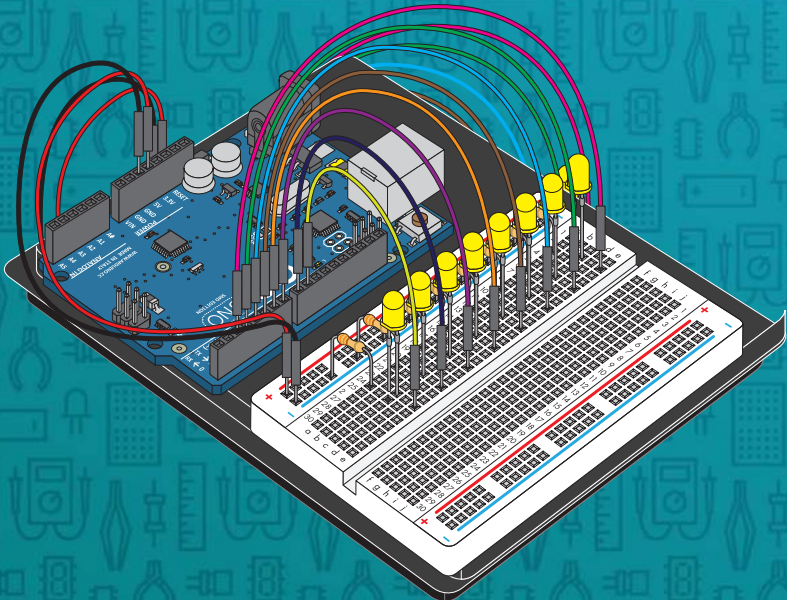


AKADEMİKPORT

Arduino Başlangıç Projeleri



AkademikPort "Arduino Başlangıç Projeleri" Eğitimine Hoşgeldiniz!

Bu kitapçık **SparkFun SIK Guide(retired)** ve bir çok yerli-yabancı kaynaktan yararlanılarak oluşturulmuş ve açık kaynaklı olarak ilk lisans şartı olan **Creative Commons CC BY SA 4.0** lisansı ile yayınlanmıştır. "**AkademikPort Arduino Başlangıç Projeleri**" Eğitimi **AkademikPort**'un açık kaynaklı elektronik kitap eğitimlerinin yedinci kitapçığı olarak yayınlanmıştır. Bu kitapçıkta sizler için hazırlanan 12 Arduino breadboard devresi sayesinde devre elemanları ve sensörlerle Arduino projeleri geliştirerek kısa süre içerisinde Arduino'yu kullanmayı ve programlamayı öğreneceksiniz.

Üretmekten korkmayan bir toplum olmamız dileğiyle...

Emre ARSLAN

AkademikPort Kurumsal İlişkiler Koordinatörü

www.akademikport.com

Orjinal Kitap Editörleri

SparkFun Education Department

Türkçe Kitap Hazırlayanlar

Emre Arslan

Hakan Çolakoğlu



Arduino Başlangıç Projeleri is licensed under a Creative Commons Attribution-NonCommercial-ShareAlike 4.0 International License.

(Arduino Başlangıç Projelerielektronik kitabı yazar isimleri belirtilip, lisans şartlarına uyularak paylaşılabilir fakat ticari bir faaliyette kullanılamaz.)

To view a copy of this license visit:

<http://creativecommons.org/licenses/by-sa/4.0/>

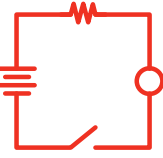
www.akademikport.com



1. Bölüm

Arduino'ya Başlamak

Arduino nedir?	1
Arduino IDE 'yi İndirmek	3
Sürücüyü Yükleme	4
Identify your Arduino	7
"Elektrikport Arduino Kod" indirme	8



2. Bölüm

Temel Elektronik ve Arduino Projeleri

Elektrik Devreleri	9
Devre Elemanları	11
Arduino Uno	13
Breadboard	15
Devre #1 - İlk Devreniz: LED Yakma	17
Devre #2 - Potansiyometre	24
Devre #3 - RGB LED	28
Devre #4 - Çoklu LED	32
Devre #5 - Buton Kullanımı	36
Devre #6 - Foto Direnç	40
Devre #7 - Sıcaklık Sensörü	44
Devre #8 - Servo Motor	48
Devre #9 - Buzzer	52
Devre #10 - Motor Döndürme	56
Devre #11 - Röle	60
Devre #12 - Shift Register	64
Notlar	68

Arduino nedir?



Arduino Çağı Başlıyor

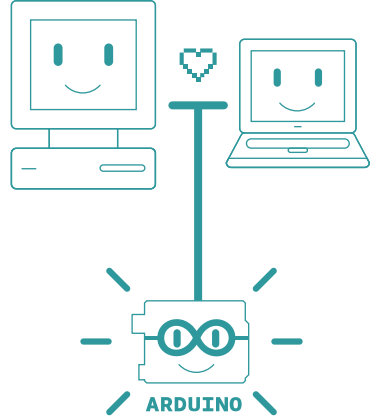
Arduino kolay bir şekilde çevresiyle etkileşime girebilen sistemler tasarlayabileceğiniz, hem acemi hem de ileri düzeydeki elektronik meraklılarına hitap eden, kolayca programlanabilen ve üzerine elektronik devre kurulabilen açık kaynaklı bir geliştirme platformudur.

arduino.cc

Fiziksel Dünya için Bir Bilgisayar

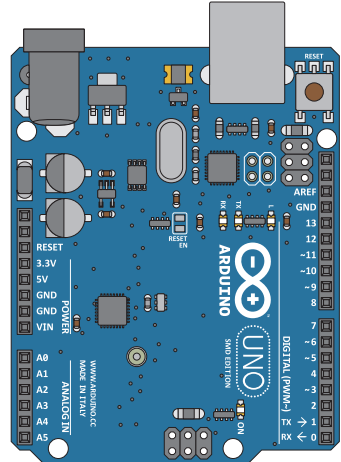
Bu elinizde tuttuğunuz dost canlısı mavi kartı masaüstü bilgisayarlarınızın ve laptoplarınızın çocuğu gibi taşınabilir küçük bir bilgisayar olarak düşünebilirsiniz. Arduino kütüphaneleri ile mikrodenetleyicileri kolaylıkla programlayabilirsiniz. Analog ve dijital girişleri sayesinde analog ve dijital verileri işleyebilir ve sensörlerden gelen verileri kullanarak dış dünyaya çıktılar (ses, ışık, hareket vs...) üretebilirsiniz.

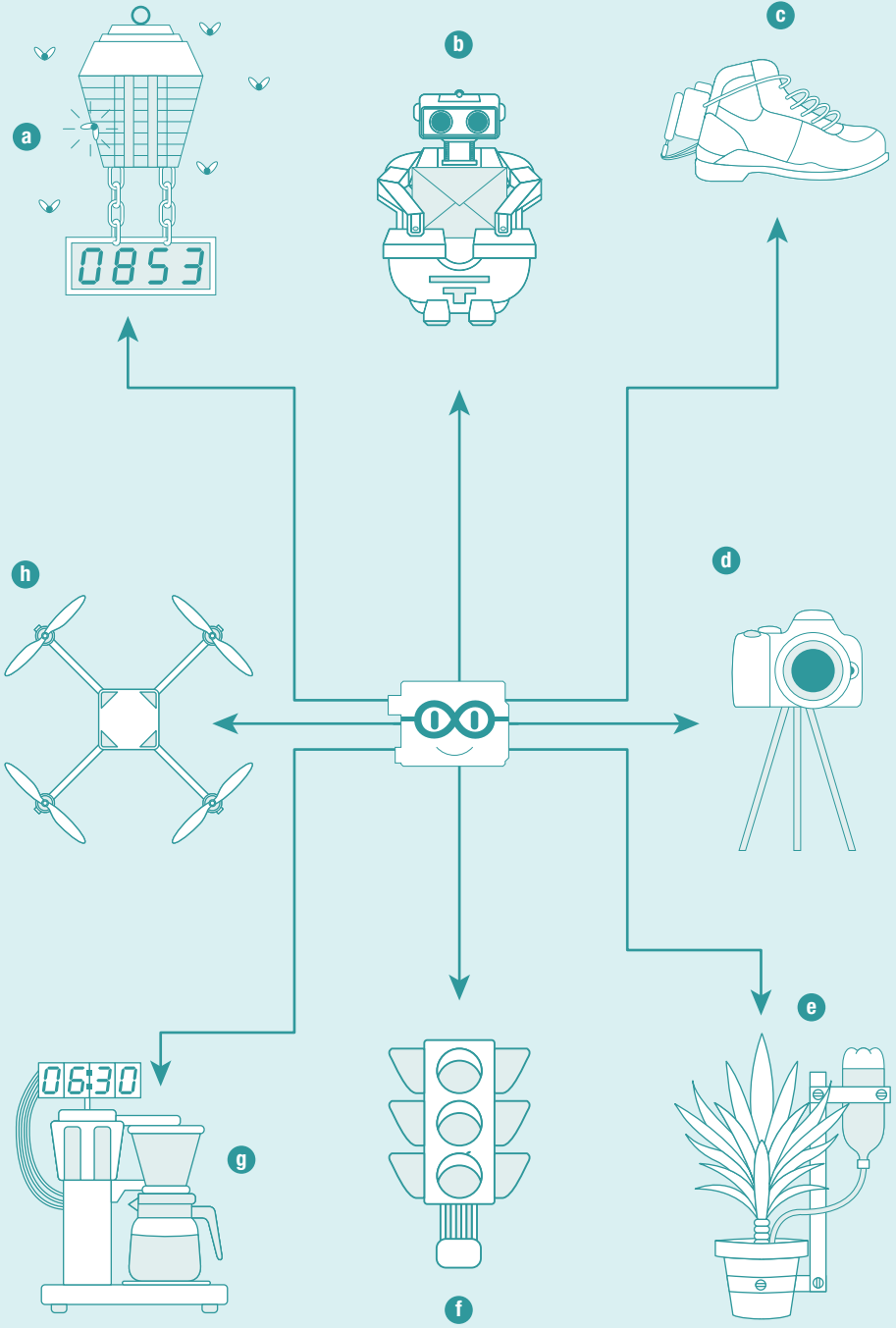
Arduino kartları bir Atmel AVR mikrodenetleyici programlama ve diğer devrelere bağlantı için gerekli yan elemanlardan oluşur. Her kartta en azından bir 5 voltluk regüle entegresi ve bir 16MHz kristal osilatör (bazılarında seramik rezonatör) bulunur. Mikrodenetleyiciye önceden bir bootloader programı yazılı olduğundan programlama için harici bir programlayıcıya ihtiyaç duyulmaz.



// Arduino UNO SMD R3

Arduino Uno en yaygın kullanılan arduino kartıdır. 14 dijital giriş / çıkış'a (6 PWM çıkışı), 6 analog girişe, bir 16 MHz kristal osilatöre, bir USB bağlantısına, bir güç girişine, bir ICSP başlığına ve bir reset düğmesine sahiptir. Heyecana kapılmaya gerek yok eğitim içerisinde tekrar bu konulara değinilecektir. Şimdilik yüzeysel olarak bilinmesi yeterlidir.





a Savaşlı Böcek Yakalama

d Kamera Zamanlama Uygulaması

g Kahve Makinesi

b Oyuncaktan Email Habercisi

e Otomatik Çiçek Sulama

h Quadcopter

c Otomatik Ayakkabı Bağcığı

f Programlanabilir Trafik Işıkları

Arduino IDE'yi İndirme (Entegre Geliştirme Ortamı)



İnternet Erişimi

Arduino ile yazılım geliştirmek ve çalıştırmak için öncelikle www.arduino.cc sitesinden bir Arduino Entegre Geliştirme Ortamı indirmeliyiz. Arduino IDE olarak bilinen bu yazılım arduino 'yu programlamak için adeta bir kelime işlemci görevi görecektir. İnternet tarayıcınızdan aşağıdaki adrese tıklayarak size uygun Arduino IDE 'yi indirebilirsiniz.

