

디지털 기본회로

기본 논리회로

기본 논리회로

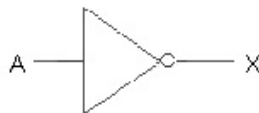
디지털 시스템의 기본 논리회로는 AND, OR, NOT Gate와 Flip-Flop 등으로 구성되어 있다. 이들 회로의 입력과 출력의 기준은 1 또는 0으로 표시하며 이것을 전압의 고저로 나타낸다. 높은 전압을 1, 낮은 전압을 0으로 나타내는 정논리계통(positive logic system)과 높은 전압을 0, 낮은 전압을 1로 나타내는 부논리계통(negative logic system), 그리고 펄스의 유무에 따라 1과 0을 규정하는 동적논리계통(Dynamic logic System) 등이 있다.

(1) 인버터(Inverter)

인버터는 NOT gate라 하며, 출력은 입력 논리신호의 반대값이 된다. 진리식은

$$X = \overline{A}$$

가 되며, 아래와 같은 기호와 진리표를 갖는다.



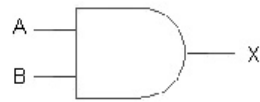
A	X
0	1
1	0

(2) AND gate

AND Gate는 입력신호가 모두 참이 되어야 출력이 참이 되는데, 진리식은

$$X = A \cdot B$$

가 되며, 아래와 같은 기호와 진리표를 갖는다.



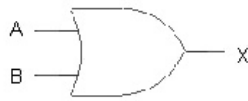
A	B	X
0	0	0
0	1	0
1	0	0
1	1	1

(3) OR GATE

OR gate는 입력신호중 어느 하나라도 참이 되며 출력이 참이 되는데, 진리식은

$$X = A + B$$

가 되며 다음과 같은 기호와 진리표를 갖는다.

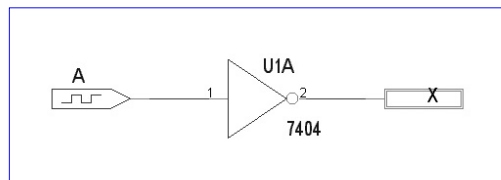


A	B	X
0	0	0
0	1	1
1	0	1
1	1	1

실험회로 I

회로개요

7404 인버터(Inverter)의 회로에서 입력신호의 변화에 따른 인버터의 출력 상태를 보기 위한 회로이다.



회로해석

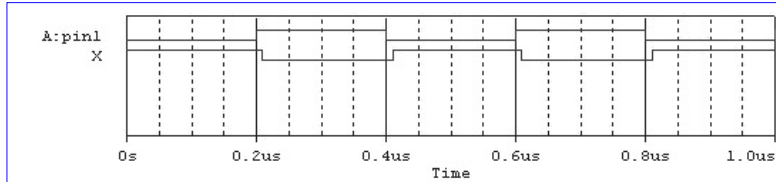
인버터(NOT gate)의 동작은 입력 논리신호의 반대가 되는 값을 출력하는 것으로 입력에 1이 인가될 때 출력은 0가 되고 입력에 0이 인가될 때 출력은 1이 된다.

시뮬레이션 조건

본 회로는 매우 간단한 디지털 회로의 NOT gate 회로로서 Final Time을 1[us]까지 20[ns]의 간격으로 Transient 해석을 한다. 입력신호는 0.2[us] 간격으로 1과 0을 반복하여 인가한다.

■ 시뮬레이션 결과

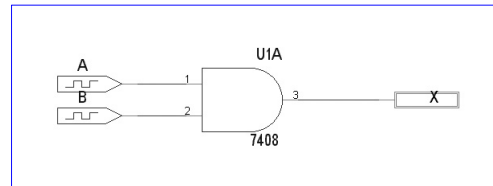
시뮬레이션 출력은 입력 신호 A:pin1에 대한 출력 상태 X를 조사하는 것으로 입력이 0이면 출력은 1, 입력이 1이면 출력은 0이 되는 것을 볼 수 있다. 7404 소자에 의해 입력과 출력신호는 약 20[ns] 정도 delay가 일어난다.



실험회로 II

■ 회로개요

7408 AND gate의 회로에서 두 입력 신호의 변화에 따른 AND gate의 출력 상태를 보기 위한 회로이다.



■ 회로해석

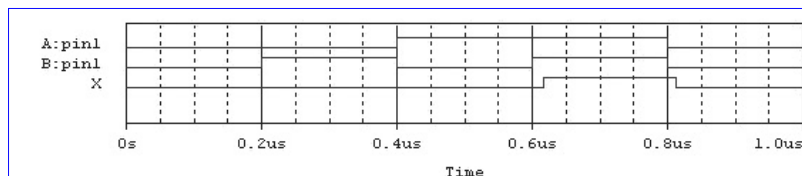
AND gate의 동작은 모든 입력 신호가 1이 될 때만 출력 상태가 1이 되며, 입력 신호중 한 개의 0 상태가 존재하며 출력 상태는 0가 된다.

■ 시뮬레이션 조건

본 회로는 간단한 AND gate로서 Final Time을 1[us]까지 20[ns]의 간격으로 Transient 해석을 한다. 입력 신호는 0.2[us] 간격으로 00, 01, 10, 11을 인가한다.

■ 시뮬레이션 결과

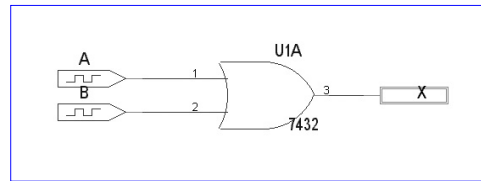
시뮬레이션 출력은 입력 신호 A:pin1와 B:pin1에 대한 AND gate의 출력 상태 X를 조사하는 것으로 입력이 00, 01, 10이면 출력은 0 상태, 입력이 11이면 출력은 1 상태가 된다. 7408 소자에 의해 입력과 출력 신호는 약 20[ns] 정도 delay가 일어난다.



