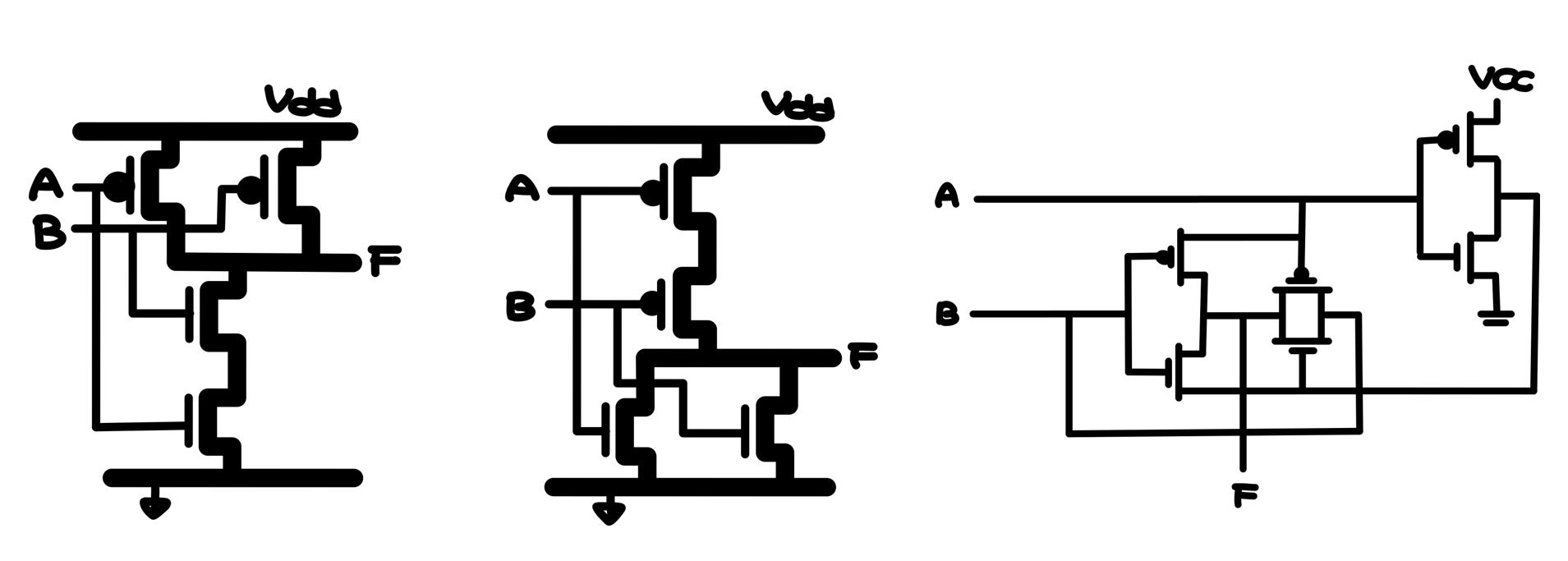
4주차 예비보고서

전공: 물리학과 학년: 3학년 학번: 20191286 이름: 김나현

1. 논리게이트 NAND, NOR, XOR의 구조를 Transistor-level로 그리시오.



<Figure 1> transistor-level의 NAND, NOR, XOR 게이트의 구조

1. NAND, NOR, XOR logic의 특성에 대해 조사하시오.

NAND 게이트는 모든 입력이 참일 때만 거짓을 출력하는 논리 게이트이고, AND 게이트의 출력이NOT 게이트를 거친 것이라고 생각하면 됩니다. NOR 게이트는 모든 입력이 거짓일 때만 참을 출력하는 논리 게이트로, OR 게이트의 출력이 NOT 게이트를 거친 것이라고 생각하면 됩니다. 이렇게 NAND 게이트와 NOR 게이트는 이름에서부터 알 수 있듯이, 각각 AND 게이트와 OR 게이트의 동작과 정확히 반대로 동작한다는 것을 알 수 있습니다. 따라서, NAND 게이트와 NOR 게이트의 진리표를 작성하면 다음과 같습니다.

<Table 1> 두 개의 입력을 갖는 NAND 게이트의 진리표

|  |  |  |
| --- | --- | --- |
| 입력 | | 출력 |
| A | B | Y |
| 0 | 0 | 1 |
| 0 | 1 | 1 |
| 1 | 0 | 1 |
| 1 | 1 | 0 |

<Table 1>을 보면 모든 입력이 1일 때만 1을 출력하는 AND 게이트와 반대로, NAND 게이트는 모든 입력이 1일 때만을 제외하고 1을 출력합니다.

