5주차 예비보고서

전공: 컴퓨터공학과 학년: 2학년 학번: 20191619 이름: 이동석

**1. De Morgan**

드 모르간의 법칙은 논리학, 집합과 같은 논리문제에서 합집합, 여집합(‘), 교집합간의 관계를 하나의 법칙으로 정리한 것이다. (과거 고등학교와 중학교 때 배웠던 것 처럼 이는 벤다이어 그램으로 표현이 가능하다.) 마찬가지로 우리가 배우는 논리 회로에도 적용이 가능하다. 드모르간의 법칙은 두 가지가 있다.

먼저 제1 법칙은 논리곱 -> 논리합 이다. 불 대수식으로 으로 쓴다.

제 2법칙은 논리합 -> 논리곱 이다. 불 대수식으로 로 쓴다.

쉽게 생각하면 OR에 NOT(‘) 연산을 하면, AND게이트가 되며 AND게이트의 경우 OR게이트가 된다. 또한, 불 대수의 기본 법칙에 따라 (A’)’ = A 가 된다. 이를 바탕으로 (A+B)’ = A’\* B’ 임을 쉽게 알 수 있다. 다음은 드모르간의 법칙 증명이다. 이는 디지털 회로개론의 강의자료를 참고했다.

테이블이(가) 표시된 사진

자동 생성된 설명

**2. 논리회로 간소화**

논리 회로는 신기하게도 복잡하게 생긴 식과 간단하게 생긴 식이 같은 기능을 할 수 도 있다. 비유를 들자면, 복잡하게 생긴 다항식들을 여러가지 법칙을 사용해 인수분해 하는 것 과 같다.

이런 논리회로의 간소화는 가히 필수라고 말할 수 있다. 논리회로가 복잡하고 사용되는 게이트가 많아질수록 당연하게도 트랜지스터의 수 역시 늘어난다. 이럴 경우, 속도가 느려지며 공간을 많이 차지하게 된다. 요즘 같이 64bit를 사용하는 시기에 사람이 하기엔 거의 불가능하여 논리회로의 간소화를 할 수 있는 Quine-McCluskey 알고리즘이 있다.

불 대수를 통한 논리회로의 간소화도 당연히 가능하다. 하지만, input의 수가 늘어나게 되면 카르노 맵이나 불 대수의 방법은 사용이 어렵다는 단점이 있다. 다음은, 불 대수를 통해 간소화 하는 예제이다.

AB + A’CD + A’BD + A’CD’ + ABCD

= AB + ABCD + A’CD + A’CD’ + A’BD \* 교환 법칙

= AB + AB(CD) + A’C(D + D’) + A’BD \* D + D’ = 1 , 결합법칙

= AB + A’C + A’BD = B(A+A’D) + A’C = B(A+D) + A’C

**3. 카르노 맵**

카르노 맵은 앞서 말했듯, graphic 접근을 통해서 논리회로를 간소화 하기 위한 방법 중 하나이다. 카르노 맵을 사용하면, 복잡한 불 식을 불 대수를 사용하지 않고 간소화가 가능하다. 그러나, 변수가 5개가 넘어 갈 경우 사용이 굉장히 어려워지는 단점이 있으며, 항상 optimal한 (minimum) 솔루션을 찾을 것이라는 보장은 없다.

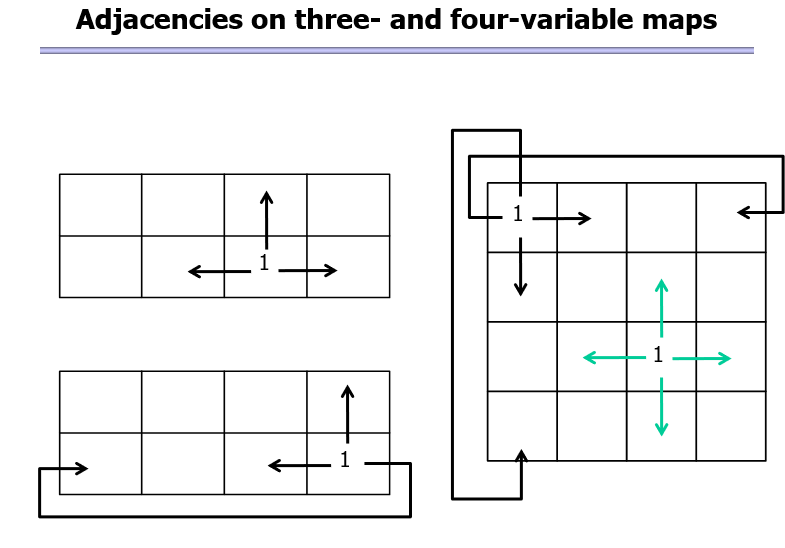
우선 OR게이트 F에 대해 다음과 같은 진리표가 주어졌다고 하자. ( m\_0 ~ 3의 위치는 a와 b를 10진수로 바꿨을 때 숫자이다. 00 = 0, 01 = 1, 10 = 2, 11 = 3 )