실험회로 III

■ 회로개요

7432 OR gate의 회로에서 두 입력 신호의 변화에 따른 OR gate의 출력 상태를 보기 위한 회로이다.



■ 회로해석

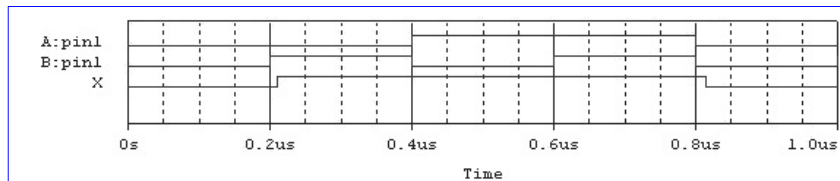
OR gate의 동작은 어떤 입력에도 1이 인가되면 출력 상태는 1 상태가 되며, 모든 입력에 0 상태가 인가될 때 출력은 0 상태가 된다.

■ 시뮬레이션 조건

본 회로는 간단한 OR gate 회로로서 Final Time을 1[us]까지 20[ns]의 간격으로 Transient 해석을 한다. 입력 신호는 0.2[us] 간격으로 00, 01, 10, 11을 인가한다.

■ 시뮬레이션 결과

시뮬레이션 출력은 입력 신호 A:OUT0와 B:OUT0에 대한 OR gate의 출력 상태 X를 조사하는 것으로 입력이 01, 10, 11이면 출력은 1 상태, 입력이 00이면 출력은 0상태가 된다. 7432 소자에 의해 입력과 출력 신호는 약 20[ns]정도 delay가 일어난다.



부울대수의 공리

부울대수의 공리

부울대수는 “0”과 “1”의 상태를 OR(+)와 AND(·)의 연산자만을 사용하는 대수.

공리 1 a) $A \neq 0$ 이면 $A = 1$, $A = 1$ 이면 $A \neq 0$
b) $A \neq 1$ 이면 $A = 0$, $A = 0$ 이면 $A \neq 1$

공리 2 a) $0 \cdot 0 = 0$, $1 \cdot 1 = 1$ b) $0 + 0 = 0$, $1 + 1 = 1$

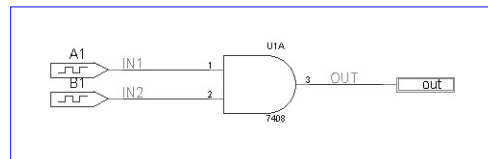
공리 3 a) $0 \cdot 1 = 1 \cdot 0 = 0$ b) $0 + 1 = 1 + 0 = 1$

공리 4 a) $\overline{\overline{A}} = A$ b) $\overline{0} = 1$

실험회로

회로개요

부울대수의 공리에 관한 실험은 앞의 NOT gate, AND gate, OR gate 회로에서 실행하였다. 본 실험은 AND gate를 이용하여 간단한 두 0과 1의 입력 신호를 인가할 때의 출력 상태를 보기 위한 것으로 앞의 실험과 같이 7408 AND gate를 사용하였다.



회로해석

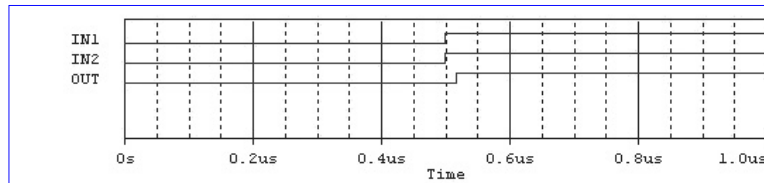
입력이 0 상태일 때와 1 상태일 때, AND gate의 출력 상태를 보기 위한 것으로 공리 2의 a)이 된다.

시뮬레이션 조건

본 회로는 부울대수의 간단한 공리를 확인하는 회로로서 Final Time을 1[us]까지 20[ns]의 간격으로 Transient 해석을 한다. 입력 신호는 0.5[us] 간격으로 1과 0을 인가한다.

■ 시뮬레이션 결과

시뮬레이션 출력은 같은 입력 신호 IN1과 IN2에 대한 출력 상태 OUT를 조사하는 것으로 입력이 1 상태이면 출력은 1 상태, 입력이 0 상태이면 출력은 0상태가 된다. 즉 이것은 공리 2의 a)이 된다. 7408 소자에 의해 입력과 출력신호는 약 20[ns] 정도 delay가 일어난다.



부울대수의 정리

부울대수의 정리

부울대수의 공리를 기본으로 교환, 결합, 분배법칙 등 다음과 같은 정리가 있다.

