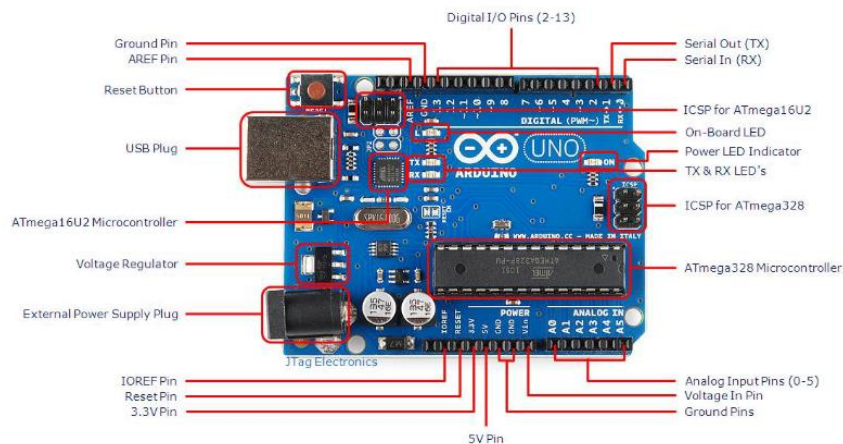


## Instrumentation & Control

In this lab we will introduce the idea of using a microcontroller as a tool for interfacing with sensor and carrying out conditioning on the data. The decision on the state of the system is based on the digital or analog input to the microcontroller device and based on the codes that we can write and compile into the microcontroller. We will use Arduino Uno R3 as an 8-bit and simple microcontroller. The Arduino kits can be used both in the lab and at home to perform experiments and build projects. The Arduino kit also includes a breadboard for fast and easy circuit assembly.



## LAB

**Create a digital lab book document detail each task you are doing. Submit this via an email to the instructor at the end of the lab.**

## Task 1 – What is Arduino

Read the introduction to Arduino from their official website. This website is a great resource for all things related to Arduino.

<https://www.arduino.cc/en/guide/introduction>

## Task 2 – Arduino development environment

Installing the Integrated development environment (IDE) if is not already installed locally.

Go to the website and download the IDE.

Follow the <https://www.arduino.cc/en/Guide>

## Task 3 – Plugin the board

[https://www.tutorialspoint.com/arduino/arduino\\_installation.htm](https://www.tutorialspoint.com/arduino/arduino_installation.htm)

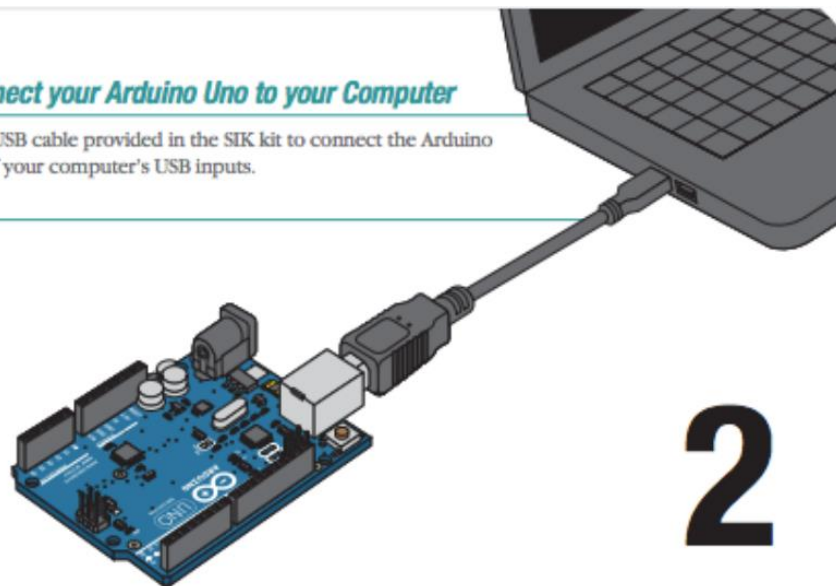
---

*// Connect your Arduino Uno to your Computer*

---

Use the USB cable provided in the SIK kit to connect the Arduino to one of your computer's USB inputs.

