

Ultrasonik Sensor HCSR-04 ile mesafe ölçme HCSR-04-LED.ino

```
1. int buzzer = 8; // Buzzer pin 8'e bağlanacak.
2. int trig = 9; // HCSR-04'ün trig ucu pin 9'a bağlanacak.
3. int echo = 10; // HCSR-04'ün echo ucu pin 10'a bağlanacak.
4. int buton2 = 4; // 2 buton pin 4'e bağlanacak (öccü butonu)
5. void setup() {
6.   Serial.begin(9600); // seri portu 9600 baud hızında başlat.
7.   pinMode(buzzer, OUTPUT); // Buzzer çıkış seçildi.
8.   pinMode(trig, OUTPUT); // trig ucu çıkış seçildi.
9.   pinMode(echo, INPUT); // echo ucu giriş seçildi.
10.  for (int i=2; i<=7; i++) pinMode(i, OUTPUT);
11.  // 2'den 7'ye kadar led'ler çıkış yapılır.
12. }
13. void loop() {
14.   digitalWrite(trig, 0); // trig ucu 0 yapılır.
15.   delayMicroseconds(2); // 2 msn bekle.
16.   digitalWrite(trig, 1); // trig ucu +5V yapılır.
17.   delayMicroseconds(10); // 10 msn bekle.
18.   digitalWrite(trig, 0); // trig ucu tekrar 0 yapılır.
19.   long sure = pulseIn(echo, 1); // Dalganın geri dönüşünü hesapla.
20.   long mesafe = (sure/2)/28,97; // Sesin hızına göre cm'yi hesapla.
21.   potDeğeri = analogRead(potPin1); // A0 pinine analog oku.
22.   Serial.println(mesafe); // Değeri: 0, 1023, 0, 255);
23.   delay(1); // 1msn bekle.
24.   Serial.print("HIZ DEĞERİ ");
25.   if (mesafe < 8) { tone(buzzer, 440); delay(240); noTone(buzzer); }
26.   if (mesafe >= 10) digitalWrite(2, 1); else digitalWrite(2, 0);
27.   if (mesafe >= 13) digitalWrite(3, 1); else digitalWrite(3, 0);
28.   if (mesafe >= 16) digitalWrite(4, 1); else digitalWrite(4, 0);
29.   if (mesafe >= 20) digitalWrite(5, 1); else digitalWrite(5, 0);
30.   if (mesafe >= 24) digitalWrite(6, 1); else digitalWrite(6, 0);
31.   if (mesafe >= 30) digitalWrite(7, 1); else digitalWrite(7, 0);
32. }
33. digitalWrite(led_ileri, 1); // led ileri söndür.
34. }
```



```
1. #include <Stepper.h>
2. #define ADIM 32 // Adım sayısı 32 olarak belirlendi.
3. int buton1 = 3; // 1. buton pin 3'e bağlanacak (ileri butonu)
4. int buton2 = 4; // 2. buton pin 4'e bağlanacak. (geri butonu)
5. int led-ileri = 5; // ileri ledi pin 5'e bağlanacak.
6. int led-geri = 6; // geri ledi pin 6'ya bağlanacak.
7. int potPini = A0; // Motor hızı potPini ile belirlenecek.
8. int potDeğeri; // Motor geri döndürüyor. // motoru ger döndür sonra
9. int motorHizi = 250; // Başlangıçta motor dönüş hızı
10. delay(1000); // 1 sn bekle
11. Stepper myStepper (ADIM, 8, 10, 9, 11); // IN1-3 IN2-4
12.
13. void setup() {
14.   Serial.begin
15.   pinMode (buton1, INPUT_PULLUP); // pin 3 pullup direncini aktif yapar.
16.   pinMode (buton2, INPUT_PULLUP); // pin 4 pullup direncini aktif yapar.
17.   pinMode (led-ileri, OUTPUT); // led ileri pin 5 çıkış yapıldı.
18.   pinMode (led-geri, OUTPUT); // led geri pin 6 çıkış yapıldı.
19.   myStepper.setSpeed (motorHizi); // ilk olarak 250 hızında çalıştırır.
20. }
21.
22. void loop() {
23.   potDeğeri = analogRead (potPini); // A0 pinine analog oku.
24.   int motorHizi = map (potDeğeri, 0, 1023, 0, 255);
25.   myStepper.setSpeed (motorHizi); // pot Değerine göre hızı ayarlar.
26.   Serial.print ("HIZ DEĞERİ ");
27.   Serial.println (motorHizi);
28.
29.   if (digitalRead (buton1) == LOW); {
30.     digitalWrite (led-ileri, HIGH); // Eğer buton 1'e basılır ise
31.     digitalWrite (led-geri, LOW); // led ileri yi yak.
32.     myStepper.step (1024); // 1024 adım ayarlar.
33.     Serial.println ("Motor ileri dönüyor"); // Motoru döndür sonra
34.     digitalWrite (led-ileri, LOW); // led ileri yi söndür.
```



```

35. delay(1000); // 1sn. bekle
36. };
37. LiquidCrystal lcd(RS, EN, D4, D5, D6, D7); // LCD pin bağlantıları
38. if (digitalRead(buton2) == LOW) {
39.   digitalWrite(led-geri, HIGH); // Eğer buton2'ye basıldı ise
40.   digitalWrite(led-ileri, LOW); // Led geri yi yak
41.   myStepper.step(-1024); // Geriye 1024 adım ayarlar.
42.   Serial.println("Motor geri dönüyor."); // motoru ger döndür sonra
43.   digitalWrite(led-geri, LOW); // led geri yi söndür.
44.   delay(1000); // 1sn bekle
45. };
46. delay(100); // 100 msn bekle
47. delay(100); // 100 msn bekle
48. }

```

```

void loop() {

```

```

  digitalWrite(trigPin, LOW);
  delayMicroseconds(5); // 5 mikrosaniye
  digitalWrite(trigPin, HIGH); // trigPin'i yapılı
  delayMicroseconds(10); // 10 mikrosaniye
  digitalWrite(trigPin, LOW); // trigPin'i

```

```

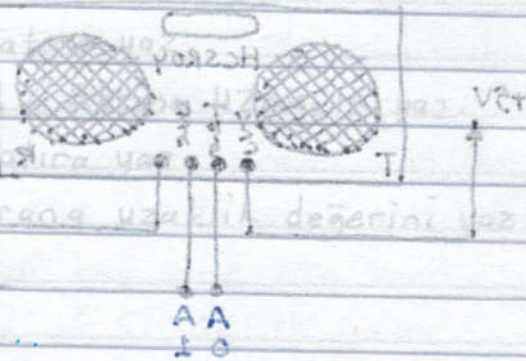
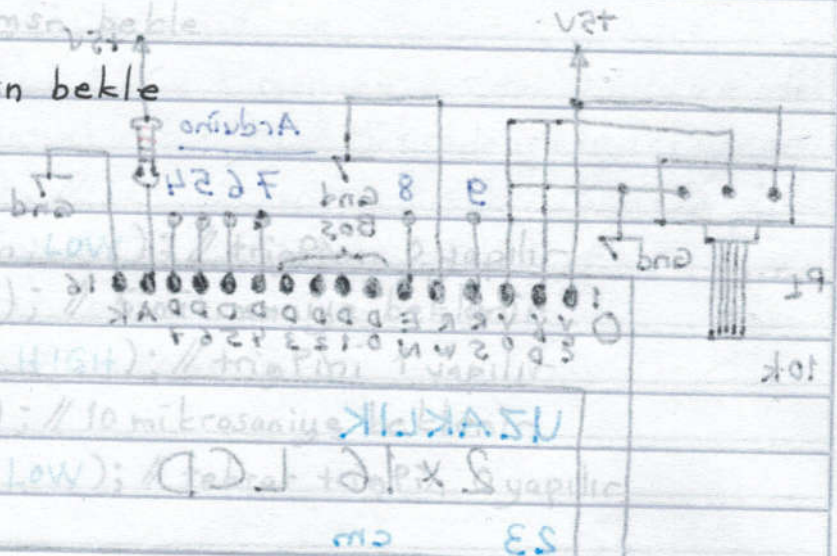
  int sure = pulseIn(echoPin, HIGH, 11600);
  int uzaklik = sure * 0.0345 / 2; // sesin havada aldığı yol
  delay(250); // 250 msn bekle

