

1.0 The Board

1.1 Original vs Compatible

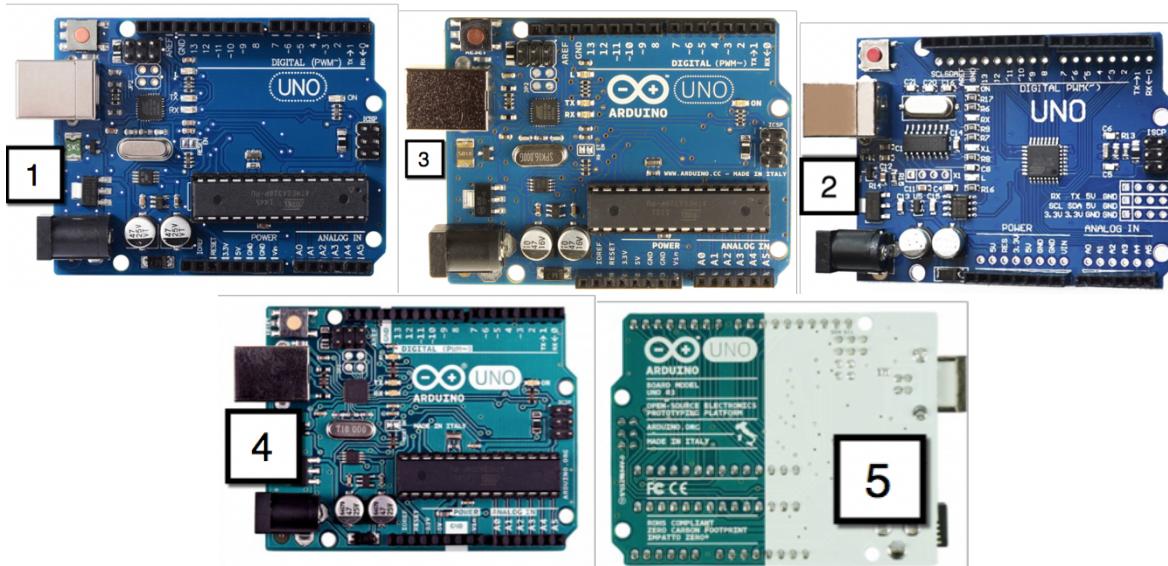


Figure 1

1.2 Arduino Family

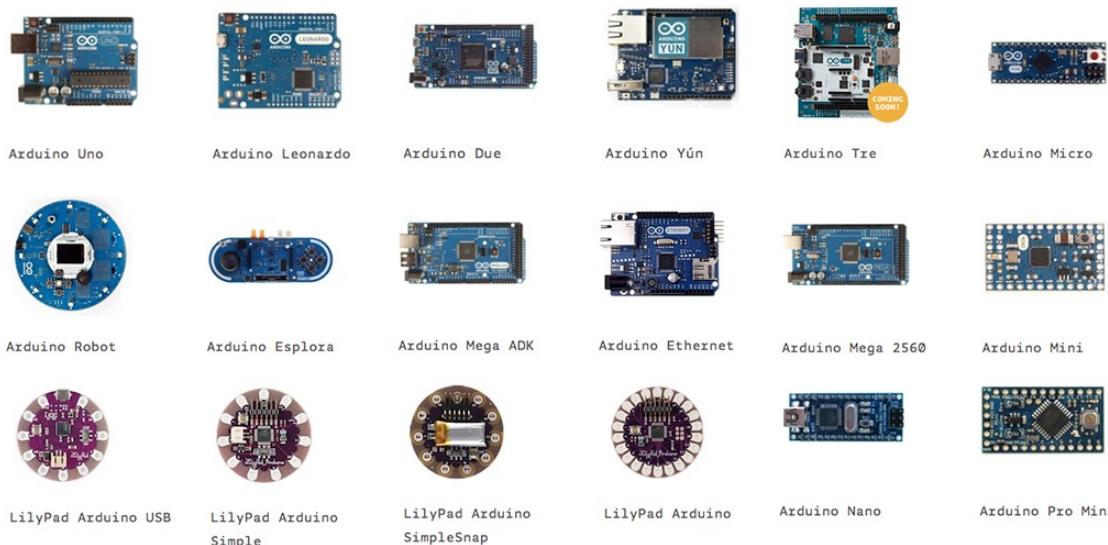


Figure 2