<Table 2> 두 개의 입력을 갖는 NOR 게이트의 진리표

|  |  |  |
| --- | --- | --- |
| 입력 | | 출력 |
| A | B | Y |
| 0 | 0 | 1 |
| 0 | 1 | 0 |
| 1 | 0 | 0 |
| 1 | 1 | 0 |

반면, <Table 2>을 보면 입력 중 하나 이상이 1이면 1을 출력하는 OR 게이트와 반대로, NOR 게이트는 입력 중 하나라도 1이면 0을 출력합니다. OR 게이트와 NAND 게이트가 서로 쌍대 관계를 이루듯 NOR 게이트와 NAND 게이트 역시 쌍대 관계라는 것을 알 수 있습니다.텍스트, 손목시계, 게이지이(가) 표시된 사진

자동 생성된 설명

<Figure 2>OR, NOR, AND, NAND 게이트의 기호

NAND 게이트와 NOR 게이트를 기호로 나타내면 위의 <Figure 2>와 같고, 버퍼 기호에 동그라미를 붙여서 표현했던 NOT 게이트와 동일하게 NAND 게이트는 AND 게이트의 앞에 동그라미를, NOR 게이트는 OR 게이트의 앞에 동그라미를 붙였다는 것을 알 수 있습니다.

NOR 연산은 Y=A+B, 즉 OR 게이트 출력에 NOT 게이트를 붙인 것이므로 Y=(A+B)’라고 표현할 수 있고, 이는 드 모르간의 법칙에 의해 Y=A’·B’가 됩니다. 마찬가지로, NAND 연산은 Y=A·B, 즉 AND 게이트 출력에 NOT 게이트를 붙인 것이므로 Y=(A·B)’라고 표현할 수 있고, 드 모르간의 법칙에 의해 Y=A’+B’가 됩니다.

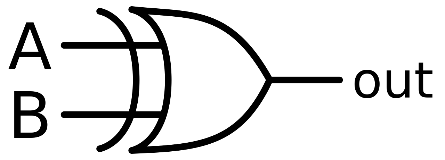
우리에게 논리 식이 주어진다면 NAND 게이트나 NOR 게이트 대신 AND 게이트와 OR 게이트만을 이용하여 schematic diagram을 구상하는 것이 편리하겠지만 실제 회로를 구현할 때에는 AND 게이트와 OR 게이트보다는 NAND 게이트와 NOR 게이트를 사용합니다. 이는 NAND 게이트와 NOR 게이트를 가지고서만으로도 모든 논리 회로를 구현할 수 있기 때문이고, NOT 게이트의 기능까지 내포하고 있기 때문에 AND 게이트, OR 게이트, NOT 게이트를 이용한 것보다 더욱 게이트를 적게 사용하고도 구현이 가능하기 때문입니다.

XOR 게이트는 XOR는 bitwise exclusive or의 약자로, 00001111이라는 입력이 있을 때, XOR 게이트를 거치면 11110000이라는 출력을 얻을 수 있기 때문에 이러한 이름을 갖습니다. XOR는 배타적 논리합이라고도 말하며 ⊕라는 기호를 사용하여 나타냅니다. Y=A ⊕ B일 경우, Y=AB’+A’B라고 연산되며 XOR 게이트의 진리표를 작성해보면 다음 <Table 3>와 같습니다.

<Table 3> 두 개의 입력을 갖는 XOR 게이트의 진리표

|  |  |  |
| --- | --- | --- |
| 입력 | | 출력 |
| A | B | Y |
| 0 | 0 | 0 |
| 0 | 1 | 1 |
| 1 | 0 | 1 |
| 1 | 1 | 0 |

위의 <Table 3>를 보면 알 수 있듯이, XOR 게이트는 참 입력의 개수가 홀수일 때 참을 출력하는 논리 게이트이므로 00001111이라는 입력과 11111111이라는 입력 두 개를 XOR 게이트에 인가할 경우, 11110000이라고 바뀌는 것입니다. XOR 게이트의 기호는 다음 <Figure 3>과 같이 표현이 가능합니다.



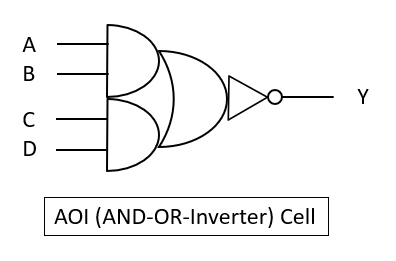
<Figure 3> XOR 게이트의 기호

1. 기본 논리게이트(AND, OR, NOT)와의 변환 관계에 대해 설명하시오.

