

3.5 Introduction to Arduino:

Arduino is a project, open-source hardware, and software platform used to design and build electronic devices. It designs and manufactures microcontroller kits and single-board interfaces for building electronics projects.

The Arduino boards were initially created to help the students with the non-technical background.

The designs of [Arduino boards](#) use a variety of controllers and microprocessors.

The Arduino board consists of sets of analog and digital I/O (Input / Output) pins, which are further interfaced to **breadboard, expansion boards**, and other **circuits**. Such boards feature the model, Universal Serial Bus (**USB**), and **serial communication interfaces**, which are used for loading programs from the computers.

It also provides an **IDE** (Integrated Development Environment) project, which is based on the Processing Language to upload the code to the physical board.

The projects are authorized under the **GPL** and **LGPL**. The **GPL** is named as **GNU General Public License**. The licensed **LGPL** is named as **GNU Lesser General Public License**. It allows the use of Arduino boards, it's software distribution, and can be manufactured by anyone.

It is also available in the form of self practicing kits.

The [Arduino](#) is used for various purposes, such as:

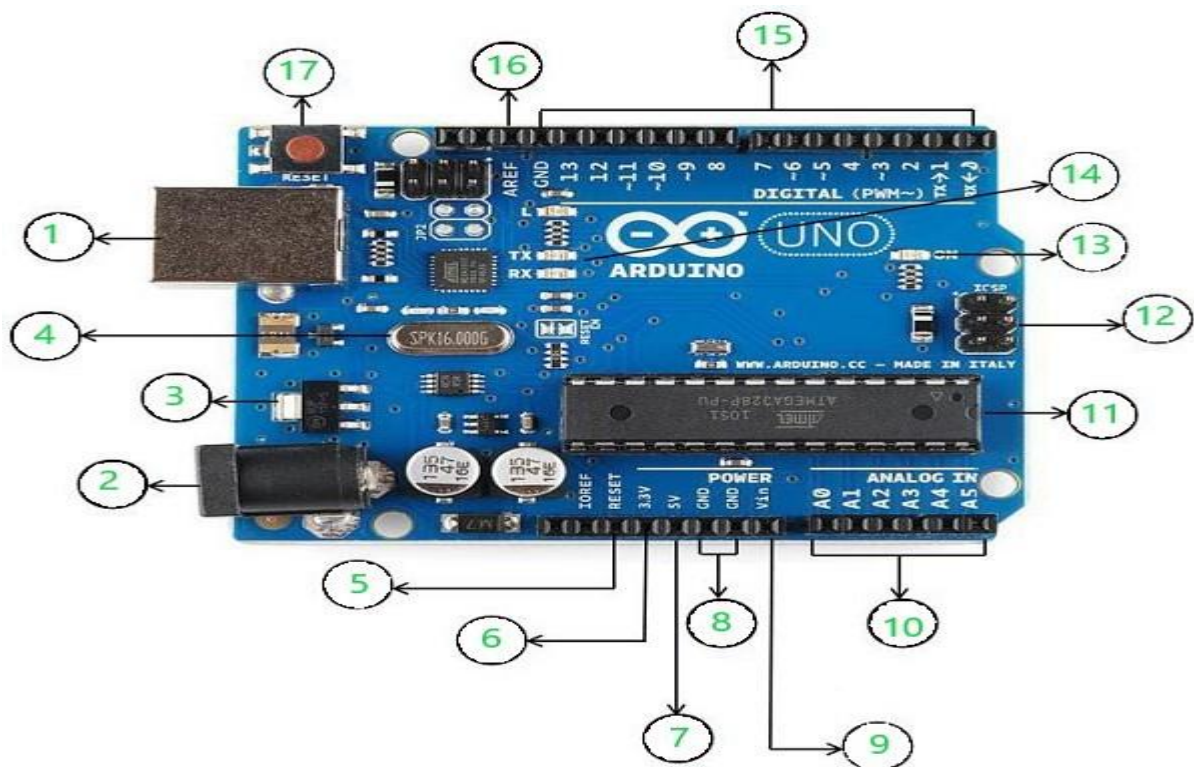
- Finger button
- Button for motor activation
- Light as a sensors
- LED button
- Designing
- The Building of electronic devices

Features

The features of Arduino are listed below:

Arduino programming is a simplified version of C++, which makes the learning process easy.

- The Arduino IDE is used to control the functions of boards. It further sends the set of specifications to the microcontroller.
- Arduino does not need an extra board or piece to load new code.
- Arduino can read analog and digital input signals.
- The hardware and software platform is easy to use and implement.
- It is necessary to understand the layout and circuitry of your Arduino Uno.



