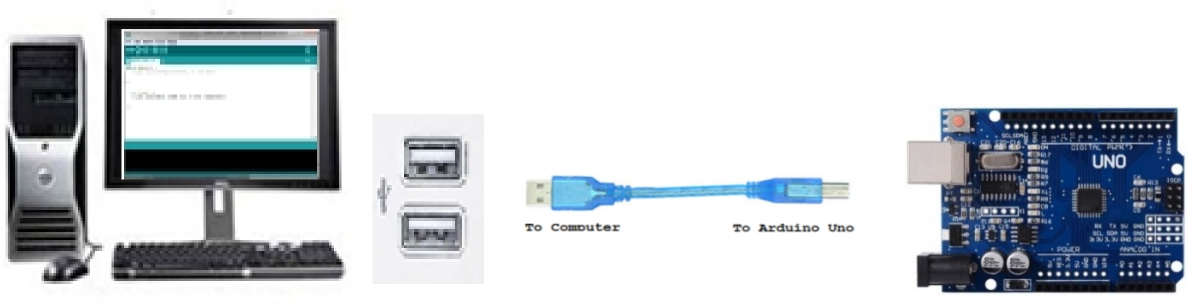
ARDUINO

Tutorial 1
Arduino Uno Quick Start

OCT 2021 by TeakSoon Ding
for STEMKRAF
https://github.com/teaksoon/stemkraf



- Basic Arduino Hardware/Software Setup
- Arduino Programming - First Program in 6 steps
- Arduino Uno Project Deployment
- Arduino IDE Software - Serial Monitor

Note:
If you do not have a Computer and do not wish to spend alot of money, the cheapest way is to get an old working computer (most likely you can get it for FREE). Give it a second life with the Linux Operating System and you can use it comfortably to program your Arduino.

Linux Operating System uses very little resources and it is FREE.

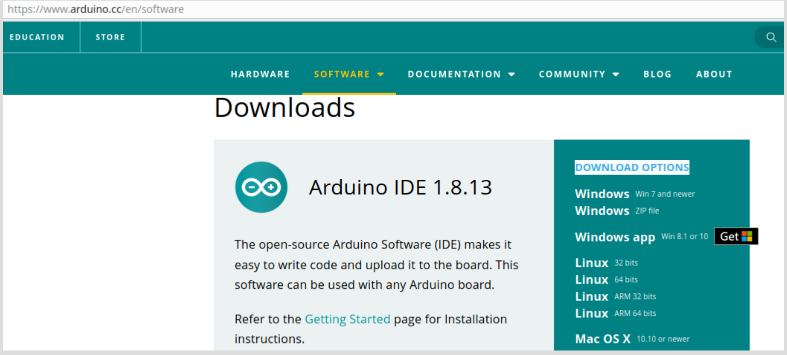
There is a small tutorial on how to do this,

https://github.com/teaksoon/stemkraf/blob/main/linux_arduino_installation.pdf

The Arduino Uno board has **Atmega328 micro-controller chip** on it and that is what we want to Program. There are no "Keyboard" or "Display Screen" on both Arduino Uno and the Atmega328 micro-controller for us to make our Program. We need to use a Computer with a Program Editor Software to key in our Program.

A Computer with USB support to run **Arduino IDE Software (Program Editor Software)**. The Arduino IDE Software requires very little Computer resources and it can run on many Operating System. We can even use an old computer with USB support from the "junk stores". For example, the Intel Core2 processor with 2Gb RAM installed with a free lightweight Linux Operating System.

The Arduino IDE Software
The Arduino IDE Software is Open Source Software, we can download and use it for FREE!!! We can contribute if we wish to. The Arduino IDE Software can be downloaded from this website,
https://www.arduino.cc/en/software



https://www.arduino.cc/en/software

EDUCATION STORE

HARDWARE SOFTWARE DOCUMENTATION COMMUNITY BLOG ABOUT

Downloads

Arduino IDE 1.8.13

The open-source Arduino Software (IDE) makes it easy to write code and upload it to the board. This software can be used with any Arduino board.

Refer to the Getting Started page for Installation instructions.

