



ARDUINO'YA GİRİŞ

ANTALYA SINAV ANADOLU LİSESİ ROBOTİK TOPLULUĞU



SINAV KOLEJİ



E-posta: yucelkilog@antalyasinavkoleji.com

Arduino Nedir?

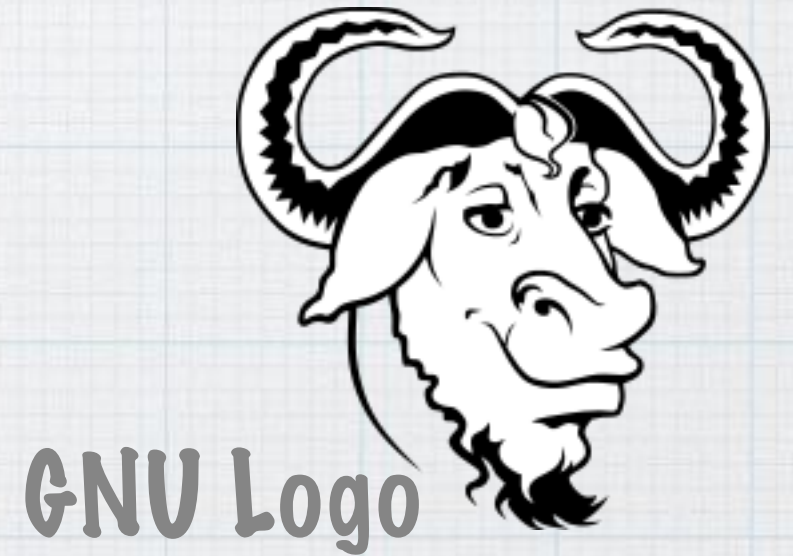
Arduino is an open-source electronics platform based on easy-to-use hardware and software.

Arduino boards are able to read inputs - light on a sensor, a finger on a button, or a Twitter message - and turn it into an output - activating a motor, turning on an LED, publishing something online. You can tell your board what to do by sending a set of instructions to the microcontroller on the board. To do so you use the Arduino programming language (based on Wiring), and the Arduino Software (IDE), based on Processing.

<https://www.arduino.cc/en/Guide/Introduction>

Özgür Yazılım Nedir?


- **Özgürlük 0:** Programı sınırsız kullanma özgürlüğü.
- **Özgürlük 1:** Programın nasıl çalıştığını inceleme ve amaçlara uygun değiştirme özgürlüğü.
- **Özgürlük 2:** Programın kopyalarını sınırsız dağıtma özgürlüğü.
- **Özgürlük 3:** Programın değiştirilmiş halini dağıtma özgürlüğü.



<https://www.gnu.org/licenses/gpl-3.0.en.html>

https://tr.wikipedia.org/wiki/GNU_Genel_Kamu_Lisansı

Arduino Çeşitleri

 Buy Download Products Learning Forum Support Blog LOGIN SIGN

Summary

Entry Level

Enhanced Features

IoT

Wearable

3D Printing

Retired

ENTRY LEVEL

ARDUINO UNOARDUINO 101ARDUINO PROARDUINO PRO MINIARDUINO MICRO

ARDUINO STARTER KITARDUINO BASIC KITMKR1000 BUNDLE

ENHANCED FEATURES

ARDUINO MEGAARDUINO ZEROARDUINO PROTO SHIELD

INTERNET OF THINGS

ARDUINO MKR1000ARDUINO WIFI SHIELD 101ARDUINO YUN SHIELD

WEARABLE

ARDUINO GEMMA

LILYPAD ARDUINO USB

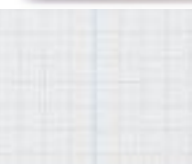




















LILYPAD ARDUINO MAIN BOARD

LILYPAD ARDUINO SIMPLE

LILYPAD ARDUINO SIMPLE SNAP

3D PRINTING

MATERIA 101



Internet of Things

Make connected devices easily with one of these IoT products and open your creativity with the opportunities of the world wide web.

