

**NAME OF PROJECT.: Wireless Home Automation System using Receiving SMS**

**How Is Wireless Home Automation System Using Receiving Sms ?**

**What Is Wireless Home Automation System Using Receiving SMS ?**

**Working Of Project**

**Applications Of Project**

**Future Scopes**

**Advantages & Disadvantages**

**Hardware Used**

**Component Information**

**Microcontroller (Arduino Uno) Information Pin Diagram**

**Software Used**

**Program Code Language**

**Arduino Uno Programming**

- **First You Must Have Your Arduino**
- **Download Arduino IDE Software.**
- **Power Up Your Board.**
- **Launch Arduino IDE.**
- **Open Your First Project.**
- **Select Your Arduino Board**
- **Select Your Serial Port.**
- **Upload The Program To Your Board.**

**Bootload An ATmega328P (Arduino IC)**

- **IC Use**
- **Use Arduino as ISP**
- **Load The Sketch**
- **Connect Your ATmega328**
- **Program The Bootloader**

**Test Arduino IC**

- **power supply pin test**
- **Digital pin test**
- **TX,RX pin test**
  - **Disconnect Power**
  - **connect Jumper Wires**
  - **Connecting Board To Pc Open Arduino Ide**
  - **If Nothing Echoes Back**

**Test Sim900a GSM Module**

- **Material Preparation**
- **Booting Up SIM900A**

- ***Pin Connection***
- ***Basic Arduino Code To Receive A Message***
- ***Power Up Your Board.***
- ***Open Your First Project.***
- ***Select Your Arduino Board***
- ***Select Your Serial Port.***
- ***Upload The Program To Your Board.***
- ***Serial Monitor***
- ***Result***

***Test Project circuit and code***

- ***Material Preparation***
- ***Booting Up SIM900A***
- ***Pin Connection***
- ***Power Up Your Board.***
- ***Open Your First Project.***
- ***Select Your Arduino Board***
- ***Select Your Serial Port.***
- ***Upload The Program To Your Board.***
- ***Result***
- ***How we tested it is explained in the video below***

<https://photos.app.goo.gl/J75SEET5sqJrMgYw8>

***AC to DC Converter 5-12v (SO12V5300)***

***Block Diagram Of Project***

***Arduino Uno Diagram Of Project***

***Microcontroller Diagram Of Project***

***project short information***

***PCB Diagram / Layout Of Project***

***PCB Manufacturing Block Diagram***

- ***Circuit Diagram***
- ***PCB Layout***
- ***Pattern Transfer***
- ***Dark Room Facility***
- ***Pattern Transfer The Check***
- ***Etching***
- ***Cleaning***
- ***Drilling***
- ***Component Mounting***
- ***Soldering***
- ***Testing***

***Board Fitting***

- **PCB Fitting**
- **Sim900a GSM Module Fitting**
- **Bulb Holder,Bulbs, and 6A 3 Pin Socket Fitting**

### **Programming Compile In Upload ATmega328P (Arduino IC)**

- **Power Up Your Board.**
- **Open Your First Project.**
- **Select Your Arduino Board**
- **Select Your Serial Port.**
- **Upload The Program To Your Board.**

### **The Last Test / Result Working**

#### **Components Costs**

- **Component Name**
- **Quantity**
- **Cost**
- **Total Cost**

#### **Conclusions**

### **how is Wireless Home Automation System using Receiving SMS ?**

The microcontroller will extract the message from the received SMS and control the relay module. The relay will turn on or off the appliances as ordered by the owner. The advantage of the project is even if the owner does not have an android phone the can send a SMS to the GSM modem to control the appliances.

### **what is Wireless Home Automation System using Receiving SMS ?**

the home appliances from remote places anywhere in the world. This system allows the owner to control the appliances and to.

#### **Working of Project:-**

In this post we are going to build a home automation circuit by which you can control your home gadgets via GSM network or in other words using SMS commands, you can control up to 5 devices independently. The proposed GSM based home automation system is loaded with several features that make this project standout, like: password protected control for all outputs, automatic recovery from power outage, acknowledgement SMS to the user. We will also see how to automate SMS commands so that you can schedule your commands and execute without your intervention.

#### **Applications of Project:-**

The project can be put to use in the following field,

1. Home automation, which was the seed for developing the idea of project.
2. Remote device control.

- This will help to eliminate need of human personnel attending the device till it has to be switched off/on. Based on experience of approximate time to switch on/off, He'she can control the device, saving time.

- It will also help to control device while attending to other work.
  - User can control device in case of forgetting to do so while leaving for some other place.
3. Energy conservation.  
4. Irrigation systems.

## **FUTURE SCOPES:-**

The project "GSM BASED HOME CONTROL" is intended to automate the certain functions of home appliances. Smart home technologies have been around for about 30 years, mostly relying on some proprietary technologies and applications. With the recent expansion of communication networks, smart home applications can be further enhanced with new dimension of capabilities that were not available before. In particular, wireless access technologies will soon enable exotic and economically feasible applications. The device is much helpful in controlling home. It reduces the wastage of valuable time and our daily life become easier and flexible.

## **ADVANTAGES & DISADVANTAGES:-**

### **ADVANTAGES:-**

- Devices can be controlled from long distances.
- Economical design
- Can be easily implemented in homes
- Can be used by everyone with just the knowledge of text SMS
- Format of the SMS is simple to understand and write

### **DISADVANTAGES:-**

- The system is network dependent. Hence, network congestion can reduce the reliability of the system.
- If user make any mistake while typing the message format, the message will not be acknowledged by Arduino hence the system will not work.

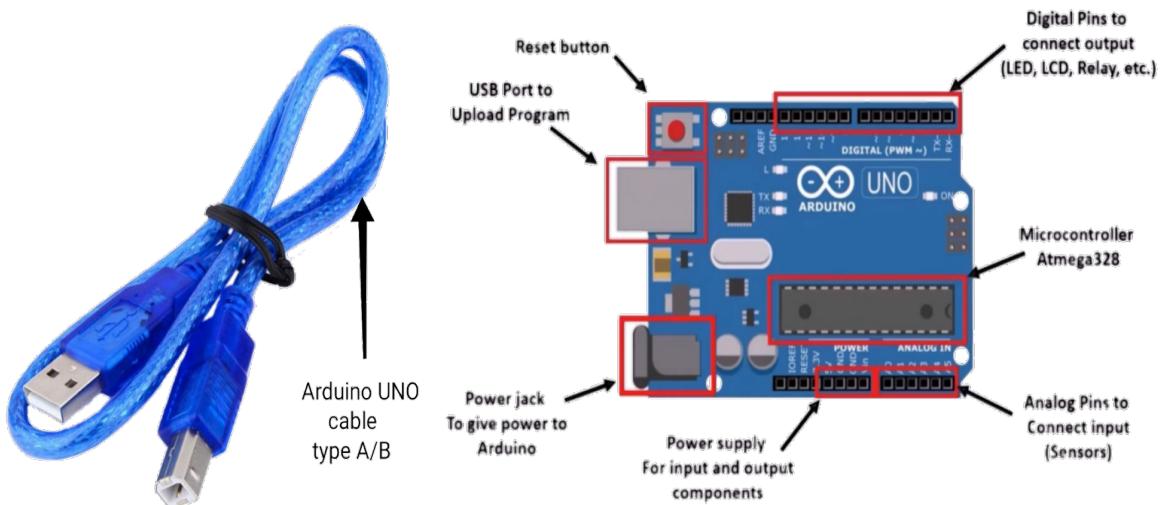
## **Hardware Used:-**

1. Arduino UNO board
2. SIM900A GSM Module
3. 4 Relay
4. relay module 4 channel
5. AC 230v Power Supply
6. LDR sensor
7. Bulb Holder
8. Bulbs
9. 6A 3 Pin Socket / PVC brand box
10. 16A 3 Pin Socket / PVC brand box
11. Jumper Wires Hardware
12. wires Solid type
13. IC Base- 28- Pin
14. ATmega328P ic
15. 2 Pin PCB Connector
16. voltage regulator IC 7805
17. 22pf capacitor

18. 100uf capacitor
19. 10uf capacitor
20. LED 5mm
21. 470 ohm resistor
22. 16MHZ Crystal Oscillator
23. IC Master 40 PIN FEMALE HEADER
24. AC to DC converter 12v/2A
25. etching pcb kit
26. PCB soldering kit
27. Photo paper print
28. mdf board 10mm(61CM×61CM)
29. other

### **Component Information:-**

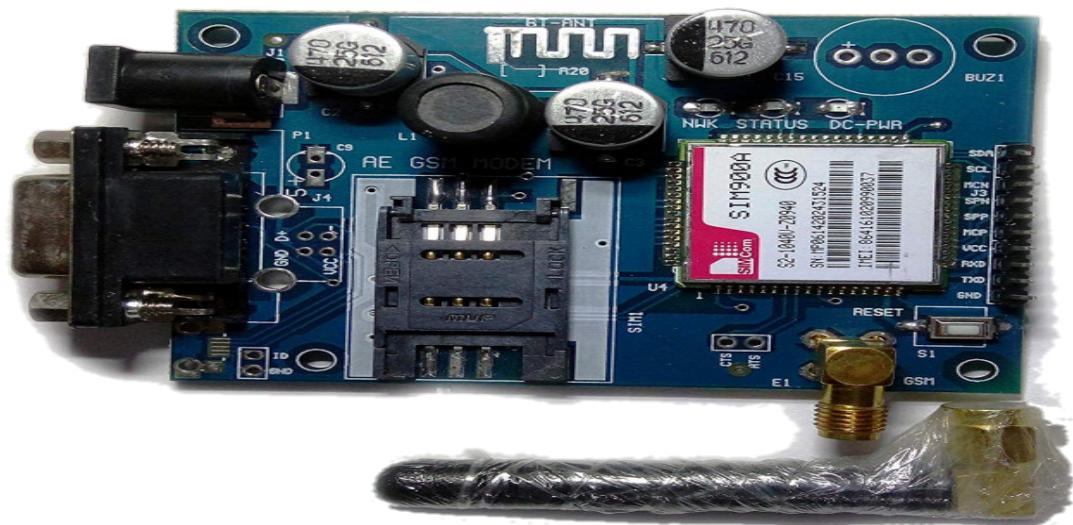
#### **Arduino UNO Hardware:-**



#### **what is a Arduino UNO Hardware?**

Arduino Uno R3 is one kind of ATmega328P based microcontroller board. It includes the whole thing required to hold up the microcontroller; just attach it to a PC with the help of a USB cable, and give the supply using AC-DC adapter or a battery to get started. The term Uno means “one” in the language of “Italian” and was selected for marking the release of Arduino’s IDE 1.0 software. The R3 Arduino Uno is the 3rd as well as most recent modification of the Arduino Uno. Arduino board and IDE software are the reference versions of Arduino and currently progressed to new releases. The Uno-board is the primary in a sequence of USB-Arduino boards, & the reference model designed for the Arduino platform.

#### **SIM 900A GSM Module:-**



### **What is the GSM module?**

A GSM modem or GSM module is a hardware device that uses GSM mobile telephone technology to provide a data link to a remote network. From the view of the mobile phone network, they are essentially identical to an ordinary mobile phone, including the need for a SIM to identify themselves to the network.

### **4 Relay:-**



### **what is a 4 Relay?**

The 4 Channel Relay Breakout is an easy way to use your Arduino, Raspberry Pi, or other microcontroller to switch high voltages and high current loads. ... Each relay has the common, normally open, and normally closed pin broken out to a convenient 5.0mm pitch screw terminal.

### **LDR sensor:-**



### ***what is Ldr sensor module ?***

The LDR Sensor Module is used to detect the presence of light / measuring the intensity of light. The output of the module goes high in the presence of light and it becomes low in the absence of light. The sensitivity of the signal detection can be adjusted using potentiometer.

### **Bulb Holder:-**



### ***what is a Bulb Holder ?***

A lightbulb socket, light socket, lamp socket or lampholder is a device which mechanically supports and provides electrical connections for a compatible electric lamp. ... Many of the later standards conform to a general coding system in which a socket type is designated by a letter or abbreviation followed by a number.

### ***Bulb:-***



### ***what is a led Bulb ?***

An LED lamp or LED light bulb is an electric light that produces light using light-emitting diodes. LED lamps are significantly more energy-efficient than equivalent incandescent lamps and can be .

### ***Board:-***



### ***what is electrical board ?***

A distribution board (also known as panelboard, breaker panel, or electric panel) is a component of an electricity supply system that divides an electrical power feed into subsidiary circuits while providing a protective fuse or circuit breaker for each circuit in a common enclosure.

### ***6A 3 Pin Socket:-***



### ***what is a 3 Pin Socket ?***

3 pin plugs are designed so that mains electricity can be supplied to electrical appliances safely. ... A 3 pin plug consists of three pins (hence the name). Each pin must be correctly connected to the three wires in the electrical cable.

### ***Jumper Wires Hardware:-***



### ***what is a Jumper Wires Hardware?***

A jump wire is an electrical wire, or group of them in a cable, with a connector or pin at each end, which is normally used to interconnect the components of a breadboard or other prototype or test circuit, internally or with other equipment or components, without soldering.

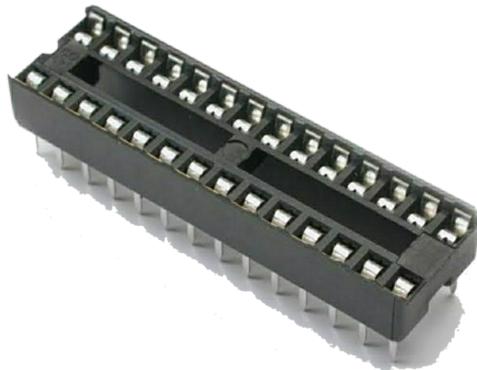
## wires Solid type:-



## what is a wires Solid type ?

Solid wire, also known as solid-core or single-strand wire, is made up of one piece of metal wire, usually surrounded by protective sheathing. It is often used for circuit breadboard wiring. Its manufacture is less expensive than stranded wire.

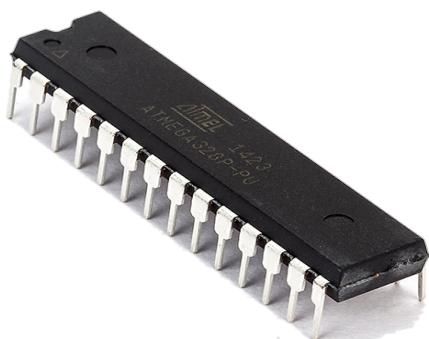
## IC Base- 28- Pin:-



## what is a IC Base- 28- Pin?

Description: This is a standard 28 pin IC base allows user to connect the ICs with the circuits. IC sockets are generally for preventing damage to IC's from soldering and while testing multiple circuits. These are made from Black Thermoplastic and tin-plated alloy contacts. One end is notched to aid in identification

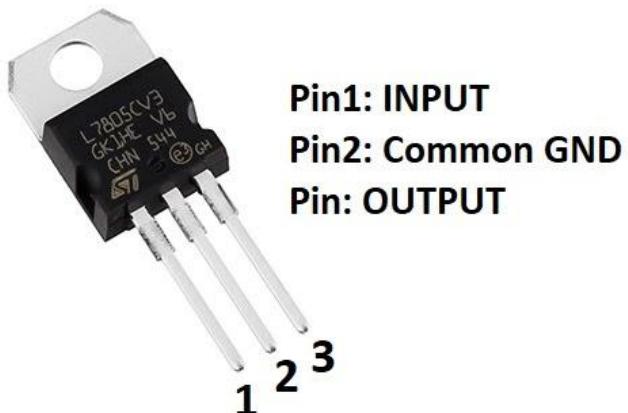
## ATmega328P ic:-



### **what is a ATmega328P ic ?**

The ATmega328 is a single-chip microcontroller created by Atmel in the megaAVR family (later Microchip Technology acquired Atmel in 2016). It has a modified Harvard architecture 8-bit RISC processor core.

### **voltage regulator IC 7805:-**



### **what is a voltage regulator ic 7805 ?**

7805 Voltage Regulator, a member of 78xx series of fixed linear voltage regulators used to maintain such fluctuations, is a popular voltage regulator integrated circuit (IC). The xx in 78xx indicates the output voltage it provides. 7805 IC provides +5 volts regulated power supply with provisions to add a heat sink.

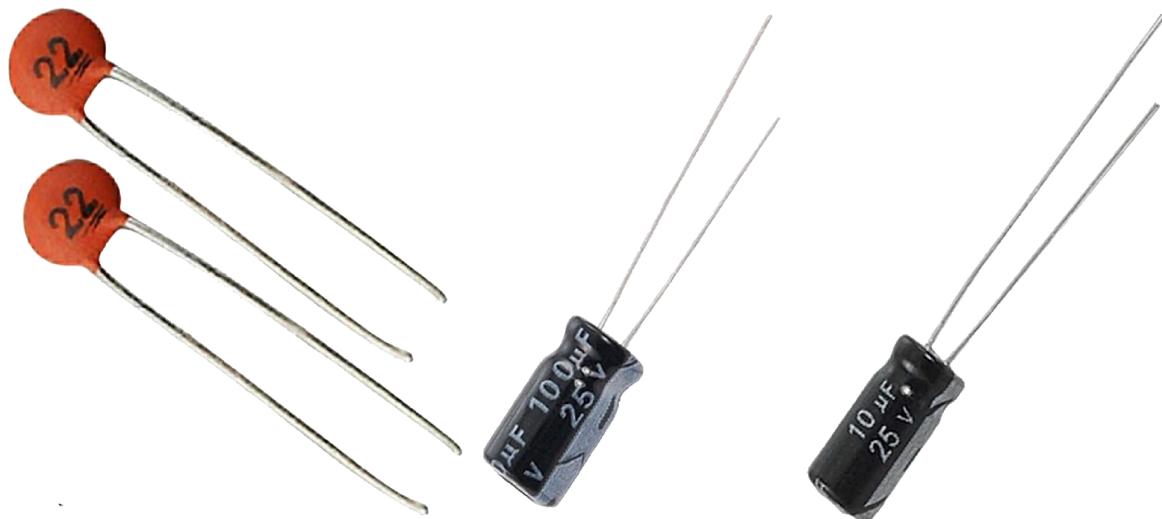
### **2 Pin PCB Connector:-**



### **what is a PCB Connector ?**

A 'PCB' or Printed Circuit Board connects electrical components on a conductive track or between pads on the board. ... PCB connectors are mounted on the PCB and are typically used to transfer signals or power from one PCB to another, or to transfer to or from the PCB from another source within the unit.

### **22pf / 100uf / 10uf capacitor:-**



### ***what is a 22pF / 100uF / 10uF capacitor ?***

Capacitor, device for storing electrical energy, consisting of two conductors in close proximity and insulated from each other. A simple example of such a storage device is the parallel-plate capacitor. ... A parallel-plate capacitor, as shown in part A, consists of two flat conducting plates, each of area A.

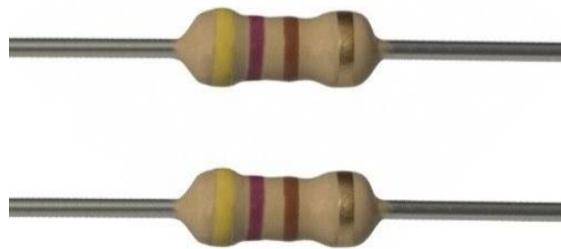
### ***LED 5mm:-***



### ***what is a LED ?***

A Light Emitting Diode (LED) is a semiconductor device, which can emit light when an electric current passes through it. ... The wavelength of the light emitted depends on the bandgap of the semiconductor material. Harder materials with stronger molecular bonds generally have wider bandgaps.

