

SEARCH THE ARDUINO WEBSITE

Getting started with Arduino Esplora

The Arduino Esplora is a device that has a tiny computer called a microcontroller, and a number of inputs and outputs. For inputs, there's a joystick, four buttons, a light sensor, a slider, a microphone, a temperature sensor, and an accelerometer. For outputs there's a buzzer and a three color LED. At first glance it looks like a videogame controller.

With the Esplora, you can write software that takes information from the inputs and use that to control the outputs on the board, or control your computer, just like a mouse or keyboard would.

The Esplora is different from all previous Arduino boards, in that it has inputs and outputs already connected to the board. This means that you don't need to know how to connect electronic sensors or actuators to use it as a result. As a result, programming for it is a bit different than for other Arduino boards. It has its own library that makes it easier to read from the input sensors and write to the output actuators. You'll see how to use the library in this guide and in the Esplora library (<http://arduino.cc/en/Reference/EsploraLibrary>) reference pages as well.

This guide will take you through the process of connecting the Esplora board to your computer and uploading an Arduino program, called a sketch.

- 1 | Get an Arduino Esplora and USB cable
- 2 | Download the Arduino environment
- 3 | Connect the board
- 4 | Install the drivers
 - Instructions for Mac OS X
 - Instructions for Windows
 - Instructions for Linux
- 5 | Programming the board
 - Open the Esplora Blink example
 - Configure Arduino for Esplora
 - Select the right USB port
 - Upload the program
- 6 | Change the RGB LED
- 7 | Inputs changing outputs
- 8 | Next steps

1 | Get an Arduino Esplora and USB cable

This tutorial is for the Arduino Esplora (<http://arduino.cc/en/Main/ArduinoBoardEsplora>). If you have another board, read the corresponding Getting Started guide (<http://arduino.cc/en/Guide/HomePage>) for that board.

You also need a proper Micro-USB cable ("A" plug to "Micro-B" plug type). Some mobile phones or portable music players use this kind of cable to transfer data to/from the PC, so you might already own one of them. This is different from a USB-Mini cable in that it's a slimmer connector. You can see the differences here (http://en.wikipedia.org/wiki/File:Usb_connectors.JPG) (the connector you need is the one on the left in this photo).

2 | Download the Arduino environment

Get the latest version of the Arduino IDE from the download page (<http://arduino.cc/en/Main/Software>). You'll need version 1.0.3 or later of the Arduino IDE.

When the download finishes, unzip the downloaded file in any directory. Make sure to preserve the folder structure. Double-click the folder to open it. There should be a few files and sub-folders inside. If you're on a Mac, the zip folder contains just the application.

