**BURSA TEKNİK ÜNİVERSİTESİ**

**Elanur BAŞARAN**

**(19360859020)**

**LED KÜP**

**FİZİK DERSİ PROJESİ**

|  |
| --- |
| **Mühendisliği ve Doğa Bilimleri** |
|  |

**Proje Danışmanı: Dr. Öğr. Üyesi Zekeriya DOĞRUYOL**

**HAZİRAN, 2020**

**İNTİHAL BEYANI**

Bu proje raporunda sunulan bilgi ve sonuçların akademik ve etik kurallara uyularak tarafımdan gözden geçirilerek belirtildiğini ve belgelediğimi, aksinin ortaya çıkması durumunda her türlü yasal sonucu kabul ettiğimi beyan ederim.

Nazlı Tuğba GÜRSOY

**İÇİNDEKİLER**

**Sayfa**

[KISALTMALAR v](#_Toc47456292)

[SEMBOLLER vi](#_Toc47456293)

[ÇİZELGE LİSTESİ vii](#_Toc47456294)

[ŞEKİL LİSTESİ viii](#_Toc47456295)

[ÖZET……… x](#_Toc47456296)

[1. GİRİŞ 1](#_Toc47456297)

[1.1 Projenin Amacı 1](#_Toc47456298)

[2. GENEL BİLGİ 2](#_Toc47456299)

[2.1 Led Tarihçesi 2](#_Toc47456300)

[2.1.1 Özellikleri 2](#_Toc47456301)

[2.1.2 Çeşitleri 3](#_Toc47456302)

[2.1.3 Kullanım Alanları 4](#_Toc47456303)

[2.1.4 LED Kullanımının Türkiye ve Dünyadaki Yeri 6](#_Toc47456304)

[2.1.4.1 Kore 6](#_Toc47456305)

[2.1.4.2 Çin Halk Cumhuriyeti 6](#_Toc47456306)

[2.1.4.3 Amerika 6](#_Toc47456307)

[2.1.4.4 Türkiye 6](#_Toc47456308)

[2.2 Projenin Parçaları Ve Tasarımı 7](#_Toc47456309)

[2.2.1 Elektronik Geliştirme Kartları Çeşitleri ve Hakkında Bilgi 7](#_Toc47456310)

[2.2.1.1 Arduino Geliştirme Kartı 7](#_Toc47456311)

[2.2.1.2 Raspberry Pi Geliştirme Kartı 7](#_Toc47456312)

[2.2.1.3 The BeagleBone Black Geliştirme Kartı 7](#_Toc47456313)

[2.2.1.4 The Intel Galileo Geliştirme Kartı 8](#_Toc47456314)

[2.2.1.5 The pcDuino Geliştirme Kartı 8](#_Toc47456315)

[2.2.1.6 The Uruk Geliştirme Kartı 9](#_Toc47456316)

[2.2.1.7 The Goldilocks 9](#_Toc47456317)

[2.2.1.8 The Extra Core Geliştirme Kartı 9](#_Toc47456318)

[2.2.1.9 The Spark Core Geliştirme Kartı 10](#_Toc47456319)

[2.2.1.10 DigiSpark Geliştirme Kartı 10](#_Toc47456320)

[2.3 Arduino Nedir? 11](#_Toc47456321)

[2.3.1 Arduino Uno Kartı Tanıtımı 12](#_Toc47456322)

[2.3.2 Teknik Özellikler 12](#_Toc47456323)

[2.3.3 Arduino Uno Kartı İle Yapılabilecek Projeler 13](#_Toc47456324)

[2.4 Yazılım Hakkında Genel Bilgi 14](#_Toc47456325)

[2.4.1 Algoritma Nedir ve Neden Kullanılır? 14](#_Toc47456326)

[2.4.2 Programlama Dilleri Hakkında Genel Bilgi 14](#_Toc47456327)

[2.4.2.1 Arduino’da Kullanılan Programlama Dilleri 15](#_Toc47456328)

[3. TASARIM 16](#_Toc47456329)

[3.1 CAD Çizimleri 16](#_Toc47456330)

[4. MALZEME VE YÖNTEM 17](#_Toc47456331)

[4.1 Kullanılan Malzemeler 17](#_Toc47456332)

[4.1.1 Direnç 17](#_Toc47456333)

[4.1.2 Jumper Kablo 18](#_Toc47456334)

[4.1.3 Montaj Kablosu 19](#_Toc47456335)

[4.1.4 Lehim Pastası 19](#_Toc47456336)

[4.1.5 Havya-Lehim Temizleme Süngeri 19](#_Toc47456337)

[4.1.6 Makaron 20](#_Toc47456338)

[4.1.7 Zımpara Kağıdı 20](#_Toc47456339)

[4.1.8 Lehim Teli 20](#_Toc47456340)

[4.1.9 Havya 21](#_Toc47456341)

[4.1.10 Kablo Soyucu 21](#_Toc47456342)

[4.2 Projenin Yapılışı 22](#_Toc47456343)

[4.2.1 Montaj İşlemleri 22](#_Toc47456344)

[5. SONUÇ 29](#_Toc47456345)

[KAYNAKLAR 30](#_Toc47456346)

[EKLER 31](#_Toc47456347)

[6. PROJE EKİBİ 55](#_Toc47456348)

KISALTMALAR

**STEM** : Science Technology Engineering and Maths

**LED** : Light Emitting Diode

**RGB** : Red Green Blue

**AR-GE** : Araştırma Ve Geliştirme

**SECOL** : Southeastern Conference On Linguistics

**PCB** : Printed Circuit Board

**IDE** : Integrated Development Environment

**USB** : Universal Serial Bus

**RAM** : Random Access Memory

**PWM** : Pulse Width Modulation

**LDR** : Light Dependent Resistance

**NTC** : Negative Temperature Coefficient

**PTC** : Positive Temperature Coefficient

SEMBOLLER

**mA** : Miliamper

**V** : Volt

**Mm** : Milimetre

**MHz** : MegaHertz

**g** : gram

**KB** : Kilobyte

**GB** : Gigabyte

**W** : İş

**R** : Direnç

**Ω** : Ohm

ÇİZELGE LİSTESİ

**Sayfa**

[Çizelge 4.1 : Malzeme listesi. 17](#_Toc42810295)

[Çizelge 6.1 : Kişisel bilgiler. 55](#_Toc42810296)

ŞEKİL LİSTESİ

**Sayfa**

[Şekil 2.1 : LED örneği. 2](#_Toc42868510)

[Şekil 2.2 : LED çeşitleri. 3](#_Toc42868511)

[Şekil 2.3 : LED’in bölümleri. 4](#_Toc42868512)

[Şekil 2.4 : Ledin kullanım alanına örnek. 4](#_Toc42868513)

[Şekil 2.5 : Ledin kullanım alanına örnek. 5](#_Toc42868514)

[Şekil 2.6 : Ledin kullanım alanına örnek. 5](#_Toc42868515)

[Şekil 2.7 : Raspberry pi geliştirme kartı. 7](#_Toc42868516)

[Şekil 2.8 : The beaglebone black geliştirme kartı. 7](#_Toc42868517)

[Şekil 2.9 : The intel galileo geliştirme kartı. 8](#_Toc42868518)

[Şekil 2.10 : The pcduino geliştirme kartı. 8](#_Toc42868519)

[Şekil 2.11 : The uruk geliştirme kartı. 9](#_Toc42868520)

[Şekil 2.12 : The goldilocks. 9](#_Toc42868521)

[Şekil 2.13 : The extra core geliştirme kartı. 10](#_Toc42868522)

[Şekil 2.14 : The spark core geliştirme kartı. 10](#_Toc42868523)

[Şekil 2.15 : DigiSpark geliştirme kartı. 11](#_Toc42868524)

[Şekil 2.16 : Arduino uno kartı tanıtımı. 12](#_Toc42868525)

[Şekil 2.17 : Arduino uno kartı ile yapılabilecek proje örneği. 13](#_Toc42868526)

[Şekil 2.18 : Arduino uno kartı ile yapılabilecek proje örneği. 13](#_Toc42868527)

[Şekil 2.19 : Algoritma şeması. 14](#_Toc42868528)

[Şekil 3.1 : Ledlerin kod ile çalışması. 16](#_Toc42868529)

[Şekil 3.2 : Ledlerin çalışırken farklı görüntüler oluşturması. 16](#_Toc42868530)

[Şekil 4.1 : Direnç. 18](#_Toc42868531)

[Şekil 4.2 : Direnç çeşitleri. 18](#_Toc42868532)

[Şekil 4.3 : Jumper kablo. 18](#_Toc42868533)

[Şekil 4.4 : Montaj kablosu. 19](#_Toc42868534)

[Şekil 4.5 : Lehim pastası. 19](#_Toc42868535)

[Şekil 4.6 : Lehim temizleme süngeri. 20](#_Toc42868536)

[Şekil 4.7 : Makaron. 20](#_Toc42868537)

[Şekil 4.8 : Zımpara kağıdı. 20](#_Toc42868538)

[Şekil 4.9 : Lehim teli. 21](#_Toc42868539)

[Şekil 4.10 : Havya 21](#_Toc42868540)

[Şekil 4.11 : Kablo soyucu. 21](#_Toc42868541)

[Şekil 4.12 : Led yerleşim tablası. 22](#_Toc42868542)

[Şekil 4.13 : Led çalışma testi. 22](#_Toc42868543)

[Şekil 4.14 : Zımparalanmış ve zımparalanmamış led farkı. 23](#_Toc42868544)

[Şekil 4.15 : Zımparalanmış ve bükülmüş ledler. 23](#_Toc42868545)

[Şekil 4.16 : Led yerleşim düzeni. 24](#_Toc42868546)

[Şekil 4.17 : Led artı bacak lehimleme işlemi. 24](#_Toc42868547)

[Şekil 4.18 : Ledlerin lehimlenmesi ile oluşan 1.katman. 25](#_Toc42868548)

[Şekil 4.19 : Ledlerin lehimlenmesi ile oluşan 1.katman. 25](#_Toc42868549)

[Şekil 4.20 : Led’lerin katmanlarının lehimlenmiş hali. 25](#_Toc42868550)

[Şekil 4.21 : Katmanlardan çıkan artı bacakları aşağıya yönlendirilmesi. 26](#_Toc42868551)

[Şekil 4.22 : Arduino’ya jumper kabloların bağlanma çizelgeleri. 26](#_Toc42868552)

[Şekil 4.23 : Eksi bacakların jumper kablolarla birleştirilip arduino’ya bağlanması. 27](#_Toc42868553)

[Şekil 4.24 : Led küpün çalışması sonucu ortaya çıkan görüntüler. 27](#_Toc42868554)

[Şekil 4.25 : Led küpün çalışması sonucu ortaya çıkan görüntüler. 28](#_Toc42868555)

**LED KÜP RAPORU**

ÖZET

Led Küp projesi hem yazılım hem de elektronik alanda bilgiler edin geliştirme olarak da tanımlanabilir. Projenin ana malzemesi olan ledlerin tarihçesini, kullanım alanlarını, çeşitlerini, Dünya genelinde birkaç ülkedeki Led kullanımının ne seviyede olduğu, projenin çalışması için gerekli olan Arduino programlama dili hakkında bilgiler, programlama dilinin ayrıntısı olarak diğer programlama dilleri nelerdir, bu programlama dilleri için gerekli olan algoritmalar için bilgiler toplanıldı ve raporda anlatılmıştır. Projenin yapım aşamasında kullanılan elektronik malzemelerin projenin yapım esnasında ne gibi yarar sağladığı, o malzemelerin ne gibi bir işlevi olduğu anlatılmıştır. Projenin yapım aşaması olarak ledlerin belirli bir zemin yüzeye yerleştirilmesi yerleştirildikten sonra ledlerin bacak farklarına uyum sağlatılmıştır. Bu uyumu CAD çizimleri yapılarak daha anlaşılır bir hale gelmesini ve kolaylıkla lehimleme işinin yapılması raporun içeriğinde belirtilmiştir. Lehimleme 4 katman boyunca sürmüş. Yapılan katmanlar birleştirilip led küpün yapısının oluşturulması bahsedilmiştir. Daha sonra Arduino’ya bağlanma, bu esnada bağlanma şeklinin nasıl olduğu bağlanmadan sonra Arduino’ya hangi kodların yazılması gerektiği raporun ek kısmında belirtilmiştir.Son olarak projenin tamamlanmış halinin dekoratif şekiller ortaya çıkardığı test edilmiş ve ortaya çıkan görüntüler raporun yapım aşamasında yer almıştır

[HAZİRAN, 2020], 65

**Anahtar kelimeler** : LED, Arduino.

# GİRİŞ

## Projenin Amacı

Elektronik severlerin ilgisini çekebilecek Led Küp projesi günlük yaşantımızda güzel ve çeşitli dekoratif görüntü oluşturmalarda kullanılır.

