**5주차 예비보고서**

전공: 아트&테크놀로지학과 학년: 4학년 학번: 20191048 이름: 김도솔

**1.**

**De Morgan의 정리**는 논리 대수와 집합 이론에서 중요한 개념 중 하나로, 논리합, 논리곱, 부정 연산 간의 관계를 설명하는 개념이다. 수학자 오거스터스 드 모르간의 이름을 따 드 모르간 정리라고 부른다. 좀 더 세부적으로는 논리 연산자의 부정과 논리 연산자의 분배에 관한 원리를 설명하는데, 식 전체에 NOT을 취하는 경우, 이를 전개할 때 각 변수에 NOT을 취하고 OR은 AND 연산으로, AND는 OR 연산으로 전환하면 된다. De Morgan의 정리는 논리식을 다룰 때 유용하며, 복잡한 논리식을 더 간단하게 표현할 수 있도록 도와준다. 또한 컴퓨터 과학 및 디지털 논리 설계와 같은 분야에서 논리 회로를 최적화하는 데 사용된다.

De Morgan의 정리의 두 가지 형태는 다음과 같다:

1)

2)

**2.**

논리 회로는 불 대수를 물리적 디바이스에 구현한 것이기 때문에 복잡한 논리식을 더 간단한 형태로 표현하면 회로의 크기와 복잡도를 줄일 수 있다. 또한 동일한 논리 동작을 수행하는 더 효율적인 회로를 설계하는 데 도움이 된다. **논리회로의 간소화** 방법에는 여러가지가 있는데, 대표적으로 불 대수 법칙을 이용한 간소화, 카르노 맵을 이용한 간소화, 퀸-맥클러스키 방법 등이 있다. 카르노맵과 퀸-맥클러스키 방법은 3, 4 문항에서 다룰 것이기 때문에 2번 문항에서는 불 대수 법칙을 이용한 간소화에 대해 자세히 알아보겠다. 우선 불 대수의 기본 정리는 다음과 같다.

|  |  |  |
| --- | --- | --- |
| 항등 법칙 | (a) x+0=x | (b) x\*1=x |
| 보수 법칙 | (a) x+x’=1 | (b) x\*x’=0 |
| 등역 법칙 | (a) x+x=x | (b) x\*x=x |
| 경계 법칙 | (a) x+1=1 | (b) x\*0=0 |
| 대합 법칙 | (a) (x’)’=x |  |
| 교환 법칙 | (a) x+y=y+x | (b) xy=yx |
| 연관 법칙 | (a) x+(y+z)=(x+y)+z | (b) x(yz)=(xy)z |
| 분배 법칙 | (a) x(y+z)=xy+xz | (b) x+yz=(x+y)(x+z) |
| 드모르간 법칙 | (a) (x+y)’=x’y’ | (b) (xy)’=x’+y’ |
| 흡수 법칙 | (a) x+xy=x(1+y)=x\*1=x | (b) x(x+y)=x |

위의 불 대수 법칙을 이용해 논리식을 단순화 할 수 있다. **예를 들어** F = X’Y + XYZ + XYZ’라는 식은 분배 법칙에 의해 F = X’Y + XY(Z + Z’)로 간소화 하고, 이때 Z + Z’는 보수 법칙에 의해 1이므로 F = XY + X’Y로 나타낼 수 있다. 다음으로 분배 법칙을 한 번 더 적용하면 F = (X + X’)Y, 여기에 보수 법칙을 한번 더 적용하면 X + X’ = 1이 되어 F = Y로 식을 간소화할 수 있다.

**3.**

**카르노 맵**(Karnaugh Map 또는 K-Map)은 논리 회로를 시각적으로 표현하고 논리식을 단순화하는 데 사용되는 방법이다. 카르노 맵은 논리식을 표로 나타내며, 주어진 입력 변수 조합에 따른 출력 값을 보여준다. 카르노 맵을 이용해 논리식을 간소화하는 순서는 다음과 같다.

1) 논리식의 변수의 개수(n)를 파악하고 2^n개의 테이블을 그린다.

2) 변수의 조합에 따라 카르노맵의 빈칸을 0 또는 1로 채워준다.

3) 규칙에 따라 묶을 수 있는 것끼리 묶어준다.

4) 묶인 값을 논리식으로 다시 표현한다.

아래 **예시**(2-변수 카르노 맵)를 통해 위의 방법을 좀 더 자세히 알아보도록 하자. 논리식은 F = A’B’+AB’라고 가정한다. 논리식이 주어지면 다음과 같이 기본 테이블을 그려준다.