```

```

  lcd.clear(); // LCD temizlenir
  lcd.setCursor(0,0); // 0. satıra yazılır
  lcd.print("UZAKLIK: ");
  lcd.setCursor(0,1); // 1. satıra yazılır
  lcd.print(uzaklik); // Ekrana uzaklik değerini yaz
  lcd.setCursor(5,1);
  lcd.print("cm");
}

```



delay(1000); // 1 sn. bekle

};

}

if (digitalRead(button) == LOW) {

digitalWrite (led_geri, HIGH); // Eger butona basıldı ise

digitalWrite (led_ileri, LOW); // led geri yak

myStepper.stop(-1024); // Geriye 1024 adım gider

Serial.println ("Motor geri döndü."); // motor geri döndü - 2000

digitalWrite (led_geri, LOW); // led geri söndür

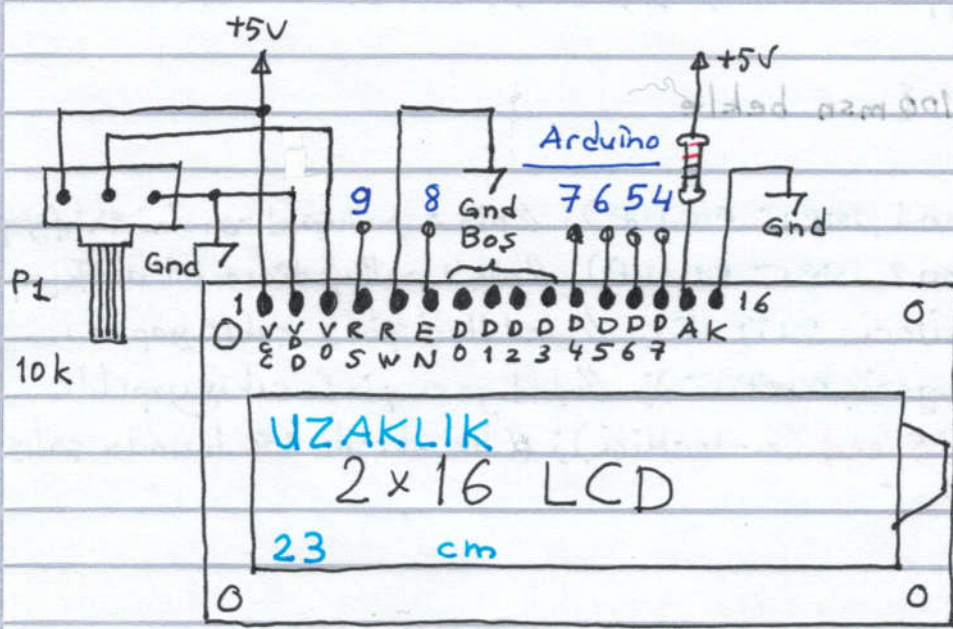
delay(1000); // 1 sn. bekle

};

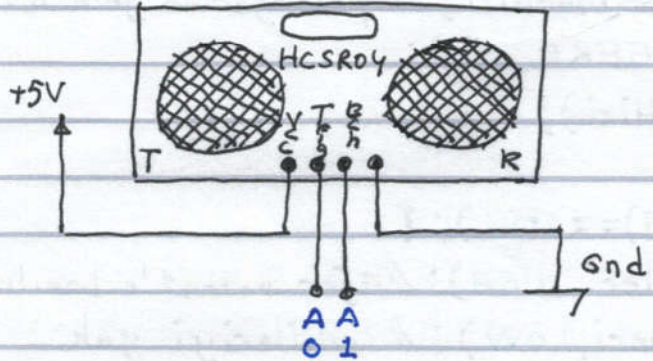
delay(100); // 100 ms bekle

}

Arduino LCD bağlantısı



Hc-SR04 bağlantısı



Arduino ile HC-SR04 ve LCD kullanımı HCSR-04-LCD.ino

```
1. #include <LiquidCrystal.h> // LCD kütüphanesi eklendi.
2. int RS=9, EN=8, D4=7, D5=6, D6=5, D7=4; // LCD kütüphanesi
3. LiquidCrystal lcd (RS,EN,D4,D5,D6,D7); // LCD pin bağlantıları.
4.
5. int trigPin = A0; // A0 pinini trig pinine bağlanacak.
6. int echoPin = A1; // A1 pinini echopinine bağlanacak.
7.
8. void setup() {
9.   pinMode(trigPin, OUTPUT); // trigPin'i çıkış seçildi
10.  pinMode(echoPin, INPUT); // echoPin'i giriş seçildi
11.  lcd.begin(16,2); // LCD 16x2 başlatılıyor.
12.  delay(100); // 100 msn bekle
13. }
14. lcd.setCursor(0,0); // LCD kursorunu 0.satır 0.satıra gönder
15. void loop() {
16.   digitalWrite(trigPin, LOW); // trigPin'i 0 yapılır.
17.   delayMicroseconds(5); // 5 mikrosaniye beklenir
18.   digitalWrite(trigPin, HIGH); // trigPin'i 1 yapılır.
19.   delayMicroseconds(10); // 10 mikrosaniye beklenir.
20.   digitalWrite(trigPin, LOW); // tekrar trigPin 0 yapılır.
21.
22.   int sure = pulseIn(echoPin, HIGH, 11600); // echoPin'den bir değeri alır
23.   int uzaklik = sure * 0.0345 / 2; // sesin havada aldığı yol.
24.   delay(250); // 250 msn bekle.
25.   lcd.clear(); // LCD temizlenir.
26.   lcd.setCursor(0,0); // 0.satıra yaz
27.   lcd.print("UZAKLIK: "); // Ekranı UZAKLIK yaz.
28.   lcd.setCursor(0,1); // 1.satıra yaz.
29.   lcd.print(uzaklik); // Ekranı uzaklik değerini yaz.
30.   lcd.setCursor(5,1); // Bir kaç satır ara verilir
31.   lcd.print("cm"); // SIZAKLIK yazdır
32. }
33. Serial.print(fahrenheit); // fahrenheit derece yazılır
    Serial.print("°F"); // Alt+248 ile ° isareti yazılabilir
    Serial.println(); // Bir alt satıra geçer
```


