

# ARDUINO RF-ID DOOR LOCK

2014555054  
2014555055  
2014555057

**EBRU SARI**  
**MAHMUT SAT**  
**DERYA ŞAHİN**

*ebrusari321@gmail.com*  
*mmahmutsat@gmail.com*  
*ssahinderya@gmail.com*

## Abstract

When this simple circuit get motion, then it will read the serial number on the card or tag in 5 seconds, display it in the serial monitor, sound a buzzer and light up some LEDs its condition.

## 1. INTRODUCTION

If you have worked in an office before, you are probably aware of those RFID cards or tags that would unlock specific doors depending on your access.

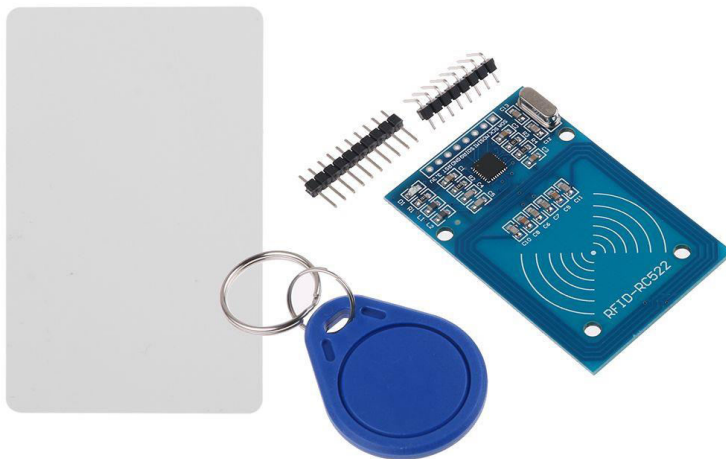
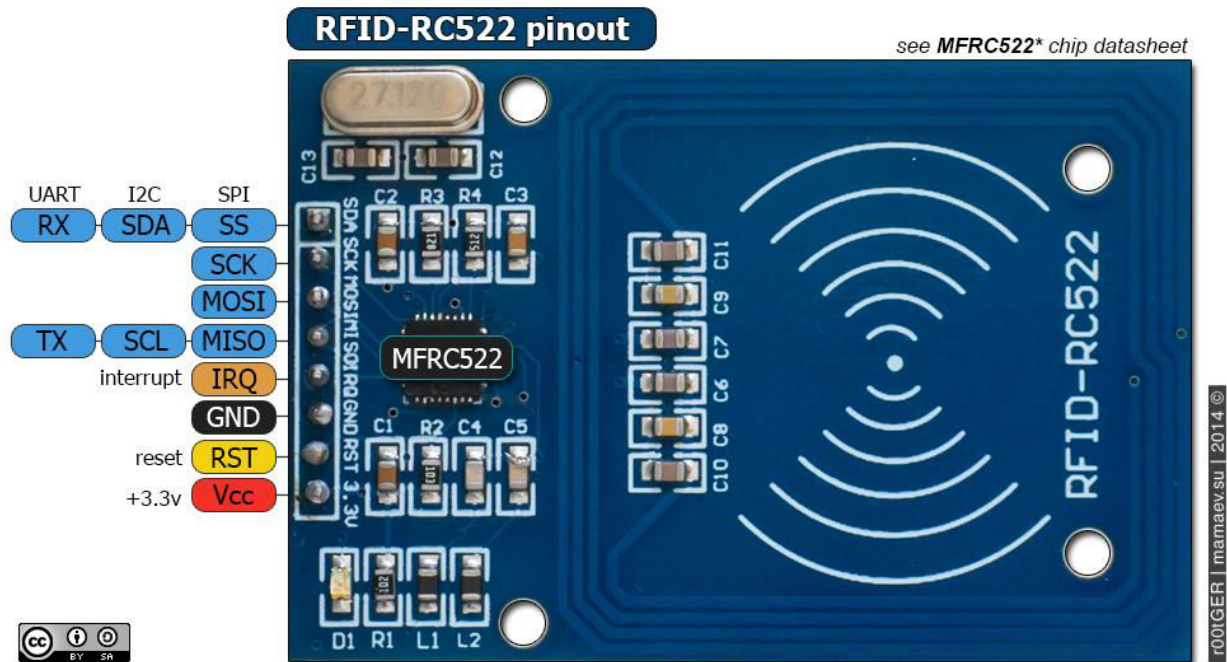
Let's make a simple Arduino RFID reader using the RC522 module, and program the Arduino to do something when the right card is detected. When this simple circuit get motion, then it will read the serial number on the card or tag in 5 seconds, display it in the serial monitor, sound a buzzer and light up some LEDs its condition.

