

# Introduction to Arduino

## Workshop by Varvara Guljajeva & Mar Canet Sola

### 19.11.2014 at TopConf, Tallinn

#### 1. [What is Arduino?](#)

(<https://vimeo.com/13781339> )

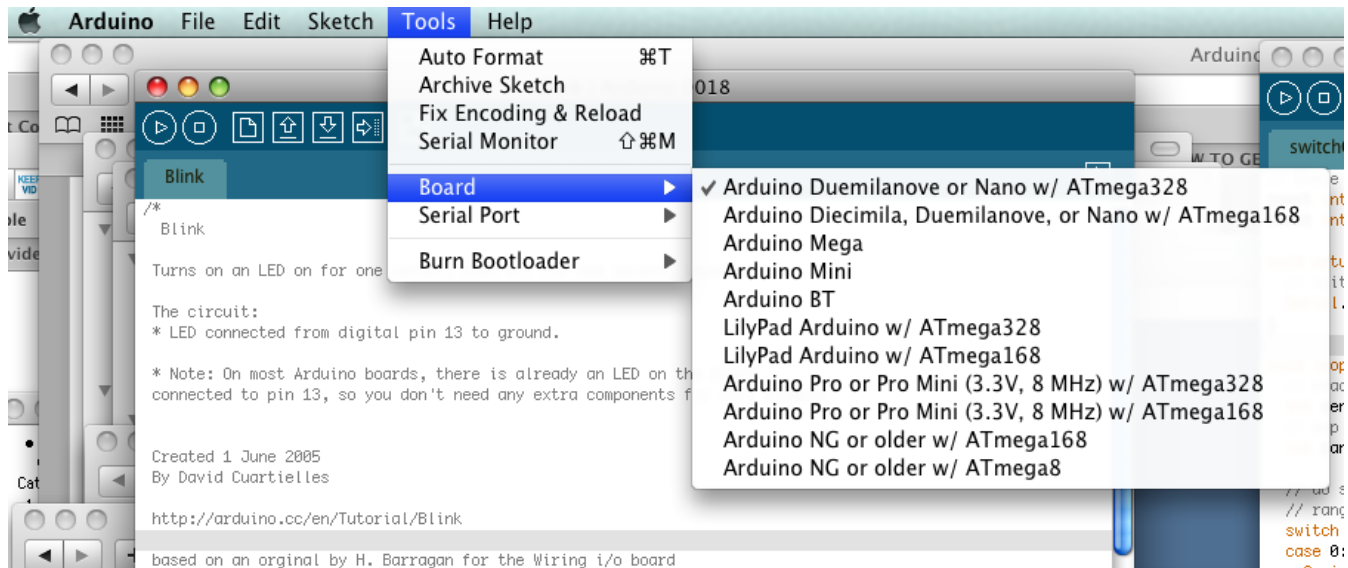
---

### INSTALL ARDUINO ENVIRONMENT

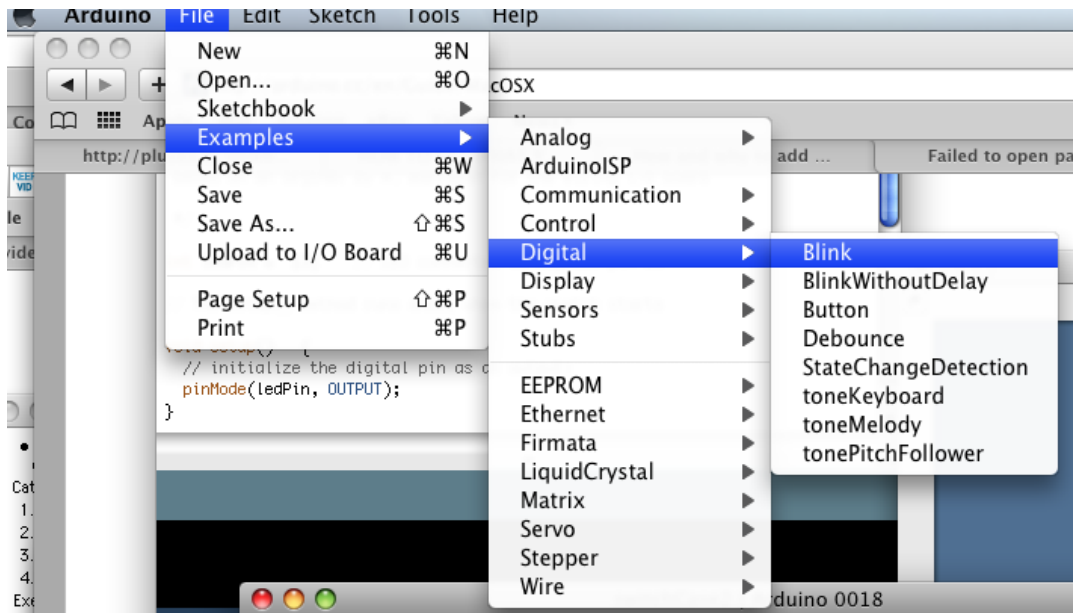
---

#### Arduino:

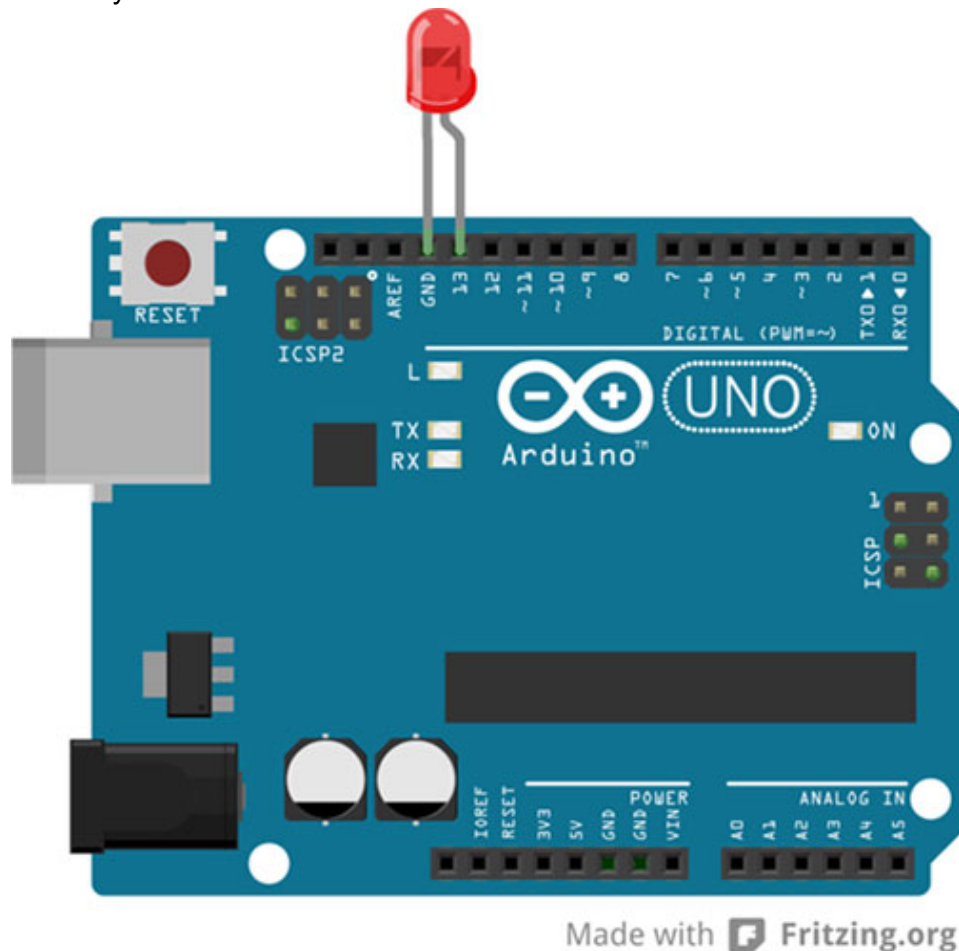
1. If you do not have Arduino program installed, do so by following this link:  
<http://arduino.cc/en/Main/Software>
2. Make sure that you have as well [FTDI driver installed](#)
3. connect Arduino to your computer
4. start Arduino program and select right BOARD and SERIAL under menu TOOLS



5. upload blink example in order to check that you did everything right (press Upload, second button in the upper left corner).



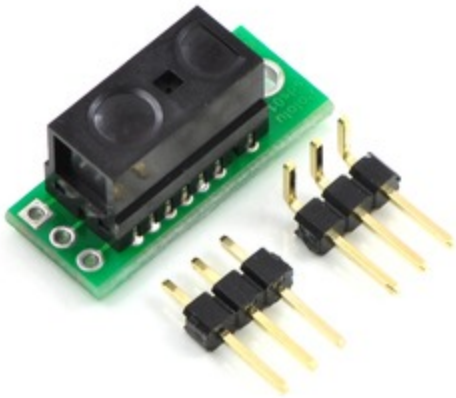
6. Insert LED to pin 13 and GND to ensure that everything is functioning as the code say.



