

WEB BASED IOT CONTROLLING OF HYBRID ENERGY SYSTEM USING ESP826

Mr.R.Madhan Mohan(8096671432),G.Saikalyani(6305589114),M.Swathi(8897847934),

P.Nandini(6302145950),G.Rambabu(6281833896),

Abstract:

In this project, we are controlling Street lights using IOT. There is various combination of energy and all of them are alternative to each other like solar energy, wind energy, bio fuel, fuel cell, etc. But the need of controlling of hybrid energy system arises when it is installed for domestic or commercial purpose. At this point IOT plays an important role in controlling system. The main criteria being switching between the two sources of energy i.e. solar and wind energy without any inconvenience through a website using Nodemcu module. The data is transmitted wirelessly to cloud through Nodemcu module which controls the sources of energy. The transmitted data is controlled remotely using IOT. This enables user to have flexible control mechanism remotely through a secured internet web connection. This system helps the user to control the sources of energy, manually and remotely using smart phone. This system is very efficient, cheaper and flexible in operation.

INTRODUCTION

Energy is the basic need for development and the requirement of energy is more due to the rapid increase in world population, technology and other political and economic condition. Now a day's electrical energy is generated by the conventional energy resources like coal, diesel, and nuclear etc. and these are depleting day by day. In this paper, authors have focused on controlling of hybrid energy system using IOT. There is various combination of energy and all of them are alternative to each other like solar energy, wind energy, bio fuel, fuel cell, etc. But the need of controlling of hybrid energy system arises when it is installed for domestic or commercial purpose. At this point IOT plays an important role in controlling system. The main criteria being switching between the two sources of energy i.e. solar and wind energy without any inconvenience through a website using ESP8266 Wi-Fi module. The data is transmitted wirelessly through website to ESP8266 module which controls the sources of energy. The transmitted data is controlled remotely using IOT. This enables user to have flexible control mechanism remotely through a secured internet web connection. This system helps the user to control the sources of energy, manually and remotely using smart phone or personal computer. This system is very efficient, cheaper and flexible in operation.

Embedded system implementation

Introduction:

An embedded system is one kind of a computer system mainly designed to perform several tasks like to access, process, and store and also control the data in various electronics-based systems. Embedded systems are a combination of hardware and software where software is usually known as firmware that is embedded into the hardware. One of its most important characteristics of these systems is, it gives the o/p within the time limits. Embedded systems support to make the work more perfect and convenient. So, we frequently use embedded systems in simple and complex devices too. The applications of embedded systems mainly involve in our real life for several devices like microwave, calculators, TV remote control, home security and neighborhood traffic control systems, etc.

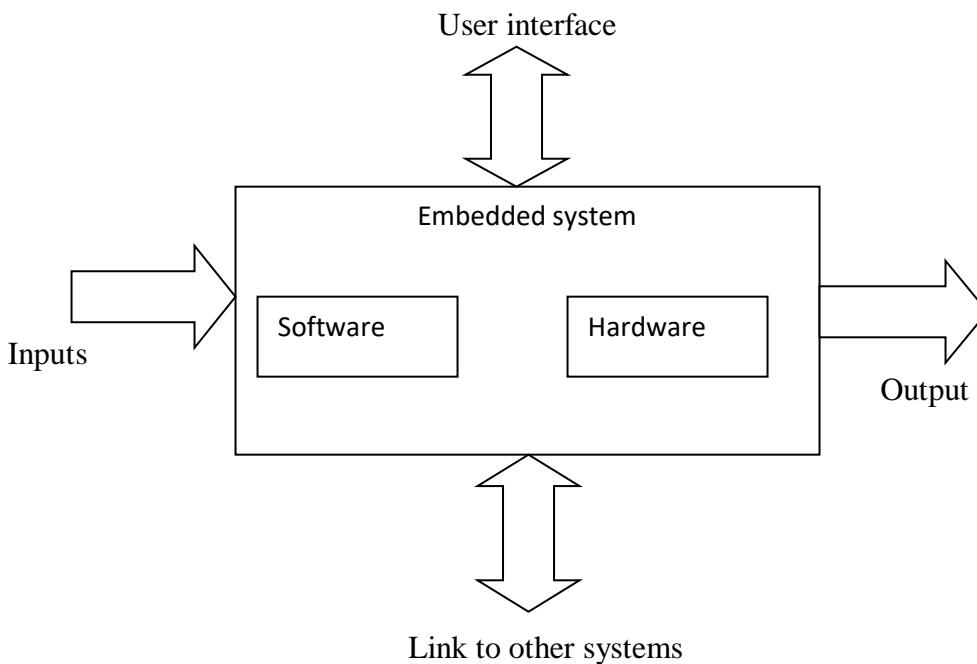
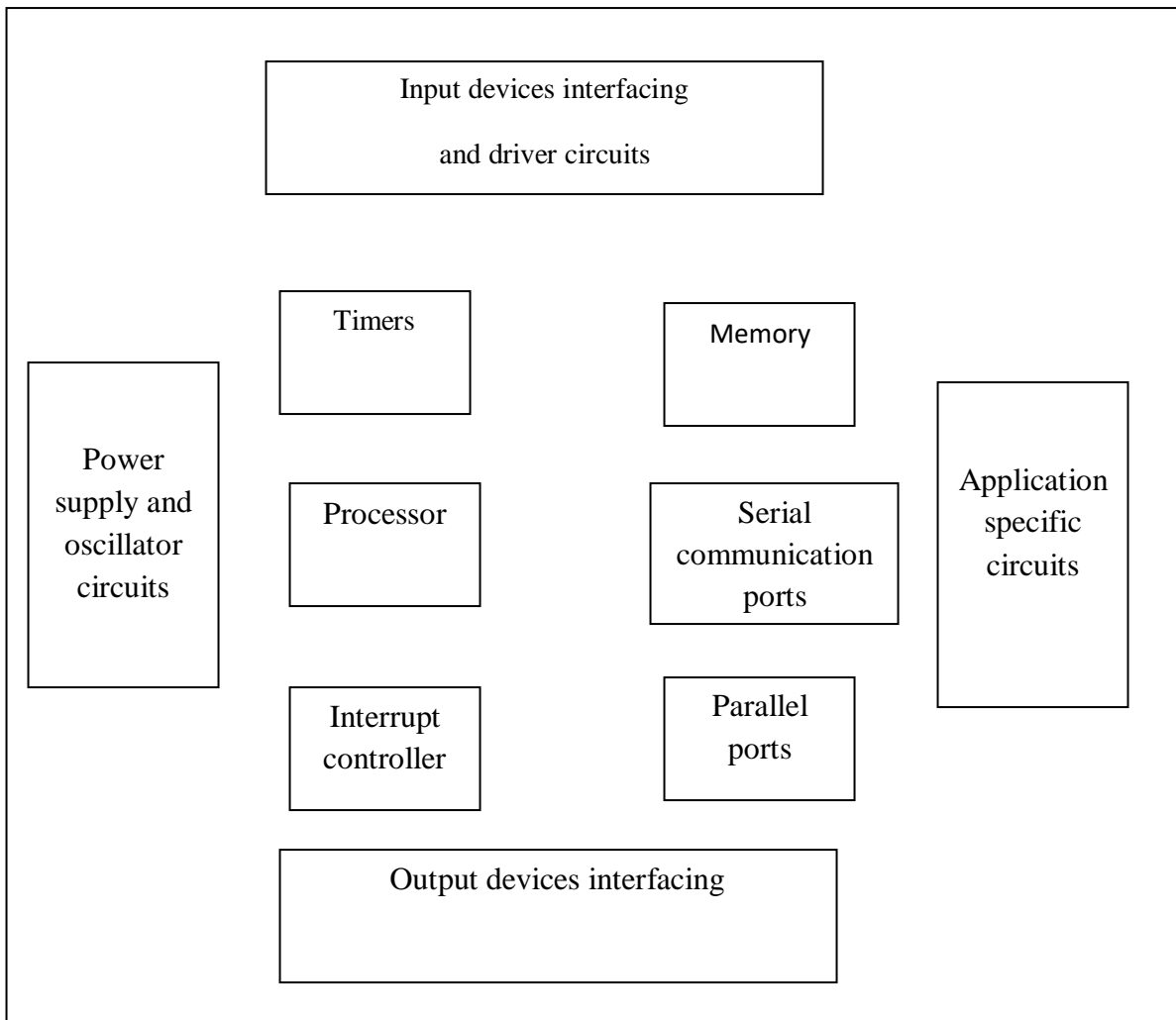


Fig: Overview of embedded system

Embedded system:

Embedded system includes mainly two sections, they are

1. Hardware
2. Software



Block diagram of embedded system

Embedded System Hardware:

As with any electronic system, an embedded system requires a hardware platform on which it performs the operation. Embedded system hardware is built with a microprocessor or microcontroller. The embedded system hardware has elements like input output (I/O) interfaces, user interface, memory and the display. Usually, an embedded system consists of:

- Power Supply
- Processor
- Memory
- Timers
- Serial communication ports

- Output/Output circuits
- System application specific circuits

Embedded systems use different processors for its desired operation. Some of the processors used are

1. Microprocessor
2. Microcontroller
3. Digital signal processor

Microprocessor vs. Microcontroller

Microprocessor

- **CPU** on a chip.
- We can attach required amount of ROM, RAM and I/O ports.
- Expensive due to external peripherals.
- Large in size
- general-purpose

Microcontroller

- **Computer** on a chip
- fixed amount of on-chip ROM, RAM, I/O ports
- Low cost.
- Compact in size.
- Specific –purpose

Embedded System Software:

The embedded system software is written to perform a specific function. It is typically written in a high level format and then compiled down to provide code that can be lodged within a non-volatile memory within the hardware. An embedded system software is designed to keep in view of the three limits:

- Availability of system memory
- Availability of processor's speed

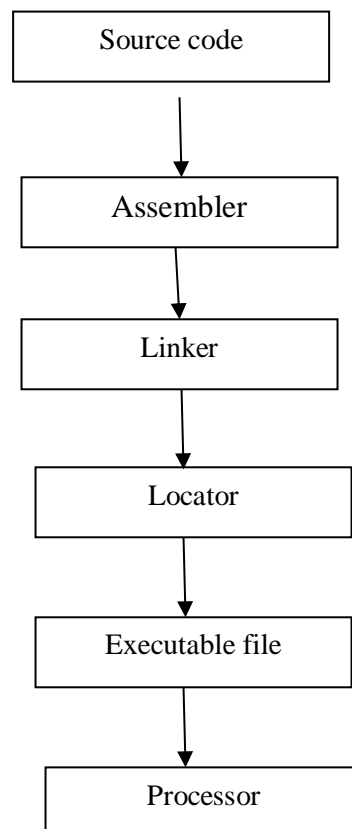
- When the system runs continuously, there is a need to limit power dissipation for events like stop, run and wake up.

