LAB MANUAL 2

Getting Started with Arduino Windows/Ubuntu (Arduino IDE, Plug In-Play)

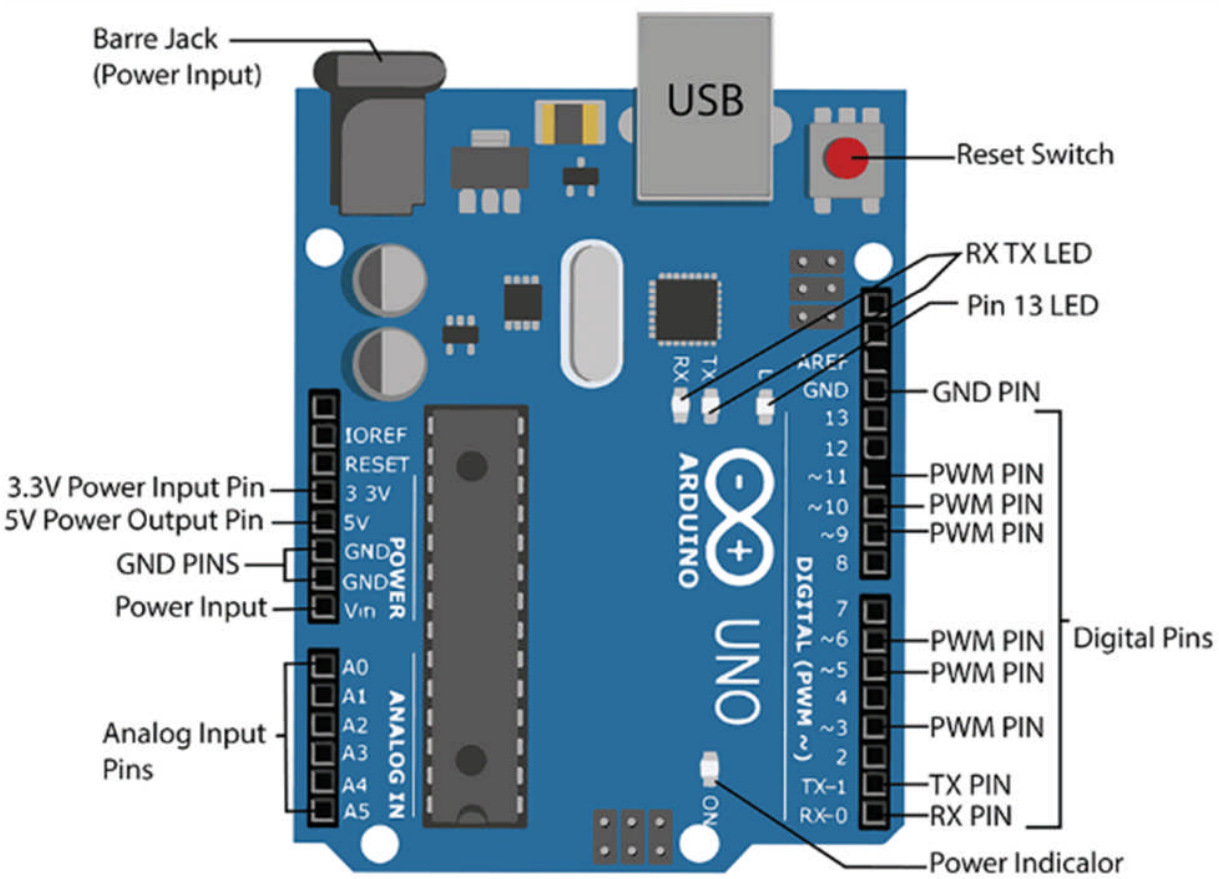
About Arduino board-

A microcontroller is embedded inside of a system to control a singular function in a device. It does this by interpreting data it receives from its I/O peripherals using its central processor. The temporary information that the microcontroller receives is stored in its data memory, where the processor accesses it and uses instructions stored in its program memory to decipher and apply the incoming data. It then uses its I/O peripherals to communicate and enact the appropriate action.

Arduino boards are available in different sizes, form factors, different no. of I/O pins etc. Some of the commonly known and frequently used Arduino boards are Arduino UNO, Arduino Mega, Arduino Nano, Arduino Mini, Arduino Leonardo and Arduino Lilypad.

