



[J02122] 컴퓨터구조

2022년 1학기

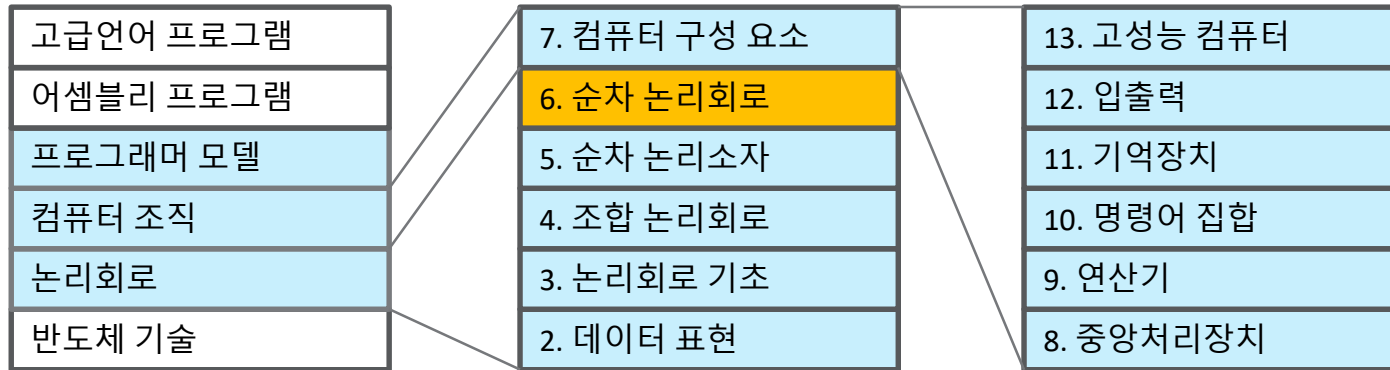
상명대학교 소프트웨어학과 박희민

- 6.1 순차 논리회로 소개
- 6.2 디지털 시스템
- 6.3 처리장치
- 6.4 요약

2022-04-13

CHAP06 순차논리회로

제6장 순차 논리회로

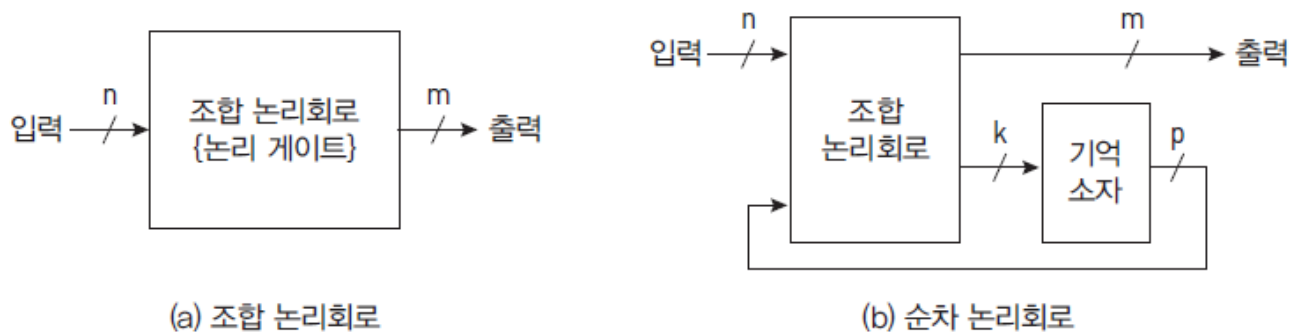


- 학습 목표
 - 상태로 표현되는 순차 논리회로의 동작을 해석할 수 있다.
 - 디지털 시스템에서 레지스터 간 데이터 전송 방법을 설명할 수 있다.
- 내용
 - 6.1 순차 논리회로 소개
 - 6.2 디지털 시스템
 - 6.3 처리장치
 - 6.4 요약

6.1 순차 논리회로 소개

- 순차 논리회로
 - 기억소자(플립플롭, 레지스터)를 포함하고 있는 논리회로
- 학습 목표
 - 순차 논리회로의 구조를 설명할 수 있다.
 - 순차 논리회로의 동작을 표현하는 방법을 이해한다.
- 내용
 - 6.1.1 순차 논리회로 구조
 - 6.1.2 순차 논리회로 동작 표현
 - 6.1.3 순서 검출기

6.1.1 순차 논리회로 구조



[그림 6-1] 조합 논리회로와 순차 논리회로

- 조합 논리회로
 - 출력 = $f(\text{입력})$

- 순차 논리회로
 - 출력 = $f(\text{입력}, \text{현재상태})$
 - 다음상태 = $f(\text{입력}, \text{현재상태})$
- 현재상태(current state)
 - 기억소자의 출력

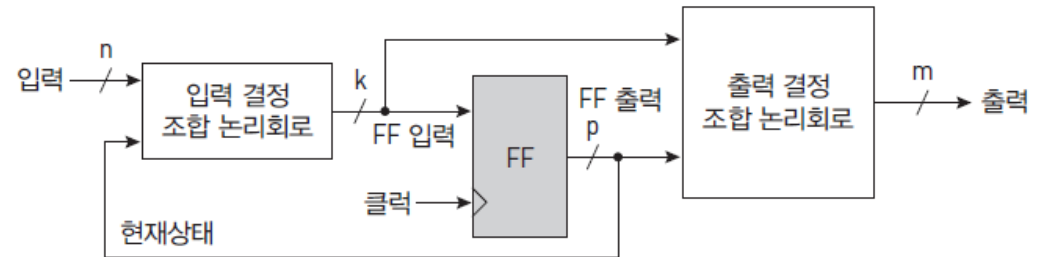
동기 순차 논리회로

- 동기/비동기 디지털 시스템
 - 동기(synchronous): 클록을 공유하고, 플립플롭들이 같은 시간에 동작
 - 비동기(asynchronous): 클록을 공유하지 않고, 플립플롭들이 각자 동작

- 동기 순차논리회로 구조

- Mealy machine

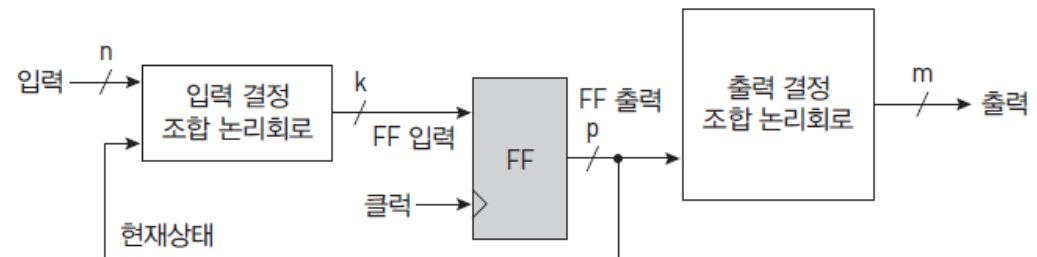
- 출력 = $f(\text{입력}, \text{현재상태})$
 - 다음상태 = $f(\text{입력}, \text{현재상태})$



