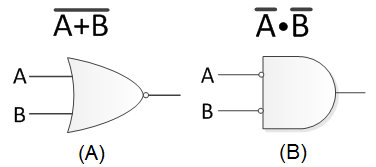
5주차 결과보고서

|  |
| --- |
| - De-Morgan 제 1, 2법칙  - (!A | !B) & !C = !((A & B) | C)  - (!A & !B) | !C = !((A | B) & C)  - 1Bit 비교기  - 추가 사항(결과 검토, 논의 사항, 이론 등) |

20141196 김성희

**1. De-Morgan 제 1, 2법칙**

**1-1. 제 1법칙**

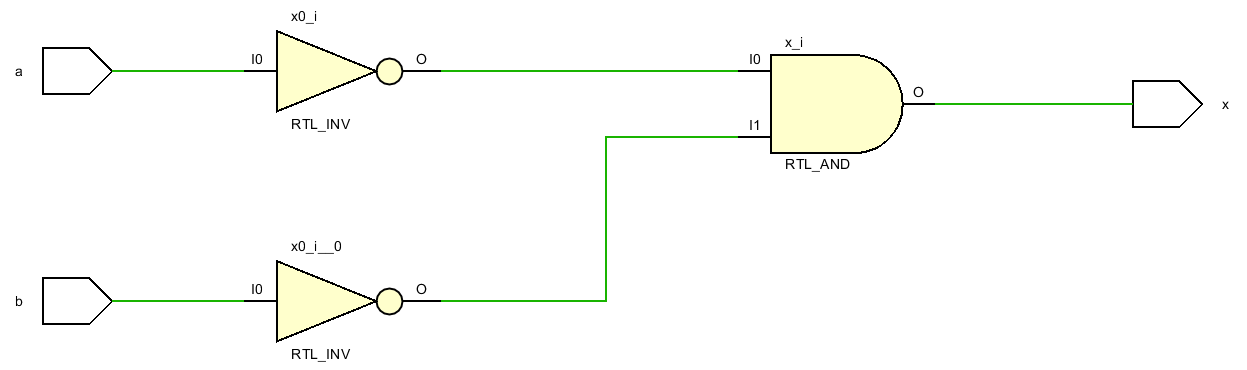


**(가). (A)와 (B)의 Schematic 비교 (with Verilog)**

(A)



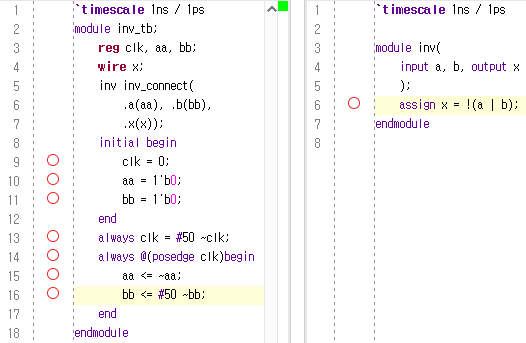
(B)



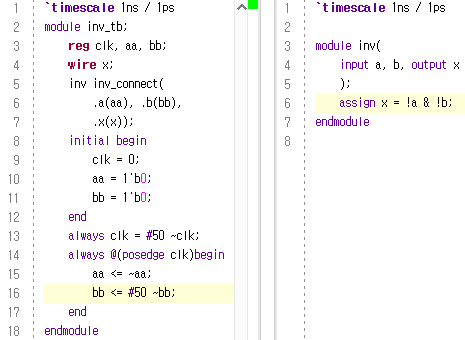
회로도에서 NOR와 input값에 inverter를 취한 AND가 같다는 것을 보여준다. inverter두개와 AND하나, 총 3개의 게이트를 NOR하나로 줄일 수 있다.

**(나). (A)와 (B)의 Verilog 코딩**

(A)

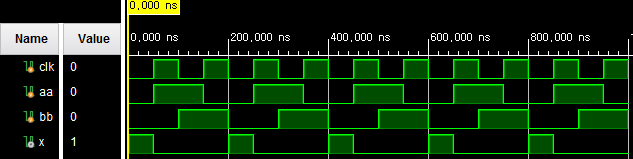


(B)