### ***470 ohm resistor:-***



### ***what is a 470 ohm resistor ?***

470 Ohm 0.5W Metal Film Resistor (MFR) with  $\pm 1\%$  Tolerance. ... Resistance: 470 Ohm, Power Rating: 0.5 Watt, Approximate Maximum Current: 32.62mA .

### ***16MHZ Crystal Oscillator:-***



### ***what is a 16 mhz crystal oscillator?***

The 16 MHz Crystal Oscillator module is designed to handle off-chip crystals that have a frequency of 4 $\times$ 16 MHz. ... The oscillator design generates low frequency and phase jitter, which is recommended for USB operation.

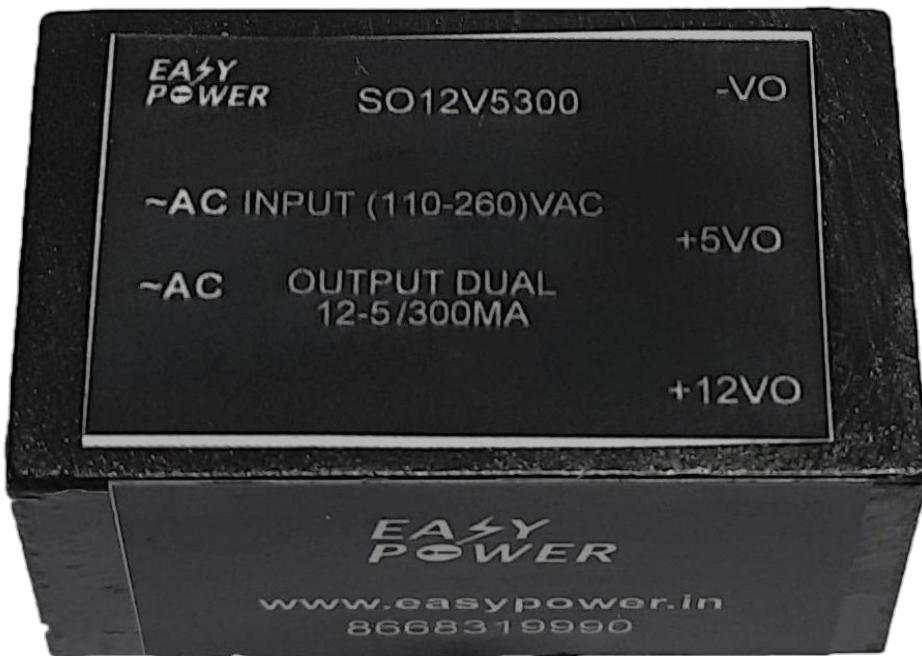
### ***IC Master 40 PIN FEMALE HEADER:-***



### ***what is a 40 PIN FEMALE HEADER?***

This 40 Pin Female Break Away Header has the industry standard spacing of 2.54mm and are designed for soldering directly into printed circuit boards (PCB). They are supplied Best for extension of pins of Microcontrollers and ICs.

### **AC to DC converter 12v:-**



#### **what is a SO12V5300 ?**

SO12V5300 Micro SMPS – a compact size power converter offered by EASY POWER. It features universal input voltage, taking AC/DC voltage, low power consumption, high reliability, and safer isolation.

### **etching pcb kit:-**



printer photo paper



copper pcb



marker



clothes iron



etching pcb



safety Gloves



Scrub Pad



Plastic Tray



Pitambari



hand drill and drill bitx2



paper tape

### ***what is a etching pcb?***

Etching is the process of material being removed from material's surface. ... The etching process is one of the major steps in the final processing of printed circuit boards (PCBs). This process removes excess copper so as to reveal the desired circuit patterns.

### ***PCB soldering kit:-***



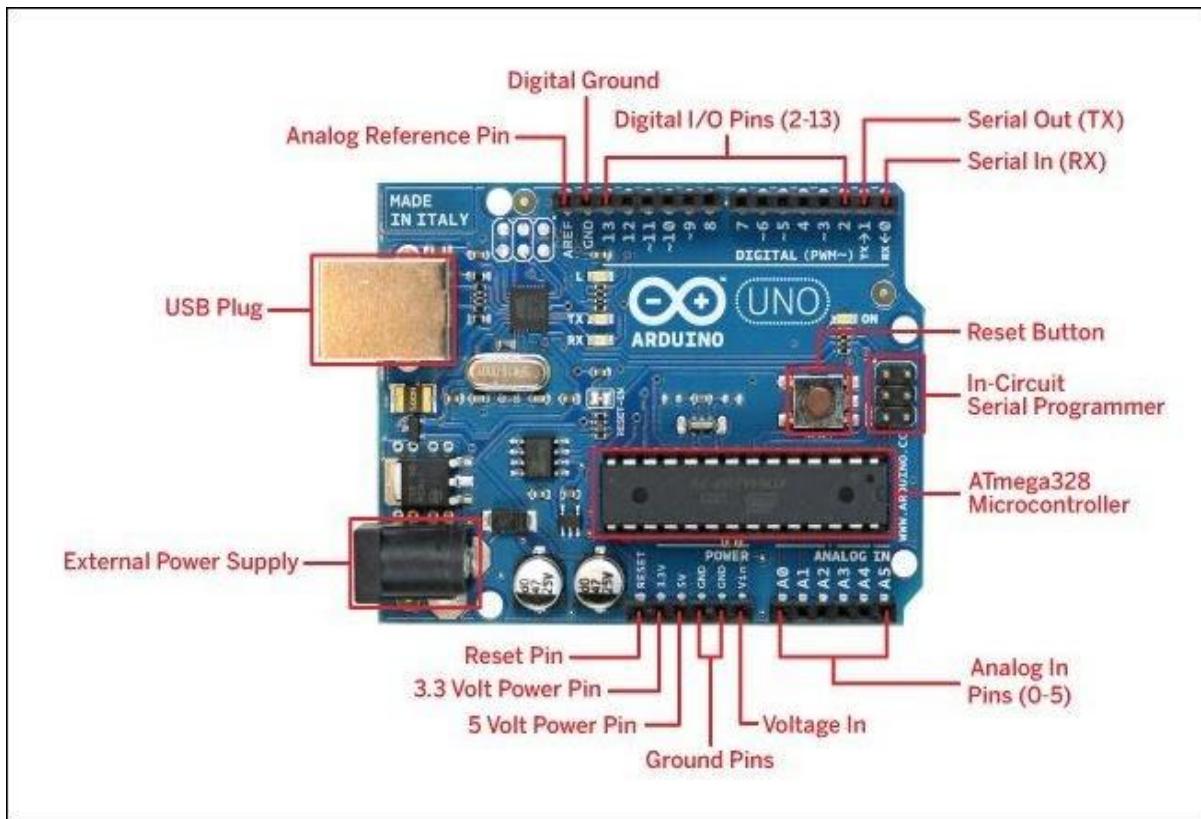
### ***what is a pcb soldering?***

PCB soldering is another term for the process of soldering electrical circuit boards. ... As the soldering iron melts this metal, it is then used a bit like glue to stick to pieces together. As the solder metal cools, it will re-harden into one large shape that connects the two parts.

### ***mdf board 10mm(61CM×61CM):-***



## **Microcontroller (Arduino Uno) Information Pin Diagram:-**



### **Arduino Uno R3 Pin Diagram**

The Arduino Uno R3 pin diagram is shown below. It comprises 14-digit I/O pins. From these pins, 6-pins can be utilized like PWM outputs. This board includes 14 digital input/output pins, Analog inputs-6, a USB connection, quartz crystal-16 MHz, a power jack, a USB connection, resonator-16Mhz, a power jack, an ICSP header an RST button.

#### **Power Supply**

The power supply of the Arduino can be done with the help of an exterior power supply otherwise USB connection. The exterior power supply (6 to 20 volts) mainly includes a battery or an AC to DC adapter. The connection of an adapter can be done by plugging a center-positive plug (2.1mm) into the power jack on the board. The battery terminals can be placed in the pins of Vin as well as GND. The power pins of an Arduino board include the following.

#### **Vin**

The input voltage or Vin to the Arduino while it is using an exterior power supply opposite to volts from the connection of USB or else RPS (regulated power supply). By using this pin, one can supply the voltage.

#### **5Volts**

The RPS can be used to give the power supply to the microcontroller as well as components which are used on the Arduino board. This can approach from the input voltage through a regulator.

#### **3V3**

A 3.3 supply voltage can be generated with the onboard regulator, and the highest draw current will be 50 mA.

### ***GND***

GND (ground) pins

### ***Memory***

The memory of an ATmega328 microcontroller includes 32 KB and 0.5 KB memory is utilized for the Boot loader), and also it includes SRAM-2 KB as well as EEPROM-1KB.

### ***Input and Output***

We know that an Arduino Uno R3 includes 14-digital pins which can be used as an input otherwise output by using the functions like pin Mode (), digital Read(), and digital Write(). These pins can operate with 5V, and every digital pin can give or receive 20mA, & includes a 20k to 50k ohm pull up resistor. The maximum current on any pin is 40mA which cannot surpass for avoiding the microcontroller from the damage. Additionally, some of the pins of an Arduino include specific functions.

### ***Serial Pins***

The serial pins of an Arduino board are TX (1) and RX (0) pins and these pins can be used to transfer the TTL serial data. The connection of these pins can be done with the equivalent pins of the ATmega8 U2 USB to TTL chip.

### ***External Interrupt Pins***

The external interrupt pins of the board are 2 & 3, and these pins can be arranged to activate an interrupt on a rising otherwise falling edge, a low-value otherwise a modify in value

### ***PWM Pins***

The PWM pins of an Arduino are 3, 5, 6, 9, 10, & 11, and gives an output of an 8-bit PWM with the function analog Write () .

### ***SPI (Serial Peripheral Interface) Pins***

The SPI pins are 10, 11, 12, 13 namely SS, MOSI, MISO, SCK, and these will maintain the SPI communication with the help of the SPI library.

### ***LED Pin***

An Arduino board is inbuilt with a LED using digital pin-13. Whenever the digital pin is high, the LED will glow otherwise it will not glow.

### ***TWI (2-Wire Interface) Pins***

The TWI pins are SDA or A4, & SCL or A5, which can support the communication of TWI with the help of Wire library.

### ***AREF (Analog Reference) Pin***

An analog reference pin is the reference voltage to the inputs of an analog I/Ps using the function like analog Reference().

### ***Reset (RST) Pin***

This pin brings a low line for resetting the microcontroller, and it is very useful for using an RST button toward shields which can block the one over the Arduino R3 board.

### ***Communication***

The communication protocols of an Arduino Uno include SPI, I2C, and UART serial communication.

### ***UART***

An Arduino Uno uses the two functions like the transmitter digital pin1 and the receiver digital pin0. These pins are mainly used in UART TTL serial communication.

### ***I2C***

An Arduino UNO board employs SDA pin otherwise A4 pin & A5 pin otherwise SCL pin is used for I2C communication with wire library. In this, both the SCL and SDA are CLK signal and data signal.

### ***SPI Pins***

The SPI communication includes MOSI, MISO, and SCK.

#### ***MOSI (Pin11)***

This is the master out slave in the pin, used to transmit the data to the devices

#### ***MISO (Pin12)***

This pin is a serial CLK, and the CLK pulse will synchronize the transmission of which is produced by the master.

#### ***SCK (Pin13)***

The CLK pulse synchronizes data transmission that is generated by the master. Equivalent pins with the SPI library is employed for the communication of SPI. ICSP (in-circuit serial programming) headers can be utilized for programming ATmega microcontroller directly with the boot loader.

### ***Please click for full detail:-***

<https://youtu.be/jXV1V33sT3U>

### ***Software Used:-***

Arduino ide

### ***program code language:-***

C++

### ***How to Arduino Uno Programming:-***

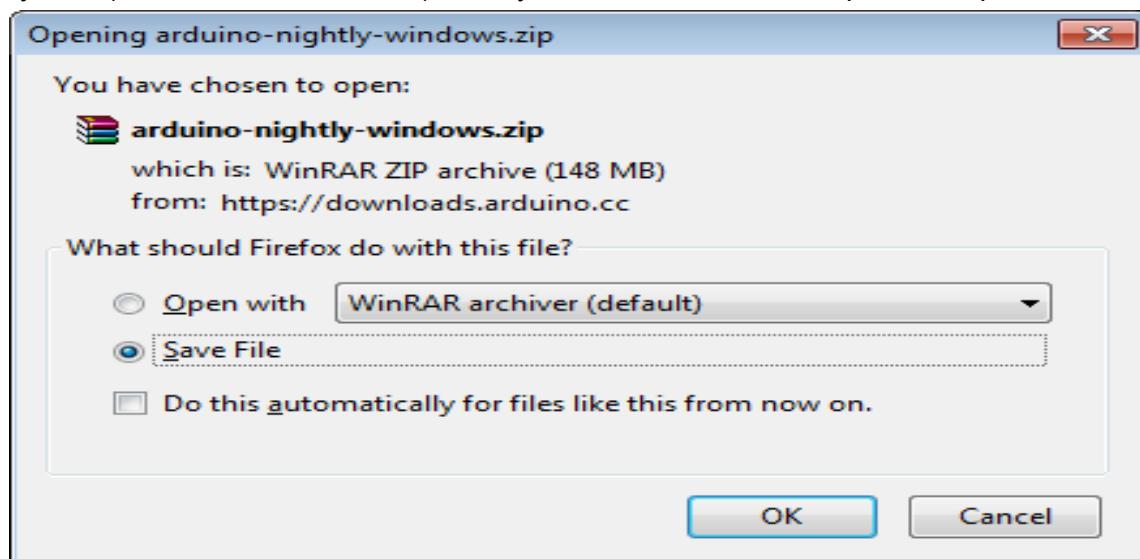
After learning about the main parts of the Arduino UNO board, we are ready to learn how to set up the Arduino IDE. Once we learn this, we will be ready to upload our program on the Arduino board.

In this section, we will learn in easy steps, how to set up the Arduino IDE on our computer and prepare the board to receive the program via USB cable.

**Step 1** – First you must have your Arduino board (you can choose your favorite board) and a USB cable. In case you use Arduino UNO, Arduino Duemilanove, Nano, Arduino Mega 2560, or Diecimila, you will need a standard USB cable (A plug to B plug), the kind you would connect to a USB printer as shown in the following image.

**Step 2** – Download Arduino IDE Software.

You can get different versions of Arduino IDE from the Download page on the Arduino Official website. You must select your software, which is compatible with your operating system (Windows, IOS, or Linux). After your file download is complete, unzip the file.



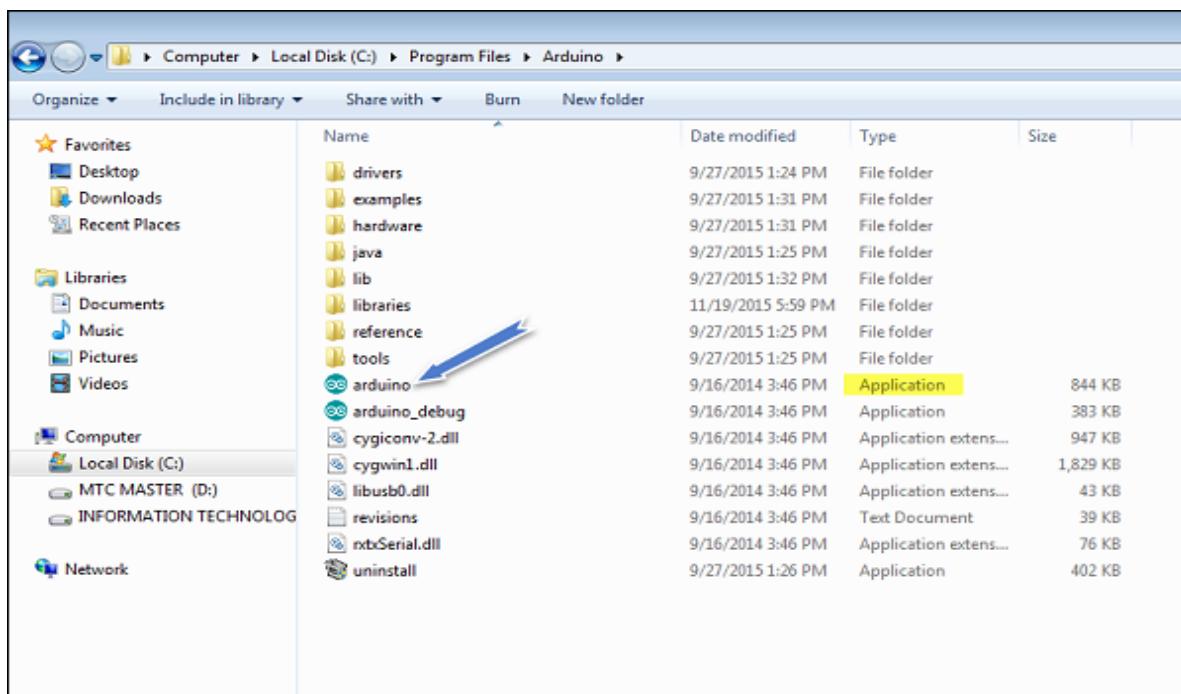
**Step 3** – Power up your board.

The Arduino Uno, Mega, Duemilanove and Arduino Nano automatically draw power from either, the USB connection to the computer or an external power supply. If you are using an Arduino Diecimila, you have to make sure that the board is configured to draw power from the USB connection. The power source is selected with a jumper, a small piece of plastic that fits onto two of the three pins between the USB and power jacks. Check that it is on the two pins closest to the USB port.

Connect the Arduino board to your computer using the USB cable. The green power LED (labeled PWR) should glow.

**Step 4** – Launch Arduino IDE.

After your Arduino IDE software is downloaded, you need to unzip the folder. Inside the folder, you can find the application icon with an infinity label (application.exe). Double-click the icon to start the IDE.

