Instructors' Manual

Electronic Skills Development (ESD) Lab

for

SE E&TC (SPPU)

by

CopperCloud IOTech Pvt Ltd

Issued at

Online Faculty Orientation Workshop

(23-26 Jun 2020)

In association with Army Institute of Technology

CopperCloud IOTech Pvt Ltd

Instructors' Manual for ESD Lab
SE E&TC (SPPU)

Table of Contents

Document: CIPL/Trg/2020/23 Dated 23 Jun 2020

Instructor's Manual

Electronic Skills Development (ESD) Lab – SE E&TC

1. Introduction

This document is meant for use by the faculty of E&TC Departments of colleges affiliated with SPPU for designing, conducting and supervising the ESD lab assignments for SE E&C students.

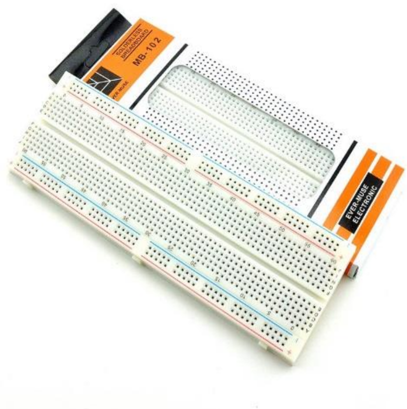
The document also includes a list and details of recommended lab assignments for this purpose, which the faculty can use as is, or adapt to their particular needs.

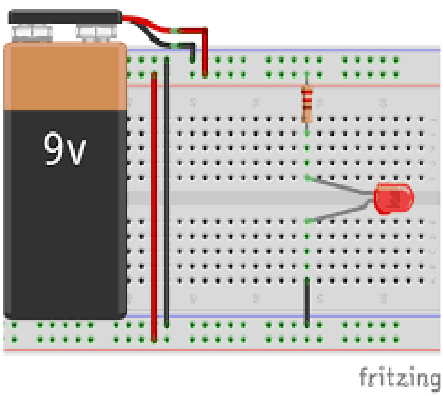
The reference material, including source code, is available at this url: https://github.com/coppercloud-iotech/sppu-esdlab-fdp-teachingaids

2. Electronic Components

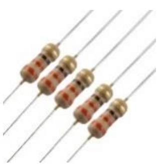
This are the basic component use in Electronic/Embedded and IOT system.

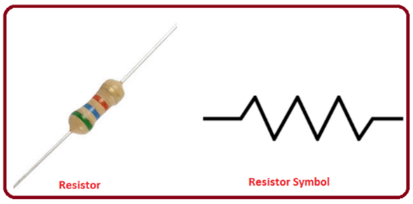
- **Bread board:** A breadboard is a rectangular plastic board with a bunch of tiny holes in it. These holes let you easily insert electronic components to prototype (meaning to build and test an early version of) an electronic circuit.



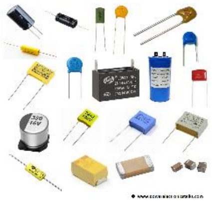


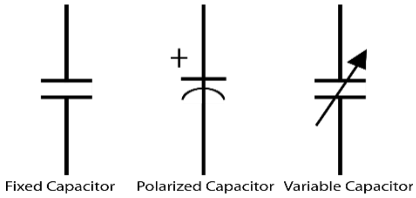
- **Resistor:** It is an electrical component that reduces the electric current. The **resistor's** ability to reduce the current is called resistance and is measured in units of ohms (symbol: Ω).



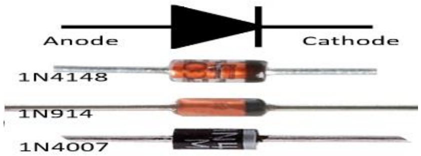


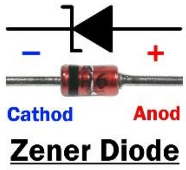
- **Capacitor:** A *capacitor* is a device that stores electrical energy in an electric field. It is a passive electronic component with two terminal



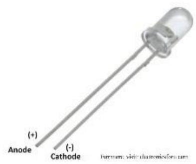


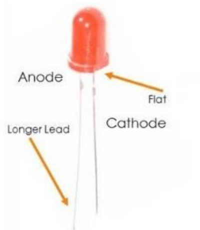
- **Diode:** A **Diode** is a semiconductor device that essentially acts as a one-way switch for current. It allows current to flow easily in one direction, but severely restricts current from flowing in the opposite direction. Diodes are also known as **rectifiers** because they change alternating current (ac) into pulsating direct current (dc).

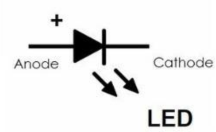




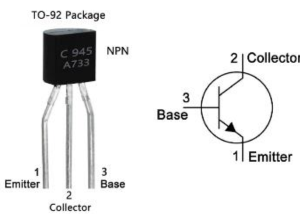
- **LED:** A **light-emitting diode** (**LED**) is a semiconductor light source that emits light when current flows through it.

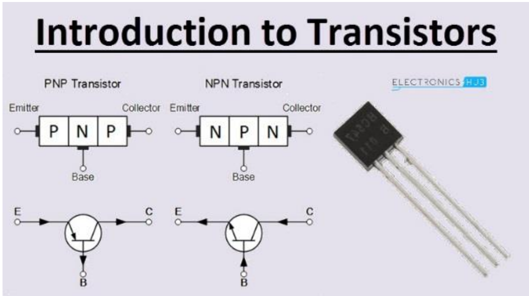




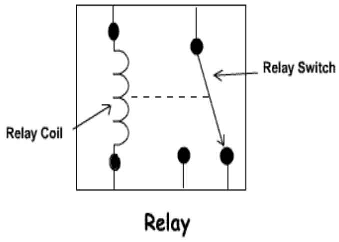


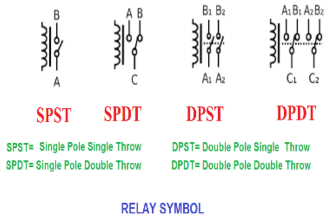
- **Transistor:** A **transistor** is a semiconductor device used to amplify or switch electronic signals and electrical power. It is composed of semiconductor material usually with at least three terminals for connection to an external circuit.



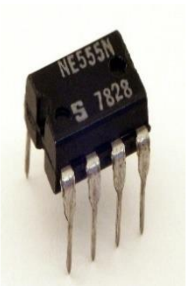


- **Relay:** A **relay** is an electrically operated switch. It consists of a set of input terminals for a single or multiple control signals, and a set of operating contact terminals. The switch may have any number of contacts in multiple contact forms, such as make contacts, break contacts, or combinations thereof.





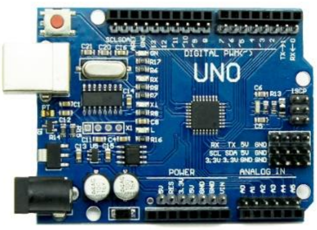
- **IC:** An **integrated circuit** or **monolithic integrated circuit** (also referred to as an **IC**, a **chip**, or a **microchip**) is a set of electronic circuits on one small flat piece (or "chip") of semiconductor material that is normally silicon.



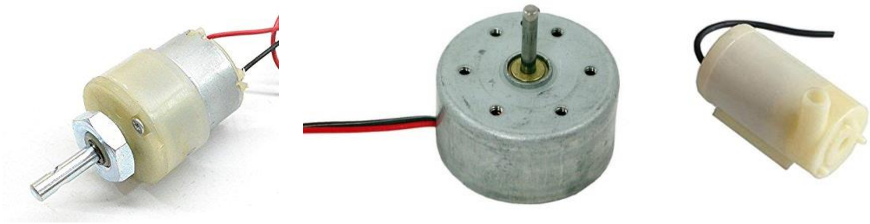

- **SMPS:** A switched-mode power supply (SMPS) is an electronic circuit that converts power using switching devices that are turned on and off at high frequencies, and storage components such as inductors or capacitors to supply power when the switching device is in its non-conduction state.


- **Microcontroller**: A **microcontroller** is a compact integrated circuit designed to govern a specific operation in an embedded system.
  Typical **microcontroller** includes a processor, memory and input/output (I/O) peripherals on a single chip


- **Arduino UNO: Arduino Uno** is a microcontroller board based on the ATmega328P (datasheet).



- **DC MOTOR:** A ***DC motor*** is any of a class of rotary electrical motors that converts direct current (DC) electrical energy into mechanical energy.

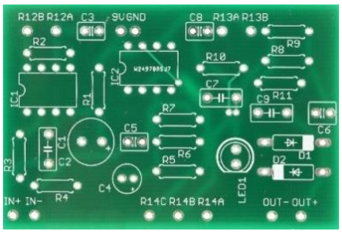


- **Servo Motor:** A **servo motor** is an electrical device which can push or rotate an object with great precision. If you want to rotate and object at some specific angles or distance, then you use **servo motor**.

- **Sensor:** A **sensor** is a device that detects and responds to some type of input from the physical environment. The specific input could be light, heat, motion, moisture, pressure, or any one of a great number of other environmental phenomena.

- **Actuator:** An **actuator** is a component of a machine that is responsible for moving and controlling a mechanism or system, for example by opening a valve. In simple terms, it is a "mover". An **actuator** requires a control signal and a source of energy

- **PCB:** A **printed circuit board** (**PCB**) mechanically supports and electrically connects electrical or electronic components using conductive tracks, pads and other features etched from one or more sheet layers of copper laminated onto and/or between sheet layers of a non-conductive substrate. Components are generally soldered onto the PCB to both electrically connect and mechanically fasten them to it.


3. Tools and Equipment

The following tools & equipment are useful for these labs:

- **Wire Stripper:-**It is for both stripping insulation off wires and cutting them, a good ***wire stripper*** is a permanent tool in every electrical lab.



- **Wire Cutter:** Wire cutter used to cut or twist metal wires.



- **Multimeter:** A **multimeter** or a multitester, also known as a VOM (volt-ohm-milliammeter), is an electronic measuring instrument that combines several measurement functions in one unit. A typical **multimeter** can measure voltage, current, and resistance.


- **Clamp Meter:** A **clamp meter** is an electrical test tool that combines a basic digital multimeter with a current sensor. Clamps measure current. Probes measure voltage.



- **Soldering Station:** It is needed to do some soldering work. If you don't do a lot of soldering, a simple soldering station will do the job. However, we really recommend getting a good soldering station with enough wattage and adjustable temperature.


- **Soldering Accessories:** Besides the soldering station you also need some soldering accessories:

Solder: the easiest type to work with is 60/40 lead/tin. Lead-free solder is recommended for ecological reasons.



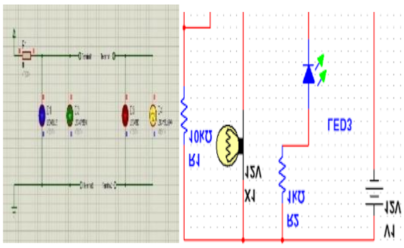

Solder wick: the wick soaks up molten solder, it's useful to clean the excess of solder.

Tweezer: Tweezers are very useful to manipulate the small electronics components.

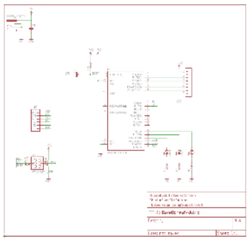
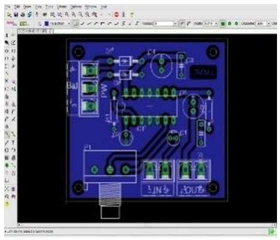
4. Software Tools

Besides the Hardware tools you also need some software tools on your machine/Laptop:

1. Multisim/PROTAEUS 8.x: Use for schematic design and simulation.

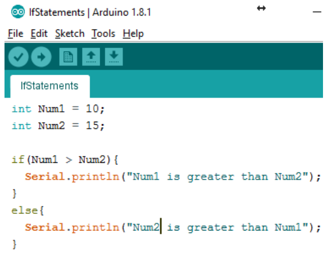



2. EAGLE 5.9.0: Schematic and PCB layout design.

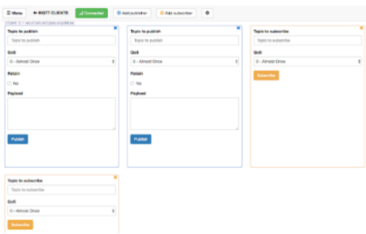



3. VIEW PLOT: View and cross check GERBER file.

4. ARDUINO IDE: Use to write code and upload it to the board.

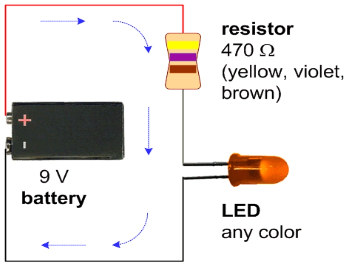


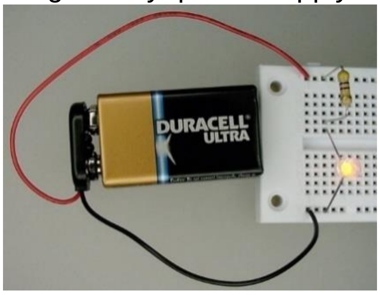
5. MQTT BOX: Connect to ***Mqtt*** brokers with TCP, TLS, Web Sockets and Secure Web Sockets. - Connect with wide range of ***Mqtt*** client settings.



5. Basic Electronics Circuit Design

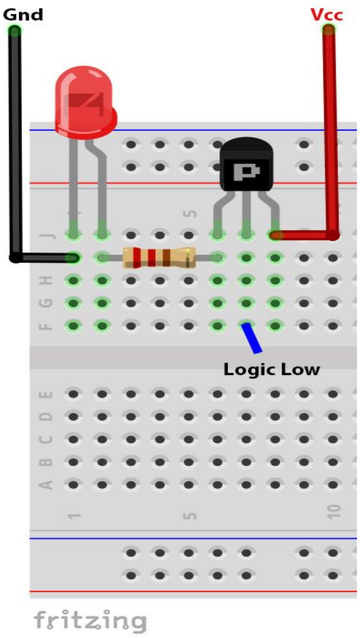
- **LED BLINKING:** Blink LED using battery/ power supply.

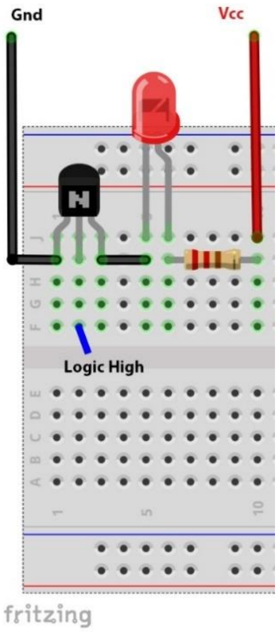


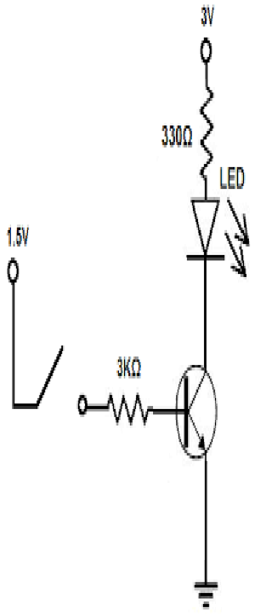


https://www.robotroom.com/Pumpkin2.html

- **Blink LED using Transistor Driver:** Blink LED using transistor driver or switch.

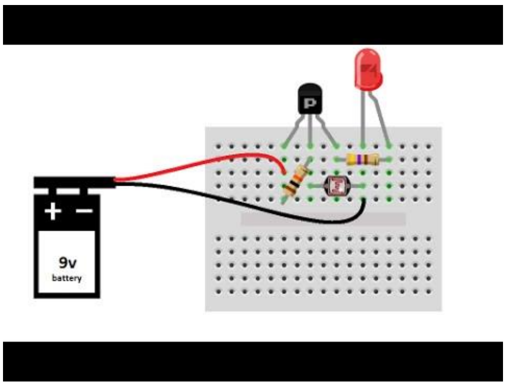


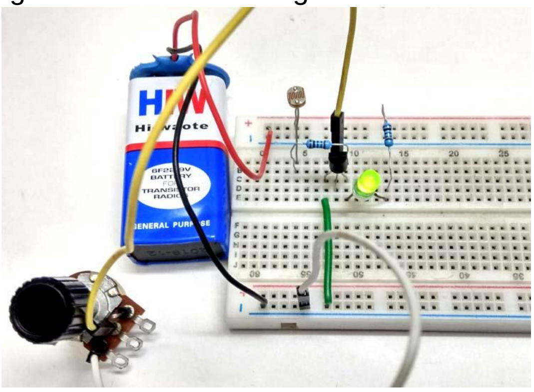




https://learn.edwinrobotics.com/tutorial-switch-transistor-and-buzzer-on-breadboard/

- **Day/Night Sensor Circuit:** Build day/night sensor circuit using transistor driver.