Arduino ile LM35 ve LCD-I2C kullanımı

LM35_LCD-I2C

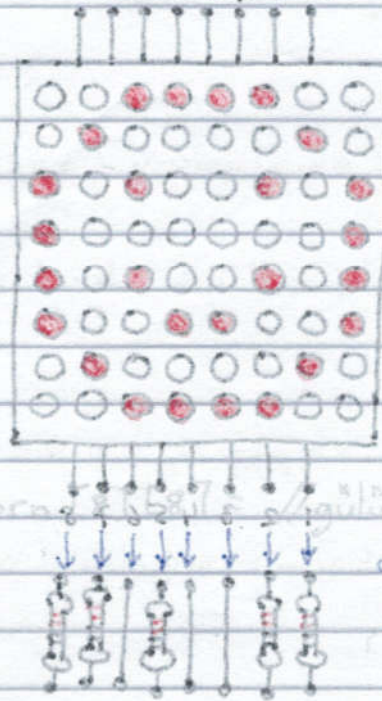
```
1. #include <Wire.h>
2. #include <LiquidCrystal_I2C.h> // LCD kütüphanesi
3. LiquidCrystal_I2C lcd (0x27, 16, 2); // LCD-I2C adresi
4. #define role 2 // role pin2'ye bağlanacak.
5. int deger; // deger değışten tanımlandı
6. int sıcaklikPin = A3; // LM35'in 2.pini A3'e bağlanacak
7. lcd.setCursor(15,0); // LCD kursorunu 15.satır, 0.sütuna gönder
8. void setup() { // Derece yazar
9.   Serial.begin(9600); // Seri portu 9600 baud hızında başlat.
10.  pinMode(role, OUTPUT); // SICAKLIK yazar.
11.  lcd.init(); // LCD'yi hazırla.
12.  lcd.backlight(); // LCD ışığını aç.
13.  lcd.setCursor(0,0); // LCD kursorunu 0.sütun 0.satıra gönder.
14.  lcd.print("Mikrodenetleyici"); // Mikrodenetleyici yazar.
15.  lcd.setCursor(0,1); // LCD kursorunu 0.sütun 1.satıra gönder
16.  lcd.print("11-T SINIFI"); // 11-T SINIFI yazar.
17.  delay(2000); // 2 sn bekle
18.  lcd.clear(); // LCD'yi temizle.
19. }
20.
21. void loop() {
22.   deger = analogRead(sıcaklikPin); // sıcaklikPin'den ham değeri oku
23.   float mVolt = (deger/1024.0) * 5000; // mVolt değerini hesapla
24.   float celcius = mVolt/10; // Santigrad dereceyi hesapla
25.   float fahrenheit = (celcius * 9)/5 + 32; // Fahrenheit'i hesapla
26.   // Aşağıdaki kodlar LCD olmadığı durumda kullanılacaktır.
27.   Serial.print("SICAKLIK="); // SICAKLIK yazar
28.   Serial.print(celcius); // santigrad derece yazılır.
29.   Serial.print("°C"); // Alt+248 ile ° işareti yazılabilir.
30.   Serial.print(" "); // Bir kaç satır ara verilir.
31.   Serial.print("SICAKLIK="); // SICAKLIK yazar
32.   Serial.print(fahrenheit); // fahrenheit derece yazılır.
33.   Serial.print("°F"); // Alt+248 ile ° işareti yazılabilir.
34.   Serial.println(); // Bir alt satıra geçer.
```



```

35. // Aşağıdaki kodlar LCD I2C olduğu durumlarda kullanılacak,
36. lcd.backlight(); // LCD'nin ışığını yak
37. // lcd.setCursor(0,0); // LCD kursorunu 0.satır, 0.sütuna gönder.
38. lcd.print("SICAKLIK"); // SICAKLIK yazar
39. lcd.setCursor(9,0); // LCD kursorunu 9.satır, 0.sütuna gönder
40. lcd.print(celcius); // santigrad dereceyi yazar.
41. lcd.setCursor(15,0); // LCD kursorunu 15.satır, 0.sütuna gönder.
42. lcd.print("C"); // Derece yazar.
43. lcd.setCursor(0,1); // LCD kursorunu 0.satır, 1.sütuna gönder.
44. lcd.print("SICAKLIK"); // SICAKLIK yazar.
45. lcd.setCursor(9,1); // LCD kursorunu 9.satır 1.sütuna gönder
46. lcd.print(fahrenheit); // fahrenheit dereceyi yazar.
47. lcd.setCursor(15,1); // LCD kursorunu 15.satır, 1.sütuna gönder.
48. lcd.print("F"); // Fahrenheit'i yazar
49.
50. if (celcius >= 32) {digitalWrite(role, HIGH); delay(50);}
51. else
52. {digitalWrite(role, LOW); delay(50);}
53. delay(100); // Buton sıçraması için 100 msn bekle
54. }

```



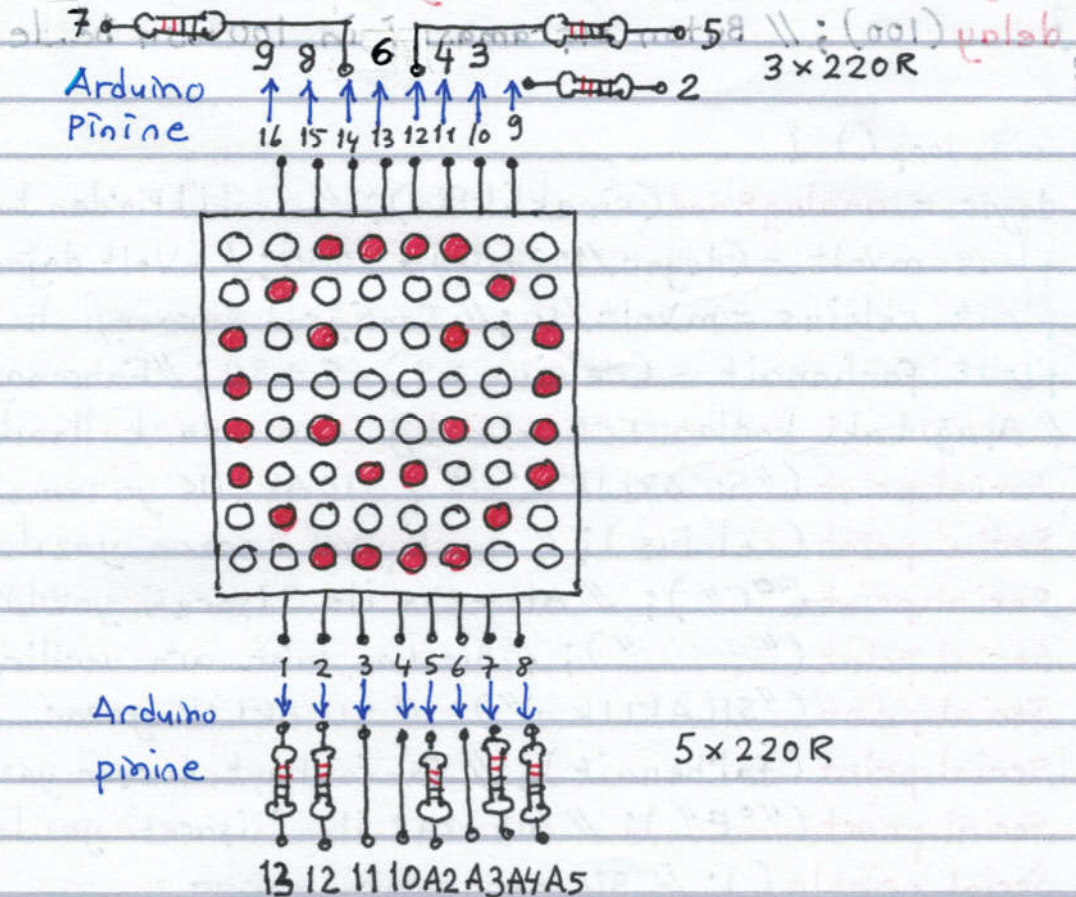
Arduino ile LM32 ve LCD-ISC kullanımı

```

35. // Aşağıdaki kodlar LCD ISC elemanı ile kullanılmak
36. lcd.backlight(); // LCD'nin ışığını yak
37. lcd.setCursor(0,0); // LCD kursorunu 0.satır 0.sütuna götüre
38. lcd.print("SICAKLIK"); // SICAKLIK yazdır
39. lcd.setCursor(9,0); // LCD kursorunu 9.satır 0.sütuna götüre
40. lcd.print("celcius"); // sıcaklığı dereceyi yazdır
41. lcd.setCursor(12,0); // LCD kursorunu 12.satır 0.sütuna götüre
42. lcd.print("C"); // Derece yazdır
43. lcd.setCursor(0,1); // LCD kursorunu 0.satır 1.sütuna götüre
44. lcd.print("SICAKLIK"); // SICAKLIK yazdır
45. lcd.setCursor(9,1); // LCD kursorunu 9.satır 1.sütuna götüre
46. lcd.print("fahrenheit"); // Fahrenheit dereceyi yazdır
47. lcd.setCursor(12,1); // LCD kursorunu 12.satır 1.sütuna götüre
48. lcd.print("F"); // Fahrenheit yazdır
49.
50. if (celcius >= 32) { digitalWrite(7,HIGH); delay(50); }
51. else
52. { digitalWrite(7,LOW); delay(50); }