Bringing software and hardware together for embedded system: To make software to work with embedded systems we need to bring software and hardware together .for this purpose we need to burn our source code into microprocessor or microcontroller which is a hardware component and which takes care of all operations to be done by embedded system according to our code. Generally we write source codes for embedded systems in assembly language, but the processors run only executable files. The process of converting the source code representation of your embedded software into an executable binary image involves three distinct steps:

1. Each of the source files must be compiled or assembled into an object file.
2. All of the object files that result from the first step must be linked together to produce a single object file, called the re-locatable program.
3. Physical memory addresses must be assigned to the relative offsets within the re-locatable program in a process called relocation.

The result of the final step is a file containing an executable binary image that is ready to run on the embedded system.

Flow of burning source code to processor



Applications:

Embedded systems have different applications. A few select applications of embedded systems are smart cards, telecommunications, satellites, missiles, digital consumer electronics, computer networking, etc.

Embedded Systems in Automobiles

- Motor Control System
- Engine or Body Safety
- Robotics in Assembly Line
- Mobile and E-Com Access

Embedded systems in Telecommunications

- Mobile computing
- Networking
- Wireless Communications

Embedded Systems in Smart Cards

- Banking
- Telephone
- Security Systems

Implementation flow:

Stage 1:

Considering the problems of existing methods and giving solution to that problem by considering the basic requirements for our proposed system

Stage 2:

Considering the hardware requirement for the proposed system

For this we need to select the below components:

1. Microcontroller

2. Inputs for the proposed system (ex: sensors, drivers etc...)
3. Outputs (ex: relays, loads)

Stage 3:

After considering hardware requirements, now we need to check out the software requirements. Based on the microcontroller we select there exists different software for coding, compiling, debugging. we need to write source code for that proposed system based on our requirements and compile, debug the code in that software. After completing all the requirements of software and hardware we need to bring both together to work our system. For this we need to burn our source code into microcontroller, after burning our source code to microcontroller then connect all input and output modules as per our requirement.

LITERATURE SURVEY

Pushkar Singh and Sanghamitra Saikia introduced the design and implementation of an Arduino-Based Smart Irrigation System. The Arduino-based communication has been created to ease the function, application, maintenance and the price. The device is fully automated and reliable. Sensors interact with the website communication system from a large distance in nanoseconds which makes the user more prolific. The entire system is based on the arduino which use inexpensive microcontroller; this system can be applied to large areas for relatively small investment.

Kalaiarasi.D, A.Anusha,D.BerslinJeni and M.Monisha- One of the primary needs for socio- economic development in any nation in the world is the provision of reliable electricity supply systems. The whole world including our country is faced with an issue of lack of back- up power, exhaustion of fossil energy and global warming. This paper describes a novel and developing electrical power generation mechanism by integrating photovoltaic solar energy. Solar energy with Nano-antenna and wind energy-a non-conventional energy sources. Thus have an interrupted power supply with affordable cost without damaging the natural balance. Currently, maintenance of power plant management is provided at the Centre through remote monitoring by using measuring sensor installed at power plant and as regular check-up or repair being performed at site is progressed by site management personnel. But there is a difference in time and accuracy depending on the ability of site management personnel. To overcome this drawback by managing hybrid power system using (INTERNET OF THINGS, IOT).It may be expanded to a technology of recognizing other power generation facility in the future. The result will obtain for both economic and various environmental condition.

EXISTING SYSTEM

In existing we are controlling lights manually which requires human interference. This is the traditional method.

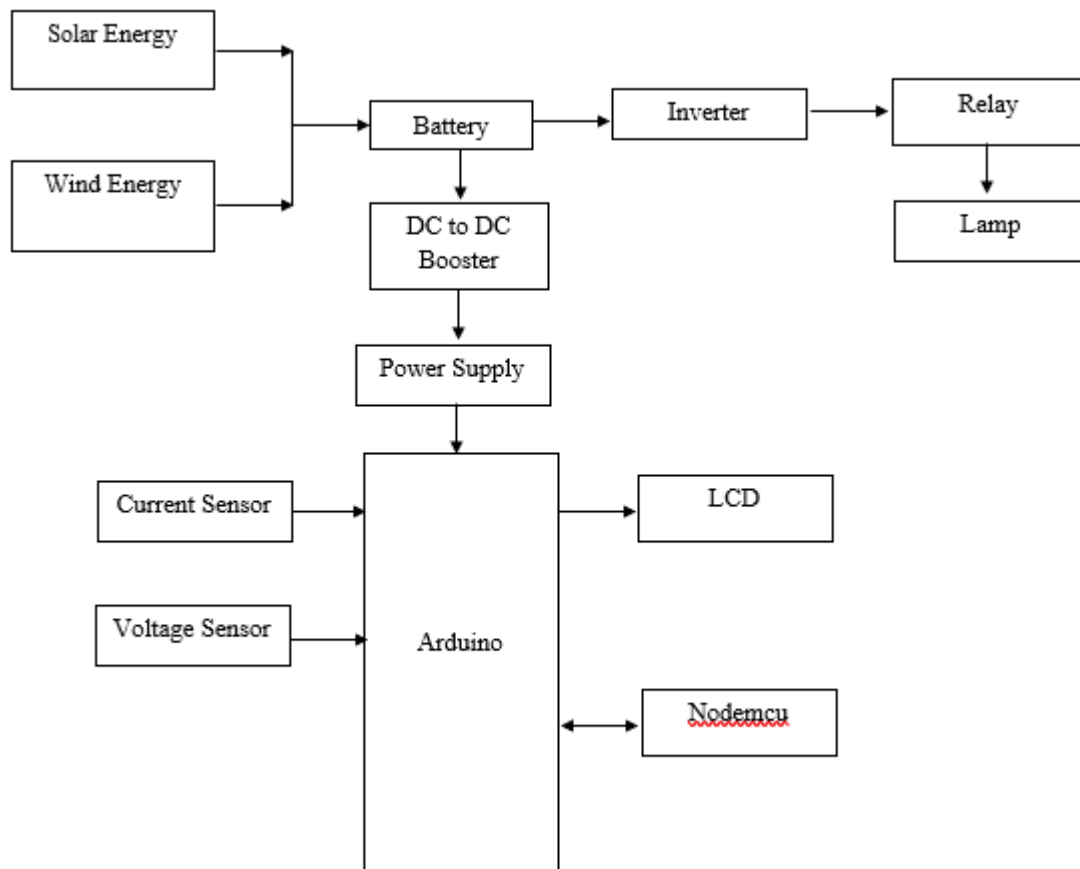
Drawbacks:

- In this electricity consumption is non renewable
- Wastage of electricity will happen if we forget to switch off the lights.

PROPOSED SYSTEM

The structure of the design and development of the proposed system is provided in the following sections. NodeMCU module is used to transmit and receive the electrical data wirelessly, which is collected from internet through designed website and the control system. The NodeMCU transmitter is interfaced with various sensing devices and reliable data reception at a receiver side of NodeMCU module. The NodeMCU receiver has been interfaced through router which is connected to the internet. The Load can be monitored and controlled remotely.

Block diagram:



HARDWARE REQUIREMENTS

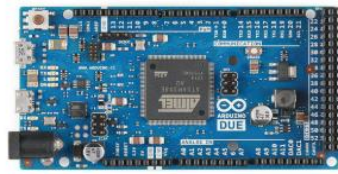
Arduino:

Arduino Uno is a very valuable addition in the electronics that consists of USB interface, 14 digital I/O pins, 6 analog pins, and Atmega328 microcontroller. It also supports serial communication using Tx and Rx pins.

There are many versions of Arduino boards introduced in the market like Arduino Uno, Arduino Due, Arduino Leonardo, Arduino Mega, however, most common versions are Arduino Uno and Arduino Mega. If you are planning to create a project relating to digital electronics, embedded system, robotics, or IoT, then using Arduino Uno would be the best, easy and most economical option.



Arduino Uno



Arduino Due



Arduino Leonardo

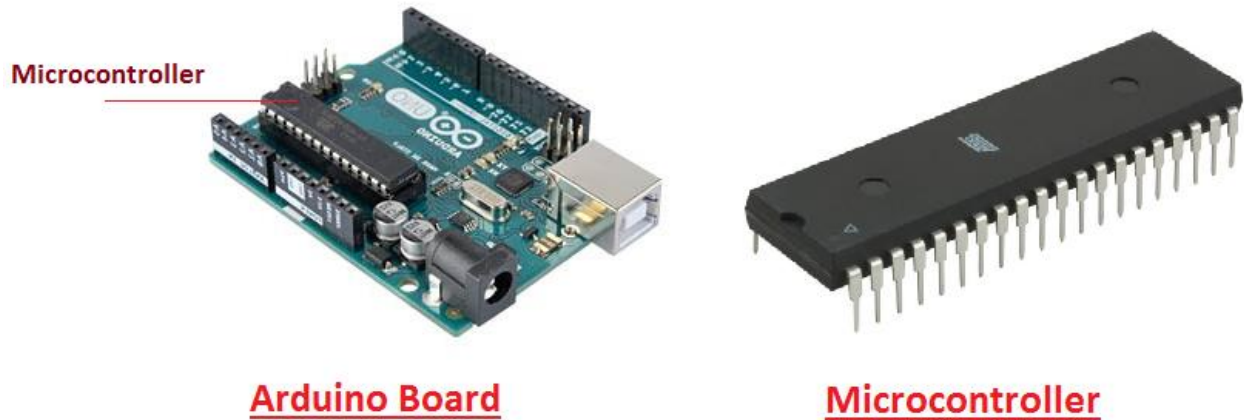


Arduino Mega

It is an open-source platform, means the boards and software are readily available and anyone can modify and optimize the boards for better functionality.

The software used for Arduino devices is called IDE (Integrated Development Environment) which is free to use and required some basic skills to learn it. It can be programmed using C and C++ language.

Some people get confused between **Microcontroller and Arduino**. While former is just an on system 40 pin chip that comes with a built-in microprocessor and later is a board that comes with the microcontroller in the base of the board, bootloader and allows easy access to input-output pins and makes uploading or burning of the program very easy.

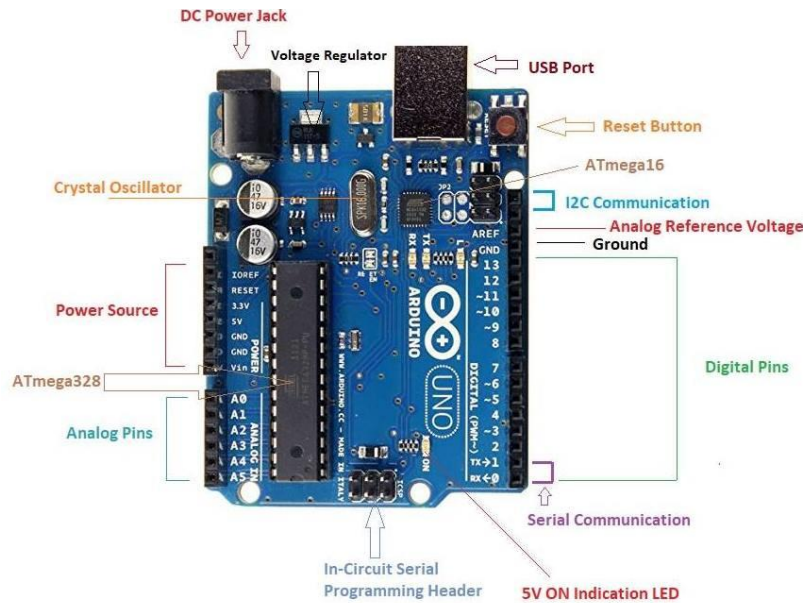


While learning microcontroller requires some expertise and skills.

