

**T.C.**

**MİLLİ EĞİTİM BAKANLIĞI**

**SİVAS MERKEZ**

**Sivas Fen Lisesi**

**Bilgisyar Bilimi Dersi**

**Rapor**

|  |  |
| --- | --- |
| **Rapor No** | Proje-1 |
| **Rapor Tarih** | 09.01.2018 |
| **Proje Adı** | Arduino ile Çizgi Takip Eden Robot |

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 | 101 | Barlas Coşkun |
| Doküman Yönetimi | 455  290 | Emine Zeynep Yıldız  Hande Yel |
| Lojistik Yönetim | 172 | Okmen Mahir Karakaş |
| Yazılım Geliştirme | 312 | Fatih Çağrı Yüksel |
| Web ve GitHub Yönetimi | 56 | Umut Yıldızdan |
| Sunum Yönetimi | 455 | Emine Zeynep Yıldız |

İÇİNDEKİLER

[ÖZET 3](#_Toc477023339)

[Anahtar Kelimeler 3](#_Toc477023340)

[ABSTRACT 3](#_Toc477023341)

[Key Words 3](#_Toc477023342)

[Proje Görev Dağılımı Listesi ve Görev Dağılımı Açıklaması 4](#_Toc477023343)

[ Görev Dağılımı ve Sorumlusu 4](#_Toc477023344)

[ Görev süresince sürdürülen eylemler 5](#_Toc477023345)

[ Görevlerin iş yükü şeması 6](file:///C:\Users\HİLAL\Desktop\düzenlendi.docx#_Toc477023346)

[ Yoklama Çizelgeleri 7](#_Toc477023347)

[ Haftalık İş Katkı Cetvelleri 8](#_Toc477023348)

[GİRİŞ 8](#_Toc477023349)

[1.Projenin Açıklaması 8](#_Toc477023350)

[2.UML Diyagramlar 8](#_Toc477023351)

[3.Donanım Yapısı: 9](#_Toc477023352)

[a.Gömülü Sistemler Mimarisi ve Devre Tasarımı 9](#_Toc477023353)

[b.Mekanik Sistem Mimarisinin Tanıtılması: 12](#_Toc477023354)

[PID 17](#_Toc477023355)

[4.Yazılım Yapısı 18](#_Toc477023356)

[SONUÇ 19](#_Toc477023357)

[1.Bilgi Düzeyine Katkıları: 19](#_Toc477023358)

[2.Teknolojik Katkıları: 19](#_Toc477023359)

[3.Ekip Çalışması Katkıları 19](#_Toc477023360)

[4.Aksayan Yönler: 20](#_Toc477023361)

[5.Görüş ve Öneriler: 20](#_Toc477023362)

## ÖZET

Birinci proje kapsamında kullanılan elemanlar ile çizgi takip eden robot yapılacaktır. Bu projeden belirlenmiş olan çizgiyi sorunsuz ve istenen bir şekilde izleyerek pisti tamamlaması beklenmektedir. Bu amaçla kullanılacak olan QTR8A çizgi sensörünün siyah ve beyaz gibi renk ayrımı yapabilmesi sayesinde gerekli kodlamalar ile robot istenilen siyah çizgi üzerinde pisti tamamlayacaktır. Proje gerçekleştirilirken devre tasarımı, devre elemenlarının lehimlenmesi, test ve deneme aşamaları, kodlama, sunum gibi aşamalar gerçekleştirilecektir.

## Anahtar Kelimeler

Arduino,Çizgi İzleyen Robot,QTR8A,Çizgi Sensörü,

## ABSTRACT

The elements used in the first project and the robot following the line will be made. It is expected to complete the runway by following the line that is determined from this project without any problems. The QTR8A line sensor to be used for this purpose can distinguish between black and white color, so that the robot will complete the run on the desired black line with the necessary coding. During the project, stages such as circuit design, soldering of circuit elements, testing and testing stages, coding, presentation will be realized.