테이블이(가) 표시된 사진

자동 생성된 설명 테이블이(가) 표시된 사진

자동 생성된 설명 테이블이(가) 표시된 사진

자동 생성된 설명

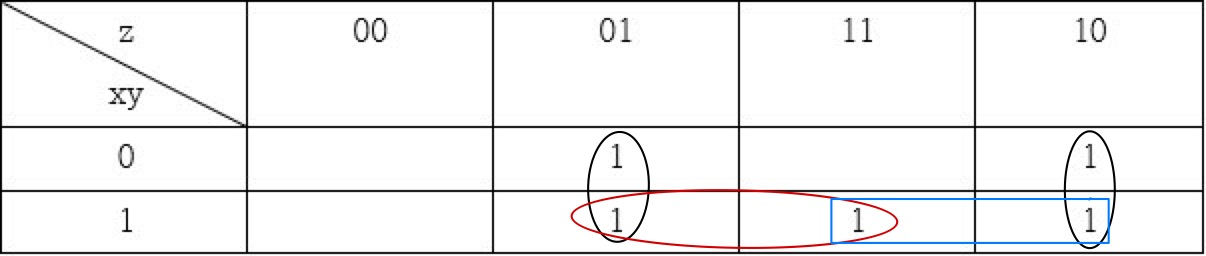
함수 F는 canonical form 중 하나인 minterm을 이용해서 다음과 같이 표기가 가능하다. F = a’b + ab’+ ab 이다. 이제 가장 첫 번째 단계인 Implicant로 부르는 2^k 개의 1의 직사각형 묶음을 모두 찾아야한다. 이때 이러한 Implicant들에서 다른 Implicant들과 결합될 수 없는 것을 Prime Implicant라 부른다. 또한, Prime Implicant들에 속해있는 1들 중에서 자신의 Prime Implicant에 속하는 것을 Essential Prime Implicant라 부른다. 간략화된 함수를 만들 때에는 **E.P.I는 모두 항상 포함**되어야 하며, 일부의 non-essential일부를 포함한다. 

이번 예제에서는 (a’b + ab) 와 (ab’ + ab) 로 묶을 수 있다. 이는 각각 b와 a로 간소화가 가능하며 최종적으로 F = a + b 가 된다.

이번엔 3변수에 대해 생각해자. 함수 F = *x* ′*yz* ′ *+ x* ′*yz + xy* ′*z* ′ *+ xy* ′*z + xyz* 라면, 카르노 맵은 다음과 같이 작성할 수 있다.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| xy z | 00 | 01 | 11 | 10 |
| 0 |  | 1 |  | 1 |
| 1 |  | 1 | 1 | 1 |

이제 Implicant를 모두 찾게 되면,



다음 2가지 경우가 나오게 되는데 둘 중 어떤 것을 사용해도 상관은 없다.

빨간색을 선택할 경우, F = x’y + xy’ + yz가 되며 파란색을 선택할 경우, F = x’y + xy’+ xz가 된다. 이유는 x’yz’ + x’yz = x’y 가 되고 xy’z’ + xy’z = xy’ 가 된다. 빨간색 부분의 경우 x’yz + xyz = yz , 파란색 부분의 경우 xyz + xy’z = xz 이기 때문이다.

Minterm뿐 아니라 maxterm을 사용해서도 가능하다. 또한 무관항(X)에 대해서는 필요한 경우에만 사용하면 된다.