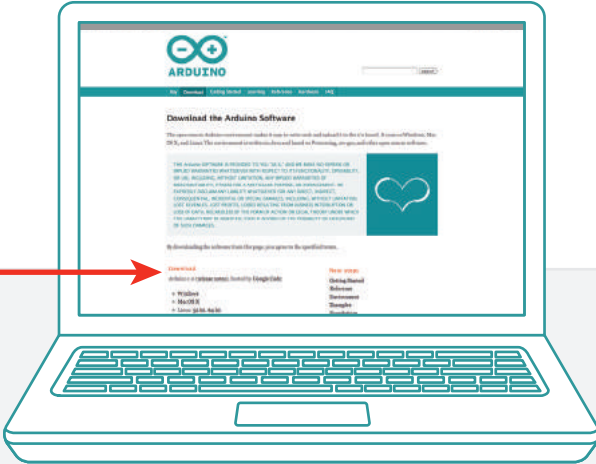
arduino.cc/en/Main/Software

1

Yükleme

Kullandığınız işletim sistemine göre size uygun olan Arduino programını seçmek için artı(+) işaretine basmanız yeterlidir.

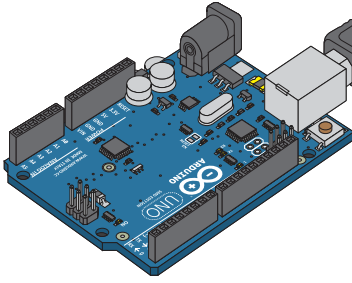
- + Windows
- + Mac OS X
- + Linux: 32 bit, 64 bit
- + source



Bilgisayarınız için uygun işletim sistemine göre yükleme paketi seçmeyi unutmayınız.

// Arduino Uno'yu Bilgisayara Bağlamak

Arduino ve Bilgisayarı USB çıkışlarından USB Kablosu ile bağlıyoruz.



2

// Sürücü Yükleme

3

Bilgisayarınızın işletim sistemine göre ihtiyacınız olan sürücüyü indirip yükliyoruz.



Windows Sürücü Yükleme

Aşağıdaki adreste Windows sürücü yüklemesi ile ilgili talimatları bulabilirsiniz.

<http://arduino.cc/en/Guide/Windows>



Macintosh OS X Sürücü Yükleme

Aşağıdaki adreste MacOSX sürücü yüklemesi ile ilgili talimatları bulabilirsiniz.

<http://arduino.cc/en/Guide/MacOSX>



Linux: 32 bit / 64 bit Sürücü Yükleme

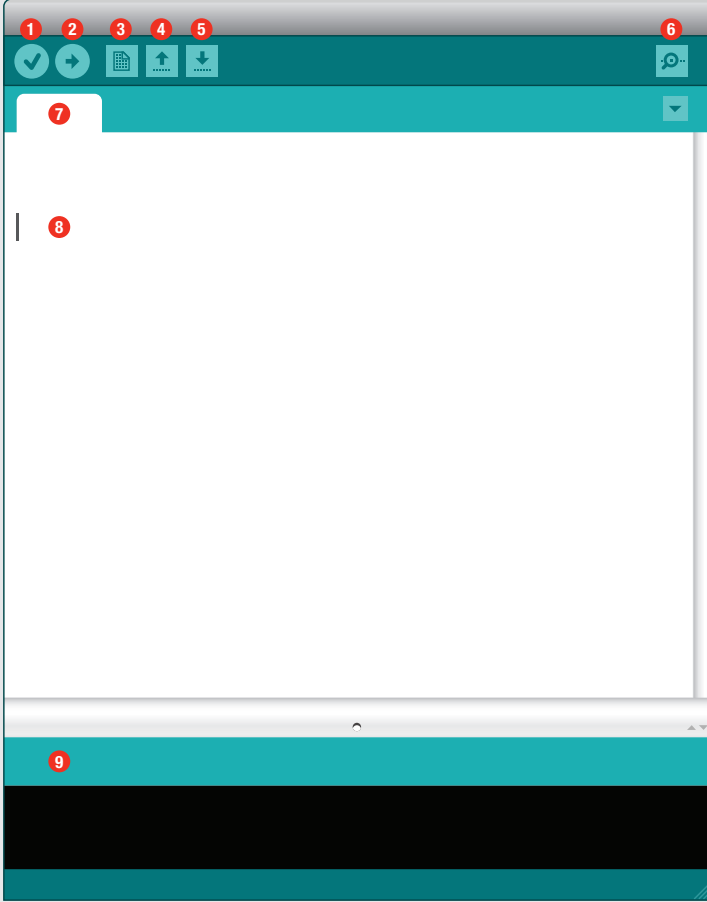
Aşağıdaki adreste Linux sürücü yüklemesi ile ilgili talimatları bulabilirsiniz.

<http://www.arduino.cc/playground/Learning/Linux>



//Arduino IDE 'yi Açıyoruz:

Öncelikle bilgisayarımıza indirdiğimiz arduino geliştirme programını açalım. Programlamaya geçmeden önce biraz geliştirme ortamını kurcalayalım.



GUI (Grafik Kullanıcı Arayüzü)

- 1 Verify(Derleme):** Yazmış olduğunuz kodu derler. Söz dizimi hatalarınızı bulur.
- 2 Upload(Yükleme):** Kodu arduino kartına gönderir.
- 3 New(Yeni):** Bu buton yeni kod penceresi açar.
- 4 Open(Açmak):** Bu buton var olan bir arduino dosyasını açmanızı sağlar.
- 5 Save(Kaydetmek):** Yazdığımız sketch'i kaydeder.
- 6 Serial Monitor(Seri monitör):** Seri bilgi görüntüleyen bir pencere açacaktır. Özellikle hata ayıklama için çok yararlıdır.
- 7 Sketch Name:** Şu an da üzerinde çalıştığımız sketch in adını gösterir.
- 8 Code Area(Kod Alanı):** Sketch için kod oluşturacağınız alandır.
- 9 Message Area(Mesaj Alanı):** Kodunuzdaki hatayı gösteren alan .

// Bu klavuz için en önemli üç komut aşağıda görülmektedir:



Open(Açmak)



Verify(Derleme)



Upload(Yükleme)

4

//Kartımızı Seçelim: Arduino Uno

File Edit Sketch **Tools** Help

Auto Format
Archive Sketch
Fix Encoding & Reload
Serial Monitor

Board

Serial Port

Programmer

Burn Bootloader

Arduino Uno

Arduino Duemilanove w/ ATmega328J

Arduino Diecimila or Duemilanove w/ ATmega168

Arduino Nano w/ ATmega328

Arduino Nano w/ ATmega168

Arduino Mega 2560 or Mega ADK

Arduino Mega (ATmega1280)

Arduino Mini

Arduino Mini w/ATmega168

Arduino Ethernet

Arduino Fio

Arduino BT w/ ATmega328

Arduino BT w/ATmega168

LilyPad Arduino w/ ATmega328

LilyPad Arduino w/ ATmega168

Arduino Pro or Pro Mini (5V, 16 MHz) w/ATmega328

Arduino Pro or Pro Mini (5V, 16 MHz) w/ATmega168

Arduino Pro or Pro Mini (3.3V, 8 MHz) w/ATmega328

Arduino Pro or Pro Mini (3.3V, 8 MHz) w/ATmega168

Arduino NG or older w/ ATmega168

Arduino NG or older w/ ATmega8



Aygıtımız için uygun olan seri portu seçmek için Tools >Serial Port menüsüne tıklarız. Genelde com3 veya üzeridir (COM1 ve COM2 genellikle donanım seri portu olarak kullanılmaktadır). Kullanacağımız arduino portunu bulmak için Arduino ile bilgisayar arasındaki bağlantıyı kesin ve menüye tekrar gelin ,bağlantıyı kestikten sonra menüde kaybolan port sizin Arduino portunuzdur. O portu seçip arduino programlamaya başlayabiliriz.

Tools Help

Auto Format
Archive Sketch
Fix Encoding & Reload
Serial Monitor

Board

Serial Port

Programmer

Burn Bootloader

com 1

com 12



Aygıtımız için uygun olan seri portu seçmek için Tools >Serial Port menüsüne tıklarız.Mac bilgisayarlarda Uno veya Mega 2560 için (/dev/tty.usbmodem) ve ya eski kartlar için (/dev/tty.usbserial) seçiyoruz.

Tools Help

Auto Format
Archive Sketch
Fix Encoding & Reload
Serial Monitor

Board

Serial Port

Programmer

Burn Bootloader

/dev/tty.usbmodem262471

/dev/cu.usbmodem262471

/dev/tty.Bluetooth-Modem

/dev/cu.Bluetooth-Modem

/dev/tty.FireFly-7256-SPP

/dev/cu.FireFly-7256-SPP

/dev/tty.tiPhone-WirelessiAP-1

/dev/cu.tiPhone-WirelessiAP-1

/dev/tty.Bluetooth-PDA-Sync

/dev/cu.Bluetooth-PDA-Sync



<http://www.arduino.cc/playground/Learning/Linux>

5



İndirmek için aşağıdaki bağlantıyı takip edin;



Start → Programs → arduino → examples

Adreste verdiğimiz sıkıştırılmış dosyayı indirip, içerisindeki "Arduino Kod" dosyasını dışarı çıkartıyoruz.

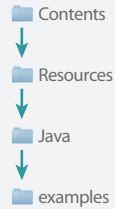
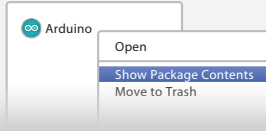
"Arduino Kod" dosyasını kopyalayıp arduino kurulumu içerisindeki "examples" klasörünün içerisine yapııştırıyoruz.



Adreste verdiğimiz sıkıştırılmış dosyayı indirip, içerisindeki "Arduino Kod" dosyasını dışarı çıkartıyoruz.



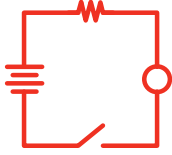
Arduino 'yu uygulama dosyasından buluyoruz. Arduino 'ya sağ tikiyoruz(ctrl + click) . "Show Package Contents" 'i seçiyoruz.



"Arduino Kod" dosyasını kopyalayıp examples klasörünün içerisine yapııştırıyoruz.



<http://www.arduino.cc/playground/Learning/Linux>



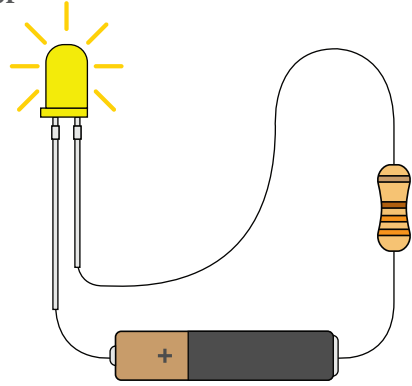
Elektrik Devresi Nedir?

Elektrik Devresi basitçe başlangıç ve bitiş noktası olan ve içerisinde bir çok devre elemanı buluduran dögüdür. Devreler her ölçüde ve şekilde yüzlerce ; direnç,diyot, bobin,sensör, motor ve diğer devre elemanlarını barındırabilir.

Devreler genelde analog devreler,dijital devreler ve karışık sinyal devreleri olmak üzere üç kategoriye ayrılır.Bu kitapçıkta her üç devre çeşidi de gösterilecektir.