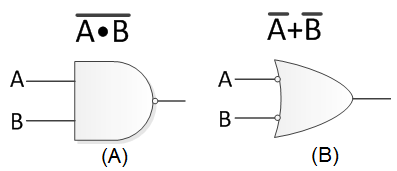
**(다). (A)와 (B)의 Simulation 출력 결과**

(A)와 (B) 동일



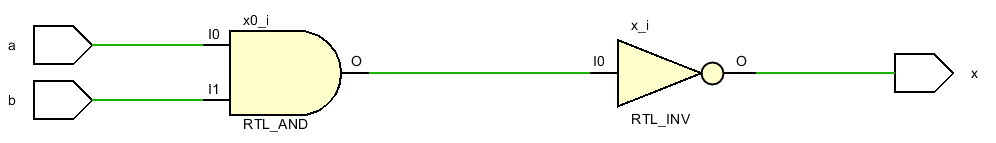
Input 값인 aa와 bb를 똑같이 했을 때 simulation이 동일하게 나온다. A와 B가 같다는 De-Morgan 제1법칙이 성립한다는 것을 보여준다.

**1-2. 제 2법칙**

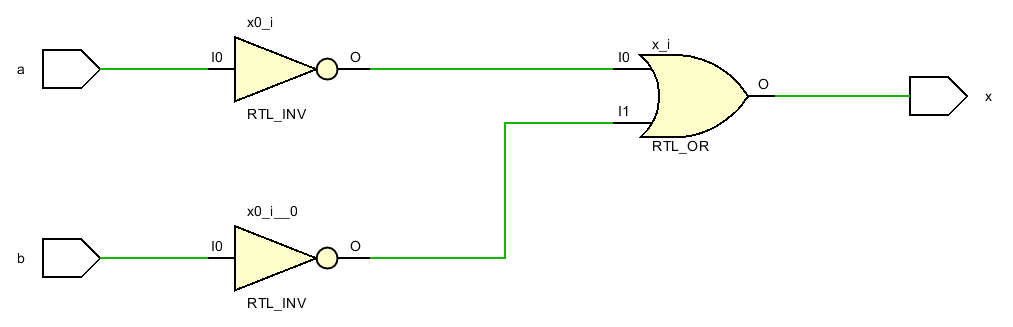


**(가). (A)와 (B)의 Schematic 비교 (with Verilog)**

(A)



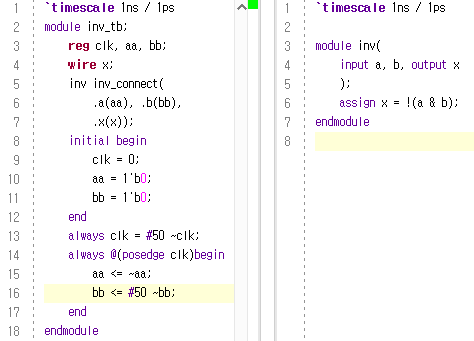
(B)



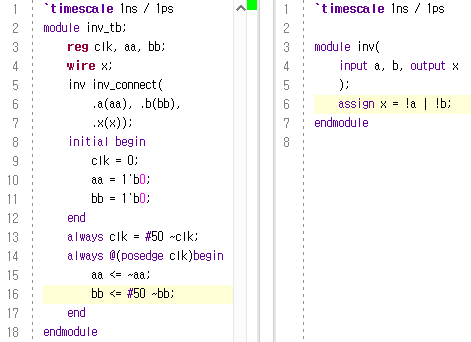
회로도에서 NAND와 input값에 inverter를 취한 OR가 같다는 것을 보여준다. inverter두개와 OR하나, 총 3개의 게이트를 NAND하나로 줄일 수 있다.

**(나). (A)와 (B)의 Verilog 코딩**

(A)

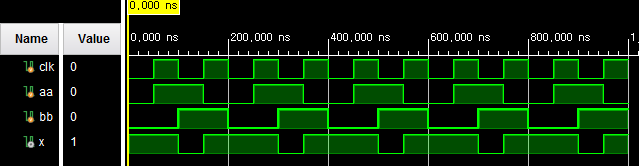


(B)



**(다). (A)와 (B)의 Simulation 출력 결과**

(A)와 (B) 동일



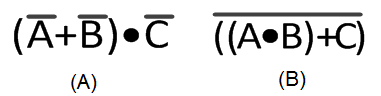
Input 값인 aa와 bb를 똑같이 했을 때 simulation이 동일하게 나온다. A와 B가 같다는 De-Morgan 제2법칙이 성립한다는 것을 보여준다.