Nevertheless, we can say every Arduino is basically a microcontroller but not every microcontroller is an Arduino.

Introduction to Arduino

- **Arduino Uno** is a microcontroller board developed by Arduino.cc which is an open-source electronics platform mainly based on AVR microcontroller Atmega328.
- First Arduino project was started in Interaction Design Institute Ivrea in 2003 by David Cuartielles and Massimo Banzi with the intention of providing a cheap and flexible way to students and professional for controlling a number of devices in the real world.
- The current version of Arduino Uno comes with USB interface, 6 analog input pins, 14 I/O digital ports that are used to connect with external electronic circuits. Out of 14 I/O ports, 6 pins can be used for PWM output.
- It allows the designers to control and sense the external electronic devices in the real world



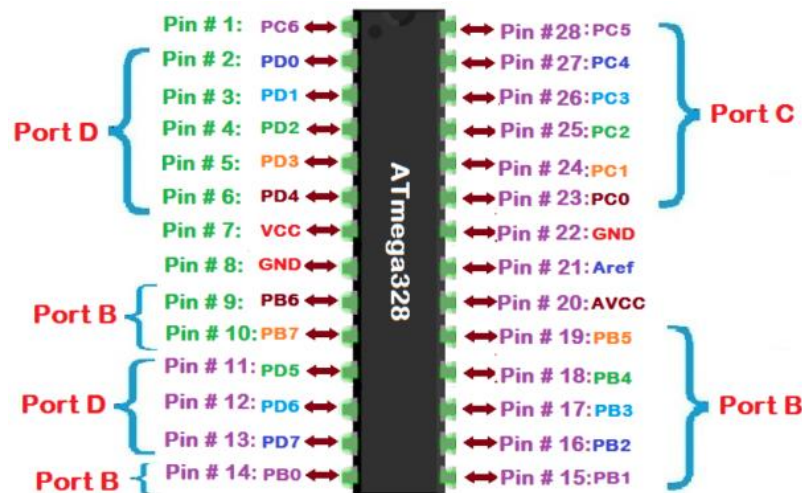
Arduino UNO

- This board comes with all the features required to run the controller and can be directly connected to the computer through USB cable that is used to transfer the code to the controller using IDE (Integrated Development Environment) software, mainly developed to program Arduino. IDE is equally compatible with Windows, MAC or Linux Systems, however, Windows is preferable to use. Programming languages like C and C++ are used in IDE.
- Apart from USB, battery or AC to DC adopter can also be used to power the board.
- Arduino Uno boards are quite similar to other boards in Arduino family in terms of use and functionality, however, Uno boards don't come with FTDI USB to Serial driver chip.
- There are many versions of Uno boards available, however, Arduino Nano V3 and Arduino Uno are the most official versions that come with ATmega328 8-bit AVR Atmel microcontroller where RAM memory is 32KB.
- When nature and functionality of the task go complex, Micro SD card can be added in the boards to make them store more information.

Features of Arduino

- Arduino Uno comes with USB interface i.e. USB port is added on the board to develop serial communication with the computer.

- Atmega328 microcontroller is placed on the board that comes with a number of features like timers, counters, interrupts, PWM, CPU, I/O pins and based on a 16MHz clock that helps in producing more frequency and number of instructions per cycle.



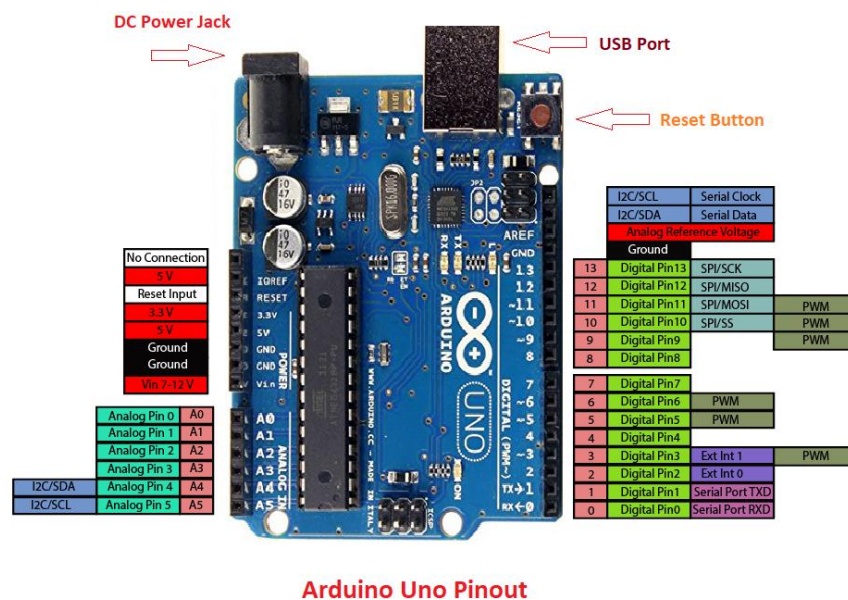
Atmega328 Microcontroller

- It is an open source platform where anyone can modify and optimize the board based on the number of instructions and task they want to achieve.
- This board comes with a built-in regulation feature which keeps the voltage under control when the device is connected to the external device.
- Reset pin is added in the board that reset the whole board and takes the running program in the initial stage. This pin is useful when board hangs up in the middle of the running program; pushing this pin will clear everything up in the program and starts the program right from the beginning.
- There are 14 I/O digital and 6 analog pins incorporated in the board that allows the external connection with any circuit with the board. These pins provide the flexibility and ease of use to the external devices that can be connected through these pins. There is no hard and fast interface required to connect the devices to the board. Simply plug the external device into the pins of the board that are laid out on the board in the form of the header.
- The 6 analog pins are marked as A0 to A5 and come with a resolution of 10bits. These pins measure from 0 to 5V, however, they can be configured to the high range using analogReference() function and AREF pin.
- 13KB of flash memory is used to store the number of instructions in the form of code.

- Only 5 V is required to turn the board on, which can be achieved directly using USB port or external adapter, however, it can support external power source up to 12 V which can be regulated and limit to 5 V or 3.3 V based on the requirement of the project.

Arduino Pinout

- Arduino Uno is based on AVR microcontroller called Atmega328. This controller comes with 2KB SRAM, 32KB of flash memory, 1KB of EEPROM. Arduino Board comes with 14 digital pins and 6 analog pins. ON-chip ADC is used to sample these pins. A 16 MHz frequency crystal oscillator is equipped on the board. Following figure shows the pinout of the Arduino Uno Board



Pin Description: There are several I/O digital and analog pins placed on the board which operates at 5V. These pins come with standard operating ratings ranging between 20mA to 40mA. Internal pull-up resistors are used in the board that limits the current exceeding from the given operating conditions. However, too much increase in current makes these resistors useless and damages the device.

LED. Arduino Uno comes with built-in LED which is connected through pin 13. Providing HIGH value to the pin will turn it ON and LOW will turn it OFF.

Vin. It is the input voltage provided to the Arduino Board. It is different than 5 V supplied through a USB port. This pin is used to supply voltage. If a voltage is provided through power jack, it can be accessed through this pin.

5V. This board comes with the ability to provide voltage regulation. 5V pin is used to provide output regulated voltage. The board is powered up using three ways i.e. USB, Vin pin of the board or DC power jack.

USB supports voltage around 5V while Vin and Power Jack support a voltage ranges between 7V to 20V. It is recommended to operate the board on 5V. It is important to note that, if a voltage is supplied through 5V or 3.3V pins, they result in bypassing the voltage regulation that can damage the board if voltage surpasses from its limit.

GND. These are ground pins. More than one ground pins are provided on the board which can be used as per requirement.

Reset. This pin is incorporated on the board which resets the program running on the board. Instead of physical reset on the board, IDE comes with a feature of resetting the board through programming.

IOREF. This pin is very useful for providing voltage reference to the board. A shield is used to read the voltage across this pin which then select the proper power source.

PWM. PWM is provided by 3, 5, 6,9,10, 11pins. These pins are configured to provide 8-bit output PWM.

SPI. It is known as Serial Peripheral Interface. Four pins 10(SS), 11(MOSI), 12(MISO), 13(SCK) provide SPI communication with the help of SPI library.

AREF. It is called Analog Reference. This pin is used for providing a reference voltage to the analog inputs.

TWI. It is called Two-wire Interface. TWI communication is accessed through Wire Library. A4 and A5 pins are used for this purpose.

Serial Communication. Serial communication is carried out through two pins called Pin 0 (Rx) and Pin 1 (Tx).

Rx pin is used to receive data while Tx pin is used to transmit data.

External Interrupts. Pin 2 and 3 are used for providing external interrupts. An interrupt is called by providing LOW or changing value.

Arduino Uno Technical Specifications

Microcontroller	<u>ATmega328P</u> – 8 bit AVR family microcontroller
Operating Voltage	5V

Recommended Input Voltage	7-12V
Input Voltage Limits	6-20V
Analog Input Pins	6 (A0 – A5)
Digital I/O Pins	14 (Out of which 6 provide PWM output)
DC Current on I/O Pins	40 mA
DC Current on 3.3V Pin	50 mA
Flash Memory	32 KB (0.5 KB is used for Bootloader)
SRAM	2 KB
EEPROM	1 KB
Frequency (Clock Speed)	16 MHz

Communication and Programming:

Arduino Uno comes with an ability of interfacing with other other Arduino boards, microcontrollers and computer. The Atmega328 placed on the board provides serial communication using pins like Rx and Tx.

The Atmega16U2 incorporated on the board provides a pathway for serial communication using USB com drivers. Serial monitor is provided on the IDE software which is used to send or receive text data from the board. If LEDs placed on the Rx and Tx pins will flash, they indicate the transmission of data.

Arduino Uno is programmed using Arduino Software which a cross-platform application called IDE is written in Java. The AVR microcontroller Atmega328 laid out on the base comes with built-in boot loader that sets you free from using a separate burner to upload the program on the board.



Applications:

Arduino Uno comes with a wide range of applications. A larger number of people are using Arduino boards for developing sensors and instruments that are used in scientific research. Following are some main applications of the board.

