

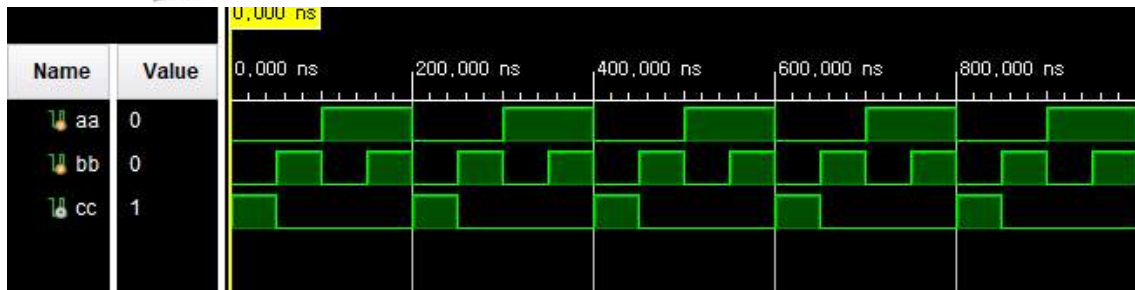
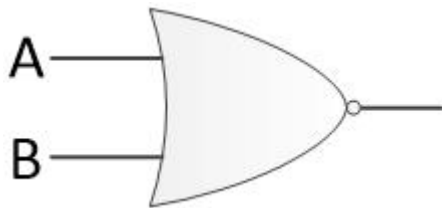
5주차 결과 보고서