**1-3. (A)와 (B)의 Truth Table**

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **1** | **2** | **3** | **4** | **5** | **6** | **7** | **8** | **9** | **10** |
| **A** | **B** | **A | B** | **A&B** | **!A** | **!B** | **!(A | B)** | **!A & !B** | **!(A&B)** | **!A | !B** |
| 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 1 |
| 0 | 1 | 1 | 0 | 1 | 0 | 0 | 0 | 1 | 1 |
| 1 | 0 | 1 | 0 | 0 | 1 | 0 | 0 | 1 | 1 |
| 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 |

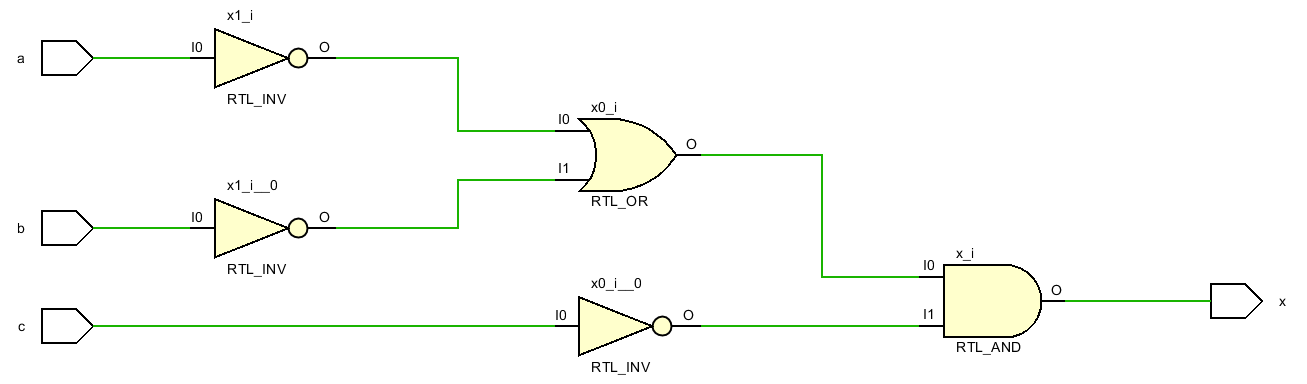
**7번과 8번의 진리값 비교를 통해 De-Morgan 제 1법칙이 참임을, 9번과 10번의 진ㄹ값 비교를 통해 제 2법칙이 참임을 알 수 있다.**

**2. (!A | !B) & !C = !((A & B) | C) (De-Morgan 법칙의 응용)**

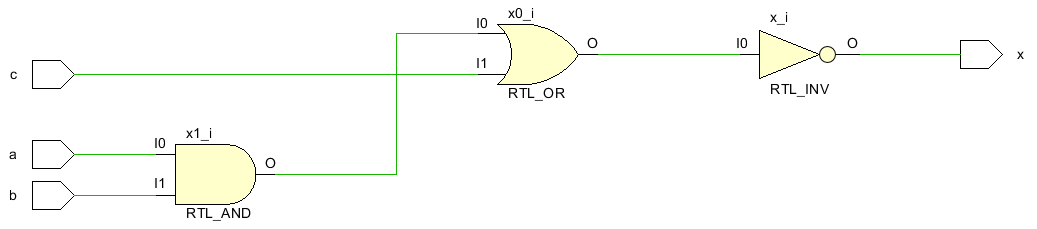


**(가). (A)와 (B)의 Schematic 비교 (with Verilog)**

(A)



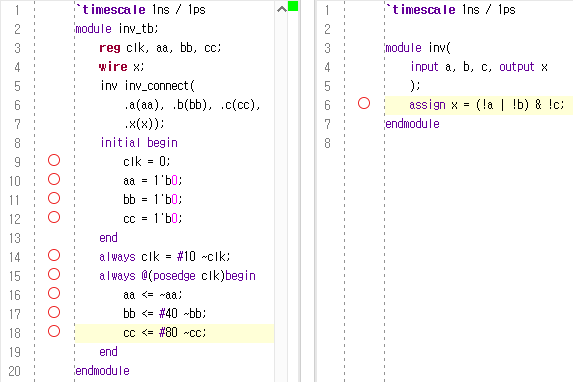
(B)



