

IoT App Development



Introduction to Arduino

Arduino UNO Board & IDE

Dr. Manas Khatua

Associate Professor

Dept. of CSE

IIT Guwahati

E-mail: manaskhatua@iitg.ac.in

What is Arduino?



- Arduino is an **open-source electronics platform**
 - based on easy-to-use hardware and software.
- It was born at *Interaction Design Institute Ivrea, Italy*
 - an easy tool **for fast prototyping**
 - aimed at students without any background in electronics and programming.
- These boards are incorporated with **microcontrollers**
 - To execute a small program, to receive input, to apply action on real world
- It has the capability to **act as an interface** for electronic systems
- These boards **are used extensively** because:
 - **Inexpensive**
 - **Cross-platform** – runs on Windows, Mac OS, and Linux OS.
 - **Easy-to-use** hardware and software environment
 - **Open source** hardware and software IDE
 - **Easy interface** with other boards and computers
 - Can **interact with sensors** and actuators
 - Facilitate **serial communication**

Microprocessor vs Microcontroller



- A **microprocessor (μP)** is essentially the **CPU on a single chip**, designed for general-purpose computing.
- A **microcontroller (μC)** is a self-contained **system-on-chip** that integrates the CPU, on-chip memory (RAM, ROM, Flash), and I/O peripherals like ADCs, timers, and communication interfaces.

	μP	μC
Architecture	needs external components	has all essentials on-chip
Performance	runs at GHz for multitasking;	runs at kHz–MHz for real-time control
Memory	uses scalable external memory	has fixed on-chip memory
I/O	I/O components are to be connected externally	I/O components are available
Power	consumes more power	highly power-efficient
Examples	Intel Core i7, AMD Ryzen, ARM Cortex-A.	ATmega328P, ESP32, RP2040, ARM Cortex-M series

Types of Arduino Boards

- **Entry Level** - easy to use and ready to power your first creative projects.
 - Arduino UNO
 - Arduino Nano
 - Arduino Micro
- **Enhanced Features** - boards with advanced functionalities, or faster performances
 - Arduino Zero
 - Arduino Mega 2560
 - Arduino Motor Shield
- **Internet of Things** - Make connected devices easily with one of these IoT products
 - Arduino Nano 33 IoT
 - Arduino Nano 22 BLE
 - UNO WiFi REV2



Arduino UNO



Arduino Mega 2560

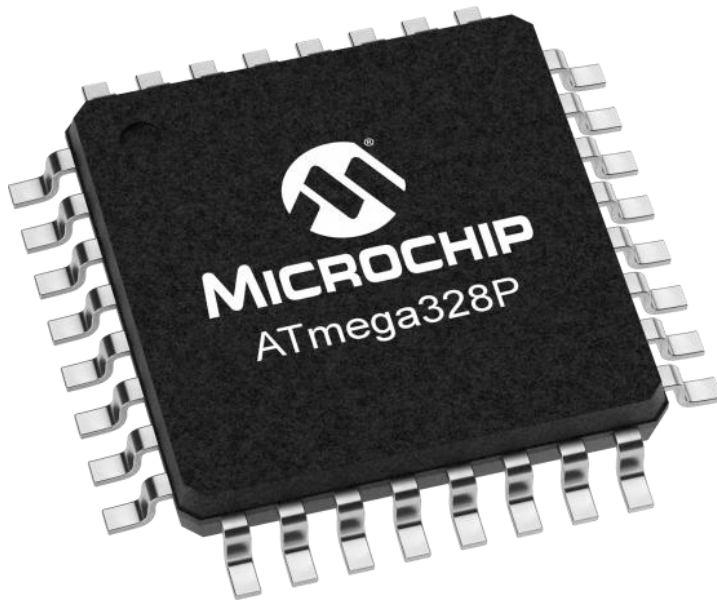


Arduino Nano 33 IoT

Source: <https://www.arduino.cc/>

Arduino UNO

- **Arduino UNO** is a Single board Microcontroller based on **ATmega328P** Processor
 - a product of **Atmel** (now Microchip)
 - **32** - represents it's **flash memory** capacity that is 32KB
 - **8** - represents it's **CPU** type that is of 8 bit
 - **p** - simply denotes **picoPower** (i.e. very low power).



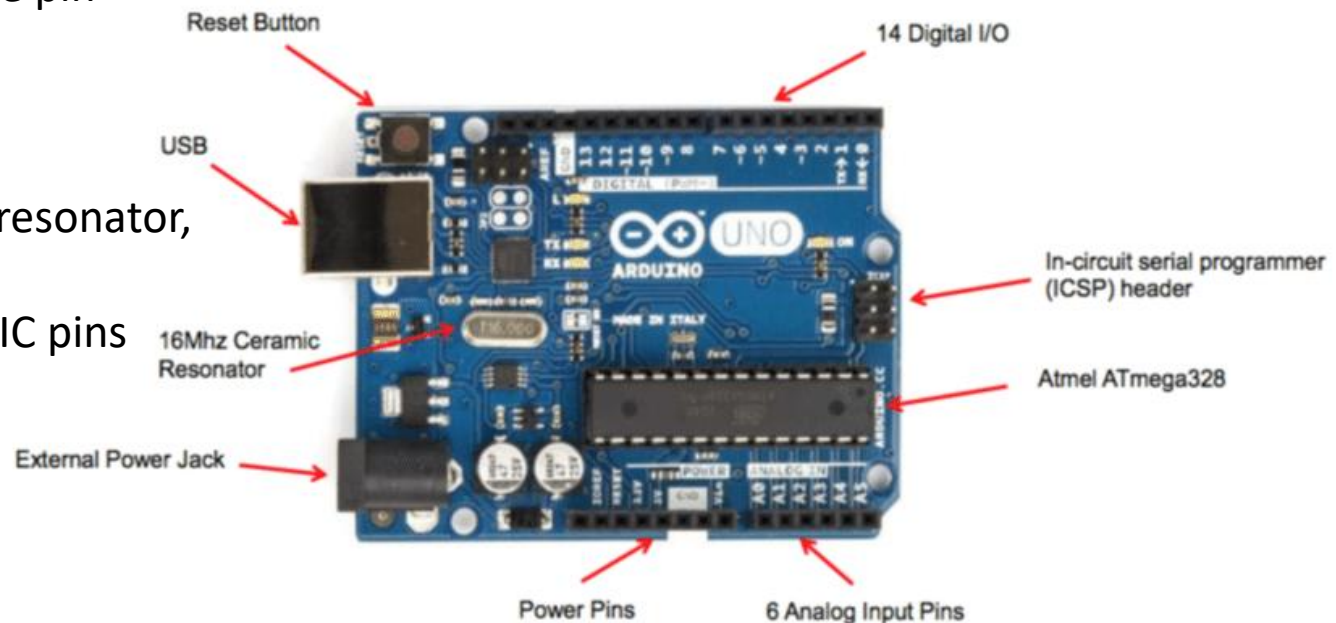
Few competitors: STM32 **ARM Cortex-M**,
MSP430, and PIC MCU



AVR CPU at up to 16 MHz, 2KB SRAM
{32KB Flash, 1KB EEPROM}->Non-volatile.

