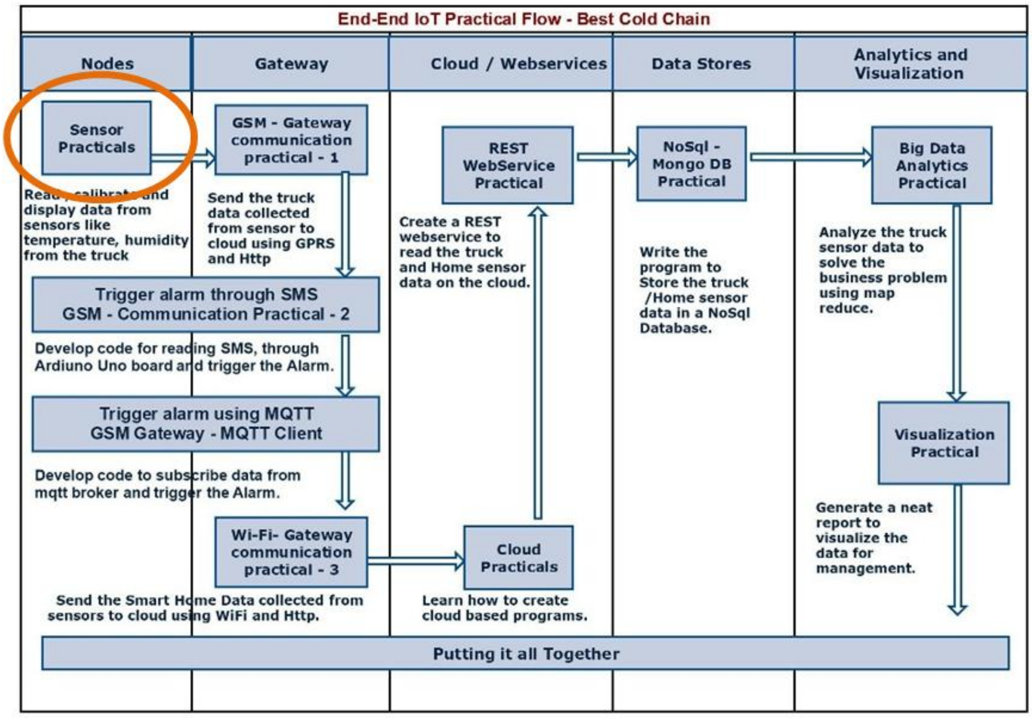
User Manual – Practical - 1

Sensor Data Displaying on LCD and Serial Monitor using Arduino UNO

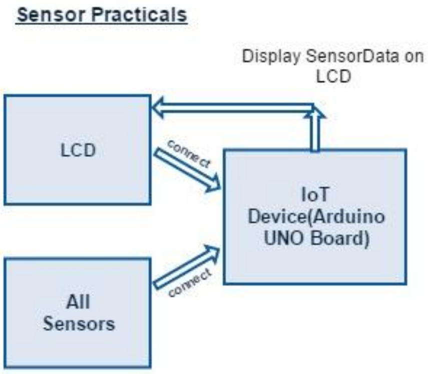
Practical's Objective:

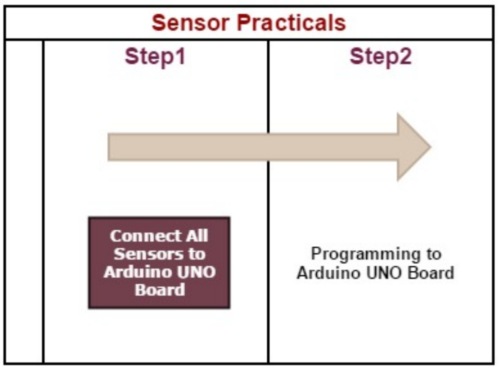
To capture data from different sensors used in Best cold chain and display the same on LCD and Serial Monitor using Arduino UNO.

End-End IoT Flow diagram:

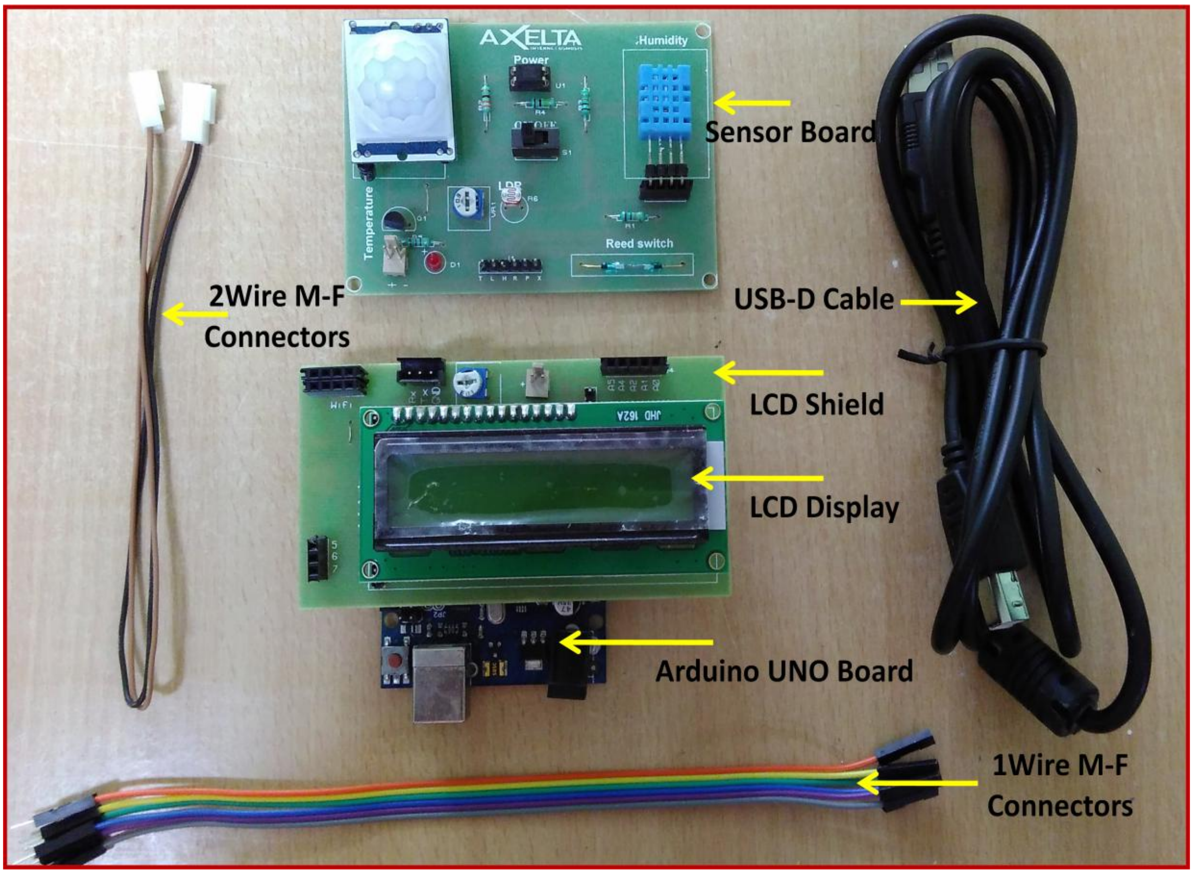

End-End IoT Practical Flow - Best Cold Chain
Nodes
Gateway
Cloud / Webservices
Data Stores
Analytics and Visualization
Sensor Practicals
Read, calibrate and display data from sensors like temperature, humidity from the truck
GSM - Gateway communication practical - 1
Send the truck data collected from sensor to cloud using GPRS and Http
REST WebService Practical
Create a REST webservice to read the truck and Home sensor data on the cloud.
NoSql - Mongo DB Practical
Write the program to Store the truck /Home sensor data in a NoSql Database.
Big Data Analytics Practical
Analyze the truck sensor data to solve the business problem using map reduce.
Trigger alarm through SMS
GSM - Communication Practical - 2
Develop code for reading SMS, through Ardiuno Uno board and trigger the Alarm.
Trigger alarm using MQTT
GSM Gateway - MQTT Client
Develop code to subscribe data from mqtt broker and trigger the Alarm.
Visualization Practical
Generate a neat report to visualize the data for management.
Wi-Fi- Gateway communication practical - 3
Cloud Practicals
Send the Smart Home Data collected from sensors to cloud using WiFi and Http.
Learn how to create cloud based programs.
Putting it all Together

Sensor Practical Hardware flow Diagram:




Sensor Practicals
Step1
Step2
Connect All Sensors to Arduino UNO Board
Programming to Arduino UNO Board

H/w Requirements:



Sensor Board notations:

T – Temperature Sensor

L – Light Dependent Resistor (LDR)

R – Magnetically operated Reed Switch

H – Humidity Sensor

P – Power Switch based on Optocoupler

X – PIR(Passive Infrared Sensor)

Software Requirement:

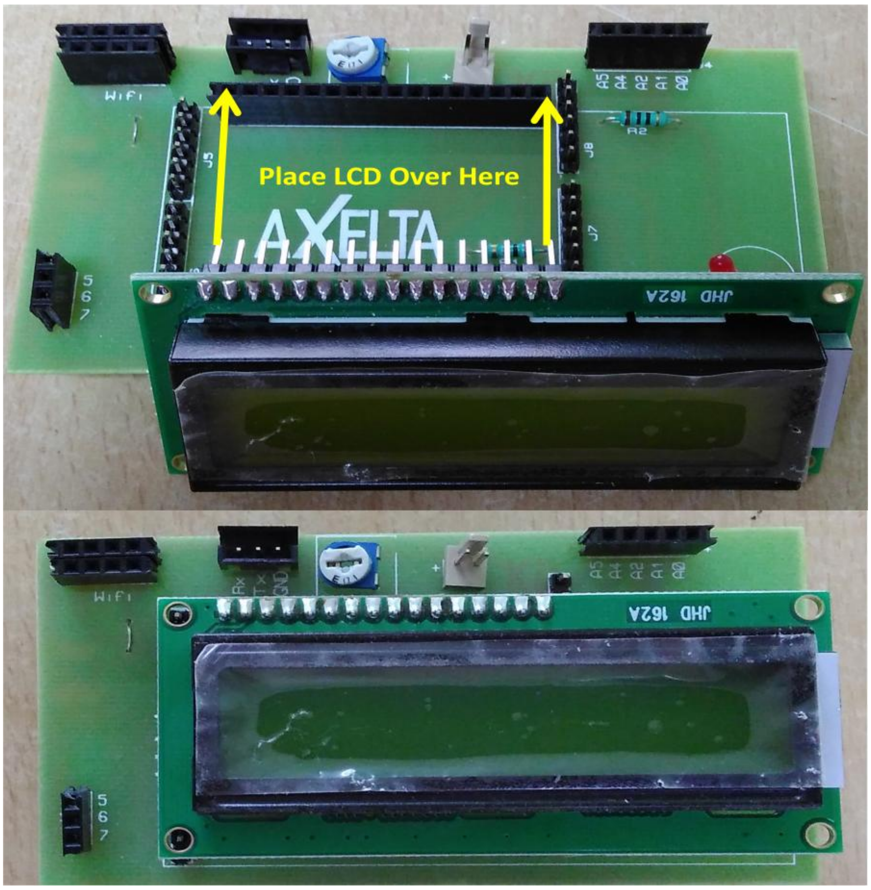
Aurdino IDE

DHT11 Library

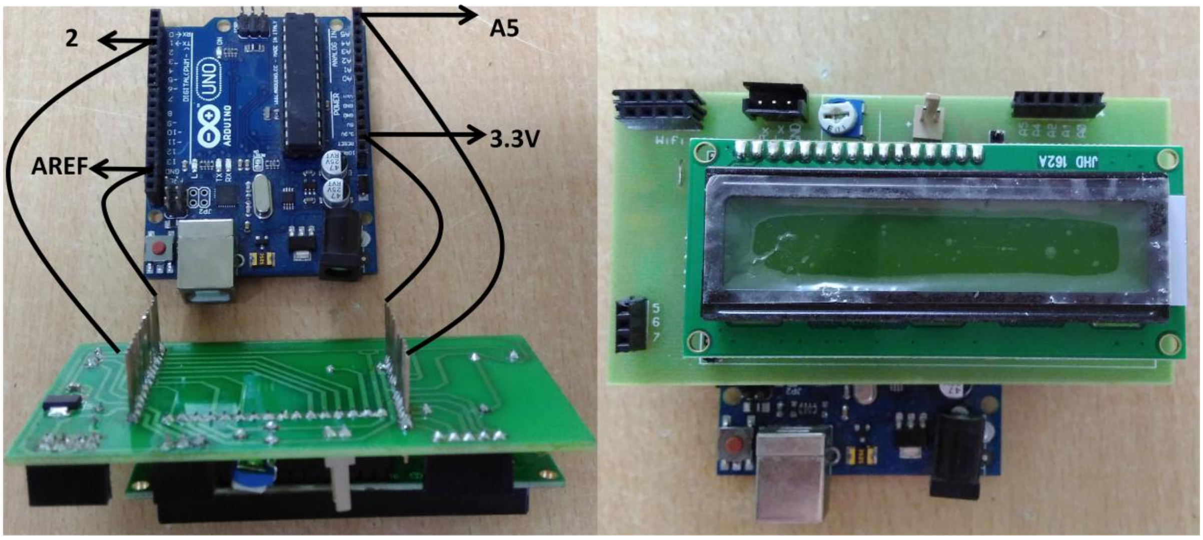
Connecting LCD to Arduino UNO Board:

Note : LCD & Shield is already connected to Arduino in the dispatched kit. Do not remove it.

