

**T.C.**  
**MİLLÎ EĞİTİM BAKANLIĞI**  
**SİVAS MERKEZ**  
**Sivas Fen Lisesi**  
**Bilgisayar Bilimi Dersi**  
**Rapor**

<b>Rapor No</b>	Proje-1
<b>Rapor Tarih</b>	09/01/2018
<b>Proje Adı</b>	Arduino ile Çizgi Takip Eden Robot

**Bilgisayar Bilimi Öğretmeni**  
**Ersin TÜTÜNCÜ**  
**2017-2018**



**T.C.**  
**MİLLÎ EĞİTİM BAKANLIĞI**  
**SİVAS MERKEZ**  
**Sivas Fen Lisesi**  
**Proje Grup**

<b>Proje Görev</b>	<b>Numara</b>	<b>Ad-Soyadı</b>
Proje Yönetimi	235	Ceren AĞBABA
Doküman Yönetimi	346	Hatice Şura AYDOĞDU
Lojistik Yönetim	478	Mehmet Tekin ÇATALTEPE
Yazılım Geliştirme	315	Elif YÜCEL
Web ve GitHub Yönetimi	33	İsra AYDOĞDU
Sunum Yönetimi	216	Sudenaz ÜNAL

ÖZET .....	4
Anahtar Kelimeler .....	4
Proje Görev Dağılımı Listesi ve Görev Dağılımı Açıklaması.....	5
1. Görev Dağılımı ve Sorumlusu.....	5
2. Görev Süresince Sürdürülen Eylemler .....	6
3. Görevlerin İş Yüğü Şeması.....	7
4. Yoklama Çizelgeleri .....	8
5. Haftalık İş Katkı Cetvelleri .....	9
GİRİŞ .....	10
Projenin Açıklaması .....	10
YAPILAN ÇALIŞMALAR .....	10
1. Donanım Yapısı .....	10
a.Gömülü Sistemler Mimarisi ve Devre Tasarımı .....	10
b.Mekanik Sistem Mimarisinin Tanıtılması .....	15
1)HC-06 Bluetooth Modülü .....	15
2. Yazılım Yapısı.....	19
a. Algoritma .....	19
b. Kodlama.....	20
1. Bilgi Düzeyine Katkıları.....	24
2. Teknolojik Katkıları.....	24
3. Ekip Çalışmasının Katkıları .....	25
4. Aksayan Yönler.....	25
5. Görüş ve Öneriler.....	25

## **ÖZET**

Bu ikinci proje kapsamında ARDUINO programlama dili yardımıyla bluetooth sinyalleri ile yönlendirilen ve mesafe sensörü sayesinde çarpmalardan sakınan tank yapılacaktır. Bu projede cep telefonumuza yüklenen Arduino Bluetooth RC Car uygulaması sayesinde üretilen sinyaller Arduino Uno mikrodenetleyici tarafından yorumlanacak ve tekerlekler çeviren motorlara hareket verecektir. Bu arada tankın ön tarafına yerleştirilen bir mesafe sensörü ile tankın engellerle arasındaki mesafe sürekli ölçülerek programlanan mesafenin korunması sağlanacaktır.

