

Mikrodenetleyici Tabanlı Oyun Makinesi Projesi

1. Çağla GÖK
Bilgisayar Mühendisliği
220201060
caglagok369@gmail.com

2. Mediha Tuğba Keyik
Bilgisayar Mühendisliği
220201044
medihatugbakeyik@gmail.com

Özetçe

Projemizin amacı çeşitli araçlar kullanarak mikrodenetleyici tabanlı bir oyun makinesi geliştirmektir. Proje Arduino IDE ve Proteus programları kullanılarak geliştirilmiştir.

I. GİRİŞ

Projemizde oyuncu, bir fiziksel palet kontrol cihazını kullanarak bir topu yansıtarak OLED ekrandaki tuğlaları kırmaya çalışacaktır. Oyuncu, fiziksel bir "palet kontrol cihazı" kullanacaktır ve bu cihaz, potansiyometre ile kontrol edilecektir.

Projenin ana istekleri şu şekildedir:

- Başlat ve Çıkış seçeneklerinden oluşan bir menü,
- Potansiyometre ile oyun kontrolü,
- Toplam 5 level ve her level farklı tuğla düzeni,
- Oyuna 3 can ile başlanması ve ek can,
- Can kaybettiğinde ledlerin ışığının sönmesi ve ek can alındığında ledlerin ışığının tekrar yanması,
- Her kırılan tuğla için skor puanın 1 artması,
- Oyun bittiğinde oyunun sıfırlanması ve yeni oyun için ana menüye dönülmesi,
- Topun hızının level atladığında belirli bir oranda artması.

II. KULLANIM

A. Kütüphane Kullanımı

Adafruit_GFX, Adafruit_SSD1306, Arduino Library gibi kütüphaneler projeye dahil edilmiştir.

B. Proteus ve Arduino Kurulumu

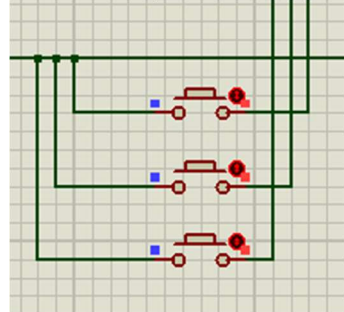
Kurulum yapılırken orijinal orijinal sitelerinden ücretsiz sürümleri indirilmiştir. İndirilen kütüphaneler eklenip çalıştırılabilir hale getirilmiştir.

C. Devre Elemanları

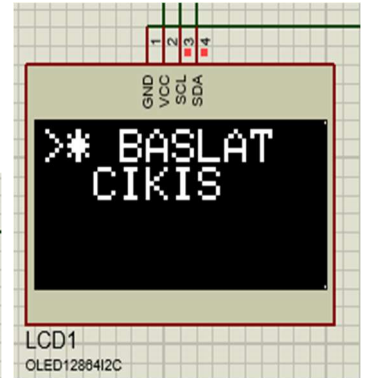
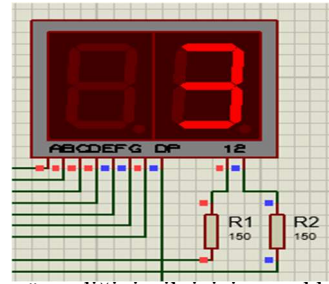
Devre yapımı sırasında Arduino Mega, Oled ekran, potansiyometre, 7 segment display(katod), 3 tane led, 3 tane buton ve 5 tane direnç kullanılmıştır.

III. YÖNTEM VE İLERLEYİŞ

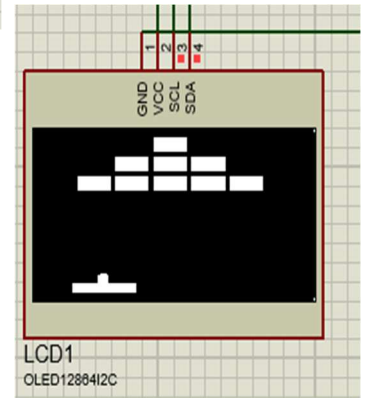
Oyunumuz başlatıldığında ilk olarak OLED ekranımızda Başlat ve Çıkış seçeneklerinin olduğu bir menüyle başlar. Bu seçenekler Proteus'taki devremizde yer alan 3 buton



ve diğer buton seçim butonudur. Oyuncu Başlat seçeneğini seçtiğinde Oyun Başlıyor... yazısı gösterildikten sonra level 1



