**ARDUINO İLE RFID TABANLI GÜVENLİK SİSTEMİ TASARIMI**

DESIGN OF RFID BASED SECURITY SYSTEM

WITH ARDUINO

**Muhammed KURTULUŞ Ragıp TEMİRTAŞ Mehmet Burak İMREN**

**ABSTRACT**

In this project, magnetic card pass application has been made by using Ardunio Uno development card. In the design, radio frequency and single and automatic recognition of objects were used by using Radio Frequency Identification (RFID) technology. The RFID consists essentially of a label and a reader. RFID-RC522 module is used for recognition.

One of the cards is a master card, the main purpose of the system is to add new cards and delete a card has been defined. In order to identify any magnetic card to the system, the master card must first be assigned to the RC522 module and then the master card must be read in order to open the door and then exit again from the system. Thus, a card can be added to the system which can open the door. To delete the card attached to the system, we do the same operations again; first the master card then the card attached to the system, and then the master card is read again and the card deletion is completed. In this way, the master card can be added to the system by opening and closing the door.

The necessary serial port display shows the necessary routing and information. The source code for the project was prepared using the Ardunio IDE compiler. Finally, the design of the card system was completed by designing the designed system.

**Keywords**: RFID-RC522, Ardunio Uno development board, Ardunio IDE, Master card

**ÖZET**

Bu projemizde, Ardunio Uno geliştirme kartı kullanarak manyetik kartlı geçiş uygulaması yapılmıştır. Tasarımda, Radyo Frekansı ile Tanımlama (RFID) teknolojisinden yararlanılarak, radyo frekansı ile nesneleri tekil ve otomatik olarak tanıma yöntemi kullanılmıştır. RFID, temel olarak bir etiket ve okuyucudan meydana gelir. Tanıma için RFID-RC522 modülü kullanılmıştır.

Kartlardan biri master kart yani asıl amacı sisteme yeni kartlar ekleyip silinmesini sağlayan bir kart tanımlanmıştır. Herhangi bir manyetik kartın sisteme tanımlanması amacıyla öncelikle RC522 modülüne master kart ardından kapıyı açması için tanımlanan kart ve ardından tekrar sistemden çıkış yapabilmesi için master kart okutulması gerekmektedir. Böylelikle sisteme kapıyı açabilen bir kart eklenmiş olur. Sistemde ekli olan kartı silmek içinse tekrar aynı işlemleri yaparız yani; önce master kart ardından sistemde ekli olan kart ve ardından tekrar master kartı okutulup kart silme işlemi tamamlanmış olur. Böylelikle sisteme master kart sayesinde kapının açılıp kapanması sağlayan kartlar eklenip çıkartılabilmiş olur.

Ardunio serial port ekranında gerekli yönlendirmeler ve bilgiler gösterilmektedir. Proje için kaynak kod Ardunio IDE derleyicisi kullanılarak hazırlanmıştır. Son olarak, tasarlanan sistem gerçeklenerek kartlı geçiş sistemi tasarımı tamamlanmıştır.

**Anahtar Kelimeler:** RFID-RC522, Ardunio Uno geliştirme kartı, Ardunio IDE, Master kart

**1.GİRİŞ**

Radio Frequency Identification (RFID) ya da “Radyo Frekanslı Tanımla” şeklinde Türkçeleştirilirse; canlıları ya da nesneleri radyo dalgaları ile tanımlamak için kullanılan teknolojilere verilen genel isimdir. Bir RFID sistemi antenli bir çipten yapılan etiket (tag), ve antenli bir okuyucudan (reader) oluşur. Okuyucu donanım elektromanyetik dalgalar yayar. Etiket anteni bu dalgaları almak için ayarlanmıştır. Pasif bir RFID etiketi, okuyucudan yayılan dalgaları algılar ve bunu mikroçipin devrelerini harekete geçirmek için kullanır. Mikroçip bu dalgalardaki dijital bilgiyi değiştirir ve okuyucuya geri gönderir. Okuyucu RFID etiketinden aldığı radyo dalgalarını dijital bilgiye dönüştürerek bilgisayar sistemine geçmesini sağlar ve bu da işe yarar bir hale gelir.

RFID kullanım alanları insanların hayal gücü ile sınırlıdır. En yaygın uygulamalar tahsilât sistemleri (fatura gibi), havaalanı güvenlik ve bagaj otomasyonu, geçiş kontrol, üretim takibi, otopark otomasyonu ve varlık takibidir. Proje kapsamında RFID, elektromanyetik kartlarla kapı geçiş sisteminde kullanılmıştır. RFID ile kurulan sistem sayesinde evlere, ofislere, okullara girişlerde kullanılabilecek güvenli bir geçiş projesi gerçekleştirilmesi amaçlanmıştır.