## **Anahtar Kelimeler**

Arduino, Bluetooth, Robot, Hc-sr04 Arduino Ultrasonic Mesafe Sensörü

## GÖREV DAĞILIMI

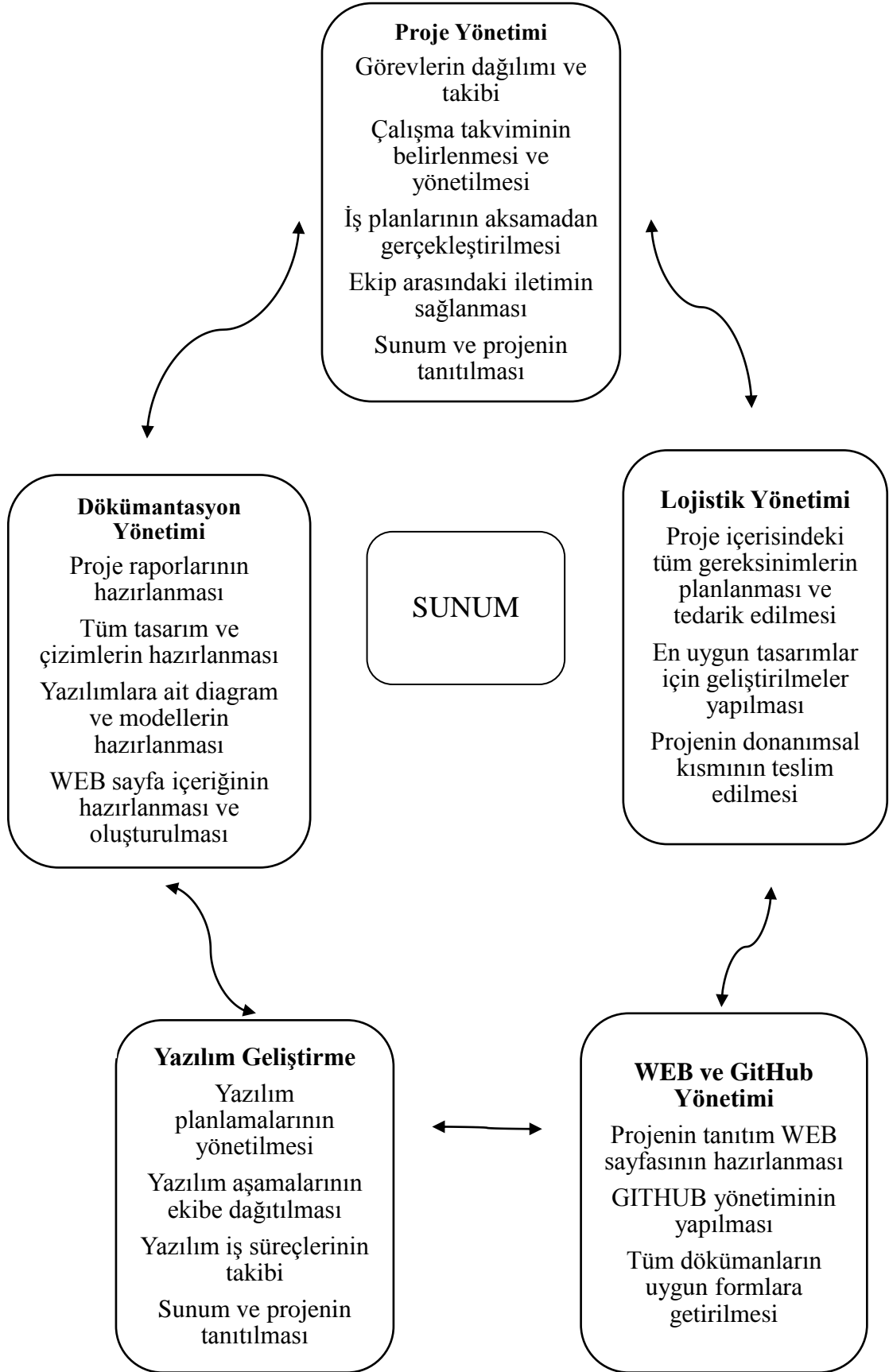
### 1. Görev Dağılımı ve Sorumlusu

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.	Ceren AĞBABA
Doküman Yönetimi	Projenin tüm tasarım ve çizimlerinden, proje raporunun sunulmasından, dokümanların uygun forma getirilmesinden kodlamaya ait diyagram ve modellerin hazırlanması ve web sitesi tasarımından sorumlu olan kişidir.	Hatice AYDOĞDU
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.	Mehmet Tekin ÇATALTEPE
Yazılım Geliştirme Yönetimi	Yazılım için araştırmaların yapılması, yazılım aşamalarının proje grubuna dağıtılması, yazılım ile ilgili raporların hazırlanarak ilgili bölüme(doküman yönetimine) aktarılması yazılım ve süreç testlerinin gerçekleştirilmesi ile ilgilenen kişidir.	Sudenaz ÜNAL
WEB ve GitHub Yönetimi	Proje tanıtımı için WEB sayfasının hazırlanması, projenin GitHub yönetiminin yapılması, doküman yöneticisinden almış olduğu raporlar ile WEB sitesine ve GitHub'a işlemekle sorumlu olan kişidir.	İsra AYDOĞDU
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.	Elif YÜCEL

## 2. Görev Süresince Sürdürülen Eylemler

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

### 3. Görevlerin İş Yüğü Şeması



#### 4. Yoklama Çizelgeleri

<b>Tarihler</b> <b>Grup</b>	24.04.2018	01.05.2018	08.05.2018	15.05.2018	22.05.2018
İsra AYDOĞDU	+	+	+	+	+
Sudenaz ÜNAL	+	+	+	+	+
Ceren AĞBABA	+	+	+	+	+
Elif YÜCEL	+	+	+	+	+
Hatice AYDOĞDU	+	+	+	+	+
Mehmet Tekin ÇATALTEPE	+	+	+	+	+



## 5. Haftalık İş Katkı Cetvelleri

Tarihler	Yapılan iş
24.04.2018	Proje araştırması
01.05.2018	Malzeme Temin Edilmesi
08.05.2018	Elektronik ve Mekanik Tasarım
15.05.2018	Yazılım
22.05.2018	Deneme ve Test Aşamaları

# GİRİŞ

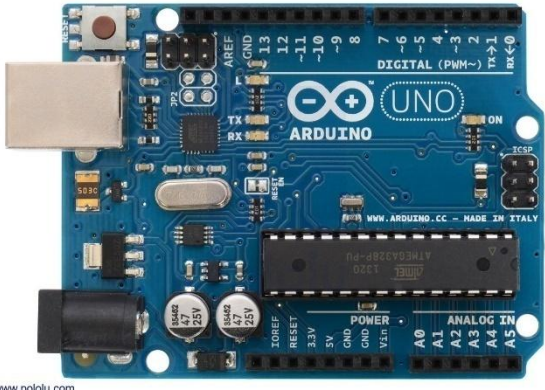
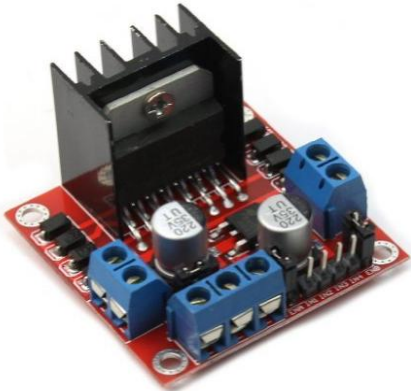
## Projenin Açıklaması

Yapılan projede amaç; ARDUINO programlama dili yardımıyla bluetooth sinyalleri ile yönlendirilen ve mesafe sensörü sayesinde çarpmalardan sakınan tank yapmaktır. Bu projede cep telefonumuza yüklenen Arduino Bluetooth RC Car uygulaması sayesinde üretilen sinyaller Arduino Uno mikrodnetleyici tarafından yorumlanacak ve tekerlekleri çeviren motorlara hareket verecektir. Bu arada tankın ön tarafına yerleştirilen bir mesafe sensörü ile tankın engellerle arasındaki mesafe sürekli ölçülerek programlanan mesafenin korunması sağlanacaktır.

## YAPILAN ÇALIŞMALAR

### 1. Donanım Yapısı

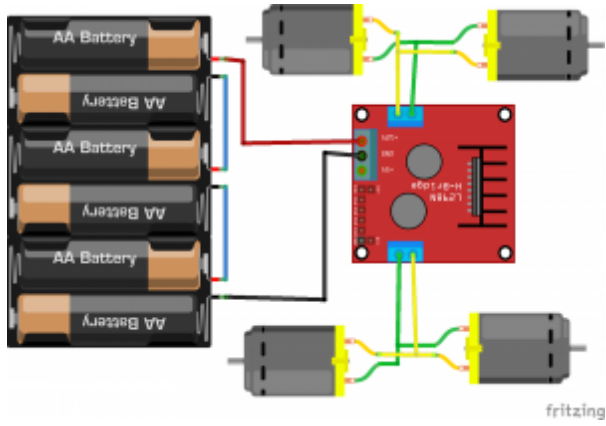
#### a.Gömülü Sistemler Mimarisi ve Devre Tasarımı

