7주차 예비보고서

전공 : 국제한국학과 학년 : 4학년 학번 : 20181202 이름 : 김수미

**1. Parity Bit 생성기 에 대해 조사하시오.**

Parity Bit는 데이터를 전송하는 과정에서 오류가 발생했는지의 여부를 검사하기 위해 사용되는 비트이다. 전체 데이터 비트에 1 비트를 추가함으로써 구현할 수 있고 주로 전달하고자 하는 데이터의 비트 크기가 작고 에러 발생 확률이 낮을 때 사용된다.  
Parity Bit는 짝수 Parity Bit와 홀수 Parity Bit 두 가지 종류가 있다. 짝수 Parity Bit는 전체 데이터에서(비트) 1의 개수가 짝수가 되도록 1 비트를 추가한다. 홀수 Parity Bit 는 전체 데이터에서(비트) 1의 개수가 홀수가 되도록 1 비트를 추가한다.  
예를 들어, 1010010 이라는 7 비트짜리 데이터를 전송한다고 할 때, 현재 1의 개수는 총 3개 이다. 해당 데이터의 맨 앞에 1bit 짜리 Parity Bit가 붙는다고 할때, 1을 붙여 11010010의 8 비트 데이터를 만들면 짝수 Parity Bit가 되고, 맨 앞에 0을 붙여 01010010의 데이터를 만들면 홀수 Parity Bit가 되는 것이다.

Parity Bit이런 식으로 데이터에 Parity Bit를 추가하는 역할을 하는 것을 Parity Bit 생성기라고 한다. XOR 게이트를 사용하면 현재 데이터에서(비트) 1 개수의 홀짝 여부를 확인할 수 있기 때문에, Parity Bit 생성기는 이러한 XOR 게이트의 특성을 활용한다.  
XOR 게이트는 데이터의(비트)1의 개수가 이 짝수이면 0, 홀수이면 1을 출력한다. 이는 짝수 Parity 와 일치한다. 홀수 Parity 를 구현하기 위해서는 XOR 게이트에 NOT 게이트를 추가하면 된다.

**2. Parity Bit 검사기 에 대해 조사하시오.(검사 부호 종류 포함)**

전달받은 데이터의 Parity Bit를 검사해 데이터에 오류가 발생하지 않았는지를 체크하는 회로를 Parity Bit 검사기(Parity checker)라고 한다..

Parity 검사 부호는 짝수와 홀수 Parity Bit로 구별된다. 홀수 Parity Bit는 데이터의(비트) 1의 개수가 홀수이면, 짝수 Parity Bit는 데이터의(비트) 1의 개수가 짝수이면 오류가 발생하지 않았다고 간주한다.  
**ASCII 짝수 Parity 부호** : 정보를 저장하는 정보 비트는 7 비트, Parity Bit는 1비트로 총 8개의 비트로 표현되는 Parity 부호이다. Error Detection은 가능하지만, Error Correction은 불가능하다.  
**Hamming 부호** : 1 Bit Error Correction이 가능한 Parity 부호이다. Hamming 부호를 이 용해 4 비트 데이터(Data Bit)(a3, a5, a6, a7)를 전송한다고 가정하자. 이때, 3개의 Parity Bit(Check Bit)를 추가할 수 있는데 이를 a1, a2, a4라고 할 때,  
a1 = a3^a5^a7 (^ = XOR) / a2 = a3^a6^a7 / a4 = a5^a6^a7 와 같이 나타낼 수 있다.  
이 때 e1 = a3^a5^a7^a1, e2 = a3^a6^a7^a2, e4 = a5^a6^a7^a4 가 되며, 이를 이용하여 Error를 Detection 및 Correction이 가능하다. 4(e4) + 2(e2) + (e1) 를 계산한 결과값이 Error 가 발생한 Bit의 자리이므로, 해당 자리의 Bit를 invert 시켜주면 Error Correction이 가능하다.

