# AFLBOT DONANIM

## Arduino Kartı C dili Program Yapısı

Arduino kartınıza C dili ile kod yazarken iki temel fonksiyonumuz vardır. Bunlar setup ve loop fonksiyonlarıdır.

Setup fonksiyonu Arduino kartına enerji verdiğiniz andan itibaren 1 defa çalıştırılır. Reset butonuna basıncaya kadar bir daha çalıştırılmaz. 1 defa çalıştırılan bu setup fonksiyonu içerisinde kurulum ayarları, donanımla ilgili özelliklerin başlatılması, diğer cihazlarla iletişimin başlatılması gibi düzenlemeler yapılabilmektedir.

Loop fonksiyonu ise programın ana döngüsünü ifade eder. Bu döngü içerisine yazacağımız kodlar sürekli olarak tekrar edilir.

void setup() {

// put your setup code here, to run once:

}

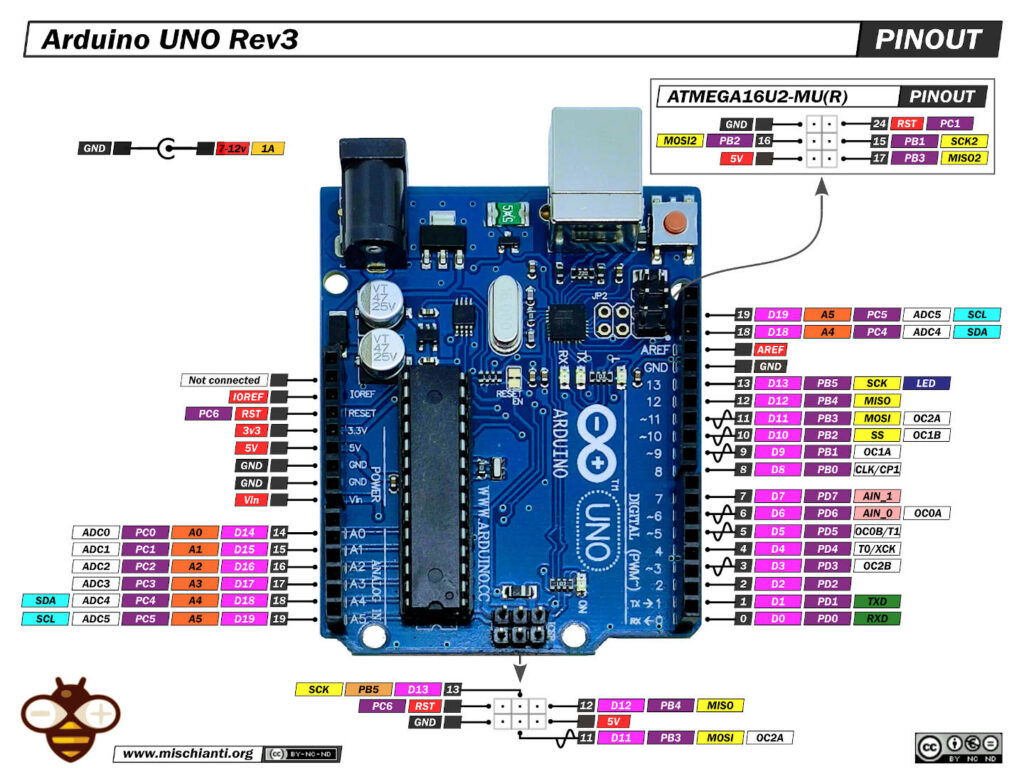
void loop() {

// put your main code here, to run repeatedly:

}

#### Pin Kontrolü:

Arduino uno kartınızın kontrol edebileceğiniz 20 adet pini vardır. Bu pinlerin her birinin dijital giriş veya dijital çıkış vermesi sağlanabilir. Bazı pinlerin ekstra özellikleri bulunmaktadır. Örneğin A0-A5 arasındaki pinler ekstra olarak analog veri girişine de izin vermektedir.



