**RASPBERRY Pİ KULLANARAK BLUETOOTH KONTROLLÜ TAŞIYICI ARAÇ**

**(OFFICE BOY)**

**Giriş**

Bu araç bluetooth üzerinden kontrol edilebilen aracın tasarımıdır. Raspberry pi, arduino ve bluetooth kullanarak bir işyerinde bölgeler arası taşımacılığı sağlamak amacıyla inşa edilmiştir. Bu projede, robotik aracı oluşturmak için ağırlıklı olarak raspberry pi, bluetooth modülü, arduino, motor sürücü ve çizgi izleyen sensörler kullanılmıştır.

Proje üzerinde iletişim sağlayabileceğimiz bir bluetooth modülü bulunmaktadır. Bluetooth modülü sayesinde kumanda ile uzaktan kontrolü sağlanmaktadır. Kumanda uygulaması için app inventor 2 kullanılmıştır. Araç 2 bölgeye göre kumanda üzerinden kontrol edilmektedir (B ve C bölgesi). Kumanda üzerinden gönderdiğimiz sinyale göre çizgileri takip ederek bölgeler arası alışverişi sağlamaktadır.

**Gerekli Donanım Bileşenleri**

1. 1 adet Raspberry Pi
2. 1 adet Arduino
3. 1 adet HC 06 Bluetooth Modül
4. 1 adet Robotik Araba
5. 1 adet Motor Driver l298N
6. 2 adet DC Motor
7. 4’lü Çizgi İzleyen Sensör Seti
8. 1 adet TCRT5000 Sensör Kartı

**Gerekli Yazılım Bileşenleri**

1. Raspbian Jessie OS ([www.raspbian.org](http://www.raspbian.org))
2. Arduino ([www.arduino.cc](http://www.arduino.cc))

**Kullanılan Bileşenleri Özellikleri**

1. Raspberry Pi: Raspberry Pi, İngiltere de bulunan Raspberry Pi Vakfı tarafından desteklenen; öğrenci, amatör ve hobicilerin kullanımına sunulan kredi kartı büyüklüğünde, tek bir board'dan oluşan mini bilgisayardır.

Projemizde raspberry pi 3 kullanılmıştır.

**Raspberry Pi 3 Teknik Özellikleri**

* Broadcom BCM2837 SoC
* 1.2 GHz 4 çekirdekli 64-bit ARM Cortex-A53 işlemci
* 2 çekirdekli Videocore IV® Multimedia işlemcisi
* 1 GB LPDDR2 bellek
* Dahili 802.11b/g/n  destekli WiFi
* Bluetooth 4.1, low-energy destekli
* 10/100 Mbit destekli Ethernet portu
* HDMI portu (HDMI 1.4 destekli)
* Kompozit video ve ses çıkışı için 3.5mm TRRS (4 uçlu) konektör
* 4 adet USB2.0 portu
* 40 adet GPIO pini, önceki Raspberry Pi modelleri ile uyumlu
* WiFi/Bluetooth için dahili çip anten
* CSI (kamera) ve DSI (ekran) konektörleri
* Mikro SD kart yuvası
* Tüm Raspberry Pi uyumlu Linux dağıtımlarını ve Windows 10 IoT Core işletim sistemini destekler
* Boyutlar: 85 x 56 x 17 mm

1. Arduino: Bir G/Ç kartı ve Processing/Wiring dilinin bir uygulamasını içeren geliştirme ortamından oluşan bir fiziksel programlama platformudur.
2. HC 06 Bluetooth Modül; Bu ürünü şuradan temin ettik (<https://www.direnc.net/hc06-arduino-bluetooth-modul>)
3. Robotik Araba; Bu ürünü şuradan temin ettik (<https://www.robotistan.com/platforma-cok-amacli-mobil-robot-platformu-turuncu>)
4. Motor Driver l298N; Bu ürünü şuradan temin ettik (<https://www.robotistan.com/l298n-voltaj-regulatorlu-cift-motor-surucu-karti>)