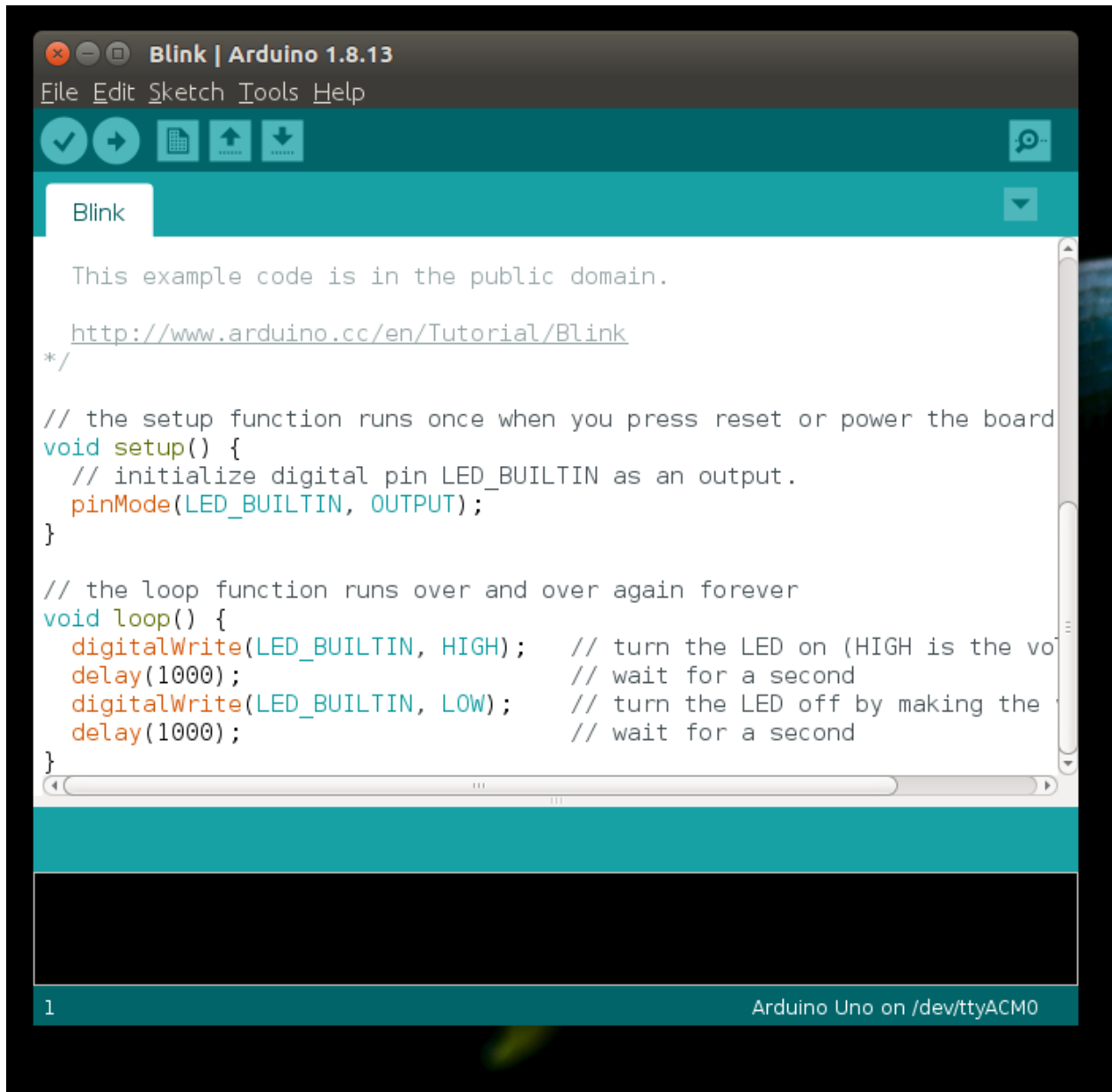
---



2

## Task 4 – Build your first program

Build run the following example program and you will have a blinking LED on your board.