LCD1
OLED12864I2C
aracılığıyla seçilebilir. Bir buton aşağı, bir buton yukarı



LCD1
OLED12864I2C
başlar. Çıkış seçeneğine basıldığında "Oyunumuza gösterdiğiniz ilgi için teşekkürler." yazısı gösterilip program

sonlanır. Oyun başladığında oyuncu potansiyometre yardımıyla top ile tuğlaları kırmaya başlar. Her kırılan tuğla için skor 1 puan artar.

Her tuğla kırıldığında yüzde 10 şans ile bir obje düşürecek şekilde ayarlanmıştır. Oyuncu bu düşen objeyi aldığı anda ek can elde eder. Eğer sönük led varsa tekrar yanmaya başlar.

Oyuncu level 1 de yer alan tüm tuğlaları kırdığında 3 saniye kadar bekletilip level 2 ye geçer. Her level için farklı tuğla düzeni kurulmuştur. Oyuncu yanana kadar oyun bu şekilde devam eder. Oyun sonlandığında Oyunu kaybettiniz ve Skor yazar ve 5 saniye gösterilir. Ardından ana menüye döner.

Fonksiyonlar şu şekildedir:

-MenuState ve GameState enum'ları tanımlanır. Bunlar, sırasıyla menü seçeneklerini ve oyun durumlarını belirlemek için kullanılır. currentMenuSelection ve currentGameState gibi değişkenler, mevcut menü seçeneği ve oyun durumunu takip eder.

-Arduino üzerindeki buton pinler tanımlanır.

-OLED ekranı için gerekli tanımlamalar yapılır.

-Oyun elemanları (plaka, top, tuğlalar) için boyutlar ve hızlar sabitlerle belirlenir.

-Tuğla matrisi ve farklı seviyeler için tuğla dizilimleri tanımlanır.

-Can sayısı, zamanlayıcılar, puan, LED pinleri, ışık sensörü pinleri ve eşik değerleri gibi diğer değişkenler ve sabitler belirlenir.

Bu adımlar, oyunun başlatılması için gerekli temel yapıyı oluşturur ve ardından oyunun ana döngüsüne geçilir.

setup() fonksiyonu, Arduino'nun başlangıç ayarlarını yapmak için kullanılır. Bu fonksiyon, Arduino kartı başlatıldığında yalnızca bir kez çağrılır.

setup() fonksiyonunda aşağıdaki işlemler gerçekleştirilir:

-Seri iletişimi başlatır: Serial.begin(9600);

-Buton pinlerini giriş olarak ayarlar: pinMode(buttonPinDown, INPUT_PULLUP);, pinMode(buttonPinUp, INPUT_PULLUP);, pinMode(buttonPinStart, INPUT_PULLUP);.

-OLED ekranını başlatır: display.begin(SSD1306_SWITCHCAPVCC, 0x3C);.

-LED pinlerini çıkış olarak ayarlar ve başlangıçta yanık konumda tutar:

```
for (int i = 0; i < 3; i++) {  
    pinMode(ledPins[i], OUTPUT); digitalWrite(ledPins[i], HIGH); }  
}
```

-Ekran temizlenir ve başlangıç ekranı gösterilir: display.clearDisplay(); displayMenu();.

-Toplam 7 segment göstergesi için pinleri çıkış olarak ayarlar: for (int i = 22; i < 29; i++) { pinMode(i, OUTPUT); } pinMode(29, OUTPUT); pinMode(30, OUTPUT);.

-Oyunu başlatır ve başlangıç skorunu göstermek için displayScore() fonksiyonunu çağırır: startGame(); displayScore();

-loop() fonksiyonu, Arduino kodunun ana döngüsünü oluşturur. Bu fonksiyon, sürekli olarak tekrarlanır ve Arduino kartı çalıştığı sürece aktif kalır.

