**T.C**

**KOCAELİ SAĞLIK VE TEKNOLOJİ ÜNİVERSİTESİ**

**MÜHENDİSLİK VE DOĞA BİLİMLERİ FAKÜLTESİ**

**BİLGİSAYAR MÜHENDİSLİĞİ**

**PROGRAMLAMA**

**Hazırlayan**

**MÜSLÜM BATIN KILIÇ**

**220501023**

**DERS SORUMLUSU:**

**DR. ÖĞR. ÜYESİ**

**Ercan ÖLÇER**

**02.06.2024**

**İÇİNDEKİLER**

[1. ÖZET 3](#_Toc149597653)

[2. GİRİŞ 3](#_Toc149597654)

[3. YÖNTEM 4](#_Toc149597655)

[4. SONUÇ VE ÖĞRENİLEN DERSLER](#_Toc149597666) 6

[5. KAYNAKÇA](#_Toc149597667) 7

# ÖZET

Bu proje, çeşitli mantık kapılarının (AND, OR, NOT, Buffer, NAND, NOR, XOR, XNOR) ve giriş/çıkış elemanlarının (Giriş Kutuları, Çıkış Kutuları ve LED'ler) görsel bir arayüz kullanılarak simülasyonunun yapılmasını sağlayan bir sayısal tasarım platformu geliştirmeyi amaçlamaktadır. Tkinter kütüphanesi kullanılarak oluşturulan bu platform, kullanıcıların çeşitli mantık kapılarını ve giriş/çıkış elemanlarını yerleştirip bağlayarak mantıksal devreler oluşturmalarını ve çalıştırmalarını sağlar. Bu proje ile dijital devre tasarımının temelleri öğrenilebilir, mantık kapılarının nasıl çalıştığı ve kombinasyonlarının sonuçları gözlemlenebilir.

# GİRİŞ

**2.1 Projenin Amacı ve Önemi**

Sayısal devre tasarımı, modern elektronik sistemlerin temelini oluşturur. Bilgisayarlar, cep telefonları ve diğer dijital cihazlar, bu temel üzerine inşa edilmiştir. Mantık kapıları, dijital devrelerin yapı taşlarıdır ve çeşitli işlemleri gerçekleştirmek için birleştirilebilirler. Dijital devre tasarımı, mühendislik ve teknoloji öğrencileri için temel bir öğrenme alanıdır. Bu proje, dijital tasarım sürecini görselleştirmek ve kullanıcıların mantık kapılarını ve diğer bileşenleri kullanarak kendi devrelerini oluşturup test edebilmelerini sağlamak amacıyla geliştirilmiştir. Bu sayede öğrenciler ve meraklılar, dijital devrelerin çalışma prensiplerini daha iyi kavrayabilirler.

**2.2 Mantık Kapıları ve Dijital Devreler**

Mantık kapıları, bir veya daha fazla giriş değeri alarak belirli bir mantıksal işleme göre tek bir çıkış değeri üretir. Temel mantık kapıları şunlardır:

* AND Kapısı: Tüm girişler 1 olduğunda çıkışı 1 olan kapıdır.
* OR Kapısı: Girişlerden en az biri 1 olduğunda çıkışı 1 olan kapıdır.
* NOT Kapısı: Girişin tersini veren kapıdır; giriş 1 ise çıkış 0, giriş 0 ise çıkış 1 olur.
* Buffer Kapısı: Girişi doğrudan çıkışa ileten kapıdır.
* NAND Kapısı: AND kapısının tersidir; tüm girişler 1 olduğunda çıkışı 0, diğer durumlarda 1 olur.
* NOR Kapısı: OR kapısının tersidir; tüm girişler 0 olduğunda çıkışı 1, diğer durumlarda 0 olur.
* XOR Kapısı: Girişler farklı olduğunda çıkışı 1 olan kapıdır.
* XNOR Kapısı: XOR kapısının tersidir; girişler aynı olduğunda çıkışı 1 olur.

# YÖNTEM

Projenin geliştirilmesi birkaç adımda gerçekleştirilmiştir. Bu adımlar, geliştirme ortamının seçilmesi, mantık kapılarının ve diğer bileşenlerin tanımlanması, kullanıcı arayüzünün oluşturulması, bağlantıların ve propagasyonun yönetilmesi ve son olarak simülasyonun gerçekleştirilmesini içermektedir.

**3.1 Geliştirme Ortamı ve Araçları**

Projenin geliştirilmesi için Python programlama dili ve Tkinter kütüphanesi kullanılmıştır. Tkinter, Python'un standart grafik kullanıcı arayüzü (GUI) kütüphanesidir ve bu proje için uygun bir seçim olmuştur. Tkinter, kullanıcı arayüzü oluşturmak için gerekli olan tüm bileşenleri sağlamaktadır.

**3.1.1 Python ve Tkinter**

Python: Python, geniş kütüphane desteği ve kolay öğrenilebilirliği ile tercih edilmiştir. Python, çeşitli kütüphaneleri sayesinde hızlı prototip geliştirmeyi sağlar.

