Chapter2 논리 회로

Point1 불 대수

불대수

* 2진수의 값으로 논리적 동작을 취급하는 대수
* 하나의 변수는 0 or 1의 값을 가짐
* 불 대수의 연산자는 논리곱(AND), 논리합(OR), 논리 부정(NOT)이 있다
* 벤다이어그램으로 이해하면 쉽다.(교집합, 합집합 개념)

|  |  |  |  |
| --- | --- | --- | --- |
| 종류 | 논리식 | 종류 | 논리식 |
| 합의 법칙 | X + 0 = X  X + 1 = X  X + X = X  X + = 1 | 곱의 법칙 | X 〮 0 = 0  X 〮 1 = X  X 〮 X = X  X 〮 = 0 |
| 교환 법칙 | X + Y = Y + X  X 〮 Y = Y 〮 X | 흡수 법칙  ★ | X + Y 〮 Y = X  X + 〮 Y = X + Y  X 〮(X + Y) = X |
| 결합 법칙 | X + (Y + Z) = (X + Y) + Z  X 〮 (Y 〮 Z) = (X 〮 Y) 〮 Z | 분배 법칙 | X 〮 (Y + Z) = (X 〮 Y) + (X 〮 Z)  X + (Y 〮 Z) = (X + Y) 〮 (X + Z) |
| 대합성의 법칙 | = X | ★드모르간의 법칙 | = 〮  = + |

* 불 대수의 X의 값에 0과 1을 대입할 수 있으며, 각각의 값을 대입해 보면 다음과 같이

X + 1 = 1의 경우 X에 0 or 1을 넣으면 0 + 1 = 1, 1 + 1 = 1이 되므로 X + 1 = 1 이 됨

논리식의 간략화

* 논리 함수를 간략화시키는 방법으로는 불대수를 이용하는 방법과 카르노 맵을 이용하는 방법이 있음

〮 Y + X 〮 + X 〮 Y

= 〮 Y + X 〮 ( + Y) <- 분배 법칙(X 〮 + X 〮 Y = X 〮 ( + Y))

= 〮 Y + X 〮 1 <- 합의 법칙( + Y = 1)

=  〮 Y + X <- 곱의 법칙(X 〮 1 = X)

= X + 〮 Y <- 교환 법칙(X + Y = Y + X)

= X + Y <- 흡수 법칙(X + 〮 Y = X + Y)

X + X 〮 Y + 〮 Y

= X + Y(X + ) <- 분배 법칙(X 〮 Y + 〮 Y = Y 〮(X + ))

= X + Y 〮 1 <- 합의 법칙(X + = 1)

= X + Y <- 곱의 법칙(Y 〮 1 = Y)

X + 〮 Y

= (X + ) 〮 (X + Y) <- 분배 법칙(X + 〮 Y = (X + ) 〮 (X + Y))

= 1 〮 (X+Y) <- 합의 법칙( + X = 1)

= X + Y <- 곱의 법칙(X 〮 1 = X)

Point2 기본 논리 회로

★★게이트

* 2진 정보를 처리하기 위한 논리 회로의 기본 소자
* 입력 논리의 필요 조건을 만족하는 결과(0 or 1)를 산출
* 컴퓨터 하드웨어의 기본 소자

AND 게이트

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 의미 | 그리고, 논리곱  두 개의 입력값이 모두 1일 때만 출력값이 1이 됨 | | | | |
| 게이트 | | 논리식 | 진리표 | | |
| A  S  B | | S = A ∩ B  = A 〮B | A | B | S |
| 0 | 0 | 0 |
| 0 | 1 | 0 |
| 1 | 0 | 0 |
| 1 | 1 | 1 |

OR 게이트

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 의미 | 또는, 논리합  두 개의 입력값 중 하나 이상이 1이면 출력값이 1이 됨 | | | | |
| 게이트 | | 논리식 | 진리표 | | |
| A  S  B | | S = A ∪ B  = A + B | A | B | S |
| 0 | 0 | 0 |
| 0 | 1 | 1 |
| 1 | 0 | 1 |
| 1 | 1 | 1 |

NOT 게이트(= Inverter)

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 의미 | 아니다, 않다, 부정을 만드는 논리 연산자  입력값의 반대값이 출력 | | | |
| 게이트 | | 논리식 | 진리표 | |
| A S | | S = = A’ | A | S |
| 0 | 1 |
| 1 | 0 |

Buffer 게이트

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 의미 | 완충 장치, 버퍼(기억), 완충역의 의미로 Delay(지연)의 개념  입력값 그대로 출력 | | | |
| 게이트 | | 논리식 | 진리표 | |
| A S | | S = A | A | S |
| 0 | 0 |
| 1 | 1 |

NAND게이트

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 의미 | NOT + AND, [그리고, 논리곱]의 부정  두 수 중 하나 이상 0이 입력될 때만 1이 출력(AND 결과의 부정) | | | | |
| 게이트 | | 논리식 | 진리표 | | |
| A  S  B | | S =  = +  드모르간 법칙 | A | B | S |
| 0 | 0 | 1 |
| 0 | 1 | 1 |
| 1 | 0 | 1 |
| 1 | 1 | 0 |

NOR 게이트

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 의미 | NOT + OR, [또는, 논리합]의 부정  두 수 모두 0이 입력될 때만 1이 출력(OR결과의 부정) | | | | |
| 게이트 | | 논리식 | 진리표 | | |
| A  S  B | | S =  = 〮  드모르간의 법칙 | A | B | S |
| 0 | 0 | 1 |
| 0 | 1 | 0 |
| 1 | 0 | 0 |
| 1 | 1 | 0 |

XOR 게이트(eXclusive OR)

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 의미 | eXclusive OR, 배타적 논리합  둘 중 하나의 값이 1일 때만(서로 다를 때) 출력값이 1이 됨 | | | | |
| 게이트 | | 논리식 | 진리표 | | |
| A  S  B | | S = 〮 B + A 〮  = A B | A | B | S |
| 0 | 0 | 0 |
| 0 | 1 | 1 |
| 1 | 0 | 1 |
| 1 | 1 | 0 |

XNOR 게이트(= Equivalence)

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 의미 | eXclusive NOR, 배타적 부정 논리합(=동치)  두 수 모두 0 또는 1일 때만(같을 때) 출력값이 1이 됨 | | | | |
| 게이트 | | 논리식 | 진리표 | | |
| A  S  B | | S = · + A · B  = A ⊙ B | A | B | S |
| 0 | 0 | 1 |
| 0 | 1 | 0 |
| 1 | 0 | 0 |
| 1 | 1 | 1 |

금지 회로(Inhibit Circuit)

A

B X

H

* AND 게이트의 여러 입력 중 한 입력을 NOT 회로를 이용하여 금지 입력으로 사용하는 회로
* 금지 입력값이 “1”인 경우 AND 게이트의 출력이 “1”이 될 수 없는 회로

AND, OR 스위치 회로

|  |  |
| --- | --- |
| 직렬과 병렬 이론으로 정리하기 - 전기1 | |
| AND: 두 개의 입력 스위치가 직렬로 연결, 둘다 동시에 ON 상태에서 불이 켜짐(=직렬 회로) | OR: 두 개의 입력 스위치가 병렬로 연결, 둘 중 하나만이라도 ON 상태면 불 켜짐(= 병렬 회로) |