12주차 예비보고서

전공: 컴퓨터공학과 학년: 3학년 학번: 20212022 이름: 이예준

**1.**

Register는 여러 bit의 데이터를 일시 저장하거나 전송하는 장치로 배열된 bit들을

이동시킨다. 여러 개의 Flipflop을 연결하여 구성하기 때문에 n-bit Register는 n개의

Flipflop으로 구성되어 있고 n-bit의 Binary 데이터를 저장한다. 이때 입력 Pulse가 가해짐에

따라 미리 인코딩된 상태들이 주기적으로 순환하는 Register를 Counter라고 하며,

Register의 특수한 상태라고 볼 수 있다. Counter의 주된 특성은 Timing, Sequencing,

Counting이다. 그 밖에도 여러 기능이 있는데 Up, Down 두가지 기능은 말 그대로 수치를

순서대로 더해서 계수하는 기능과 수치를 빼면서 계수하는 기능이며 또한 순차회로로써

입력 Pulse의 수를 Counting하는 기능도 있다. 입력 Pulse에 따라 발생한 여러 Process나

Event들의 횟수를 저장하여 계산 목적으로 여러 디지털 전자제품에 사용되는데, 그 예로는

타이머, 주파수 분배기, 주파수 계수기 등이 있다.

**2.**

Decade Counter는 Decimal Counter의 대표적인 Counter로 Decimal Counter는 0~9까지 10개의 상태를 Counting한다. 10진수를 2진 코드로 표현하기 위해서는 4-bit가 필요하기 때문에 총 4개의 Flipflop으로 구성된다. 아래는 Decade Counter의 회로도와

각 카운터 순서에 따른 2진 코드를 작성한 표이다.

텍스트, 스크린샷, 직사각형, 폰트이(가) 표시된 사진

자동 생성된 설명

\*Decade Counter 회로도

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 카운터 순서 | | | | |
| 순서 | X | Y | Z | W |
| 1 | 0 | 0 | 0 | 0 |
| 2 | 0 | 0 | 0 | 1 |
| 3 | 0 | 0 | 1 | 0 |
| 4 | 0 | 0 | 1 | 1 |
| 5 | 0 | 1 | 0 | 0 |
| 6 | 0 | 1 | 0 | 1 |
| 7 | 0 | 1 | 1 | 0 |
| 8 | 0 | 1 | 1 | 1 |
| 9 | 1 | 0 | 0 | 0 |
| 10 | 1 | 0 | 0 | 1 |

\*Decade Counter 순서

위의 회로도는 JK-Flipflop으로 구성된 Counter 회로도이며, J와 K 입력이 모두 1인 이유는

Clock Pulse에 의해서만 값이 바뀌거나 유지시켜야 하기 때문이다. 첫번째 Flipflop의 CLK을

제외하고 다른 Flipflop의 CLK은 이전 Flipflop의 출력과 연결되어 있다. 또한 NAND gate가

두번째와 네번째 Flipflop의 출력을 입력으로 받고 출력은 CLR에 연결되어 있다. 두번째와 네번째 Flipflop의 출력이 모두 1일 때 1010(10)인 뜻으로 NAND gate에서 0이 출력으로 나오면서 모든 Flipflop을 초기화시켜준 뒤 가시 0~9까지 Counting한다.

**3.**

Counter는 크게 두가지 종류로 구분할 수 있는데 Asynchronous Counter와 Synchronous Counter가 있다. Asynchronous Counter는 Clock Pulse에 모든 Flipflop이 동기화되어 있지 않고 동작하는 Counter로 Ripple Counter라고도 한다. 따라서 처음 Flipflop만 Main Clock에 의해서 작동하며 나머지 Flipflop은 이전 Flipflop의 출력에 의해 작동한다. 위에서 설명한 Decade Counter가 그 예이다.

텍스트, 도표, 평면도, 라인이(가) 표시된 사진

자동 생성된 설명

\*2-bit Asynchronous Counter Diagram

위 그림을 보게 되면 CLK이 작동한 뒤 Q0 Counter가 작동하고, Q0의 출력을 받아

Q1 Counter가 작동하는 것을 볼 수 있다. Asynchronous Counter는 주파수 분배기로써

활용되어 실제 높은 Clock 주파수를 더 낮고 안정적으로 사용 가능한 주파수로 줄일 수 있다. 실생활에서 디지털 전자, 타이밍 관련 어플리케이션 및 디지털 Clock, 인터럽트 소스를 제공한다. 하지만 단점도 있는데 Flipflop을 재동기화하기 위해 추가적인 Flipflop이 필요하며, bit수가 많아질수록 Ripple형식이기 때문에 전파 Delay가 길어진다. 또한 높은 Clock 주파수가 적용될 경우 계산 오류가 있을 수 있다.