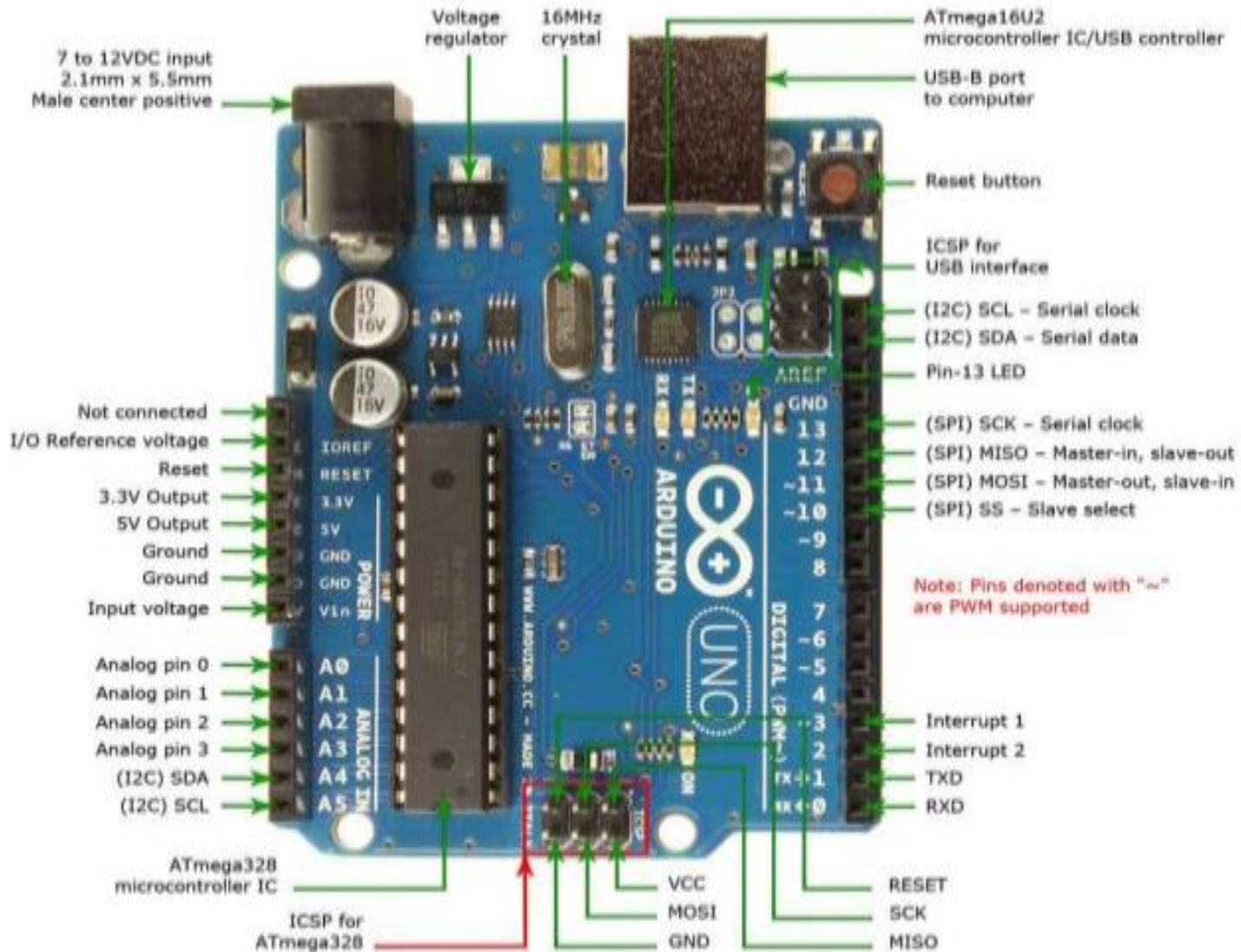
Pins/Jacks in Arduino UNO R3

- It has the following major pins/jacks:
 - ✓ 14 digital input/output pins (of which 6 can be used as PWM outputs),
 - ✓ 6 analog inputs,
 - ✓ 6 pins related to energy/power
 - ✓ a reset pin
 - ✓ an analog reference pin
 - ✓ a reset button
 - ✓ a USB connection,
 - ✓ a power jack,
 - ✓ a 16 MHz ceramic resonator,
 - ✓ two ICSP header
 - ✓ Atmel ATmega328 IC pins



Source: <https://docs.arduino.cc/hardware/uno-rev3>

Detailed Pin Diagram



Pin Description

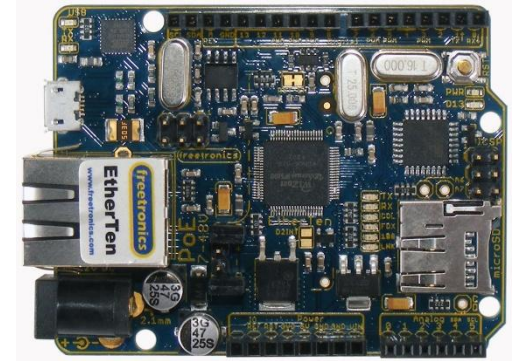
Pin category	Pin Name	Details
Power Pins	Vin, 3.3V, 5V, GND, RESET	Vin : Input voltage to Arduino when using an external power source. 5V : Regulated power supply used to power microcontroller and other components on the board. 3.3V : 3.3V supply generated by on-board voltage regulator. Maximum current draw is 50mA. GND : ground pins. Reset: Reset the microcontroller
ICSP: In-Circuit Serial Programming	ICSP pins: MISO, VCC, SCK, MOSI, RESET, GND	Used to code and boot an Arduino from an external source. Allow inter workings of two or more Arduino boards. Allow you to upload your firmware.

Cont...

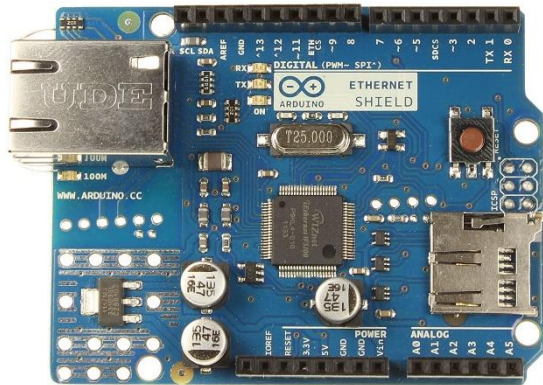
Pin category	Pin No / name	Details
Analog pin	A0 - A5	Used to provide analog input in the range 0-5V.
Digital Input/output pin	Digital Pins 2 - 13	Can be used as input or output pins.
Serial Communication	0(Rx),1(Tx)	Used to receive and transmit TTL serial data.
External Interrupts	2, 3	To trigger an interrupt.
PWM: Pulse Width Modulation	3, 5, 6, 9, 10, 11	Provides 8-bit PWM output.
SPI: Serial Peripheral Interface	10 (SS), 11 (MOSI), 12 (MISO) and 13 (SCK)	Used for SPI communication.
Inbuilt LED	13	To turn on the inbuilt LED.
I2C: Inter-IC, or TWI: Two Wire Interface	A4 (SDA: Serial Data), A5 (SCL: Serial Clock)	Used for TWI / I2C communication.
AREF	AREF : Analog Reference Voltage	To provide reference voltage from an external power supply for analog-to-digital conversion of inputs to the analog pins. E.g. if AREF is 4V, the analogRead() range of 0~1023 will relate to 0~4V and not 0~5V.