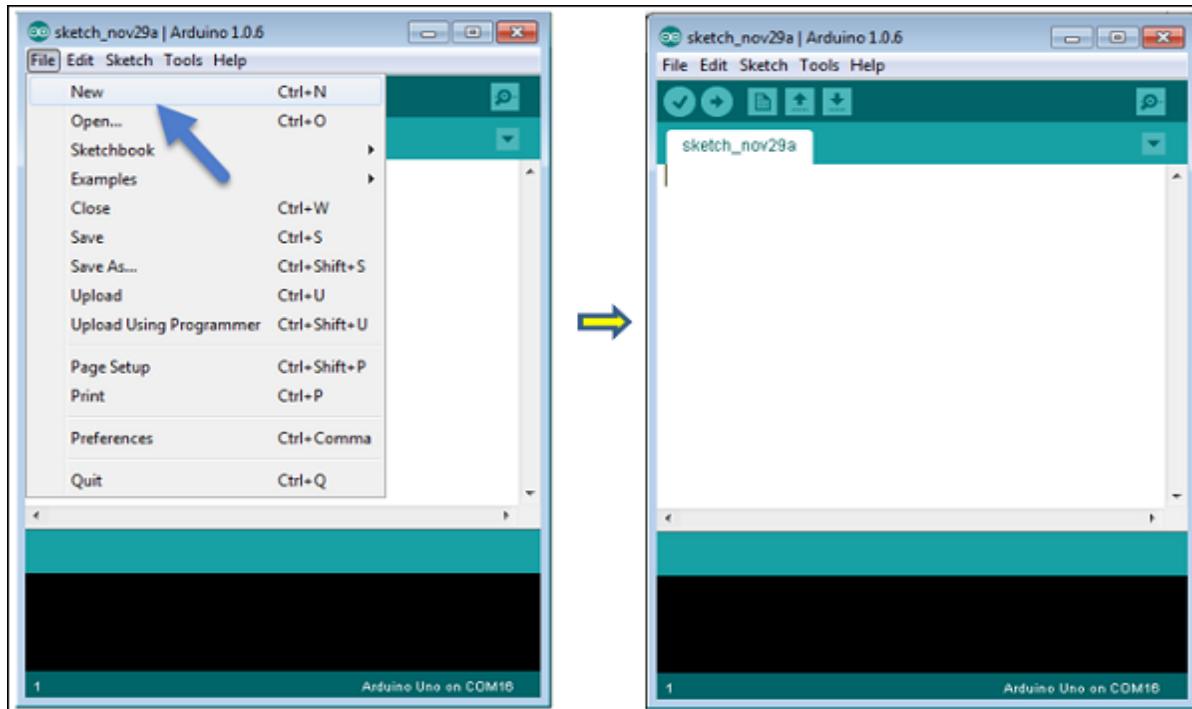


### Step 5 – Open your first project.

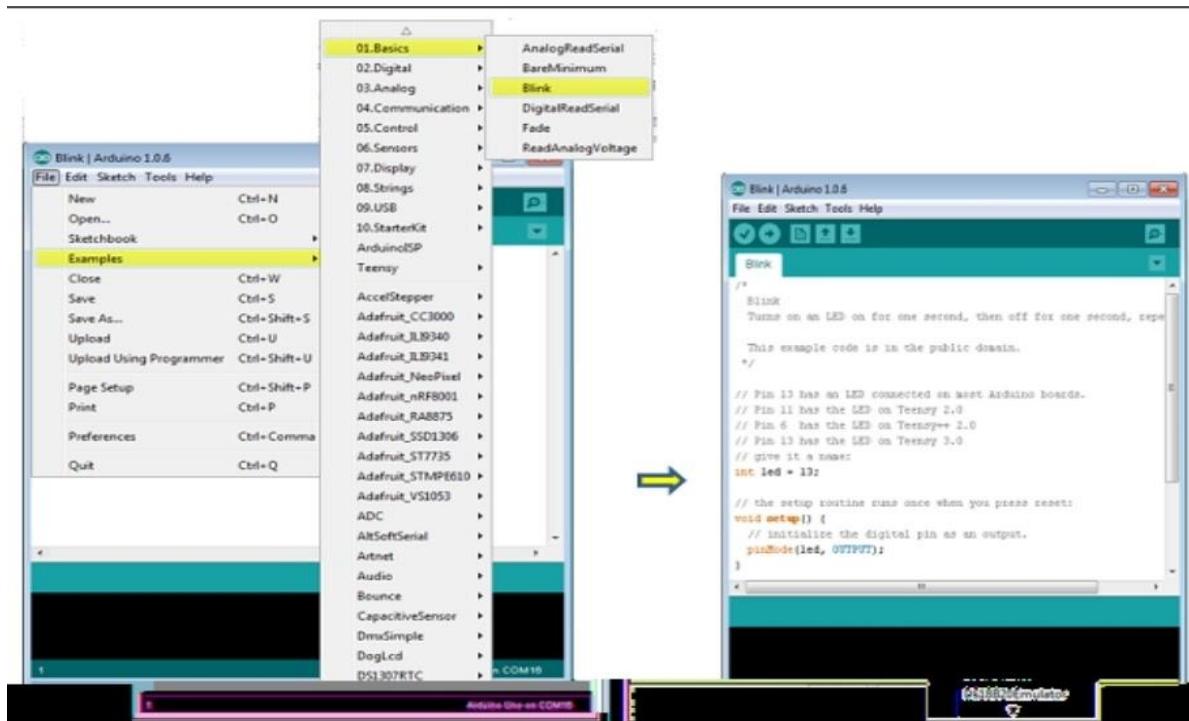
Once the software starts, you have two options – Create a new project.

Open an existing project example.

To create a new project, select File → New.



To open an existing project example, select File → Example → Basics → Blink.

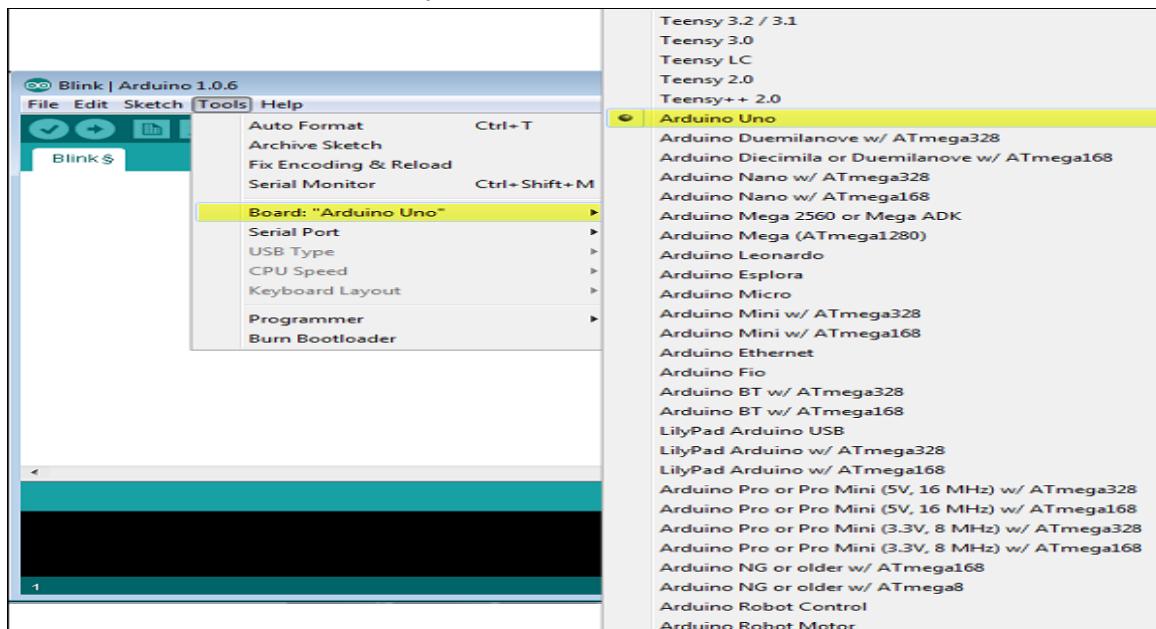


Here, we are selecting just one of the examples with the name Blink. It turns the LED on and off with some time delay. You can select any other example from the list.

### Step 6 – Select your Arduino board.

To avoid any error while uploading your program to the board, you must select the correct Arduino board name, which matches with the board connected to your computer.

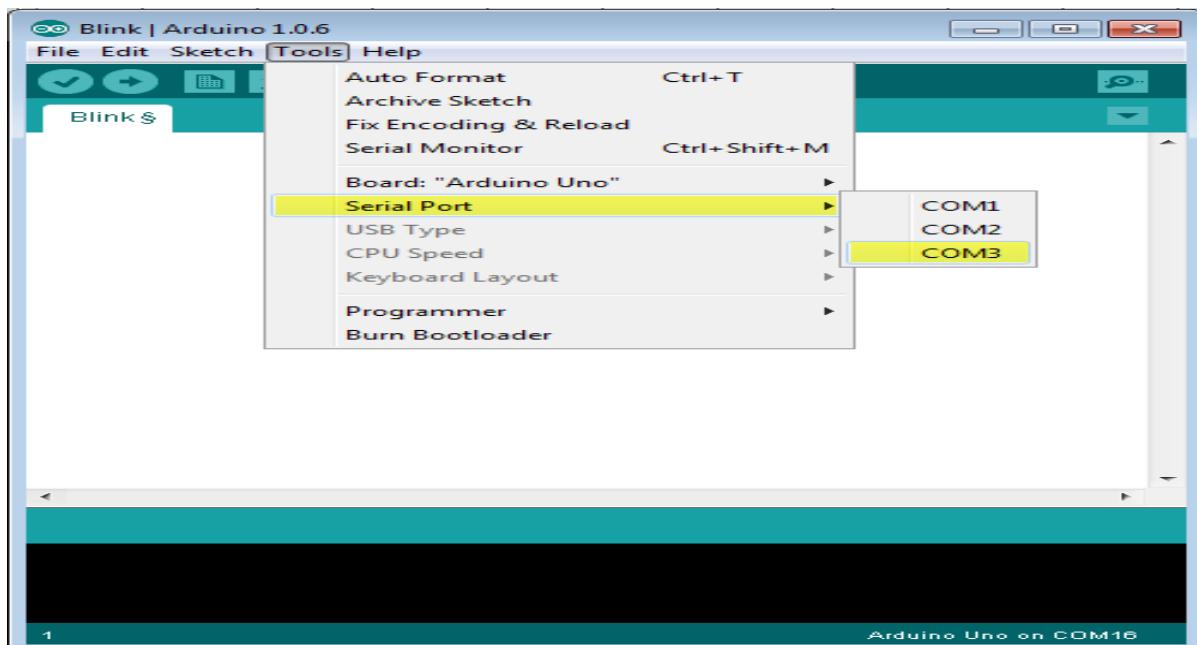
Go to Tools → Board and select your board.



Here, we have selected Arduino Uno board according to our tutorial, but you must select the name matching the board that you are using.

### Step 7 – Select your serial port.

Select the serial device of the Arduino board. Go to Tools → Serial Port menu. This is likely to be COM3 or higher (COM1 and COM2 are usually reserved for hardware serial ports). To find out, you can disconnect your Arduino board and re-open the menu, the entry that disappears should be of the Arduino board. Reconnect the board and select that serial port.



#### **Step 8 – Upload the program to your board.**

Before explaining how we can upload our program to the board, we must demonstrate the function of each symbol appearing in the Arduino IDE toolbar.



A – Used to check if there is any compilation error.

B – Used to upload a program to the Arduino board.

C – Shortcut used to create a new sketch.

D – Used to directly open one of the example sketch.

E – Used to save your sketch.

F – Serial monitor used to receive serial data from the board and send the serial data to the board.

Now, simply click the "Upload" button in the environment. Wait a few seconds; you will see the RX and TX LEDs on the board, flashing. If the upload is successful, the message "Done uploading" will appear in the status bar.

Note – If you have an Arduino Mini, NG, or other board, you need to press the reset button physically on the board, immediately before clicking the upload button on the Arduino Software.

### **How to Bootload an ATmega328P (arduino IC):-**

A bootloader is "a small program that has been loaded on to the microcontroller. It allows you to upload code without using any additional hardware." So first if you were to do anything on those microcontrollers you need a bootloader burned on those chips.

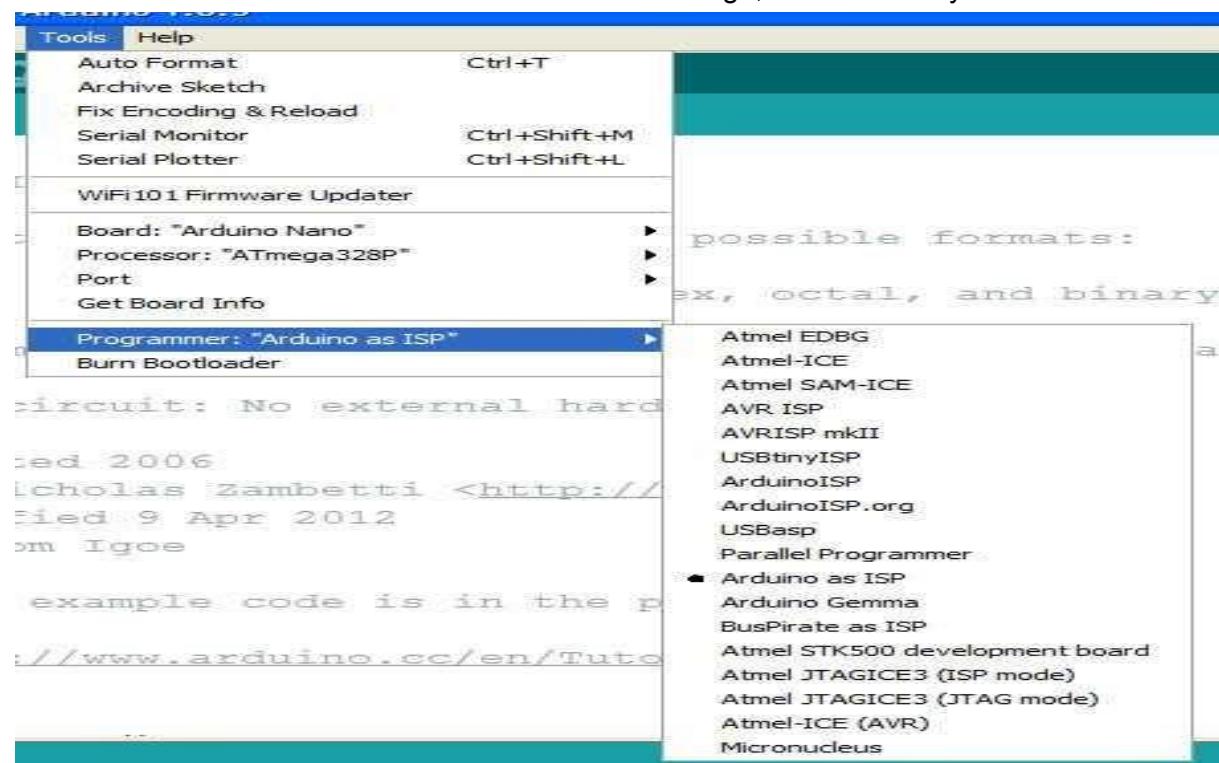
The IC we take from outside, the bootloader is locked in such a way that the bootloader has to be unlocked. Below is the way to unlock the bootloader. It consists of 5 steps in total.

#### **STEP 1: IC Use**

The Arduino software has the facility to unlock the bootloader, that is the ATmega328P only. If other than this, the bootloader of any other IC has to be unlocked, then the file has to be changed.

#### **STEP 2: Use Arduino as ISP**

The whole process of loading the bootloader code and burning properly the fuses to make an ATmega microcontroller an "Arduino" is managed by the Arduino Software (IDE): it provides a specific menu item and allows you to use a variety of programming devices. Among the programmers, the "Arduino as ISP" is the cheapest and most practical solution to burn a bootloader on another Arduino board with ATmega, 32U4 or ATTiny.



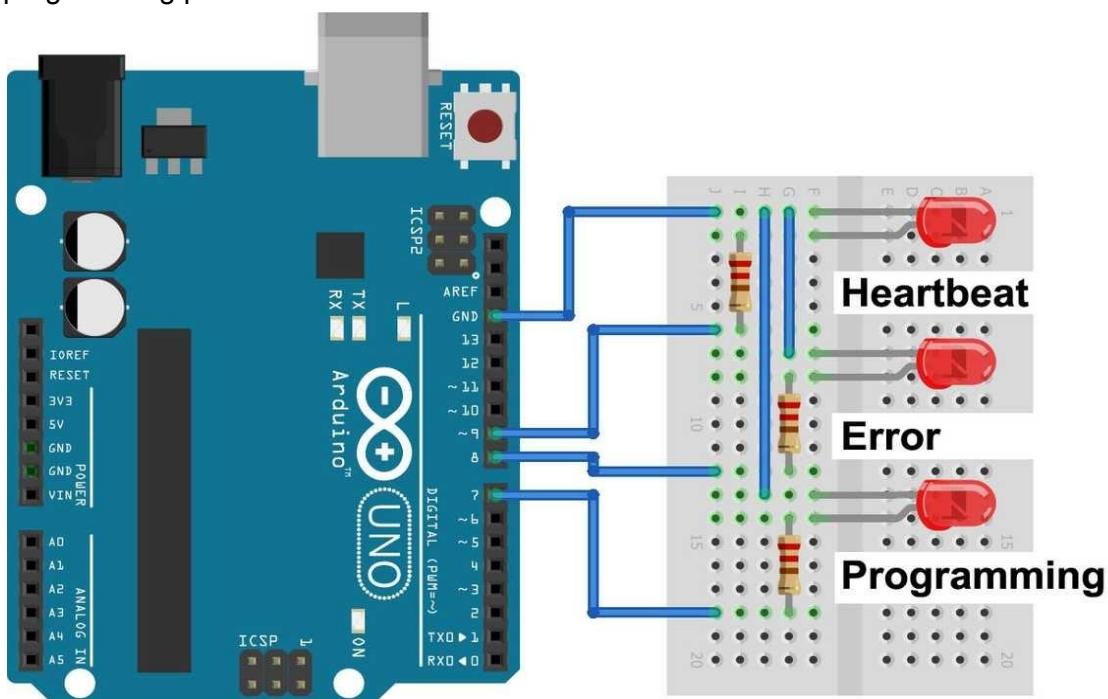
The programming process uses VCC, GND and four data pins. Three pins connect MISO, MOSI and SCK between the programming micro and the target micro, the fourth pin from the programming micro goes to the reset pin of the target.

### **STEP 3:** Load the sketch

The Arduino that you will use as programmer needs a specific sketch. You find it under Examples > 11. ArduinoISP > ArduinoISP .



Going through the lines of the sketch you find a number of parameters that need to be set according to the target board. These parameters are, however, set by a specific file available for each bootloader/board supported by the Arduino Software (IDE). Other parameters are clearly explained by the comments and should be changed only if you know what you are doing. The sketch also supports three LEDs that give you a visual feedback about the programming process.



To upload the sketch to your Arduino board - the one that you will use as the programmer - you need to select board type and port, then you can proceed as usual.

#### **STEP 4: CONNECT YOUR ATMEGA328**

Now connect your ATmega to your UNO as follows:

Arduino 1.      Arduino 2.

UNO 5v    ---> UNO 5v (VCC)

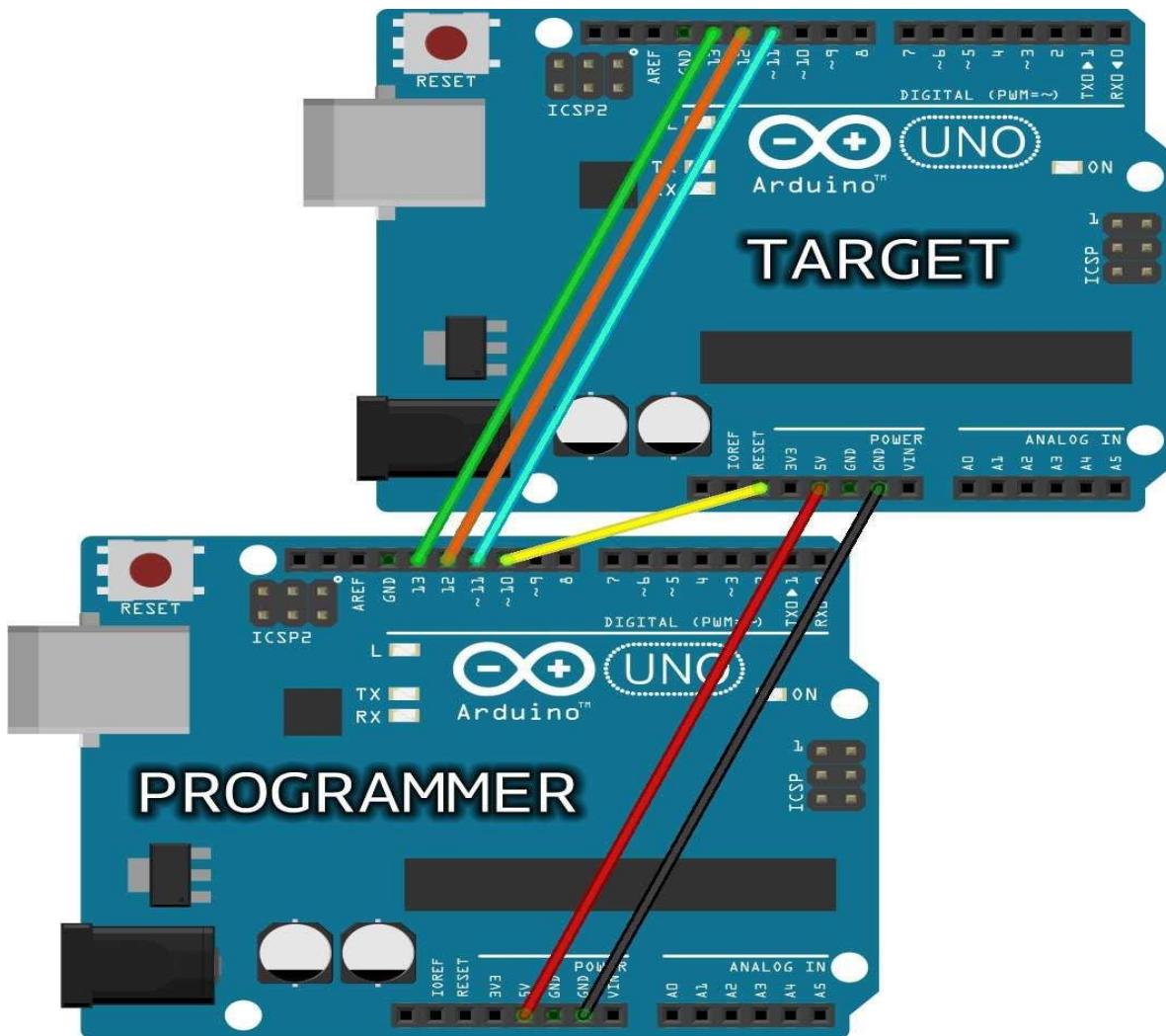
UNO GND    ---> UNO GND (GND)

UNO pin 10 ---> RESET

UNO pin 11 ---> UNO pin 11 (MOSI)

UNO pin 12 ---> UNO pin 12 (MISO)

UNO pin 13 ---> UNO pin 13 (SCK)



Make sure that you don't have anything else connected to the ATmega pins used above.