(a) 밀리 머신

- Moore machine

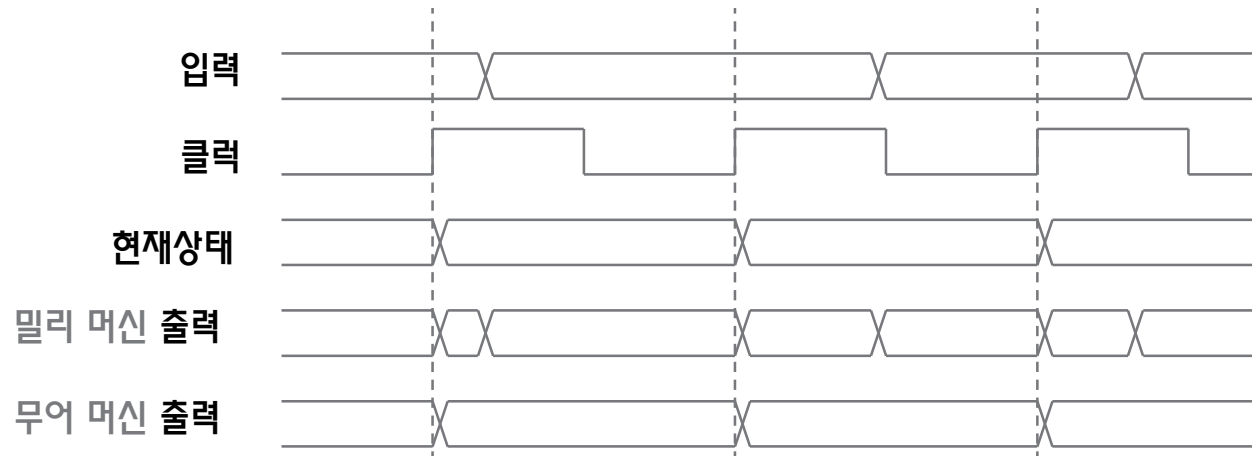
- 출력 = $f(\text{상태})$
 - 다음상태 = $f(\text{입력}, \text{현재상태})$



(b) 무어 머신

밀리 머신과 무어 머신의 동작 타이밍

- Mealy machine
 - 출력 = $f(\text{입력}, \text{현재상태})$
 - 다음상태 = $f(\text{입력}, \text{현재상태})$
- Moore machine
 - 출력 = $f(\text{상태})$
 - 다음상태 = $f(\text{입력}, \text{현재상태})$
 - 대부분의 디지털 시스템은 무어 머신



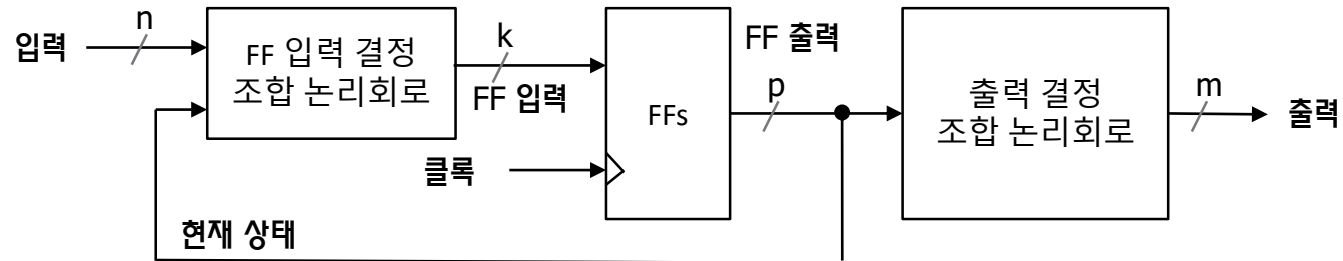
6.1.2 순차 논리회로 동작 표현



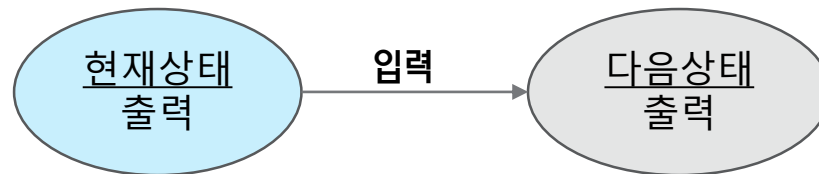
[그림 6-3] 동기 순차회로의 동작 표현

- 상태표
 - 입력, 현재상태, 다음상태, 출력을 나타내는 표
 - 동작 해석이 어렵다/입력 및 출력 논리식을 유도하기 쉽다.
- 상태도
 - 입력, 현재상태, 다음상태, 출력을 나타내는 그림
 - 해석이 쉽다.

무어 머신의 동작 표현



상태도(state diagram)
상태 수 = 2^p

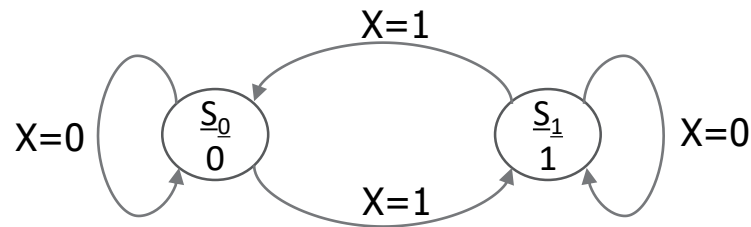


상태표(state table)

	현재상태(p)	입력(n)	다음상태(p)	출력(m)
$2^{(n+p)}$

[예제 6-1] 상태도와 상태표

- 상태도에 대한 상태표 구하기



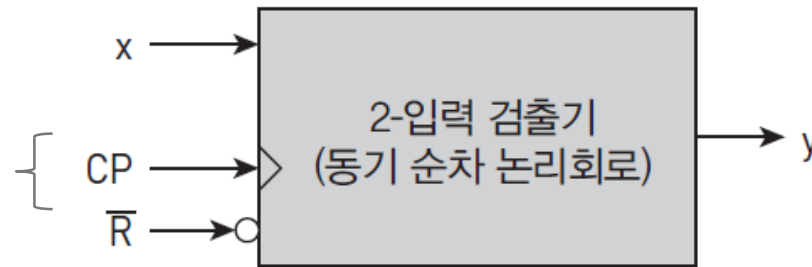
- [상태도]

현재상태	입력(X)	출력(Y)	다음상태
S_0	0	0	
S_0	1	0	
S_1	0	1	
S_1	1	1	

6.1.3 순서 검출기

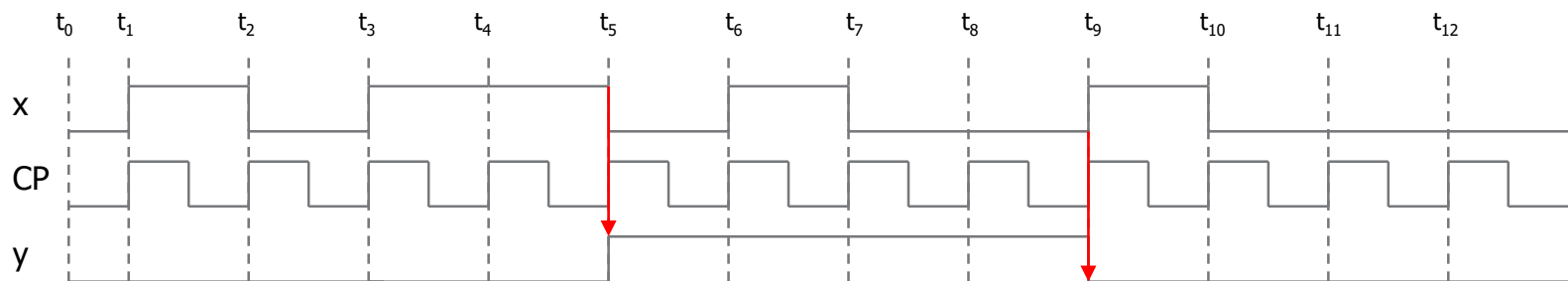
- 순서 검출기(sequence detector)
 - 특정한 입력 패턴이 들어오는 순서를 검출하는 회로
 - 순차 논리회로를 순차 검출기로 사용할 수 있다.
 - 입력이 들어온 순서를 상태로 기억한다.
- 예제: 2-입력 검출기
 - 현재 출력이 0일 때, 1이 연속해서 2번 입력되면 출력을 1로 변경하고,
 - 현재 출력이 1일 때, 0이 연속해서 2번 입력되면 출력을 0으로 변경한다.