Arduino UNO is a basic and inexpensive Arduino board and is the most popular of all the Arduino boards with a market share of over 50%. Arduino UNO is considered the best prototyping board for beginners in electronics and coding.

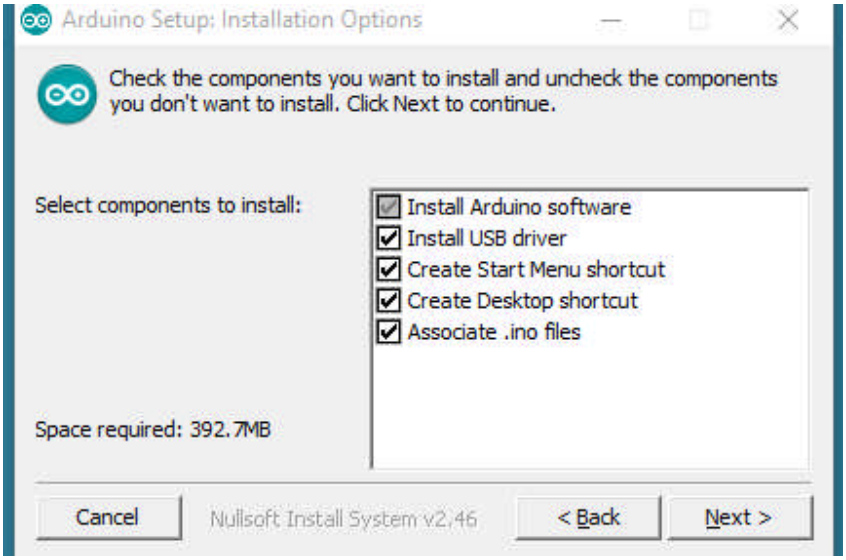
- UNO is based on ATmega328P microcontroller, which is an 8-bit microcontroller based on the AVR architecture.
- It has 14 digital input/output pins (of which 6 can be used as PWM outputs), 6 analog inputs, a USB connection, a power jack, a reset button and more. It contains everything needed to support the microcontroller; simply connect it to a computer with a USB cable or power it with an AC-to-DC adapter or battery to get started.