7. Make the LED blink differently from the example one.

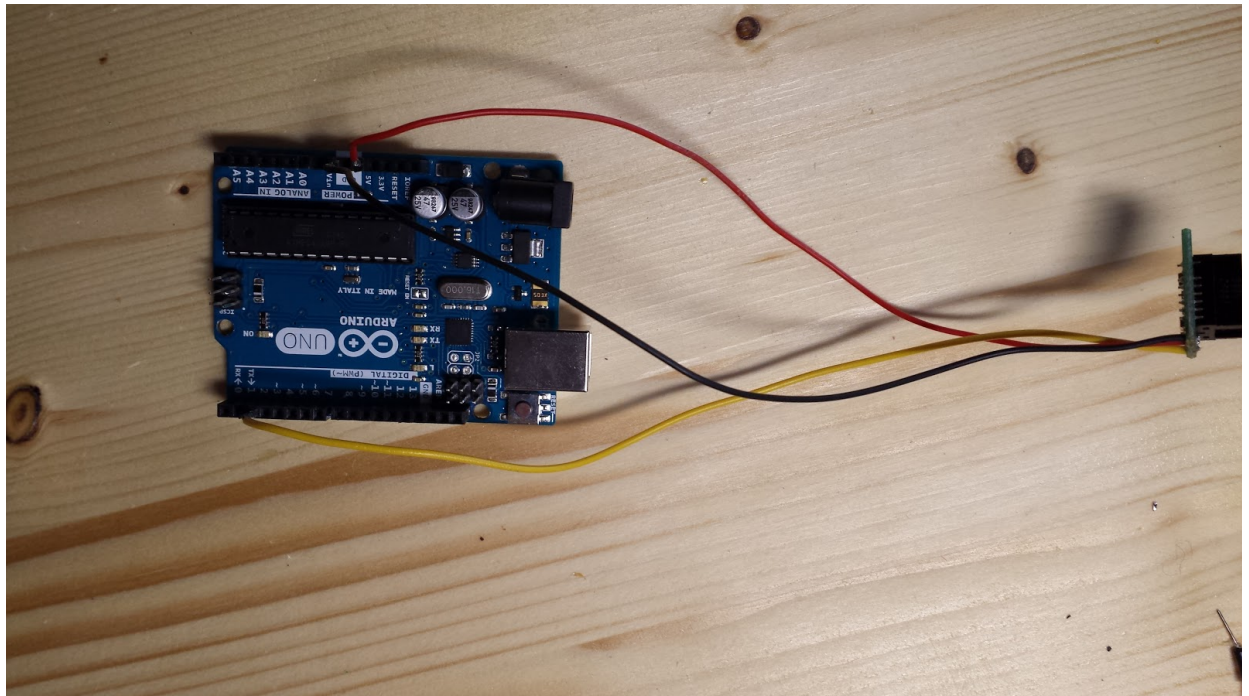
Varvara Guljajeva - [varvarag@gmail.com](mailto:varvarag@gmail.com) - [www.varvarag.info](http://www.varvarag.info)  
 Mar Canet Sola - [mar.canet@gmail.com](mailto:mar.canet@gmail.com) - [www.mcanet.info](http://www.mcanet.info)