**3. Parity Bit 검사기 외의 다른 오류 검출기 및 오류 정정기를 조사하시오.**

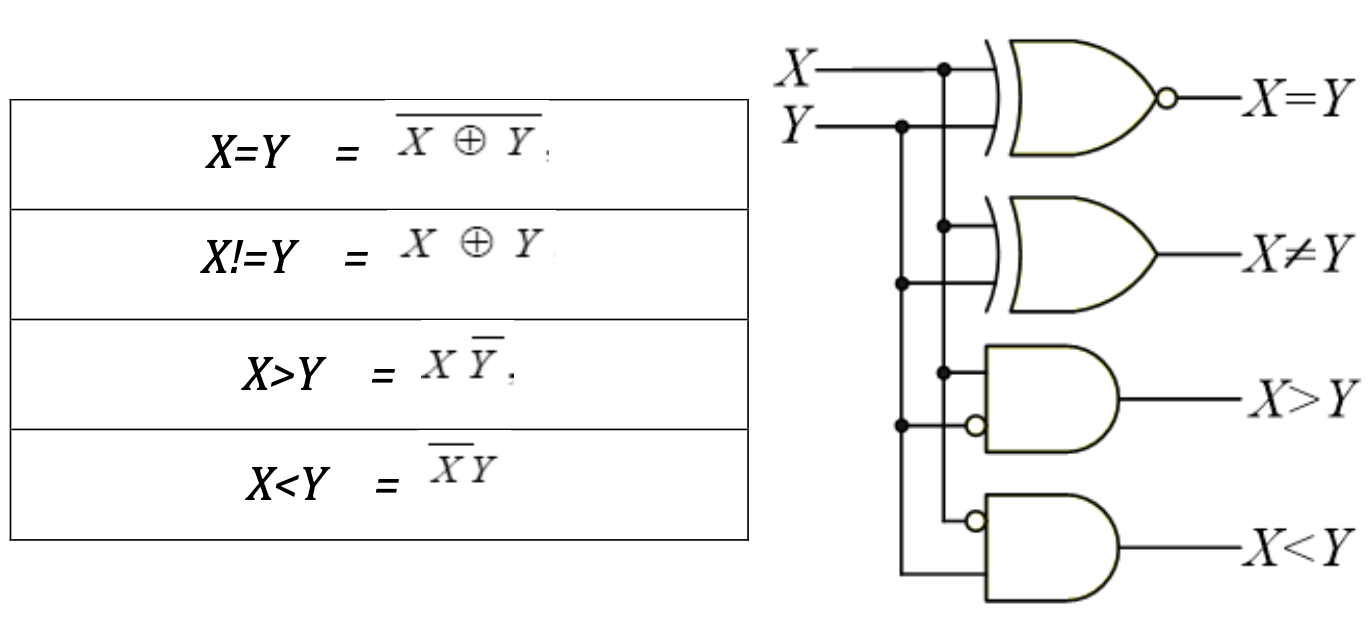
**Checksum** : 모든 데이터를 16 비트로 구분하며, 16 비트 데이터에 1의 보수를 취한 결과값의 합을 데이터와 함께 전송하면 수신 측에서도 똑같이 전달받은 데이터의 합을 계산해 오류가 발생했는지의 여부를 확인하는 방식이다. 간단하여 사용이 쉽지만 데이터의 순서가 바뀌는 오류는 검출하지 못한다는 단점이 있다.

**Cyclic Redundancy Checking(CRC)** : 에러 Detection 능력이 우수한 오류 검출기이다. 데이터를 전송할 때, 특정 다항식으로 데이터를 나눈 결과를 데이터와 함께 전송하면 수신 측 에서도 전달받은 데이터에 대해 동일한 연산을 수행해 그 결과를 비교함으로써 에러를 확인하는 방식이다. 에러 검출 능력이 우수하면서 하드웨어로 구현하기 쉽다는 장점이 있다.

**Reed-Solomon Code(RS Code)** : 에러 Correction 능력이 매우 우수한 오류 정정기이다. 해밍 코드의 비-이진 형태로 각종 디지털 통신 시스템과 TV 등에서 광범위하게 사용된다. 다른 방식에 비해 Decoding이 간편하다는 장점이 있다.

