



Bilgisayarca
Düşünmeyi
Destekler



21. yy
Öğrencileri
için
Tasarlandı



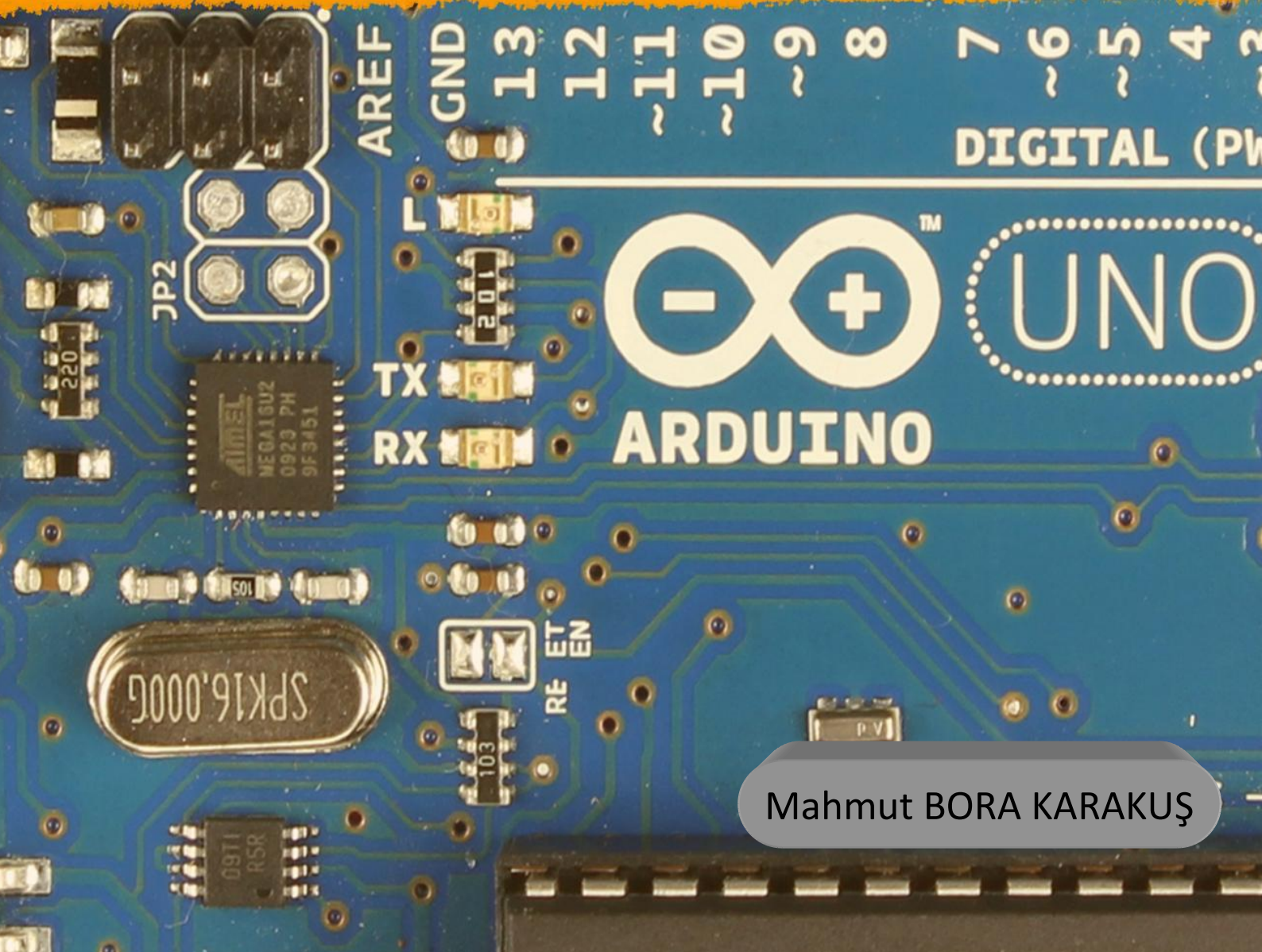
SCRATCH

ile

Güncellenmiş
2. Versiyon



ARDUINO
Programlama



Mahmut BORA KARAKUŞ



Bütün ümidim gençliktedir.

Mustafa Kemal ATATÜRK



Hayallerinle Dünyanı Programla



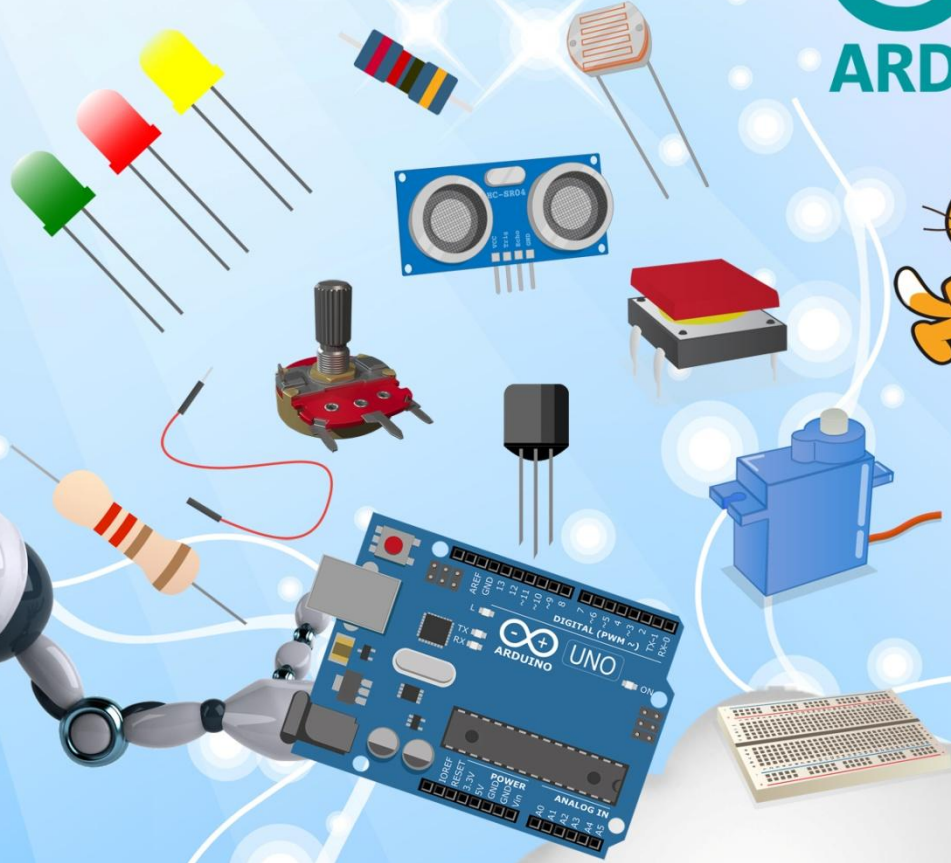


[Scratch ile Programlamaya Giriş E-Kitabımı okumak için Tıklayınız!](#)

Scratch ile Arduino Süper Başlangıç Seti

ali's
robotics

ARDUINO



Ücretsiz
E-Kitap

Scratch ile Arduino Süper Başlangıç Setini Almak için Tıklayınız!

Not: Aldığınız her set ile bir öğrencinin robotik ve kodlama eğitimine katkıda bulunmuş olacaksınız!

İÇİNDEKİLER

BÖLÜM 1: SCRATCH ile ARDUİNO PROGRAMLAMA

Arduino Nedir?	12
Arduino'yu Nasıl Programlarız?	13
Arduino'muzu Tanıyalım	14
Mblock ile Kullanılabilecek Kartlar	15
Arduino'yu Bilgisayara Bağlama	16
Mblock Programı	17
İlk Programımızı Yazalım	19
Temel Elektronik	20
Breadboard üzerindeki Led'i Yakıp Söndürme	24
Kara Şimşek Devresi	25
Dijital Pinden Değer Okuma	26
RGB Led	27
Buton ile RGB Led Kontrolü	28
Potansiyometre	30
Uzay Savaş Oyunu	31
Sensörler	32
Hava Kararınca Yanan Led	33
Mesafe Sensörü	35
Zıplama Oyunu	36
Yazdığımız Programı Arduino'ya Yükleme	37
Su seviyesi Sensörü	39
Hava Kalite Sensörü	40
Alev Sensörü	41
Toprak Nem Sensörü	42
Servo Motor	45
L9110 Pervaneli Motor	47
Potansiyometre ve Pervaneli Motor	49
Motor sürücü Kartları	53



Kitapta kullanılan uygulamaların kaynak dosyalarını indirmek için **tıklayınız!**

Programlama Nedir?

Programlama: Bilgisayarın ne yapması gerektiğinin adım adım söylenmesidir. Bilgisayarlar bizim isteklerimiz doğrultusunda çalışan elektronik aletlerdir. İstedığımız iş ve işlemleri yapabilmeleri için programlanmaları gerekir. Programlama ile biz bilgisayara işleri nasıl yapması gerektiğini anlatırız.

Bilgisayarda çalışan bir çok program vardır. Bu programlar programlama dilleri kullanılarak yazılır. İnsanların günlük hayatta iletişim kurmak için nasıl bir dil kullanmaları gerekiyorsa bizim de bilgisayarlar ile haberleşebilmemiz için bir dil kullanmamız gerekir. Bu diller yazılı dillerdir . Üst seviye(kolay) ve alt seviye(zor) diller vardır. Okullarımızda okutulan Bilişim Teknolojileri ve Yazılım derslerinde öğrendiğimiz Scratch programlama dili üst seviye(Kolay) bir programlama dilidir.



Scratch MIT (Massachusetts Institute of Technology) üniversitesi tarafından geliştirilen bir görsel programlama dilidir. Bu dil özellikle 9-16 yaş grubu öğrenciler için oluşturulmuştur. Bu dili kullanarak kendi oyununuzu programlayabilir, kendi animasyonlarınızı oluşturabilir ve yaratıcılığınız dahilinde birbirinden güzel çalışmalar üretebilirsiniz.

Bilgisayarca Düşünme

“Computational Thinking (Bilgisayarca Düşünme) terimini bilgisayarları üretim amaçlı olarak hayat problemlerinin çözümünde kullanabilmek için gerekli olan bilgi, beceri ve tutumlara sahip olmak diye tanımlamak mümkündür.”

M. Yaşar ÖZDEN

Bilgisayarca düşünmenin temelinde problem çözme vardır. Günümüz öğrencileri teknolojiyi etkin bir biçimde kullanmaktadırlar. Bilgisayarca düşünmede ise amaç teknolojiyi ve bilgisayarları problem çözmede kullanmaktır. Bilgisayarca düşünme, bilgisayarların işlem gücü ile insanın düşünme yeteneklerinin birleşmesidir. Bu ikisi, yaşam kalitemizi arttırmak ya da yaratıcı çözümler üretmek için kullanılır.

Bilgisayarca düşünme öğrencilerin sahip olması gereken 21.yy yeteneklerini de içeren bir düşünme biçimidir.

Bu yetenekler aşağıdaki gibidir.

- ⊗ Kodlama
- ⊗ Algoritmik düşünme
- ⊗ Eleştirel düşünme
- ⊗ Yaratıcı düşünme
- ⊗ Problem çözme
- ⊗ İletişim
- ⊗ İş birliği

Bu yüzyıl geçtiğimiz yüzyıldan farklı olarak hızlı bir değişimi bünyesinde barındırmaktadır. Günümüz öğrencileri meslek hayatına atıldığında karşılaşacakları dünya bugünkü dünyadan çok farklı olacaktır. Bu değişimin sonucu olarak bu yüzyılın gerektirdiği yeteneklere sahip olan bireyler, her zaman bir adım önde olacaklardır.

Yazar Hakkında



• İlk ve orta öğrenimimi Mersin'de tamamladım.

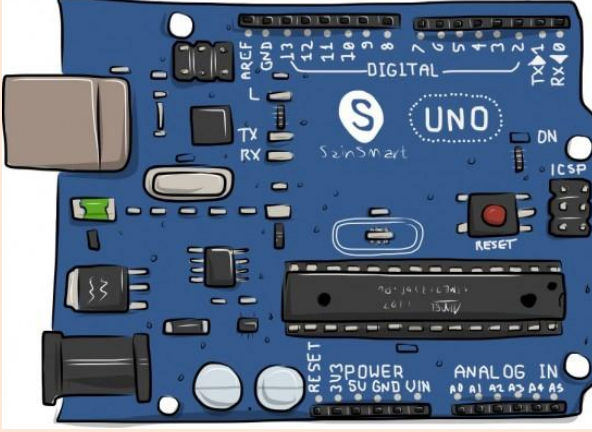
• Çanakkale 18 Mart Üniversitesi Bilgisayar ve Öğretim Teknolojileri Öğretmenliği mezunuyum.

• Ahmet Yesevi Üniversitesi Bilgisayar Mühendisliği 4. Sınıf öğrencisiyim.

• Bir devlet okulunda Bilişim Teknolojileri ve Yazılım Dersi öğretmeni yapmaktayım.



Arduino Nedir?