Tkinter: Tkinter, Python’un standart GUI kütüphanesi olup, kolay kullanımı ve geniş özellikleri ile kullanıcı arayüzü geliştirme sürecini hızlandırır. Tkinter ile düğmeler, tuval, metin kutuları ve diğer interaktif bileşenler oluşturulabilir.

**3.2 Mantık Kapıları ve Diğer Bileşenler**

Proje kapsamında kullanılan mantık kapıları ve diğer bileşenler aşağıda detaylandırılmıştır:

**3.2.1 Mantık Kapıları**

Mantık kapıları, dijital devrelerin temel bileşenleridir. Her kapı, belirli bir mantık fonksiyonunu gerçekleştirir ve bu fonksiyonlara göre giriş değerlerini işleyerek bir çıkış değeri üretir.

* AND Kapısı (ANDGate): İki giriş alır ve her iki giriş de 1 olduğunda çıkış 1, diğer durumlarda çıkış 0 olur.
* OR Kapısı (ORGate): İki giriş alır ve en az bir giriş 1 olduğunda çıkış 1, her iki giriş de 0 olduğunda çıkış 0 olur.
* NOT Kapısı (NOTGate): Tek bir giriş alır ve bu girişin tersini çıkış olarak verir.
* Buffer Kapısı (BufferGate): Tek bir giriş alır ve bu girişi değiştirmeden çıkışa iletir.
* NAND Kapısı (NANDGate): AND kapısının tersidir; her iki giriş de 1 olduğunda çıkış 0, diğer durumlarda çıkış 1 olur.
* NOR Kapısı (NORGate): OR kapısının tersidir; her iki giriş de 0 olduğunda çıkış 1, diğer durumlarda çıkış 0 olur.
* XOR Kapısı (XORGate): İki giriş alır ve girişler farklı olduğunda çıkış 1, aynı olduğunda çıkış 0 olur.
* XNOR Kapısı (XNORGate): XOR kapısının tersidir; girişler aynı olduğunda çıkış 1, farklı olduğunda çıkış 0 olur.

**3.2.2 Diğer Bileşenler**

Giriş Kutuları (InputBox): Kullanıcıların giriş değerlerini (0 veya 1) belirlemelerini sağlar.

Çıkış Kutuları (OutputBox): Mantık kapılarının hesapladığı sonuçları gösterir.

LED'ler (LED): Çıkış değerine göre yeşil (1) veya kırmızı (0) olarak yanar.

Bağlantı Düğümleri (ConnectionNode): Bileşenler arasındaki bağlantıları temsil eder.

**3.3 Kullanıcı Arayüzü**

Kullanıcı arayüzü, mantık kapılarının ve diğer bileşenlerin eklenebileceği, taşınabileceği ve bağlanabileceği bir tuval sunar. Arayüzdeki kontroller ve düğmeler, kullanıcıların bileşenleri eklemesini ve simülasyonu çalıştırmasını sağlar. Kullanıcı arayüzü, kullanıcı dostu bir tasarıma sahip olup, kullanıcıların bileşenleri kolayca ekleyip bağlayabilmeleri için uygun bir kullanıcı deneyimi sunar.

**3.3.1 Kontrol Paneli**

Kullanıcıların mantık kapılarını ve diğer bileşenleri ekleyebileceği kontrol düğmeleri içerir. Her bir mantık kapısı için bir düğme ve ayrıca giriş kutusu, çıkış kutusu ve LED eklemek için düğmeler bulunur.

**3.3.2 Tuval**

Bileşenlerin yerleştirilebileceği, taşınabileceği ve bağlanabileceği bir alan sağlar. Kullanıcılar, bileşenleri sürükleyip bırakarak tuval üzerinde istedikleri konuma yerleştirebilirler.

**3.3.3 Simülasyon Kontrolleri**

Simülasyonu başlatmak, durdurmak ve sıfırlamak için düğmeler içerir. Bu kontroller, kullanıcıların devrelerini çalıştırarak sonuçları gözlemlemelerine ve gerektiğinde devreyi sıfırlayarak yeni bir tasarıma başlamalarına olanak tanır.

**3.4 Bağlantılar ve Propagasyon**

Bağlantılar, giriş kutularından mantık kapılarına ve mantık kapılarından çıkış kutularına veya LED'lere veri akışını sağlar. Kullanıcılar, bileşenler arasındaki bağlantıları belirleyerek devreyi oluştururlar. Bağlantıların doğru şekilde kurulması ve giriş değerlerinin doğru bileşenlere iletilmesi önemlidir.

**3.4.1 Bağlantı Kurulumu**

Kullanıcılar, bileşenleri tuval üzerinde seçerek ve bağlamak istedikleri bileşene tıklayarak bağlantılar kurabilirler. Bağlantılar, çizgilerle görselleştirilir ve bileşenler arasında veri akışını sağlar.

**3.4.2 Propagasyon**

Giriş kutularına girilen değerler, bağlı oldukları mantık kapılarına iletilir. Mantık kapıları, aldıkları giriş değerlerine göre çıkış değerlerini hesaplar ve bu değerleri bağlı oldukları bileşenlere iletir. Bu süreç, tüm devre boyunca devam eder ve çıkış kutuları veya LED'ler, son değerleri gösterir.