2.0 The Comm Port

2.1 Open Device Manager [#1 NO Arduino attach & #2 WITH Arduino attached]

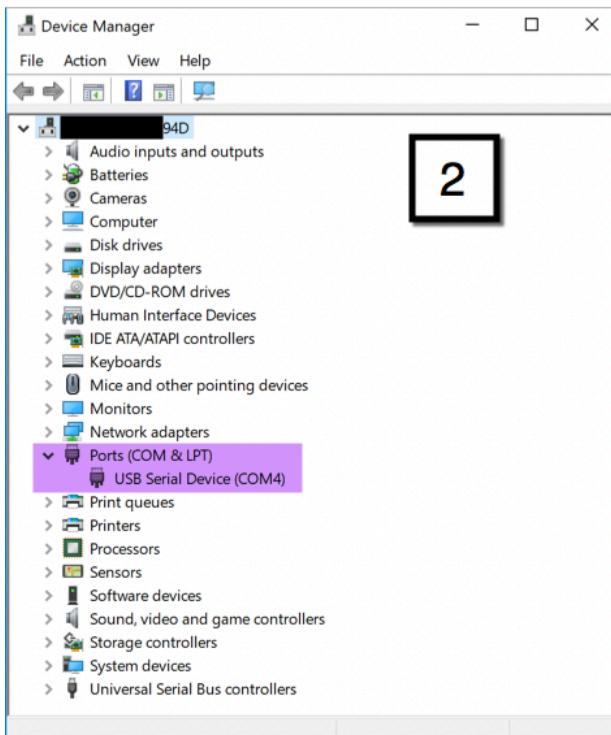
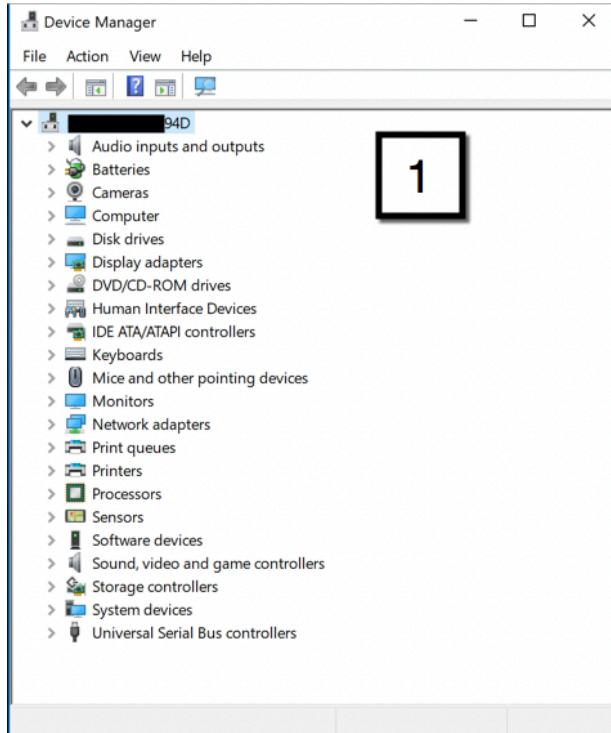