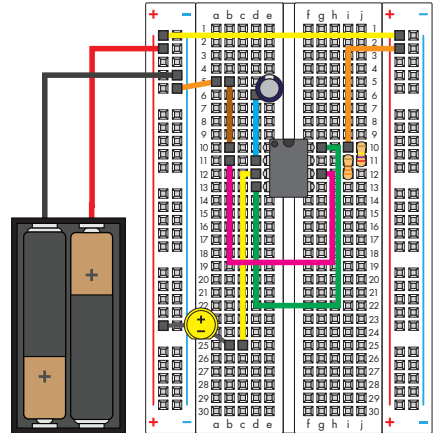
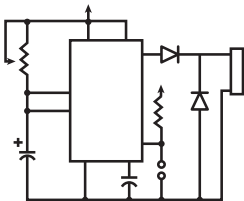
Dünya Bu Devreler Üzerinde Dönüyor

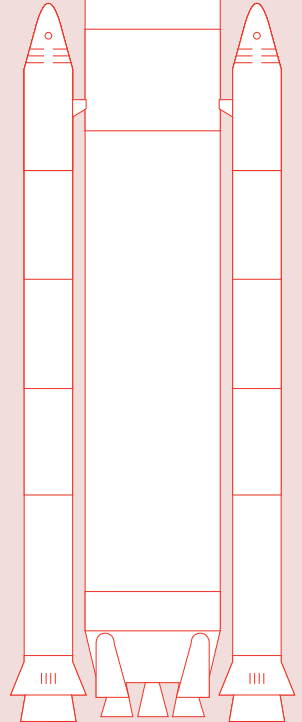
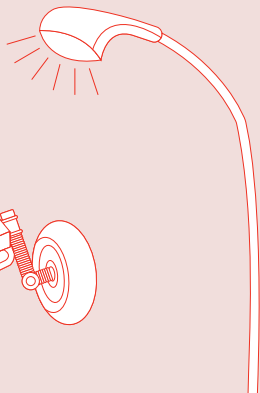
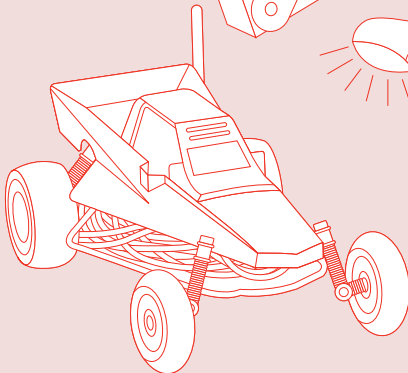
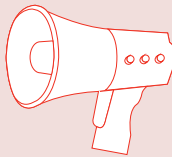
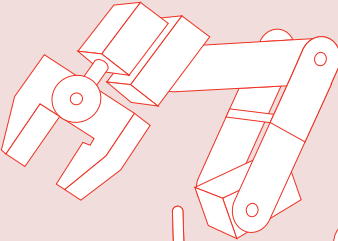
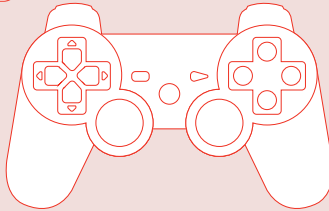
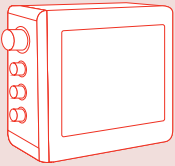
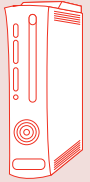
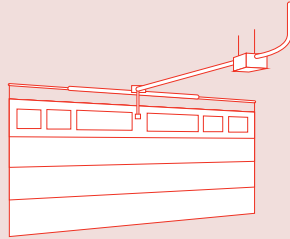
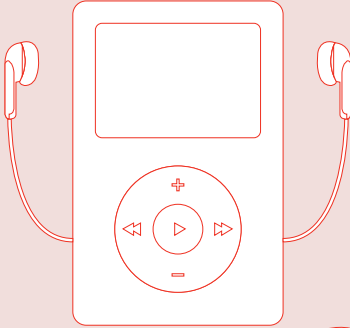
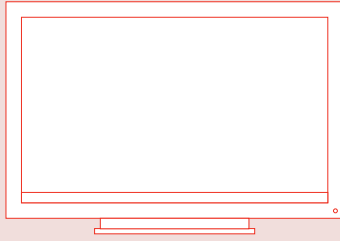
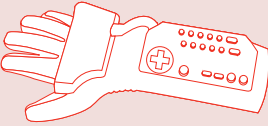
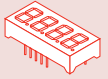
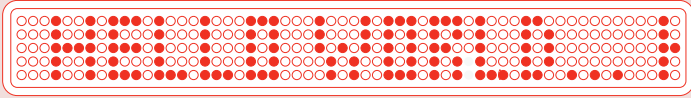
Nereye bakarsanız bakın devreler ile karşılaşacaksınız. Cebinizdeki telefon, aracınızın emisyon sistemini kontrol eden bilgisayar,oyun konsollarımız,elektrikli mutfak aletleriniz , bunların hepsi elektrik devreleriyle dolu. Bu kitapçıkta bazı basit devreleri deneyecek ve gömülü elektronik sistemlerinin temeline adım atacaksınız.



// Basit ve Karmaşık Devreler

Bu kitapçıkta ilk olarak basit devreleri keşfedeceksiniz. Ama bu sizin basit araçlarla büyüleyici şeyler yapamayacağınız anlamına gelmiyor elbette. Bu eğitimi tamamladıktan sonra devreler hakkında öğrendiğiniz bilgiler hayal gücünüze de bağlı olarak muhteşem projeler yapmanızı sağlayacak.

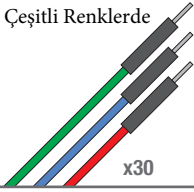




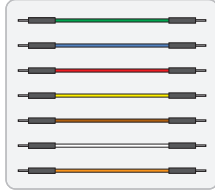
Devre Elemanları

Atlama Kablosu

Çeşitli Renklerde

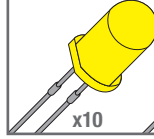


x30

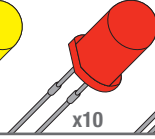


LED (5mm)

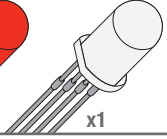
(Light Emitting Diode)



x10



x10



x1

330Ω Direnç



x25



*Gerçek Görünüm

10KΩ Direnç

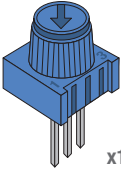


x25

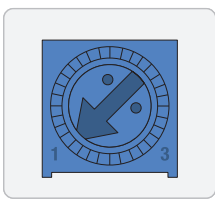


*Gerçek Görünüm

Potansiyometre

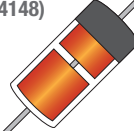


x1



Diyot

(1N4148)

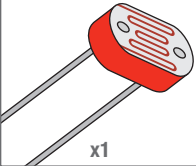


x2

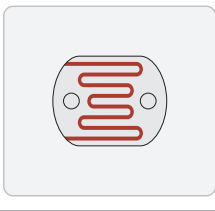


*Gerçek Görünüm

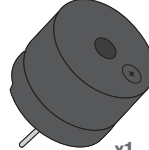
Foto Direnç (LDR)



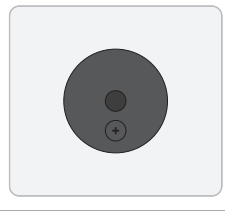
x1



Buzzer

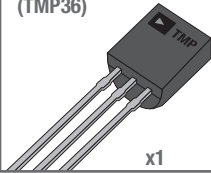


x1

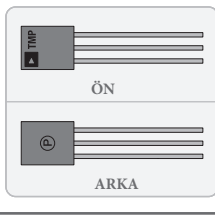


Sıcaklık Sensörü

(TMP36)

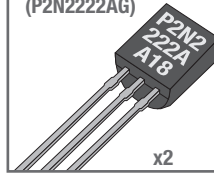


x1

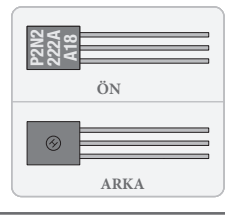


Transistor

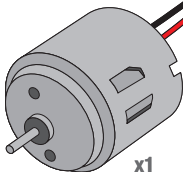
(P2N2222AG)



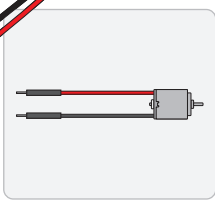
x2



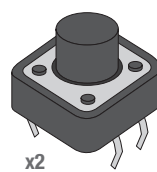
DC Motor



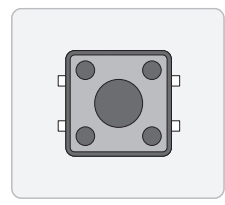
x1



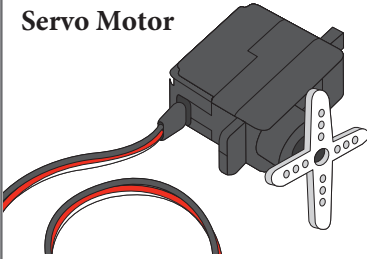
Push Buton



x2

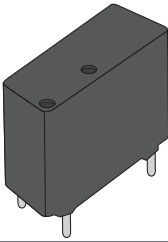


Servo Motor



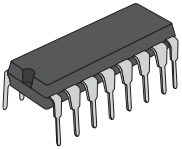
x1

Röle

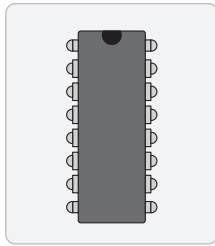


x1

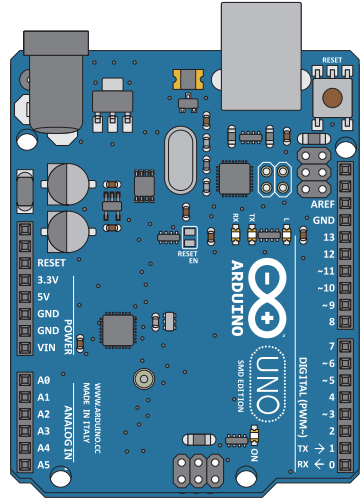
Entegre (IC)



x1

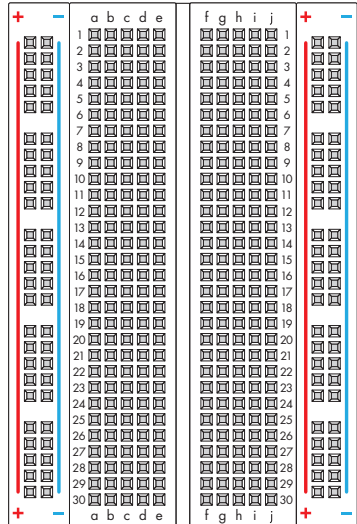


Arduino Kartı UNO

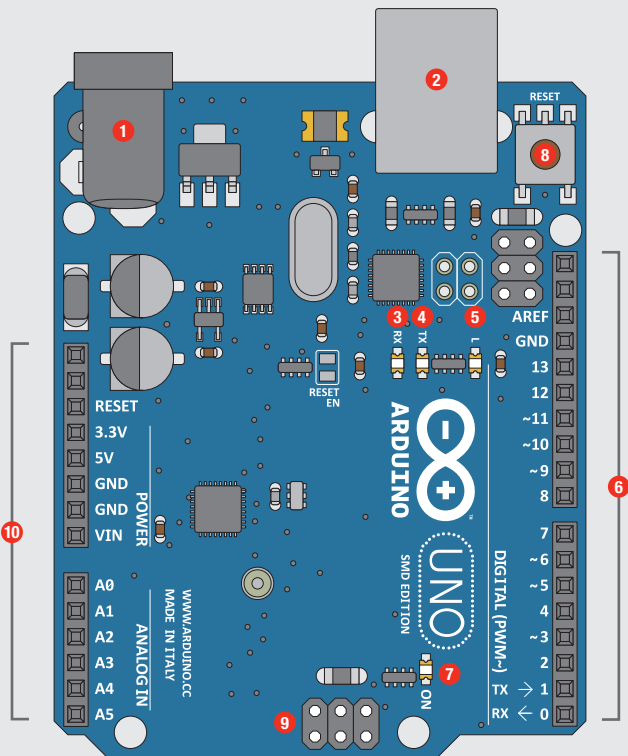


x1

Breadboard (Standart Lehimsiz)