## Key Words

Arduino, Line Follower Robot, QTR8A,Line Follower Sensor

## 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. | Barlas Coşkun  Hande Yel |
| 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. | Emine Zeynep Yıldız  Umut Yıldızdan |
| 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. | Ökmen Mahir Karakaş |
| 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. | Fatih Çağrı 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. | Umut Yıldızhan |
| 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. | Emine Zeynep Yıldız |

## 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) |

## Görevlerin iş yükü şeması

SUNUM

## Yoklama Çizelgeleri

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Grup**  **Tarihler** | **05.12.2017** | **12.12.2017** | **19.12.2017** | **26.12.2017** | **02.01.2018** | **09.01.2018** |
| **Ad –Soyad**  **Emine Zeynep Yıldız** |  |  |  |  |  |  |
| **Umut Yıldızdan** |  |  |  |  |  |  |
| **Barlas Coşkun** |  |  |  |  |  |  |
| **Fatih Çağrı Yüksel** |  |  |  |  |  |  |
| **Hande Yel** |  |  |  |  |  |  |

## Haftalık İş Katkı Cetvelleri

|  |  |
| --- | --- |
|  | **Yapılan iş** |
| **05.12.2017** | Proje araştırması |
| **12.12.2017** | Malzeme Teminatı |
| **19.12.2017** | Elektronik ve Mekanik Tasarım |
| **26.12.2017** | Yazılım |
| **02.01.2018** | Deneme ve Test Aşamaları |

# 

# GİRİŞ

## 1.Projenin Açıklaması

Yapılan projede amaç; hazırlanan pistte, çizgiyi takip edebilecek bir çizgi izleyen robot tasarımı gerçekleştirebilmektir. Robot yolu öz denetimli olarak takip etmelidir.Robotun çalışma prensibi;QTR-8A çizgi sensörü ile döndürülen değerler şart yapıları ile hesaplamaları sayesinde robotun konumunu güncelleyerek pisti tamamlamaktır.Robot ilerlerken hedeflenen çizgi robotun sağ tarafında kalırsa motorlar gerekli komutlarla sağ tarafa,sol tarafında kalırsa sol tarafa yönecelecektir.Dönme işlemi de hedeflenen çizgiye gelene kadar devam edecektir.Çizgi üzerindeyken robot düz devam edecektir.

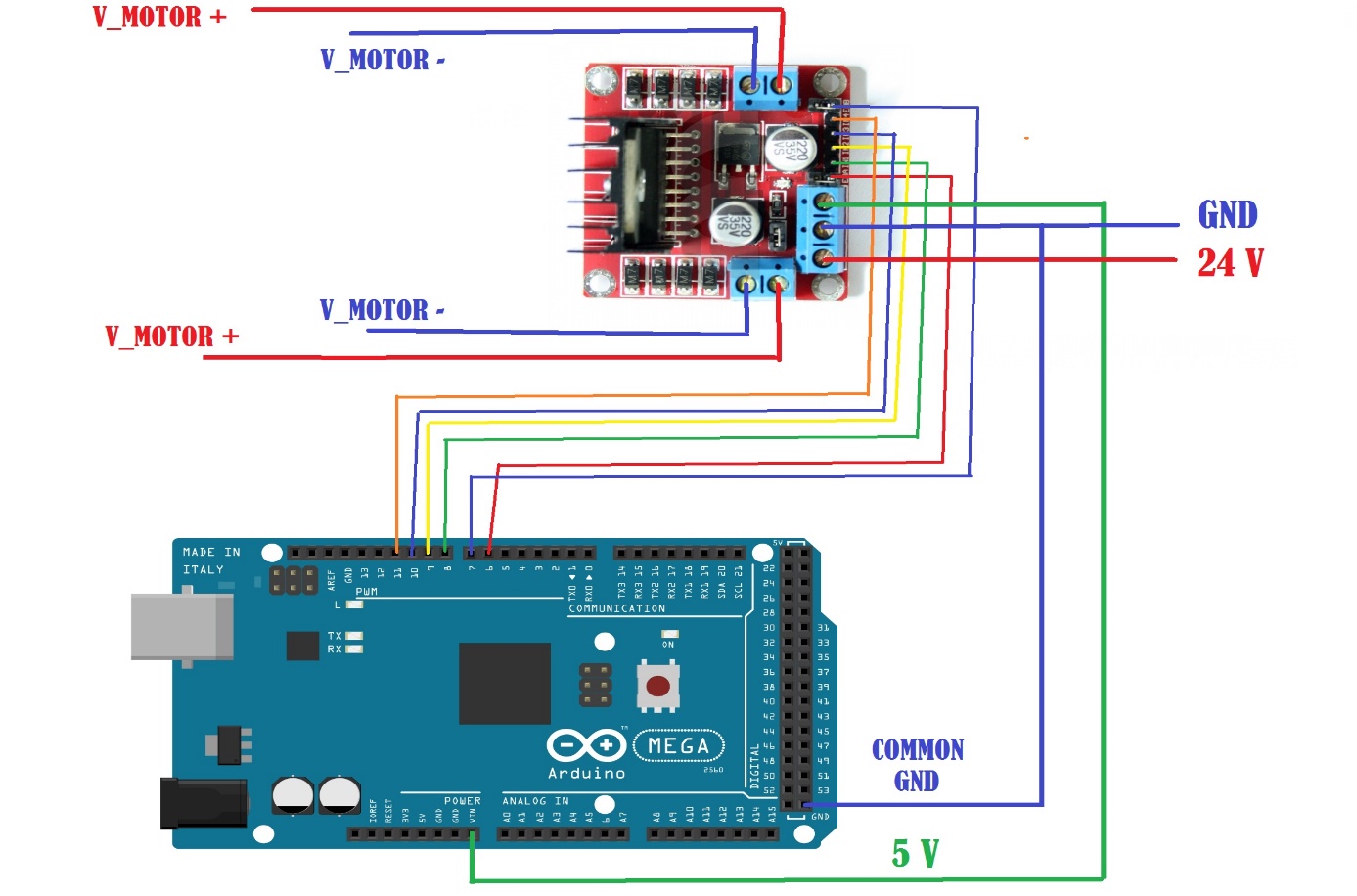
## 3.Donanım Yapısı:

### a.Gömülü Sistemler Mimarisi ve Devre Tasarımı

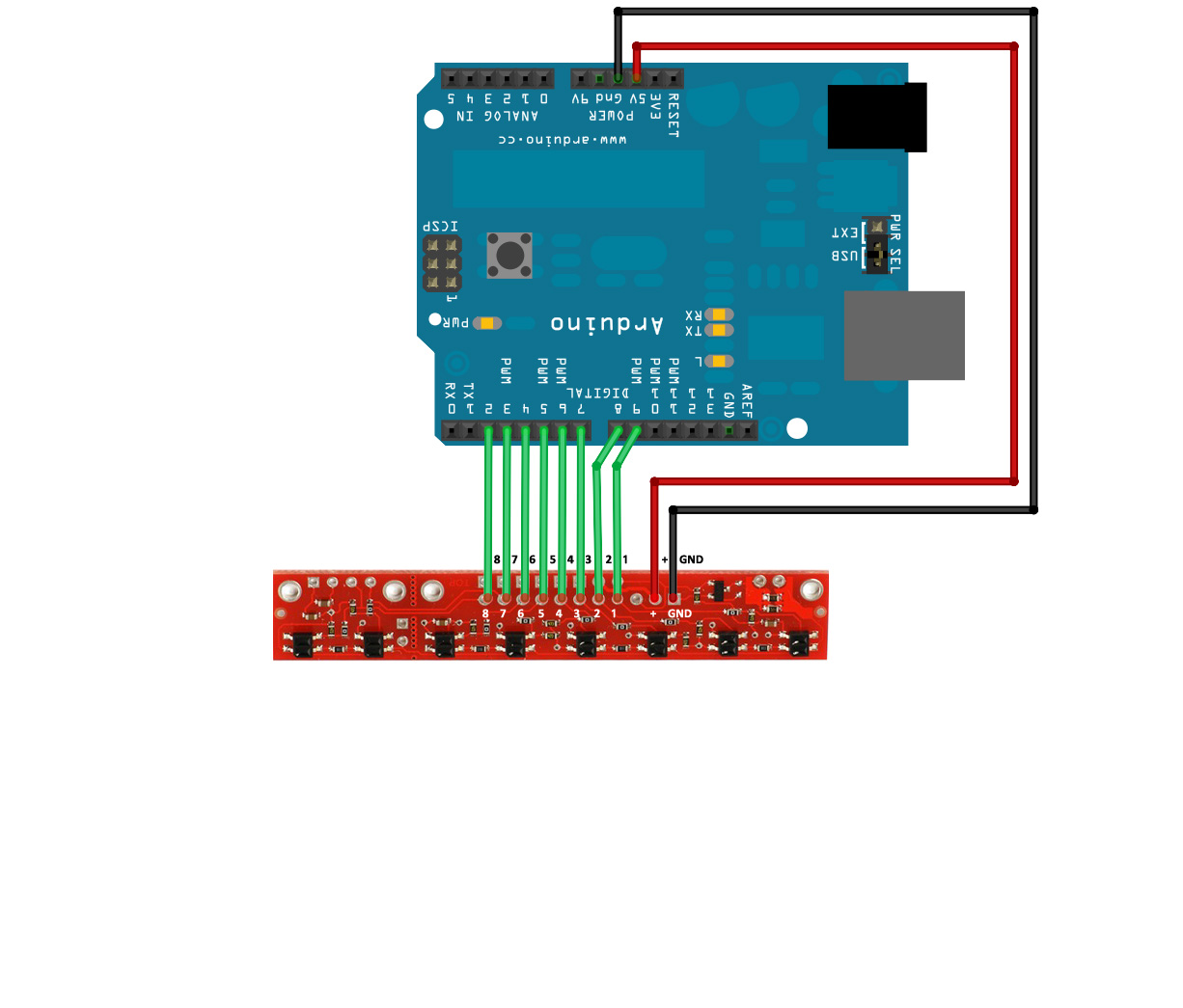
|  |  |
| --- | --- |
|  | **ARDUINO UNO**  Bunlar projelerde çokça kullanılan ve genelde Arduino Uno R3 pinlerinin yetmedi dediğimiz RX/TX, SCL/SDA, 3.3V/5V ve GND pinleri olur. Konum olarakta güzel bir yerde bulunan bu çıkışları kendiniz ister erkek, ister dişi header lehimleyerek kullanabilirsiniz.   En yeni Arduino klonlarında, standart UNO'daki Atmega 16U2 çipi veya FTDI çipi yerine USB CH340 Serial Dönüştürücüsü kullanılmıştır. |
|  | **L298N Voltaj Regulatörlü Çift Motor Sürücü Kartı**  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 imkan sağlamaktadır. |
|  | **QTR-8A Çizgi Sensörü**  Çizgi izleyen robotlar için yapılmış olan bu sensör kartı 1cm arayla yerleştirilmiş 8 IR LED/fototransistör çifti barındırmaktadır.LED çiftlerinin her biri ayrı birer MOSFET transistörle sürülmektedir ve ek hassasiyet ya da güç tasarrufu için LED’ler kapatılabilir.Kart üzerindeki her sensör ayrı bir analog voltaj çıkışı sağlar.Her bir sensöre bir pull-up direnci bağlanmıştır. |
|  | **Jumper Kablo (E-E/D-E)**  Devre elemanlarının bağlantılarını gerçekleşirmek için kullanabiliriz. |