**4. Quine-McCluskey**

이 알고리즘은 변수가 5개 이상이 되어 카르노 맵을 사용하기 어려울 때 사용한다.비슷한 결과를 도출하지만, 컴퓨터에 알고리즘을 구현하기 편한 방식으로 동작한다. 방법은 크게 PI 식별과 / 선택으로 구성된다.

PI 식별 단계에서는 카르노 맵과 마찬가지로 모든 Implicant를 찾는다. 우선 다음과 같은 예제가 있다고 하자. f(A,B,C,D) = sigma m(0,1,3,7,8,9,11,15)

우선, 0~15에 해당하는 모든 이진수를 1의 수에 따라 그룹핑 시킨다.

|  |  |  |
| --- | --- | --- |
| Group | Minterm | Binary |
| 0 | M0 | 0000 |
| 1 | M1,M8 | 0001,1000 |
| 2 | M3.M9 | 0011,1001 |
| 3 | M7,M11 | 0111,1011 |
| 4 | M15 | 1111 |

이제, Matced Pair의 개념을 사용한다. ( ex 0000과 0001은 한자리만 다르며 이를 일치 쌍이라 한다.) 이 개념을 그룹 n과 n+1을 선택해 비교한다. ( 0과 1, 1과 2 ..)

이런식으로 비교한 후 다음과 같은 표를 작성 가능하다.

|  |  |  |
| --- | --- | --- |
| Group | Matched Pairs | Binary |
| 0 | (0,1), (0,8) | 000-,-000 |
| 1 | (1,3),(1,9),(8,9) | 00-1,-0001,100- |
| 2 | (3,7),(3,11),(9,11) | 0-11,-011,100- |
| 3 | (7,15),(11,15) | -111,1-11 |

이를 더 이상 결합할 수 없을 때 까지 반복한다. 최종적 표를 얻을 수 있으며 Minimized form을 얻을 수 있다. 또한, 이 과정에서 결합이 한번도 이루어지지 않은 항을 선택하면 Prime Implicant를 구할 수 있다.

테이블이(가) 표시된 사진

자동 생성된 설명

이제 선택 단계이다. 다음표에서 X가 한번만 나타난 항을 확인한다.

테이블이(가) 표시된 사진

자동 생성된 설명

따라서 최종 간소화된 함수 f(A,B,C,D) = B’C’ + CD 가 됨을 알 수 있다. ( 0,1,3,7,8,9,11,15 가 모두 포함되어 있음.)

그러나, 변수의개수 n이 증가하면 주내포항의 개수는 3^n/ n으로 증가하기 때문에 알고리즘 수행시간과 메모리는 기하급수적으로 증가하여 효율적이지 못한 단점을 갖고있다. (이상운. (2014). 회로 최소화를 위한 개선된 Quine-McCluskey 알고리즘. 한국컴퓨터정보학회논문지 , 19(3), 109-117.)

**5. 기타**

Minterm은 모든 변수가 항상 한번씩 사용된 product term을 말하며 Maxterm은 모든 변수가 항상 한번씩 사용된 sum term을 말한다. 예로, 변수 X,Y,Z가 있을 때 Minterm은 X’YZ, XYZ’ 등이며 Maxterm은 X+Y+Z, X’+Y+Z’ 등이다.

테이블이(가) 표시된 사진

자동 생성된 설명

또한, 모든 불 함수는 1-minterms의 합으로 표현 가능하며, 0-maxterms의 곱으로 표현 가능하다. 이를 Canonical From이라 한다. 또한, minterm과 maxterm은 서로 보수의 관계를 가진다.

종종 Quine-McCluskey 알고리즘을 사용 한 후 후보항들이 모든 최소항을 표현하지 못할 수 있다. 이럴 경우 Petrick’s Method를 사용하면 추가 과정을 통해 항을 구할 수 있다.

**6. 참고문헌**

디지털회로개론 강의자료 chap3 The Karnaugh Map. 김주호 교수

디지털회로개론 강의자료 chap2 combinational systems 김주호 교수

<https://www.techtarget.com/whatis/definition/Karnaugh-map-K-map>

<https://www.watelectronics.com/quine-mccluskey-method/>