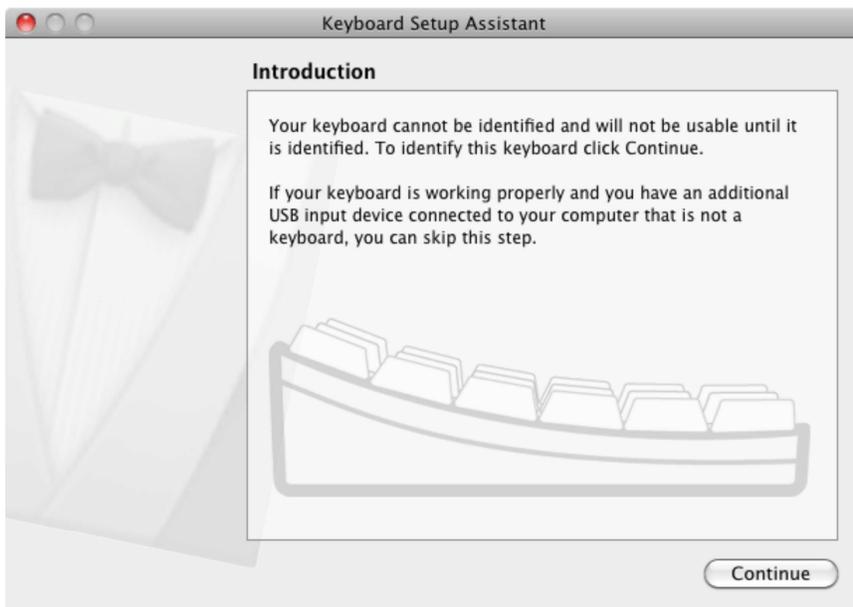
3 | Connect the board

Connect the Arduino board to your computer using the USB cable. The green power LED (labelled ON) should go on, and the yellow LED marked "L" should start glowing. After about 8 seconds, the yellow LED should start blinking on and off.

4 | Install the drivers

Instructions for Mac OS X

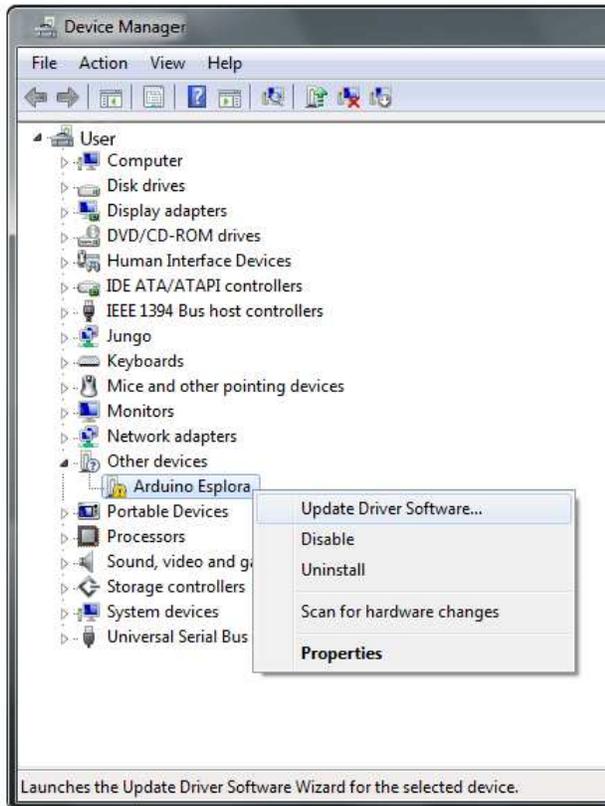
- The first time you plug an Esplora into a Mac, the "Keyboard Setup Assistant" will launch. There's nothing to configure with the Esplora, so you can close this dialogue by clicking the red button in the top left of the window.



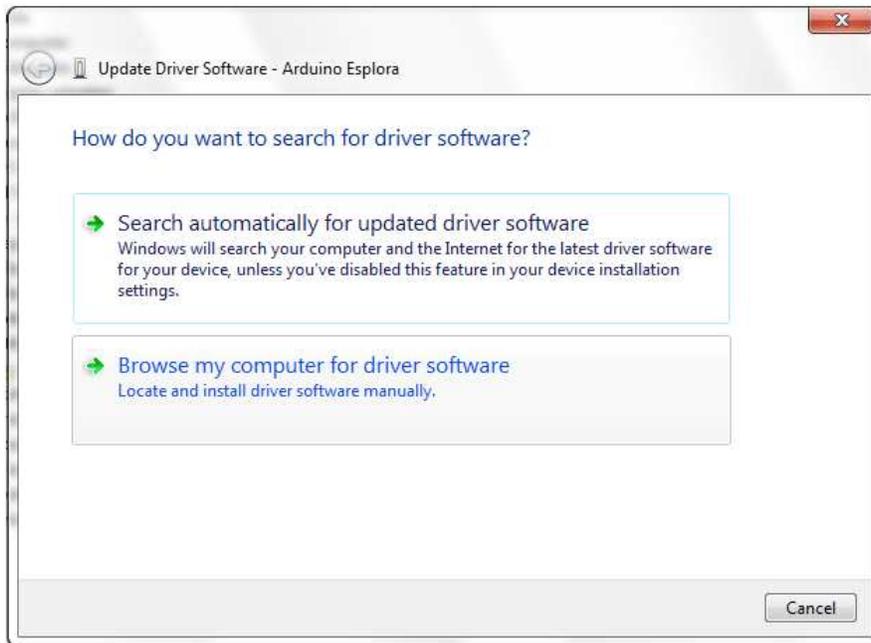
Instructions for Windows

The following instructions are for Windows 7. They are valid also for Windows XP, with small differences in the dialog windows.

