

**T.C.**

**MİLLİ EĞİTİM BAKANLIĞI**

**SİVAS MERKEZ**

**Sivas Fen Lisesi**

**Bilgisyar Bilimi Dersi**

**Rapor**

|  |  |
| --- | --- |
| **Rapor No** | Proje-2 |
| **Rapor Tarih** | 12.05.2018 |
| **Proje Adı** | LED Cube |

Bilgisayar Bilimi Öğretmeni

Ersin TÜTÜNCÜ

2017-2018



**T.C.**

**MİLLİ EĞİTİM BAKANLIĞI**

**SİVAS MERKEZ**

**Sivas Fen Lisesi**

**Proje Grup**

|  |  |  |
| --- | --- | --- |
| **Proje Görev** | **Numara** | **Ad-Soyad** |
| Proje Yönetimi | 107 | Beyza Dişli |
| Doküman Yönetimi | 440 | Zülal Toy |
| Lojistik Yönetim | 409 | Elif Sultan Akgül |
| Yazılım Geliştirme | 252 | Sena Belinay Yüksek |
| Web ve GitHub Yönetimi | 377 | Ömer Karakoç |
| Sunum Yönetimi | 307 | Aydın Can Çelik |

**ÖZET**

Bu proje kapsamında yapılan led cubeün hazırlanan koda göre yanması beklenmektedir. Kod üzerinde yapılan değişiklikler ile ledlerin farklı şekilde yandığı gözlemlenebilmektedir. Proje gerçekleştirilirken devre tasarımı, devre elemanlarının lehimlenmesi, test ve deneme aşamaları, kodlama, sunum gibi aşamalar gerçekleştirilecektir.

**Anahtar Kelimeler**

Led cube, kod, koda göre yanma

**ABSTRACT**

The led cube built within the scope of this project is expected to burn according to the prepared code.With the changes made on the code, it can be observed that the LEDs are lit up differently. During the project, stages such as circuit design, soldering of circuit elements, testing and testing stages, coding, presentation will be realized.

**Key Words**

Led cube, code, combustion by code

**Proje Görev Dağılımı Listesi ve Görev Dağılımı Açıklaması**

* **Görev Dağılımı ve Sorumlusu**

|  |  |  |
| --- | --- | --- |
| Proje Yönetimi | Görev dağılımı ve takibinden sorumlu kişi, aynı zamanda proje grubunun çalışma takvimini ve düzenini ayarlamaktadır.Grupta bulunan kişilerle iletişim halinde olup projenin yönetimini sağlar. | Ad-Soyad  Beyza Dişli |
| Döküman Yönetimi | Projenin tüm tasarım ve çizimlerinden,proje raporunun sunulmasından,dökümanların uygun forma getirilmesinden kodlamaya ait diagram ve modellerin hazırlanması ve web sitesi tasarımından sorumlu olan kişidir. | Zülal Toy |
| Lojistik Yönetimi | Projede kullanılacak tüm elemanların, malzemelerin belirlenmesi ve temin edilmesi,en uygun tasarımın yapılması için geliştirmelerin yapılmasıyla ve projenin donanımsal kısmının tanıtılması ile ilgilenen kişidir. | Elif Sultan Akgül |
| Yazılım Geliştirme Yönetimi | Yazılım için araştırmaların yapılması, yazılım aşamalarının proje grubuna dağıtılması,Yazılım ile ilgili raporların hazırlanarak ilgili bölüme(döküman yönetimine) aktarılması yazılım ve süreç testlerinin gerçekleştirilmesi ile ilgilenen kişidir. | Sena Belinay Yüksel |
| WEB ve GitHub Yönetimi | Proje tanıtımı için WEB sayfasının hazırlanması, projenin GitHub yönetiminin yapılması,döküman yöneticisinden almış olduğu raporlar ile WEB sitesine ve GitHub'a işlemekle sorumlu olan kişidir. | Ömer Karakoç |
| Sunum Yönetimi | Proje teslim zamanında sunumun, yapılan tüm işlemlerin uygun bir biçimde anlatılmasından,rapor ve evrakların eksiksiz bir şekilde sunulmasından ve önerilere,  sorulara uygun çözümler üretmekten sorumludur. | Aydın Can Çelik |