Projede bir elektromanyetik kart master kart olarak seçilir ve RFID okuyucuya okutulur. Bu okutulma sonucunda sitem açılır ve sisteme yeni kartların tanıtılması durumuna geçiş yapılır. Tanıtılan her kart güvenli geçiş noktalarında ki kapıların açılması işlemini yapabilir duruma gelir. Kart tanıtma işleminin ardından yeniden master kart kullanılarak sistemden çıkış yapılır ve yalnızca tanıtılan kartların kullanılabildiği bir sistem oluşturulmuş olur. Sisteme tanımlı olmayan bir kartın okutulup geçiş yapılmaya çalışılması durumunda ise sistem buna izin vermeyip, tercih edilebilecek değişik yöntemlerle bu izinsiz girişi ifade edecek şekilde düzenlenir. Projede bu izinsiz geçiş isteği durumu için led kullanılmıştır.

Tasarım projesi kapsamında belirlenen konu Arduino kullanılarak donanımsal ve yazılımsal olarak gerçekleştirilmiştir. Bu kontrol için PC ve Arduino seri port üzerinden haberleşmektedir. Arduino platformunu oluşturan elemanlar Arduino Geliştirme Ortamı (IDE), Arduino Kütüphaneleri, AVRDude, Derleyici ve Optiboot’tur.

Projede kullanılan tüm donanımsal malzemeler ise Arduino Uno R3, RFID-RC522, Board, Jumper, Led’tir.

**2. DONANIMSAL BİLGİLER**

**2.1 Arduino UNO**

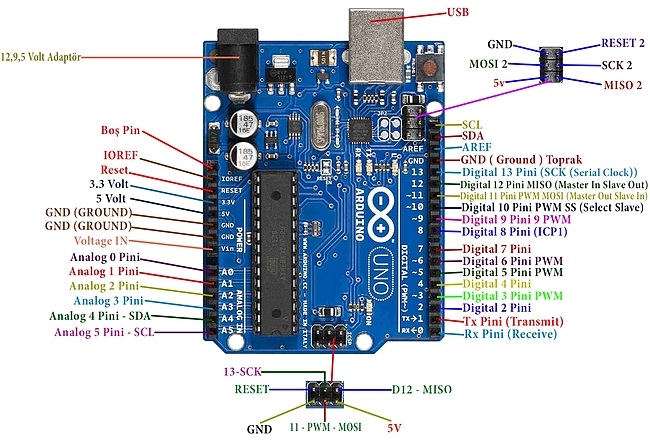
Arduino, temel olarak açık kaynaklı donanıma sahip bir fiziksel programlama platformudur. Arduino tek başına çalışan interaktif nesneler geliştirmek için kullanılabileceği gibi bilgisayar üzerinde çalışan yazılımlara da bağlanabilir.

Arduino geliştirme kartı üzerindeki mikroişlemci arduino programlama dili ile programlanır ve bu program Processing tabanlı Arduino Yazılım Geliştirme Ortamı (IDE) yardımı ile karta yüklenir. Arduino kütüphaneleri birçok işlemi donanım seviyesine inmeden yani kaydediciler üzerinde işlem yapmaya gerek kalmadan yapmayı sağlar. Projede en son güncellenen arduino geliştirme kartlarından biri olan Arduino UNO kullanılmıştır.

Arduino kartları bir Atmel AVR mikrodenetleyici (Eski kartlarda ATmega8 veya ATmega168, yenilerinde ATmega328) ve diğer devrelere bağlantı için gerekli yan elemanlardan oluşur. Arduino kartları üzerinde ATmega firmasının 8 ve 32 bit mikrodenetleyicileri bulunmaktadır. 14 dijital giriş/çıkış pini bulunup, 6’sı PWM çıkışı olarak kullanılabilmektedir. PWM, çıkışta istenilen bant genişliğini ve dalga şeklini belirlemek amacıyla kullanılmaktadır. Günümüzde en çok DC motorların hızlarının ayarlanması için kullanılmaktadır.

ATmega328 mikroişlemcisi:

Arduino kartları üzerinde Atmel firmasının çeşitli mikrodenetleyicileri bulunuyor. Arduino Uno, Mini gibi modellerde Atmega328p’yi görüyoruz. Yüksek performanslı Atmel 8-bit AVR RISC tabanlı mikroişlemci okuma-yazma yeteneklerine sahiptir. ATmega328 Arduino Uno üzerinde bulunduğundan aynı özelliklere sahiptirler.



