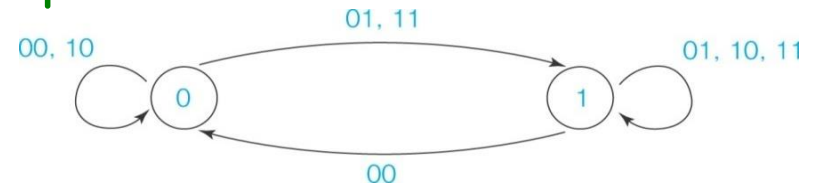
[표 7-2] 그림 7-29의 회로에 대한 상태표

현재 상태 A	입력 $x \quad y$		다음 상태 $A(t+1)$
0	0	0	0
0	0	1	1
0	1	0	0
0	1	1	1
1	0	0	0
1	0	1	1
1	1	0	1
1	1	1	1



한 개의 **D Flip-Flop**이 포함된 **FSM** 분석 1-5

[5] 상태 다이어그램으로부터의 분석



- 상태 전이 ($0 \rightarrow 1$)

- 상태가 0일 때 입력 조합 $xy = 01$ ($x=0, y=1$) 혹은 11 ($x=1, y=1$)이라면, 상태는 1로 전이.

- 상태 전이 ($1 \rightarrow 0$)

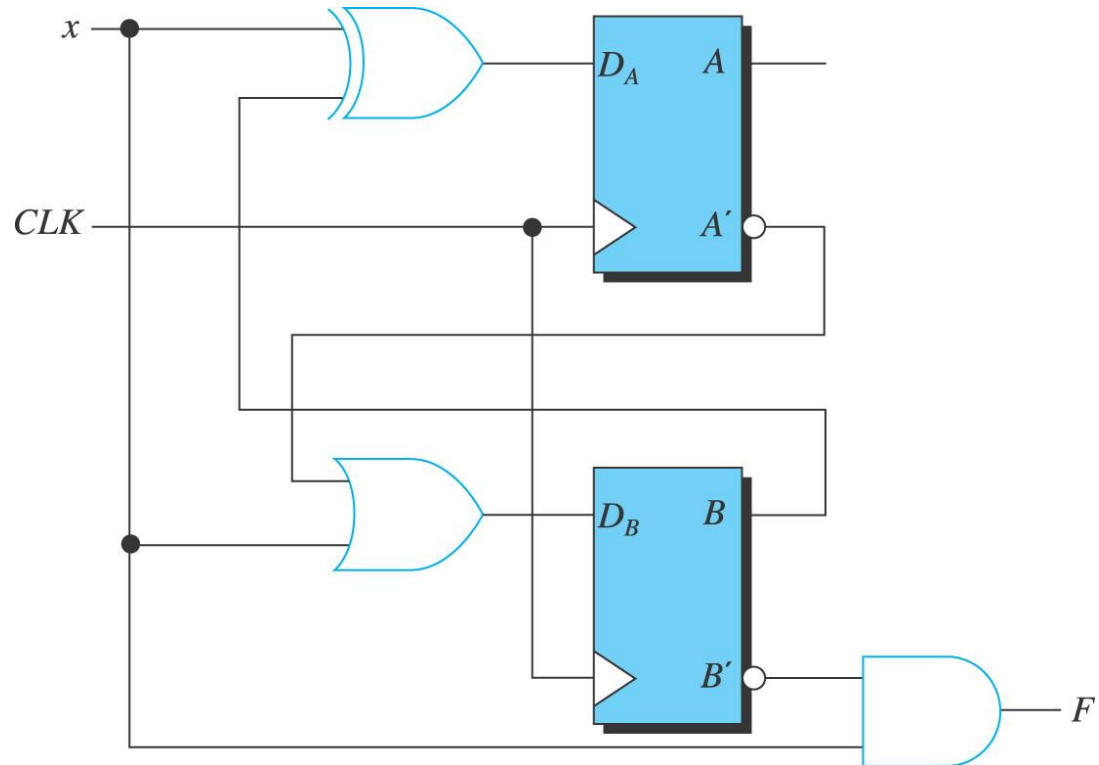
- 상태가 1일 때 $xy = 00$ 이라면, 상태는 0으로 전이.

- 상태 불변

- 상태가 0일 때 $xy = 00$ 혹은 10 이라면 상태는 변하지 않고 그대로 유지
- (그림에서 0으로 되돌아가는 것으로 표시).
- 또한, 상태가 1일 때 $xy = 01, 10$ 혹은 11 이라면, 상태는 변하지 않고 1을 유지 (그림에서 1로 되돌아가는 것으로 표시)

두 개의 **D Flip-Flop**이 포함된 **FSM** 분석 2-1

[1] 모델분석



<밀리 모델> 출력 신호가 상태 A 와 B 뿐 아니라,
입력 신호 x 에 의해서도 영향을 받음.