Arduino, mikrodenetleyici temelli programlanabilir bir geliştirme kartıdır. Arduino ile kendi robotlarınızı yapabileceğiniz gibi kendi elektronik devrelerinizi de oluşturabilirsiniz.

Not: Arduino'ya elektronik bir beyin diyebiliriz. Bu beyini projelerimizde kullanabilmek için programlamamız gerekmektedir.



Arduino'yu Nasıl Programlarız?

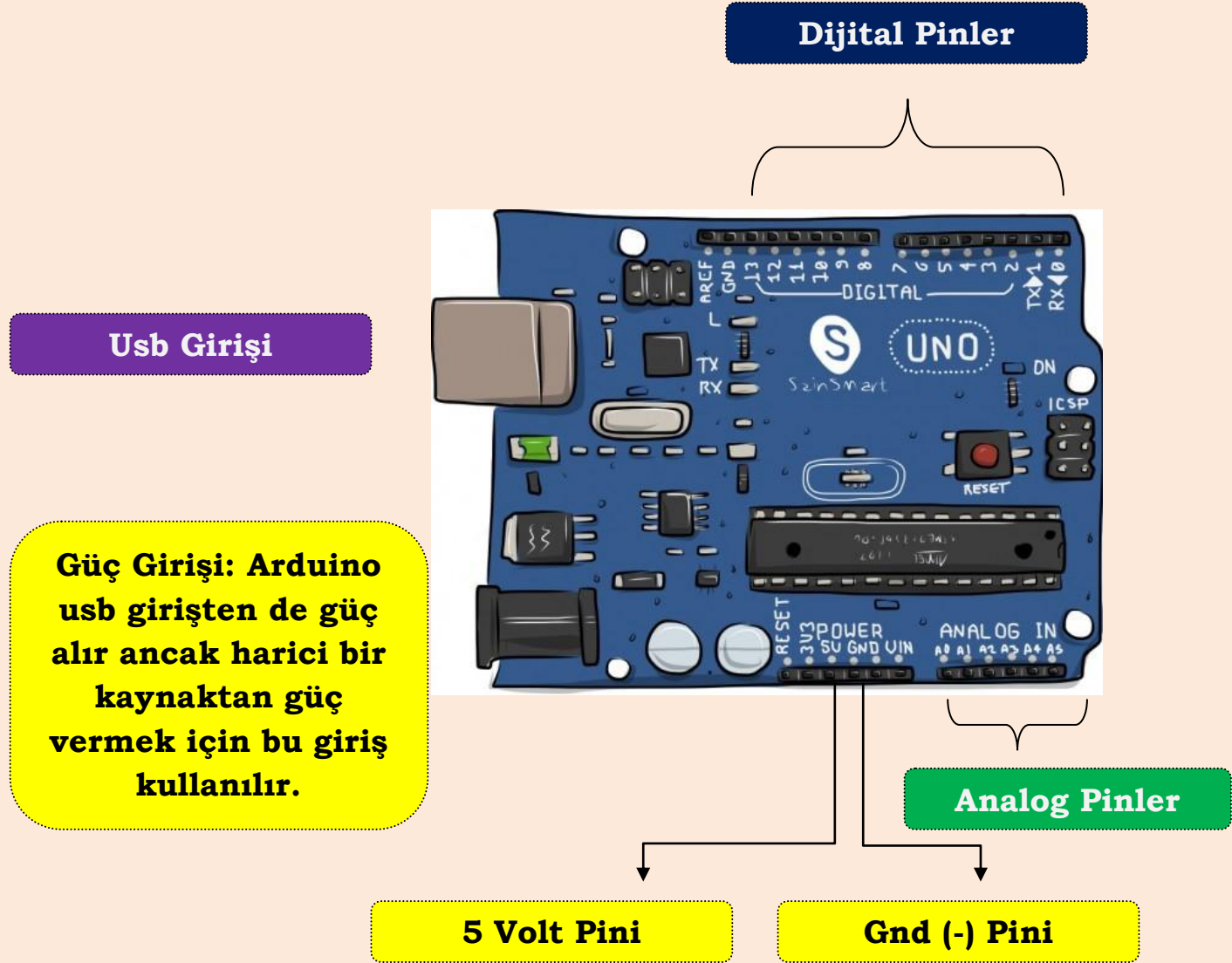
Arduino'yu programlamak için bir programlama diline ihtiyacımız var. Biz kitabımızda Arduino'yu programlamak için Scratch programlama dilini kullanacağız.



Mblock programı Makeblock firmasının geliştirmiş olduđu Scratch programlama dili üzerine inşa edilmiş bir programdır. Şu anda Arduino programlamada kullanılan en popüler görsel programlama aracıdır. Biz de kitabımızda Arduino programlarken genellikle bu programı kullanacağız.

[Mblock Programını İndirmek İçin Tıklayınız!](#)

Arduino'muzu Tanıyalım



Usb Girişi: Arduino'yu bilgisayara bağlamak için

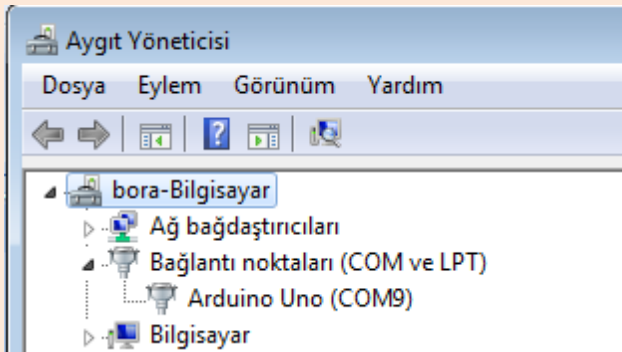
Dijital Pinler: Arduino'muzu çevresiyle etkileşim halinde tutabilmek için pinleri kullanırız. Dijital pinleri Arduino'dan dışarıya akım vermek ya da gelen akımları akımak için kullanırız. Örneğin; Arduino'ya bağladığımız bir Led'i yakmak için dijital pinleri kullanırız.

Analog Pinler: Arduino'ya bağladığımız sensörleri okumak için analog pinleri kullanırız.

Arduino'yu Bilgisayara Baęlama



Arduino'yu bilgisayarımıza usb kablosu aracılıęıyla baęlarız. Bilgisayarınızın Arduino'yu tanıyıp tanımadıęını anlamak ya da nasıl baęlandıęını grmek iin bilgisayarımıza saę tıklayın, zellikler mensnden **aygıt yneticisini** aın.

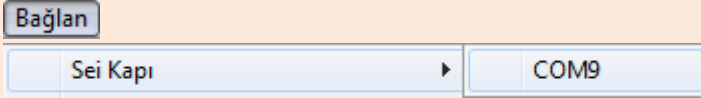


Arduino bilgisayara Com baęlantı noktasından baęlanır. Bende baęlantı noktası Com 9 olarak gzkmekte. Sizde farklı bir Com noktası gzkebilir.

Mblock Programını Tanıyalım



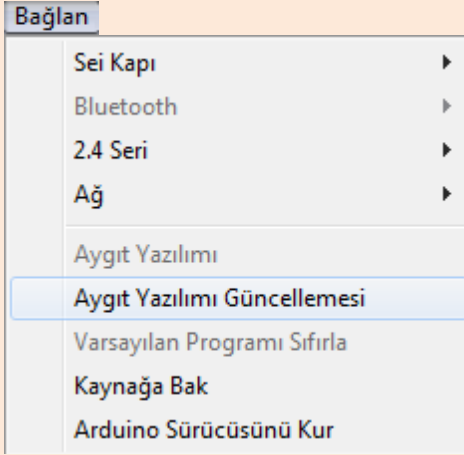
Arduino'yu Mblock ile Programlama



Öncelikli olarak Arduino'yu bilgisayara bağlayın. Mblock programını açın. Bağlan menüsünden seri kapı seçeneği ile Arduino'nun bağlı olduğu Com numarasını seçin.

Youtube Videosu İçin Tıklayınız!





Daha sonra yine bağlan menüsü altında bulunan “Aygıt Yazılımı Güncellenmesi” kısmından Arduino’yu programlamamız için gerekli programı Arduino’ya yüklüyoruz.

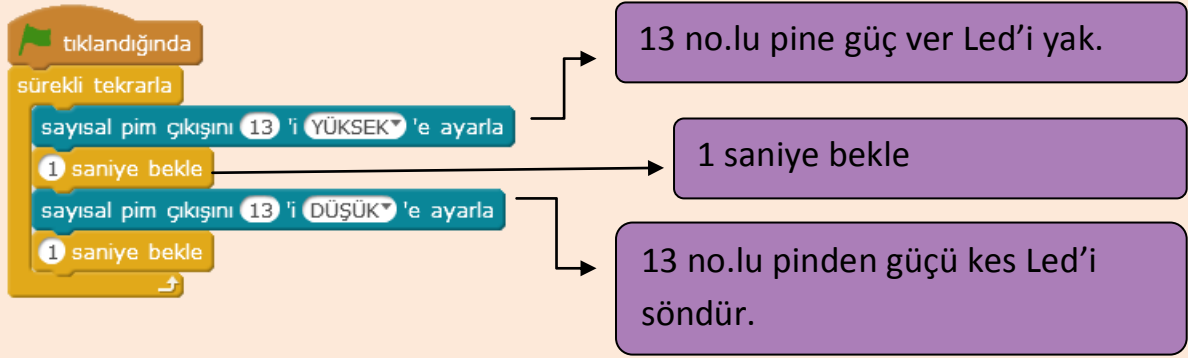


Yükleme bitti yazısı çıktıktan sonra Arduino’muzu programlamaya başlayabiliriz.

İlk Programımızı Yazalım

Arduino üzerinde dijital 13 no.lu pine bağlı bir Led mevcut. Biz bu Led'i Scratch ile programlayacağız.

LED'ler elektrik enerjisini ışığa dönüştüren yarı iletken devre elemanlarıdır.



Not: Bağlan menüsünden Arduino'nun bağlı ve aygıt yazılımının yüklü olduğundan emin olun!

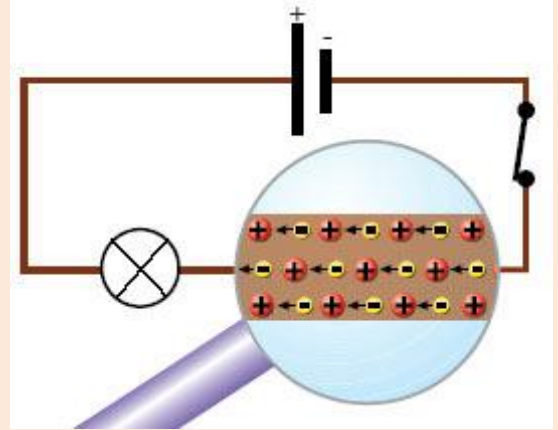
Temel Elektronik

Arduino'yu programlarken pek fazla elektronik bilgisi gerekmez ancak elektronik hakkında temel bilgiler edinmekte fayda var.

Elektronik, elektronik aygıtları çalıştırmak için küçük elektrik akımlarının nasıl kumanda edilmesi gerektiğini inceleyen bilim dalıdır. Kullandığımız bilgisayar, televizyon, akıllı telefonlar hepsi elektronik bilimi sayesinde hayatımıza girdi. Bizde kitabımızda bazı elektronik kavramlarını ve elektronik devre elemanlarını inceleyeceğiz. İnceleyeceğimiz devre elemanları, kullandığımız bütün elektronik aletlerde bulunan parçalardır.

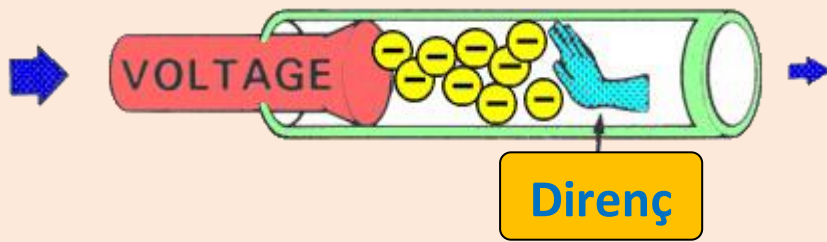
Elektrik Akımı

Elektrik akımı, elektron adı verilen çok küçük parçacıkların bir tel boyunca akışıdır. Bütün maddeler elektron içerir. Elektronlar her şeyi oluşturan atomların bir parçasıdır.