#### **STEP 5: Program the bootloader**

If all the wires are set, you need to switch to the board type you want to program with the bootloader. This step is mandatory to select the proper bootloader code and the fuses configurations. The programming procedure checks the signature of the microcontroller

before any writing action is taken, but many boards share the same microcontroller and each board has its own bootloader. The port remains the one of the ArduinoISP.

Choose "Burn bootloader" under tools and wait for the confirmation message in the lower part of the Arduino Software (IDE) interface. If you connected the LEDs you may follow the process visually.



### How to test arduino IC :-

We are only using 5v, 3.3v, GND, Digital 4, Digital 7, Digital 9, Digital10, Tux and Rx pins for this project. So we will only do so many pinky tests.

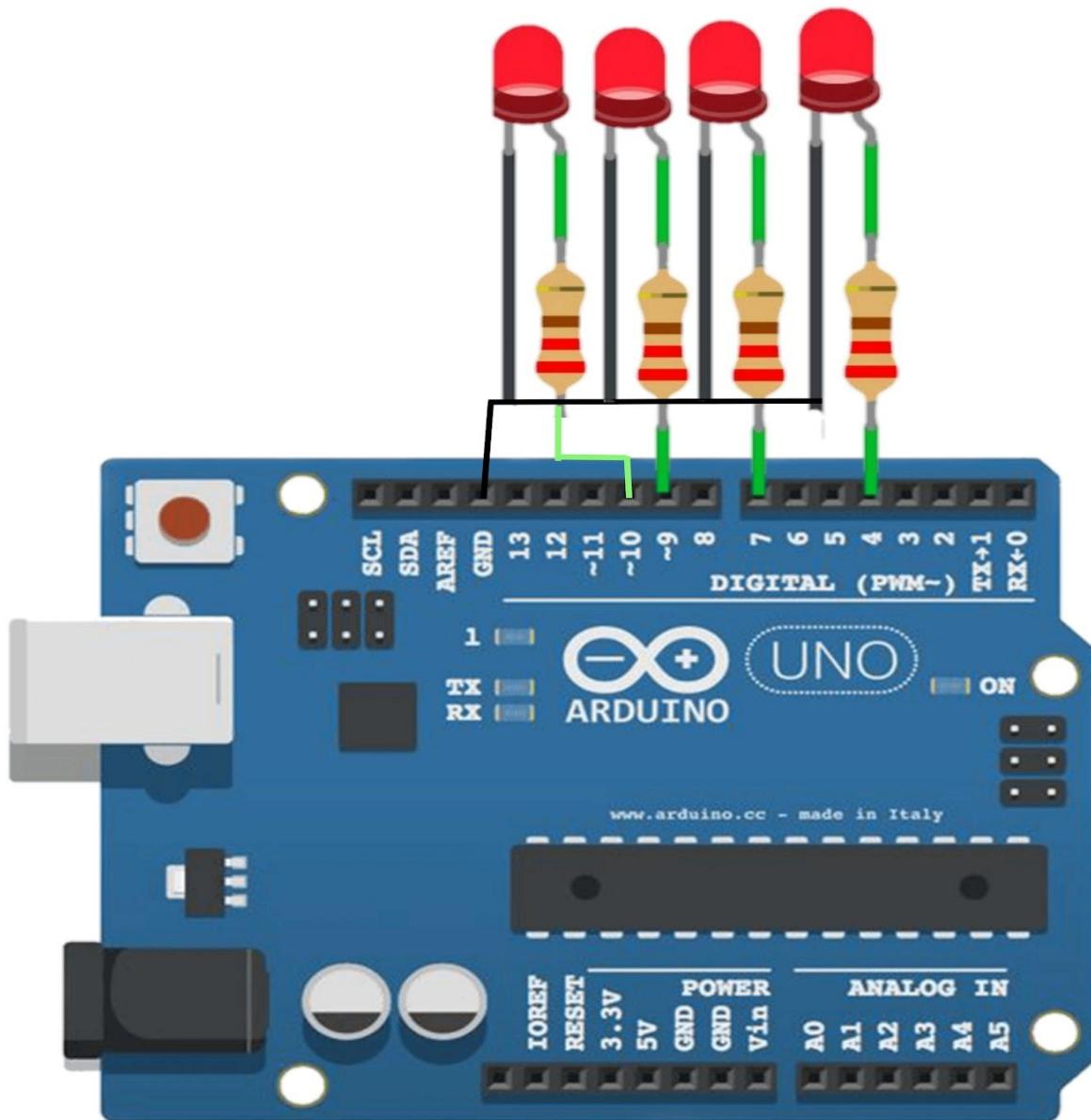
#### STEP 1: power supply pin test

5v, 3.3v, GND:-This pin is the power supply pin. So we tested with multimeter of people.



### **STEP 2:** Digital pin test

Digital 4, Digital 7, Digital 9, Digital10:-This pin is the Digital pin. So we will upload and check the LED blinking program in arduino UNO.



### **STEP 3:** TX,RX pin test

Tux, Rx:-This pin is the transmitter and receiver pin. There is 4 STEP to test the transmitter and receiver pins.

#### **1: DISCONNECT POWER**

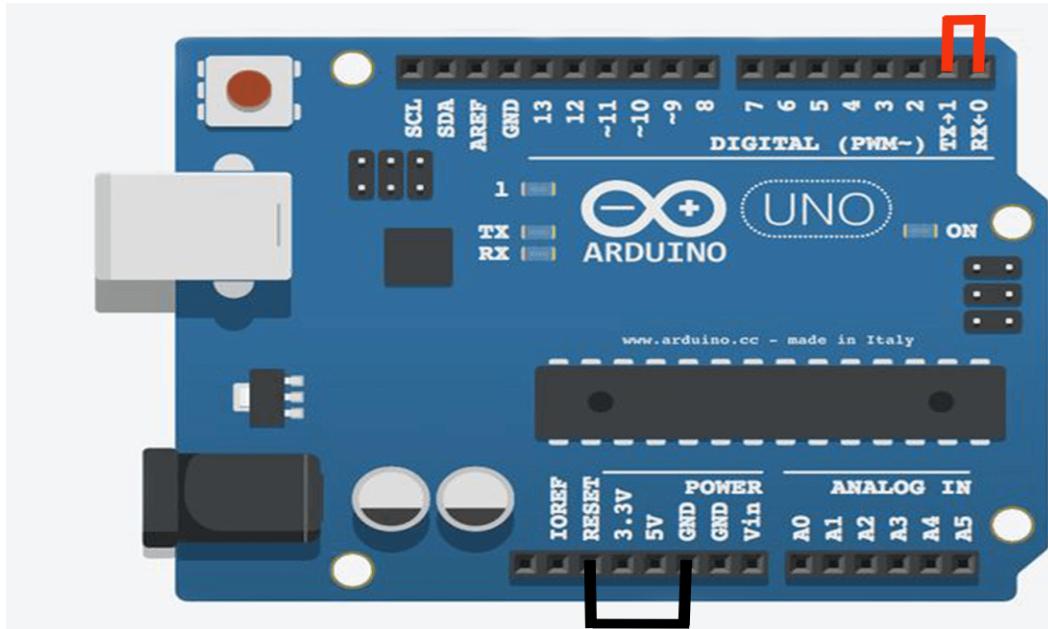
Disconnect power from the board.

Also remove all connections and shields from the board.

#### **2: CONNECT JUMPER WIRES**

1.)Force the processor to remain in reset by connecting a jumper from RESET to GND(black wire in photo).

2.)Connect a jumper wire from TX pin(digital pin 1) to the RX pin(digital pin 0). (Red wire in photo)



### 3: CONNECTING BOARD TO PC

Open Arduino IDE .

Start terminal application like serial monitor after selecting your port.

Send data by typing.Whatever you write should be echoed back.

If exactly what you write is echoed back then the board passes the test.And so the host computer hardware driver,USB cable, and USB to serial converter are all working.

### 4: IF NOTHING ECHOES BACK!

If nothing is echoed back,check:

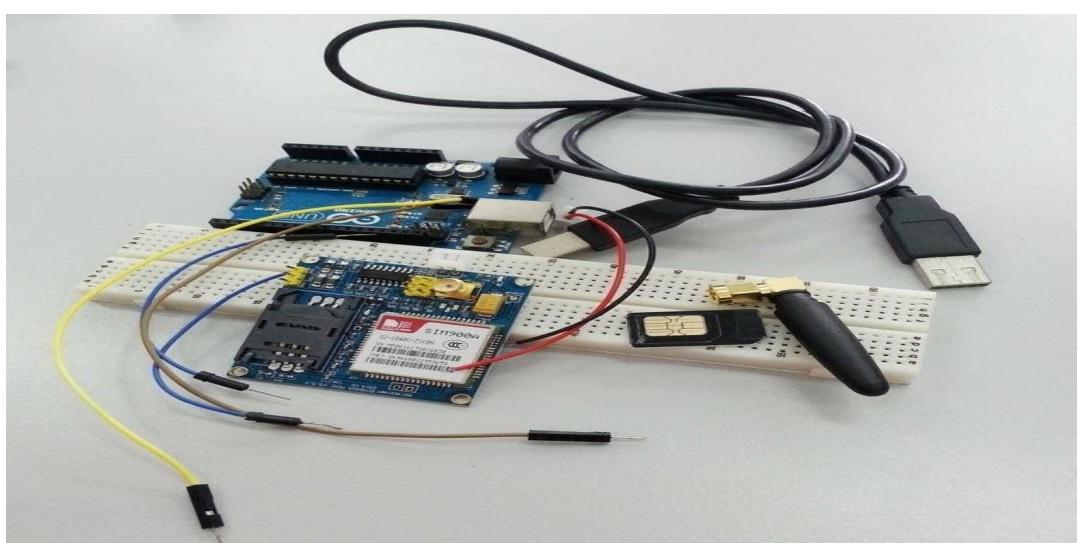
- \* You have the correct com port.

- \* Try a different cable.Some cheap USB cables only have power wires and not data wires.

- \* Check the device driver for the Arduino is installed .

After unlocking the bootloader of ic, ic is working correctly. We have done all this to find out.

## How to test SIM 900A GSM Module :-



### STEP 1: MATERIAL PREPARATION

In this tutorial, you will need :

1. GSM SIM900A (MINI V3.9.2)

2. Arduino Uno Board and USB
3. Jumper Wire
4. Power adapter 5V
5. SIM card
6. Breadboard

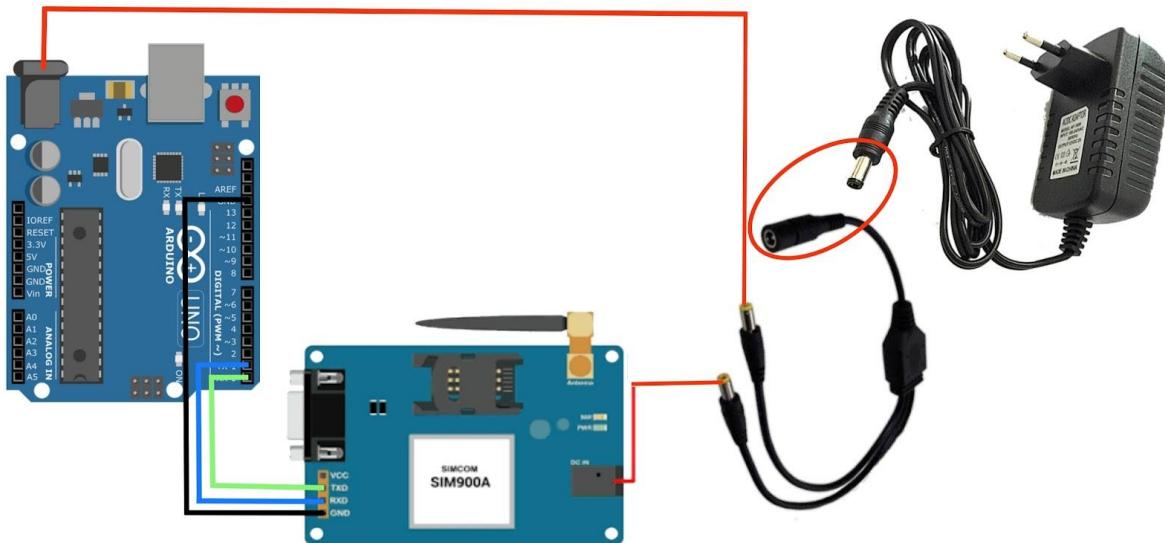
### **STEP 2: BOOTING UP SIM900A**



1. Insert your SIM card to GSM module and lock it. (picture 1 and 2)
2. power up your gsm by connecting it to Arduino's 5V and GND (picture 3)
3. Connect the Antenna (picture 4)
4. Now wait for some time (say 1 minute) and see the blinking rate of 'status LED' or 'network LED' (D6, refer picture 5) //GSM module will take some time to establish connection with mobile network//
5. Once the connection is established successfully, the status/network LED will blink continuously every 3 seconds. You may try making a call to the mobile number of the sim card inside GSM module. If you hear a ring back, the gsm module has successfully established network connection.

### **STEP 3: PIN CONNECTION**

You can see a TTL pin with 3VR, 3VT, 5Vr, 5VT, VCC and GND on your sim900a near your power supply pin. You have to connect GSM's 5VT to Arduino D9 and GSM's 5VR to Arduino's D10 for serial communication between arduino and sim900a module.



#### **STEP 4:** BASIC Arduino Code to receive a Message

```

char str[15];
int i = 0;

//Message Example: #Hello World*

void setup()
{
    Serial.begin(9600);
    Serial.println("AT+CNMI=2,2,0,0,0");
    delay(2000);
    Serial.println("AT+CMGF=1");
    delay(1000);
}

void loop()
{
    i = 0;
    delay(1000);
}

void serialEvent()
{
    while(Serial.available())
    {
        if(Serial.find("#"))
        {
            delay(1000);
            while (Serial.available())
            {
                char inChar=Serial.read();

```

```
str[i++]=inChar;
if(inChar=='*')
{
    Serial.println((String)str);
    return;
}
}
}
}
}
}
```

**Step 5:-** Power up your board.

The Arduino Uno, Mega, Duemilanove and Arduino Nano automatically draw power from either, the USB connection to the computer or an external power supply. If you are using an Arduino Diecimila, you have to make sure that the board is configured to draw power from the USB connection. The power source is selected with a jumper, a small piece of plastic that fits onto two of the three pins between the USB and power jacks. Check that it is on the two pins closest to the USB port.

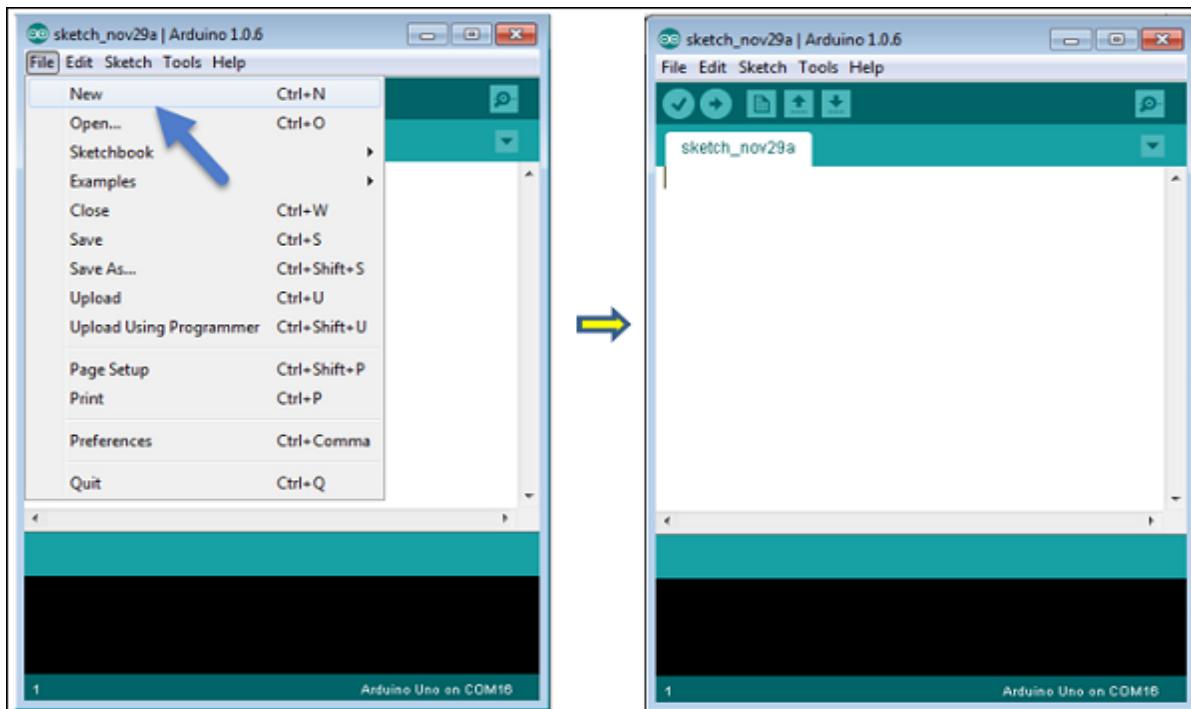
Connect the Arduino board to your computer using the USB cable. The green power LED (labeled PWR) should glow.