정리 1 교환법칙

a) $A \cdot B = B \cdot A$ b) $A + B = B + A$

정리 2 결합법칙

a) $(A \cdot B) \cdot C = A \cdot (B \cdot C)$
b) $(A + B) + C = A + (B + C)$

정리 3 분배법칙

a) $(A + B) \cdot (A + C) = A + B \cdot C$
b) $A \cdot B + A \cdot C = A \cdot (B + C)$

정리 4

a) $A \cdot 0 = 0$ b) $A + 0 = A$
c) $A \cdot 1 = A$ d) $A + 1 = 1$

정리 5

a) $A \cdot \overline{A} = 0$ b) $A + \overline{A} = 1$

정리 6 동일법칙

a) $A \cdot A = A$ b) $A + \overline{A} = A$

정리 7 흡수법칙

a) $A \cdot (A + B) = A$ b) $A + A \cdot B = A$

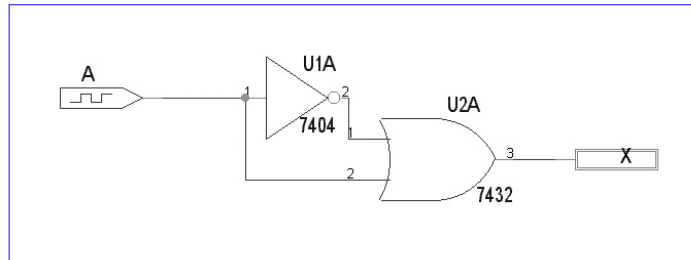
정리 8 부정법칙

$\overline{\overline{A}} = A$

실험회로 I

■ 회로개요

정리 4와 정리 6의 동일 법칙은 앞의 AND gate와 OR gate의 실험에서 충분하게 검증되었으며, 교환법칙, 결합법칙, 분배법칙 등은 기본적으로 성립함을 알 수 있다. 본 실험은 정리 5를 검증하기 위한 것으로 7432 OR gate의 두 입력에 A와 반진된 A 신호를 입력할 때 출력 상태를 확인하기 위한 회로



■ 회로해석

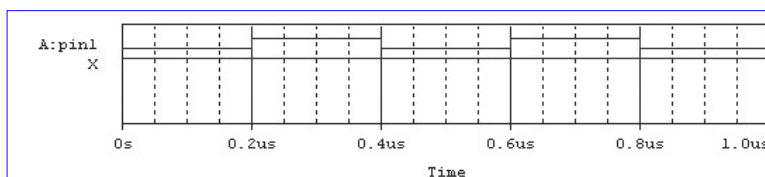
정리 5 b)와 같이 두 입력 신호가 반대가 되어 OR gate에 입력될 때, 그 출력은 항상 1 상태가 된다. 즉 $A + \bar{A} = 1$ 이 된다.

■ 시뮬레이션 조건

본 회로는 부울대수의 정리 5 b)를 확인하는 회로로서 Final Time을 1[us]까지 20[ns]의 간격으로 Transient 해석을 한다. 입력 신호는 0.2[us] 간격으로 0과 1을 반복 인가한다.

■ 시뮬레이션 결과

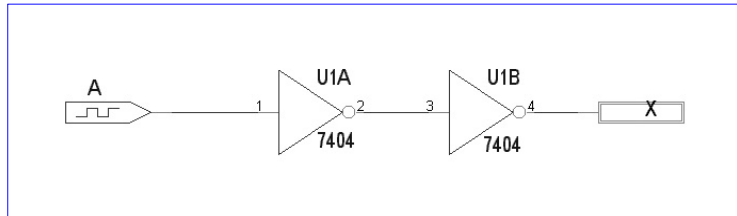
시뮬레이션 출력은 입력 신호 A:pin1에 대한 출력 상태 X를 확인하는 것으로 입력상태에 관계없이 출력은 항상 1 상태가 된다. 즉 $A + \bar{A} = 1$ 이 된다.



실험회로 II

회로개요

정리 8의 부정법칙을 검증하기 위한 회로로서 7404 인버터 두 개를 직렬로 연결할 때 출력 상태를 보기 위한 회로이다.



회로해석

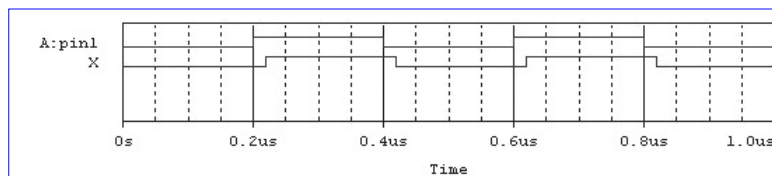
정리 8의 부정법칙과 같이 두 개의 인버터가 직렬로 연결되면 출력은 입력과 \neg 같은 값이 된다. 즉 $\overline{\overline{A}}=A$ 이 된다.

시뮬레이션 조건

본 회로는 정리 8의 부정법칙을 검증하는 것으로 Final Time을 1[us]까지 20[ns]의 간격으로 Transient 해석을 한다. 입력 Source의 조건은 0.2[us] 간격으로 0과 1을 반복 인가한다.

시뮬레이션 결과

시뮬레이션 출력은 입력 신호 A:pin1에 대한 출력 상태 X를 확인하는 것으로 입력이 0 상태이면 출력은 1 상태, 입력이 1 상태이면 출력은 0 상태가 된다. 2개의 7404 소자가 직렬로 연결되어 있는 관계로 입력과 출력 신호는 약 40[ns] 정도 delay가 일어난다.



De-Morgan의 정리

De-Morgan의 정리

De-Morgan의 정리는 NAND gate와 NOR gate의 응용 및 논리회로를 간소화하는데 이용된다.