**Şekil 2.1.** Ardunio Uno pin girişleri.

Arduino uygulama kartı üzerinde DC besleme girişi, reset butonu ve seri programlama portu bulunmaktadır. Bir adet 5 Volt, bir adet 3.3 Volt çıkış pini, üç adet GND pini, bir reset pini, bir adet analog referans pini, bir adet güç giriş pini ve serial programlama pinleri bulunmaktadır. Arduino Uno üç adet led barındırır bunlardan RX ve TX ledeleri seri haberleşme durumunda çalışmaktadırlar. Diğer pin ise 13. Pine bağlıdır ve led uyarısı istenilen durumlardan ayrıca led bağlanmadan kullanılabilir

***Arduino Uno temel donanım özellikleri;***

• Besleme gerilim sınırı 6-20 Volt olup tavsiye edilen ara 7-12 Volt’ tur,

• Dijital giriş/çıkış pinleri 14 pin,

• Analog giriş 6 pin,

• Pin başına 40 mA,

• 32 KB Flash,

• 2 KB SRAM,

• 1 KB EEPROM,

• 16 MHz saat frekansı şeklindedir.

Arduino’nun çalışması için gerekli olan güç USB üzerinden veya güç kaynağı bağlantı noktasından sağlanır.

***Analog:*** Analog giriş pinleridir. Başlarında A harfi (A0, A1 gibi) vardır. Bu pinler sensörler ile iletişim kurulmasını sağlar.

***Digital:*** Arduino UNO ‘da 13 adet digital pin bulunur. Bu pinleri digital giriş ve çıkış için kullanılır.

***PWM:*** “~” işareti gördüğünüz digital pinlerden analog çıkış alınabileceğini belirtir.

***AREF:*** Analog referans pinidir. Arduino regülatörü 1023 adıma sahiptir. Hassas uygulamalarda işe yarayacak bir pindir.

***Reset Butonu:*** Bu butona basıldığında Arduino ve yüklenilen kodlar yeniden başlar.

***Güç Göstergesi:*** Arduinonun çalışıp çalışmadığını bu led sayesinde anlaşılır.

***TX ve RX Ledleri:*** İşlemciye yazılım yüklerken bu ledlerin yanıp söndüğünü görülecektir. Seri iletişimi ifade eder. TX verici RX ise alıcıdır. Aynı harfler 0 ve 1 nolu pinlerde göze çarpacaktır. Bu pinler seri iletişim için kullanılabilir.

***İşlemci:*** ATMEL firmasının ATmega işlemcisi. Yazılan programlar bu işlemciye gönderilir.

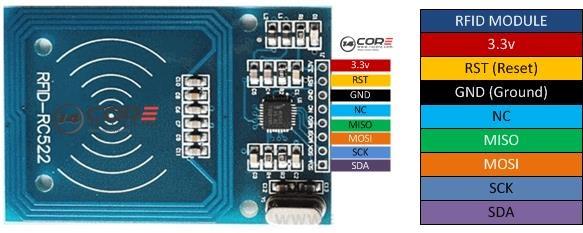
***Regülatör:*** Devreye zarar verebilecek gerilim değerlerinden korur.

Arduino Uno barındırdığı ATmega 328 sayesinde 32 Kb lık bir hafızaya sahiptir. Ancak bunun 0.5 Kb lık kısmı Arduino bootloaderı tarafından kullanılmaktadır. Ayrıca Arduino Uno 2 Kb RAM ve 1 Kb EEPROM barındırır.

Projede Arduino ile seri haberleşme prensibiyle haberleşilmektedir. Seri iletişim haberleşmenin tek bir hat üzerinden 1 ve 0’lar şeklinde yapılması prensibine dayanmaktadır. Arduino üzerindeki UART birimiyle bu seri haberleşme işlemi yapılmaktadır. Arduino içinde bu seri haberleşme için Serial kütüphanesi bulunmaktadır. Program yazılıp derlendikten sonra iletişimin nasıl yapıldığı Arduino IDE’sinde bulunan Serial Monitor kısmından takip edilebilmektedir. Yapılan seri haberleşmenin iletim hızı baud hızı sekmesinden veya fonksiyonundan ayarlanabilmektedir. Serial Monitor yardımıyla Arduino’ya veri yollanabilir ve aynı şekilde veri alınabilmektedir.