**Step 6:-** Open your first project.

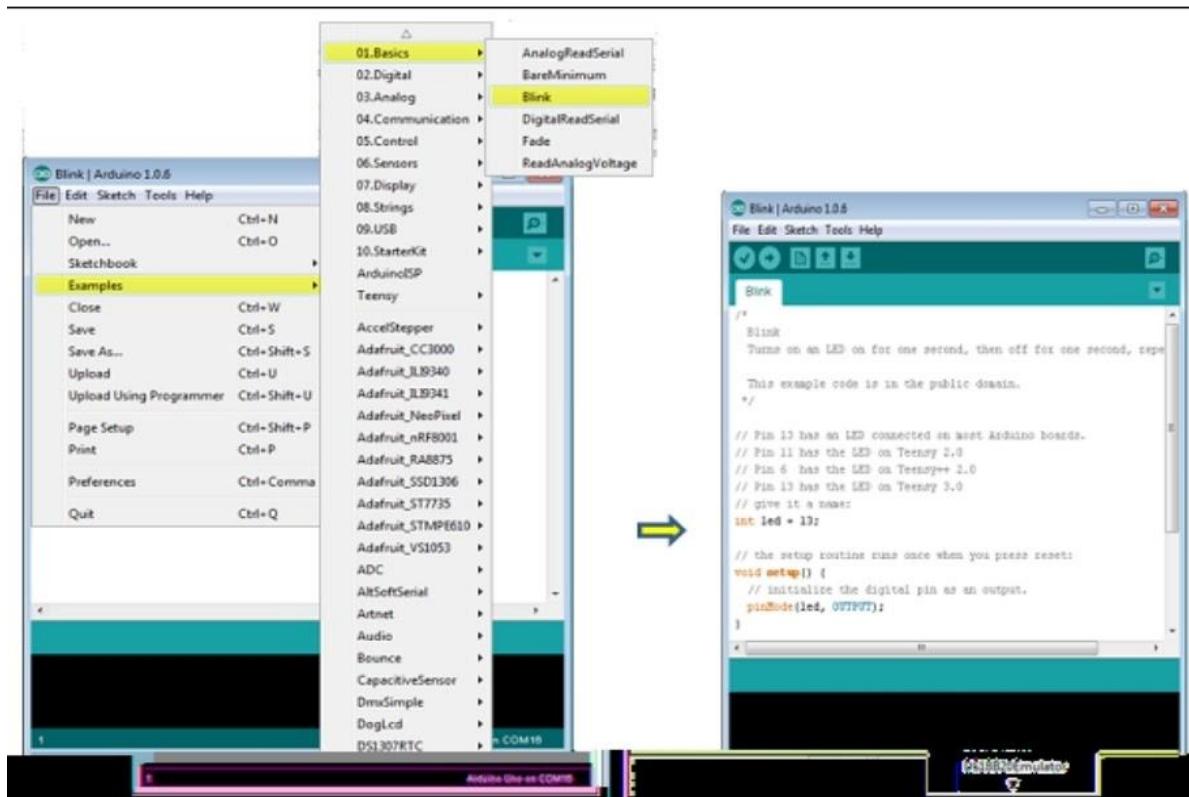
Once the software starts, you have two options – Create a new project.

Open an existing project example.

To create a new project, select File → New.



To open an existing project example, select File → Example → Basics → GSM.

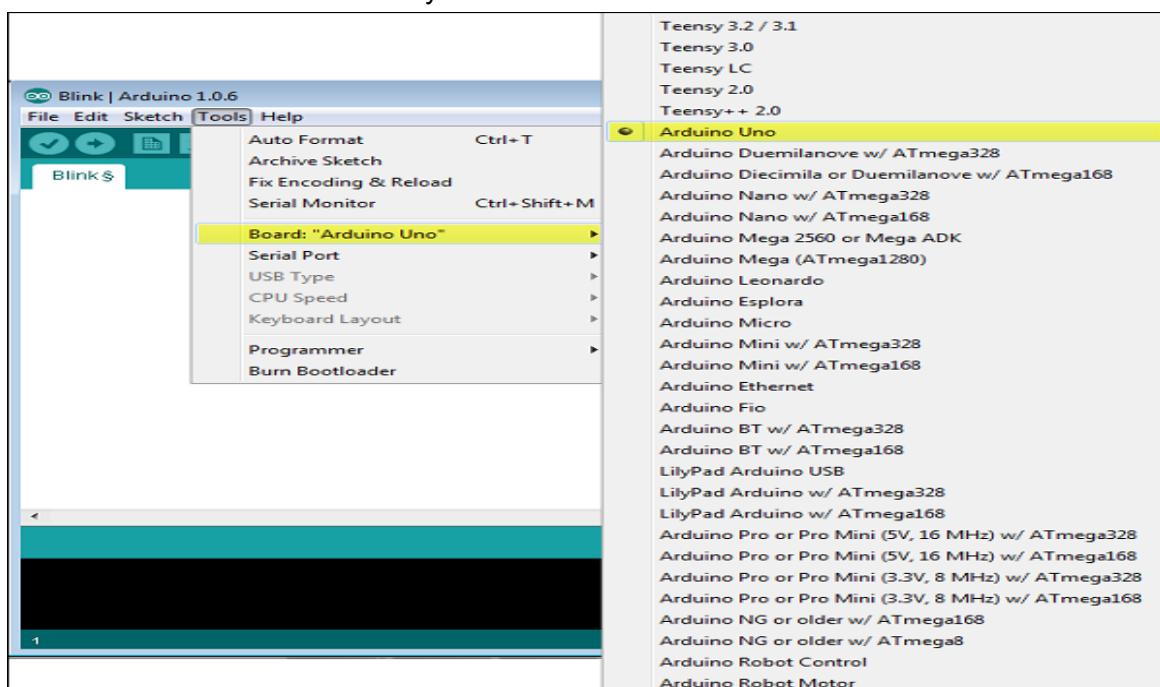


Here, we are selecting just one of the examples with the name Blink. It turns the LED on and off with some time delay. You can select any other example from the list.

### Step 7:- Select your Arduino board.

To avoid any error while uploading your program to the board, you must select the correct Arduino board name, which matches with the board connected to your computer.

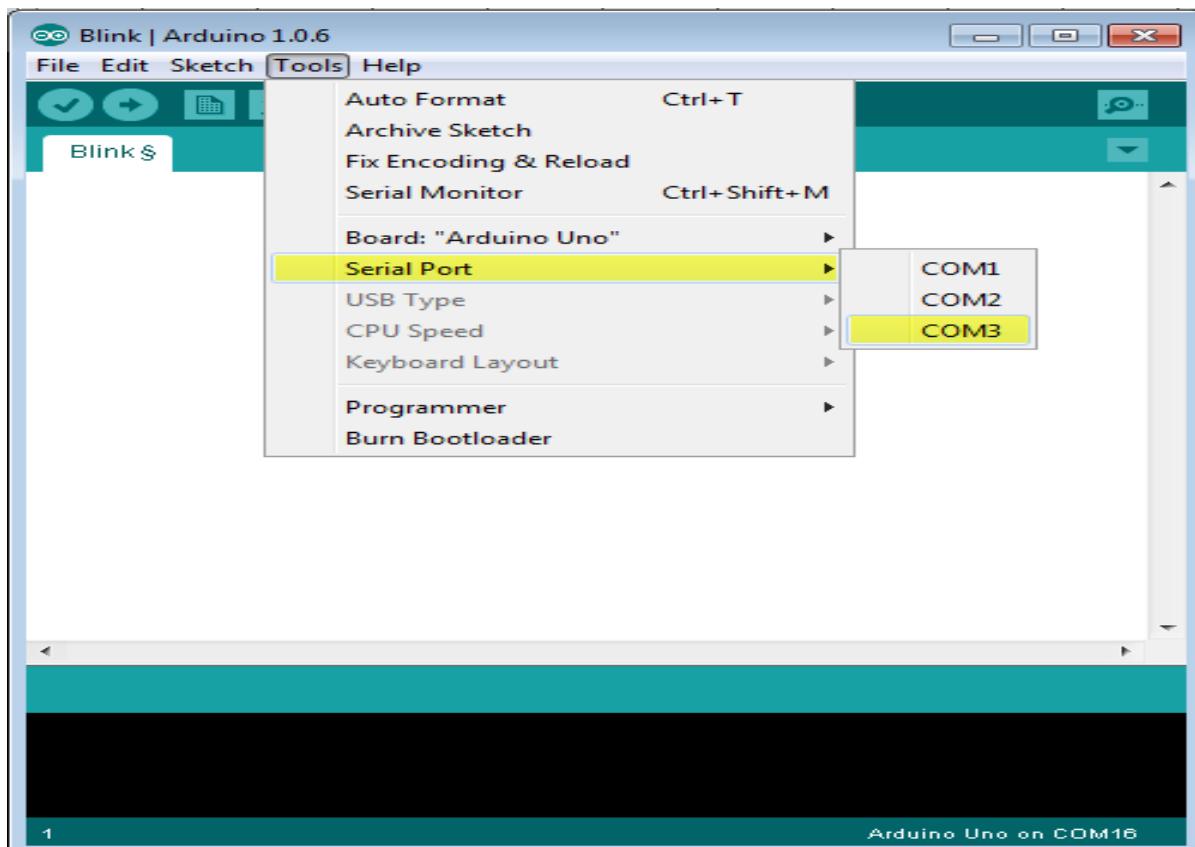
Go to Tools → Board and select your board.



Here, we have selected Arduino Uno board according to our tutorial, but you must select the name matching the board that you are using.

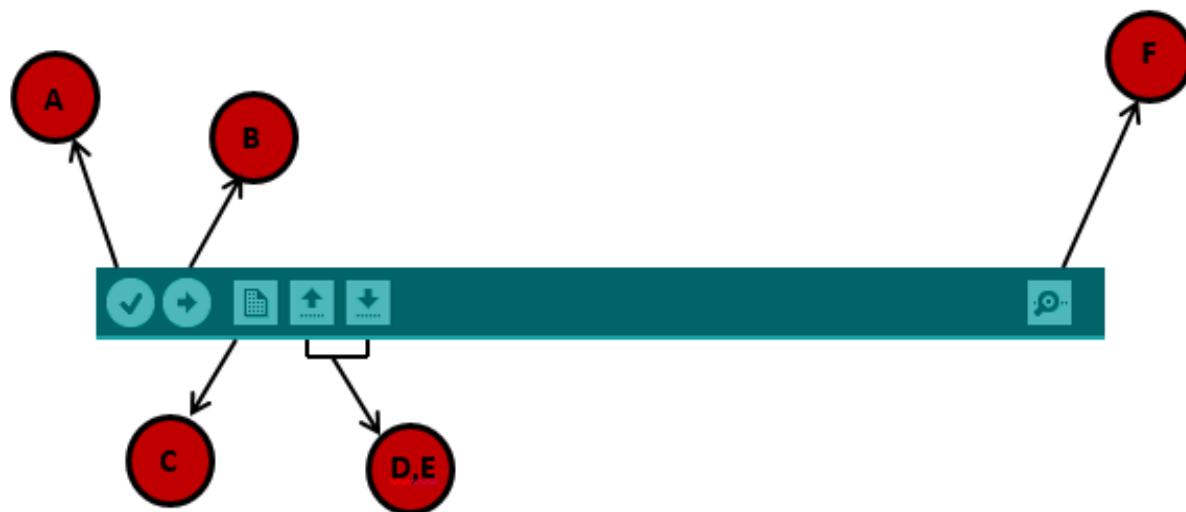
**Step 8:-** Select your serial port.

Select the serial device of the Arduino board. Go to Tools → Serial Port menu. This is likely to be COM3 or higher (COM1 and COM2 are usually reserved for hardware serial ports). To find out, you can disconnect your Arduino board and re-open the menu, the entry that disappears should be of the Arduino board. Reconnect the board and select that serial port.



**Step 9:-** Upload the program to your board.

Before explaining how we can upload our program to the board, we must demonstrate the function of each symbol appearing in the Arduino IDE toolbar.



A – Used to check if there is any compilation error.

B – Used to upload a program to the Arduino board.

C – Shortcut used to create a new sketch.

D – Used to directly open one of the example sketch.

E – Used to save your sketch.

F – Serial monitor used to receive serial data from the board and send the serial data to the board.

Now, simply click the "Upload" button in the environment. Wait a few seconds; you will see the RX and TX LEDs on the board, flashing. If the upload is successful, the message "Done uploading" will appear in the status bar.

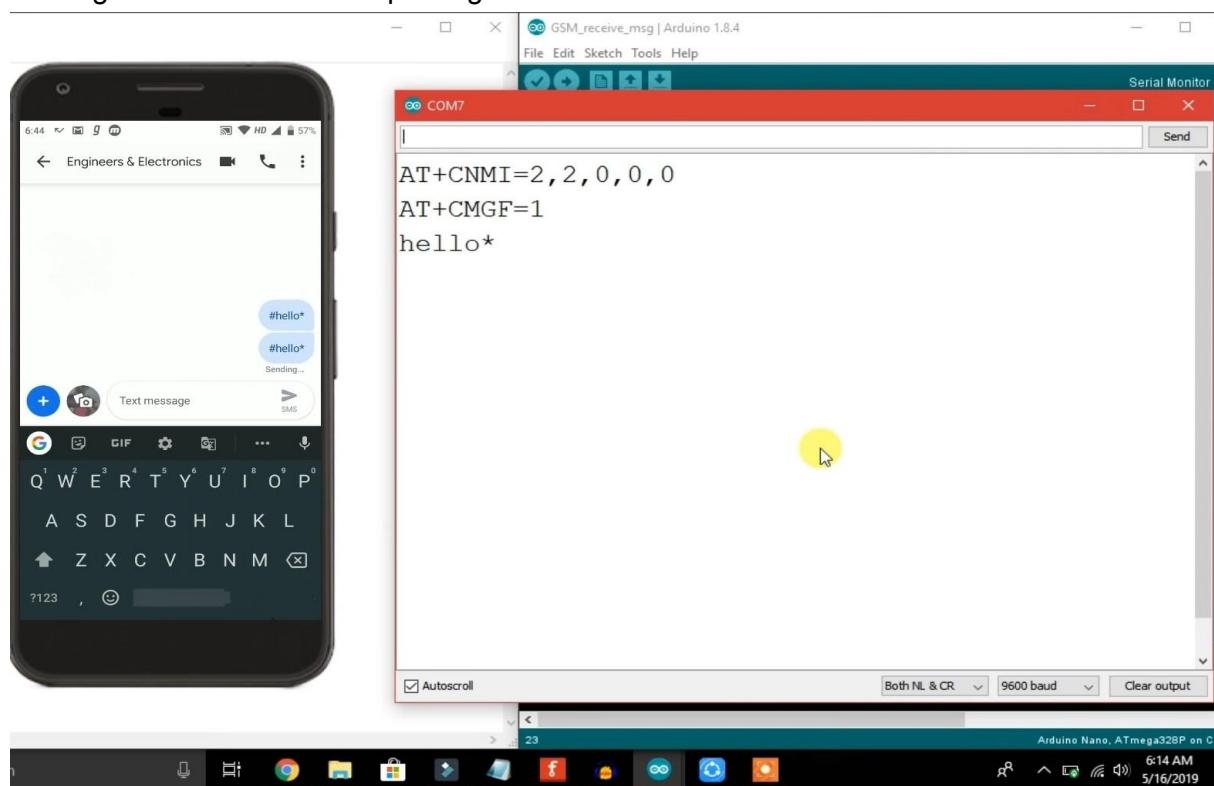
Note – If you have an Arduino Mini, NG, or other board, you need to press the reset button physically on the board, immediately before clicking the upload button on the Arduino Software.

#### **STEP 10: SERIAL MONITOR**

After you has successfully uploaded your source code, open your serial monitor. Serial monitor will display as shown in the picture above.

#### **STEP 11: RESULT**

Message sent from mobile is printing on serial monitor.



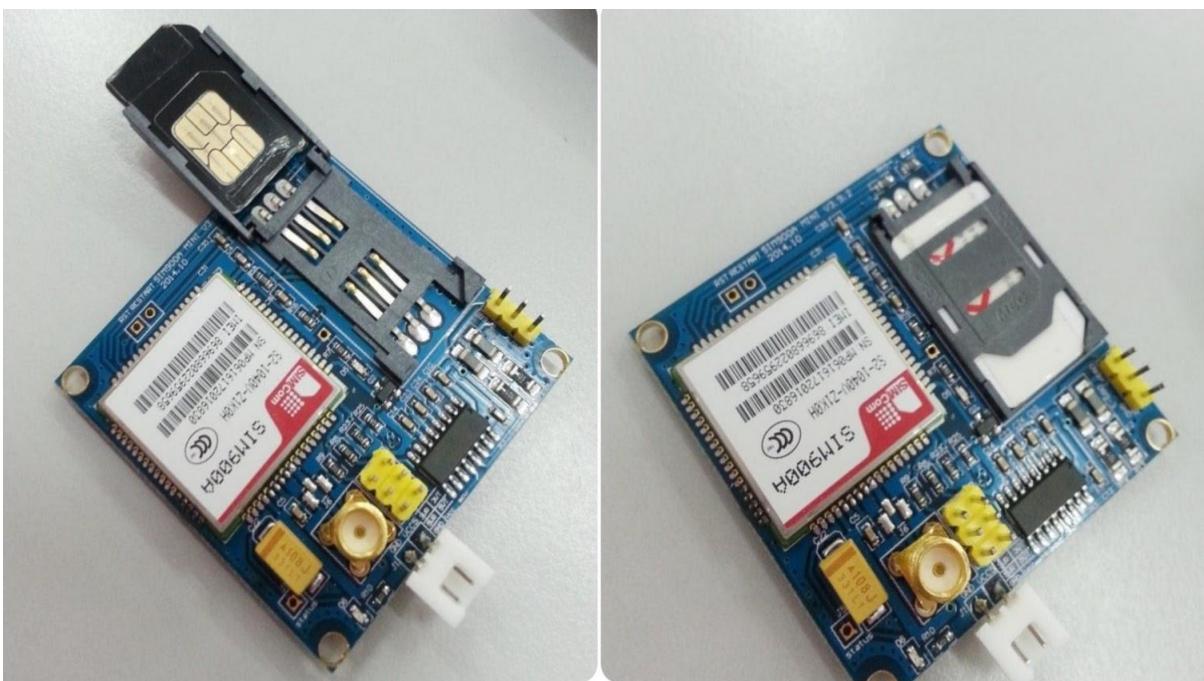
## **Test Project circuit and code:-**

### **STEP 1: MATERIAL PREPARATION**

In this tutorial, you will need :

1. GSM SIM900A (MINI V3.9.2)
2. Arduino Uno Board and USB
3. Jumper Wire
4. Power adapter 12V
5. SIM card
6. Breadboard
7. 4 LED (different colors)
8. LDR sensor module

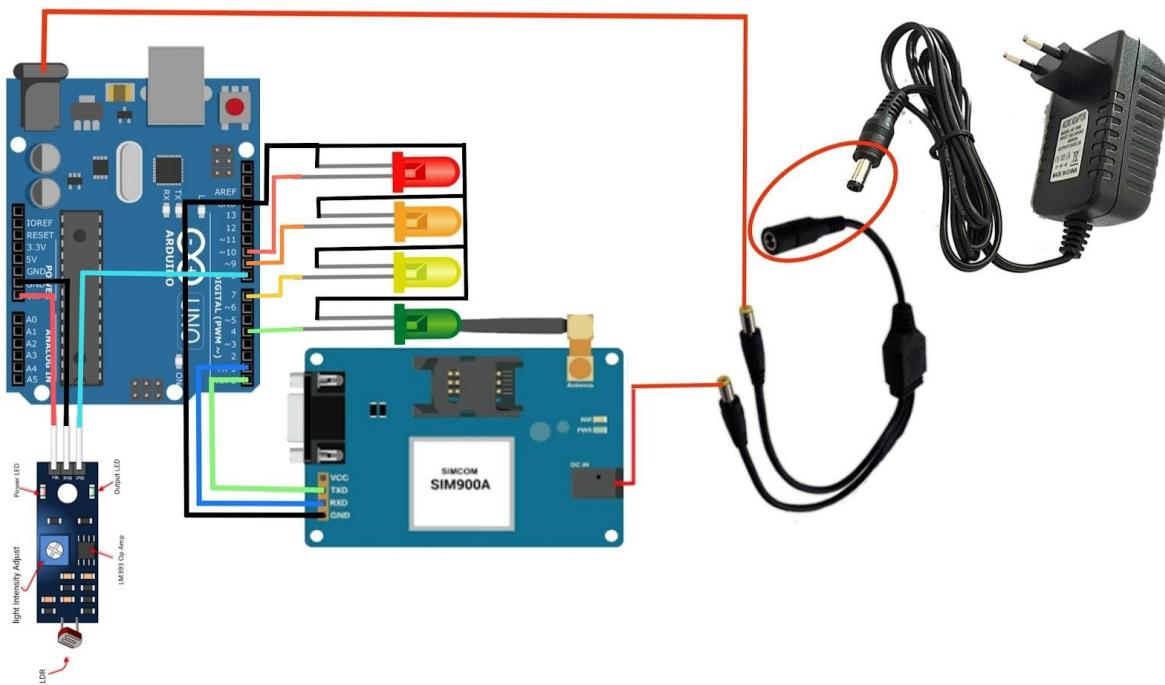
### **STEP 2: BOOTING UP SIM900A**



1. Insert your SIM card to GSM module and lock it. (picture 1 and 2)
2. power up your gsm by connecting it to Arduino's 5V and GND (picture 3)
3. Connect the Antenna (picture 4)
4. Now wait for some time (say 1 minute) and see the blinking rate of 'status LED' or 'network LED' (D6, refer picture 5) //GSM module will take some time to establish connection with mobile network//
5. Once the connection is established successfully, the status/network LED will blink continuously every 3 seconds. You may try making a call to the mobile number of the sim card inside GSM module. If you hear a ring back, the gsm module has successfully established network connection.