Arduino in IoT

- Arduinos are used to create IoT projects.
- But, it requires either a specialized Arduino or shields to provide network capabilities
- The **network interface** could be Ethernet / WiFi / Cellular



EtherTen



Arduino Ethernet Shield



Arduino + Ethernet Shield



Arduino UNO WiFi Rev2

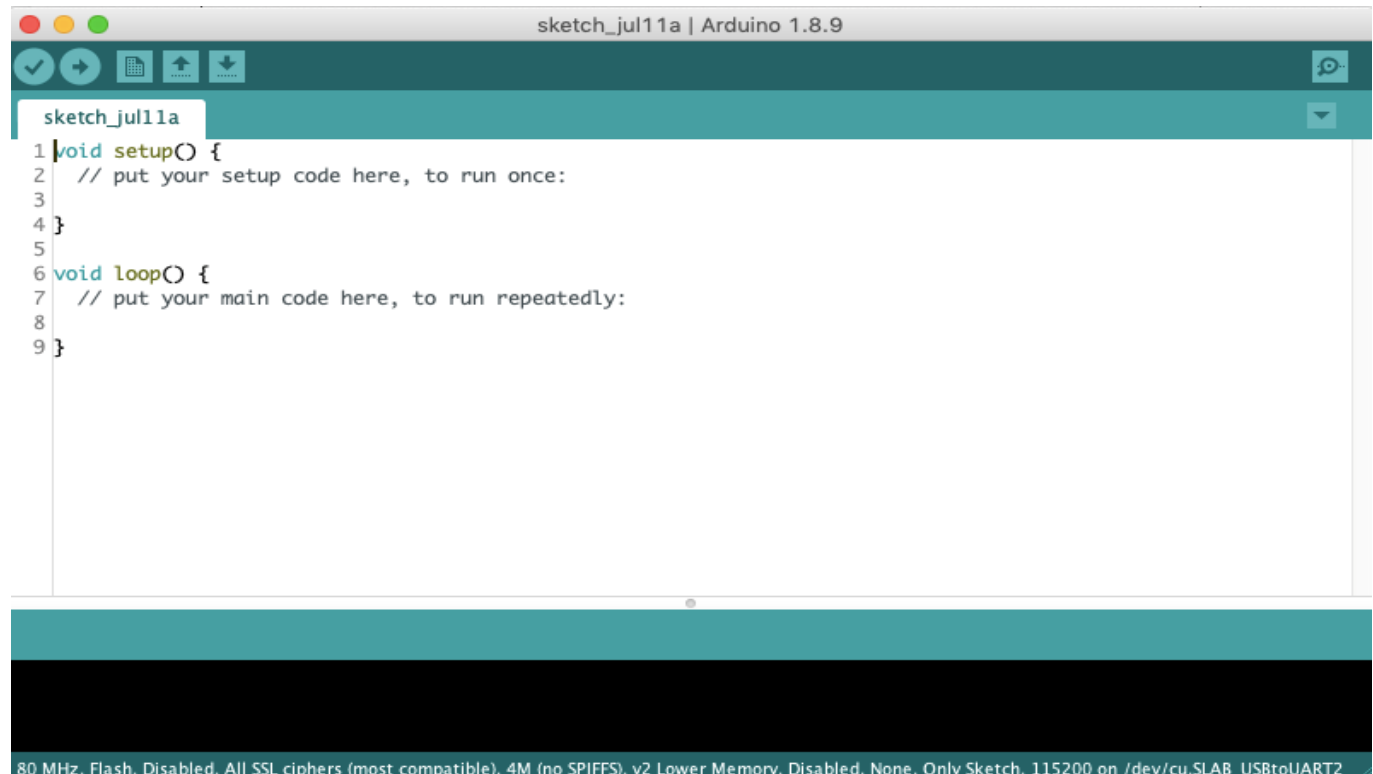
Configure Arduino IDE

- Download and Install Arduino IDE <https://www.arduino.cc/en/Main/Software>
- The Arduino Software (IDE) allows you to write programs and upload them to your board.
- When the Arduino IDE first opens, this is what you should see:

setup () { ... }



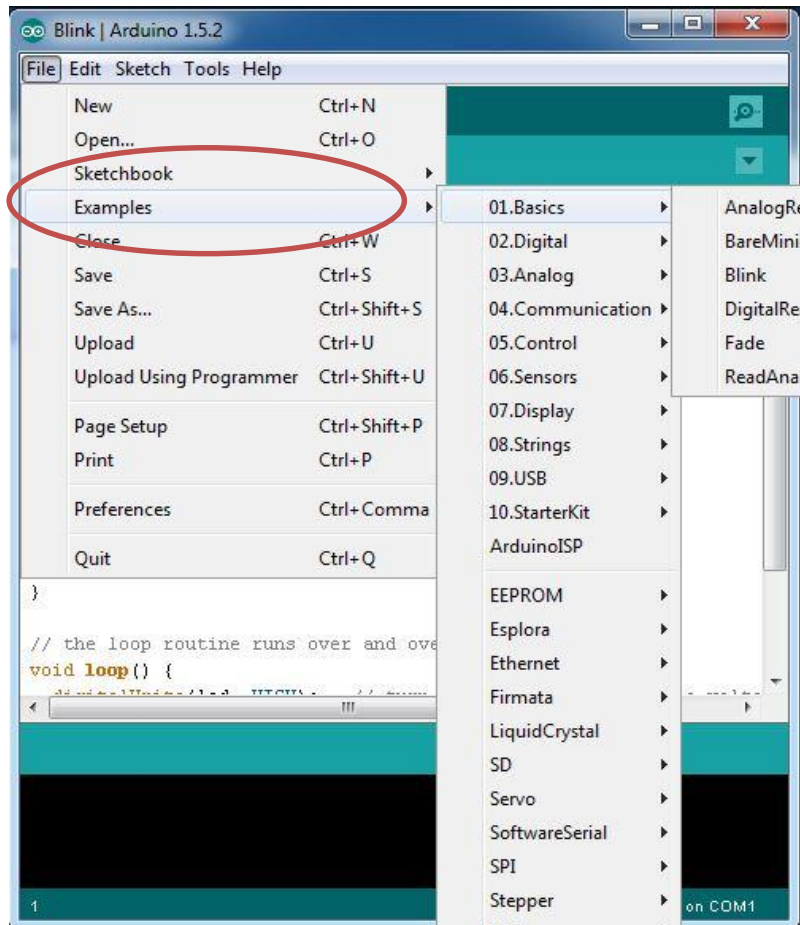
loop () { ... }



```
sketch_jul11a
1 void setup() {
2   // put your setup code here, to run once:
3 }
4
5
6 void loop() {
7   // put your main code here, to run repeatedly:
8 }
9
```