|  | **50x11mm Mavi Renk Geçmeli Tekerlek Seti**  Bu tekerlek seti 3mm, 4mm D tipi şaftlı motorlar ve L tipi plastik redüktörlü motorlara uyumlu şaft adaptörleri ile birlikte gelmektedir. Bu sayede bir çok motor ile birlikte kullanma imkanı sunmaktadır. |
|  | **6V 350 RPM Redüktörlü DC Motor**  DC 6V’ta 350Rpm hıza sahip mikro redüktörlü motordur. Yüksek torka ihtiyaç duyduğunuz birçok projenizde kullanabileceğiniz motorlardandır. Mini sumo robotlar için ideal, engelden kaçan, ışık takip eden ve çizgi izleyen robot gnu projelerde kullanabilirsiniz. |
|  | **Sarhoş Tekerlek**  Plastik bilyalı bu küçük sarhoş tekerin bilya çapı 12,7 mm'dir. Birleştirilmiş halde kit uzunluğu 12,7 ile 20,32 mm arasında değişmektedir.  Bu sarhoş teker seti; top yuvası, 12,7 mm çapında plastik bilyalı, 2 adet vida seti ve 2 adet yükselticiden oluşur |

#### Devre Tasarımı:

**L298N bağlantısı:**

****

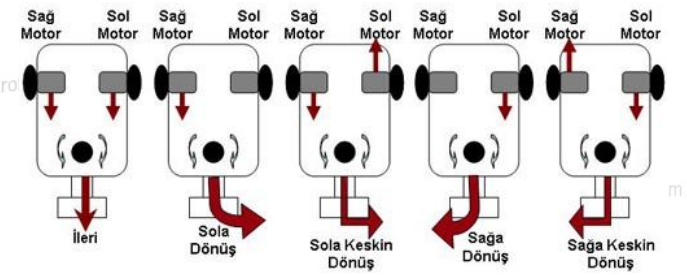
**QTR8-A bağlantısı:**

****

**Siyah Kablo:GND , Kırmızı Kablo: 5V**

### b.Mekanik Sistem Mimarisinin Tanıtılması:

Robotun dönüşü birbirinden bağımsız çalışan iki motorun arasındaki hız farkıyla sağlanır. Aşağıdaki şemada bu sürüş sisteminin çalışılması gösterilmiştir.

