12주차 예비보고서

전공: 컴퓨터공학과 학년: 3학년 학번: 20191599 이름: 송경호

**1.**

Counter는 디지털 논리 분야에서 특정 이벤트 혹은 프로세스의 발생 횟수를 추적하고 저장하거나 표시해주는 장치이다. Counter의 가장 일반적인 형태는 Flip-Flop 관련 회로에서 언급되었던 clock을 입력으로 갖고 출력을 발생시키는 회로이다. 이때 출력은 이진수 혹은 BCD number이며 clock의 pulse에서 counter를 증가/감소시킨다.

또한 Counter는 일반적으로 여러 개의 Flip-Flop을 이어서 구성한다. 이러한 Counter는 별도의 회로로 제작되어 사용되기도 하고, Integrated Circuit의 일부분으로 사용되는 등 회로에 굉장히 빈번히 사용된다. 이 때 여러 Filp-Flop을 이어서 구성하는 가장 일반적인 형태의 sequential digital logic은 모든 Flip-Flop이 공통된 clock을 사용하는지 혹은 각각의 Flip-Flop이 별도의 clock을 사용하는 지에 따라 각 각 synchronous sequential logic과 asynchronous sequential logic으로 나뉜다.

Asynchronous sequential logic counter의 경우 외부 clock에 의해 trigger 되는 첫 비트를 제외하고 나머지 flip-flop은 이전 비트에 의해 trigger된다. 따라서 외부 clock이 첫 비트를 trigger한 뒤 다음 flip-flop들은 연쇄적으로 trigger되며 상태가 변경된다. 예를 들어, 외부 clock에 의해 trigger 받는 0번째 bit가 1번째 bit를 trigger시키고 이는 다시 2번째 bit를 trigger 시키는 방식이다.

Synchronous sequential logic counter의 경우 모든 Flip-Flop의 입력 clock이 서로 연결되어 있기 때문에 하나의 공통 clock에 의해 동시에 trigger된다. 즉, 모든 Flip-Flop은 병렬적으로 상태가 변경된다.

Counter에는 다양한 형태가 존재한다. 먼저 2번에서 설명할 Decade Counter는 10진수로 표현되는 숫자인 0부터 9까지의 숫자를 세는 카운터이다. 다음으로 Ring Counter는 n개의 flip-flop이 이어진 형태이며 마지막 flip-flop의 출력이 첫 flip-flop의 입력이 된다는 특이한 형태이다. 결과적으로 n clock 주기마다 특정 상태가 반복된다. 마지막으로 Johnson Counter는 링 카운터의 변형 형태로 마지막 flip-flop의 출력이 첫 flip-flop의 입력이 되기 전 반전되어 들어간다는 특징이 있다. 이는 D Flip-Flop이나 JK Flip-Flop을 이용하여 비교적 쉽게 구현 가능하다.

counter는 다양한 형태로 존재하며, clock 신호를 활용하여 여러 응용 분야의 이벤트 및 프로세스를 효과적으로 추적하고 계산하는 능력을 갖고 있다.

**2.**

앞서 말했듯이, Decade Counter(십진 카운터)는 10진수로 표현된 숫자인 0부터 9까지의 숫자를 세는 카운터이다. 따라서 0붙 9까지 10개의 서로 다른 상태를 가지게 된다.

Decade Counter는 일반적으로 4개의 JK Flip-Flop을 이용해 구현한다. 이때 각 Flip-Flop은 이전 Flip-Flop의 출력을 입력으로 받아들이고, 0부터 9까지의 이진수를 나타내기 위해 한 번에 한 비트씩 증가한다는 특징이 있다. 또 clock이 각 Flip-Flop에 도달할 때마다 상태가 전이되고 9를 나타내는 1001에 도달한 이후 다음 주기에는 0으로 리셋 된다는 특징이 있다. 회로도는 다음과 같다.