-loop() fonksiyonunda aşağıdaki işlemler gerçekleştirilir:

Oyun Menüsü Kontrolü (GAME_MENU Durumu):

Eğer oyun menüsünde (GAME_MENU durumunda) ise: isButtonPressed() fonksiyonuyla yukarı ve aşağı butonlarının durumu kontrol edilir.

-Başlatma/onaylama butonunun durumu kontrol edilir.

-Kullanıcının menü seçimini yapmasını ve ilgili duruma geçiş yapmasını sağlayan fonksiyonlar kullanılır: displayMenu().

-Oyun Çalışma Kontrolü (GAME_RUNNING Durumu):

-Eğer oyun çalışır durumda (GAME_RUNNING durumunda) ise:

moveBall() fonksiyonuyla topun hareketi kontrol edilir ve çarpışma durumları kontrol edilir.

checkBallLoss() fonksiyonuyla can kaybı kontrol edilir ve gerekli işlemler yapılır.

checkBricks() fonksiyonuyla tuğlaların tamamının yok olup olmadığı kontrol edilir ve yeni seviyeye geçiş yapılır.

-Oyun ekranı temizlenir ve plaka, top, tuğlalar çizilir.

-Her döngüde ekrana güncel bilgiler yansıtılır.

-Oyun Bitiş Kontrolü (GAME_EXIT Durumu):

-Eğer oyun bitiş durumunda (GAME_EXIT durumunda) ise: Teşekkür mesajı ekrana yazdırılır.

-Belirli bir süre boyunca teşekkür mesajı gösterilir ve ardından program sonlandırılır.

Bu işlemler, loop() fonksiyonunun içinde bulunan alt fonksiyonlar kullanılarak gerçekleştirilir.

-displayMenu(): OLED ekranda başlangıç menüsünü gösterir.

-movePlate(): Plakanın konumunu potansiyometre değerine göre ayarlar.

-moveBall(): Topun hareketini kontrol eder, ekran kenarlarına çarpmaları ve tuğlaları vurması durumunda gerekli işlemleri gerçekleştirir.

-drawBricks(): Tuğla dizisini OLED ekranda çizer.

-displayScore(): Skoru 7 segment göstergesine yazdırır.

-resetGame(): Oyunu sıfırlar ve başlangıç menüsünü gösterir.

-increaseBallSpeed(): Topun hızını artırır.

-isButtonPressed(): Belirtilen butonun basılıp basılmadığını kontrol eder.

-updateLivesDisplay(): Can göstergesini günceller.
-checkBallLoss(): Topun düşüp düşmediğini kontrol eder ve gerekli işlemleri gerçekleştirir.
-updateDisplayColors(): Işık sensörüne göre ekran renklerini günceller.
-checkBricks(): Tuğlaların tümünün yok olup olmadığını kontrol eder ve gerekli işlemleri gerçekleştirir.
-moveLifeUp(): Can artırma objesinin hareketini kontrol eder.
-brickHit(): Tuğla vurulduğunda can artırma objesi oluşturur.
-resetBall(): Topu sıfırlar ve oyunun devam etmesini sağlar.
-displayLevelComplete(): Seviye tamamlandığında ekrana mesaj gösterir.

