

# Arduino Oyun Makinesi

Berat Ölmez, Sait Ömer San  
210201074, 220201001  
Bilgisayar Mühendisliği Bölümü  
Kocaeli Üniversitesi  
Kocaeli/Türkiye  
beratolmez123@hotmail.com  
sainsaitomer@gmail.com

**Özetçe**—Arduino ve çeşitli araçlar yardımı ile mikroişlemci tabanlı, etkileşimli, dijital gösterimli, çeşitli bölümlerden oluşan oyun konsolunun geliştirilme aşamaları gösterilecektir. Mikroişlemci programlama, baskı devre tasarlama, devre elemanlarını kullanma becerilerinin edinilmesi, geliştirilmesi ve uygulanması amaç edinilmiştir.

## I. GİRİŞ

Bu proje temeli yirminci yüzyılın ikinci yarısının başlangıcına uzanan; teknolojik gelişmeler, pazar ve ekonomik faktörler sonucunda eğlence, bilimsel merak gibi çeşitli amaçlar için geliştirilen; çeşitli firmalar tarafından farklı versiyonları üretilmiş, son yıllarda popülerliği tekrar artan taşınabilir oyun konsollarına basit bir örnek niteliği taşımaktadır. Platform olarak Arduino Uno, platform hareketi için potansiyometre, dijital ekran gösterimi için 128x64 OLED ekran modülü, oyuna giriş veya oyundan çıkış yapılmasını sağlamak amacı ile 3 adet push buton, oyuncunun kalan haklarını (can) ifade etmek amacı ile 3 adet led, ekran renklerini tersine çevirmek amacı ile ışık sensörü, oyuncunun skorunu görebilmesi amacı ile de 7 segment display kullanılmıştır.

## II. YÖNTEM

### A. Şema Tasarımı

Arduino Uno, elektronik projelerin gerçekleştirilmesi için kodlanabilir bir geliştirme kartıdır. ATmega328 mikrodenetleyiciye sahip olan bu kart, geniş bir GPIO (Genel Amaçlı Giriş/Çıkış) pin yelpazesi, analog girişler ve USB bağlantısı gibi özelliklere sahiptir. Kart, kullanıcı dostu bir programlama ortamı olan Arduino IDE ile programlanabilir. USB üzerinden bilgisayara bağlanarak kodlar yüklenir. Bu projede Arduino Uno

kartı kullanılarak OLED ekran üzerinde butonlar ve potansiyometre kullanılarak kontrol edilen tuğla kırma oyunu yapılmıştır. Bu oyunda kullanıcı 3 cana sahip olarak oyuna başlar; top yere düştükçe can kaybeder ve canı 0 olduğunda oyun biter. Can kaybetmesine karşı tuğla kırıldıkça %10 ihtimalle tuğlanın kırıldığı yerden ödül düşürmektedir ve eğer bu ödül platform tarafından toplanırsa can sayısı artmaktadır. Oyun 5 farklı bölümden oluşmaktadır ve level atlandıkça topun hızı artmaktadır. Bu proje Proteus programı kullanılarak gerçekleştirilmiştir. Proteus kullanılarak Arduino Uno kartını OLED ekrana bağlamak için I2C haberleşme protokolüyle uyumlu olan A4 ve A5 pinleri kullanılmıştır. OLED ekranın güç ve toprak bağlantıları yapıldıktan sonra can bilgisini tutmak amacıyla 3 adet led 9,10 ve 11. pinlere bağlanmıştır. Kullanıcının oyunu oynarken kırıdığı tuğlaların 0 ve 9 arasında kayıt edilebilmesi amacıyla ortak katot 7 segment display kullanılmıştır. 7 segment display üzerinde bulunan A,B,C,D,E,F,G pinleri Arduino Uno kartında sırasıyla 2,3,4,5,6,7,8 numaralı pinlere bağlanmıştır. Kullanıcının oyunu daha konforlu oynayabilmesi için ışık değişimlerine bağlı olarak ekranın renkleri siyah ve beyaz renklerine değiştirilebilmektedir. Işığa bağlı olan bu değişimi sağlamak için LDR(light dependent resistor) kullanılmıştır. Bu malzemenin direnci ışık şiddeti arttıkça azalır ve ışık şiddeti azaldıkça artar. Işık değişimine bağlı olan direnç değişimi gerilim bölücü devresi kullanılarak gerilim değişimine dönüştürülmüştür ve Arduino Uno kartının analog pinlerinden olan A2 pinine bağlanmıştır. Oyun oynanırken platformun hareketinin kontrolü için potansiyometre kullanılmıştır bu potansiyometre analog bir pin olan A0 pinine bağlanmıştır. Menü üzerinden kullanıcının “Başlat” ve “Çıkış” seçeneklerini seçebilmesi için 3 adet buton

kullanılmıştır. Bu butonlar menüde yukarı gitme seçim yapma ve aşağı gitme işlemlerini yapmaktadır. Menü üzerinde gezinmeyi ve seçim yapmayı sağlayan butonlar 12, A1 ve 13 numaralı pinlere bağlanmıştır.