- Embedded System
- Security and Defense System
- Digital Electronics and Robotics
- Parking Lot Counter
- Weighing Machines
- Traffic Light Count Down Timer
- Medical Instrument
- Emergency Light for Railways
- Home Automation
- Industrial Automation

There are a lot of other microcontrollers available in the market that are more powerful and cheap as compared to Arduino board. So, why you prefer Arduino Uno?

Actually, Arduino comes with a big community that is developing and sharing the knowledge with a wide range of audience. Quick support is available pertaining to technical aspects of any electronic project. When you decide Arduino board over other controllers, you don't need to arrange extra peripherals and devices as most of

the functions are readily available on the board that makes your project economical in nature and free from a lot of technical expertise.

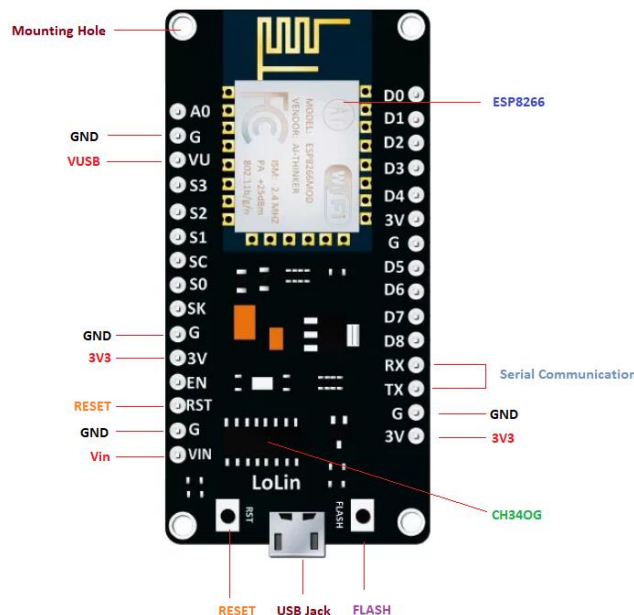
NodeMCU:

Introduction to NodeMCU

NodeMCU is an open-source firmware and development kit that plays a vital role in designing your own IoT product using a few Lua script lines. Multiple GPIO pins on the board allow you to connect the board with other peripherals and are capable of generating PWM, I2C, SPI, and UART serial communications.

- The interface of the module is mainly divided into two parts including both Firmware and Hardware where former runs on the ESP8266 Wi-Fi SoC and later is based on the ESP-12 module.

The firmware is based on Lua – A scripting language that is easy to learn, giving a simple programming environment layered with a fast scripting language that connects you with a well-known developer community.



And open source firmware gives you the flexibility to edit, modify and rebuilt the existing module and keep changing the entire interface until you succeed in optimizing the module as per your requirements.

- USB to UART converter is added on the module that helps in converting USB data to UART data which mainly understands the language of serial communication.

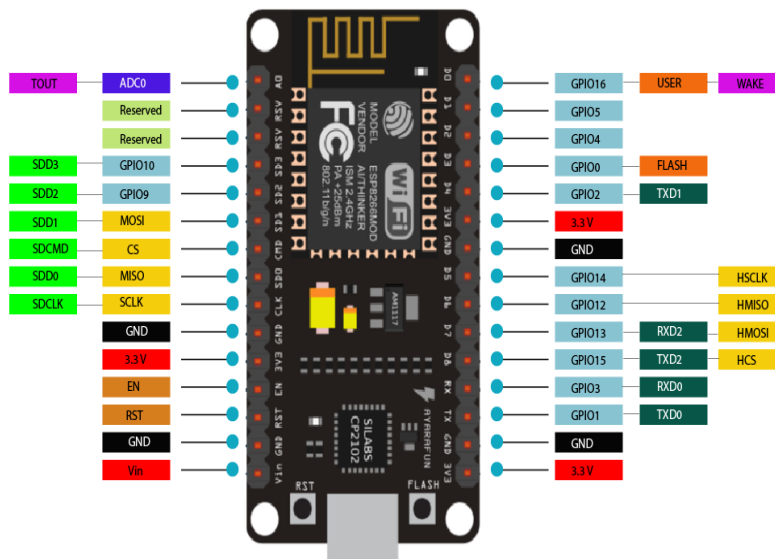
Instead of the regular USB port, MicroUSB port is included in the module that connects it with the computer for dual purposes: programming and powering up the board.

- The board incorporates status LED that blinks and turns off immediately, giving you the current status of the module if it is running properly when connected with the computer.

The ability of module to establish a flawless WiFi connection between two channels makes it an ideal choice for incorporating it with other embedded devices like Raspberry Pi.

NodeMCU Pinout:

NodeMCU comes with a number of GPIO Pins. Following figure shows the Pinout of the board.



- There is a candid difference between VIN and VU where former is the regulated voltage that may stand somewhere between 7 to 12 V while later is the power voltage for USB that must be kept around 5 V.

Features:

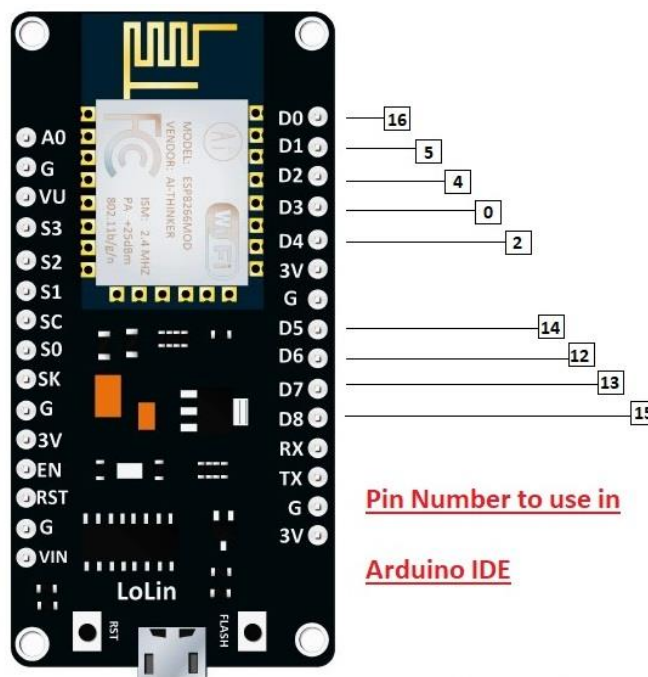
- Open-source
- Arduino-like hardware

- Status LED
- MicroUSB port
- Reset/Flash buttons
- Interactive and Programmable
- Low cost
- ESP8266 with inbuilt wifi
- USB to UART converter
- GPIO pins

As mentioned above, a cable supporting micro USB port is used to connect the board. As you connect the board with a computer, LED will flash. You may need some drivers to be installed on your computer if it fails to detect the NodeMCU board. You can download the driver from [this](#) page.

Note: We use Arduino IDE software for programming this module. It is important to note that the pin configuration appearing on the board is different from the configuration we use to program the board on the software i.e. when we write code for targeting pin 16 on the Arduino IDE, it will actually help in laying out the communication with the D0 pin on the module.

Following figure shows the pin configuration to use in Arduino IDE.



How to Power NodeMCU

You can see from the pinout image above, there are five ground pins and three 3V3 pins on the board. The board can be powered up using the following three ways.

USB Power. It proves to an ideal choice for loading programs unless the project you aim to design requires separate interface i.e. disconnected from the computer.

Provide 3.3V. This is another great option to power up the module. If you have your own off-board regulator, you can generate an instant power source for your development kit.

Power Vin. This is a voltage regulator that comes with the ability to support up to 800 mA. It can handle somewhere between 7 to 12 V. You cannot power the devices operating at 3.3 V, as this regulator unable to generate as low as 3.3V.

Applications:

NodeMCU V3 is mainly used in the WiFi Applications which most of the other embedded modules fail to process unless incorporated with some external WiFi protocol. Following are some major applications used for NodeMCU V3.

- Internet Smoked Alarm
- VR Tracker
- Octopod
- Serial Port Monitor
- ESP Lamp
- Incubator Controller
- IoT home automation
- Security Alarms

LCD:

LCD (Liquid Crystal Display) is the innovation utilized in scratch pad shows and other littler PCs. Like innovation for light-producing diode (LED) and gas-plasma, LCDs permit presentations to be a lot more slender

than innovation for cathode beam tube (CRT). LCDs expend considerably less power than LED shows and gas shows since they work as opposed to emanating it on the guideline of blocking light.

A LCD is either made with a uninvolved lattice or a showcase network for dynamic framework show. Likewise alluded to as a meager film transistor (TFT) show is the dynamic framework LCD. The uninvolved LCD lattice has a matrix of conductors at every crossing point of the network with pixels. Two conductors on the lattice send a current to control the light for any pixel. A functioning framework has a transistor situated at every pixel crossing point, requiring less current to control the luminance of a pixel.

Some aloof network LCD's have double filtering, which implies they examine the matrix twice with current in the meantime as the first innovation took one sweep. Dynamic lattice, be that as it may, is as yet a higher innovation.

A 16x2 LCD show is an essential module that is generally utilized in various gadgets and circuits. These modules more than seven sections and other multi fragment LEDs are liked. The reasons being: LCDs are affordable; effectively programmable; have no restriction of showing exceptional and even custom characters (not at all like in seven fragments), movements, etc.

A 16x2 LCD implies 16 characters can be shown per line and 2 such lines exist. Each character is shown in a lattice of 5x7 pixels in this LCD. There are two registers in this LCD, in particular Command and Data.

The directions given to the LCD are put away by the order register. An order is a direction given to LCD to play out a predefined assignment, for example, introducing it, clearing its screen, setting the situation of the cursor, controlling presentation, and so forth. The information register will store the information that will be shown on the LCD. The information is the character's ASCII incentive to show on the LCD.

Data/Signals/Execution of LCD

Now that was all about the signals and the hardware. Let us come to data, signals and execution.

Two types of signals are accepted by LCD, one is data and one is control. The LCD module recognizes these signals from the RS pin status. By pulling the R / W pin high, data can now also be read from the LCD display. Once the E pin has been pulsed, the LCD display reads and executes data at the falling edge of the pulse, the same for the transmission case.

It takes 39-43 μ S for the LCD display to place a character or execute a command. It takes 1.53ms to 1.64ms except for clearing display and searching for cursor to the home position.

Any attempt to send data before this interval may result in failure in some devices to read data or execute the current data. Some devices compensate for the speed by storing some temporary registers with incoming data.

There are two RAMs for LCD displays, namely DDRAM and CGRAM. DDRAM registers the position in which the character would be displayed in the ASCII chart. Each DDRAM byte represents every single position on the display of the LCD.