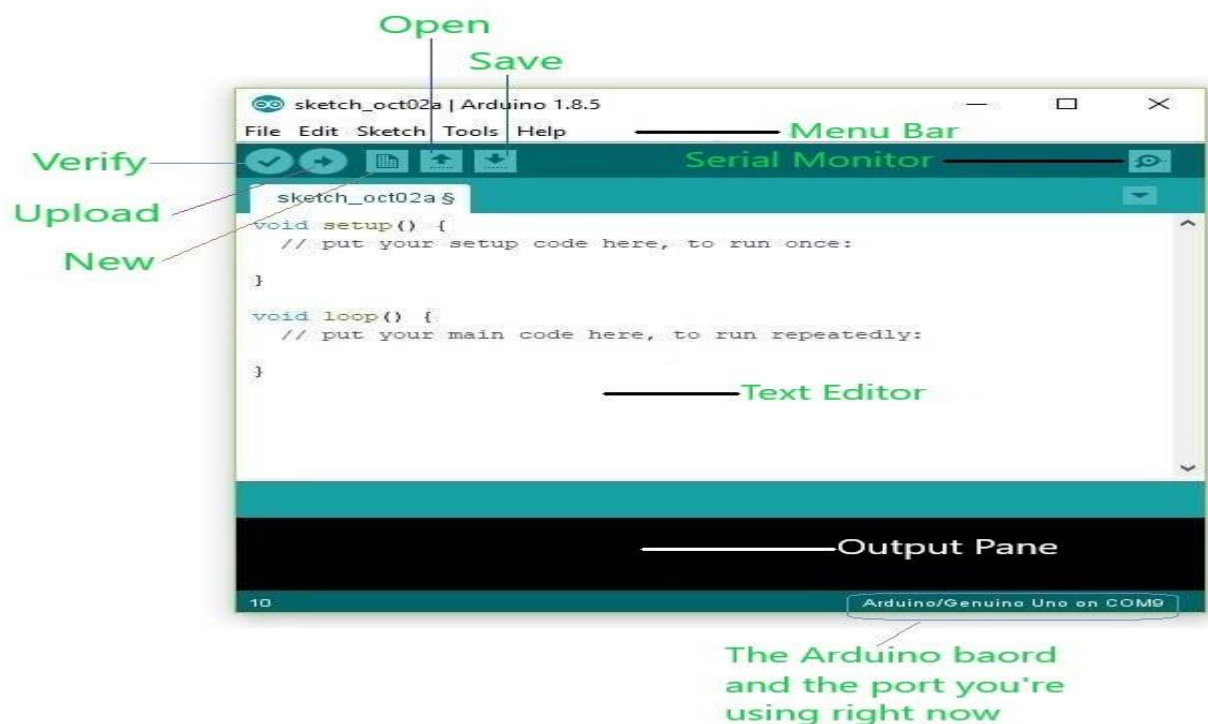
Using the above image as a reference, the labeled components of the board respectively are-

1. **USB:** can be used for both power and communication with the IDE
2. **Barrel Jack:** used for power supply
3. **Voltage Regulator:** regulates and stabilizes the input and output voltages
4. **Crystal Oscillator:** keeps track of time and regulates processor frequency
5. **Reset Pin:** can be used to reset the Arduino Uno
6. **3.3V pin:** can be used as a 3.3V output
7. **5V pin:** can be used as a 5V output
8. **GND pin:** can be used to ground the circuit
9. **Vin pin:** can be used to supply power to the board
10. **Analog pins(A0-A5):** can be used to read analog signals to the board

11. **Microcontroller(ATMega328):** the processing and logical unit of the board
12. **ICSP pin:** a programming header on the board also called SPI
13. **Power indicator LED:** indicates the power status of the board
14. **RX and TX LEDs:** receive(RX) and transmit(TX) LEDs, blink when sending or receiving serial data respectively
15. **Digital I/O pins:** 14 pins capable of reading and outputting digital signals; 6 of these pins are also capable of PWM
16. **AREF pins:** can be used to set an external reference voltage as the upper limit for the analog pins
17. **Reset button:** can be used to reset the board

Getting started with the Arduino IDE

Now that you're familiar with the hardware, its time to learn about the development environment using which you're going to program your Uno. The Arduino IDE is the best place to start your journey in programming your Uno. Download the latest build of the Arduino IDE for your Mac or PC. Go ahead and install the IDE on your PC or Mac and open it.



As you open the IDE, you'll be greeted by a window similar to the one shown in the above image. The text editor is where you'll be writing your code; you'll use the verify button to compile and debug the written program, the save button to save the program and the upload button to upload the program to the board. Before you click on the upload button, it is necessary to select your board, Uno in this case, from the tools menu in the Menu Bar. After you choose your appropriate board, make sure you specify the correct port on your PC or Mac that you've connected your Uno to, in the IDE.