

## 14. 해밍코드 - 오류 검출 및 교정 코드

- 해밍코드는 원래의 데이터(실제 정보)에 여러 개의 패리티 비트를 추가하여 만든다.
- 추가된 패리티 비트를 이용하여 데이터 오류 검출 및 교정까지 가능하다.
- 해밍코드는 가변길이 코드이다. 실제 정보 크기에 따라 추가되는 비트 수가 다르다.

### ● 해밍코드에서 패리티 비트 위치 및 값 구하기(상세한 것은 뒤에서 설명)

먼저, 해밍코드에서 패리티 비트 위치는  $2^p$  ( $p = 0, 1, 2, 3, 4, 5, \dots$ )이 되는 위치이다. 즉, 다음 예제에서  $P_1, P_2, P_4, P_8 \dots$  등이 패리티 비트 위치가 된다.

- $P_1$  위치의 패리티 비트 값은  
1, 3, 5, 7, 9, 11, 13, 15, ... 자리의 1의 개수 합이 짝수 또는 홀수가 되도록 한다.
- $P_2$  위치의 패리티 비트 값은  
2, 3, 6, 7, 10, 11, 14, 15, ... 자리의 1의 개수 합이 짝수 또는 홀수가 되도록 한다.
- $P_4$  위치의 패리티 비트 값은  
4, 5, 6, 7, 12, 13, 14, 15, ... 자리의 1의 개수 합이 짝수 또는 홀수가 되도록 한다.



### 연습

BCD 코드 "1000"에 대한 홀수 및 짝수 패리티 방식의 해밍코드는?

비트 위치	1	2	3	4	5	6	7	
해밍코드 구조	$P_1$	$P_2$	D1	$P_4$	D2	D3	D4	→ $P_i$ 는 패리티 비트 위치
실제 정보(4비트)			1		0	0	0	→ BCD 코드(4비트)
홀수 해밍코드	0	0	1	1	0	0	0	→ 패리티 비트 3개 추가 됨
짝수 해밍코드	1	1	1	0	0	0	0	→ 패리티 비트 3개 추가 됨

- 홀수 해밍코드와 짝수 해밍코드의 패리티 비트 위치의 값은 서로 다르다.



예제

[예제 1] BCD 코드 '1000' 에 대한 **홀수** 패리티 방식의 해밍코드는?

비트 위치	1	2	3	4	5	6	7	
패리티 비트 위치	P <sub>1</sub>	P <sub>2</sub>		P <sub>4</sub>				
실제 정보(4비트)			1		0	0	0	→ BCD 코드(4비트)
패리티 비트 값	0	0		1				→ 추가된 비트
전송하는 정보	0	0	1	1	0	0	0	→ 해밍코드

- 전송 전에 원래의 데이터(실제 정보)에 여러 개의 패리티 비트를 추가한 것이 해밍코드이다.
- 해밍코드에서 실제 정보(데이터 비트)가 4비트이면 패리티 비트는 3개가 필요하고
- 해밍코드에서 실제 정보(데이터 비트)가 8비트이면 패리티 비트는 4개가 필요하다.
- 해밍코드는 휴대폰에서도 사용된다.

[예제 2] 8비트 크기의 실제 정보에서 **짝수** 방식의 해밍코드로 전송하는 경우

비트 위치	1	2	3	4	5	6	7	8	9	10	11	12
실제 정보(8비트)			1		0	0	1		1	0	1	1
전송하는 정보	0	1	1	0	0	0	1	1	1	0	1	1
패리티 비트	P <sub>1</sub>	P <sub>2</sub>		P <sub>4</sub>				P <sub>8</sub>				

[예제 3] 홀수 패리티 비트를 가진 해밍코드를 수신하였다. 오류가 있으면 수정하여라.

비트 위치	1	2	3	4	5	6	7
해밍코드	1	1	1	1	0	0	1
패리티 비트	P <sub>1</sub>	P <sub>2</sub>		P <sub>4</sub>			

- [풀이] P<sub>1</sub>은 1, 3, 5, 7 비트 위치를 검사 : 1이 3개 → 정상  
 P<sub>2</sub>는 2, 3, 6, 7 비트 위치를 검사 : 1이 3개 → 정상  
 P<sub>4</sub>는 4, 5, 6, 7 비트 위치를 검사 : 1이 2개 → 비정상(오류 발생)

따라서, P<sub>4</sub>의 패리티 값은 0이어야 한다. 수정된 해밍코드 : 1110001

[Tip] 해밍코드에서는 패리티 비트 위치와 값을 정하는 개념이 중요하다.



