**실험 2. 불 대수식의 단순화**

2022. 03. 25

20210054 정하우

1. **개요**

불 대수식을 단순화 하는 방법인 카노 맵(Karnaugh-Map, K-Map)과 퀸 매클러스키 (Quine-McCluskey, QM) 알고리즘이 있음을 알고, 2-Bit Magnitude Comparator 불 대수식을 카노 맵을 이용하여 직접 단순화해본다. 그리고 단순화 전후로 회로의 차이를 Verilog를 통해 구현해 확인한다. 세부적인 학습 목표는 다음과 같다.

- K-map 알고리즘 이해

- 와이어와 논리 게이트 개수를 확인하여 단순화의 효과 확인

1. **이론적 배경**
2. 불 대수식의 단순화

불 대수식은 단순화를 통해 gate와 wire 개수를 줄일 수 있다. 이를 줄임으로서 전력효율을 높힐 수 있다는 장점이 있다. 불 대수식은 분배, 결합법칙이나 드모르간의 법칙등으로 단순화가 가능하다. 이 외에 2-level combinational logic을 단순화 하는데 쓰이는 카노 맵(Karnaugh-Map, K-Map)과 퀸 매클러스키 (Quine-McCluskey, QM) 알고리즘이 있다.

이번 실험에서는 카노 맵을 사용할 예정이다.

1. 2-Bit Magnitude Comparator

2-Bit Magnitude Comparator는 2Bit 크기의 두 수(0~3)의 대소관계를 비교하는 회로이다. A>B 이면 GT가 1을, A=B 이면 EQ는 1을, A<B 이면 LT가 1을 출력한다. 조건에 맞지 않으면 0을 출력한다.

1. **실험 준비**
2. 2-Bit Magnitude Comparator의 세 출력 각각에 대한 식을 단순화하지 않고 작성한다.
3. A>B(GT)

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 입력 | | A1A0 | | | |
| 00 | 01 | 11 | 10 |
| B1  B0 | 00 | 0 | 1 | 1 | 1 |
| 01 | 0 | 0 | 1 | 1 |
| 11 | 0 | 0 | 0 | 0 |
| 10 | 0 | 0 | 1 | 0 |

1. A=B(EQ)

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 입력 | | A1A0 | | | |
| 00 | 01 | 11 | 10 |
| B1  B0 | 00 | 1 | 0 | 0 | 0 |
| 01 | 0 | 1 | 0 | 0 |
| 11 | 0 | 0 | 1 | 0 |
| 10 | 0 | 0 | 0 | 1 |

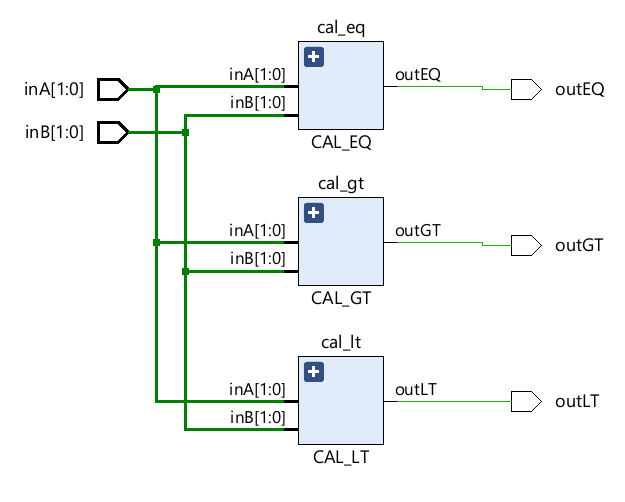
1. A<B(LT)

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 입력 | | A1A0 | | | |
| 00 | 01 | 11 | 10 |
| B1  B0 | 00 | 0 | 0 | 0 | 0 |
| 01 | 1 | 0 | 0 | 0 |
| 11 | 1 | 1 | 0 | 1 |
| 10 | 1 | 1 | 0 | 0 |