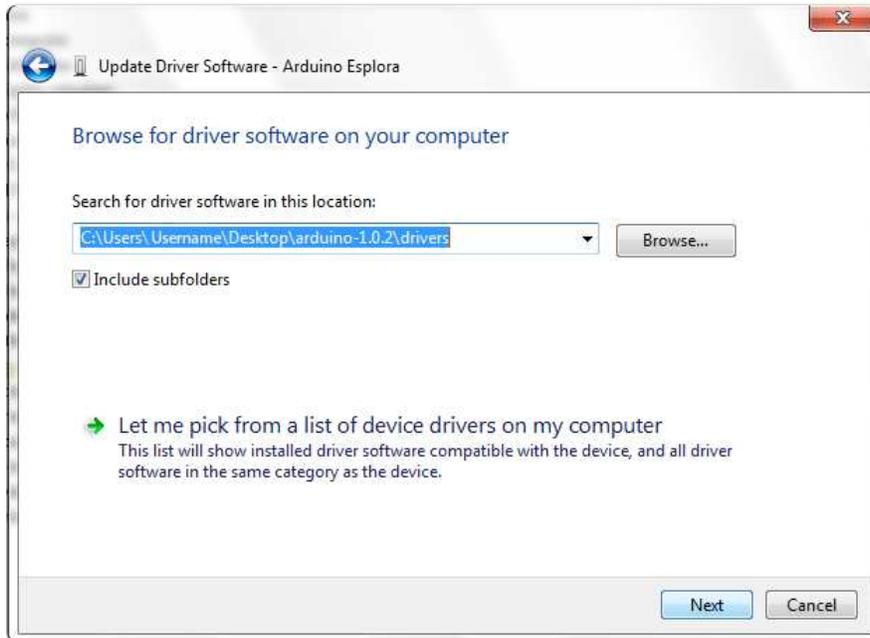
- Plug in your board and wait for Windows to begin its driver installation process. If the installer does not launch automatically, navigate to the Windows Device Manager (Start>Control Panel>Hardware) and find the Arduino Esplora listing. Right click and choose **Update driver**.



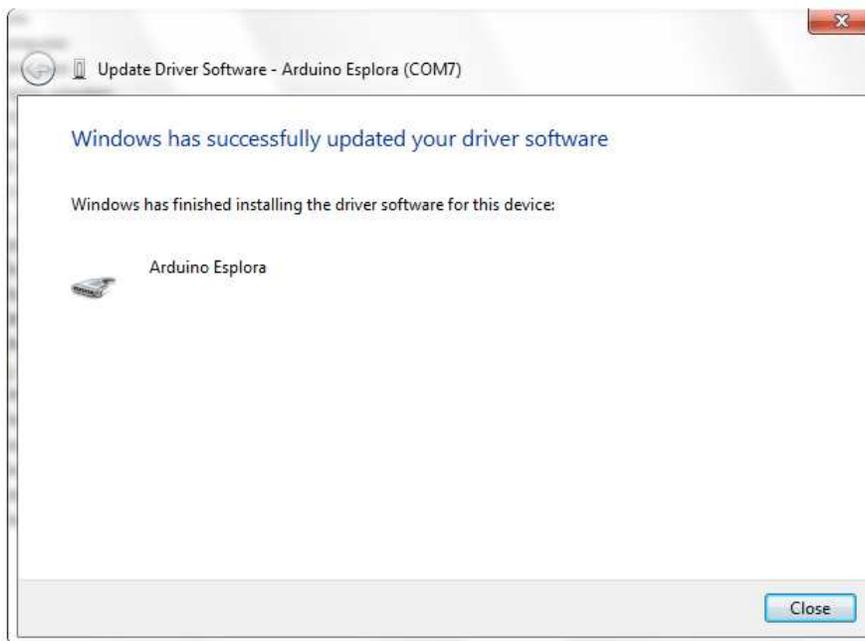
— At the next screen, choose "Browse my computer for driver software", and click **Next**.