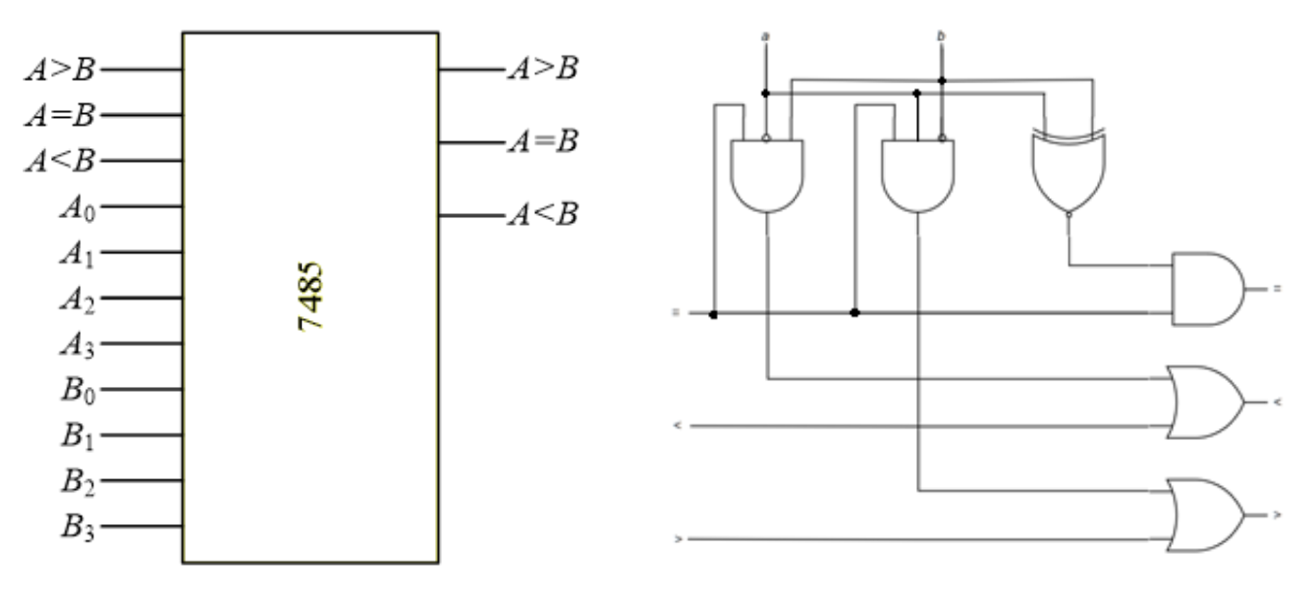
**4. N bit 비교기 에 대해서 조사하시오.**

N bit 비교기란, 두 개의 N 비트 이진수의 크기를 비교하는 회로이다. N 개의 비트 2 개, 총 2N개의 입력을 가지며, A=B, A!=B, A>B, A<B 연산의 결과값을 출력한다. 높은 자리의 비트부터 비교해 대소를 출력하고, 대소 비교 결과가 동일하다면 같다면 다음 비트를 비교하는 방식으로 연산을 수행한다.



**5. IC 7485 비교기 에 대하여 조사하시오.**

IC 7485 비교기는 A3-A0 와 B3-B0 의 크기를 비교하는 회로이다. 4 비트짜리 두 수를 비교해 A>B 인 경우 AGBO 의 출력이 1, A<B 인 경우 ALBO 의 출력이 1, A=B 일 때 AEBO 의 출력이 1이 되며 이 출력은 다시 입력된다. 상위 비트 비교 결과가 A<B 나 A>B 였다면 그 값을 다시 출력하고, 두 값이 같았다면, 하위 비트의 비교 값에 따라 출력값이 결정된다. 이 과정을 최하위 비트에 도달할때까지 반복하여 비교연산을 수행한다.



**6. 기타이론**

- **Gray Code** : Error Detection에 사용되는 Code이다. Number 0 부터 15까지에 대응되는 4bit 짜리 Code를 가지며, 각 Code는 연속적으로 1bit 씩만 차이가 난다. 이렇게 1bit 씩 차이나는 Code를 Counter라고 부르며, Stack이나 Queue같이 Continuous한 device 구현에 주로 사용된다.