### **STEP 3: PIN CONNECTION**

You can see a TTL pin with 3VR, 3VT, 5Vr, 5VT, VCC and GND on your sim900a near your power supply pin. You have to connect GSM's 5VT to Arduino D9 and GSM's 5VR to Arduino's D10 for serial communication between arduino and sim900a module.



#### **Step 4:- Power up your board.**

The Arduino Uno, Mega, Duemilanove and Arduino Nano automatically draw power from either, the USB connection to the computer or an external power supply. If you are using an Arduino Diecimila, you have to make sure that the board is configured to draw power from the USB connection. The power source is selected with a jumper, a small piece of plastic that fits onto two of the three pins between the USB and power jacks. Check that it is on the two pins closest to the USB port.

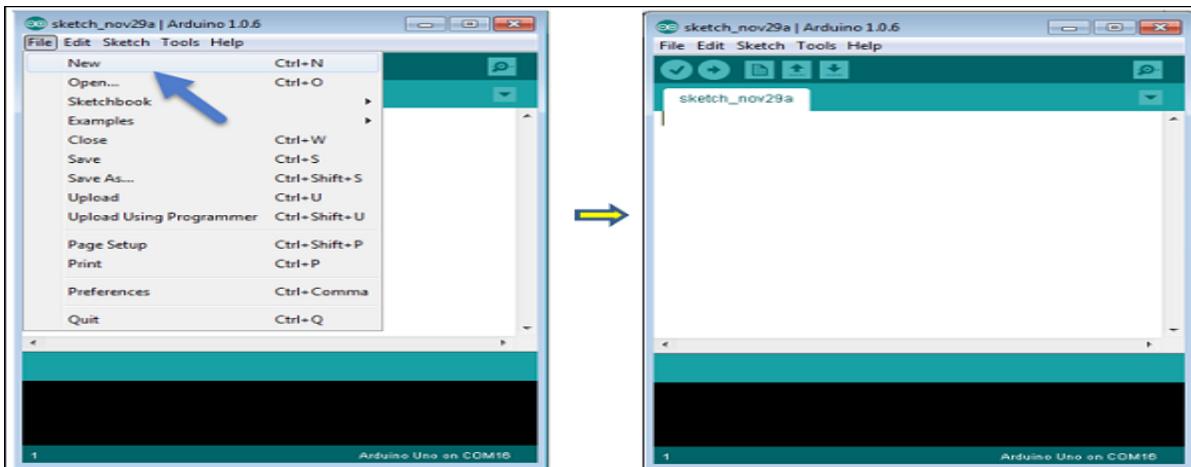
Connect the Arduino board to your computer using the USB cable. The green power LED (labeled PWR) should glow.

#### **Step 5:- Open your first project.**

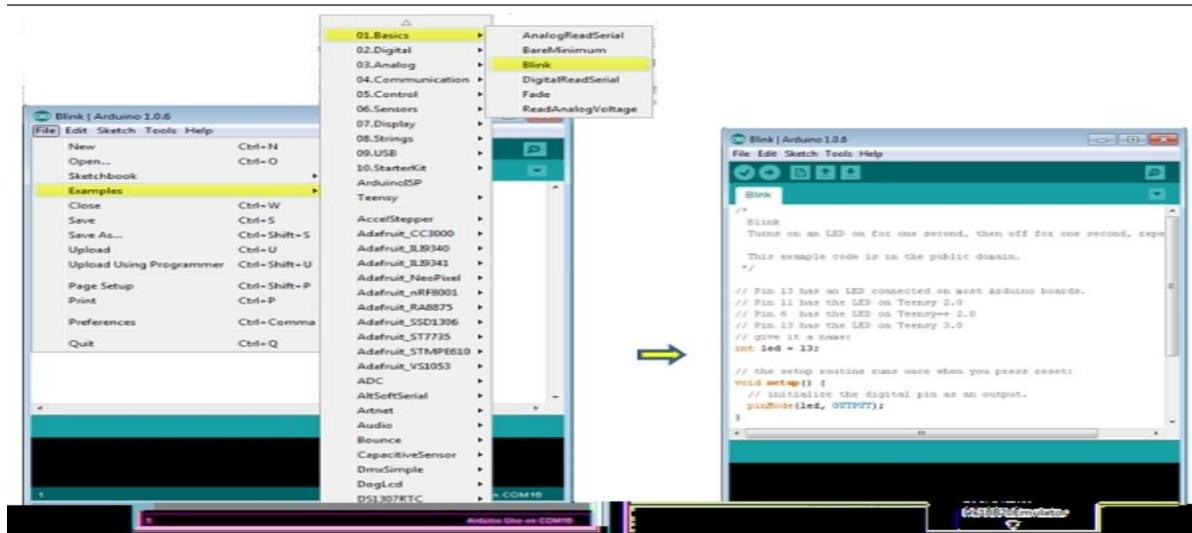
Once the software starts, you have two options – Create a new project.

Open an existing project example.

To create a new project, select File → New.



To open an existing project example, select File → Example → GSM → ReceiveSMS → Edit code



We will take an example from GSM, that example will be the message received and we will edit it according to our project, after editing the court will look like the court below, this is also created by editing the code and copy pasting we have this Will use the code in the project

### Code:-

```
#define Fan 7
#define Light 9
#define TV 10
#define bulb 4
#define ldr 8
int temp=0,i=0;
int led=13;
char str[15];

void setup()
{
  initialize_system();
}

void loop()
{
  GSM_system();
  void serialEvent();
  void check();
  LDR_system();
}

void initialize_system()
{
  pinMode(led, OUTPUT);
  pinMode(Fan, OUTPUT);
```

```

pinMode(Light, OUTPUT);
pinMode(TV, OUTPUT);
pinMode(bulb, OUTPUT);
pinMode(Idr, INPUT);
Serial.begin(9600);

Serial.println("AT+CNMI=2,2,0,0,0");
delay(500);
Serial.println("AT+CMGF=1");
delay(1000);
}

void GSM_system()
{
if(temp==1)
{
check();
temp=0;
i=0;
delay(1000);
}
}

void serialEvent()
{
while(Serial.available())
{
if(Serial.find("#A."))
{
digitalWrite(led, HIGH);
delay(1000);
digitalWrite(led, LOW);
while (Serial.available())
{
char inChar=Serial.read();
str[i++]=inChar;
if(inChar=='*')
{
temp=1;
return;
}
}
}
}
}

void check()
{
if(!(strncmp(str,"tv on",5)))

```

```

{
  digitalWrite(TV, HIGH);
}
else if(!(strncmp(str,"tv off",6)))
{
  digitalWrite(TV, LOW);
}
if(!(strncmp(str,"bulb on",7)))
{
  digitalWrite(bulb, HIGH);
}
else if(!(strncmp(str,"bulb off",7)))
{
  digitalWrite(bulb, LOW);
}
else if(!(strncmp(str,"fan on",6)))
{
  digitalWrite(Fan, HIGH);
}
else if(!(strncmp(str,"fan off",7)))
{
  digitalWrite(Fan, LOW);
}
  else if(!(strncmp(str,"all on",6)))
{
  digitalWrite(Fan, HIGH);
  digitalWrite(TV, HIGH);
  digitalWrite(bulb, HIGH);
}
else if(!(strncmp(str,"all off",7)))
{
  digitalWrite(Fan, LOW);
  digitalWrite(TV, LOW);
  digitalWrite(bulb, LOW);
}
}
}

```

```

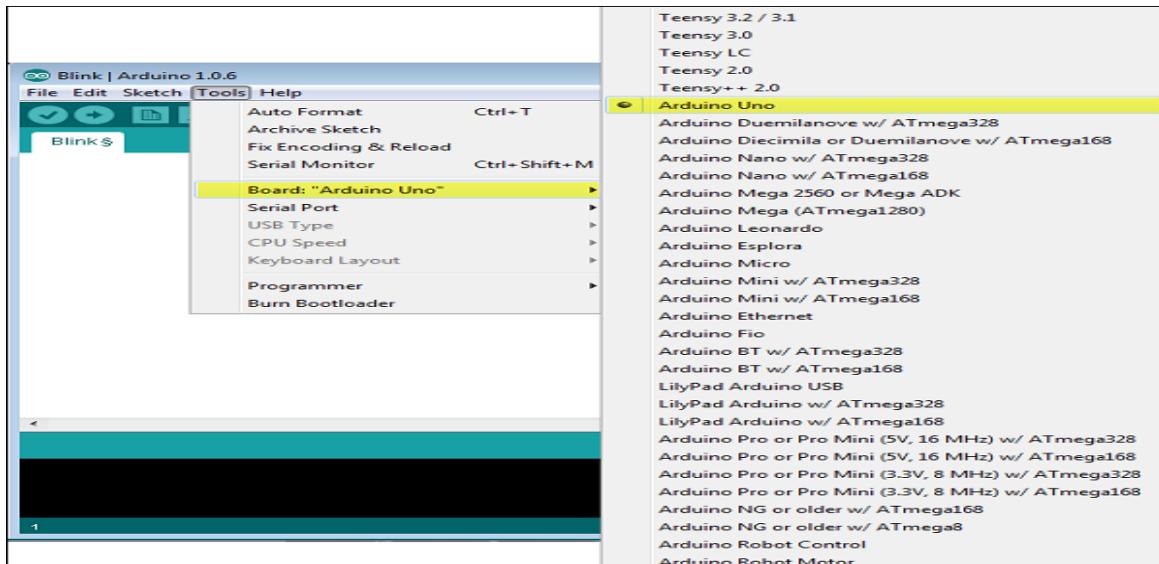
void LDR_system()
{
  int temp=digitalRead(Idr);
  if(temp==HIGH)
digitalWrite(Light,HIGH);
else
digitalWrite(Light,LOW);
}

```

**Step 6:-** Select your Arduino board.

To avoid any error while uploading your program to the board, you must select the correct Arduino board name, which matches with the board connected to your computer.

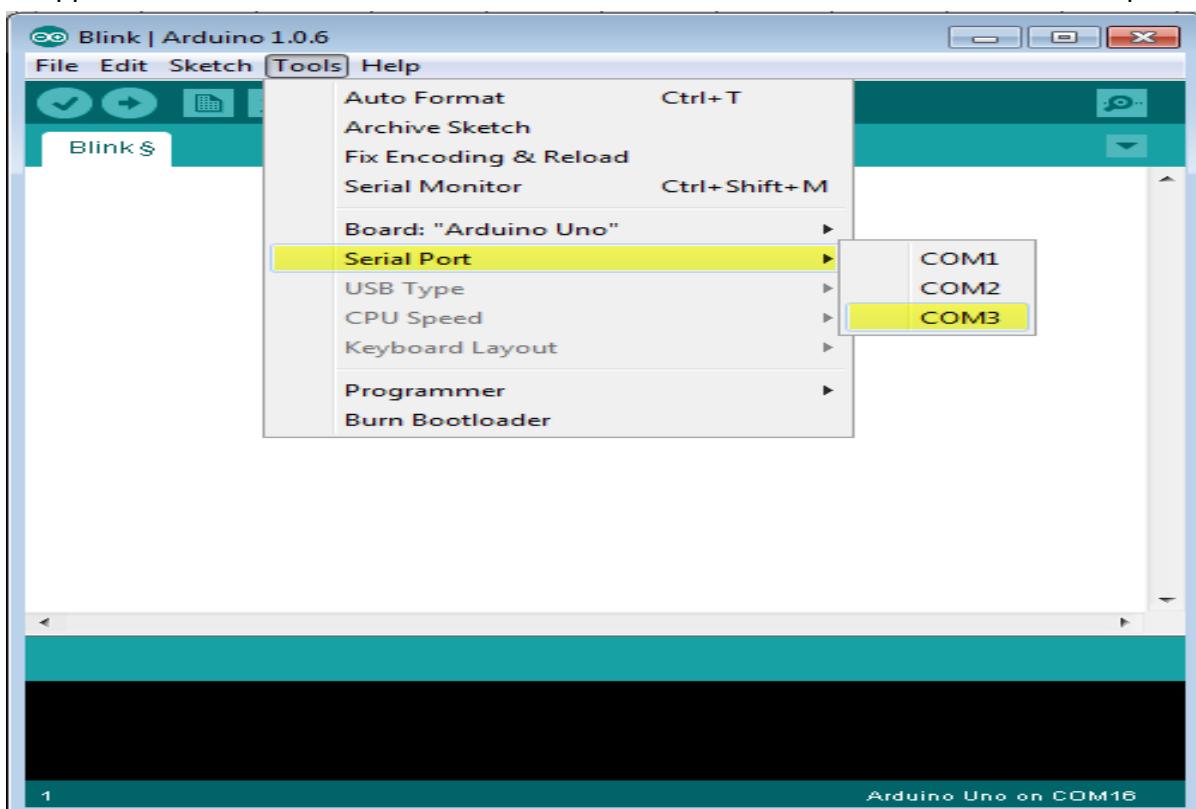
Go to Tools → Board and select your board.



Here, we have selected Arduino Uno board according to our tutorial, but you must select the name matching the board that you are using.

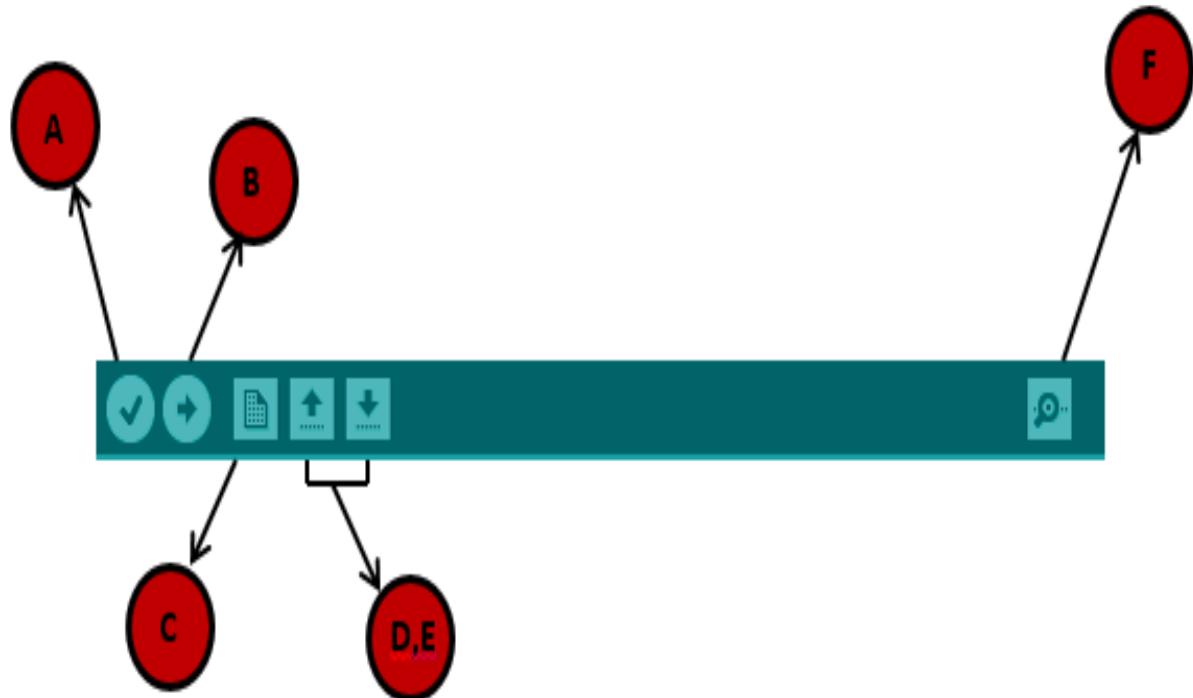
#### **Step 7:-** Select your serial port.

Select the serial device of the Arduino board. Go to Tools → Serial Port menu. This is likely to be COM3 or higher (COM1 and COM2 are usually reserved for hardware serial ports). To find out, you can disconnect your Arduino board and re-open the menu, the entry that disappears should be of the Arduino board. Reconnect the board and select that serial port.



**Step 8:-** Upload the program to your board.

Before explaining how we can upload our program to the board, we must demonstrate the function of each symbol appearing in the Arduino IDE toolbar.



A – Used to check if there is any compilation error.

B – Used to upload a program to the Arduino board.

C – Shortcut used to create a new sketch.

D – Used to directly open one of the example sketch.

E – Used to save your sketch.

F – Serial monitor used to receive serial data from the board and send the serial data to the board.

Now, simply click the "Upload" button in the environment. Wait a few seconds; you will see the RX and TX LEDs on the board, flashing. If the upload is successful, the message "Done uploading" will appear in the status bar.