위에서 언급하였듯이 NAND 게이트와 NOR 게이트만을 이용해서 AND, OR, NOT 게이트를 모두 구현할 수 있습니다. 우선, Y=~X과 같은 NOT 게이트를 구현하기 위해서는 NAND 게이트의 입력 부분을 묶어 하나의 입력 X를 인가해주면 되는데 NAND 게이트는 AND 게이트와 반대로 모든 입력이 0이면 1을, 모든 입력이 1이면 0을 출력하기 때문입니다. 다음으로, Y=A·B와 같은 AND 게이트를 구현하기 위해서는 NAND 게이트에 출력 부분에 NOT 게이트를 연결하면 됩니다. 마지막으로 Y=A+B는 Y=(A’B’)’라고도 표현이 가능하므로 OR 게이트는 두 입력을 각각 NOT 게이트를 거친 후, NAND 게이트의 입력으로 인가하게 하면 구현이 가능합니다. XOR 게이트는 A와 B의 보수를 논리 곱한 출력과 A의 보수와 B를 논리 곱한 출력을 논리 합하여 얻을 수 있으므로, 단순하게 AND, OR, NOT 게이트를 이용하여 표현하자면 두 개의 NOT 게이트와 두 개의 AND 게이트 그리고 한 개의 OR 게이트로 나타낼 수 있습니다.

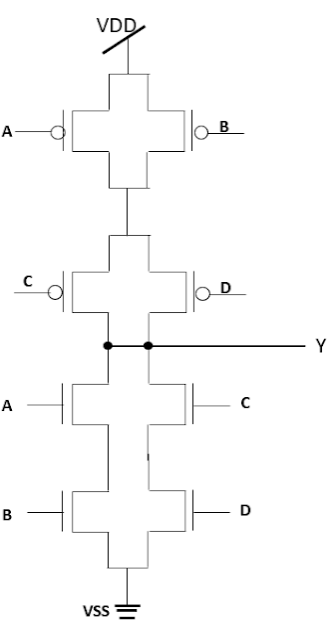
1. AND-OR-INVERT logic의 응용에 대해 설명하시오.

Q=(AB+CD)’라는 논리 함수가 있을 때, 드 모르간의 법칙에 의해 Q=(AB)’(CD)’로 바꿀 수 있고, 다시 드 모르간의 법칙에 의해 Q=(A’+B’)(C’+D’)로 바꿀 수 있습니다. A’+B’는 A와 B를 각각 NOT 게이트를 거치게 한 후, OR 게이트의 입력으로 인가한 것으로 NAND 게이트와 동일합니다. 마찬 가지도 C’+D’는 NAND 게이트를 거친 것과 동일하고 마지막으로 (A’+B’)와 (C’+D’)는 AND 게이트를 거치면 Q라는 출력을 얻게 됩니다. 또, NAND 게이트는 AND 게이트 뒤에 NOT 게이트를 거친 것이고, NOT 게이트를 거친 두 입력이 인가된 AND 게이트는 NOR 게이트로 변환이 가능합니다. 따라서, 위의 논리 함수 Q=(A’+B’)(C’+D’)는 아래 <Figure 4>이며, 실제로 Q=(AB+CD)’ 논리 함수를 그대로 AND, OR, NOT 게이트를 이용하여 구현한 diagram과 동일하다는 것을 알 수 있습니다.



<Figure 4> AND-OR-INVERTER

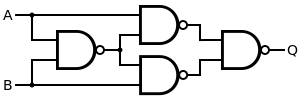
이러한 AOI Cell을 transistor-level로 그려보면 다음 <Figure 5>와 같고, 이는 표준 셀의 하나로써 널리 사용되는 셀 중 하나입니다.



<Figure 5> Transistor-level의 AND-OR-INVERTER Cell

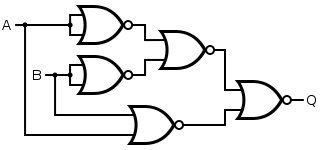
1. XOR의 logic의 구현 방법에 대해 설명하시오.

