

# PROLAB2/PROJE-2

## Arduino ile Akıllı Ev Simülasyonu

1<sup>st</sup> Melih Selami ÜRKMEZ  
Kocaeli Üniversitesi  
Bilgisayar Mühendisliği  
Kocaeli,Türkiye  
200202010@kocaeli.edu.tr

**Abstract**—Proje konusu özetlenecek olursa Arduino kartına yangın sensörü, hareket sensörü, keypad ve sıcaklık ölçer entegre ederek bir akıllı ev tasarımı yapmamız istenmiştir. Her sensörün kendine göre koşulları vardır. Örnek verecek olursak sıcaklık ölçer sensörü 20 derece altına düştüğünde lcd'ye sıcaklık düştü yazar. Eğer 30 derece üstüne çıkarsa sıcaklık yükseldi yazar. Örnek koşullardan biridir.



### I. GİRİŞ

**Akıllı ev sistemleri** aydınlatma, ısıtma, güvenlik ve ev aletlerinin yönetilmesini sağlayan otomasyon sistemlerdir. Akıllı ev teknolojisinde kurulan sistem akıllı cep telefonu ya da tablet gibi cihazlarınıza bağlanarak evde veya evden çok uzakta iken sistemi yönetebilmenizi sağlar. Akıllı ev özellikleri gelişen teknoloji ile her geçen gün yeni imkânlar ile gelişmektedir.

**Akıllı ev sistemleri** zamanlayıcı sistemler ve tetikleyici sistemler olarak kurulur. Zamanlayıcı sistemler evinizde yapılmasını istediğiniz olayları otomatik olarak gerçekleştirir. Tetikleyici faktörler ise bir şarta bağlı otomatik çalışan ve akıllı telefonunuzdan yapacağınız işlem ile çalışabilen sistemlerdir.

**Akıllı ev sistemlerinde** bulunabilecek bazı özellikler şunlardır;

- Otomatik ısı sabitleme
- Odalarda aydınlık durumuna göre ışık kontrolü
- Perdelerin kullanıcıya bağlı olarak açılıp açılmama kontrolü
- Garaj kapısı kontrolü
- Hırsız alarm sistemi
- Otomatik toprak sulama sistemi vb.

Akıllı ev sistemleri elektronik kartlar ile oluşturulabilir. Bu kartlardan biri de **Arduino**'dur.

**Arduino**, interaktif projeler geliştirmek için tasarlanan, elektronik donanım ve yazılım temelli bir geliştirme platformudur. Arduino kartlarında bir adet Atmel AVR mikrodenetleyici ve devre bağlantıları için çeşitli elektronik komponentler bulunur.

Yukarıdaki görsel bir Arduino görselidir.

### II. TEMEL BİLGİLER

Program Arduino dilinde geliştirilmiş olup,tümleşik geliştirme programı olarak "Arduino Ide" kullanılmıştır.

### III. YÖNTEM

Proje kodları geliştirilirken kodun temiz ve kolay anlaşılabilir olması amaçlanmıştır.Bu yüzden her sensörün işlemleri farklı fonksiyonlar üzerinde yazılıp daha sonra sonsuz döngü içerisinde bu fonksiyonlar çağırılmıştır. Böylece elimizde daha kolay okunabilir ve anlaşılabilir bir kod bulunmaktadır.

- include Keypad.h  
Keypadi daha efektif bir şekilde kullanmak için içerisinde keypad fonksiyonları bulunan Keypad.h kütüphanesini projemize ekliyoruz.
- include LiquidCrystal.h  
Projede lcd kullanmamız gerekmektedir. Bu yüzden içerisinde lcd fonksiyonlarını bulunduran LiquidCrystal.h kütüphanesini projemize ekliyoruz.
- const int rs=2,en=3,d4=4,d5=5,d6=6,d7=7;  
Yukarıdaki yazdığımız satır bize lcd girişlerinin hangi pinler olduğunu söyleyen değişkenleri bulunduran satırdır.
- LiquidCrystal lcd(rs,en,d4,d5,d6,d7);  
Yukarıda lcdmizin bağlı olduğu pinleri tanımladıktan sonra LiquidCrystal kütüphanesi içerisinde bulunan

lcd fonksiyonumuza girişleri yollayarak lcdimizi oluşturuyoruz.

