

BİLGİSAYAR MİMARİSİ PROJE ÖDEVİ



23360859035

Esmâ BİLEN

Haziran 2025

1. Giriş

Bu proje, bilgisayar ağları ve iletişim sistemlerinde kullanılan hata tespit ve düzeltme tekniklerinden biri olan Hamming kodlarının simülasyonunu amaçlamaktadır. Hamming SEC-DED kodlama algoritması, tek bitlik hataları otomatik olarak düzeltebilmekte ve çift bit hatalarını tespit edebilmektedir. Bu sayede veri iletiminde güvenilirlik artırılır.

Proje kapsamında, Hamming kodlama algoritması temel alınarak; rastgele tek bit ve çift bit hata oluşturma, tespit etme ve düzeltme işlemleri simüle edilmiştir.

2. Projenin Amacı ve Kapsamı

Bu proje ile amaçlanan;

- Hamming kodlarının çalışma prensiplerini anlamak,
- Tek bitlik hataların nasıl oluşturulup tespit edildiğini göstermek,
- Çift bit hatalarının tespiti ve etkilerini simüle etmek,
- Öğrenilen teorik bilgileri pratikte deneyimlemek,
- Basit bir simülatör aracılığıyla hata kontrol mekanizmalarını görselleştirmektir.

3. Proje Görselleri ve Proje Anlatımı

Simülatör şu adımları takip eder:

1. 32 bitlik veri girişi yapılır.
2. Tek ya da çift bitlik hata rastgele oluşturulur.
3. Hamming kodu hesaplanır.
4. Hata veriye uygulanır.
5. Hata tespiti ve varsa düzeltme işlemi gerçekleştirilir.
6. Sonuç kullanıcıya gösterilir.

Hamming SEC-DED Code Simülatörü

Veri Bit Uzunluğu Seçin:

☒ 8 Bit ☐ 16 Bit ☐ 32 Bit

Veri (sadece 0 ve 1):

Örn: 00011101

Hamming Kodu Oluştur Rastgele Tek Bit Hata Oluştur Rastgele Çift Bit Hata Oluştur

Hata Tespit Et Hata Düzelt

Veri Bitleri Parite Bitleri SEC Bitleri Hatalı Bit

Hamming Kod:

Oluşturulan Hatalı Kod:

Demoya tıkladığımızda bizi bu ekran karşılıyor. Buradan kaç bitlik veri gireceğimizi seçip o boyutta verimizi yazarak hamming kodu oluşturma butonuna tıklıyoruz.

Veri Bit Uzunluğu Seçin:

☒ 8 Bit ☐ 16 Bit ☐ 32 Bit

Veri (sadece 0 ve 1):

10101010

Hamming Kodu Oluştur Rastgele Tek Bit Hata Oluştur Rastgele Çift Bit Hata Oluştur

Hata Tespit Et Hata Düzelt

Veri Bitleri Parite Bitleri SEC Bitleri Hatalı Bit

Hamming Kod:

1	1	0	1	0	0	1	0	1	1	0	0	0
1010	1010	1010	1010	1010	0011	1010	1010	1010	1010	1010	0011	1010

Oluşturulan Hatalı Kod:

Görselde de gözüktüğü gibi hamming kodumuz oluşuyor.

Veri bitleri: mavi

Parite bitleri: sarı

Sec biti: pembe

Veri (sadece 0 ve 1):

10101010

Hamming Kodu Oluştur

Rastgele Tek Bit Hata Oluştur

Rastgele Çift Bit Hata Oluştur

Hata Tespit Et

Hata Düzelt

Veri Bitleri

Parite Bitleri

SEC Bitleri

Hatalı Bit

Hamming Kod:

1

1

0

1

0

0

1

0

1

1

0

0

0

Oluşturulan Hatalı Kod:

1

1

0

1

0

0

1

0

1

1

0

0

1

Tek bit hatalı.Hatalı bitin sendromu: 1

Rastgele bir bit hata oluştur butonuna tıkladığımızda görselde görüldüğü gibi o butonun yanında değiştiğine dair uyarı işareti beliriyor.

Hamming Kod:

1

1

0

1

0

0

1

0

1

1

0

0

0

Oluşturulan Hatalı Kod:

1

1

0

1

0

0

1

0

1

1

0

0

1

Tek bit hatalı.Hatalı bitin sendromu: 1

Sonuçlar:

1.bit düzeltildi.

1

1

0

1

0

0

1

0

1

1

0

0

0

Hata düzelt butonuna tıkladığımızda yukarıdaki görselde gözüktüğü gibi hatalar düzeltiliyor hangi hatanın düzeltildiği daha net olsun diye uyarı işareti yanından hala kalkmıyor.

Benzer işlemler rastgele çift bit hata oluştur butonuna tıklayınca da oluyor ama kritik nokta çift bit hatalar düzeltilmiyor ve düzeltilmek istendiğinde mesaj ekrana basılıyor:



4.Kullanılan Teknolojiler ve Araçlar

- Programlama dili: JavaScript
- Arayüz: HTML/CSS

5. Teknik Detaylar

- **Hata Modeli:** Tek bitlik ve çift bitlik hatalar rastgele bit pozisyonlarında simüle edilmiştir.
- **Düzeltilme Mekanizması:** Tek bitlik hatalar, parite bitlerinin kontrolü sonucunda belirlenip otomatik olarak düzeltilmektedir.
- **Teknolojiler:** Proje HTML ve JavaScript kullanılarak geliştirilmiş olup, görsel arayüzle kullanıcı etkileşimi sağlanmıştır.

6. Kullanıcı Arayüzü ve Fonksiyonlar

Proje arayüzünde kullanıcı;

- Veri girişini yapabilir,
- Tek bit hata oluşturma ve düzeltme işlemini çalıştırabilir,
- Çift bit hata oluşturma ve tespit simülasyonunu izleyebilir,

Daha fazla bilgi için github repomu ziyaret edebilirsiniz:
<https://github.com/esmabilen37/hamming-code-simulator>

Demo linki: <https://esmabilen37.github.io/hamming-code-simulator/>

Youtube videosu: <https://youtu.be/KR6ZeaimbMs>