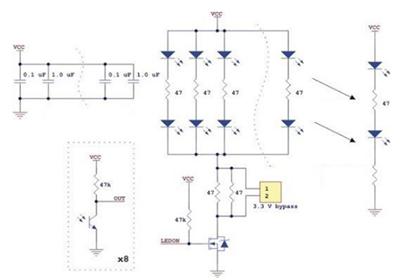


##### 

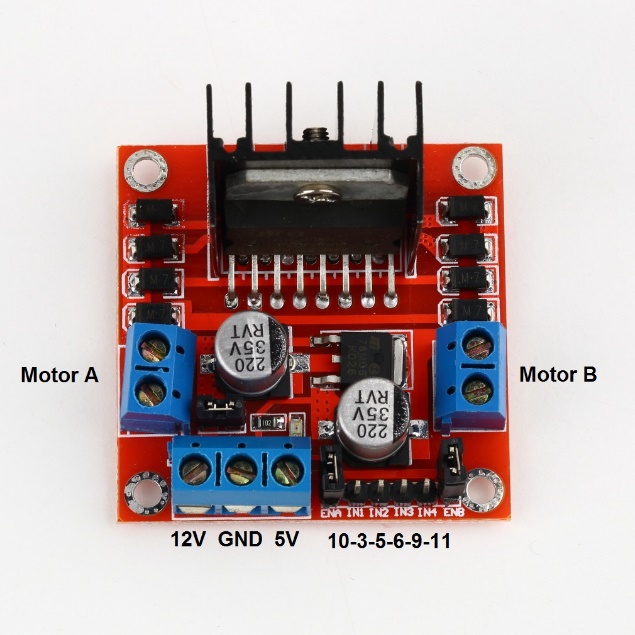
Burada bulunan 8 tane çizgi sensörü pin’i arduınonun analog bacaklarına bağlıyoruz.Vcc ve led on pin’ini +5V’a Gnd pin’inide  toprağa bağlıyoruz.QTR-8A Sensör kartı bir kaç farklı şekilde çalıştırılabilir. Bir mikrodenetleyicinin analog giriş pinlerine bağlanarak ADC(Analog Dijital Çevirici) işlemine tabi tutulabilir. Eşik değeri ayarlanabilir bir karşılaştırıcı kullanılarak gelen analog voltajı dijitale çevirerek işlemler yaptırılabilir. Her çıkışı mikrodenetleyicinin I/O pinlerine bağlanarak mikrodenetleyici içindeki karşılaştırıcı kullanılarak okuma yapılabilir. (Bu yöntemle yüksek yansıma olan ortamlarda daha iyi sonuç alınır.).QTR-8A sensörü 8 sensörün tamamını kullanmak istemeyen kullanıcılara da imkan sağlamaktadır.Ayrıca sensör yakınlık ve cisim algılama sensörü olarak da kullanılabilmektedir.Çalışma voltajı 3.3 volt ile 5 votl arasındadır.Çektiği maksimum akım 100mAh tir.İdeal algılama mesafesi 3mm dir.Bu uzaklıktan fazla olduğunda kararlı çalışmaz.Sorunlar çıkartabilir.

##### 1.QTR-8A Çizgi Sensörü

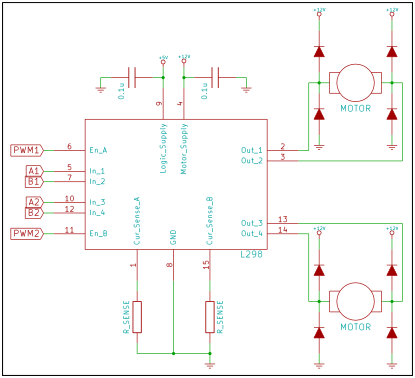
**QTR sensör iç yapısı:**

****

##### 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.

****

##### 3. Orjinal Arduino UNO R3

Teknik Özellikler:

- Mikrodenetleyici: ATmega328

- Çalışma Gerilimi : 5V

- Giriş Gerilimi (önerilen): 7-12V

- Giriş Gerilimi (limit): 6-20V

- Dijital I/O Pinleri: 14 (6 tanesi PWM çıkışı)