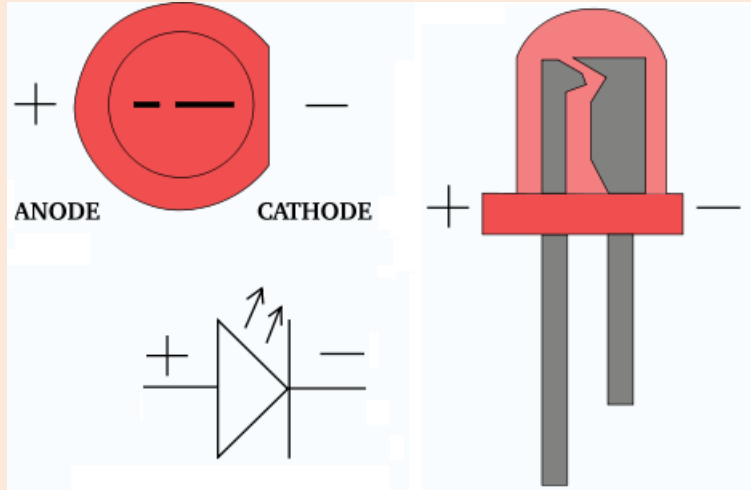


[Detaylı Bilgi İçin Tıklayınız!](#)

Elektronlar pilin (+) ucundan (-) ucuna doğru hareket ederler. Bu hareket sonucu bir gerilim oluşur, biz bu gerilime Voltaj diyeceğiz ve V ile göstereceğiz. Elektronların (+) uctan (-) uca doğru hareketine akım diyeceğiz ve I ile göstereceğiz. Akım (I) ve gerilim (V) bir devrenin temel bileşenlerinden ikisidir. Üçüncü temel bileşen dirençtir. Direnç, kimi maddelerin elektron akışına yani akıma karşı gösterdiği zorluktur.

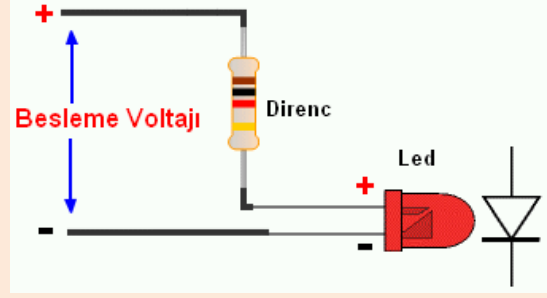


Elektronik Devre Elemanları



Led'ler, üzerinden akım geçtiğinde ışık yayan devre elemanlarıdır. Uzun ucu (+) kısa ucu (-). Yalnız dikkat etmemiz gereken husus: Asla Led'imizi direnç olmadan devremize bağlamayacağız. Çünkü Led'ler hassas devre elemanlarıdır, uzun süre yüksek akıma maruz kaldığında yanabilir ve bir gaz açığa çıkabilir! Bu gaz tehlikelidir ve asla solumayınız. Direnç akıma zorluk göstererek akımı azaltacak ve Led'imize uygun hale getirecektir.

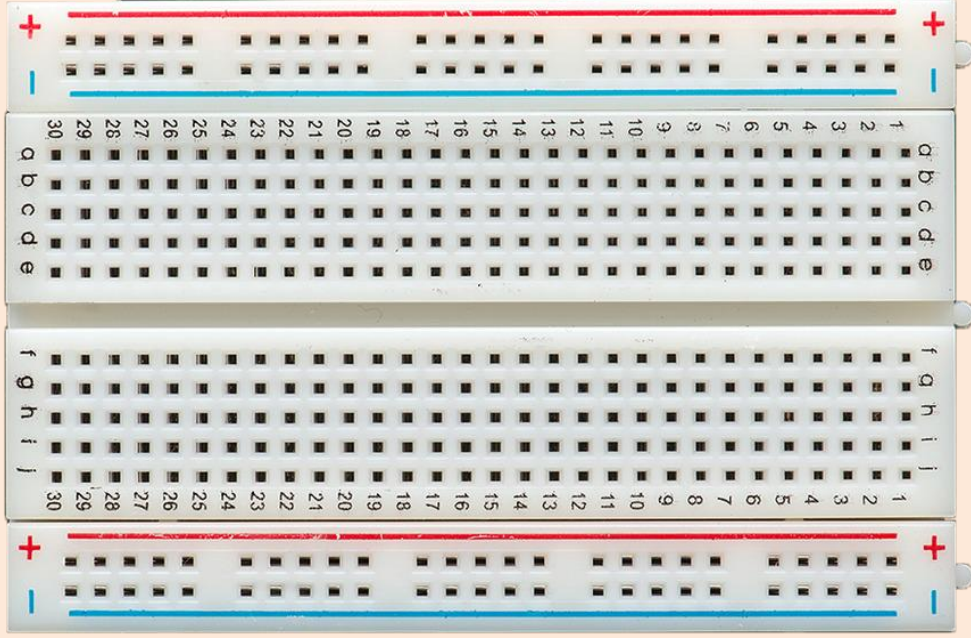
Dirençler akıma zorluk gösteren devre elemanlarıdır. Arduino ile pinlerden vereceğimiz akım Led'ler için yüksek değerdedir. Dirençler bu değeri düşürerek Led'e uygun hale getirirler.



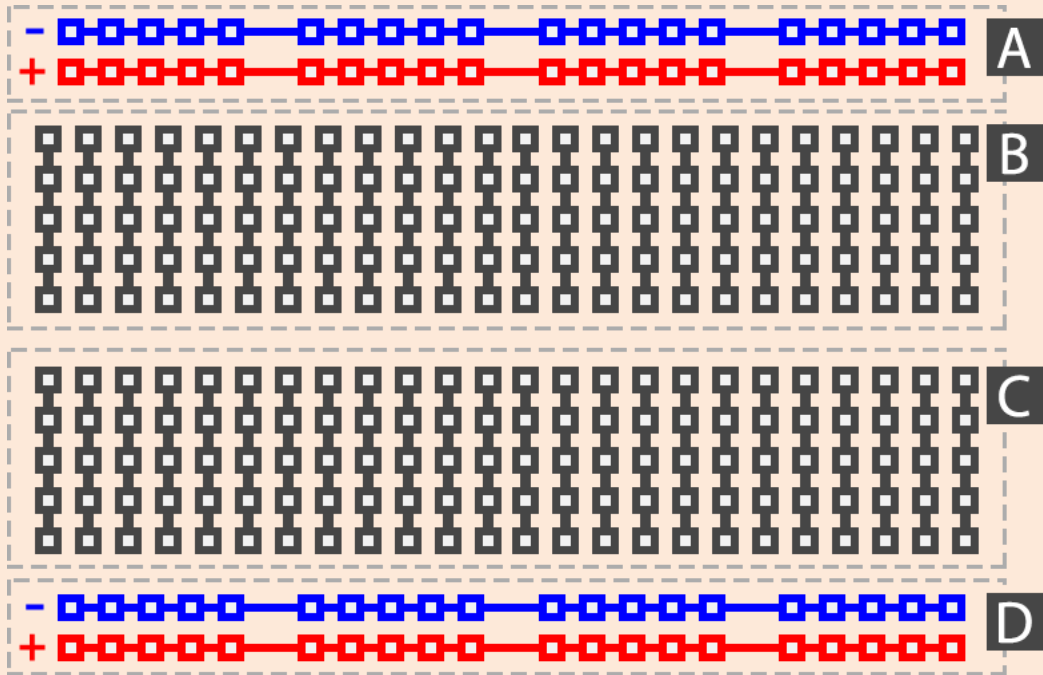
Jumper kablo Devre elemanlarını birbirine bağlamak için kullanılır. Bir ucunu Arduino'muza diğer ucunu da breadboarda bağlayacağız.



Breadboard

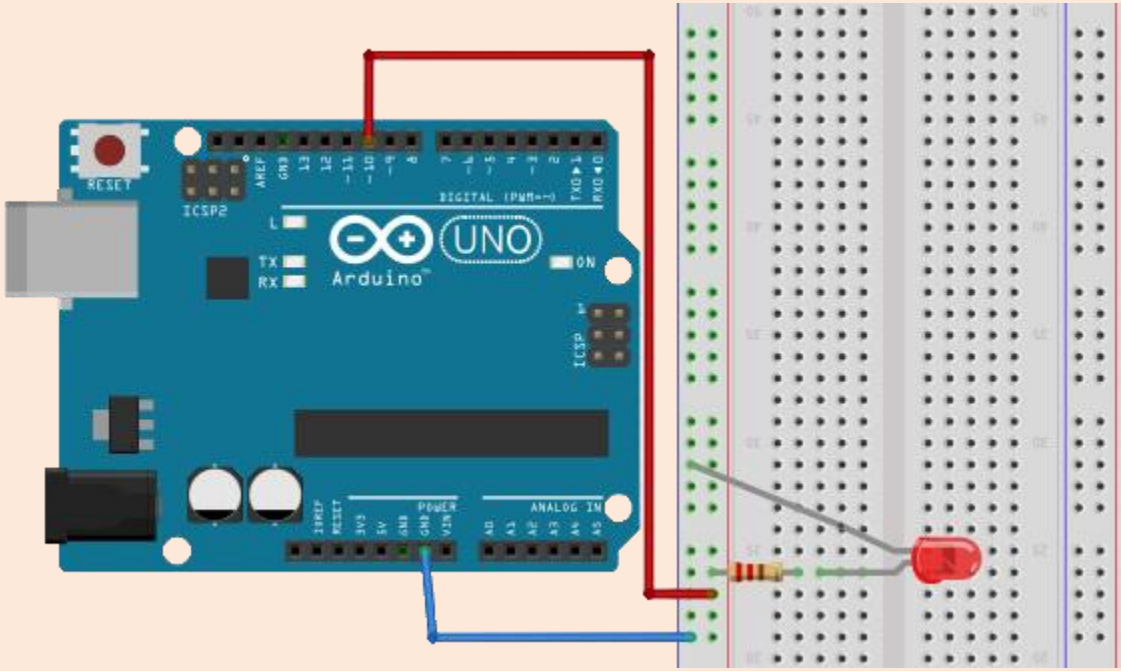


Breadboard devre elemanlarını birbirine bağlamak için kullanılır. Tekrar tekrar kullanabilirsiniz.



A ve D kısımları yatay olarak birbirine bağlıdır. B ve C kısımları ise dikey olarak birbirine bağlıdır.

Breadboard Üzerindeki Led'i Yakıp Söndürme



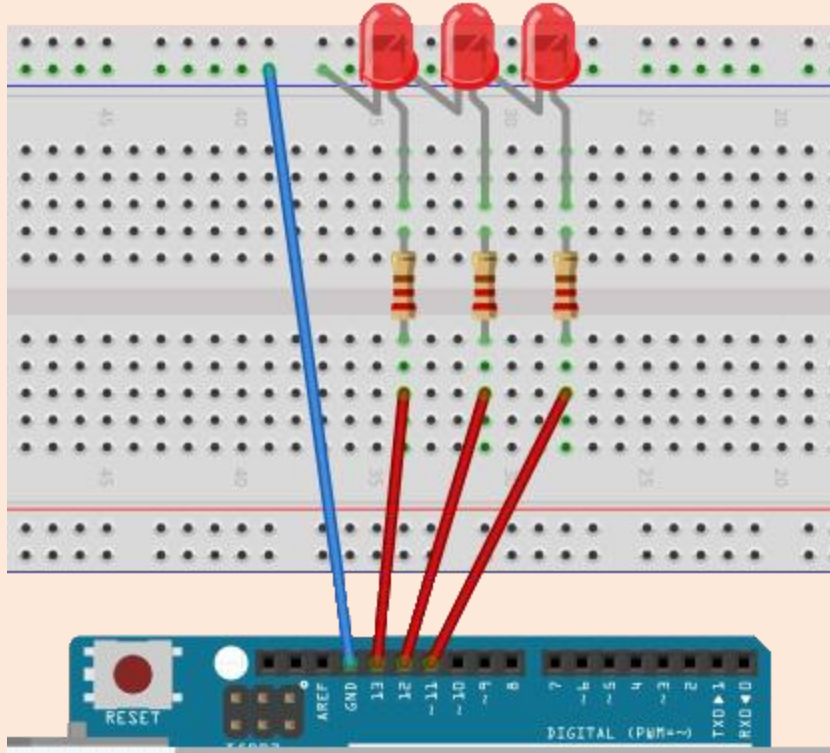
Gerekli Malzemeler

- 1 Adet Arduino Uno
- 1 Adet breadbord
- 1 Adet Led
- 1 Adet 220 Ohm ya da 330 Ohm direnç

220 Ohm direncin rengi kırmızı, kırmızı, kahverengidir. 330 Ohm direncin ise rengi turuncu, turuncu, kahverengidir.

```
tıklandığında  
sürekli tekrarla  
sayısal pim çıkışını 10 'i YÜKSEK'e ayarla  
1 saniye bekle  
sayısal pim çıkışını 10 'i DÜŞÜK'e ayarla  
1 saniye bekle  
↴
```