Bu proje inşa ve kodlama aşamasında STEM kabiliyetini geliştirmektedir. (STEM: Science Technology Engineering and Maths).

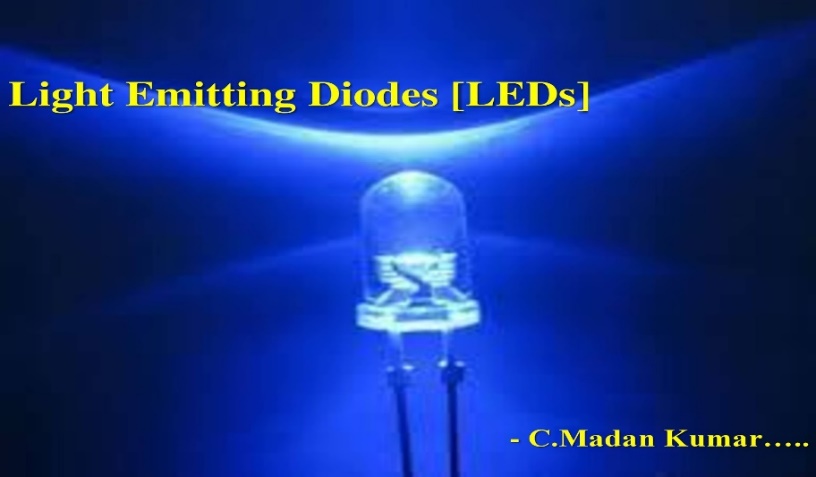
# GENEL BİLGİ

## Led Tarihçesi

Light Emitting Diode (Şekil 2.1) yani “ışık yayan diyot” kavramının kısaltması olarak bilinmektedir. 1907 yılında Marconi Laboratuvarlarında H.J. Round tarafından icat edildi ve yıllar boyunca pratik bir kullanım alanı bulunamadı. Yeni buluşla uğraşan başka bilim adamları da vardı, örneğin Rusya’dan Oleg Vladimirovich ve Amerikan Radyo İstasyonu’ nda çalışan Rubin Braunstein. Ayrıca Texas Enstrümanlarından Gary Pittman ve Bob Biard 1961 yılında infrared LED için patent almış.[1]

1962 yılında ilk kırmızı veya görülebilir spektrum Nick Holonyak Jr, General Elektrikte çalışan bir bilim adamı, tarafından keşfedildi. Holonyak ona “LED ’lerin babası” unvanını kazandırdı. Ve onun bir öğrencisi, M. George Craford, ilk sarı LED’i 1972’de geliştirdi. İlk mavi LED Shuji Nakamura tarafından, ilk beyaz LED Alberto Barbieri tarafından 1995’te geliştirilmiş. LED’ler hem gelişme kaydediyordu hem de daha verimli hale ulaşıyordu.[1]

1980-90’lı yıllar ise led tarihinde bir dönüm noktası olarak görülmüş. Çünkü bu tarih aralığında led lambalar beyaz, yeşil, mavi ve sarı olarak farklı renklerde üretilmeye başlanmıştır ve verimlilikleri arttırılmaya çalışılmıştır. Farklı boyutlarda ve şekillerde üretilebilen led lambaların düşük enerji ile uzun süre kullanılabilmesi çok tercih edilme nedeni olmuştur. Öyle ki bir adet Flaman lambanın kullanım süresi 1500 saat, floresan lambanın kullanım süresi 30.000 saat iken, led lambanın kullanım süresi 50.000 saatten fazla ölçülmüş. Ledlerin önemli bir diğer özelliği de çabuk ısınmaması olarak belirtilir.[2]



Şekil 2.1 : LED örneği.

### Özellikleri

* Doğru akımla çalışırlar.
* Enerji tüketimi düşüktür.
* Kolay kullanıma sahiptir.
* Uzun ömürlüdür.
* Tepki süresi kısadır.
* Işık verimliliği fazladır.
* Isınma süresi uzun olmasına rağmen açığa çıkan ısı miktarı düşüktür.
* İstenilen renkte ve yoğunlukta kullanılabilmektedir.
* Darbe ve sarsıntıya karşı dayanıklıdır.[2]

### Çeşitleri

Çakan (flashing) Ledler standart LED olarak bilinmektedir. Ancak saniyede bir yanıp sönmesini sağlayan multivibratör devresi ile düzenlendikleri zaman çakan Ledler olarak adlandırılabilmektedir. Genellikle tek renk olmalarına karşın iki renkte çakan ya da RGB renk karışımı ile düzenlenmiş tipleri de yer almaktadır.

İki renkli (Bi-color) Ledler bir durum için iki farklı LED kullanılarak yapılır. Aynı kaynaktan beslenen iki paralel bağlantı vardır. Akım bir hattan akarken bir renk, diğer hattan geçerken diğer renk yayılır. Üçüncü bir renk ortaya çıkması için iki rengin uygun frekansta yayılması gerekmektedir. Örneğin yeşil/kırmızı iki LED uygun bir frekans düzenlemesiyle sarı rengi ortaya çıkarmıştır.

Üç renkli Ledler yine iki LED grubundan oluşur. Ancak bu defa Ledler aynı anda ayrı ayrı kontrol etmeyi sağlayan iki hat üzerinden beslenmektedir.

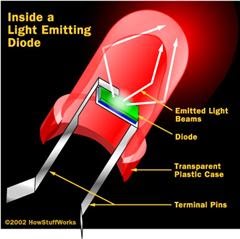
RGB Ledler çok farklı renklerin elde edilebildiği Led olarak görülmektedir. Işığın rengini değiştirmek için kırmızı (Red), yeşil (Green) ve mavi (Blue) renkli Ledlerin birlikte kullanıldığını 3 ana rengin diğer ara renklerin elde edilmesinde de kullanıldığı bilinmektedir. (Şekil 2.2)

Diğer yöntem ise 3 adet LED çipini tek bir kılıf içerisinde barındıran RGB LED kullanmaktır.3 adet LED içeren RGB Ledlerin 4 adet bacağı vardır. Ledlerin bağlantı şekli ortak anotlu veya ortak katotlu olabilmektedir.[19]



Şekil 2.2 : LED çeşitleri.

Diğer diyotlar gibi LED’in yapısında da p-tipi ve n-tipi olmak üzere iki farklı çeşit yarı-iletken madde bulunmaktadır. P-tipi yarı-iletkende pozitif yük taşıyıcılar, n-tipi yarı-iletkende ise negatif yük taşıyıcılar bulunmaktadır. Bu sebepten dolayı, diyot üzerinden yalnızca anottan katot yönünde elektrik akımı geçişi mümkün olmaktadır. (Şekil 2.3).[18]



Şekil 2.3 : LED’in bölümleri.

LED’de ise yine aynı durum geçerlidir. Standart diyotlardan farklı olmak üzere Ledlerde, p ve n tipi yarı-iletkenlerin birleştiği noktadaki elektron alış-verişi, ışık oluşmasına sebep olmaktadır. Bu olaya elektrolüminesans adı verilmektedir.[18]

### Kullanım Alanları

LED ‘ler önceleri yüksek maliyetliydi. Bu yüzden dekoratif olarak kullanılmaktaydı. Her geçen gün gelişen teknolojinin de etkisiyle LED teknolojisi de gelişmiştir. Yapılan çalışmalar sonucunda LED kullanım avantajlarının artışı ve çevreye olan olumlu etkilerinin sonucunda kullanım alanları artmaktadır. Gelişmiş ülkelerde rutin aydınlatmalarda da kullanılmaktadır. Ülkemizde de bu alanda Ledlerin genel aydınlatma için kullanılmasına yönelik çalışmalar yapılmaktadır. Ledlerin kullanım alanları başlıca aşağıdaki belirtilmiştir;

* Fabrikalar
* Alışveriş merkezleri
* Sokak ve cadde aydınlatmaları (Şekil 2.4)



Şekil 2.4 : Ledin kullanım alanına örnek.

* Restaurantlar
* Tarihi eser aydınlatmaları ve cami aydınlatması



Şekil 2.5 : Ledin kullanım alanına örnek.

* Plazalar
* Fabrika çevre aydınlatmaları
* İş Merkezleri
* Dış duvar aydınlatmaları
* Barlar
* Hastaneler
* Otopark aydınlatmaları
* Genel kullanım amaçlı binalar
* Karayolları ve çevre yolları aydınlatmaları
* Köprü, geçit vb. yapıların aydınlatılması (Şekil 2.6)



Şekil 2.6 : Ledin kullanım alanına örnek.

* Mağazalar
* Konferans salonları
* Oteller [20]

### LED Kullanımının Türkiye ve Dünyadaki Yeri

LED kullanımında dünyanın gelişmiş ülkeleri adeta birbirleri ile yarış içerisinde olduğu gözlemleniyor. Bu ülkelerden bazıları;

* Kore
* Çin Halk Cumhuriyeti
* Amerika
* Türkiye [21]

#### Kore

Kore LED üretimi ve Ledin geliştirilmesi üzerine oldukça gelişme kat etmiş durumdadır. Kore Ledler için kimyasalların hazırlanıp hazır çip (ledin ışık veren Core kısmı) halde dünyada led üreticilerinin birçoğuna destek ve stok sunmaktadır. 2000 yılından bu yana patent almış 58 patentli led modelinin neredeyse yarısı Koreli firmalara aittir. Kore LED ürünlerine dönüşümünü neredeyse tamamlamış maksimum enerji tasarrufu sağlanmıştır.[21]

#### Çin Halk Cumhuriyeti

Çin Halk Cumhuriyeti hem LED üretiminde hem LED ürünler ile ilgili tasarım ve AR-GE de hem de bu tasarımların ürün haline getirmede oldukça ileri seviyelerde yer alıyor.

Çin Halk Cumhuriyeti stadyumlar, otoyollar, fabrikalar, kamu binalarında ve evlerde LED dönüşümünü %70 oranında tamamlamış vaziyettedir.[21]

#### Amerika

Amerika LED ile ilgili AR-GE faaliyetlerini tamamlamış, bu konuda uzmanlaşmış tercih edilen bir LED üretici markaya sahiptir. Bu firmanın gözetiminde ürünler geliştiren firmaların ürünleri ile LED dönüşümünü hızlı şekilde devam ettirmektedirler.[21]

#### Türkiye

Türkiye LED dönüşümüne Ledli ürünlerin kullanımına önem vermekte bu konu ile ilgili yasal düzenlemeyi hazırlamıştır. SECOL, ülkede LED üretimi konusunda ilk ve tek yerli firma olma unvanına sahiptir. Şirketin iki yıldan fazla süren araştırmaları ve çalışmaları sonucunda 2009 yılının ilk çeyreğinde gerekli ilgiyi görememiştir. Ürettiği Led'i kullanarak ürün haline getirecek sektör ve alt üretici yetersizliği nedeniyle istediği yerlere maalesef ulaşamamıştır. Ve bu konudaki birçok girişimci bu pazar sıkıntılarından dolayı bu hevesleri kırılmaktadır. Bu konuda birçok kurum kuruluş ve özel sektöre görev düşmektedir.