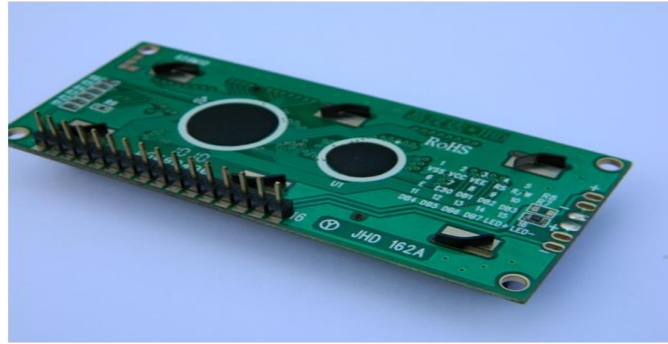
The DDRAM information is read by the LCD controller and displayed on the LCD screen. CGRAM enables users to define their personalized characters. Address space is reserved for users for the first 16 ASCII characters.

Users can easily display their custom characters on the LCD screen after CGRAM has been set up to display characters.

Images of LCD Display:-

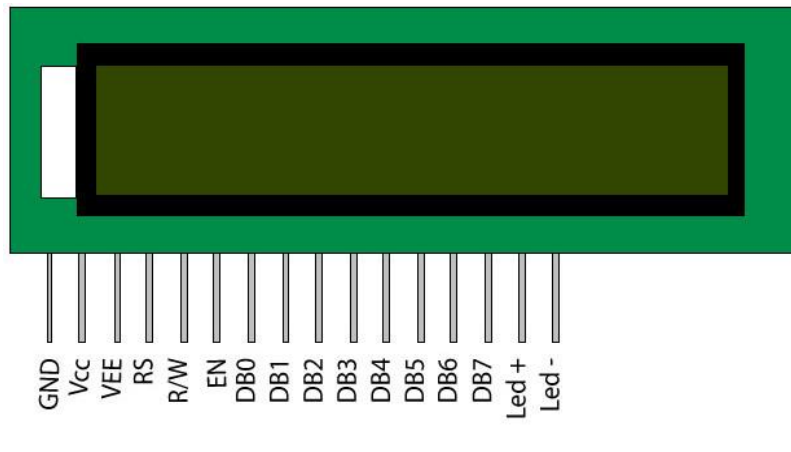


LCD – Front View



LCD – Back View

Pin Diagram:



Pin Description:

Pin No	Function	Name
1	Ground (0V)	Ground
2	Supply voltage; 5V (4.7V – 5.3V)	Vcc
3	Contrast adjustment; through a variable resistor	VEE
4	Selects command register when low; and data register when high	Register Select
5	Low to write to the register; High to read from the register	Read/write
6	Sends data to data pins when a high to low pulse is given	Enable
7		DB0
8		DB1
9		DB2
10		DB3
11		DB4
12		DB5
13		DB6
14		DB7

15	Backlight V _{CC} (5V)	Led+
16	Backlight Ground (0V)	Led-

RS (Register select)

A 16X2 LCD has two order and information registers. The determination of the register is utilized to change starting with one register then onto the next. RS=0 for the register of directions, while RS=1 for the register of information.

Command Register

The guidelines given to the LCD are put away by the direction register. An order is a direction given to LCD to play out a predefined assignment, for example, instating it, clearing its screen, setting the situation of the cursor, controlling showcase, and so on. Order preparing happens in the direction register.

Data Register:

The information register will store the information that will be shown on the LCD. The information is the character's ASCII incentive to show on the LCD. It goes to the information register and is prepared there when we send information to the LCD. While choosing RS=1, the information register.

Read and Write Mode of LCD:

As stated, the LCD itself comprises of an interface IC. This interface IC can be perused or composed by the MCU. A large portion of the occasions we're simply going to keep in touch with the IC since perusing will make it increasingly perplexing and situations like that are exceptionally uncommon. Information such as cursor position, status completion interrupts, etc. can be read if necessary.

LCD Commands:

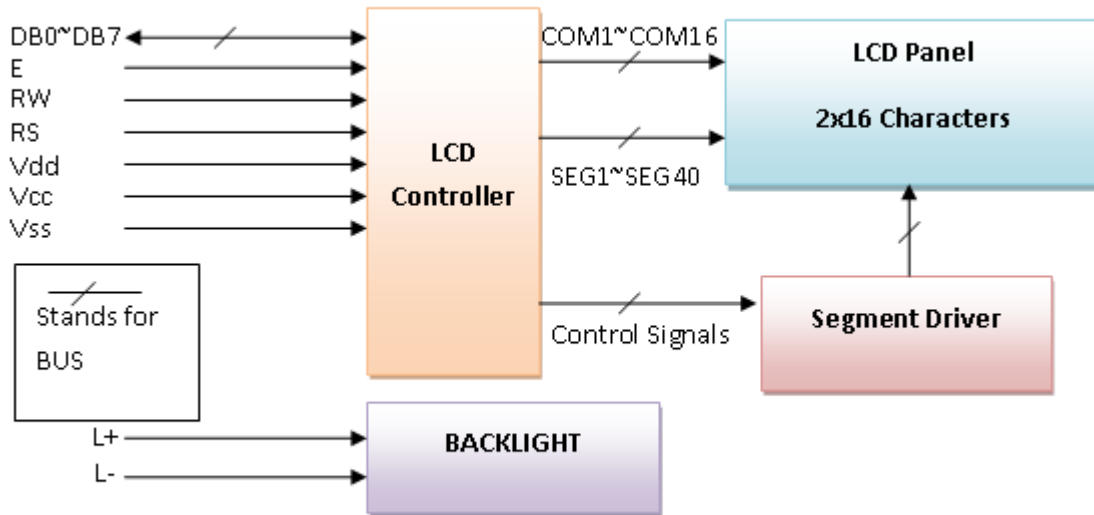
There are some preset commands in the LCD that we need to send to the LCD via some microcontroller. The following are some important command instructions:

Sr. No.	Hex Code	Command to LCD instruction Register
1	01	Clear display screen

2	02	Return home
3	04	Decrement cursor (shift cursor to left)
4	06	Increment cursor (shift cursor to right)
5	05	Shift display right
6	07	Shift display left
7	08	Display off, cursor off
8	0A	Display off, cursor on
9	0C	Display on, cursor off
10	0E	Display on, cursor blinking
11	0F	Display on, cursor blinking
12	10	Shift cursor position to left
13	14	Shift cursor position to right
14	18	Shift the entire display to the left
15	1C	Shift the entire display to the right
16	80	Force cursor to beginning (1st line)
17	C0	Force cursor to beginning (2nd line)
18	38	2 lines and 5×7 matrix

Command codes for LCD

Block Diagram of LCD Display:-



Control and display commands

Instruction n	Instruction Code										Instruction Description	Code	Execution time
	R/S	R/W	DB7	DB6	DB5	DB4	DB3	DB2	DB1	DB0			
Read Data From RAM	1	1	D7	D6	D5	D4	D3	D2	D1	D0	Read data from internal RAM		1.53-1.64ms
Write data to RAM	1	0	D7	D6	D5	D4	D3	D2	D1	D0	Write data into internal RAM (DDRAM/CGRAM)		1.53-1.64ms
Busy flag & Address	0	1	BF	AC6	AC5	AC4	AC3	AC2	AC1	AC0	Busy flag (BF: 1→ LCD Busy) and contents of address counter in bits AC6-AC0.		39 μs
Set DDRAM	0	0	1	AC	AC	AC	AC	AC	AC	AC	Set DDRAM address in address		39 μs

Address				6	5	4	3	2	1	0	counter.	
Set CGRAM Address	0	0	0	1	AC 5	AC 4	AC 3	AC 2	AC 1	AC 0	Set CGRAM Address in address counter.	39 μ s
Function Set	0	0	0	0	1	DL	N	F	X	X	Set interface data length (DL: 4bit/8bit), Numbers of display line (N: 1-line/2-line) display font type (F:0 \rightarrow 5 \times 8 dots, F:1 \rightarrow 5 \times 11 dots)	39 μ s
Cursor or Display Shift	0	0	0	0	0	1	S/C	R/L	X	X	Set cursor moving and display shift control bit, and the direction without changing DDRAM data	39 μ s
Display & Cursor On/Off	0	0	0	0	0	0	1	D	C	B	Set Display(D),Cursor(C) and cursor blink(b) on/off control	39 μ s
Entry Mode Set	0	0	0	0	0	0	0	1	I/D	SH	Assign cursor moving direction and enable shift entire display.	0 μ s
Return Home	0	0	0	0	0	0	0	0	1	X	Set DDRAM Address to "00H" from AC and return cursor to its original position if shifted.	43 μ s
Clear Display	0	0	0	0	0	0	0	0	0	1	Write "20H" to DDRAM and set DDRAM Address to "00H" from AC	43 μ s

4-bit and 8-bit Mode of LCD:

The LCD can work in two striking modes, the 4-bit mode and the 8-bit mode. We send the information snack through snack in 4 bit mode, first upper chomp, by then lower snack. For those of you who don't have the

foggiest idea what a goody is: a chomp is a four-piece gathering, so a byte's lower four bits (D0-D3) are the lower snack, while a byte's upper four bits (D4-D7) are the higher snack. This enables us to send 8 bit data. This connects with us to send 8 bit data. Whereas in 8 bit mode we can send the 8-bit information truly in one stroke since we utilize all the 8 information lines. You need to get it now; yes 8-bit mode is quicker and immaculate than 4-bit mode. In any case, the fundamental shortcoming is that it needs 8 microcontroller-related information lines. This will result in our MCU coming up short on I/O pins, so 4-bit mode is extensively utilized. To set these modes, no control pins are used.

Solar panel:

Photovoltaic solar panels absorb sunlight as a source of energy to generate electricity. A photovoltaic (PV) module is a packaged, connected assembly of typically 6x10 photovoltaic solar cells. Photovoltaic modules constitute the photovoltaic array of a photovoltaic system that generates and supplies solar electricity in commercial and residential applications.

The most common application of solar energy collection outside agriculture is solar water heating systems.

Theory and construction

Photovoltaic modules use light energy (photons) from the Sun to generate electricity through the photovoltaic effect. The majority of modules use wafer-based crystalline silicon cells or thin-film cells. The structural (load carrying) member of a module can either be the top layer or the back layer. Cells must also be protected from mechanical damage and moisture. Most modules are rigid, but semi-flexible ones based on thin-film cells are also available. The cells must be connected electrically in series, one to another.

A PV junction box is attached to the back of the solar panel and it is its output interface. Externally, most of photovoltaic modules use MC4 connectors type to facilitate easy weatherproof connections to the rest of the system. Also, USB power interface can be used.

Module electrical connections are made in series to achieve a desired output voltage or in parallel to provide a desired current capability (amperes). The conducting wires that take the current off the modules may contain silver, copper or other non-magnetic conductive transition metals. Bypass diodes may be incorporated or used externally, in case of partial module shading, to maximize the output of module sections still illuminated.

Some special solar PV modules include concentrators in which light is focused by lenses or mirrors onto smaller cells. This enables the use of cells with a high cost per unit area (such as gallium arsenide) in a cost-effective way.

Solar panels also use metal frames consisting of racking components, brackets, reflector shapes, and troughs to better support the panel structure.

