

Introduction to Arduino[®] Programming

Introduction

Microcontrollers are electronic devices that are programmed to perform a set of instructions. They complete any given task as often as prompted until they are reprogrammed. You can find them everywhere, from lights that automatically turn on when it's dark to garage door openers that respond to sensor input to open, close, or reverse direction.

Arduino[®] is a versatile microcontroller that can easily be programmed. It was originally designed as a low-cost device to provide easy access for non-technical people to create devices that are interactive. [Arduino has an interesting history](#) as an open-source hardware and software company.

Objectives

- Understand the basic elements of an Arduino microcontroller
- Understand the basic elements of a sketch
- Create a simple sketch to control an LED

Materials

SparkFun[®] RedBoard (or equivalent) with USB cable and power supply
Computer or Chromebook[™] with Arduino software

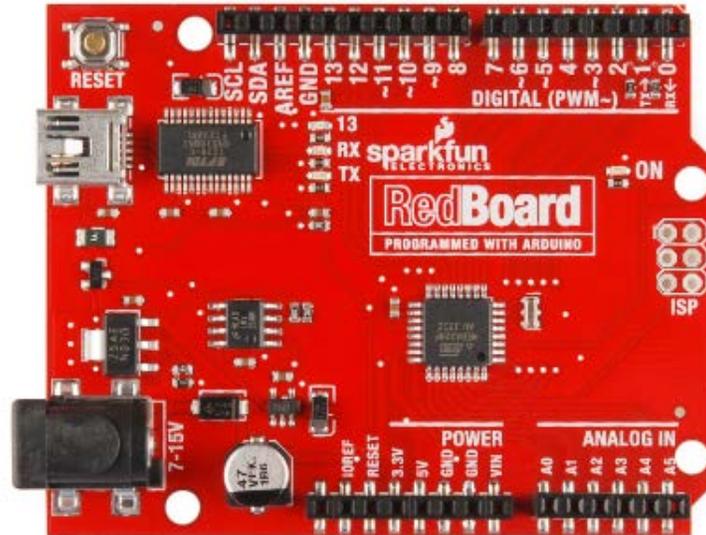
Preliminary Activity

The RedBoard has a program (called a “sketch” in Arduino terminology) installed so that it will run “out of the box.” Try these simple steps to get started.

1. Connect your external power supply to the Arduino board.
2. Locate the built-in LED that is tied to Pin 13 and observe what happens.
3. Press the Arduino’s Reset button and again observe.
4. Disconnect the power supply and then connect the Arduino to your computer using a USB cable.
5. Observe and compare to having a power supply connected.

Background

Arduino has developed both its hardware and software as open source. The SparkFun RedBoard is equivalent (although not identical) to the Arduino Uno, and when using the Arduino software, it should be treated as an Uno.



Identify the following elements on your RedBoard: Reset button, USB connector, external power port, Power pins, Analog In pins, Digital pins, LED 13, and the TX and RX communication LEDs.

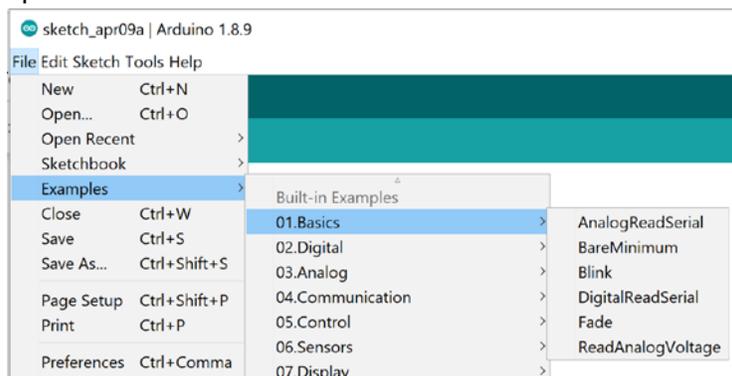
If you are not using a RedBoard, the parts may appear slightly different.

Procedure

Download and install the Arduino IDE (Integrated Development Environment) from <https://www.arduino.cc/en/Main/Software> or create an account for the Arduino Web Editor. If you are using a Chromebook, you will need to download and use the Arduino Web Editor and install a plug-in. The plug-in is available as a subscription from the Chrome Web Store. The Arduino Web Editor has processes that will vary slightly from those provided in these activities.

While you are at the Arduino website, it may be helpful to bookmark the Reference pages (<https://www.arduino.cc/reference/en/>). They will be very useful as you work on these activities.

After the download and installation is complete, open the Arduino IDE. The sketch that is installed on the Arduino is called Blink. To start, let's understand how the Blink program works. Click on File > Examples > 01.Basics > Blink



You will see three sections in this sketch (Arduino's terminology for a program):

1. A set of comments bracketed by these characters: `/* ...*/`
Note: These lines are ignored when running the sketch and are only there to help the user understand how the program works. Single lines can be "commented" using the characters `//` in front of the part of the line to be ignored.
2. Code that starts with "void setup()"
Note: This code is referred to as the `setup()` function and runs one time when the program is started. It can be used to specify communication protocols and to set the `pinMode` (whether certain pins are treated as inputs or outputs). You can also provide preliminary information to be printed in the Serial Monitor.
3. Code that starts with "void loop()"
Note: This code is referred to as the `loop()` function and will run repeatedly after the setup portion of the sketch has completed. This code will continue to run until the power is removed from the Arduino or a new program is uploaded.

Notice that there is only one line of code in the `setup()` function of this program. Answer each of the following:

The entire `setup()` function is contained between these two characters: _____

There is a line that is "commented out". It starts with these characters: _____

The line of code ends with this character: _____

Look at the `loop()` function and state whether these same rules seem to apply:

You are now ready to test your connection by following these steps:

1. Confirm that you still have your Arduino board connected to your computer using the USB cable.
2. In the Arduino IDE, click on Tools > Board and select your Arduino board. The Redboard is equivalent to the Uno, so you will select Arduino/Genuino Uno.
3. Verify that the correct COM port is selected. Click on Tools > Port and choose the COM Port used by your device. You may need to disconnect the USB cable to determine which port disappears in order to choose the correct port.
4. Change the code so that the LED stays on for 1.5 seconds and is off for 0.5 seconds.
5. Save the file in a folder location of your choosing and click the Upload button (the right-facing arrow in the toolbar). This will verify the code and upload the sketch to the Arduino board.
6. Verify that the LED flashing is working as expected.