Türkiye led ve ledli ürünleri maalesef sadece tüketir durumdayız. Ve sektörel olarak sadece ciddi paralar kazanmak için yapılan yatırımların birçoğunda ülkeye kalitesi düşük ledli ürünler ithal edilmekte ve enerji tasarrufu sağlayacakken hüsranla sonuçlanıp maddi israfa yol açmaktadır.[21]

## Projenin Parçaları Ve Tasarımı

### Elektronik Geliştirme Kartları Çeşitleri ve Hakkında Bilgi

#### Arduino Geliştirme Kartı

Arduino geliştirme kartı, dünyanın en ünlü açık kaynaklı geliştirme kartı olarak karşımıza çıkmaktadır. Günümüzde, PCB çiziminden tüm kütüphaneleri açık kaynak olarak kullanıcılarla paylaşılmaktadır.[5]

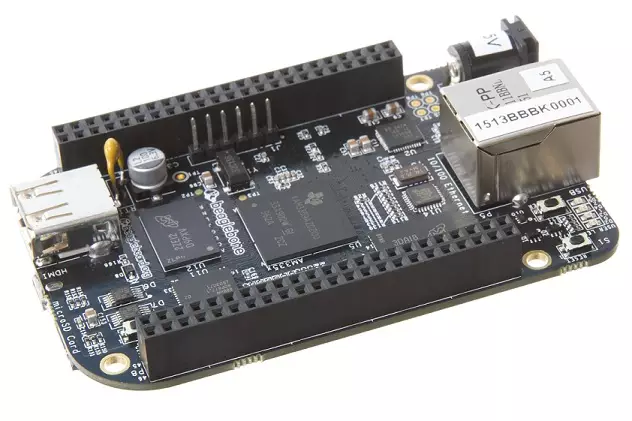
#### Raspberry Pi Geliştirme Kartı



Şekil 2.7 : Raspberry pi geliştirme kartı.

Raspberry Pi(Şekil 2.7), ufak boyutları ile dünyanın en küçük bilgisayarlarından biri olarak karşımıza çıkmaktadır. Üzerindeki 1 GB RAM’i, işlemcisi, USB ve HDMI portları ile tam donanımlı bir bilgisayar da denilebilmektedir.[5]

#### The BeagleBone Black Geliştirme Kartı



Şekil 2.8 : The beaglebone black geliştirme kartı.

BeagleBone Black (Şekil 2.8), Raspberry Pi geliştirme kartı ile aynı kategoride değerlendirilebilecek bir karttır. Raspberry Pi 3’e göre daha iyi bir işlemcidir. Kullanıma açık daha fazla pin sayısı ve fazladan 2 GB kullanılabilir dahili hafızası ile Raspberry Pi’ den daha iyi bir şekilde görünür olsa bile multimedya ve fiyat açısından bakıldığında Raspberry Pi’nin gerisinde kalmaktadır.[5]

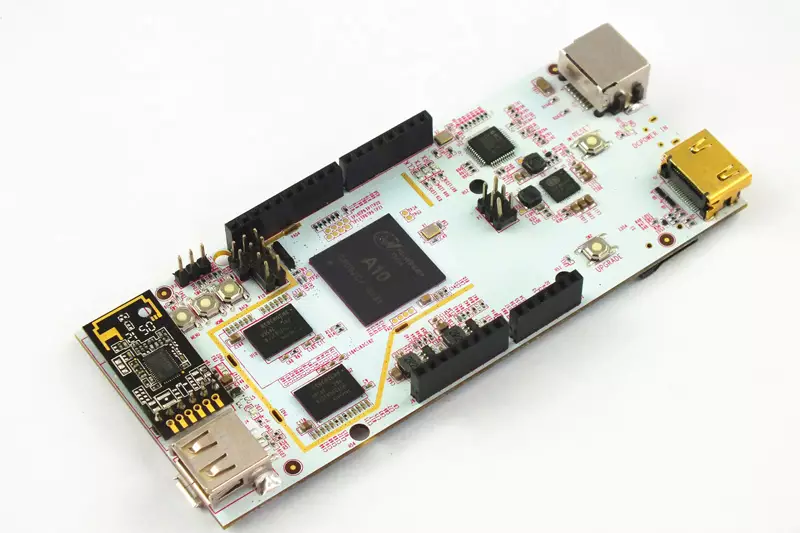
#### The Intel Galileo Geliştirme Kartı



Şekil 2.9 : The intel galileo geliştirme kartı.

Intel Galileo (Şekil 2.9) yine Raspberry Pi ve BeagleBone’den sonra en iyi alternatif olarak görülmektedir. Linux, Mac OS, Windows 7, Windows 8 ile uyumludur. C tabanlı açık kaynak yazılımı ile gelen geliştirme kartı birçok programlama dili tarafından desteklenmektedir. En büyük özelliği ise Arduino IDE ve Shield’ ler ile olan uyumluluğudur. İntel Quark işlemcinin güç verdiği kart, gerek etiket fiyatı gerekse boyut nedeniyle kullanıcıları diğer seçeneklere itmektedir.[5]

#### The pcDuino Geliştirme Kartı

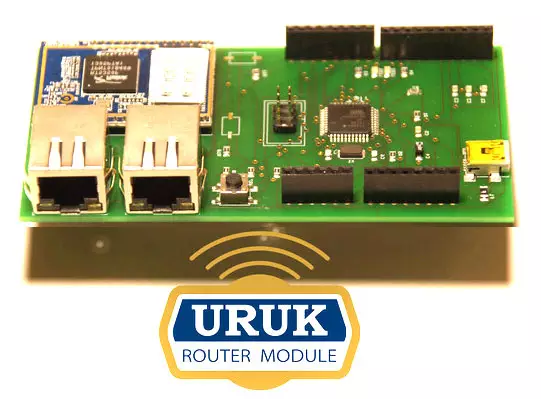


Şekil 2.10 : The pcduino geliştirme kartı.

PcDuino (Şekil 2.10), Raspberry Pi’nin bazı eksikliklerini kapatacak şekilde tasarlanmış bir geliştirme kartı olarak karşımıza çıkmaktadır.

Üzerindeki işlemci sayesinde birçok Linux işletim sistemlerini desteklemektedir. Kartın orta kısmında Arduino’ların pin dizimine benzer pinleri bulunmaktadır.Ayrıca 2 GB dahili hafıza, 1 GB RAM'e sahip olan pcDuino, grafik işlemci olarak Android telefonlarda görmeye alışık olduğumuz Mali 400’ü barındırmaktadır.[5]

#### The Uruk Geliştirme Kartı



Şekil 2.11 : The uruk geliştirme kartı.

Uruk Router Modülü (Şekil 2.11), ATMega32u4 işlemcisi bulunan Arduino IDE ile uyumlu bir Wi-Fi router olarak karşımıza çıkmaktadır. Herhangi bir ekstra kütüphane kullanılmasına gerek duyulmadan doğrudan Wi-Fi özelliği aktif olabilmektedir. Wi-Fi ve Ethernet üzerinden internete bağlanmayı sağlayan Uruk, aynı zamanda access point olarak da kullanılabilmektedir.[5]

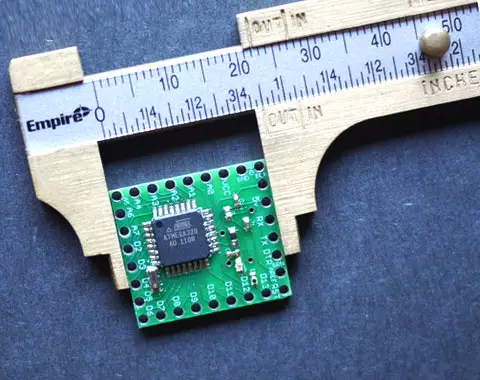
#### The Goldilocks



Şekil 2.12 : The goldilocks.

Goldilocks (Şekil 2.12) aslında Arduino’nun klonu olarak çıkmış bir geliştirme kartı olarakta bilinmektedir. Tek farklılığı Arduino UNO ve Mega gibi düşük hızlı kartlardan daha fazla SRAM sunmasıdır.[5]

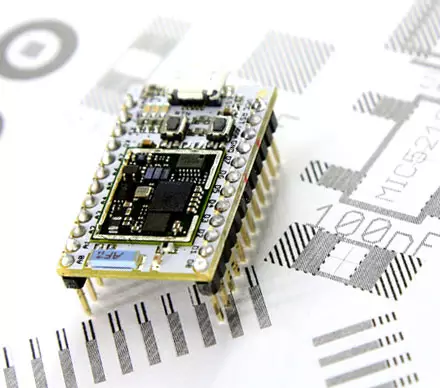
#### The Extra Core Geliştirme Kartı



Şekil 2.13 : The extra core geliştirme kartı.

ExtraCore geliştirme kartı (Şekil 2.13), Arduino IDE ile uyumlu 1 inçlik en ve boy uzunluğuna 1,7 gr ağırlığa sahip geliştirme kartıdır. 22 tane I/O pinlerine sahiptir.[5]

#### The Spark Core Geliştirme Kartı



Şekil 2.14 : The spark core geliştirme kartı.

Spark Core (Şekil 2.14), ufak boyutlara sahip Arduino IDE destekli bir geliştirme kartıdır. Dahili Wi-Fi modülü bulunan kart, aynı zamanda bulut servisine de sahiptir. Bu özellikleri ile Arduino kartlarından ayrılmaktadır.[5]

#### DigiSpark Geliştirme Kartı



Şekil 2.15 : DigiSpark geliştirme kartı.

DigiSpark (Şekil 2.15), Attiny85 tabanlı Arduino IDE uyumlu karttır. 6 adet dijital I/O pinlerine sahiptir. Aynı zamanda üreticisi tarafından ufak boyutlarda üretilmiş Shield’ler de bulunmaktadır.[5]

## Arduino Nedir?

Öncelikle Arduino'nun ortaya çıkış şeklini belirtelim. İlk defa 1995 yılında İtalya’nın Evrea şehrinde bulunan İnteractive Design İnstitute kurumunda öğretim üyeliği yapan Massimo Banzi tarafından dile getirilmiş. Massimo Banzi ’nin amacı öğrencilerin projelerinde kullanmak için ucuz, kullanımı kolay ve güvenilir bir kart tasarımı yapmaktı. Ortak amaçları olan Malmö Üniversitesinden gelen David Cuartiellese ile çalışarak Arduino’yu ürettiler.

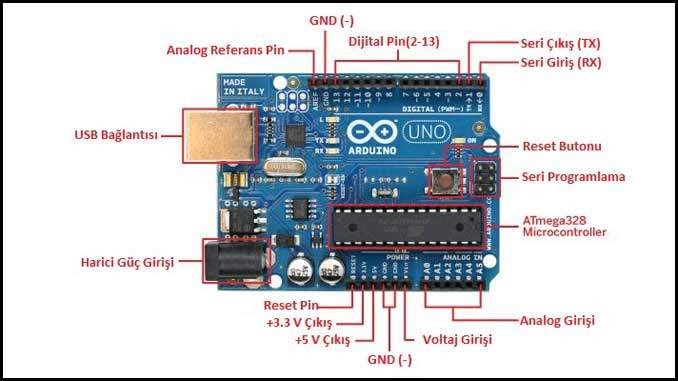
Arduino bulunduğumuz zamanda kullanım açısından damga vurmuş bir programlama platformu olarak karşımıza çıkmaktadır.Arduino içinde giriş-çıkış pinleri, mikrodenetleyici, osilatör gibi elektronik bileşenleri bulunduran basit bir bilgisayar olarak da adlandırabilir.

Kullanımı kolay, açık kaynak kodlu yazılım ve donanıma sahip bir mikrodenetleyici prototipleme platformu olarak da bilinmektedir.

Kullanımının kolay olması ve açık kaynak kodlu yazılıma sahip olması kullanıcılar için çok önemlidir. Çünkü bunlar kodlara erişiminin serbest olması ve isteğe göre değiştirilebilir anlamını ifade etmektedir.