Şekil 1arduino uno pinout

Bir pinin giriş ya da çıkış olarak kullanılacağı mikrodenetleyiciye bildirilmelidir. Bunun için pinMode fonksiyonu kullanılır.

**pinMode(parametre1, parametre2)** fonksiyonu iki parametre alır. Bu parametrelerden ilki hangi pin üzerinde işlem yapılacağını bildirir. İkinci parametre ise yapılacak işlemi bildirir.

Bir pin üzerinde yapılabilecek işlemler şunlardır:

**OUTPUT:** Bir pini kullanarak başka bir donanıma bilgi sağlamak üzere dijital çıkış işleminin yapılacağını bildirir. Bir pinden alınabilecek max akım değeri 40mA olarak sınırlandırılmıştır. Bu değerin üzerine çıkıldığında kartınız bozulabilir. Pinin düşük empedansa sahip olmasını sağlar.

Bir pin çıkış olarak ayarlandığında dijital çıkış elde edilebilir. Pin lojik 0 yapıldığında pinden 0V çıkar. Pin lojik 1 yapıldığında ise pinden 5V çıkışı sağlanır.

**INPUT:** Bir pinin giriş olarak ayarlanacağını bildirir. Dışarından girilecek verinin Arduino kartı tarafından alınarak değerlendirilmesini sağlar. Pinin yüksek empedansa sahip olmasını sağlar.

**INPUT\_PULLUP:** Bir pinin giriş olarak ayarlanırken default durumunun YÜKSEK seviyede olmasına imkan sağlar. Bu sayede input pinine bağlayacağınız bir butonun boşta kalan ucunu gnd’ye bağlarsanız butona basıldığında ilgili pin lojik 0 seviyesine gideceğinden lojik durum değişecektir. Böylece butona basıldığı anlaşılacaktır.

**digitalWrite(parametre1, parametre2)** bu fonksiyonda 2 parametre alır. İlk parametre üzerinde işlem yapılacak olan pini belirtir. İkinci parametre ise belirtilen pinin dijital olarak hangi değeri alacağını bildirir.

Uygulayınız: Örnek kodu tinkercad üzerinde uygulayınız.

void setup() {

pinMode(13, OUTPUT); // 13 numaralı pin çıkış olarak ayarlanır.

}

void loop() {

digitalWrite(13, HIGH); // 13 numaralı pin lojik 1 seviyesine ayarlanır.

delay(1000); // 1000 ms yani 1 sn beklenir.

digitalWrite(13, LOW); // 13 numaralı pin lojik 0 seviyesine ayarlanır.

delay(1000); // 1 saniye beklenir.

}

**analogWrite(parametre1, parametre2)** bu fonksiyon da iki parametre alır. Burada ilk parametre Arduino kartınızın PWM özelliğine sahip pinlerinden birisidir. Bu pinlerin hangisi olduğunu görmek için pinlerin yanında sinüs ~işareti yer alır. Pwm özelliği led parlaklığı, motor hız kontrolü gibi uygulamalarda kullanacağımız bir özelliktir.

PWM sinyalini incelediğimizde üretilen sinyalin bir kare dalga sinyal olduğunu görmekteyiz. Buna göre üretilen sinyal bir periyotu boyunca, yüksek seviyede kalan bölümü % olarak ifade edilmiştir. Sinyalin bir periyotu boyunca yüksek seviyede kalan bölümüne duty cycle-görev süresi- denir. Duty cycle %100 olduğunda sinyalin o periyot için ürettiği toplam gerilim 5V olacaktır. Duty cycle %50 olduğunda üretilen sinyalin ortalama gerilimi 2.5V olacaktır.