BOARDS






MODULES

SHIELDS

KITS

ACCESSORIES

COMING



<https://www.arduino.cc/en/Main/Products>

Arduino Uno

Technical specs

Microcontroller	ATmega328P
Operating Voltage	5V
Input Voltage (recommended)	7-12V
Input Voltage (limit)	6-20V
Digital I/O Pins	14 (of which 6 provide PWM output)
PWM Digital I/O Pins	6
Analog Input Pins	6
DC Current per I/O Pin	20 mA
DC Current for 3.3V Pin	50 mA
Flash Memory	32 KB (ATmega328P) of which 0.5 KB used by bootloader
SRAM	2 KB (ATmega328P)
EEPROM	1 KB (ATmega328P)
Clock Speed	16 MHz
LED_BUILTIN	13
Length	68.6 mm
Width	53.4 mm
Weight	25 g



<https://www.arduino.cc/en/Main/ArduinoBoardUno>

E-Belge/E-Kaynak



The screenshot shows the Arduino.cc website's Language Reference page. The header is teal with the Arduino and Genuino logos, a search bar, and navigation links. The 'Learning' menu is open, showing options like 'Getting started', 'Tutorials', 'Reference', 'CTC Program', and 'Playground'. The main content area is titled 'Language Reference' and includes a paragraph about the structure of Arduino programs. Below this, there are three columns: 'Structure' with links to 'setup()' and 'loop()'; 'Control Structures' with links to 'if', 'if...else', 'for', 'switch case', and 'while'; 'Variables' with sub-sections for 'Constants' (listing HIGH, LOW, INPUT, OUTPUT, INPUT_PULLUP, LED_BUILTIN, true, false, integer constants, and floating point constants) and 'Data Types' (listing void); and 'Functions' with sub-sections for 'Digital I/O' (listing pinMode(), digitalWrite(), and digitalRead()) and 'Analog I/O' (listing analogReference(), analogRead(), and analogWrite() - PWM).

Arduino **Genuino**
USA ONLY OUTSIDE USA

Search the Arduino Website

Home Buy Download Products **Learning** Forum Support Blog LOG IN SIGN UP

Reference Language | Libraries | Components

Language Reference

Arduino programs can be divided in three main parts: *structure*, *values* (variables and constants), and *functions*.

Structure

- `setup()`
- `loop()`

Control Structures

- `if`
- `if...else`
- `for`
- `switch case`
- `while`

Variables

Constants

- `HIGH` | `LOW`
- `INPUT` | `OUTPUT` | `INPUT_PULLUP`
- `LED_BUILTIN`
- `true` | `false`
- integer constants
- floating point constants

Data Types

- `void`

Functions

Digital I/O

- `pinMode()`
- `digitalWrite()`
- `digitalRead()`

Analog I/O

- `analogReference()`
- `analogRead()`
- `analogWrite()` - PWM

Page# on this page in a new tab

<https://www.arduino.cc/en/Reference/HomePage>

Arduino IDE Kurulumu

[Home](#)[Buy](#)[Download](#)[Products](#)[Learning](#)[Forum](#)[Support](#)[Blog](#)[LOG IN](#)[SIGN UP](#)

DOWNLOAD

ENGLISH

Download the Arduino Software



ARDUINO 1.6.12

The open-source Arduino Software (IDE) makes it easy to write code and upload it to the board. It runs on Windows, Mac OS X, and Linux. The environment is written in Java and based on Processing and other open-source software.

This software can be used with any Arduino board. Refer to the [Getting Started](#) page for installation instructions.

[Windows Installer](#)[Windows ZIP file for non admin install](#)[Mac OS X 10.7 Lion or newer](#)[Linux 32 bits](#)[Linux 64 bits](#)[Linux ARM \(experimental\)](#)[Release Notes](#)[Source Code](#)[Checksums \(sha512\)](#)

CONNECT, COLLABORATE, CREATE. [Learn more about the Create platform.](#)

[Try out the new
Arduino Web Editor](#)

<https://www.arduino.cc/en/Main/Software>

Klon Arduino?

<http://maker.robotistan.com/arduino-uno-suruculeri-nasil-yuklenir-ch340-ciqli-klon/>

S4A Kurulumu



The screenshot shows the S4A website homepage. At the top is a dark blue header with the S4A logo (two overlapping circles, one orange with 'S4' and one blue with 'A') on the left and the Citilab logo on the right. Below the header is a navigation bar with links: About, Docs, Android, Changelog, Downloads, Kit, FAQ, Team, Contact, Snap!, and Blog. The main content area has a light gray background. On the left, there's a section titled 'About S4A' with a paragraph describing it as a Scratch modification for Arduino programming. Below this is another paragraph about the project's aim. On the right, there are two orange boxes. The top one is titled 'Donate to S4A' and contains a paragraph about donations and a 'Donate' button with a dropdown menu set to 'any amount' and a currency selector set to '€'. The bottom one is titled 'Hardware? Child's Play!' and contains a paragraph about S4A being a modified version of Scratch developed in 2010 by the Citilab Smalltalk Team.