Kara Şimşek Devresi



tıklandığında

sürekli tekrarla

sayısal pim çıkışı 13 'i YÜKSEK'e ayarla

0.3 saniye bekle

sayısal pim çıkışı 13 'i DÜŞÜK'e ayarla

sayısal pim çıkışı 12 'i YÜKSEK'e ayarla

0.3 saniye bekle

sayısal pim çıkışı 12 'i DÜŞÜK'e ayarla

sayısal pim çıkışı 11 'i YÜKSEK'e ayarla

0.3 saniye bekle

sayısal pim çıkışı 11 'i DÜŞÜK'e ayarla

sayısal pim çıkışı 12 'i YÜKSEK'e ayarla

0.3 saniye bekle

sayısal pim çıkışı 12 'i DÜŞÜK'e ayarla

Gerekli Malzemeler

1 Adet Arduino Uno

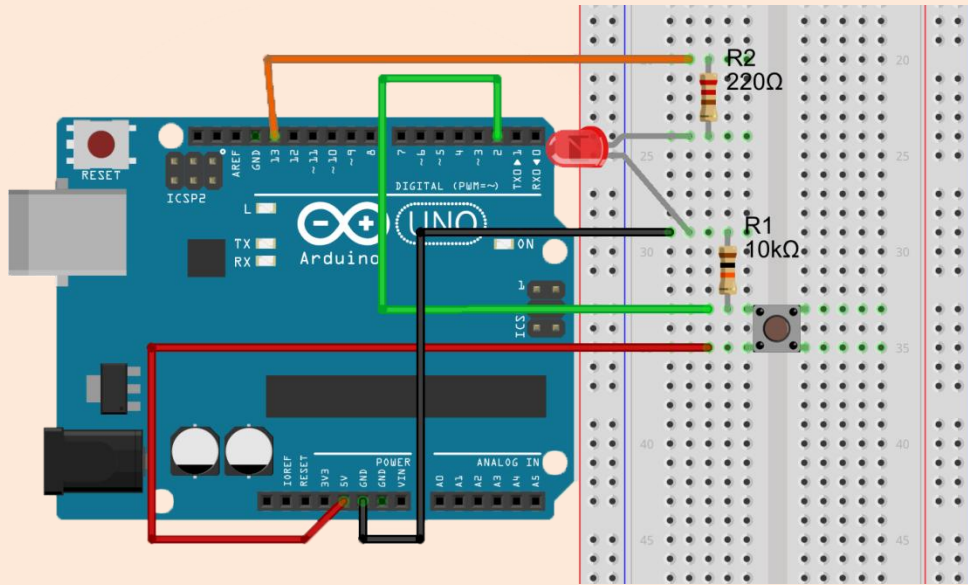
1 Adet breadbord

3 Adet Led

3 Adet 220 Ohm ya da 330 Ohm direnç

Dijital Pinlerden Deęer Okuma

Arduino ile dijital pinlerden akım verebildiđimiz gibi gelen akımları da okuyabiliyoruz. Burada bir adet buton kullanarak buton basıldıđında gelen akımı okuyarak butonun basılı olup olmadıđını anlayacađız.



Gerekli Malzemeler

1 Adet Arduino Uno

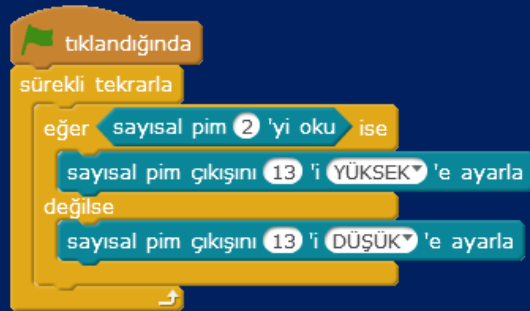
1 Adet breadbord

1 Adet Led

1 Adet buton

1 Adet 220 Ohm ya da 330 Ohm direnç

1 Adet 10 Kilo Ohm direnç (kahverengi,siyah,turuncu)

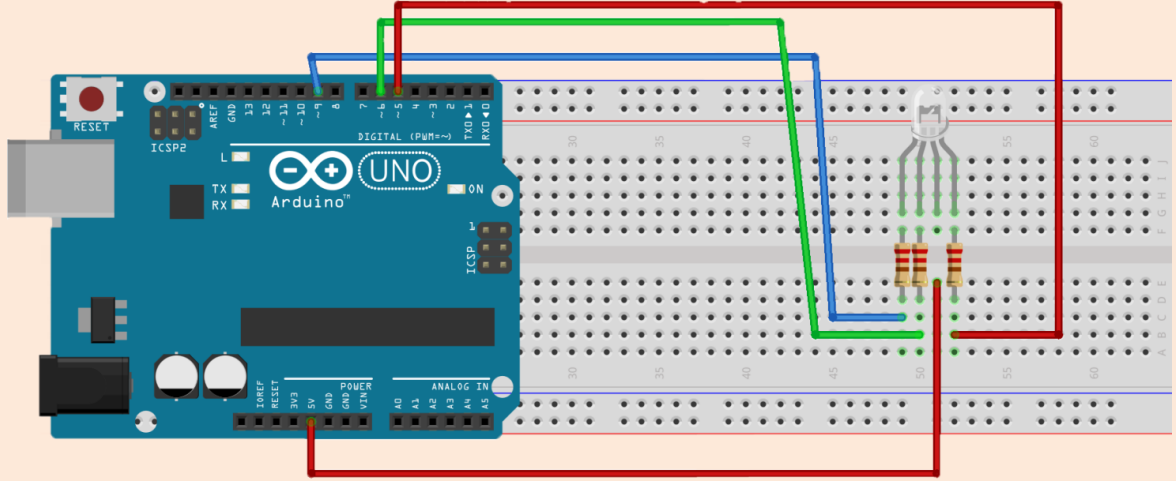


Youtube Videosu İin Tıklayınız!



RGB Led

Rgb Led'ler üç ana rengi kullanarak(Kırmızı, Yeşil, Mavi) bütün renklerde yanabilen Led'lerdir.



Gerekli Malzemeler

1 adet Arduino Uno

1 adet Breadboard

1 adet RGB Anot(+) Led

3 adet 220Ω direnç

4 adet Jumper kablo

İki tür RGB Led var. Anot(+) ve Katot(-). Biz devremizde 5v'ye bağlanan anot Led kullandık. Led'in uzun ucu 5v'ye bağlanacak. Türkiye'de genellikle anot Led satılır.

m tuşu basılınca

pwm pimi 9 çıkışını 0'e ayarla

pwm pimi 6 çıkışını 255'e ayarla

pwm pimi 5 çıkışını 255'e ayarla

y tuşu basılınca

pwm pimi 9 çıkışını 255'e ayarla

pwm pimi 6 çıkışını 0'e ayarla

pwm pimi 5 çıkışını 255'e ayarla

k tuşu basılınca

pwm pimi 9 çıkışını 255'e ayarla

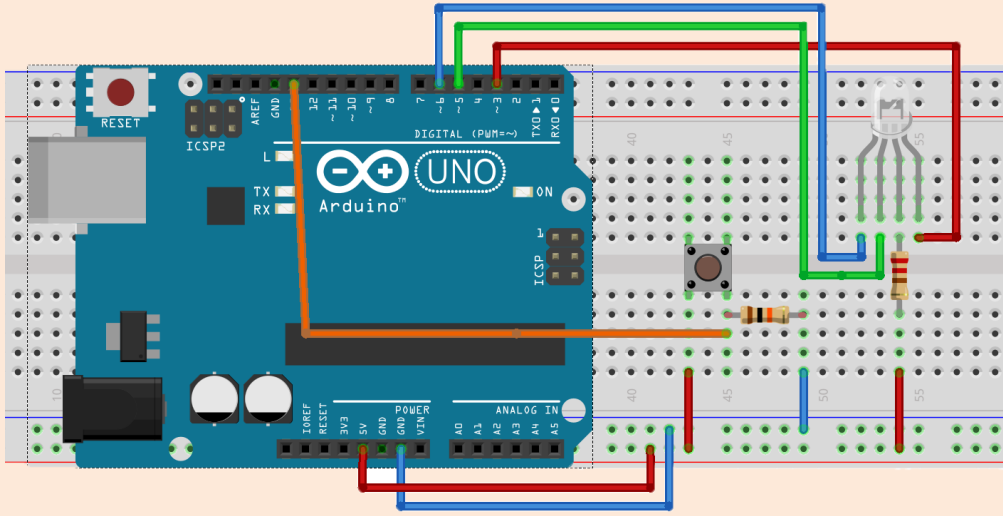
pwm pimi 6 çıkışını 255'e ayarla

pwm pimi 5 çıkışını 0'e ayarla

Youtube Videosu için Tıklayınız!



Buton ile RGB Led Kontrolü



```
graph TD
    Start([tıklandığında]) --> Init[x, 0 olsun]
    Init --> Loop[sürekli tekrarla]
    Loop --> Pin13{13 sayısal pini oku = 1}
    Pin13 -- ise --> Mod3_0{x modüler 3 = 0}
    Mod3_0 -- ise --> Red[tanımla kırmızı]
    Red --> PWM3_0[3 pwm pini 0 yap]
    PWM3_0 --> PWM5_255[5 pwm pini 255 yap]
    PWM5_255 --> PWM6_255[6 pwm pini 255 yap]
    Mod3_0 -- değil --> Pin13
    Pin13 -- değil --> Mod3_1{x modüler 3 = 1}
    Mod3_1 -- ise --> Green[tanımla yeşil]
    Green --> PWM3_255[3 pwm pini 255 yap]
    PWM3_255 --> PWM5_0[5 pwm pini 0 yap]
    PWM5_0 --> PWM6_255[6 pwm pini 255 yap]
    Mod3_1 -- değil --> Pin13
    Pin13 -- değil --> Mod3_2{x modüler 3 = 2}
    Mod3_2 -- ise --> Blue[tanımla mavi]
    Blue --> PWM3_255[3 pwm pini 255 yap]
    PWM3_255 --> PWM5_255[5 pwm pini 255 yap]
    PWM5_255 --> PWM6_0[6 pwm pini 0 yap]
    Mod3_2 -- değil --> Pin13
    Pin13 -- değil --> Wait1[13 sayısal pini oku = 1 değil olana kadar bekle]
    Wait1 --> Inc[x i 1 arttır]
    Inc --> Wait0[0.1 saniye bekle]
    Wait0 --> Loop
```

Bu örneğimizde butona 1. tıklayışımızda RGB Led'imizi kırmızı 2. tıklayışımızda yeşil 3. tıklayışımızda ise mavi renkte yanacağız.



Bir eğer ise bloğu ile butona basılıp basılmadığını kontrol ediyoruz. Butona basıldığında komutlarımız çalışmaya başlıyor. Olana kadar bekle bloğuyla da butona basılma olayının değilini yani butona basılı olmama durumunu kontrol ediyoruz. Butondan elimizi çekinceye kadar program olana kadar bekle kısmında takılı kalıyor. Elimizi butondan çektiğimizde oluşturduğumuz X değişkeninin değerini 1 arttırıyoruz. Araya da 0.1 saniye bekleme koyuyoruz ki işlem kolayca yapılabilsin.

Butona her bastığımızda X değişkeninin değeri 1 artıyor.

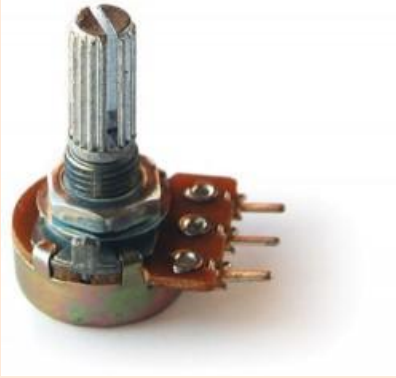


3 tane eğer ise bloğuyla da X değişkeninin 3'e bölümünden kalan değerine göre RGB Led'imizi kırmızı, yeşil ve mavi renklerde yakıyoruz.

Butona ilk tıkladığımızda X'in 3'e bölümünden kalan 0 olduğu için Led'imiz kırmızı yanar ve elimizi butondan çektiğimizde X'in değeri 1 artar.

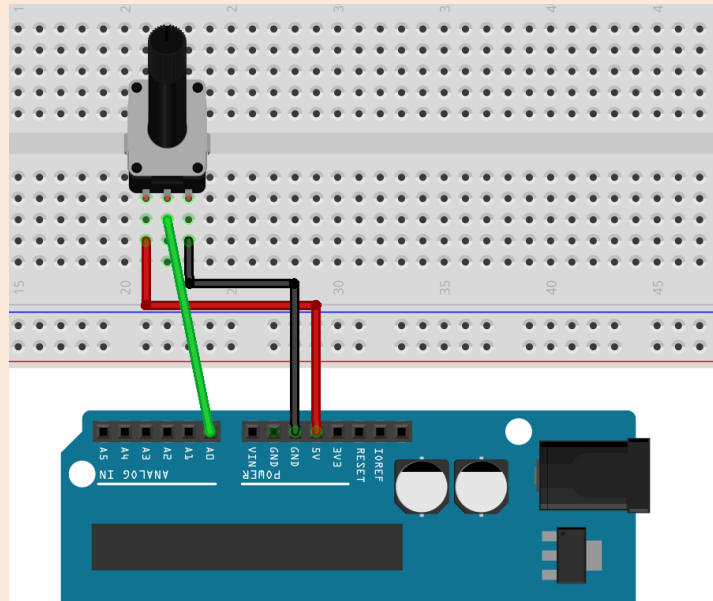
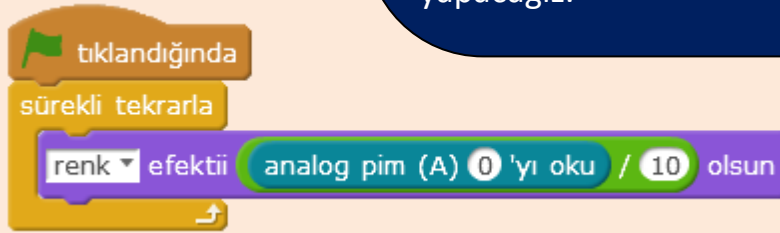
Butona 2. kez bastığımızda kalan 1, 3. Kez bastığımızda ise kalan 2 olur ve ilgili Ledler yanar.

