

# **Arduino 101**

---

## **Introduction and Programming**

Midcoast Mini Maker's Faire

9/12/2015

## **What is it????**

The Arduino Uno is a single board computer which has a microprocessor, memory, communication, variable voltage reading and output capability, and digital input/output all on one circuit board.

Once programmed it executes the program every time power is applied. It does not lose its programming if it loses power.

**And you get it all for less than \$20!!!**

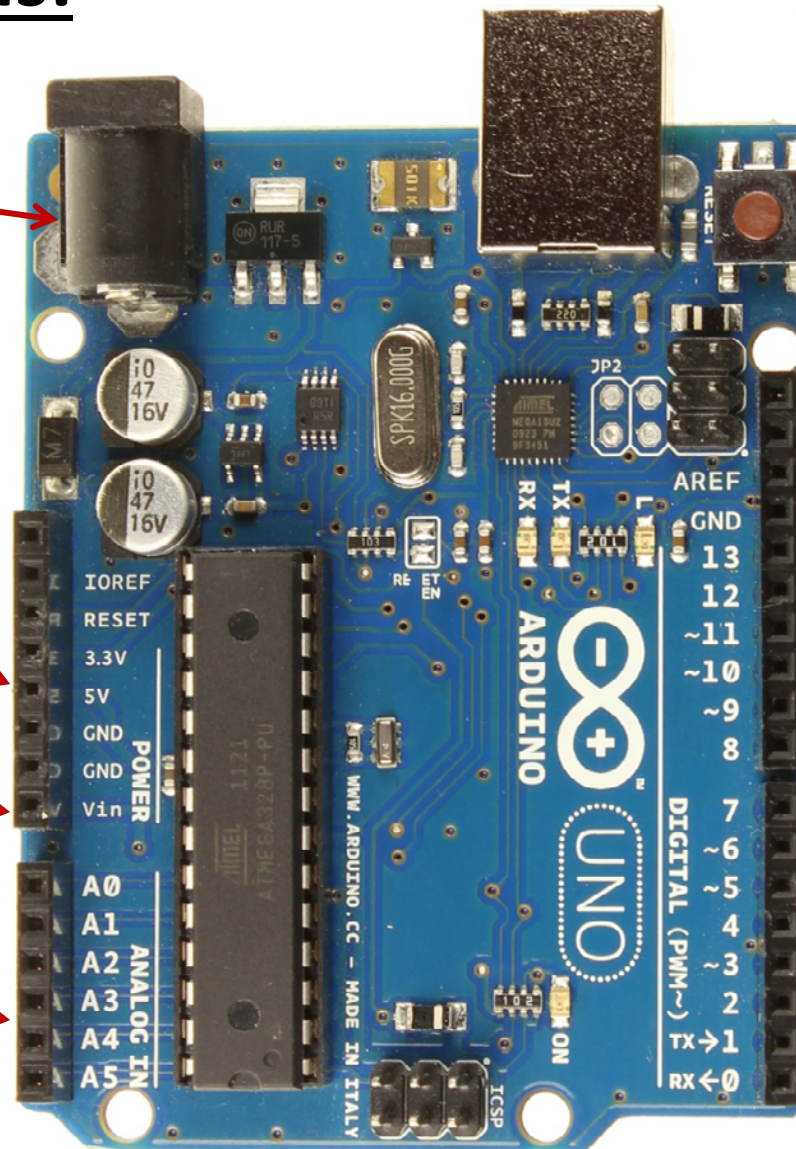
# And here it is:

External 9-12V  
power input

Power output for  
things like variable  
resistance sensors

External 9-12V  
power input

Pins to read  
variable voltages



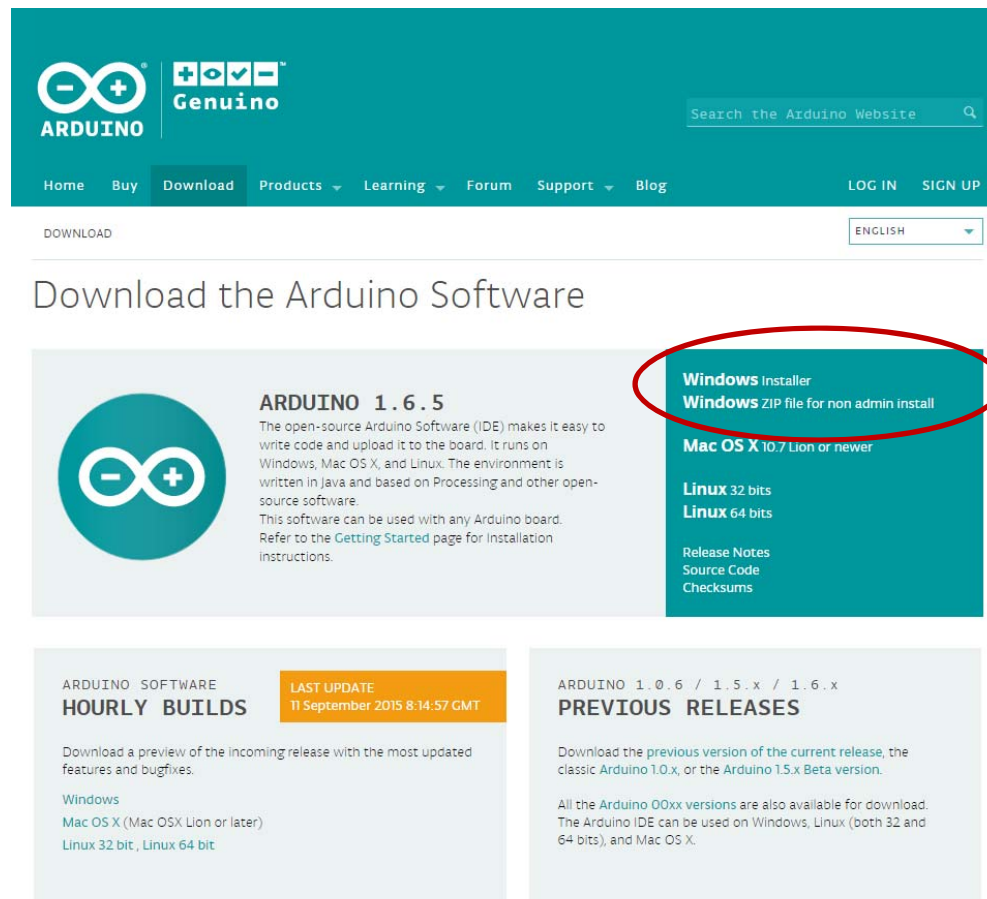
Digital  
input/output pins.  
Use these to signal  
to turn things on  
or off, read if a  
switch is open or  
closed, etc.

PWM or pulse  
width modulation  
allows the Uno to  
make variable  
voltages to do  
things like dim a  
light or change the  
speed of a motor.

# Cool!! So how do I get started??

Programming the Arduino requires a computer with a USB port. It can be a Mac or a PC. First go to:

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



The screenshot shows the Arduino website's software download page. The header includes the Arduino and Genuino logos, a search bar, and navigation links. The main heading is "Download the Arduino Software". Below this, there's a large section for "ARDUINO 1.6.5" with a description and a list of download links. A red circle highlights the "Windows Installer" and "Windows ZIP file for non admin install" links. To the right of this circle, a red arrow points to the text "Click here and install the FREE programming software". Below the main section, there are two smaller sections: "ARDUINO SOFTWARE HOURLY BUILDS" and "ARDUINO 1.0.6 / 1.5.x / 1.6.x PREVIOUS RELEASES".

ARDUINO 1.6.5

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

Release Notes  
Source Code  
Checksums

ARDUINO SOFTWARE  
HOURLY BUILDS

LAST UPDATE  
11 September 2015 8:14:57 GMT

Download a preview of the incoming release with the most updated features and bugfixes.