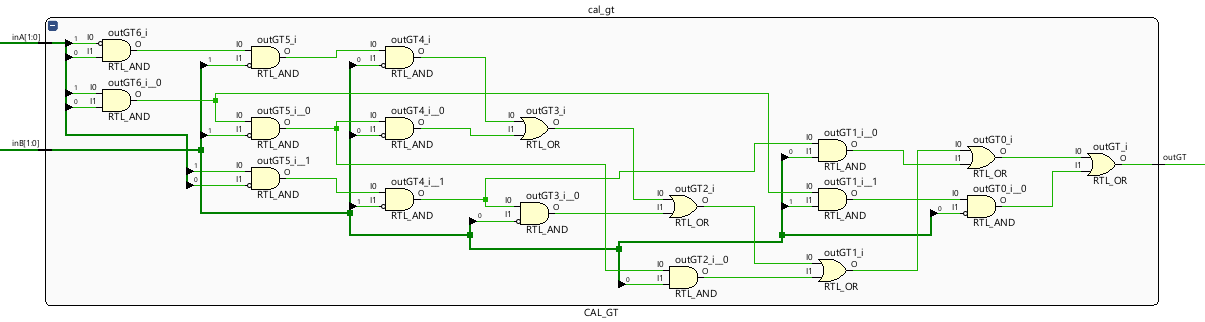
1. K-map을 활용하여 세 식을 단순화한다.
2. A>B(GT)
3. A=B(EQ)
4. A<B(LT)
5. **결과**
6. 단순화 이전
7. 2-Bit Magnitude Comparator를 단순화하기 전 식을 구현한다.

실험준비에서 구한 식을 assign keyword와 “~”,“&”,“|” operator를 이용하여 나타내었다.

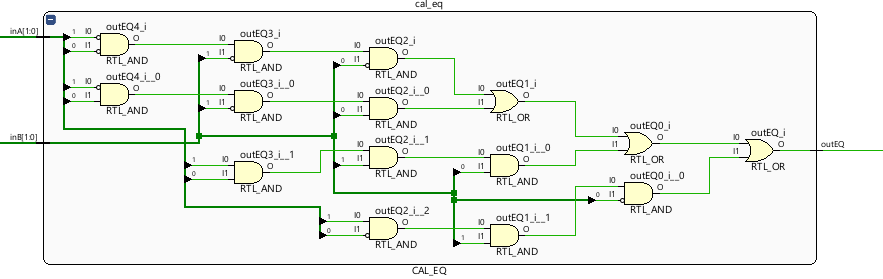
1. Schematic 기능으로 회로가 잘 구현되었는지 확인한다.