Asynchronous Counter에서 발생하는 전파 Delay를 해결하기 위한 회로가 Synchronous

Counter이며, Asynchronous Counter와 달리 Synchronous Counter는 모든 Flipflop을

작동시키는 하나의 Global Clock이 존재한다. 아래 그림을 보면 Clock이 Falling Edge Trigger를 가지고 있어 1에서 0으로 바뀔 때 신호를 입력받는다. H는 항상 0이기 때문에

J와 K가 항상 1이며, Clock을 입력받을 때 첫번째 Flipflop의 출력은 반전된다.

또한 첫번째 Flipflop의 출력이 1일 때만 두번째 Flipflop의 J와 K 입력이 1이 되며, 역시 두번째 Flipflop의 출력이 반전된다. 마지막 Flipflop은 두번째가 1일때 반전되면 010일 때 출력이 1이 되면서 정상적인 Counting이 되지 않기 때문에 첫번째와 두번째 Flipflop이 모두 1일 때만 세번째 Flipflop의 J와 K 입력이 1이 될 수 있도록 첫번째와 두번째 Flipflop의 출력을 AND gate로 묶어서 세번째 Flipflop으로 연결한다. 이렇게 하여 0~7까지 정상적인 Counting하는 것을 밑에 있는 Timing Diagram을 보면 알 수 있다.

\*Counter는 순차 회로이기 때문에 Flipflop을 하나의 Clock으로 엮었다고 Flipflop이 동시에

작동하는 것은 아니다.

도표, 텍스트, 평면도, 낱말맞추기 퍼즐이(가) 표시된 사진

자동 생성된 설명

\*3-bit Synchronous Counter Diagram

**4.**

FSM은 Finite-State Machine의 약자로 여러 제한을 가진 조건부 상태의 시스템의 수학적 모델을 설명하는 용어이다. FSM은 취할 수 있는 유한개의 상태를 가지며 그 중에서

단 하나의 상태를 가지고, 그 상태를 기준으로 다음으로 어떤 동작을 수행할지 결정한다.

현재 상태는 특정 조건에 따라 다른 상태로 변하기 때문에 FSM은 취할 수 있는 유한 개의 상태의 집합과 각 상태들이 전이할 수 있는 조건들로 구성된다. FSM의 장점은 직관적이며 프로그래머 외의 제 3자가 쉽게 확인 및 설계할 수 있다는 장점이 있다. 단점으로는 상태의 개수가 늘어날수록 추가된 전이조건들을 각 상태들과 연결하는 것이 매우 복잡해진다.

원, 도표, 폰트, 텍스트이(가) 표시된 사진

자동 생성된 설명

\*FSM의 State Diagram

FSM은 크게 Moore Machine과 Mealy Machine로 나뉘는데 Moore Machine은 출력이 현재

상태에 의해서만 결정되며, Mealy Machine은 출력이 현재 상태와 입력에 의해서 결정된다.

Moore Machine의 장점은 행위를 단순화할 수 있다는 장점이 있다. 예를 들어 전구의 불을

키고 끌 때 Moore Machine으로 표현하면 “전구가 꺼진 상태”, “전구가 켜지는 상태”,

“전구가 켜진 상태”, “전구가 꺼지는 상태” 이렇게 표현할 수 있다. 그래서 직관적이다는

장점도 있다. Mealy Machine의 장점은 상태의 개수를 줄일 수 있다는 장점이 있다. 위의

전구 예를 다시 들어 Mealy Machine으로 표현하면 “전구가 꺼진 상태”, “전구가 켜진 상태”

이렇게 두가지 상태로 표현 가능하고, “전구가 켜지는 상태”, “전구가 꺼지는 상태”는 없다.

단 입력값으로 전구를 키는 명령어와 전구를 끄는 명령어가 있다.

**5.**

FSM을 표현할 때 추상기계(Abstract Machine)로 표현하는데, 추상기계는

실재하는 것일지라도 그것을 재현하기 위해 기능하지 않고, 새로운 유형 즉, 현실에

존재하지 않는 것을 구축하는데도 사용된다. 표현할 때 가장 대표적인 예가 Diagram이다.

또한 현실에 존재하지 않는 것을 구축하는 것의 예로는 “복소수”를 들 수 있다.

위에서 FSM에 추상기계를 사용한 방법은 컴퓨터의 동작을 추상화하는데에 적용되었다.