### B. Kodun Çalışma Mantığı

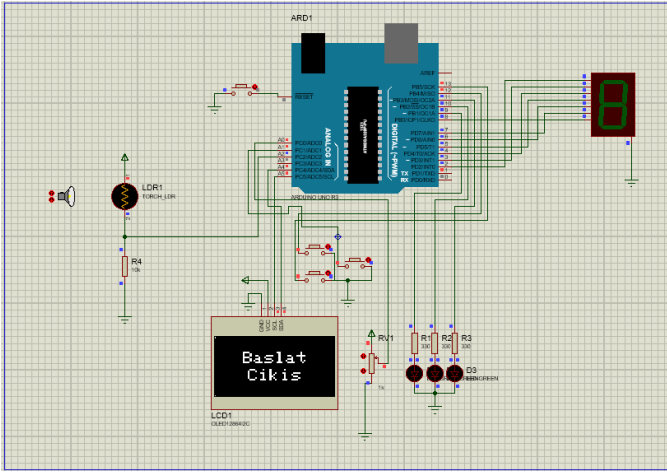
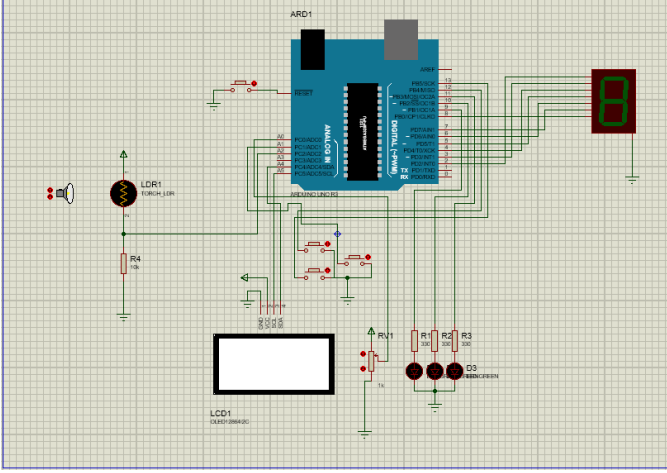
Proje Arduino IDE kullanılarak kodlanmıştır. OLED ekranın kullanılabilmesi için I2C protokolünü içeren “<Wire.h>” kütüphanesi ve OLED ekranda grafiksel işlemlerin yapılabilmesini sağlayan “<Adafruit\_SSD1306.h>” kütüphanesi kurulmuştur. Bu kütüphaneler kullanılarak OLED ekranda yazı yazdırma tuğla çizdirme gibi işlemler gerçekleştirilmiştir. Kodda ilk olarak devrenin şemasında kullanılan pinler define edilmiştir. Pinler define edildikten sonra ADC(Analog to Digital Converter) içeren pinlerden veri okuyan LDR ve potansiyometreye bağlı olan pinler input olarak ayarlanmıştır. Dışarıya veri veren butonlara ve ledlere bağlı olan pinler output olarak ayarlanmıştır. Kodda “tuğlasiyahap” ve “tuğlabeyazyap” isimli fonksiyonlar LDR’den okuduğu değere göre ekranın rengini değiştirmektedir. Kodda tuğlalar “bricks” matrisi kullanılarak konumlandırılmıştır bu matriste 1 olan yerlerde tuğlalar durmakta 0 olan yerlerde ise tuğlalar silinmektedir. Bu matris kullanılarak 5 farklı bölüm için harita oluşturulmuştur. Harita oluşturulduktan sonra topun hareketini ayarlayan “updateBall” fonksiyonu yazılmıştır. Bu fonksiyon topun tuğlaların olduğu alana gittikçe tuğlaların silinmesini sağlamak, top yere düştüğünde canın azalmasını sağlamak, %10 ihtimalle kırılan tuğladan ödül düşmesini sağlamak, 7 segment üzerine kırılan tuğla sayısını yazmak ve topun konumunu değiştirmek gibi işlevleri yerine getirmektedir. Düşen ödülün platformla temasının kontrolü için “updatebrickFall” fonksiyonu kullanılmıştır. Potansiyometrenin hareketine bağlı olarak platformu hareket ettirmek için kodda “updatePaddle” fonksiyonu kullanılmıştır. Projede topun hareketi “updateBall” fonksiyonu kullanılarak 2 farklı yöntemde sağlanmıştır. Bu yöntemlerden birincisinde topun fizik yasalarına uygun yani geldiği açıyla aynı açıda yansıması sağlanmıştır top platforma ve tuğlalara çarptığında fizik kurallarına uygun olarak yansımaktadır. İkinci yöntemde ise tuğla kırma temalı oyunlarda genellikle kullanılan oyuncunun topa daha esnek yön verebilmesini sağlayan bir yöntem kullanılmıştır. Bu yöntemde topun sektiği platform

parçalara ayrılmıştır ve bu parçalara ayrılan platformda top sağa gelirse sağa sola gelirse sola doğru yansımaktadır. Bu yöntemde topun yansımasının açısı da platforma geldiği bölgeye göre değişebilmektedir. Bu iki hareket yöntemi arasında geçiş A3 pinine bağlanmış olan anahtarın açılıp kapanması aracılığıyla sağlanmaktadır. Proje Arduino IDE üzerinde derlendikten sonra HEX kodu oluşturulmuştur. Oluşturulan HEX kodunun uzantısı Proteus programının içinde bulunan Arduino Uno kartının içine yerleştirilmiştir. Bu sayede Proteus içinde bulunan Arduino Uno kartının simülasyonu yapılmıştır.

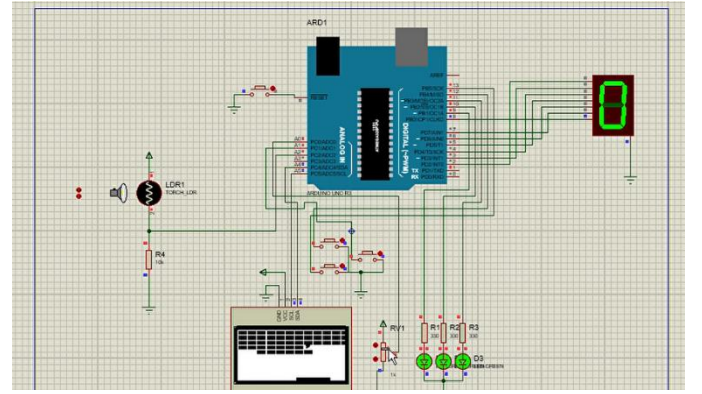
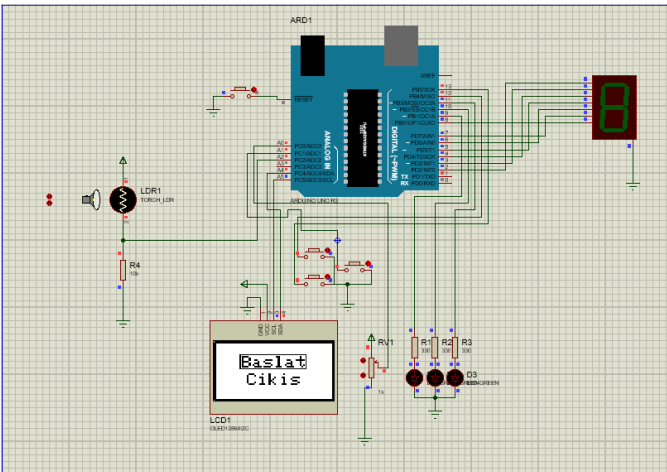
### III.DENEYSEL YÖNTEM

Kodun çalıştırılması ve oyunun farklı ışık seviyelerindeki hali sonraki sayfalarda bulunan resimlerde verilmiştir. Not: Alınan ekran görüntülerinde devrenin tamamının gösterilmesinin sebebi, deneyin yürütüldüğü ortamın projenin aslı olduğunun ispatı çabasıdır.