1. De-Morgan의 정리

$$\text{a) } \overline{A+B} = \overline{A} \cdot \overline{B} \quad \text{b) } \overline{A \cdot B} = \overline{A} + \overline{B}$$

2. NAND gate를 이용한 기본논리회로

$$\text{a) } A \cdot B = A \cdot B \quad \text{b) } A+B = A+B = \overline{\overline{A} \cdot \overline{B}}$$

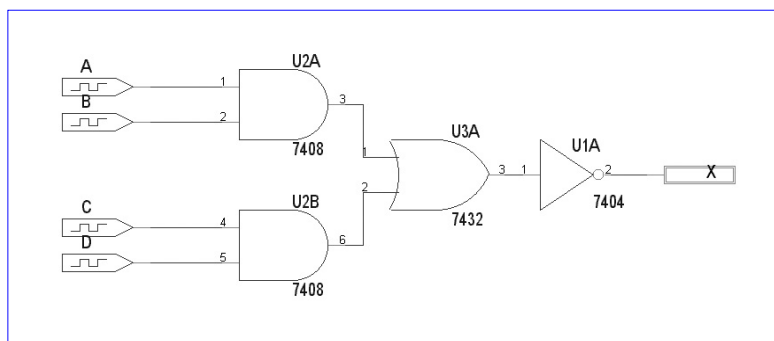
3. NOR gate를 이용한 기본논리회로

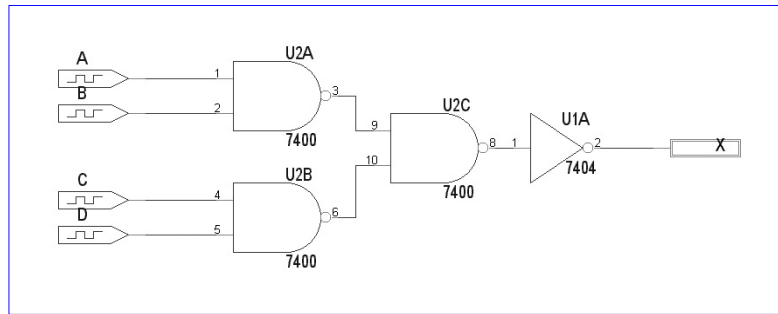
$$\text{a) } A+B = A+B \quad \text{b) } A \cdot B = A \cdot B = \overline{\overline{A} + \overline{B}}$$

실험회로

회로개요

De-Morgan의 정리는 NAND gate나 NOR gate를 이용하여 논리회로를 간소화하는데 사용된다. 본 실험은 일반 회로인 $\overline{A \cdot B + C \cdot D}$ 의 회로를 NAND gate로 구성된 $\overline{\overline{A \cdot B} \cdot \overline{C \cdot D}}$ 회로의 변경에 따른 입력에 대한 출력 상태를 보기 위한 회로이다.





■ 회로해석

본 회로는 De-Morgan의 정리에 의해 AND gate, OR gate로 이루어진 일반 회로를 NAND gate로 이루어진 회로로 변환하는 것이다. 본 실험에서는 4개의 gate의 사용으로 회로의 간소화는 이루어지지 않았으나 NAND로 이루어진 회로의 구성으로 회로의 실제 구현이 간소화되었다.

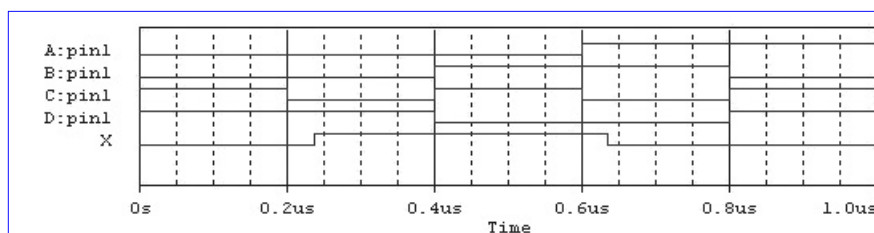
즉, $\overline{A \cdot B + C \cdot D} = \overline{A \cdot B} \cdot \overline{C \cdot D}$ 로 변환되어 NAND gate로 구성된 회로가 된다.

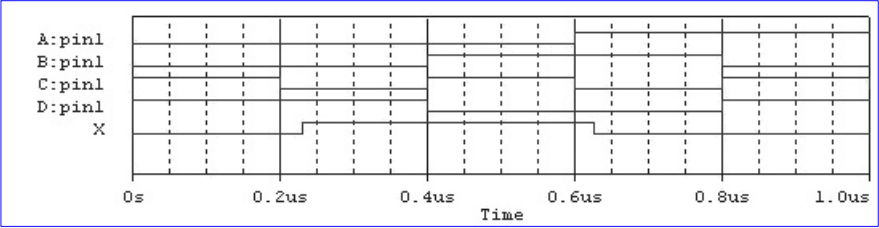
■ 시뮬레이션 조건

본 회로는 De-Morgan의 정리에 의해 NAND gate로 구성된 회로로 변환하는 것으로 Final Time을 1[us]까지 20[ns]의 간격으로 Transient 해석을 한다. 입력은 0.2 [us] 간격으로 0011, 0010, 0110, 1100, 1011 순으로 A, B, C, D에 각각 입력한다.

■ 시뮬레이션 결과

시뮬레이션 결과는 입력 신호 A:pin1, B:pin1, C:pin1, D:pin1에 대한 출력 상태 X를 확인하는 것으로 위의 두 회로에 대한 결과를 각각 나타내었다. 첫 번째 결과는 $\overline{A \cdot B + C \cdot D}$ 의 일반적인 회로에 대한 결과이며, 두 번째 결과는 NAND gate로 구성된 $\overline{A \cdot B} \cdot \overline{C \cdot D}$ 회로에 대한 결과이다. 입력 신호의 변화에 대한 두 결과는 모두 일치한다. 두 결과 모두 세 개의 gate를 통해야 하므로 입력과 출력 신호는 약 50~60[ns] 정도 delay가 일어난다.