— Click the **Browse . . .** button. Another dialog appears: navigate to the folder with the Arduino software that you just downloaded. Select the **drivers** folder and click **OK**, then click **Next**.



— You will receive a notification that the board has not passed Windows Logo testing. Click on the button **Continue Anyway**.



— After a few moments, a window will tell you the wizard has finished installing software for Arduino Esplora. Press the **Close** button.

Instructions for Linux

There is no need to install drivers for Ubuntu 12.04.

5 | Programming the board

Double-click the Arduino application. (Note: if the Arduino software loads in the wrong language, you can change it in the preferences dialog. See the environment page (<http://arduino.cc/en/Guide/Environment>) for details.)

Open the Esplora Blink example

Open the LED Blink example sketch: File > Examples > Esplora > Beginners > EsploraBlink.

Note: If you do not see the EsploraBlink example, you may need to update the Esplora library. To do this, quit the Arduino IDE and download this .zip file (<http://arduino.cc/en/uploads/Guide/Esplora.zip>). When you unzip it, you'll have a folder called Esplora. Look in your Documents folder (on OSX or Ubuntu) or My Documents folder (on Windows) for the Arduino sketches folder. It will be called "Arduino" and there should be a folder inside called "libraries". If not, create a libraries folder inside the sketches folder. Move the Esplora folder into the libraries folder. Then re-start the Arduino IDE. Now you should be able to open the EsploraBlink sketch as described above.

Configure Arduino for Esplora

Since the Arduino IDE is used for many different Arduino boards, you need to tell it that you're working with the Esplora. Open the **Tools > Board** menu and choose **Arduino Esplora**.

Select the right USB port

The Arduino IDE needs to know which of your USB ports the Esplora is connected to. The **Tools > Serial** menu lists the available ports:

- if only one item is shown, click on that one;
- if two or more are shown, you can disconnect the Esplora and re-open the menu; the entry that disappears should be the Esplora board. Reconnect the board and select that serial port.

Upload the program

Now, simply click the "Upload" button in the environment. Wait a few seconds - you should see the RX and TX leds on the board flashing. If the upload is successful, the message "Done uploading." will appear in the status bar of the software.

A few seconds after the upload finishes, you should see the yellow RGB LED on the board start to blink, changing color from red to green to blue to yellow to cyan to magenta to white. If it does, congratulations! You've gotten Esplora up and running.

6 | Change the RGB LED

Now that you've verified that the Esplora can connect to your computer and you can upload code, it's time to try a few of its functions. The Esplora has input sensors, which you get information from using **read** commands, and output actuators, which you control using **write** commands. The RGB LED and the buzzer are the two main output actuators. The joystick, linear potentiometer slider, microphone, accelerometer, temperature sensor, and pushbuttons are the input sensors. You can also add your own inputs and outputs using the TinkerKit input and output connectors.

In this example, you'll get the RGB LED to blink each of its colors once, then all three at the same time. The Esplora library, included at the beginning of the sketch, has a number of methods that will simplify the code you need to use.

The sketch below is similar to the EsploraBlink example, but controls the LED in a slightly different way. You can either use the `Esplora.writeRGB()` command, and give it red, green, and blue values like so:

```
Esplora.writeRGB(255, 255, 255); // sets red, green and blue all to full brightness
```

[Get Code] (<http://arduino.cc/en/Guide/ArduinoEsplora?action=sourceblock&num=1>)

Or you can turn on each individual color using the `Esplora.writeRed()`, `Esplora.writeGreen()`, and `Esplora.writeBlue()` commands, as you see below.

