

**T.C.**

**MİLLİ EĞİTİM BAKANLIĞI**

**SİVAS MERKEZ**

**Sivas Fen Lisesi**

**Bilgisayar Bilimi Dersi**

**Rapor**

|  |  |
| --- | --- |
| **Rapor No** | Proje-1 |
| **Rapor Tarih** | 06.01.2018 |
| **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 | 140 | Muhammed Bera KOÇ |
| Doküman Yönetimi | 355,38 | Mevlüde AVCI,Gizem BULUT |
| Lojistik Yönetim | 140 | Muhammed Bera Koç |
| Yazılım Geliştirme | 325 | Osman Sefa KILIÇ |
| Web ve GitHub Yönetimi | 38 | Gizem BULUT |
| Sunum Yönetimi | 236,181 | Şevval CEYLAN,Asya Dila SUBAŞI |

İÇİNDEKİLER

[ÖZET 4](#_Toc503031036)

[Anahtar Kelimeler 4](#_Toc503031037)

[ABSTRACT 4](#_Toc503031038)

[Key Words 4](#_Toc503031039)

[Proje Görev Dağılımı Listesi ve Görev Dağılımı Açıklaması 5](#_Toc503031040)

[ Görev Dağılımı ve Sorumlusu 5](#_Toc503031041)

[ Görev süresince sürdürülen eylemler 6](#_Toc503031042)

[ Görevlerin iş yükü şeması 7](file:///C:\Users\Pronet\Desktop\Gizem\bilişm\bilisim%20%20(Onarıldı).docx#_Toc503031043)

[ Yoklama Çizelgeleri 8](#_Toc503031044)

[ Haftalık İş Katkı Cetvelleri 9](#_Toc503031045)

[GİRİŞ 9](#_Toc503031046)

[1.Projenin Açıklaması 9](#_Toc503031047)

[3.Donanım Yapısı: 10](#_Toc503031048)

[a.Gömülü Sistemler Mimarisi ve Devre Tasarımı 10](#_Toc503031049)

[b.Mekanik Sistem Mimarisinin Tanıtılması: 13](#_Toc503031050)

[PID 19](#_Toc503031051)

[4.Yazılım Yapısı 21](#_Toc503031052)

[SONUÇ 25](#_Toc503031053)

[1.Bilgi Düzeyine Katkıları: 25](#_Toc503031054)

[2.Teknolojik Katkıları: 25](#_Toc503031056)

[3.Ekip Çalışması Katkıları 25](#_Toc503031057)

[4.Aksayan Yönler: 26](#_Toc503031058)

[5.Görüş ve Öneriler: 26](#_Toc503031059)

## 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. | Muhammed Bera Koç |
| 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. | Mevlüde AVCI  Gizem BULUT |
| 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. | Muhammed Bera Koç |
| 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. | Osman Sefa  KILIÇ |
| GitHub Yönetimi | Projenin GitHub yönetiminin yapılması,döküman yöneticisinden almış olduğu raporlar ile GitHub'a işlemekle sorumlu olan kişidir. | Gizem BULUT |
| 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. | Asya Dila SUBAŞI  Şevval CEYLAN |

## Görev süresince sürdürülen eylemler

|  |  |
| --- | --- |
| 1.Hafta | Proje Hakkında Bilgi Edinme  Görev Dağılımını Yapma |
| 2.Hafta | Malzeme Seçimi ve malzemelerin temin edilişi |
| 3.Hafta | Mekanik ve Elektronik Tasarım |
| 4.Hafta | Yazılım |
| 5.Hafta | Test aşamaları ve ortaya çıkan sorunların çözümü |
| 6.Hafta | Grup elemanlarına ait iş yükünün tamamlanması (rapor,,github) |

## Görevlerin iş yükü şeması

SUNUM

## Yoklama Çizelgeleri

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Tarihler**  **Grup** | **28.11.2017** | **05.12.2017** | **12.12.2017** | **19.12.2017** | **26.12.2017** | **02.01.2018** |
| **Bera KOÇ** | **+** | **+** | **+** | **+** | **+** | **+** |
| **Asya Dila SUBAŞI** | **+** | **+** | **+** | **+** | **+** | **+** |
| **Gizem BULUT** | **+** | **+** | **+** | **+** | **+** | **+** |
| **Şevval CEYLAN** | **+** | **+** | **+** | **+** | **+** | **+** |
| **Mevlüde AVCI** | **+** | **+** | **+** | **+** | **+** | **-** |
| **Osman Sefa**  **KILIÇ** | **+** | **+** | **+** | **+** | **+** | **+** |