Figure 3

3.0 Installation of Arduino IDE

3.1 <https://wwwarduino.cc/> SOFTWARE > DOWNLOADS

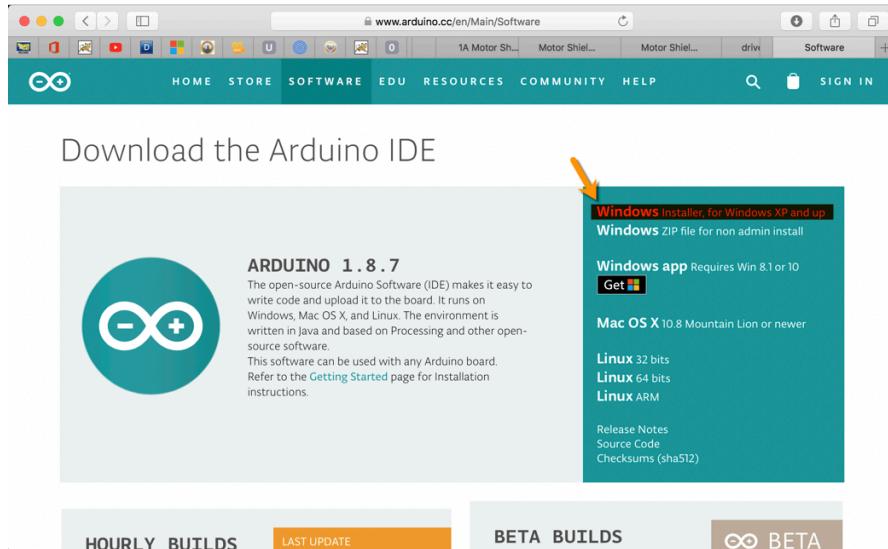


Figure 4

3.2 arduino-1.8.7-windows.exe

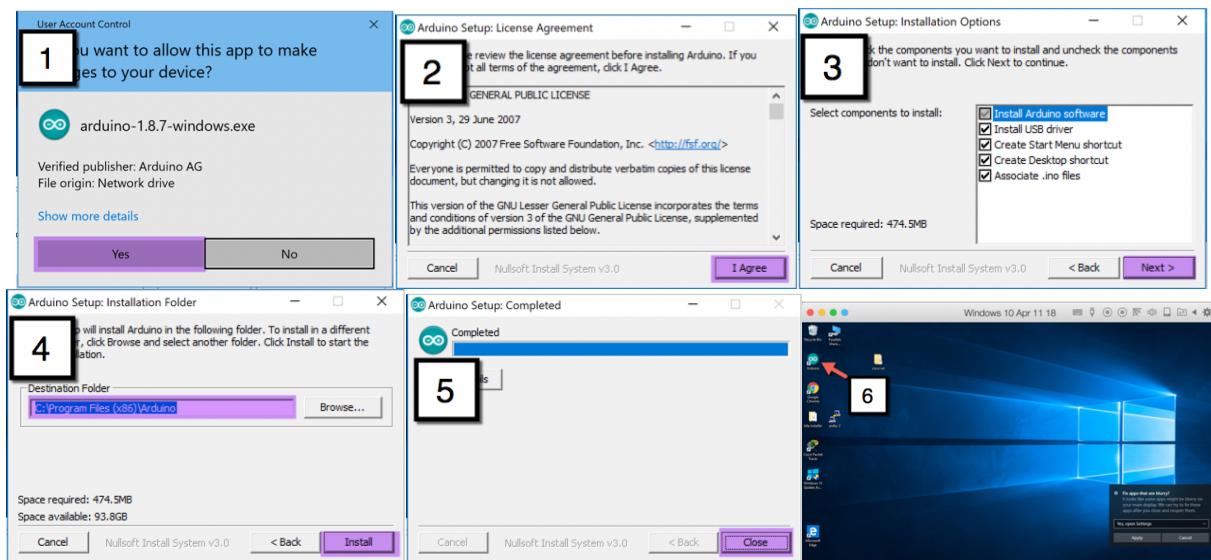
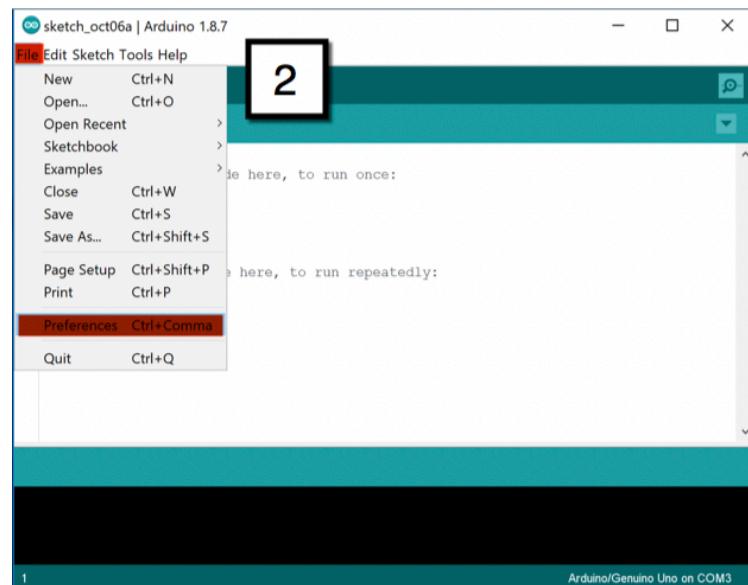
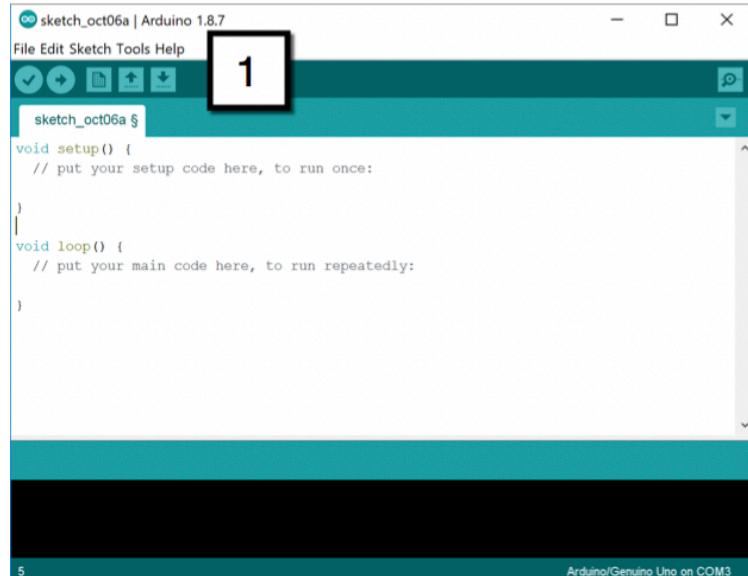