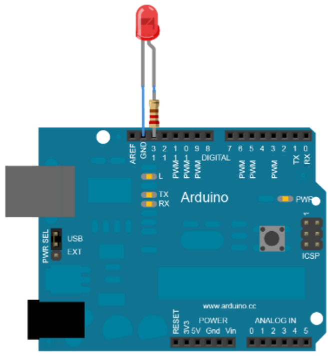




https://www.youtube.com/watch?v=KMO5g0WNfJ8
https://circuitdigest.com/electronic-circuits/light-detector-ldr-circuit

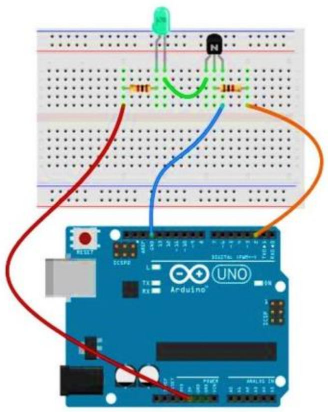
6. Arduino Interfacing

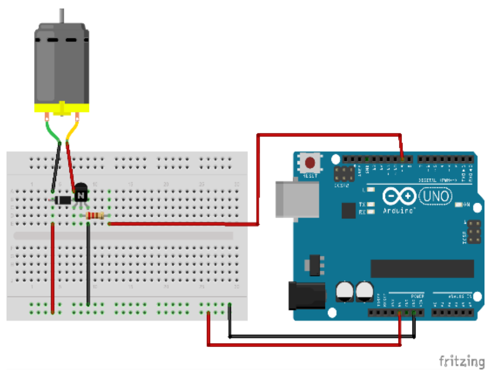
- **Blink LED:** Blink LED using Arduino UNO.



https://www.arduino.cc/en/Tutorial/Blink

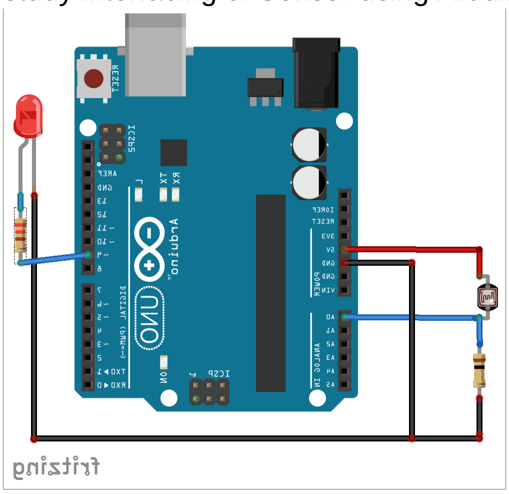
- **Arduino interfacing with Driver:** Turn on transistor based driver circuit using Arduino.





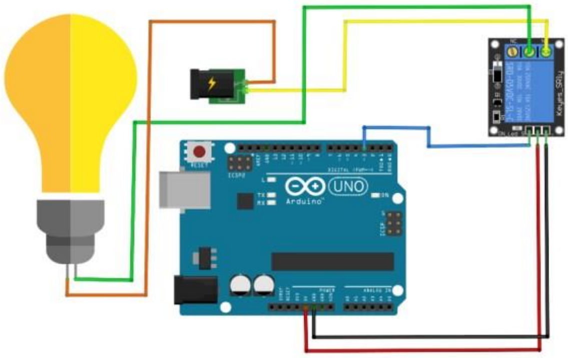
https://create.arduino.cc/projecthub/sumanbargavr/switching-using-transistor-c9114a
https://forum.arduino.cc/index.php?topic=390379.0

- **Sensor interfacing with Arduino:** Build day night sensor using Arduino also study interfacing of Sensor using Arduino.



https://mechatronicslabrpi.blogspot.com/2019/02/arduino-light-sensor-circuit-using-ldr.html

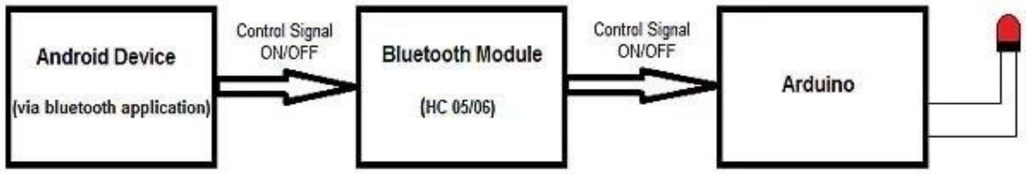
- **Relay interfacing with Arduino:** Interfacing Relay with Arduino and control DC or AC Power Supply



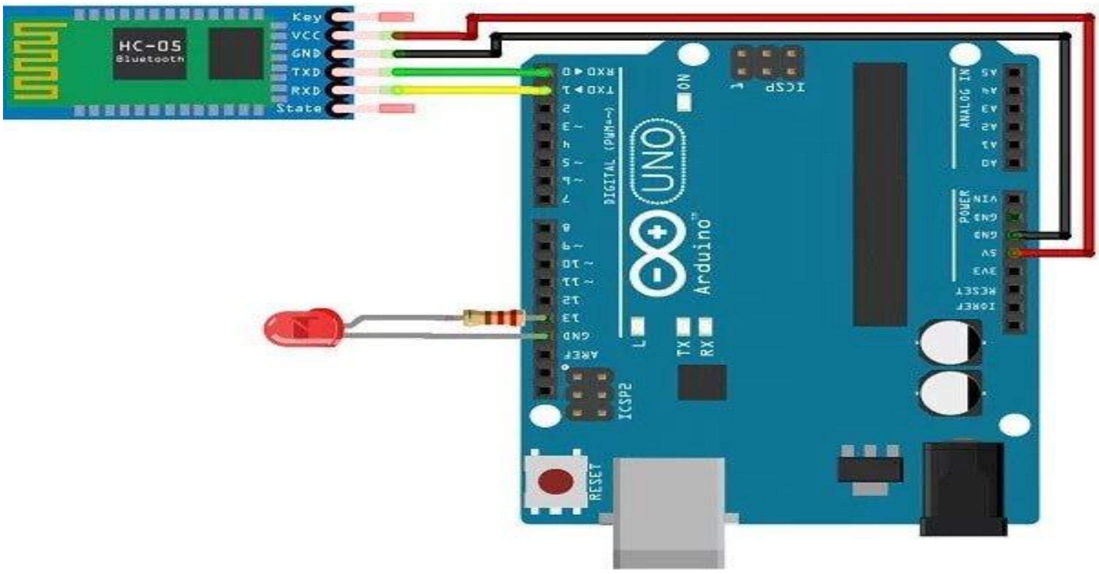
.

https://www.teachmemicro.com/arduino-relay-module-tutorial/

- **Wireless Connectivity to Arduino:**
  Interfacing an Android smartphone with Arduino via Bluetooth



The Android app is designed to send serial data to the Arduino Bluetooth module when a button is pressed on the app. The Arduino Bluetooth module at the other end receives the data and sends it to the Arduino through the TX pin of the Bluetooth module (connected to RX pin of Arduino). The code uploaded to the Arduino checks the received data and compares it. If the received data is 1, the LED turns ON. The LED turns OFF when the received data is 0.

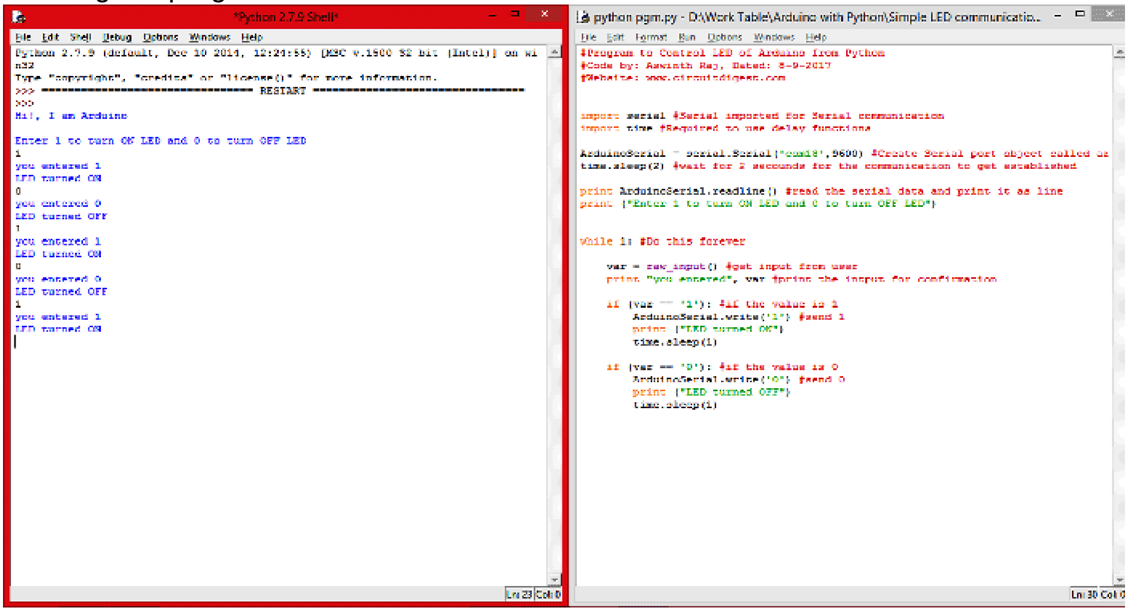


https://maker.pro/arduino/tutorial/bluetooth-basics-how-to-control-led-using-smartphone-arduino#:~:text=The%20Android%20app%20is%20designed,to%20RX%20pin%20of%20Arduino).

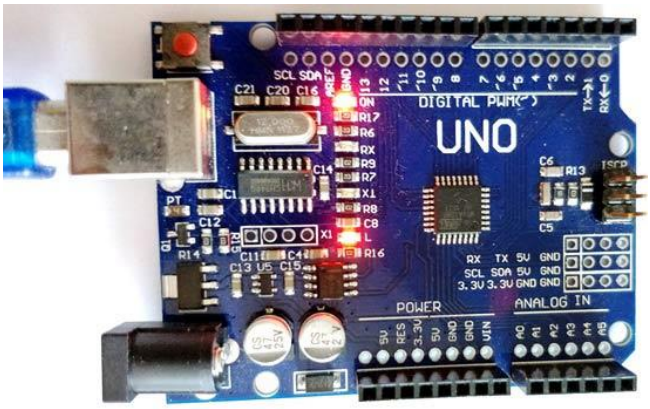
- **Python with Arduino:**

  Controlling the in-built Arduino board LED using Python script.

Upload the program to your Arduino and verify it is connected to the same COM port as mentioned in the python program. Then Launch the Python program. This will launch a python shell script as shown below. The window on the left is the shell window showing the output and the window on the right is the script showing the program.



As you can see the string *"Hi! I am Arduino"* entered in the Arduino program is received by the Python and displayed on its shell window

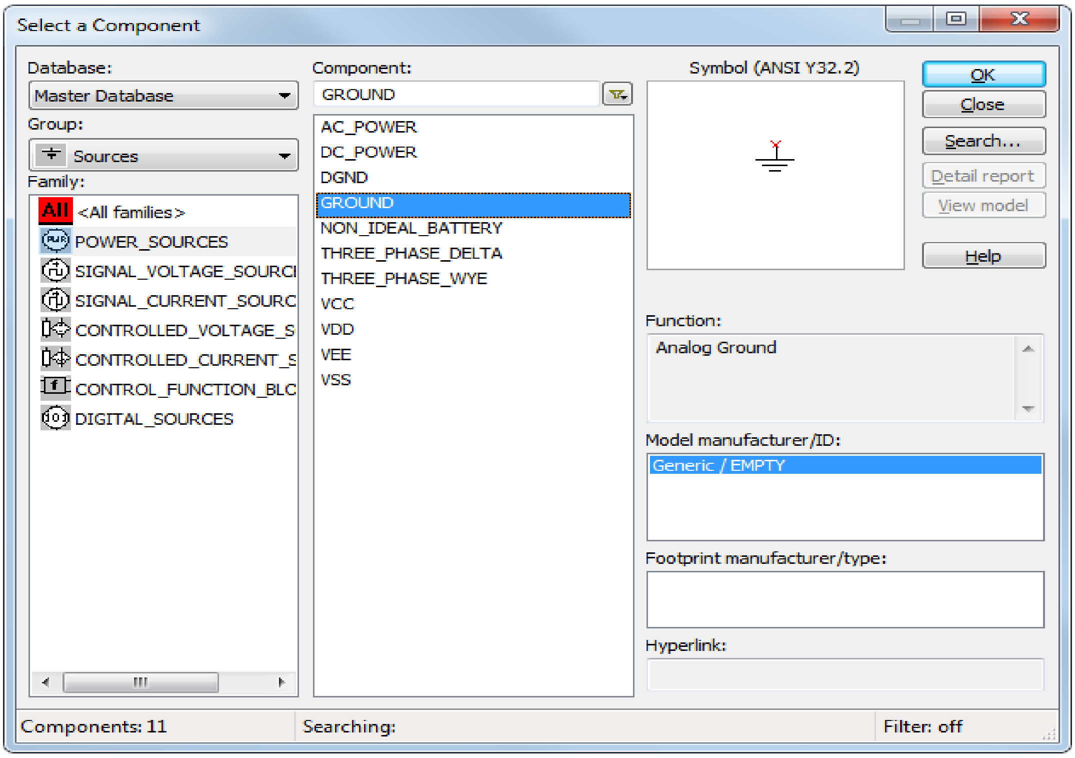


When the shell window asks to enter values, we can enter either 0 or 1. If we send 1 the LED on the Arduino Board will turn ON and if we send 0 the LED on our Arduino Board will turn OFF. Showing a successfully connection between our Arduino Program and Python.
https://circuitdigest.com/microcontroller-projects/arduino-python-tutorial

6. PCB Design

There are many designing software available for designing PCBs such as Express PCB, Eagle PCB, PCB-Web Designer, Zenith PCB, PCB Elegance, Free PCB, Open Circuit Design and Proteus etc. We will be using Multisim /Proteus to design our PCB circuit and simulate it.

- **Designing and Simulate Circuit in Multisim:**
  Begin by drawing your schematic in the Multisim environment.
  1. Open Multisim by selecting **All Programs »National Instruments » Circuit Design Suite 14.0 » Multisim 14.0**.
  2. Select **Place » Component**. The **Select a Component** window appears (also known as the **Component Browser**), as shown in Figure 2.



The **Component Browser** organizes the database components into three logical levels. The **Master Database** contains all shipping components in a read-only format. The **Corporate Database** is where you can save custom components to be shared with colleagues. Finally, the **User Database** is where custom components are saved that can be used only by the specific designer.

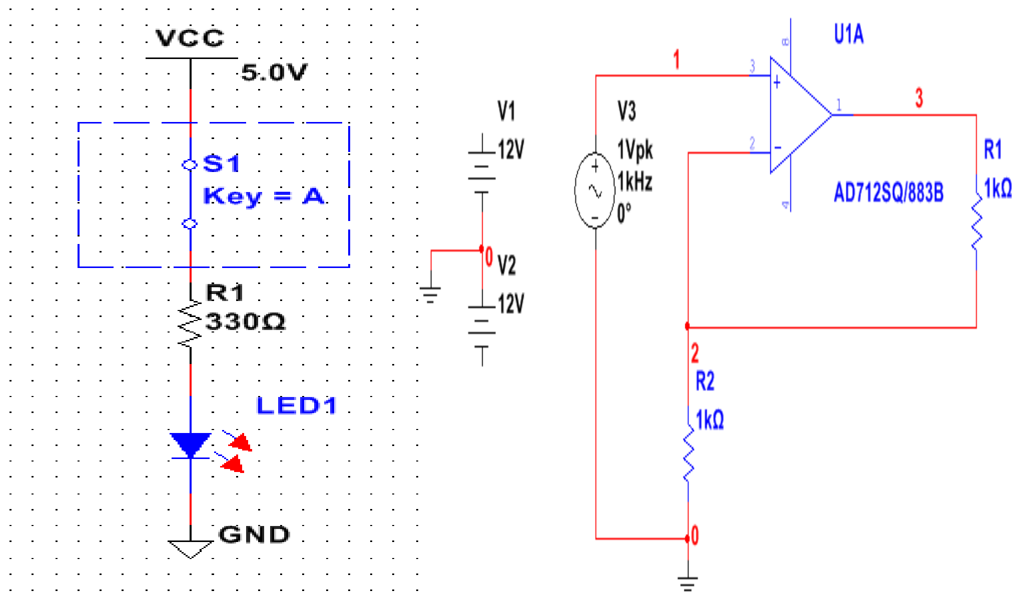
Additional Points

- Without a power and ground your simulation cannot run.
- If you need multiple components you can repeat the placement steps as shown, or place one component and use copy (Ctrl+C) and paste (Ctrl+V) to place additional components as needed.
- By default, the **Select a Component** window keeps returning as a pop-up until you have completed placing your components. Close the window to return to the schematic entry window.