The image shows a screenshot of the Arduino IDE interface. The title bar reads "Blink | Arduino 1.8.13". The menu bar includes "File", "Edit", "Sketch", "Tools", and "Help". Below the menu bar is a toolbar with icons for checking, running, sketching, uploading, and downloading. The main text area displays the "Blink" program code. The code includes a comment about the public domain, a URL to the Arduino tutorial, and the C++ code for the setup and loop functions. The setup function initializes the built-in LED pin as an output. The loop function turns the LED on for one second and off for one second. The status bar at the bottom shows "1" on the left and "Arduino Uno on /dev/ttyACM0" on the right.

```
Blink | Arduino 1.8.13
File Edit Sketch Tools Help

This example code is in the public domain.

http://www.arduino.cc/en/Tutorial/Blink
*/

// the setup function runs once when you press reset or power the board
void setup() {
  // initialize digital pin LED_BUILTIN as an output.
  pinMode(LED_BUILTIN, OUTPUT);
}

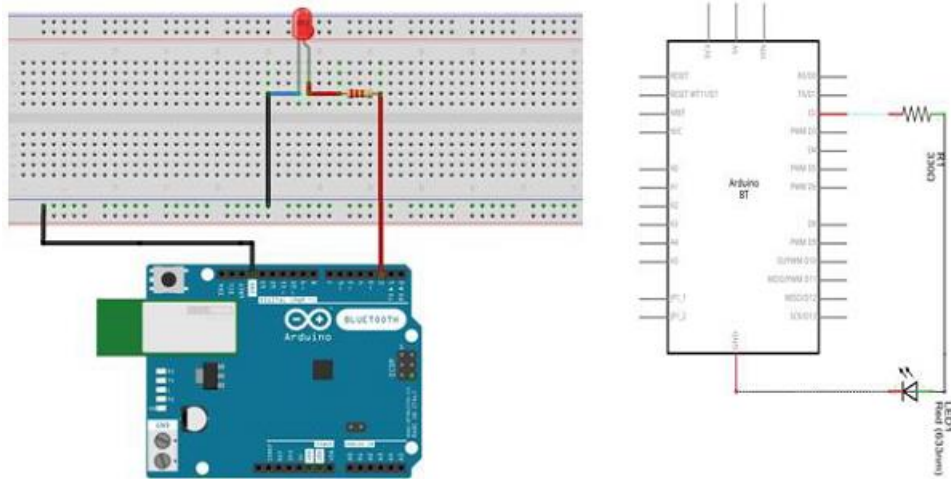
// the loop function runs over and over again forever
void loop() {
  digitalWrite(LED_BUILTIN, HIGH); // turn the LED on (HIGH is the voltage level)
  delay(1000);                     // wait for a second
  digitalWrite(LED_BUILTIN, LOW);  // turn the LED off by making the voltage LOW
  delay(1000);                     // wait for a second
}

1 Arduino Uno on /dev/ttyACM0
```

## Task 5

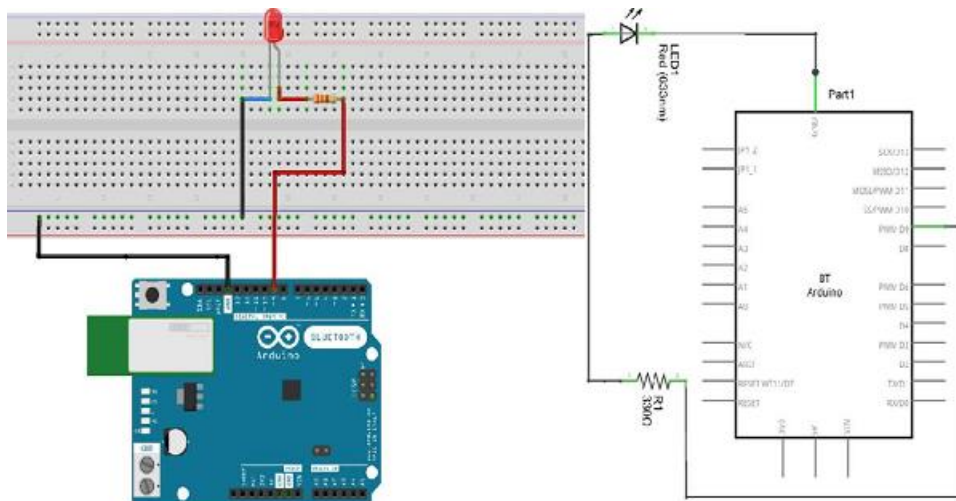
After you have your board setup you are ready for your first embedded program that will light up your own LED.

[https://www.tutorialspoint.com/arduino/arduino\\_blinking\\_led.htm](https://www.tutorialspoint.com/arduino/arduino_blinking_led.htm)



## Task 6

[https://www.tutorialspoint.com/arduino/arduino\\_fading\\_led.htm](https://www.tutorialspoint.com/arduino/arduino_fading_led.htm)



## Task 7

Expand on task 5,6 and two extra LEDs.

1. Show that you can

[https://www.tutorialspoint.com/arduino/arduino\\_led\\_bar\\_graph.htm](https://www.tutorialspoint.com/arduino/arduino_led_bar_graph.htm)

## Task 8

Add a pot to the board

2. Show that you can do

[https://www.tutorialspoint.com/arduino/arduino\\_reading\\_analog\\_voltage.htm](https://www.tutorialspoint.com/arduino/arduino_reading_analog_voltage.htm)