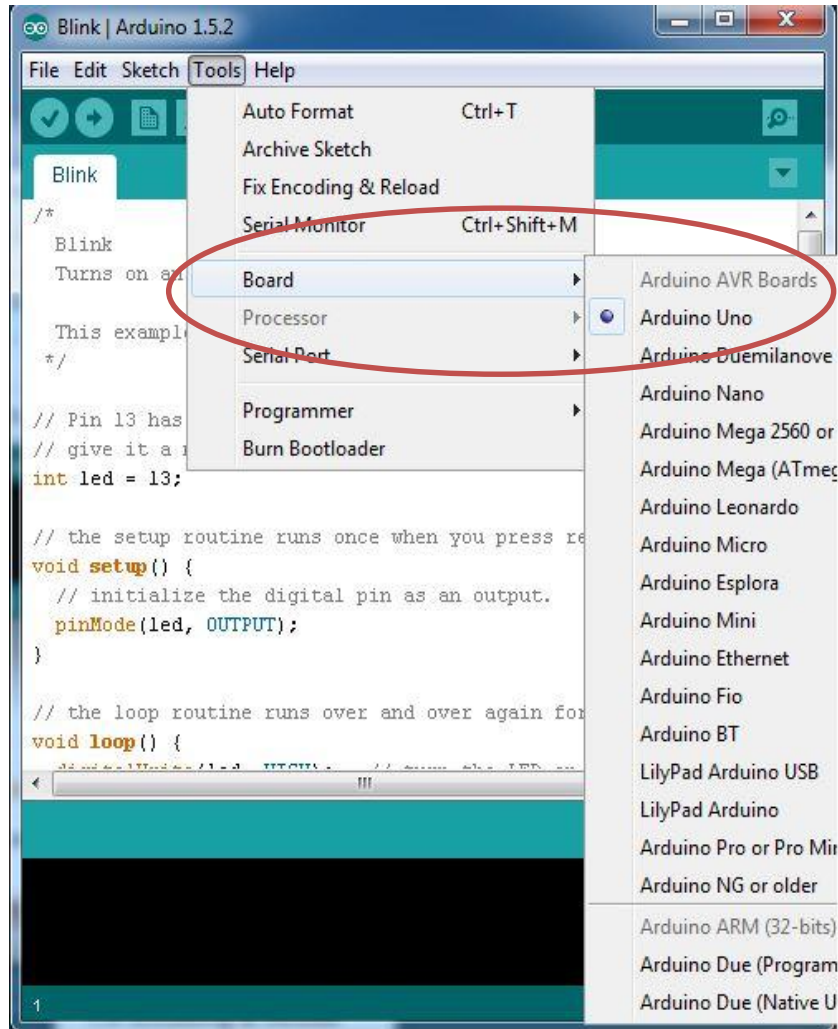
80 MHz, Flash, Disabled, All SSL ciphers (most compatible), 4M (no SPIFFS), v2 Lower Memory, Disabled, None, Only Sketch, 115200 on /dev/cu.SLAB_USBtoUART2

Built-in Examples



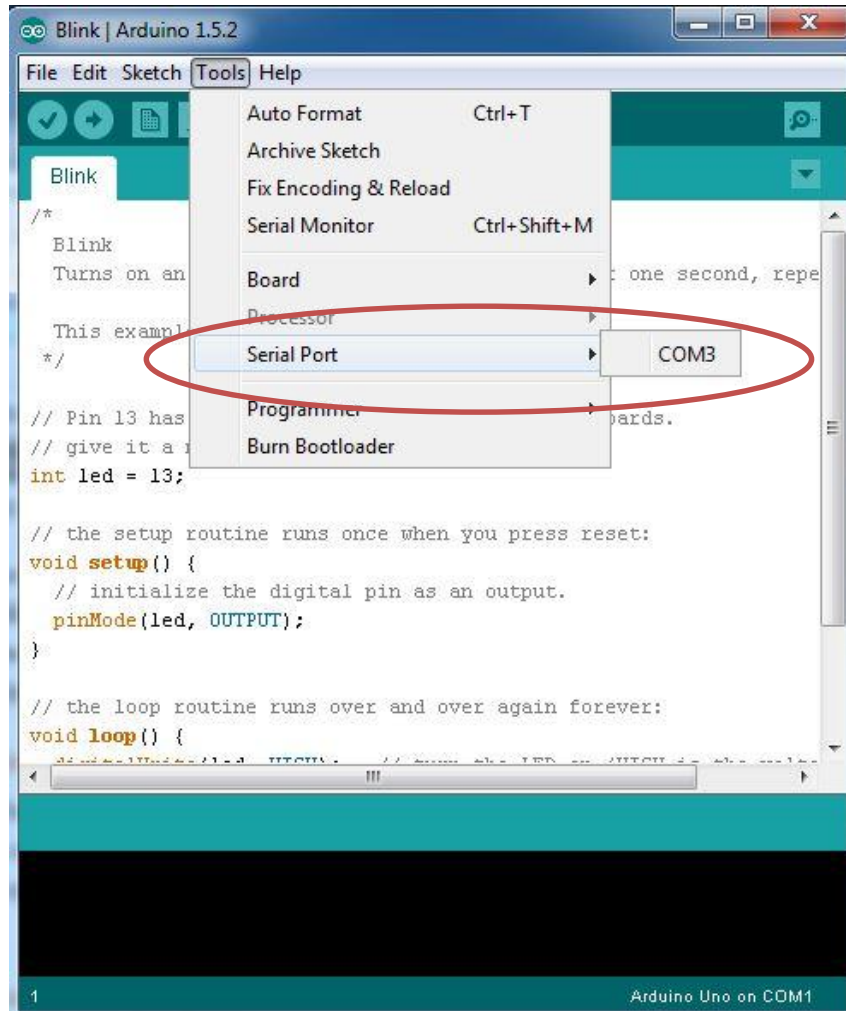
- Launch the Arduino application
- Programs written using Arduino IDE are called **sketches**.
- There are many **built-in examples / sketches**.
- To open built-in examples: select **File** -> **examples**.
- These simple programs demonstrate all the basic Arduino commands.

Set Arduino Board



- Plug in your board through cable
- Select the **type of Arduino board** you're using:
 - ✓ **Tools -> Board -> (your board type)**
 - ✓ e.g. Arduino UNO

Set Serial Port



- Select the **serial/COM** port that your Arduino is attached to:
✓ **Tools > Port > COMxx**

Note: If you're not sure in which serial port your Arduino is connected, take a look at the available ports, then unplug your Arduino and look again. The one that disappeared is your Arduino.