Bu geliştirme platformunun içinde değişik Arduino kartlar, birçok işlem yapmayı sağlayan eklentiler ve program yazmayı sağlayan Arduino IDE bulunur. Bizlerde bunları kullanarak Arduino kartı yapabiliriz.[17]

### Arduino Uno Kartı Tanıtımı



Şekil 2.16 : Arduino uno kartı tanıtımı.

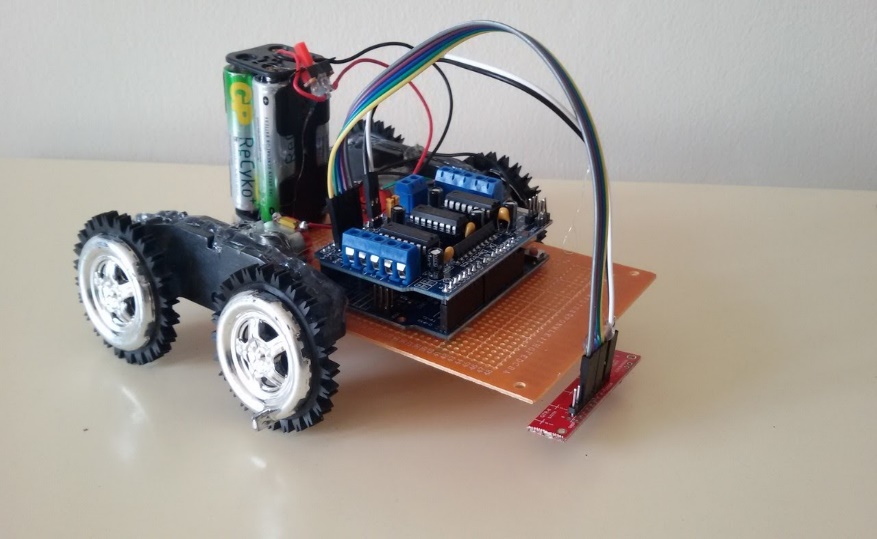
### Teknik Özellikler

Arduino Uno üzerinden bulunan resetlenebilir sigorta, bilgisayarınızın USB portunu kısa devreden veya aşırı akım tüketimi durumlarından korumaktadır. Kart, USB portu üzerinden 500 mA’ den fazla akım çektiğinde otomatik olarak USB ’den aldığı gücü koruma amacıyla kullanmaktadır. Fazla akım durumu veya kısa devre ortadan kalktığında sigorta normal konuma döner ve tekrar bağlantı (Şekil 2.16) kurulur. Genel özellikleri şöyledir:

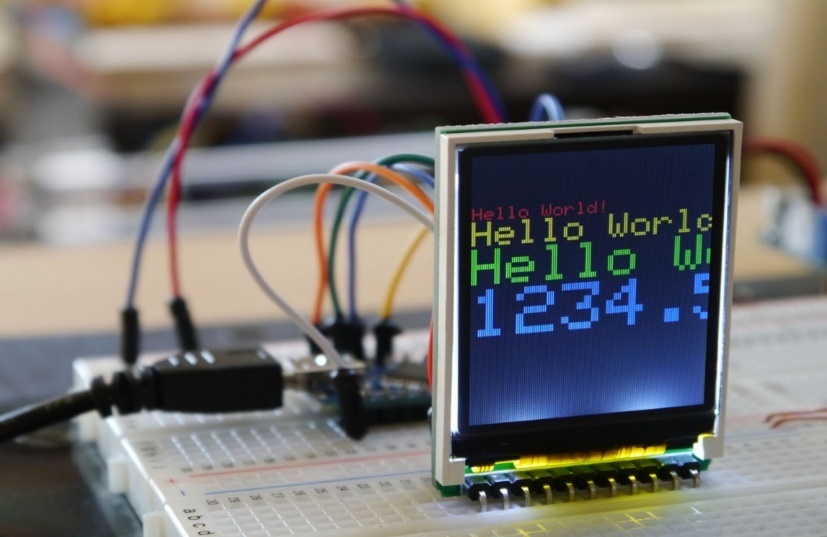
* **Mikrodenetleyici** : ATmega328
* **Çalışma Gerilimi** : 5V
* **Giriş Gerilimi (önerilen)** : 7-12V
* **Giriş Gerilimi (limit)** : 6-20V
* **Dijital Giriş/Çıkış Pinleri** : 14 (6 tanesi PWM çıkışı)
* **Analog Giriş Pinleri** : 6
* **Her Giriş/Çıkış için Akım** : 40 mA
* **3.3V Çıkış için Akım** : 50 mA
* **Flash Hafıza** : 32 KB (ATmega328)
* **SRAM** : 2 KB (ATmega328)
* **EEPROM** : 1 KB (ATmega328)
* **Saat Hızı** : 16 MHz
* **Uzunluk** : 68.6 mm
* **Genişlik** : 53.4 mm
* **Ağırlık** : 25 g [16]

### Arduino Uno Kartı İle Yapılabilecek Projeler

* Robotik uygulamalar (Şekil 2.17)
* Medikal uygulamalar
* Mobil uygulamalar (Şekil 2.18)
* Giyilebilir uygulamalar
* Kablosuz haberleşme uygulamaları
* Algoritmik uygulamalar
* RFID uygulaması (radyo frekansı ile tanımlama teknolojisi)
* Ev otomasyonu

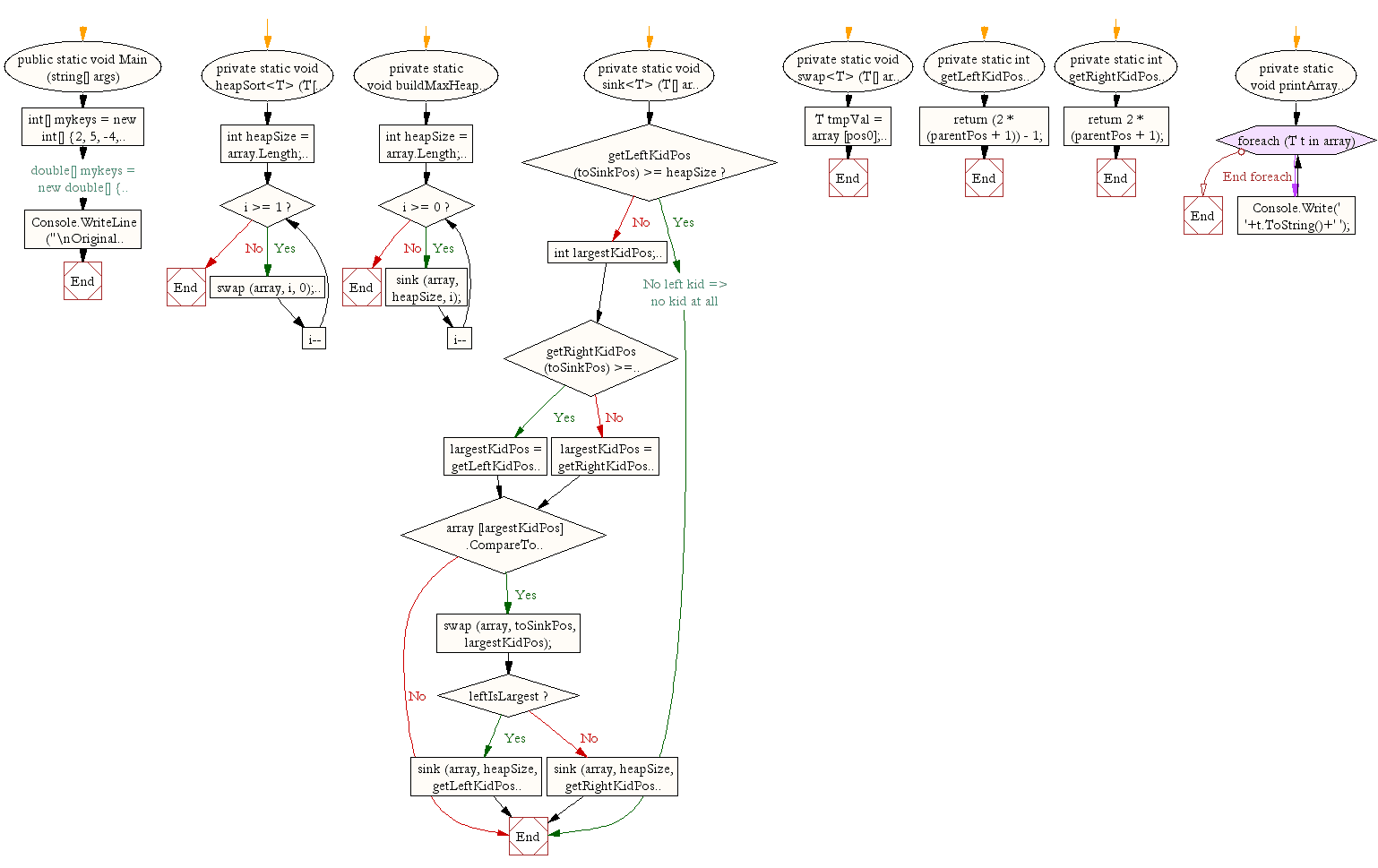


Şekil 2.17 : Arduino uno kartı ile yapılabilecek proje örneği.



Şekil 2.18 : Arduino uno kartı ile yapılabilecek proje örneği.

## Yazılım Hakkında Genel Bilgi



Şekil 2.19 : Algoritma şeması.

### Algoritma Nedir ve Neden Kullanılır?

Bir sorunu çözmek veya belirlenmiş bir hedefe ulaşmak için tasarlanan çözüm yolları, takip edilen işlem basamaklarına algoritma denmektedir. Algoritmalar (Şekil 2.19) açıkça belirtilmiş bir başlangıcı ve sonu olan işlemler kümesi olarak da bilinmektedir. Amaca ulaşılmak için işlenecek çözüm yolları ve sıralamaları belirlenir ve algoritma bu sırayı takip ederek en mantıklı çözüme ulaşmaya çalışır. Bilgisayar biliminde algoritmalar yazılımda da karşımıza çıkmaktadır. Bilgisayarda offline olarak yapılan bütün işlemler, çalıştırılan programlar, web siteleri, mobil uygulamalar, oynanan oyunlar, robotlar,vb. örnek verilebilmektedir.Kısacası sanal ortamdaki her işlem algoritmalar ile gerçekleşiyor ve bunlarda bizlere sunulmaktadır.[4]

### Programlama Dilleri Hakkında Genel Bilgi

Tüm programlama dillerinin temelinde algoritma yer almaktadır. Algoritmalar, programlama dillerinin yardımıyla uygunabilirler.

Programlama dili, yazılımda işlenecek algoritmayı ifade etme yoludur.Tüm elektronik cihazlar ve bilgisayarlar sadece “1” ve “0” rakamlarının bulunduğu binary makine dilinde işlem yapmaktadırlar. Bu nedenle yazılım hangi dilde yazılırsa yazılsın bir derleyici tarafından makine diline çevrilip işlenir.

Programlama dilleri 3 gruptur.

* Alt Seviye Programlama Diller
* Orta Seviye Programlama Diller
* Yüksek Seviye Programlama Dilleri

Olarak bilinmektedir. Bunların içeriğini de aşağıda belirtilenler doldurmaktadır:

Python, C, Java, C++, C#, R, JavaScript, PHP, Go, Swift, Arduino, Ruby, Assembly, Scala, Matlab, HTML, Shell, Perl, Visual Basic, Cuda. [4]

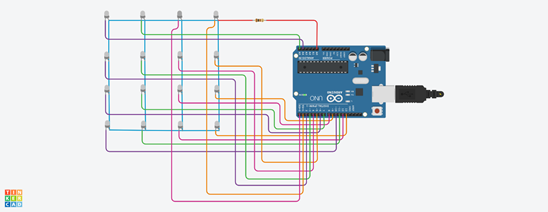
#### Arduino’da Kullanılan Programlama Dilleri

Arduino geliştirme ortamı (IDE), Arduino bootloader (Optiboot), Arduino kütüphaneleri, AVRDude (Arduino'da mikrodenetleyici programlayan yazılım) ve derleyiciden (AVR-GCC) oluşmaktadır.