## Haftalık İş Katkı Cetvelleri

|  |  |
| --- | --- |
|  | **Yapılan iş** |
| **28.112017** | Proje araştırması |
| **05.12.2017** | Malzeme Teminatı |
| **12.12.2017** | Elektronik ve Mekanik Tasarım |
| **19.12.2017** | Yazılım |
| **26.12.2017** | Deneme ve Test Aşamaları. Çıkan Sorunların düzeltilmesi |
| **02.01.2018** | Rapor,Github düzenlenmesi.Sunuma hazırlık yapımı. |

# 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.

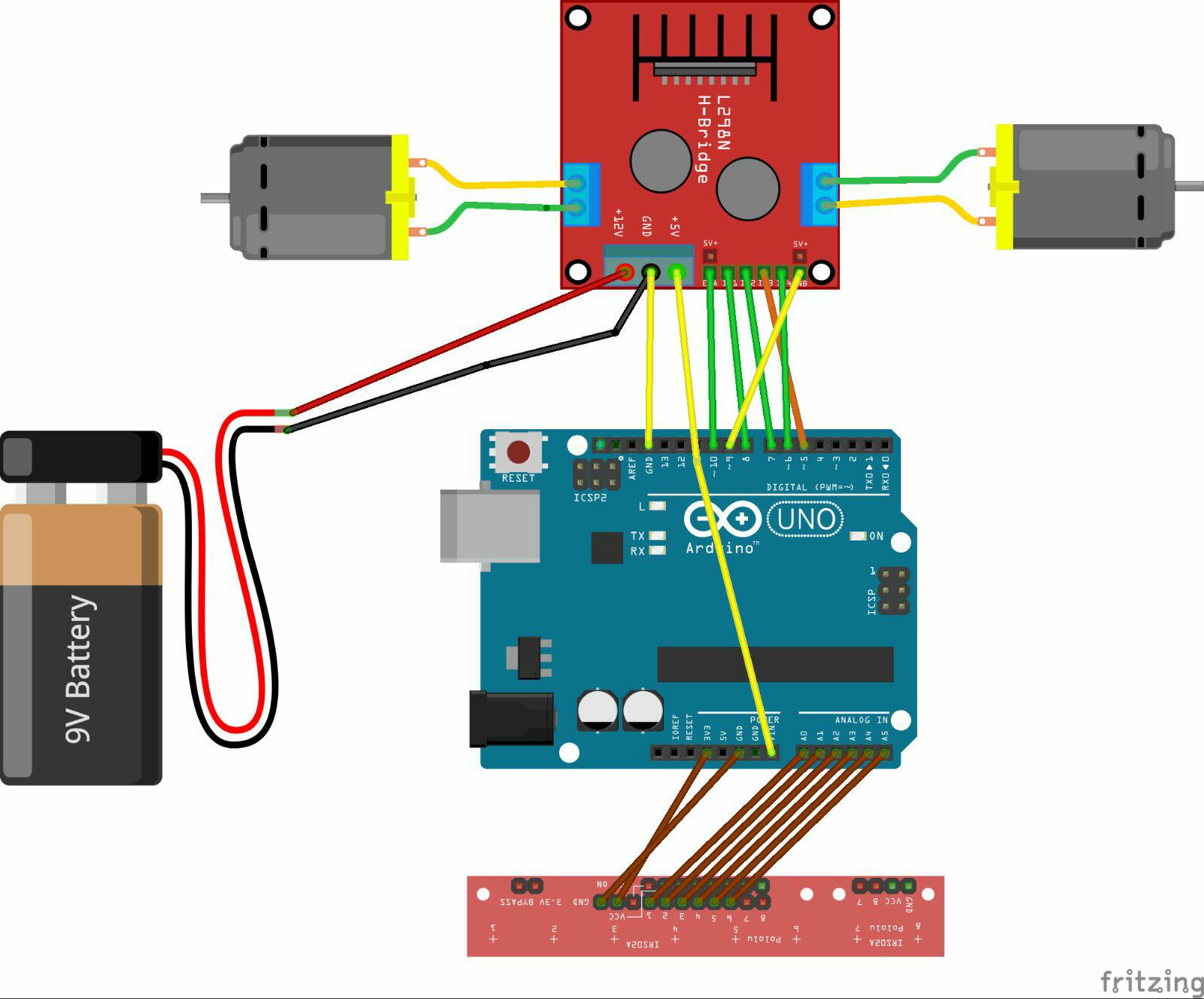
## 3.Donanım Yapısı:

### a.Gömülü Sistemler Mimarisi ve Devre Tasarımı

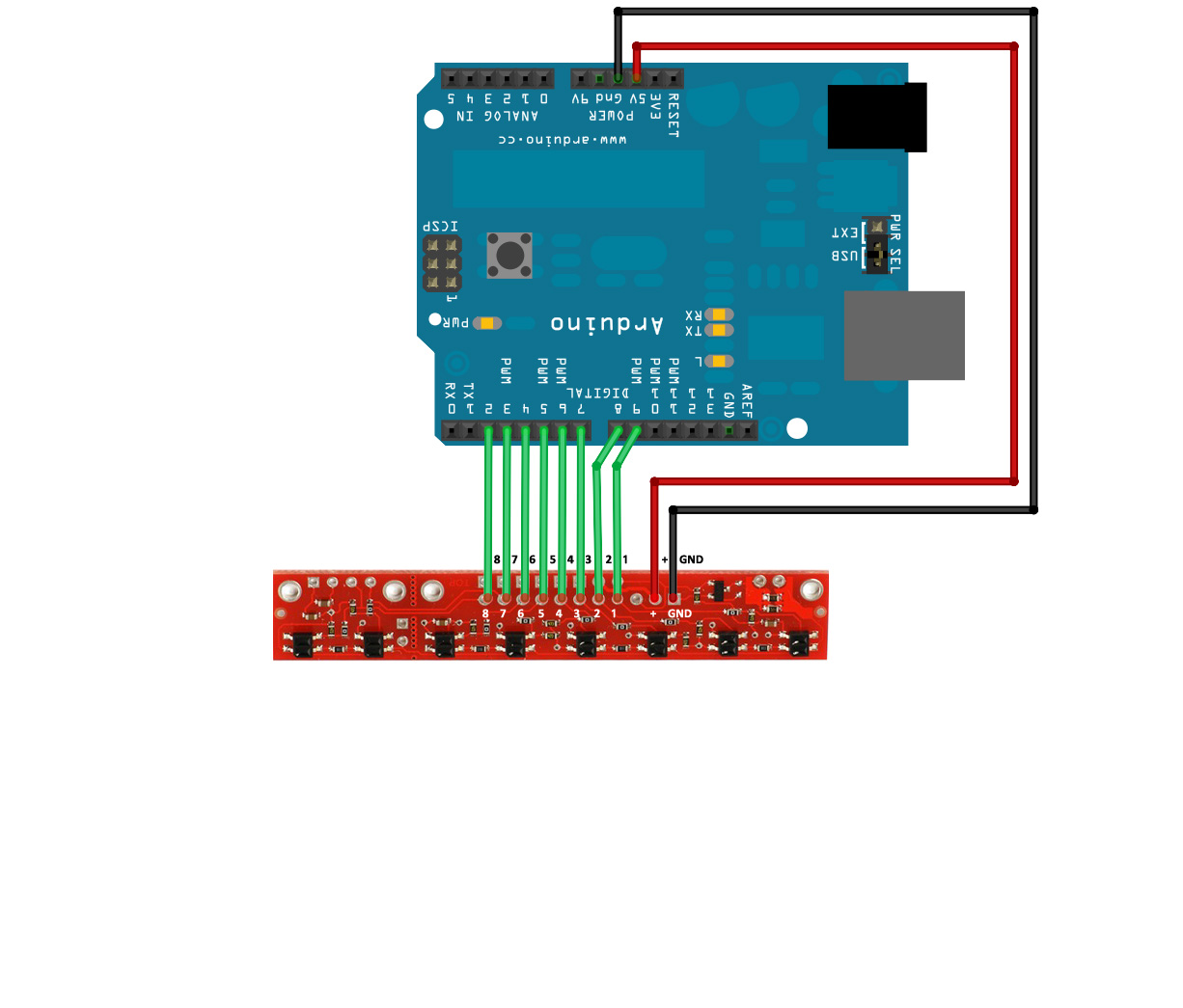
|  |  |
| --- | --- |
|  | * **Arduino Uno R3** * Arduino Uno; Atmega328 temelli bir mikrodenetleyici kartıdır. Üzerinde 14 adet dijital giriş/çıkış pini (6 tanesi PWM çıkışı olarak kullanılabilir), 6 analog giriş, 16Mhz kristal, usb soketi, güç soketi, ICSP konektörü ve reset tuşu bulundurmaktadır. Kart üzerinde mikrodenetleyicinin çalışması için gerekli olan her şey bulunmaktadır. Kolayca usb kablosu üzerinden bilgisayara bağlanabilir, adaptör veya pil ile çalıştırılabilir. |
|  | **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. |
|  | **Tekerlek**  65 mm çapında yumuşak ve yüksek sürtünmeli bir tekerlektir. 250 RPM, micro ve mini metal redüktörlü motorlarla tam uyumludur.  Çapı:65mm  Genişlik:30mm  Ağırlık:38gr |
|  | **250 RPM DC motor**  **Motor Özellikleri:**   * Çalışma Voltajı: 3-12V * Redüksiyon Oranı: 1:48 * Hız: 250 Rpm(@6V) * Akım: 95mA (maks. 160mA) * Ağırlık: 29gr |
|  | **Delikli Metal Sarhoş Teker 14mm**  M4 vida delikleri sayesinde robot gövdelerinize ve farklı ortamlara kolayca sabitlenebilir.  Bilye kısmının çapı 14mm, yüksekliği 21mm'dir. Çizgi izleyen robot başta olmak üzere bir çok robot projesinde kullanılabilir. Bilyesi metaldir. |
|  | Mavi mini breadboard 170 deliğe sahip ve çeşitli yüzeylere yapıştırabilmeniz için altı yapışkanlıdır. Ufak devre çalışmalarınızda ve devre kartlarının üzerine yapıştırarak hızlı bir şekilde prototipleme işlemini gerçekleştirebilir, devrelerinizi çalıştırabilirsiniz.    Boyutları: 45,72x35,56mm. |