* **Görev süresince sürdürülen eylemler**

|  |  |
| --- | --- |
| 1.Hafta | Proje Hakkında Bilgi Edinme |
| 2.Hafta | Malzeme Seçimi |
| 3.Hafta | Mekanik ve Elektronik Tasarım |
| 4.Hafta | Yazılım |
| 5.Hafta | Grup elemanlarına ait iş yükünün tamamlanması (rapor,web,github) |

* **Yoklama Çizelgeleri**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| AD-SOYAD  Elif Sultan Akgül | **18.04.2018**    **+** | **25.04.2018**  **+** | **02.05.2018**  **+** | **09.05.2018**  **+** | **16.05.2018**  **+** |
| Zülal Toy | **+** | **+** | **+** | **+** | **+** |
| Beyza Dişli | **+** | **+** | **+** | **+** | **+** |
| Sena Belinay Yüksel | **+** | **+** | **+** | **+** | **+** |
| Ömer Karakoç | **+** | **+** | **+** | **+** | **+** |
| Aydın Can Çelik | **+** | **+** | **+** | **+** | **+** |

**Haftalık İş Katkı Cetvelleri**

|  |  |
| --- | --- |
|  | **Yapılan iş** |
| **18.04.2018** | Proje araştırması |
| **25.04.2018** | Malzeme Teminatı |
| **02.05.2018** | Elektronik ve Mekanik Tasarım |
| **09.05.2018** | Yazılım |
| **16.05.2018** | Deneme ve Test Aşamaları |

**GİRİŞ**

**1.Projenin Açıklaması**

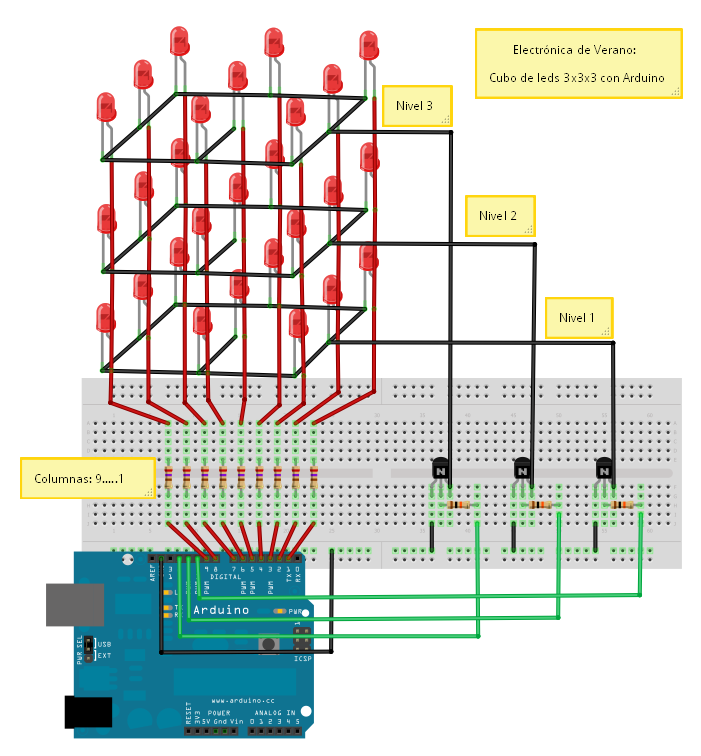
Yapılan projede amaç; hazırlanan koda göre yanabilecek bir led cube tasarımı gerçekleştirebilmektir.Led cubeün çalışma prensibi birbirine kablolar ile bağlanan ve arduıno nanodan destek alan ledin istek üzerine çalışmasıdır.