Figure 5

Change Setting File > Preferences



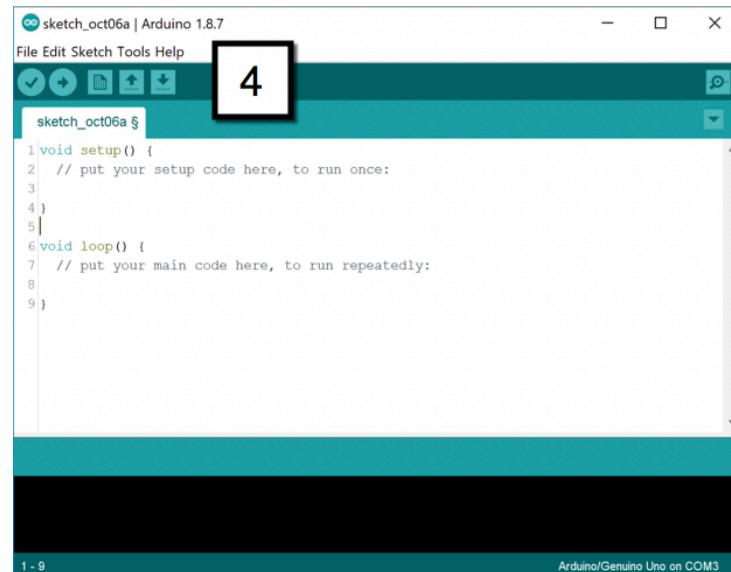
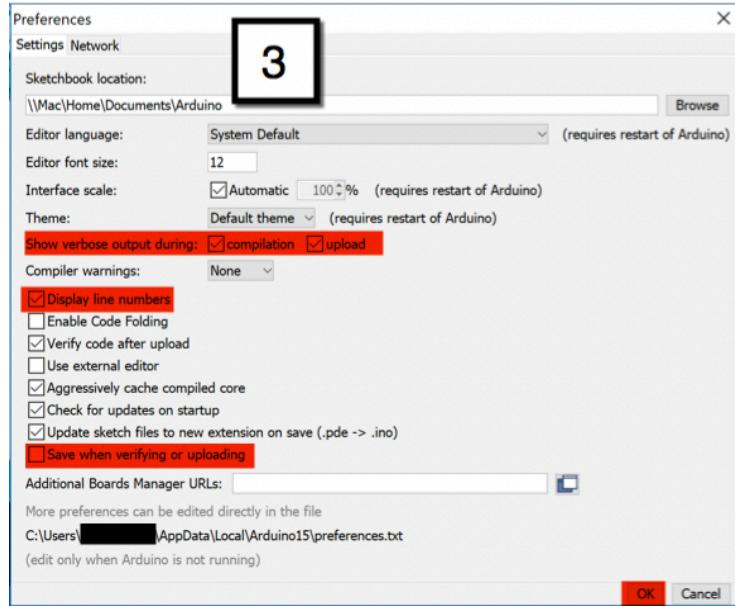


Figure 6

4.0 Installation of Additional Arduino Board Driver

4.1 CH430 – Refer to Board #2 at Fig 1

- i. Download the Windows CH340 Driver from

<https://roboindia.com/tutorials/CH340-usb-to-serial-driver-installation>

- ii. Unzip the file
- iii. Run the installer which you unzipped

- iv. In the Arduino IDE when the CH340 is connected you will see a COM Port in the *Tools > Serial Port* menu, the COM number for your device may vary depending on your system.