**2.2. RFID-RC522 Kart Okuyucu Modül**

Projede RFID-RC522 modülü kullanılmıştır. Radyo Frekans ile Tanımlama (RFID) basit anlamda; herhangi bir nesnenin üzerinde bulunan ve o nesneye ait bilgiyi içeren elektronik bir etiket (TAG) ile radyo frekansı üzerinden kendisini tanıtacak bir alıcı (OKUYUCU) arasında kurulan bir tanımlama teknolojisidir. RFID Okuyucu, Tag'ların belleklerindeki bilgiyi okumak, yazmak veya değiştirmek için kullanılır. Projede kartlar EEPROM tarafından okunurken karların ID’si RFID ile taranmaktadır. Master kart olarak tanımlanan sistemi açan kartın ID’si de EEPROM’a aktarılmaktadır. Arduino ve diğer mikrodenetleyicilerle kullanılabilir özelliktedir. Şekil 2.3’te verilen RFID-RC522 modülünün çalışma gerilimi 3.3 Volttur.



**Şekil 2.2.** RFID-RC522 pin girişleri.

Çalışma frekansı 13.56 MHz, çalışma akımı 13-26 mA’dir. Haberleşme protokolü olarak SPI (Seri Çevresel Arayüz) kullanmaktadır ve haberleşme hızı 424 kbit/s’dir. Kartın boyutları 40x60 mm’dir. Sekiz adet bağlantı pin adedi bulunmaktadır.

**3. Proje Devresinin Tasarımı ve Programlanması**

**3.1. Devre Tasarımı**

Kartlı kapı geçiş sistemi projesinin devre tasarımı Şekil 2.4’te verlmiştir. Devrenin Led’ler, RFID ve Arduino kullanılarak nasıl bağlantılarla oluşturulduğu açıkça belirtilmiştir.

Devre tasarımında ve uygulamasında kullanılan malzemeler ise şöyledir;

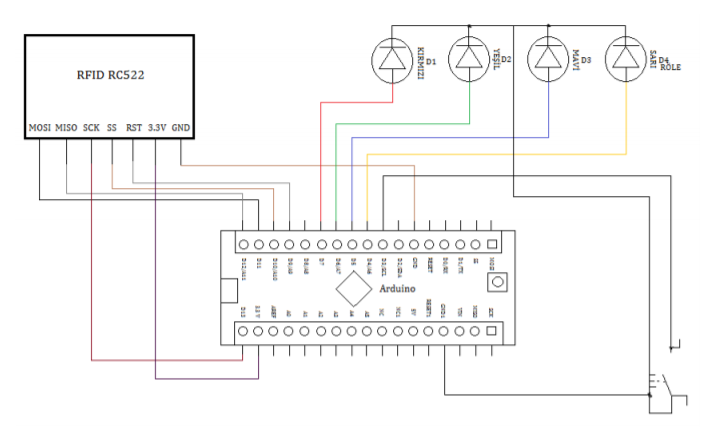
1. RFID-RC522 (1 Adet)

2. Elektromanyetik Kartlar (Master Kart ve Diğer Kartlar)

3. Board (1 Adet)

4. Jumper kablolar

5. Led (4 Adet)



**Şekil 3.1.** Proje bağlantı şeması.

**3.2. Sistemin Programlanması**

Projenin tamamında Arduino geliştirme ortamı kullanılmıştır. Arduino’nun kendi programlama temelleri dikkate alınarak program kodu yazılmıştır.

Proje kapsamında Arduino programına gerekli kütüphanelerin eklenmesiyle başlanmıştır. Arduino ile Seri Port üzerinden haberleşme yapıldığında okuma ve yazma işlemlerinin kaydı EEPROM’a yapılacağından ilgili EEPROM.h kütüphanesi programa include edilir. Ayrıca RFID kullanmak için gerekli olan MFRC522.h kütüphanesi ve yine RFID’in kullandığı protokol SPI protokolü için gerekli olan SPI.h kütüphanesi programa include edilir. Bu üç kütüphaneden MFRC522.h internet üzerinden elde edilip Arduino klasörünün içerisine atılarak kullanılabilmektedir. Diğer kütüphaneler Arduino IDE’sinin içinde bulunmaktadır. Arduino kodları C++ kodlarıyla yazıldığından tüm kütüphaneler .h ve .cpp uzantılıdır.