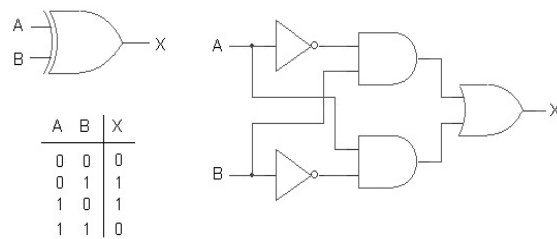
Exclusive OR

Exclusive OR

Exclusive OR(XOR)의 대수식은

$$X = A\bar{B} + \bar{A}B$$

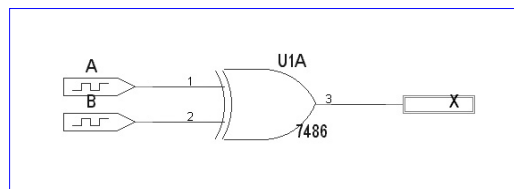
로 나타내며, 흔히 $X=A\oplus B$ 로도 표시한다. 즉 두 입력이 서로 다르면 출력은 1이 되고 두 입력이 같으면 출력은 0가 된다. 기호와 진리표, 그리고 회로는 다음과 같다.



16.5.1 실험회로

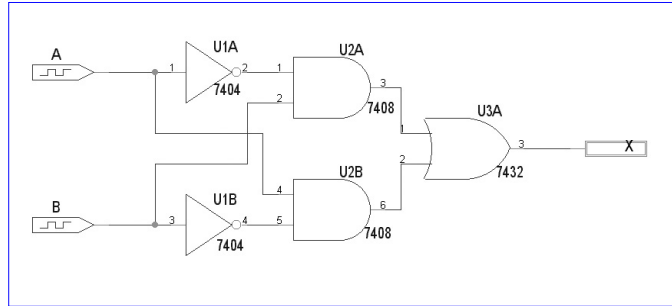
■ 회로개요

Exclusive OR gate는 식 $X=A\bar{B} + \bar{A}B$ 와 같이 나타낸다. 본 실험은 Exclusive OR 회로를 기본 gate를 이용하여 구성한 출력 상태와 7486의 Exclusive OR gate의 출력 상태를 비교한다.



■ 회로해석

Exclusive OR 회로의 동작을 확인하기 위해 식 $X=A\bar{B} + \bar{A}B$ 회로를 일반 논리 gate로 구성하고 7486의 Exclusive OR gate의 출력과 비교한다.

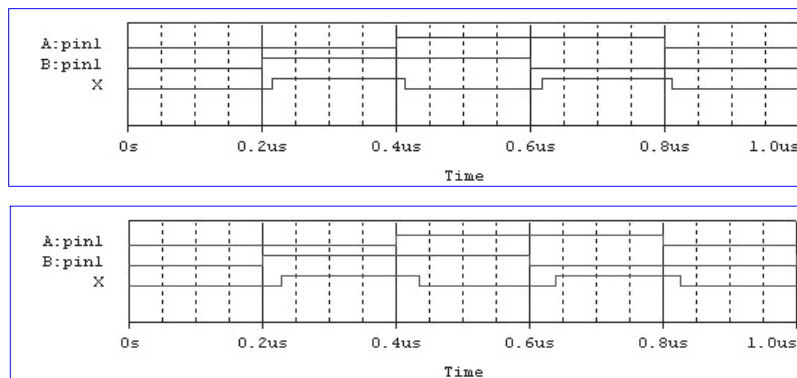


■ 시뮬레이션 조건

본 회로는 Exclusive OR 회로로서 Final Time을 1[us]까지 20[ns]의 간격으로 Transient 해석을 한다. 입력 신호는 0.2[us] 간격으로 00, 01, 11, 10의 신호를 인가한다.

■ 시뮬레이션 결과

시뮬레이션 출력은 입력 신호 A:pin1과 B:pin1에 대한 출력 상태 X를 확인하는 것으로 두 입력이 같으면 0 상태, 다르면 1 상태가 된다. 입력과 출력 사이에 3개의 gate가 있으므로 약 50~60[ns] 정도 delay가 일어난다.



가산기

가산기

가산기는 반가산기(half adder)와 전가산기(full adder)가 있다.

1. 반가산기

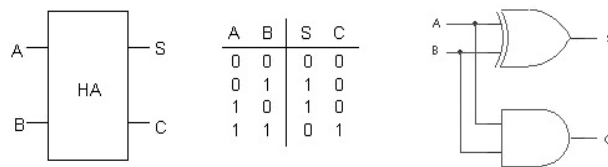
한자리수 A와 B를 합할 때 발생하는 결과는 A와 B의 합과 다음 자리의 자리올림수(Carry)가 된다. 예를 들어

$$\begin{array}{r} \square \square \square \\ \text{자리올림 수} \end{array} \quad \begin{array}{r} 1 \leftarrow A \\ + 1 \leftarrow B \\ \hline 10 \square \square \\ \text{합} \end{array}$$

이 된다. 이러한 계산을 하기 위한 합(S)과 자리올림수(C)의 논리식은

$$\begin{aligned} S &= \overline{A}B + A\overline{B} = A \oplus B \\ C &= A \cdot B \end{aligned}$$

로 나타내며, 기호와 진리표 그리고 회로는 다음과 같다.