Potansiyometre



Potansiyometreler, ayarlanabilir dirençlerdir. Kullandığımız cihazlardaki sesi arttırıp azaltmaya yarayan düğmeler potansiyometre ile yapılır. Potansiyometreyi ayarlayarak direnci azaltırız ses artar, direnci arttırırız ses azalır.

Potansiyometrenin üç bacağı vardır. En soldaki bacağı (+) ya , orta bacağı analog 0'a, en sağdaki bacağı ise (-) ye bağlayarak uygulamalarımızı yapacağız.



Uzay Savaşı Oyunu

Bu oyunumuzda bir adet potansiyometre ve bir adet buton kullanacağız. Potansiyometre ile karakterimizin hareketini kontrol ederken buton ile uzay gemisinin ateş etmesini sağlayacağız.

Potansiyometreyi **A0** pinine, butonu da **D2** pinine bağlayınız. Kaynak dosyalarda bulunan uzayoyunu.sb2 dosyasını açınız. Kaynak kodların hepsi burada irdelenmeyecektir.

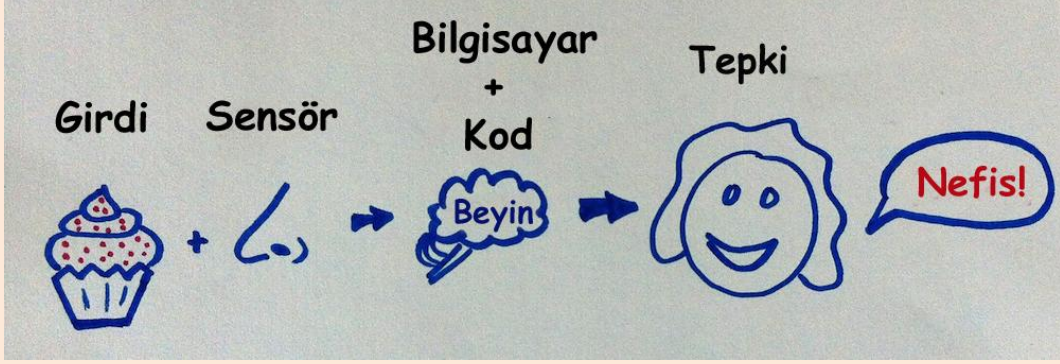
Potansiyometre ile Analog pinden 0-1023 arası bir değer okuruz. Potansiyometreyi döndürdüğçe okunan değer değişecektir ve bu değişen değere göre Potansiyometre'nin hangi yöne döndüğünü ya da hangi konumda olduğunu belirleyebiliriz. Biz bu değeri okuyarak karakterin X konumunu kontrol etmekte kullanacağız.



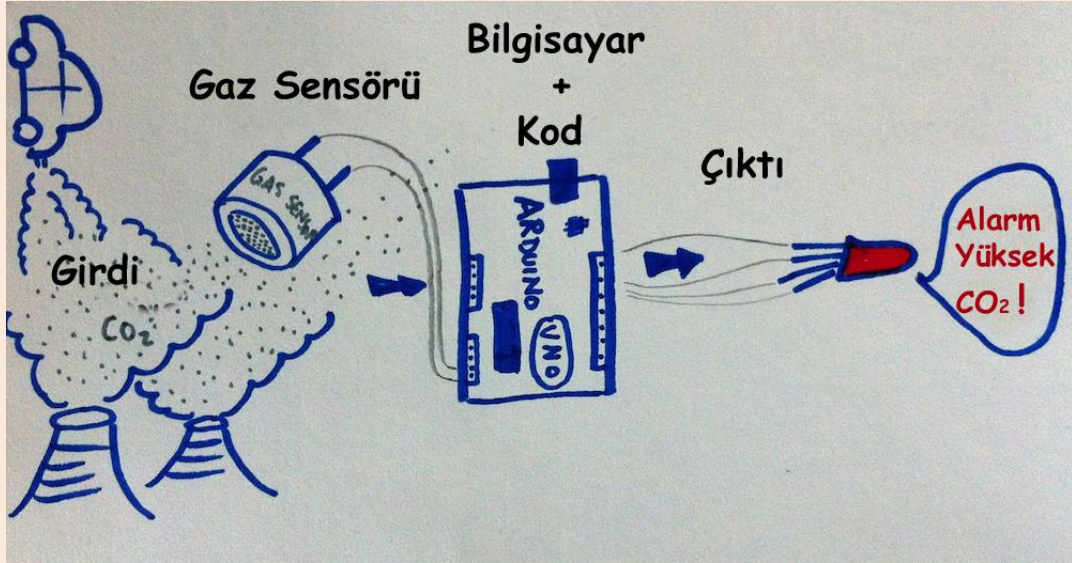
Potansiyometre'nin konumunu en sola getirdiğinizde Analog pinden 1023 değeri okunurken en sağa getirdiğinizde ise 0 değeri okunur. Analog pinden okunan değeri 2.7'e bölüp 190'dan çıkarıyoruz. Potansiyometre en soldayken Analog pinden 1023 değeri okunur 1023 değerini 2.7'e böldüğümüzde 379 gibi değer gelecektir. 190'dan 378 değeri çıkarıldığında yaklaşık -190 değeri bulunur. Karakterin X konumu -190 olur. Potansiyometre sağa doğru çevrildikçe Potansiyometreden gelen değer azalacaktır ve karakter sağa doğru hareket edecektir.

Sensörler

İnsanlar çevrelerindeki dünyayı ve çevrelerinde olup bitenleri algılamak için duyu organlarını kullanır.

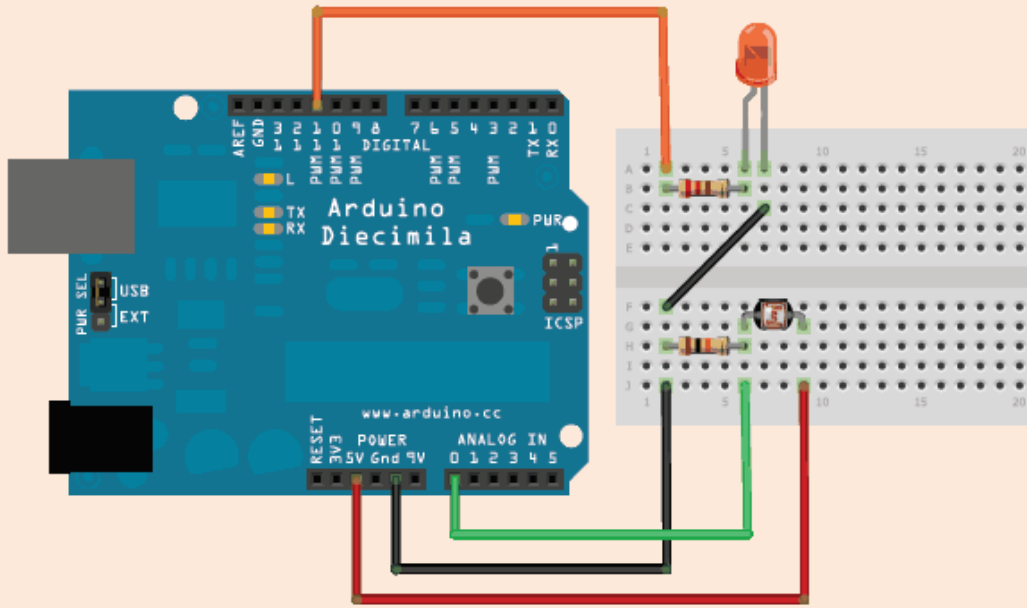


Makineler (Bilgisayarlar, Robotlar) ise çevrelerinde olup bitenleri algılamak için sensörleri kullanır.



Hava Kararınca Yanan Led

LDR'ler ışığa duyarlı dirençlerdir. Aydınlıkta dirençleri minimum, karanlıkta ise maksimumdur. Biz bu özelliğini kullanarak gece olduğunda yani karanlıkta yanan Led yapacağız. Hava kararınca Ldr'nin direnci maksimum olacak ve yazacağımız kod ile bu durum gerçekleştiğinde Led



Gerekli Malzemeler

1 adet Arduino uno
bağlanacak)

1 adet breadboard

1 adet Ldr(foto direnç)

1 adet 220Ω direnç (Led'e

1 adet 10000Ω (10KΩ) direnç
(Ldr'ye bağlanacak)

1 adet Led



Eğer sensör değeri 500'den küçük ise, Led'imizin bağlı olduğu digital 11 pini aktif olacak yani Led yanacak. Sensör değeri 500'den küçük değilse, digital 11 pini pasif olacak, giden akım kesilecek, Led sönecek.

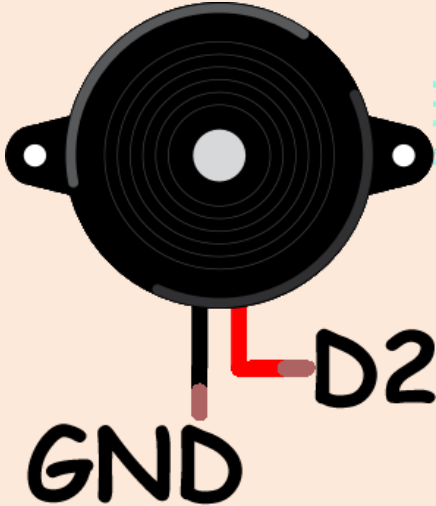
Bu uygulamada LDR'nin bir ucu Arduinonun analog 0 pinine bağlıdır ve buradan değeri okumaktadır. Bu değeri "analog pin A(0) (analog 0 pinine bağlı sensörün değeri) " komutu ile kontrol ederiz.

Bu pinden 0 ile 1023 arası değerler okunur. 1023 aydınlık, 0 karanlık. Işığın değerine göre de 0-1023 arası değer okunur.

Burada bir sürekli bloğunun içerisinde bir eğer ise değilse yapısı kullandık. Bir koşul belirledik. Sensör değeri 500'ün altında mı , değil mi?

Buzzer Kullanımı

Buzzerlar elektronik devrelerde uyarı sesleri vermek için kullanılır.



Buzzerları ilgili pine akım vererek kullanabileceğimiz gibi özel oluşturulmuş kod bloğunu da kullanarak istediğimiz notada ses üretebiliriz.

9 ses tonu pini C4 notasında Yarım vuruş çal

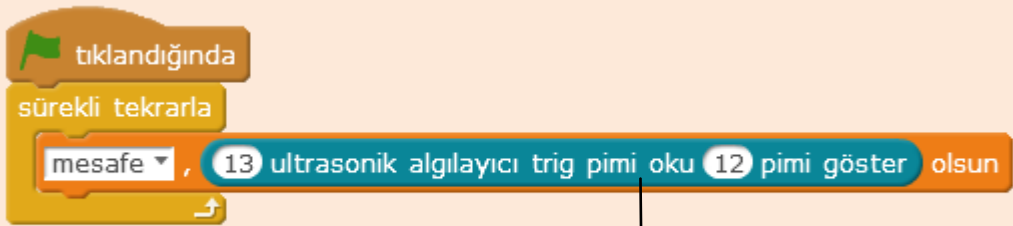
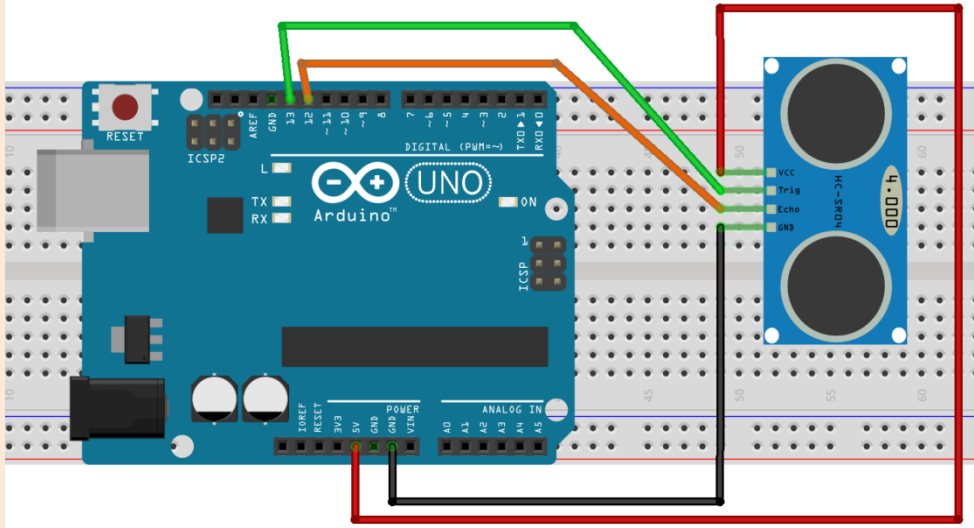
Pin Numarasını buradan değiştirerek 2 yapınız!