20191621 이민영

1. De-Morgan의 제 1,2 법칙의 simulation 결과 및 과정에 대해서 설명하시오. (NAND, NOR과 비교 포함)

드 모르간(De-Morgan)의 제 1 법칙

(A) $\overline{A+B}$

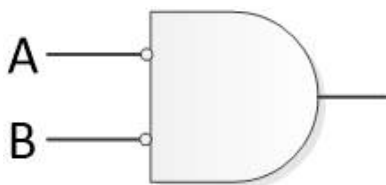


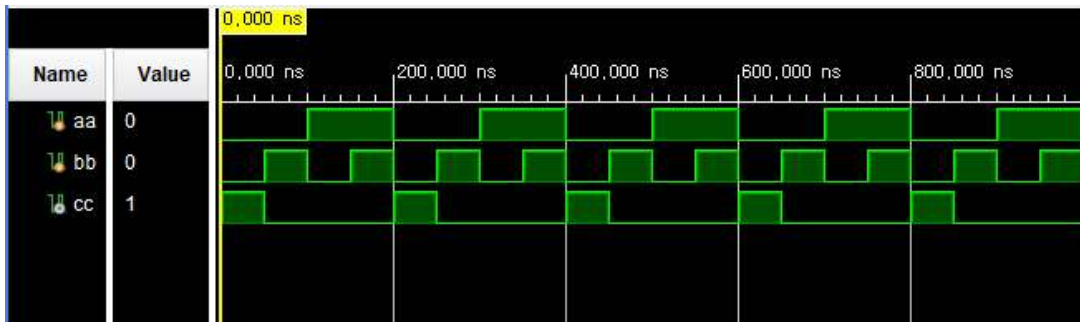
<제 1 법칙의 (A) Simulation>



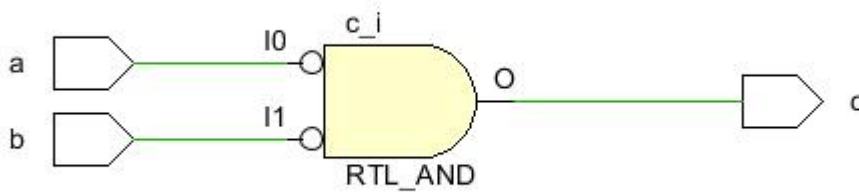
<제 1 법칙의 (A) Schematic>

(B) $\overline{A} \cdot \overline{B}$



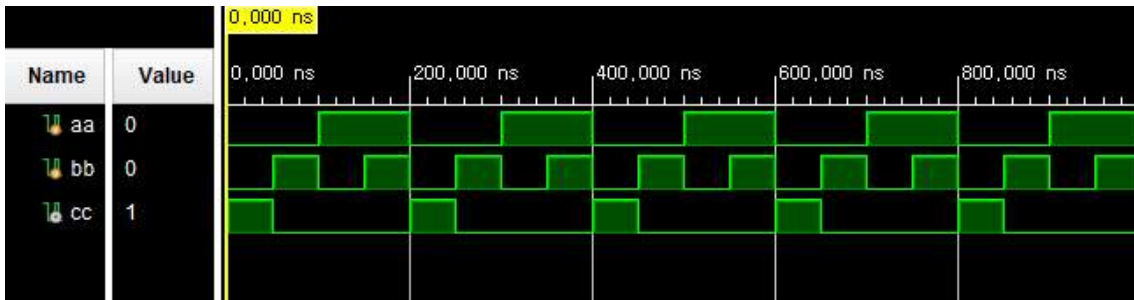


<제 1 법칙의 (B) Simulation>



<제 1 법칙의 (B) Schematic>

Simulation 결과를 보면, 각각의 입력에 맞추어 나온 값을 통해 $\overline{A+B} = \overline{A} \cdot \overline{B}$ 가 성립한다는 것을 알 수 있다. 즉, 드모르간의 제 1법칙이 성립한다.



또한 NOR를 실행한 결과와도 동일한 것을 알 수 있다.

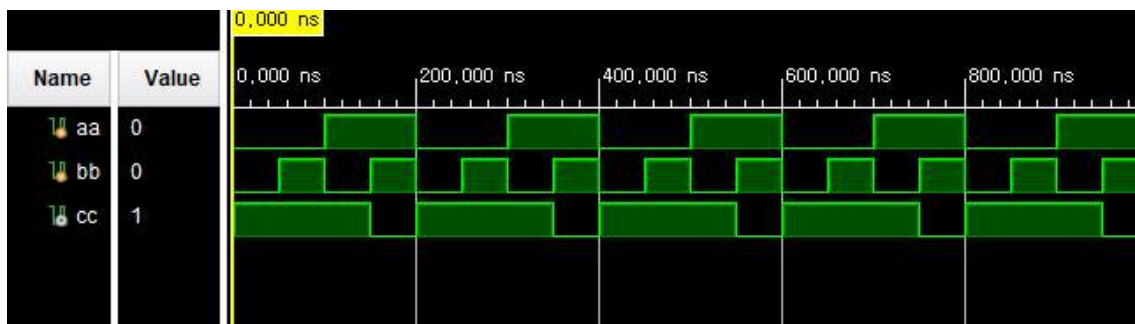
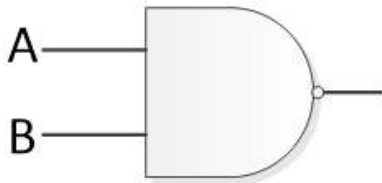
즉, $\text{NOR} = \overline{A+B} = \overline{A} \cdot \overline{B}$ 이다.

위의 결과를 통해서 A와 B의 합의 부정이 A의 부정과 B의 부정의 곱과 같다는 것을 알 수 있다.

Simulation 결과를 자세히 살펴보면 A가 0이고 B가 0일 때 결과 값은 $(0+0)' = 1$, $0' \cdot 0' = 1$ 로 결과 값이 1로 동일하게 나온다는 것을 알 수 있다. 이와 마찬가지로 A와 B의 모든 값이 $(A+B)'$ 와 $A' \cdot B'$ 가 같다는 것을 알 수 있다.