Copy the code below, and paste it into the Arduino IDE. Make sure you have the right serial port and the Esplora board selected, as you did previously. Plug your board in and upload the code. Once it is uploaded, you should see the RGB LED flashing.

```
// include the Esplora library
#include <Esplora.h>

void setup() {
  // nothing to setup
}

void loop() {
```

```

// write light level to the red LED
// 0 means the LED is off, 255 is full brightness
Esplora.writeRed(255);

// add a delay to keep the LED lit for
// 1000 milliseconds (1 second)
delay(1000);

// turn the red LED off, and the green LED on
Esplora.writeRed(0);
Esplora.writeGreen(255);

// add a delay
delay(1000);

// turn the green LED off, and the blue LED on
Esplora.writeGreen(0);
Esplora.writeBlue(255);

// add a delay
delay(1000);

// turn all the LEDs on together
Esplora.writeRGB(255, 255, 255);

// add a delay
delay(1000);

// turn the LEDs off
Esplora.writeRGB(0, 0, 0);

// add a delay
delay(1000);
}

```

[Get Code] (<http://arduino.cc/en/Guide/ArduinoEsplora?action=sourceblock&num=2>)

7 | Inputs changing outputs

The Esplora has a number of sensors right on its surface. In this example, you'll use the linear potentiometer (the slider at the bottom of the board) to change the brightness of the RGB LED.

Inputs on the Esplora give you values between 0 and 1023, except the pushbuttons, which give you 0 or 1. Outputs don't have the same range, however. The LED output ranges from 0 to 255. To get the value from the input to scale to the value for the output, you'll divide the input by 4. This new number will be the brightness level of the LED.

Copy the code below, and upload it as you did the previous examples. Once uploaded, move the slider back and forth. You should see the Red LED change its brightness.

```

// include the Esplora library
#include <Esplora.h>

void setup() {
// nothing to setup
}

void loop() {
// read the sensor into a variable
int slider = Esplora.readSlider()/4;

// convert the sensor readings to light levels
byte bright = slider/4;

// write the light levels to the Red LED
Esplora.writeRed(bright);

// add a small delay to keep the LED from flickering:
delay(10);
}

```

[Get Code] (<http://arduino.cc/en/Guide/ArduinoEsplora?action=sourceblock&num=3>)

8 | Next steps

There are a number of example sketches in the **File -> Examples -> Esplora** that show you more of what you can do with your Esplora. They're divided into examples for programming Beginners or Experts. If you're just getting started in programming, or you want to see how to use one input or output at a time, start with the Beginners examples. If you're an experienced programmer, then the Advanced examples will give you some more ideas.

If you've never programmed before at all, there are more examples included with the IDE to show you some of the basic programming structures. Note that these examples will need to be modified a bit to work with the Esplora, however. You'll need to include the Esplora library by choosing Import Library... -> Esplora from the Tools menu, and you'll need to change the general Arduino inputs and outputs for the Esplora inputs and outputs. For more on this, see the Guide to using Esplora with the Arduino Examples (<http://arduino.cc/en/Guide/ArduinoEsploraExamples>)

You should look at the Esplora library (<http://arduino.cc/en/Reference/EsploraLibrary>) reference pages for using various sensors and actuators with the Esplora library. You can see more examples on the examples (<http://arduino.cc/en/Tutorial/HomePage>) page.

If you have problems, please see the troubleshooting suggestions (<http://arduino.cc/en/Guide/Troubleshooting>).

The text of the Arduino getting started guide is licensed under aCreative Commons Attribution-ShareAlike 3.0 License (<http://creativecommons.org/licenses/by-sa/3.0/>). Code samples in the guide are released into the public domain.