- const byte pirpin=29;  
Hareket sensörümüzü Arduino kartımıza bağladığımızda hangi pinden giriş alacağımızı tanımladık.
- const byte flamepin=30;  
Yangın sensörümüzü Arduino kartımıza bağladığımızda hangi pinden giriş alacağımızı tanımladık.
- const byte yesilpin=12,kirmizipin=13;  
Kullanıcı tarafından keypad üzerinden şifre girildiğinde eğer şifre yanlış ise kırmızı led yanmalıdır.Eğer şifre doğruysa yeşil led yanmalıdır. Bu yüzden şifrenin doğruluğuna göre yanacak olan ledlerin çıkış yapacağı pini tanımladık.
- const byte lamppin=11;  
Eğer hareket sensörümüzde bir hareket algılanırsa hareket algılandığı an lamp yanmalıdır. Lamp için çıkış pinimizi tanımladık.
- const byte buzzerpin=10;  
Yangın sensöründen aktif bir değer geldiği zaman buzzerin ses çıkarması istenmektedir. Bu yüzden buzzerin ses çıkarabilmesi için buzzera bir çıkış pini tanımlanmıştır.
- void const byte satirpin=4,sutunpin=3;  
Keypadimizde bulunan satır sayısını ve sütun sayısını değişkenler içerisinde aktardık.
- const String parola="1907",kullanicidegeri="";  
Keypad sayesinde kullanıcıdan alınan veri kullanicidegeri adlı değişkenin içerisinde toplanır. Doğru diyeceğimiz parola ise parola değişkeninin içerisinde saklanmaktadır.
- char eleman[satirpin][sutunpin]=...;  
Yukarıda keypadte bulunan elemanları bir matrise karşılık gelecek şekilde tanımladık.
- byte satirpinleri[]=25,26,27,28,sutunpinleri[]=24,23,22;  
Yukarıda keypadin satır ve sütunlarının bağlı olduğu pinler dizi şeklinde gösterdik. Boyuttan tasarruf yapmak için değişkenler byte tipinde tanımladık.
- Keypad keypad1=Keypad(makeKeymap(eleman),satirpinleri,...);  
Keypad kütüphanesinin içerisinde bulunan Keypad fonksiyonuna giriş pinlerimizi,satır ve sütun adetimizi ve keypad elemanlarını tanımladığımız matrisi yollayarak keypadimizi oluşturuyoruz.
- void KeypadFonksiyonu()

Keypad fonksiyonu içerisinde keypad üzerinden kullanıcıdan bir değer almayı bekler. Eğer kullanıcı bir karaktere basmışsa koşullara girmeye başlar. Eğer kullanıcı '(Hashtag)' bastıysa;  
Kullanıcının girdiği değer ile parola değerini karşılaştırır. Eğer bu iki ifade birbirine eşitse yeşil ledi yakar, kırmızı ledi söndürür. Eğer bu iki değer birbirine eşit değilse kırmızı ledi yakar, yeşil ledi söndürür.  
Eğer kullanıcı '\*' bastıysa;  
Kullanıcının şu ana kadar girdiği tüm değerleri temizler. Ve hem kırmızı hem yeşil ledi söndürür.

- void PirFonksiyonu()

Bu fonksiyon hareket sensöründen gelen değeri okur. Eğer değer 1 ise lampin bağlı olduğu pine güç yollayarak lamp'in yanmasını sağlar.  
Eğer değer 0 ise lampin bağlı olduğu pine güç yollamayı lamp'in yanmamasını sağlar.

- void FlameFonksiyonu()

Bu fonksiyon yangın sensöründen gelen değeri okur. Eğer yangın sensöründen gelen değer 1 ise buzzerin çalışması için buzzerin olduğu pine güç gönderilmesini sağlar. Eğer gelen değer 0 ise buzzerin olduğu pine güç göndermeyerek buzzerin çalışmamasını sağlar.