#### Devre Tasarımı:

**L298N bağlantısı:**

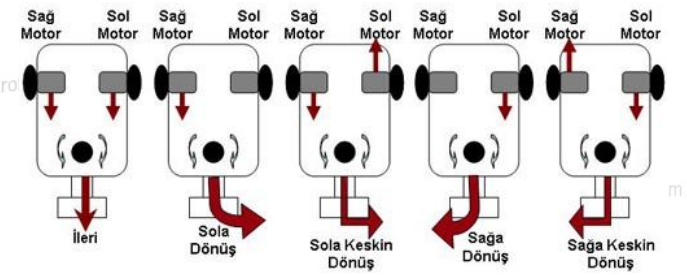
****

**QTR8-A bağlantısı:**

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

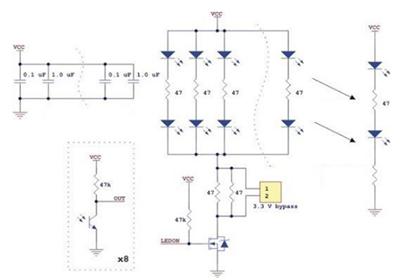
### b.Mekanik Sistem Mimarisinin Tanıtılması:

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

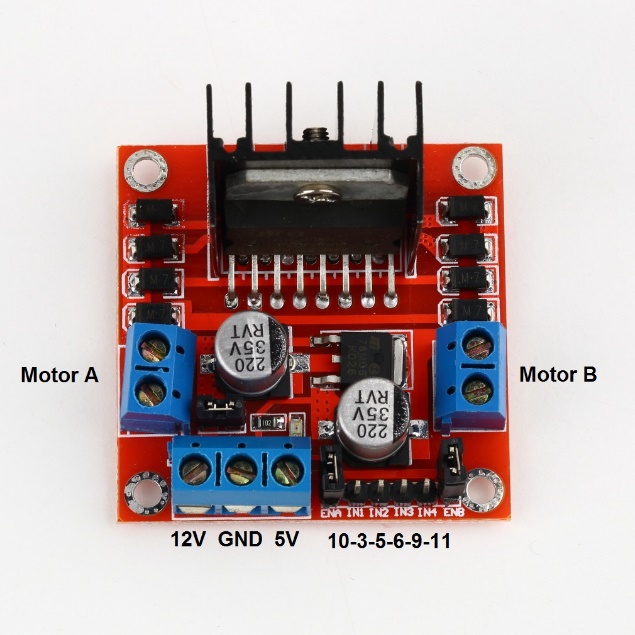


##### 1.QTR-8A Çizgi Sensörü

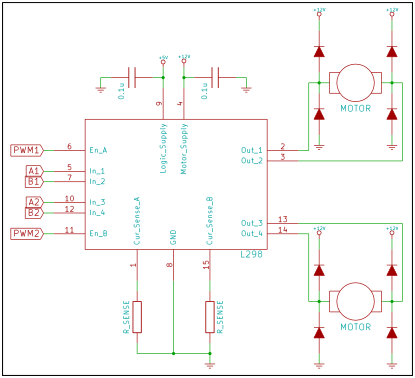
Burada bulunan 8 tane çizgi sensörü pin’i arduınonun analog bacaklarına bağlıyoruz.Vcc ve led on pin’ini +5V’a Gnd pin’inide  toprağa bağlıyoruz.QTR-8A Sensör kartı bir kaç farklı şekilde çalıştırılabilir. Bir mikrodenetleyicinin analog giriş pinlerine bağlanarak ADC(Analog Dijital Çevirici) işlemine 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.

**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.Arduino UNO R3



**Özellikler**

|  |  |
| --- | --- |
| Mikrodenetleyici | AT mega328 |
| Çalışma Gerilimi | 5V |
| Giriş Gerilimi(önerilen) | 7-12V |
| Dijital I/O Pinleri | * 14(6 tanesi PWM çıkışı) |
| Analog Giriş Pinleri | 6 |
| Her I/O için Akım | 40mA |
| * 3.3V Çıkış için Akım | * 50 mA |
| EEPROM | * 1KB (ATmega328) |
| SRAM | * 2KB(ATMega328) |
| Flash Hafıza | * 32KB (Atmega32bootloader) |
| Uzunluk | * 68.8mm |
| Genişlik | 53.4mm |
| Ağırlık | * 25gr |

