5주차 결과보고서

전공 : 컴퓨터공학과 학년 : 2학년 학번 : 20211558 이름 : 윤준서

**1. 실험 목적**

• De-Morgan의 정리와 Boolean 함수의 동작을 이해하고 확인한다.

• Verilog를 사용하여 De-Morgan의 정리와 Boolean 함수의 동작을 구현한다.

• 입력 신호 생성 후 Simulation을 통해 구현된 결과를 확인한다.

• FPGA를 통해서 Verilog로 구현된 회로의 동작을 확인한다.

**2. De-Morgan Simulation**

De-Morgan 제 1법칙을 논리식으로 표현하면 다음과 같다.

**(A • B)' = A' + B'**

Verilog를 통해 구한 Schematic과 Simulation 결과는 다음과 같다.

**도표, 라인, 그래프이(가) 표시된 사진

자동 생성된 설명도표, 라인, 그래프, 디자인이(가) 표시된 사진

자동 생성된 설명**

스크린샷, 사각형이(가) 표시된 사진

자동 생성된 설명

NAND 게이트는 AND 게이트에 NOT 게이트를 적용한 것으로, **(A • B)'**와 형태가 일치하다. 따라서 **A NAND B**는 De-Morgan 제 1법칙 **(A • B)' = A' + B'**과 동일하다.

De-Morgan 제 2법칙을 논리식으로 표현하면 다음과 같다.

**(A + B)' = A' • B'**

Verilog를 통해 구한 Schematic과 Simulation 결과는 다음과 같다.

**도표, 라인, 그래프, 디자인이(가) 표시된 사진

자동 생성된 설명도표, 라인, 디자인이(가) 표시된 사진

자동 생성된 설명**

스크린샷, 텍스트, 사각형이(가) 표시된 사진

자동 생성된 설명

NOR 게이트는 OR 게이트에 NOT 게이트를 적용한 것으로, **(A + B)'**와 형태가 일치하다. 따라서 **A NOR B**는 De-Morgan 제 2법칙 **(A + B)' = A' • B'**과 동일하다.

**3. (A' + B') • C' = ((A • B) + C)'와 (A' • B') + C' = ((A + B) • C)' Simulation**

Verilog를 통해 구한 **(A' + B') • C'** 의 Schematic과 Simulation 결과는 다음과 같다.

도표, 평면도, 라인이(가) 표시된 사진

자동 생성된 설명

스크린샷, 도표, 사각형이(가) 표시된 사진

자동 생성된 설명

Verilog를 통해 구한 **((A • B) + C)'** 의 Schematic과 Simulation 결과는 다음과 같다.

도표, 라인, 그래프, 평면도이(가) 표시된 사진

자동 생성된 설명

스크린샷, 사각형이(가) 표시된 사진

자동 생성된 설명

De-Morgan의 정리에 의해 **(A' + B') • C'** 과 **((A • B) + C)'** 이 같으므로 Simulation 결과가 일치하는 것을 확인할 수 있다.

Verilog를 통해 구한 **(A' • B') + C'** 의 Schematic과 Simulation 결과는 다음과 같다.

도표, 라인, 평면도, 그래프이(가) 표시된 사진

자동 생성된 설명

스크린샷, 도표, 사각형이(가) 표시된 사진

자동 생성된 설명

Verilog를 통해 구한 **((A + B) • C)'** 의 Schematic과 Simulation 결과는 다음과 같다.

도표, 라인, 평면도, 그래프이(가) 표시된 사진

자동 생성된 설명

스크린샷, 사각형, 디스플레이이(가) 표시된 사진

자동 생성된 설명

De-Morgan의 정리에 의해 **(A' • B') + C'** 과 **((A + B) • C)'** 이 같으므로 Simulation 결과가 일치하는 것을 확인할 수 있다.

**4. One-Bit Comparison Simulation**

One-Bit 비교기의 논리식과 Boolean 식은 다음과 같다.

**A = B Y = ~(A ^ B)**

**A ≠ B Y = A ^ B**

**A > B Y = A & ~B**

**A < B Y = ~A & B**

Verilog를 통해 구한 Schematic과 Simulation 결과, 진리표는 다음과 같다.

도표, 평면도이(가) 표시된 사진

자동 생성된 설명

스크린샷, 사각형이(가) 표시된 사진

자동 생성된 설명

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| A | B | A = B | A ≠ B | A > B | A < B |
| 0 | 0 | 1 | 0 | 0 | 0 |
| 0 | 1 | 0 | 1 | 0 | 1 |
| 1 | 0 | 0 | 1 | 1 | 0 |
| 1 | 1 | 1 | 0 | 0 | 0 |

**5. 결과 검토 및 논의 사항**

One-Bit 비교기를 구현할 때, 입력 변수의 값이 0 또는 1이기 때문에 비트 연산으로 표현이 가능하다는 것을 알 수 있다.

**A = B**와 **A ≠ B**의 경우, 두 입력 값이 다를 때 1을 출력하는 **XOR 게이트**를 이용하여 나타낼 수 있다.

**A > B**와 **A < B**의 경우, 결과 값이 1인 경우는 하나 뿐이므로 **AND 게이트**를 이용하여 나타낼 수 있다.

**6. 추가 이론 및 추가 작성**

NAND 게이트와 NOR 게이트는 De-Morgan의 정리와 형태가 일치하다는 것을 확인했다. 트랜지스터의 개수가 적을 수록 높은 성능을 발휘하는 논리 회로의 특성 상, 상대적으로 사용하는 트랜지스터의 개수가 적은 NAND 게이트와 NOR 게이트를 주로 사용하는 것이 유리하다. 즉 논리 회로에서 De-Morgan의 정리를 이용해 그 효율성을 추구한다고 볼 수 있다.

논리 회로를 간소화할 때 POS 식을 SOP 식으로 변환하거나, SOP 식을 POS 식으로 변환하는 과정이 중요한데, 이 과정을 De-Morgan을 통해 쉽게 진행할 수 있다.

예시로 SOP **Y = A'B' + CDE** 을 POS로 변형하는 과정은 다음과 같다.

1. 양변에 NOT을 취한다. **Y' = (A'B' + CDE)'**

2. De-Morgan의 정리를 두 번 사용하여 식을 정리한다. **Y' = (A'B')'(CDE)' = (A + B)(C' + D' + E')**

이처럼 NOT을 취하는 것만으로도 De-Morgan의 정리에 의해 SOP 식을 POS 식으로 변환할 수 있다.