## 1. Proteus ile Arduino'nun çalıştırılması

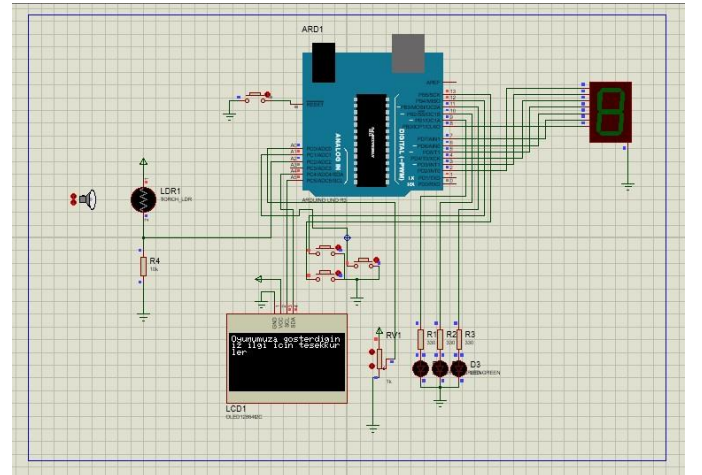
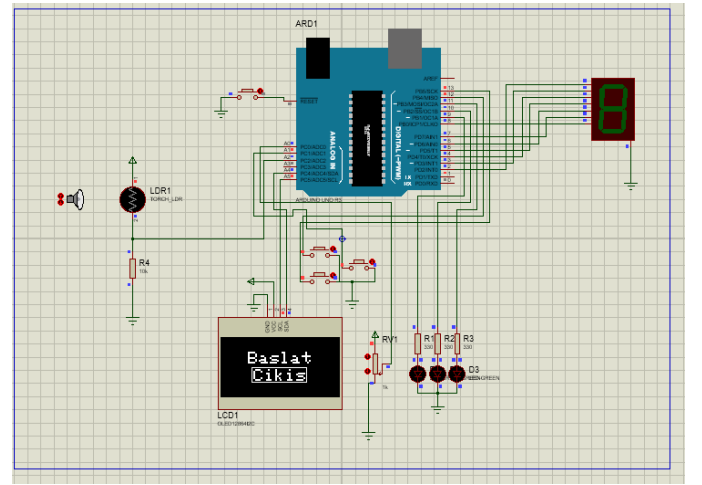


## 2. Işık sensörü testi

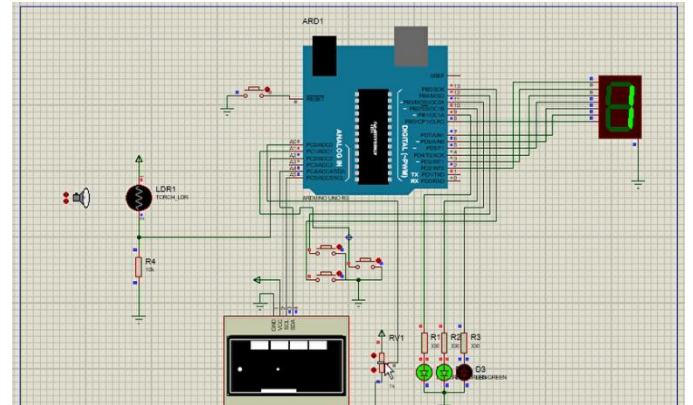
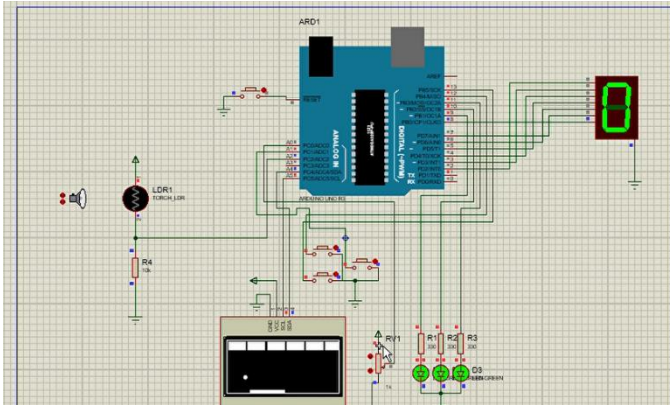
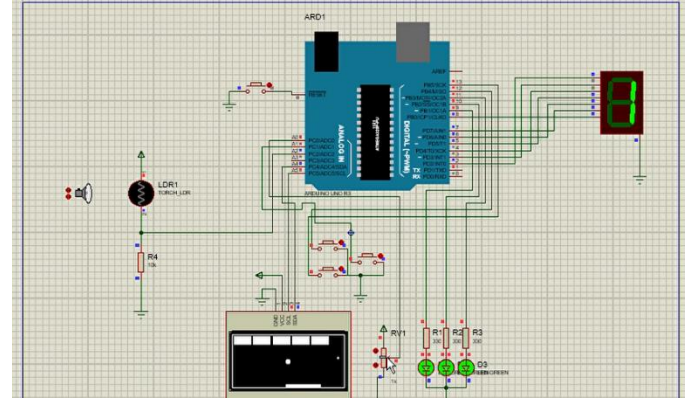
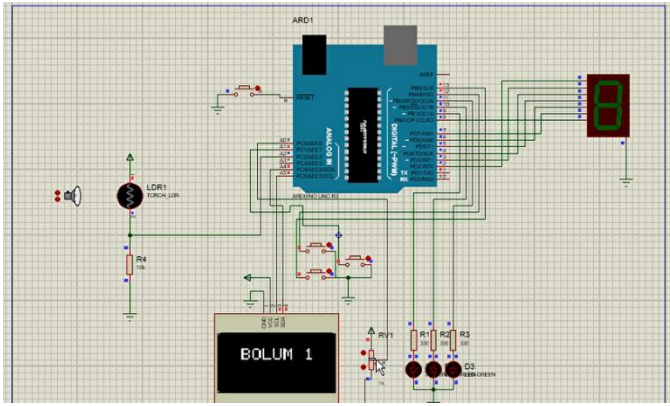
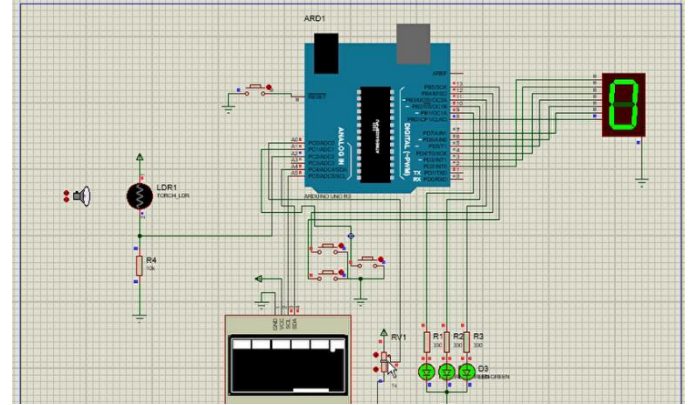
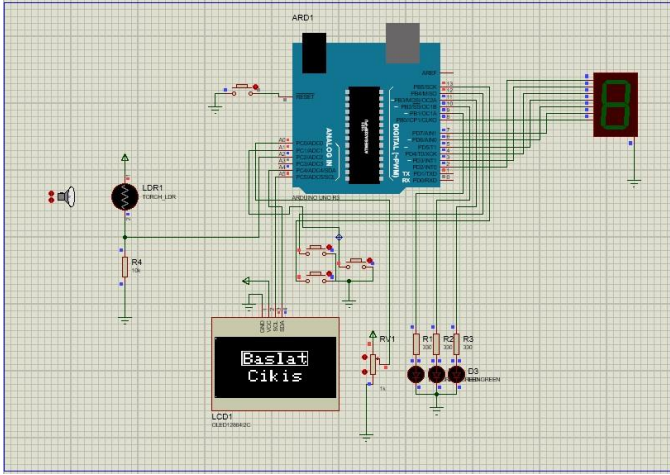