Technology:

Most solar modules are currently produced from crystalline silicon (c-Si) solar cells made of multicrystalline and monocrystalline silicon. In 2013, crystalline silicon accounted for more than 90 percent of worldwide PV production, while the rest of the overall market is made up of thin-film technologies using cadmium telluride, CIGS and amorphous silicon^[16]

Emerging, third generation solar technologies use advanced thin-film cells. They produce a relatively high-efficiency conversion for the low cost compared to other solar technologies. Also, high-cost, high-efficiency, and close-packed rectangular multi-junction (MJ) cells are preferably used in solar panels on spacecraft, as they offer the highest ratio of generated power per kilogram lifted into space. MJ-cells are compound semiconductors and made of gallium arsenide (GaAs) and other semiconductor materials. Another emerging PV technology using MJ-cells is concentrator photovoltaics (CPV).

Thin film

In rigid thin-film modules, the cell and the module are manufactured in the same production line. The cell is created on a glass substrate or superstrate, and the electrical connections are created *in situ*, a so-called "monolithic integration". The substrate or superstrate is laminated with an encapsulant to a front or back sheet, usually another sheet of glass. The main cell technologies in this category are CdTe, or a-Si, or a-Si+uc-Si tandem, or CIGS (or variant). Amorphous silicon has a sunlight conversion rate of 6–12%

Flexible thin film cells and modules are created on the same production line by depositing the photoactive layer and other necessary layers on a flexible substrate. If the substrate is an insulator (e.g. polyester or polyimide film) then monolithic integration can be used. If it is a conductor then another technique for electrical connection must be used. The cells are assembled into modules by laminating them to a transparent colourless fluoropolymer on the front side (typically ETFE or FEP) and a polymer suitable for bonding to the final substrate on the other side.



Wind Energy:

Wind energy is the kinetic energy associated with the movement of atmospheric air. It has been used for hundreds of years for sailing, grinding grain and for irrigation. Wind energy systems convert this kinetic energy to more useful forms of power. Wind energy systems for irrigation and milling have been in use since ancient times and at the beginning of the 20th century it is being used to generate electric power. Windmills for water pumping have been installed in many countries particularly in the rural areas. Wind turbines transform the energy in the wind into mechanical power, which can then be used directly for grinding etc. or further converting to electric power to generate electricity. Wind turbines can be used singly or in clusters called 'wind farms.



There is a growing awareness that renewable energy such as photovoltaic system and Wind power have an important role to play in order to save the situation. Hybrid power system consist of a combination of renewable energy source such as wind generators, solar etc of charge batteries and provide power to meet the energy demand, considering the local geography and other details of the place of installation. These types of systems are not connected to the main utility grid. They are also used in stand-alone applications and operate independently and reliably. The best application for these type of systems are in remote places, such as rural villages, in telecommunications etc. The importance of hybrid systems has grown as they appear to be the right solution for a clean and distributed energy production.

SOFTWARE REQUIREMENTS

Arduino IDE:

Arduino IDE where IDE stands for Integrated Development Environment – An official software introduced by Arduino.cc, that is mainly used for writing, compiling and uploading the code in the Arduino Device. Almost all Arduino modules are compatible with this software that is an open source and is readily available to install and start compiling the code on the go.

Introduction to Arduino IDE:

- Arduino IDE is an open source software that is mainly used for writing and compiling the code into the Arduino Module.
- It is an official Arduino software, making code compilation too easy that even a common person with no prior technical knowledge can get their feet wet with the learning process.
- It is easily available for operating systems like MAC, Windows, and Linux and runs on the Java Platform that comes with inbuilt functions and commands that play a vital role for debugging, editing and compiling the code in the environment.
- A range of Arduino modules available including Arduino Uno, Arduino Mega, Arduino Leonardo, Arduino Micro and many more.

- Each of them contains a microcontroller on the board that is actually programmed and accepts the information in the form of code.
- The main code, also known as a sketch, created on the IDE platform will ultimately generate a Hex File which is then transferred and uploaded in the controller on the board.
- The IDE environment mainly contains two basic parts: Editor and Compiler where former is used for writing the required code and later is used for compiling and uploading the code into the given Arduino Module.
- This environment supports both C and C++ languages.

How to install Arduino IDE:

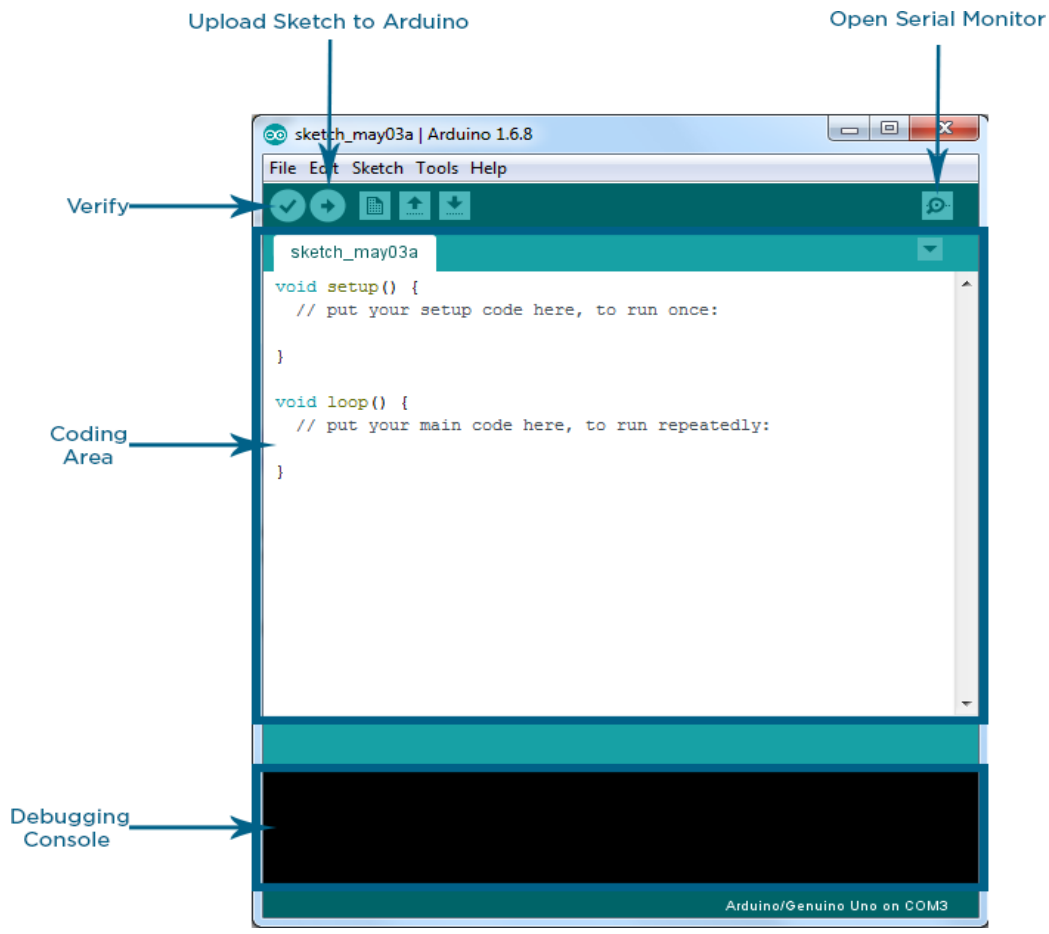
You can download the Software from [Arduino](#) main website. As I said earlier, the software is available for common operating systems like Linux, Windows, and MAX, so make sure you are downloading the correct software version that is easily compatible with your operating system.

- If you aim to download Windows app version, make sure you have Windows 8.1 or Windows 10, as app version is not compatible with Windows 7 or older version of this operating system.

The IDE environment is mainly distributed into three sections

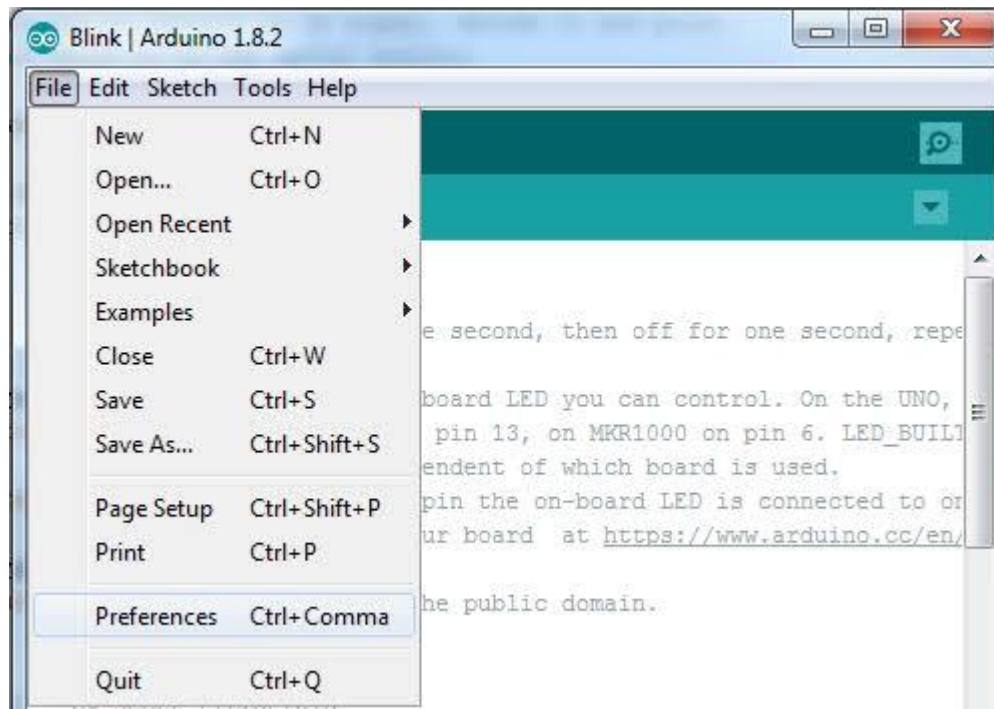
- **1. Menu Bar**
- **2. Text Editor**
- **3. Output Pane**

As you download and open the IDE software, it will appear like an image below.