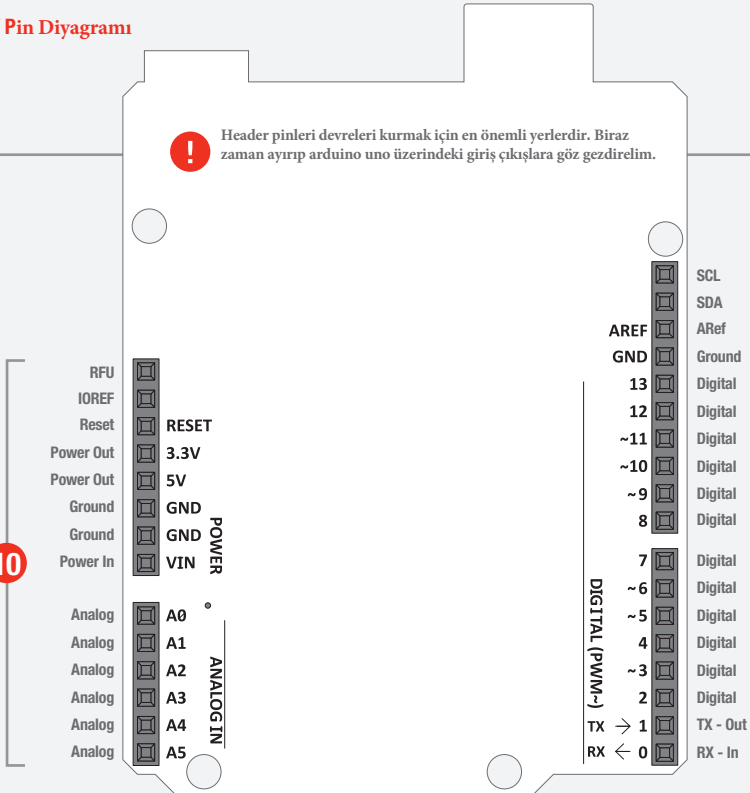
x1



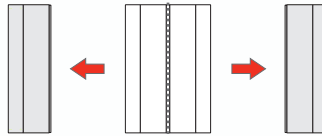
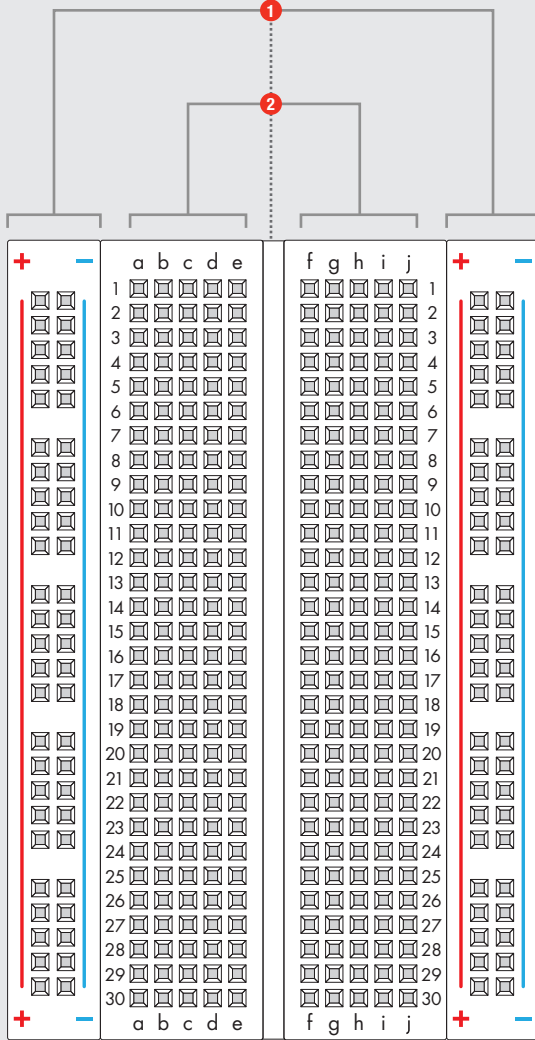
Arduino Uno

- 1 9-12V DC Güç Girişi(Barrel Jack) - 9V veya 12V güç bağlanabilen güç girişidir .
- 2 USB Bağlantı Konnektörü(USB Port) - Arduino 'ya program yüklemek ve bilgisayar ile haberleşmek için kullanılan bağlantı konnektörüdür.
- 3 LED(RX: Receiving) - Seri haberleşme için kullanılan RX pininin durumunu gösteren LED'dir. Veri alışverişi olduğunda bu led yanar.
- 4 LED(TX: Transmitting) - Seri haberleşme için kullanılan TX pininin durumunu gösteren LED'dir. Veri alışverişi olduğunda bu led yanar.
- 5 LED (Pin 13: Troubleshooting) - 13 Nolu dijital pine bağlı olan LED'dir. Programları test etmek için kullanılabilir.
- 6 Dijital Giriş-Çıkış Pinleri- Dijital giriş-çıkış için kullanılan pinler burada bulunuyor.Yanında işaretli bulunan pinler aynı zamanda analog çıkış(PWM) almak içinde kullanılıyor.
- 7 LED (Güç LED'i)-Kartımızın güç göstergesi LED 'idir. Devrelerinizi kurarken kısa devre yapıp yapmadığımız bu LED 'e bakarak anlayabilirsiniz.
- 8 Reset Butonu -Arduinoyu yeniden başlatır.Programlarımız setup() fonksiyonundan itibaren yeniden başlar.
- 9 Kart Üzerinde Programlama(ICSP) Pinleri - Atmega microdenetleyiciji harici bir programlayıcı ile programlama pinleri.
- 10 Analog Giriş Pinleri ve Güç Bağlantıları -Analog giriş ve gerilim bağlantıları bu bölümde bulunuyor.Ayrıca analog-dijital çeviricinin referans giriş pini ve seri iletişim pinleri de (RX ve TX) burada bulunuyor.

// Pin Diyagramı



~ = PWM/Analog çıkış uyumlu (i.e.~ 3)



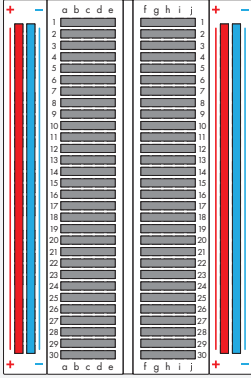
Bu çizgi breadboardumuzu ikiye bölüyor, Her bölüm kendi içinde bağımsızdır.

Breadboard

1 Dikey Bağlantı (+ Güç ve - Toprak // Aşağıdaki diyagrama Bakınız)

2 Yatay Bağlantı (a-e & f-j // Aşağıdaki diyagrama Bakınız)

Bağlantılar Nasıldır?



+ Güç:

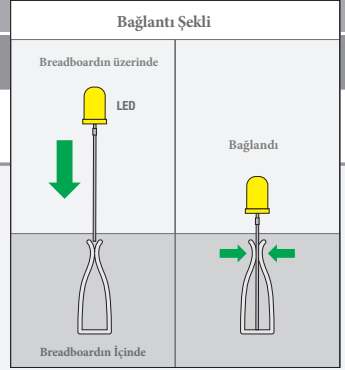
Her + işareti dikey sütundan güç geçişi sağlar.

- Toprak:

Her - işareti dikey sütunda her yerde toprak görevi görür.

Yatay Satırlar:

1 den 30 a kadar numaralı satırların herbiri kendi içerisinde (abcde ve ya fg hij) güç iletimi sağlar.



Breadboardın İç Görünümü>>>

Devreler #1 - İlk Devreniz

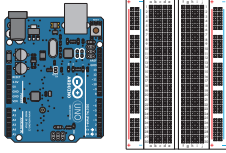
Nasıl Çalıştıracamız;

1 Parçaları Birleştir

2 Kodu Yaz

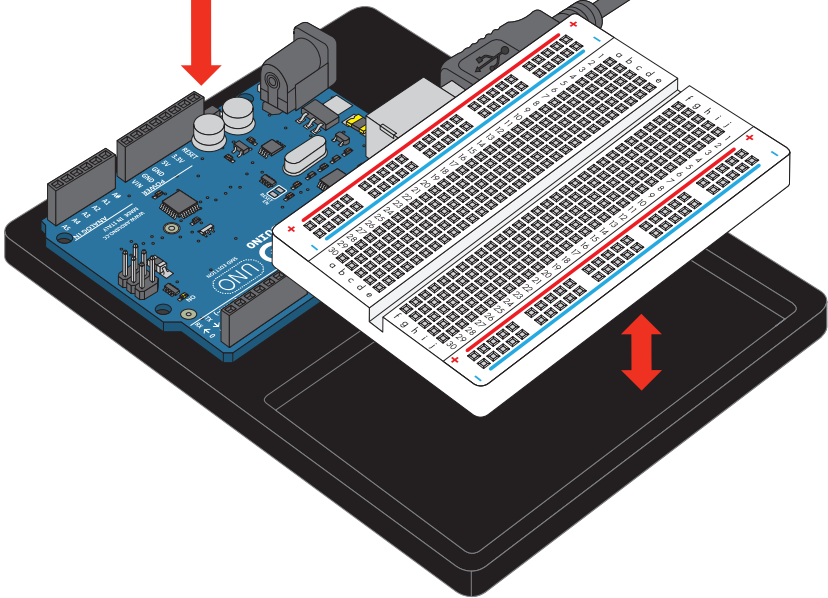
3 Arduino'ya Yükle

+ Breadboard hakkında genel bilgiler verildi, şimdi arduino ile breadboardu yan yana yerleştirelim ve talimatlara dikkat edelim.



+ Arduino ve Breadboardı yan yana yerleştir

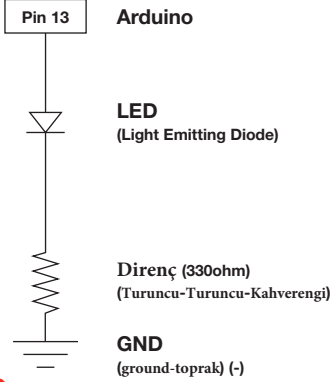
+ USB Kablosunu Bağla



5V Akım Arduino 5V da çalışır. Arduino'yu bilgisayara bağladığımızda aldığımız güç kurduğunuz devreleri çalıştıracaktır. Bilgisayarımızı Arduino'ya bağlayarak doğru gerilimi alabilirsiniz. 5V size zarar vermez bu yüzden devre içerisinde dokunmaktan korkmayın.

LED YAKMAK

LED'ler (light-emitting diodes) bir çok elektronik devrede kullanılan güçlü ışık kaynaklarıdır. Çalışmalarımıza ilk olarak her elektronik meraklının yaptığı gibi led yakarak başlayacağız. Bir çoğunuz için çok basit olabilir ama hem devre kurma hem de programlama olarak bize katacakları şeyler daha kompleks projeleri yapmak için bizlere birer basamak olacaktır.