## 3. Butonlar ile seçenek seçimi ve seçeneğin onaylanması

Çıkış seçeneği:



Başlat seçeneği:

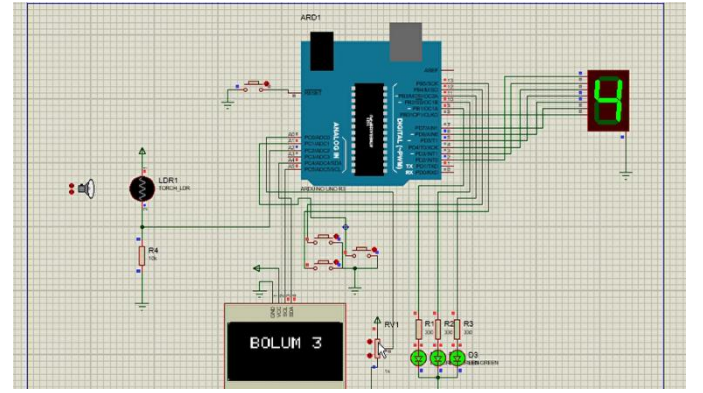
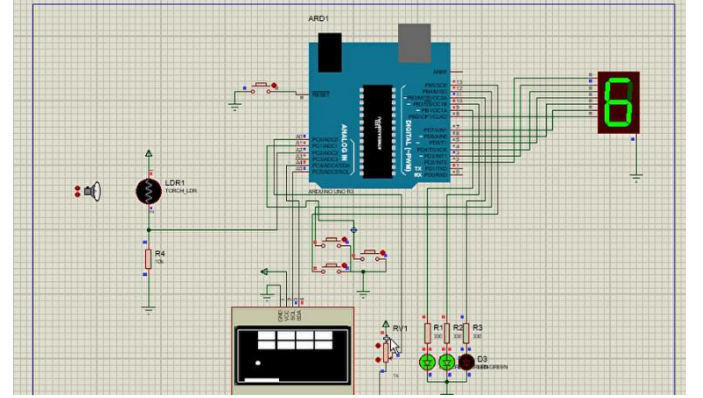
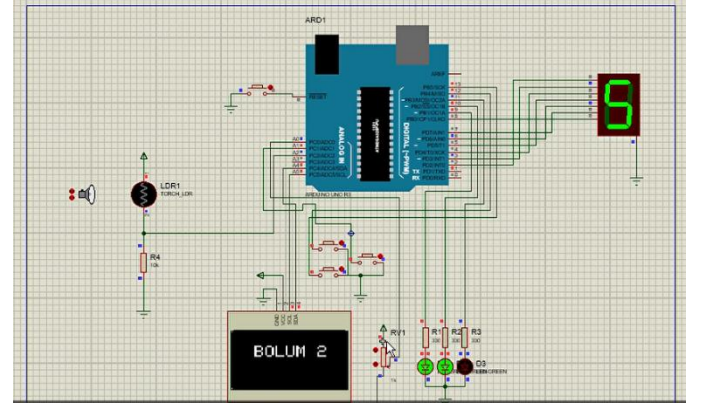


4. Potansiyometre kullanarak platform hareket ettirme, tuğla kırma, puan kazanma, kırılan tuğla sonucu şans eseri can düşürme

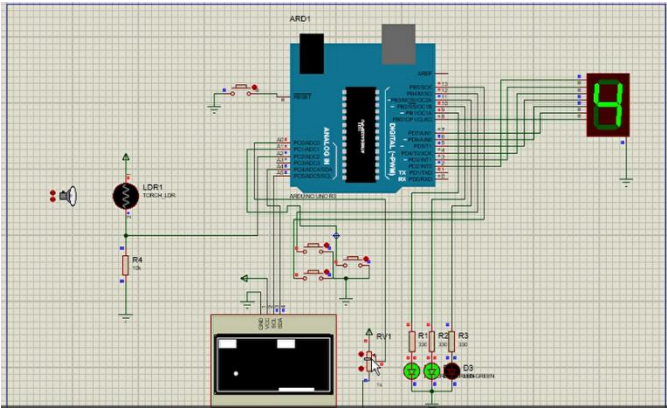
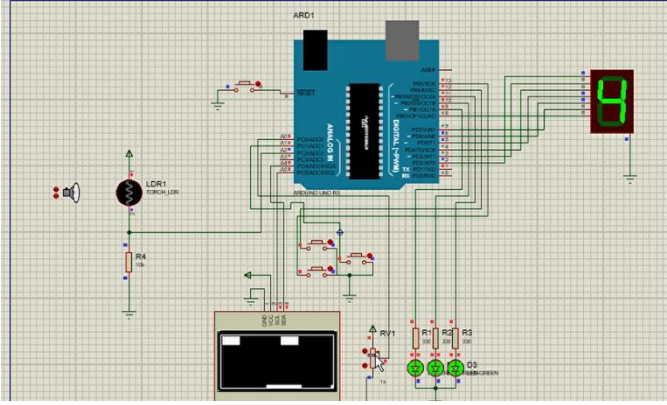
5. Canın platform ile yakalanması sonucu can kazanılması