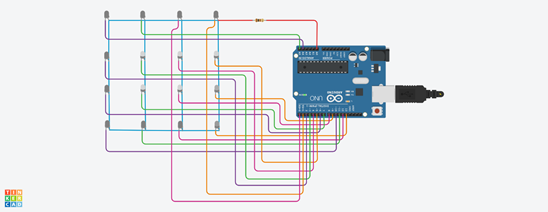
Arduino yazılımı, geliştirme ortamı (IDE) ve kütüphanelerden oluşur. IDE, Java programlama dilinde yazılmıştır ve Processing adlı dilin ortamına dayanmaktadır. Kütüphaneleri ise C ve C++ dillerinde yazılmıştır ve derleyici olarakta AVR-GCC ve AVR Libc. kullanılmıştır. Arduino'nun Optiboot bileşeni bootloader bileşenidir. Bu bileşen, Arduino kartlarının üzerindeki mikrodenetleyicinin programlanmasını sağlayan bileşen olarak karşımıza çıkar. [15]

# TASARIM

## CAD Çizimleri



Şekil 3.1 : Ledlerin kod ile çalışması.



Şekil 3.2 : Ledlerin çalışırken farklı görüntüler oluşturması.

# MALZEME VE YÖNTEM

## Kullanılan Malzemeler

Çizelge 4.1 : Malzeme listesi.

|  |  |  |
| --- | --- | --- |
| Miktarı | Malzemenin Adı | Fiyatı |
| 80 adet | 5 mm Şeffaf beyaz led | 23,43 ₺ |
| 10 adet | ¼ W 100R Direnç | 0.37 ₺ |
| 40 pin | Ayrılabilen M-M jumper kablo 200 mm | 4.39 ₺ |
| 50 gr | Lötfett lehim pastası | 13,18 ₺ |
| 1 adet | Hayva-lehim temizleme süngeri | 4,39 ₺ |
| 1 metre | 1.5/1.6 Makaron | 1,61 ₺ |
| 15 metre | Montaj kablosu tek damar mavi | 6,53 ₺ |
| 1 adet | Arduino uno R3 klon USB kablolu | 24,45 ₺ |
| 2 adet | Zımpara kağıdı | 4 ₺ |
| 2 metre | Lehim teli | 16 ₺ |
| - | Havya | - |
| - | Kablo soyucu | - |

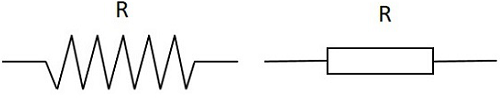
Son 4 malzeme hariç diğerleri www.robotistan.com web sitesinden alınmıştır.

Lehim teli nalburdan, zımpara kağıdı Görükle çarşından alınmıştır.

### Direnç

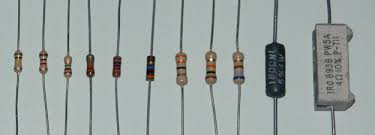
Elektrik devrelerinde direnç (Şekil 4.1), bir iletken üzerinden geçen elektrik akımının karşılaştığı zorlanma olarak belirtilir.

Mekanik sistemlerdeki sürtünmeye benzer özellikler de gösterir. Direncin birimi Ohm (Ω)’dur. Denklemlerde R harfi ile belirtilebilir. Direnç sembolü 2 farklı şekilde gösterilebilir. Bunlar ;



Şekil 4.1 : Direnç.

Elektrikli devrelerde dirençler,akımı sınırlayarak belli bir değerde tutmaya yardımcı olurlar. Bunun haricinde hassas devre elemanlarının üzerinden yüksek akım geçmesini önlemektedirler, besleme gerilimini ve akımı bölmek için de kullanılabilmektedirler. Farklı tipteki bazı dirençler (Şekil 4.2) (LDR,NTC,PTC gibi), dış ortamdaki fiziksel değişimleri kontrol ederek pasif sensör görevi görürler. Ayrıca dirençlerin üzerlerine düşen akım değerleri arttıkça ısınmalarından da faydalanılmaktadır.[14]



Şekil 4.2 : Direnç çeşitleri.

### Jumper Kablo



Şekil 4.3 : Jumper kablo.

Bu tarz bağlantı kabloları (Şekil 4.3), özellikle devre tahtası ile Arduino, EasyPİC gibi geliştirme kartlarının bir arada kullanıldığı devreler için oldukça uygundur. Uçlarında dişi ve erkek girişlerinin olduğu üç çeşidi bulunmaktadır.

* Erkek-erkek
* Erkek-dişi
* Dişi-dişi.

Bağlantı yapacağımız girişlere göre, bu çeşitlerden uygun olanları seçilebilmektedir. Farklı çeşitleri bir arada kullanılabilmektedir Bu da kabloları birbirine bağlayarak daha uzun iletim kabloları oluşturmamıza da olanak sağlamaktadır.[12]

### Montaj Kablosu



Şekil 4.4 : Montaj kablosu.

### Lehim Pastası

Lehim pastası (Şekil 4.5) daha güzel görünümlü bir lehimleme yapmak için gerekli malzemelerden biridir.

Lehim pastasının kullanılmasının amacı metal yüzde temizlemeden oluşan ve ısınmadan kaynaklanan oksitlenmeleri önlemek amacıyla kullanılmalıdır. Lehim yapılacak yerde gözle görülemeyecek kadar yağ tabakası vardır. Bu sayede yağ tabakları temizlenir ve kusursuz bir lehimleme yapılmasını sağlar. Lehim yapılacak bacağın veya telin her yerine dağılması önemlidir.[9]



Şekil 4.5 : Lehim pastası.

### Havya-Lehim Temizleme Süngeri

Havya temizleme süngeri (Şekil 4.6) hafifçe ıslatılarak kullanılır. Havyanın ucunda birikmiş olan lehim parçacıklarını temizlemede kullanılır.Lehim işlemini kolaylaştırır havyanın ömrünü uzatır.[7]



Şekil 4.6 : Lehim temizleme süngeri.

### Makaron

Makaron (Şekil 4.7), ısı etkisi ile belirli bir oranda daralarak çapı küçülen özel izolasyon malzemesidir. Daralan makaronlar, koruma, elektriksel izolasyon, etiketleme, sınıflama ve benzeri operasyonlarda güvenle kullanılan malzemelerdir.[12]



Şekil 4.7 : Makaron.

### Zımpara Kağıdı

Zımpara kağıdı (Şekil 4.8) üstüne zımpara tozu yapıştırılmış kalınca bir kâğıttır. Ledlerin yüzünü aşındırıp düzeltmeye ve parlatmaya yaramaktadır.Bu sayede ledlerin parlaklıkları daha canlı gözükmektedir.[8]



Şekil 4.8 : Zımpara kağıdı.

### Lehim Teli

Elektronik devrelerde bir sistemi oluşturmak amacıyla elemanları ve telleri birbirine tutturmak için belirli sıcaklıklarda eriyebilen tellere lehim teli denir.[10]



Şekil 4.9 : Lehim teli.

### Havya

Havya (Şekil 4.10),elektronik elemanların birbirine lehimlemesinde kullanılan bir eleman olarak bilinmektedir Lehimleme için yüksek ve hızlı bir ısı kaynağına ihtiyaç vardır. Bu ihtiyacı karşılamak üzere havyalar kullanılmaktadır. Havyalar 200 ile 500 derece arasında ısı yayabilecek şekilde üretilmektedirler.[11]



Şekil 4.10 : Havya

### Kablo Soyucu

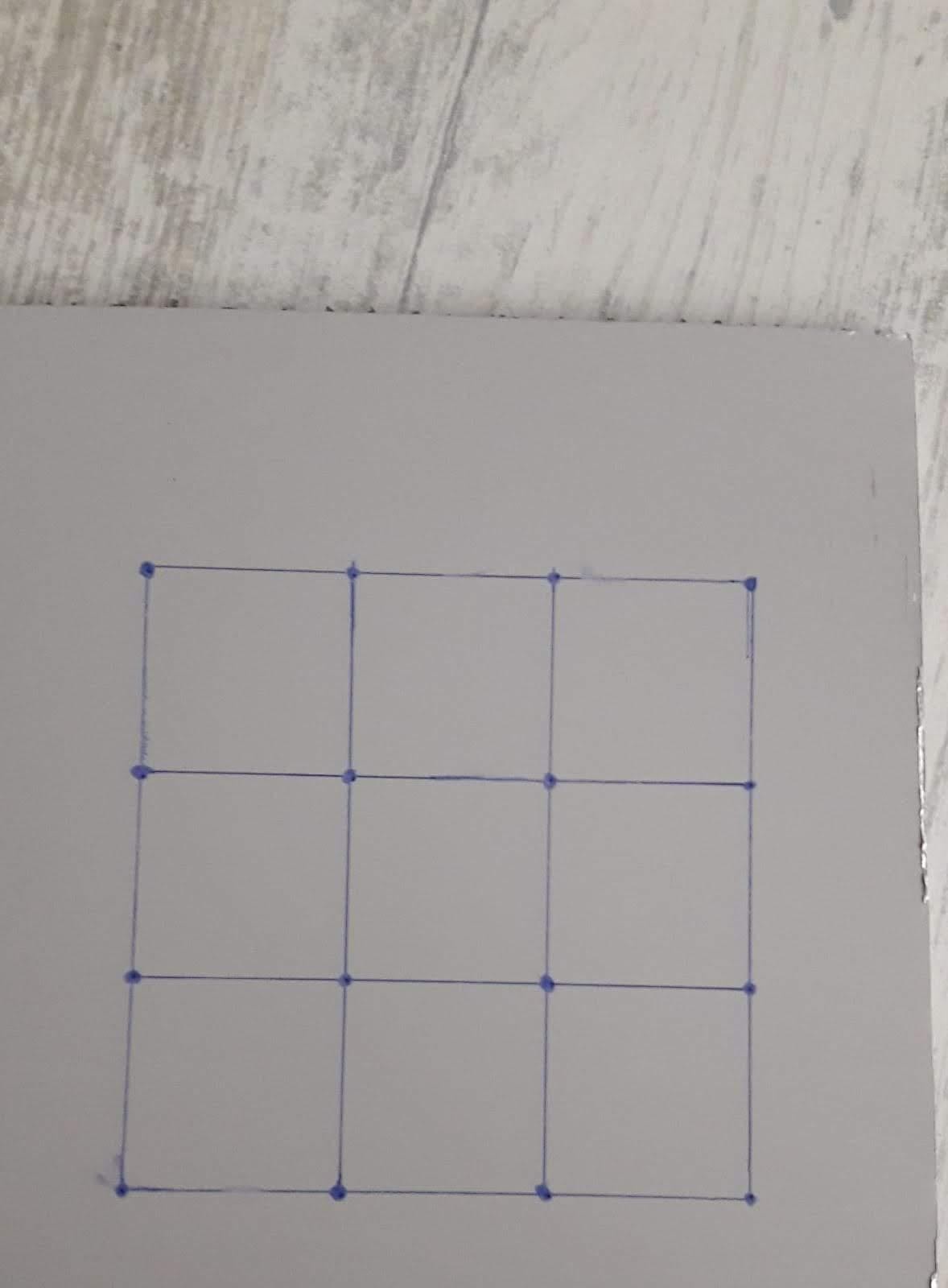


Şekil 4.11 : Kablo soyucu.

## Projenin Yapılışı

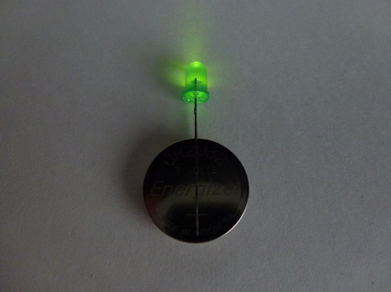
### Montaj İşlemleri

İlk olarak 4\*4\*4 Led Küpün yerleşimi için ahşap bir parçanın üzerinde delikler belirlendi. Deliklerin arası yaklaşık 2,7 cm olarak ölçüldü. (Şekil 4.12)



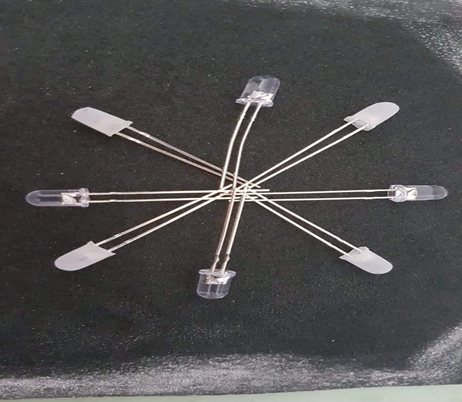
Şekil 4.12 : Led yerleşim tablası.