- void DereceFonksiyonu()

Bu fonksiyon analog girişten gelen değeri okur.Daha sonra bu gelen değeri belli formüller sayesinde celsius tipine çevirir.Celsius tipine çevirdikten sonra bazı koşullar bizi karşılar;  
Eğer celsius değeri 20'den küçükse lcd'nin üst kısmına 'Sıcaklık Düştü' yazılır ve alt kısmına da o anki celsius değeri yazılır.  
Eğer celsius değeri 30'dan büyükse lcd'nin üstüne 'Sıcaklık yükseldi' yazılır ve alt kısmına da o anki celsius değeri yazılır.  
Eğer celsius değeri 20 ile 30 arasında ise lcd'ye sadece o anki celsius değeri yazılır.

- void setup()

Serial.begin(9600); ile hızlı iletişim yolu kullanılır.  
lcd.begin(16,2); ile lcdnin enine 16 karakter ve boyuna 2 karakter sığacak şekilde ayarlanır.  
pinMode(yesilpin,OUTPUT); Yeşil pinin çıkış pini olduğu belirlenir.  
pinMode(kirmizipin,OUTPUT); Kırmızı pinin çıkış pini olduğu belirlenir.  
pinMode(lamppin,OUTPUT); Lamp pininin çıkış pini olduğu belirlenir.  
pinMode(buzzerpin,OUTPUT); Buzzer pininin çıkış pini olduğu belirlenir.

- void loop()

İçerisinde her sensörün fonksiyonu teker teker çağınlarak bu döngü sonsuz kez döndürülür. Böylece her sensördeki kontrol sağlanmış olur.

#### IV. KABAKOD

- 1) Gerekli kütüphaneleri dahil et.
- 2) LCD'nin pinlerini değişkenlere ata.
- 3) LiquidCrystal kütüphanesinde lcd fonksiyonu ile lcd yi oluşturun.
- 4) Sensörlerin giriş ve elemanların çıkış yapacak pinlerini tanımla.
- 5) Keypad fonksiyonunda doğru diyeceğimiz ve kullanıcıdan alacağımız değer için değişken tanımla.
- 6) Keypad oluşturun.
- 7) Döngü içerisinde tüm fonksiyonları sırasıyla çalıştır.
- 8) Eğer kullanıcı keypadten veri girmeye çalışırsa KeypadFonksiyonu() fonksiyonunu çağır. Girilen değerin doğru olup olmadığı koşuluna göre ledi yak.
- 9) Eğer hareket sensöründe hareket algılanırsa PirFonksiyonu() fonksiyonunu çağırarak lamp'ı yak.
- 10) Eğer yangın sensöründe yangın algılanırsa FlameFonksiyonu() fonksiyonunu çağırarak buzzeri çalıştır.
- 11) Derece sensöründeki değeri sürekli analogtan okuyarak celsius'a çevir. Gerekli koşullara göre lcd'de çıktısını ver.