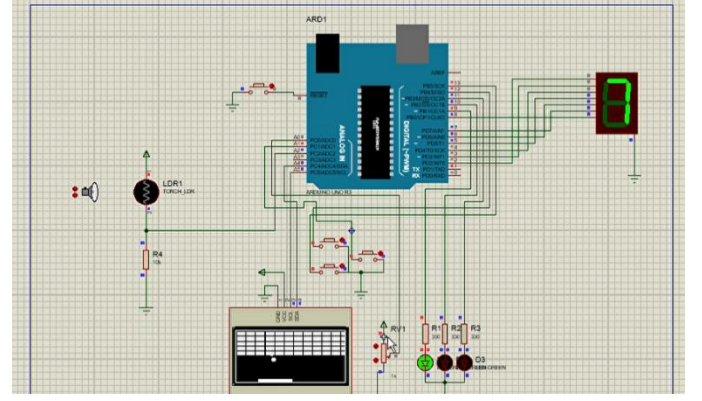
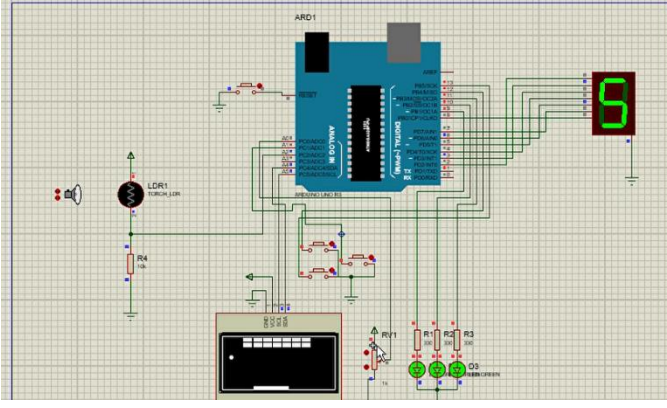
7. Başarıyla tamamlanan bölümler sonucu bir sonraki bölümlere geçme ve bölüm tasarımları



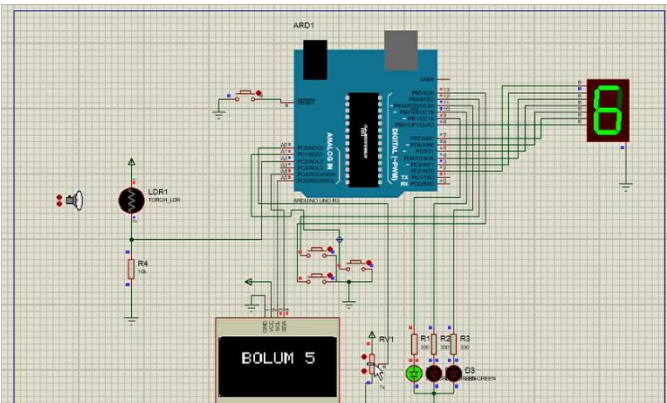
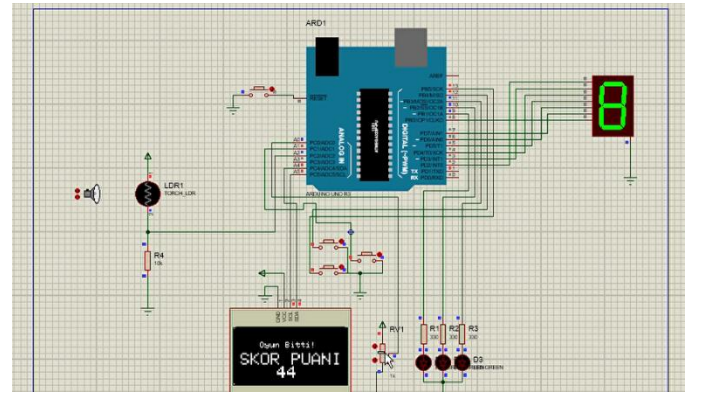
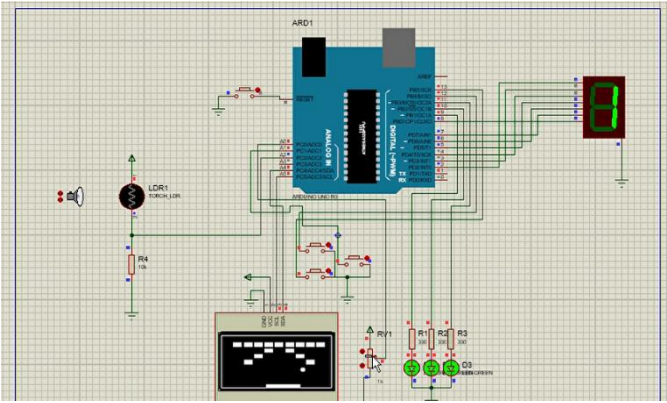
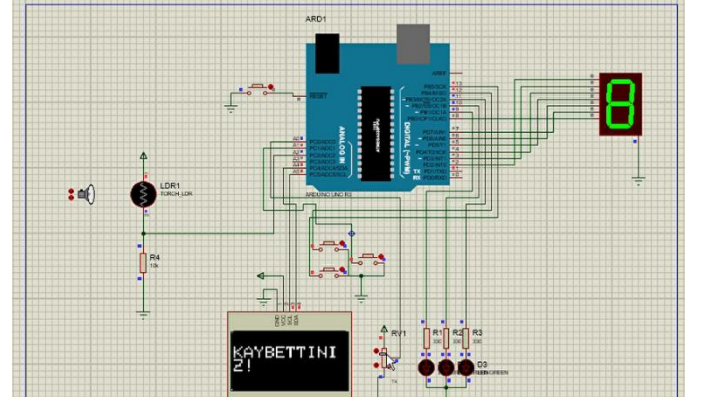
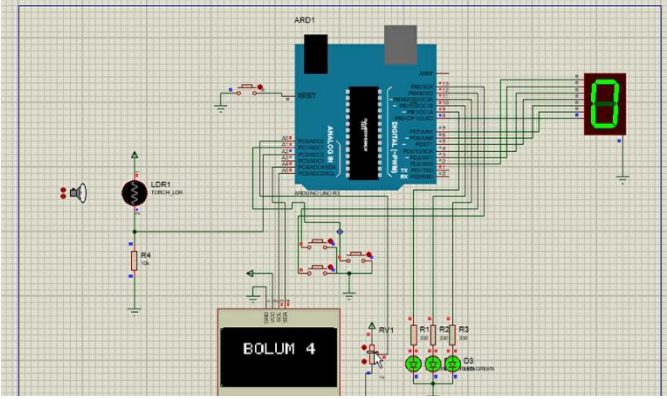
6. Topun zemine düşmesi sonucu can kaybetme







#### 8. Tüm canların kaybedilmesi sonucu oyunun bitişi ve skor gösterimi



#### IV.SONUÇ

Proje Arduino IDE ve Proteus programları kullanılarak çalıştırılmıştır. Başlangıç, çıkış ve seçim yapma butonları projede oyunun başlatılması ve çıkış yapılması için kullanılmıştır. LDR ışık sensörü kullanarak aydınlık ve karanlığa duyarlı şekilde OLED ekrandaki baskın renk ayarlanmıştır. Ortak katot 7 segment kullanarak kırılan tuğlalar 1 ve 9 arasında yazdırılmıştır. 3 adet led kullanarak oyuncunun can bilgisi gösterilmiştir. 5 farklı

tasarımla 5 farklı level oluşturulmuştur. Oyun başladığında ilk level başlatılarak kullanıcının canı bitinceye kadar kırıdığı tuğlalar kayıt altına alınmıştır. %10 ihtimalle kırılan tuğlalardan toplanması halinde canı arttıran ödül düşürülmesi sağlanmıştır. Oyun bitiminde başlangıç menüsüne dönülüp oyunun tekrar baştan başlatılmasına hazır hale getirilmiştir.