**2.Donanım Yapısı:**

**a.Gömülü Sistemler Mimarisi ve Devre Tasarımı**

|  |  |
| --- | --- |
| arduino nano ile ilgili gÃ¶rsel sonucu | **Arduino Nano**  Bu ürünümüz, Klon olup bazı farklılıklar ile popüler Arduino Nano ürününün pin ve özellikleri ile uyumlu bir üründür. Bu klon ürünün orjinali [Arduino NANO](https://www.robotistan.com/arduino-nano-328-usb-kablolu) modelidir. Bu ürünün farklılıkları ise USB dönüştürücü entegresi ile birkaç ufak değişiklikte bulunulmuştur. |
| Ä°lgili resim | **¼ W 10K Direnç**  0.25W değerine kadar enerjiye dayanıklı 1K ohm dirençtir. |
| 3 mm Åeffaf Beyaz Led ile ilgili gÃ¶rsel sonucu | **3 mm Şeffaf Beyaz Led** 3mm Şeffaf Beyaz Led Paketi içerisinde 10 adet led bulunur.Elektronik devrelerin neredeyse tamamında kullanılan bir komponenttir. Şeffaf olduğu için devrelerinize şık bir görüntü de katacaktır. 1.5-3V arası gerilimde çalışır. 5V ve üstü voltaj değerleri için gerekli dirençlerle kullanılması gerekir. |
|  | **Pertinaks**  Prototip devre elemanlarını pertinaks üzerine lehimleyip farklı uygulamalar için kullanabiliriz. |
|  | **Jumper Kablo (E-E/D-E)**  Devre elemanlarının bağlantılarını gerçekleşirmek için kullanabiliriz. |

Devre Tasarımı:



**1.Arduino NANO**

**Özellikler**

|  |  |
| --- | --- |
| Mikrodenetleyici | Atmel Atmega328 |
| Çalışma Gerilimi | 5V |
| Giriş Gerilimi(önerilen) | 6-20V |
| Dijital I/O Pinleri | * 14 (6 tanesi PWM çıkışı) |
| Analog Giriş Pinleri | 8 |
| Her I/O için Akım | 40mA |
| EEPROM | * 1 KB (Atmega328P) |
| SRAM | * 2KB(Atmega 328P) |
| Flash Hafıza | * 32 KB |

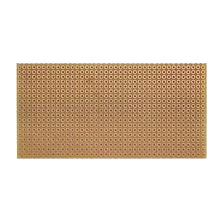
**2. ¼ W 10K Direnç**



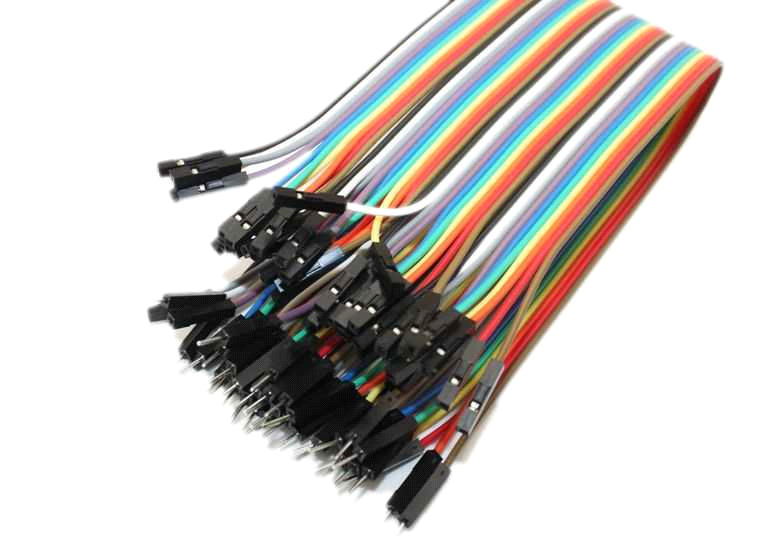
**3. 3mm Şeffaf Beyaz Led**

****

**4.Pertinaks**



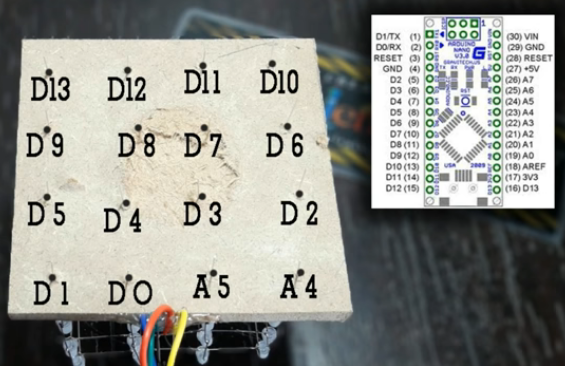
**5. Jumper Kablo**



**3.Yazılım Yapısı**

**a.Algoritmik olarak:**

Led cubede bağlanan kablolara göre yanacaktır. Birkaç örnek;



Şekilde görüldüğü gibi led cıkışlarından kablo uzatılarak arduıno nanoya bağlanılacaktır.