드 모르간(De-Morgan)의 제 2 법칙

(A) $\overline{A \cdot B}$

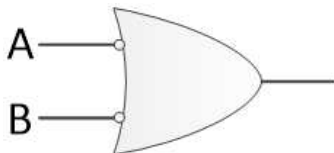


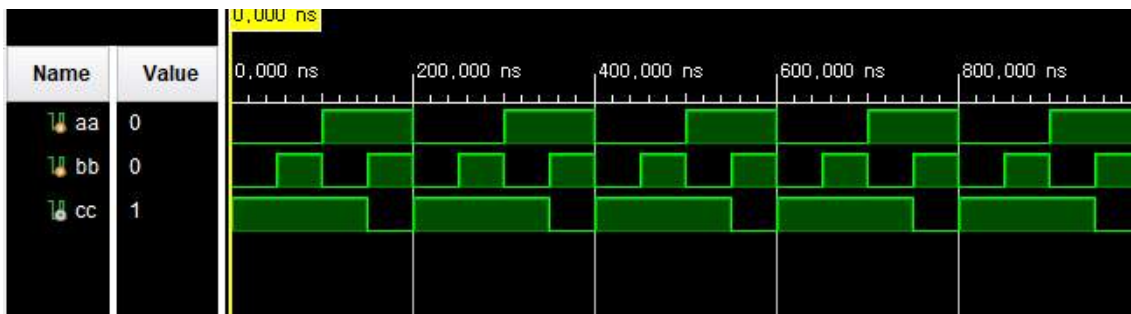
<제 2 법칙의 (A) Simulation>



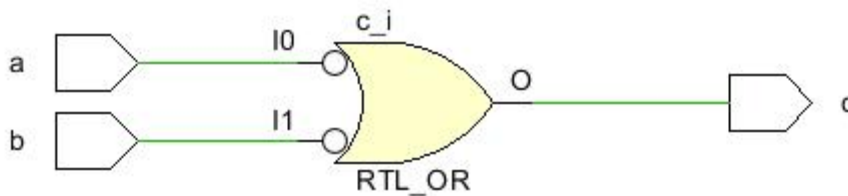
<제 2 법칙의 (A) Schematic>

(B) $\overline{A + B}$



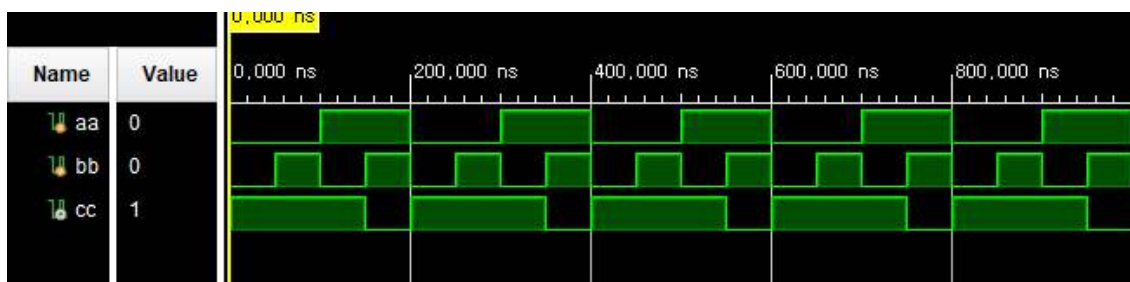


<제 2 법칙의 (B) Simulation>



<제 2 법칙의 (B) Schematic>

Simulation결과를 비교해보면, $\overline{A \cdot B} = \overline{A} + \overline{B}$ 가 성립함을 알 수 있다. 즉, 드모르간의 제 2 법칙이 성립한다.



또한 NAND의 값과도 동일한 것을 Simulation결과를 통해서 알 수 있다.

즉, $\overline{A \cdot B} = \overline{A} + \overline{B} = \text{NAND}$ 이다.