**3.5 Simülasyon ve Çıktı**

Kullanıcılar giriş kutularına değerler girdikten sonra simülasyonu çalıştırarak mantık kapılarının çıkışlarını ve devrenin genel davranışını gözlemleyebilirler. Çıkış kutuları ve LED'ler, devrenin sonuçlarını görselleştirmek için kullanılır. Simülasyon sırasında, giriş değerleri mantık kapılarına iletilir ve kapılar kendi işlevlerine göre çıkış değerlerini hesaplar ve propagasyon gerçekleştirir.

**3.5.1 Giriş Değerlerinin Ayarlanması**

Kullanıcılar giriş kutularına çift tıklayarak değerleri ayarlayabilirler. Giriş değerleri 0 veya 1 olarak ayarlanabilir ve bu değerler mantık kapılarına iletilir.

**3.5.2 Çıkışların Hesaplanması**

Mantık kapıları, aldıkları giriş değerlerine göre çıkış değerlerini hesaplar ve bu değerleri bağlı oldukları bileşenlere iletir. Örneğin, AND kapısı, girişlerinin her ikisi de 1 olduğunda çıkış olarak 1 üretir.

**3.5.3 LED ve Çıkış Kutularının Güncellenmesi**

LED'ler ve çıkış kutuları, kendilerine iletilen değerleri görüntüler. LED'ler, aldıkları değere göre yeşil (1) veya kırmızı (0) olarak yanar. Çıkış kutuları, hesaplanan değerleri metin olarak gösterir.

# SONUÇ VE ÖĞRENİLEN DERSLER

Bu proje, mantık kapıları ve dijital devre tasarımının temel prensiplerini öğretmek için etkili bir araç olarak hizmet etmektedir. Kullanıcıların kendi devrelerini tasarlayıp test edebilmeleri, sayısal tasarım sürecini daha iyi anlamalarını sağlamaktadır. Projenin geliştirilmesi sırasında karşılaşılan zorluklar ve çözümler şu şekildedir:

**4.1 Zorluklar**

Bağlantı Yönetimi: Mantık kapıları ve diğer bileşenler arasındaki bağlantıları yönetmek ve doğru şekilde propagasyon sağlamak başlıca zorluklardan biriydi. Bileşenler arasındaki bağlantıların doğru şekilde kurulması ve giriş değerlerinin doğru bileşenlere iletilmesi için kapsamlı bir bağlantı yönetimi sistemi geliştirilmiştir.

Kullanıcı Etkileşimi: Kullanıcıların arayüz ile etkileşimde bulunarak bileşenleri kolayca ekleyip bağlayabilmeleri için uygun bir kullanıcı deneyimi tasarlamak önemliydi. Kullanıcı dostu bir arayüz sağlamak, projenin başarısı için kritik bir faktördü.

Görselleştirme ve Animasyon: Bileşenlerin ve bağlantıların doğru şekilde görselleştirilmesi ve animasyonların doğru zamanlamayla çalıştırılması, kullanıcıların devreyi daha iyi anlamalarını sağladı.

**4.2 Öğrenilen Dersler**

Modüler Tasarım: Mantık kapıları ve diğer bileşenlerin modüler bir yapıda tasarlanması, projenin genişletilebilir ve sürdürülebilir olmasını sağladı. Her bir bileşen, belirli bir işlevi yerine getiren bağımsız bir modül olarak tasarlandı. Bu modüler yaklaşım, bileşenlerin yeniden kullanılabilirliğini artırdı ve yeni bileşenlerin eklenmesini kolaylaştırdı.

Hata Yönetimi: Kullanıcı girişlerinin ve bağlantıların doğruluğunu kontrol etmek ve uygun hata mesajları göstermek, kullanıcı deneyimini iyileştirdi. Kullanıcıların hatalı girişler yapmasını önlemek ve doğru bağlantılar kurmalarını sağlamak için kapsamlı bir hata yönetimi sistemi geliştirildi. Bu sistem, kullanıcıların doğru devreler oluşturmalarına ve simülasyon sonuçlarını doğru bir şekilde gözlemlemelerine yardımcı oldu.

Kullanıcı Eğitimi: Kullanıcıların proje arayüzünü ve bileşenleri kullanmayı öğrenmeleri için açıklayıcı metinler ve kılavuzlar sağlandı. Bu sayede, kullanıcılar proje arayüzünü daha hızlı ve etkili bir şekilde kullanmayı öğrendi.

Simülasyon Performansı: Büyük ve karmaşık devrelerin simülasyonu sırasında performansın optimize edilmesi gerektiği anlaşıldı. Simülasyonun hızlı ve verimli bir şekilde çalışabilmesi için algoritmalar ve veri yapıları optimize edildi.

# KAYNAKÇA

Python Tkinter Documentation

ChatGPT

Github Bağlantı :

MÜSLÜM BATIN KILIÇ : https://github.com/batinkilic?tab=repositories