Of course this project could be interfaced to open doors, switch on a relay, light up an LED, or anything else.

## 2. MATERIALS AND METHODS

Here are the electronic component:

### 1. RC522 RF-ID NFC Kit



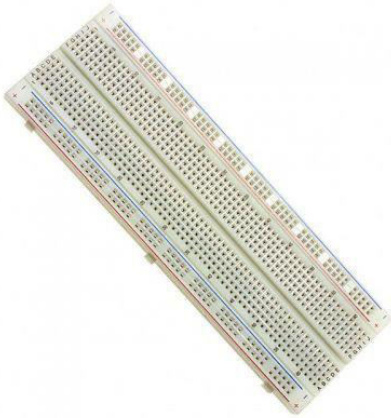
The RC522 RFID card is a small size card with low power consumption that can read and write on tags operating at a frequency of 13.56 MHz, which is NFC frequency. It can be easily used with many microcontroller platforms, especially Arduino. It has a communication speed of 424 kbit / s. It supports different types of encryption on RFID.

## 2. Arduino UNO R3 Clone



Arduino UNO R3; is the latest version of Arduino which contains the CH340 USB-Serial converter.

## 3. Breadboard



2 power buses have 60 columns, 10 lines total 630 holes.

## 4. Buzzer



It can use to get sound output on our projects.

## 5. PIR Motion Sensor



PIR sensors are sensors that are used to detect live motion in an environment.

## 6. LED



It operates at the voltage between 1.5-3V. It must be used with the necessary resistances for voltage values of 5V and above.

