컴퓨터 공학 기초 설계 및 실험1

예비 & 결과 보고서

실험제목: OR, AND, NOT, NAND, NOR, Boolean algebra

실험일자: 2018년 03월 15일 (금)

제출일자: 2018년 03월 21일 (목)

학 과:컴퓨터정보공학부

담당교수: 이혁준

실습분반: 03-금012

학 번: 2018202074

성 명: 김상우

결과보고서

1. 제목 및 목적
   1. 제목

OR, AND, NOT, NAND, NOR, Boolean algebra

* 1. 목적

기본 논리 게이트인 OR 게이트, AND 게이트, NOT 게이트를 비롯, 그들의 결합인 NAND게이트 등을 사용해본다. 또한 이들을 통해서 주어진 회로를 구현함과 동시에 결과값을 예측, 측정, 비교해본다. 이를 통해 회로 해석, 연결 능력을 향상시키는 것을 주 목적으로 한다. 추가적으로 부울 대수를 바탕으로 논리식을 줄여보며 식 해석능력 및 변형능력의 향상도 목적으로 둔다.

1. 실험 결과

-(실험 1.3) OR 게이트 회로 입/출력 전압 측정

논리식: A + B = Z

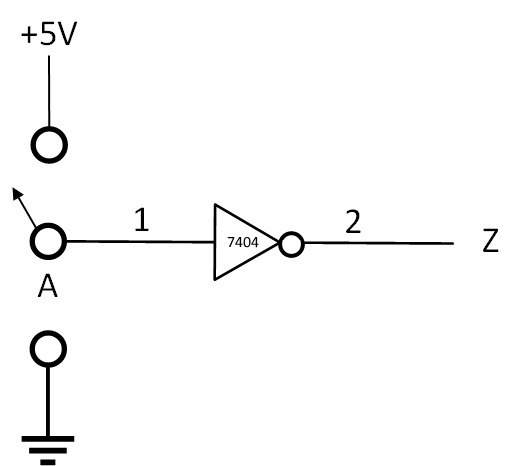
|  |  |  |
| --- | --- | --- |
| 입력 | | 출력 |
| A | B | Z |
| 0 | 0 | 0 |
| 0 | +5 | 1 |
| +5 | 0 | 1 |
| +5 | +5 | 1 |

-(실험 1.4) AND 게이트 회로 입/출력 전압 측정

논리식: A \* B = Z

|  |  |  |
| --- | --- | --- |
| 입력 | | 출력 |
| A | B | Z |
| 0 | 0 | 0 |
| 0 | +5 | 0 |
| +5 | 0 | 0 |
| +5 | +5 | 1 |

-(실험 1.6) NOT회로 입/출력 전압 측정

논리식: A’(상단 막대 기호 대체)=Z

|  |  |
| --- | --- |
| 입력 | 출력 |
| A | Z |
| 0 | 1 |
| +5 | 0 |

-(실험 2.3) NAND를 사용한 AND 동작 회로



|  |  |  |  |
| --- | --- | --- | --- |
| 입력 | | 출력 | |
| A | B | Y | Z |
| 0 | 0 | 1 | 0 |
| 0 | +5 | 1 | 0 |
| +5 | 0 | 1 | 0 |
| +5 | +5 | 0 | 1 |

논리식: { ( A \* B )’ \* ( A \* B )’ }’ = { ( A \* B )’ }’ = A \* B = Z \* 정리 7, 9참고

-(실험 2.5) NAND를 사용한 OR 동작 회로



|  |  |  |
| --- | --- | --- |
| 입력 | | 출력 |
| A | B | Z |
| 0 | 0 | 0 |
| 0 | +5 | 0 |
| +5 | 0 | 0 |
| +5 | +5 | 1 |

논리식: ( ( A \* A )’ \* ( B \* B )’ )’ = (A’ \* B’) ’ = A \* B = Z \* 정리 7 참고

-(실험 3.2) OR 게이트와 AND 게이트를 사용한 동작 회로



|  |  |  |  |
| --- | --- | --- | --- |
| 입력 | | | 출력 |
| A | B | C | Z |
| 0 | 0 | 0 | 0 |
| 0 | 0 | +5 | 0 |
| 0 | +5 | 0 | 0 |
| 0 | +5 | +5 | 1 |
| +5 | 0 | 0 | 1 |
| +5 | 0 | +5 | 1 |
| +5 | +5 | 0 | 1 |
| +5 | +5 | +5 | 1 |

논리식{ ( A + B ) \* ( B \* C ) } + A = { (A \* B \* C) + ( B \* B \* C ) } + A = { A \* B \* C + (B \* C) ) } + A = (A + 1) \* (B \* C) + A= B \* C + A