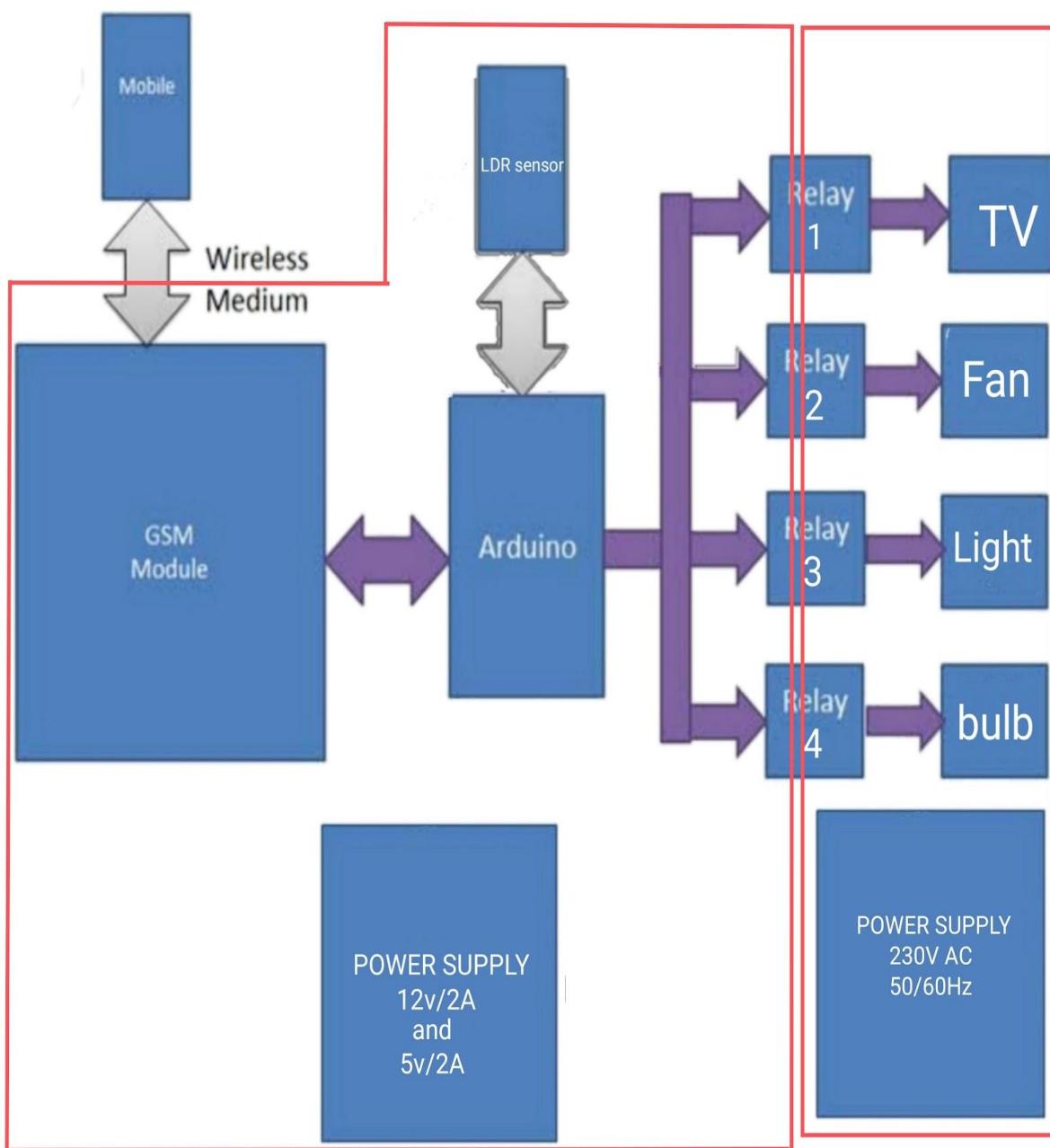
Note – If you have an Arduino Mini, NG, or other board, you need to press the reset button physically on the board, immediately before clicking the upload button on the Arduino Software.

**Step 8:-Result**

How we tested it is explained in the video below

<https://photos.app.goo.gl/J75SEET5sqJrMgYw8>

**Block diagram:-**

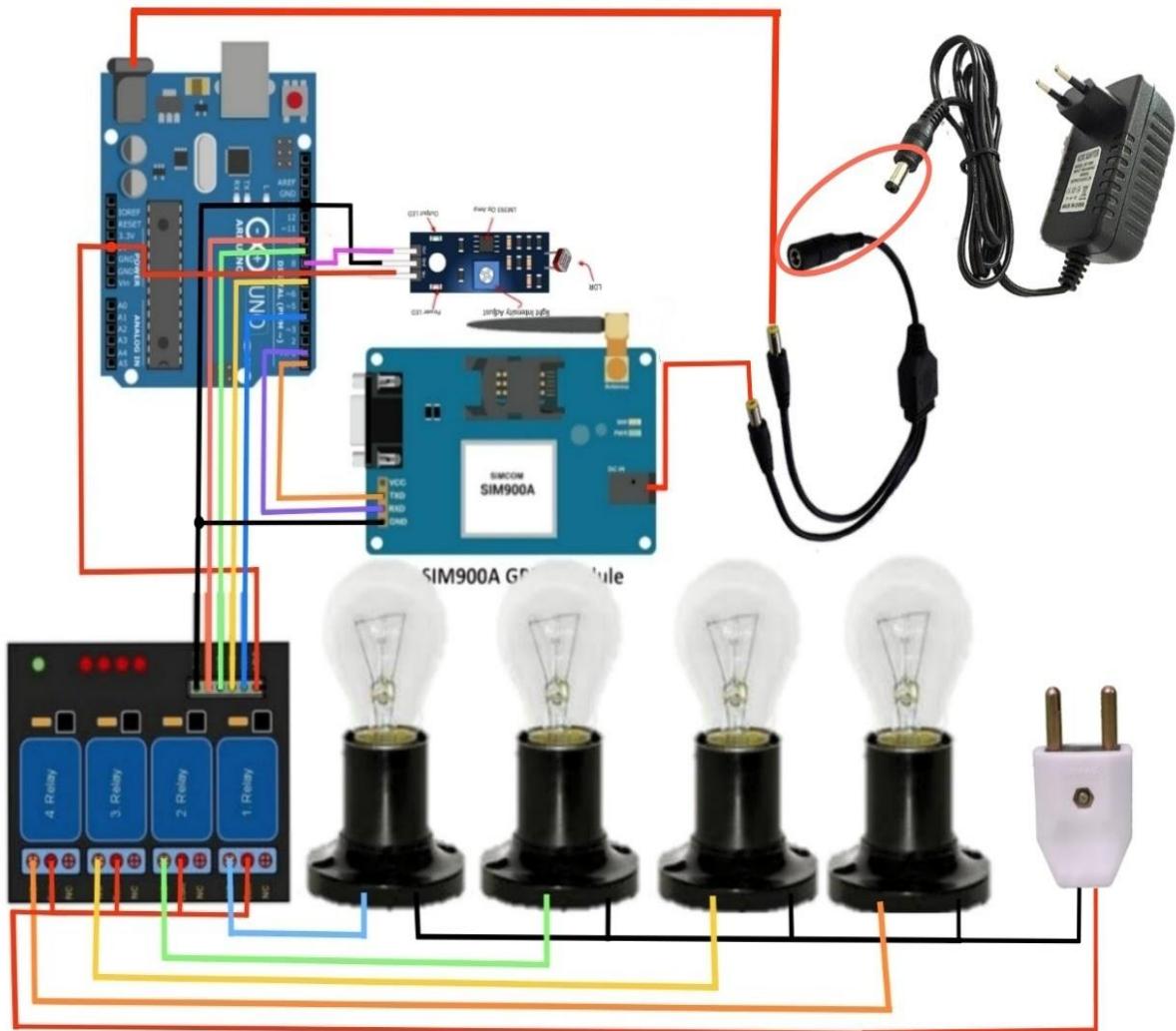


### **what is Block diagram ?**

A block diagram is a diagram of a system in which the principal parts or functions are represented by blocks connected by lines that show the relationships of the blocks. They are heavily used in engineering in hardware design, electronic design, software design, and process flow diagrams.

The diagram above describes how the circuit diagram will be connected.

### **Arduino UNO diagram:-**



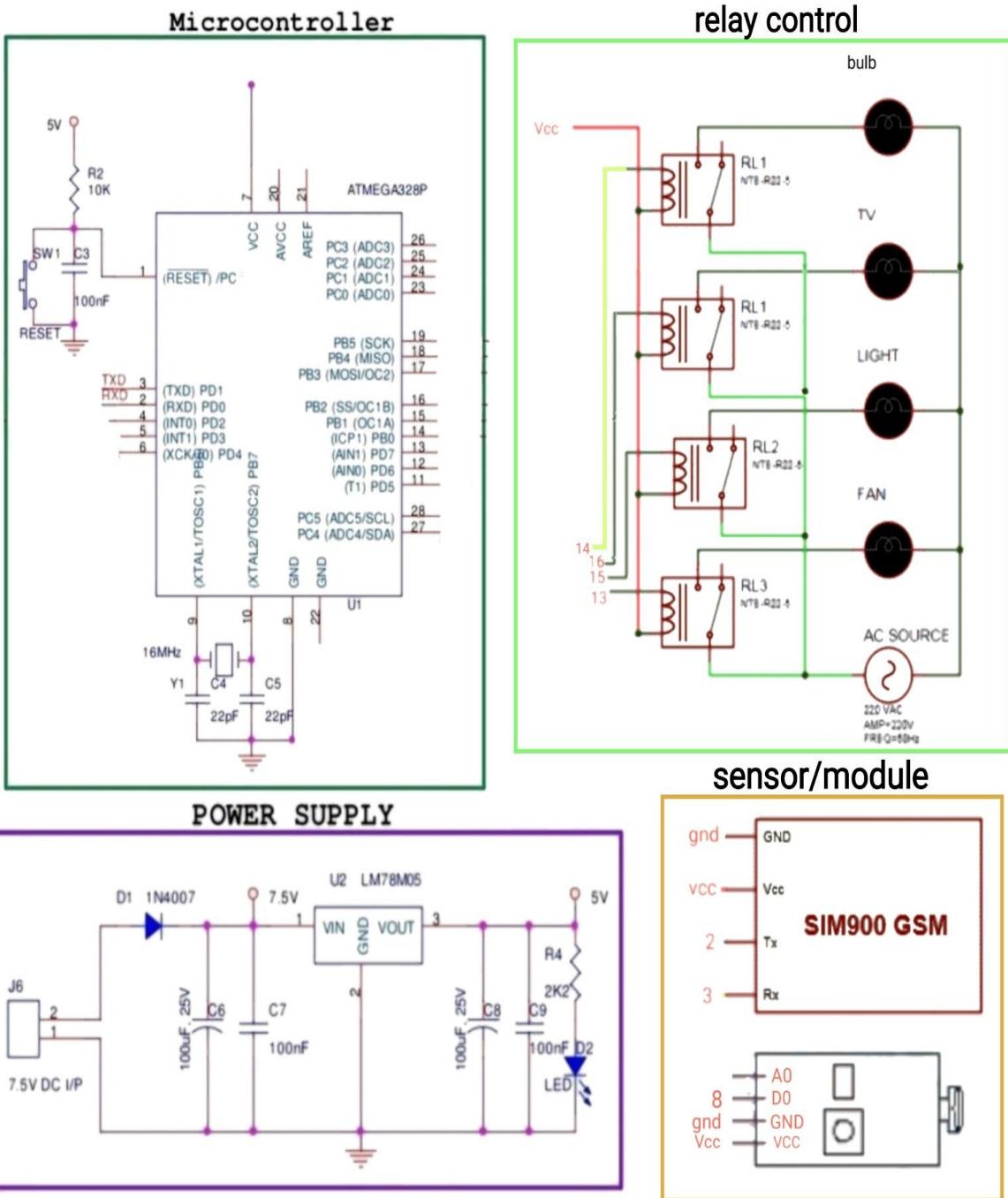
The Arduino can be powered via VIN pin or USB or even using on-board DC jack, however the GSM modem should be powered only via on-board DC jack using a wall adapter. The whole circuit can also be powered from a 9V wall adapter rated at 1 amp as shown in the diagram.

The Tx of the Arduino should connect to Rx of the GSM module and Rx of the Arduino should connect to Tx of the GSM module and ground to ground connection between Arduino and GSM must be made.

There are 5 relays to control five different gadgets independently. The relay voltage can be 9 to 12V, if you are using 12V relays then you should supply 12V to the circuit and this is the maximum.

The 5 relays are where it takes 5V signal from Arduino and switches 9 to 12V rated relay ON and OFF.

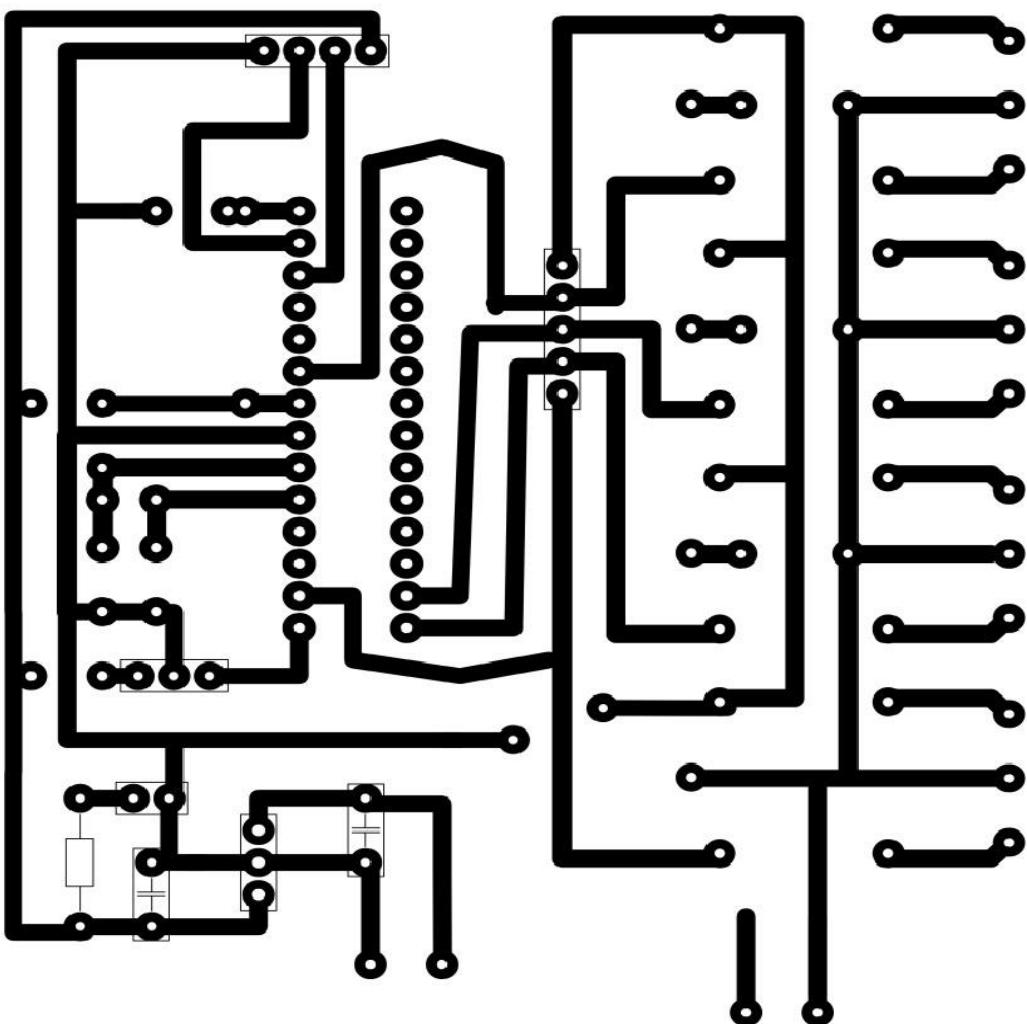
### ***Microcontroller diagram:-***



### how is a Microcontroller diagram?

The diagram above describes how to install the Arduino ic and how the circuit diagram will look like how the module, power supply and relay will be connected.

### PCB diagram:-



### ***what is PCB diagram ?***

•Printed circuit board (PCB) design brings your electronic circuits to life in the physical form. Using layout software, the PCB design process combines component placement and routeing to define electrical connectivity on a manufactured circuit board.

### ***project short information:-***

Here we have used a prefix in command string that is "#A.". This prefix is used to identify that the main command is coming next to it and \* at the end of string indicates that message has been ended.

When we send SMS to GSM module by Mobile, then GSM receives that SMS and sends it to Arduino. Now Arduino reads this SMS and extract main command from the received string and stores in a variable. After this, Arduino compare this string with predefined string. If match occurred then Arduino sends signal to relay via relay driver for turning ON and OFF the home appliances.

When we send data to arduino by LDR. Now Arduino reads this data and extract main command from the received string and stores in a variable. After this, Arduino compare this string with predefined string. If match occurred then Arduino sends signal to relay via relay driver for turning ON and OFF the home appliances.

Here in this project we have used 3 zero watt bulb for demonstration which indicates Fan, Light and TV.

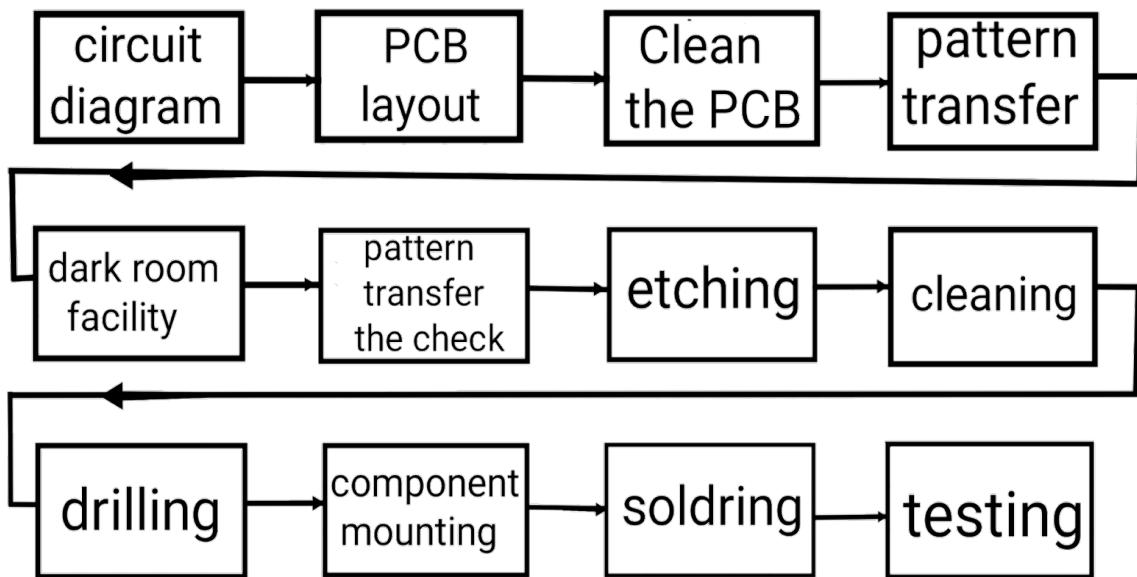
Below is the list of messages which we send via SMS, to turn On and Off the Fan, bulb and TV:

S.no	Message	Operation
1	#A.fan on*	Fan ON
2	#A.fan off*	Fan OFF
3	#A.bulb on*	bulb ON
4	#A.bulb off*	bulb OFF
5	#A.tv on*	TV ON
6	#A.tv off*	TV OFF
7	#A.all on*	All ON
8	#A.all off*	All OFF

Below is the list of Data which we send via LDR, to turn On and Off the Light

S.no	Data	Operation
1	Intensity=1	Light ON
2	Intensity=0	Light OFF

***PCB manufacturing block diagram:-***



### ***what is PCB manufacturing?***

Printed Circuit Board (PCB) fabrication is the assembly method for circuit boards used in electronic and internet viagra sales computer devices. ... Manufacturers of electronic goods can either fabricate in-house or contract this operation out to third party specialists.

## **STEP 1:** circuit diagram

Proteus is a simulation and design software tool developed by Labcenter Electronics for Electrical and Electronic circuit design. It is a software suite containing schematic, simulation as well as PCB designing.

## **STEP 2: PCB layout**

First you need to design the PCB layout using a service like Eagle,  
Take the printout of PCB layout bottom view on Photo paper using LASER Printer.

**Note:-**

For printing, use LASER PRINTER or PHOT COPIER only because the ink used in them is plastic toner ink.

In other printer ink used is in liquid form which will not transfer to copper after heating  
Use Photo Paper OR Press & Peel magazine cover with gloss lamination. paper OR  
any

### **STEP 3: Clean the PCB**

THE COPPER PLATE SO THERE IS A ROUGH SURFACE FOR THE DESIGN TO STICK TO WHEN TRANSFERED

#### **STEP 4:** pattern transfer

Place the print out sheet face down on copper  
Fix it using the paper tape .

Turn on Electric Iron and set the maximum temperature available.  
Give the maximum pressure and temperature up to 5 minute to the copper laminated sheet from photo paper side.  
After 5 minute keep this copper laminated sheet water for soaking.  
Wait for 1 minute or until paper softens  
Photo paper softens almost immediately, turning into a pulp that is easy to remove rubbing with your thumb.  
Keep rubbing until the paper gets dissolved (usually less than 1 minute).