탐구

해밍코드에서 패리티 비트 값 구하는 원리

• 왜? 해밍코드에서

$P_1$ 의 값은 1, 3, 5, 7,... 위치를 검사하여 1의 개수가 짝수 또는 홀수가 되도록 하는가?

이를 설명하기 위해 다음은 10진수를 4자리 형태의 2진수로 나타낸 것이다.

10진수	2진수	
1	0 0 0 1	• 먼저, 2진수에서 0과 1이 배치된 형태를 잘 살펴보고 • 다음을 이해하면 문제의 해답을 찾을 수 있다. ① $P_1$ 은 2진수 끝자리가 1인 10진수 1, 3, 5, 7, 9, 11, 13, 15 → 하나의 수를 건너감
2	0 0 1 0	
3	0 0 1 1	
4	0 1 0 0	
5	0 1 0 1	
6	0 1 1 0	② $P_2$ 는 2진수 끝에서 두 번째 자리가 1인 10진수 2, 3, 6, 7, 10, 11, 14, 15 → '두 수를 연속해서 취하고, 두 수를 건너감'을 반복
7	0 1 1 1	
8	1 0 0 0	
9	1 0 0 1	③ $P_4$ 는 2진수 끝에서 세 번째 자리가 1인 10진수 4, 5, 6, 7, 12, 13, 14, 15 → '네 수를 연속해서 취하고, 네 수를 건너감'을 반복
10	1 0 1 0	
11	1 0 1 1	④ $P_8$ 은 2진수 끝에서 네 번째 자리가 1인 10진수 8, 9, 10, 11, 12, 13, 14, 15 → '8개의 수를 연속해서 취하고, 8개의 수를 건너감'을 반복
12	1 1 0 0	
13	1 1 0 1	
14	1 1 1 0	
15	1 1 1 1	

결론	$P_1$ 은 1이므로 1부터 시작되고, 1개씩 처리하고, 1개씩 건너가고, $P_2$ 는 2이므로 2부터 시작되고, 2개씩 처리하고, 2개씩 건너가고, $P_4$ 는 4이므로 4부터 시작되고, 4개씩 처리하고, 4개씩 건너가는 원리이다.
----	---

**기출문제 분석**

1. 짝수 패리티를 갖는 7비트 데이터 '0011111'이 수신되었다. 해밍코드를 이용하여 전송 중 발생한 오류를 찾아 바르게 정정한 것은? [2009년 지방 9급]

- ① 0011100                      ② 0011011
- ③ 0001111                      ④ 0010111

☞ 짝수 패리티를 가지는 해밍코드

비트 위치	1	2	3	4	5	6	7
수신된 해밍코드	0	0	1	1	1	1	1
데이터 비트 위치			D1		D2	D3	D4
패리티 비트 위치	p1	p2		p4			

- p1은 1, 3, 5, 7 비트 위치의 1의 개수를 검사 : 1이 3개 → 비정상(오류 발생)
- p2는 2, 3, 6, 7 비트 위치의 1의 개수를 검사 : 1이 3개 → 비정상(오류 발생)
- p4는 4, 5, 6, 7 비트 위치의 1의 개수를 검사 : 1이 4개 → 정상

// 결론

- p4는 정상이므로 비트 위치 1, 2, 3에서 오류가 발생된 것을 유추할 수 있다.
- 따라서, p1과 p2가 공유하는 비트 위치 3에서 오류가 발생된 것이다.