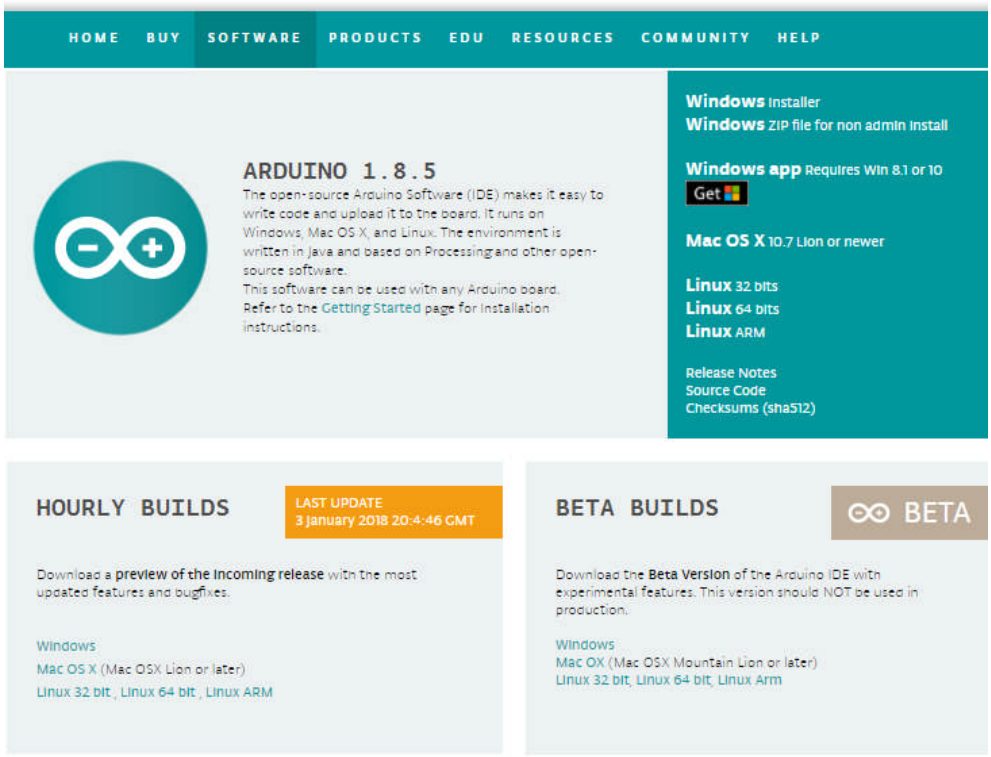


Step-1

Download Arduino Software – Integrated Development Environment

Access the Internet – In order to use your Arduino, you will first need to download Arduino software from the Internet (it is free!). This software, known as the Arduino integrated development environment (IDE), allows us to program Arduino to do what we want. It is called integrated, because we can write the code, compile and execute it from the same program window. You can download the latest or a previous version of the software from Arduino.cc, the repository for Arduino development. We can also code online using Arduino Web Editor, with the code stored on the cloud: http://arduino.cc/en/Main/Software



