

**Aim**

To study the Arduino UNO kit with its features and Pin configurations for performing IoT based projects

**Introduction to Arduino UNO:**

In IoT applications the Arduino is used to collect the data from the sensors/devices to send it to the internet and receives data for purpose of control of actuators.

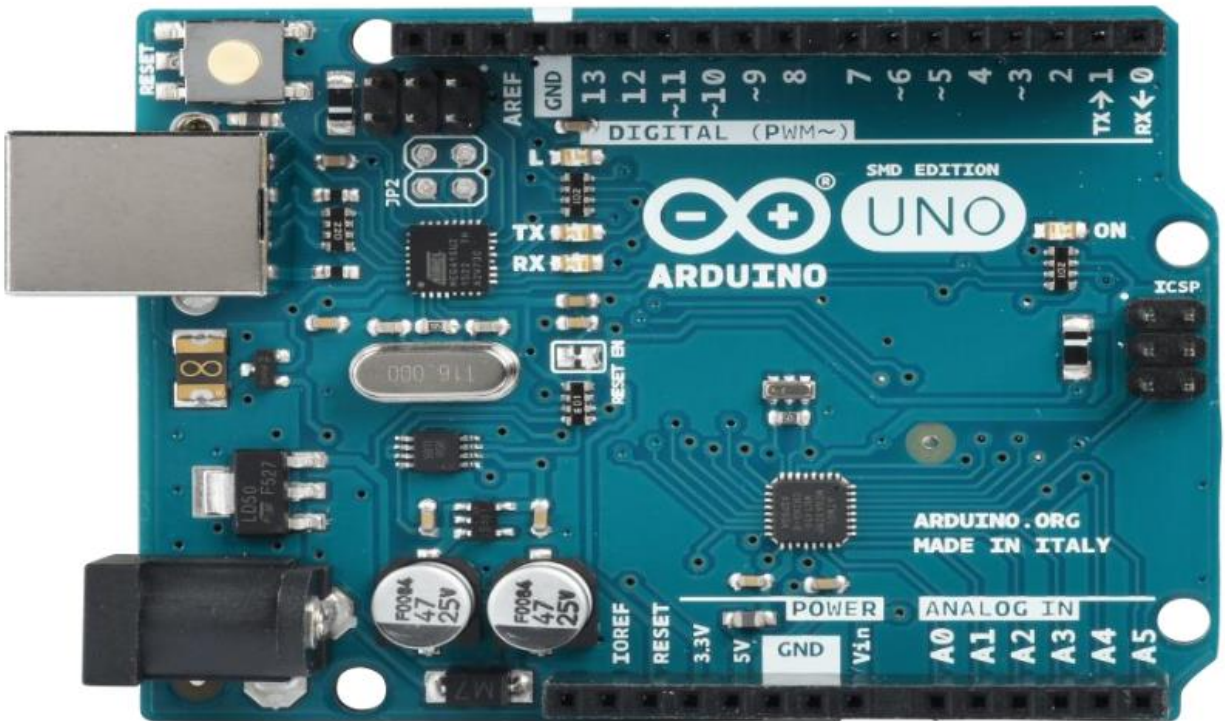


The Arduino Uno is an open-source microcontroller board based on the Microchip ATmega328P microcontroller and developed by Arduino.cc. The board is equipped with sets of digital and analog input/output (I/O) pins that may be interfaced to various expansion boards (shields) and other circuits. The board has 14 digital I/O pins (six capable of PWM output), 6

analog I/O pins, and is programmable with the Arduino IDE (Integrated Development Environment), via a type B USB cable. It can be powered by the USB cable or by an external 9-volt battery, though it accepts voltages between 7 and 20 volts. The word "uno" means "one" in Italian and was chosen to mark the initial release of Arduino Software.