***Kitaplık tanımlamalarının yapıldığı program bölümü;***

#include <EEPROM.h> **//Okuma ve yazma işini UID ile EEPROM a kaydetme işlemi.**

#include <SPI.h> **//RC522 modülü SPI protokolünü kullanır.**

#include <MFRC522.h> **//Mifare RC522 kütüphanesi**

Kütüphanelerin include edilmesinin ardından programın genelinde kullanılan değişkenlerin, yapılan pin bağlantılarının define tanımlamaları yapılmıştır.

***Kullanılan değişkenlerin ve yapılan pin bağlantılarının define tanımlamaları bölümü;***

#define COMMON\_ANODE

#ifdef COMMON\_ANODE

#define LED\_ON HIGH

#define LED\_OFF LOW

#else

#define LED\_ON LOW

#define LED\_OFF HIGH

#endif

#define kirmizi 7 //**Led pinleri seçildi.**

#define yesil 6

#define mavi 5

#define relay 4 //**Röle pini seçildi.**

#define buton 3 //**Master kartını silmek için buton pini seçildi.**

boolean match=false;

boolean programMode=false; //**Programlama modu başlatma.**

int successRead; //**Başarılı bir şekilde sayı okuyabilmek için integer atıldı.**

byte storedCard[4]; //**Kart EEPROM tarafından okundu.**

byte readCard[4]; //**RFID modül ile ID tarandı.**

byte masterCard[4]; //**Master kart ID'si EEPROM'a aktarıldı.**

#define SS\_PIN 10

#define RST\_PIN 9

MFRC522 mfrc522(SS\_PIN,RST\_PIN); //**MFRC522 örneği oluşturuldu.**

Arduino’nun programlanması kendi yazılım düzeninde olan iki temel fonksiyonun koşulmasıyla gerçekleştirilmektedir. Bunlar setup() ve loop() fonksiyonlarıdır. setup() fonksiyonu Arduino başlatıldığında yalnızca bir kez koşulurken loop() fonksiyonu Arduino açık kaldığı sürece kendini sürekli çağırarak sonsuz döngü içerisinde çalışmaktadır. Uygulama yazılırken genel ayarlar, değişken atamaları vs. gibi işlemler setup() içerisinde yapılmaktadır. Arduino’nun asıl yapmasını istediğimiz kodlar ise loop() içerisinde döngü yapısına uygun olacak şekilde yazılmaktadır.

Projede program başlatıldığı zaman ilk olarak çalışması istenen özellikler ve işlevler setup() kısmında programlanmıştır. Sistem ilk çalıştığı zaman pinMode() fonksiyonu ile bağlanan ledlerin ve diğer pinlerin çalışma modları (giriş/çıkış) atanmıştır. Başlangıçta yalnızca kapının kilitli durumda olduğunu belirtmek amacıyla sarı led (röle) HIGH modunda yanıyor olarak ayarlanmıştır. Ayrıca baud hızı Serial.begin (9600) satırıyla 9600 değerine setlenmiştir. Protokollerin düzgün çalışması için SPI.begin() ve mfrc-c522.PCD Init() fonksiyonları setup() bölümünde çağrılmıştır.

***Kullanıcı EEPROM’daki bilgilerin sıfırlanmasını-silinmesini isterse reset butonuna basar ve şu kod kısmı çalışır;***

if(digitalRead(buton)==LOW)

{**//Butona basıldığında EEPROM içindeki bilgileri siler.**

digitalWrite(kirmizi,LED\_ON);

**//Kırmızı led, sildiğimizi bilgilendirmek amacıyla yanık kalır.**

delay(5000);

if (digitalRead(buton)==LOW)

{

Serial.println(F("EEPROM silinmeye baslaniyor."));

for (int x=0; x<EEPROM.length();x=x+1)**//EEPROM adresinin döngü sonu.**

{

if(EEPROM.read(x)==0)**//EEPROM adresi sıfır olursa.**

{

}

else

{

EEPROM.write(x,0);

}

}

Serial.println(F("EEPROM Basariyla Silindi.."));

digitalWrite(kirmizi,LED\_OFF);

delay(200);

digitalWrite(kirmizi,LED\_ON);

delay(200);

digitalWrite(kirmizi,LED\_OFF);

delay(200);

digitalWrite(kirmizi,LED\_ON);

delay(200);

digitalWrite(kirmizi,LED\_OFF);

}

else

{

Serial.println(F("Silme islemi iptal edildi."));

digitalWrite(kirmizi,LED\_OFF);

}

}

Silme işlemi sırasında kırmızı led yanarken, silme işlemi gerçekleştirilince yanıp sönmeyi 2 saniye aralıklarla gerçekleştirilmektedir.

setup() bölümünde son olarak sistem ilk çalıştırıldığında kontrol edilmesi istenen özellik sistemde bir Master kartın tanımlı olup olmadığının kontrolüdür. Eğer sistemde bir master kart bulunmamaktaysa yeni bir master kart kaydı yapılmaktadır.