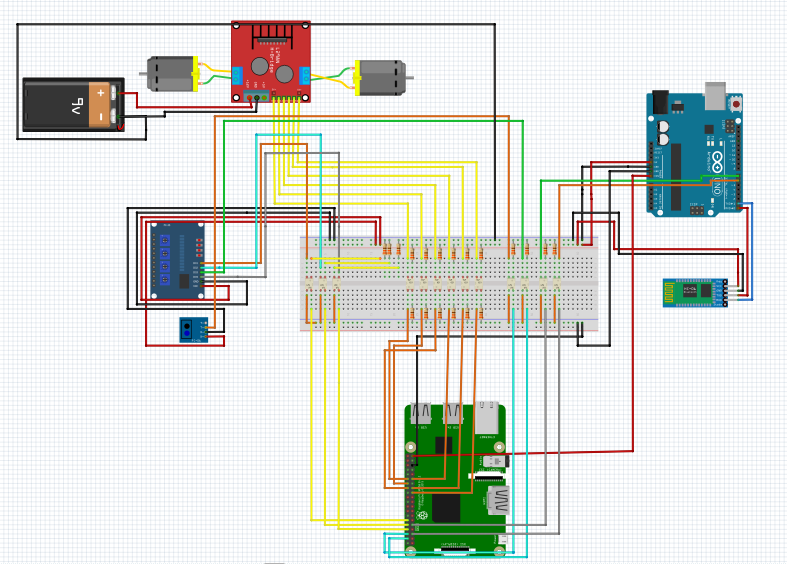
Sürücü kartı motorun hızını ve yönünü ayarlamak amacıyla kullanılmıştır.

1. DC Motor; Bu ürünü şuradan temin ettik (<https://www.robotistan.com/12v-16mm-800rpm-reduktorlu-dc-motor>)

DC motor arabanın ileri hareketini ve dönmesini sağlamak amacıyla kullanılmıştır.

1. 4’lü Çizgi İzleyen Sensör Seti; Bu ürünü şuradan temin ettik (<https://urun.n11.com/diger/4lu-cizgi-izleyen-sensor-seti-4-way-tracking-module-P161032872?cid=604001&gclid=EAIaIQobChMIm5qRzYnL2QIVQY0bCh1ZhAIIEAkYASABEgLQN_D_BwE&gclsrc=aw.ds>)
2. TCRT5000 Sensör Kartı; Bu ürünü şuradan temin ettik; (<https://www.direnc.net/tcrt5000-optical-switches-reflective-phototransistor-output-10974>)

**Şematik Çizimi**

****

**Yapım Aşamaları**

**1.Adım**: İlk olarak Raspberry Pi’deki Raspbian OS’yi güncellemek için aşağıdaki komutu çalıştırın:

**sudo apt-get update**

**2.Adım:** Raspberry pi gpiozero python 3 yükleme

sudo apt install python3-gpiozero

**Python Kodu**

from gpiozero import Button, LED

from time import sleep

mr\_2 = LED(17)

mr\_1 = LED(27)

mr\_e = LED(22)

ml\_2 = LED(18)

ml\_1 = LED(23)

ml\_e = LED(24)

button\_1 = Button(5)

button\_2 = Button(6)

button\_3 = Button(13)

button\_4 = Button(19)

button\_5 = Button(26)

button\_6 = Button(12)

button\_7 = Button(16)

while True:

mr\_1.on()

mr\_2.off()

mr\_e.off()

ml\_1.on()

ml\_2.off()

ml\_e.off()

if button\_4.is\_pressed==1 and button\_5.is\_pressed==1:

while not button\_6.is\_pressed==1 or not button\_7.is\_pressed==1:

if not button\_1.is\_pressed and button\_2.is\_pressed and not button\_3.is\_pressed:

mr\_1.off()

mr\_2.off()

mr\_e.off()

ml\_1.off()

ml\_2.off()

ml\_e.off()

sleep(0.045)

mr\_1.on()

mr\_2.off()

mr\_e.on()

ml\_1.on()

ml\_2.off()

ml\_e.on()

sleep(0.030)

if button\_1.is\_pressed and not button\_2.is\_pressed and not button\_3.is\_pressed:

mr\_1.off()

mr\_2.off()

mr\_e.off()