Wiring the Schematic:

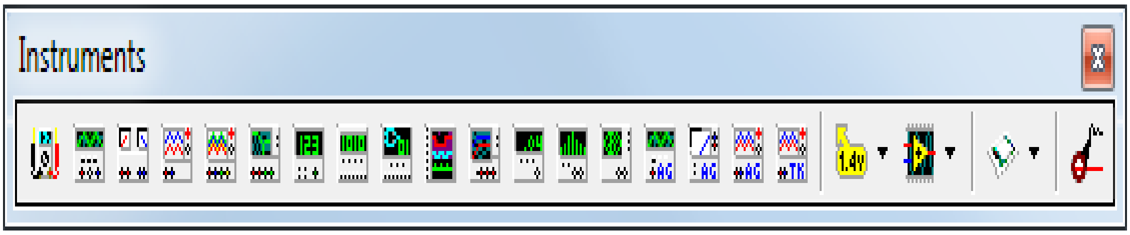
Multisim is a modeless wiring environment. This means that Multisim determines the functionality of the mouse pointer by the position of the mouse. You do not have to return to the menu to select between the placement, wiring, and editing tools.

1. Begin wiring by moving the mouse pointer close to a pin of a component. The mouse appears as a crosshair rather than the default mouse pointer.
2. Place an initial wire junction by clicking on the pin/terminal of the part (in this case, the output pin of the opamp).
3. Complete the wire by moving the mouse to another terminal or just double-click to anchor the termination point of the wire to a floating location somewhere in the schematic window.
4. Create a copy of the ground symbol using Copy <Ctrl+C> and Paste <Ctrl+V>.
5. Complete the wiring as shown in Figure 5. Do not worry about the labeled numbers on the wires (also called nets).

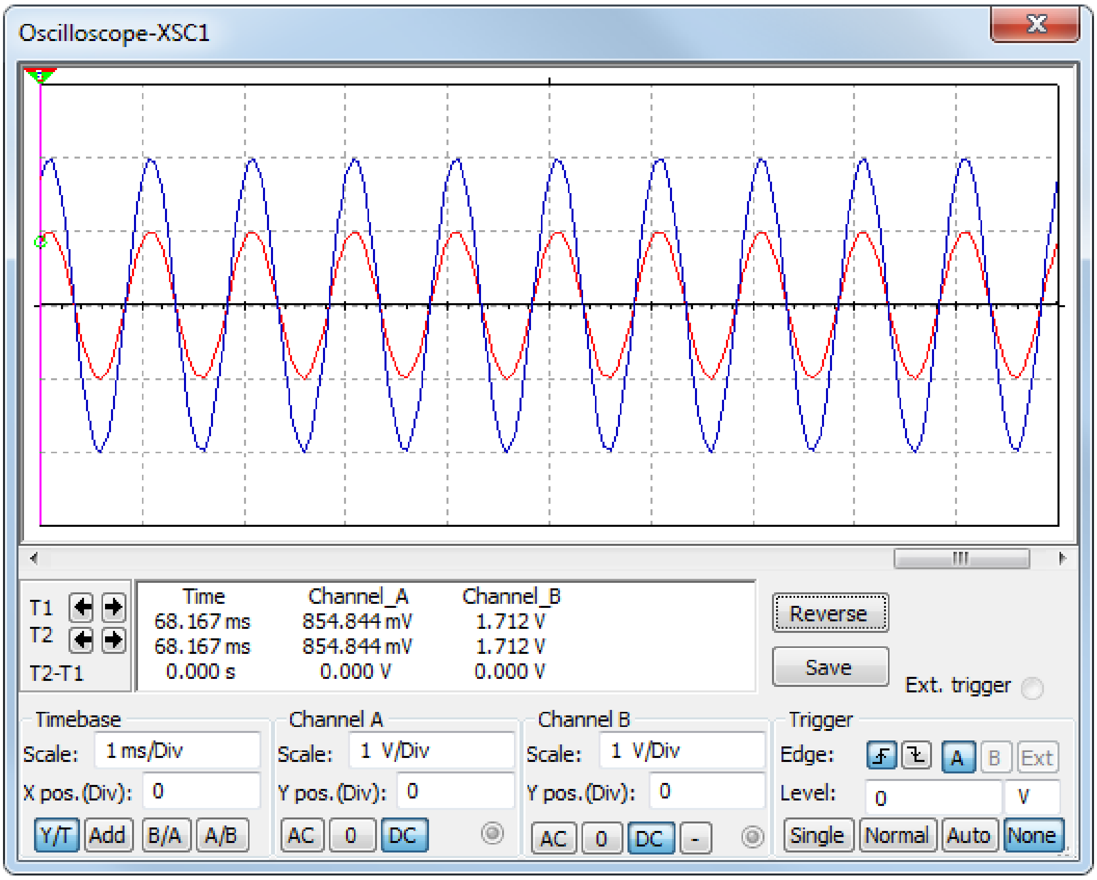


Simulating the Circuit:

You are now ready to run an interactive Multisim simulation; however, you need a way to visualize the data. Multisim provides instruments to visualize the simulated measurements. Instruments can be found on the right menu bar and are indicated by the following icons.



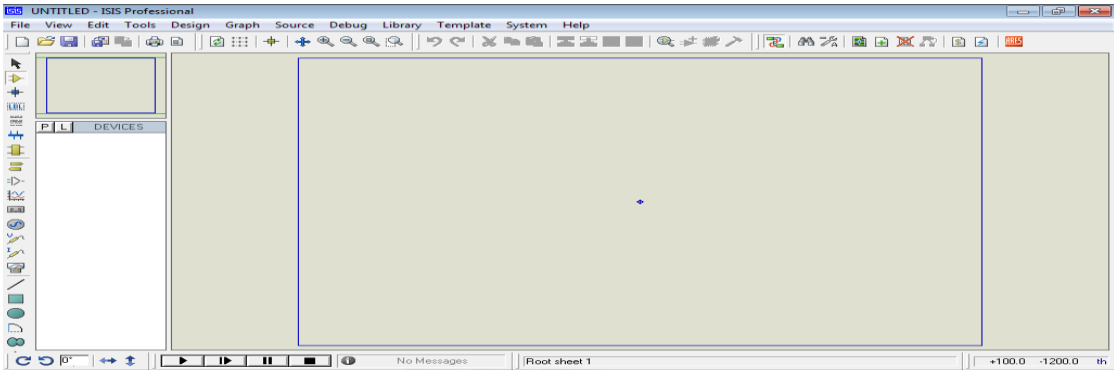
Select the **Oscilloscope** from the menu and place this onto the schematic. Select a shade of blue and click the **OK** button. The schematic should look like Figure. Select **Simulate »Run** to start the simulation.



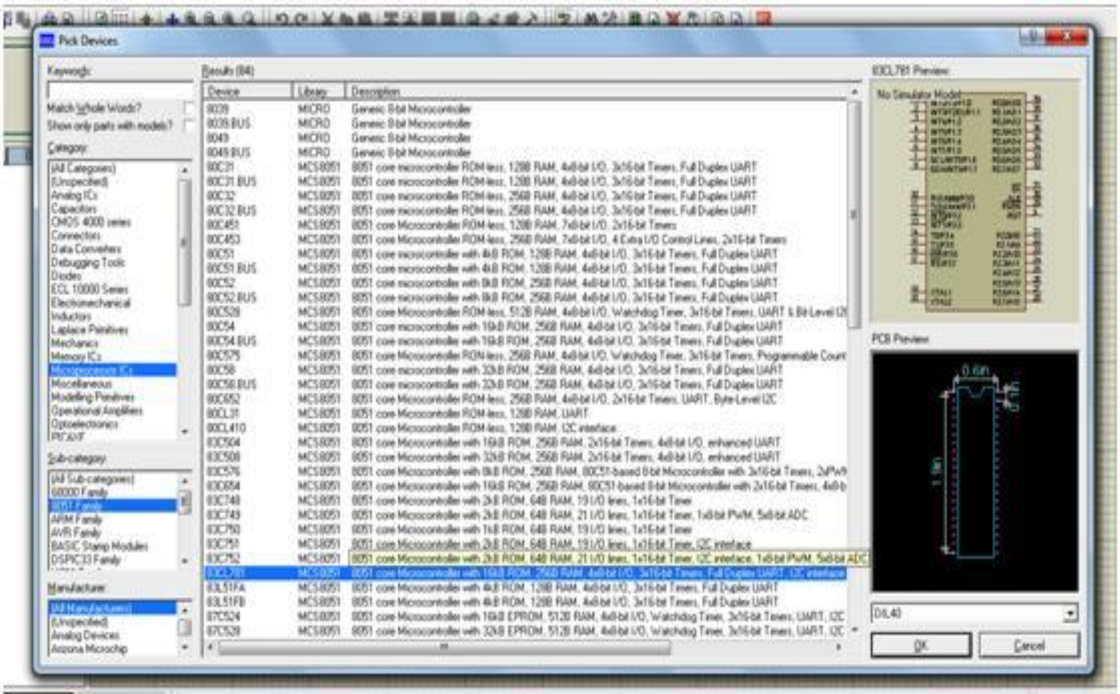
http://www.ni.com/tutorial/10710/en/

- **Designing and Simulate Circuit in Proteus:**

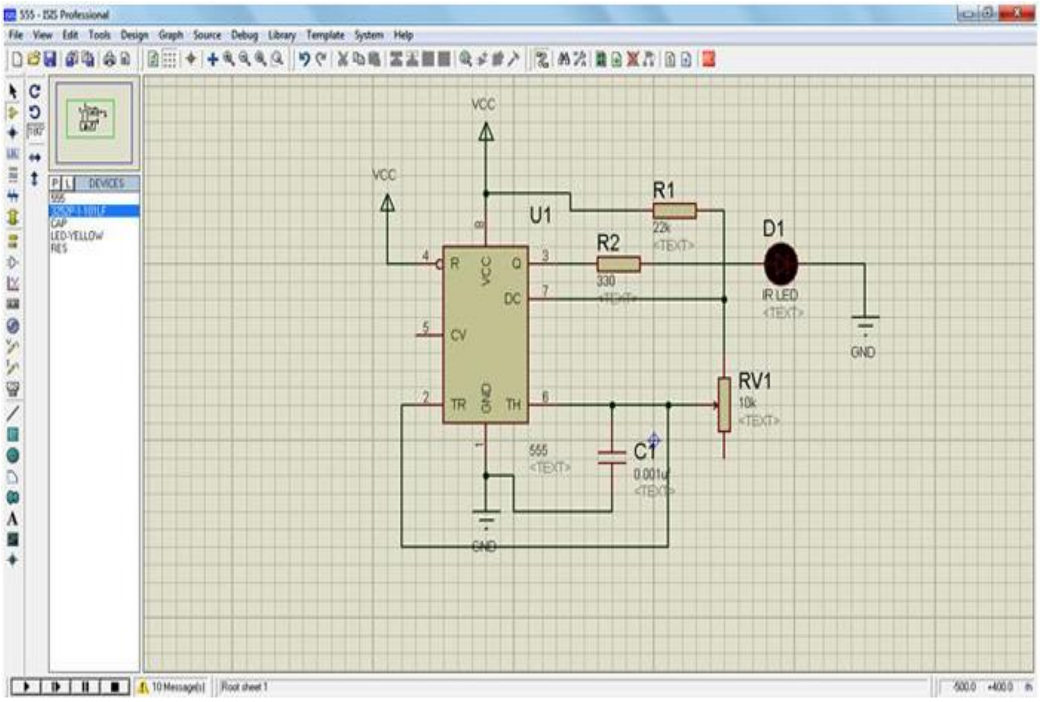
  Run the program by clicking on the icon, and a new splash screen will appear. Next, a grid-like workspace will appear as shown. There are buttons available that will help us in designing our PCB. There is a blue rectangle outline; our circuit will be designed within this rectangular space.



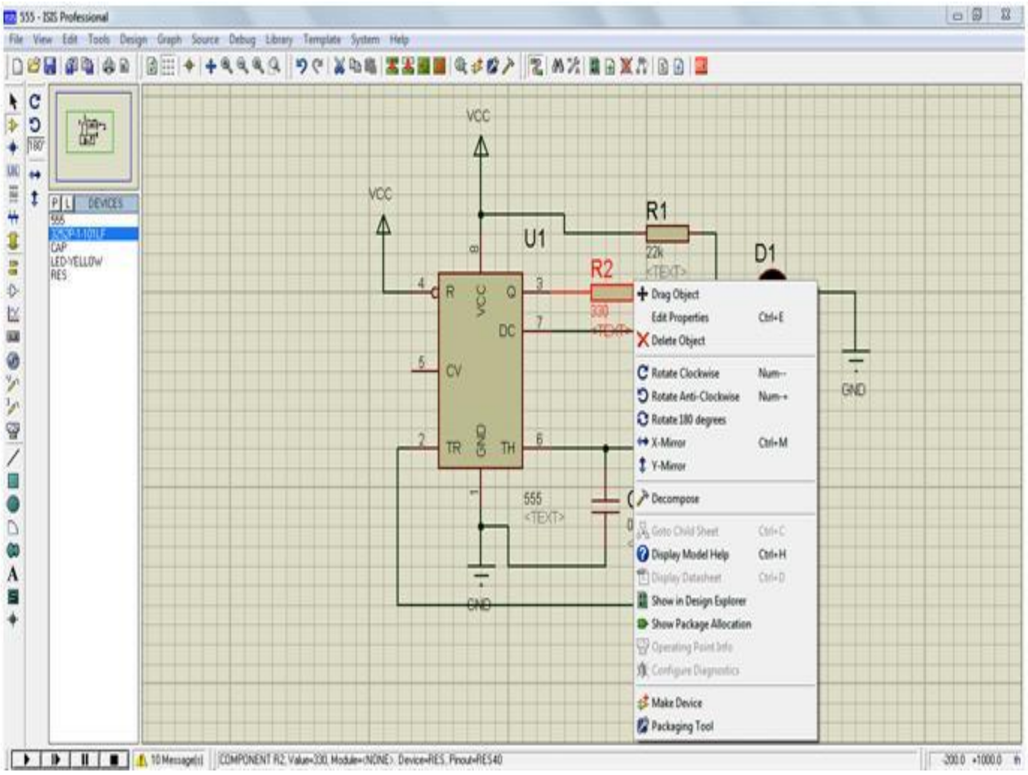
Now go to 'library' and select 'Pick device/symbol'. A window will appear. Another way to select components would be to use the toolbar on the left side of your workspace. Click on "component model" or pick from the library.



Now select all the components you require, these will be added to the devices list. The device you select can be rotated by using the rotate buttons. You can then place your component into the workspace. After placing all your components, place your cursor at the component pin end and draw the connections. Connect all the components to get a designed circuit as shown.



If you want to modify any component, simply right click on the component and a dropdown will appear as shown. After completion save and debug the saved file.



After connecting the circuit, click on the play button to run the simulation.

https://www.electronicslovers.com/2018/09/design-your-own-pcb-by-using-proteus.html
https://www.circuitstoday.com/proteus-software-introduction

- **Design Schematic and Layout of PCB in EAGLE:**

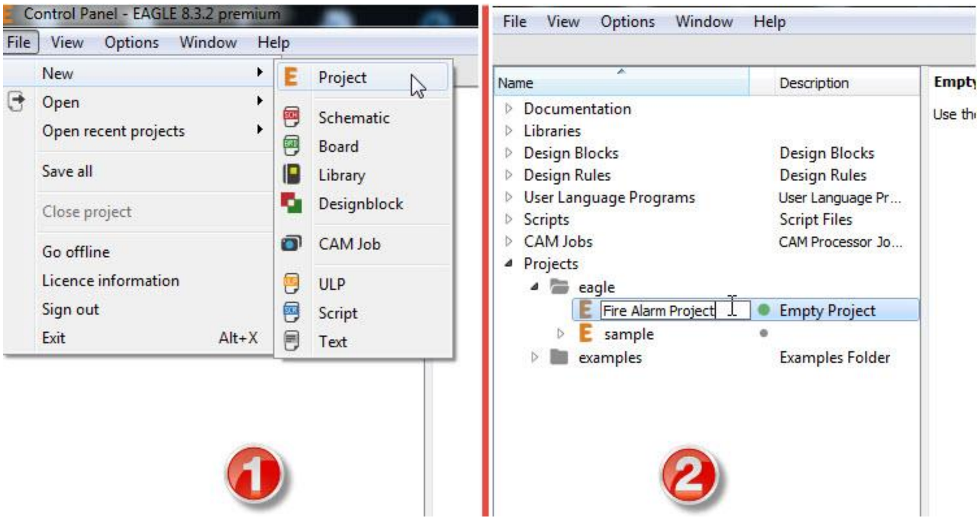
The steps for How to Design PCB using Eagle CAD Software. After designing the PCB in Autodesk's Eagle CAD, I'll show you to make your own PCB (Printed Circuit Board) at home.