```

Dot Matrix: (8x8) 88 AS kullanımı



Arduino ile Dot Matris 8x8 (1088AS) kullanımı 8x8_dotmatris.ino

```
1. // Satır numaraları sırasıyla
2. int R[] = {2,7,A5,5,13,A4,12,A2} // 8'li dizi tanımlandı
3. // Sütun numaraları sırasıyla
4. int C[] = {6,11,10,3,A3,4,8,9} // 8'li dizi tanımlandı.
5. 1,1,0,0,0,0,1,1,
6. unsigned char smile[8][8] = //gülümse
7. {
8. 1,1,0,0,0,0,1,1,
9. 1,0,1,1,1,1,0,1,
10. 0,1,0,1,1,0,1,0, <8;i++)
11. 0,1,1,1,1,1,1,0, //i sırasıyla aktif yap.
12. 0,1,0,1,1,0,1,0, OUTPUT); // Sütunlar çıkış seçildi
13. 0,1,1,0,0,1,1,0, OUTPUT); // Satırlar çıkış seçildi
14. 1,0,1,1,1,1,0,1,
15. 1,1,0,0,0,0,1,1,
16. };
17. void loop()
18. { unsigned char sad[8][8] = //gülümse 2
19. { for(int i=0; i<200; i++) // Displayde 200 defa tekrarla
20. 1,1,0,0,0,0,1,1,
21. 1,0,1,1,1,1,0,1, ); // Displayde smile gösterir
22. 0,1,0,1,1,0,1,0,
23. 0,1,1,1,1,1,1,0,
24. 0,1,1,1,1,1,1,0, i<150; i++) // Displayde 150 defa tekrarla
25. 0,1,1,1,1,1,1,0,
26. 1,0,1,1,1,1,0,1, // Displayde smile 2 gösterir.
27. 1,1,0,0,0,0,1,1,
28. };
29. for(int i=0; i<100; i++) // Displayde 100 defa tekrarla
30. unsigned char stubborn[8][8] = //gülümse 3
31. { Display(stubborn); // Displayde smile 3 gösterir.
32. 1,1,0,0,0,0,1,1,
33. 1,0,1,1,1,1,0,1,
34. 0,1,0,1,1,0,1,0,
```



```

35.   0,1,1,1,1,1,1,0, signed char dot[8][8])
36.   { 0,1,0,0,0,0,1,0,
37.     0,1,1,1,1,1,1,0, <8; r++)
38.     1,0,1,1,1,1,0,1,
39.     1,1,0,0,0,0,1,1, [c], HIGH); // Sütunları kullan
40. }
41.   for (int r=0; r<8; r++)
42.   void setup()
43.   { digitalWrite(R[c], dot[r][c]);
44.     for (int i=0; i<8; i++)
45.     { // çıkış pinlerini sırayla aktif yap.
46.       pinMode(R[i], OUTPUT); // Sütunlar çıkış seçildi.
47.       pinMode(C[i], OUTPUT); // Satırlar çıkış seçildi.
48.     }
49.   } void clear()
50.   {
51.   void loop() {
52.     for (int i=0; i<200; i++) // Displayde 200 defa tekrarla
53.     { digitalWrite(C[0], LOW); // Sütunlar 0 yapıldı
54.       Display(smile); // Displayde smile gösterir.
55.     }
56.     for (int i=0; i<150; i++) // Displayde 150 defa tekrarla
57.     {
58.       Display(sad); // Displayde smile 2 gösterir.
59.     }
60.     for (int i=0; i<100; i++) // Displayde 100 defa tekrarla
61.     {
62.       Display(stubborn); // Displayde smile 3 gösterir.
63.     }
64.     for (int i=0; i<50; i++) // Displayde 50 defa tekrarla
65.     {
66.       Display(sad); // Displayde smile 2 gösterir.
67.     }
68.   }

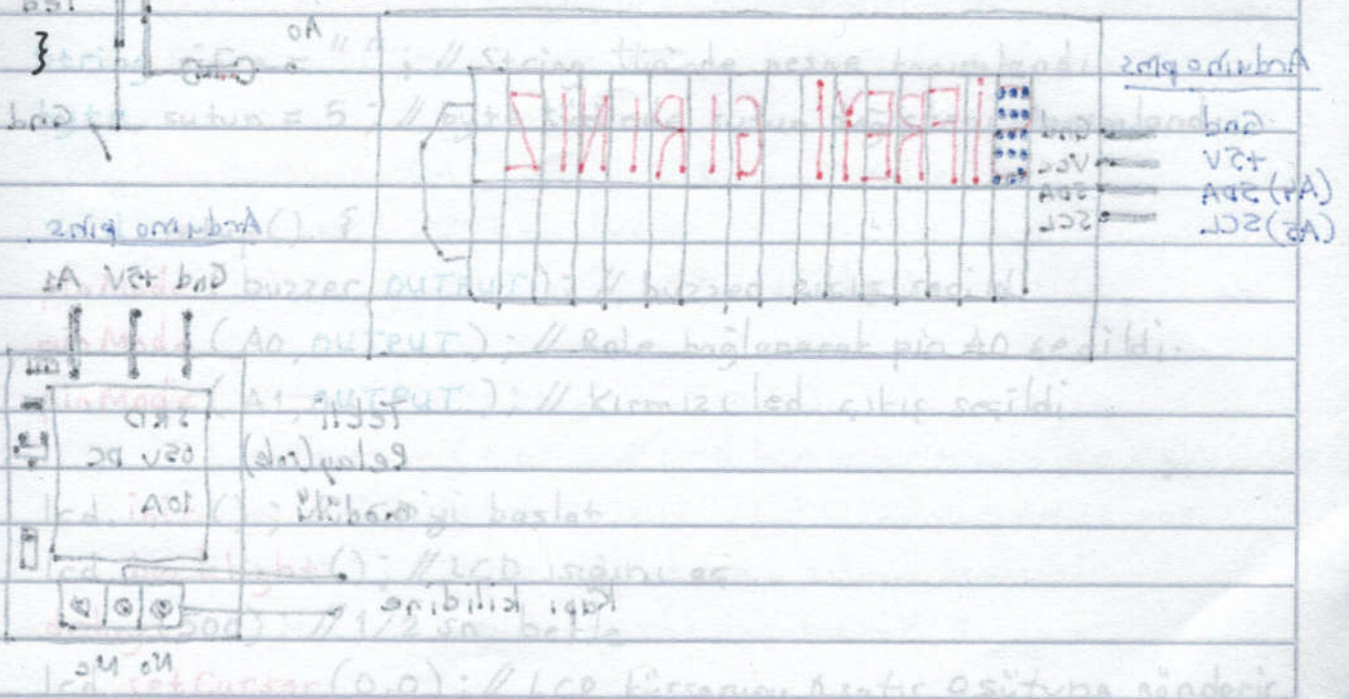
```



```

69. void Display(unsigned char dat[8][8])
70. {
71.   for(int c=0; c<8; c++)
72.   {
73.     digitalWrite(C[c], HIGH); // Sütunları kullan
74.     // döngü
75.     for(int r=0; r<8; r++)
76.     {
77.       digitalWrite(R[r], dat[r][c]);
78.     }
79.     delay(1); // 1msn bekle
80.     Clear(); // Displayi sil.
81.   }
82. }
83. void Clear()
84. {
85.   for(int i=0; i<8; i++)
86.   {
87.     digitalWrite(R[i], HIGH); // Satırlar 1 yapıldı
88.     digitalWrite(C[i], LOW); // Sütunlar 0 yapıldı
89.   }
90. }
91. }

