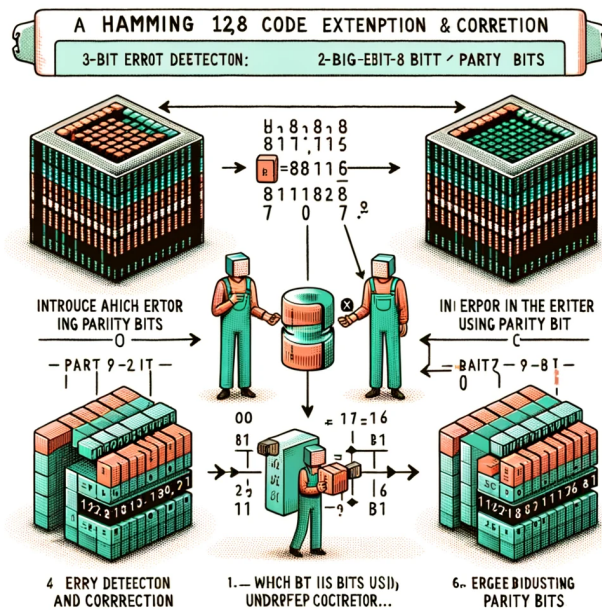


HAMMING ALGORİTMASI (12,8)

Hamming (12,8) kodu, hata tespiti ve düzeltme alanında kullanılan bir yöntemdir. Bu yöntemde, 8 bitlik veriye 4 bit eklenerek toplamda 12 bitlik bir kod oluşturulur. Bu eklenen 4 bit, verinin doğru şekilde iletildiğini doğrulamak ve olası hataları düzeltmek için kullanılır.

Hamming (12,8) Algoritmasının Açıklaması

- **Amacı:** Hata tespiti ve düzeltme.
- **Veri Bitleri:** İlk 8 bit (D1, D2, ..., D8) orijinal veri bitleridir.
- **Parite Bitleri:** Eklenen 4 bit (P1, P2, P4, P8) parite bitleridir. Bu bitler, veri bitlerine bağlı olarak hesaplanır.
- **Parite Bitlerinin Hesaplanması:**
 - **P1:** 1, 3, 5, 7, 9, 11. bitlerin XOR'u (D1, D3, D5, D7 gibi).
 - **P2:** 2, 3, 6, 7, 10, 11. bitlerin XOR'u.
 - **P4:** 4-7 ve 12. bitlerin XOR'u.
 - **P8:** 8-11. bitlerin XOR'u.



Parite Bitlerinin Hesaplanması

P1, P2, P4, P8 Bunlar, belirli veri bitlerinin XOR işlemiyle hesaplanır.

- **P1:** 1, 3, 5, 7, 9, 11. bitlerin XOR'u.
- **P2:** 2, 3, 6, 7, 10, 11. bitlerin XOR'u.
- **P4:** 4, 5, 6, 7, 12. bitlerin XOR'u.
- **P8:** 8, 9, 10, 11. bitlerin XOR'u.

Örnek Hesaplama

Veri Bitleri

- Diyelim ki göndermek istediğimiz veri 1101 0110 olsun. Bu, 8 bitlik veri kısmımızı oluşturur (D1, D2, ..., D8).

Parite Bitlerinin Hesaplanması

Parite bitleri, belirli veri bitlerinin XOR (özel veya) işlemiyle hesaplanır. Hamming (12,8) kodunda 4 parite biti vardır: P1, P2, P4 ve P8. Her birini aşağıdaki gibi hesaplarız:

- **P1:** 1, 3, 5, 7, 9, 11. bitlerin XOR'u.
 - Bu durumda: D1 (1), D3 (0), D5 (0), D7 (1) → $P1 = 1 \text{ XOR } 0 \text{ XOR } 0 \text{ XOR } 1 = 0$
- **P2:** 2, 3, 6, 7, 10, 11. bitlerin XOR'u.
 - Bu durumda: D2 (1), D3 (0), D6 (1), D7 (1) → $P2 = 1 \text{ XOR } 0 \text{ XOR } 1 \text{ XOR } 1 = 1$
- **P4:** 4, 5, 6, 7, 12. bitlerin XOR'u.
 - Bu durumda: D4 (0), D5 (0), D6 (1), D7 (1) → $P4 = 0 \text{ XOR } 0 \text{ XOR } 1 \text{ XOR } 1 = 0$
- **P8:** 8-11. bitlerin XOR'u.
 - Bu durumda: D8 (1), D9 (0), D10 (1), D11 (0) → $P8 = 1 \text{ XOR } 0 \text{ XOR } 1 \text{ XOR } 0 = 0$

Sonuç

- Hesaplanan parite bitleriyle birlikte, 12 bitlik Hamming kodumuz şu şekilde oluşur: **P1 P2 D1 P4 D2 D3 D4 P8 D5 D6 D7 D8**
- Bu durumda kodumuz: 0 1 1 0 1 0 0 0 1 1 0 1

Başarılar ♥