5.0 Getting Started in the Arduino IDE

5.1 Familiarization using the IDE

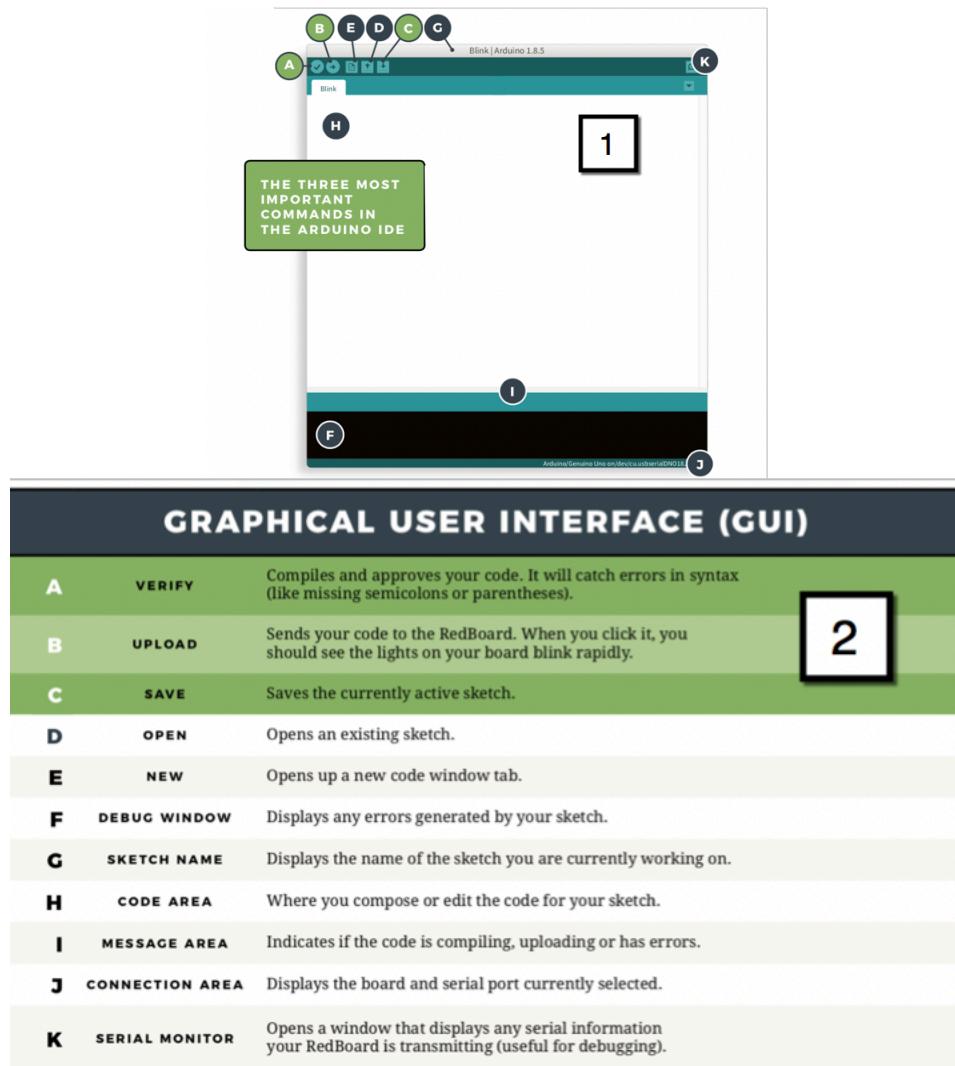


Figure 7

6.0 Connecting Your Board

6.1 Set your board type.

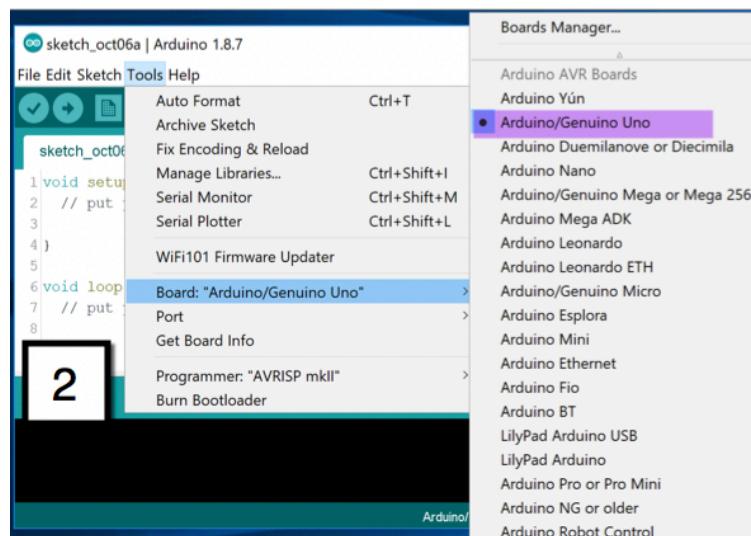
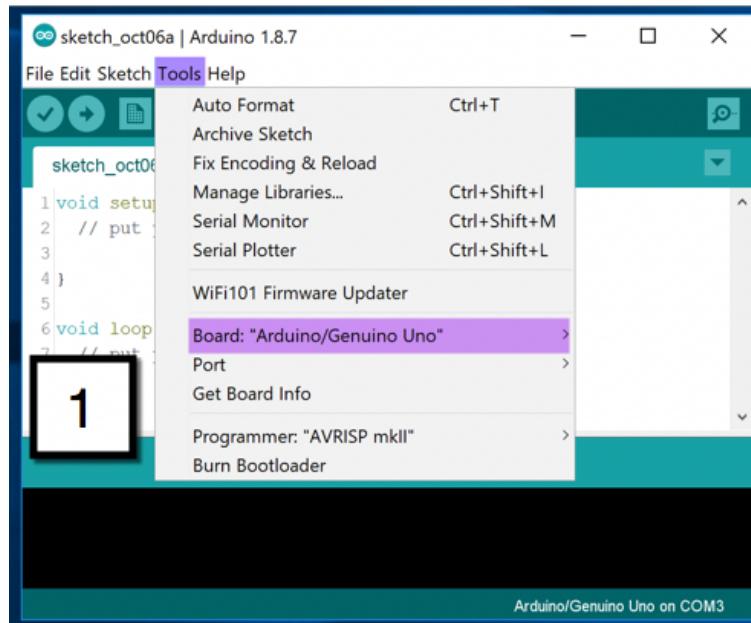