2-입력 검출기 입출력 신호



[그림 6-5] 2-입력 검출기의 입출력신호

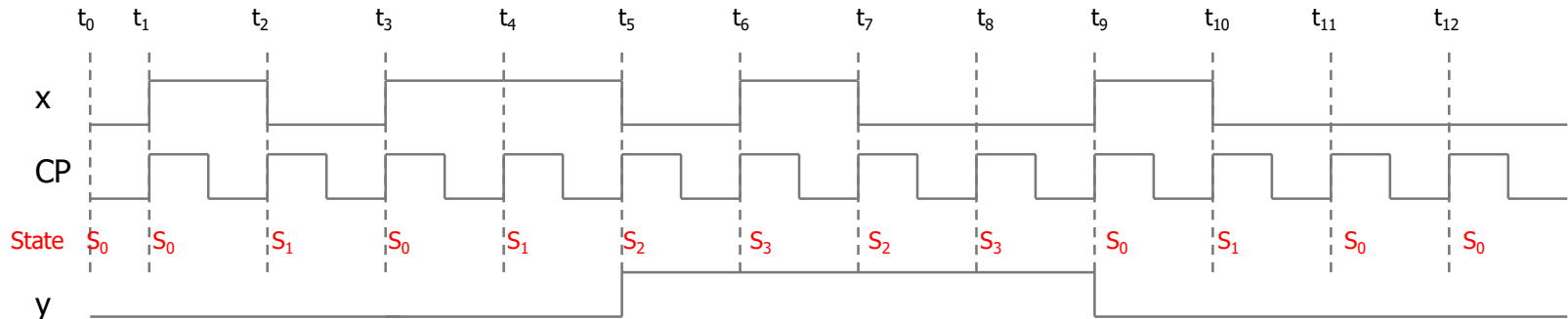
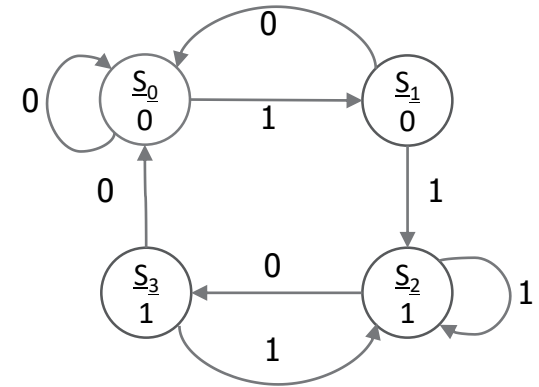
- 동기 순차논리회로
 - 클록 펄스(CP)와 비동기 리셋(/R) 기본
 - x의 값이 두 번 연속해서 출력 y와 다르면 상태 변경
 - 초기 상태를 0으로 가정



<2-입력 검출기 동작 예시>

2-입력 검출기 동작: 상태도

상태/출력	조건	비고
$S_0/0$	출력 1일 때 0이 연속해서 두 번 입력된 상태	초기상태
$S_1/0$	출력 0일 때 1이 한번 입력된 상태	
$S_2/1$	출력 0일 때 1이 연속해서 두 번 입력된 상태	
$S_3/1$	출력 1일 때 0이 한번 입력된 상태	

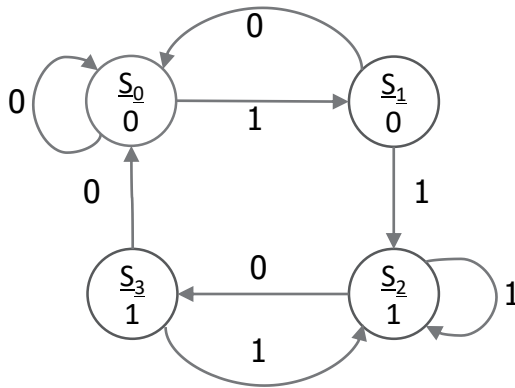


< 2-입력 검출기의 동작 예 (상태 표시) >

2-입력 검출기: 상태표

- 상태에 2진수 할당
 - $S_0 = 00, S_1 = 01, S_2 = 10, S_3 = 11$

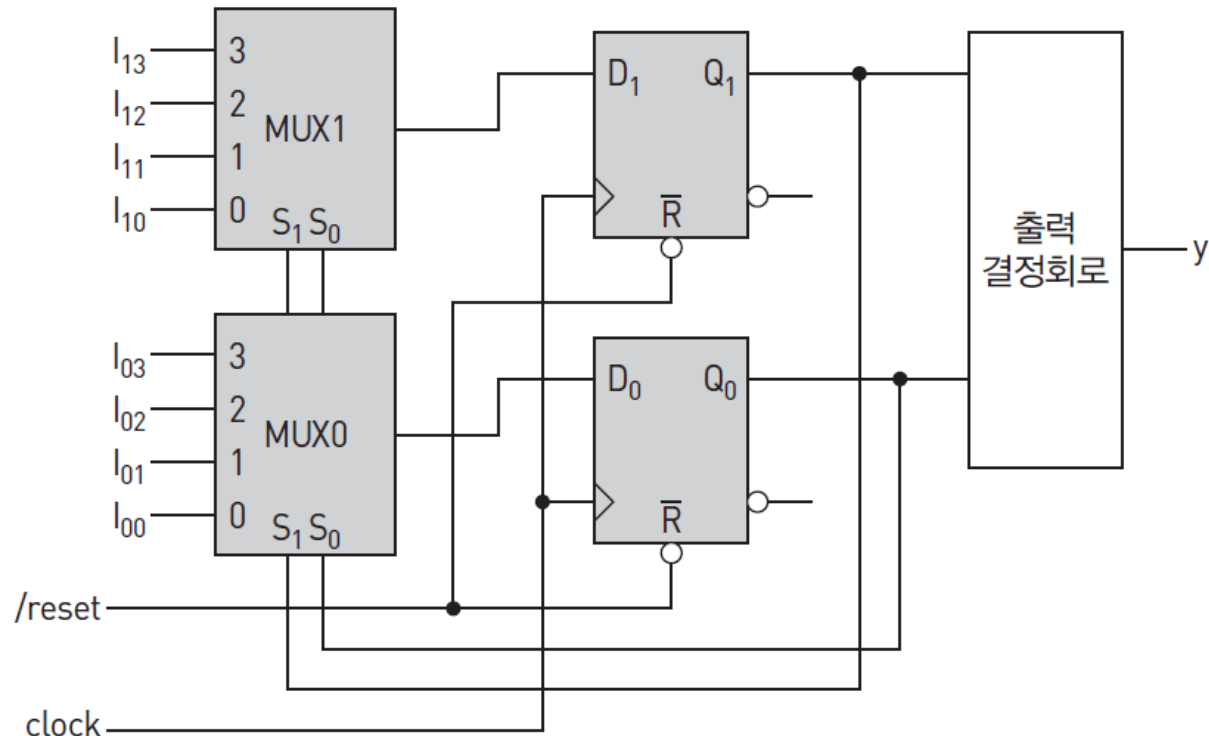
- 상태도 \rightarrow 상태표



현재상태		입력	출력	다음상태	
상태	$Q_1 Q_0$	x	y	상태	$Q_1 Q_0$
S_0	0 0	0	0	S_0	0 0
S_0	0 0	1	0	S_1	0 1
S_1	0 1	0	0	S_0	0 0
S_1	0 1	1	0	S_2	1 0
S_2	1 0	0	1	S_3	1 1
S_2	1 0	1	1	S_2	1 0
S_3	1 1	0	1	S_0	0 0
S_3	1 1	1	1	S_2	1 0