Buzzerı istediğiniz pine bağlayabilirsiniz.

Mesafe Sensörü

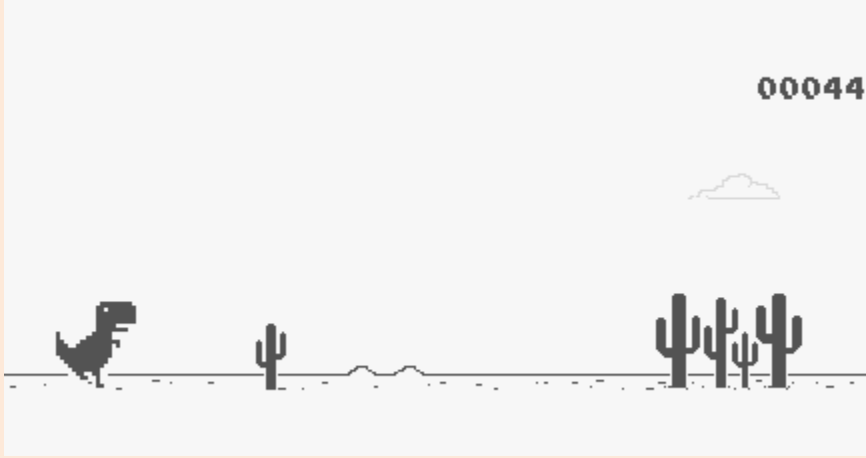


Ultrasonik mesafe sensörünü projelerimizde cisimlerle olan mesafeleri ölçmek için kullanırız. Engelleri algılayıp aşan robotlarda sıkça kullanılır.



Ultrasonik sensörden okunan değeri tutan kod bloğu

Zıplama Oyunu



Google Chrome web tarayıcı programında internet olmadığında gelen bir dinazor oyunu vardır. Gelin hep birlikte bu oyunun değişik bir versiyonunu yapalım. Ultrasonik mesafe sensörünü zıplama mesafesine koyalım ve biz zıpladığımızda sensörden okunan değerin düşmesine göre oyundaki karakteri de zıplatalım.

Ultrasonik mesafe sensörünün bağlantılarını yapalım ve kitabın başında verilen kaynak dosyalarının içindeki **dinazorgamemegauno.sb2** dosyasını Mblock programında açalım. Mesafe sensörünü zıplama mesafesine sabitleyelim ve oyunun keyfini çıkaralım.

Not: Trig Pin: 13 , Echo Pin: 12

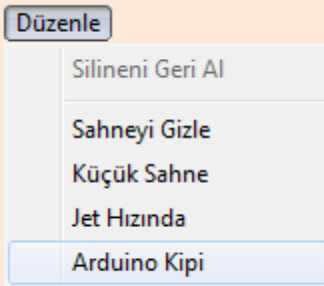
Oyunun Örneğini İzlemek için Tıklayınız!



Scratch ile Yazdığımız Programı Arduino'ya Yükleme

Mblock programı ile yazdığımız programlar normalde bilgisayar üzerinden çalışır. Program çalışmayı durdurduğunda ya da bilgisayarı kapattığımızda Arduino programı da durur. Eğer bilgisayar açık değilken de Arduino programının çalışmasını istiyorsanız yazdığımız kodları Arduino'ya yüklemeniz gerekmektedir.

Arduino'ya kodları yüklemek için öncelikli olarak düzenle menüsünden Arduino kipini seçelim.



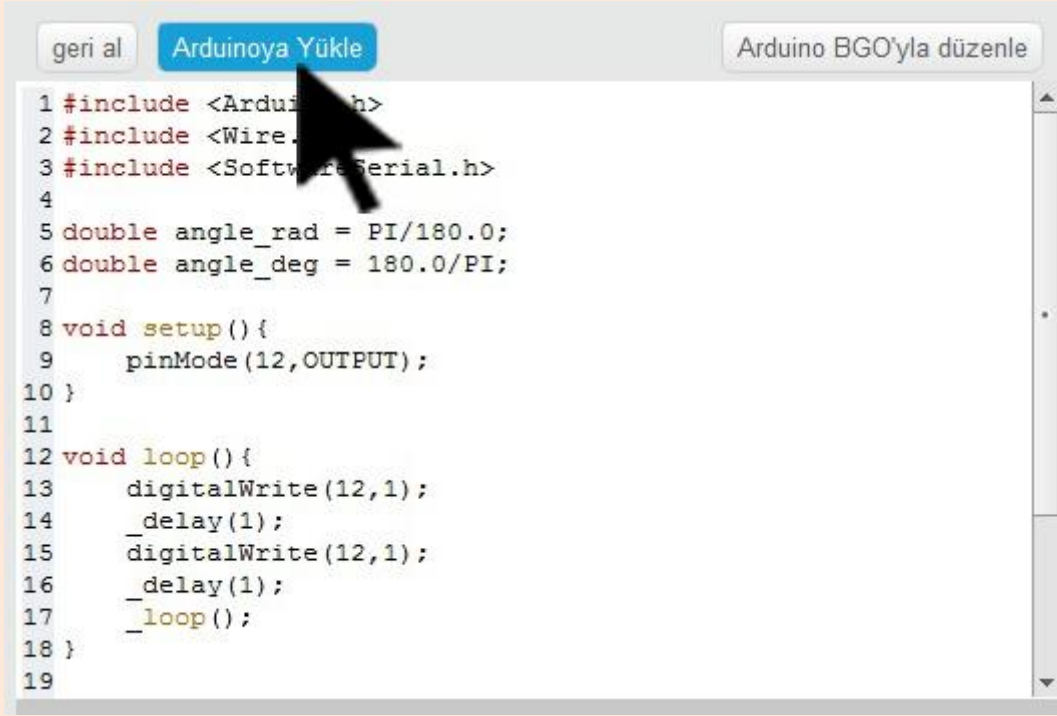
Arduino kipini seçtiğinizde bazı kod bloklarının aktif olmadığını görebilirsiniz. Arduino kipi seçiliyken bütün kod bloklarını kullanamıyoruz. Sadece aktif olan kod bloklarını kullanabiliyoruz.

Arduino kipinde program yazarken başlangıç olarak yeşil bayrak yerine **Arduino programını** kullanıyoruz.



NOT: Yazacağımız bütün programlar tek bir Arduino programı altında olmalıdır.

Programımızı yazdıktan sonra **Arduino'ya yükle** butonu ile Arduino karta yüklüyoruz.



```
1 #include <Arduino.h>
2 #include <Wire.h>
3 #include <SoftwareSerial.h>
4
5 double angle_rad = PI/180.0;
6 double angle_deg = 180.0/PI;
7
8 void setup() {
9   pinMode(12, OUTPUT);
10 }
11
12 void loop() {
13   digitalWrite(12, 1);
14   _delay(1);
15   digitalWrite(12, 1);
16   _delay(1);
17   _loop();
18 }
19
```

Arduino'ya yükle dediğimizde Arduino'nun kendi diline çevrilmiş programı Arduino'ya yüklenecektir.

NOT: Arduino'ya program yüklerken bağlan menüsünden kartın bağlı ve kartlar menüsünden Arduino Uno seçili olduğundan emin olun!

NOT: Arduino'ya program yüklerken kullandığınız değişken isimlerinde ve özel blok isimlerinde Türkçeye has karakterler (ş,ç,ğ,ö,ü,ı) kullanmayınız!

Youtube Videosu İçin Tıklayınız! 

Su Seviyesi ve Yağmur Sensörü



Su seviyesi ve yağmur sensörü ile yağmur yağıp yağmadığını ya da belli bir ölçüde suyun seviyesini kontrol edebiliriz.

Su seviyesi sensörü Analog çıkış verdiği için Analog pinlerden birine bağlanmalıdır.

Sensör üzerinde bulunan pinlerden - pinini kart üzerindeki GND pinine, + pinini kart üzerindeki VCC pinine, S yazan pini de Analog 0 pinine (A0) bağlayacağız. Bağladığımız sensörün yolladığı değerleri Analog pinden okuyacağız.

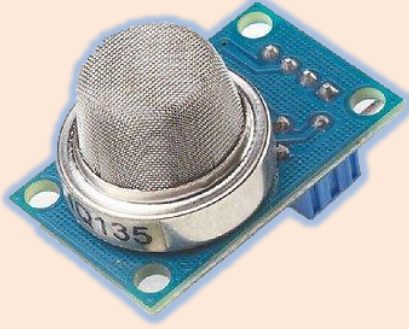
Bunun için sensör adında bir değişken oluşturalım ve bu değişkene Analog 0 pininden okuduğumuz değeri atayalım.

Sensör , 0 olsun
(A) 0 analog pini oku

tıklandığında
sürekli tekrarla
sensör , analog pim (A) 0 'yı oku olsun

Su seviyesi düşükken ya da hiç su yokken sensörden 0 değeri okunur. Su seviyesi arttıkça bu seviye de artar.

Hava Kalite Sensörü (MQ135)



MQ135 hava kalite sensörü havadaki duman ve CO₂ miktarını ölçen bir sensördür.

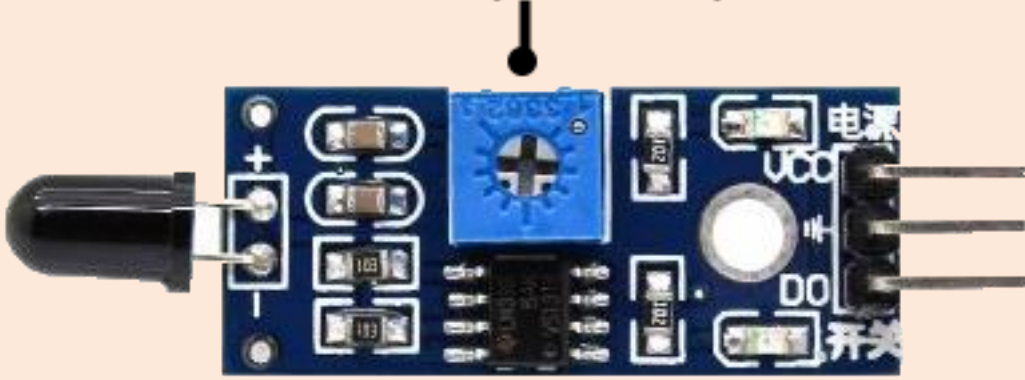
Üzerinde A0, D0, GND, VCC pinleri mevcuttur.

GND ve VCC pinlerini bağladıktan sonra A0 yazan pini Analog pinlerden birine bağlayınız ve sensörün değerini okuyunuz.

NOT: Sensörü bağladıktan sonra ısınması için biraz bekleyiniz. Okunan değer sabitlendiğinde artık sensörü kullanabilirsiniz. Havadaki CO₂ miktarı arttığında ya da duman algılandığında sensörden okunan değer artacaktır. Artan değere göre gerekli kodları yazarak istediğiniz gibi kullanabilirsiniz.

Alev Sensörü

Hassasiyet Ayarı



Alev sensörü ateşin dalga boyunu algılayabilir böylece menzil mesafesinde ateş olup olmadığını algılar.

Üzerinde 3 adet pin vardır. GND, VCC, D0.

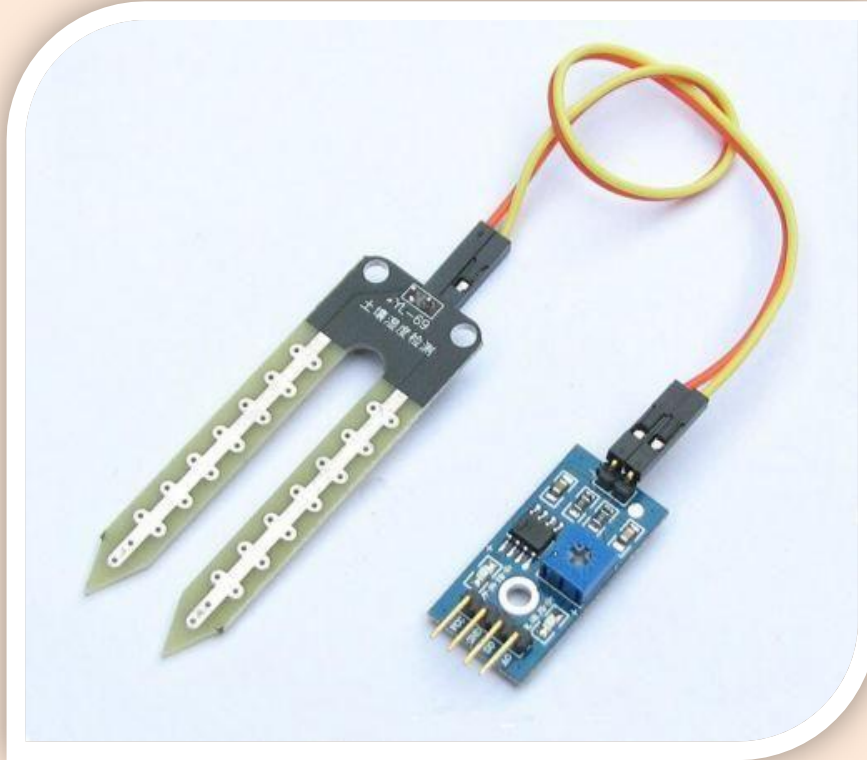
GND ve VCC pinlerini bağladıktan sonra D0 pinini Arduino kartı üzerindeki D9 pinine bağlayınız.