Arduino Uno gücünü usb üzerinden veya harici güç kaynağından alabilir. Harici güç kaynağı AC-DC adaptör olabileceği gibi bataryada olabilir. Adaptör kart üzerindeki 2.1mm merkez-pozitif güç soketinden bağlanabilir. Batarya kart üzerindeki GND ve Vin pinleri üzerinden bağlanabilir.

Kartın çalışması için sürekli olarak usb'nin bağlı olması şart değildir. Kart sadece adaptör veya batarya ile çalıştırılabilir. Bu sayede kart bilgisayardan bağımsız olarak çalıştırılabilir.

Harici güç kaynağı olarak 6-20V arası kullanılabilir. Ancak bu değerler limit değerleridir. Kart için önerilen harici besleme 7-12V arasıdır. Çünkü kart üzerinde bulunan regülatör 7V altındaki değerlerde stabil çalışmayabilir. 12V üstündeki değerlerde de aşırı ısınabilir.

Uno kartının üzerindeki mikrodenetleyicinin çalışma gerilimi 5V'dur.

Güç pinleri:

* **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.

##### 4.6V 250 RPM DC Motor

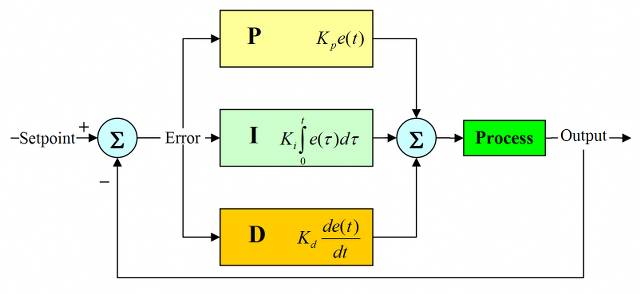


**Şekil 8.DC motor**

**Özellikler:**

* **Çalışma voltajı:**6V
* **Maks.Çalışma Voltajı:** 12V
* **Devir:**101-300 RPM
* **Boşta Çektiği Akım(6V):** 50mA
* **Zorlanma Akımı(6V):** 160mA
* **Motor Çapı:**2.5 cm
* **Redüksiyon Oranı:** 1:48
* **Mil Çapı:** 3mm
* **Mil Uzunluğu:**
* **Uçtan Uça Uzunluk:**6.5 cm
* **Ağırlık:** 29 gr