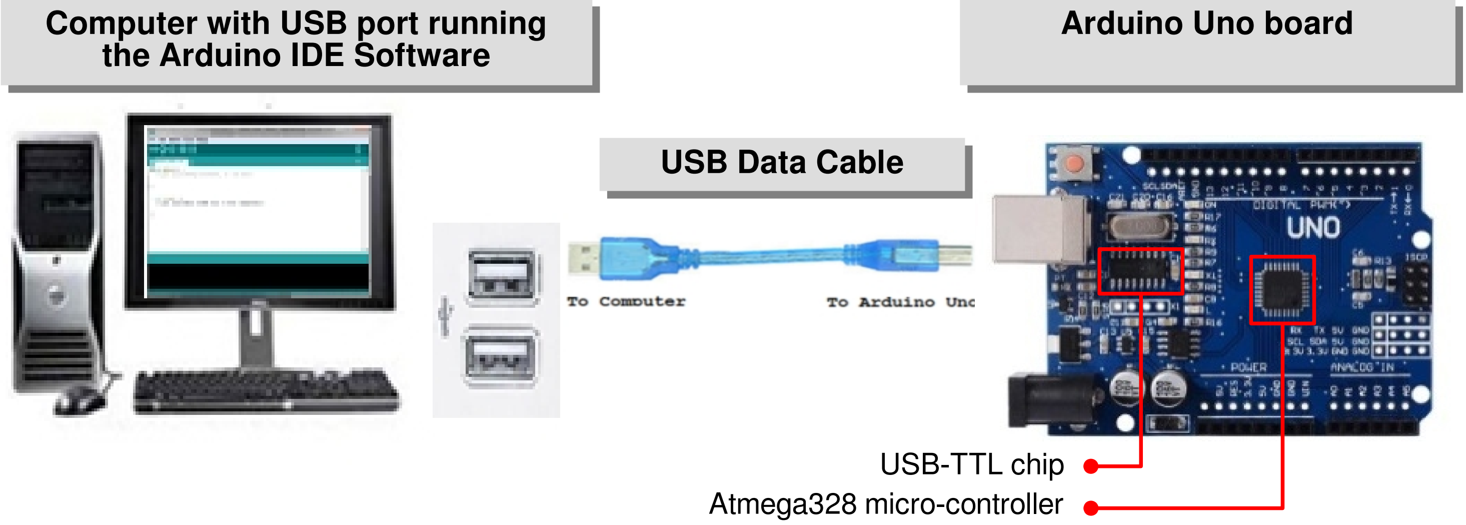
DOWNLOAD OPTIONS

Windows Win 7 and newer
Windows ZIP file
Windows app Win 8.1 or 10 Get
Linux 32 bits
Linux 64 bits
Linux ARM 32 bits
Linux ARM 64 bits
Mac OS X 10.10 or newer

Choose version according to the Computer Operating System.
1.Download
2.Install (use default options)

The Arduino Uno board has a USB-TTL chip connected to the Atmega328 micro-controller. This chip allows us to use **USB data cable** to do data transfer between Computer and the Atmega328 micro-controller on the Arduino Uno board.

Basic Arduino Hardware/Software Setup



Computer with USB port running the Arduino IDE Software

Arduino Uno board

USB Data Cable

USB-TTL chip

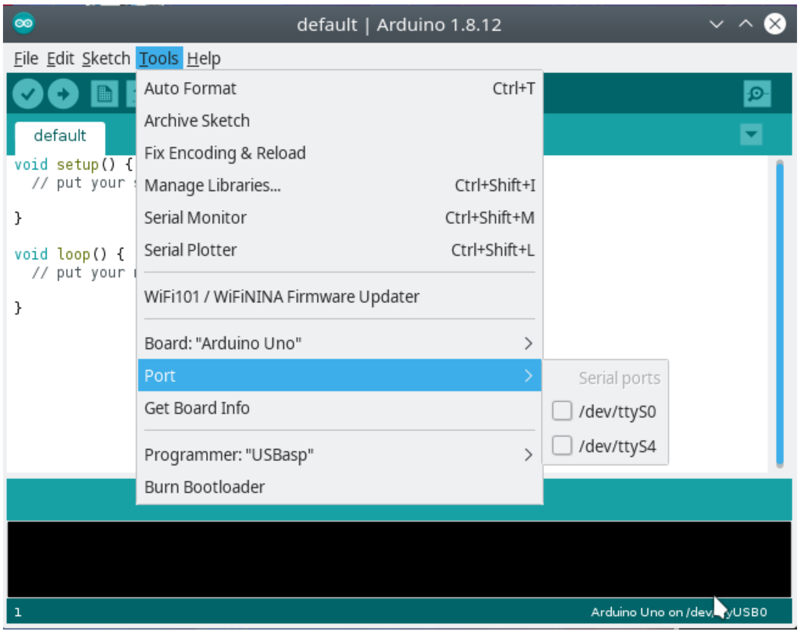
Atmega328 micro-controller

The Arduino IDE Software runing on our Desktop Computer are now able to send our Program from the Desktop Computer into the Atmega328 micro-controller and also to receive data from the Atmega328 micro-controller.