* 정리 3(분배법칙), 정리 8 참고

1. 고찰

해당 실험을 하면서 트랜지스터가 세게 고정되어 있지 않거나 측정기구와의 연결이 느슨할 경우 실험 결과가 확실하게 나오지 않는 경우가 있었습니다. (대표적으로 채널 1번과 연결했을 때 기기에 표시되는 수가 0과 1을 반복하거나 혹은 일부분이 옅게 나오기도 했다.) 이러한 점은 간단한 실험인 경우에는 문제가 되지 않지만, 실제 부품 및 제품을 출시할 때나 혹은 새로운 이론을 낼 때 이러한 오류가 생긴다면 큰 문제가 발생될 것이다. 이에 대해 고정을 굳세게 바꿔주는 부품을 추가적으로 장착하는 등의 노력이 필요할 것이다. 추가적으로 실험 3.2와 같이 더 간결한 방법으로 연결할 수 있는 회로도 있을 수 있으므로 이후에 회로를 간결하고 효율적으로 바꾸는 것 또한 필요해 보인다.

예비보고서

1. 제목 및 목적
   1. 제목

De Morgan’s theorem, karnaugh map, Xor gate

* 1. 목적

‘드모르간 정리’를 실습을 통해 배우면서 논리 게이트를 다루는 방법에 대해 심화적으로 배운다. 또한 카르노도 맵을 통해서 논리게이트에 대한 새로운 공식 및 표현을 배운다. 추가적으로 Xor gate를 사용하여 조립할 수 있는 회로의 범위를 넓혀 컴퓨터공학이라는 분야에 대한 견문을 넓히는 걸 목적으로 두고 있다.

1. 원리(배경지식)

**드모르간 정리**: 1) ( A \* B )’ = A’ + B’ 논리곱을 논리합으로 변경한다.

=

드 모르간의 제 1 정리는 A와 B의 논리곱의 보수는 A’와 B’의 논리합이 같음을 의미.

2) ( A + B )’ = A’ \* B’ 논리합을 논리곱으로 변경한다.

=

드 모르간의 제 2 정리는 A와 B의 논리합의 보수는 A’와 B’의 논리곱이 같음을 의미.

\*위의 공식을 응용 시 NAND나 NOR만으로 AND, OR, NOT을 구현할 수 있음.

**카르노 맵(Kamaugh Map)**

부울 식을 간소화하기 위한 방법 중 하나로, 진리표와 유사한 형태를 띄고 있으며 각 셀에는 입력변수의 2진값이 포함되어있다. 변수의 개수에 무관하게 적용할 수 있으며 진리표와 달리 행과 열로 구성되지 않았다. 주로 말아감기를 통해 논리식을 구할 수 있는데 방법에는 크게 최소항(Minterm)과 최대항(Maxterm)이 있다.

최소항(Minterm): 두 칸 항에서 변하는 변수는 버리고 변하지 않는 변수가 1이면 그대로 0이면 보수를 취해 곱하는 방법.

최대항(Maxterm): 두 칸 항에서 변하는 변수는 버리고 변하지 않는 변수가 0이면 그대로 1이면 보수를 취해 곱하는 방법.

\*쌍방 변하는 변수는 버리나 보수를 취해 곱하는 대상이 각각 0,1이라는 점에서 다르다.

Ex)3 input And회로는 이렇게 표현한다.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| A | BC | 00 | 01 | 11 | 10 |
| 0 | | 0 | 0 | 0 | 0 |
| 1 | | 0 | 0 | 1 | 0 |

**XOR 게이트(베타적 논리합)**

|  |  |  |  |
| --- | --- | --- | --- |
| 이름 | **XOR 게이트(베타적 논리합)** | | |
| 논리기호 |  | | |
| 논리식 | Z= A \* B’ + A’ \* B | | |
|  | 입력 | | 출력 |
|  | A | B | Z |
|  | 0 | 0 | 0 |
|  | 0 | 1 | 1 |
|  | 1 | 0 | 1 |
|  | 1 | 1 | 0 |
| 정리 | A와 B가 동일할 시 0을, A와 B가 다를 시 1을 출력 | | |

NAND와 NOR처럼 XNOR가 존재.

XNOR의 경우 A와 B가 동일할 시 1을, A와 B가 다를 시 0을 출력한다.

1. 참고문헌

서적:

이원석,정길수/논리회로 실험/생능출판/2010

웹:

카르노도/ https://terms.naver.com/entry.nhn?docId=754013&cid=42341&categoryId=42341

XOR 게이트/https://terms.naver.com/entry.nhn?docId=2835925&cid=40942&categoryId=32830