A) Designing schematic :

Schematic and circuit board layout are done with the help of a CAD Tool. There are many CAD Tools for designing PCBs like Altium Designer, Cadence OrCAD, Mentor Graphics PADS, Autodesk Eagle, KiCad, etc. In this tutorial, we are going to use Autodesk Eagle CAD. Eagle is available in three variants: Eagle Free, Eagle Standard and Eagle Premium.

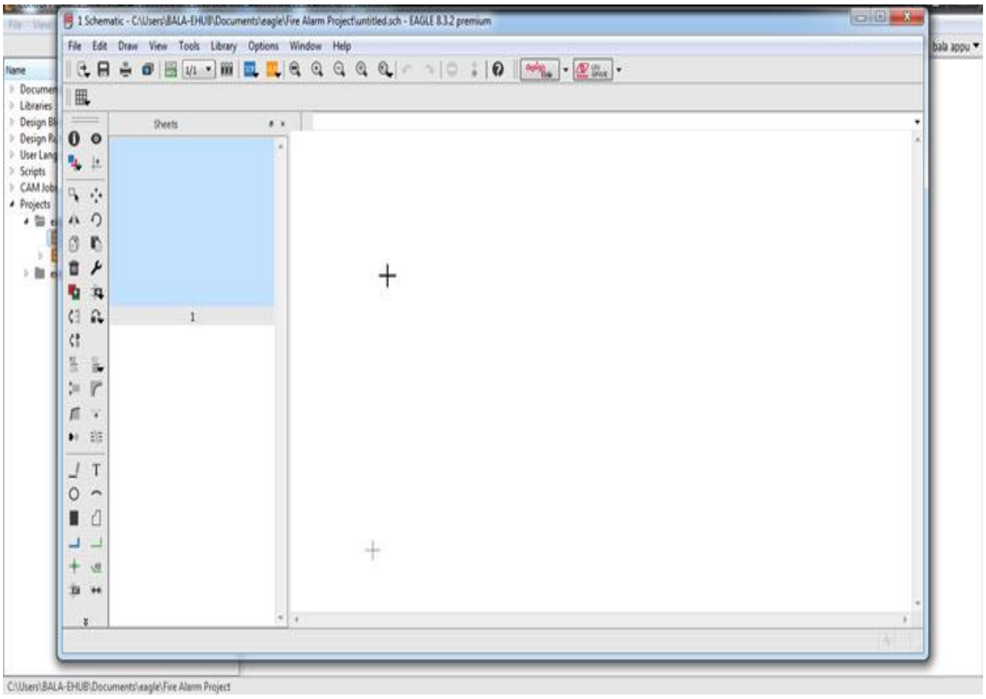
You can download the free version of the Eagle PCB from the Autodesk's official website or follow this link. https://www.autodesk.com/products/eagle/free-download

Click on File and select New Project. Rename the Project to an appropriate name (I will be designing the PCB for Fire Alarm Project). Right click on the project and create new schematic. Save the schematic file with extension .sch.

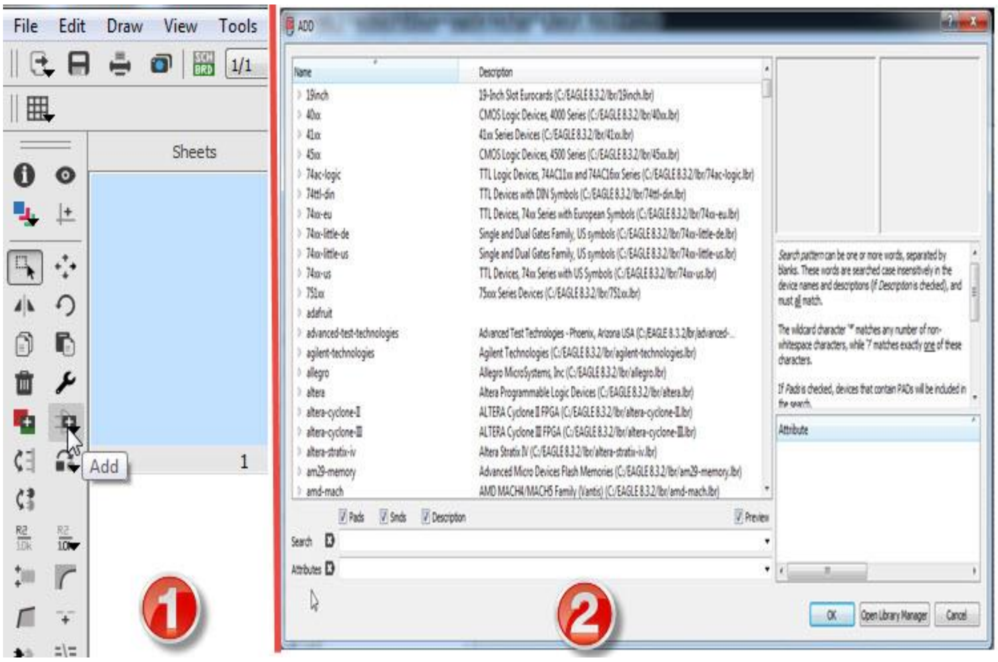


After this, we have to add the necessary components which we are going to use in our schematic. But, before that, we need to adjust the grid size of the schematic. Select the Grid option and set the size to 1mm. Also set the alternate grid to a smaller value. You can turn on the grid by selecting display on option.

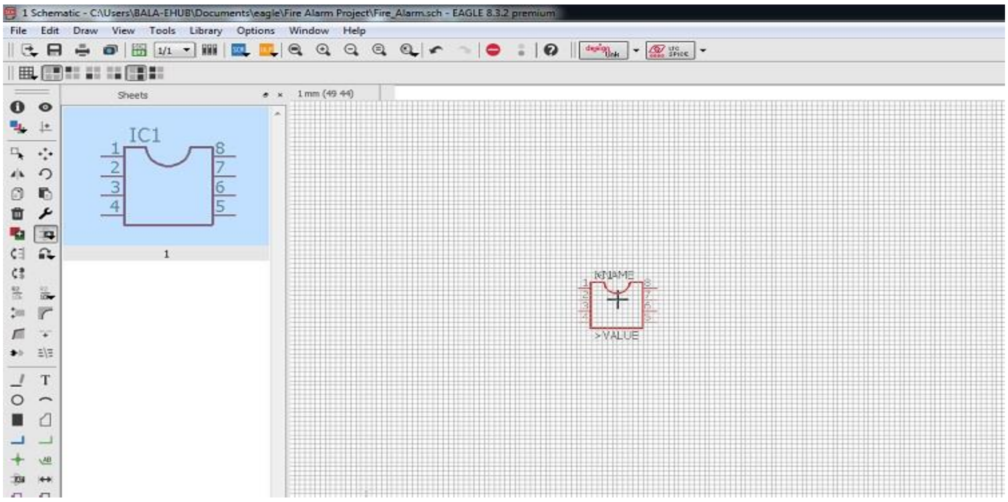
NOTE: Setting Grid is optional and set the values you are comfortable with.



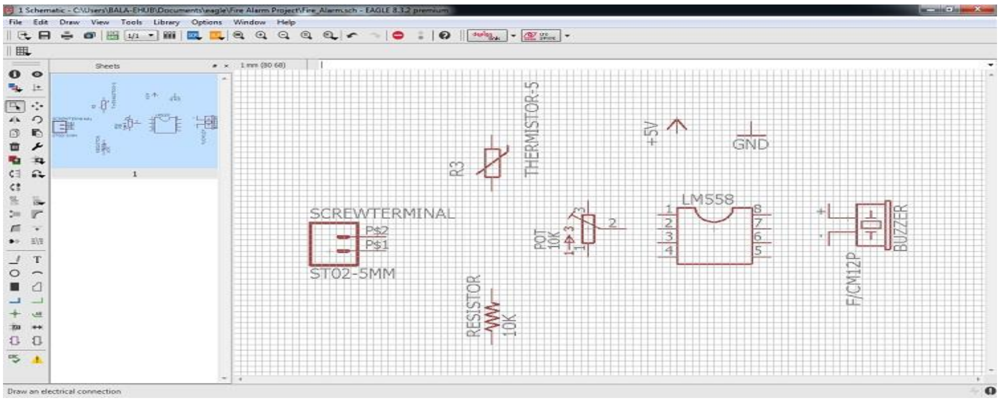
Now, we have to add components to our schematic. For this, select Add Option from the side tool bar. A new window opens with a list of all the components available in the libraries.



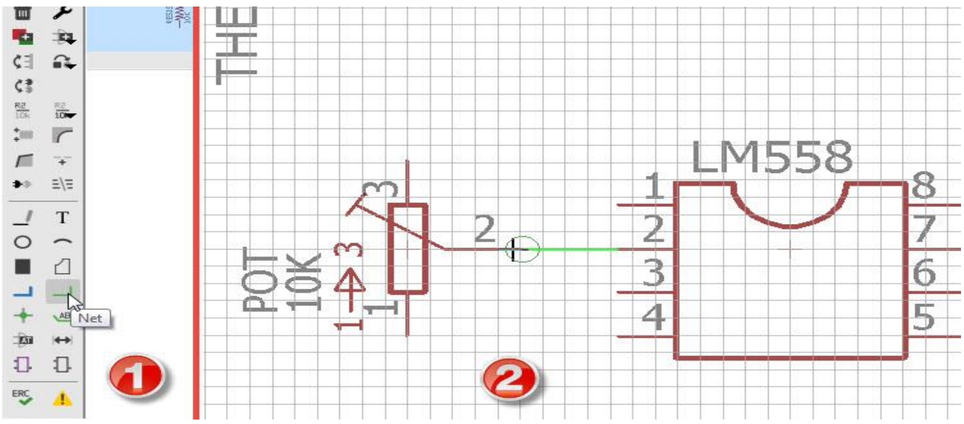
After selecting the component, click on OK and then you can place the component on the schematic sheet. Click on the sheet once to place the component and if you want to rotate the component, right click. After placing the component on the schematic, press Esc to return to the component selection window.



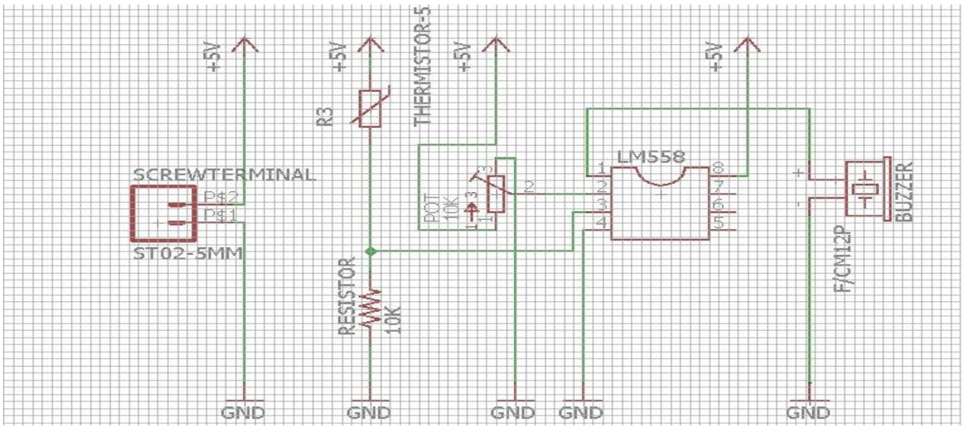
After placing all the components, rename the components to something understandable. Also, mention the values like 10KΩ Resistor etc.



Next step is to connect these components. You have to use the net option from the side tool bar and start making the connections.

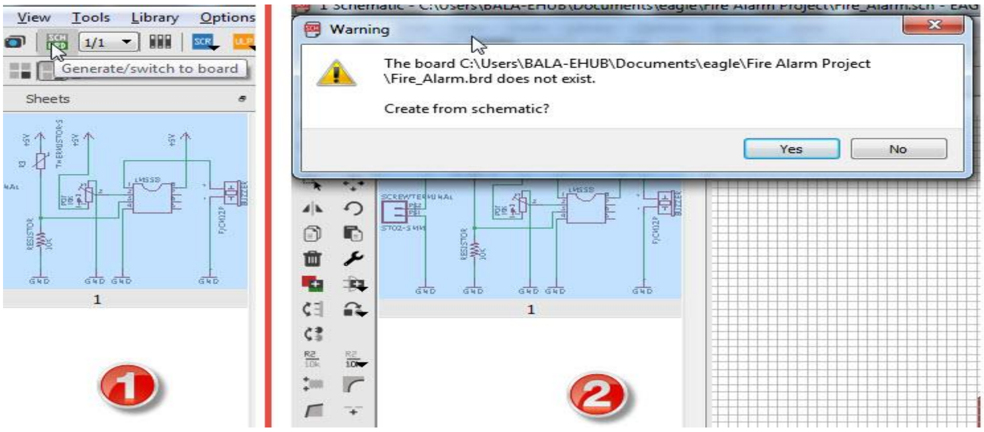


After making all the connections, the final schematic will look something like the circuit in the following image. Save the schematic file.

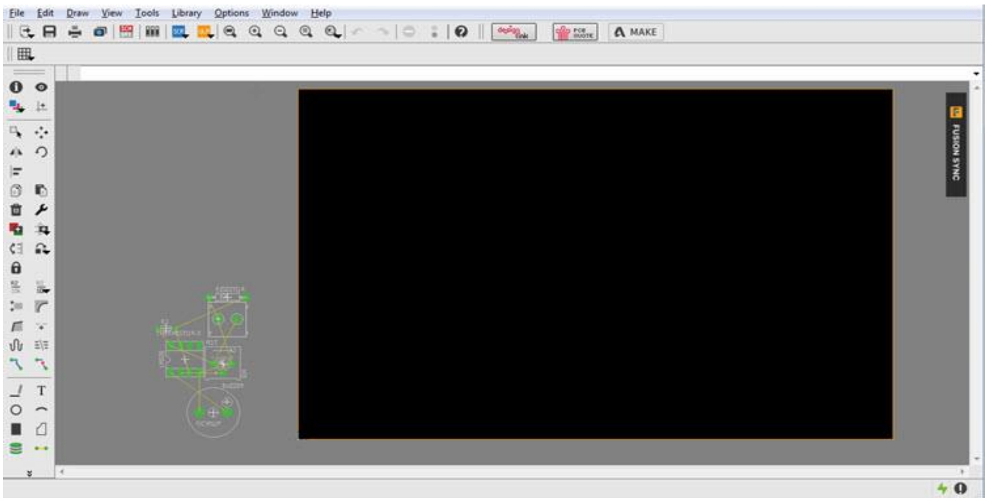


B) Drawing the layout for the Printed Circuit Board (PCB)

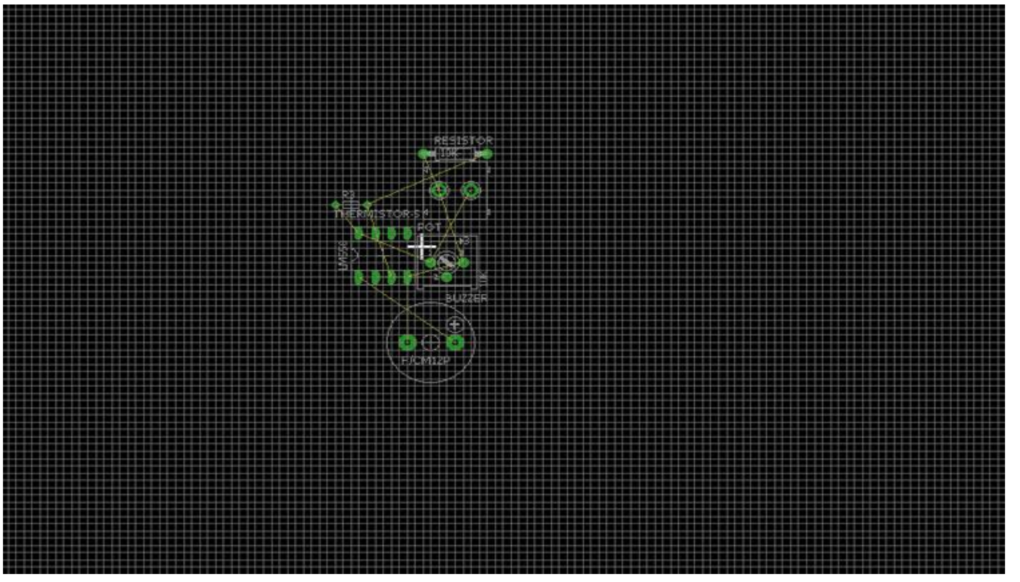
After completing the schematic, we have to proceed with the design of the PCB layout. Select the switch to board option from the top tool bar. You can create the board fie from the schematic.