2. 전가산기

한자리수 A와 B, 그리고 자리올림수를 합할 때에 사용되는 것으로 결과는 A와 B의 합(S)과 자리올림수(Carry)가 된다. 예를 들어

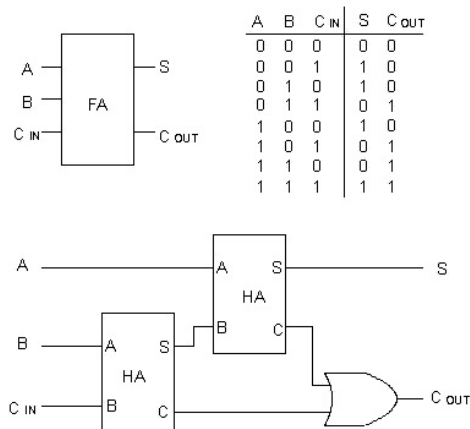
$$\begin{array}{r} \square \square \square \\ \text{자리올림 수} \end{array} \quad \begin{array}{r} 1 \leftarrow \text{자리올림수} \\ 1 \leftarrow A \\ + 1 \leftarrow B \\ \hline 11 \square \square \\ \text{합} \end{array}$$

이 된다. 이러한 계산을 하기 위한 합(S)과 자리올림수(C)의 논리식은

$$\begin{aligned} S_n &= \overline{A_n} \overline{B_n} C_{n-1} + \overline{A_n} B_n \overline{C}_{n-1} + A_n \overline{B_n} \overline{C}_{n-1} + A_n B_n C_{n-1} \\ C_n &= A_n \cdot B_n \end{aligned}$$

로 나타내며, 기호와 진리표 그리고 회로는 다음과 같다.

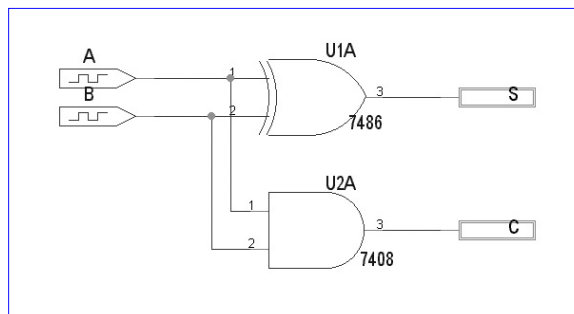
로 나타내며, 기호와 진리표 그리고 회로는 다음과 같다.



실험회로 I

회로개요

본 실험은 반가산기(half adder) 실험으로 7486 EX-OR gate의 합(S) 회로와 7408 AND gate의 자리올림수(C) 회로를 구성하여 반가산기의 동작을 보기 위한 회로이다.



회로해석

$S = \overline{A}B + A\overline{B} = A \oplus B$ 의 합 회로와 $C = A \cdot B$ 의 자리올림 회로에 대한 반가산기의 동작을 본다.

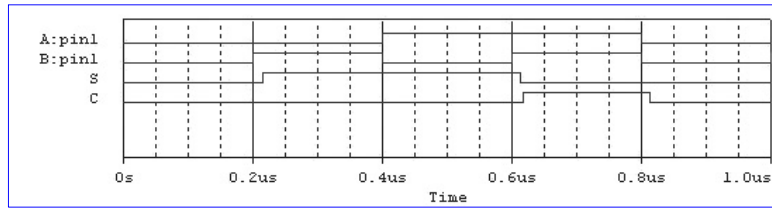
시뮬레이션 조건

본 회로는 반가산기 회로로서 Final Time을 1[us]까지 20[ns]의 간격으로 Transient 해석을 한다. 입력 신호는 0.2[us] 간격으로 00, 01, 10, 11을 인가한다.

시뮬레이션 결과

시뮬레이션 출력은 입력 신호 A:pin1와 B:pin1에 대한 출력 상태 S와 C를 관찰하는 것으로

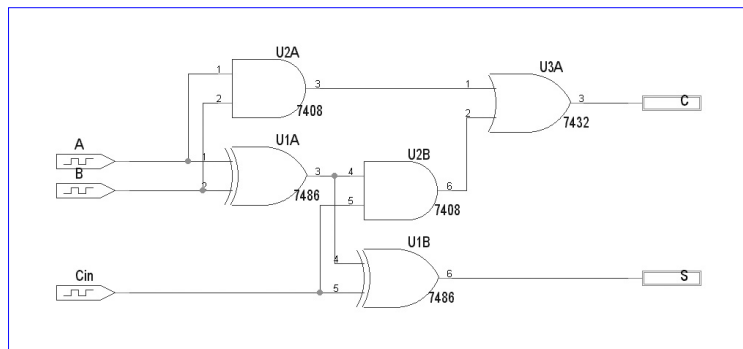
Exclusive OR의 합 회로와 AND gate의 자리올림 회로의 결과를 확인할 수 있다. 1개의 gate를 통한 입력과 출력은 약 20[ns] 정도 delay가 일어난다.



실험회로 II

회로개요

두 개의 반가산기와 한 개의 OR gate로 구성된 전가산기의 회로를 구성하여 전가산기의 동작을 보기 위한 회로이다.



회로해석

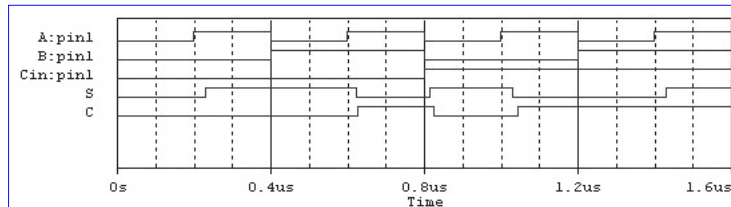
$S_n = \overline{A}_n \overline{B}_n C_{n-1} + \overline{A}_n B_n \overline{C}_{n-1} + A_n \overline{B}_n \overline{C}_{n-1} + A_n B_n C_{n-1}$ 의 합 회로와 $C=A \cdot B$ 의 숫자올림 회로 대신에 반가산기 두 개와 한 개의 OR gate로 전가산기 회로를 구성한다.