Windows  
Mac OS X (Mac OSX Lion or later)  
Linux 32 bit, Linux 64 bit

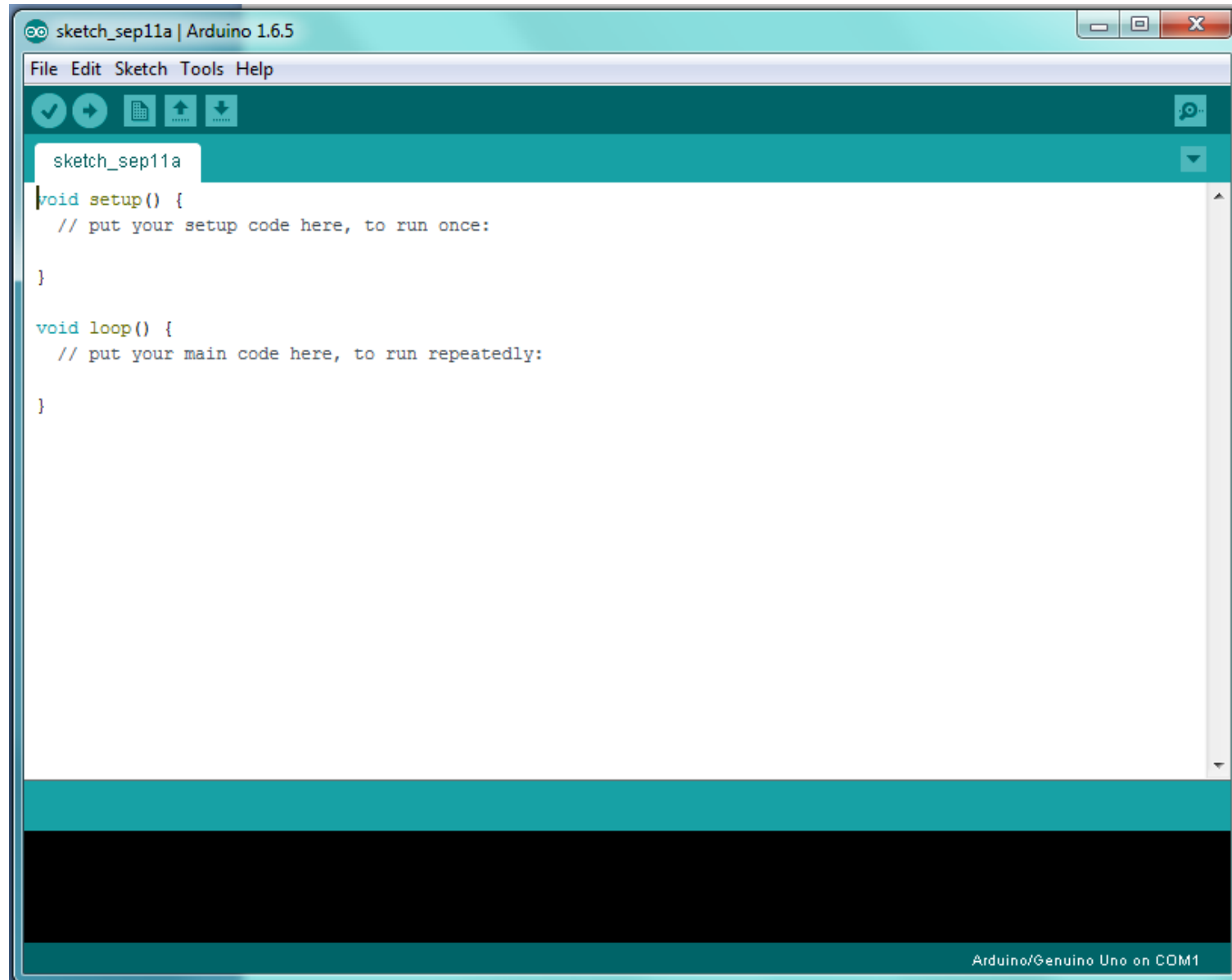
ARDUINO 1.0.6 / 1.5.x / 1.6.x  
PREVIOUS RELEASES

Download the previous version of the current release, the classic Arduino 1.0.x, or the Arduino 1.5.x Beta version.