Sensörü bağladığımız dijital pinden sensörün değerini okuduğumuz kod bloğu.

NOT: Sensör alev algılandığında dijital pinden 0 değeri, alev algılanmadığında ise dijital pinden 1 değeri gönderir.

Toprak Nem Sensörü

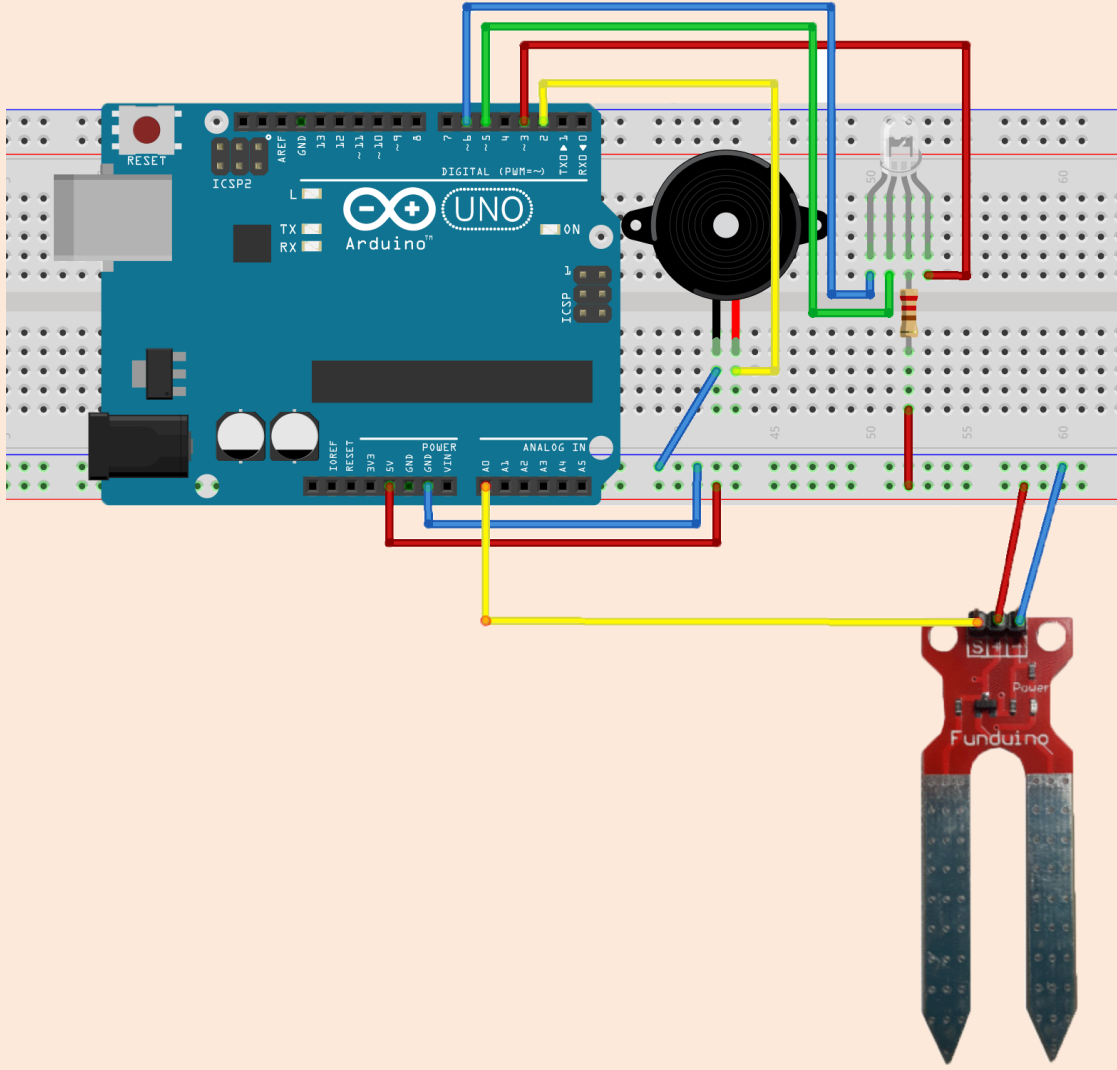


Toprak nem sensörünü topraktaki su miktarı ölçmek için kullanırız.

Sensörün uç kısmı toprağa saplanacaktır. Uç kısmına iki kablo aracılığıyla bağladığımız kontrol devresi uç kısımdan aldığı değerleri değerlendirerek arduino karta yollar.

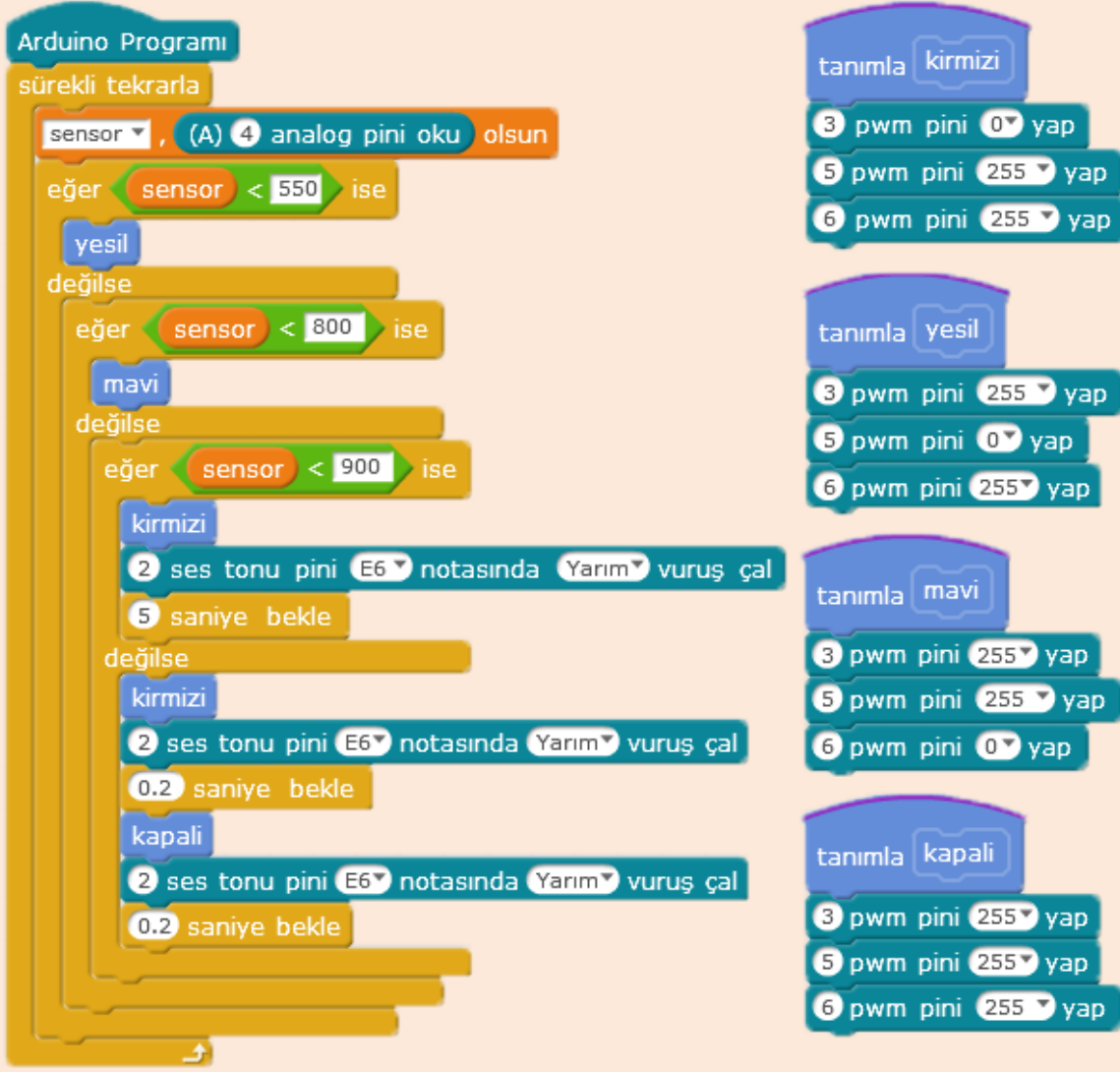
GND pinini Arduino kartın üzerindeki GND pinine, VCC pinini VCC pinine, A0 yazan pinide A0 pinine bağlayınız.

Eğer topraktaki nem oranı yüksekse 500'e yakın bir değer okunur. Nem oranı düşükçe okunan değer artar. En fazla 1023 değeri okunur.



Sensörden okunan değere göre. Nem oranı iyi olduğunda RGB Led'in yeşil, orta olduğunda RGB Led'in mavi, kötü olduğunda da RGB Led'in kırmızı yandığı ve Buzzer'ın öttüğü bir program yapalım.

Bu program için iç içe eğer ise değilse yapıları kullanacağız. İlk eğer ise bloğunda sensör değerinin 550'de küçük olmasını kontrol edeceğiz. Eğer değer 550'den küçükse nem oranı iyidir. RGB Led yeşil yanar. Değilse kısmına (değer 550'den küçük değilse büyüktür.) bir eğer ise değilse bloğu daha koyuyoruz. Bu eğer ise değilse bloğuyla da değer 800'den küçük olup olmadığını ölçüyoruz. Eğer değer 550'den büyük 800'den küçükse RGB Led mavi yanar.



Devam ediyoruz ve değilse kısmına bir eğer ise değilse daha koyuyoruz. Bu şekilde iç içe üç tane eğer ise değilse koymuş olduk. Bu eğer ise değilse kısmında da değer 900'den küçük olup olmadığını kontrol ediyoruz. Eğer değer 800'den büyük ve 900'den küçükse RGB Led kırmızı yanıyor.

En son değilse kısmına yani değer 550'den küçük değil, 800'den küçük değil, 900'den küçük değilse RGB Led'i yakıp söndürüyoruz ve Buzzer'ı öttürerek uyarı vermesini sağlıyoruz.

Bu programı Arduino karta yüklüyoruz.

NOT: Arduino'ya bir program yükledikten sonra tekrar Arduino kartı bilgisayar üzerinden Scratch ile programlamak için **Aygıt Yazılımı Güncellemesini** tekrar yapınız!

NOT: Programdaki değerleri kendi kullandığınız toprağa göre düzenleyebilirsiniz.

Servo Motor



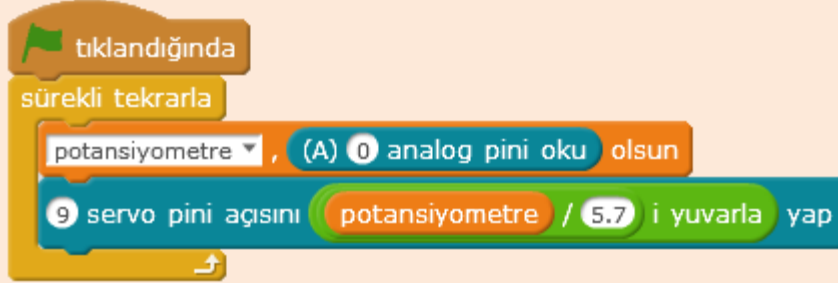
Servo motorlar açılı dönebilen motorlardır. Servo motoru 0-180 arasında istediğimiz bir açığa ayarlayarak döndürebiliriz.

Servo motor üzerinde üç adet kablo bulunur. Kahverengi kablo GND pinine, Kırmızı kablo VCC pinine, Sarı kablo ise D9 pinine bağlanmalıdır.

9 servo pini açısını 90° yap

Yukarıdaki kod bloğuyla da servo motoru istediğimiz bir açığa ayarlayabiliriz.