NAND 게이트만을 이용하거나, NOR 게이트만을 이용해서 모든 논리 함수를 구현할 수 있으므로 XOR 게이트도 이 두 개의 게이트를 각각 이용해서 구현할 수 있고, 또는 NAND 게이트와 AND 게이트, OR 게이트를 혼합하여 구현할 수도 있습니다. 우선, NAND 게이트만을 이용하여 구현한 XOR 게이트 회로에 대해 설명해보자면 다음 <Figure 6>의 예시를 들 수 있습니다.



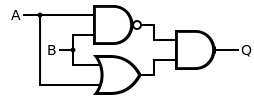
<Figure 6> NAND 게이트 네 개를 이용하여 구현한 XOR 게이트 회로

위 <Figure 6> schematic diagram을 Q, A, B에 대한 함수로 표현해보면, Q=((A(AB)’)’(B(AB)’)’)’이므로 Q=(A(AB)’)+(B(AB)’)가 되고, Q는 다시 Q=A(A’+B’)+B(A’+B’)=(A+B)(A’+B’)=AB’+A’B가 되므로 XOR 게이트 회로가 알맞게 구현되었음을 알 수 있습니다. 다음으로는 NOR 게이트만을 이용하여 구현한 XOR 게이트 회로에 대해 알아보겠습니다.



<Figure 7> NOR 게이트 다섯 개를 이용하여 구현한 XOR 게이트 회로

NOR 게이트를 이용하여 위의 <Figure 7>처럼 XOR 게이트 회로를 구현할 수 있는데 마찬가지로 Q, A, B에 대한 함수로 표현해보면, Q=((A’+B’)’+(A+B)’)’이므로 Q=(A’+B’)(A+B)=AB’+A’B가 되므로 XOR 게이트 회로가 올바르게 구현되었음을 알 수 있습니다. 마지막으로 NAND, AND, OR 게이트, 세 개의 게이트를 혼합하여 만든 XOR 게이트 회로에 대해 알아보겠습니다.



<Figure 8> NAND, OR, AND 게이트를 이용하여 구현한 XOR 게이트 회로

세 개의 게이트를 혼합하여 XOR 게이트 회로를 만들려면 위의 <Figure 8>처럼 구현할 수 있는데 마찬가지로 Q, A, B에 대한 함수로 표현해보면 Q=(AB)’(A+B)이므로 Q=(A’+B’)(A+B)=AB’+A’B가 됩니다.

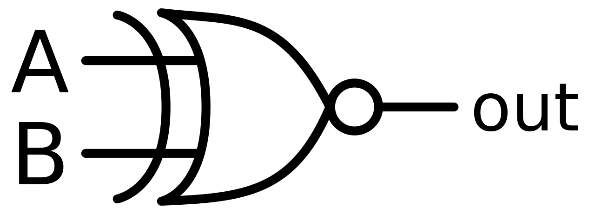
1. 기타 이론에 대해 설명하시오.

XNOR 게이트는 exclusive NOR gate로 약자로, 한국어로는 배타적 부정 논리합이라고 불립니다. XNOR 게이트는 두 개의 입력이 있을 때, 둘 다 참일 때나 둘 다 거짓일 때만 1을 출력하고 두 입력 값이 다르면 0을 출력하고, 여러 개의 입력이 있을 때는 참인 입력이 짝수 개 있을 때 1을 출력하는 성질을 보입니다. XNOR 게이트의 진리표를 작성하면 다음 <Table 4>와 같습니다.

<Table 4> 두 개의 입력을 갖는 XNOR 게이트의 진리표

|  |  |  |
| --- | --- | --- |
| 입력 | | 출력 |
| A | B | Y |
| 0 | 0 | 1 |
| 0 | 1 | 0 |
| 1 | 0 | 0 |
| 1 | 1 | 1 |

XNOR 연산은 ⊙를 이용하여 Y=A⊙B와 같이 표기하며 이 식을 정리하면 Y=AB+A’B’가 됩니다. XNOR 게이트의 기호는 다음 <Figure 9>와 같습니다.



<Figure 9> XNOR 게이트의 기호

XNOR 게이트 회로를 NOR, AND, OR 게이트만을 이용해서 구현한다면 두 개의 AND 게이트와 두 개의 NOT 게이트 그리고 한 개의 OR 게이트가 필요할 것입니다. 여러 개의 입력 중 참인 입력이 홀수 개인 경우, 1을 출력하는 XOR 게이트와 정반대의 역할을 하는 XNOR 게이트는 XOR 게이트의 기호, <Figure 3 >과 <Figure 9>을 비교해보면 XOR 게이트의 출력단에 동그라미를 그려 표현한다는 것을 알 수 있습니다.