2-입력 검출기 멀티플렉서 구현

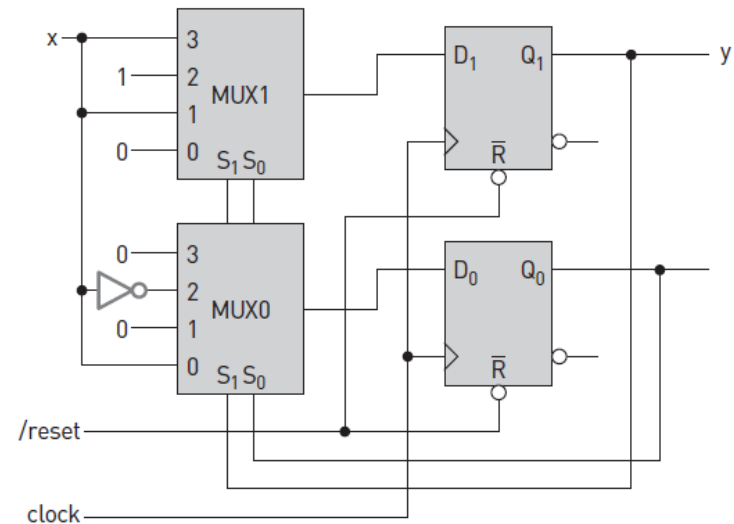
- 회로 형태는 항상 같음
- 현재상태를 멀티플렉서 선택선으로 사용
- (입력 변수가 한 비트일 때) 멀티플렉서 입력 값 결정 필요: 0, 1, x, x' 중 하나



[그림 6-9] 멀티플렉서 구현의 일반 회로

2-입력 검출기 멀티플렉서 구현

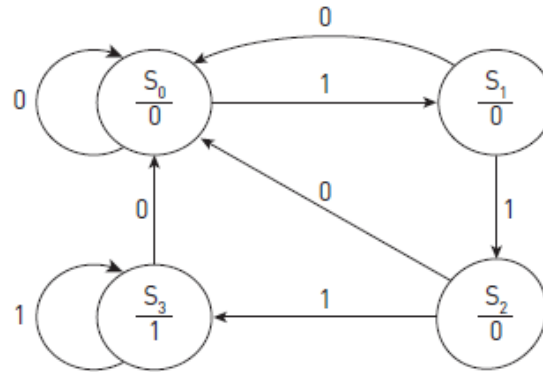
현재상태 ($Q_1 Q_0$)	입력 (x)	다음상태 (Q_1)	MUX1 출력	현재상태 ($Q_1 Q_0$)	입력 (x)	다음상태 (Q_0)	MUX2 출력
S_0	0	0		S_0	0	0	
S_0	1	0		S_0	1	1	
S_1	0	0		S_1	0	0	
S_1	1	1		S_1	1	0	
S_2	0	1		S_2	0	1	
S_2	1	1		S_2	1	0	
S_3	0	0		S_3	0	0	
S_3	1	1		S_3	1	0	



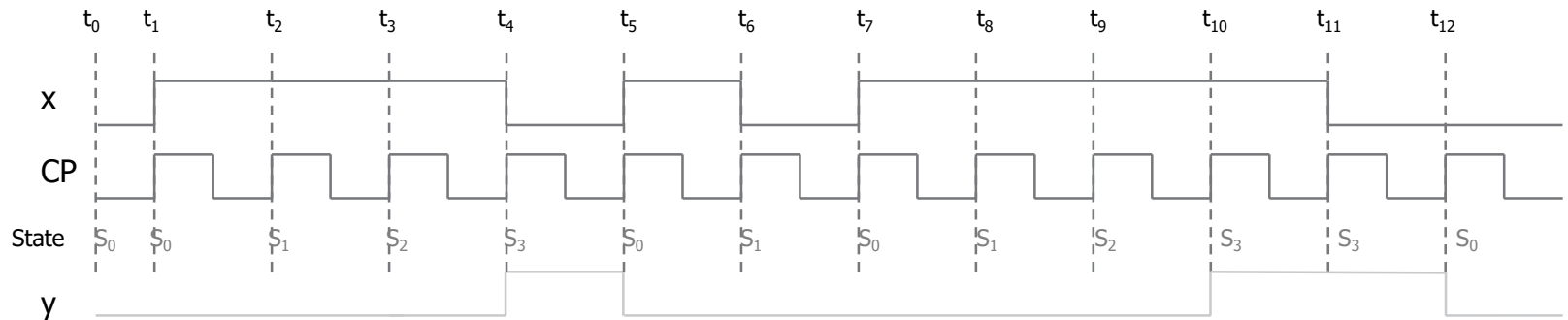
[그림 6-10] 2-입력 검출기의 논리회로도

[예제 6-2] 1-1-1 검출기

< 1-1-1 검출기 상태도 >

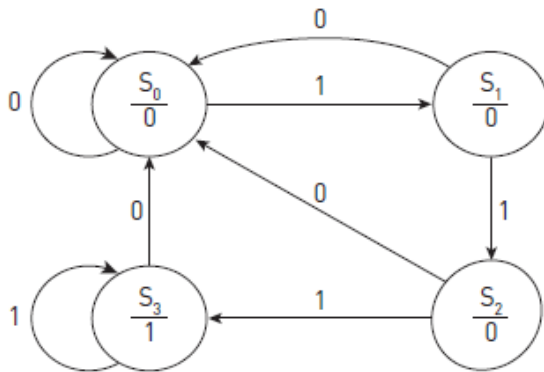


- (1) 입력 신호에 대한 출력 파형과 상태 변화는?



[예제 6-2] 1-1-1 검출기

- (2) $S_0=00$, $S_1=01$, $S_2=10$, $S_3=11$ 로 상태를 할당할 때, 상태표를 구하라.

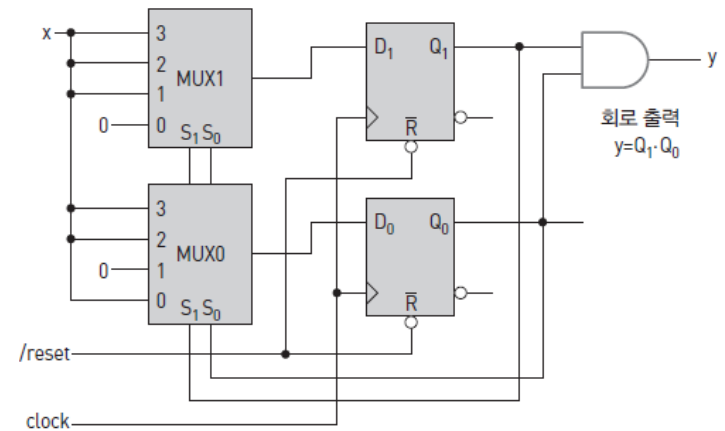


현재상태		입력	출력	다음상태	
상태	$Q_1 Q_0$	x	y	상태	$Q_1 Q_0$
S_0	0 0	0	0		
S_0	0 0	1	0		
S_1	0 1	0	0		
S_1	0 1	1	0		
S_2	1 0	0	0		
S_2	1 0	1	0		
S_3	1 1	0	1		
S_3	1 1	1	1		

[예제 6-2] 1-1-1 검출기

- (3) 멀티플렉서 회로로 구현하라.