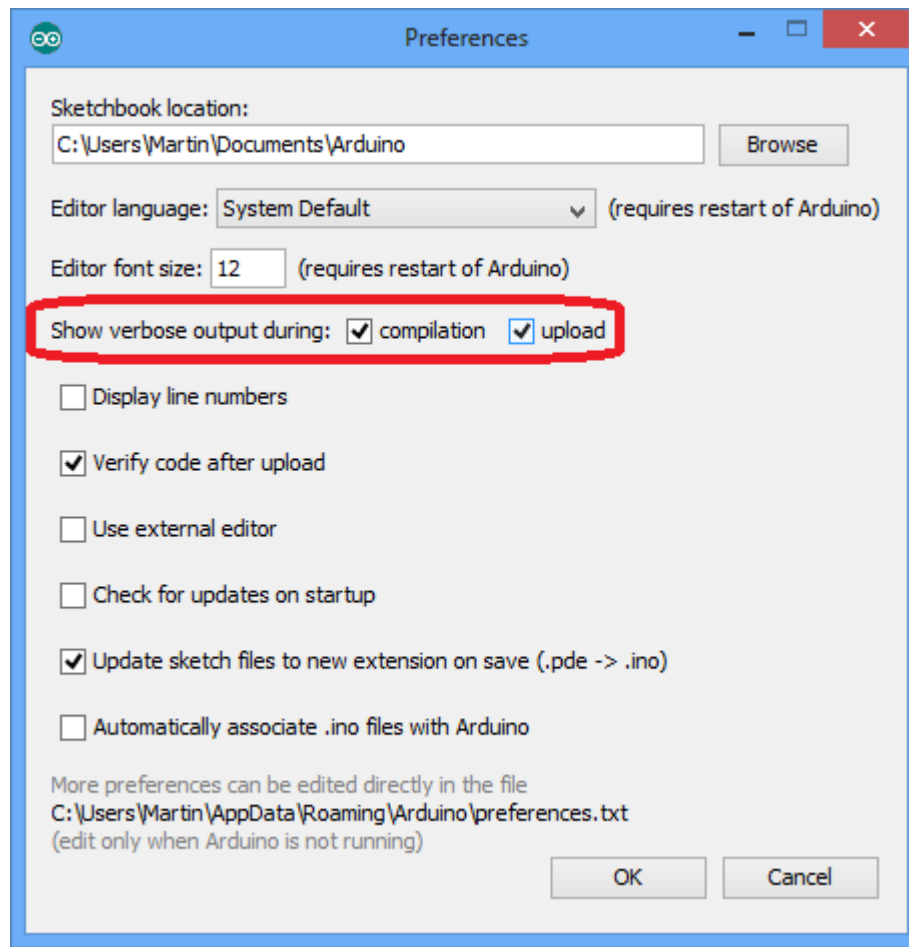


The bar appearing on the top is called **Menu Bar** that comes with five different options as follow

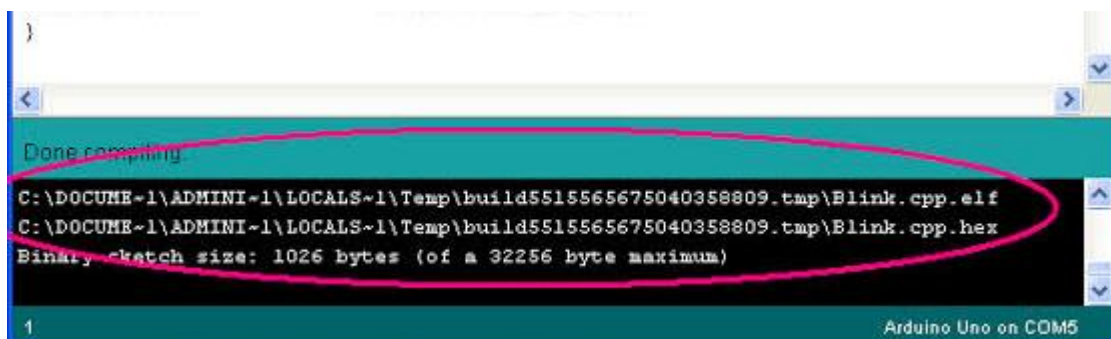
- **File** – You can open a new window for writing the code or open an existing one. Following table shows the number of further subdivisions the file option is categorized into.



As you go to the preference section and check the compilation section, the Output Pane will show the code compilation as you click the upload button.



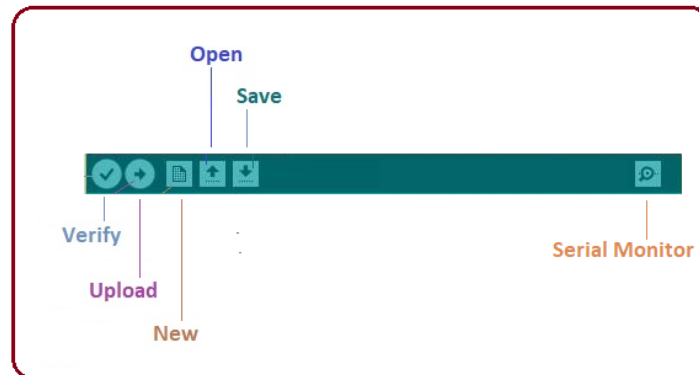
And at the end of compilation, it will show you the hex file it has generated for the recent sketch that will send to the Arduino Board for the specific task you aim to achieve.



- **Edit** – Used for copying and pasting the code with further modification for font
- **Sketch** – For compiling and programming
- **Tools** – Mainly used for testing projects. The Programmer section in this panel is used for burning a bootloader to the new microcontroller.

- **Help** – In case you are feeling skeptical about software, complete help is available from getting started to troubleshooting.

The **Six Buttons** appearing under the Menu tab are connected with the running program as follow.



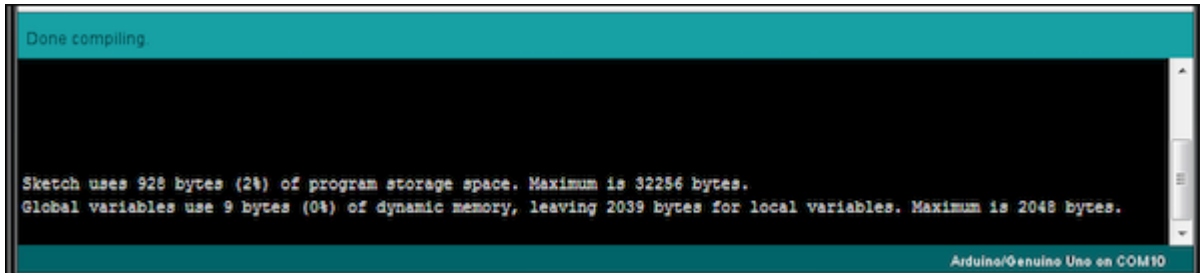
- The check mark appearing in the circular button is used to verify the code. Click this once you have written your code.
- The arrow key will upload and transfer the required code to the Arduino board.
- The dotted paper is used for creating a new file.
- The upward arrow is reserved for opening an existing Arduino project.
- The downward arrow is used to save the current running code.
- The button appearing on the top right corner is a **Serial Monitor** – A separate pop-up window that acts as an independent terminal and plays a vital role for sending and receiving the Serial Data. You can also go to the Tools panel and select Serial Monitor, or pressing Ctrl+Shift+M all at once will open it instantly. The Serial Monitor will actually help to debug the written Sketches where you can get a hold of how your program is operating. Your Arduino Module should be connected to your computer by USB cable in order to activate the Serial Monitor.
- You need to select the baud rate of the Arduino Board you are using right now. For my Arduino Uno Baud Rate is 9600, as you write the following code and click the Serial Monitor, the output will show as the image below.



The main screen below the Menu bard is known as a simple text editor used for writing the required code.



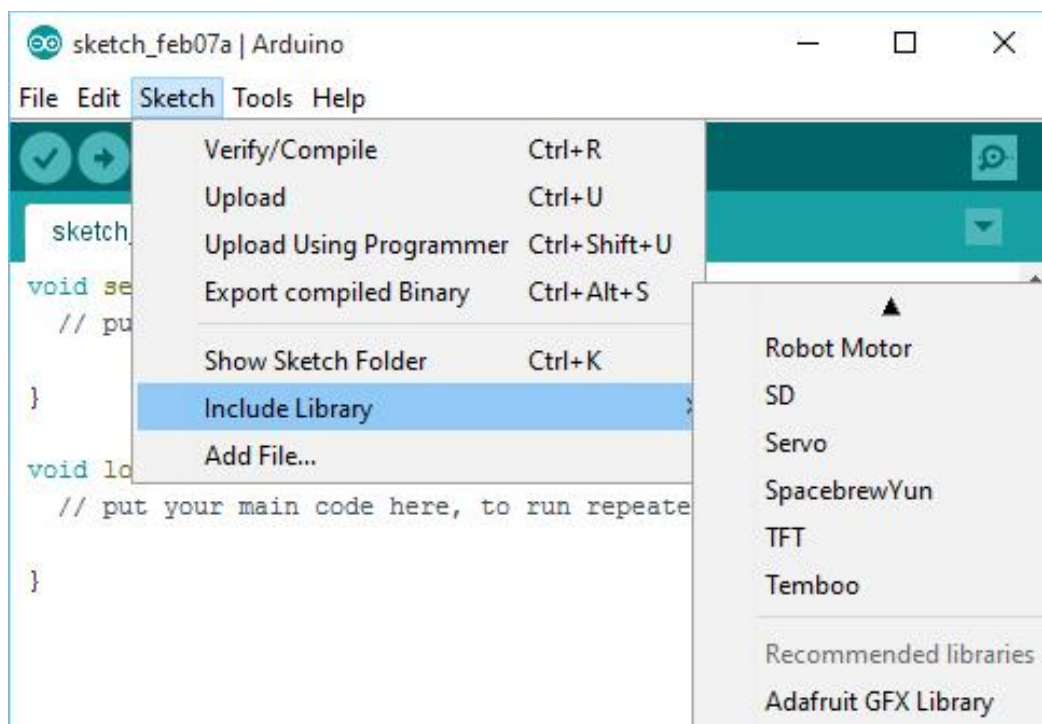
The bottom of the main screen is described as an Output Pane that mainly highlights the compilation status of the running code: the memory used by the code, and errors occurred in the program. You need to fix those errors before you intend to upload the hex file into your Arduino Module.



More or less, Arduino C language works similar to the regular C language used for any embedded system microcontroller, however, there are some dedicated libraries used for calling and executing specific functions on the board.

Libraries:

Libraries are very useful for adding the extra functionality into the Arduino Module. There is a list of libraries you can add by clicking the Sketch button in the menu bar and going to Include Library.



As you click the Include Library and Add the respective library it will on the top of the sketch with a #include sign. Suppose, I Include the EEPROM library, it will appear on the text editor as

```
#include <EEPROM.h>.
```

Most of the libraries are preinstalled and come with the Arduino software. However, you can also download them from the external sources.