EEPROM genel çalışma mantığı gereği EEPROM kayıtlarının haricinde 143 EEPROM adresini tutabilmektedir. Bu sebeple 143 sayısı EEPROM’da bilgiler bulunup bulunmadığının kontrolünde de kullanılmaktadır.

***Sistemde master kart bulunuyor mu bulunmuyorsa master kartın tanıtılması işlemi şu kod parçasıyla yapılmaktadır;***

if (EEPROM.read(1) != 143)

{

Serial.println(F("Master Kart Secilmedi."));

Serial.println(F("Master Karti secmek icin kartinizi okutunuz.."));

do

{

successRead=getID();

**//successRead 1 olduğu zaman okuyucu düzenlenir aksi halde 0 olacaktır**.

digitalWrite(mavi,LED\_ON);

**//Master kartın kaydedilmesi için gösterildiğini ifade eder.**

delay(200);

digitalWrite(mavi,LED\_OFF);

delay(200);

}

while(!successRead);

**//Başarılı bir şekilde okuyamadıysa başarılı okuma işlemini** **yapmayacaktır.**

for (int j=0; j<4; j++)**//4 kez döngü.**

{

EEPROM.write(2+j,readCard[j]);**//UID EPPROM’a yazıldı.**

}

EEPROM.write(1,143);**//EEPROM’a master Kartı kaydettik**.

Serial.println(F("Master Kart kaydedildi.."));

}

Kod parçasından da anlaşıldığı üzere Master kart okunması gerçekleşmediği durumda mavi ışık yanıp sönmektedir. Master Kart okunduğu zaman EEPROM’a UID’si yazılmaktadır ve kayıt gerçekleştirilmektedir. Böylece setup() bölümünde sistem ilk açıldığında gerçekleştirilmesi istenen işlevler gerçekleştirilmiş olmaktadır. Master kartı okuyan sistem artık yeni kartları okuyup güvenli geçişi sağlayabilecek konuma gelmiş bulunmaktadır. Kullanıcıya hazır durumunu bildiren bir mesajda iletilmektedir.

Programın loop() kısmında sistem her açık olduğu an yapılması istenen durumların kodları yazılmaktadır. Sistemin kart okutulmasını beklediği duruma program mod, kart okutulmasını beklemediği duruma normal mod denilerek kodlama başlanmaktadır. Programın bu aşamasında master kartın sisteme yeniden okutulması durumunda sistemden çıkılmaktadır. Bu sebeple okutulan kartın master kart olup olmadığı kontrolünün de yapılması gerekmektedir. loop() kısmında kullanılan fonksiyonlardan getID() fonksiyonu kart okunmasında kullanılan hazır bir fonksiyondur. master() fonksiyonu ise master kartın doğruluğunun test edildiği fonksiyondur. Okutulan kart master kart değilse başka bir karttır. Bu kart ya sisteme kayıt edilmek isteniyordur ya da sistemden silinmek isteniyordur. Bu durumun sağlanması için kullanılan iki fonksiyon bulunmaktadır. Okunan kart silinmek isteniyorsa deleteID() fonksiyonu, okunan kart sisteme kayıt edilmek isteniyorsa writeID() fonksiyonu kullanılmaktadır. Kart tanıtma işlemleri bu kısımlarla son bulmaktadır.

Master kartın sisteme okutulmasının ardından sistem açılmaktadır. Açılan sistemde kullanıcıya ilk olarak sistemde kayıtlı olan kullanıcı sayısı bildirilmektedir. Ardından okutulan herhangi bir kartın ID’si sistemde bulunuyorsa kapı açılmaktadır, sistemde bulunmuyorsa ise geçişe izin verilmemektedir.