A new window opens which is the PCB layout editor. The black space is the board area and all the components are at the outside bottom left of the board area. Now, we need to place the components in to the editor. You can adjust the grid size of the PCB Layout editor, if you want.

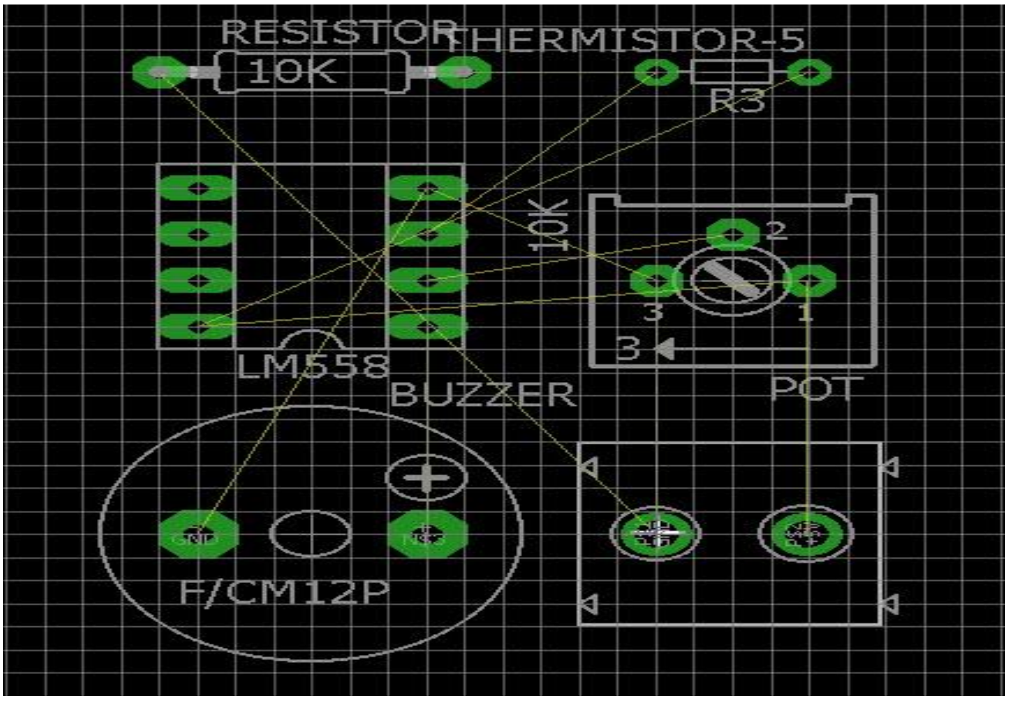


Now, using the group option from the side tool bar, select all the components and using move option move all the components and populate the board area.

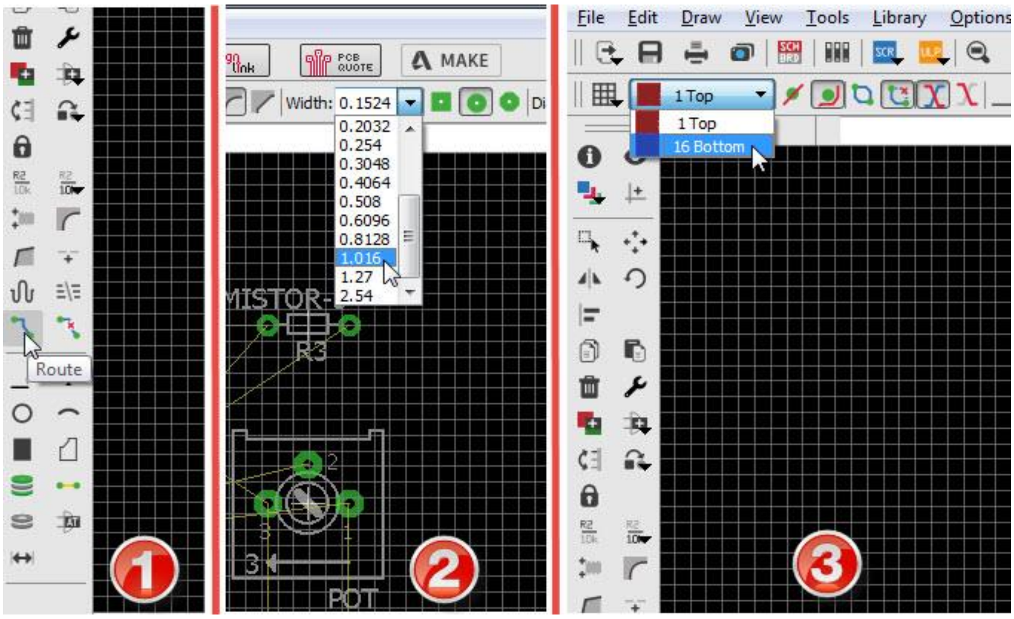


Using the move option, place the components on the board as per the position you want the component to be on the board. You can see thin yellow wires running between the components. These wires are called air wires and are representation of connections between components.

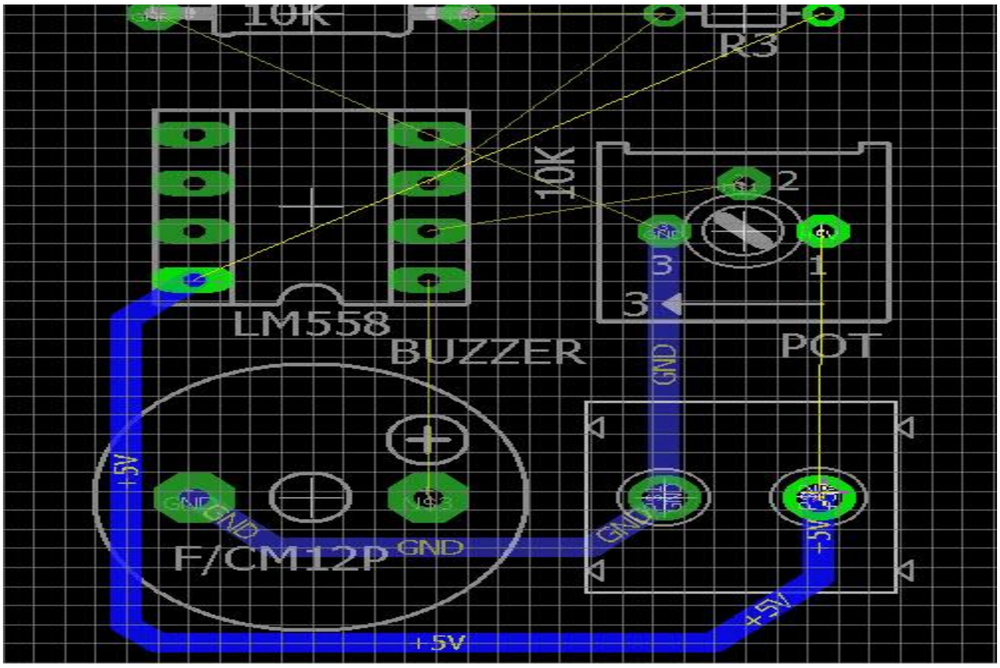
When we route the path between components, these air wires will disappear as an indication of successful connection.



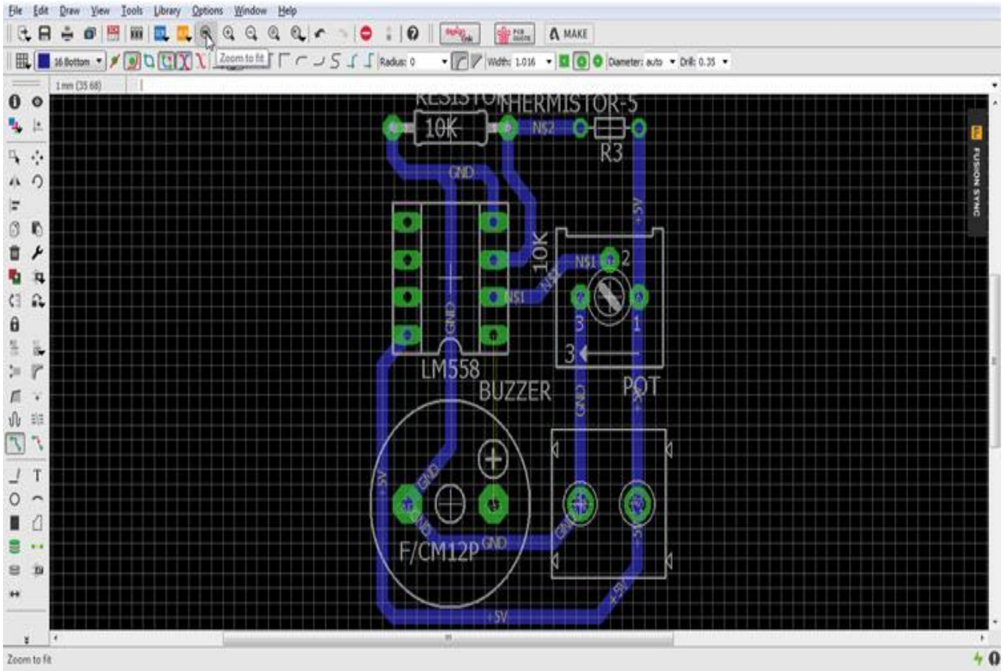
Now, it is time to make the connections or traces for the Printed Circuit Board. For this, we are going to use the Route tool from the side tool bar. Also set the width of the trace to be routed as your requirement. Here, I've set the trace width to approximately 1mm. Then, select the signal layer i.e. bottom layer, as our PCB is a single sided board.



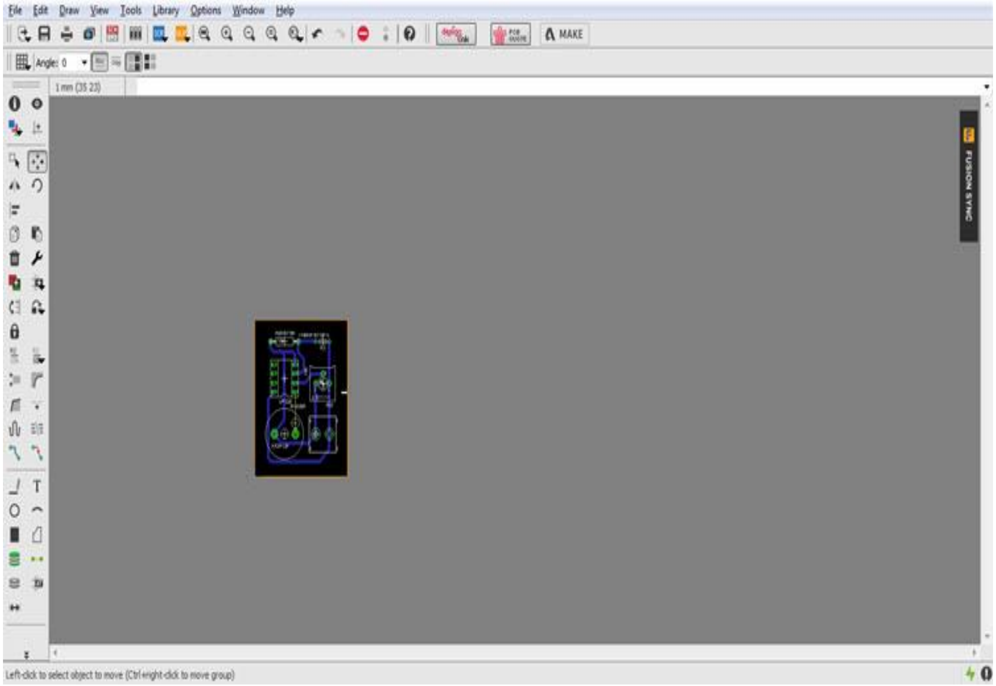
Start routing the traces from pin to pin. As you progress with routing, the air wires will start disappearing.



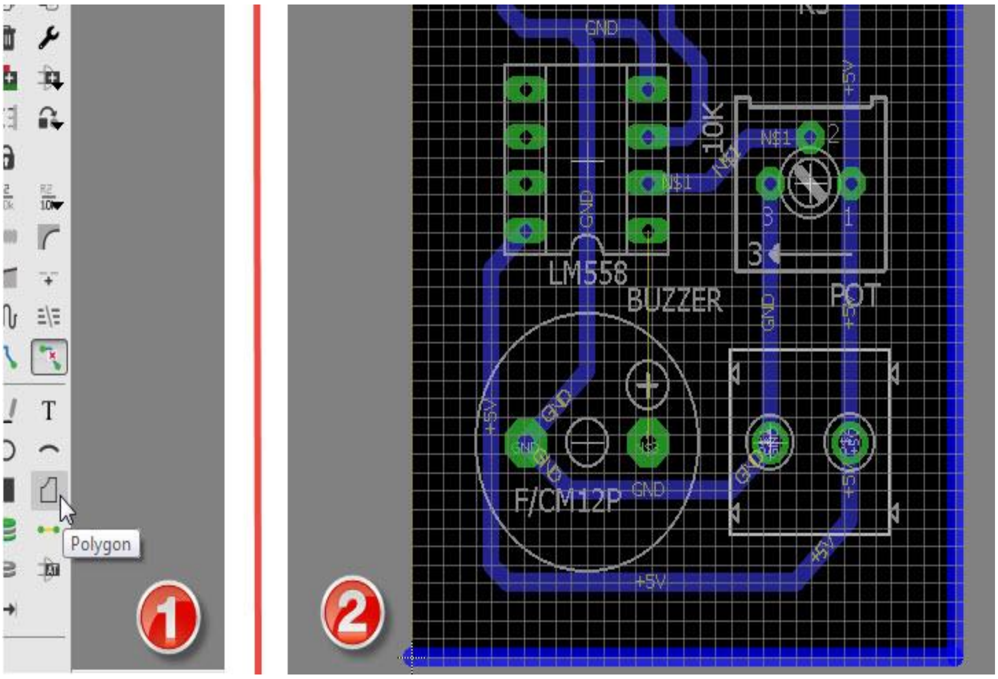
Complete all the connections and make sure that one trace do not interact with other (as this is considered as a short circuit).



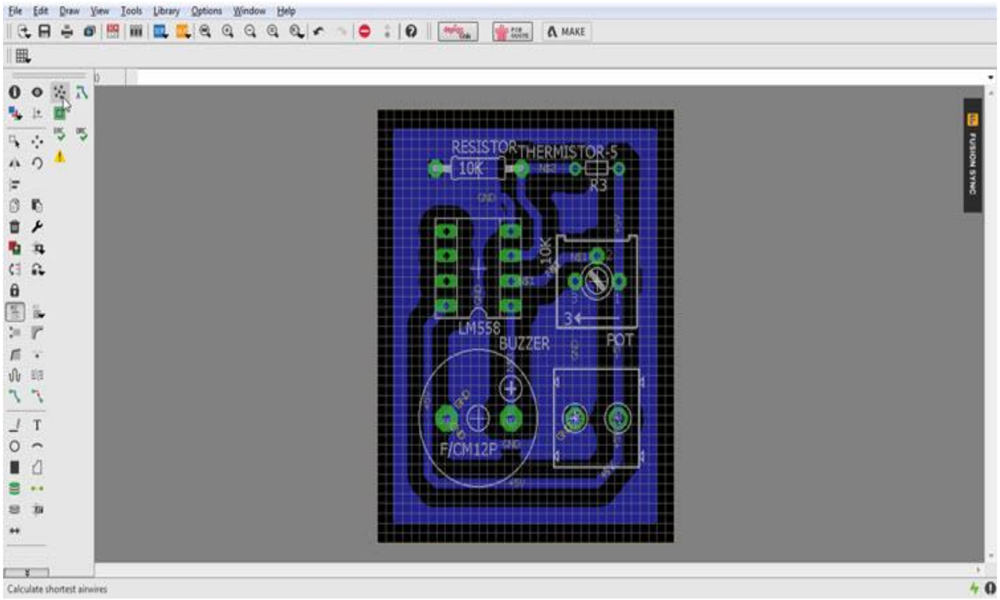
Adjust the size of the board as per the required dimensions and save the file. The board file will be save with .brd extension. Before proceeding with how to make your own PCB at home, I'll show you one more important step: the Ground Pour.