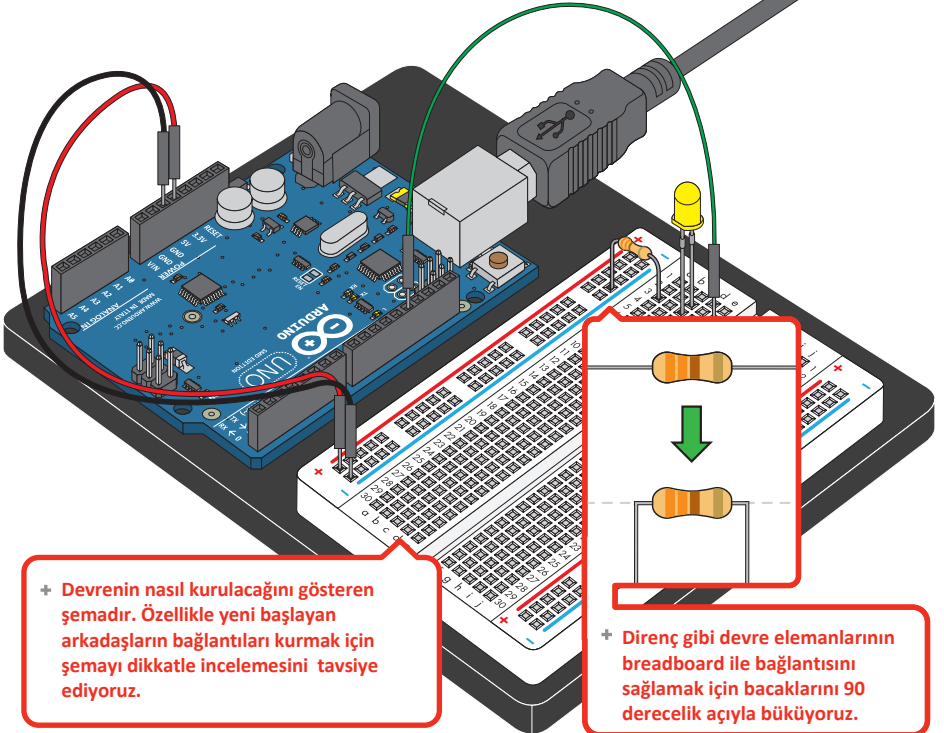
+ Bu LED yakma devremizin şemasıdır.

+ Bu bölümde devrelerimizi kurmadan önce kuracağımız devre ile ilgili kısa açıklamalar yer alır.

Elemanlar



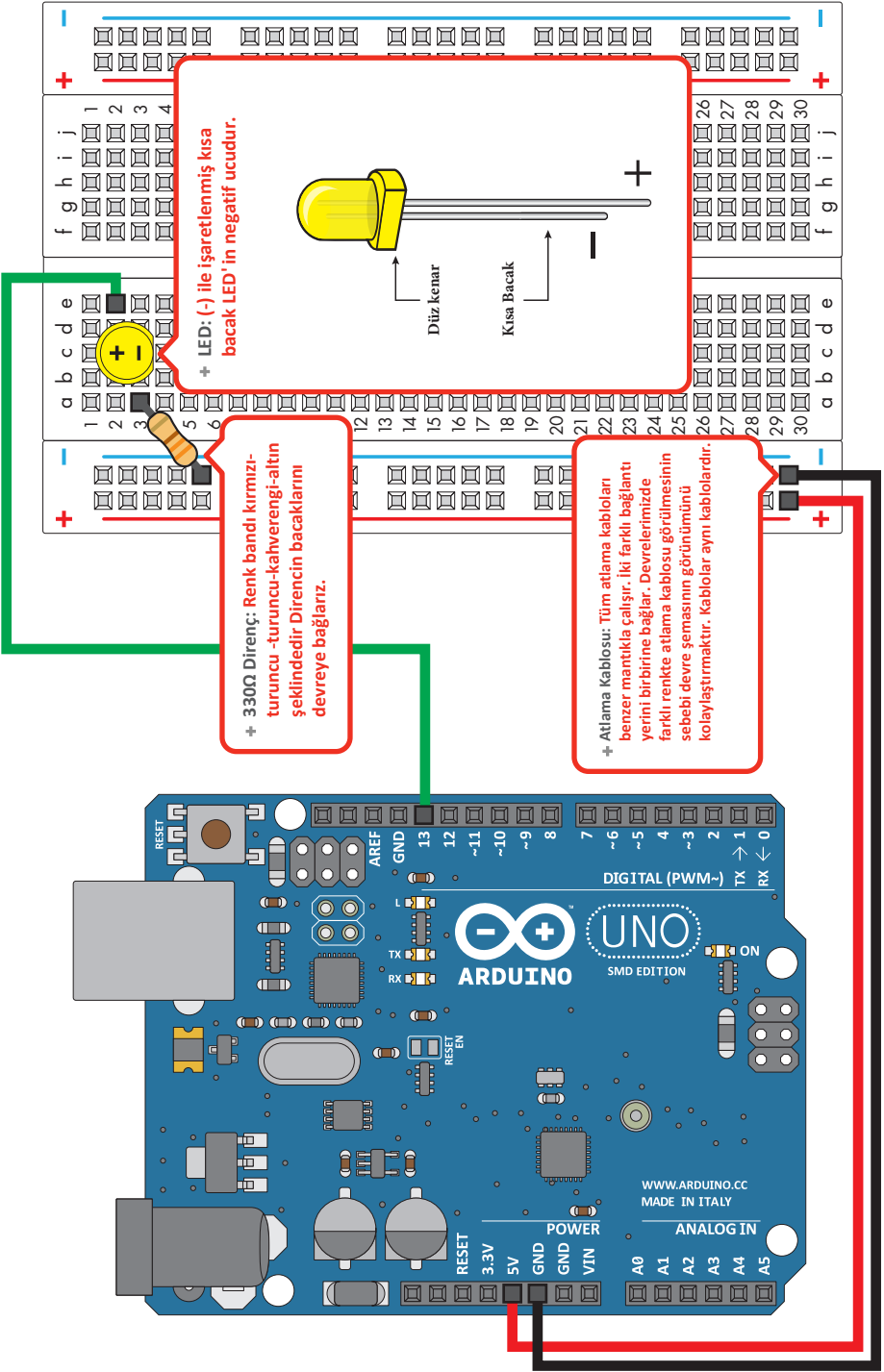
+ Bu bölümde, devreyi tamamlamak için gereken parçalar listelenir.













+ Devrenin nasıl kurulacağını gösteren şemadır. Özellikle yeni başlayan arkadaşların bağlantıları kurmak için şemayı dikkatle incelemesini tavsiye ediyoruz.

+ Direnç gibi devre elemanlarını breadboard ile bağlantısını sağlamak için bacaklarını 90 derecelik açıyla büküyoruz.

Devre 1: Led Yakmak

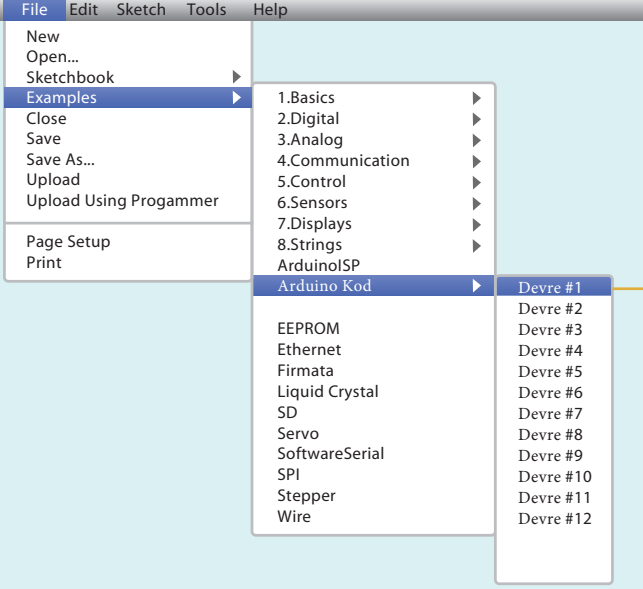


Elemanlar:	Gerçek Görünümleri:		
LED (5mm)			+ LED'i c2 (uzun bacak) c3 (kısa bacak) şeklinde breadboard'ın soketlerine takıyoruz.
330Ω Direnç			+ Dirençler yalnızca Breadboard üzerindeki soketlere yerleştirilir. "-" işaretli satıra ve LED'in bacağına bağlıyoruz.
Atlama Kablosu			+ Arduino üzerindeki "GND" pinini breadboard üzerindeki "-" işaretli satıra bağlıyoruz.
Atlama Kablosu			+ Arduino üzerindeki "5V" pinini breadboard üzerindeki "+" işaretli satıra bağlıyoruz.
Atlama Kablosu			+ Arduino üzerindeki "Pin 13" pinini breadboard üzerindeki "e2" soketine bağlıyoruz.
			+ Breadboard: Beyaz şerit breadboard soket bağlantılarını temsil eder
			+ Arduino: Mavi şerit Arduino header pinlerini temsil eder.



İlk Kodumuzu Açalım:

Bilgisayarımızdaki Arduino IDE yazılımını açıyoruz. Daha sonra aşağıdaki şemayı takip ederek "examples" klasörü içerisindeki "Arduino Kod" bölümünden "Devre 1" kodunu seçiyoruz.



//Devre #1

```
Circuit #1
/*
Led Yakmak (Blink=Göz Kırpma)
LED' i bir saniye yakıp bir saniye söndüren program
*/

int LEDcikis = 13;
void setup() {

// Dijital Pini çıkış olarak ayarlayacağız.

pinMode(LEDcikis, OUTPUT);
}
void loop() {
digitalWrite(LEDcikis, HIGH); //LED'i yak
delay(1000); // 1 saniye bekle
digitalWrite(LEDcikis, LOW); // LED'i söndür
delay(1000); //1 saniye bekle
}
```




Derleme

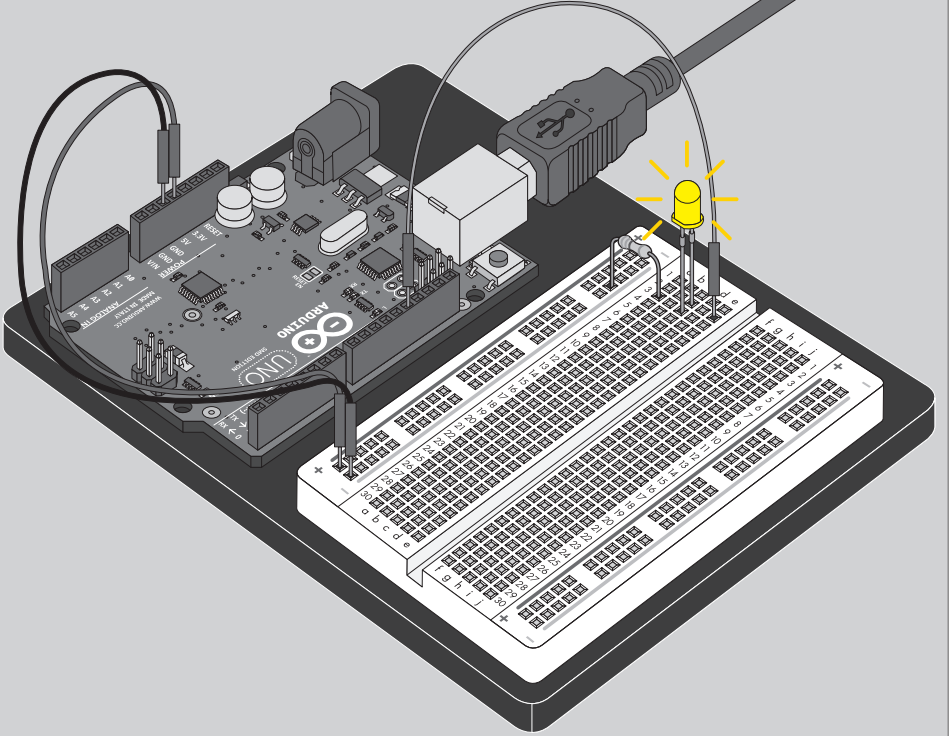
Kodu derliyoruz. Yazılım kodu arduino üzerindeki entegrenin anlayabileceği dile çeviriyor.