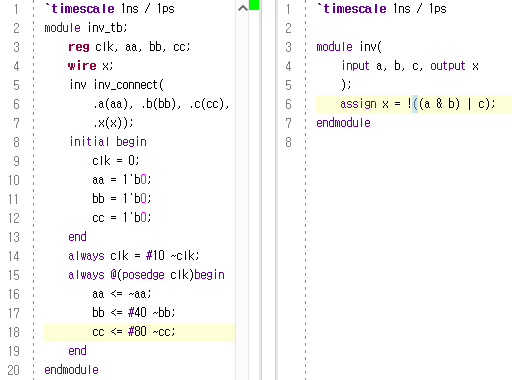
(A)의 a, b의 inverter와 OR를 합쳐서 NAND를 만들었다. 그리고 NAND의 output의 inverter와 c의 inverter 그리고 AND를 합쳐서 NOR를 만들었다. 최종적으로 (B)에서 AND와 NOR를 이용한 셈이다. 즉 De-Morgan 제 1법칙을 회로에 적용한 뒤 이어서 제 2법칙을 회로에 적용하였다. 결과적으로 (A)에 비해 (B)가 더 효율이 좋다.

**(나). (A)와 (B)의 Verilog 코드**

(A)

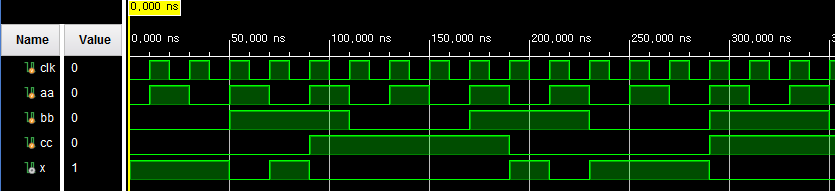


(B)



**(다). (A)와 (B)의 Simulation 출력 결과**

(A)와 (B) 동일



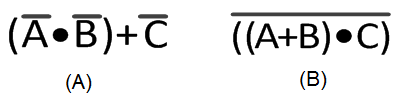
Input 값인 aa, bb, cc를 똑같이 했을 때 simulation이 동일하게 나온다. A와 B가 동치임을 보여준다.

**(라). (A)와 (B)의 Truth Table**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **1** | **2** | **3** | **4** | **5** |
| **A** | **B** | **C** | **(!A | !B) & !C** | **!((A & B) | C)** |
| 0 | 0 | 0 | 1 | 1 |
| 0 | 0 | 1 | 0 | 0 |
| 0 | 1 | 0 | 1 | 1 |
| 0 | 1 | 1 | 0 | 0 |
| 1 | 0 | 0 | 1 | 1 |
| 1 | 0 | 1 | 0 | 0 |
| 1 | 1 | 0 | 0 | 0 |
| 1 | 1 | 1 | 0 | 0 |

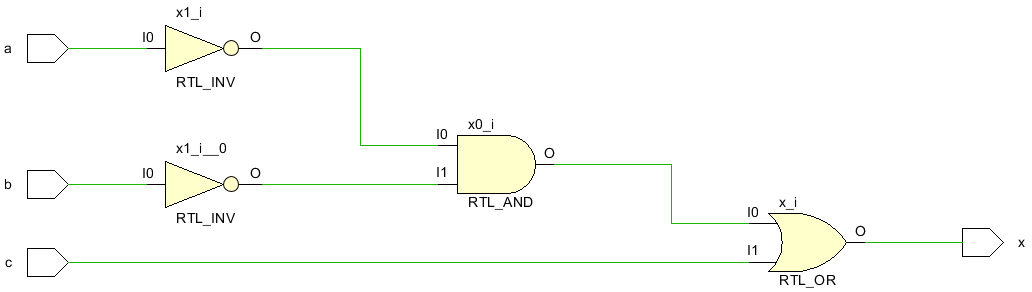
4번과 5번의 진리값이 같다는 것을 통해 (A)와 (B)가 동치임을 알 수 있다.

**3. (!A & !B) | !C = !((A | B) & C)**

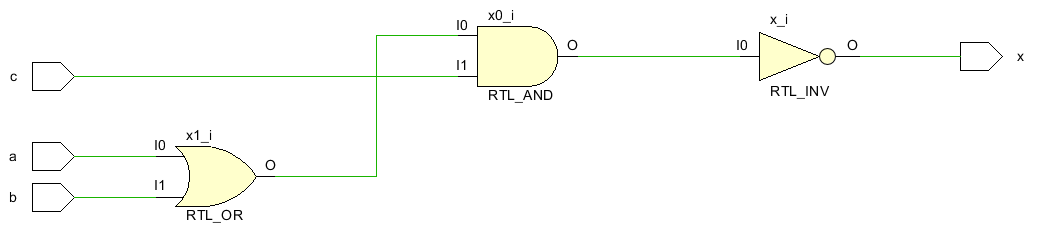


**(가). (A)와 (B)의 Schematic 비교 (with Verilog)**

(A)