Select the polygon option from the side tool bar and start drawing the polygon along the edges of the board. Make sure that the signal layer is bottom. The starting and ending points of the polygon must meet.



You will get a dotted line along the edges of the board. Select name option and rename the polygon to GND. Hit ratsnest option from the side tool bar to see the ground pour.



The layout for the PCB is ready. The next step is to check for errors and generate the necessary Gerber files for sending them to PCB Manufacturers.

https://www.electronicshub.org/pcb-design-eagle/#:~:text=Eagle%20Free%2C%20as%20the%20name,layers%20and%20more%20board%20area.

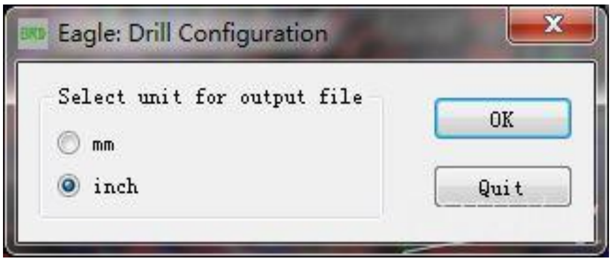
C) Generate GERBER File in EAGLE:

Before generating Gerber files from Eagle, firstly you need to confirm the silkscreen is on single side or double sides, because only the top silkscreen layer (tPlace and tNames) would be generated by default in Eagle software. When your PCB has double sides silkscreen, you need to add the bottom layer as well (Dimension, bPlace and bName). If there is any milling design in your brd file, please add extra outline layer and check the"Dimension" /span> "Milling". After following the 4 steps you will have all the necessary files needed for PCB manufacturing.

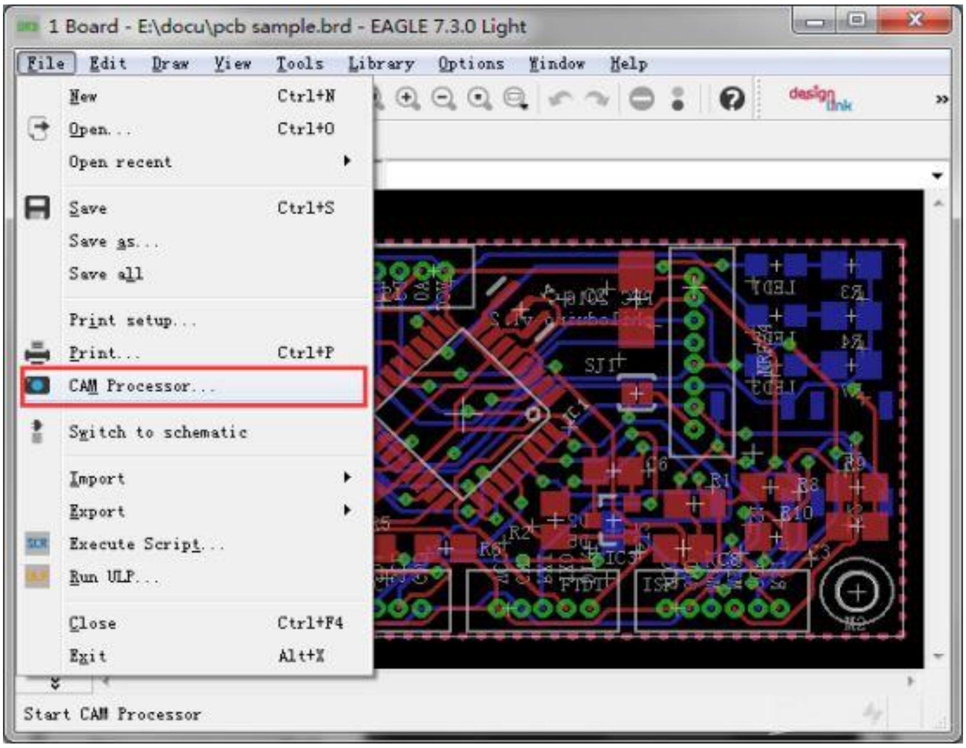
Generating Drill Files

To create a Gerber file in Eagle, you should run the drillcfg command first:

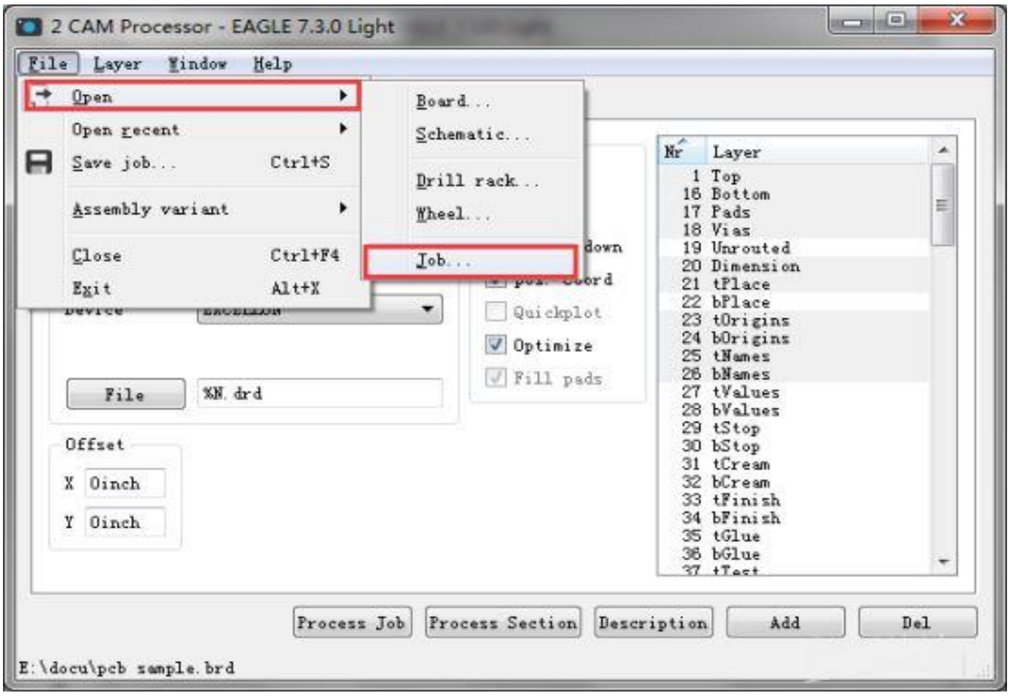
File -->Run ULP --> the pop-up "Drill Configuration" dialog box and click the "OK" button to generate the corresponding drill configuration file.



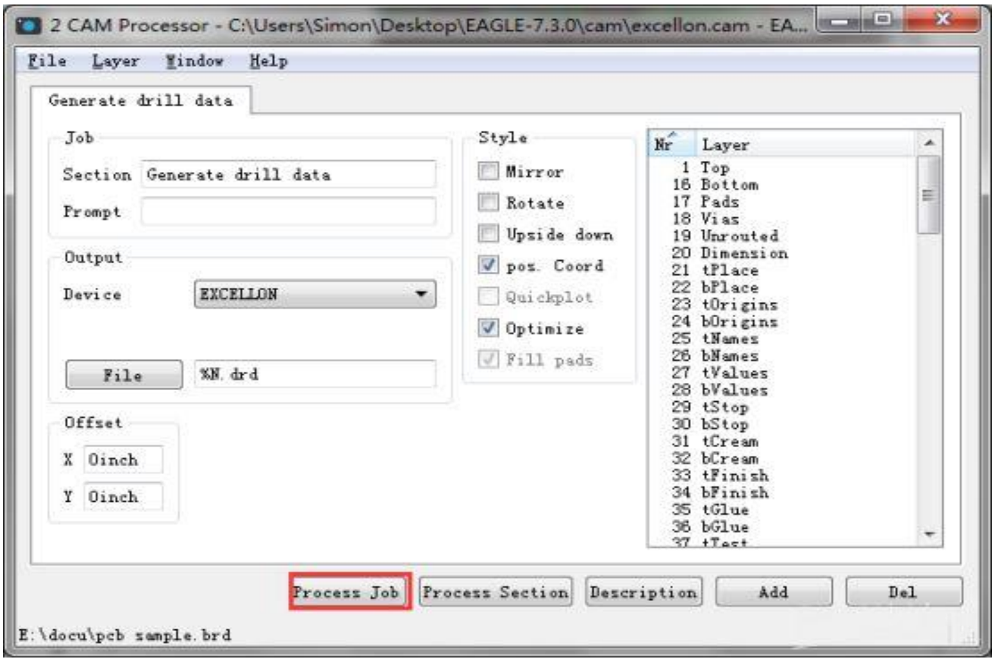
Open the .brd file, and then click the "File" menu to open the "CAM Processor" dialog box.



In the "CAM Processor" dialog box, select "File" -> "Open" -> "Job ..." command, open the "Open CAM Job" dialog box, select the "excellentone.cam" and click "Open" button.

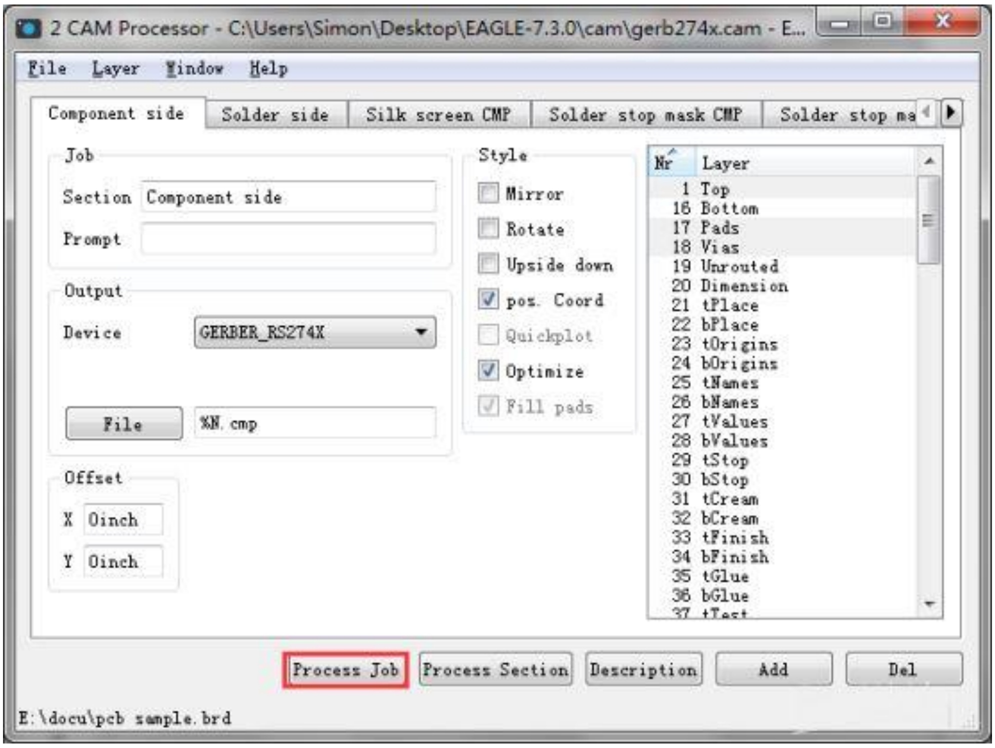


After loading it, click the "Process Job" command to generate the corresponding drill file:



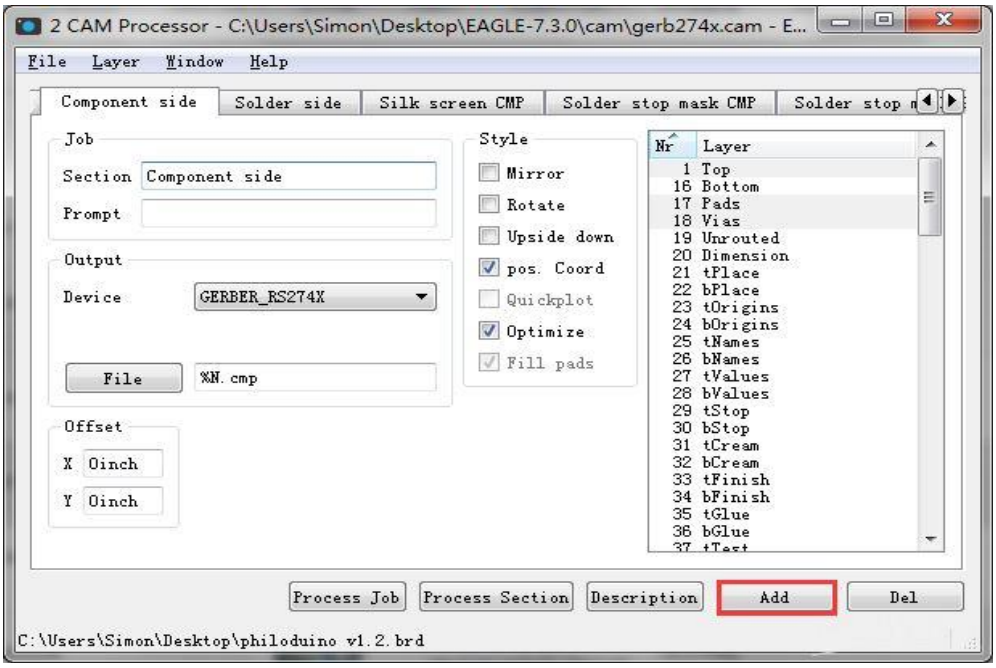
Generating Gerber Files

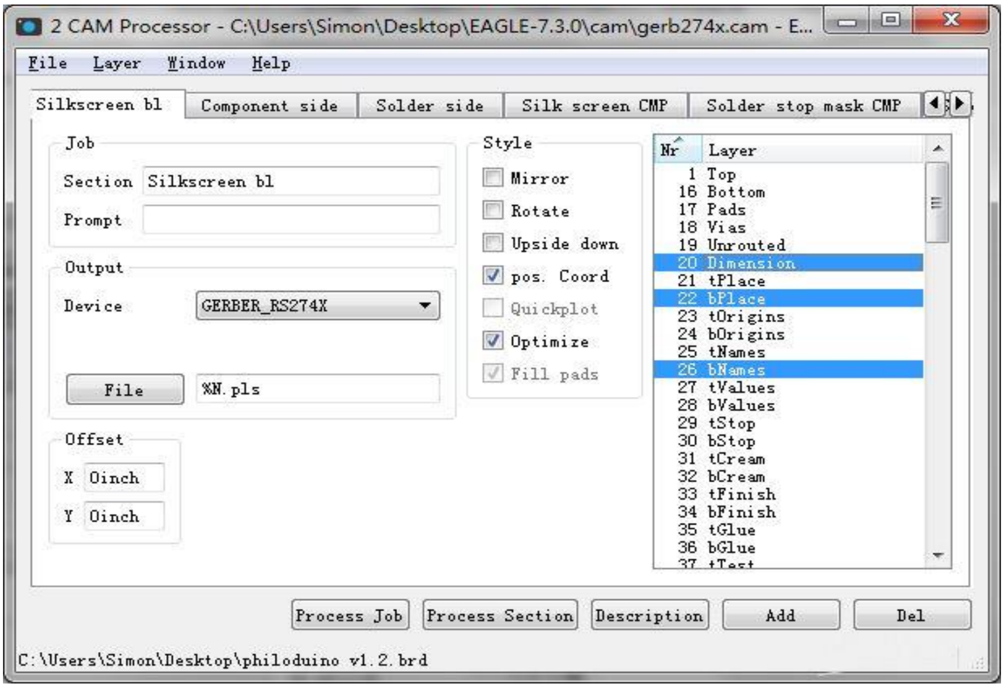
In the "CAM Processor" dialog box, select "File" -> "Open" -> "Job ..." command, open the "Open CAM Job" dialog box, select one of the "gerb274x.cam" and click "Open" After it is loaded, click the "Process Job" command to generate the corresponding Gerber file:



Please note: When it is the double layers silkscreen of your PCB, you need to add the bottom layer as well in the CAM Processor (20 Dimension,22 bPlace and 26 bNames).