#### V. DENEYSEL SONUÇLAR

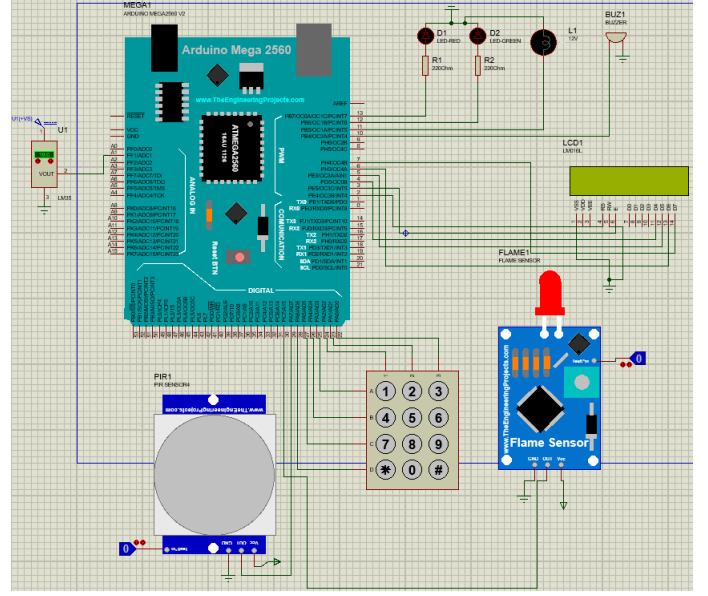


Fig. 1. Projenin genel Proteus Görüntüsü

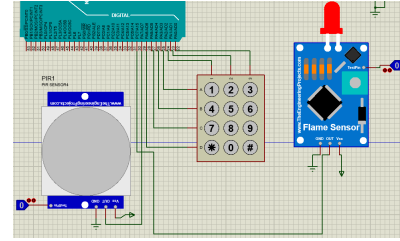


Fig. 2. Hareket, keypad ve yangın sensörü bağlantısı

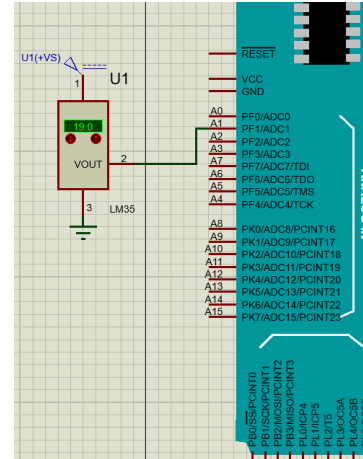


Fig. 3. LM35 sensörü bağlantısı

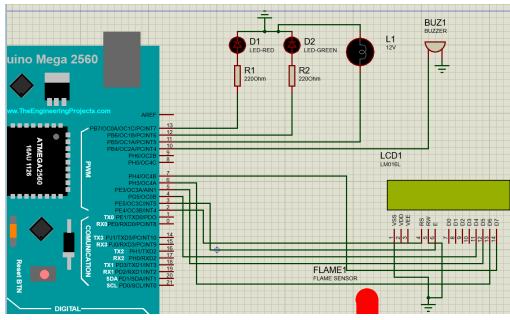


Fig. 4. Çıkışların ve lcd'nin bağlantısı

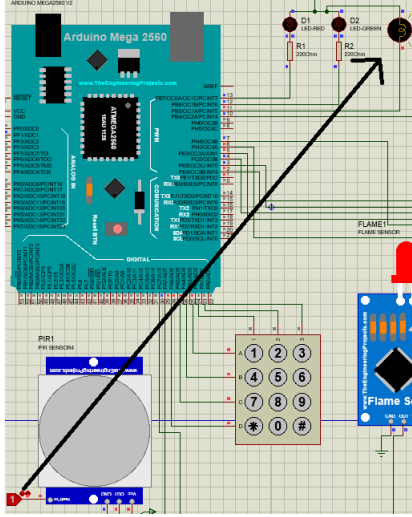


Fig. 5. Hareket sensörü aktifken lamba görüntüsü

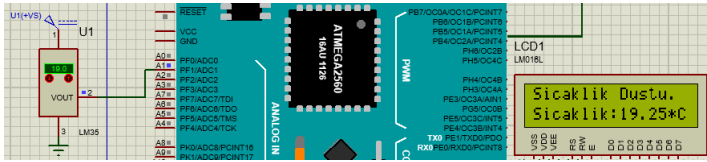


Fig. 6. Sıcaklık 20 derecenin altına düşerse

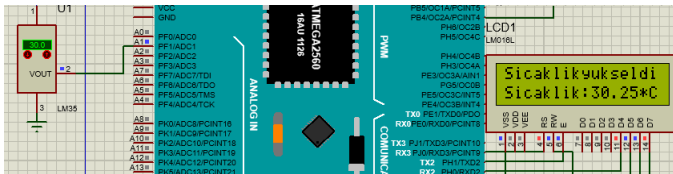


Fig. 7. Sıcaklık 30 derecenin üstüne çıkarsa

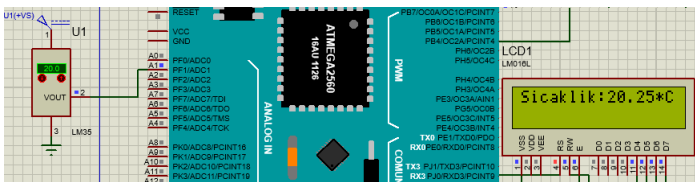


Fig. 8. Sıcaklık değeri 20-30C arasındaysa

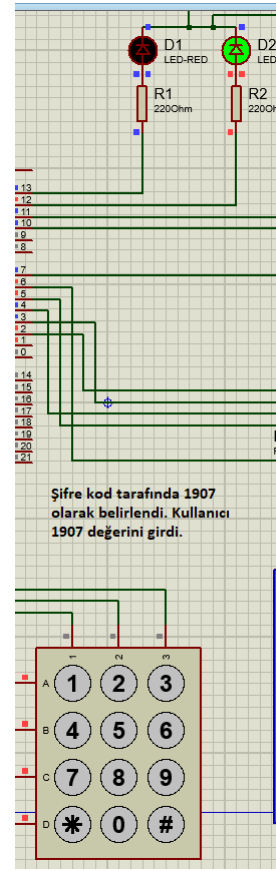


Fig. 9. Eğer kullanıcı doğru şifre girerse

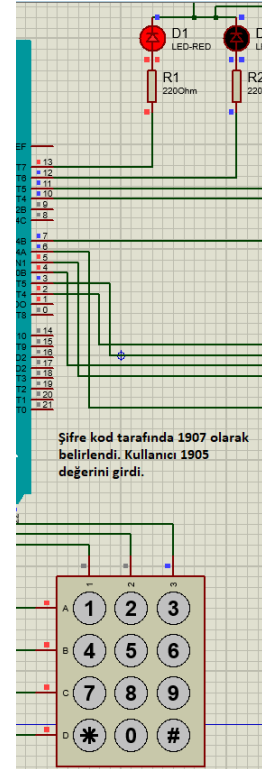


Fig. 10. Eğer kullanıcı yanlış şifre girerse

## I. SONUÇ

Bu proje bana akıllı ev sistemleri hakkında fikir edindirdi. Edindiğim fikirlere dayanarak çıkarım yapacak olursam aslında bu sistemler sayesinde günlük hayatta yaptığımız işlemlerin otomatikleştirilip kendi kendine gerçekleştirilmesinin hayati ne kadar rahat kıldığını gördüm.

Neler öğrendiğim konusuna gelecek olursak;

1)Akıllı ev sistemlerinin bir kart ile nasıl kodlandığını, bağlantılarının nasıl yapıldığını öğrendim ve uyguladım.

2)Arduino kartın ne olduğunu, üzerindeki yapıların ne işe yaradığını öğrendim.

3)Proteusta nasıl bir projenin simüle edileceğini öğrendim.