About S4A

S4A is a Scratch modification that allows for simple programming of the Arduino open source hardware platform. It provides new blocks for managing sensors and actuators connected to Arduino. There is also a sensors report board similar to the PicoBoard one.

The main aim of the project is attracting people to the programming world. The goal is also to provide a high level interface to Arduino programmers with functionalities such as interacting with a set of boards through user events.

The interface



The screenshot shows the S4A Scratch interface. It features a standard Scratch workspace with a stage, a script area, and a block palette. The block palette includes S4A-specific blocks for interacting with Arduino hardware, such as 'read sensor', 'write sensor', 'digital write', and 'analog write'. The stage area shows a Scratch character and a 'Sensors' report board.

Donate to S4A

If you wish to help us continue developing and maintaining this project, you can always make a donation!

Donate of € to S4A

Hardware? Child's Play!

Scratch for Arduino (S4A) is a modified version of [Scratch](#), ready to interact with [Arduino](#) boards. It was developed in 2010 by the [Citilab Smalltalk Team](#) and

<http://s4a.cat>

S4A Kurulumu

Download and Install



Installing S4A requires you to install software both in your PC and your Arduino board. Here you'll find the detailed steps to get it up and running.

Installing S4A into your computer

S4A works in the three major consumer operating systems. Download and install the one that fits your configuration:

- Windows
- Mac
- Linux (Debian)
- Linux (Fedora) (version 1.5)
- Raspbian (Debian for RaspberryPi) (version 1.5)

<http://s4a.cat>

S4A Kurulumu

Installing the Firmware into your Arduino

This firmware is a piece of software you need to install into your Arduino board to be able to communicate with it from S4A.

- Download and install the Arduino environment by following the instructions on <http://arduino.cc/en/Main/Software>. Take in account Arduino Uno requires at least version 0022.
- Download our firmware from [here](#)
- Connect your Arduino board to a USB port in your computer
- Open the firmware file (S4AFirmware16.ino) from the Arduino environment
- In the Tools menu, select the board version and the serial port where the board is connected
- Load the firmware into your board through File > Upload

Arduino drivers

If you are a Microsoft Windows user, you may need to install the Arduino drivers into your computer:

[Arduino drivers for Microsoft Windows](#)