KAYNAKLAR

- [1] [Proteus'a Arduino Kütüphanesi Nasıl Eklenir | How to Add Arduino Library in Proteus | - YouTube](#)
- [2] [Arduino Library For Proteus V3.0 - The Engineering Projects](#)
- [3] [https://github.com/SabariKannanM/Arduino-UNO-Ping-Pong-Game-v2.0/blob/master/Circuit%20diagrams/Arduino%20UNO%20-%20Ping%20Pong%20game%20v2.0%20\(1\).png](https://github.com/SabariKannanM/Arduino-UNO-Ping-Pong-Game-v2.0/blob/master/Circuit%20diagrams/Arduino%20UNO%20-%20Ping%20Pong%20game%20v2.0%20(1).png)
- [4] [Arduino-UNO-Ping-Pong-Game-v2.0/Arduino code/arduino uno ping pong game v2 0 at master · SabariKannanM/Arduino-UNO-Ping-Pong-Game-v2.0](#)
- [5] [PROTEUS DERSLERİ: Proteus ile neler yapılır | Proteus Ders Notları | - YouTube](#)
- [6] [Arduino ile Proteus Nasıl Çalışır ? | Arduino Eğitimleri - YouTube](#)
- [7] [OLED with Arduino in proteus - YouTube](#)
- [8] [PROTEUS-ARDUINO-SIMULATION/proteus_simulation.ino at master · ArduinoHocam/PROTEUS-ARDUINO-SIMULATION](#)
- [9] [Devre Kurmadan Arduino Projenizi TEST EDİN-PROTEUS ARDUINO SIMULATION - YouTube](#)
- [10] [Arduino-UNO-Ping-Pong-Game-v2.0/Circuit diagrams/Arduino UNO Ping Pong Game v2.0.png at master · SabariKannanM/Arduino-UNO-Ping-Pong-Game-v2.0](#)
- [11] [Arduino Game Project: Pong Game using an Arduino Uno and Color OLED display \(SSD1331\). Easy tutorial - YouTube](#)
- [12] [Ping Pong Oynadık! 0.96" Oled Ekran Kullanımı - YouTube](#)
- [13] [PROTEUS DERSLERİ: Proteus ile neler yapılır | Proteus Ders Notları | - YouTube](#)
- [14] <https://www.protus.com.tr/hakkimizda.html>
- [15] <https://www.arduino.cc/>
- [16] <https://lezzetlirobotarifleri.com/7-segment-display/>
- [17] <https://www.youtube.com/watch?v=kKWK6ZdvC3A>
- [18] <https://www.theengineeringprojects.com/2021/03/arduino-library-for-proteus-v2.html>
- [19] <https://www.youtube.com/watch?v=E27WfriOIU8>
- [20] <https://www.youtube.com/watch?v=Dvsy9AS5bjo>
- [21] <https://devreyakan.com/arduino-ile-7-segment-0-99-sayici-ve-proteus-simulasyonu/>

BAŞLA
MENÜYÜ GÖSTER
KULLANICI SEÇENEK SEÇENE KADAR BEKLE
EĞER SEÇENEK "BAŞLAT" İSE
OYUNU BAŞLAT
EĞER SEÇENEK "ÇIKIŞ" İSE
OYUNDAN ÇIK
OYUNU BAŞLAT
OYUNCU CAN SAYISINI AYARLA (3)
SEVİYEYİ AYARLA (1)
OYUN DEVAM EDİYOR = TRUE
SÜREÇ DEVAM ETTİĞİ SÜRECE
EĞER SEVİYE 1 İSE
TUĞLA DİZİLİMİNİ YÜKLE (SEVİYE 1)
EĞER SEVİYE 2 İSE
TUĞLA DİZİLİMİNİ YÜKLE (SEVİYE 2)
...
EĞER SEVİYE 5 İSE
TUĞLA DİZİLİMİNİ YÜKLE (SEVİYE 5)
OYUNCU PLAKAYI HAREKET ETTİR
TOPUN HAREKETİNİ GÜNCELLE
EĞER TOPLAM CAN 0 İSE
OYUNU SONLANDIR
EĞER TÜM TUĞLALAR KIRILMIŞ İSE
SEVİYEYİ ARTIR
EĞER SEVİYE 6'YA ULAŞTIYSA
SEVİYEYİ 1'E SIFIRLA
SKORU GÜNCELLE
OYUN EKCRANINI GÜNCELLE
OYUNU SONLANDIR
OYUNU SONLANDIR
EKRANI TEMİZLE
"OYUN BİTTİ" MESAJI GÖSTER
SEÇENEKLERİ GÖSTER
KULLANICI SEÇENEK SEÇENE KADAR BEKLE
EĞER SEÇENEK "BAŞLAT" İSE
OYUNU BAŞLAT
EĞER SEÇENEK "ÇIKIŞ" İSE
OYUNDAN ÇIK
OYUNDAN ÇIK