Arduino Programming – First Program in 6 steps (1/3)

Step 1. Computer and Arduino Uno, USB connection Setup

From the Arduino IDE Software Menu, select **“Tools | Port >”**



```
void setup() {
  // put your
}

void loop() {
  // put your
}
```

Test 1: Connect Arduino Uno the Desktop Computer with USB Cable, from Arduino IDE Software Menu, select **“Tools | Port >”** a new “Port” will appear.

Test 2: Disconnect Arduino Uno the Desktop Computer from USB Cable, from Arduino IDE Software Menu, select **“Tools | Port >”** a “Port” will disappear.

The “Port” that “appear” and “disappear” is the Connection name between our Desktop Computer and the Arduino Uno USB-TTL chip. **Make sure it is “Selected”**

Optional (for troubleshooting):
If you cannot see the “Port” appear/disappear in **Test 1:** and **Test 2:**, something has gone wrong.

It is most likely the USB-TTL driver for the USB-TTL chip is not present on the Computer. We will need to install the USB-TTL driver. Most China made Arduino Uno clones uses the CH340 USB-TTL chip. We can download this USB-TTL driver from the following link

http://www.wch.cn/download/ch341ser_exe.html

If the Arduino Uno is using other USB-TTL chip, we have to find its driver and install it on the Computer.

Step 2. Arduino UNO Board selection

From the Arduino IDE Software Menu, select **“Tools | Board >”**. Make sure **Arduino Uno board** is selected

By default Arduino Uno is already selected. The Arduino IDE software can work with other type of micro-controller board, that is when we select another board instead of Arduino Uno.

NOTE:
Step 1 and Step 2 are usually optional, we only need to do it once and they will become the default. However, it is good for us to confirm them each time before we start Programming.

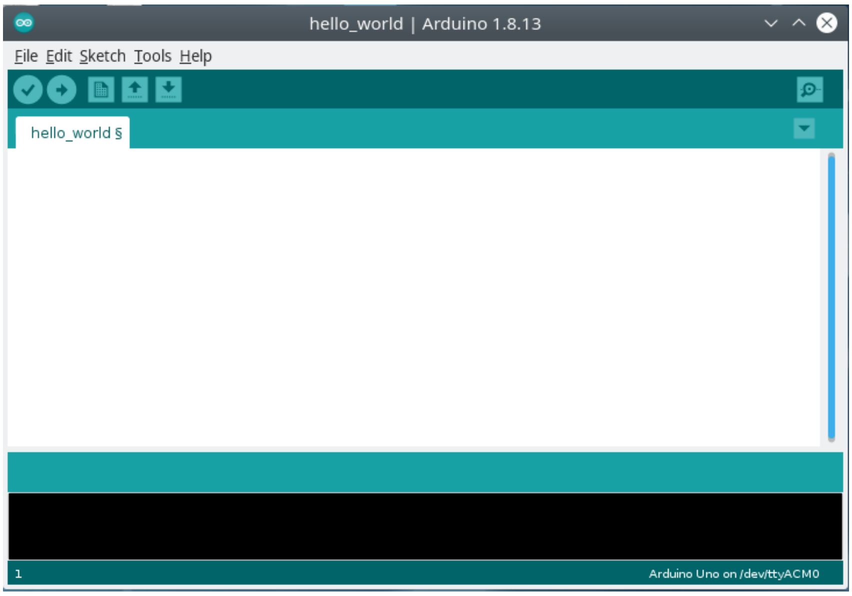
Arduino Programming – First Program in 6 steps (2/3)

Step 3. Create a new program using the Arduino IDE Software

The Arduino IDE Software starts with its **Code Editor** screen with a default Program loaded.

To prevent potential mistakes of “over-writing” our existing Program, we do not immediately key-in our Program codes into the Arduino IDE Software Code Editor, we always start by making a new copy.