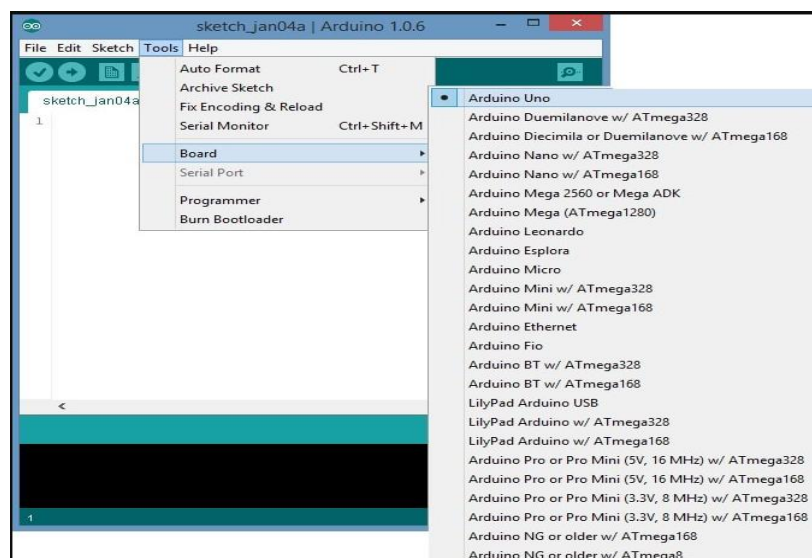
Making pins Input and output:

The `digitalRead` and `digitalWrite` commands are used for addressing and making the Arduino pins as an input and output respectively.

These commands are text sensitive i.e. you need to write them down the exact way they are given like `digitalWrite` starting with small “d” and write with capital “W”. Writing it down with `Digitalwrite` or `digitalwrite` won’t be calling or addressing any function.

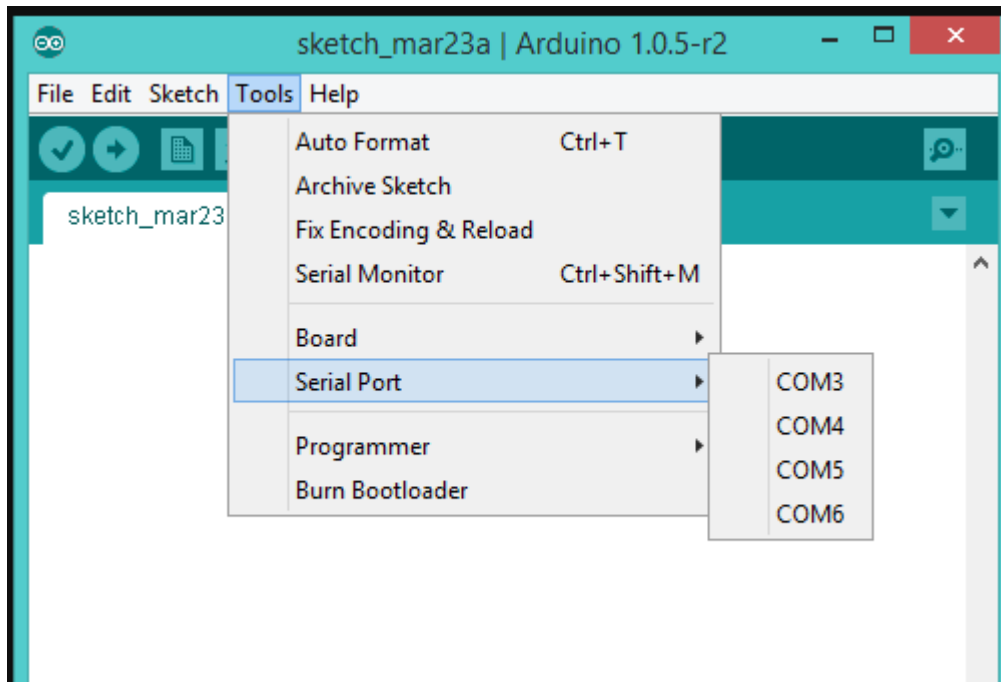
How to select the board:

In order to upload the sketch, you need to select the relevant board you are using and the ports for that operating system. As you click the Tools on the Menu, it will open like the figure below.



- Just go to the “Board” section and select the board you aim to work on. Similarly, COM1, COM2, COM4, COM5, COM7 or higher are reserved for the serial and USB board. You can look for the USB serial device in the ports section of the Windows Device Manager.

Following figure shows the COM4 that I have used for my project, indicating the Arduino Uno with COM4 port at the right bottom corner of the screen.



- After correct selection of both Board and Serial Port, click the verify and then upload button appearing in the upper left corner of the six button section or you can go to the Sketch section and press verify/compile and then upload.
- The sketch is written in the text editor and is then saved with the file extension .ino.

It is important to note that the recent Arduino Modules will reset automatically as you compile and press the upload button the IDE software, however, older version may require the physical reset on the board.

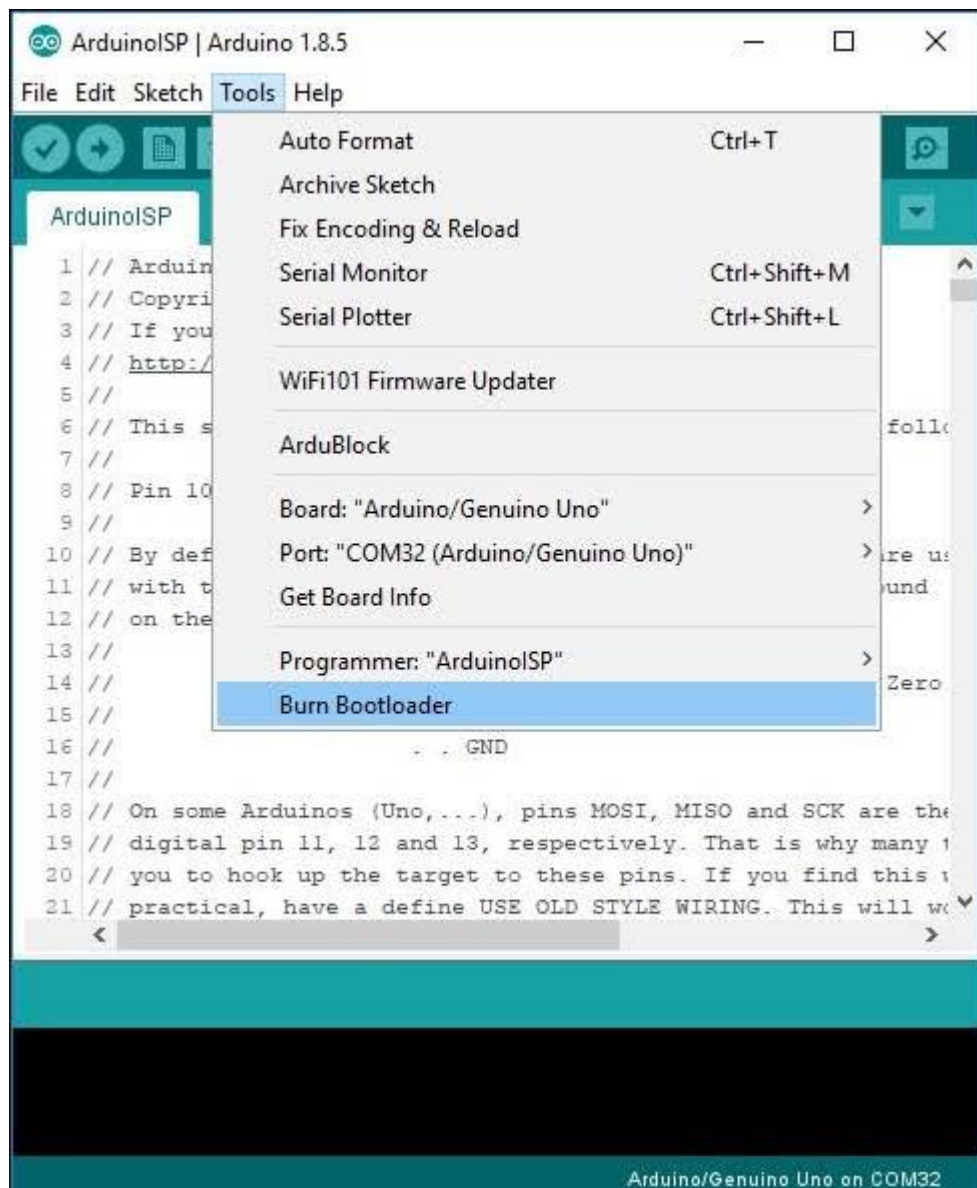
- Once you upload the code, TX and RX LEDs will blink on the board, indicating the desired program is running successfully.

Note: The port selection criteria mentioned above is dedicated for Windows operating system only, you can check this [Guide](#) if you are using MAC or Linux.

- The amazing thing about this software is that no prior arrangement or bulk of mess is required to install this software, you will be writing your first program within 2 minutes after the installation of the IDE environment.

BootLoader:

As you go to the Tools section, you will find a bootloader at the end. It is very helpful to burn the code directly into the controller, setting you free from buying the external burner to burn the required code.



When you buy the new Arduino Module, the bootloader is already installed inside the controller. However, if you intend to buy a controller and put in the Arduino module, you need to burn the bootloader again inside the controller by going to the Tools section and selecting the burn bootloader.

ADVANTAGES

- Highly sensitive
- Works according to the soil condition
- Low cost and reliable circuit
- Complete elimination of manpower

APPLICATIONS

- Industrial Automation
- Weather stations
- Home Automation
- Roof Gardens
- Lawns
- Agriculture Lands
- Home Gardens

CONCLUSION

This combination of solar-wind energy source will be highly effective in commercial areas. It is eco-friendly at the same time prevents accidents due to lightening. It is used to cut short power charge. By this system electricity charge could be saved as very less maintenance charge is required for equipment. More over there is no power cut or load shedding at any times. In addition to this, the system is controlled by INTERNET OF THINGS as site manager is able to receive detailed information of facility at site, efficient maintenance for regular checkup and failure could be performed conveniently. It is the most reliable and cost efficient. This research is at an underdeveloped stage and may take years to bring it into market. We encourage the scientific community to consider this technology along with others when contemplating efforts and resources for renewable energy.

REFERENCES

- P. Singh and S. Saikia, “Arduino-based smart irrigation using water flow sensor, soil moisture sensor, temperature sensor and ESP8266 WiFi module”, 2016 IEEE Region 10 Humanitarian Technology Conference (R10-HTC), Agra, India, 2016, pp. 1-4.
- Kalaiarasi.D, A.Anusha,D.BerslinJeni and M.Monisha,“ENHANCEMENT OF HYBRID POWER SYSTEMS USING IOT”, International Journal of Advanced Research Trends in Engineering and Technology (IJARTET), Vol. 3, Special Issue 19, April 2016.
- S. Darshna1, T.Sangavi, Sheena Mohan, A.Soundharya,Sukanya,“Smart Irrigation System”, IOSR-JECE, May – Jun 2015.
- Ali Dehgani, Mojtaba Hakimzadeh, Amir Habibi, Navid Mehdizadeh Afroozi, “Small Signal Stability Enhancement for Hybrid Power Systems By SVC” World Academy of Science, Engineering and Technology International Journal of Electrical, Computer, Energetic, Electronic and Communication Engineering Vol:8, No:10, 2014.
- Jaehak Byun, Sooyeop Kim, Jaehun Sa, Sangphil Kim, Yong-Tae Shin and Jong-Bae Kim, “Smart City Implementation Models Based on IoT Technology”, Advanced Science and Technology Letters Vol.129 (Mechanical Engineering 2016), pp.209-212.