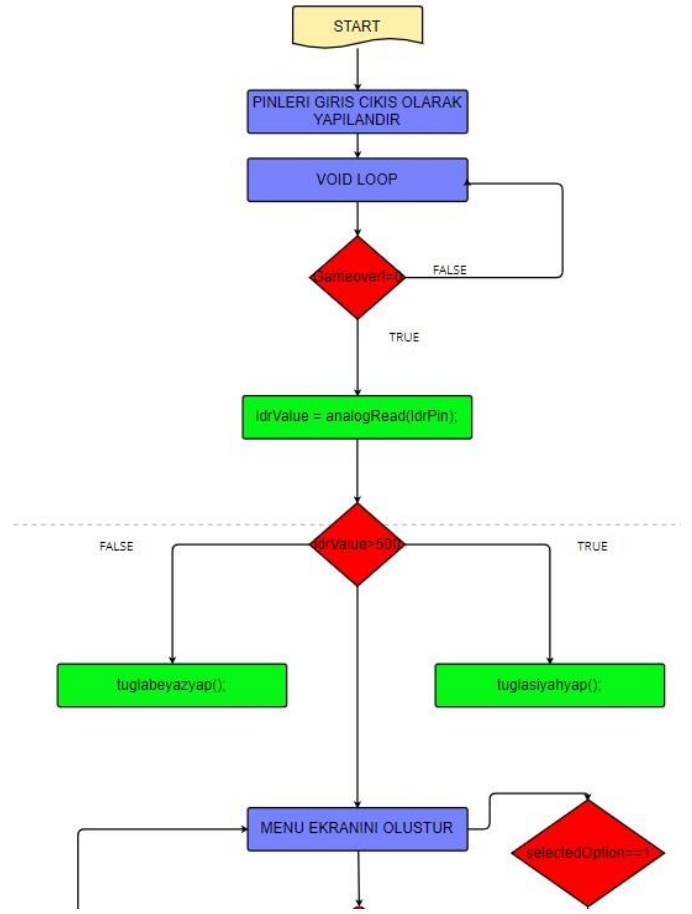
## V.UML DİYAGRAMI

### ArdunioGame

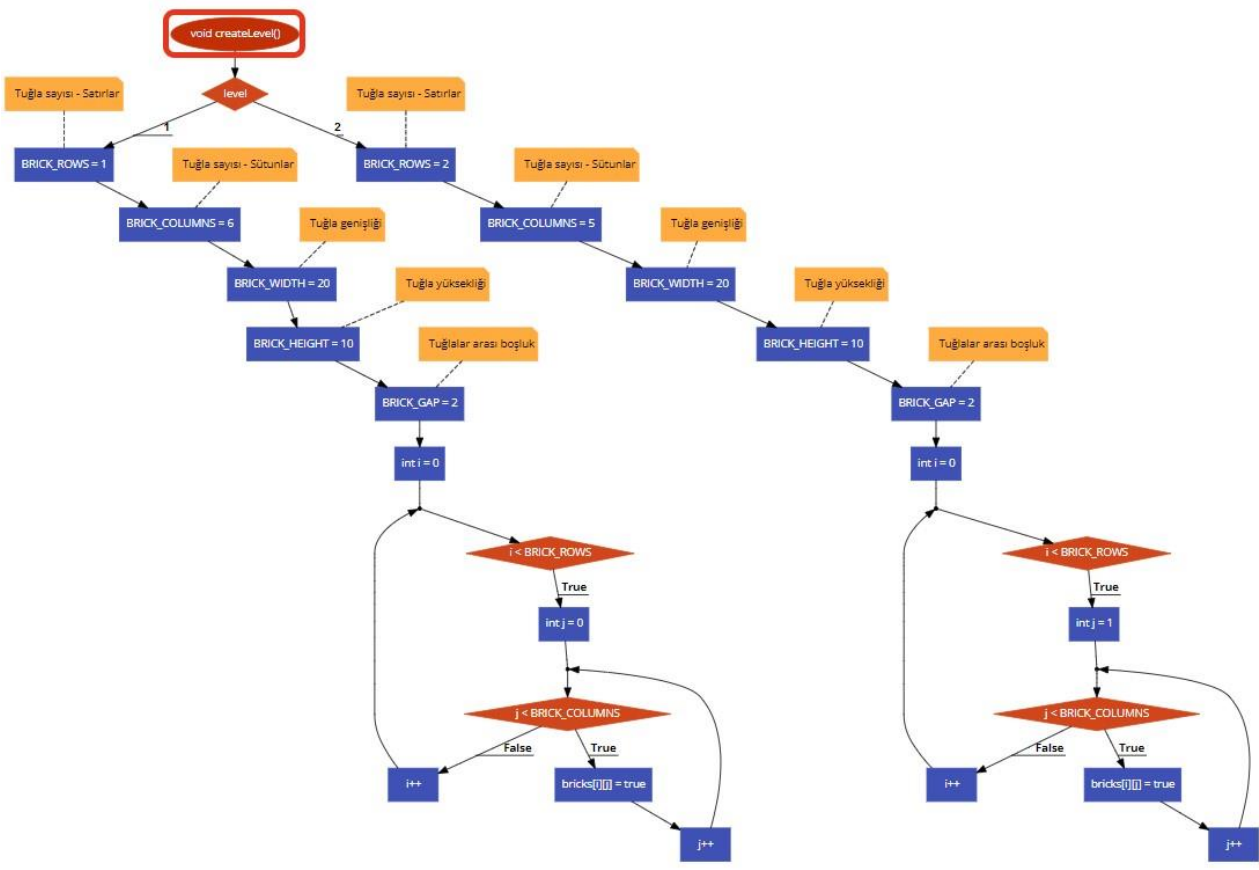
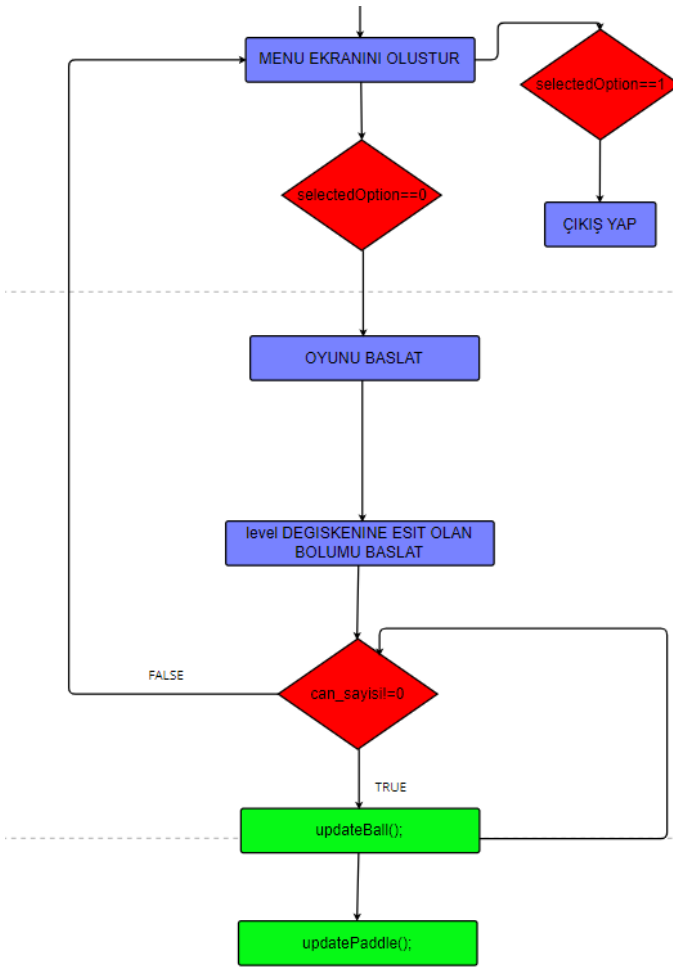
```
- SCREEN_WIDTH: int
- SCREEN_HEIGHT: int
- PADDLE_WIDTH: int
- PADDLE_HEIGHT: int
- PADDLE_SPEED: int
- BALL_SIZE: int
- BALL_SPEED_X: int
- BALL_SPEED_Y: int
- BRICK_ROWS: int
- BRICK_COLUMNS: int
- BRICK_WIDTH: int
- BRICK_HEIGHT: int
- BRICK_GAP: int
- SCREEN_PADDING: int
- BALL_DROP_CHANCE: int
- BUTTON_PIN_1: int
- BUTTON_PIN_2: int
- BUTTON_PIN_3: int
- selectedOption: int
- gameStart: bool
- gameOver: bool
- nextLevel: bool
- level: int
- digit[10]: int
- can_sayisi: int
- brickFallX: int
- brickFallY: int
- brickFallActive: bool
- kirilantugla: int
- ktugla: int
- display: Adafruit_SSD1306

+ setup(): void
+ loop(): void
+ createLevel(): void
+ dis(num: int): void
+ temizle(): void
+ updatePaddle(): void
+ displayHealthLed(): void
+ updateBall(): void
+ skorEkrani(): void
+ resetGame(): void
+ drawFrame(): void
+ drawScreen(): void
+ updateBrickFall(): void
+ tuglasiyahyap(): void
+ tuglabeyazyap(): void
```