1.From the Arduino IDE Software menu, select **“File | Save As...”**
2.Key in a new Program name = **“t1_helloworld”** and then Click **“Save”** button
3.In this tutorial, remove all existing codes from the Code Editor so that we can have a clean start


hello_world | Arduino 1.8.13
File Edit Sketch Tools Help
hello_world §
1
Arduino Uno on /dev/ttyACM0

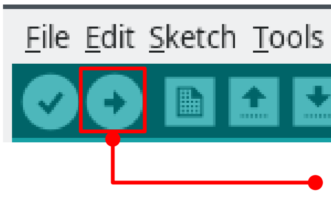
Step 4. Key in our new Program

Key in the codes below into the Arduino IDE Software Code Editor

```
// Program: t1_helloworld
//
void setup() {
  pinMode(13, OUTPUT);
}
void loop() {
  digitalWrite(13, HIGH);
  delay(1000);
  digitalWrite(13, LOW);
  delay(1000);
}
```

Step 5. Upload the Program into the Atmega328 micro-controller

Click the **“Upload Icon”** or From Arduino IDE Software menu, select **“Sketch | Upload”**

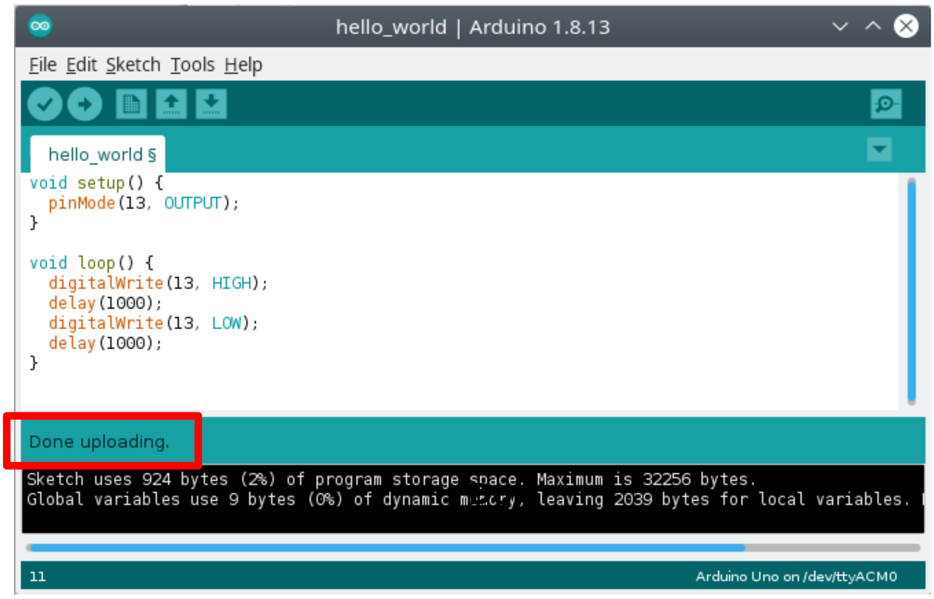


Upload Icon

Arduino Programming – First Program in 6 steps (3/3)

Step 6. Done Uploading

The Arduino IDE Software will show a status at the bottom of the screen “Done Uploading”, means it has successfully sent our Program into the Atmega328 micro-controller. We are Done!!!


hello_world | Arduino 1.8.13
File Edit Sketch Tools Help
hello_world §

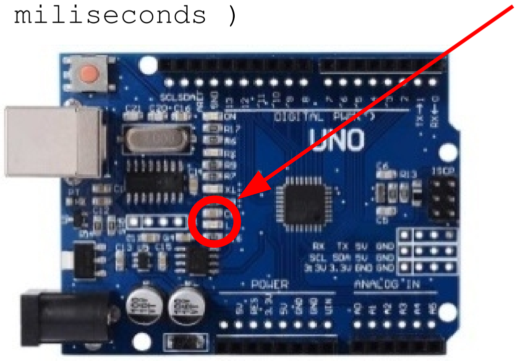
```
void setup() {
  pinMode(13, OUTPUT);
}

void loop() {
  digitalWrite(13, HIGH);
  delay(1000);
  digitalWrite(13, LOW);
  delay(1000);
}
```

Done uploading.

```
Sketch uses 924 bytes (2%) of program storage space. Maximum is 32256 bytes.
Global variables use 9 bytes (0%) of dynamic memory, leaving 2039 bytes for local variables.
```

11
Arduino Uno on /dev/ttyACM0

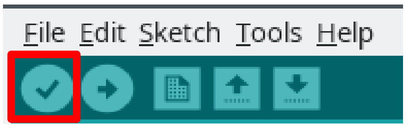
As soon as the Program is successfully uploaded into the Atmega328 micro-controller “Done Uploading”, the micro-controller will “reset” itself and run the instructions codes inside this Program (Watch the LED on the Arduino Uno board, it will be switched ON and OFF every 1000 miliseconds)



This program will stay inside the micro-controller permanently (until we replaced it with a new Upload). Whenever we “Power-Up” or “Reset” this Arduino Uno/Atmega328 micro-controller, this Program will run automatically.

