**LOJİK DEVRE BENZETİMİ**

Onur Kuş, Aziz Yelbay

Bilgisayar Mühendisliği Bölümü

Kocaeli Üniversitesi

[onurkus58g@gmail.com](mailto:onurkus58g@gmail.com), aziz1594@hotmail.com

Özet

*Programlama Laboratuvarı 1 dersi ikinci projede Programlama 1, Programlama 2 Mantıksal Tasarım ve Uygulamaları derslerinde edinilen bilgilerin uygulamaya geçirilmesi amaçlanmıştır.*

*Projede bizden istenen C/C++ programlama dili ve bize sağlanan örnek devreler aracılığı lojik devre benzetiminin gerçekleşmesi ve kullanıcıdan alınan komutlar doğrultusunda yeni değerlerin oluşturduğu devrenin son halini dosya sistemi ile metin belgesine yazdırmasını sağlar. Bu istekler doğrultusunda biz C Programlama dili ve CodeBlocks programını kullandık.*

*Program kullanıcının daha önceden belirlediği devreleri ve devre değerlerinin .txt uzantılı dosyaları programa çekerek belirli komutlarla devredeki kapıların değerlerinin değişimini loglanması ve devrenin son durumunu simüle edilmesini dosya sistemi yardımıyla gerçekleşmesini hedefledik.*

# Giriş

Lojik Devre Benzetimi programı devre.txt ve deger.txt metin belgelerini işleyerek konsol ekranına simülasyonunu daha sonra kullanıcının giriş kapılarını değiştirmesiyle (setlenme) diğer kapıların etkilenmesini ve bunla birlikte yayılma süresi değişimini ekrana ve metin belgesine yazdırır.

# Temel Bilgiler

Program C programlama dilinde geliştirilmiş olup, tümleşik geliştirme ortamı olarak “CodeBlocks” kullanılmıştır.

# Tasarım

Lojik Devre Benzetiminin programlanma aşamaları altta belirtilen başlıklar altında açıklanmıştır.

## Algoritma

Program kullanıcıdan aldığı Y<devre.txt> veya y<devre.txt> komutuyla devrenin bilgilerini programa çeker. Daha sonra I<deger.txt> veya i<deger.txt> komutuyla da devrenin değer bilgilerini programa yükler.

S veya s komutuyla devredeki son durumdaki kapıların değerlerini ve yayılma sürelerini nsn (nanosaniye) cinsinden ekrana çıktısını verir.

H veya h komutuyla giriş uçlarının değerini Lojil-1’e yükseltirken. L veya l giriş uçlarının değerlerini Lojik-0’a düşürülür.

**3.2 Kullanılan Fonksiyonlar**

void swap(char \*str1, char \*str2);

baska\_dosya.txt’i devre.txt’e uyarlar.

void giris5yap();

Tüm girişlerin sonradan değerleri değişeceği için 5 atar. Girişleri ismini sonradan belli olacağı için boşluk karakteri atar.

int not(int deger);

Giriş değerini NOT kapısı manntığıyla tersine çevirerek çıkışa atar.

int and (int giris1, int giris2, int giris3);

Giriş değerleri AND kapısı mantığıyla çıkışa atar.

int or (int giris1, int giris2, int giris3);

Giriş değerleri OR kapısı mantığıyla çıkışa atar.

int nand (int giris1, int giris2, int giris3);

Giriş değerleri NAND kapısı mantığıyla çıkışa atar.

int nor (int giris1, int giris2, int giris3);

Giriş değerleri NOR kapısı mantığıyla çıkışa atar.

int xor (int giris1, int giris2, int giris3);

Giriş değerleri XOR kapısı mantığıyla çıkışa atar.

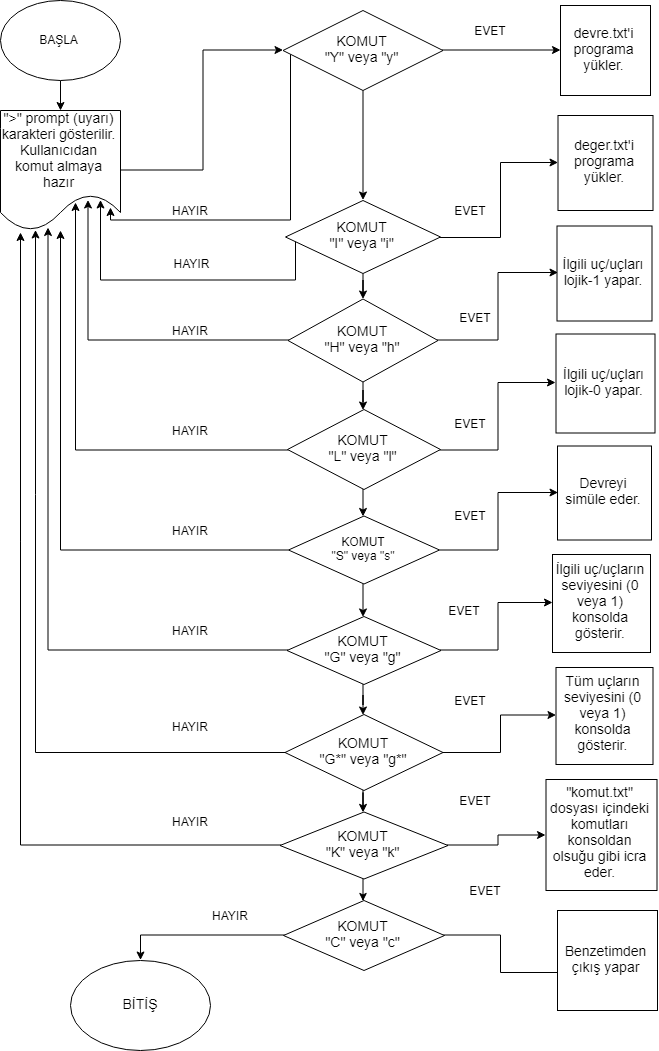
void yenile(int kapi\_sayisi);

Değerleri değişen kapıların son halini günceller.