## 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)

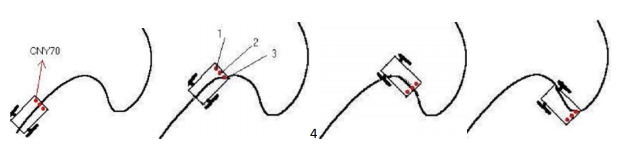
D = (Kd \* HD)/dt

PID = P + I + D

EHata = Hata

### 4.Yazılım Yapısı

a.Algoritmik olarak:



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:

#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:

## Her şeyden önce grup olarak nasıl çalışılacağını , nasıl işbirliğini yapılacağını sorunlarla başa çıkmayı öğrendik.Bu süreçte çeşitli malzemeleri , ne işe yaradıklarını öğrendik.Bir grup olarak çalıştık ve robotumuzu meydana getirdik.Bu çalışma bize hem grup olarak çalışma konusundaki tecrübemizi artırmamızı ağlarken,aynı zamanda daha önce robot yapmamış olmamıza rağmen grupça bu konuda bir bilgi sahibi olduk.Proje sayesinde devre tasarımında devre elemanlarının zarar görmemesi için yapılması gerekenleri devre tasarımının nasıl olması gerektiğini ve bağlantıları öğrendik. Devrenin kurulumu,gerekli kodlamanın yapılması, devrelerin şematik olarak gösterilmesi gibi konuları öğrendik.Robotun mekanik yapısını ve kodlamanın nasıl yapılacağını öğrendik.

## 2.Teknolojik Katkıları:

Endüstriyel lojistik uygulamalarında çizgi izleyen robotların kullanımı artmaktadır. Yakın gelecekte üretim hatlarında operatörlerin kullandığı taşıma ve kaldırma araçlarının yerini çizgi izleyen robotlar alarak hatlarda otonom olarak besleme işlemlerini gerçekleştireceklerdir. Çizgi Takip Eden Robotlar gelecekte sanayinin lojistik ve taşıma problemlerine çözüm getirerek endüstriyel anlamda otomasyon sistemlerinde kullanılmaya başlayacaktır Robot dünyası bütün dünyada olduğu gibi Türkiye ‘de de ilgi görmektedir.Bizler de şimdiden kolay ve basit programlanan robotlarımızla projeler geliştirerek kendimizi bu alanda ilerletmeyi ve ülkemize faydalı olmak istiyoruz.

3.Ekip Çalışması Katkıları**:**

Ekip çalışmasının bize kattıkları:

* İlk olarak hepimizin bu konudaki teknik bilgisi gelişti.Herkes yapabileceği görevleri üstlendi.Bu sayede de çalışmamız bir düzen ve bütünlülük içinde tamamlandı.
* İşbirliği ve yardımlaşma içinde çalıştık ve bu çalışmamızın daha verimli oluşunu sağladı.
* Bu konu üzerinde hep birlikte uzun bir süre içinde çalıştık.Bu ilgilerimizin daha kalıcı olmasına katkı sağladı
* Karşılaştığımız sorunlarda birlikte çözüm aradık.Çözüm önerilerinde bulunduk.Grup olarak çalışınca sorunlar çok daha hızlı çözüm buldu.
* Birbirimizle iletişim alışanlığımız gelişti.
* Hatalarımızı gördüğümüzde bunları düzelttik.Bu çalışmamızın çok daha verimli olmasını sağladı.

## 4.Aksayan Yönler:

Projeyi gerçekleştirirken karşılaştığımız sorunlar:

Projeyi gerçekleştirirken öncelikle sensörümüzde sıkıntı çıktı. Seri portta değişik değerler veriyordu. Ardından kod ile ilgili bir sorunla karşılaştık ve kodumuzla bir yerden aldığımız kodu değiştik.Böylece sorunlarımız çözülmüş oldu.

## 5.Görüş ve Öneriler:

* Grup olarak iyi bir çalışma ortaya koyduğumuzu düşünüyor ve bu çalışmamızın bizi geliştirdiğine inanıyoruz.
* Ortaya çıkan sorun bizi endişelendirdi ve nasıl çözüm bulabileceğimizi düşündük.Bu sorunlarla karşılaşmamız ise bize bir grup olarak sorunlarla başa çıkmayı öğretti.
* Okulda işleri birlikte yürüttük fakat okul dışında toplu bir şekilde buluşup işi yürütmeyi tercih etmedik.Bunun yerine herkes belli bölümleri üstlendi ve yaptıklarından bizi de bilgilendirerek kendisi tamamladı.Belki buluşup yapsaydık çok daha verimli bir çalışma olabileceğini düşünüyoruz.