## SHARP Distance sensor GP2Y0D805Z0F

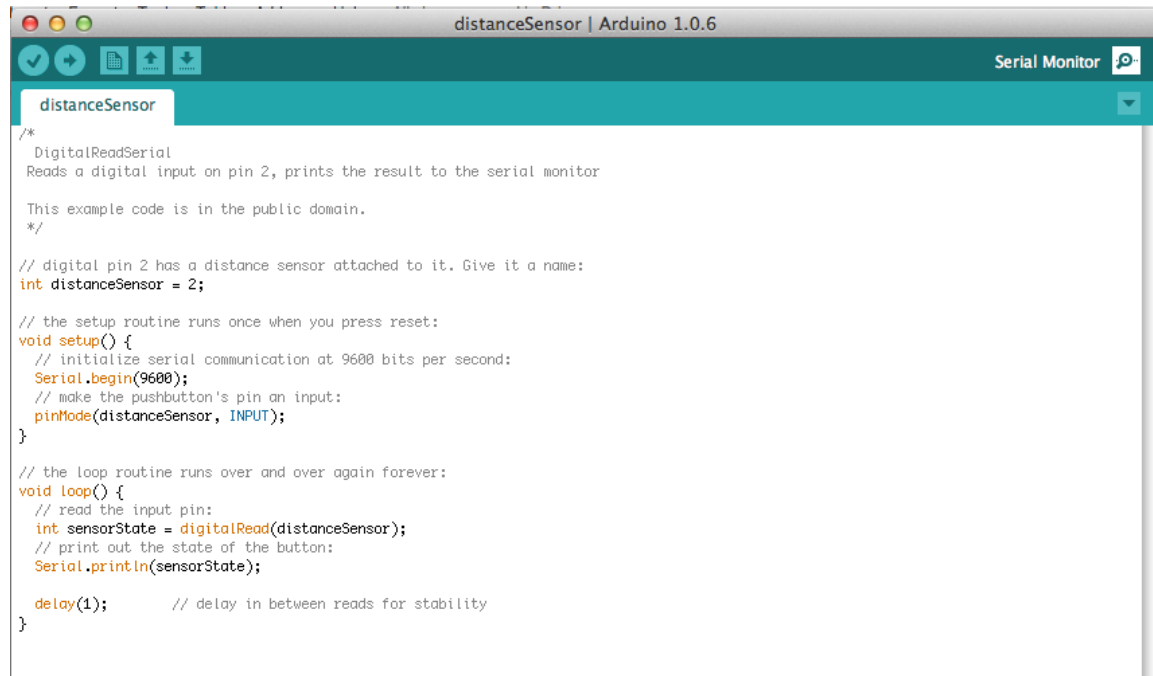


This small digital distance sensor detects objects between 0.5 cm and 5 cm (0.2" and 2") away. It gives LOW signal to Arduino when an object is detected in its range.

1. Connect sensor to Arduino: RED to pin 5V, BLACK to pin GND, and YELLOW to pin2.



2. upload distance sensor code to your arduino (the same way how you did for the LED).  
Code you find here:  
[https://github.com/var-mar/workshop\\_topconf\\_introduction\\_arduino/tree/master/code](https://github.com/var-mar/workshop_topconf_introduction_arduino/tree/master/code)
3. Open serial monitor (upper right corner) and see how it reacts of the objects in and out of it range.

The image shows a screenshot of the Arduino IDE's Serial Monitor window. The title bar at the top reads "distanceSensor | Arduino 1.0.6". Below the title bar is a toolbar with icons for running, stopping, and other functions. The main area of the window displays the source code for a program named "distanceSensor". The code is a C++ sketch that uses the DigitalReadSerial library to read a digital input on pin 2 and prints the result to the serial monitor. It includes a setup routine for initializing serial communication and pin modes, and a loop routine for continuously reading the sensor state and printing it. A delay of 1 unit is used between reads for stability. The code is as follows:

```
/*
  DigitalReadSerial
  Reads a digital input on pin 2, prints the result to the serial monitor

  This example code is in the public domain.
  */

// digital pin 2 has a distance sensor attached to it. Give it a name:
int distanceSensor = 2;

// the setup routine runs once when you press reset:
void setup() {
  // initialize serial communication at 9600 bits per second:
  Serial.begin(9600);
  // make the pushbutton's pin an input:
  pinMode(distanceSensor, INPUT);
}

// the loop routine runs over and over again forever:
void loop() {
  // read the input pin:
  int sensorState = digitalRead(distanceSensor);
  // print out the state of the button:
  Serial.println(sensorState);

  delay(1);      // delay in between reads for stability
}
```

**Challenge:**

Make the LED to light up when sensor detects an object in its range.

## SERIAL COMMUNICATION

Receive serial values that Arduino is sending in Processing or any different programming environment.

1. Download and install Python  
<https://www.python.org/downloads/>
  2. Install the pySerial dependency: <http://pyserial.sourceforge.net/pyserial.html>
  3. Run the code for read a sensor in python
- PSS. close Arduino serial monitor before running Python code

A screenshot of a code editor with two tabs: 'receive\_sensor.py' and 'arduino\_serial\_led'. The 'receive\_sensor.py' tab is active, showing the following Python code:

```
1 import serial
2
3 ser = serial.Serial('COM3', 9600)
4 while True:
5     print ser.readline()
```

5. understand and modify the code

## CHALLENGE

Use SimpleWrite example for lighting up LED.