Figure 8

6.2 Set your Port number.

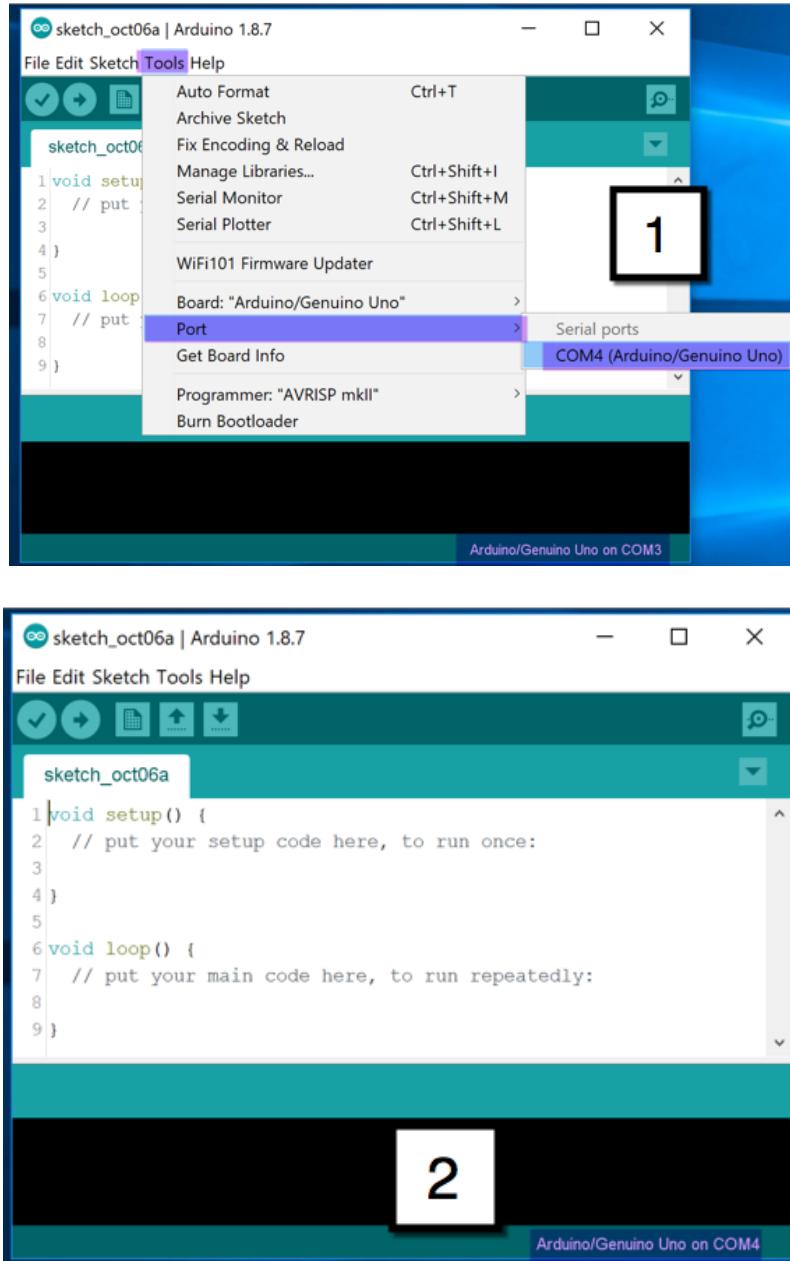


Figure 9

7.0 Check Your Board

- i. Upload Blink & AnalogReadSerial sketch.
- ii. FILE>EXAMPLES>BASICS>BLINK.
- iii. FILE>EXAMPLES>BASICS>ANALOGREADSERIAL.
- iv. The board considered **faulty** if any of the tests failed.