Oluşan toplam 16 deliğin tümü matkap yardımı ile açıldı. Alınan ledler düğme pil kullanarak çalışıp çalışmadığı test edildi. Yapılan adım projenin sonunda arıza çıkmasını önüne geçti (Şekil 4.13).



Şekil 4.13 : Led çalışma testi.

Sonrasında bütün ledler çevresine daha çok ışık vermesi için zımpara kağıdı ile zımparalandı (Şekil 4.14).



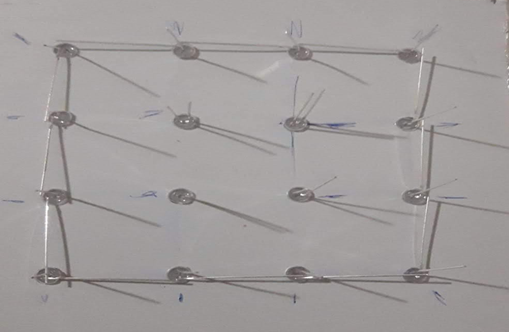
Şekil 4.14 : Zımparalanmış ve zımparalanmamış led farkı.

Daha sonra ahşap parçanın üzerine ledin anot bacakları dışarı gelecek biçimde pense ile büküldü (Şekil 4.15). Bükülme işlemi yaparken ledlerin hepsinin uzunluklarının eşit olmasına sebep oluyor.

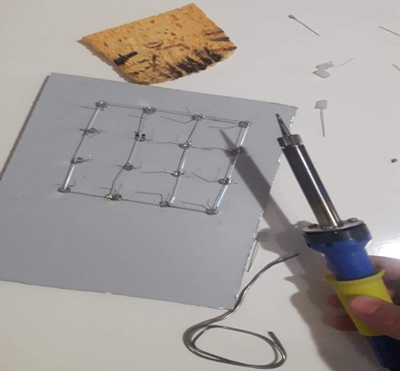


Şekil 4.15 : Zımparalanmış ve bükülmüş ledler.

Ahşap üzerindeki deliklere 16 tane bükülmüş led katot bacakları birleşecek şekilde led yerleşim düzenine göre ayarlandı (Şekil 4.16). Birleşim yerleri havya ile lehimlendi (Şekil 4.17).

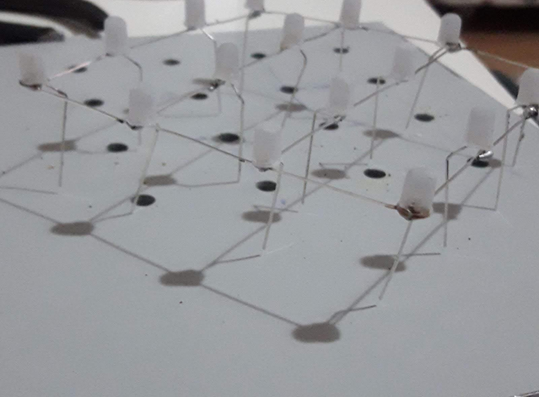


Şekil 4.16 : Led yerleşim düzeni.

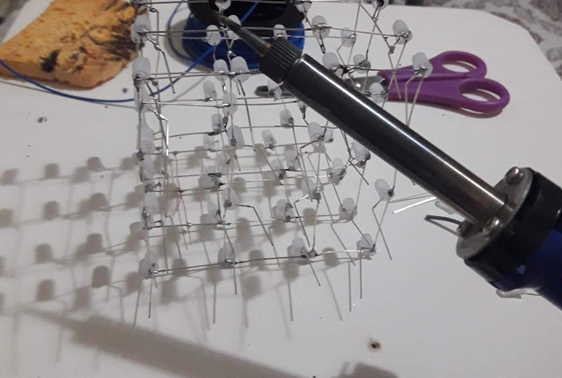
****

Şekil 4.17 : Led artı bacak lehimleme işlemi.

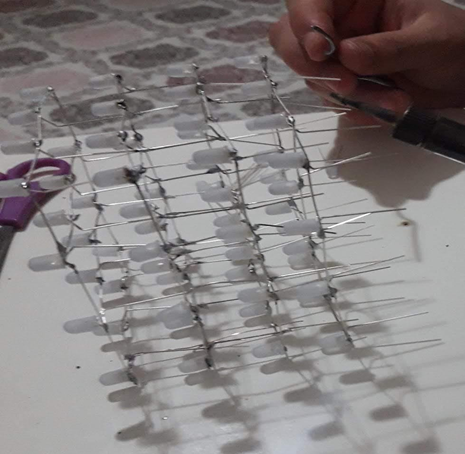
Anot bacakları yukarı doğru bir şekil aldırılmış olundu. Katmandan 1 tane anot bacak dışarı doğru çıkmış oldu. Böylece 1. katman tamamlanmış olundu (Şekil 4.18). Bu şekilde lehimleme işine (Şekil 4.19) devam edildi. Toplam 4 katman yapıldı (Şekil 4.20).

****

Şekil 4.18 : Ledlerin lehimlenmesi ile oluşan 1.katman.

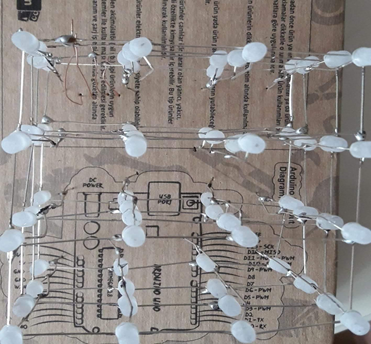
****

Şekil 4.19 : Ledlerin lehimlenmesi ile oluşan 1.katman.

****

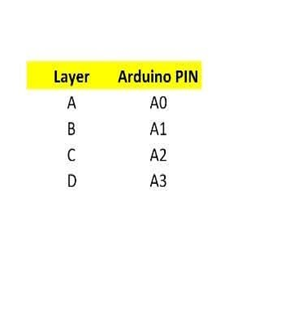
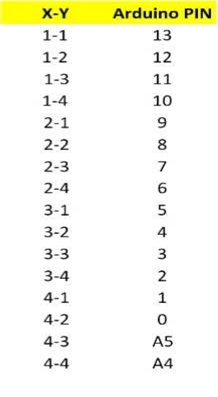
Şekil 4.20 : Led’lerin katmanlarının lehimlenmiş hali.

Daha sonra her katmadan çıkan artı uçlara jumper kablo yardımı ile aşağı yönlendirme yapıldı (Şekil 4.21).

****

Şekil 4.21 : Katmanlardan çıkan artı bacakları aşağıya yönlendirilmesi.

Aşağıya yönlendirilen artı bacaklara 100 ohm’luk dirençler bağlandı. Aşağıda belirtilen bağlanma çizelgesine göre Arduino ‘ya takıldı (Şekil 4.22). 16 tane bulunan eksi bacakda jumper kablolar yardımı ile arduinoya bağlandı (Şekil 4.23). Projenin mekanik kısmı böylece tamamlandı.



Şekil 4.22 : Arduino’ya jumper kabloların bağlanma çizelgeleri.

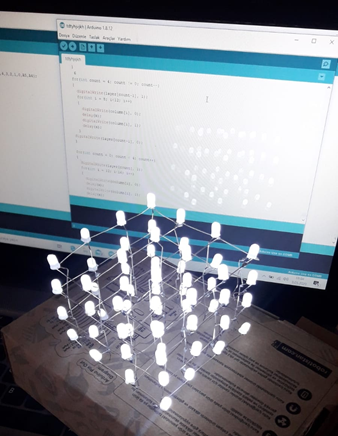
****

Şekil 4.23 : Eksi bacakların jumper kablolarla birleştirilip arduino’ya bağlanması.

Arduino platformu üzerinden kodlar yazıldı. USB kablo ile Arduino bilgisayara bağlandı. Kodlar programa yüklenip çalıştırıldı (Şekil 4.24). Projemiz tamamlanmış ve amacına ulaşılmış oldu (Şekil 4.25).

****

Şekil 4.24 : Led küpün çalışması sonucu ortaya çıkan görüntüler.

****

Şekil 4.25 : Led küpün çalışması sonucu ortaya çıkan görüntüler.

# 

# SONUÇ

Led küp projesinin mekanik kısmının tamamlanmış hali sonunda projeye uygulanılan kodlar sayesinde çeşitli görüntüler elde edildi. Bunun haricinde projenin ana malzemesi olan ledin 1907 yılında icat edildi. İcat edilmesinin ardından lede farklı özellikler kazandırılmak için gelişmeler meydana geldiği bilgisi elde edilmiştir. Ledin farklı çeşitleri olduğu bunun sayesinde led ile birçok projenin yapımı ve hayatın çeşitli alanlarında renkli, güzel görüntüler elde etmek için kullanılmıştır. Dünya genelinde de led kullanımı ülkeler arasında bir rekabet ortamı oluşturmuştur. Günümüzde, projede kullanılan Arduino Uno haricinde farklı elektronik geliştirme kartları bulunmaktadır. Bunlara örnek olarak Raspberry pi geliştirme kartı, The beaglebone black geliştirme kartı…vb. Arduino Uno kartının çalışması için gerekli yazılım platformu Arduinodur. Arduino’da kullanımının kolay, açık kaynak kodlu yazılım ve donanıma sahip bir mikrodenetleyicidir. Böyle olmasından dolayı Led küp projesinin çalıştırılması için bir avantaj sağlanmıştır. Projenin yapım aşamasında kullanılan elektronik malzemelerin nasıl bir işlevi olduğu araştırılmıştır. Malzemeler içinden jumper kablo ledlerin artı veya eksi bacaklarının Arduino Uno’ya bağlamak için kullanılmıştır. Projenin yapım aşamasına geçilmeden önce projenin CAD çizimleri yapıldı. Bu çizimlerden projeyi yaparken faydalanıldı. Projenin yapım aşamasında ledler belirli bir düzene göre hizalanlandı. Hizalanan ledler lehimleme aşamasından geçirildi. Bu esnada katmanlardan hangi bacakların dışarıya çıkacağı ve bu çıkan bacakların bazılarının dirençlerle beraber bazılarının ise direkt jumper kablolar yardımıyla Arduino’ya bağlanılması ayarlandı. Ayarlanma şekli bağlanma çizelgelerine uyarlanıp yapıldı. Böylece projenin kalıp olarak şekli tamamlandı. Projenin çalıştırılması için kodlar Arduino platformu yazıldı. Bu esnada Arduino’ nun içeriğinde Java, C, C++ dillerinin özellikleri kullanıldığına ulaşıldı Kodlama sonrasında Led küp dekoratif görüntüler oluşturdu.

Led küp projesi montajlama ve yazılım konusunda bilgi ve beceriler kazanılmasına ve sonunda değişik görüntüler elde edilmesine yardımcı oldu.