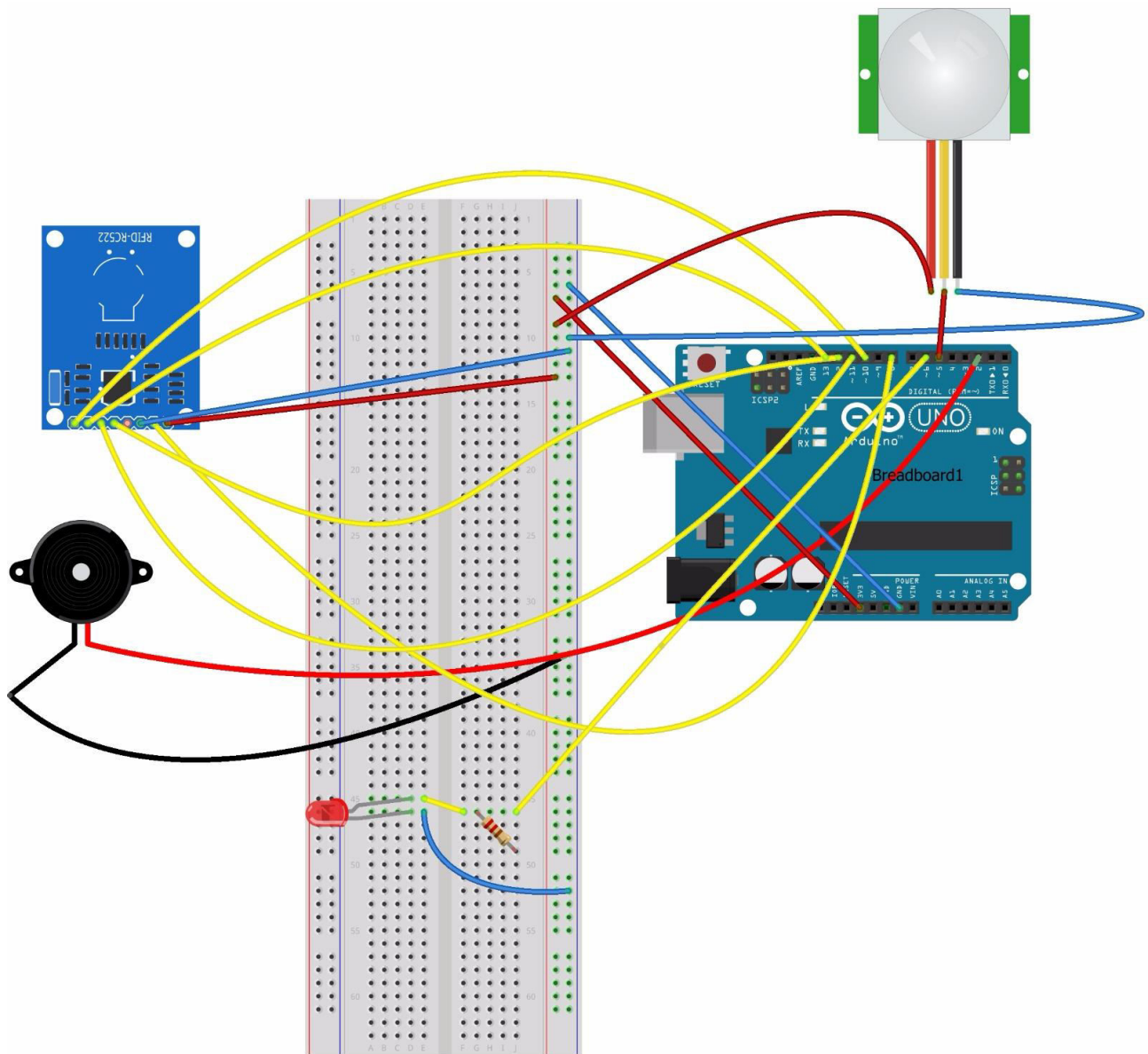
## 7. Resistance



It is a 150K resistor with resistance to 0.5W.

## 8. Cables





fritzing

Here are the connection needed for this tutorial:

The GND of the UNO is connected to the negative (-) pin of the buzzer, GND pin of the RFID Module and GND pin of the sensor to the negative (-).

The 3.3V pin of the UNO is connected to the VCC pin of the LED and the 3.3V pin of the RFID Module.

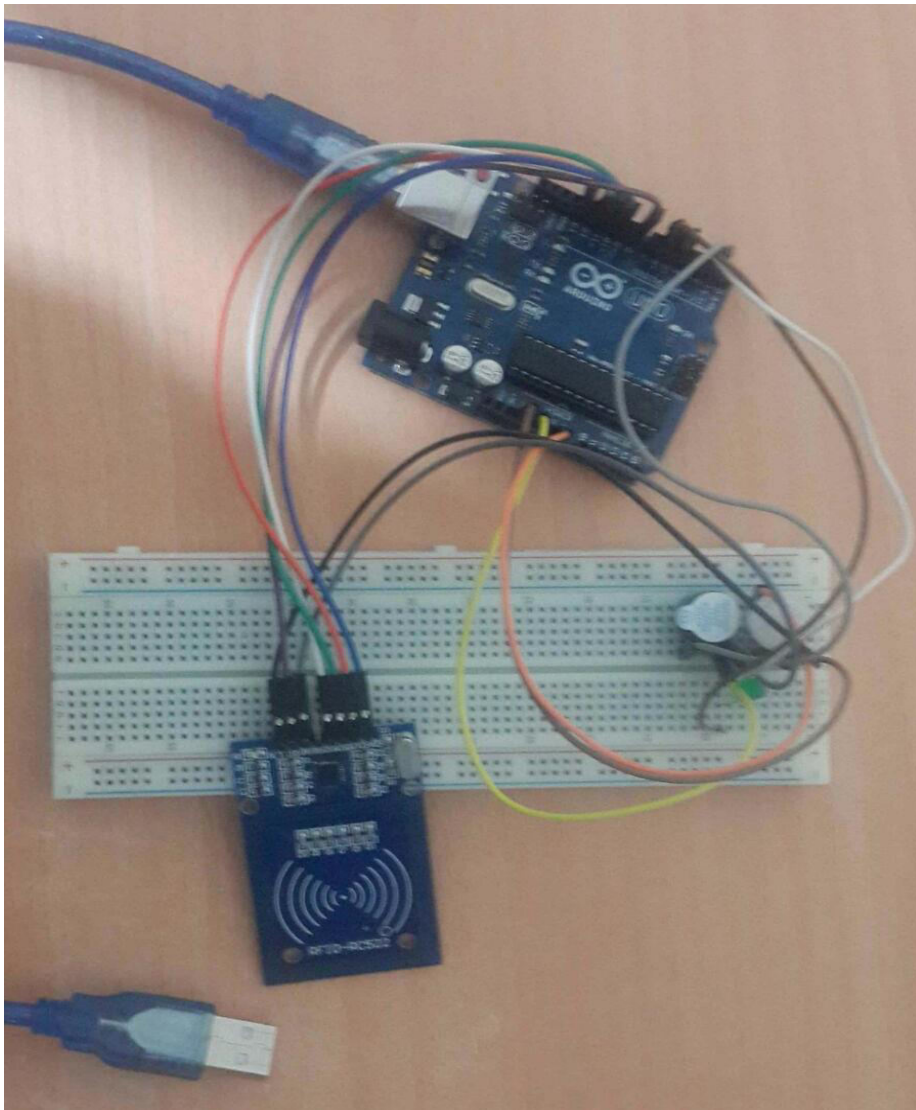
Pin 8-10-11-12-13 of the UNO are connected to pin RST-SDA-MOSI-MISO-SCK of the RFID Module.