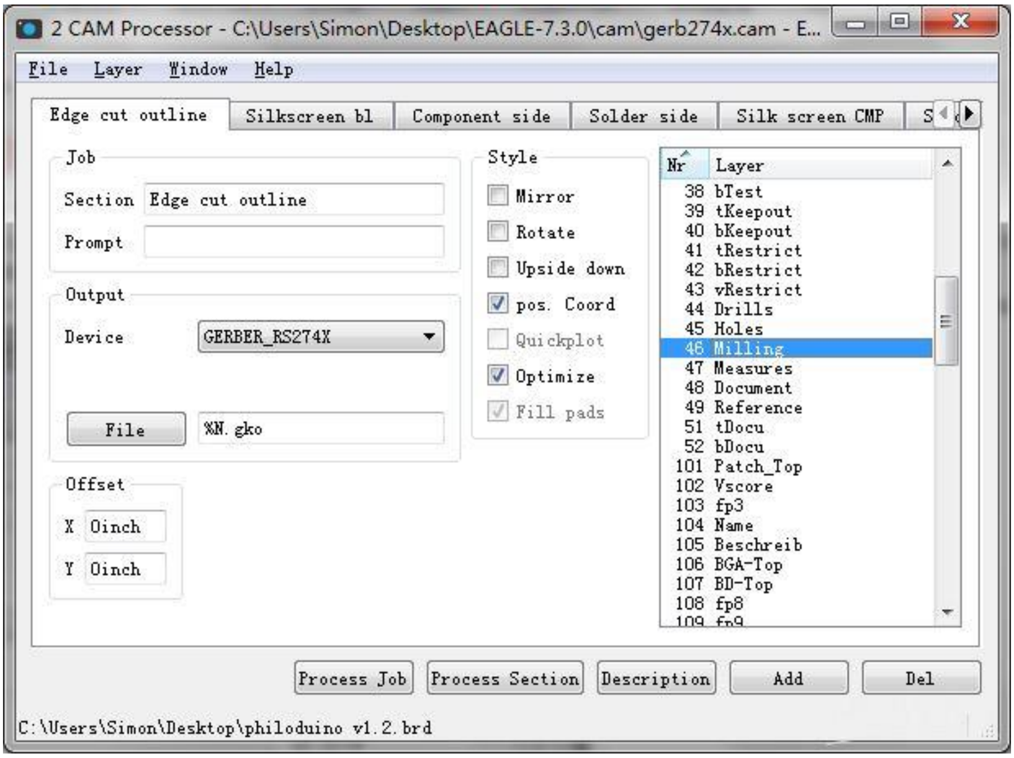
Click "Add" --> Change Section to something like “Silkscreen bl”-- > Change File to “%N.pls”--> Deselect all layers --> Select layers 20 Dimension, 22 bPlace and 26 bNames



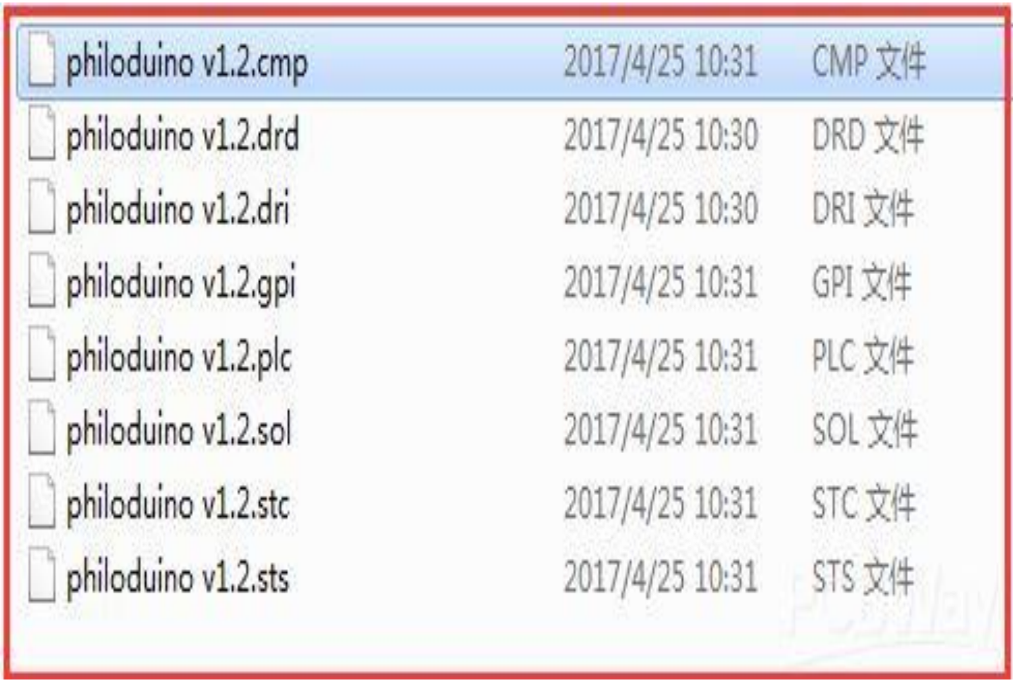


If there is any milling design in your brd file, please add extra outline layer and check the "20 Dimension" and "46 Milling".

Click "Add" --> Change Section to something like “Edge cut outline”-- > Change File to “%N.gko”--> Deselect all layers --> Select layers 20 Dimension and 46 milling.



After completing the above steps, in you. Brd file where the directory will generate some other documents --- Gerber files, which can be sent to the PCB manufacturers to produce.



But before you upload the Gerber files to PCBWay online system or other manufacturing fab, you should always check all the layers and look at them using a Gerber viewer to make sure everything is ok.

Compress all the files in a single .zip file

The final step is to compress all the files in a single .zip file, then you can send to manufacturer.

Link for how to make PCB Layout in Eagle:

https://www.pcbway.com/helpcenter/technical_support/Generate_Gerber_files_in_Eagle.html

https://www.youtube.com/watch?v=a1I6N7BVINA

https://www.youtube.com/watch?v=1AXwjZoyNno

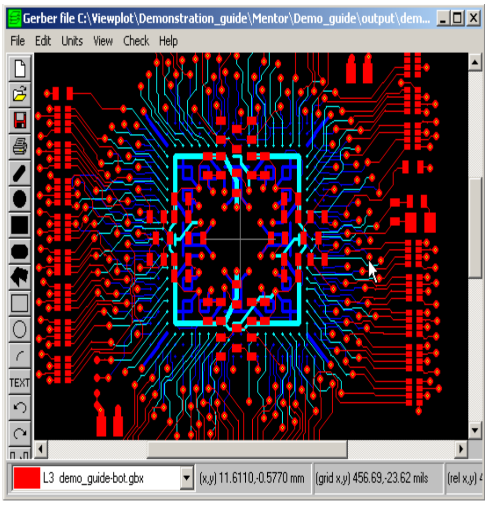
Link for how to make GERBER File in Eagle:

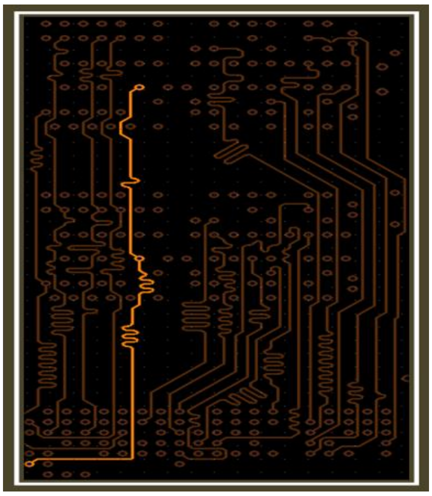
https://www.youtube.com/watch?v=eBwfRZkkPII

https://www.youtube.com/watch?v=-AuJfxQ199U

D) Check GERBER File in Viewplot:

Viewplot allows you to inspect various CAD formats, take measurements and manipulate data. Create high resolution PDF documentation within the graphical user interface or in batch mode.





Verify design data and or Design complexity.
http://www.viewplot.com/info_files/viewplot.html

7. Solar Power and Batteries

The main components of a solar power system are photovoltaic (PV) panels, a DC to AC power converter (called an inverter) and a rack system that holds the PV panels in place. Solar Photovoltaic (PV) panels are generally fitted on the roof. For good power generation panels should face in an easterly, northerly or westerly direction. South facing panels will work but generation is much reduced, especially in winter. Ideally panels should be tilted to a minimum of 10 degrees to ensure they are naturally cleaned by the rain.

Solar PV panels on the roofs of homes and businesses generate clean electricity when light energy hits them. This conversion takes place within the cells of solar panels which are specially fabricated, usually of silicon. Solar power generally has no moving parts with the exception of the inverter which sometimes comes with a small cooling fan.

Solar panels are different to solar hot water systems, which are also mounted on household roof-tops but use the heat from the sun to provide hot water for household uses, in a similar principle like a hose in summer contains hot water after a few hours in the sun.

The technology to convert sunlight into electricity was developed in the 19th century, but it was only in the second half of the 20th century that development accelerated behind the need to provide reliable supplies of electricity in remote locations, to satellites in space and to homes and businesses in Australia and around the world.

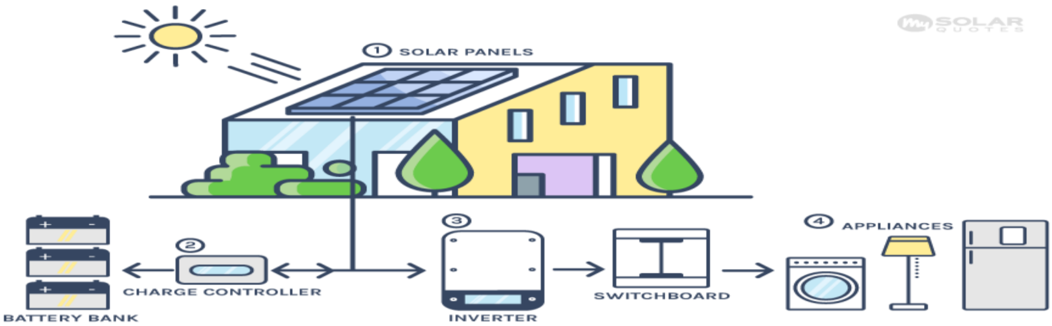
https://www.lgenergy.com.au/faq/did-you-know/what-is-a-solar-power-system#:~:text=What%20is%20a%20solar%20power,the%20PV%20panels%20in%20place.&text=This%20conversion%20takes%20place%20within,specially%20fabricated%2C%20usually%20of%20silicon.

- **HOW AN OFF-GRID SOLAR POWER SYSTEM WORKS**

An off-grid solar power system is similar to a grid-connected system, the difference being, an off-grid system uses batteries to store solar power. Excess power is not exported to the grid.

This is how an off-grid solar power system works:

1. DC electricity is created by the solar PV panels.
2. The DC electricity runs through a charge controller, controlling the voltage going into the batteries whilst protecting them.
3. When power is required by the home or business it is drawn from the batteries through the inverter, which is then converted into AC electricity.
4. This electricity can be used by most appliances. However, 12-volt appliances can use power directly from the battery without having to go through the inverter.
   An optional backup generator can be added to the system e.g. a diesel generator when you need to use more power.



https://mysolarquotes.co.nz/about-solar-power/residential/how-an-off-grid-system-works/

- **Type of Battery's:**
  A battery is a collection of one or more cells that go under chemical reactions to create the flow of electrons within a circuit. There is lot of research and advancement going on in battery technology, and as a result, breakthrough technologies are being experienced and used around the world currently. Batteries came into play due to the need to store generated electrical energy. As much as a good amount of energy was being generated, it was important to store the energy so it can be used when generation is down or when there is a need to power standalone devices which cannot be kept tethered to the supply from the mains. Here it should be noted that only DC can be stored in the batteries, AC current can't be stored.

  Battery cells are usually made up of three main components;

1. The Anode (Negative Electrode)
2. The Cathode (Positive Electrode)
3. The electrolytes

The anode is a negative electrode that produces electrons to the external circuit to which the battery is connected. When batteries are connected, an electron build up is initiated at the anode which causes a potential difference between the two electrodes. The electrons naturally then try to redistribute themselves, this is prevented by the electrolyte, so when an electrical circuit is connected, it provides a clear path for the electrons to move from the anode to the cathode thereby powering the circuit to which it is connected.

Batteries generally can be classified into different categories and types, ranging from chemical composition, size, form factor and use cases, but under all of these are two major battery types;

1. Primary Batteries
2. Secondary Batteries

Primary Batteries:

Primary batteries are batteries that **cannot be recharged** once depleted. Primary batteries are made of electrochemical cells whose electrochemical reaction cannot be reversed.

They are commonly used in standalone applications where charging is impractical or impossible. A good example of which is in military grade devices and battery powered equipment.

Secondary Batteries:

Secondary batteries are batteries with electrochemical cells whose chemical reactions can be reversed by applying a certain voltage to the battery in the reversed direction. Also referred to as **rechargeable batteries**, secondary cells unlike primary cells can be recharged after the energy on the battery has been used up.
Small capacity secondary batteries are used to power portable electronic devices like **mobile phones**, and other gadgets and appliances while heavy-duty batteries are used in powering diverse **electric vehicles** and other high drain applications like load levelling in electricity generation. They are also used as standalone power sources alongside **Inverters to supply electricity**.
Secondary batteries can be further classified into several other types based on their chemistry**.** This is very important because the chemistry determines some of the attributes of the battery including its specific energy, cycle life, shelf life, and price to mention a few.

There are basically four major chemistries for rechargeable batteries;

1. Lithium-ion(Li-ion)
2. Nickel Cadmium(Ni-Cd)
3. Nickel-Metal Hydride(Ni-MH)
4. Lead-Acid

Nickel-Cadmium Batteries

The nickel–cadmium battery (NiCd battery or NiCad battery) is a type of rechargeable battery which is developed using nickel oxide hydroxide and metallic cadmium as electrodes. Ni-Cd batteries excel at maintaining voltage and holding charge when not in use. However, NI-Cd batteries easily fall a victim of the dreaded “memory” effect when a partially charged battery is recharged, lowering the future capacity of the battery.

Some of the properties of Nickel-Cadmium batteries are listed below.

- Specific Energy: 40-60W-h/kg
- Energy Density: 50-150 W-h/L
- Specific Power: 150W/kg
- Charge/discharge efficiency: 70-90%
- Self-discharge rate: 10%/month
- Cycle durability/life: 2000cycles

Nickel-Metal Hydride Batteries

Nickel metal hydride (Ni-MH) is another type of chemical configuration used for rechargeable batteries. The chemical reaction at the positive electrode of batteries is similar to that of the nickel–cadmium cell (NiCd), with both battery type using the same nickel oxide hydroxide (NiOOH). However, the negative electrodes in Nickel-Metal Hydride use a hydrogen-absorbing alloy instead of cadmium which is used in NiCd batteries

Below are some of the properties of batteries based on the Nickel-metal hydride chemistry;

- Specific Energy: 60-120h/kg
- Energy Density: 140-300 Wh/L
- Specific Power: 250-1000 W/kg
- Charge/discharge efficiency: 66% - 92%
- Self-discharge rate: 1.3-2.9%/month at 20ºC
- Cycle Durability/life: 180 -2000

Lithium-ion Batteries

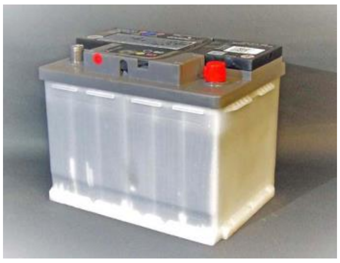
Lithium ion batteries are one of the most popular types of rechargeable batteries. They are found in different portable appliances including mobile phones, smart devices and several other battery appliances used at home. They also find applications in aerospace and military applications due to their lightweight nature.