	<p><b>ArduinoUno</b> ArduinoUno; Atmega328 temelli bir mikrodnetleyici kartıdır. Üzerinde 14 adet dijital giriş/çıkış pini (6 tanesi PWM çıkışı olarak kullanılabilir), 6 analog giriş, 16Mhz kristal, usb soketi, güç soketi, ICSP konektörü ve reset tuşu bulundurmaktadır. Kart üzerinde mikrodnetleyicinin çalışması için gerekli olan her şey bulunmaktadır. Kolayca usb kablosu üzerinden bilgisayara bağlanabilir, adaptör veya pil ile çalıştırılabilir.</p>
	<p><b>L298N Voltaj Regülatörlü Çift Motor Sürücü Kartı</b> 24V'a kadar olan motorları sürmek için hazırlanmış olan bu motor sürücü kartı iki kanallı olup, kanal başına 2A akım vermektedir. Kart üzerinde L298N motor sürücü entegresi kullanılmıştır. DC motorlardan ayrı olarak step motor kontrolüne de imkân sağlamaktadır.</p>

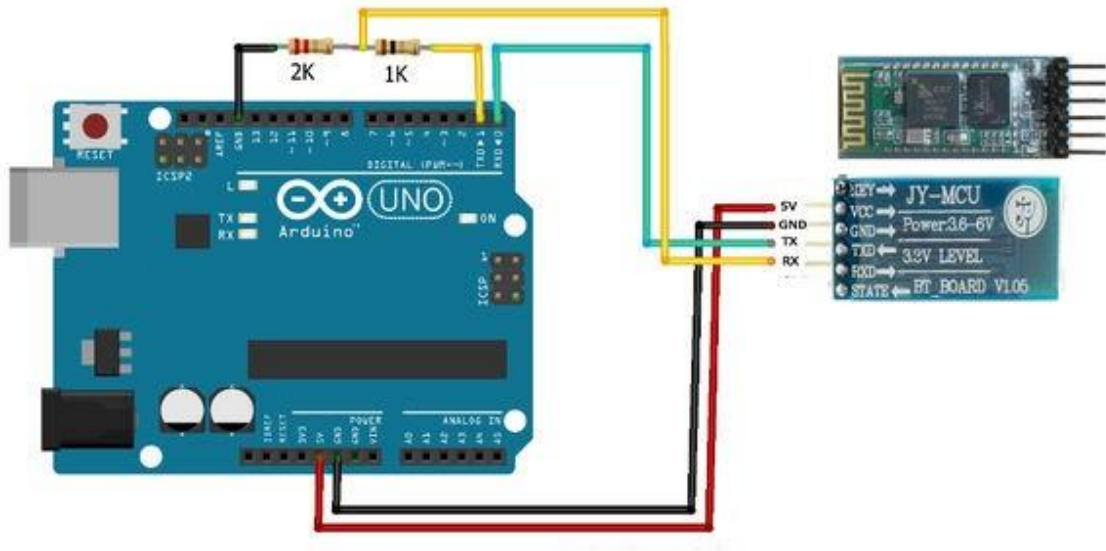
	<p><b>HC-06 Bluetooth Modülü</b>  Slave modunda <b>bluetooth modüle</b> dışardan bir bağlantı yapılabilirken, modül ile başka bir bluetooth cihaza ilk bağlantı yapılamıyor. Yani siz <b>HC-06</b> ile başka bir cihaza isteğiniz doğrultusunda bağlanamazsınız. Ancak <b>HC-06 modülüne</b> dışardan bir cihaz bağlanabilir. Master mod ise sizin elinizdeki modül ile dışardaki bir bluetooth cihazına direk olarak ilk bağlantıyı yapmanıza olanak sağlıyor.</p>
	<p><b>Pertinaks</b>  Prototip devre elemanlarını pertinaks üzerine lehimleyip farklı uygulamalar için kullanabiliriz.</p>
	<p><b>Jumper Kablo (E-E/D-E)</b>  Devre elemanlarının bağlantılarını gerçekleştirmek için kullanabiliriz.</p>

	<p><b>Tekerlek</b>  42 mm çapında yumuşak ve yüksek sürtünmeli bir tekerlektir. 16mm çaplı, mikro ve mini metal redüktörlü motorlarla tam uyumludur.  Çapı:42mm  Genişlik:19mm  Ağırlık:16gr</p>
	<p><b>6V 250 Rpm Motor</b>  <b>Motor Özellikleri:</b>  Çalışma Voltajı: 3-12V  Redüksiyon Oranı: 1:48  Hız: 250 Rpm(@6V)  Akım: 95mA (maks. 160mA)  Ağırlık: 29gr</p>
	<p><b>Hc-cs04 Arduino Ultrasonic Mesafe Sensörü</b></p> <p>Ultrasonik sensörler, mesafe ölçümlerinde yaygın olarak kullanılmaktadır. Bu ultrasonik ses dalgaları, insan kulağının duyamayacağı 20 kHz – 500 kHz frekans bandına sahiptir. Ultrasonik ses dalgalarını kullanan sensörler ile nesnelerle herhangi bir temas halinde olmadan mesafe ölçümleri yapılabilir.</p>