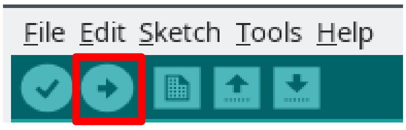
The difference between “Compile” and “Upload” in the Arduino IDE Software

Compile



“Compile” will check the C-Language Codes in the Program Code Editor for errors, if no errors are found, a “Machine Code” file will be generated otherwise errors will be shown at the bottom of the screen.

Upload



“Upload” will run the “Compile” option, if the “Machine Code” file is sucessfully generated by the “Compile” option, the “Upload” will continue to sent the “Machine Code” into the into the micro-controller.

After we have uploaded our Program into the Atmega328 micro-controller, Arduino Uno can be removed from the Computer.

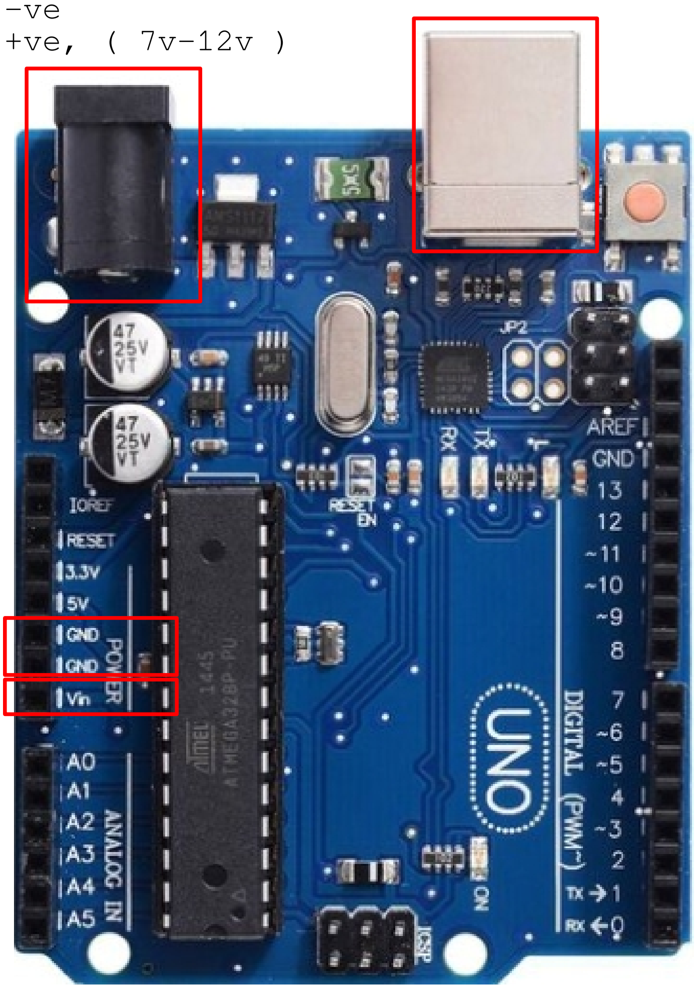
The Arduino Uno and its parts can work independently from the Computer. All we need to do is, to give it Power Supply.

Power supply to Arduino Uno can be from any ONE of these methods

1. DC Jack from Power Supply 5.5x2.1mm (7v-12v)

2. USB Cable from Power Supply (5v)

3. 2-Wire Power Supply - +ve Wire to Vin Pin and -ve Wire to Ground Pin



DC Jack 5.5x 2.1mm
Outer -ve
Inner +ve, (7v-12v)

USB Cable (5v)

2-Wire Power Supply
-ve Termnial to any GND pin
+ve Terminal to Vin pin
Same as *DC Jack
(7v to 12v)