Pin 5 of the UNO is connected to motion sensor.

Pin 6 of the UNO is connected to sensor LED.

Pin 2 of the UNO is connected to the (+) pin of the Buzzer.





We are using the MFRC522 library to communicate with RC522 module. The code will read the serial number of the card when it gets motion and it's close enough to be energized by the RC522 reader, and then compare that information to the value that we entered before. If the serial numbers match, then the buzzer will sound a short tone and light up the LED, if not then a longer tone will be sounded and the LED will be red. The card that entered room will be read again and that means the user goes out.

\*\*\*\*\*

Kapı açıldı

Kart okutulması için bekleniyor...

Kartın UID'sini taratin...

9A32AE29

Hosgeldin, gecis izni verildi.

BASARILI

COM3 (Arduino/Genuino Uno)

\*\*\*\*\*

Kapı açıldı

Kart okutulması için bekleniyor...

Kartın UID'sini taratin...

9A32AE29

Hosgeldin, gecis izni verildi.

BASARILI

Kartın UID'sini taratin...

9A32AE29

Alarm Kuruluyor....

5....

4....

3....

2....

1....

Alarm Kuruldu....

☐ Otomatik Kaydetme

Hosgeldin, gecis izni verildi.

BASARILI

Kartın UID'sini taratin...

9A32AE29

Alarm Kuruluyor....

5....

4....

3....

2....

1....

Alarm Kuruldu....

\*\*\*\*\*

Kapı açıldı

Kart okutulması için bekleniyor...

Kartın UID'sini taratin...

9A32AE29

Hosgeldin, gecis izni verildi.

BASARILI

### 3. CONCLUSION

This project helps us about basic security problems. It uses identification and provides safe entry. It can be used payments and phones.

As a conclusion RFID eases entry and it makes do safely. RFID is a solution of security problems. Today this circuit uses everywhere and unlocks specific doors with right card. If the master card will shown, after we can add a new card or we can delete the card added before. Maybe RFID door lock utilization can be increase a lot.