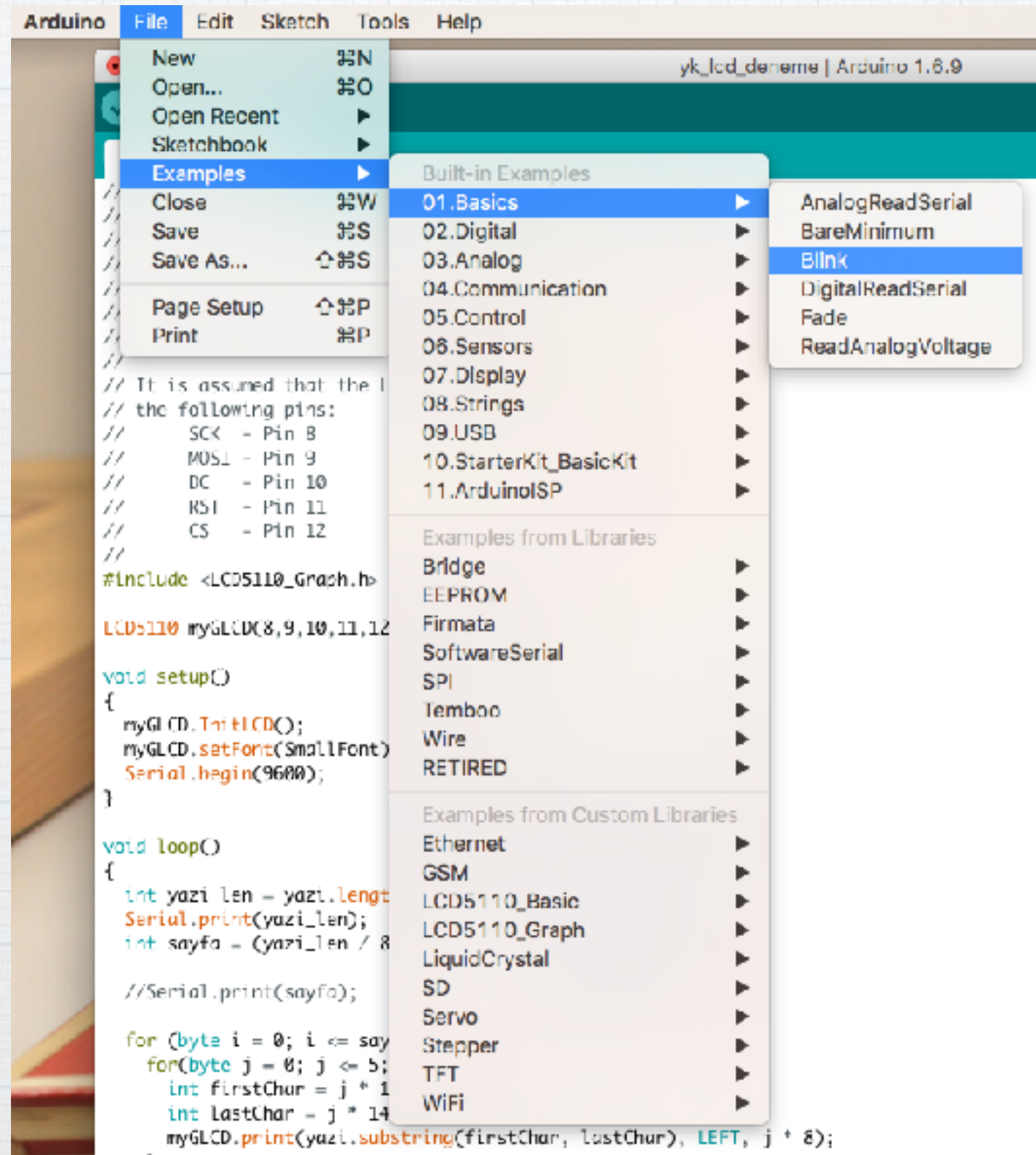
<http://s4a.cat>

“Merhaba Dünya!”

```
void setup() {  
  // Seri iletişim hızı yaklaşık olarak 1000 karakter olarak başlatılır.  
  Serial.begin(9600);  
  
}  
  
void loop() {  
  // Ekran yazı yazdırıp, satır atlıyor.  
  Serial.println("Merhaba Dünya!");  
  // Döngü içindeki bekleme zamanı  
  delay(1000);  
  
}
```

