5주차 결과보고서

전공: 컴퓨터공학과 학년: 2학년 학번: 20211533 이름: 남정연

**1.**

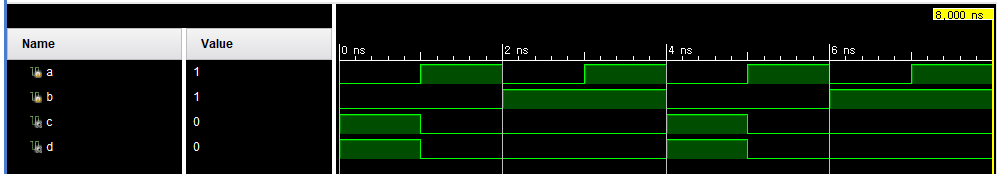
1. De Morgan의 정리 및 Boolean 함수의 동작을 이해하고 및 확인한다.

2. Verilog를 사용하여 De Morgan의 정리 및 Boolean 함수의 동작을 구현한다.

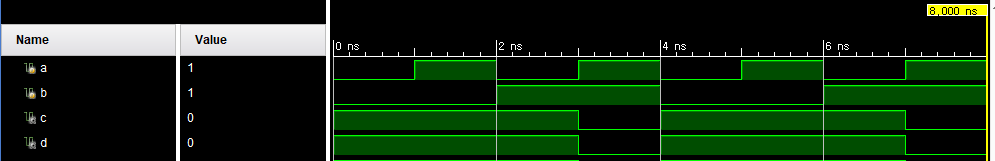
**2.**

1. 드모르간의 제 1법칙 : (A+B)’=A’B’

Input a, b에 대하여 assign c = ~(a|b), assign d = (~a)&(~b)의 코드를 작성하였을 때 아래와 같은 시뮬레이션 결과가 나타난다. 즉 두 식의 진리표가 같음을 알 수 있으며, 두 논리 식이 같은 논리 값을 가짐을 알 수 있다. (A+B)’는 NOR 게이트의 정의이므로 드모르간의 제 1법칙은 NOR게이트를 A’B’로도 나타낼 수 있음을 알려준다.

****

2. 드모르간의 제 2법칙 : (AB)’=A’+B’

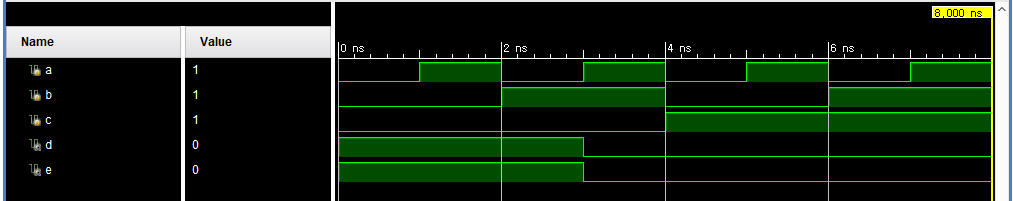
****

Input a, b에 대하여 assign c = ~(a&b), assign d = (~a)|(~b)의 코드를 작성하였을 때 아래와 같은 시뮬레이션 결과가 나타난다. 즉 두 식의 진리표가 같음을 알 수 있으며, 두 논리 식이 같은 논리 값을 가짐을 알 수 있다. (AB)’는 NAND 게이트의 정의이므로 드모르간의 제 2법칙은 NAND게이트를 A’+B’로도 나타낼 수 있음을 알려준다.

**3.**

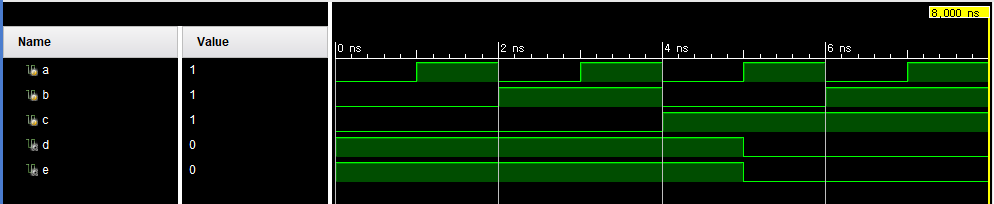
1. (A’+B’)\*C’ = ((AB)+C)’