***Geçiş izninin verilmesi durumunda izinverildi() fonksiyonu, verilmemesi durumunda ise izinyok() fonksiyonu çalıştırılmaktadır;***

void izinverildi(int setDelay)

{

digitalWrite(mavi,LED\_OFF);**//mavi led sönük.**

digitalWrite(kirmizi,LED\_OFF);**//kırmızı led sönük.**

digitalWrite(yesil,LED\_ON);**//yeşil led yanık.**

digitalWrite(relay,LOW);**//kapı kilidi açıldı.**

delay(setDelay);**//kapının açılması için zaman kazanıyoruz.**

digitalWrite(relay,HIGH);**//kapı kilitlendi.**

delay(1000);**//1 saniye sonra yeşil led de sönecektir.**

}

Bu fonksiyon çalıştırıldığında mavi ve kırmızı led sönük, yeşil led yanar durumdadır. Role değişkenide LOW değerine setlenerek kapı açılır ve gecikme değişkenine atanan değer kadar beklenir. Ardından role HIGH pozisyonuna getirilerek kapı kapanmaktadır.

void izinyok()

{

digitalWrite(yesil,LED\_OFF);//**yeşil led sönük**

digitalWrite(mavi,LED\_OFF);//**mavi led sönük**

digitalWrite(kirmizi,LED\_ON);//**kırmızı led yanık**

delay(1000);

}

şeklinde düzenlenerek kartın sahibi kullanıcıya izin verilmediğinde sadece kırmızı ledin yanması sağlanmaktadır.

loop() fonksiyonunun içerisinde bu durumların çalışması sağlanmaktadır. Genel olarak mantık bu şekilde olmakla birlikte program genelinde loop() fonksiyonunun içerisinden çağrılan bazı fonksiyonlar ve işlevleri aşağıdaki şekildedir;

getID() fonksiyonu kart okunmasında kullanılan hazır olarak alınan bir fonksiyondur.

showReaderDetails() fonksiyonu da ayrıntıların gösterilmesinde kullanılan bir diğer hazır fonksiyondur. Program genelinde hazır olarak kullanılan yalnızca bu iki fonksiyon bulunmaktadır.

programMod() fonksiyonu sistem program modundayken kullanılan fonksiyondur. Sistem bu moddayken yeşil, mavi, kırmızı ledler sırasıyla yanıp sönmektedir.

normalMod() fonksiyonu ise program normal mod olarak tanımlanan modda iken kullanılan fonksiyondur. Bu modda mavi led yanarken program kart okumaya hazır şekildedir. role HIGH pozisyonunda ve kapı kilitli halde bulunmaktadır.

***readID() fonksiyonu EEPROM içerisinden ID okuma işlemi yapılmasını sağlamaktadır;***

void readID( int number )

{

int start=(number\*4)+2;//**başlama pozisyonu.**

for (int i=0; i<4; i++)//**4 byte alabilmek için 4 kez döngü kuruldu.**

{

storedCard[i]=EEPROM.read(start+i);

}

}

***writeID() fonksiyonu EEPROM içerisine ID ekleme durumunda kullanılan fonksiyondur;***

void writeID(byte a[])

{

if (!findID(a))

{

int num=EEPROM.read(0);

int start= (num\*4)+6;

num++;

EEPROM.write(0,num);

for(int j=0;j<4;j++)

{

EEPROM.write(start+j,a[j]);

}

successWrite();

Serial.println(F("Basarili bir sekilde ID kaydi EEPROM'a eklendi.."));

}

else

{

failedWrite();

Serial.println(F("! Yanlis ID veya kullanici sinirina ulasildi."));

}

}

***findID() fonksiyonu EEPROM içerisinde ID bulma işlemi gerçekleştirileceği zaman kullanılmaktadır;***

boolean findID (byte find[] )

{

int count = EEPROM.read(0);**//EEPROM'daki ilk byte'ı oku.**

for(int i=1; i <= count; i++)**//Önceden giriş yapılmış mı kontrolü.**

{

readID(i);

if(checkTwo(find,storedCard) )

{

return true;

break;