텍스트, 스크린샷, 도표이(가) 표시된 사진

자동 생성된 설명

회로도를 확인하면 4 개의 JK Flip-Flop이 존재하고 각 출력이 다음의 입력이 된다. 또한 10 이상의 값이 발생하는지 modular 연산을 통해 확인하고 이를 NAND Gate를 통해 Clear bit에 전달하여 Flip-Flop을 0으로 초기화한다. 따라서 Decade Counter의 진리표와 Timing Diagram은 아래와 같다.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Clk Pulse** | **Q3** | **Q2** | **Q1** | **Q0** | **Dec** |
| 1 | 0 | 0 | 0 | 0 | 0 |
| 2 | 0 | 0 | 0 | 1 | 1 |
| 3 | 0 | 0 | 1 | 0 | 2 |
| 4 | 0 | 0 | 1 | 1 | 3 |
| 5 | 0 | 1 | 0 | 0 | 4 |
| 6 | 0 | 1 | 0 | 1 | 5 |
| 7 | 0 | 1 | 1 | 0 | 6 |
| 8 | 0 | 1 | 1 | 1 | 7 |
| 9 | 1 | 0 | 0 | 0 | 8 |
| 10 | 1 | 0 | 0 | 1 | 9 |
| **11** | **0** | **0** | **0** | **0** | **0** |

**텍스트, 스크린샷, 음악이(가) 표시된 사진

자동 생성된 설명**

Clock Pulse의 하강 엣지 마다 첫 비트를 나타내는 Flip-Flop의 값이 변하며 이 값이 계속해서 연쇄적으로 다음 Flip-Flop으로 전달되는 것을 알 수 있다. 또한 진리표와 Timing Diagram 모두에서 1001 이후에 reset pulse가 발생하여 count값이 0으로 초기화되는 것을 확인할 수 있다.

**3.**

비동기식 Counter는 1번 문항에서 설명했던 Asynchronous sequential logic counter를, 동기식 Counter는 Synchronous sequential logic counter를 의미한다.

**3-1) Asynchronous**

먼저 비동기식 counter는 각 Flip-Flop이 별도의 clock 신호를 갖고 상태 전이가 독립적으로 이루어진다. 또한 비동기식 카운터에는 상향 카운터와 하향 카운터 두 가지 주요 형태가 존재하는데 이 둘은 카운터의 증가 또는 감소 방향에 따라 구분된다.

먼저 상향 카운터는 가장 낮은 최소 카운트값에서 카운트 값을 증가시키는 카운터이다. 이때 카운트가 최대값에 도달한 뒤 다시 최소값으로 돌아가며 순환하는 구조를 가진다는 특징이 있다. 이를 이진수로 표현하면 000, 001, 010, 011, …, 111, 000 의 증가와 순환 구조를 가지는 것이다.

스크린샷, 블랙, 흑백, 사각형이(가) 표시된 사진

자동 생성된 설명

상향 카운터의 회로도는 위와 같은 방식으로 이루어지며, 가장 좌측의 Flip-Flop으로만 clock pulse 입력이 이루어지며 해당 Flip-Flop의 출력이 다음 Flip-Flop의 입력으로 사용되는 연쇄적인 구조를 갖는다. 이에 따라 Timing Diagram은 아래와 같다.

스크린샷, 다채로움이(가) 표시된 사진

자동 생성된 설명

clock의 하강 엣지에서 첫 Flip-Flop의 출력인 Q0의 값이 변화하며 해당 값은 다음 Flip-Flop에 입력으로 들어간다. 따라서 이에 따라 Q1의 값도 변화한다. 결과적으로 clock의 하강 엣지에서 Q­0, Q­1, Q­2, Q­3의 값이 변화하며 0000 -> 0001 -> 0010 -> … -> 1111의 순으로 변화한다. 이후에는 다시 Q­0, Q­1, Q­2, Q­3의 값이 모두 0이 되며 Count가 0이 된다. 이를 진리표로 나타내면 아래와 같다.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Q3** | **Q2** | **Q1** | **Q0** | **Count** |
| 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 0 | 1 | 1 |
| 0 | 0 | 1 | 0 | 2 |
| 0 | 0 | 1 | 1 | 3 |
| …. | | | | |
| 1 | 1 | 1 | 0 | 14 |
| 1 | 1 | 1 | 1 | 15 |