- Analog Giriş Pinleri: 6

- Her I/O için Akım : 40 mA

- 3.3V Çıkış için Akım: 50 mA

- Flash Hafıza: 32 KB (ATmega328) 0.5 KB kadarı bootloader

- SRAM: 2 KB (ATmega328)

- EEPROM: 1 KB (ATmega328)

- Saat Hızı : 16 MHz

- Uzunluk: 68.6 mm

- Genişlik: 53.4 mm

- Ağırlık: 25 g

Arduino Uno; Atmega328 temelli bir mikrodenetleyici 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 mikrodenetleyicinin çalışması için gerekli olan her şey bulunmaktadır. Kolayca usb kablosu üzerinden bilgisayara bağlanabilir, adaptör veya pil ile çalıştırılabilir.

Güç pinleri aşağıdaki gibidir:

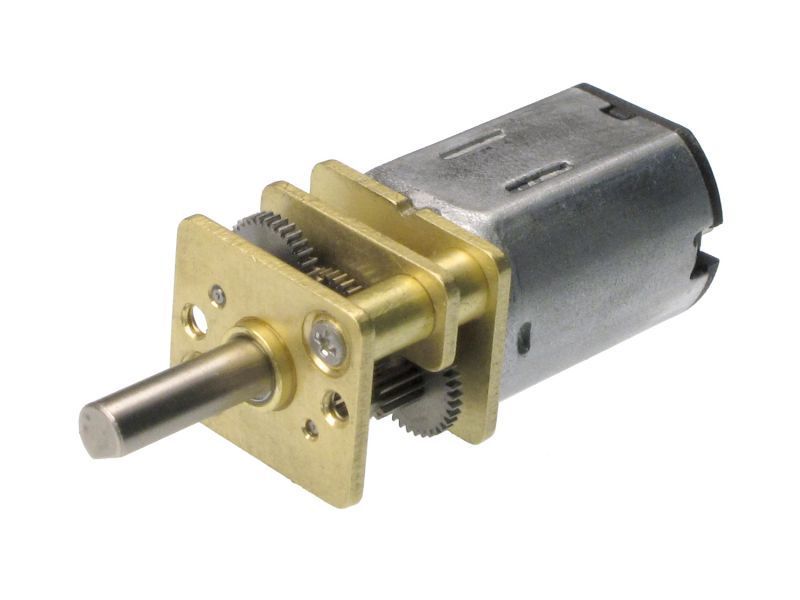
VIN: Harici güç kaynağı kullanılırken 7-12V arası gerilim giriş pini.

5V: Bu pin 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.

3V3: Kart üzerinde bulunan 3.3V regülatörü çıkış pinidir. Maks. 50mA çıkış verebilir.

GND: Toprak pinleridir.

##### 4.Force Up 6V 600RPM DC Motor

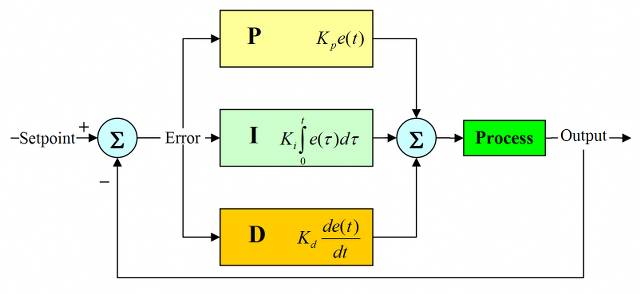
****

**Şekil 8.DC motor**

**Özellikler:**

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

## PID

****

**Şekil 9: PID kontrol Diyagramı**

PID,Oransal İntegral Türev için kullanılan bir kısaltmadır.En genel tanımıyla bir kontrol geri bildirim mekanizmasıdır.PID yönteminin en temel amacı hatayı minimize etmek,en aza indirmektir.PID kontrolünü uygulamak ve kavramak oldukça zordur. PID kontrolünde öncelikle hata tanımlaması yapılmalıdır.Hata ise referans değere olan uzaklık olarak tanımlanabilir.Açıklamak gerekirse;

Referans=İstenilen değer

Gelen=Şuan ki Konum

HATA=Referans-Gelen

**Oransal Terim(P):**

Oransal terim, sistemden gelen hatayı bir katsayı ile çarparak hatayı küçültmeyi hedefler. Bozucu etkileri de mevcuttur.Projemizin yazılım kısmında oldukça ağırlık verdiğimiz PID kontrolünde deneme ve test aşamasında bu katsayıya büyük bir değer vermemiz gerektiğini anladık.