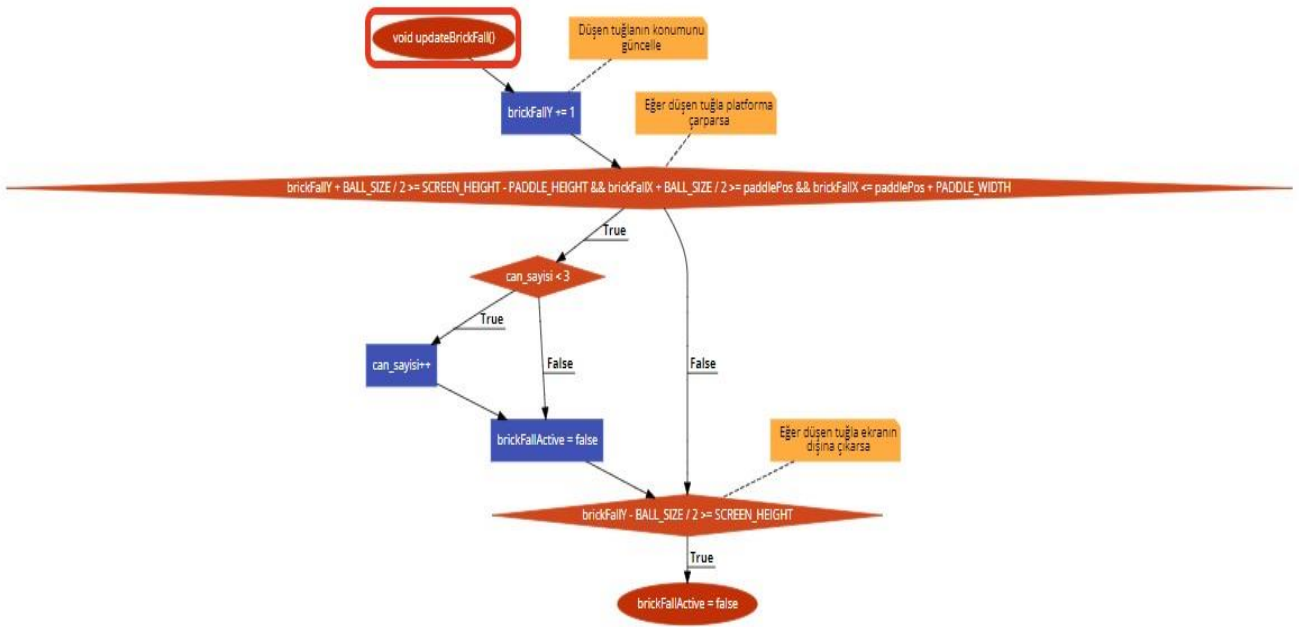
## VI.AKIŞ DİYAGRAMLARI



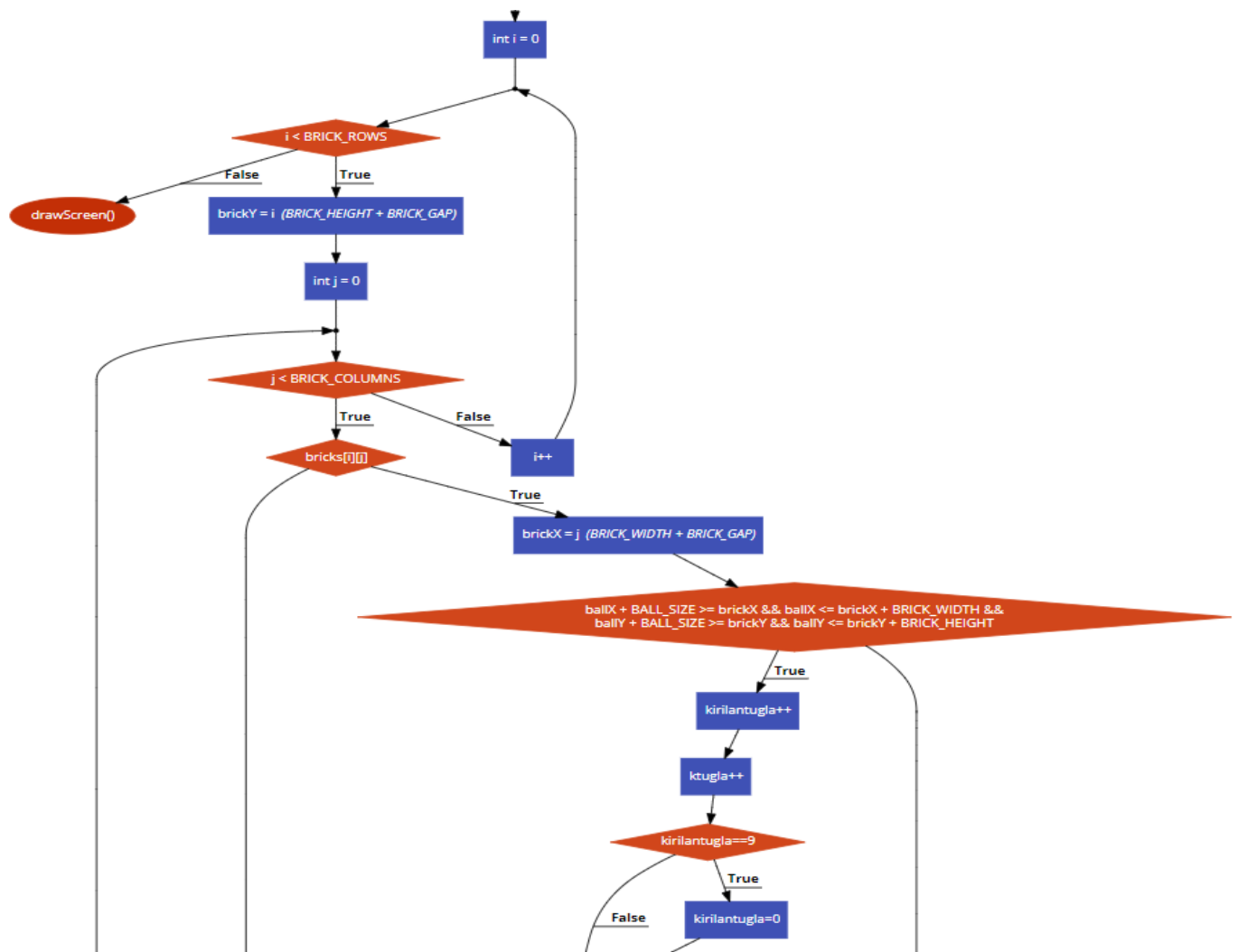


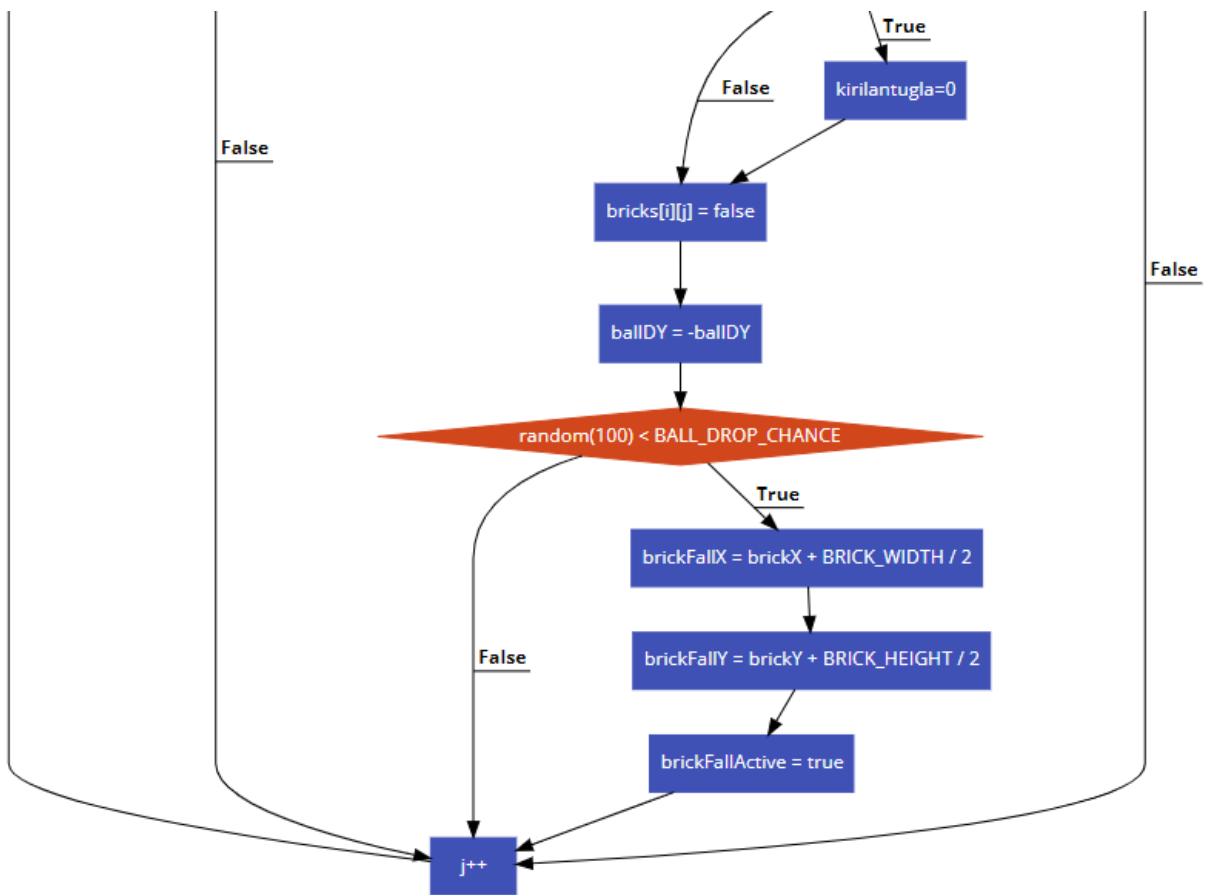












## KAYNAKLAR

- [https://www.youtube.com/watch?v=EqNeh8wu\\_-E](https://www.youtube.com/watch?v=EqNeh8wu_-E)
- <https://www.arduino.cc/reference/en/>