Yükleme

USB Kablo aracılığıyla kodu Arduino üzerindeki entegreye gönderiyoruz. Daha sonra devre otomatik olarak çalışmaya başlayacaktır.

// Verilen kod ile derledikten ve yükledikten sonra breadboard üzerindeki LED yanıp sönmeye başlayacaktır



1

+ Tüm devrelerde kodların ne olduğunu açıklayan bölümdür.



Arduino IDE 'yi Aç // File > Examples > Arduino Kod > Devre # 1

Kod notları:

+ Kodun nasıl çalıştığını anlamak için aşağıdaki açıklamaları inceleyin.

+ Verilen Kodu derlemeyi(verify) ve yüklemeyi(upload) unutmayın



pinMode(13, OUTPUT);



Arduinoki pinleri kullanmadan önce pini INPUT (giriş) yada OUTPUT (çıkış) olarak tanımlamanız gerekmektedir . Bunu yapmak için *pinMode()* yerleşik fonksiyonunu kullanacağız.

digitalWrite(13, HIGH);

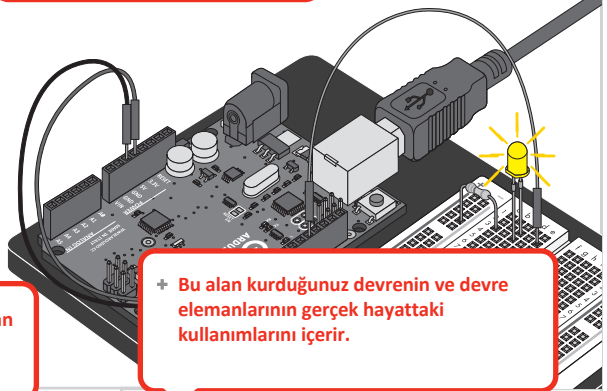


Bir pini OUTPUT (çıkış) olarak tanımladığımızda, o pinin HIGH (output 5 Volts-5 Volts çıkış), veya LOW (output 0 Volts-0 Volt çıkış) olarak davranmasını sağlayabilirsiniz .

Ne göreceğiz?

+ Devrenizi tamamladıysanız , aşağıdaki gibi LED yanıp sönecektir

Yanıp sönen bir LED göreceksiniz. Eğer çalışmıyorsa kodu kontrol edin ve tekrar derleyin sonra yükleyin ve devreyi tekrar kontrol edin veya aşağıdaki sorun giderme ipuçlarını bakın.



+ Bu bölüm devre montajı sırasında yapılan en yaygın hataları içerir.

+ Bu alan kurduğunuz devrenin ve devre elemanlarının gerçek hayattaki kullanımlarını içerir.

Sorun Giderme:

LED Yanmıyor

LED sadece tek yönlü çalışır. Bu tür sorunlar genelde LED in yanlış bacağına bağlanmasından kaynaklanıyor. Endişeye gerek yok Led in bacaklarını çevirin ve yeniden bağlayın.

Program Yüklenmiyor

Yükleme ile ilgili sorunlar genelde seri port'dadır. tools>serial port> sekmesinden portunuzu değiştirip tekrar yüklemeye çalışın.

Hâlâ Çalışmıyor mu?

Tamamlayamadığınız her devreniz hakkında yardım almak için emrearslan@elektrikport.com adresine mail yollayabilirsiniz.

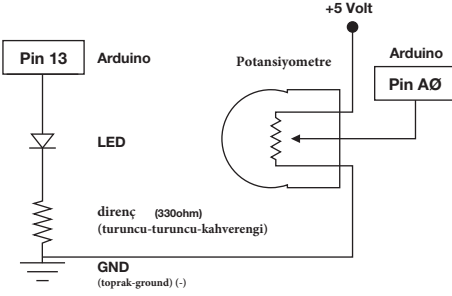
Gerçek Hayatta Uygulamaları:

Hemen hemen tüm modern televizyon ve monitörlerde ürünün çalıştığını ya da standby durumunda olduğunu gösteren LED ler bulunur.



Potansiyometre

Bu devrede potansiyometrenin ne işe yaradığını öğreneceğiz. Potansiyometre değişken bir direnç olarak bilinir. İki çıkış pini 5 Volta bağlandığında ortadaki pin potansiyometredeki kontrol düğmesinin konumuna göre 0V ile 5V arasında çıkış verir. Bu devrede, LED parlaklığını kontrol etmek için potansiyometreyi nasıl kullanacağınızı öğreneceksiniz.



Elemenlar:

Potansiyometre



x1

LED



x1

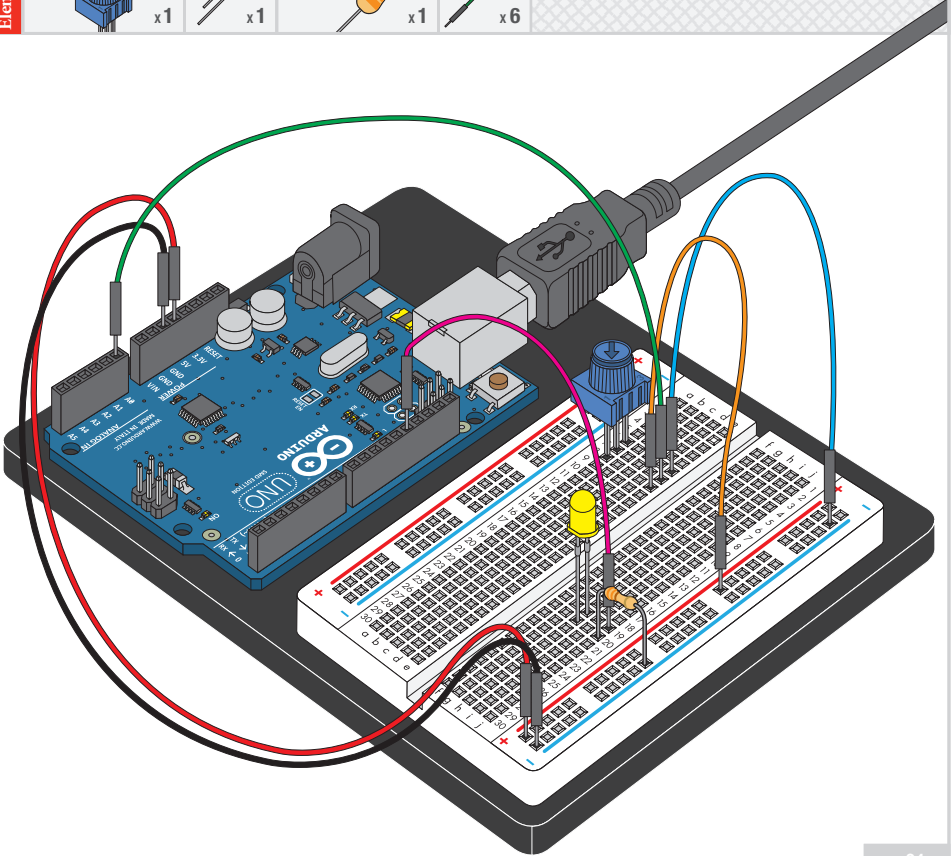
330Ω
Direnç

x1

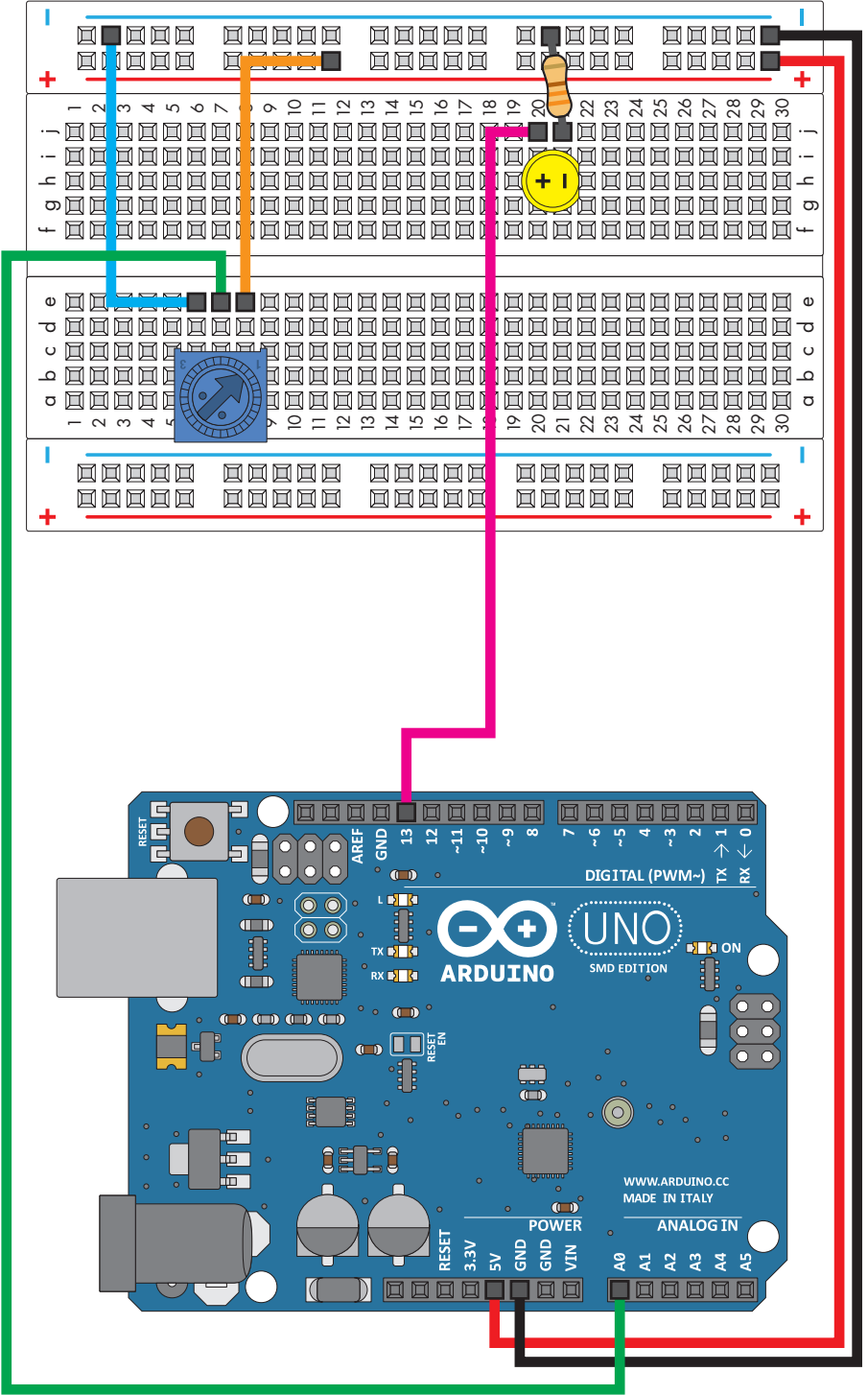
Kablo













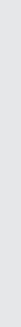

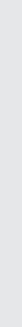

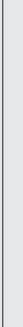



x6



Devre 2: Potansiyometre

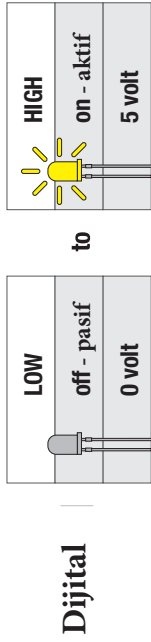


Elemanlar:	Gerçek Görünüm:		
Potansiyometre			
LED (5mm)			
330Ω Direnç			
Atlama Kablosu			
Atlama Kablosu			A0
Atlama Kablosu			
Atlama Kablosu			Pin 13
Atlama Kablosu			5V
Atlama Kablosu			GND

Dijital V Analog:

Eğer Arduinoya yakından bakarsanız bazı pinlerin "Dijital" bazı pinlerin ise "Analog" olduğunu göreceksiniz. Peki ama ne farkı var bu pinlerin?