Araçlar > Serial Port Ekranı

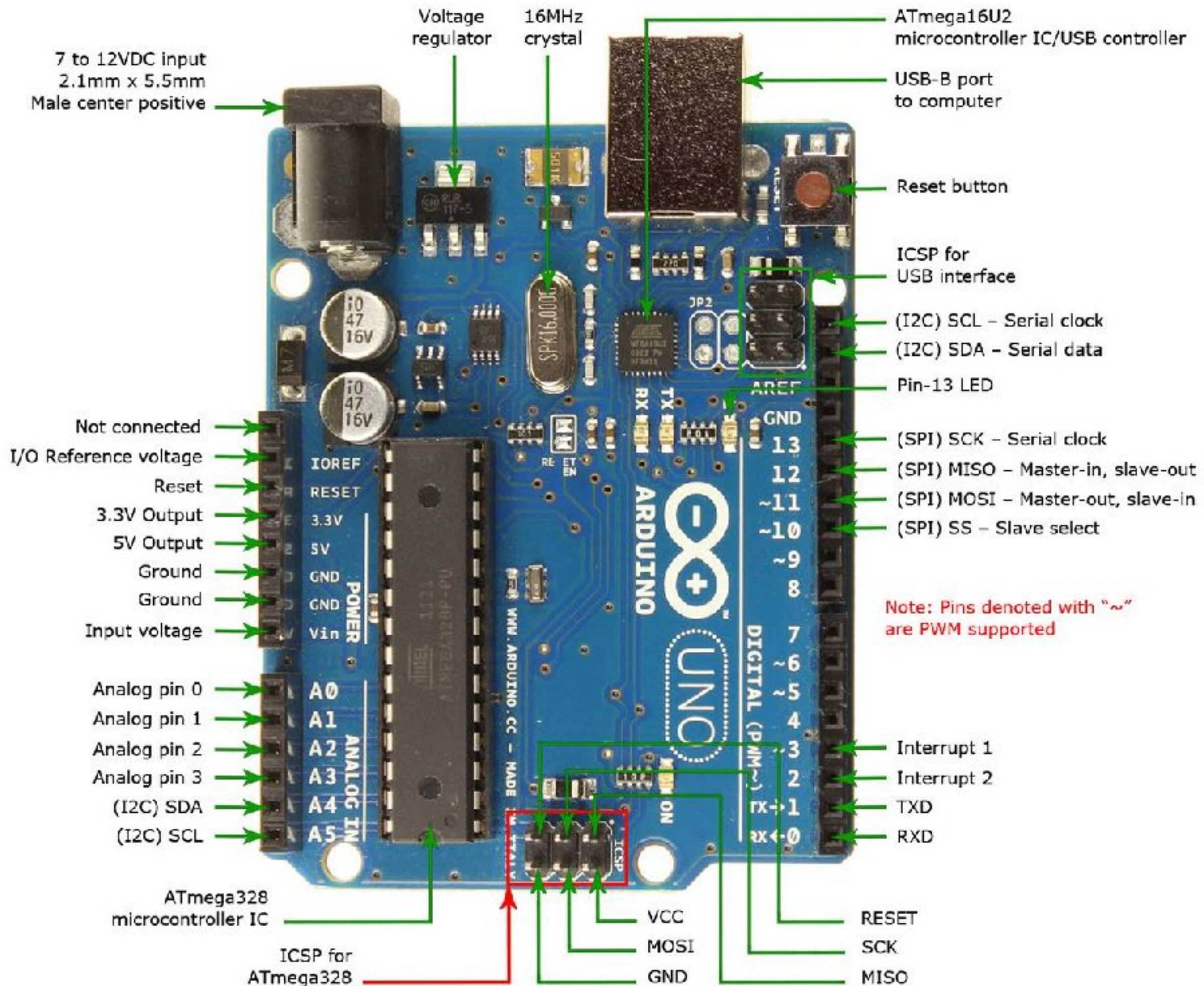
LED YAKIP SÖNDÜRME



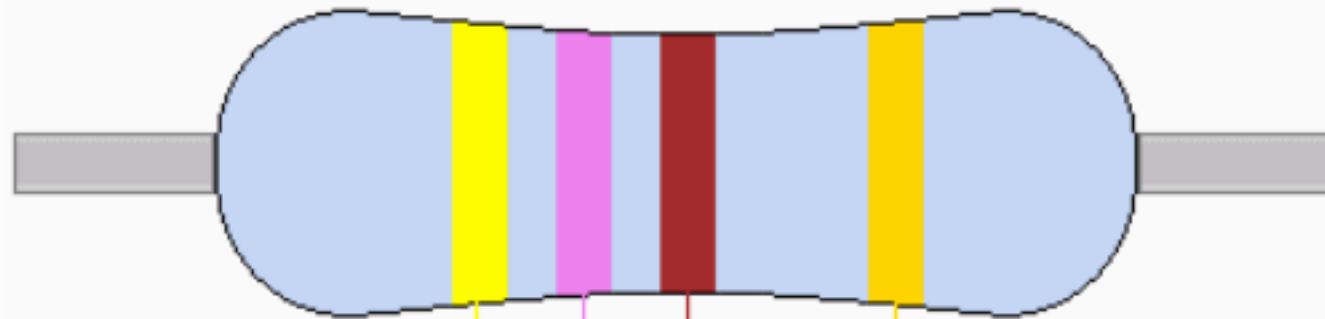
Arduino ile LED Yakma

Gerekenler;

- * Arduino UNO
- * Breadboard
- * 1 Adet 470 Ohm Direnç
- * Jumper Kablolar
- * İlgil ve Merak!