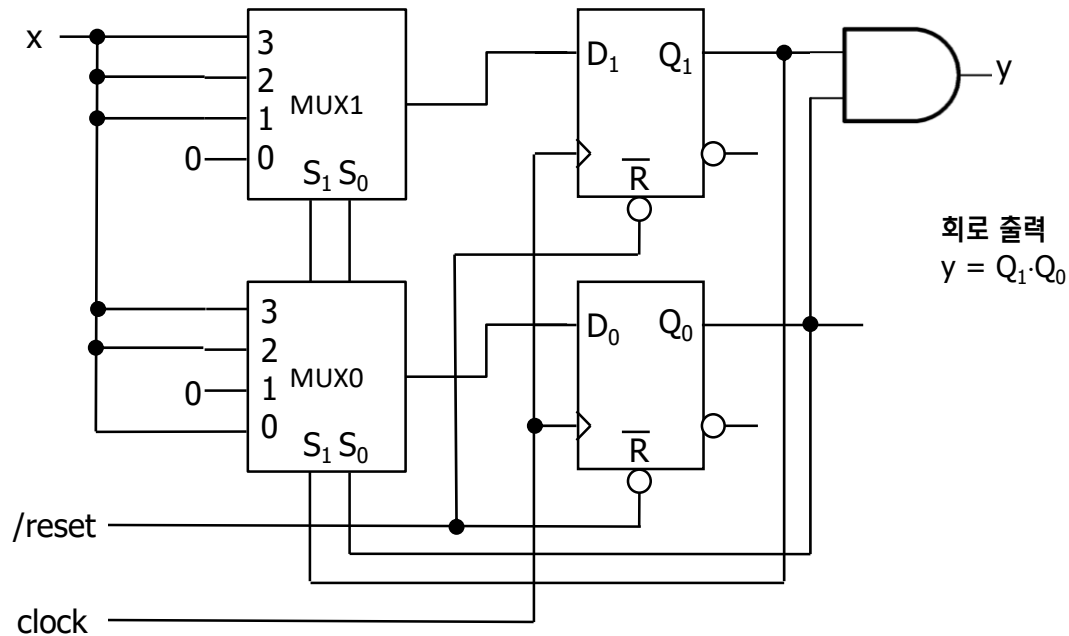
현재상태 ($Q_1 Q_0$)	입력 (x)	다음상태 (Q_1)	MUX1 출력	현재상태 ($Q_1 Q_0$)	입력 (x)	다음상태 (Q_0)	MUX0 출력
S_0	0	0		S_0	0	0	
S_0	1	0		S_0	1	1	
S_1	0	0		S_1	0	0	
S_1	1	1		S_1	1	0	
S_2	0	0		S_2	0	0	
S_2	1	1		S_2	1	1	
S_3	0	0		S_3	0	0	
S_3	1	1		S_3	1	1	



[그림 6-14] 1-1-1 검출기 회로도

출력 논리식: $y = S_3 = Q_1 \cdot Q_0$

[예제 6-2] 1-1-1 검출기



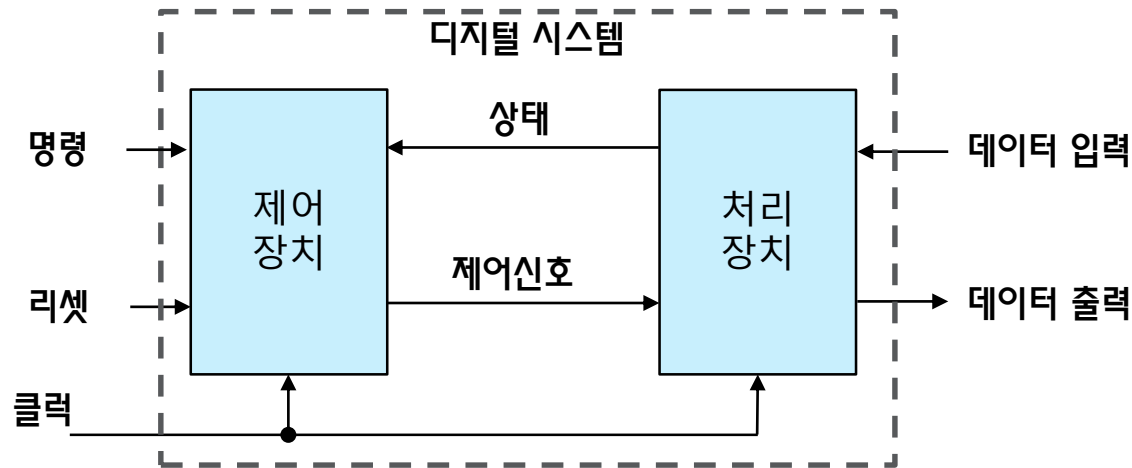
6.1 순차 논리회로 소개 요약

- 순차 논리회로
 - 기억소자 포함
 - 현재상태 = 기억소자의 출력
- 동기 순차 논리회로
 - 시스템 클럭 공유
- 동작 표현
 - 상태도, 상태표
- 동기 순차 논리회로 활용
 - 순서 검출기(sequence detector)
 - 중앙처리장치의 제어기 구현

6.2 디지털 시스템

- 디지털 시스템
 - 디지털 데이터를 처리하는 동기 순차 논리회로
- 학습 목표
 - 디지털 시스템의 구조를 제시하고 기능을 설명할 수 있다.
 - 타이밍 제어기로 실행 시간을 구분하는 방법을 이해한다.
- 내용
 - 6.2.1 디지털 시스템의 구조
 - 6.2.2 타이밍 신호 발생기

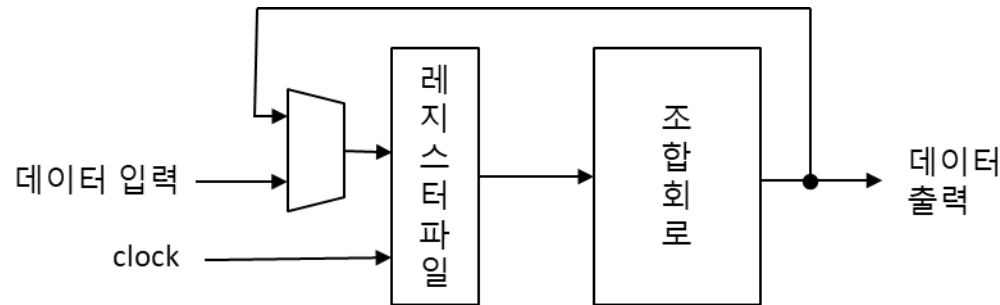
6.2.1 디지털 시스템



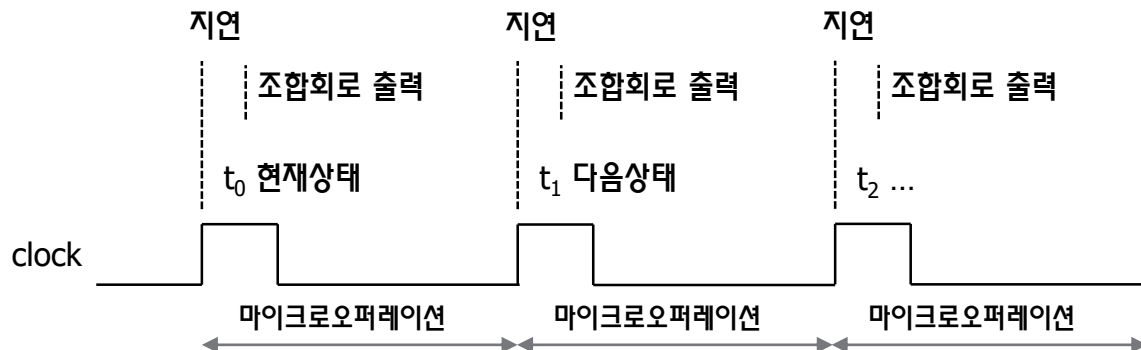
- 동기 순차논리회로
 - 명령(command): 외부에서 동작 지시
 - 제어장치: 명령과 상태(조건)를 참고하여 제어신호 생성
 - 상태도 기반 동작
 - 처리장치: 처리할 데이터를 임시 저장하고 데이터 조작(계산)
 - 비동기 리셋: 시스템 초기화
 - 시스템 클럭: 기억소자 동작 시간 제어