Arduino IDE – Serial Monitor

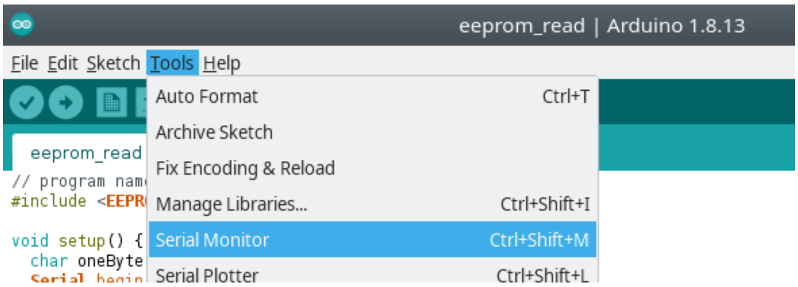
We are able to send our Program into the ATMEGA328P micro-controller with our Arduino IDE Software “Upload” facility. The Arduino IDE Software can also receive information from our Program running inside the Atmega328 micro-controller via the Arduino IDE Software, “Serial Monitor” facility.

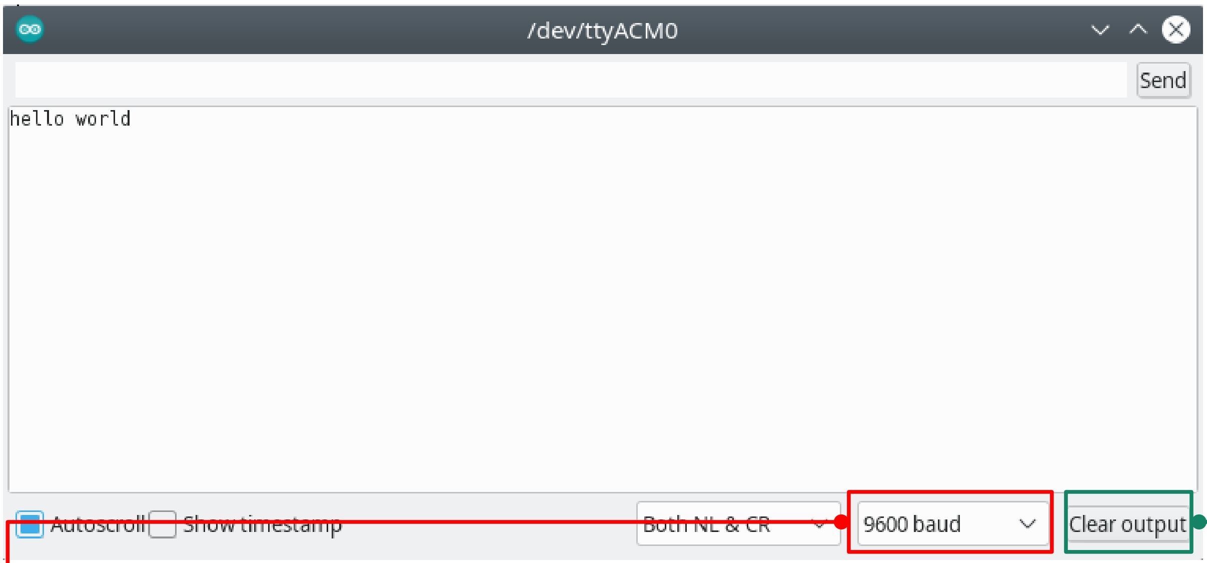
Use the Arduino IDE Software
1.Save as “t1 helloworld serial monitor”
2.Key in the codes below into the Arduino IDE Software Code Editor and Upload

```
// Program: t1_helloworld_serial_monitor
//
void setup() {
  Serial.begin(9600);
  Serial.println("hello world, serial monitor");
}
void loop() {
}
```

This time we will not see blinking LED but we will see something on the “Serial Monitor” screen

From Arduino IDE Software menu, select **“Tools | Serial Monitor”**


eeprom_read | Arduino 1.8.13
File Edit Sketch Tools Help
Auto Format Ctrl+T
Archive Sketch
Fix Encoding & Reload
Manage Libraries... Ctrl+Shift+I
Serial Monitor Ctrl+Shift+M
Serial Plotter Ctrl+Shift+L
eeprom_read



```
hello world
```

Make sure “baud” is set to the same value as “Serial.begin(9600);” that is coded in our Program

Sometimes the Serial Monitor contains data from previous session, Click on “Clear Output” button and press “Reset” button on the Arduino Uno board to get a new clean session

We are able to see the text “hello world” in our Serial Monitor screen. When our micro-controller CPU executes the Serial.println("hello world") instruction code, the “hello world” text will be sent from our Atmega328 micro-controller to our Computer via the USB interface, and is picked up by our Arduino IDE Software’s Serial Monitor facility.