

# Introduction

The Arduino microcontroller (Figure 1-1) is like a little command center that is awaiting your orders. With a few lines of code, you can make your Arduino turn a light on or off, read a sensor value and display it on your computer screen, or even use it to build a homemade circuit to repair a broken kitchen appliance. Because of the versatility of the Arduino and the massive support available from the online community of Arduino users, it has attracted a new breed of electronics hobbyists who have never before touched a microcontroller, let alone programmed one.

*Figure 1-1. An Arduino microcontroller*

The basic idea of the Arduino is to create an atmosphere where *anyone* who is interested can participate and contribute with little upfront cost. A basic Arduino board can be found online for around 800 INR, and all of the software needed to program the Arduino is open-source (free to use and modify). You need only a computer and a standard USB cable

## Setting Up the Arduino

### Setting Up

An Arduino is an embedded computing platform, consisting of a single-board microcontroller and a software suite for programming it. It can interact with the environment through the use of sensors, motors, lights etc. as guided by the program burnt on it. A series of boards are sold under the Arduino moniker, of which the Arduino Uno (shown above) is the latest. The Arduino \*\*can be

connected to LEDs, displays, buttons, switches, motors, temperature sensors, pressure sensors, distance sensors, GPS receivers, Ethernet modules, \*\*or just about anything that outputs data or can be controlled

The Arduino can also be extended with the use of shields, which are circuit boards containing other devices (e.g. GPS receivers, LCD Displays, Ethernet modules, etc.) that you can simply connect to the top of your Arduino to get extra functionality. Shields\*\* also extend the pins to the top of its own circuit board\*\* so you still have access to all of them. You don't have to use a shield if you don't want to; you can make the exact same circuitry using a breadboard, or by making your own PCB.

Arduino programs are called sketches, and are in a language very similar to C. To program the Arduino, you use the Arduino IDE. Download it now from <http://arduino.cc/en/Main/Software> (<http://arduino.cc/en/Main/Software>). It is available for Windows, Mac OS and Linux.

1. Now plug your Arduino into a USB port. To install drivers properly on a Windows system:
2. Plug in your board and wait for Windows to begin its driver installation process. After a few moments, the process will fail, despite its best efforts
3. Click on the Start Menu, and open up the Control Panel.
4. While in the Control Panel, navigate to System and Security. Next, click on System. Once the System window is up, open the Device Manager.
5. Look under Ports (COM & LPT). You should see an open port named "Arduino UNO (COMxx)"
6. Right click on the "Arduino UNO (COMxx)" port and choose the "Update Driver Software" option.
7. Next, choose the "Browse my computer for Driver software" option.
8. Finally, navigate to and select the Uno's driver file, named "ArduinoUNO.inf", located in the "Drivers" folder of the Arduino Software download (not the "FTDI USB Drivers" sub-directory, uncheck subdirectory option).
9. Windows will finish the driver installation from there. Now Fire up the IDE.
10. If you're using Windows 8, the drivers require disabling the signature requirements of drivers. Follow the steps here (<http://www.bryonconnolly.com/windows-8-arduino-driver-install>) to do so in Windows 8, and then follow the steps above.

The IDE consists of a menu bar, a quick-access toolbar, and messages at the bottom. Set your board from the Tools > Board menu (e.g. we are using an Arduino UNO) and your serial port from Tools > Serial Port. (as shown earlier in the Device Manager) Go to File > Examples > Basics > Blink to try out your first Arduino sketch. Don't worry about the code right now, but first let's see if everything is working by uploading it on the board.

Click on the Verify (tick mark) button to compile the code and check that your code is correct and error free, then click Upload (right arrow) button to upload it to the Arduino board. When the process is completed, you should see the orange LED on the board blinking. If it is blinking, congratulations.