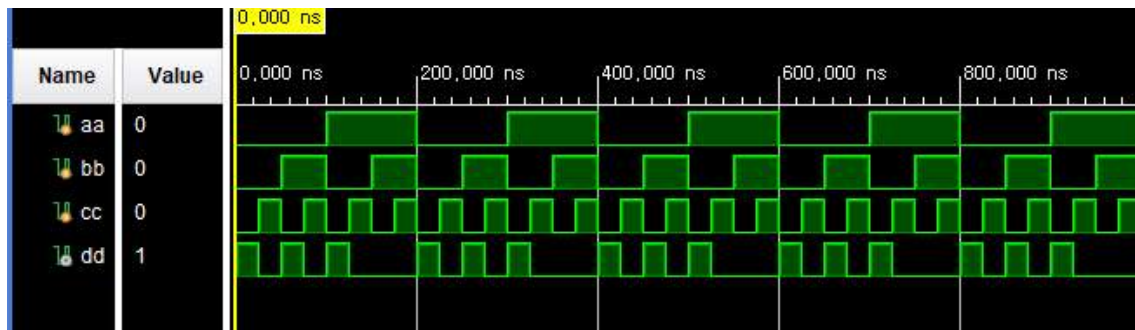
이를 통해서 곱에 보수를 취한 것이 보수끼리의 합과 같으며, 합에 보수를 취한 것이 보수끼리의 곱과 같다는 것을 알 수 있다.

Simulation 결과를 자세히 살펴보면 A가 0이고 B가 0일 때 결과 값은 $(0 \cdot 0)' = 1$, $0' + 0' = 1$ 로 결과 값이 1로 동일하게 나온다는 것을 알 수 있다. 이와 마찬가지로 A와 B의 모든 값이 $(A \cdot B)'$ 와 $A' + B'$ 가 같다는 것을 알 수 있다.

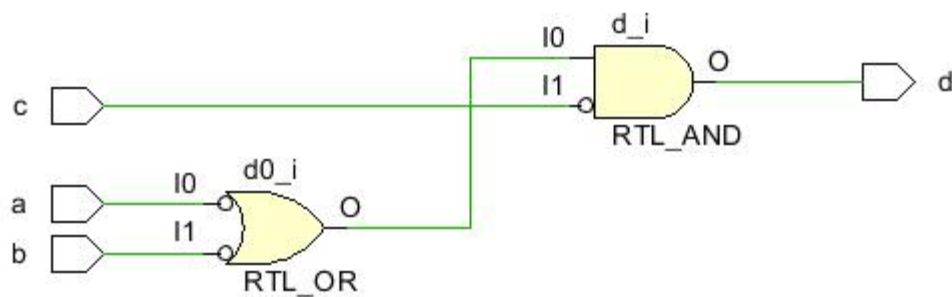
2. $(A'+B')*C' = ((A*B)+C)'$ 의 simulation 결과 및 과정에 대해서 설명하시오. [+ 및 * 위치 바꾼 모양도 수행]