P=Kp\*HATA

**Integral Terimi(I):**

Integral hatanın alanını bulmak anlamına gelir. Integralin çok yükselmesini önlemek için sınırlandırmak gereklidir. Sürekli toplandığı için integral çok artarsa tekrar azalmasını beklemek zaman alır. Bu yüzden integrali sınırlamak sistemin çabuk toparlamasını sağlayacaktır.

I = I + (Ki \* HATA \* dt)

dt: PID fonksiyonuna her girdiğinde geçen zaman.

**Türev Terimi(D):**

Türev sistemdeki iki örnek arasındaki zamanı hesaplar.Eğer hatada bir değişim olmadıysa türev sıfır olur.

EHata=Bir önceki hatanın değeri

HD=HATA-EHata

D=(Kd\*HD)/dt

**PID Algoritması**

Kp , Ki ve Kd katsayılarından oluşur.Bu katsayılar deneme yanılma yöntemiyle bulunur. Yapacağınız sistemde optimum katsayıları bulmak için değerde değişiliklik yapıp sistemi gözlemlemeniz gerekmektedir.

HATA = Referans - Gelen

HD = HATA - EHata

P = Kp \* Hata

I = I + (Ki \* HATA\* dt)

D = (Kd \* HD)/dt

PID = P + I + D

EHata = Hata

### 

### 4.Yazılım Yapısı

a.Algoritmik olarak: Algoritmik olarak:

Robotun çalışma prensibi, çizgiyi ortadaki algılayıcıda tutarak ilerlemesine dayanmaktadır. Robot kısa aralıklarla algılayıcının durumunu kontrol ederek çizgiye göre konumunu belirler. Bunun için tasarım fikrine bağlı olarak belli sayıda ve belli aralıklarla ve yerleşim şekli ile çizgi algılama sensörleri kullanılır ve motorlar buradan gelen bilgilere göre kontrol edilir. Robot ilerlerken, çizgi robotun sağ tarafına gelirse, 3 nolu algılayıcı çizgiyi görecektir. Böylece robot çizginin sol tarafında olduğunu anlayacak ve programı içinde belirtilen sağa dönüş komutunu uygulayacaktır. Dönme işlemi, ortadaki algılayıcının çizgiyi tekrar görmesine kadar devam eder. Bu algılayıcı çizgiyi gördüğünde, robot tekrar düz hareket etmeye başlar. Eğer, çizgi robotun sol tarafına gelirse, aynı işlemler bu taraf için tekrarlanır ve böylelikle çizgi izleme işlemi yerine getirilir.

b:Kod yapısı: b:Kod yapısı:

#include <QTRSensors.h>

#define SolMotorileri 9

#define SolMotorGeri 10

#define SagMotorileri 11

#define SagMotorGeri 12

#define ena 3

#define enb 5

#define KP .2

#define KD 5

#define M1\_minumum\_hiz 80

#define M2\_minumum\_hiz 80

#define M1\_maksimum\_hiz 85

#define M2\_maksimum\_hiz 85

#define MIDDLE\_SENSOR 3

#define NUM\_SENSORS 5

#define TIMEOUT 2500

#define EMITTER\_PIN 2

#define DEBUG 0

int lastError = 0;

int last\_proportional = 0;

int integral = 0;

QTRSensorsRC qtrrc((unsigned char[]) { A2, A3, A4, A5} ,NUM\_SENSORS, TIMEOUT, EMITTER\_PIN);

unsigned int sensorValues[NUM\_SENSORS];

void setup() {

pinMode(SolMotorileri,OUTPUT);

pinMode(SolMotorGeri,OUTPUT);

pinMode(SagMotorileri,OUTPUT);

pinMode(SagMotorGeri,OUTPUT);

pinMode(ena,OUTPUT);

pinMode(enb,OUTPUT);

delay(1500);

manual\_calibration();

set\_motors(0,0);

}

void loop() {

unsigned int sensors[5];

int position = qtrrc.readLine(sensors);

int error = position - 3000;

int motorSpeed = KP \* error + KD \* (error - lastError);

lastError = error;

int leftMotorSpeed = M1\_minumum\_hiz + motorSpeed;

int rightMotorSpeed = M2\_minumum\_hiz - motorSpeed;

set\_motors(leftMotorSpeed, rightMotorSpeed);