다음으로 하향 카운터는 가장 낮은 최대 카운트값에서 카운트 값을 감소시키는 카운터이다. 이때 카운트가 최소값에 도달한 뒤 다시 최대값으로 돌아가며 순환하는 구조를 가진다. 즉 상향 카운터의 완전히 반대 방식으로 작동한다. 해당 회로는 상향 카운터의 회로에서 counter입력의 Not 연산을 제거하여 구현할 수 있다.

**3-2) Synchronous**

동기식 Counter는 비동기식과 달리 모든 Flip-Flop에 공통으로 Clock을 연결하고 신호를 동시에 가해 병렬적인 연산 처리가 가능하다.

스크린샷, 블랙, 흑백, 디자인이(가) 표시된 사진

자동 생성된 설명

위는 동기식 카운터의 회로이며, 하나의 CP가 모든 Flip-Flop에 입력되는 것을 알 수 있다. 따라서 비동기식 카운터의 경우 이전 Flip-Flop이 이후 Flip-Flop으로 들어가 마치 이전에 다룬 4 bit sequential adder처럼 전파 지연이 발생하지만 동기식 Counter는 전파지연이 발생하지 않는다는 장점이 있다.

**4.**

FSM은 Finite State Machine의 준말로 유한 상태 기계 혹은 유한 오토마타 Finite Automaton이라고 불린다. 이는 프로그램 설계와 전자 논리 회로 설계에 사용되는 수학적 모델로 상태와 상태의 전이로 이루어진다. 이때 Finite State인 만큼 유한개의 상태를 가지며 한 번에 오직 하나의 상태만을 가질 수 있다는 특징이 있고 해당 상태 간의 변화를 transition이라고 부른다.

FSM은 크게 두가지 유형으로 나눌 수 있다.

**4-1) Moore Machine**

Moore Machine은 현재 상태에 따라 출력이 달라지는 유형이다. Moore Machine은 입력에 영향을 받지 않으므로 현재 상태에 따라 고정된 출력을 가진다. 예를 들어, 주어진 상태 A에 대하여 항상 출력 X를, B에 대하여 항상 출력 Y가 발생한다.

**4-2) Mealy Machine**

다음으로 Mealy Machine은 입력에도 출력이 영향을 받는다. 따라서 상태가 같아도 입력에 따라 다양한 출력을 가질 수 있다. 앞선 위의 예시에서 주어진 상태가 A여도 입력에 따라 출력이 반드시 X가 아니게 된다.

**텍스트, 도표, 폰트, 라인이(가) 표시된 사진

자동 생성된 설명텍스트, 원, 도표, 폰트이(가) 표시된 사진

자동 생성된 설명**

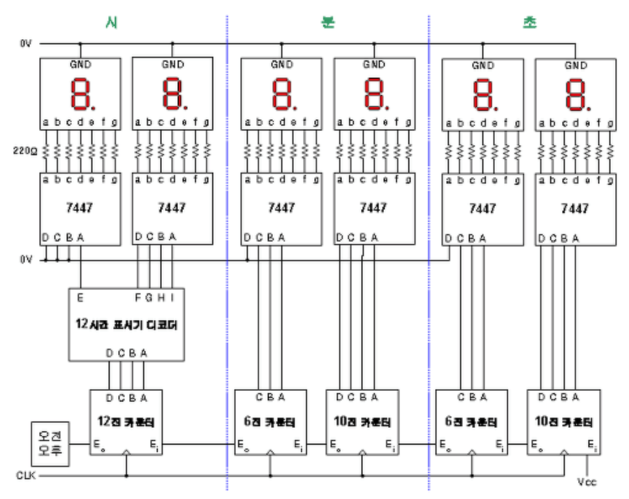
위 두사진은 Moore Machine과 Mealy Machine의 차이를 나타낸다. 좌측이 Moore Machine을 우측이 Mealy Machine을 의미한다. 결과만을 보았을 때, Mealy Machine의 경우 입력에 따라 다양한 출력을 발생시킬 수 있기 때문에 전체 상태의 수를 줄일 수 있다는 장점이 있지만 입력을 고려해야 하기 때문에 Moore Machine에 비해 덜 직관적이고 복잡성이 높아진다는 단점이 있다.