두 개의 **D Flip-Flop**이 포함된 **FSM** 분석 **2-2**

[2] 상태 함수(state equation)의 유도

– 입력 함수:

$$D_{A(next)} = B \oplus x$$

$$D_{B(next)} = A' + x$$

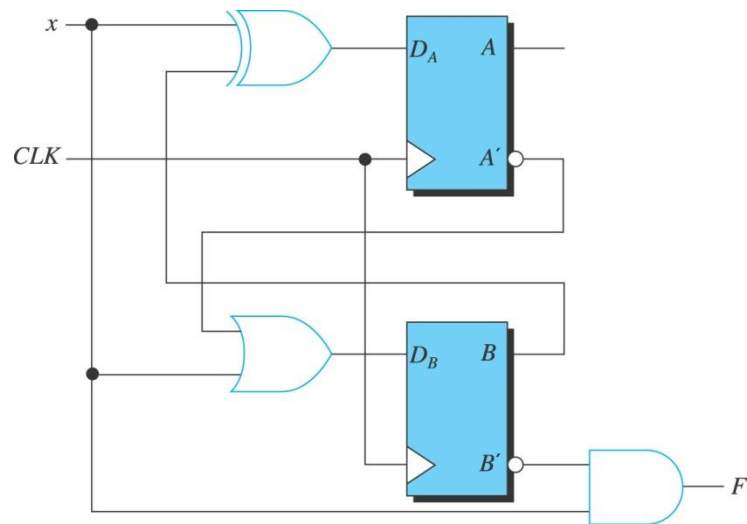
– 상태 함수들 :

$$A(t+1)_{(next)} = B \oplus x = B'x + Bx'$$

$$B(t+1)_{(next)} = A' + x$$

– 출력 함수:

$$F(t) = B'(t)x(t) \text{ 혹은 } F = B'x$$



두 개의 **D Flip-Flop**이 포함된 **FSM** 분석 **2-3**

[3] 상태표 [유형 1]

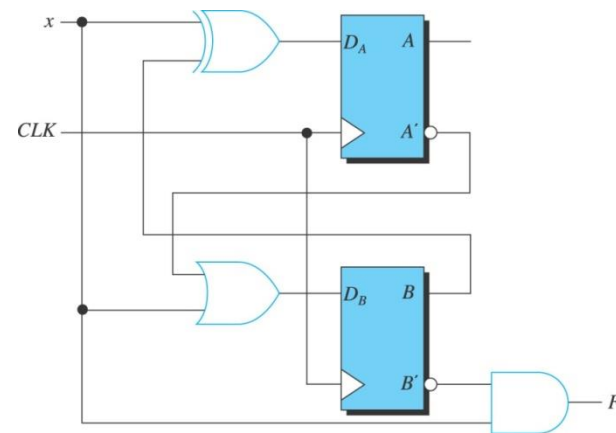
$$A(t+1)_{(next)} = B \oplus x = B'x + Bx'$$

$$B(t+1)_{(next)} = A' + x$$

$$F(t) = B'(t)x(t) \text{ 혹은 } F = B'x$$

[표 7-3] 그림 7-31의 회로에 대한 상태표

현재 상태		입력 x	다음 상태		현재 출력 F
A	B		A	B	
0	0	0	0	1	0
0	0	1	1	1	1
0	1	0	1	1	0
0	1	1	0	1	0
1	0	0	0	0	0
1	0	1	1	1	1
1	1	0	1	0	0
1	1	1	0	1	0



두 개의 **D Flip-Flop**이 포함된 **FSM** 분석 2-4

[3] 상태표 [유형 2]

[표 7-3] 그림 7-31의 회로에 대한 상태표

현재 상태		입력	다음 상태		현재 출력
<i>A</i>	<i>B</i>	<i>x</i>	<i>A</i>	<i>B</i>	<i>F</i>
0	0	0	0	1	0
0	0	1	1	1	1
0	1	0	1	1	0
0	1	1	0	1	0
1	0	0	0	0	0
1	0	1	1	1	1
1	1	0	1	0	0
1	1	1	0	1	0