4)Proteusta bulunmayan kütüphanelerin nasıl eklendiğini öğrendim.

5)Arduino programlama dili öğrendim.

6)Dijital ve analog farkını öğrenerek hangisini kullanmam gerektiği konusunda fikir edindim.

Projeyi geliştirirken;

Kodu temiz yazmaya çalıştım. Her sensörün fonksiyonunu ayrı ayrı yazarak fonksiyonları sonsuz döngü içerisinde çağırdım. Bunun bana ne gibi faydası oldu dersek sensörleri teker teker test ederken hangisinde sorun var ise direkt onun fonksiyonuna giderek sorunumu çözdüm. Bu da bana temiz kodun ne kadar rahatlık sağladığını gösterdi.

## II. KAYNAKÇA

- 1) <https://www.youtube.com/watch?v=pVEa99oUayk>
- 2) <http://arduinoturkiye.com/arduino-mega-2560-nedir/>
- 3) <http://arduinoturkiye.com/kategori/arduinoya-giris/>
- 4) <https://www.bilisimkonulari.com/proteus-isis-ve-ares-kullanimi-resimli-anlatim.html>
- 5) <https://www.circuitstoday.com/proteus-software-introduction>
- 6) <https://www.elektrikport.com/teknik-kutuphane/proteus-ile-arduinosisimulasyonu/12005ad-image-0>
- 7) <https://www.theengineeringprojects.com/2016/01/pir-sensor-library-proteus.html>
- 8) <https://microdigisoft.com/interfacing-temperature-sensor->

lm35-with-arduino-in-proteus/

- 9) <https://forum.arduino.cc/t/how-to-get-lm35-to-read-negative-temperature-with-arduino/407736>
- 10) <http://robojax.com/node/1163>
- 11) <https://www.theengineeringprojects.com/2021/03/arduino-mega-2560-library-for-proteus-v2-0.html>
- 12) <https://www.theengineeringprojects.com/2016/07/flame-sensor-library-proteus.html>