8.0 Complete Arduino-based System Overview

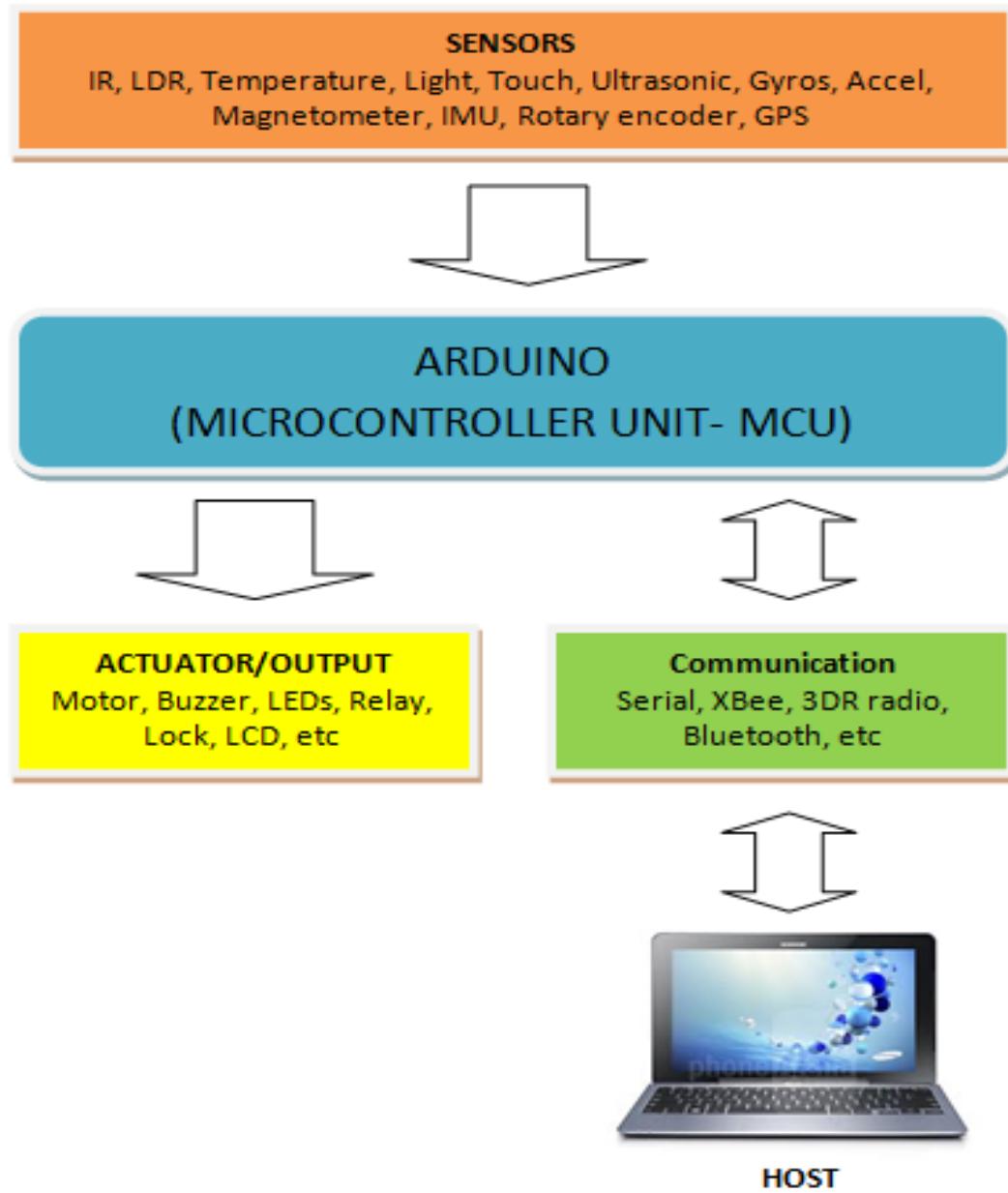


Figure 10

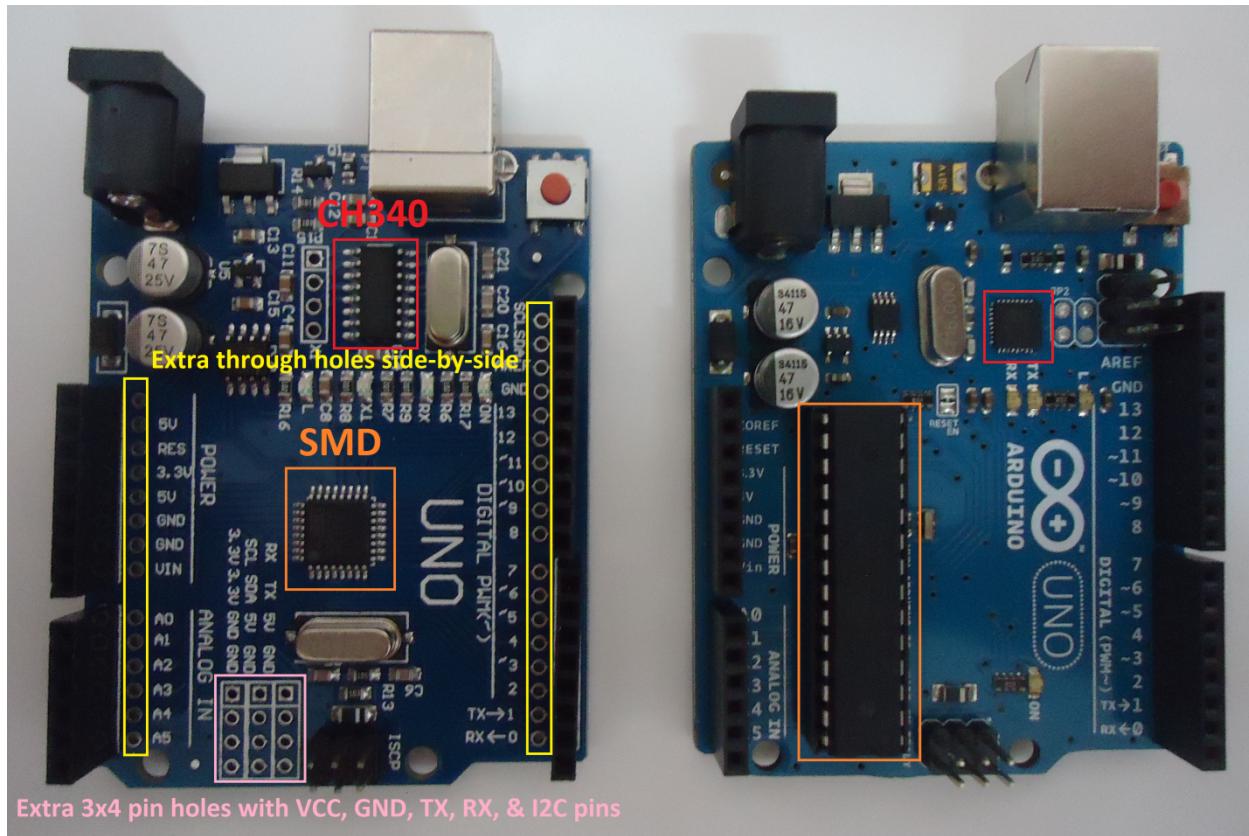


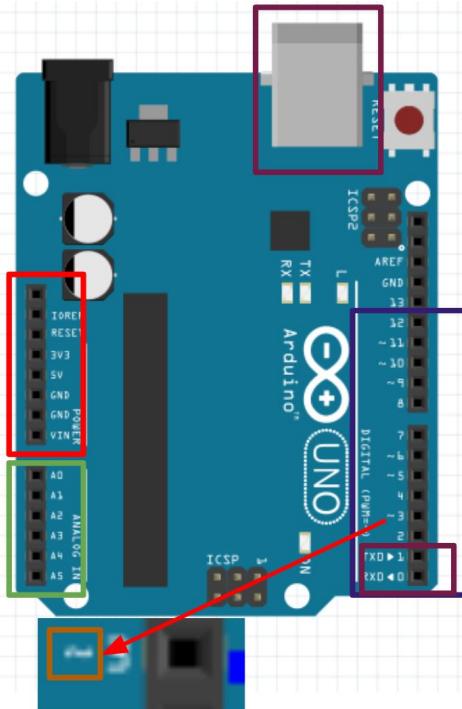
Figure 11

THE ARDUINO

MCU : Atmega 328
Input voltage : 7V-12V
Operating voltage : 5V
CPU Speed : 16MHZ
Analog In/Out : 6/0
Digital IO/PWM : 14/6
EEPROM : 1KB
SRAM : 2KB
Flash : 32KB
UART : 1
USB : Regular

Sensor's Power - 200 mA Max

Analog In -
 Access with `analogRead(pin#)`
 returns mapped 0-5V => int 0-1023
Digital In/Out -
 Set pin with `pinmode(pin#, INPUT/OUTPUT)`
 OUTPUT - 40 mA Max
 INPUT - return HIGH if > 3V else LOW
PWM (Pulse Width Modulated) signal
 Access with `analogWrite(pin#, Value)`
 $0 < \text{Value} < 255$ maps to 0 - 100% Duty cycle
Serial Communication Access Points



ARDUINO PIN	MICROCONTROLLER PIN
0	-
1	-
2	-
3	-
4	-
5	-
6	-
7	-
8	-
9	-
10	-
11	-
12	-
13	-
A0	PC0
A1	PC1
A2	PC2
A3	PC3
A4	PC4(SDA)
A5	PC5(SCL)

Pin out from: <http://icircuit.net/arduino-boards-pin-mapping/11>

Figure 12

END OF TUTORIAL