KAYNAKLAR

1. [https://www.elektrikport.com/teknik-kutuphane/aydinlatmanin-tarihcesi-ampulden-lede/4382#ad-image-0](https://www.elektrikport.com/teknik-kutuphane/aydinlatmanin-tarihcesi-ampulden-lede/4382" \l "ad-image-0)
2. https://www.bilgiustam.com/led-lambalarin-icadi-ve-kullanim-alanlari/
3. <https://www.instructables.com/id/4x4x4-LED-Cube-Arduino-Uno/>
4. <https://maker.robotistan.com/programlama-dilleri/>
5. <https://hayaletveyap.com/en-iyi-12-gelistirme-karti/>
6. <https://github.com/Robotistan/YouTube-DIY-Projeler/tree/master/LED_Kup_4x4x4_Arduino>
7. <https://www.robotistan.com/havya-lehim-temizleme-sungeri>
8. [https://www.nedirnedemek.com/zımpara-kağıdı-ne-demek](https://www.nedirnedemek.com/z%C4%B1mpara-ka%C4%9F%C4%B1d%C4%B1-ne-demek)
9. <http://elektronikblogum.blogspot.com/2011/06/lehim-pastas-nedir.html>
10. <http://elektrikelektronikegitimi.blogspot.com/2018/04/elektronikte-kullanlan-lehim-teli-nedir.html>
11. <https://teknolojiprojeleri.com/elektronik/havya>
12. <https://diyot.net/makaron/#:~:text=Makaron%2C%20%C4%B1s%C4%B1%20etkisi%20ile%20belirli,benzeri%20operasyonlarda%20g%C3%BCvenle%20kullan%C4%B1labilen%20malzemelerdir.>
13. <http://devreokulu.com/BaslangicBaglanti.html>
14. <https://maker.robotistan.com/direnc/>
15. <https://www.makerteknoloji.com/bizden-yazilar/arduino-yu-programlamak-icin-hangi-yazilim-dili-gerekiyor>
16. <https://maker.robotistan.com/arduino-uno/>
17. <https://teknolojiprojeleri.com/arduino/arduino-nedir-ne-ise-yarar>
18. <https://maker.robotistan.com/led/#LED-Nasil-Calisir>
19. <https://docplayer.biz.tr/5084639-Led-li-isik-kaynaklari.html>
20. <https://www.elektrikport.com/teknik-kutuphane/led-aydnlatmann-hayatmzdaki-yeri-ve-onemi/8569#ad-image-0>
21. http://shellscript.info/yazi-kategorisi/elektronik/ledler-hakkinda-her-sey/9/181

EKLER

**EK A : KOD DİZİNİ** [6]

//initializing and declaring led rows

int column [16] ={13,12,11,10,9,8,7,6,5,4,3,2,1,0,A5,A4};

//initializing and declaring led layers

int layer [4] ={A3,A2,A1,A0};

int time =250;

void setup()

{

//setting rows to output

for (int i = 0; i<16; i++)

{

pinMode(column[i], OUTPUT) ;

}

//setting layers to output

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

{

pinMode(layer[i], OUTPUT);

}

//seeding random for random pattern

randomSeed (analogRead (10));

}

//xxxxxxxxxxxxxxxxxxxxFUNCTION LOOPxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx

//xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx

void loop ()

{

turnEverythingOff () ;//turn all off

flickerOn () ;

turnEverythingOn () ;//turn all on

delay(time) ;

turnOnAndOffAllByLayerUpAndDownNotTimed ();

layerstompUpAndDown ();

turnOnAndOffAllByColumnSideways ();

delay (time);

aroundEdgeDown ();

turnEverythingOff ();

randomflicker ();

randomRain ();

diagonalRectangle ();

goThroughAllLedsOneAtATime ();

propeller ();

spiralInAndOut ();

lickerOff () ;

turnEverythingOff();

delay(2000);

}

//xxxxxxxxxxxxxxxxxxxxFUNCTIONSxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx

//xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx

///////////////////////////////////////////////////////////turn all off

void turnEverythingOff()

{

for(int i = 0; i<16; i++)

{

digitalWrite(column[i], 1);

}

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

{

digitalWrite(layer[i], 0);

}

}

////////////////////////////////////////////////////////////turn all on

void turnEverythingOn()

{

for(int i = 0; i<16; i++)

{

digitalWrite(column[i], 0);

}

//turning on layers

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

{

digitalWrite(layer[i], 1);

}

}

///////////////////////////////////////////////////////turn columns off

void turnColumnsOff()

{

for(int i = 0; i<16; i++)

{

digitalWrite(column[i], 1);

}

}

/////////////////////////////////////////////////////////////flicker on

void flickerOn()

{

int i = 150;

while(i != 0)

{

turnEverythingOn();

delay(i);

turnEverythingOff();

delay(i);

i-= 5;

}

}

//////////////turn everything on and off by layer up and down NOT TIMED

void turnOnAndOffAllByLayerUpAndDownNotTimed()

{

int x = 75;

for(int i = 5; i != 0; i--)

{

turnEverythingOn();

for(int i = 4; i!=0; i--)

{

digitalWrite(layer[i-1], 0);

delay(x);

}

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

{

digitalWrite(layer[i], 1);

delay(x);

}

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

{

digitalWrite(layer[i], 0);

delay(x);

}

for(int i = 4; i!=0; i--)

{

digitalWrite(layer[i-1], 1);

delay(x);

}

}

}

//////////////////////////turn everything on and off by column sideways

void turnOnAndOffAllByColumnSideways()

{

int x = 75;

turnEverythingOff();

//turn on layers

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

{

digitalWrite(layer[i], 1);

}

for(int y = 0; y<3; y++)

{

//turn on 0-3

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

{

digitalWrite(column[i], 0);

delay(x);

}

//turn on 4-7

for(int i = 4; i<8; i++)

{

digitalWrite(column[i], 0);

delay(x);

}

//turn on 8-11

for(int i = 8; i<12; i++)

{

digitalWrite(column[i], 0);

delay(x);

}

//turn on 12-15

for(int i = 12; i<16; i++)

{

digitalWrite(column[i], 0);

delay(x);

}

//turn off 0-3

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

{

digitalWrite(column[i], 1);

delay(x);

}

//turn off 4-7

for(int i = 4; i<8; i++)

{

digitalWrite(column[i], 1);

delay(x);

}

//turn off 8-11

for(int i = 8; i<12; i++)

{

digitalWrite(column[i], 1);

delay(x);

}

//turn off 12-15

for(int i = 12; i<16; i++)

{

digitalWrite(column[i], 1);

delay(x);

}

//turn on 12-15

for(int i = 12; i<16; i++)

{

digitalWrite(column[i], 0);

delay(x);

}

//turn on 8-11

for(int i = 8; i<12; i++)

{

digitalWrite(column[i], 0);

delay(x);

}

//turn on 4-7

for(int i = 4; i<8; i++)

{

digitalWrite(column[i], 0);

delay(x);

}

//turn on 0-3

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

{

digitalWrite(column[i], 0);

delay(x);

}

//turn off 12-15

for(int i = 12; i<16; i++)

{

digitalWrite(column[i], 1);

delay(x);

}

//turn off 8-11

for(int i = 8; i<12; i++)

{

digitalWrite(column[i], 1);

delay(x);

}

//turn off 4-7

for(int i = 4; i<8; i++)

{

digitalWrite(column[i], 1);

delay(x);

}

//turn off 0-3

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

{

digitalWrite(column[i], 1);

delay(x);

}

}

}

/////////////////////////////////////////up and down single layer stomp

void layerstompUpAndDown()

{

int x = 75;

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

{

digitalWrite(layer[i], 0);

}

for(int y = 0; y<5; y++)

{

for(int count = 0; count<1; count++)

{

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

{

digitalWrite(layer[i], 1);

delay(x);

digitalWrite(layer[i], 0);

}

for(int i = 4; i !=0; i--)

{

digitalWrite(layer[i-1], 1);

delay(x);

digitalWrite(layer[i-1], 0);

}

}

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

{

digitalWrite(layer[i], 1);

delay(x);

}

for(int i = 4; i!=0; i--)

{

digitalWrite(layer[i-1], 0);

delay(x);

}

}

}

////////////////////////////////////////////////////////////flicker off

void flickerOff()

{

turnEverythingOn();

for(int i = 0; i!= 150; i+=5)

{

turnEverythingOff();

delay(i+50);

turnEverythingOn();

delay(i);

}

}

///////////////////////////////////////////around edge of the cube down

void aroundEdgeDown()

{

for(int x = 200; x != 0; x -=50)

{

turnEverythingOff();

for(int i = 4; i != 0; i--)

{

digitalWrite(layer[i-1], 1);

digitalWrite(column[5], 0);

digitalWrite(column[6], 0);

digitalWrite(column[9], 0);

digitalWrite(column[10], 0);

digitalWrite(column[0], 0);

delay(x);

digitalWrite(column[0], 1);

digitalWrite(column[4], 0);

delay(x);

digitalWrite(column[4], 1);

digitalWrite(column[8], 0);

delay(x);

digitalWrite(column[8], 1);

digitalWrite(column[12], 0);

delay(x);

digitalWrite(column[12], 1);

digitalWrite(column[13], 0);

delay(x);

digitalWrite(column[13], 1);

digitalWrite(column[15], 0);

delay(x);

digitalWrite(column[15], 1);

digitalWrite(column[14], 0);

delay(x);

digitalWrite(column[14], 1);

digitalWrite(column[11], 0);

delay(x);

digitalWrite(column[11], 1);

digitalWrite(column[7], 0);

delay(x);

digitalWrite(column[7], 1);

digitalWrite(column[3], 0);

delay(x);

digitalWrite(column[3], 1);

digitalWrite(column[2], 0);

delay(x);

digitalWrite(column[2], 1);

digitalWrite(column[1], 0);

delay(x);

digitalWrite(column[1], 1);

}

}

}

/////////////////////////////////////////////////////////random flicker

void randomflicker()

{

turnEverythingOff();

int x = 10;

for(int i = 0; i !=750; i+=2)

{

int randomLayer = random(0,4);

int randomColumn = random(0,16);

digitalWrite(layer[randomLayer], 1);

digitalWrite(column[randomColumn], 0);

delay(x);

digitalWrite(layer[randomLayer], 0);

digitalWrite(column[randomColumn], 1);

delay(x);

}

}

////////////////////////////////////////////////////////////random rain

void randomRain()

{

turnEverythingOff();

int x = 100;

for(int i = 0; i!=60; i+=2)

{

int randomColumn = random(0,16);

digitalWrite(column[randomColumn], 0);

digitalWrite(layer[0], 1);

delay(x+50);

digitalWrite(layer[0], 0);

digitalWrite(layer[1], 1);

delay(x);

digitalWrite(layer[1], 0);

digitalWrite(layer[2], 1);

delay(x);

digitalWrite(layer[2], 0);

digitalWrite(layer[3], 1);

delay(x+50);

digitalWrite(layer[3], 0);

digitalWrite(column[randomColumn], 1);

}

}

/////////////////////////////////////////////////////diagonal rectangle

void diagonalRectangle()

{

int x = 350;

turnEverythingOff();

for(int count = 0; count<5; count++)

{

//top left

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

{

digitalWrite(column[i], 0);

}

digitalWrite(layer[3], 1);

digitalWrite(layer[2], 1);

delay(x);

turnEverythingOff();

//middle middle

for(int i = 4; i<12; i++)

{

digitalWrite(column[i], 0);

}

digitalWrite(layer[1], 1);

digitalWrite(layer[2], 1);

delay(x);

turnEverythingOff();

//bottom right

for(int i = 8; i<16; i++)

{

digitalWrite(column[i], 0);

}

digitalWrite(layer[0], 1);

digitalWrite(layer[1], 1);

delay(x);

turnEverythingOff();

//bottom middle

for(int i = 4; i<12; i++)

{

digitalWrite(column[i], 0);

}

digitalWrite(layer[0], 1);

digitalWrite(layer[1], 1);

delay(x);

turnEverythingOff();

//bottom left

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

{

digitalWrite(column[i], 0);

}

digitalWrite(layer[0], 1);

digitalWrite(layer[1], 1);

delay(x);

turnEverythingOff();

//middle middle

for(int i = 4; i<12; i++)

{

digitalWrite(column[i], 0);

}

digitalWrite(layer[1], 1);

digitalWrite(layer[2], 1);

delay(x);

turnEverythingOff();

//top right

for(int i = 8; i<16; i++)

{

digitalWrite(column[i], 0);

}

digitalWrite(layer[2], 1);

digitalWrite(layer[3], 1);

delay(x);

turnEverythingOff();

//top middle

for(int i = 4; i<12; i++)