**b:Kod yapısı:**

**int layer[4]={A3,A2,A1,A0}; //initializing and declaring led layers**

**int column[16]={13,12,11,10,9,8,7,6,5,4,3,2,1,0,A5,A4}; //initializing and declaring led rows**

**int time = 250;**

**void setup()**

**{**

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

**{**

**pinMode(column[i], OUTPUT); //setting rows to ouput**

**}**

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

**{**

**pinMode(layer[i], OUTPUT); //setting layers to output**

**}**

**randomSeed(analogRead(10)); //seeding random for random pattern**

**}**

**void loop()**

**{**

**turnEverythingOff();**

**flickerOn();**

**turnEverythingOn();**

**delay(time);**

**turnOnAndOffAllByLayerUpAndDownNotTimed();**

**layerstompUpAndDown();**

**spiralInAndOut();**

**turnOnAndOffAllByColumnSideways();**

**delay(time);**

**aroundEdgeDown();**

**turnEverythingOff();**

**randomflicker();**

**randomRain();**

**diagonalRectangle();**

**goThroughAllLedsOneAtATime();**

**propeller();**

**spiralInAndOut();**

**flickerOff();**

**turnEverythingOff();**

**delay(2000);**

**}**

**//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);**

**}**

**}**

**}**

**SONUÇ**

**1.Bilgi Düzeyine Katkıları:**

Herşeyden önce proje yönetiminin nasıl olması gerektiğini ve projenin sunumunun nasıl yapılması gerektiğini öğrendik. 4 Haftalık süreçte birçok devre elemanının kullanımıyla ilgili tecrübeler edindik.Devrenin kurulumu,gerekli kodlamanın yapılması, devrelerin şematik olarak gösterilmesi, fritzing, proteus kullanımı ve daha birçok konuda bilgi edindik. Bir robotun elektonik ve mekanik tasarımının nasıl olması gerektiğini öğrendik ve yaptığımız yanlışlar ile tasarım aşamasında yapılmaması gerekenleri fark ettik.

Bu proje sayesinde kodlamyaı daha iyi kavrayarak kabloların nasıl yerleştirileceğini, arduıno ile nasıl bağlantı kurulacağını öğrendik.Test aşamasında gördüğümüz hatalarımızı düzelttik.

**2.Teknolojik Katkıları:**

Bu projeyi yaparken kodlamanın önemini ve hayatımızın her alanında kullanabileceğimizi öğrendik. Artık basitçe led döşeyip geçmeyi değil, bu ledlerle farklı şekillerde hatta isteğimiz şekle göre yanmasını sağlayabildik.

**3.Ekip Çalışması Katkıları:**

Ekip çalışmasının bize kattıkları:

* Her grup elemanının teknik bilgisini geliştirdi. Konuya bütünlük açısından bakmamıza yardımcı oldu
* Proje çalışanların sorun çözme alışkanlığını geliştirdi.
* Çalışanların birbiriyle iletişim alışkanlıklarını geliştirdi.
* Ekip, bir kişinin tek başına üretebileceğinden daha fazla fikir üretebilir ve çözüm önerisi geliştirebilir.Yani; bir kişinin çözüm üretmesinin zor olduğu durumlarda ekip olarak çalışmanın faydalı olduğunu gördük.
* Yanlış karar verme ve yanlış uygulama riski ekip çalışması ile en aza indirgenmiş oldu.

**4.Aksayan Yönler:**

Projeyi gerçekleştirirken karşılaştığımız sorunlar:

* Lehimleme işinin fazla zaman alması karşılaştığımız tek sorundu.

**5.Görüş ve Öneriler:**

Bu proje ile alakalı görüşümüz projenin her alanda kullanılabileceği için çok fonksiyonlu oluşu. Bu proje çeşitli mekanlarda aydınlatma sağlanılabilir.

Önerilerimiz ise bu projenin farklı şekillerde de çalışabileceği yönünde. Örneğin bu proje Vu Metre eklenerek daha profesyonel hale getirilebilir ve ses düzeyine göre yanıp sönebilir.