시뮬레이션 조건

본 회로는 전가산기 회로로서 Final Time을 1.6[us]까지 20[ns]의 간격으로 Transient 해석을 한다. 입력 신호는 000, 001, 010, 011, 100, 101, 110, 111의 신호를 Cin, B, A에 각각 인가한다.

■ 시뮬레이션 결과

시뮬레이션 출력은 입력 신호 A:pin1, B:pin1, Cin:pin1에 대한 출력 상태 S와 C를 조사하는 것으로 입력이 000이면 출력은 S=0, C=0 상태이고 입력이 001, 010, 100이면 출력은 S=1, C=0 상태이다. 또 입력이 011, 101, 110일 경우 출력은 S=0, C=1이 되며, 입력이 111일 경우 출력은 S=1, C=1이 된다. 입력과 출력 사이에는 gate의 개수에 따라 약 20~60[ns] 정도 delay가 일어난다.



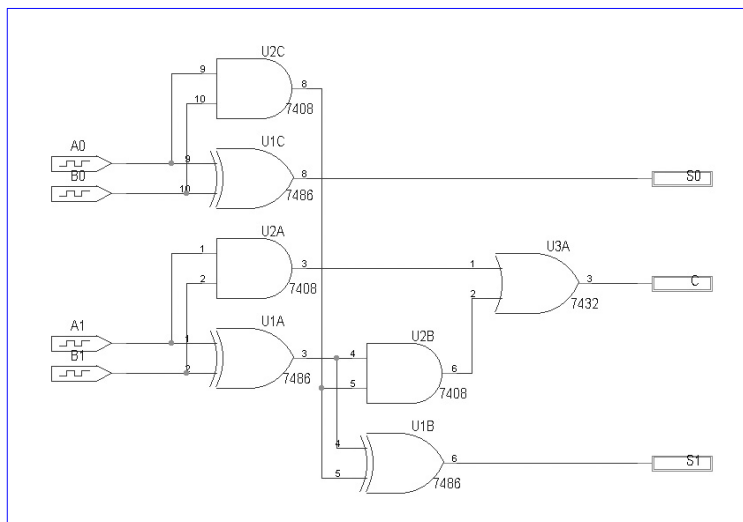
실험회로 III

■ 회로개요

전가산기와 반가산기를 이용한 2비트 병렬 가산기의 회로이다.

■ 회로해석

2비트 가산은 한 개의 반가산기와 한 개의 전가산기를 필요로 한다. 먼저 입력되는 하위 비트는 반가산기로 계산되며, 상위 비트는 하위 비트에서 출력되는 자리올림수와 함께 전가산기에 입력되어 계산된다.

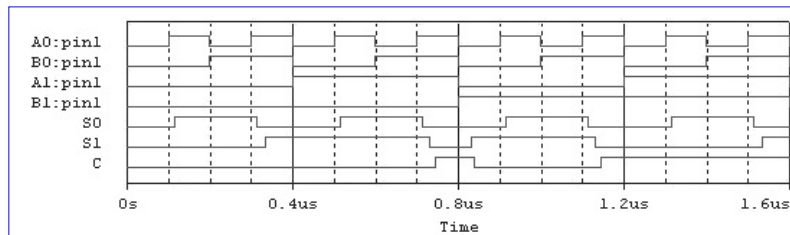


■ 시뮬레이션 조건

본 회로는 2비트 병렬 가산기 회로로서 Final Time을 1.6[us]까지 20[ns]의 간격으로 Transient 해석을 한다. 입력 신호는 0.1[us] 간격으로 0000, 0001, 0010, ..., 1100, 1101, 1111의 16가지 신호를 인가한다.

■ 시뮬레이션 결과

시뮬레이션 출력은 입력 신호 A0:pin1, B0:pin1, A1:pin1, B1:pin1의 4가지에 대한 출력 상태 S0, S1, C를 조사하는 것으로 입력과 출력의 관계에서 2비트 병렬 가산기의 동작을 볼 수 있다. 입력과 출력 사이에 gate수에 따라 약 20~60[ns] 정도 delay가 일어난다.

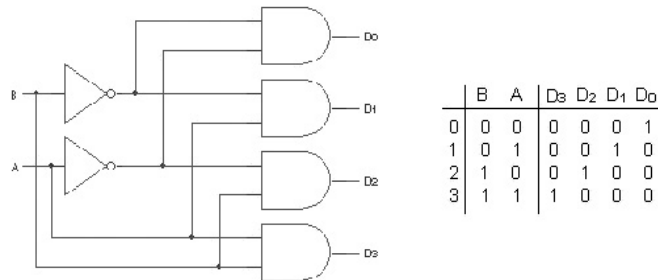


디코드 및 인코드

디코드 및 인코드

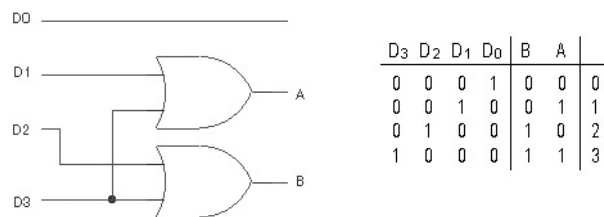
1. 디코드(Decoder)

이진수로 표시된 입력 코드를 어떠한 상태 또는 명령을 나타내는가를 해독하는 장치를 말하며, BCD 코드를 7-segment 숫자표시기로 나타내는 등 여러 가지 경우가 있다. 다음 회로는 2선-4선 디코드 회로로서 2진수 입력 A와 B의 4가지 조합에 대하여 구분되는 4개의 출력으로 변환하는 회로이다.