처리장치 동작

- 동기 순차 논리회로 모델

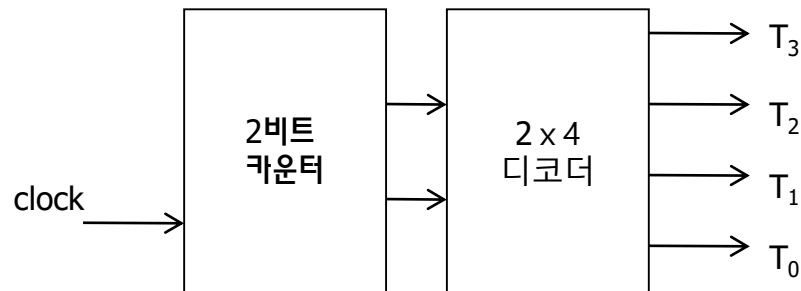
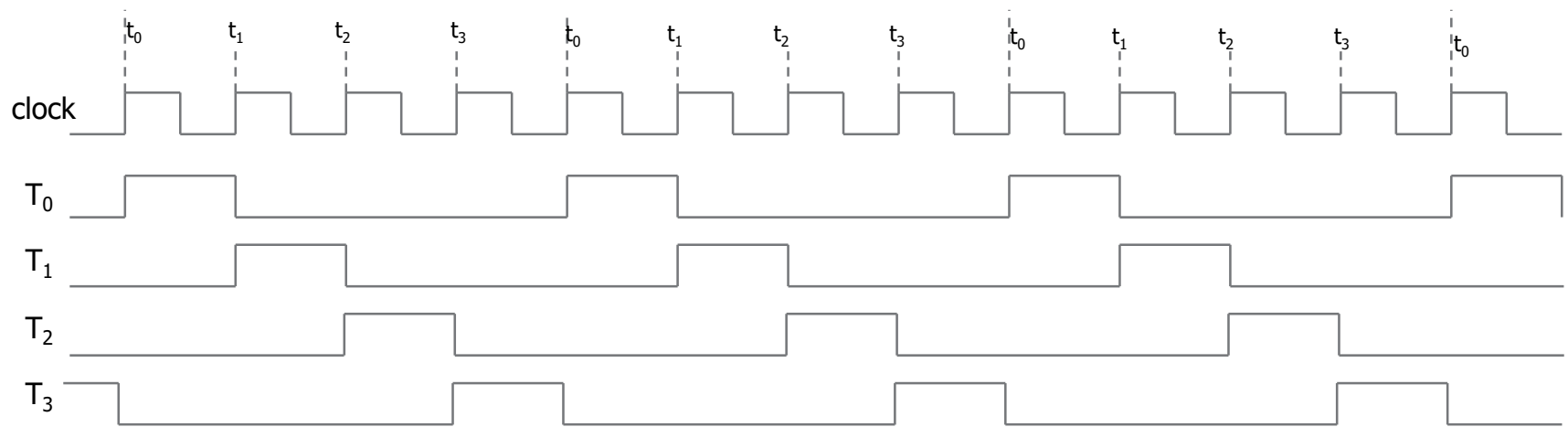


- 마이크로오퍼레이션
 - 한 개의 클럭 구간 동안 수행하는 동작



6.2.2 타이밍 신호 발생기

- 제어장치의 기능 중 하나
- 현재 몇 번째 클럭 펄스가 지나가는지 시간 구분



6.2 디지털 시스템 요약

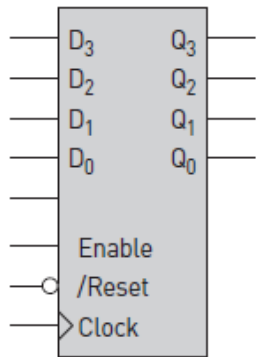
- 제어장치
 - 처리장치로 제어신호 공급
 - 타이밍 신호 발생기 포함: 클럭 펄스를 구분하는 타이밍 신호 제공
- 처리장치
 - 레지스터 파일과 연산기로 구성
 - 데이터 처리 담당
- 마이크로오퍼레이션
 - 한 개의 클럭 구간에서 수행하는 데이터 전달 동작

6.3 처리장치

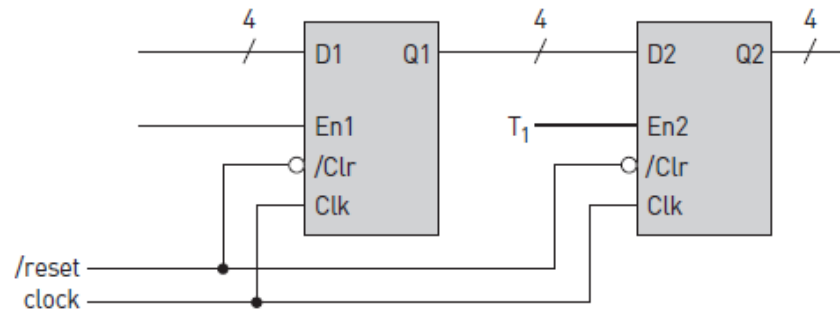
- 처리장치
 - 레지스터의 데이터를 연산기에서 계산
- 학습목표
 - 처리장치의 구조를 제시하고,
 - 타이밍 신호에 따른 데이터 처리와 전달 과정을 설명할 수 있다.
- 내용
 - 6.3.1 레지스터 간 데이터 전송
 - 6.3.2 레지스터 전송 언어

6.3.1 레지스터 간 데이터 전송

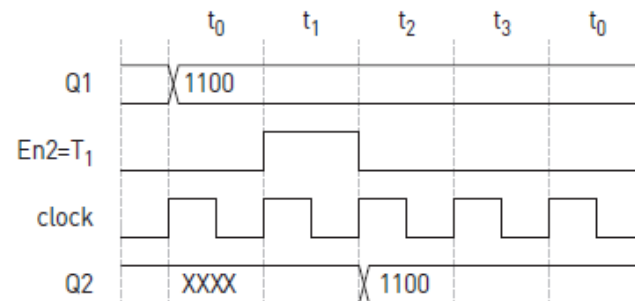
- T1: $R2 \leftarrow R1$
 - R1의 출력을 R2의 입력으로 연결
 - 제어장치의 타이밍 신호 T1을 R2의 적재 가능(enable)에 연결
 - T1 이후에 R1의 값을 R2로 전달



< 4비트 레지스터 >

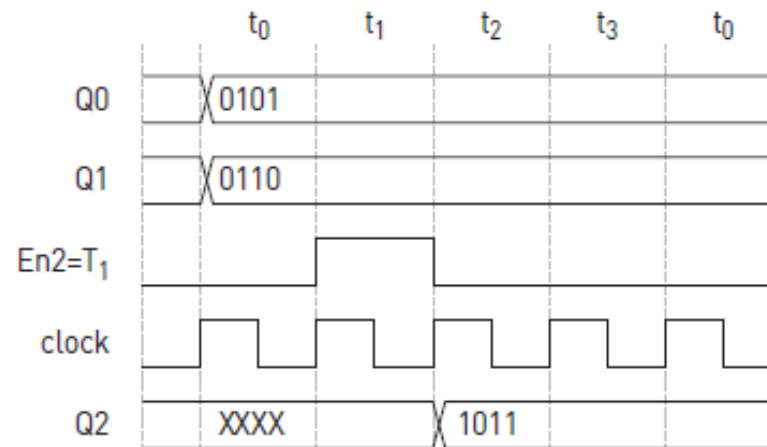
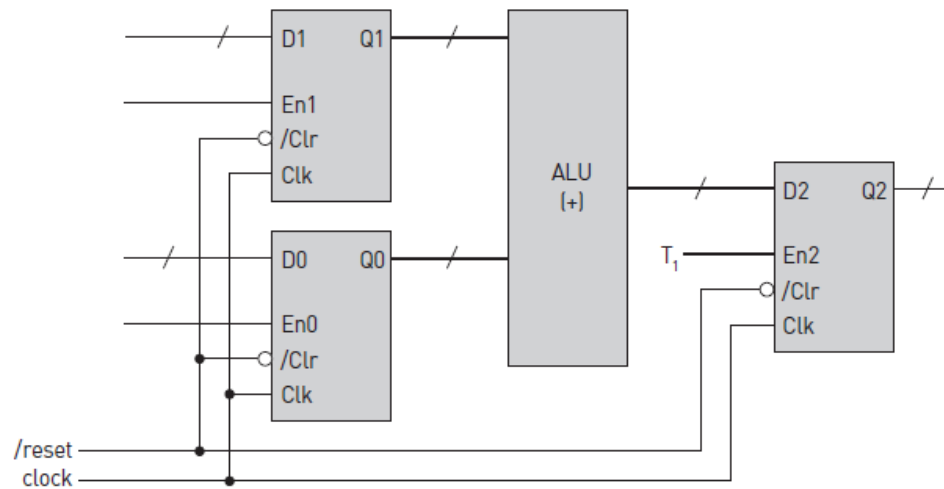