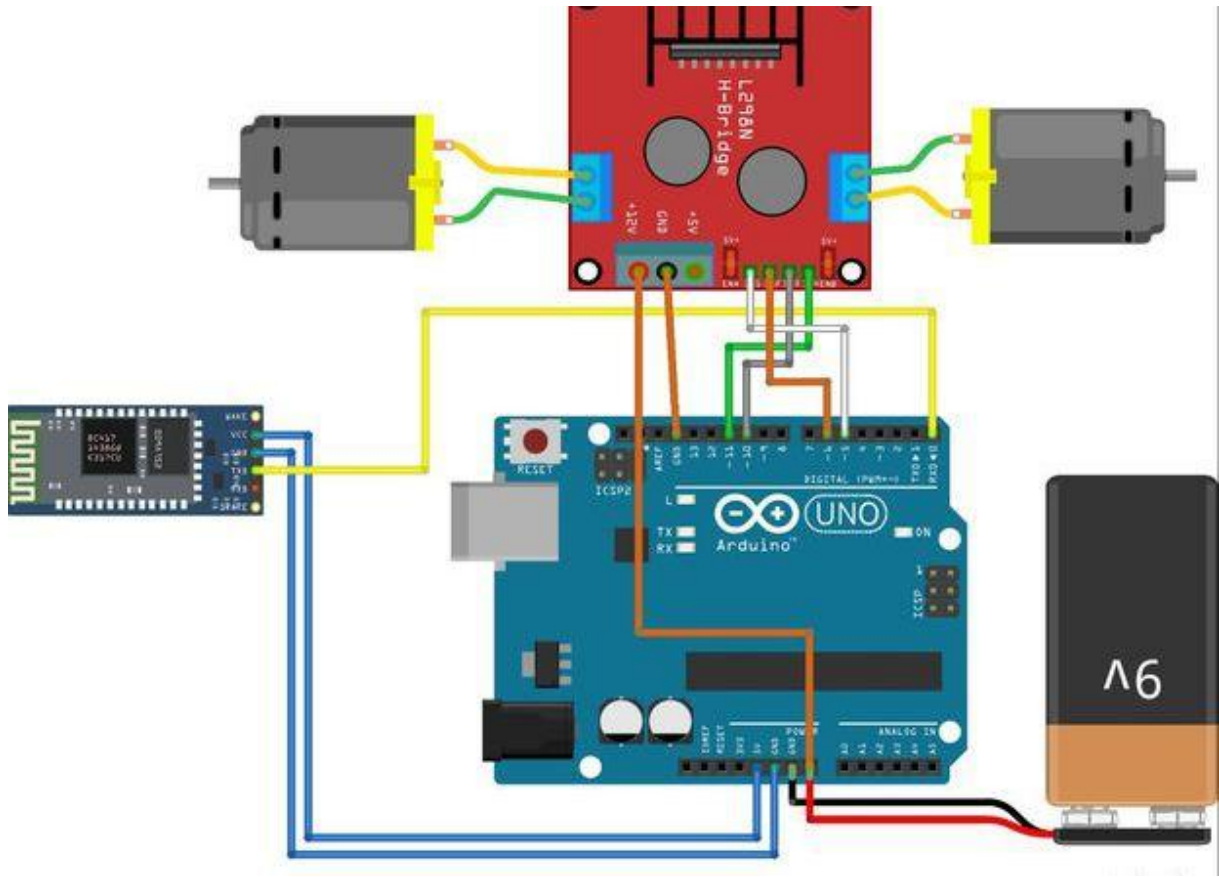
## L298N bağlantısı



## HC-06 bağlantısı:

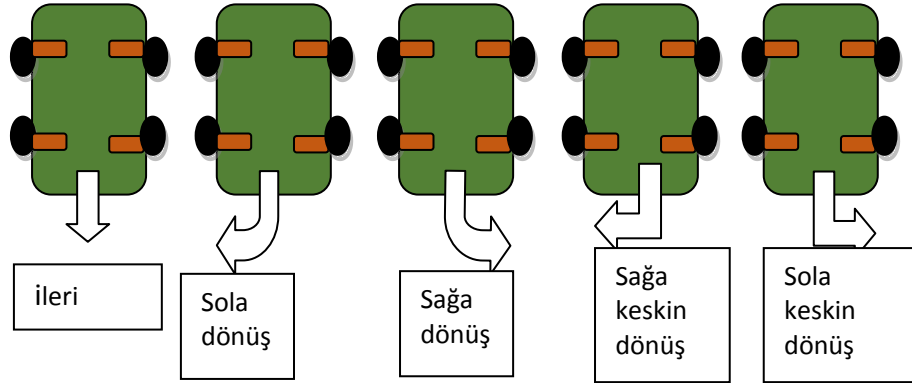


## Devre Şeması:





## b.Mekanik Sistem Mimarisinin Tanıtılması

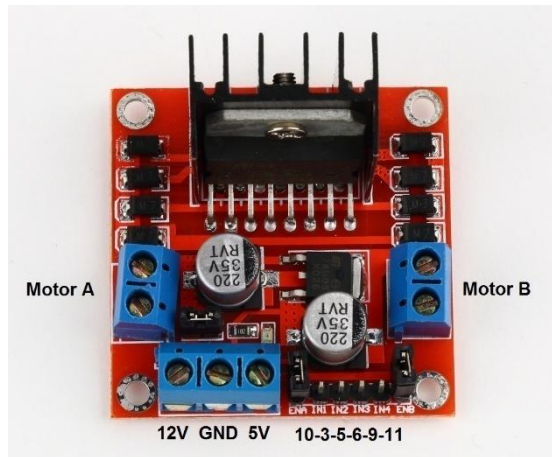


### 1)HC-06 Bluetooth Modülü



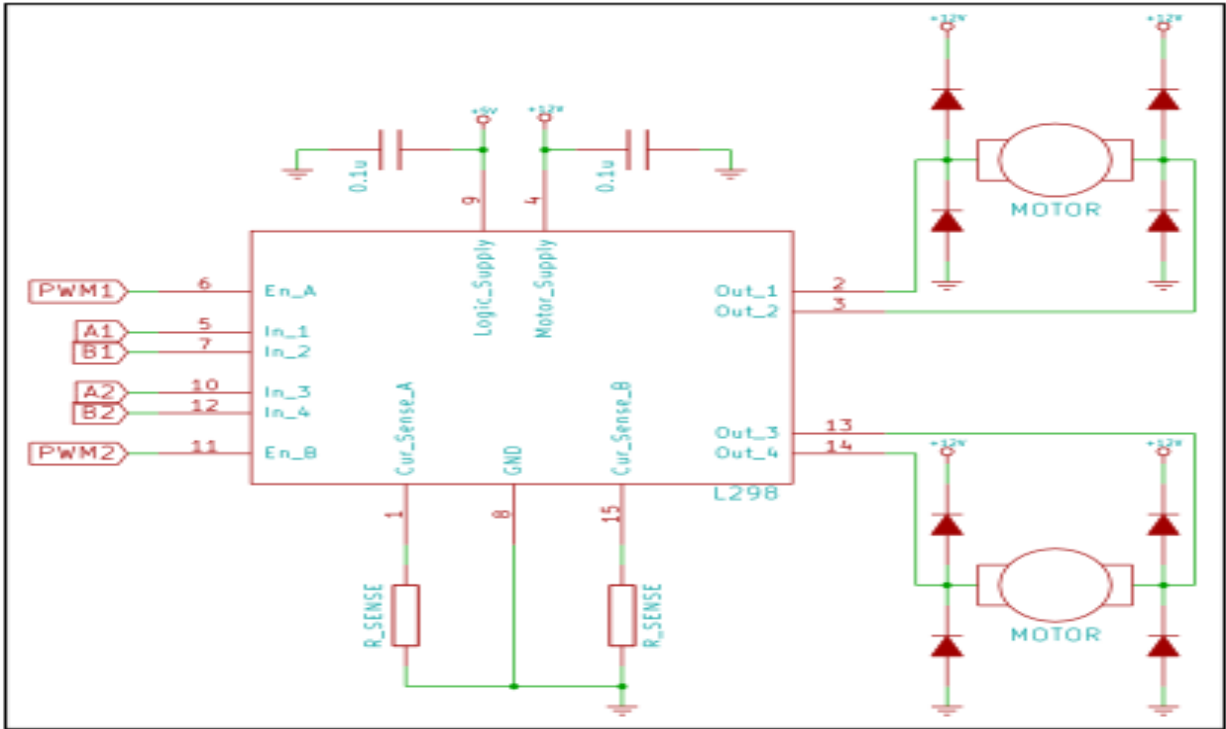
Çalışma mantığı ise seri haberleşme esasına dayanır. **Tx ucu verici ucudur**, Rx ise alıcı ucudur. 3.3V ve 5V arasındaki gerilimlerde çalışabilir. Ancak veri alış verişini 3.3V ile yapmaktadır. Verici ucundan gönderilen veriyi mikrodnetleyici algılar ama mikrodnetleyicinin çıkış voltajı 5V olduğu için bluetooth modülümüze zarar verir. Bu yüzden mikrodnetleyici çıkış ucuna gerilim bölücü uygulayıp istenilen gerilime indirmeliyiz.