Code Compilation

Code Uploading



```
local_server | Arduino 1.8.9

1 #include <ESP8266WiFi.h> //Including ESP8266 library
2 #include <ESP8266WebServer.h> //Including ESP8266WebServer library for web serv
3 #include <ThingSpeak.h> //Including ThingSpeak library
4
5 IPAddress IP(192,168,4,15); //Static IP address of local server
6 IPAddress gateway(192,168,4,1); //Gateway of the network
7 IPAddress mask(255, 255, 255, 0); //Subnet mask of the network
8
9 WiFiClient client;
10 WiFiServer server(80);
11
12 unsigned long myChannelNumber = 814887; //Replace with channelID of ThingSpeak
13 const char * myWriteAPIKey = "EK4LTPHNU4GGE0VP"; //Replace with WriteAPIKey of
14
15 const char* softAPssid = "ESP8266"; //SSID of the hotspot of ESP8266 acting
16 const char* password = "12345678"; //Password of the hotspot of ESP8266 act
17
18 const char* wifissid = "Tenda_8060A0"; //Replace with SSID of WIF router provi
19 const char* pass = "12345678"; //Password of WIFI router providing inte

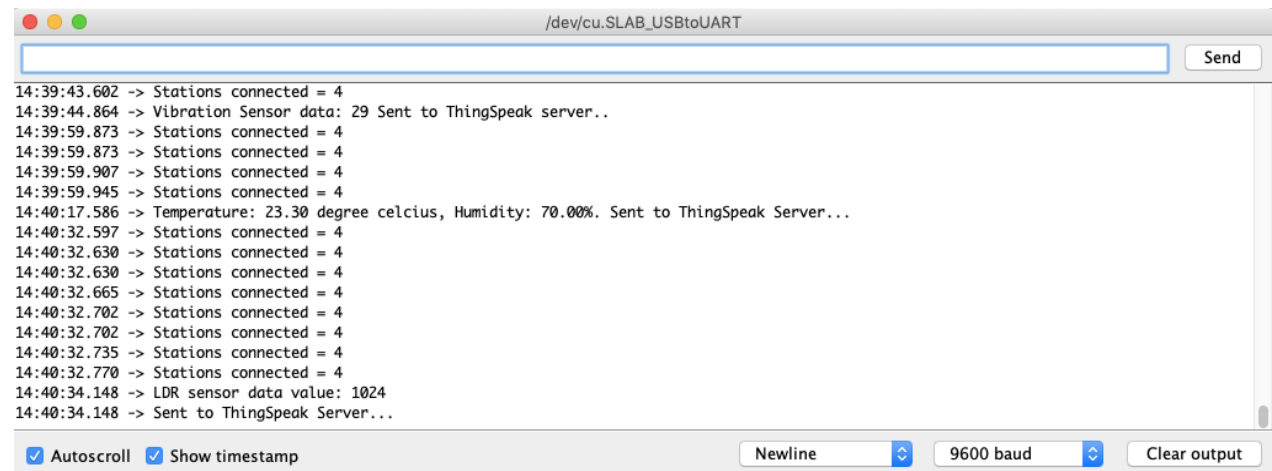
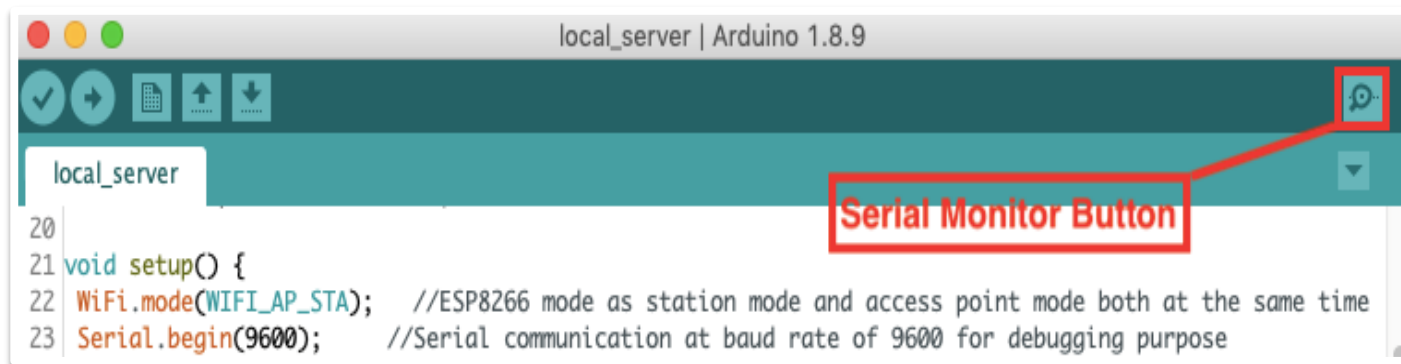
Done uploading.
TRACE +0.000 Received full packet: 011202000000000000000000
Hard resetting via RTS pin...

NodeMCU 1.0 (ESP-12E Module) on /dev/cu.SLAB_USBtoUART
```

- With your **Arduino board connected**, and the Blink sketch open, press the '**Upload**' button
- After a second, you should see some LEDs flashing on your Arduino, followed by the message '**Done Uploading**' in the status bar of the Blink sketch.
- If everything worked, the **on-board LED on your Arduino should now be blinking!**

Serial Monitor

- The serial monitor is the 'tether' **between the computer and your Arduino** - it lets you send and receive text messages.
- First **select the port** (go to **Tools -> Port:**) to which the board is connected then click the icon of **Serial Monitor** on the top right side of the Arduino IDE