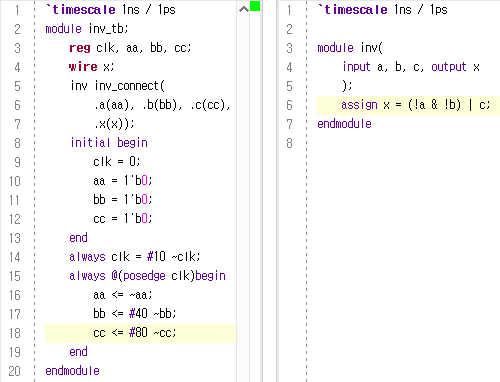
(B)



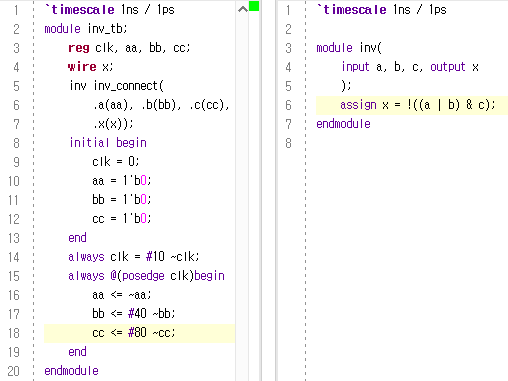
(A)의 a, b의 inverter와 AND를 합쳐서 NOR를 만들었다. 그리고 NOR의 output의 inverter와 c의 inverter 그리고 OR를 합쳐서 NAND를 만들었다. 최종적으로 (B)에서 OR와 NAND를 이용한 셈이다. 즉 De-Morgan 제 2법칙을 회로에 적용한 뒤 이어서 제 1법칙을 회로에 적용하였다. 결과적으로 (A)에 비해 (B)가 더 효율이 좋다.

**(나). (A)와 (B)의 Verilog 코드**

(A)

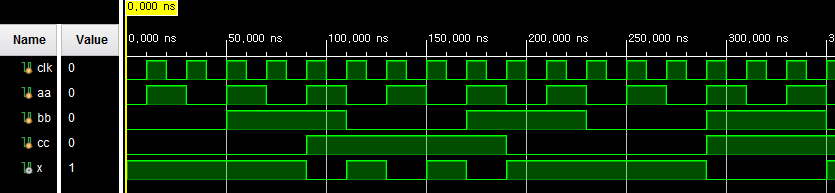


(B)



**(다). (A)와 (B)의 Simulation 출력 결과**

(A)와 (B) 동일



Input 값인 aa, bb, cc를 똑같이 했을 때 simulation이 동일하게 나온다. A와 B가 동치임을 보여준다.

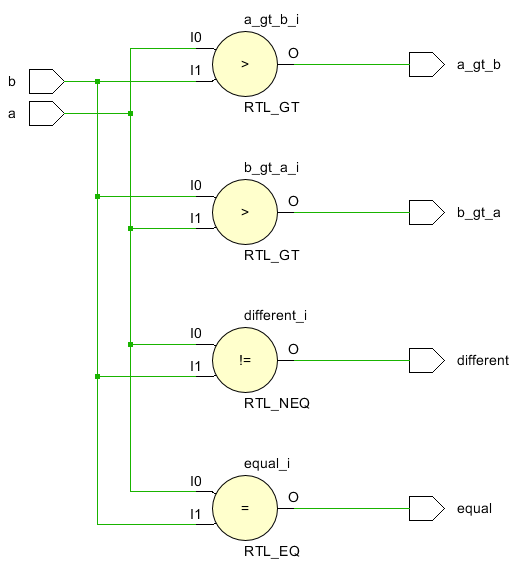
**(라). (A)와 (B)의 Truth Table**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **1** | **2** | **3** | **4** | **5** |
| **A** | **B** | **C** | **(!A & !B) | !C** | **!((A | B) & C)** |
| 0 | 0 | 0 | 1 | 1 |
| 0 | 0 | 1 | 1 | 1 |
| 0 | 1 | 0 | 1 | 1 |
| 0 | 1 | 1 | 0 | 0 |
| 1 | 0 | 0 | 1 | 1 |
| 1 | 0 | 1 | 0 | 0 |
| 1 | 1 | 0 | 1 | 1 |
| 1 | 1 | 1 | 0 | 0 |