ml\_1.off()

ml\_2.off()

ml\_e.off()

sleep(0.045)

mr\_1.on()

mr\_2.off()

mr\_e.off()

ml\_1.on()

ml\_2.off()

ml\_e.on()

sleep(0.030)

if not button\_1.is\_pressed and not button\_2.is\_pressed and button\_3.is\_pressed:

mr\_1.off()

mr\_2.off()

mr\_e.off()

ml\_1.off()

ml\_2.off()

ml\_e.off()

sleep(0.045)

mr\_1.on()

mr\_2.off()

mr\_e.on()

ml\_1.on()

ml\_2.off()

ml\_e.off()

sleep(0.030)

if button\_4.is\_pressed==0 and button\_5.is\_pressed==1:

while not button\_6.is\_pressed==0 or not button\_7.is\_pressed==1:

if not button\_1.is\_pressed and button\_2.is\_pressed and not button\_3.is\_pressed:

mr\_1.off()

mr\_2.off()

mr\_e.off()

ml\_1.off()

ml\_2.off()

ml\_e.off()

sleep(0.045)

mr\_1.on()

mr\_2.off()

mr\_e.on()

ml\_1.on()

ml\_2.off()

ml\_e.on()

sleep(0.030)

if button\_1.is\_pressed and not button\_2.is\_pressed and not button\_3.is\_pressed:

mr\_1.off()

mr\_2.off()

mr\_e.off()

ml\_1.off()

ml\_2.off()

ml\_e.off()

sleep(0.045)

mr\_1.on()

mr\_2.off()

mr\_e.off()

ml\_1.on()

ml\_2.off()

ml\_e.on()

sleep(0.030)

if not button\_1.is\_pressed and not button\_2.is\_pressed and button\_3.is\_pressed:

mr\_1.off()

mr\_2.off()

mr\_e.off()

ml\_1.off()

ml\_2.off()

ml\_e.off()

sleep(0.045)

mr\_1.on()

mr\_2.off()

mr\_e.on()

ml\_1.on()

ml\_2.off()

ml\_e.off()

sleep(0.030)

if button\_4.is\_pressed==1 and button\_5.is\_pressed==0:

while not button\_6.is\_pressed==1 or not button\_7.is\_pressed==0:

if not button\_1.is\_pressed and button\_2.is\_pressed and not button\_3.is\_pressed:

mr\_1.off()

mr\_2.off()

mr\_e.off()

ml\_1.off()

ml\_2.off()

ml\_e.off()

sleep(0.045)

mr\_1.on()

mr\_2.off()

mr\_e.on()

ml\_1.on()

ml\_2.off()

ml\_e.on()

sleep(0.030)

if button\_1.is\_pressed and not button\_2.is\_pressed and not button\_3.is\_pressed:

mr\_1.off()

mr\_2.off()

mr\_e.off()

ml\_1.off()

ml\_2.off()

ml\_e.off()

sleep(0.045)

mr\_1.on()

mr\_2.off()

mr\_e.off()

ml\_1.on()

ml\_2.off()

ml\_e.on()

sleep(0.030)

if not button\_1.is\_pressed and not button\_2.is\_pressed and button\_3.is\_pressed:

mr\_1.off()

mr\_2.off()

mr\_e.off()

ml\_1.off()

ml\_2.off()

ml\_e.off()

sleep(0.045)

mr\_1.on()

mr\_2.off()

mr\_e.on()

ml\_1.on()

ml\_2.off()

ml\_e.off()

sleep(0.030)

**Arduino Kodu**

int data;

int led1=7;

int led2=6;

void setup() {

pinMode(led1,OUTPUT);

pinMode(led2,OUTPUT);

pinMode(3,OUTPUT);

pinMode(5,OUTPUT);

Serial.begin(9600);

}

void loop() {

analogWrite(3,90);

analogWrite(5,90);

if(Serial.available()){

int data = Serial.read();

delay(100);

if(data=='A'){

digitalWrite(led1,1);

digitalWrite(led2,1);

Serial.print("data==A");

}

if(data=='B'){

digitalWrite(led1,0);

digitalWrite(led2,1);