### 2.L298N Motor Sürücü Kartı:

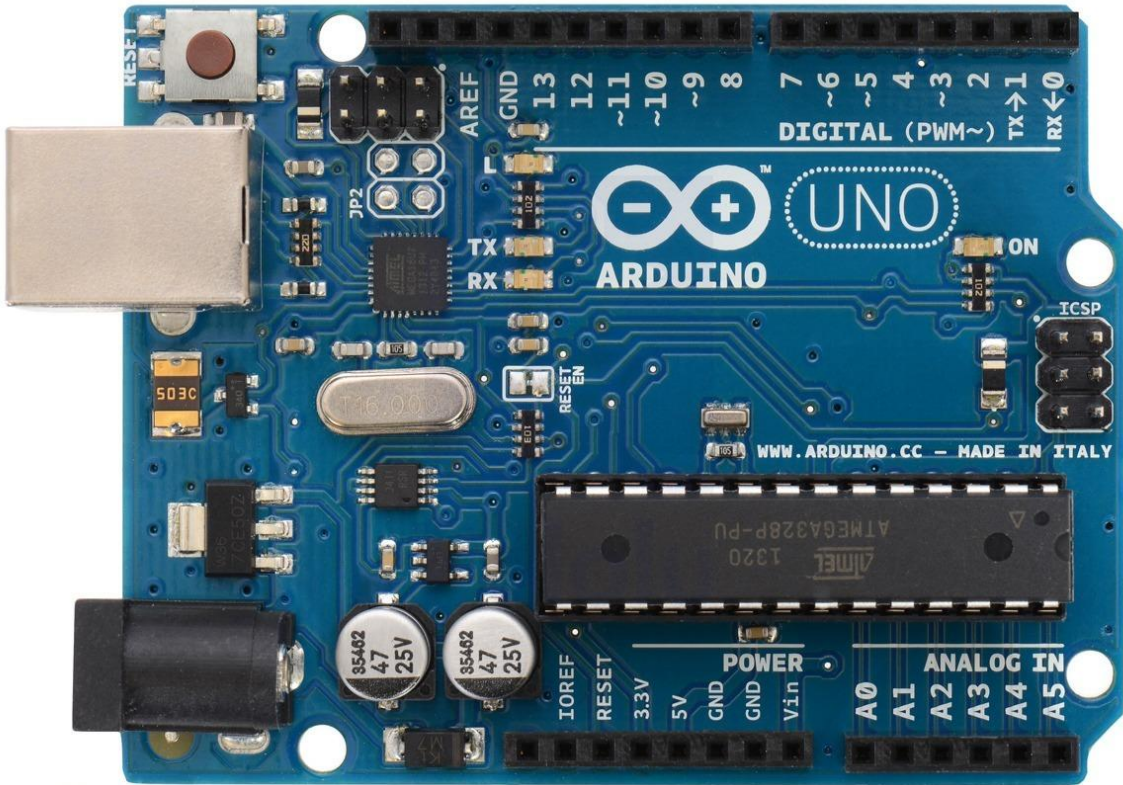


2 adet H köprüsü bulunur. H köprüsü DC motorun iki yönde de hareket etmesini sağlayan faydalı bir yöntemdir.Bu yöntem için 4 adet transistör kullanılır.Bu entegre içerisinde toplam 15 adet bacak bulunmaktadır.Bunlardan IN1,IN2,OUT1,OUT2,ENA,SENSA A köprüsü için IN3,IN4,OUT3,OUT4,ENB,SENSB B köprüsü içindir.IN1 ve IN2 girişleri 5V'a duyarlı girişlerdir.OUT1 ve OUT2 isminden de anlaşılacağı gibi çıkış işlemleri içindir.

## Devre Şeması



## 3.Arduino Uno



## Özellikler



<b>Mikro denetleyici</b>	ATUno 2560
<b>Çalışma Gerilimi</b>	5V
<b>Giriş Gerilimi(önerilen)</b>	7-12V
<b>Dijital I/O Pinleri</b>	54 (15 tanesi PWM çıkışı)
<b>Analog Giriş Pinleri</b>	16
<b>Her I/O için Akım</b>	40mA
<b>3.3V Çıkış için Akım</b>	50 mA
<b>EEPROM</b>	4 KB (ATmega2560)
<b>SRAM</b>	8KB(ATMega 2560)
<b>Flash Hafıza</b>	256 KB (Atmega2560) 8 KB kadarı bootloader tarafından kullanılmaktadır.
<b>Uzunluk</b>	101.6 mm
<b>Genişlik</b>	53.4mm
<b>Ağırlık</b>	36gr

Arduino Uno 2560 gücünü USB üzerinden veya harici bir güç kaynağından alabilir. Kartın çalışması için USB kablusunun sürekli bağlı olması gerekli değildir. Kart sadece adaptör veya batarya ile de çalıştırılabilir. Bu da kartın bilgisayardan bağımsız da çalışabileceğini gösterir. Harici güç kaynağı olarak 6-20V arası kullanılabilir. Bu değerler limit değerleridir. Önerilen harici besleme 7-12V arasındadır. 12V üzerindeki değerlerde kart aşırı ısınabilir. Mega kartının üzerindeki mikro denetleyicinin çalışma gerilimi 5V'dur.