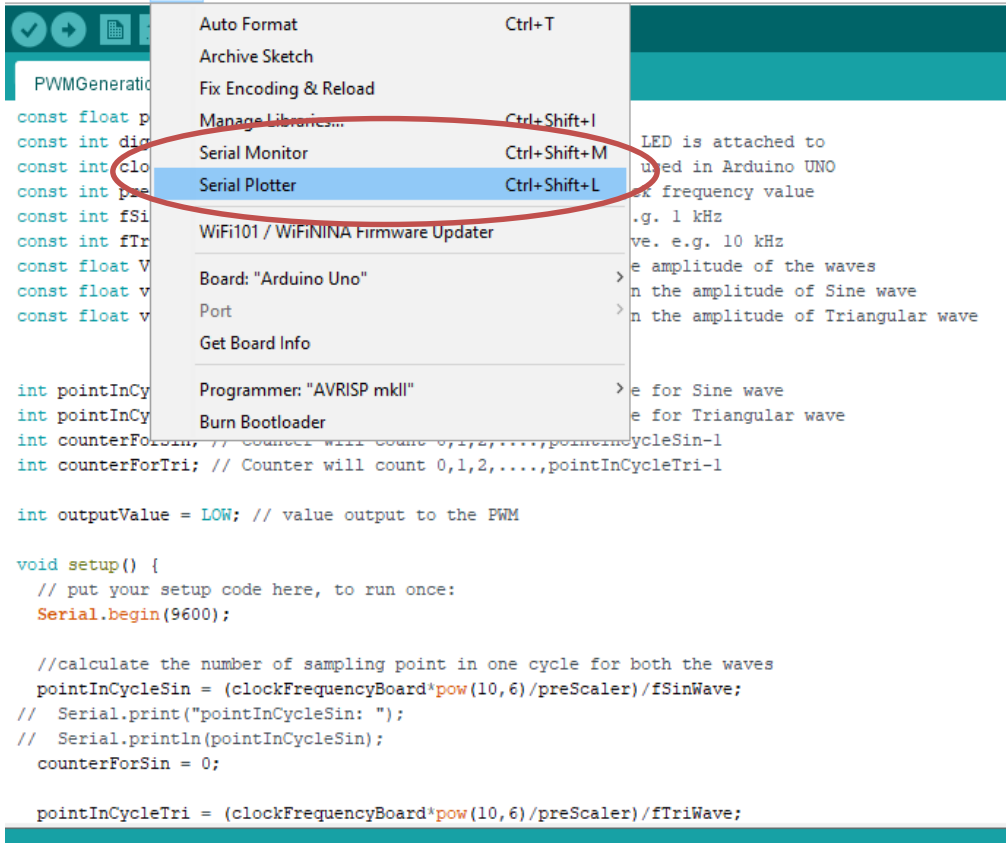


Serial Monitor
output

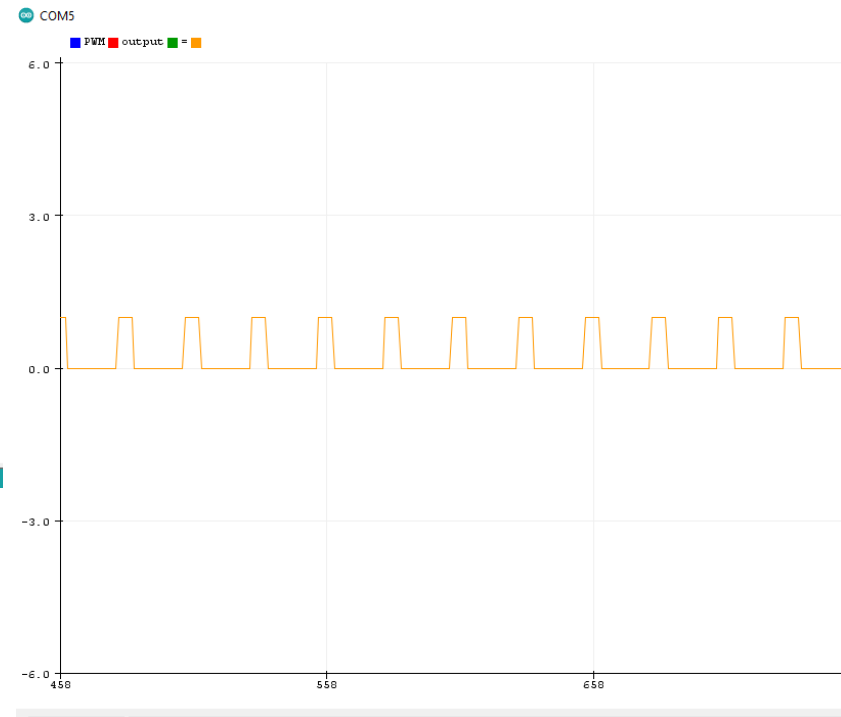
Serial Plotter

PWMGenerationSineTriangularWave | Arduino 1.8.19 (Windows Store 1.8.57.0)

File Edit Sketch Tools Help

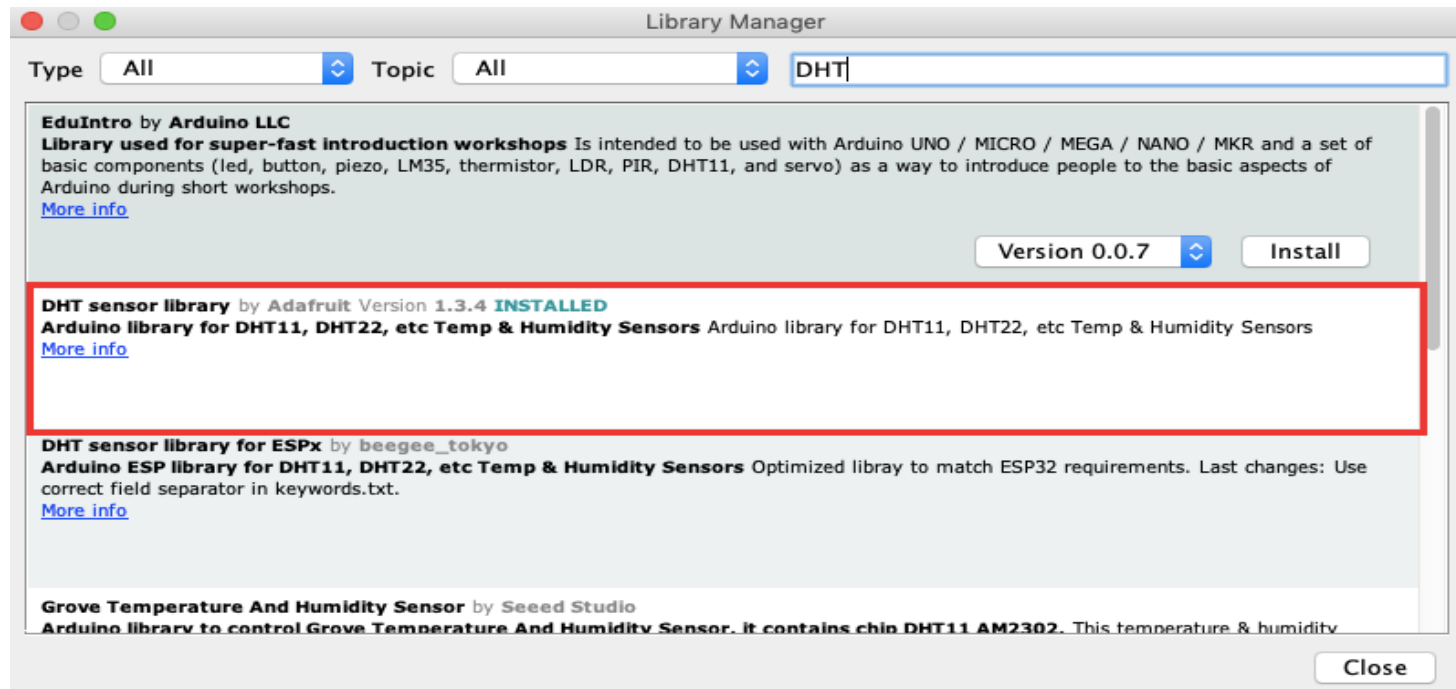


- Can use Serial Plotter to plot the output signal
- See the below image for example