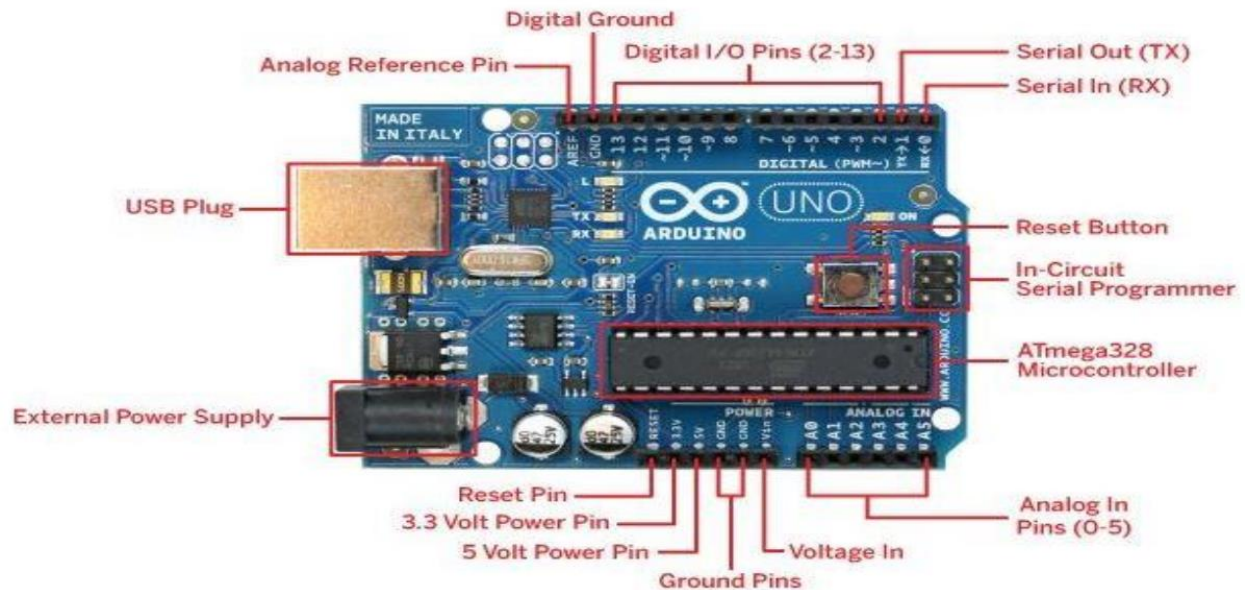
### **Features of the Arduino UNO:**

1. Arduino boards are able to read analog or digital input signals from different sensors and turn it into an output such as activating a motor, turning LED on/off, connect to the cloud and many other actions.
2. The board functions can be controlled by sending a set of instructions to the microcontroller on the board via Arduino IDE.
3. Arduino IDE uses a simplified version of C++, making it easier to learn to program
4. Arduino provides a standard form factor that breaks the functions of the micro- controller into a more accessible package.



### **Arduino UNO Pin Configuration:**

Arduino boards sense the environment by receiving inputs from many sensors, and affect their surroundings by controlling lights, motors, and other actuators. Arduino boards are the microcontroller development platform that will be at the heart of your projects. When making something you will be building the circuits and interfaces for interaction, and telling the microcontroller how to interface with other components. Here the anatomy of Arduino UNO.



1. **Digital pins** Use these pins with `digitalRead()`, `digitalWrite()`, and `analogWrite()`. `analogWrite()` works only on the pins with the PWM symbol.
2. **Pin 13 LED** The only actuator built-in to your board. Besides being a handy target for your first blink sketch, this LED is very useful for debugging.
3. **Power LED** Indicates that your Arduino is receiving power. Useful for debugging.
4. **ATmega** microcontroller The heart of your board.
5. **Analog in** Use these pins with `analogRead()`.
6. **GND and 5V pins** Use these pins to provide +5V power and ground to your circuits.
7. **Power connector** This is how you power your Arduino when it's not plugged into a USB port for power. Can accept voltages between 7-12V.
8. **TX and RX LEDs** These LEDs indicate communication between your Arduino and your computer. Expect them to flicker rapidly during sketch upload as well as during serial communication. Useful for debugging.
9. **USB port** Used for powering your Arduino UNO, uploading your sketches to your Arduino, and for communicating with your Arduino sketch (via `Serial.println()` etc.).
10. **Reset button** Resets the ATmega microcontroller.

### Result:

Thus the Arduino UNO kit is studied with its features and its various Pin Configurations which is helpful for creating IoT based projects using Arduino UNO.