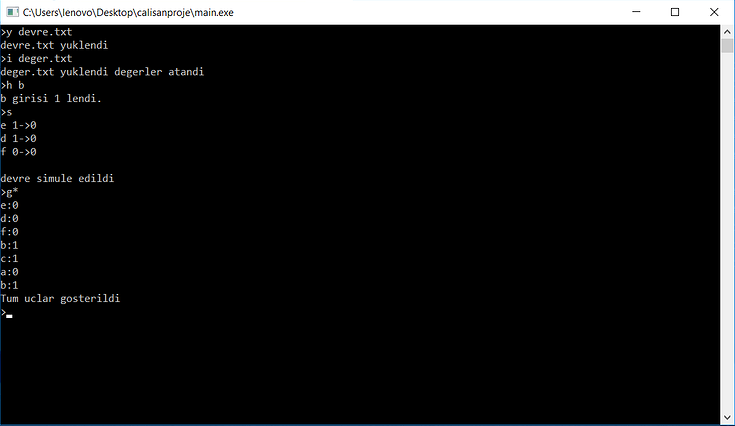
void yazdir(int kapi\_sayisi);

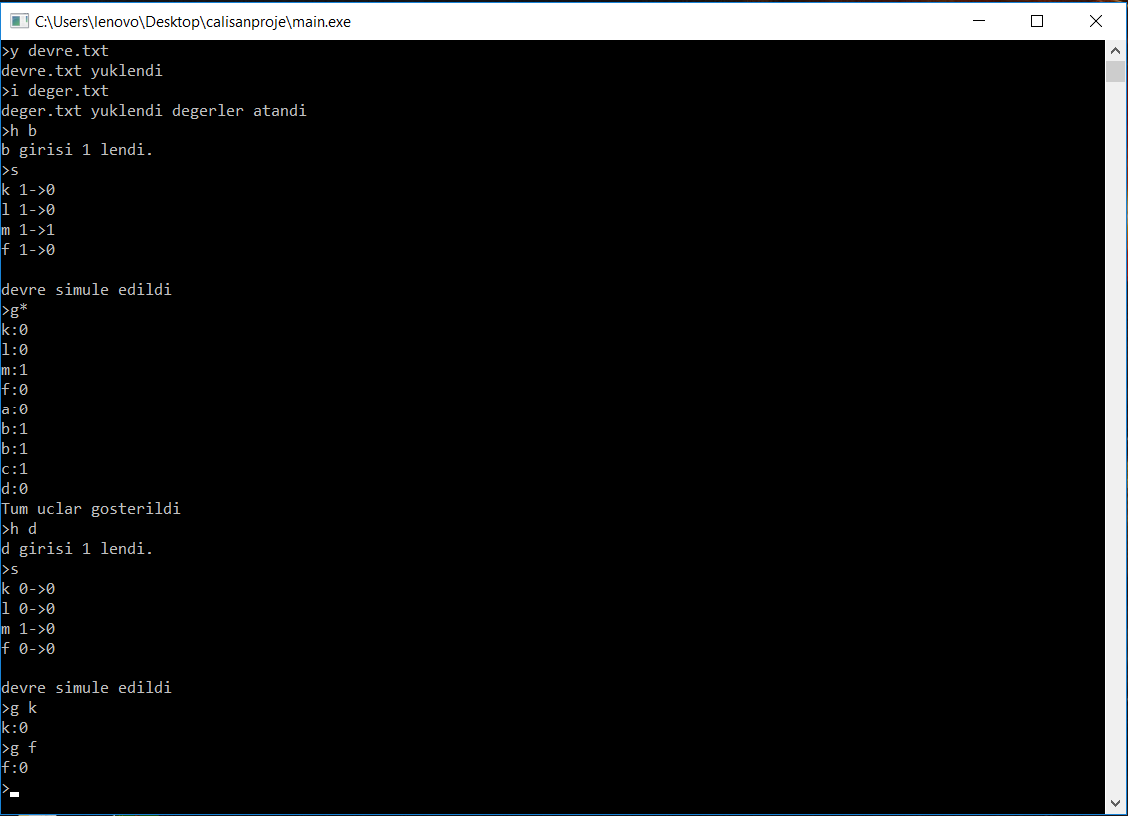
Devrenin son durumunu simüle ederek ekrana yazdırır.

**3.3 Akış Diyagramı**



**4.Ekran Çıktıları**





1. **5.Sonuçlar**

Program devre, değer ve başka dosya metinlerini programa çekmekte olup bütün komutları konsolda çalıştırmaktadır. Yalnızca yayılma sürelerinin değişimini verememektedir.

**6.Kaynakça**

1. https://www.geeksforgeeks.org/swap-strings-in-c/

/\*void swap2(char \*str1, char \*str2)

{

char \*temp = (char \*)malloc((strlen(str1) + 1) \* sizeof(char));

strcpy(temp, str1);

strcpy(str1, str2);

strcpy(str2, temp);

free(temp);

}\*/

( kod parçacığı)