2. 인코드(Encoder)

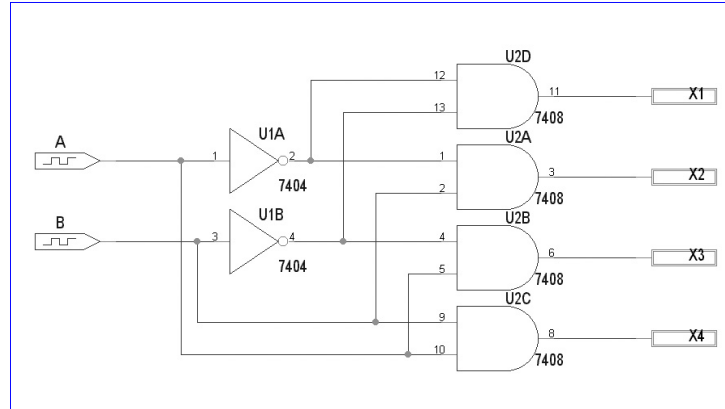
상태 또는 명령들을 이진수의 코드로 변환하는 장치로 디코드의 반대 기능을 갖는다. 다음 회로는 4개의 입력을 2진수 출력으로 변환하는 회로이다.



실험회로 I

■ 회로개요

2선-4선 디코드 회로로서 A와 B의 두 2진수 입력의 4가지 조합에 대응하는 4개의 출력을 변환하는 회로이다.



■ 회로해석

두 개의 2진수 입력을 4가지 조합으로 변환하기 위해서는 각 출력과 입력 관계는 아래와 같다.

$$X_1 = \overline{A} \cdot \overline{B}$$

$$X_2 = \overline{A} \cdot B$$

$$X_3 = A \cdot \overline{B}$$

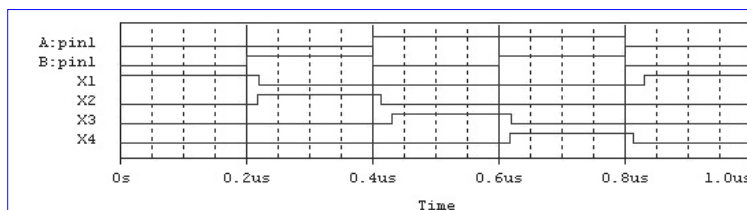
$$X_4 = A \cdot B$$

■ 시뮬레이션 조건

본 회로는 간단한 디지털 회로로서 20[ns]의 간격으로 Final Time을 1[us]으로 하여 Transient 해석을 하면 된다. 입력 Source의 조건은 0.2[us] 간격으로 0과 1을 반복 인가한다.

■ 시뮬레이션 결과

다음 출력은 입력 Source A:pin1, B:pin1에 대한 X1, X2, X3, X4의 출력 상태를 확인하는 것으로 A와 B의 입력이 00상태이면 X1이 1 상태, 01 입력이면 X2가 1 상태, 10이면 X3가 1 상태, 11이면 X4가 1 상태가 되는 것을 볼 수 있다.



실험회로 II

■ 회로개요

8개의 각각 다른 입력신호인 8진수를 2진수로 변환하는 8선-3선 엔코드 회로이다.

■ 회로해석

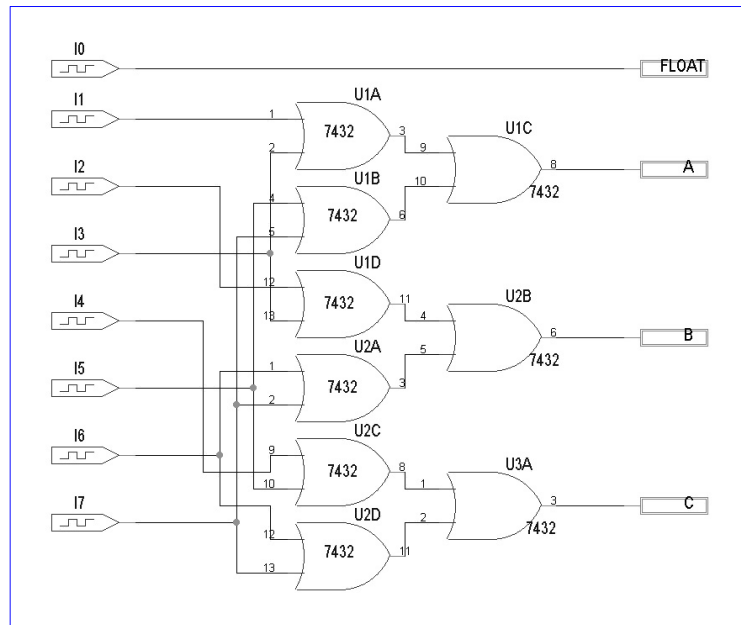
10진수를 2진수로 변환하기 위한 회로로서 각 2진수의 출력은

$$A = I_1 + I_3 + I_5 + I_7 + I_9$$

$$B = I_2 + I_3 + I_6 + I_7$$

$$C = I_4 + I_5 + I_6 I_7$$

가 된다. 여기서 I는 입력을 나타내며 A, B, C, D는 2진 출력을 나타낸다.



■ 시뮬레이션 조건

본 회로는 8-3 엔코더회로로서 20[ns]의 간격으로 Final Time을 1.6[us]으로 하여 Transient 해석을 하면 된다. 입력신호는 0.2[us] 간격으로 0와 1을 I1, I2, I3, ..., I6, I7순으로 인가한다.

■ 시뮬레이션 결과

출력은 8입력 대한 출력 상태 A, B, C를 확인하는 것이다. 입력과 출력 사이에 약 20[ns] 정도 delay가 일어난다.