1) $(A'+B')*C' = ((A*B)+C)'$

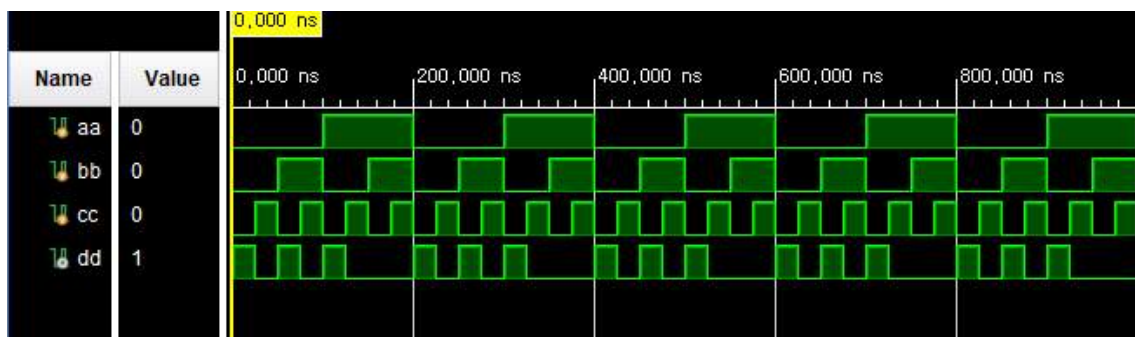
(A) $(A'+B')*C'$



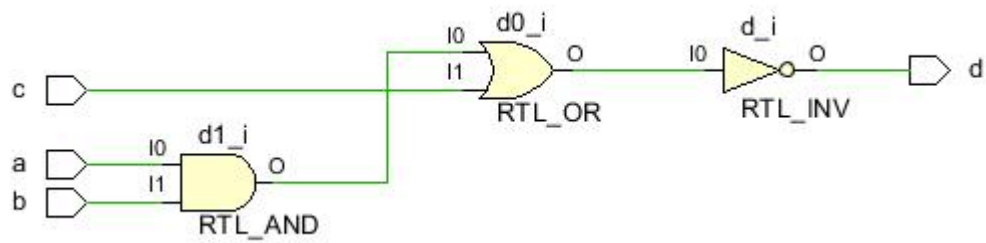
< $(A'+B')*C'$ 의 Simulation 결과>



(B) $((A*B)+C)'$



< $((A*B)+C)'$ 의 Simulation 결과>

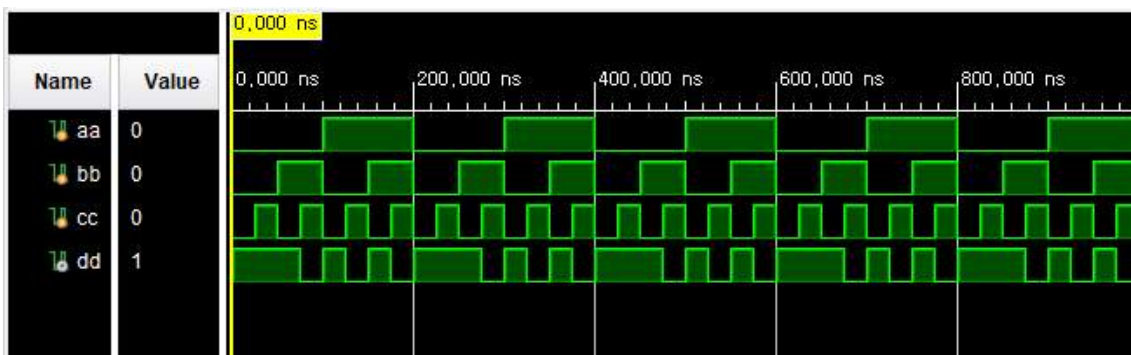


Simulation 결과를 비교해보면, $(A'+B')*C' = ((A*B)+C)'$ 가 성립함을 알 수 있다.

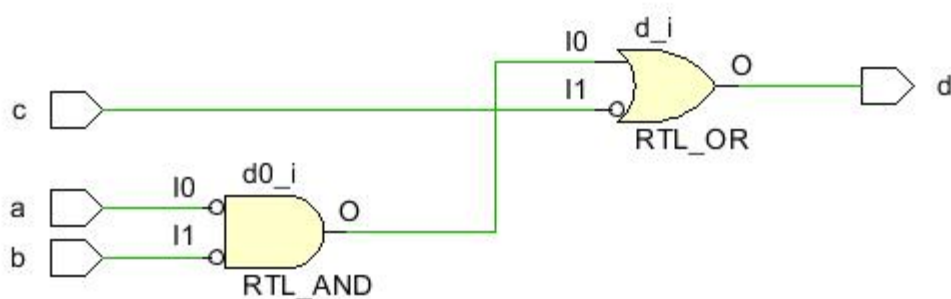
Simulation 결과를 자세히 살펴보면 A가 0이고 B가 0이고 C가 0일 때 결과 값은 $(0'+0')*0' = 1$, $((0*0)+0)' = 1$ 로 결과 값이 1로 동일하게 나온다는 것을 알 수 있다. 이와 마찬가지로 A와 B의 모든 값이 $(A'+B')*C' = ((A*B)+C)'$ 가 같다는 것을 알 수 있다.