Some of the attributes of lithium ion batteries are listed below;

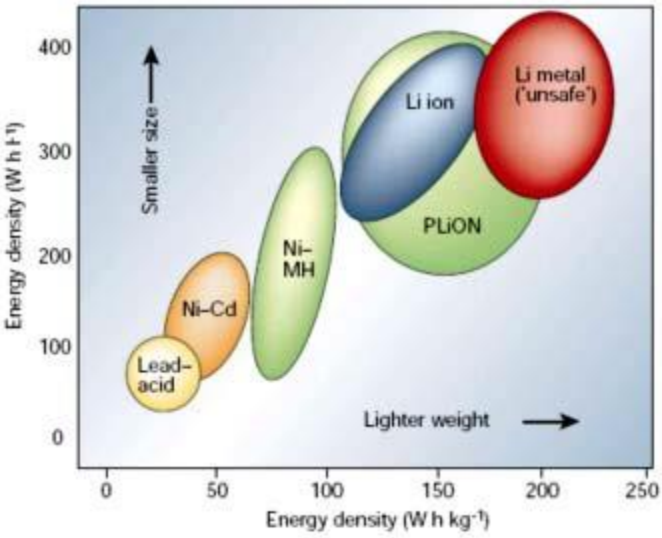
- Specific Energy: 100: 265W-h/kg
- Energy Density: 250: 693 W-h/L
- Specific Power: 250: 340 W/kg
- Charge/discharge percentage: 80-90%
- Cycle Durability: 400: 1200 cycles
- Nominal cell voltage: NMC 3.6/3.85V

Lead-Acid Batteries

Lead acid batteries are a low-cost reliable power workhorse used in heavy duty applications. They are usually very large and because of their weight, they're always used in non-portable applications such as solar-panel energy storage, vehicle ignition and lights, backup power and load levelling in power generation/distribution. The lead-acid is the oldest type of rechargeable battery and still very relevant and important into today's world.



Each of these batteries has its area of best fit and the image below is to help choose between them.



Selecting the right battery for your application:

One of the main problems hindering technology revolutions like IoT is power, battery life affects the successful deployment of devices that require long battery life and even though several power management techniques are being adopted to make the battery last longer, a compatible battery must still be selected to achieve the desired outcome.

Below are some factors to consider when selecting the right type of battery for your project.

1. **Energy Density:** The energy density is the total amount of energy that can be stored per unit mass or volume. This determines how long your device stays on before it needs a recharge.

2. Power Density: Maximum rate of energy discharge per unit mass or volume. Low power: laptop, i-pod. High power: power tools.

3. Safety: It is important to consider the temperature at which the device you are building will work. At high temperatures, certain battery components will breakdown and can undergo exothermic reactions. High temperatures generally reduces the performance of most batteries.

4. Life cycle durability: The stability of energy density and power density of a battery with repeated cycling (charging and discharging) is needed for the long battery life required by most applications.

5. Cost: Cost is an important part of any engineering decisions you will be making. It is important that the cost of your battery choice is commensurate with its performance and will not increase the overall cost of the project abnormally.

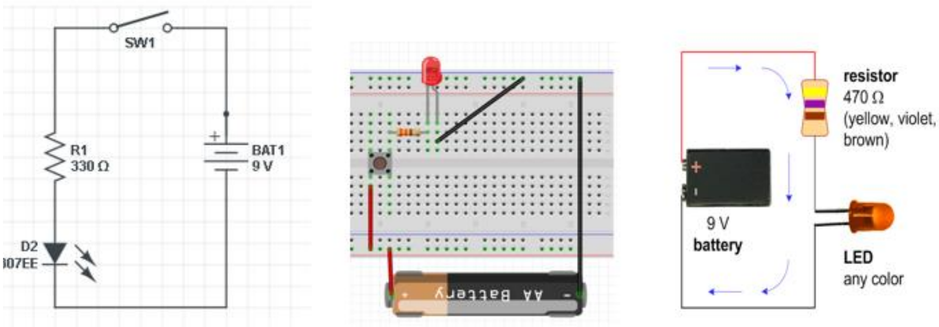
https://circuitdigest.com/article/different-types-of-batteries

8. Recommended Practical Assignments

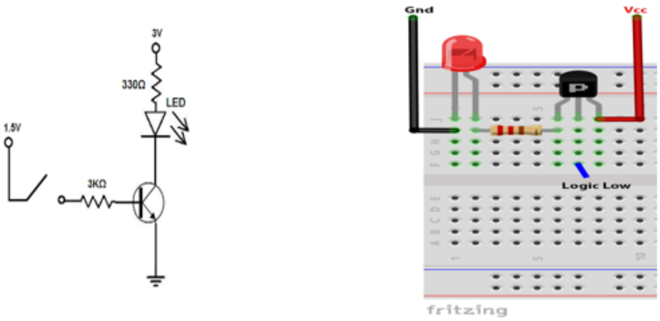
Online Training Material: https://github.com/coppercloud-iotech/sppu-esdlab-fdp-teachingaids

Arduino Source Code: https://github.com/coppercloud-iotech/sppu-esdlab-fdp-teachingaids/tree/master/Arduino%20Source%20Code

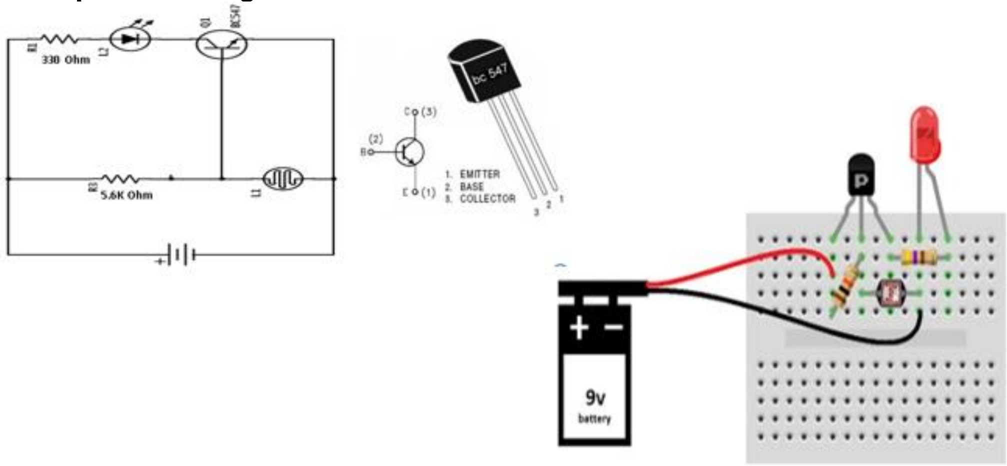
1. **Example Lab1: Basic Resistor + LED Circuit:**



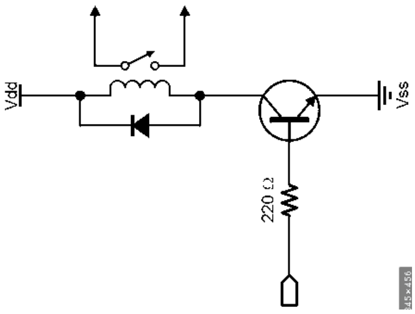
2. **Example Lab2: Transistor-driven LED Circuit:**



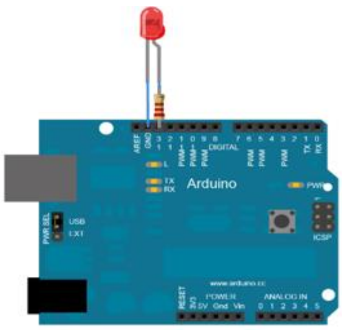
3. **Example Lab3: Light/Dark Detector Circuit:**



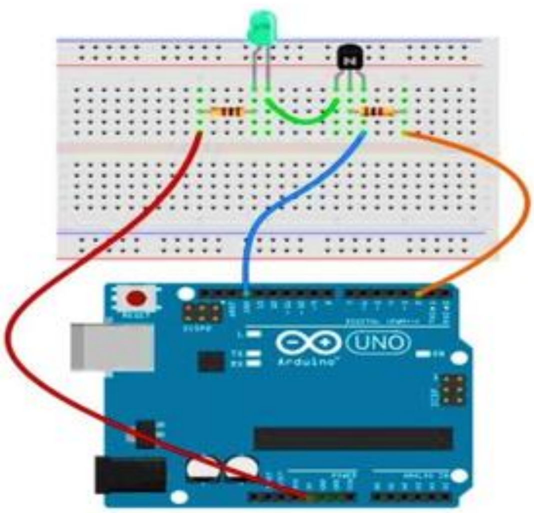
4. **Example Lab4: Relay-Driver Circuit:**

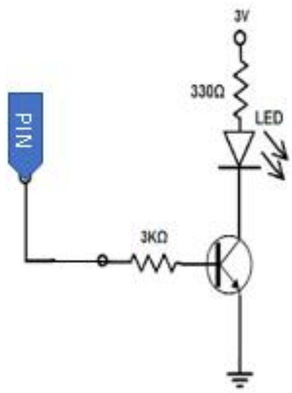


5. **Example Lab5: Arduino with Blinking LED:**

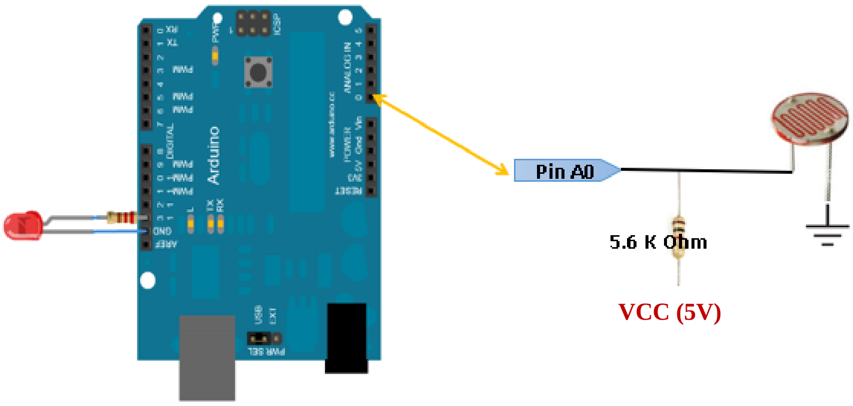


6. **Example Lab6: Arduino with Transistor Driver for LED:**

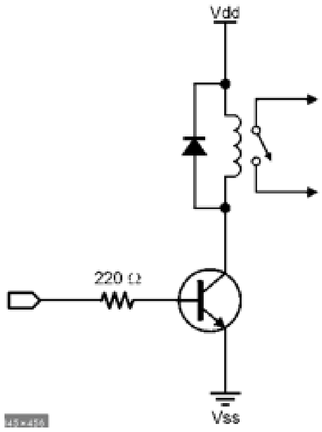


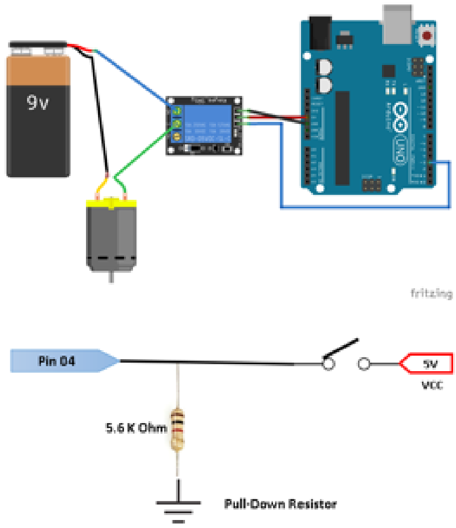


7. **Example Lab7: Arduino with Analog Input (LDR):**

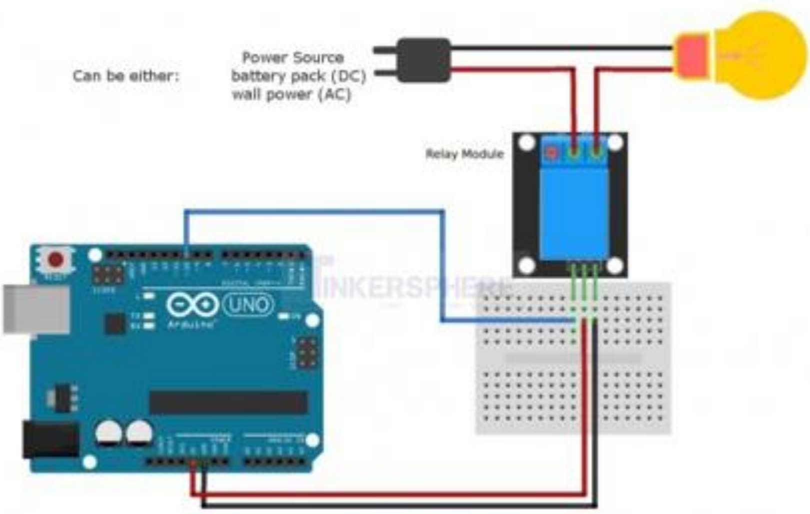


8. **Example Lab8: Arduino with Relay (DC Motor):**

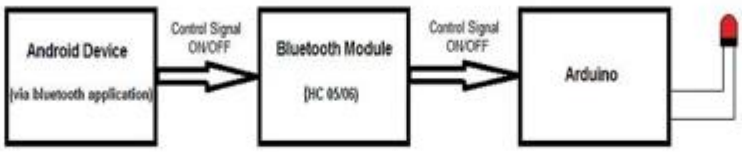


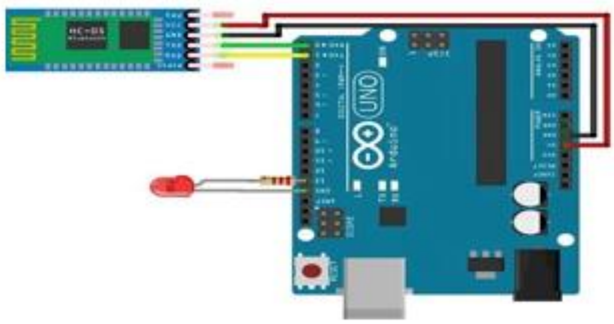


9. **Example Lab9: Arduino with Relay (AC Bulb):**



10. Example Lab10: Arduino Wireless Communication (with HC05 Bluetooth Module)

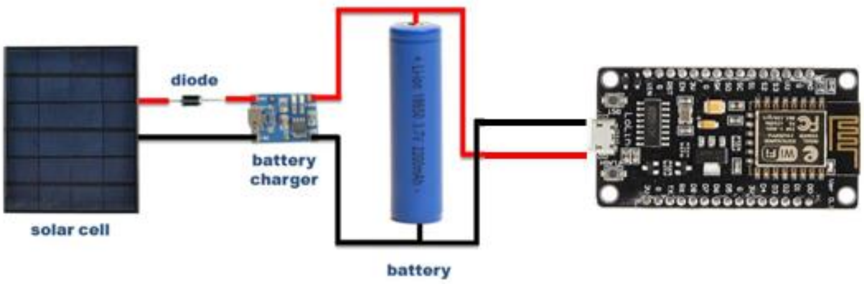




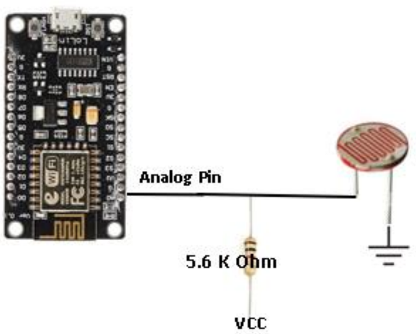
11. Example Lab11: PCB Schematic, Simulation & Layout

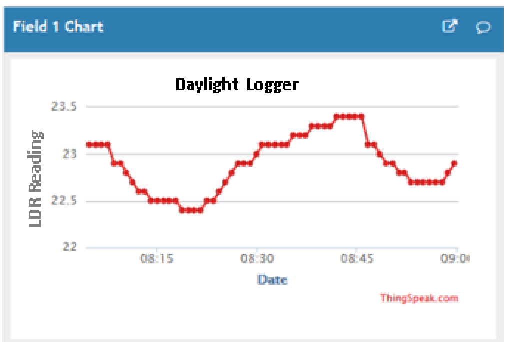
Please refer to the detailed **Section 6**

12. Example Lab 12: Run Microcontroller on Solar power



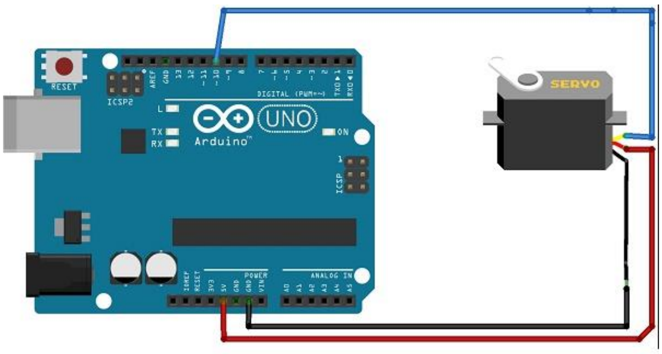
13. Example Lab 13 (Extra): Beginning of IoT – Send Daylight readings (LDR) to ThingSpeak





URL: thingspeak.com

14. Example Lab 14 (Extra): Servo Motor with Arduino



Authored by : CopperCloud IOTech Technical Team
Abhijeet Deogirikar
Yogesh Yewale

For Support, please contact: iot@coppercloud.in
www.coppercloud.in
https://www.linkedin.com/company/coppercloud-iot/