Arduino üzerindeki dijital pinler ile dış dünyadan sinyal almak ve dış dünyaya sinyal vermek çok kolaydır. Hatta LED leri çok hızlı yakıp söndürerek dim sergilemek gibi numaralar ve seri iletişim de "HIGH" ve "LOW" gibi kalıplar ile başka cihazlara veri aktarımında kolaylıkla kullanılabilir.



Ama sadece on ya da off olarak tanımlamayacağımız birçok şey vardır. Sıcaklık seviyeleri, kontrol düğmeleri gibi birçok şeyde pasif ve aktif (HIGH ve LOW) arasında sürekli değişebilen değerler alabilirler. Bu durumlarda Arduino bir giriş voltajını 0 (0 volt) ile 1023 (5 volt) arasında değişebilen sayılara dönüştürebilen altı analog giriş sunar. Analog girişler tüm bu gerçek değerleri ölçmekte çok iyidirler ve bu tarz değerleri Arduino'ya aktarmanızı sağlarlar.





Arduino IDE 'yi Aç// File > Examples > Arduino Kod > Devre # 2

Kod Notları:



int sensorValue;



"Değişken" sizin isim verdiğiniz bir rakamdır. Değişkeni kullanmadan önce onu tanımlamalısınız veya bildirmelisiniz; burada sensorValue adında bir değişkeni "int" (integer-tamsayı) olarak bildiriyoruz. Bu değişken isimlerinin küçük-büyük harf duyarlı olduğunu unutmayın.

sensorValue = analogRead(sensorPin);



Analog pindeki değeri okumak için analogRead() fonksiyonunu kullanırız. analogRead() kullanmak istediğiniz analog pindeki ("sensorPin") değeri okur ve bir rakama ("sensorValue") dönüştürür, bu rakam 0 (0 Volt) ile 1023 (5 Volt) arasındadır.

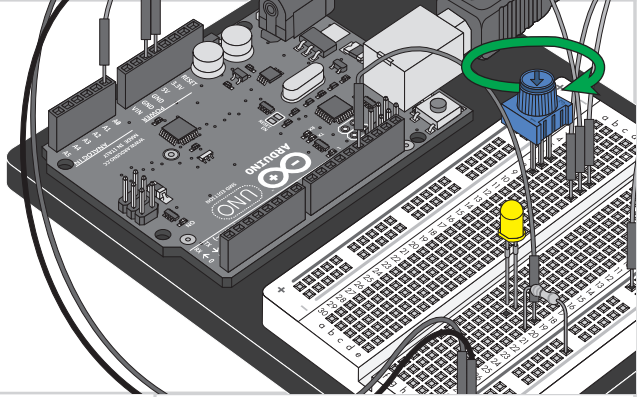
delay(sensorValue);



Arduino her saniye binlerce satır kodu çalıştıracak kapasiteye sahip, çok hızlı bir yapıdır. Onu yavaşlatıp neler yaptığını gözlemlemek için bazen kodlar arasına ("delay") gecikmeler ekleriz. Delay() fonksiyonu milisaniye ile sayar; yani 1 s gecikme için parantez içine 1000ms yazılmalıdır.

Ne göreceğiz?

Potansiyometrenin ayarını değiştirdikçe LED'in hızlı ve yavaş bi şekilde yanıp söndüğünü göreceksiniz. Eğer çalışmıyorsa kodu kontrol edin ve tekrar derleyin sonra yükleyin ve devreyi tekrar kontrol edin veya aşağıdaki sorun giderme ipuçlarını bakın.



Sorun Giderme:

Düzensiz Çalışma Durumu

Büyük olasılıkla potansiyometrenin pinlerinin breadboardta tam oturmasından kaynaklanabilir. Potansiyometreyi bastırarak bu sorunu çözebilirsiniz.

Çalışmama Durumu

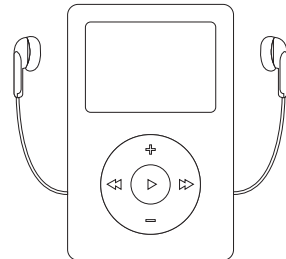
Potansiyometrenin 2. pinini analog pin yerine dijital pine bağlamadığınızdan emin olunuz.

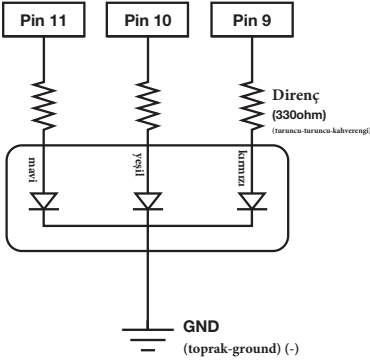
Hâlâ Çalışmıyor mu?

Tamamlayamadığınız her devrenin hakkında yardım almak için emrearslan@elektrikport.com adresine mail yollayabilirsiniz.

Gerçek Hayatta Uygulamaları:

MP3 çalarlardaki ses kontrolü potansiyometre kullanımına bir örnektir.

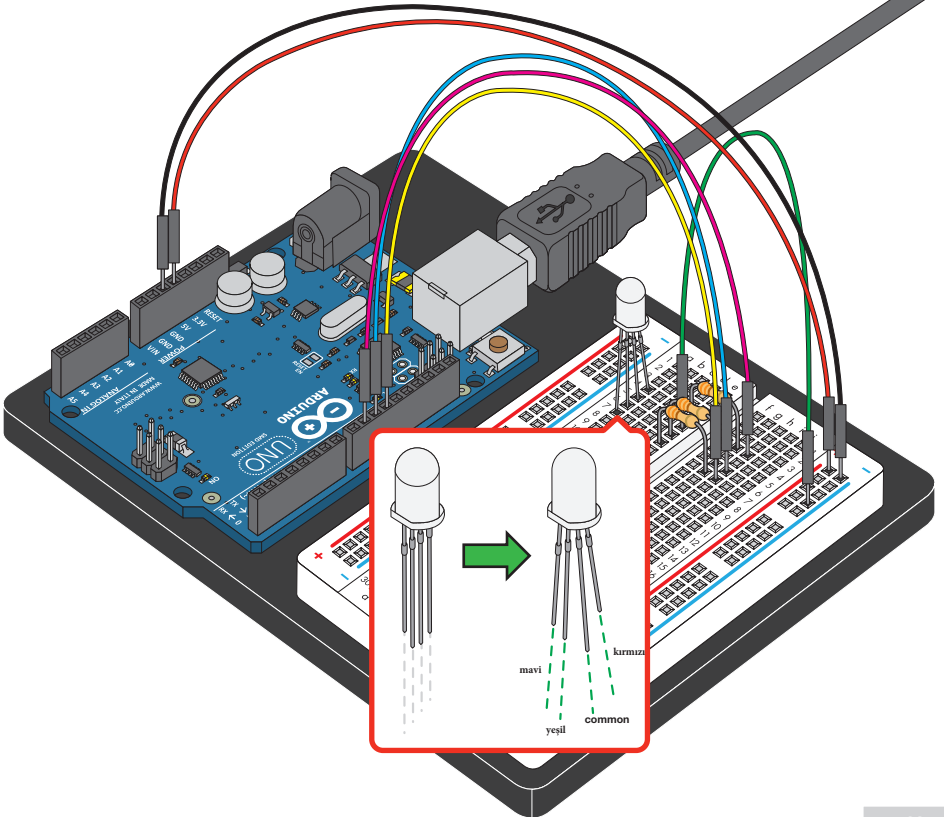




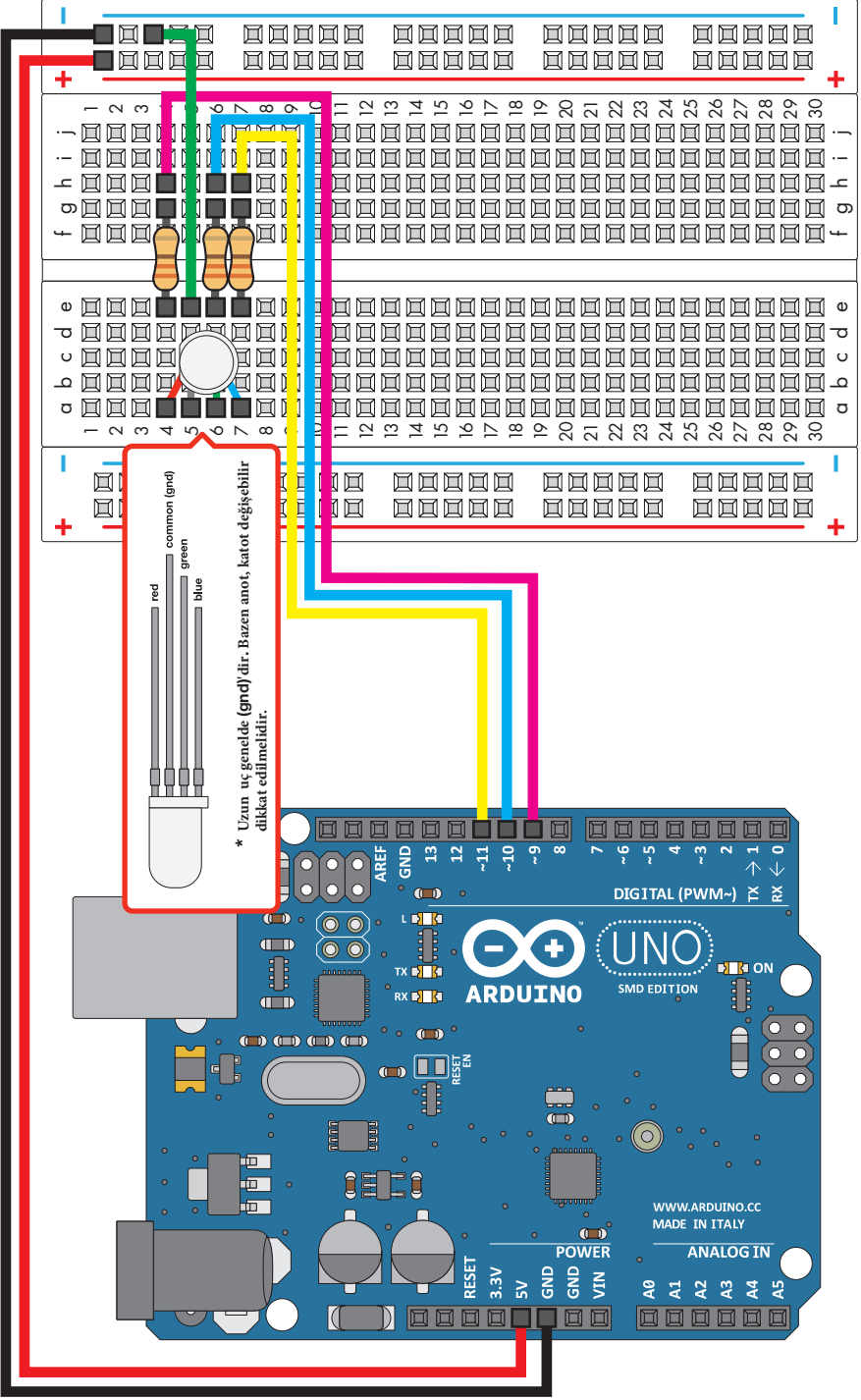
RGB LED

Yanıp sönen bir LED'den daha eğlenceli şey nedir sizce? Tabii ki renkli LED'ler! RGB, yani red-blue-green LED'ler 3 adet renk barındıran ve her tür renk için kombin edilebilen bir diyet türüdür. Bu devrede Bir RGB LED kullanarak nasıl renk kombinasyonları oluşturabileceğinizi öğreneceksiniz. Her bir diyetün parlaklığına bağlı olarak neredeyse bütün renkleri elde etmeniz mümkün.