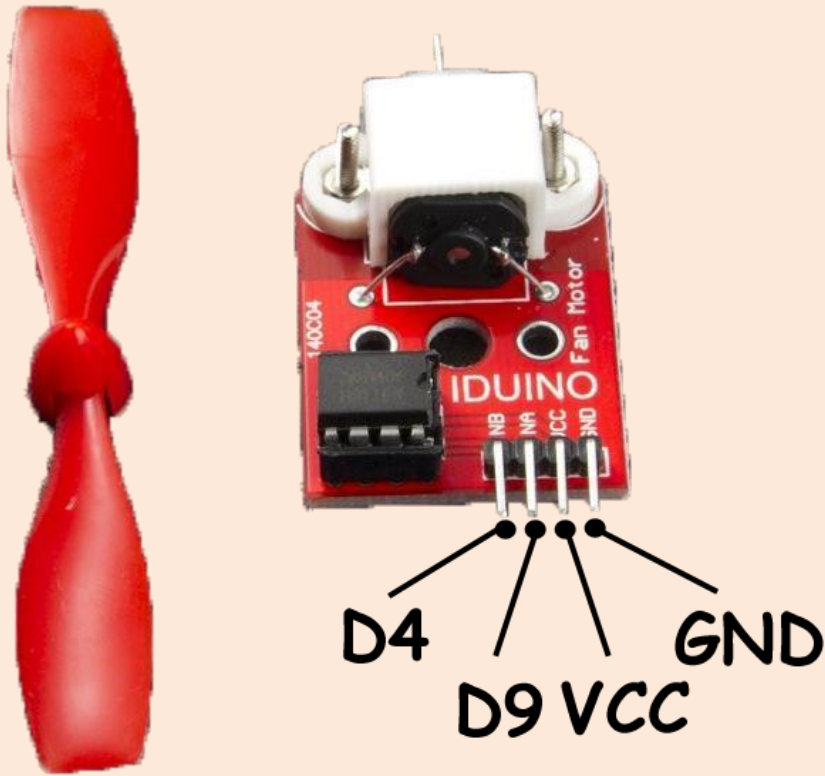
Potansiyometre ile Servo Kontrolü



Potansiyometre Arduino kartın Analog 0 pinine bağlıdır. Bu pinden okuduğumuz değeri oluşturduğumuz potansiyometre değişkenine aktarıyoruz.

Potansiyometreden 0-1023 arası değerler okunmakta. Bize ise 0-180 arası değerler gerekmektedir. Bu yüzden potansiyometreden okuduğumuz değeri direkt kullanmak yerine 0-180 arası bir değere dönüştürmemiz gerekiyor. Bunun içinde potansiyometreden okuduğumuz değeri 5.7 değerine bölüyoruz. Küsurat olmaması için de yuvarlıyoruz. Bu elde ettiğimiz potansiyometreden gelen değeri de servo motor için açı değerinin olduğu yere sürükleyip bırakıyoruz.

L9110 Pervaneli Motor



L9110 pervaneli motor üzerinde oyuncak arabalarda olan motorlardan bir tane vardır. Ucundaki pervane yardımıyla da projelerimizi gerçekleştirmek için çok uygun.

Üzerindeki GND pinini Arduino kartın üzerindeki GND pinine, VCC pinini VCC pinine, INA pinini D9 pinine, INB pinini de D4 pinine bağlayınız.

sağ ok ▾ tuşu basılınca
9 sayısal pini YÜKSEK ▾ yap
4 sayısal pini DÜŞÜK ▾ yap

sol ok ▾ tuşu basılınca
9 sayısal pini DÜŞÜK ▾ yap
4 sayısal pini YÜKSEK ▾ yap

any ▾ tuşu bırakılınca
9 sayısal pini DÜŞÜK ▾ yap
4 sayısal pini DÜŞÜK ▾ yap

Motoru kontrol etmek için iki adet dijital pin kullanıyoruz. Dijital pinlerden bir tanesi **Yüksek** diğeri **Düşük** olduğunda bir yöne döner. **Yüksek** ve **Düşük** değerli pinlerin yeri değiştirildiğinde ise ters yöne yöner. Motoru durdurmak içinse iki dijital pinde **Düşük** olmalıdır.

NOT: Mblock'ta Scratch'a ek olarak Tuşu basılincaya ek olarak Tuşu bırakılınca bloğu da vardır. Çünkü tuşa bastığımızda motorun çalışması için sinyal gönderildikten sonra elimizi tuştan çektiğimizde motorun durması için sinyal gitmezse motor dönmeye devam eder. Bu örnekte any tuşu bırakılınca (herhangi bir tuş) motor durur.

NOT: Motoru taktığınızda motor hemen çalışmaya başlayabilir. Dikkatli olunuz!

Potansiyometre ve Pervaneli Motor

Şimdi potansiyometre ve butonu kullanarak motorun dönme yönünü belirleyeceğimiz bir uygulama geliştirelim. Potansiyometrenin konumu 0 iken butona basılırsa motor dursun. Potansiyometreyi biraz çevirdiğimizde D7 pinine bağlı Led yansın ve butona bastığımızda motor bir yöne dönsün. Potansiyometreyi biraz daha çevirdiğimizde D7 ve D8 pinlerine bağlı Ledler yansın ve butona bastığımızda motor tam ters yöne dönsün. Bu şekilde motoru kontrol edeceğimiz basit bir program yapalım.

NOT: Bu uygulama için D7 ve D8 pinlerine birer Led, A0 pinine bir potansiyometre ve D13 pinine bir buton bağlayınız. İsterseniz pinleri değiştirebilirsiniz.

Arduino Programı

sürekli tekrarlar

```
eğer (A) 0 analog pini oku < 100 ise
  7 sayısal pini DÜŞÜR yap
  8 sayısal pini DÜŞÜR yap
  mod , 0 olsun
değilse
  eğer (A) 0 analog pini oku < 300 ise
    7 sayısal pini YÜKSER yap
    8 sayısal pini DÜŞÜR yap
    mod , 1 olsun
  değilse
    7 sayısal pini YÜKSER yap
    8 sayısal pini YÜKSER yap
    mod , 2 olsun
eğer 13 sayısal pini oku = 1 ise
  eğer mod = 0 ise
    9 sayısal pini DÜŞÜR yap
    4 sayısal pini DÜŞÜR yap
  eğer mod = 1 ise
    9 sayısal pini DÜŞÜR yap
    4 sayısal pini YÜKSER yap
  eğer mod = 2 ise
    9 sayısal pini YÜKSER yap
    4 sayısal pini DÜŞÜR yap
```

```
eğer (A) 0 analog pini oku < 100 ise
7 sayısal pini DÜŞÜK yap
8 sayısal pini DÜŞÜK yap
mod , 0 olsun
değilse
eğer (A) 0 analog pini oku < 300 ise
7 sayısal pini YÜKSEK yap
8 sayısal pini DÜŞÜK yap
mod , 1 olsun
değilse
7 sayısal pini YÜKSEK yap
8 sayısal pini YÜKSEK yap
mod , 2 olsun
```

Öncelikli olarak iki tane iç içe eğer ise değilse bloğu kullanıyoruz. İlk eğer ise değilse kısmında potansiyometreden okunan değer 100'den küçük olup olmama durumuna bakıyoruz. Eğer değer 100'den küçükse 7 ve 8 no.lu dijital pine bağlı Led'i söndürüyoruz ve oluşturduğumuz mod değişkenini 0 yapıyoruz.

Eğer değer 100'den küçük değilse eğer ise değilse kısmına geçiliyor. Burada da değer 300'den küçük olup olmama durumuna bakılıyor. Eğer değer 100'den büyük 300'den küçük ise 7 nolu dijital pine

bağlı Led'i yakıyoruz ve mod değişkenini 1 yapıyoruz.

Eğer değer 100'den ya da 300'den küçük değilse bu sefer ikinci eğer ise değilse kısmına geçiliyor. 7 ve 8 nolu Ledleri yakıyoruz ve mod değişkenini 2 yapıyoruz.

Bu şekilde potansiyometre ile motorun dönme modunu ayarlıyoruz.

```
eğer 13 sayısal pini oku = 1 ise
  eğer mod = 0 ise
    9 sayısal pini DÜŞÜK yap
    4 sayısal pini DÜŞÜK yap
  eğer mod = 1 ise
    9 sayısal pini DÜŞÜK yap
    4 sayısal pini YÜKSEK yap
  eğer mod = 2 ise
    9 sayısal pini YÜKSEK yap
    4 sayısal pini DÜŞÜK yap
```

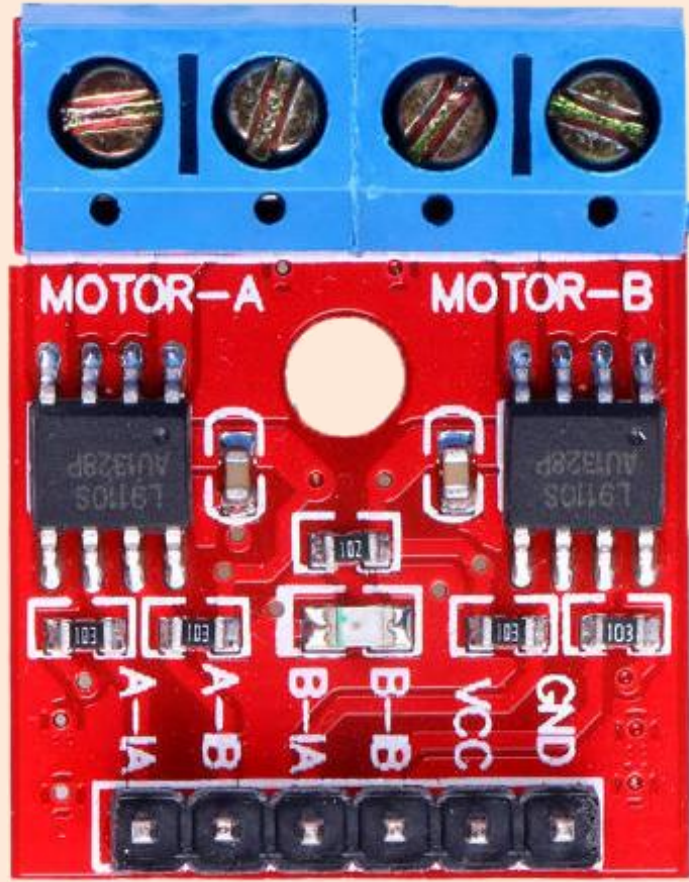
Potansiyometre ile motorun dönme modunu belirledikten sonra buton ile bunu onaylıyoruz.

Butona tıkladığında 13 no.lu dijital pinden 1 değeri okunur.

Yazdığımız eğer ise bloğuyla butonun tıklanmasını kontrol ediyoruz. Buton tıkladığında mod değişkeni 0 ise motor durur. Mod değişkeni 1 ise motor dışa doğru döner. Mod değişkeni 2 ise motor içe doğru döner.

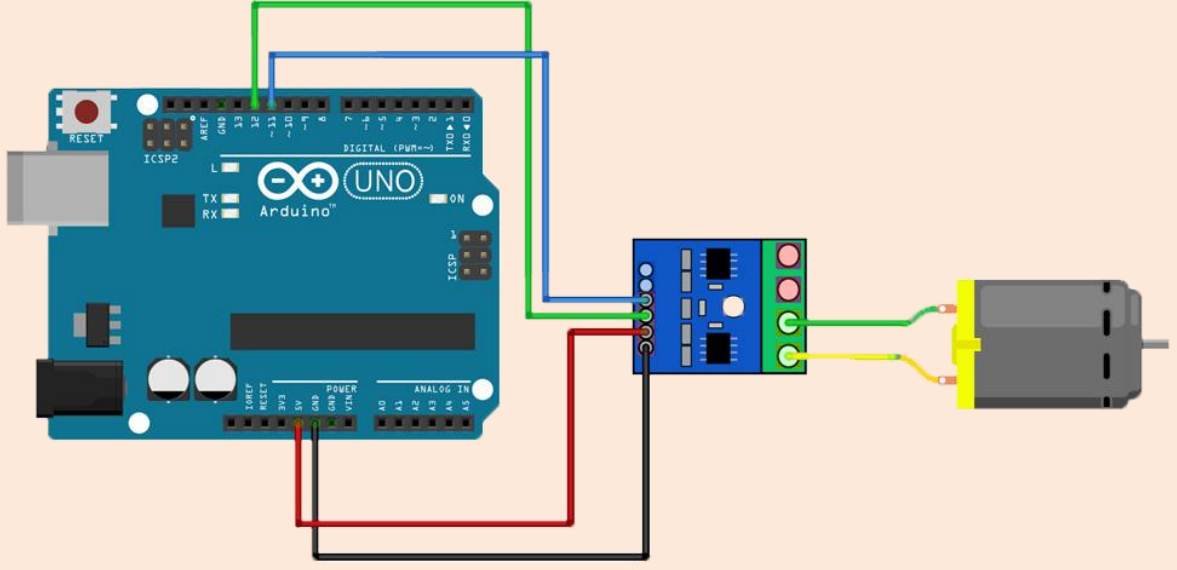
L9110 Motor Sürücü Kartı

Robotları hareket ettirmek için kullandığımız motorları, direkt Arduino karta bağlayıp çalıştıramıyoruz. Arduino'nun gücü buna yetmiyor. Motorları çalıştırmak için ara devrelere ihtiyacımız var. Motor sürücü kartlar bu ihtiyacımızı karşılar.



Çok güç gerektirmeyen işlerinizde L9110 çift motor sürücü kartı kullanabilirsiniz. 2.5V - 12V arası Voltaj değerlerinde çalışır. Oyuncak araba motorlarını bağlayabilirsiniz. Toplamda iki adet motor bağlayabilirsiniz.

A-IA ve A-IB girişleri Motor-A için, B-IA ve B-IB girişleri Motor-B kontrolü için kullanılır. İki motor girişini de düşük yaparsanız motor durur. İki motor girişinden birini yüksek diğerini düşük yaparsanız motor bir yöne döner. Girişlerin yüksek, düşük sırasını değiştirirseniz diğer yöne döner.



L9110 Motor Sürücü Kartı Örnek Bağlantı