```

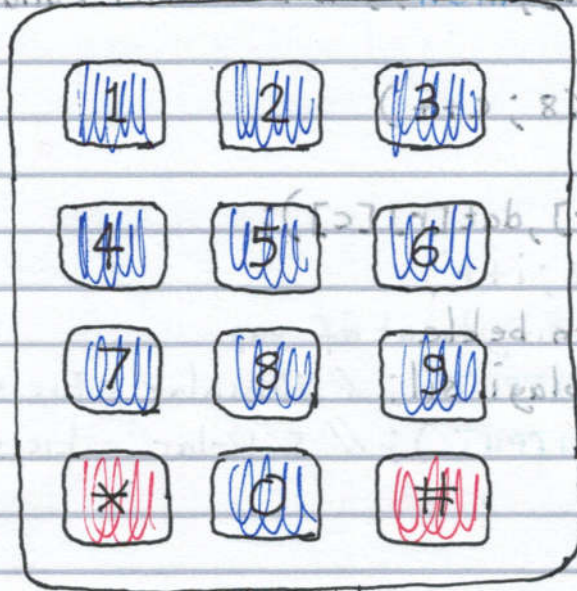


Arduino ile Dot Matrix 8x8 (10882) Fullscreen

void Display (unsigned char dot[8][8])

for (int c=0; c<8; c++)

digitalWrite (C[c], HIGH);

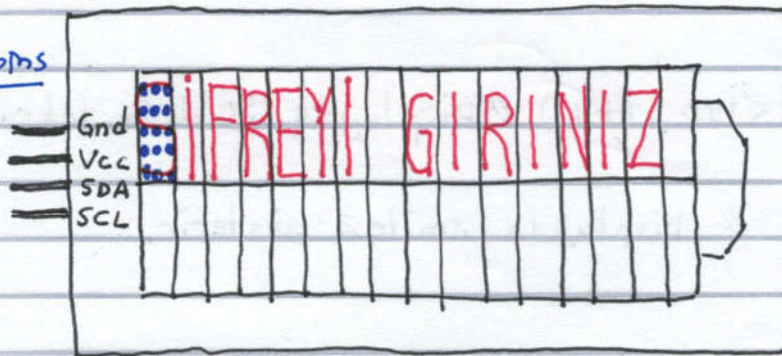


Arduino Pins 12 11 10 9 8 7 6

Arduino ile LCD bağlantısı

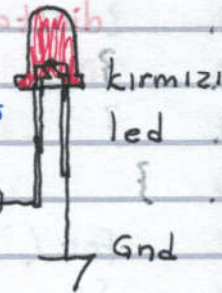
Arduinopins

Gnd
+5V
(A4) SDA
(A5) SCL



Arduinopins

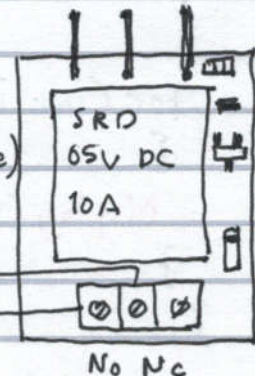
A0



Arduinopins

Gnd +5V A1

Tekli
Relay (role)
modülü



Kapı kilidine

Proje 1 - Arduino ile LCD'li şifreli kilit

şifreliKilit.ino

Alt+35

```

1. #include <Wire.h> // LCD'ye SIFREYİ GİRİNİZ yaz
2. #include <LiquidCrystal_I2C.h>
3. #include <Keypad.h>
4. #define buzzer 4 // Pasif buzzer pin 4'e bağlı.
5. // Likit kristal I2C hattına SDA ve SCL pinlerine bağlı.
6. LiquidCrystal_I2C lcd(0x27, 16, 2);
7. if (key) { // key değişkeninde bir veri varsa;
8.   const byte ROWS = 4; // Satır sayısı tanımlandı. delay(50);
9.   const byte COLS = 3; // Sütun sayısı tanımlandı.
10.   şifre += key; // Girilen karakterleri yan yana ekle.
11.   char keys[ROWS][COLS] = {
12.     {'1', '2', '3'},
13.     {'4', '5', '6'},
14.     {'7', '8', '9'},
15.     {'*', '0', '#'}
16.   };
17.   if (key == '#') { // Eğer # karakterine basıldıysa;
18.     byte rowPins[ROWS] = {12, 11, 10, 9}; // Satır pinleri 4 tane
19.     byte colPins[COLS] = {8, 7, 6}; // Sütun pinleri 3 tane
20.     // KEYPAD HARİTASINI OLUŞTURALIM
21.     Keypad keypad = Keypad(makeKeymap(keys), rowPins, colPins, ROWS, COLS);
22.     lcd.setCursor(2, 1); // Kursoru 2. sütun ve 1. satıra gönder
23.     String şifre = ""; // String türünde nesne tanımlandı.
24.     byte sutun = 5; // byte türünde sutun değişkeni tanımlandı.
25.     digitalWrite(A1, HIGH); // Role'yi salıstır
26.     void setup() { // 3 sn. bekle
27.       pinMode(buzzer, OUTPUT); // buzzer çıkış seçildi.
28.       pinMode(A0, OUTPUT); // Role bağlanacak pin A0 seçildi.
29.       pinMode(A1, OUTPUT); // kırmızı led çıkış seçildi
30.       lcd.setCursor(0, 0); // LCD kursorunu 0. satır ve 0. sütuna gönder
31.       lcd.init(); // LCD'yi başlat.
32.       lcd.backlight(); // LCD ışığını aç.
33.       delay(500); // 1/2 sn. bekle
34.       lcd.setCursor(0, 0); // LCD kursorunu 0. satır 0. sütuna gönderir.