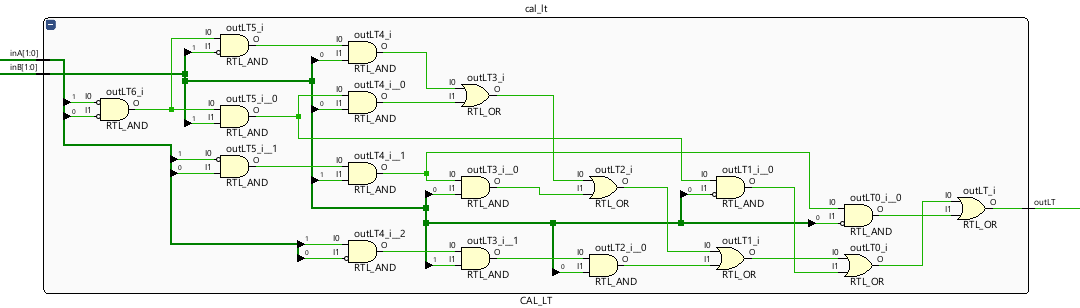


1. A>B(GT)



1. A=B(EQ)



1. A<B(LT)
2.  RTL ANALYSIS > Netlist를 참조하여 와이어와 논리 게이트 개수를 확인한다.
3. A>B(GT)

와이어 : 22개

논리 게이트 : 18개

1. A=B(EQ)

와이어 : 19개

논리 게이트 : 15개

1. A<B(LT)

와이어 : 22개

논리 게이트 : 18개

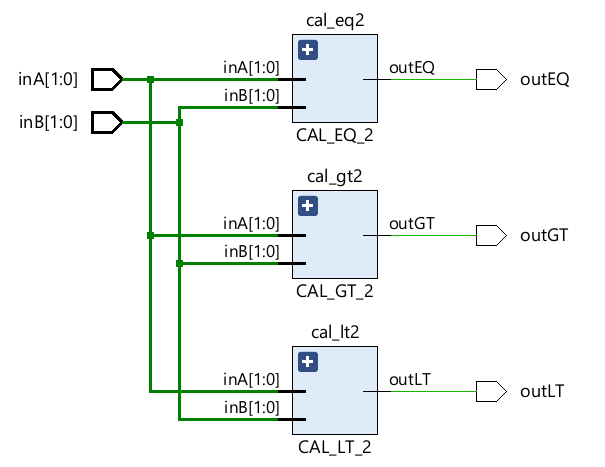
총 와이어 : 63개

총 논리 게이트 : 51개

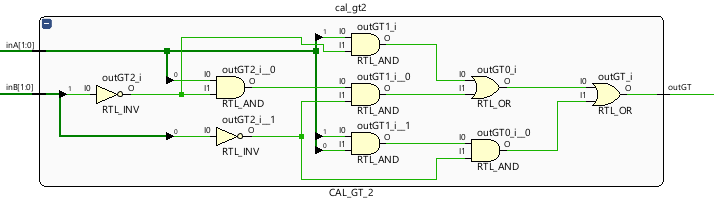
1. 단순화 이후
2. K-map으로 단순화한 세 가지 출력을 구현한다.

실험준비에서 단순화한 식을 assign keyword와 “~”,“&”,“|” operator를 이용하여 나타내었다.

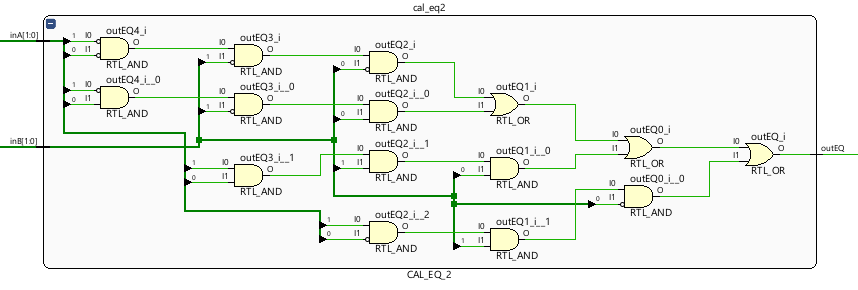
1. Schematic 기능으로 회로가 잘 구현되었는지 확인한다.



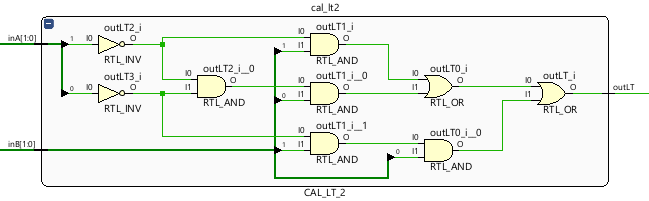
1. A>B(GT)



1. A=B(EQ)



1. A<B(LT)



1. RTL ANALYSIS > Netlist를 참조하여 와이어와 논리 게이트 개수를 확인한다.
2. A>B(GT)

와이어 : 13개

논리 게이트 : 9개

1. A=B(EQ)

와이어 : 19개

논리 게이트 : 15개

1. A<B(LT)

와이어 : 13개

논리 게이트 : 9개

총 와이어 : 45개

총 논리 게이트 : 33개

1. 실험 1과 와이어와 논리 게이트 개수를 비교한다.

총 와이어 개수 : 63->45

총 논리 게이트 개수 : 51->33

와이어 개수와 논리 게이트 개수 모두 감소하였다

1. **논의 및 결론**

본 실험에서 2-Bit Magnitude Comparator를 불 대수식으로 나타내고, K-map로 단순화 하는 작업을 하였다. 단순화 전 후로 회로도를 그리고 와이어와 논리 게이트의 개수를 비교하였다. 결과는 예상대로 단순화를 하였을 때 와이어와 논리 게이트 개수가 감소함을 확인하였다. 이를 통해 불 대수식의 단순화는 회로의 복잡도 감소로 나타남을 알게 되었다.