Input a, b, c에 대하여 assign d = ((~a)|(~b))&(~c), assign e = ~((a&b)|c)의 코드를 작성하였을 때 아래와 같은 시뮬레이션 결과가 나타난다. 즉 두 식의 진리표가 같음을 알 수 있으며, 두 논리 식이 같은 논리 값을 가짐을 알 수 있다.



|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| A | B | C | ((A’+B’)\*C’)’ | ((AB)+C)’ |
| 0 | 0 | 0 | 1 | 1 |
| 1 | 0 | 0 | 1 | 1 |
| 0 | 1 | 0 | 1 | 1 |
| 1 | 1 | 0 | 0 | 0 |
| 0 | 0 | 1 | 0 | 0 |
| 1 | 0 | 1 | 0 | 0 |
| 0 | 1 | 1 | 0 | 0 |
| 1 | 1 | 1 | 0 | 0 |

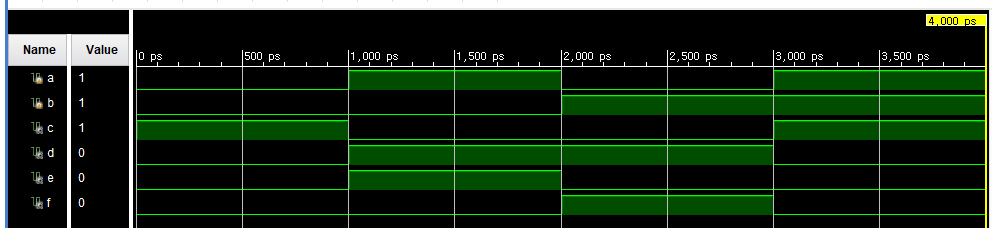
2. (A’B’)+C’ = ((A+B)\*C)’



Input a, b, c에 대하여 assign d = ((~a)&(~b))|(~c), assign e = ~((a|b)&c)의 코드를 작성하였을 때 아래와 같은 시뮬레이션 결과가 나타난다. 즉 두 식의 진리표가 같음을 알 수 있으며, 두 논리 식이 같은 논리 값을 가짐을 알 수 있다.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| A | B | C | (A’B’)+C’ | ((A+B)\*C)’ |
| 0 | 0 | 0 | 1 | 1 |
| 1 | 0 | 0 | 1 | 1 |
| 0 | 1 | 0 | 1 | 1 |
| 1 | 1 | 0 | 1 | 1 |
| 0 | 0 | 1 | 1 | 1 |
| 1 | 0 | 1 | 0 | 0 |
| 0 | 1 | 1 | 0 | 0 |
| 1 | 1 | 1 | 0 | 0 |

**4.**

Input A, B에 대해 Output C = (A^B)’, D = A^B, E = AB’, F= A’B이다.

C는 A와 B의 XOR 연산의 역으로, 두 개의 비트가 같은 값을 가지는지 확인할 수 있고, D는 그 역이다. E는 A가 B보다 클 경우 1, F는 B가 A보다 클 경우 1을 나타내며 아래 결과에서 볼 수 있듯이 정확한 결과가 나타남을 알 수 있다.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| A | B | C | D | E | F |
| 0 | 0 | 1 | 0 | 0 | 0 |
| 1 | 0 | 0 | 1 | 1 | 0 |
| 0 | 1 | 0 | 1 | 0 | 1 |
| 1 | 1 | 1 | 0 | 0 | 0 |

**5.**

드 모르간의 법칙을 실제 verilog를 통하여 실행시켜보며 법칙이 일치하는지를 확인할 수 있었고, 보다 복잡한 식에서도 드 모르간의 법칙이 성립함을 알 수 있었다. 또한 1 Bit 비교기를 직접 구현하여 봄으로써 실제 비트 비교에 있어서 어떠한 방법을 활용하는지, 그 방식의 정확성이 어떤지 확인할 수 있었다.

**6.**

2 bit 비교기의 경우 A1A2와 B1B2 값을 축으로 하는 카르노 맵을 그려 논리식을 단순화 한 뒤 비교기를 작성할 수 있다. 이때 F1= (A1 XNOR B1)(A2 XNOR B2), F2=(A1 XOR B1)+(A2 XOR B2), F3 = A1B1’+A2B1’B2’+A1A2B2’, F4=A1’B1+A1’A2’B2+A2’B1B2로 나타난다.