}

void set\_motors(int motor1speed, int motor2speed)

{

if (motor1speed > M1\_maksimum\_hiz ) motor1speed = M1\_maksimum\_hiz; //MAKSİMUM MOTOR 1 HIZ LİMİTİ

if (motor2speed > M2\_maksimum\_hiz ) motor2speed = M2\_maksimum\_hiz; // MAKSİMUM MOTOR 2 HIZ LİMİTİ

if (motor1speed < 0) motor1speed = 30; // MİNIMUMMOTOER 1 HIZ LİMİTİ

if (motor2speed < 0) motor2speed = 30; // MİNİMUM MOTOR 2 HIZ LİMİTİ

analogWrite(ena,motor1speed);

analogWrite(enb,motor2speed);

digitalWrite(SolMotorileri,HIGH);

digitalWrite(SagMotorileri,HIGH);

digitalWrite(SolMotorGeri,LOW);

digitalWrite(SagMotorGeri,LOW);

}

void manual\_calibration() {

int i;

for (i = 0; i < 250; i++)

{

qtrrc.calibrate(QTR\_EMITTERS\_ON);

delay(20);

}

if (DEBUG) {

Serial.begin(9600);

for (int i = 0; i < NUM\_SENSORS; i++)

{

Serial.print(qtrrc.calibratedMinimumOn[i]);

Serial.print(' ');

}

Serial.println();

for (int i = 0; i < NUM\_SENSORS; i++)

{

Serial.print(qtrrc.calibratedMaximumOn[i]);

Serial.print(' ');

}

Serial.println();

Serial.println();

}

}

# 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 farkettik.

PID kontrolü konusuna yoğunlaştık. Hata oranını minimize ederek, robotun sorunsuz bir şekilde çizgiyi tam anlamıyla takip etmesi için araştırmalar yapıp Kp, Kd ve Ki değerlerine deneme ve yanılma yöntemiyle uygun katsayıları bulmaya çalıştık.Bizi en zorlayan kısım PID kontrolü ve robotun çizgi takip işlemini gerçekleştirirken hız ve açısal konumunun ayarlanması oldu. Test aşamasında gördüğümüz hatalarımızı düzelttik. 1.Proje sayesinde devre tasarımında devre elemanlarının zarar görmemesi için yapılması gerekenleri devre tasarımının nasıl olması gerektiğini ve bağlantıları öğrendik.

Projeyi sunma ve raporlamada neler yapılması gerektiğini öğrendik.Altı hafta boyunca kullandığımız devre elemanları ile ilgili bilgi birikimimizi genişlettik.

En zorlandığımız yer ise kodlama bölümüydü. Fakat biraz mantık kurarak birazda araştırarak bununda üstesinden geldik.

## 2.Teknolojik Katkıları:

Teknolojinin gelişmesiyle birlikte bu robotlar birçok sektörde hayata girmiş durumda özellikle lojistik bölümünde bunlara çok ihtiyaç duyuluyor.Bu da bize hem iş imkanı hemde tecrübe olarak dönüyor.

3.Ekip Çalışması Katkıları**:**

Ekip çalışmasının bize kattıkları:

• Başarıya ulaşma surecimizi hızlandırdı.Problemlerimiz daha hızlı çözüldü,zamanımızı daha verimli kullandık.Görev dağıtımı sayesinde çeşitlenen fikirler ve gelişen bakış açılarıyla sonuca daha çabuk ulaştık.

• Birbirimizle olan ilişkilerimizi güçlendirdik.Bunun yanı sıra yapmamız gerekenleri ve takımımızın bizden beklentilerini karşılamak için çalışmalarımızı bir düzen içinde yürüttük.

• Farklı fikirlere olan bakış açımız olumlu yönde etkilendi.

## 4.Aksayan Yönler:

Projeyi gerçekleştirirken karşılaştığımız sorunlar:

• Kullandığımız pillerden ötürü motorumuz yanmıştı.

• Raporda fazla bir araya gelemedigimiz için kayıt altına almada aksaklıklar çıktı.

## 5.Görüş ve Öneriler:

Bilgi haznemizi gelistirmemize katkıda bulunmustur.Aynı zamanda edindiğimiz bu bilgileler sayesinde ileride yapacak olduğumuz projeler hakkında bir tecrübe edindik.Yapmış olduğumuz robotuda geliştirilmeye açık bir hale getirdik.