$$2) (A'*B')+C' = ((A+B)*C)'$$

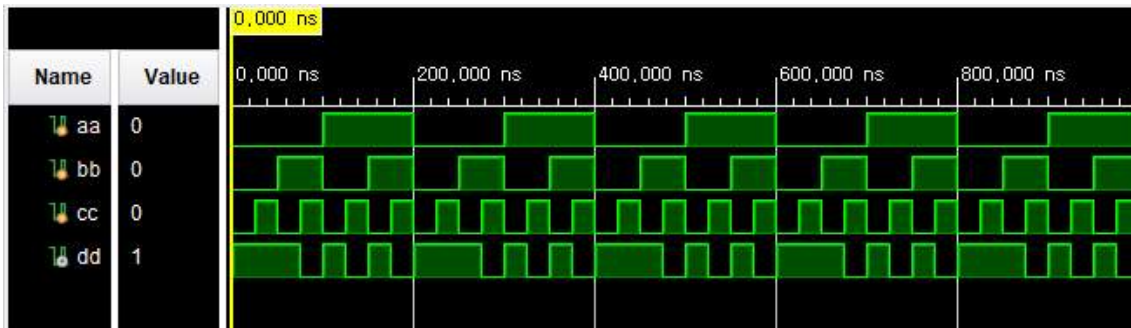
$$(A) (A'*B')+C'$$



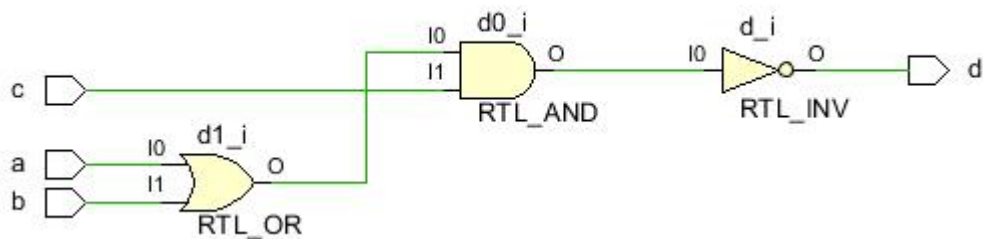
< $(A'*B')+C'$ 의 Simulation결과>



(B) $((A+B)*C)'$



< $((A+B)*C)'$ 의 Simulation 결과>



Simulation 결과를 비교해보면, $(A'*B')+C' = ((A+B)*C)'$ 가 성립함을 알 수 있다.

Simulation 결과를 자세히 살펴보면 A가 0이고 B가 0이고 C가 0일 때 결과 값은 $(0'*0')+0' = 1$, $((0+0)*0)' = 1$ 로 결과 값이 1로 동일하게 나온다는 것을 알 수 있다. 이와 마찬가지로 A와 B의 모든 값이 $(A'*B')+C' = ((A+B)*C)'$ 가 같다는 것을 알 수 있다.

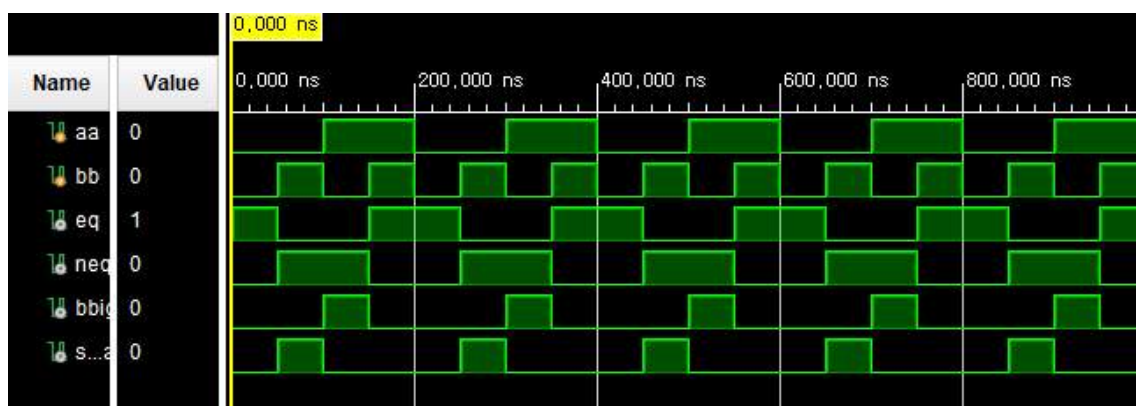
3. 1Bit 비교기의 simulation 결과 및 과정에 대해서 설명하시오.