## REFERENCES

- [1] <https://brainy-bits.com/blogs/tutorials/card-reader-with-an-arduino-rfid-using-the-rc522-module>
- [2] <http://roboturka.com/arduino/arduino-ile-pir-hareket-sensoru-uygulamasi/>
- [3] <http://www.microsolution.com.pk/product/rfid-reader-pakistan/>
- [4] <https://www.robotistan.com/rc522-rfid-nfc-kiti-rc522-rfid-nfc-modulu-kart-ve-anahtarlik-kiti-1356>
- [5] <https://www.robotistan.com/arduino-uno-r3-klon-usb-kablo-hediyeli-usb-chip-ch340>
- [6] <https://www.robotistan.com/breadboard-2>
- [7] <http://rees52.com/434-piezoelectric-buzzer-b10.html>
- [8] <https://www.robotistan.com/buzzer>
- [9] <https://www.robotistan.com/mini-pir-hareket-algilama-modulu>
- [10] <http://www.goldmine-elec-products.com/prodinfo.asp?number=G19084>
- [11] <http://uk.rs-online.com/web/p/through-hole-fixed-resistors/0132494/>
- [12] <https://www.robotistan.com/adan-bye-usb-kablosuyazici-kablosu-50cm>
- [13] <https://www.robotistan.com/40-pin-ayrilabilen-disi-erkek-m-f-jumper-kablo-200-mm>

## APPENDIX

### A. PROJECT SOURCE CODE

```
#include <EEPROM.h>      //Okuma ve yazma işini UID ile EEPROM a kaydedicez.
#include <SPI.h>          //RC522 modülü SPI protokolünü kullanır.
#include <MFRC522.h>      //Mifare RC522 kütüphanesi

#define RESET 3//Master Card'ı silmek için buton pinini sseçtik.(3 ü direk grounda
baglayarak resetleyebiliriz.)
#define BUZER 2
#define SENSOR 5
#define sensorLed 6
boolean match=false;
boolean programMode=false; //Programlama modu başlatma.

int successRead=0 ;      //Başarılı bir şekilde sayı okuyabilmek için integer atıyoruz.
int sensorDeger = LOW;
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 8
MFRC522 mfrc522(SS_PIN,RST_PIN);

////////////////////YÜKLEME////////////////////////////////////

void setup()
{
    pinMode(RESET,INPUT_PULLUP);
    pinMode(BUZER,OUTPUT);
    digitalWrite(BUZER,LOW);
    pinMode(SENSOR,INPUT);
    pinMode(sensorLed, OUTPUT);
    sensorDeger=LOW;

    //Protokol konfigrasyonu
    Serial.begin(9600);    //PC ile seri iletim başlat.
    SPI.begin();           //MFRC522 donanımı SPI protokolünü kullanır.
    mfrc522.PCD_Init();    //MFRC522 donanımını başlat.

}
void loop()
{
    delay(1000);
```

```
sensorDeger=LOW;
sensorDeger=digitalRead(SENSOR);
```

```
if(sensorDeger==HIGH)
{
    start();
    digitalWrite(sensorLed,HIGH);
    successRead=getID();
    for(int i = 0;i<20;i+=1)
    {
        if(i<10)
        {
            while(successRead==0)
            {
                successRead=getID();
                digitalWrite(BUZER,HIGH);
                successRead=getID();
                delay(100);
                digitalWrite(BUZER,LOW);
                delay(500);
                successRead=getID();
                digitalWrite(BUZER,LOW);
                i++;
                if(i==10)
                    break;
            }
        }
    }
}
```

```
    while(successRead==0)
    {
        digitalWrite(BUZER,HIGH);
        successRead=getID();
    }
}
```

```
digitalWrite(BUZER,LOW);
digitalWrite(sensorLed,LOW);
```

```
if(programMode)
{
    if(isMaster(readCard))    //Master Kart tekrar okutulursa programdan çıkar.
    {
        Serial.println(F("Master Kart Okundu.."));
    }
}
```

```

    Serial.println(F("Programdan cikis yapiliyor.."));
    Serial.println(F("*****"));
    programMode=false;
    return;
}
else
{
    if(findID(readCard))    //Okunan kart silinmek isteniyorsa
    {
        Serial.println(F("Okunan kart siliniyor.."));
        deleteID(readCard);
        Serial.println(F("*****"));
    }
    else                    //Okunan kart kaydedilmek isteniyorsa
    {
        Serial.println(F("Okunan kart hafizaya kaydediliyor.."));
        writeID(readCard);
        Serial.println(F("*****"));
    }
}
}
else
{
    if(isMaster(readCard))    //Master kart okunursa programa giriş yapılıyor.
    {
        programMode=true;
        Serial.println(F("Merhaba Sayin Kullanıcı, Programa giris yapiyorum."));
        int count=EEPROM.read(0);
        Serial.print(F("Sahip oldugum kullanici "));
        Serial.print(count);
        Serial.print(F("sayisi kadardir."));
        Serial.println("");
        Serial.println(F("Ekleme veya cikarmak istediginiz karti okutunuz."));
        Serial.println(F("*****"));
    }
    else
    {
        if(findID(readCard))
        {
            Serial.println(F("Hosgeldin, gecis izni verildi."));
            Accepted();
        }
        else
        {
            Serial.println(F("Gecis izni verilmedi."));