{

digitalWrite(column[i], 0);

}

digitalWrite(layer[2], 1);

digitalWrite(layer[3], 1);

delay(x);

turnEverythingOff();

}

//top left

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

{

digitalWrite(column[i], 0);

}

digitalWrite(layer[3], 1);

digitalWrite(layer[2], 1);

delay(x);

turnEverythingOff();

}

//////////////////////////////////////////////////////////////propeller

void propeller()

{

turnEverythingOff();

int x = 90;

for(int y = 4; y>0; y--)

{

for(int i = 0; i<6; i++)

{

//turn on layer

digitalWrite(layer[y-1], 1);

//a1

turnColumnsOff();

digitalWrite(column[0], 0);

digitalWrite(column[5], 0);

digitalWrite(column[10], 0);

digitalWrite(column[15], 0);

delay(x);

//b1

turnColumnsOff();

digitalWrite(column[4], 0);

digitalWrite(column[5], 0);

digitalWrite(column[10], 0);

digitalWrite(column[11], 0);

delay(x);

//c1

turnColumnsOff();

digitalWrite(column[6], 0);

digitalWrite(column[7], 0);

digitalWrite(column[8], 0);

digitalWrite(column[9], 0);

delay(x);

//d1

turnColumnsOff();

digitalWrite(column[3], 0);

digitalWrite(column[6], 0);

digitalWrite(column[9], 0);

digitalWrite(column[12], 0);

delay(x);

//d2

turnColumnsOff();

digitalWrite(column[2], 0);

digitalWrite(column[6], 0);

digitalWrite(column[9], 0);

digitalWrite(column[13], 0);

delay(x);

//d3

turnColumnsOff();

digitalWrite(column[1], 0);

digitalWrite(column[5], 0);

digitalWrite(column[10], 0);

digitalWrite(column[14], 0);

delay(x);

}

}

//d4

turnColumnsOff();

digitalWrite(column[0], 0);

digitalWrite(column[5], 0);

digitalWrite(column[10], 0);

digitalWrite(column[15], 0);

delay(x);

}

//////////////////////////////////////////////////////spiral in and out

void spiralInAndOut()

{

turnEverythingOn();

int x = 60;

for(int i = 0; i<6; i++)

{

//spiral in clockwise

digitalWrite(column[0], 1);

delay(x);

digitalWrite(column[1], 1);

delay(x);

digitalWrite(column[2], 1);

delay(x);

digitalWrite(column[3], 1);

delay(x);

digitalWrite(column[7], 1);

delay(x);

digitalWrite(column[11], 1);

delay(x);

digitalWrite(column[15], 1);

delay(x);

digitalWrite(column[14], 1);

delay(x);

digitalWrite(column[13], 1);

delay(x);

digitalWrite(column[12], 1);

delay(x);

digitalWrite(column[8], 1);

delay(x);

digitalWrite(column[4], 1);

delay(x);

digitalWrite(column[5], 1);

delay(x);

digitalWrite(column[6], 1);

delay(x);

digitalWrite(column[10], 1);

delay(x);

digitalWrite(column[9], 1);

delay(x);

///////////////////////////////////////spiral out counter clockwise

digitalWrite(column[9], 0);

delay(x);

digitalWrite(column[10], 0);

delay(x);

digitalWrite(column[6], 0);

delay(x);

digitalWrite(column[5], 0);

delay(x);

digitalWrite(column[4], 0);

delay(x);

digitalWrite(column[8], 0);

delay(x);

digitalWrite(column[12], 0);

delay(x);

digitalWrite(column[13], 0);

delay(x);

digitalWrite(column[14], 0);

delay(x);

digitalWrite(column[15], 0);

delay(x);

digitalWrite(column[11], 0);

delay(x);

digitalWrite(column[7], 0);

delay(x);

digitalWrite(column[3], 0);

delay(x);

digitalWrite(column[2], 0);

delay(x);

digitalWrite(column[1], 0);

delay(x);

digitalWrite(column[0], 0);

delay(x);

///////////////////////////////////////spiral in counter clock wise

digitalWrite(column[0], 1);

delay(x);

digitalWrite(column[4], 1);

delay(x);

digitalWrite(column[8], 1);

delay(x);

digitalWrite(column[12], 1);

delay(x);

digitalWrite(column[13], 1);

delay(x);

digitalWrite(column[14], 1);

delay(x);

digitalWrite(column[15], 1);

delay(x);

digitalWrite(column[11], 1);

delay(x);

digitalWrite(column[7], 1);

delay(x);

digitalWrite(column[3], 1);

delay(x);

digitalWrite(column[2], 1);

delay(x);

digitalWrite(column[1], 1);

delay(x);

digitalWrite(column[5], 1);

delay(x);

digitalWrite(column[9], 1);

delay(x);

digitalWrite(column[10], 1);

delay(x);

digitalWrite(column[6], 1);

delay(x);

//////////////////////////////////////////////spiral out clock wise

digitalWrite(column[6], 0);

delay(x);

digitalWrite(column[10], 0);

delay(x);

digitalWrite(column[9], 0);

delay(x);

digitalWrite(column[5], 0);

delay(x);

digitalWrite(column[1], 0);

delay(x);

digitalWrite(column[2], 0);

delay(x);

digitalWrite(column[3], 0);

delay(x);

digitalWrite(column[7], 0);

delay(x);

digitalWrite(column[11], 0);

delay(x);

digitalWrite(column[15], 0);

delay(x);

digitalWrite(column[14], 0);

delay(x);

digitalWrite(column[13], 0);

delay(x);

digitalWrite(column[12], 0);

delay(x);

digitalWrite(column[8], 0);

delay(x);

digitalWrite(column[4], 0);

delay(x);

digitalWrite(column[0], 0);

delay(x);

}

}

//////////////////////////////////////go through all leds one at a time

void goThroughAllLedsOneAtATime()

{

int x = 15;

turnEverythingOff();

for(int y = 0; y<5; y++)

{

//0-3

for(int count = 4; count != 0; count--)

{

digitalWrite(layer[count-1], 1);

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

{

digitalWrite(column[i], 0);

delay(x);

digitalWrite(column[i], 1);

delay(x);

}

digitalWrite(layer[count-1], 0);

}

//4-7

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

{

digitalWrite(layer[count], 1);

for(int i = 4; i<8; i++)

{

digitalWrite(column[i], 0);

delay(x);

digitalWrite(column[i], 1);

delay(x);

}

digitalWrite(layer[count], 0);

}

//8-11

for(int count = 4; count != 0; count--)

{

digitalWrite(layer[count-1], 1);

for(int i = 8; i<12; i++)

{

digitalWrite(column[i], 0);

delay(x);

digitalWrite(column[i], 1);

delay(x);

}

digitalWrite(layer[count-1], 0);

}

//12-15

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

{

digitalWrite(layer[count], 1);

for(int i = 12; i<16; i++)

{

digitalWrite(column[i], 0);

delay(x);

digitalWrite(column[i], 1);

delay(x);

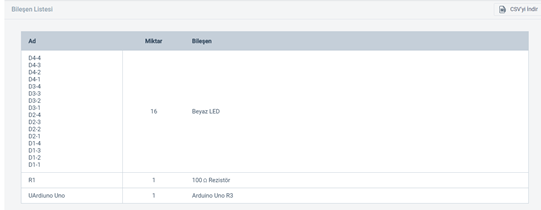
}

digitalWrite(layer[count], 0);

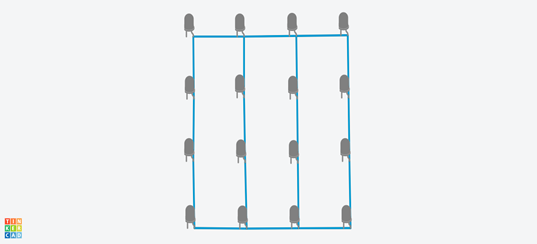
}}}

**EK B : AYRINTILI ÇİZİM**

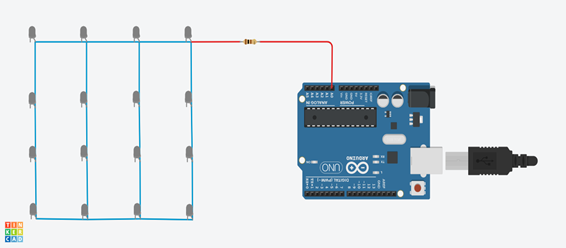
1. Çizim yapılırken kullanılan bileşenler

****

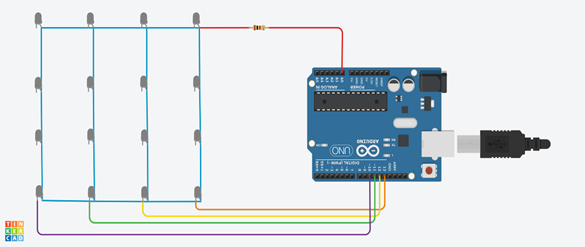
1. Ledlerin artı bacaklarının birleşimi

****

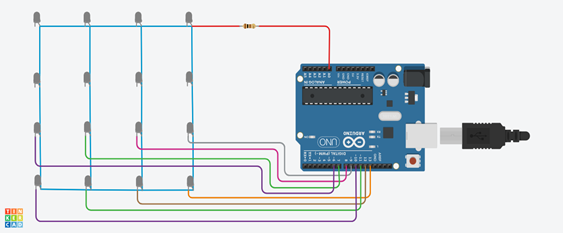
1. Ledlerin her katmandan çıkan artı bacaklarına 100 ohm’luk dirençlerle Arduino’ya bağlanması

****

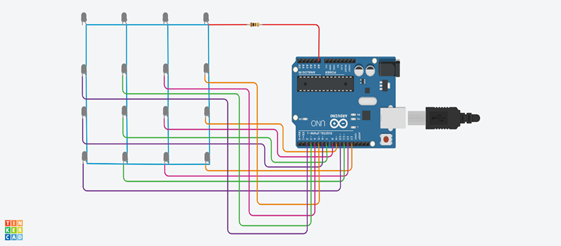
1. Ledlerin yerleşiminde eksi bacaklarının Arduino’ya bağlanması (1.bağlantı)

****

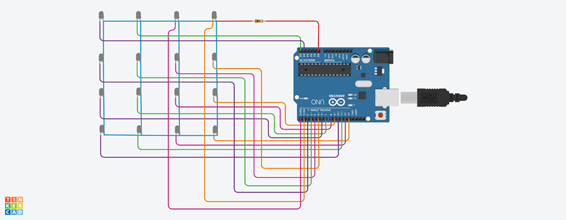
1. Ledlerin yerleşiminde eksi bacaklarının Arduino’ya bağlanması (2. bağlantı)

****

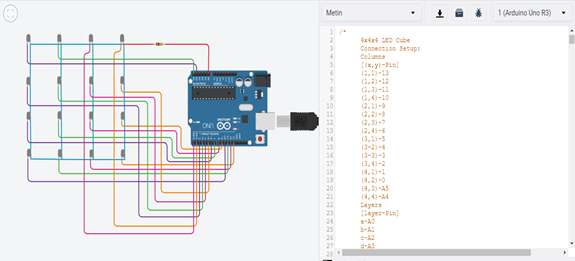
1. Ledlerin yerleşiminde eksi bacaklarının Arduino’ya bağlanması (3. bağlantı)

****

1. Ledlerin yerleşiminde eksi bacaklarının Arduino’ya bağlanması (4.bağlantı)

****

1. Arduino kodları yazılıp Arduino Uno’ nun takılması

****

# PROJE EKİBİ

Çizelge 6.1 : Kişisel bilgiler.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **AD** | **SOYAD** | **MAİL** | **OKUL NO** | **GÖREVİ** |
| Beyzanur | GÜRSES | beyzanurgurses1036  @gmail.com | 18360859051 | Raportör |
| Elanur | BAŞARAN | elanurbasaran7  @gmail.com | 19360859020 | Lider,Yapan |
| Nazlı Tuğba | GÜRSOY | nzligursoy61  @gmail.com | 19360859037 | D.yazman,  İntihal |