```




```
35. lcd.print("SIFREYİ GIRINIZ"); // LCD'ye SIFREYİ GIRINIZ yaz.
36. }
37. tone(buzzer, 1000, 500); delay(200); noTone(buzzer); delay(50);
38. void loop() { // LCD'yi temizle
39.   char key = keypad.getKey(); // Keypad'den veri gelince key'e kaydet.
40.   lcd.print("SIFRE YANLIŞ."); // SIFRE YANLIŞ yaz.
41.   if (key) { // key değişkeninde bir veri varsa;
42.     tone(buzzer, 1000, 500); delay(250); noTone(buzzer); delay(50);
43.     if (key != '#') { Eğer #'e basılmadıysa;
44.       sifre += key; // Girilen karakterleri yan yana ekle.
45.       lcd.setCursor(0, 0); // LCD kursorunu 0. satır ve 0. sütuna gönder.
46.       lcd.print("# TUSUNA BASINIZ"); // # TUSUNA BASINIZ yaz.
47.       lcd.setCursor(sutun, 1); // İlgili sütun ve 1. satıra git. (sutun=5)
48.       lcd.print("*"); // LCD'ye * karakteri yazar.
49.       sutun++; // Sütun değerini arttır.
50.     }
51.     if (key == '#') { // Eğer # karakterine basıldıysa;
52.       if (sifre == "1923") { // Doğru sifre girilirse;
53.         lcd.clear(); // LCD'yi temizle.
54.         lcd.setCursor(2, 0); // Kursoru 2. sütun ve 0. satıra gönder
55.         lcd.print("SIFRE DOĞRU"); // SIFRE DOĞRU yazar
56.         lcd.setCursor(2, 1); // Kursoru 2. sütun ve 1. satıra gönder
57.         lcd.print("KAPI ACILDI"); // KAPI ACILDI yazar
58.         digitalWrite(A1, HIGH); // Role'i çalıştır.
59.         delay(3000); // 3 sn bekle
60.         digitalWrite(A1, LOW); // Role'i durdur.
61.         lcd.clear(); // LCD'yi temizle.
62.         lcd.setCursor(0, 0); // LCD kursorunu 0. satır ve 0. sütuna gönder
63.         lcd.print("SIFREYİ GIRINIZ"); // SIFREYİ GIRINIZ yaz.
64.         sifre = ""; // sifre değişkenini tekrar temizle
65.         sutun = 5; // sutun değişkenini tekrar 5 yap.
66.       }
67.     }
68.   }
```

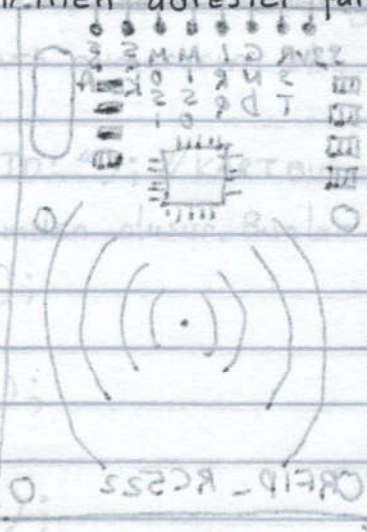
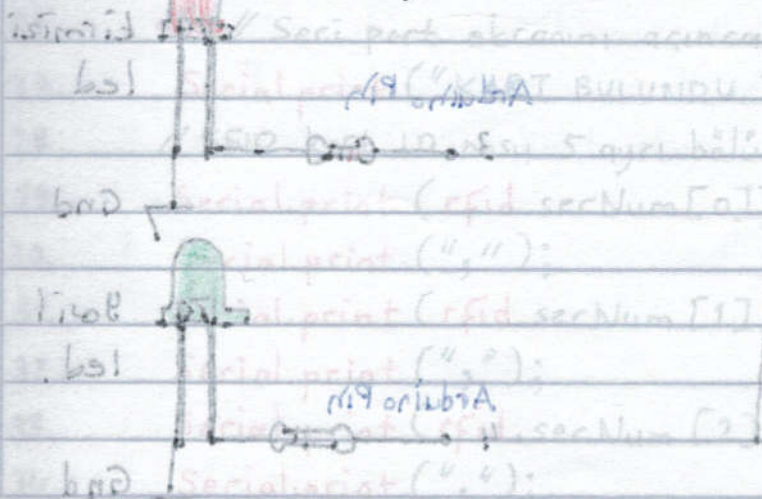


```

69. else // Bu şartların dışındaki durumlarda
70. { // Bıp sesini daha uzun çal
71. tone(buzzer, 1000, 500); delay(700); noTone(buzzer); delay(50);
72. lcd.clear(); // LCD'yi temizle
73. lcd.setCursor(2, 0); // LCD kursorunu 2. sütun ve 0. satıra gönder
74. lcd.print("SIFRE YANLIŞ..!"); // SIFRE YANLIŞ..! yaz
75. lcd.setCursor(0, 1); // kursoru 0. sütun ve 1. satıra gönder
76. lcd.print("TEKRAR DENEYİN"); // TEKRAR DENEYİN yaz
77. digitalWrite(A0, HIGH); // Kırmızı led'i yak.
78. delay(2000); // 2 sn bekle
79. digitalWrite(A0, LOW); // Kırmızı led'i söndür.
80. lcd.clear(); // LCD'yi temizle.
81. lcd.setCursor(0, 0); // LCD kursorunu 0. sütun ve 0. satıra gönder.
82. lcd.print("SIFREYİ GİRİNİZ"); // SIFREYİ GİRİNİZ yaz
83. sifre = ""; // sifre değişkenini temizle
84. sutun = 5; // sutun değişkenini 5 yap.
85. }
86. }
87. }
88. }

```

Not: 6. satırda **LiquidCrystal_I2C**.lcd(0x27, 16, 2) i yazılarak 8 LCD ekranda görüntü oluşmuyorsa 0x27 yerine 0x3F yazılacaktır. Bazı LCD ekranlarda I2C yolunda belirlenen adresler farklı olabildiği için haberleşme sağlanamamaktadır.