Serial.print("data==B");

}

if(data=='C'){

digitalWrite(led1,1);

digitalWrite(led2,0);

Serial.print("data==C");

}

delay(100);

}

}

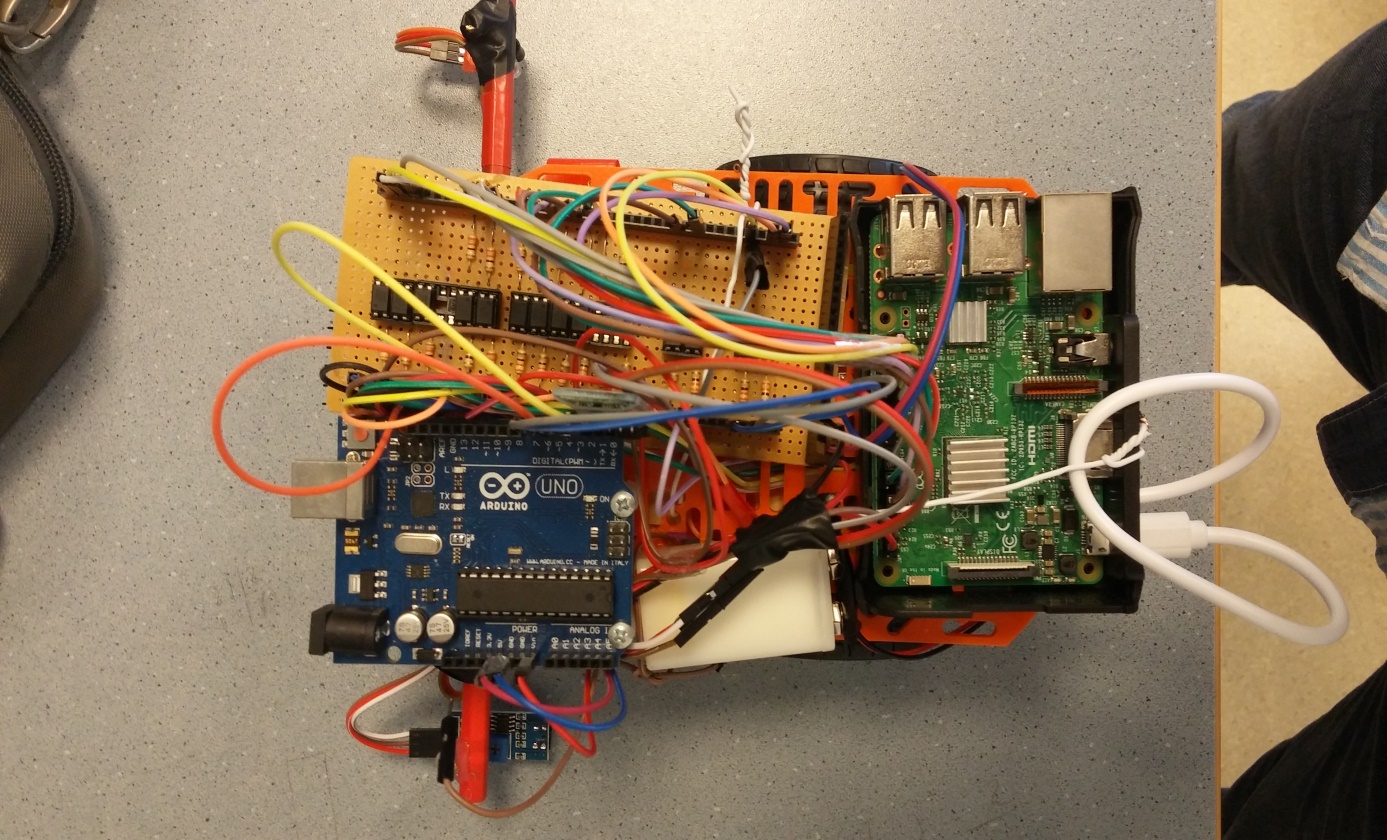
**Nasıl Kullanılır**

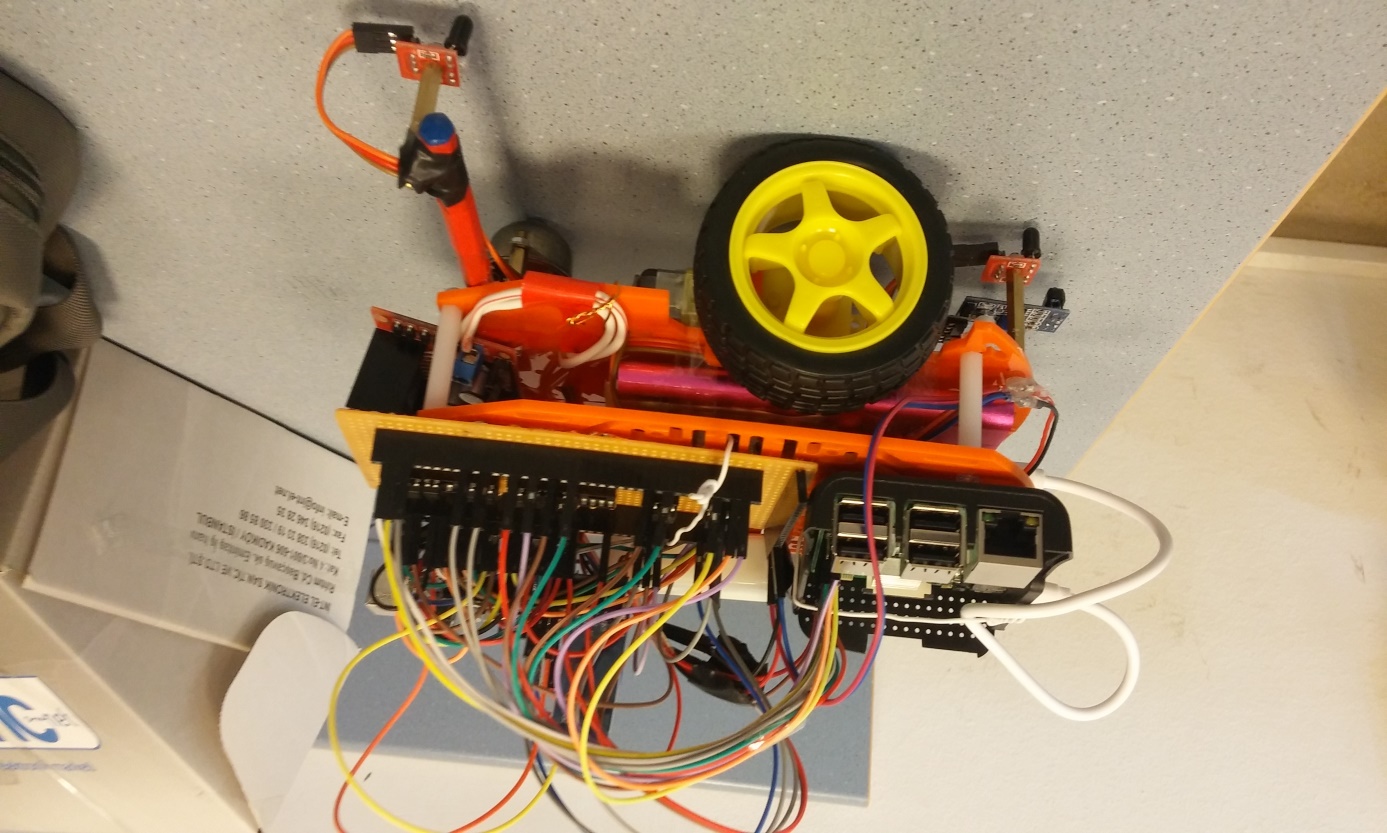
Gerekli bağlantılar yapıldıktan sonra bash ekranına aşağıdaki kod yazılır:

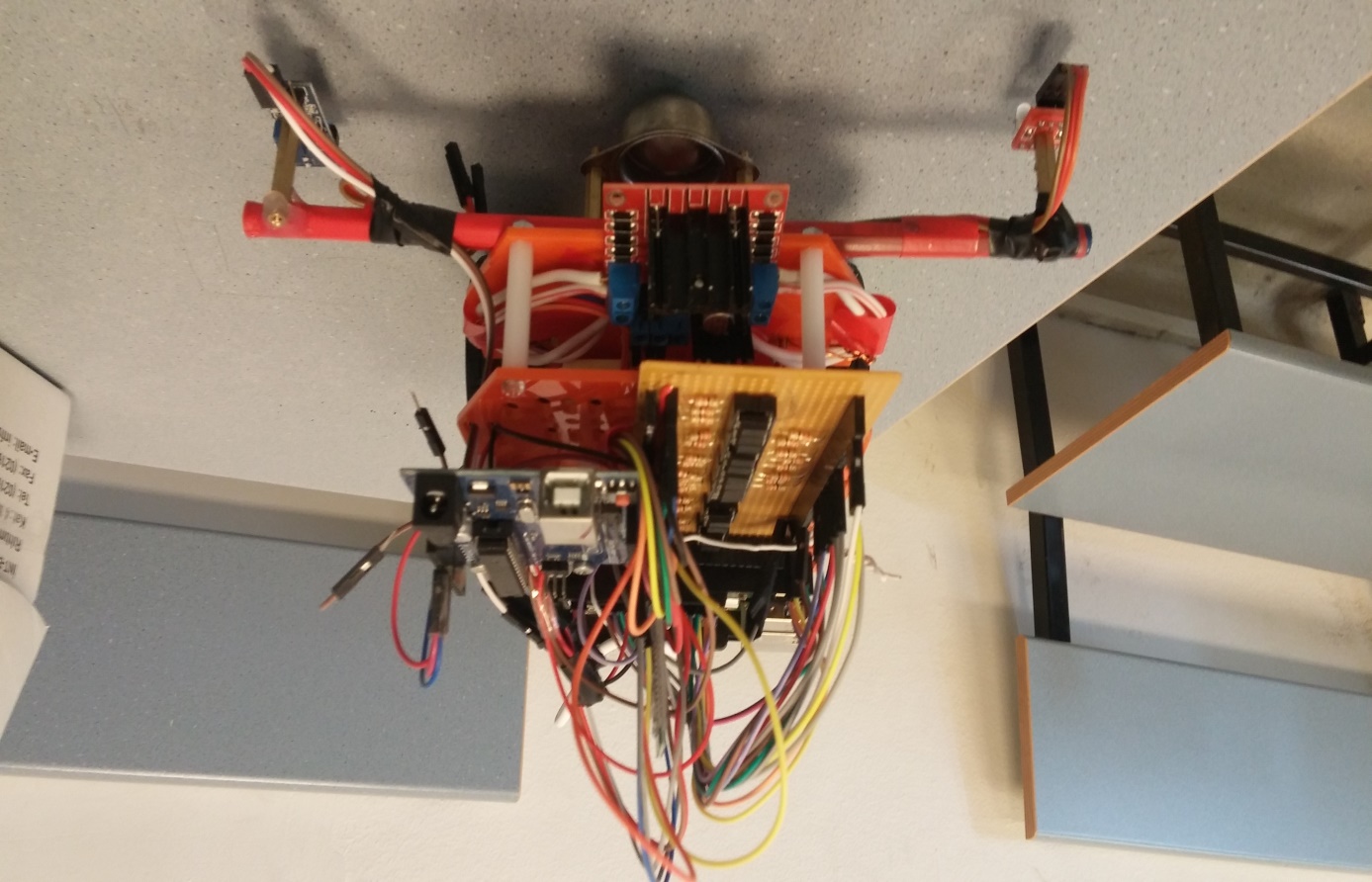
Python proje3.py

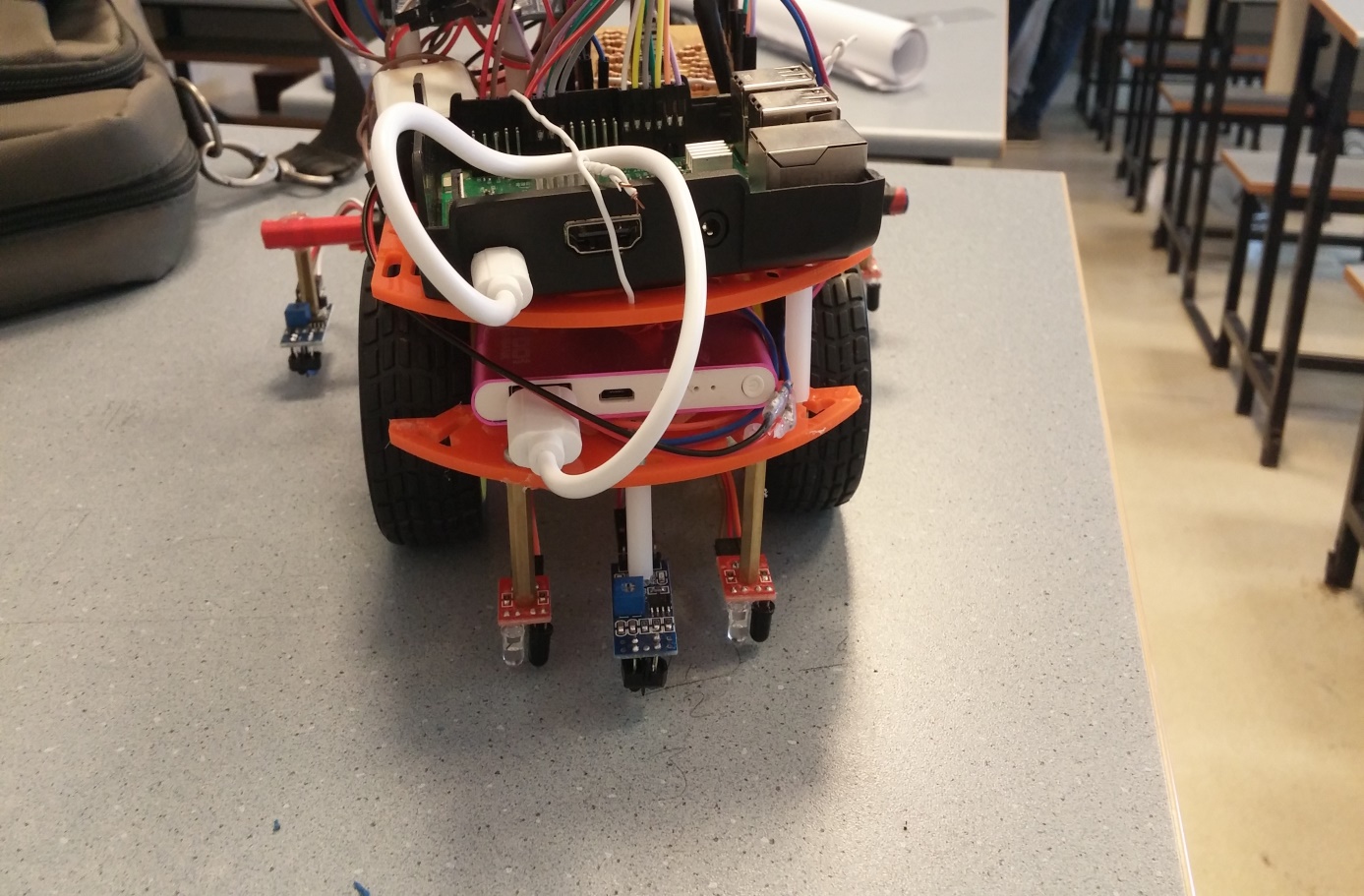
Daha sonra telefondaki uygulama ile araç üzerinde bluetooth cihazına bağlantı sağlanır. Bluetooth üzerinden arabanın gideceği konum seçilir. Aynı seçim telefondaki konum uygulamasında da gerçekleştirilir. Araç istenilen yere ulaştığı zaman konum uygulamasından işlem tamamlandı tuşuna basılarak sistemin çalışması tamamlanmış olur.