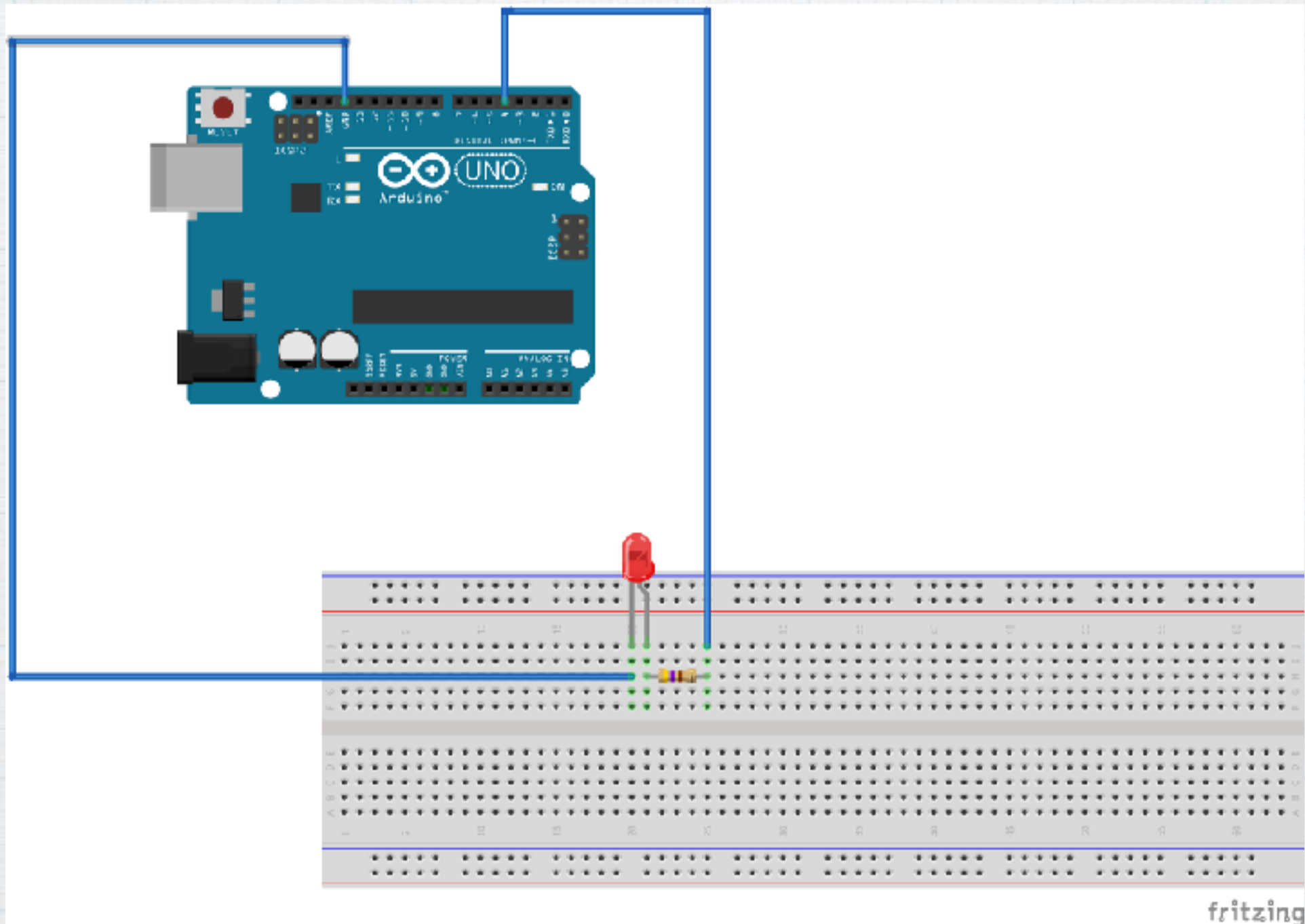


470R / 470 ohm Resistor Colour Code



	Band			Multiplier	Tolerance
	1	2	3		
Black	0	0	0	1	-
Brown	1	1	1	10	±1%
Red	2	2	2	100	±2%
Orange	3	3	3	1000	-
Yellow	4	4	4	10 000	-

Arduino ile LED Yakma (Fritzing Şema)



Arduino ile LED Yakma (Kod)

```
void setup() {  
  // LED'e güç vereceğimiz çıkış pini belirtiliyor.  
  pinMode(4, OUTPUT);  
  
}
```

```
void loop() {  
  // 4 numaralı pine güç veriyoruz.  
  digitalWrite(4, HIGH);  
  
}
```