(a) 레지스터 데이터 전달 회로



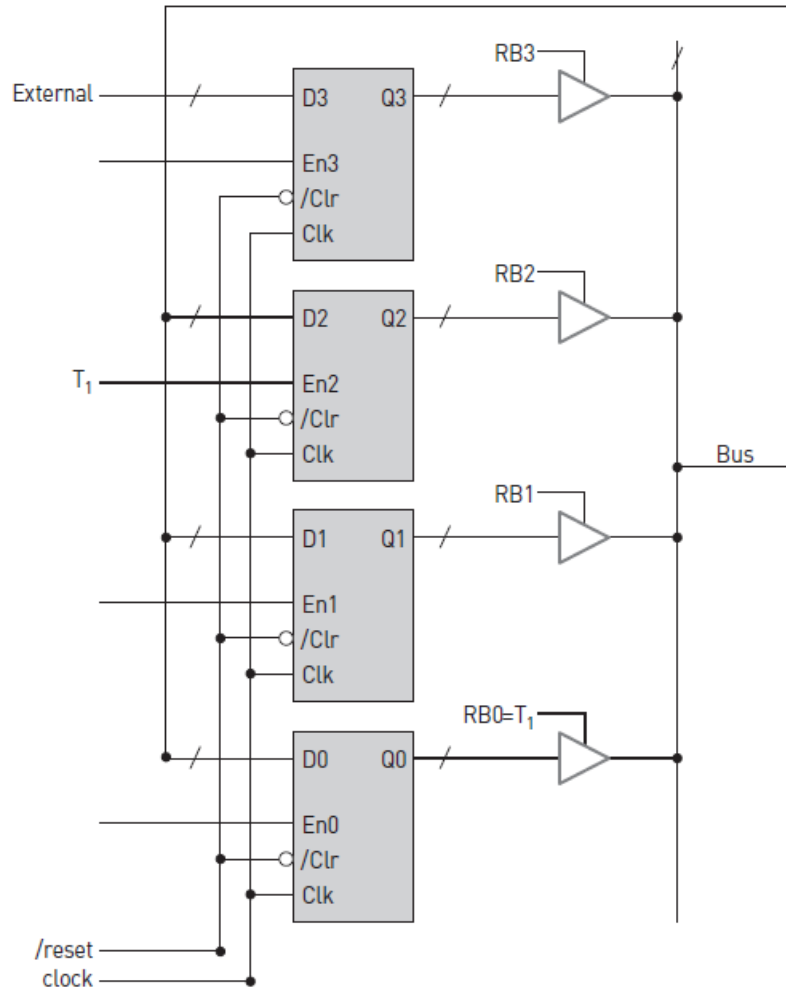
(b) 데이터 전송 타이밍 다이어그램

병렬 덧셈 $T_1: R2 \leftarrow R0 + R1$

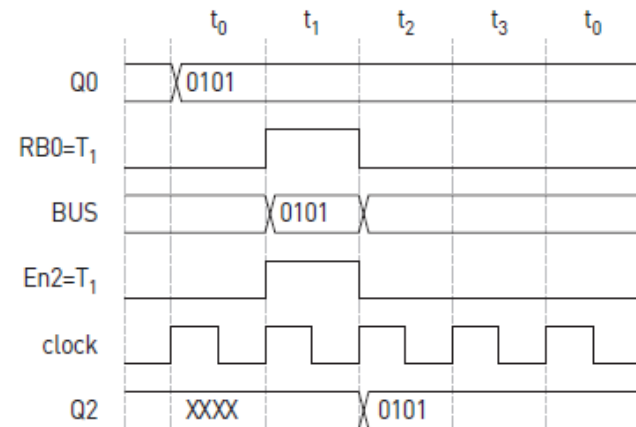


(b) 타이밍 다이어그램

버스 전송 $T_1: R2 \leftarrow R0$

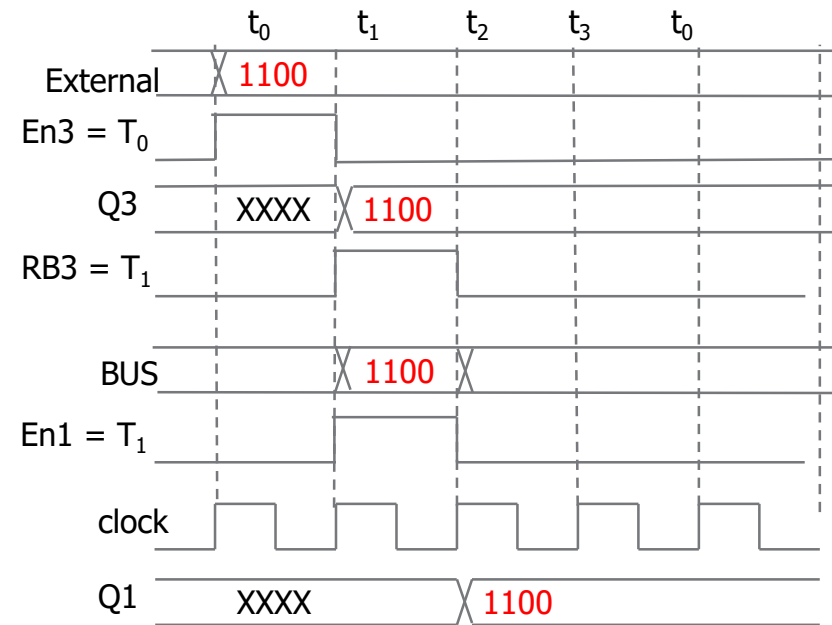
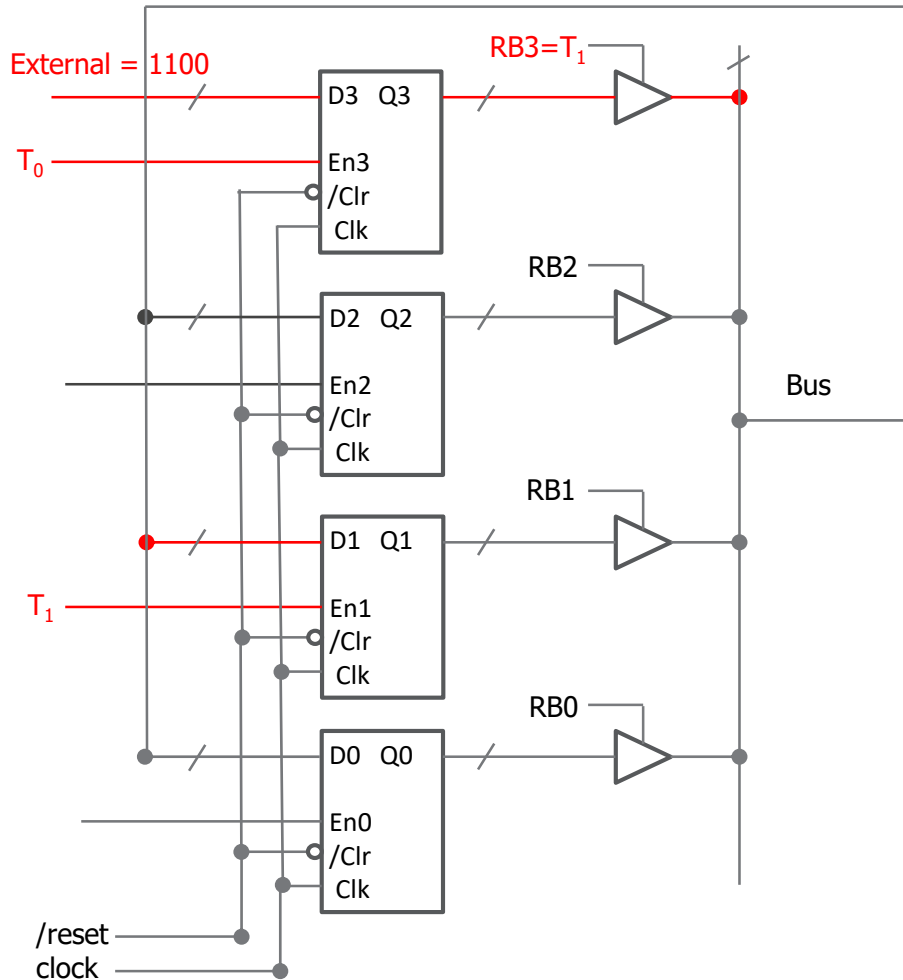