**STEP 5:** dark room facility

Some space will be needed to build the project or build a PCB. There should be a dark room facility at that place which takes care of our safety.

**STEP 6:** pattern transfer the check

Check what has happened on the pattern transfer

**STEP 7:** etching

Using FeCl<sub>3</sub> in a plastic tray, do the etching of PCB and remove the unwanted copper.

**Note:**

FeCl<sub>3</sub> is an acid so while doing the etching keep your body and cloths away from the FeCl<sub>3</sub>.

It is not too dangerous for body; you can wash your body by soap.

But it is dangerous for cloths. Once you cloths are in contact with FeCl<sub>3</sub>, your cloths are damaged.

**STEP 8:** cleaning

After etching, the next procedure is to clean the PCB by removing the plastic toner.

Using acetone remove the plastic toner.

**STEP 9:** drilling

Take the drill machine and drill bit.

**STEP 10:** component mounting

The two diagrams below show: 1. Track side of a PCB (normally the underneath side) and 2.The component side (normally the top side) of the same circuit.

Take the 5v power supply PCB Mount the components as per the circuit diagram.

**Note:**

Components are placed on the non-copper side, and then the PCB is turned over to solder the component leads to the copper tracks.

Take precaution of terminals while mounting 16×2 LCD Display,Potentiometer 10K,Connecting Jumper Wires,Pulse Oximeter Sensor MAX30100 Module,DC Male and Female Jack,Eveready 9V Battery and Battery Holder Clip and IC.

### **STEP 11: soldering**

Take the soldering gun. And prepare it for soldering.

Place the soldering iron in its stand and plug in.

The iron will take a few minutes to reach its operating temperature of about 400°C.

Wait a few minutes for the soldering iron to warm up.

You can check if it is ready by trying to melt a little solder on the tip.

Wipe the tip of the iron on the damp sponge. This will clean the tip.

Melt a little solder on the tip of the iron.

This is called 'tinning' and it will help the heat to flow from the iron's tip to the joint. It only needs to be done when you plug in the iron.

Do the soldering.

Check the soldering and if there is any dry soldering, do the corrective action using soldering

After soldering cut the terminals by cutter.

#### **Note:**

Don't cut the terminals before the soldering. Always cut the terminals after soldering.

### **STEP 12: testing**

Is not soldering perfect? Check with multimeter.

## **Board Fitting**

A distribution board (also known as panelboard, breaker panel, or electric panel) is a component of an electricity supply system that divides an electrical power feed into subsidiary circuits while providing a protective fuse or circuit breaker for each circuit in a common enclosure.

### **STEP 1:-PCB Fitting**

Tighten the screw by sealing the PCB by piercing the board.

### **STEP 2:-Sim900a GSM Module Fitting**

Tighten the screw by piercing the board and sitting gsm.

### **STEP 3:-Bulb Holder,Bulbs,and 6A/16A 3 Pin board Socket Fitting**

Tighten the screw by piercing the MDF board and sitting Bulb Holder.

Tighten the screw by piercing the MDF board and sitting 6A/16A 3 Pin board Socket.

## ***Programming Compile In Upload ATmega328P (Arduino IC)***

The Arduino programming language is based on a very simple hardware programming language called processing, which is similar to the C language. After the sketch is written in the Arduino IDE, it should be uploaded on the Arduino board for execution.

### **Step 1:- Power up your board.**

The Arduino Uno, Mega, Duemilanove and Arduino Nano automatically draw power from either, the USB connection to the computer or an external power supply. If you are using an Arduino Diecimila, you have to make sure that the board is configured to draw power from

the USB connection. The power source is selected with a jumper, a small piece of plastic that fits onto two of the three pins between the USB and power jacks. Check that it is on the two pins closest to the USB port.

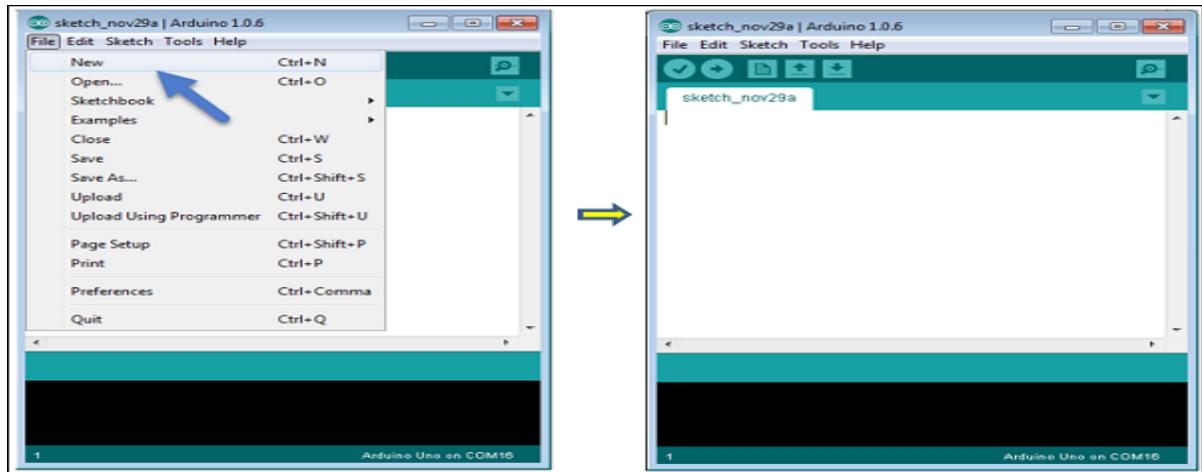
Connect the Arduino board to your computer using the USB cable. The green power LED (labeled PWR) should glow.

### **Step 2:- Open your first project.**

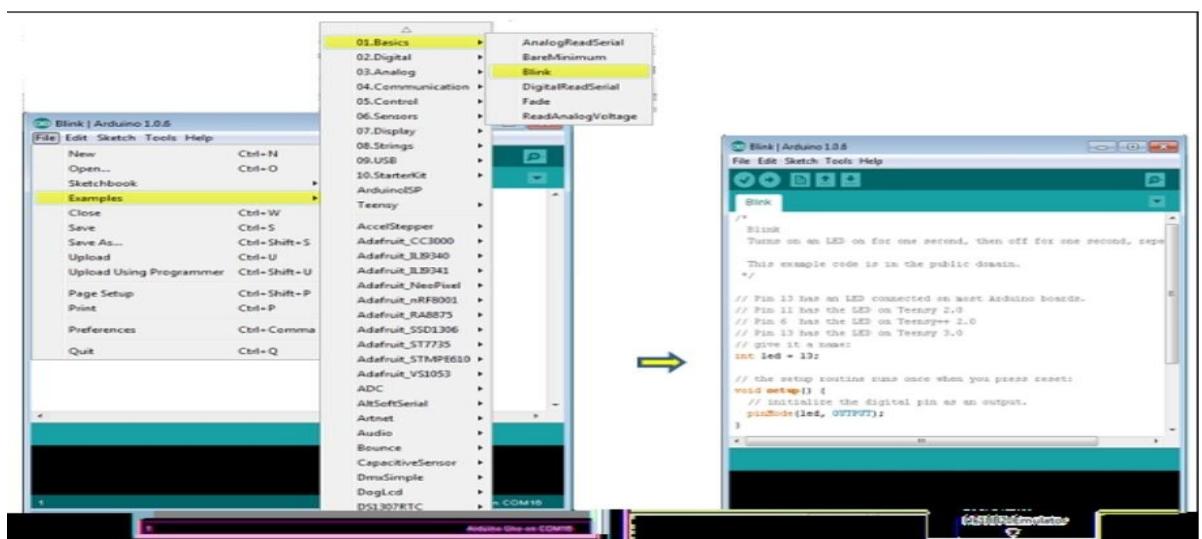
Once the software starts, you have two options – Create a new project.

Open an existing project example.

To create a new project, select File → New.



To open an existing project example, select File → Examples → GSM → ReceiveSMS → Edit code



We will take an example from GSM, that example will be the message received and we will edit it according to our project, after editing the court will look like the court below, this is also created by editing the code and copy pasting we have this Will use the code in the project

### **Code:-**

```

#define Fan 7
#define Light 9
#define TV 10
#define bulb 4
#define ldr 8
int temp=0,i=0;
int led=13;
char str[15];

void setup()
{
  initilize_system();
}

void loop()
{
  GSM_system();
  void serialEvent();
  void check();
  LDR_system();
}

void initilize_system()
{
  pinMode(led, OUTPUT);
  pinMode(Fan, OUTPUT);
  pinMode(Light, OUTPUT);
  pinMode(TV, OUTPUT);
  pinMode(bulb, OUTPUT);
  pinMode(ldr,INPUT);
  Serial.begin(9600);

  Serial.println("AT+CNMI=2,2,0,0,0");
  delay(500);
  Serial.println("AT+CMGF=1");
  delay(1000);
}

void GSM_system()
{
  if(temp==1)
  {
    check();
    temp=0;
    i=0;
    delay(1000);
  }
}

```

```

void serialEvent()
{
    while(Serial.available())
    {
        if(Serial.find("#A."))
        {
            digitalWrite(led, HIGH);
            delay(1000);
            digitalWrite(led, LOW);
            while (Serial.available())
            {
                char inChar=Serial.read();
                str[i++]=inChar;
                if(inChar=='*')
                {
                    temp=1;
                    return;
                }
            }
        }
    }
}

void check()
{
    if(!(strncmp(str,"tv on",5)))
    {
        digitalWrite(TV, HIGH);
    }
    else if(!(strncmp(str,"tv off",6)))
    {
        digitalWrite(TV, LOW);
    }
    if(!(strncmp(str,"bulb on",7)))
    {
        digitalWrite(bulb, HIGH);
    }
    else if(!(strncmp(str,"bulb off",7)))
    {
        digitalWrite(bulb, LOW);
    }
    else if(!(strncmp(str,"fan on",6)))
    {
        digitalWrite(Fan, HIGH);
    }
    else if(!(strncmp(str,"fan off",7)))
    {
        digitalWrite(Fan, LOW);
    }
}

```

```

        }
        else if(!(strcmp(str,"all on",6)))
        {
            digitalWrite(Fan, HIGH);
            digitalWrite(TV, HIGH);
            digitalWrite(bulb, HIGH);
        }
        else if(!(strcmp(str,"all off",7)))
        {
            digitalWrite(Fan, LOW);
            digitalWrite(TV, LOW);
            digitalWrite(bulb, LOW);
        }
    }

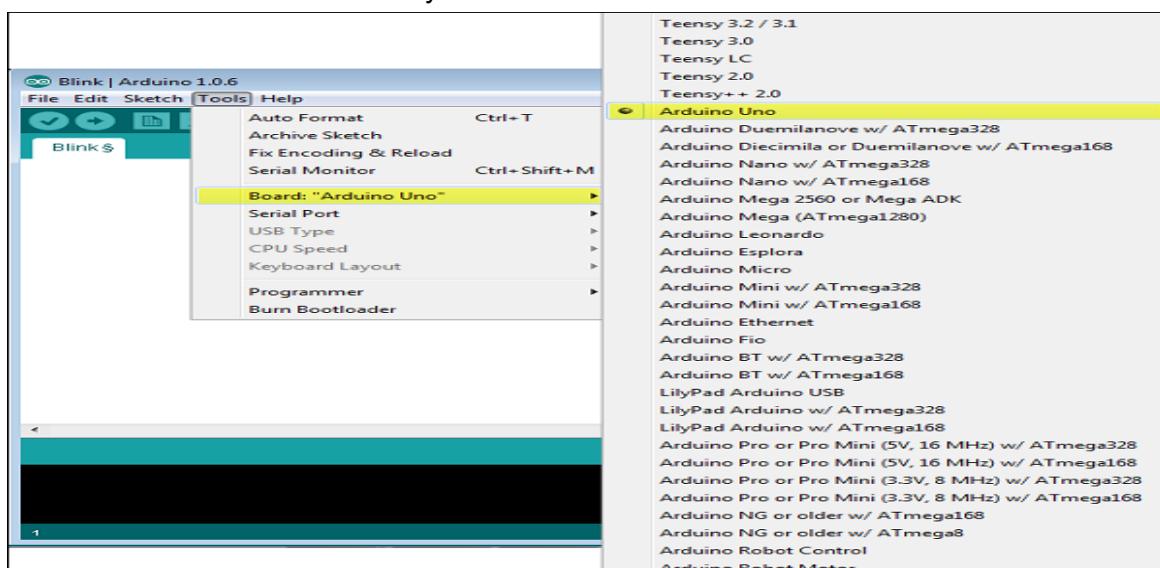
void LDR_system()
{
    int temp=digitalRead(Idr);
    if(temp==HIGH)
        digitalWrite(Light,HIGH);
    else
        digitalWrite(Light,LOW);
}

```

**Step 3:- Select your Arduino board.**

To avoid any error while uploading your program to the board, you must select the correct Arduino board name, which matches with the board connected to your computer.

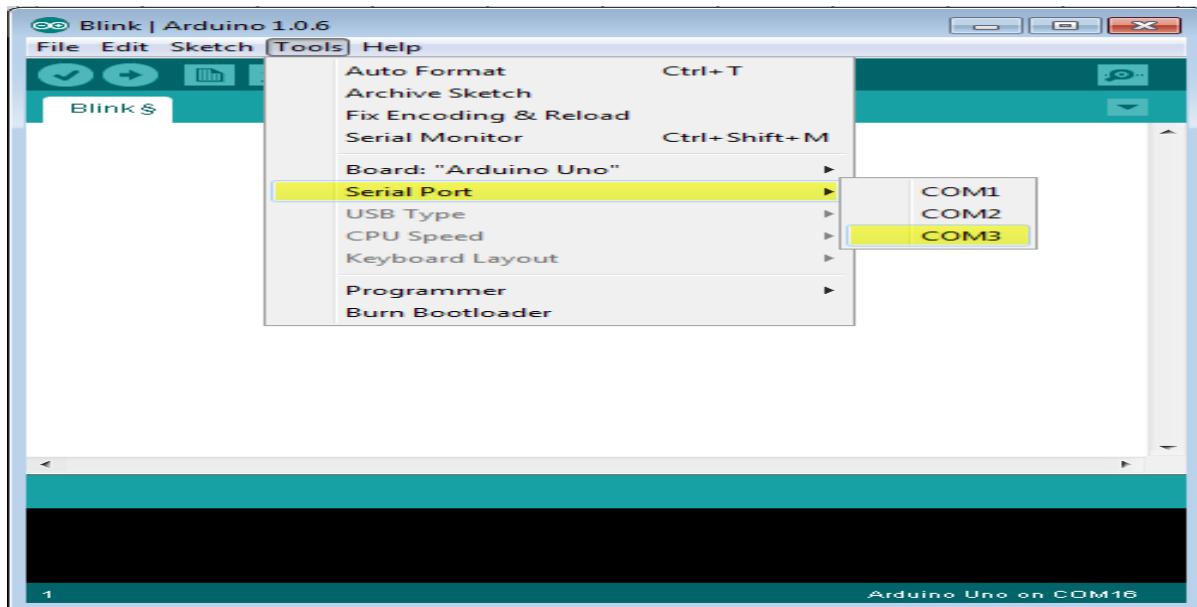
Go to Tools → Board and select your board.



Here, we have selected Arduino Uno board according to our tutorial, but you must select the name matching the board that you are using.

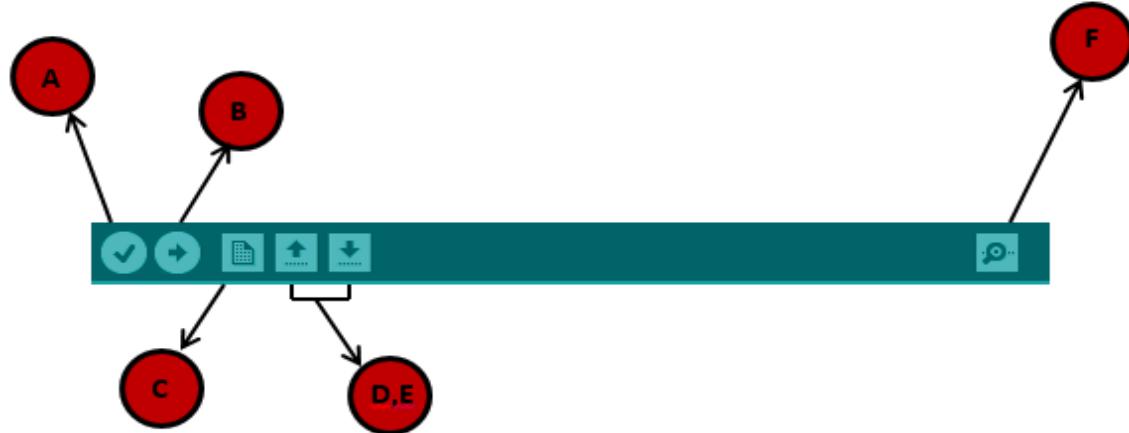
**Step 4:- Select your serial port.**

Select the serial device of the Arduino board. Go to Tools → Serial Port menu. This is likely to be COM3 or higher (COM1 and COM2 are usually reserved for hardware serial ports). To find out, you can disconnect your Arduino board and re-open the menu, the entry that disappears should be of the Arduino board. Reconnect the board and select that serial port.



**Step 5:-** Upload the program to your board.

Before explaining how we can upload our program to the board, we must demonstrate the function of each symbol appearing in the Arduino IDE toolbar.



A – Used to check if there is any compilation error.

B – Used to upload a program to the Arduino board.

C – Shortcut used to create a new sketch.

D – Used to directly open one of the example sketch.

E – Used to save your sketch.

F – Serial monitor used to receive serial data from the board and send the serial data to the board.

Now, simply click the "Upload" button in the environment. Wait a few seconds; you will see the RX and TX LEDs on the board, flashing. If the upload is successful, the message "Done uploading" will appear in the status bar.

Note – If you have an Arduino Mini, NG, or other board, you need to press the reset button physically on the board, immediately before clicking the upload button on the Arduino Software.

### ***The Last Test / Result Working***

Message sent from mobile:-

### **Components cost:-**

No	COMPONENT NAME	QUANTITY	COST
1	Arduino UNO board	1	450
2	SIM900A GSM Module	1	1850
3	4 Relay	1	80
4	AC 230v Power Supply	-----	---
5	LDR sensor	1	40
6	Bulb Holder	2	150
7	Bulbs	2	450
8	6A 3 Pin Socket and PVC brand box	1	50
9	16A 3 Pin Socket and PVC brand box	1	200
10	Jumper Wires Hardware	40	190
11	wires Solid type	2 metre	20
12	IC Base- 28- Pin	1	8
13	ATmega328P ic	1	100
14	2 Pin PCB Connector	12	120
15	voltage regulator IC 7805	1	15
16	22pf capacitor	2	6
17	100uf capacitor	1	3
18	10uf capacitor	1	3
19	LED 5mm	1	2
20	470 ohm resistor	1	2

21	16MHZ Crystal Oscillator	1	10
22	IC Master 40 PIN FEMALE HEADER	1	8
23	AC to DC converter 12v/2A	1	200
24	etching pcb kit	-----	300
25	PCB soldering kit	-----	200
26	Photo paper print	1	30
27	mdf board 10mm(61CM×61CM)	1	318
28	other	-----	1250
--	-----	Total COST	

### **CONCLUSIONS:-**

The main purpose of the project is to provide a smart and efficient way to control our home appliances with the help of modern communication technology. With the combination of Arduino and GSM this project is not only a modern approach to next generation home but also a very friendly and easy to use system. This project will give our daily life a new dimension at the same time it will help us to save energy.