FSM는 게임 AI, 자동제어 시스템, 프로토콜 디자인, 컴파일러 구성 등 다양한 분야에서 활용된다. 실제 현재 컴퓨터공학 전공자 3,4학년을 대상으로 한 기초 컴파일러 구성 강의 내에 FSM을 활용하여 컴파일러의 FE에서 parser와 lexer에 대한 코드를 자동으로 생성해주는 시스템에 대해 공부할 수 있다. FSM은 상태 전이를 시각적으로 표현하기 쉽고 정해진 규칙에 따라 상태 전이가 발생하므로 안정적인 프로그램 구현에 도움을 준다.

**5.**

**5-1) Counter의 활용**

Counter가 일상생활에서 사용되는 가장 대표적인 예시는 디지털 시계이다. 이전에 배운 7 segment display와 BCD-to-7 Segment decoder, 그리고 counter를 이용하여 디지털 시계를 구성한다. counter를 통해 발생한 값을 decoder를 통해 display에 전달하는 방식으로 이루어진다.



이외에도 주파수 측정기, 주소 생성기 등 주기적으로 동작을 제어해야 하는 다양한 분야에서 counter가 활발하게 사용된다.

**5-2) Moore, Mealy model 구성**

Flip-Flop을 이용하여 두 모델 회로를 구성할 수 있다.

먼저 D Flip-Flop을 이용한 Moore model 구성이다.

도표, 평면도, 기술 도면, 라인이(가) 표시된 사진

자동 생성된 설명 텍스트, 스크린샷, 번호, 폰트이(가) 표시된 사진

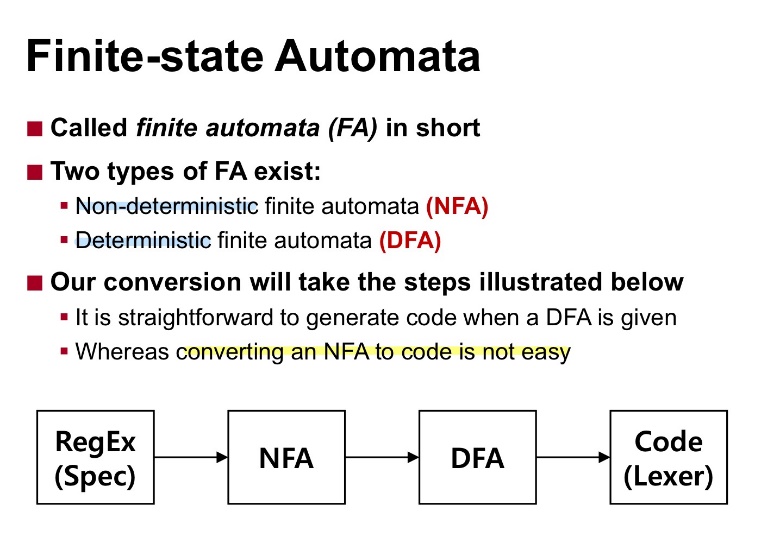
자동 생성된 설명

다음으로 Mealy model 구성이다.

도표, 라인, 평면도, 기술 도면이(가) 표시된 사진

자동 생성된 설명

**5-3) Compiler Construction**

****

컴파일러 구성 시 정규표현식을 입력받아 NFA와 DFA를 거쳐 Lexer 코드를 생성한다.