(a) 버스 연결 회로도

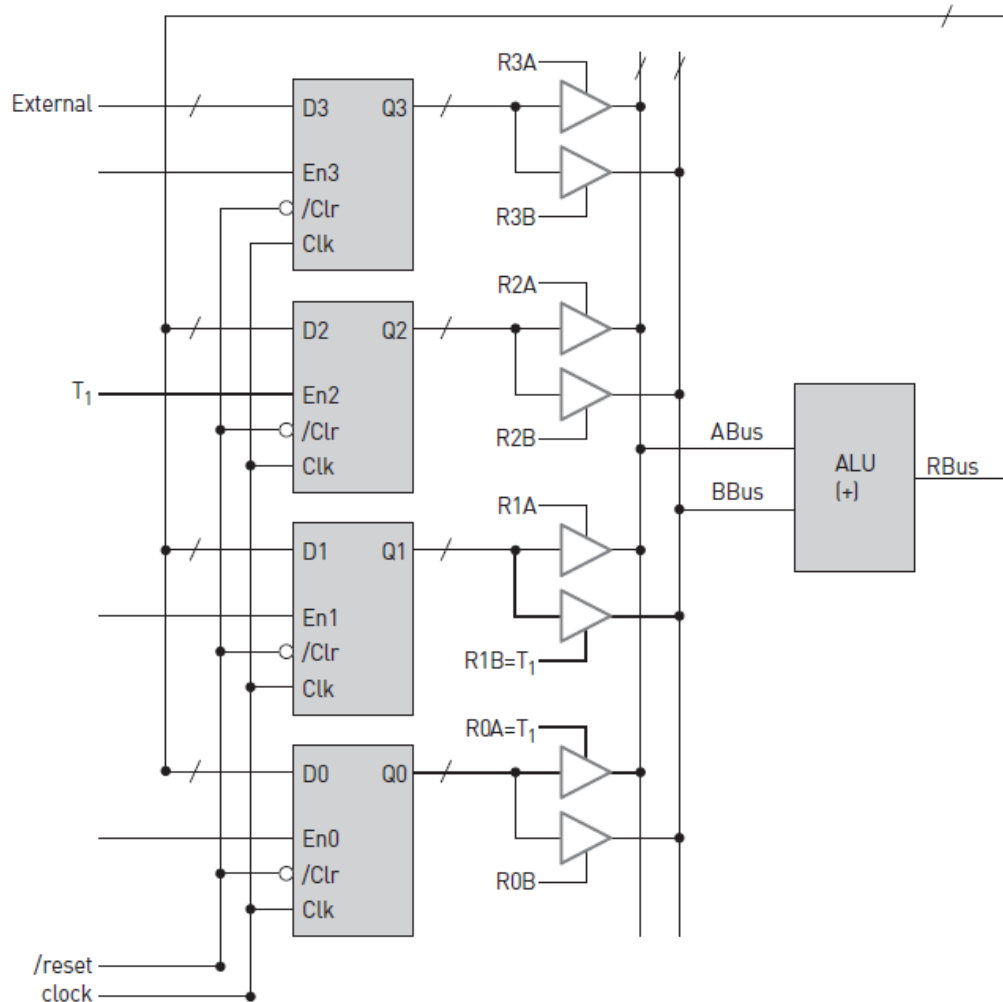


(b) 버스 타이밍 다이어그램

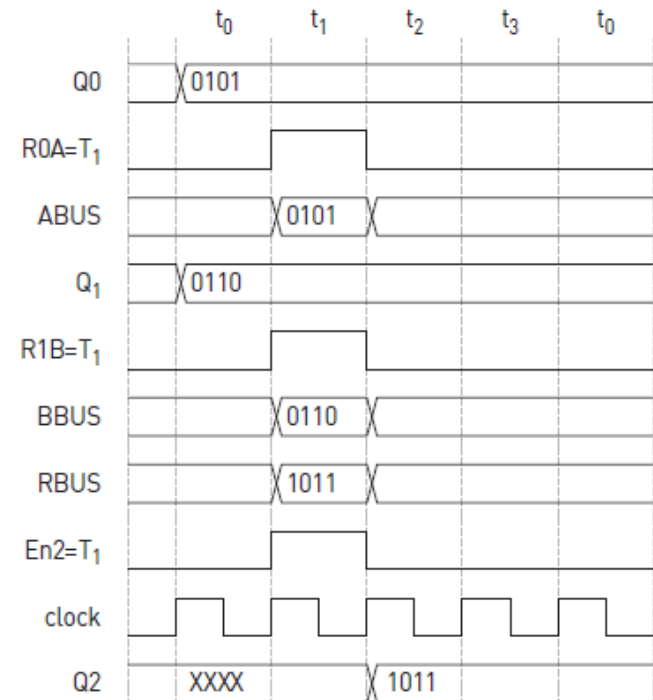
버스 전송 $T_0: R3 \leftarrow \text{External} / T_1: R1 \leftarrow R3$



버스 연산 $T_1: R2 \leftarrow R0 + R1$



(a) 버스 연산 회로도



(b) 연산 타이밍 다이어그램

6.2.2 레지스터 전송 언어

- RTL(Register Transfer Language)
 - 레지스터 간 데이터 처리를 표현하는 문장
 - 하드웨어로 구현 가능해야 함.
- 표현 방법
 - 절대적 규칙은 없고, 자유롭게 표현 가능.

기호	설명	예제
문자(숫자)	레지스터	MAR, PC, R2
괄호	레지스터의 일부	IR[opcode], MBR[7:0]
화살표 \leftarrow	정보의 전달	$R2 \leftarrow R1$
coma (,)	동시 실행 마이크로 동작의 구분	$R2 \leftarrow R1, R1 \leftarrow R2$
콜론 (:)	왼쪽: 동작 실행 조건 또는 시간 오른쪽: 실행할 동작	$T_3: AC \leftarrow AC + MAR$ $T_1+T_2: R1 \leftarrow R2 \text{ AND } R3$

6.3 처리장치 요약

- 처리장치
 - 레지스터 파일과 연산기로 구성되어 있다.
 - 제어장치가 제공하는 타이밍 신호를 제어신호로 받는다.
- 타이밍 신호
 - 어느 레지스터를 출력할 것인지
 - 어떤 연산을 수행할 것인지
 - 연산 결과를 어느 레지스터에 적재할 것인지 결정한다.
- 레지스터 전송 언어
 - 일종의 의사 코드(pseudo code)
 - 디지털 시스템의 동작 표현

6.3 요약

6.1 순차 논리회로 소개

- 동기 순차 논리회로
- 밀리 머신: 다음상태 = $f(\text{현재상태}, \text{입력})$, 출력 = $f(\text{현재상태})$

6.2 디지털 시스템

- 제어장치/처리장치
- 타이밍 신호 발생기: 시간 구분

6.3 처리장치

- 레지스터 전송 타이밍
- 버스 구조
- 레지스터 전송 언어

• 제7장 컴퓨터 구성 요소

- 컴퓨터 구성 요소(중앙처리장치, 기억장치, 입출력장치)의 역할과 기능