Güç pinleri:

- **VIN:** Harici güç kaynağı kullanırken 7-12V arası gerilim giriş pini.
- **5V:** Regülatörden çıkan 5V çıkışı verir. Eğer kart sadece USB (5V) üzerinden çalışıyor ise USB üzerinden gelen 5V doğrudan bu pin üzerinden çıkış olarak verilir. Eğer karta güç Vin (7-12V) veya güç soketi (7-12V) üzerinden veriliyorsa regülatörden çıkan 5V doğrudan bu pin üzerinden çıkış olarak verilir.
- **3V:** Kart üzerinde bulunan 3.3V regülatörlü çıkış pinidir. Maks. 50mA çıkış verebilir.
- **GND:** Toprak pinleridir.

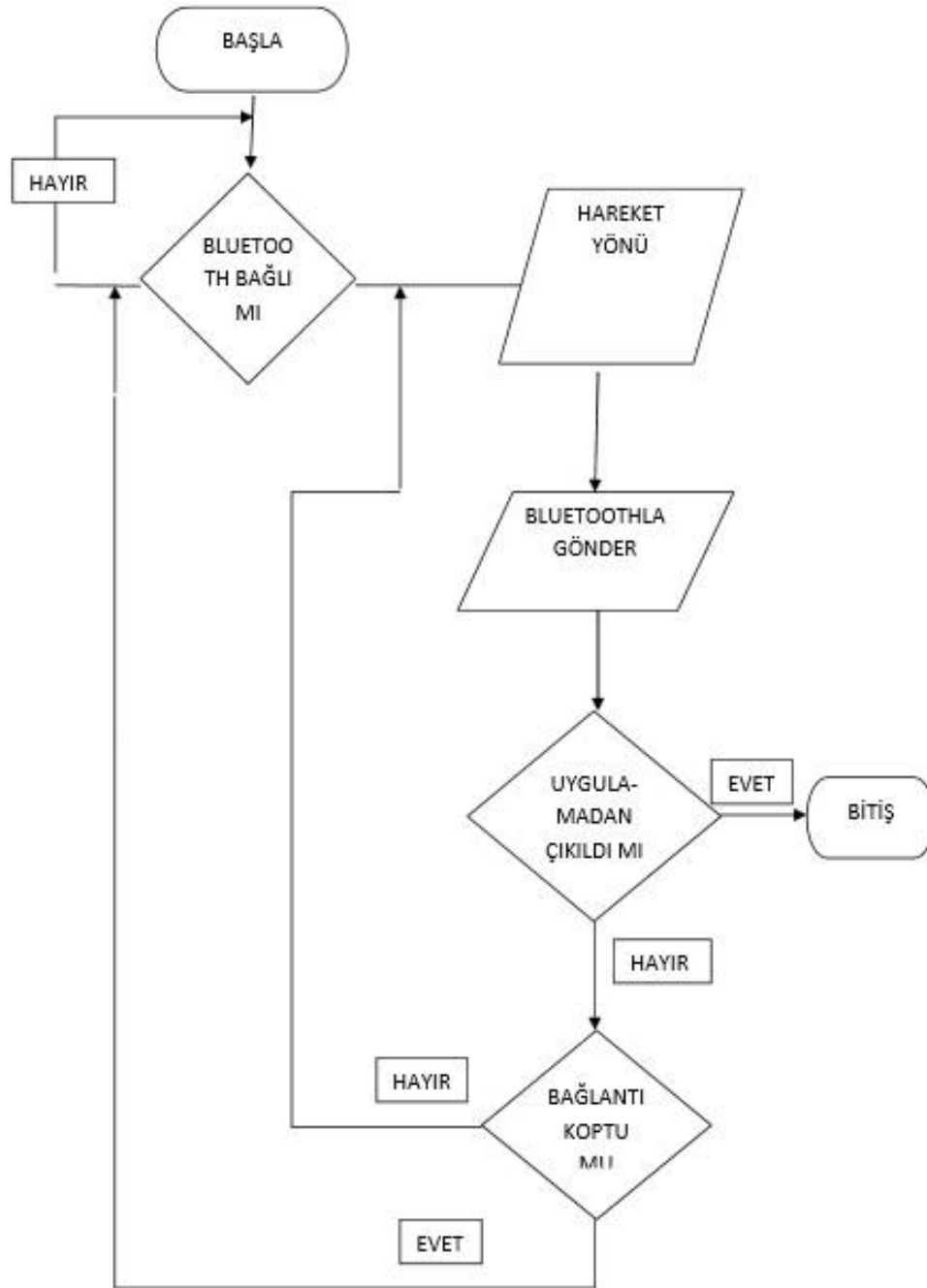
#### 4. DC Motor 6V



##### Özellikler:

- |                          |          |                             |           |
|--------------------------|----------|-----------------------------|-----------|
| • Çalışma Voltajı        | : 6V     | • Zorlanma Akımı(12V)       | : 2.9A    |
| • Azami Çalışma Voltajı  | : 12V    | • Motor Çapı                | : 12-12mm |
| • Devir                  | : 600Rpm | • Redüktör Çapı- Kare       | : 13x12mm |
| • Boşta Çektiği Akım(6V) | : 50mA   | • MilOrtadan Çıkışlı D Şaft | : 3mm     |
| • Zorlanma Akımı(6V)     | : 1.6A   | • Mil Uzunluğu              | : 10mm    |
| • Ağırlık                | : 16 gr  | • Uçtan Uca Uzunluk         | : 42mm    |

## 2. Yazılım Yapısı



### a. Algoritma

- Android uygulamasında ileri tuşuna basıldı ise; tüm motorlar sürülür, bu arada sürekli olarak mesafe sensörü ile mesafe kontrol edilir. Mesafe programlanan seviyeye geldiğinde motorlar durdurulur.
- Android uygulamasında geri tuşuna basıldı ise; tüm motorlar ters yönde sürülür.
- Android uygulamasında sağ tuşuna basıldı ise; sağ tekerleklere bağlı motorlar ileri yönde, sol tekerleklere bağlı motorlar tersi yönde sürülür.