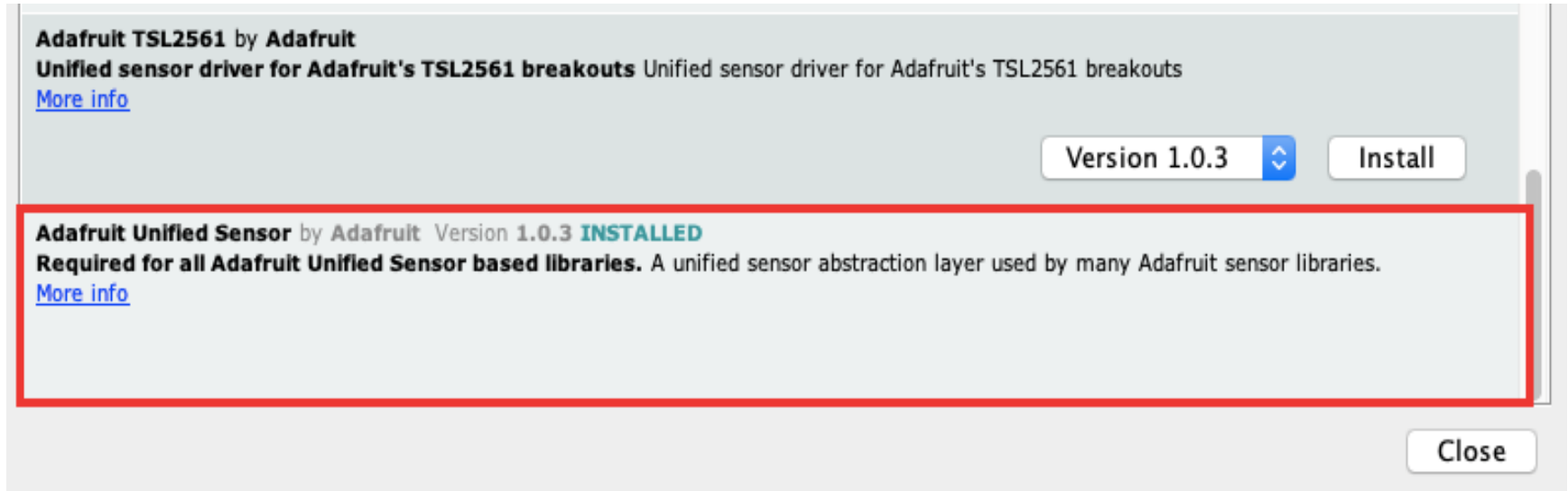
How to Install Sensor Libraries

- Let we will use **DHT11 sensor** for which we need **DHT.h** header file
- So, this header file needs to be **installed** first.
- **Install Using the Library Manager**
 - click to **Sketch** menu -> **Include Library** -> **Manage Libraries**
 - Search for “**DHT**” on the Search box and **install** the DHT library from **Adafruit**.



Cont...

- After installing the DHT library from Adafruit, **install** “**Adafruit Unified Sensor**” libraries.



- **There exist other methods for installing libraries**
 - **Importing a .zip Library**
 - Sketch --> Include Library --> Add .Zip Library
 - **Manual Installation of Library**
 - Download the library as .Zip --> extract it
 - Place the files in File --> Preferences --> Sketchbook location
 - Restart Arduino IDE

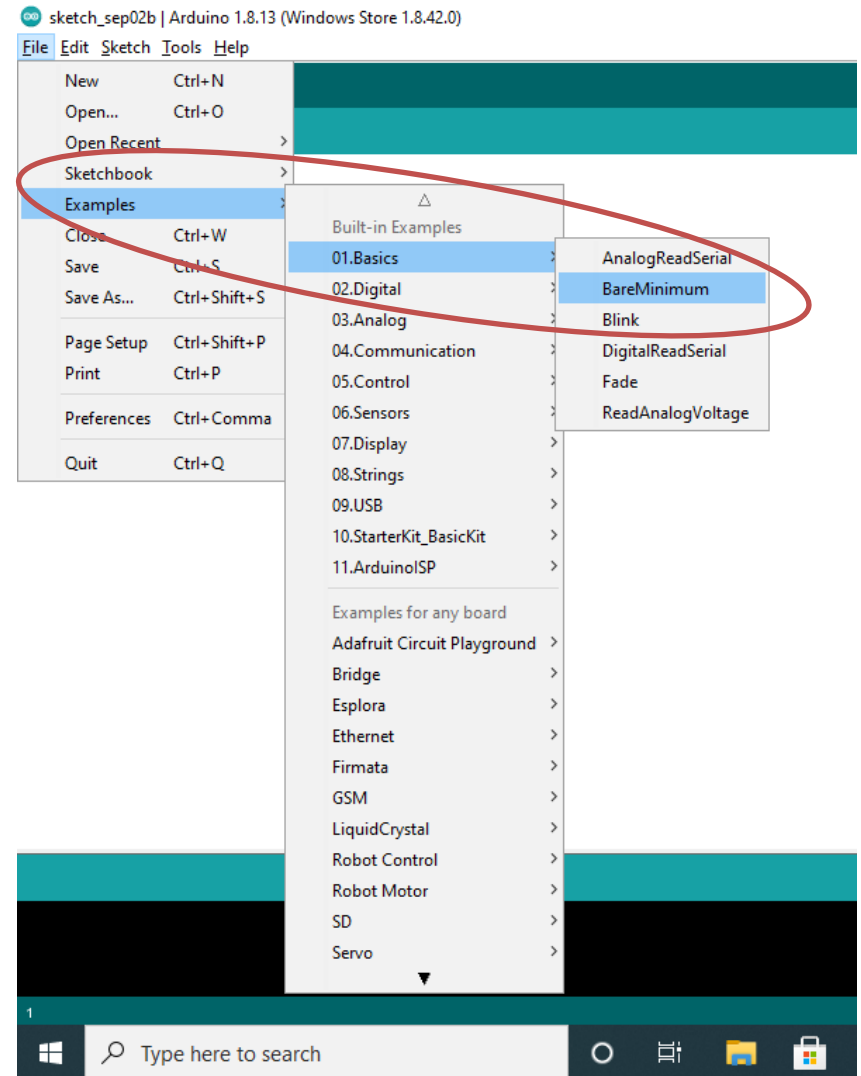
Demo: LED Blink



- See the Demo using Arduino UNO circuit board
 - 1) Blink the in-built LED of Arduino Board
 - 2) Blink the additionally attached LEDs