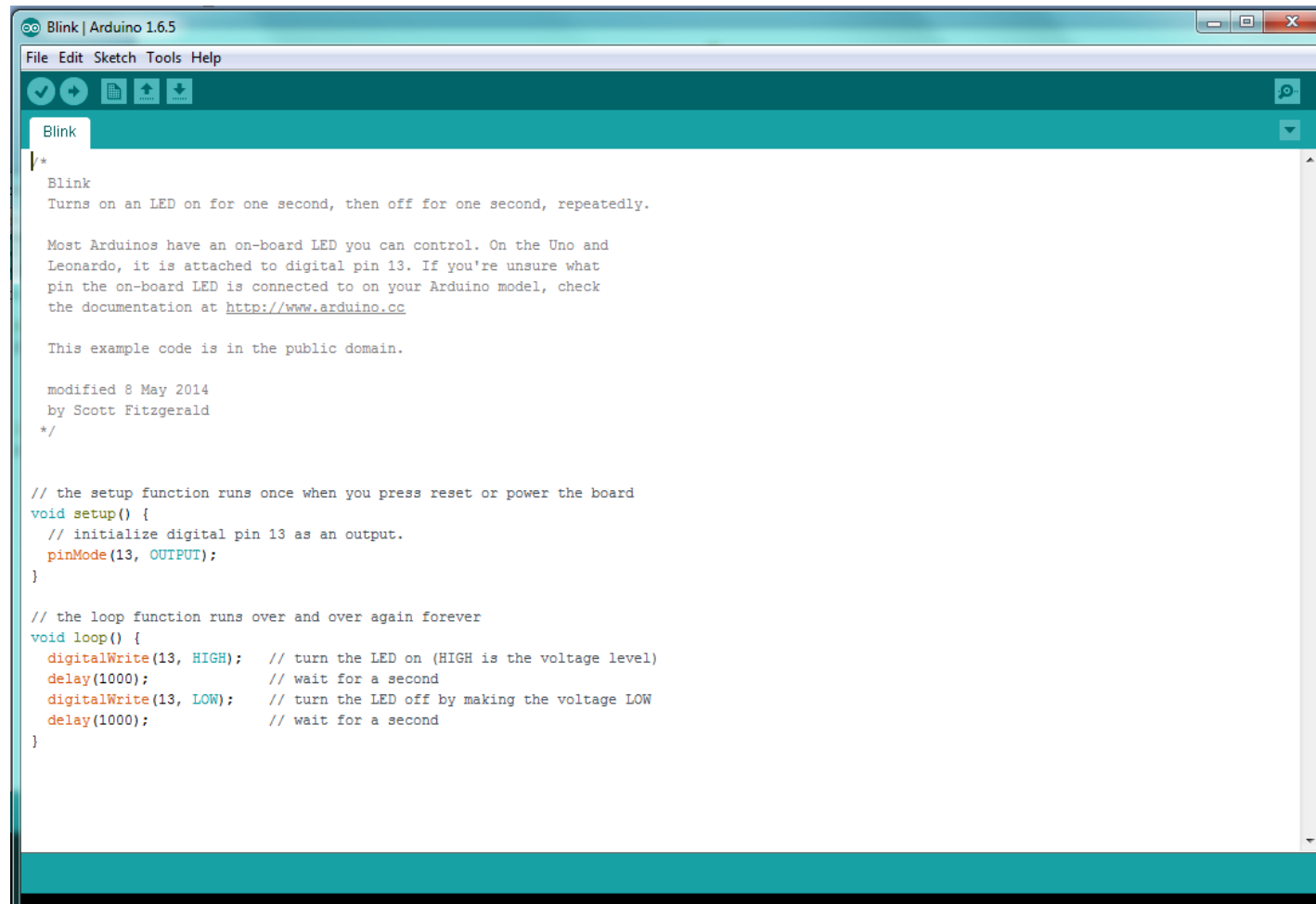
All the Arduino 00xx versions are also available for download. The Arduino IDE can be used on Windows, Linux (both 32 and 64 bits), and Mac OS X.

Click here and  
install the  
**FREE**  
programming  
software

Once the software is installed you access it like any other program  
(ie. Start > Arduino)



Under Files > Examples there are many examples which you can hack. The easiest is probably “Blink”