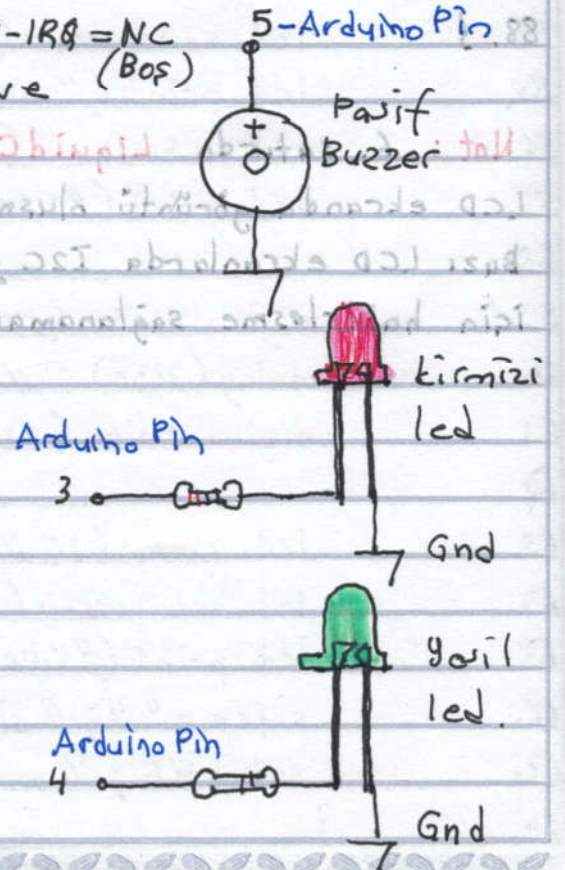
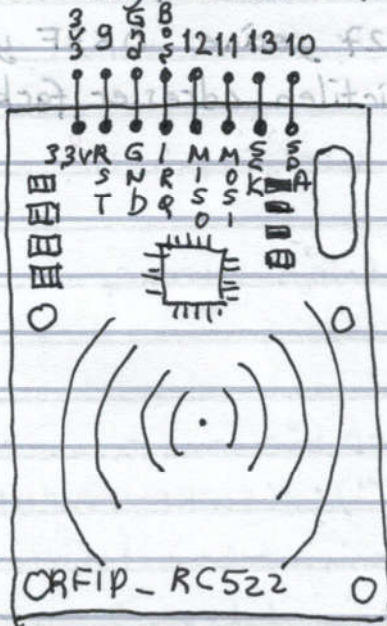

```

50. else // Bu şartların dışındaki durumlarda
51. { // Bıp sesini daha uzun süre
52. tone(buzzer, 1000, 200); // delay(500); // buzzer (buzzer); // delay(20);
53. lcd.setCursor(0, 0); // LCD kursorunu 0. satıra ve 0. sütuna döndür
54. lcd.print("SİFRE YANLIŞ!"); // SİFRE YANLIŞ! yaz
55. lcd.setCursor(0, 1); // kursoru 0. satıra ve 1. satıra döndür
56. lcd.print("TEKRAR DENEYİN"); // TEKRAR DENEYİN yaz
57. digitalWrite(A0, HIGH); // Kilitin açılması
58. delay(5000); // 5 sn bekle
59. digitalWrite(A0, LOW); // Kilitin kapanması
60. lcd.setCursor(0, 0); // LCD kursorunu 0. satıra ve 0. sütuna döndür
61. lcd.print("SİFREYİ GİRİNİZ"); // SİFREYİ GİRİNİZ yaz
62. // Sifre değeri için
63. // Sifre değeri için
64. // Sifre değeri için
65. // Sifre değeri için
66. // Sifre değeri için
67. // Sifre değeri için
68. // Sifre değeri için
69. // Sifre değeri için
70. // Sifre değeri için
71. // Sifre değeri için
72. // Sifre değeri için
73. // Sifre değeri için
74. // Sifre değeri için
75. // Sifre değeri için
76. // Sifre değeri için
77. // Sifre değeri için
78. // Sifre değeri için
79. // Sifre değeri için
80. // Sifre değeri için
81. // Sifre değeri için
82. // Sifre değeri için
83. // Sifre değeri için
84. // Sifre değeri için
85. // Sifre değeri için
86. // Sifre değeri için
87. // Sifre değeri için
88. // Sifre değeri için
89. // Sifre değeri için
90. // Sifre değeri için
91. // Sifre değeri için
92. // Sifre değeri için
93. // Sifre değeri için
94. // Sifre değeri için
95. // Sifre değeri için
96. // Sifre değeri için
97. // Sifre değeri için
98. // Sifre değeri için
99. // Sifre değeri için
100. // Sifre değeri için

```

Arduino Pin Bağlantıları

1-3,3V=3,3V'a 2-RST=9'a 3-GND=GND'ye 4-IRQ=NC (Boş) 5-Arduino Pin
 5-MISO=12'ye 6-MOSI=11'e 7-SCK=13'e ve 8-SDA=10'a bağlanacaktır.




```
1. #include <SPI.h>
2. #include <RFID.h>;
3. const byte kirmizi_led = 3;
4. const byte yesil_led = 4;
5. #define hoparlör 5
6. for (int i=0; i<5; i++) {
7. RFID rfid(10,9); // SDA 10.pine ve RST 9.pine bağlanacak.
8. // Kayıtlı olan kartın ID nosundan farklıysa;
9. byte kart[5] = {39,0,200,115,156}; // Buraya kendi RFID kartının
10. // ID numarası yazılır. Seri port ekranından okunan numaralardır.
11. boolean izin = true; // ilk anda izin bilgisi true yani '1' olur.
12. if (izin == true) {
13. void setup() {
14. Serial.begin(9600); // Seri iletişimi 9600 Baud hızında başlat.
15. SPI.begin(); // SPI haberleşme protokolunu başlat.
16. rfid.init(); // RFID kart okumasını başlat.
17. pinMode(kirmizi_led, OUTPUT);
18. pinMode(yesil_led, OUTPUT);
19. pinMode(hoparlör, OUTPUT);
20. }
21. Serial.println("İZİNSİZ GİRİŞ"); // İZİNSİZ GİRİŞ yaz
22. void loop() {
23. izin = true; // izin değeri '1' olur.
24. if (rfid.isCard()) { // Eğer RFID kartı okunduysa;
25. if (rfid.readCardSerial()) // Kart numarasını oku
26. { // Seri port ekranını açınca
27. Serial.print("KART BULUNDU ID:"); // KART BULUNDU ID: yaz
28. // RFID kart ID nosu 5 ayrı bölümden oluşur. Bunlar ayrı ayrı yazılır.
29. Serial.print(rfid.serNum[0]);
30. Serial.print(",");
31. Serial.print(rfid.serNum[1]);
32. Serial.print(",");
33. Serial.print(rfid.serNum[2]);
34. Serial.print(",");
```



```
35. Serial.print(rfid.serNum[3]);
36. Serial.print(",");
37. Serial.print(rfid.serNum[4]);
38. }
39.
40. for (int i=0; i<5; i++) {
41.   if (rfid.serNum[i] != kart[i]) { // Eğer okunan kartın ID
42.     // Kayıtlı olan kartın ID'nden farklıysa;
43.     izin = false; // izin değerini "0" yap.
44.   }
45. }
46. if (izin == true) {
47.   Serial.println(" İZİN VERİLDİ"); // Ekranı İZİN VERİLDİ yaz
48.   Serial.println("-----");
49.   digitalWrite(yesil-led, HIGH); // yeşil-led'i yak.
50.   tone(hoparlör, 1000, 100); // Bırp sesi çalar.
51.   delay(2000); // 2 sn bekle
52.   digitalWrite(yesil-led, LOW); // yeşil-led'i söndür.
53. }
54. else {
55.   Serial.println(" İZİNSİZ GİRİŞ"); // İZİNSİZ GİRİŞ yaz.
56.   Serial.println("-----");
57.   digitalWrite(kırmızı-led, HIGH); // kırmızı-led'i yak.
58.   tone(hoparlör, 1000, 750); // Bırp sesini uzun çal.
59.   delay(2000); // 2 sn bekle
60.   digitalWrite(kırmızı-led, LOW); // kırmızı-led'i söndür.
61. }
62. rfid.halt(); // Kart okuma işlemini bitir.
63. }
64. }
65.
```