도표, 텍스트, 평면도, 기술 도면이(가) 표시된 사진

자동 생성된 설명

이때 각 칸 안의 식은 변수의 조합을 나타낸다. A가 0이고 B가 0일 때 출력이 1이라면 첫 번째 칸을 1로 채우면 된다. 원래 카르노 맵의 각 칸은 비워진 상태로 그려야 하며 이해를 돕기 위해 식을 채워넣었다. 다음으로 입력 A, B 그리고 출력 F에 대해 진리표를 작성한다. 주어진 논리식의 진리표는 다음과 같다.

도표, 텍스트, 평면도, 기술 도면이(가) 표시된 사진

자동 생성된 설명

위의 진리표를 가지고 맵을 채운 결과는 다음과 같다. A, B 순서대로 00인 자리와 10인 자리에 1을 채워주고 나머지 칸은 0으로 채운다.

도표, 텍스트, 평면도, 기술 도면이(가) 표시된 사진

자동 생성된 설명

위의 테이블에서 1로 채워진 인접한 두 칸을 묶어준다. 이때 묶는 칸의 수는 2의 제곱(2, 4, 8 …) 수여야 한다.

도표, 텍스트, 평면도, 기술 도면이(가) 표시된 사진

자동 생성된 설명

마지막으로 묶인 칸을 보면 B’는 고정이고 A와 A’는 변하는 값이라는 것을 알 수 있다. 이때 고정이라는 것은 그 값이 남아야 한다 것이고, 변하는 것은 어느 쪽이든 상관이 없다는 것을 의미한다. 따라서 고정된 값인 B’만 남겨 F = B’로 식을 간단하게 정리할 수 있다.

이러한 방법은 다변수 카르노맵에도 동일하게 적용되어 변수가 늘어나더라도 이 원리를 이용해 논리 회로를 간소화할 수 있다.