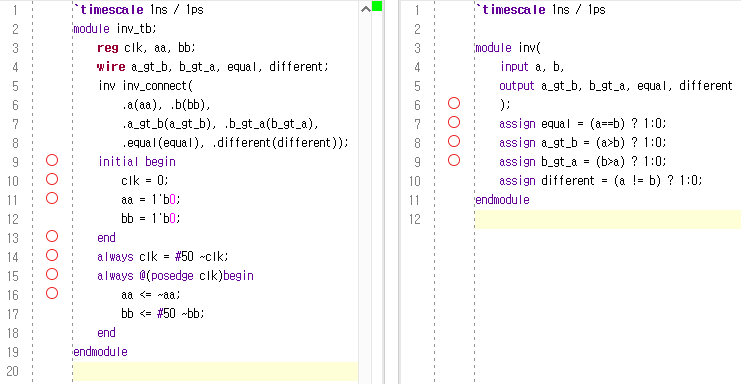
4번과 5번의 진리값이 같다는 것을 통해 (A)와 (B)가 동치임을 알 수 있다.

**4. 1Bit 비교기**

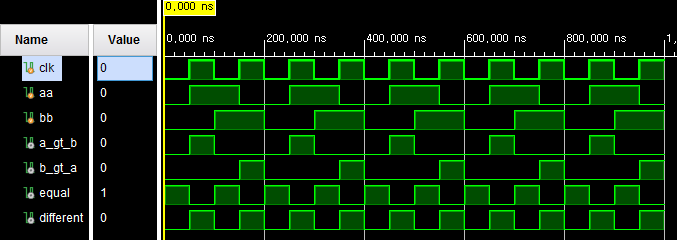
**(가). 1Bit 비교기 schematic**



**(나). 1Bit 비교기 Verilog 코드**



**(다). 1Bit 비교기 Simulation 결과**



|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **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 |

1. bb > aa 일 때, b\_gt\_a=1.

Truth Table

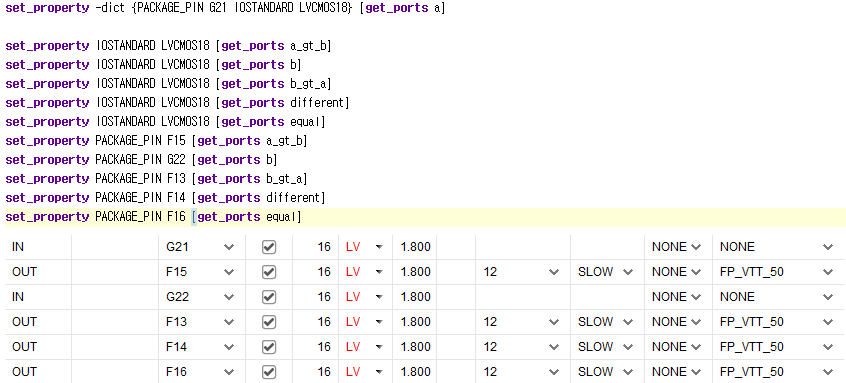
2. aa > bb 일 때, a\_gt\_b=1.

3. aa == bb 일 때, equal=1.

4. aa != bb 일 때, different=1.

**(라). 1Bit 비교기 FPGA 동작 결과**

**I/O port 할당 코드**

****

F15, F13, F14, F16은 순서대로 LD1~LD4(LED)를 의미하며, G21, G22는 switch MID, RIGHT을 의미한다. 아래 표를 보자.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| MID | RIGHT | LD1  (MID>RIGHT) | LD2  (MID<RIGHT) | LD3  MID != RIGHT | LD4  MID==RIGHT |
| NO PUSH | NO PUSH | OFF | OFF | OFF | ON |
| NO PUSH | PUSH | OFF | ON | ON | OFF |
| PUSH | NO PUSH | ON | OFF | ON | OFF |
| PUSH | PUSH | OFF | OFF | OFF | ON |

위 표에 따라 FPGA가 동작한다. Switch PUSH는 1을 의미하고 NO PUSH는 0을 의미한다. LED ON은 1을 의미하고 OFF는 0을 의미한다.