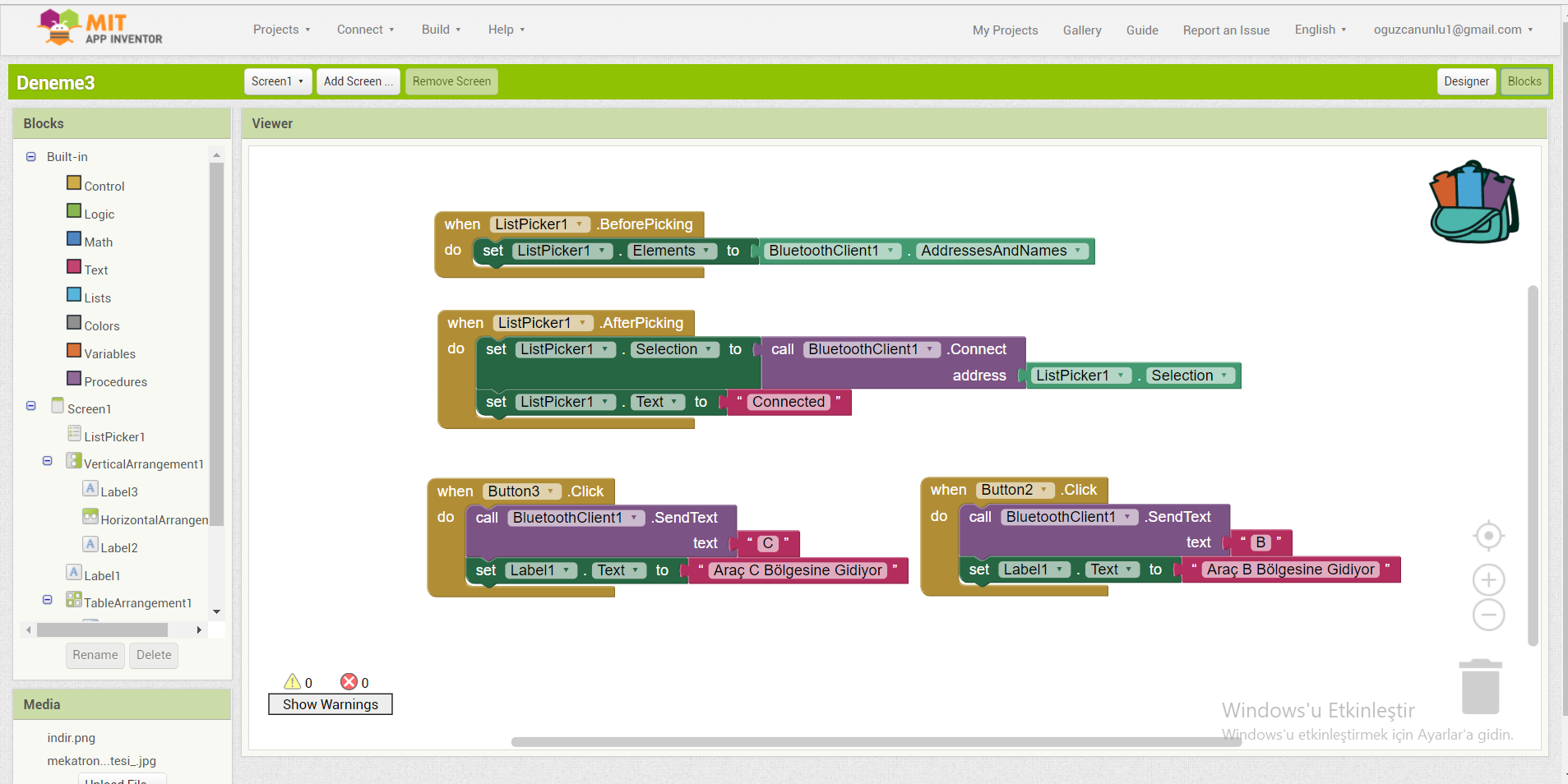
**Proje Resimleri**

****

****

****

****

****

****

**Öneriler**

Yaptığımız bu projede bluetooth yerine wifi ile kontrol sağlanabilir. Bu sayede daha uzun mesafelerde de iletişim sağlanabilir. Araç tasarımında değişiklikler yapılabilir.

**Kaynak**

https://gpiozero.readthedocs.io/en/stable/index.html