- Android uygulamasında sol tuşuna basıldı ise; sol tekerleklere bağlı motorlar ileri yönde, sağ tekerleklere bağlı motorlar tersi yönde sürülür.
- Android uygulamasında sağ çapraz tuşuna basıldı ise; sol tekerleklere bağlı motorlar ileri yönde, sağ tekerleklere bağlı motorlar daha yavaş olarak ileri yönde sürülür.
- Android uygulamasında sol çapraz tuşuna basıldı ise; sağ tekerleklere bağlı motorlar ileri yönde, sol tekerleklere bağlı motorlar daha yavaş olarak ileri yönde sürülür.
- Android uygulamasında sağ geri çapraz tuşuna basıldı ise; sol tekerleklere bağlı motorlar ters yönde, sağ tekerleklere bağlı motorlar daha yavaş olarak ters yönde sürülür.
- Android uygulamasında sol çapraz tuşuna basıldı ise; sağ tekerleklere bağlı motorlar ters yönde, sol tekerleklere bağlı motorlar daha yavaş olarak ters yönde sürülür.

## **b. Kodlama**

```
const int motorA1 = 5; // L298N'in IN3 Girişi

const int motorA2 = 6; // L298N'in IN1 Girişi

const int motorB1 = 10; // L298N'in IN2 Girişi

const int motorB2 = 9; // L298N'in IN4 Girişi


int i=0; //Döngüler için atanan rastgele bir değişken
int j=0; //Döngüler için atanan rastgele bir değişken
char state; //Bluetooth cihazından gelecek sinyalin değişkeni
int vSpeed=255; // Standart Hız, 0-255 arası bir değer alabilir

#define trigPin 13 //Sensörün Echo pini Arduinonun 13. pinine bağlanır
#define echoPin 12 //Sensorün Trig pini Arduinonun 12. pinine bağlanır

void setup() {

    // Pinlerimizi belirleyelim

    pinMode(motorA1, OUTPUT);

    pinMode(motorA2, OUTPUT);

    pinMode(motorB1, OUTPUT);

    pinMode(motorB2, OUTPUT);

    // 9600 baud hızında bir seri port açalım

    Serial.begin(9600);

    pinMode(trigPin, OUTPUT); //13. yani trigpini çıkış olarak ayarlıyoruz
    pinMode(echoPin, INPUT); //12. yani echoPini giriş olarak ayarlıyoruz
```

```
}
```

```
void loop() {
```

```
/*Bluetooth bağlantısı koptuğunda veya kesildiğinde arabayı durdur.
```

```
(Aktif etmek için alt satırın "/" larını kaldırın.)*/
```

```
if(digitalRead(state)==LOW) { state='S'; }
```

```
//Gelen veriyi 'state' değişkenine kaydet
```

```
if(Serial.available() > 0){
```

```
    state = Serial.read();
```

```
    Serial.println(state);
```

```
}
```

```
//if(distance>40) {
```

```
/* Uygulamadan ayarlanabilen 4 hız seviyesi.(Değerler 0-255 arasında olmalı)*/
```

```
if (state == '0'){
```

```
    vSpeed=0;}
```

```
else if (state == '1'){
```

```
    vSpeed=100;}
```

```
else if (state == '2'){
```

```
    vSpeed=180;}
```

```
else if (state == '3'){
```

```
    vSpeed=200;}
```

```
else if (state == '4'){
```

```
    vSpeed=255;}
```

```
/******İleri******/
```

```
//Gelen veri 'F' ise araba ileri gider.
```

```

if (state == 'F') {
    long duration, distance;
    digitalWrite(trigPin, LOW);
    delayMicroseconds(5);
    digitalWrite(trigPin, HIGH);
    delayMicroseconds(10);
    digitalWrite(trigPin, LOW);
    duration = pulseIn(echoPin, HIGH);
    distance = (duration/2) / 29.1; //Ölçüm fonksiyonu
    Serial.print(distance);
    if(distance>30){
        analogWrite(motorA1, vSpeed); analogWrite(motorA2, 0);
        analogWrite(motorB1, vSpeed); analogWrite(motorB2, 0);
        Serial.println(" -> ileri");
    }else {
        Serial.println(" -> DUR");
        state='S';
    }
}

/*****Geri*****/
//Gelen veri 'T' ise araba ileri sol(çapraz) gider.
else if (state == 'T') {
    analogWrite(motorA1,250 ); analogWrite(motorA2, 0);
    analogWrite(motorB1, 100); analogWrite(motorB2, 0);
}

/*****İleri Sağ*****/
//Gelen veri 'G' ise araba ileri sağ(çapraz) gider.
else if (state == 'G') {
    analogWrite(motorA1, 100); analogWrite(motorA2, 0);

```

```

    analogWrite(motorB1, 250);    analogWrite(motorB2, 0);
}

//Gelen veri 'B' ise araba geri gider.

else if (state == 'B') {

    analogWrite(motorA1, 0);  analogWrite(motorA2, vSpeed);

    analogWrite(motorB1, 0);  analogWrite(motorB2, vSpeed);

}

/*****Sol*****/

//Gelen veri 'R' ise araba sola gider.

else if (state == 'R') {

    analogWrite(motorA1, vSpeed);  analogWrite(motorA2, 0);

    analogWrite(motorB1, 0); analogWrite(motorB2, vSpeed);

}

/*****Sağ*****/

//Gelen veri 'L' ise araba sağa gider.

else if (state == 'L') {

    analogWrite(motorA1, 0);  analogWrite(motorA2, vSpeed);

    analogWrite(motorB1, vSpeed);  analogWrite(motorB2, 0);

}

/*****İleri Sol*****/

/*****Geri Sol*****/

//Gelen veri 'J' ise araba geri sol(çapraz) gider

else if (state == 'H') {

    analogWrite(motorA1, 0);  analogWrite(motorA2, 100);

    analogWrite(motorB1, 0); analogWrite(motorB2, 250);

}

/*****Geri Sağ*****/

//Gelen veri 'H' ise araba geri sağ(çapraz) gider

```

```

else if (state == 'J') {

    analogWrite(motorA1, 0); analogWrite(motorA2, 250);

    analogWrite(motorB1, 0); analogWrite(motorB2, 100);

}

/*****Stop*****/

//Gelen veri 'S' ise arabayı durdur.

else if (state == 'S'){

    analogWrite(motorA1, 0); analogWrite(motorA2, 0);

    analogWrite(motorB1, 0); analogWrite(motorB2, 0);

}

// }

}

```

## SONUÇ

### 1. Bilgi Düzeyine Katkıları

Her şeyden 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 elektronik ve mekanik tasarımının nasıl olması gerektiğini öğrendik ve yaptığımız yanlışlar ile tasarım aşamasında yapılmaması gerekenleri fark ettik.