(2 input, 4 output) [진리표 작성]

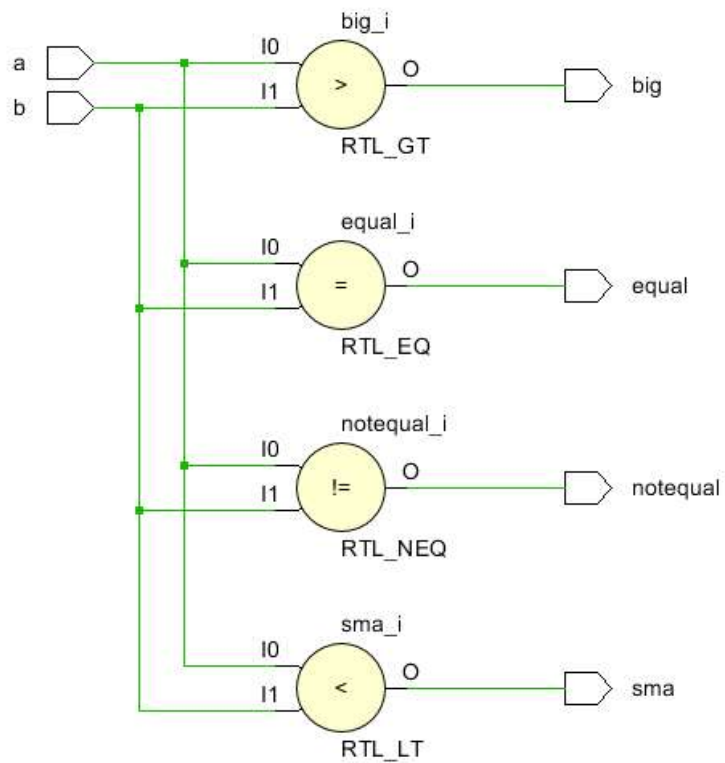
input A	input B	A=B	A≠B	A>B	A<B
0	0	1	0	0	0
0	1	0	1	0	1
1	0	0	1	1	0
1	1	1	0	0	0

One_Bit_Comp	One_Bit_Comp_tb
<pre>`timescale 1ns / 1ps module One_Bit_Comp(input a,</pre>	<pre>`timescale 1ns / 1ps module One_Bit_Comp_tb; reg aa; reg bb;</pre>

<pre> input b, output equal, output notequal, output big, output sma); assign equal = (a==b); assign notequal = (a!=b); assign big = (a>b); assign sma = (a<b); endmodule </pre>	<pre> wire eq; wire neq; wire bbig; wire ssma; One_Bit_Comp u_One_Bit_Comp(.a(aa), .b(bb), .equal(eq), .notequal(neq), .big(bbig), .sma(ssma)); initial aa = 1'b0; initial bb = 1'b0; always aa = #100 ~aa; always bb = #50 ~bb; initial begin #1000 \$finish; end endmodule </pre>
---	--



<Simulation 결과>



<Schematic>

Simulation의 결과, input에 따른 output의 결과가 진리표와 동일하게 나오는 것을 확인할 수 있다.

a가 0, b가 0으로 입력되었을 때 a와 b는 같은 값이기 때문에 $a==b$ 를 나타내는 eq는 1을 가르키며, 동시에 a와 b는 같은 값이므로 $a!=b$ 를 나타내는 notequal은 0을 나타낸다. 또한 $a>b$, $a<b$ 모두 거짓이므로 0을 나타낸다.

이와 같은 방식으로 a와 b의 값을 비교한 결과를 얻어낼 수 있었다.