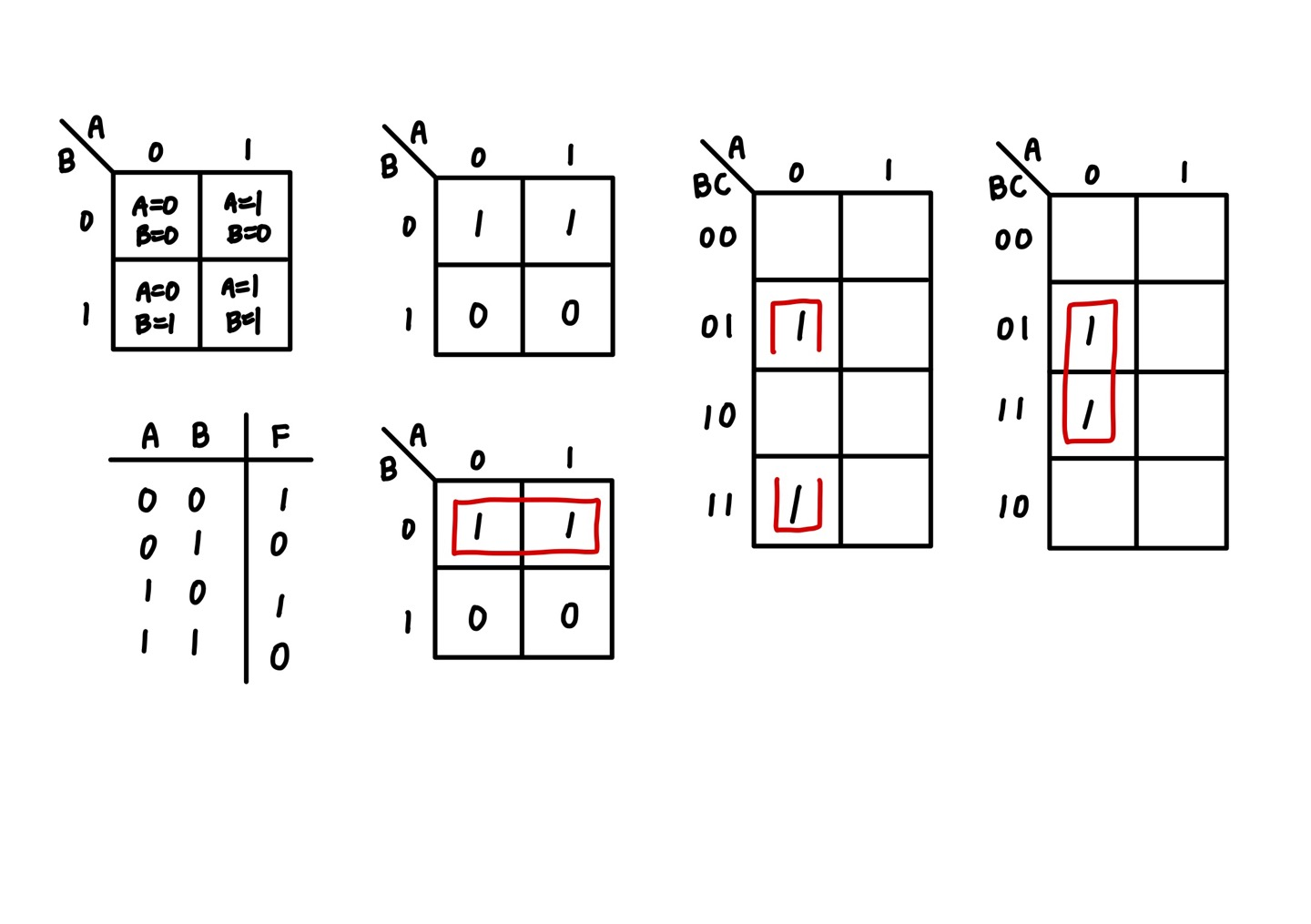
**4.**

**Quine-McCluskey 최소화 알고리즘**은 불 대수를 사용하여 논리식을 최소화하고, 최적화된 논리 회로를 설계하는 데 사용되는 방법이다. 내부적으로는 카르노 맵과 같지만, 직접 그림을 그리는 카르노 맵과는 달리 표를 사용하기에 컴퓨터 상에서 돌릴 수 있다. 따라서 카르노 맵은 변수가 약 5개 이상이 되면 작성하기가 어려워지는 단점이 있지만 QM 알고리즘은 도표를 이용하기에 변수의 개수가 많아도 적용할 수 있다. 또한 카르노 맵은 논리식의 최소 형태를 보장하지 못하지만, QM 알고리즘은 논리식의 최소 형태를 보장한다.

Quine-McCluskey 알고리즘의 초반 단계는 카르노 맵과 유사하다. 논리식과 진리표를 보고 도표를 채운 후에는 크게 두 단계로 구성되는데, 먼저 주어진 논리식에서 Prime Implicant를 모두 찾는다. Prime Implicant는 상하좌우 어느 방향으로든 더 크게 확장할 수 없고, 다른 Implicant에 완전히 포함되지 않는 Implicant를 말한다. 다음으로 찾은 PI 중에서 Essential Prime Implicant를 추출한다. Essential Prime Implicant란 다른 PI로 덮을 수 없는, 즉 자신에게만 속하는 요소(셀)를 적어도 하나 가지는 Prime Implicant를 말한다. 마지막으로 구한 EPI에 따라 논리식을 간소화하여 표현하면 모든 단계가 끝난다. Quine-McCluskey 알고리즘은 논리식을 수학적으로 분석하고 모든 가능한 Implicants를 생성할 수 있으며, 이 중에서 가장 필수적인 부분을 찾아내어 최소 형태의 논리식을 생성한다. 이를 통해 최적의 논리 회로를 설계하고, 불필요한 논리 게이트를 제거하여 회로의 성능과 효율성을 향상시킬 수 있다.

**5.**

**그레이 코드(Gray Code)**는 이진수의 한 형태로서, 연속된 숫자 간의 변환 시 한 비트만 변경되는 특징을 가지는 이진 코드이다. 카르노 맵을 작성하는 경우 변수의 배열을 나타내기 위해 바이너리 코드 대신 그레이 코드를 사용하는데, 그 이유는 그레이 코드로 인코딩된 입력을 사용하면 카르노 맵에서 인접한 셀 간에 한 비트만 변경되므로, 카르노 맵을 효과적으로 사용하여 논리식을 간소화할 수 있기 때문이다. 예를 들어 논리식 F = B’C + BC가 주어졌다고 하자. 이를 각각 바이너리 코드와 그레이 코드로 나타낸 카르노 맵은 다음과 같다.



왼쪽의 바이너리 코드로 나타낸 카르노 맵의 경우 B’C + BC를 C로 묶을 수 있지만 맵 상에서는 떨어져 있어 하나의 Prime Implicant로 묶어내기 어렵다. 반면 오른쪽의 그레이 코드로 나타낸 카르노 맵은 인접한 셀끼리 한 비트씩 차이나기에 B’C + BC를 하나의 Prime Implicant로 묶어내기 간편하다는 장점이 있다