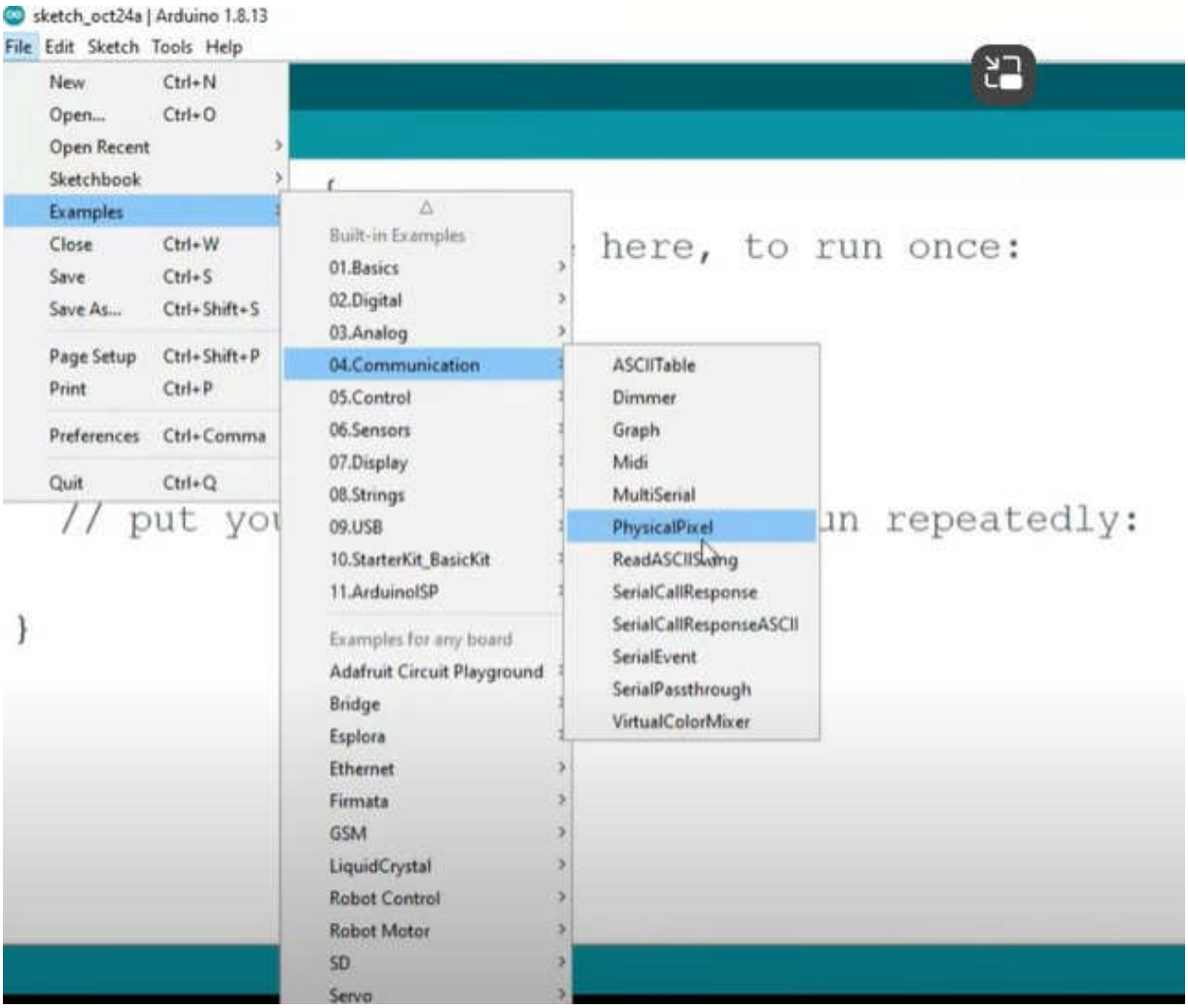


The Arduino Integrated Development Environment - or Arduino Software (IDE) - contains a text editor for writing code, a message area, a text console, a toolbar with buttons for common functions and a series of menus. It connects to the Arduino hardware to upload programs and communicate with them.

Step-2

Procedure for practical setup-

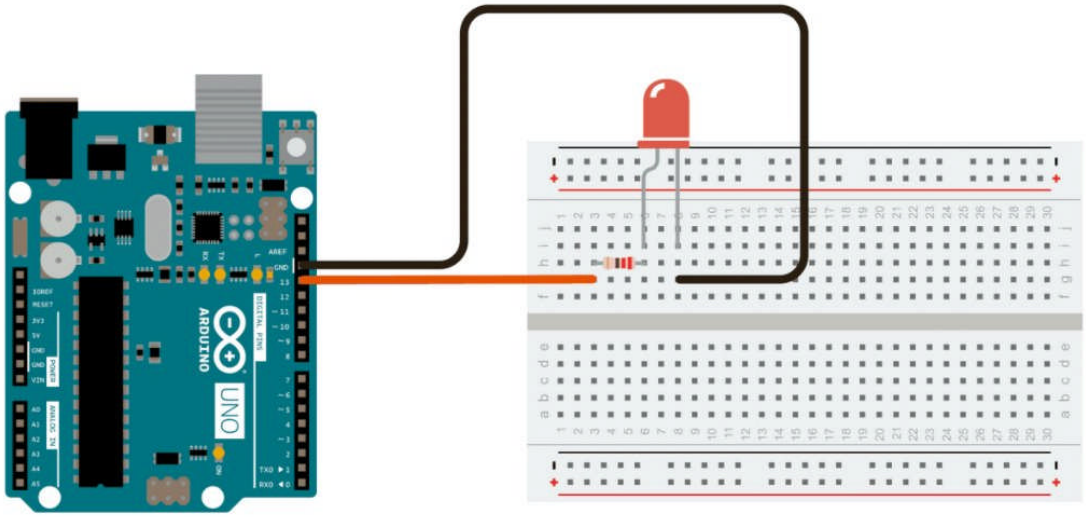
- Check out: http://arduino.cc/en/Guide/HomePage
- Download & install the Arduino environment (IDE)
- Connect the board to your computer via the UBS cable
- If needed, install the drivers (not needed in lab)
- Launch the Arduino IDE
- Select your board
- Select your serial port
- Open the examples select communication and then physicalPixcel and run code
- Open our window terminal and follow the steps in the next slide.



Procedure for blinking led :

LED Resistor Circuit: The LED's legs are connected to two pins on the Arduino: ground and pin 13. The component between the LED and pin 13 is a resistor, which helps limit the current to prevent the LED from burning itself out.

Blink Arduino Code: When the code editor is open, you can click the drop-down menu on the left and select "Blocks + Text" to reveal the Arduino code generated by the code blocks. All the extra symbols are part of Arduino's syntax, but don't be intimidated.



Led blink with Arduino

Step-3

Run below code for blinking Led:-
Install the pyserial library if you haven't already:

```
pip install pyserial
```

Create a Python script

```
import serial
import time
ser = serial.Serial('COM3', 9600)
time.sleep(2)
try:
    while True:
        ser.write(b'1')
        time.sleep(1)
        ser.write(b'0')
        time.sleep(1)
except KeyboardInterrupt:
    ser.close()
```