A screenshot of the Arduino IDE interface. The title bar reads "Blink | Arduino 1.6.5". The menu bar includes "File", "Edit", "Sketch", "Tools", and "Help". Below the menu bar is a toolbar with icons for opening files, saving, and other functions. The main text area displays the "Blink" example code. The code is a C++ program that turns an LED on and off every second. It includes a multi-line comment at the top explaining the purpose of the sketch and providing information about the on-board LED and the code's origin. The code is structured with a setup function and a loop function. The setup function initializes digital pin 13 as an output. The loop function writes HIGH to pin 13, delays for 1000ms, writes LOW to pin 13, and delays for 1000ms again.

```
Blink
/*
  Blink
  Turns on an LED on for one second, then off for one second, repeatedly.

  Most Arduinos have an on-board LED you can control. On the Uno and
  Leonardo, it is attached to digital pin 13. If you're unsure what
  pin the on-board LED is connected to on your Arduino model, check
  the documentation at http://www.arduino.cc

  This example code is in the public domain.

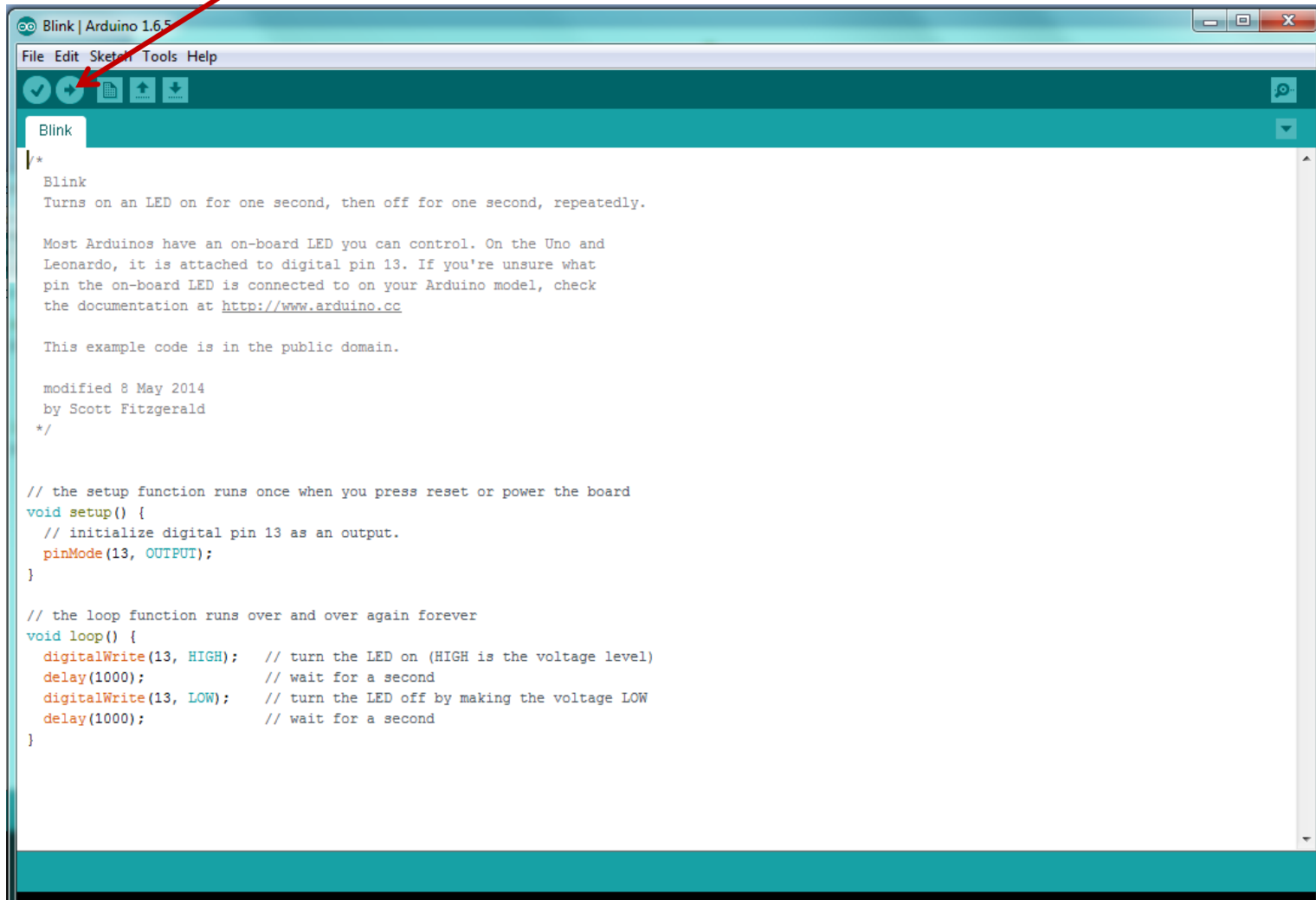
  modified 8 May 2014
  by Scott Fitzgerald
  */

// the setup function runs once when you press reset or power the board
void setup() {
  // initialize digital pin 13 as an output.
  pinMode(13, OUTPUT);
}

// the loop function runs over and over again forever
void loop() {
  digitalWrite(13, HIGH);  // turn the LED on (HIGH is the voltage level)
  delay(1000);             // wait for a second
  digitalWrite(13, LOW);   // turn the LED off by making the voltage LOW
  delay(1000);             // wait for a second
}
```

The Arduino environment uses the standard “C” language.

To upload the code into the Arduino simply attach the Arduino via a USB cable and click this button



There is no need to compile the code before uploading it. All of the power needed to program the Arduino is supplied by the computer's USB port.

Now whenever the Uno is connected to a 9V battery, the board mounted LED should flash.