수정된 해밍코드 : 0 0 0 1 1 1 1

정답 : ③

2.  $0 \sim (64^{10} - 1)$ 에 해당하는 정수를 이진코드로 표현하기 위해 필요한 최소 비트 수는? [2019년 국가 9급]

- ① 16비트                      ② 60비트
- ③ 63비트                      ④ 64비트

☞ 10진수와 2진수와의 관계

•  $64^{10} = (2^6)^{10} = 2^{60}$ 이므로 60비트가 필요

정답 : ②

3. <보기>는 자료의 표현과 관련된 설명이다. 옳은 것을 모두 고른 것은? [2010년 계리직]

ㄱ. 2진수 0001101의 2의보수(complement)는 1110011이다.  
 ㄴ. 부호화 2의보수 표현방법은 영(0)이 하나만 존재한다.  
 ㄷ. 패리티(parity) 비트로 오류를 수정할 수 있다.  
 ㄹ. 해밍(hamming) 코드로 오류를 검출할 수 있다.

- ① ㄱ, ㄹ                      ② ㄴ, ㄷ
- ③ ㄱ, ㄴ, ㄷ                ④ ㄱ, ㄴ, ㄹ

☞ 패리티 비트 검사

- 패리티 비트 검사 방식은 오류를 수정할 수 없다. 발견만 할 수 있다.
- 패리티 비트는 데이터를 송신하기 전에 원래의 데이터에 추가로 덧붙이는 비트이다.
- 추가되는 비트는 1bit이다.

ㄴ. 부호화 2의보수 표현방법은 영(0)이 하나만 존재한다.(○)  
 → 부호화 1의보수 표현방법은 영(0)이 2개 존재한다. 즉, +0과 -0이 존재한다는 것!

정답 : ④

4. 어떤 시스템은 7비트의 데이터에 홀수 패리티 비트를 최상위 비트에 추가하여 8비트로 표현하여 저장한다. 다음과 같은 데이터를 저장 장치에서 읽어 왔을 때 오류가 발생한 경우는? [2019년 서울 9급]

- ① 011010111                ② 101101111
- ③ 011001110                ④ 101001101

☞ 홀수 패리티

- ① 0110 1011 1 → 1의 개수 : 6개
- ② 1011 0111 1 → 1의 개수 : 7개
- ③ 0110 0111 0 → 1의 개수 : 5개
- ④ 1010 0110 1 → 1의 개수 : 5개

- 문제에서. 7비트 데이터라고 했는데, 7비트가 아니고 8비트로 표현되어 있다.

정답 : ①

5. 네트워크에서 1비트의 패리티 비트(Parity bit)를 사용하여 데이터의 전송 에러를 검출하려 한다. 1바이트 크기의 데이터 A, B, C, D, E 다섯 개를 전송하였다. 그 중 두 개의 데이터에서 1비트 에러가 발생하였고 나머지는 정상적으로 전송되었다고 가정하자. 다음 표에서 에러가 발생한 두 개의 데이터는? [2009년 국가 9급]

데이터 이름	데이터 비트열	패리티 비트
A	01001101	1
B	01110110	1
C	10111000	0
D	11110001	0
E	10101010	0

- ① A, D                      ② B, C                      ③ B, E                      ④ C, E

☞ 패리티 비트

- 패리티 비트는 데이터를 송신하기 전에 원래의 데이터에 추가로 덧붙이는 비트이다.
- 추가되는 비트는 1bit이다.
- 패리티 비트는 데이터 송수신 과정에서 오류 발생 여부를 검사하기 위한 것이다.
- 기수 패리티 검사 : 코드에 1의 개수가 패리티 비트를 포함하여 홀수개가 되도록 구성
- 우수 패리티 검사 : 코드에 1의 개수가 패리티 비트를 포함하여 짝수개가 되도록 구성
- 즉, 운용하는 시스템에 따라 전송하기 전의 코드에 1로 된 비트들 개수의 합이 항상 짝수 또는 홀수가 되도록 추가로 1비트 값을 강제로 덧붙인다.

// 문제에서 각 데이터의 1의 개수(패리티 포함)는 다음과 같다

```

A : 5 → 홀수  ┌
B : 6 → 짝수  │
C : 4 → 짝수  │ 홀수 2개(A와 D), 짝수 3개(B, C, E)
D : 5 → 홀수  │
E : 4 → 짝수  └
    
```

- ↓ 문제에서 전송 시 두 개의 데이터에서 에러가 발생했다는 것은
- ↓
- 1의 개수가 홀수인 A와 D에서 발생된 것임을 유추할 수 있다.