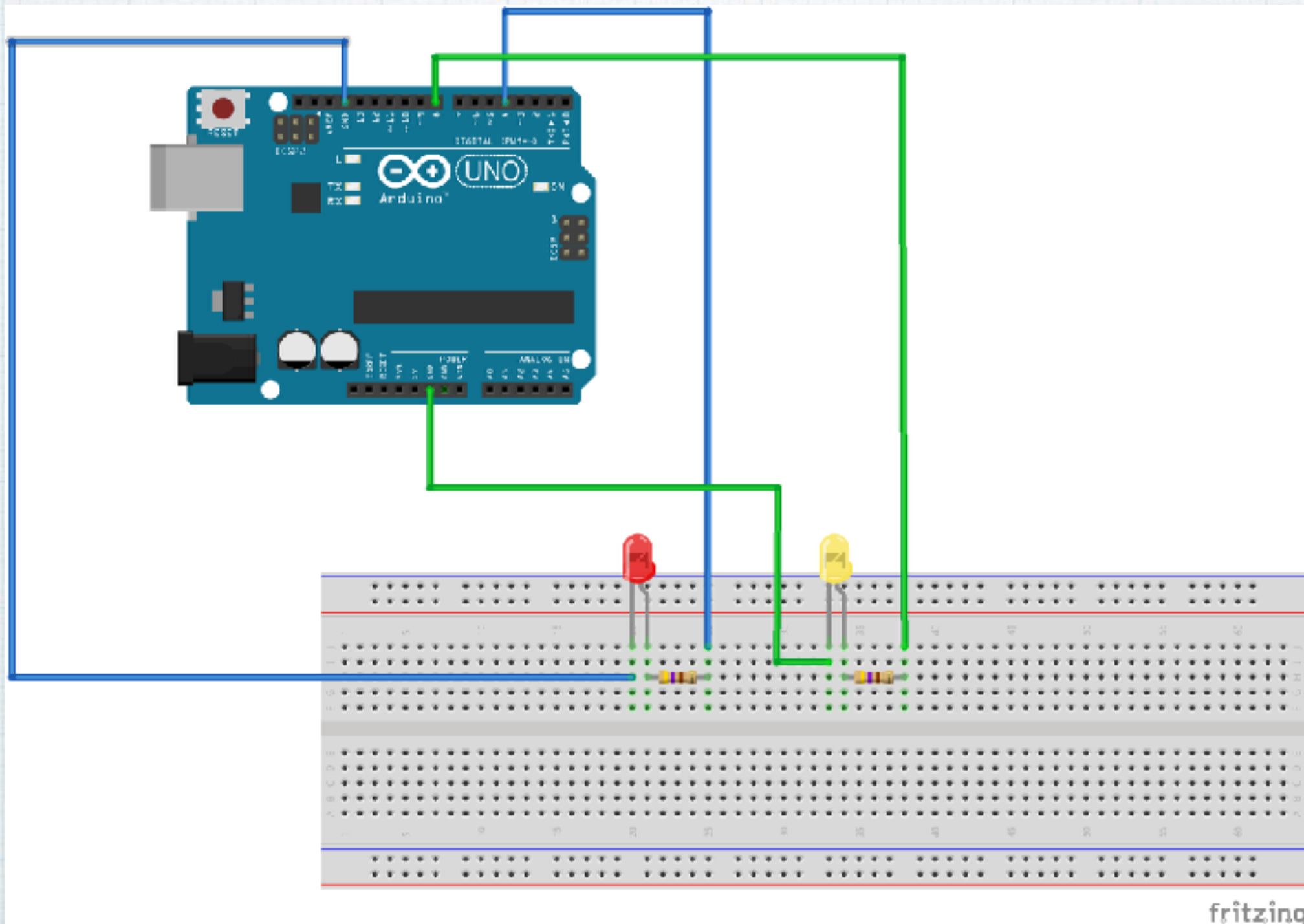

Arduino ile LED Yakma (Kod, Blink)

```
void setup() {  
  // LED'e güç vereceğimiz çıkış pini belirtiliyor.  
  pinMode(4, OUTPUT);  
  
}
```

```
void loop() {  
  // 4 numaralı pine güç veriyoruz.  
  digitalWrite(4, HIGH);  
  // Gecikme zamanı.  
  delay(500);  
  // 4 numaralı pinden gücü kesiyoruz.  
  digitalWrite(4, LOW);  
  delay(500);  
  
}
```

Challenge!

Ardışık LED Yakıp Söndürme!



Challenge!

Ardışık LED Yakıp Söndürme!

(Kod)

* To be continued...