}

else

{

//değilse return false

}

}

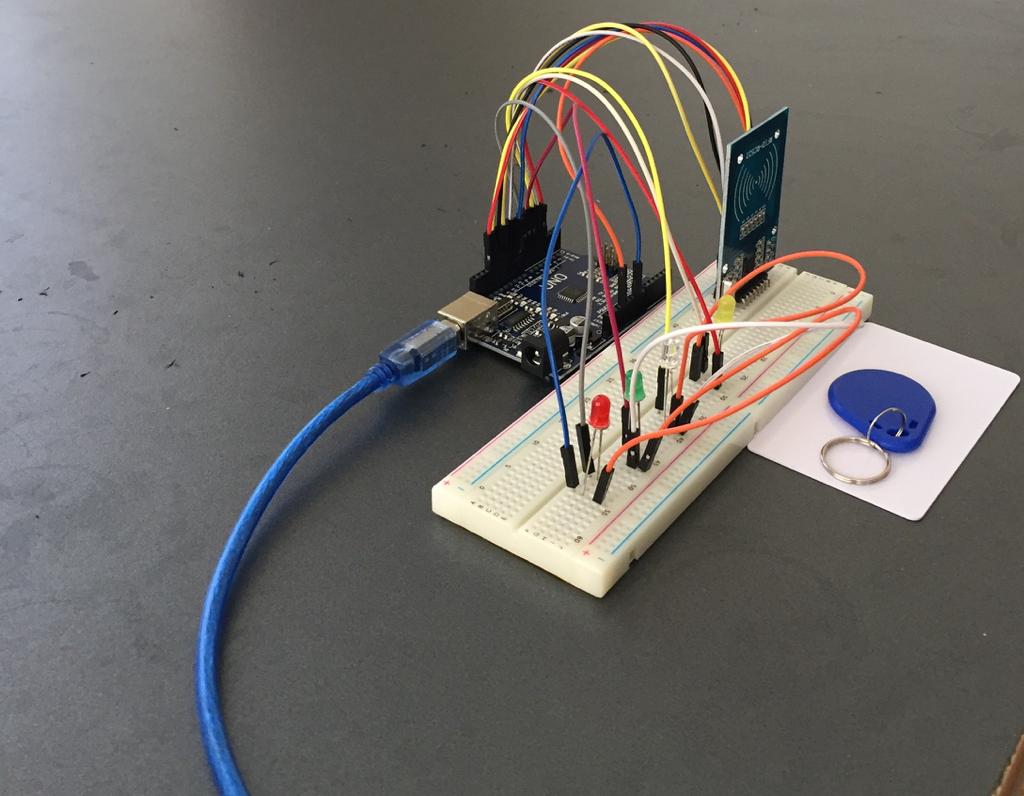
return false;

}

successWrite() fonksiyonu EEPROM’a kayıt başarılı şekilde yapıldığı zaman çağrılmaktadır. Bu fonksiyon çağrılınca ilk olarak mavi, kırmızı, yeşil ledler sönük hale gelmektedir. Ardından yeşil led yanıp sönmektedir.

failedWrite() fonksiyonu EEPROM’a kayıt işlemi başarısız olunca kullanılan fonksiyondur. Kayıt başarısız olunca en başta hepsi sönen ledlerden kırmızı led yanıp sönmeye başlamaktadır.

successDelete() fonksiyonu kart silme işlemi gerçekleştiği zaman kullanılan fonksiyondur. İlk olarak tüm ledler sönük durumdayken işlem sonucunda mavi led yanıp sönmeye başlamaktadır.



**Şekil 3.2.** Hazırlanan devrenin çalıştırılması.

**4. SONUÇLAR**

Proje, giriş yapılan binalarda veya herhangi bir kapının ardındaki ortam güvenliğine vurgu yapmaktadır. Günümüzde artan hırsızlık olayları, bina güvenliği ihtiyacı gibi sebeplerle önlemlerin nasıl alınabileceği konusu önemli hale gelmiştir. Bu güvenliğin sağlanabilmesi için de çeşitli yöntemler öne sürülmektedir. Proje dahilinde günümüz ihtiyacının giderilmesi için bir yöntem önerilmiştir. Güvenliği istenilen ortama yalnızca izin verilen kişilerin girebilmesi için bir sistem geliştirilmiştir.

Proje kapsamında çeşitli yazılı ve internet kaynaklarından yararlanılarak Arduino kullanımı, bu donanımın kullanılabilmesi için yazılımın nasıl kullanılacağı ve bağlantıların nasıl yapılacağı öğrenilmiş oldu. Bu edinilen bilgiler ışığında yazılım ve donanım kısmının bağlantısı sağlanarak projenin tamamlanmasından sonra kurulan sistem sayesinde her binada, her güvenli geçiş gereken yerde kullanılabilecek bir kartlı geçiş sistemi oluşturuldu.

Devre tasarımı ve programlanması başarıyla uygulamaya geçirilip istenilen hedeflere ulaşıldı.

**KAYNAKLAR**

<https://www.nxp.com/docs/en/data-sheet/MFRC522.pdf>

<https://playground.arduino.cc/Learning/MFRC522/>

<https://datasheet.octopart.com/A000066-Arduino-datasheet-38879526.pdf>

<https://blog.direnc.net/rc522-rfid-okuyucu-ile-role-kontrolu/>