Proje - 3 Deprem algılama sensörü

MPU6050.ino

```
1. #include <MPU6050.h>
2. #include <Wire.h>
3. MPU6050 MPU = MPU6050();
4. int GyroX, GyroY, GyroZ;
5. int buzzer = 10; // Buzzer Digital 10'a bağlanacak
6. // Pasif buzzer bağlanacaktır. Devreli buzzer.
7. void setup() {
8.   pinMode(10, OUTPUT); // Buzzer çıkışı yapıldı.
9.   pinMode(11, OUTPUT); // Led çıkışı (kırmızı)
10.  Serial.begin(9600); // Seri iletişimi 9600 baud başlat
11.  Wire.begin(); // Kablo iletişimini başlat
12.  MPU.initialize(); // MPU sensörünü başlat
13. }
14. float frequencySetting = 97.5; // FREKANS FM
15. void loop() {
16.   MPU.getRotation(&GyroX, &GyroY, &GyroZ);
17.   if (GyroX < -1000 || GyroX > 1000 || GyroY > 1000 ||
18.   GyroY < -1000 || GyroZ > 1000 || GyroZ < -1000) {
19.     // Eğer GyroX, GyroY veya GyroZ' bir hareket algılsa
20.     tone(buzzer, 1000); // Buzzer'i öttür.
21.     digitalWrite(11, HIGH); // Kırmızı ledi yak
22.     delay(1000); // 1sn bekle
23.   } else { // Değilse
24.     noTone(buzzer); // Buzzeri sustur.
25.     digitalWrite(11, LOW); // Kırmızı ledi söndür.
26.   }
27. }
28. float comingByte;
29. float frequencySetting;
30.
31. if (Serial.available() > 0) {
32.
33.   comingByte = Serial.parseFloat();
34.   frequencySetting = comingByte;
```


Proje 4 Arduino ve TEA5767 LCD Radyo

```
1. #include <Wire.h>
2. #include <TEA5767Radio.h>
3. TEA5767Radio radio = TEA5767Radio();
4. #include <LiquidCrystal.h>
5. const int rs = 8, en = 9, d4 = 4, d5 = 5, d6 = 6, d7 = 7;
6. LiquidCrystal lcd(rs, en, d4, d5, d6, d7);
7.
8. void setup()
9. {
10.   Serial.begin(9600);
11.   Wire.begin();
12.   // Frekans bilgisi seri porttan 106.0 girilirse VAV FM çıkar.
13.   float frequencySetting;
14.   frequencySetting = 97.5; // EREĞLİ FM FREKANSI
15.   radio.setFrequency(frequencySetting);
16.   // LCD'ye mesaj yazılıyor.
17.   lcd.begin(16, 2);
18.   lcd.print("FM RADYO V.2");
19.   lcd.setCursor(0, 1);
20.   lcd.print("FREKANS=");
21.   lcd.setCursor(8, 1);
22.   lcd.print(frequencySetting); // Ayarlanan frekans yaz
23.   delay(1000); // 1 sn bekle
24. }
25.
26. void loop()
27. {
28.   float comingByte;
29.   float frequencySetting;
30.   delay(100); // 100 ms bekle
31.   if (Serial.available() > 0) {
32.
33.     comingByte = Serial.parseFloat();
34.     frequencySetting = comingByte;
```

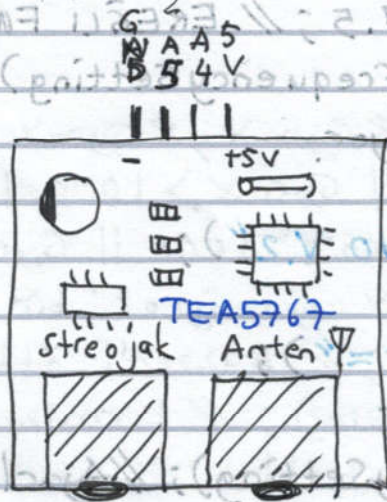


```

1 #include <Wire.h>
2 #include <TEA5767Radio.h>
3 TEA5767Radio radio = TEA5767Radio();
4 #include <LiquidCrystal.h>
5 const int rs = 8, rw = 9, rd = 10, qc = 4, qd = 5, qf = 6;
6 LiquidCrystal lcd(rs, rw, rd, qc, qd, qf);
7
8 void setup()
9 {
10     Serial.begin(9600);
11     Wire.begin();
12     // Frekans dijital seri porttan 106.0 dijital VAV FM çıkış
13     float frequencySetting;
14     frequencySetting = 97.5; // Frekans FM FREKANZI
15     radio.setFrequency(frequencySetting);
16     // LCD'ye mesaj yazdırma
17     lcd.begin(16, 2);
18     lcd.print("FM RAYON");
19     lcd.setCursor(0, 1);
20     lcd.print("FREKANZI");
21     lcd.setCursor(8, 1);
22     lcd.print(frequencySetting);
23     delay(1000); // 1 sn bekle
24 }
25
26 void loop()
27 {
28     float comingByte;
29     float frequencySetting;
30
31     if (Serial.available() > 0) {
32
33         comingByte = Serial.parseFloat();
34         frequencySetting = comingByte;

```

Arduino



Proje 4 Arduino ve TEA5767 LCD Radyo

```
35. // while (Serial.available() == 0) {
36.     // Bu satırlar istenirse konabilir. Frekans 0.0 oluncaya
37. // }
38.
39.     radio.setFrequency(frequencySetting);
40.
41.     lcd.setCursor(0,0);
42.     lcd.print("FM FR. Giriniz..");
43.     lcd.setCursor(0,1);
44.     lcd.print("FREKANS=");
45.     lcd.setCursor(8,1);
46.     lcd.print(frequencySetting);
47.     lcd.setCursor(10,1);
48.     lcd.print(".");
49.     lcd.setCursor(13,1);
50.     lcd.print("");
51.
52.     if (frequencySetting > 99.9) {
53.         lcd.setCursor(0,1);
54.         lcd.print("FREKANS=");
55.         lcd.setCursor(8,1);
56.         lcd.print(frequencySetting);
57.         lcd.setCursor(11,1);
58.         lcd.print(".");
59.     } // if digitalWrite(LED, HIGH); // ONBOARD led'i yak
60.     // mic değeri yani A0 değeri 50'den büyükse
61. } digitalWrite(LED, LOW); // Led'i söndür
62. Serial.println("Lamba söndü");
63. Serial.println(frequencySetting); // Frekansı seri porta yaz.
64. delay(100); // 100 msn bekle
65.
66. }
67.
```


Alkışla Çalışan Onboard Led uygulaması.

Analog çıkıslı mikrofon modülünden gelen belirli bir sinyal ile çıkışta ONBOARD (Arduino üzerindeki) LED'i yakar. Örneğin alkış yapınca seviye geçilir. Sonra belli bir zaman geçince tekrar alkış yapınca seviye yine geçilir. Yanmış olan ONBOARD LED söner. Alkış yaptıkça tekrar yanıp sönmeye devam eder.

```
1. const byte led = 13; // ONBOARD led tanımlandı.
2. int mic; // mic adlı değişken tanımlandı.
3. bool degistir = 1; // degistir adlı değişken 1 yapıldı.
4. void setup() {
5.     pinMode (led, OUTPUT);
6.     Serial.begin (9600); // Seri portu 9600 baud hızla başlat.
7. }
8.
9. void loop() {
10.    mic = analogRead (A0);
11.    if (mic > 59) { // mic A0 çıkış değeri 59'u geçerse
12.        digitalWrite (led, degistir); // degistir'i tersle.
13.        degistir = !degistir; // Önce led'i yak.
14.        delay (100);
15.    }
16.    Serial.println (mic); // mic değerini seri portta yaz.
17. }
18.    digitalWrite (led, HIGH); // ONBOARD led'i yak
19.    else // mic değeri yani A0 değeri 59'dan küçükse
20.        digitalWrite (led, LOW); // Led'i söndür
21.    Serial.println ("lambda söndü");
22. }
```