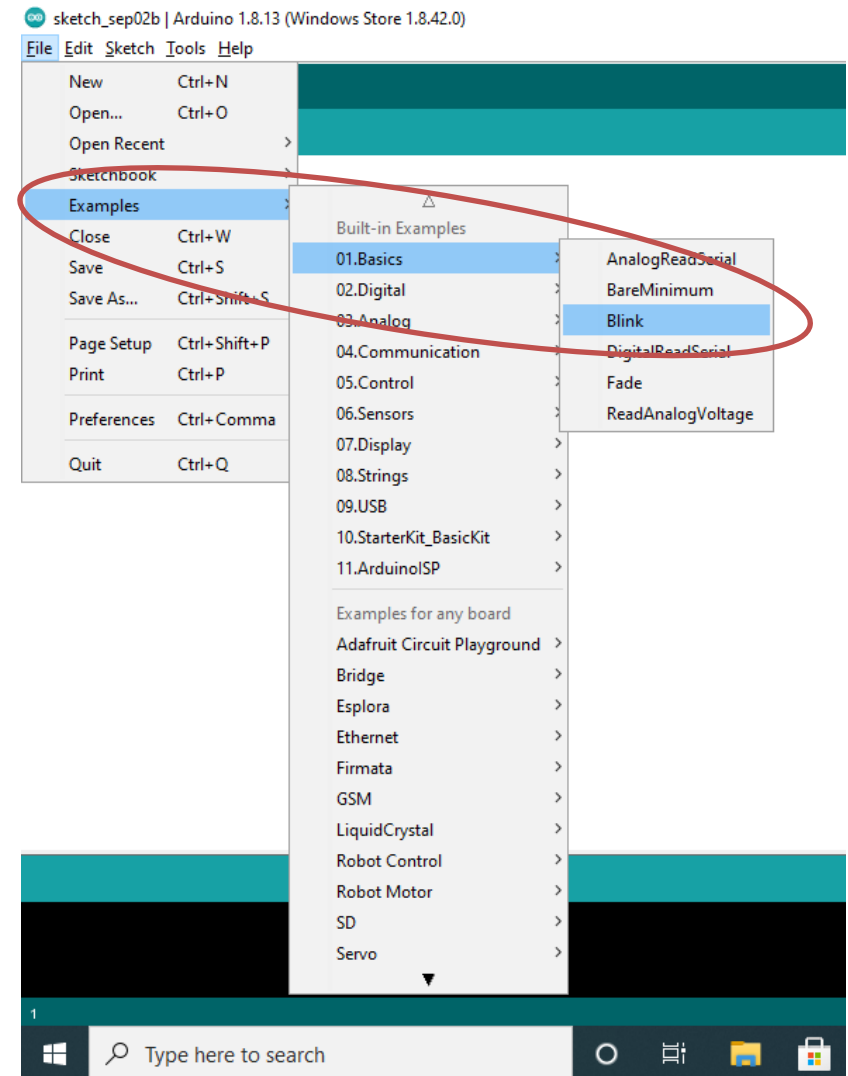
Blink In-built LED Continuously

- First upload the bare minimum example:
 - ✓ Files -> Examples -> Basics -> BareMinimum
- Output:
 - ✓ In-built LED will **glow continuously**

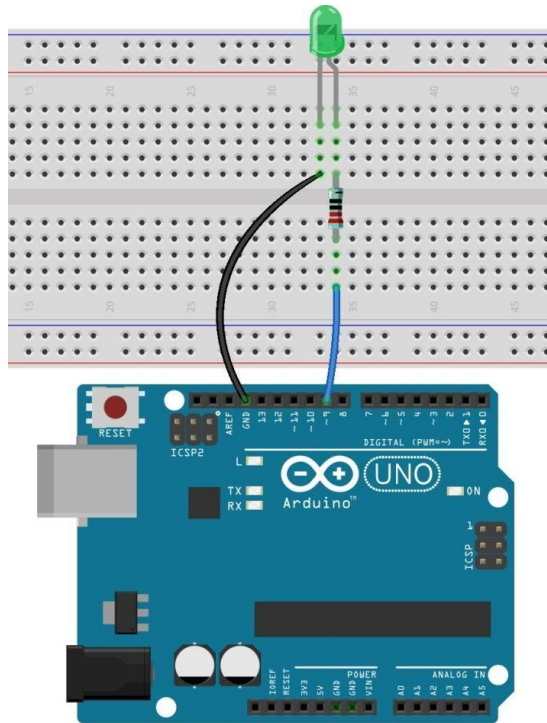


Blink In-built LED Periodically

- First upload the bare minimum example:
 - ✓ Files -> Examples -> Basics -> BareMinimum
- Output:
 - ✓ In-built LED will **glow continuously**
- Then, upload the blink example:
 - ✓ Files -> Examples -> Basics -> Blink
- Output:
 - ✓ In-built LED will **glow periodically**



Blink External LED



Connect the “digital out” **pin 9** with the “**Anode** pin” of LED (i.e. long leg of LED) through **1K Ohm resistor**, and “ground” pin with the **Cathode** pin of LED

```
BlinkExternalLED | Arduino 1.8.13 (Windows Store 1.8.42.0)
File Edit Sketch Tools Help

BlinkExternalLED

int animationSpeed = 0;
void setup() {
    // put your setup code here, to run once:
    pinMode(9,OUTPUT);
}

void loop() {
    // put your main code here, to run repeatedly:
    animationSpeed = 1000;
    digitalWrite(9,HIGH);
    delay(animationSpeed);
    digitalWrite(9,LOW);
    delay(animationSpeed);
}
```

Upload this sketch in
Arduino UNO

OUTPUT:

- LED connected with digital pin 9 will blink periodically.

Demo on LED Blink



IoT_PPT06_Intro_Arduino_Board_and_IDE - PowerPoint (Product Activation Failed)

File Home Insert Design Transitions Animations Slide Show Review View Add-Ins Foxit Reader PDF Tell me what you want to do... Manas Khatua Share

Clipboard Slides Font Paragraph Drawing Editing

Live Demo: LED Blink

- See the Live Demo using Arduino UNO circuit board
- 1) Blink the in-built LED of Arduino Board
- 2) Blink the additionally attached LEDs

17-09-2020 Dr. Manas Khatua 17

Click to add notes

Slide 17 of 22 English (India)

Notes Comments

Type here to search

Lessons Learned



- ✓ What is Arduino
- ✓ Types of Arduino Board
- ✓ Arduino UNO pin diagram
- ✓ Arduino in IoT
- ✓ Arduino IDE
- ✓ Built-in Sketch in IDE
- ✓ Compiling and Uploading a sketch using IDE
- ✓ LED blink program and system setup

Thanks!



Which is better? ATmega328P vs STM32 vs MSP430



	ATmega328P	STM32	MSP430
Brand	ATmel (now MicroChip)	Cortex (STMicroelectronics)	Texas Instruments
Cost	Low	High	Low
Architecture	Advanced RISC architecture	Power Architecture technology designed for embedded applications	Older, von-Neumann architecture
Power Consumption	Low	Medium	Low
Performance	Medium, suitable for complex projects	High, fast processing speed, Running 32 bit ARM processor core with sufficient RAM	Low, more suitable for only simple projects
Ease of Usage	Easy to use, 8 bit and high compatibility with Arduino boards	Complicated due to its nature of being a 32 bit microcontroller	Complex relative to Arduino boards