Duty cycle’ın ayarlanması için ise analogWrite fonksiyonu kullanılır. Analogwrite fonksiyonunda ilk parametre pwm pinini ifade eder ikinci parametre ise duty cycle değerini 0-255 aralığında yüzde karşılığını ifade eder. 255 değeri %100 duty cycle iken 64 değeri %25 duty cycle anlamına gelir.

metin, ekran görüntüsü, diyagram, tasarım içeren bir resim

Açıklama otomatik olarak oluşturuldu

Şekil 2PWM sinyalinin incelenmesi

Pwm ile led diyotun parlaklığını kontrol edebiliriz. Bunun için 11 numaralı pine led diyotun anotu(led diyotun uzun bacağı), gnd pinine ise led diyotun kısa bacağı(katodu) bağlanır.

metin, ekran görüntüsü, devre, tasarım içeren bir resim

Açıklama otomatik olarak oluşturuldu

Şekil 3pwm ile led kontrolü

Uygulayınız:

void setup() {

  // put your setup code here, to run once:

  pinMode(11, OUTPUT);

}

void loop() {

  // put your main code here, to run repeatedly:

  analogWrite(11, 20);

  delay(1000);

  analogWrite(11, 120);

  delay(1000);

  analogWrite(11, 220);

  delay(1000);

}

## MOTOR SÜRÜCÜ İLE MOTOR KONTROLÜ

Bu bölümde motor sürücü modülü L298 kullanılarak bir motorun hızının ve yönünün nasıl değiştirildiğini inceleyeceğiz.

L298 motor sürücü modülü 2 kanala sahip bir sürücü modülüdür. Her bir kanaldan max 2A’e kadar akım geçişine imkan sağlar. Motor sürücü modülünün nalları istenirse köprü bağlanarak daha yüksek akım ihtiyaç duyan 1 motoru çalıştırabilir.

metin, devre, elektronik donanım, elektronik mühendisliği içeren bir resim

Açıklama otomatik olarak oluşturuldu

Şekil 4motor sürücü modülü pinout

Motor sürücü modülünü incelediğimizde 2 kanala motor bağlantıları için 2 ayrı klemens sağ ve sol tarafta yerleştirildiği görülmektedir. Bunun yanında motorlara güç sağlamak için 3’lü klemens kullanılmıştır. Bu klemens 12V-GND-5V sıralaması ile sıralanmış pinlere sahiptir.

Arduino kartınızın bu motor sürücüyü kontrol edebilmesi için pin kontrolü ve pwm çıkışlarına ihtiyaç vardır. Motor sürücü modülü üzerinde sırası ile ENA-IN1-IN2-IN3-IN4-ENB pinleri yer almaktadır. ENA pini direkt olarak jumper ile 5V’a bağlandığında bağlı olan motor son hızda dönecektir. Jumper çıkartılıp Arduino kartınızın bir pwm pinine bağlantı sağlarsanız l298 modülüne bağlı olan motorların hız kontrolünü de yapmış oluruz. IN1 ve ıN2 pinleri ise yüksek-yüksek olacak şekilde Arduino kartı programlanırsa motorlar frenleme yapacaktır. Motorun tek bir yönde hareket etmesi isteniyorsa IN1 ve IN2 zıt lojik seviyelerde olmalıdır. IN1 HIGH,IN2LOW şeklinde olmalıdır.

### Devre Şeması:

metin, diyagram, plan, ekran görüntüsü içeren bir resim

Açıklama otomatik olarak oluşturuldu

Şekil 5Motor sürücü modülüne bağlanan motorlar ve kontrolü sağlayan arduino kartı bağlantısı

### Uygulama Kodları:

void setup() {

  pinMode(8, OUTPUT);

  pinMode(9, OUTPUT);

  pinMode(10, OUTPUT);

}

void dur() {

  analogWrite(10, 0);

  digitalWrite(8, LOW);

  digitalWrite(9, LOW);

}

void geri() {

  analogWrite(10, 128);

  digitalWrite(9, LOW);

  digitalWrite(8, HIGH);

}

void ileri() {

  analogWrite(10, 255);

  digitalWrite(8, LOW);

  digitalWrite(9, HIGH);

}

void loop() {

  ileri();

  delay(5000);

  geri();

  delay(5000);

  dur();

  delay(3000);

}