### 2. Teknolojik Katkıları

Çizgi izleyen robotlar günümüzde gelişen teknolojiyle birlikte endüstriyel alanda bir süredir kullanılmaktadır. Genellikle lojistik ve otomasyon bölümleri içerisinde oldukça gereksinim duyulan niteliksiz insan gücü ile yapılan taşıma işlemlerini bir süredir çizgi izleyen robotlar yapmaya başlamışlardır. Günümüzde bu kadar yaygın olarak kullanılan çizgi takip eden robot tasarımını gerçekleştirmiş olmamız bize iş hayatımızda kazanım olarak döneceğini ve bizim için önemli bir tecrübe, deneyim olduğunu düşünüyoruz.



### 3. Ekip Çalışmasının Katkıları

Ekip çalışmasının bize kattıkları:

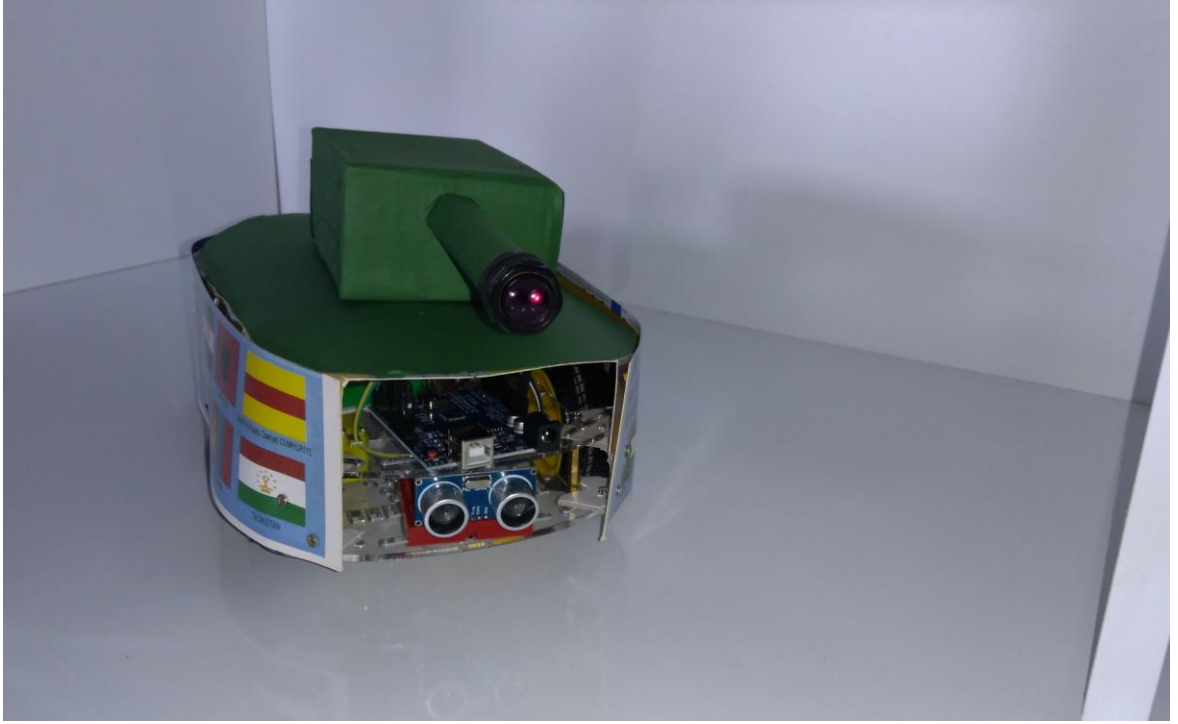
- Her grup elemanının teknik bilgisi arttı.
- Konuya bütünlük açısından bakmamıza yardımcı oldu.
- Proje, çalışanların sorun çözme becerisini geliştirdi.
- Çalışanların birbiriyle iletişim alışkanlıklarını geliştirdi.
- Ekip, bir kişinin tek başına üretebileceğinden daha fazla fikir üretebilir ve çözüm önerisi geliştirebileceğinden ekip olarak çalışmanın faydalı olduğunu gördük.
- Yanlış karar verme ve yanlış uygulama riski ekip çalışması ile en aza indirgenmiş oldu.

### 4. Aksayan Yönler

- Devre ve program hazır hale getirildikten sonra zaman zaman komutların işletilmesinde sorunlar yaşadık. Hiçbir ekleme veya çıkarma yapmamıza rağmen tankın bazen komutları yerine getirdiği, bazen ise tepki vermediği durumlarla karşılaştık. Öncelikle bu sorunun pilin akımının yeterli olmamasından kaynaklanacağı üzerinde durduk. Kuvvetli pillerle yeniden denemeler yaptık. Pilin değişmesi ile bazen sorunun çözüldüğü oldu, ancak hala aynı sorunla bazen karşılaşabiliyoruz. Sebepleri üzerine araştırmalarımız devam etmektedir.
- Devre Tasarımının Çizimi (Fritzing, Proteus, Circuits ): Devre tasarımının çiziminde kullandığımız devre elemanlarının fritzing, proteus gibi devre tasarımı programlarında bulunmaması karşımıza bir sorun olarak çıktı. Çözüm olarak ‘Paint’ çizim programını kullanarak devre tasarımını gerçekleştirdik.

### 5. Görüş ve Öneriler

Bu projede elektronik ve mekanik iki kısım bulunmakta olması hem elektronik ve hem de mekanik becerilerimizi artırmamızı sağladı. Bu iki kısmın birbirleriyle uyumu açısından irdelendiğinde hem teoride hem uygulamada proje grubumuza birçok bilgi birikimi sağlamıştır. Edindiğimiz bu bilgiler sayesinde yapacağımız diğer projelerde daha başarılı olacağımıza inanıyoruz. İlerleyen aşamalarda yapacağımız yeni çalışmalarla mesafe sensörü sayısı artırılarak geri harekette, çarpaz hareketlerde de çarpmanın önüne geçebiliriz.



GitHub Adresi

[https://github.com/KullanAt/9C\\_Grup1\\_Proje2](https://github.com/KullanAt/9C_Grup1_Proje2)