Elementler:	LED	330Ω Direnç	Kablo
	 x1	 x3	 x6



Devre 3: RGB LED

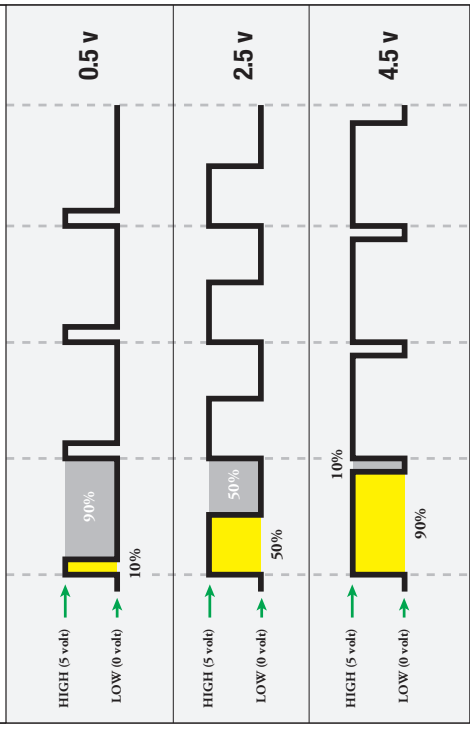


Elemanlar:	Gerçek Görünüm:		
RGB LED (5mm)			
330Ω Direnç			
330Ω Direnç			
330Ω Direnç			
Atlama Kablosu			
Atlama Kablosu			
Atlama Kablosu			
Atlama Kablosu			
Atlama Kablosu			
Atlama Kablosu			

analogWrite() Arkasındaki Şok Edici Gerçek:

Şimdiye kadar Arduino'nun analog voltajları (0 ve 5 voltaj arasında ki değerleri) analogRead() fonksiyonu kullanarak okuyabildiğini gördük. Peki Arduino'nun analog voltaj çıktısı vermek gibi bir imkanı var mı diye soracak olursak, cevabımız hem hayır hem evet olacaktır. Arduino gerçek bir analog voltaj çıkıtısına sahip değildir ama Arduino'nun çok hızlı olmasından dolayı PWM ("Pulse-Width Modulation") kullanılarak bu çıktıyı uyandırabilir*

Arduino o kadar hızlı çalışır ki bir pini saniyede 1000 kez açıp kapatabilir. PWM HIGH ve LOW olarak harcanan zamanı kullanıyor. Eğer HIGH konumunda daha fazla vakit harcamıyorsa bu pine bağlı olan LED parlak yanacaktır. Fakat LOW konumunda daha fazla vakit harcamıyorsa LED daha sönük kalacaktır çünkü pin gözün görebileceğinden daha hızlı açıp kapama yapacaktır. İşte bu Arduino'nun "gerçek" analog çıktısı için yaptığı bir illüzyondur.





Arduino IDE 'yi Aç// File > Examples > Arduino Kod > Devre # 3

Kod Notları:



for (x = 0; x < 768; x++)
{|



Bir for() döngüsü bir aralıkta sayı yerleştirmek için kullanılır ve tekrarlayarak brackets{} içerisinde kodu çalıştırır. Değişken bir "x" 0 olarak başlıyor ve 767 sayısına kadar her adımda bir artarak ilerliyor ("x++")..

if (x <= 255)
{|
else
{|



"If / else" ifadeleri programınızda seçim yapmak için kullanılır. Parantez () içerisindeki ifade değerlendirildiğinde, eğer ifadesiniz doğru ise ilk brackets{} içerisindeki ifade yürütülür. Şayet doğru değil ise ikinci brackest {} içerisindeki ifade yürütülür.

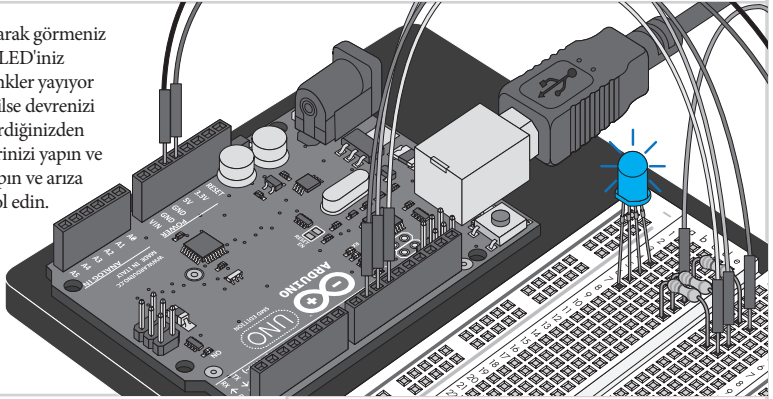
delay(sensorValue);



Arduino çok hızlı çalışabilmektedir. Öyle ki her saniye binlerce kod satırını çalıştırabilecek kapasiteye sahiptir. Biz de cihazın ne yaptığını görebilmek için yavaşlatmak adına gecikmeler ekliyoruz. Delay() milisaniye olarak hesaplanıyor; 1 saniye 1000 milisaniye.

Ne göreceğiz?

LED'inizi çalışıyor olarak görmemiz lazım, fakat bu sefer LED'iniz sırayla farklı farklı renkleri yayıyor olmalı. Eğer öyle değilse devrenizi doğru şekilde birleştirdiğinizden emin olun, kontrollerinizi yapın ve kartınıza yüklemeyi yapın ve arıza tespiti kısmını kontrol edin.



Sorun Giderme:

LED'in Işık Vermemesi veya Yanlış Renk Vermesi
LED'in dört pini birbirine çok yakın bulunmakta, kolaylıkla yerlerini karıştırabilirsiniz. Pinlerin nerede olması gerektiğini iki kez kontrol edin. Ayrıca uzun uç bazen GND olmayabilir, uzun ucu Arduino kartınızın 5v çıkışına bağlayıp tekrar deneyiniz.

Kırmızı Renk Görmek

RGB LED içerisindeki kırmızı diyot diğerlerinden daha parlak olabilir. Renklerinizi daha dengeli hale getirmek için daha yüksek bir direnç kullanın veya kodunuzda ayarlamalar yapın.

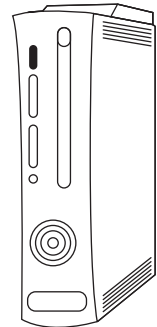
```
analogWrite(RED_PIN, redIntensity);
```

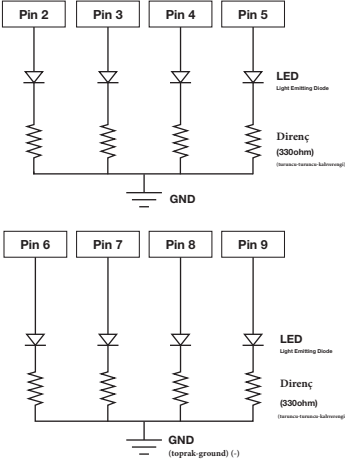
```
to
```

```
analogWrite(RED_PIN, redIntensity/3);
```

Gerçek Hayatta Uygulamaları:

Videogame konsolları gibi bir çok elektrikte RGB LED kullanılmaktadır. Bu LED'ler aynı bölgede farklı renkleri göstermek için kullanılıyor. Sıklıkla farklı renkler farklı çalışma şartlarını ifade eder



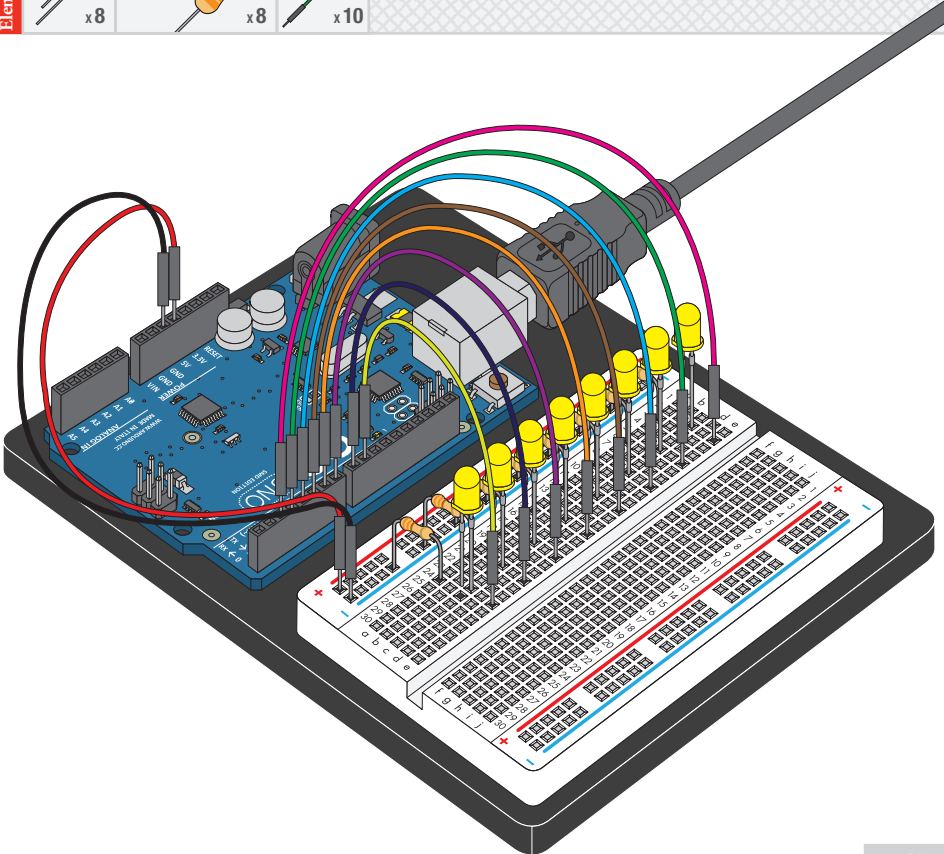


Çoklu LED

Elimizde yakıp söndürmek için bir LED var. Gelin şimdi **SEKİZ LED'i** anda bağlayarak çıtayı biraz yükseltelim. Böylece çeşitli renkler oluşturma konusunda Arduinomuzu da ufak bir teste tabi tutmuş olacağız. Bu devre kendi programınızı yazma pratikleri için güzel bir başlangıç adımı olacak. LED'leri kontrol aşamasında yazdığımız programı düzenli tutmanızı sağlayacak bir kaç ipucu öğreneceksiniz.

for() loops - Bu döngü birden fazla kez aynı kodu kullanmanızı sağlar.

arrays[] - Birden fazla değişkeni grup haline getirerek yönetilmelerini kolaylaştırır.



Devre 4: Çoklu LED