```

```

        denied();
    }
}
closeAlarm();
}
}

////////////////////ERİŞİM İZNİ////////////////////
void CloseAlarm()
{
    successRead=0;
    while(successRead == 0)
    {
        successRead=getID();
        sensorDeger=LOW;
    }
    Serial.println(F("Alarm Kuruluyor...."));
    delay(1000);
    Serial.println(F("5...."));
    delay(1000);
    Serial.println(F("4...."));
    delay(1000);
    Serial.println(F("3...."));
    delay(1000);
    Serial.println(F("2...."));
    delay(1000);
    Serial.println(F("1...."));
    delay(1000);
    Serial.println(F("Alarm Kuruldu...."));

}
void Accepted()
{
    sensorDeger=LOW;
    successRead=0;
    Serial.println(F("BASARILI"));
    digitalWrite(BUZER,HIGH);
    delay(100);
    digitalWrite(BUZER,LOW);
    delay(100);
    digitalWrite(BUZER,HIGH);
    delay(100);
    digitalWrite(BUZER,LOW);
    delay(100);
}

```

```
digitalWrite(BUZER,HIGH);
delay(100);
digitalWrite(BUZER,LOW);
}
```

```
void start()
{
```

```
// silme kodu butona basıldığında EEPROM içindeki bilgileri siler.
if(digitalRead(RESET)==LOW)
{
    Serial.println(F("Silme butonuna basıldı."));
    Serial.println(F("5 saniye icinde iptal edebilirsiniz.."));
    Serial.println(F("Tum kayitlar silinecek "));
    delay(5000); //kullanıcıya iptal işlemi için zaman
    if (digitalRead(RESET)==LOW)
    {
        Serial.println(F("EEPROM siliniyor."));

        for (int x=0; x<EEPROM.length();x=x+1) //EEPROM adresinin döngü sonu
        {
            if(EEPROM.read(x)!=0)
            {
                EEPROM.write(x,0);
            }
        }
        Serial.println(F("EEPROM Basariyla Silindi.."));
    }

    else
    {
        Serial.println(F("Silme islemi iptal edildi."));
    }
}
}
```

```
/*silinme işleminden sonra yeni bir master card tanımı istenicek.EEPROM kayıtlarının
haricinde 143 EEPROM adresi tutulabilir.
```

```
EEPROM adresi sihirli sayısı 143.
```

```
*/
```

```
if (EEPROM.read(1) != 143)
{
    Serial.println(F("Master Kart Secilmedi.));
    Serial.println(F("Master Karti secmek icin kartinizi okutunuz.."));
}
```



```

do
{
    successRead=getID();
}
while(!successRead);
for (int j=0; j<4; j++)          //4 kez döngü
{
    EEPROM.write(2+j,readCard[j]);    //UID EPPROM a yazıldı, 3. adres başla.
}
EEPROM.write(1,143);    //EEPROM a Master Kartı kaydettik.
Serial.println(F("Master Kart kaydedildi.."));
}

Serial.println("");
Serial.println(F("*****"));
Serial.println(F("Kapı açıldı"));
Serial.println(F("Kart okutulması için bekleniyor..."));
}

//////////ERİŞİM İZNİ VERİLMEDİ//////////
void denied()
{
    Serial.println(F("!!!!BASARISIZ!!!!"));
    digitalWrite(BUZER,HIGH);
    while((successRead=getID()) != 0)
    {
        successRead=getID();
        if(findID(readCard)==true)
        {
            digitalWrite(BUZER,LOW);
            break;
        }
    }
    successRead=0;
}

//////////KART OKUYUCU UID AYARLAMA//////////

int getID()
{
    //Kart okuyucuyu hazır ediyoruz
    if(!mfrc522.PICC_IsNewCardPresent())    //yeni bir kart okutun vve devam edin.
    {
        return 0;
    }
}