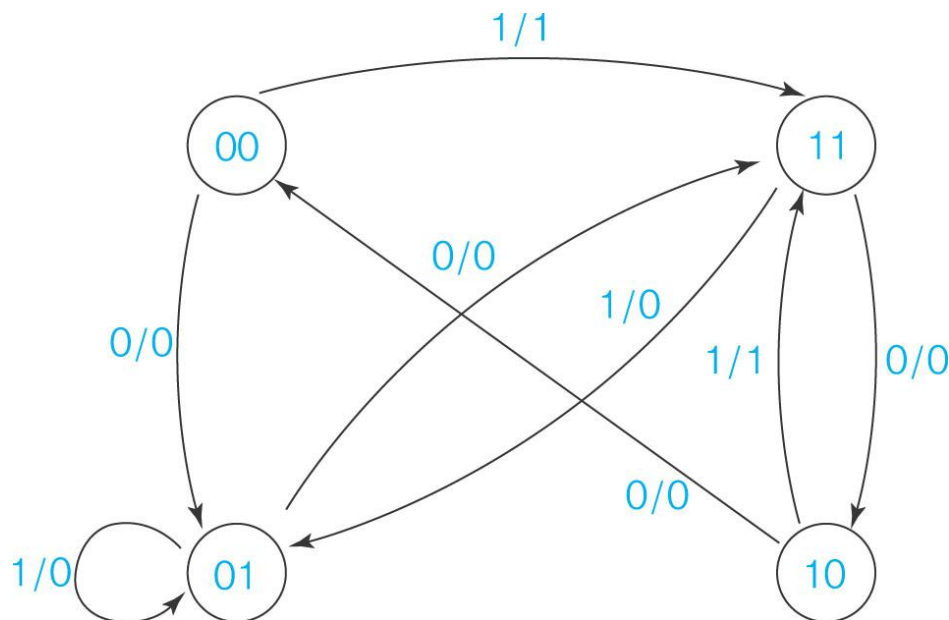
[표 7-4] 그림 7-31의 회로에 대한 다른 형태의 상태표

현재 상태		다음 상태		현재 출력	
<i>A</i>	<i>B</i>	<i>x</i> = 0	<i>x</i> = 1	<i>x</i> = 0	<i>x</i> = 1
		<i>A</i> <i>B</i>	<i>A</i> <i>B</i>	<i>F</i>	<i>F</i>
0	0	0 1	1 1	0	1
0	1	1 1	0 1	0	0
1	0	0 0	1 1	0	1
1	1	1 0	0 1	0	0

두 개의 **D Flip-Flop**이 포함된 **FSM** 분석 2-5

[4] 상태 다이어그램(state diagram)의 작성

- **상태**: 두 비트로 표현된 상태를 포함한 원으로 표시
- **상태 전이**: 화살표를 가진 선과 해당 상태 전이를 야기하는 입력 및 그 순간에 발생하는 출력을 표시 (**입력/출력**)



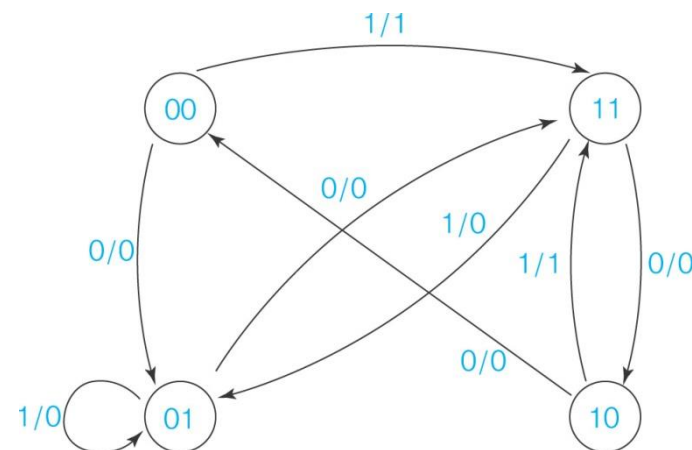
[표 7-4] 그림 7-31의 회로에 대한 다른 형태의 상태표

현재 상태 <i>A B</i>	다음 상태		현재 출력	
	<i>x = 0</i>	<i>x = 1</i>	<i>x = 0</i>	<i>x = 1</i>
	<i>A B</i>	<i>A B</i>	<i>F</i>	<i>F</i>
0 0	0 1	1 1	0	1
0 1	1 1	0 1	0	0
1 0	0 0	1 1	0	1
1 1	1 0	0 1	0	0

두 개의 **D Flip-Flop**이 포함된 **FSM** 분석 2-6

[5] 상태 다이어그램으로부터의 분석

- 그림 같은 동작 특성을 가지는 순차회로에 아래와 같은 입력 신호들이 연속적으로 들어올 때, 상태 전이 및 출력 값은 ?



입력 시퀀스 : 0 1 0 0 1 0 0 1 1 0 1 0 0 1 1
(단, 초기 상태 = 00)

클록 주기	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
현재 상태	00	01	01	11	10	11	10	00	11	01	11	01	11	10	11
입력(x)	0	1	0	0	1	0	0	1	1	0	1	0	0	1	1
다음 상태	01	01	11	10	11	10	00	11	01	11	01	11	10	11	01
출력(F)	0	0	0	0	1	0	0	1	0	0	0	0	0	1	0

Q & A