```

```

if(!mfrc522.PICC_ReadCardSerial())    //kartın serial numarasını alın ve devam edin.
{
    return 0;
}
//4 ve 7 byte UID'ler mevcut biz 4 byte olanı kullanacağız
Serial.println(F("Kartın UID'sini taratın..."));
for (int i=0; i<4; i++)
{
    readCard[i]=mfrc522.uid.uidByte[i];
    Serial.print(readCard[i],HEX);
}
Serial.println("");
mfrc522.PICC_HaltA(); //Okuma durduruluyor.

return 1;
}

////////////////////////////////EEPROM için ID Okuma////////////////////////////////

void readID( int number )
{
    int start=(number*4)+2;    //başlama pozisyonu
    for (int i=0; i<4; i++)    //4 byte alamabilmek için 4 kez döngü kurucuz
    {
        storedCard[i]=EEPROM.read(start+i); // EEPROM dan diziye okunabilen değerler atayın.
    }
}

////////////////////////////////EEPROM a ID Ekleme////////////////////////////////

void writeID(byte a[])
{
    if (!findID(a))    //biz eeprom a yazmadan önce önceden yazılıp yazılmadığını kontrol
    edin.
    {
        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]);
        }
        Serial.println(F("Basarili bir sekilde ID kaydi EEPROM'a eklendi.."));
    }
    else

```

```

    {
        Serial.println(F("Basarisiz! Yanlıs ID"));
    }
}

```

////////////////////////////////EEPROM'dan ID Silme////////////////////////////////

```

void deleteID(byte a[])
{
    if (!findID(a))          //Önceden EEPROM'dan silinen karta sahipmiyiz kontrol et.
    {
        Serial.println(F("Basarisiz! Yanlıs ID veya kotu EEPROM. :-("));
    }
    else
    {
        int num=EEPROM.read(0);
        int slot;
        int start;
        int looping;
        int j;
        int count=EEPROM.read(0); //Kart numarasını saklayan ilk EEPROM'un ilk byte'ını oku
        slot=findIDSLOT(a);
        start=(slot*4)+2;
        looping=((num-slot)*4);
        num--;                      //tek sayacı azaltma
        EEPROM.write(0,num);
        for(j=0;j<looping;j++)
        {
            EEPROM.write(start+j,EEPROM.read(start+4+j));
        }
        for(int k=0;k<4;k++)
        {
            EEPROM.write(start+j+k,0);
        }
        Serial.println(F("Silme basarili.));
        Serial.println(F("Basarili bir sekilde ID kaydi EEPROM'dan silinmistir..));
    }
}

```

////////////////////////////////Byte'ların Kontrolü////////////////////////////////

```

boolean checkTwo (byte a[],byte b[])
{
    if (a[0] != NULL)

```

```

match=true;

for (int k=0; k<4; k++)
{
    if(a[k] != b[k])
    {
        match=false;
    }
    if(match)
    {
        return true;
    }
    else
    {
        return false;
    }
}
}

////////////////////////////////Boşluk Bulma////////////////////////////////

int findIDSLOT (byte find[] )
{
    int count = EEPROM.read(0);    //EEPROM ile ilk byte ı okuyacağız.
    for(int i=1; i<=count; i++)    //Döngüdeki her EEPROM girişi için
    {
        readID(i);                //EEPROM daki ID yi okuyacak ve storedcard[4] de
saklayacağız.
        if (checkTwo(find, storedCard)) // Saklı kartlar da olup olmadığının kontrolü.
        { //aynı ID'e sahip kart bulursa geçişe izin vericek.
            return i;    //kartın slot numarası
            break;        //aramayı durdurucak.
        }
    }
}

////////////////////////////////EEPROM'da ID Bulma////////////////////////////////

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;
}
```

//////////////////Master Kartın Doğruluğunun Tespiti//////////////////

```
boolean isMaster (byte test[])
{
    if(checkTwo(test,masterCard))
        return true;
    else
        return false;
}
```