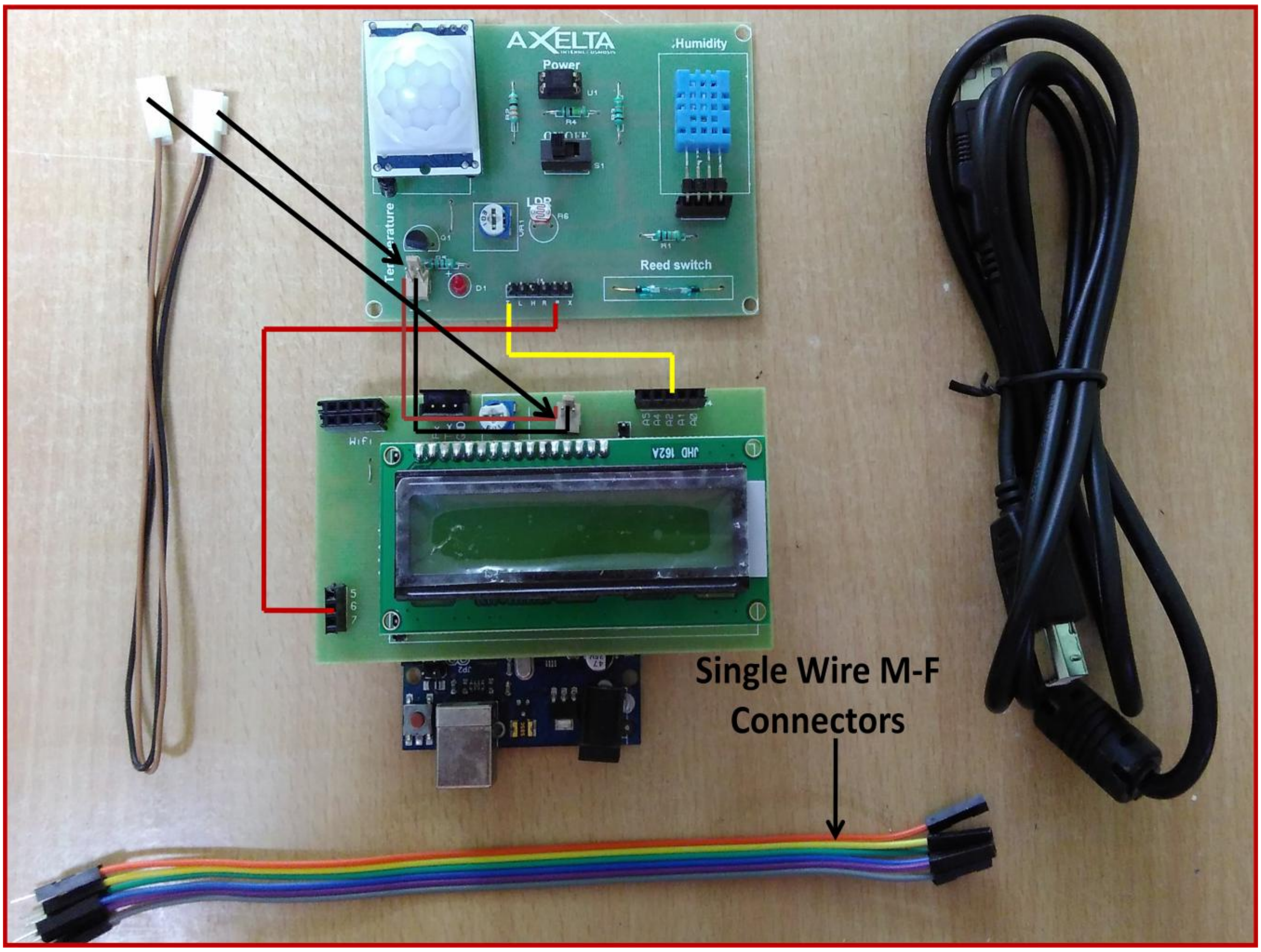
In case if you removed it then follow the steps given below to reconnect it.



Insert the LCD shield in the Arduino UNO Board as shown below LCD shield covers Pin 2-AREF on left side and Pin Analog5 – 3V3 on right side, while connecting make sure you are matching the gaps between male connectors back of LCD Shield.



Connecting Sensors to Arduino UNO Board :



Single Wire M-F Connectors

Sensor Board	**LCD Shield on ARDUINO**
T (Temp sensor)	**A2 (**Analog i/p pin)
P (Power Sensor)	**6 (**Digital input pin)

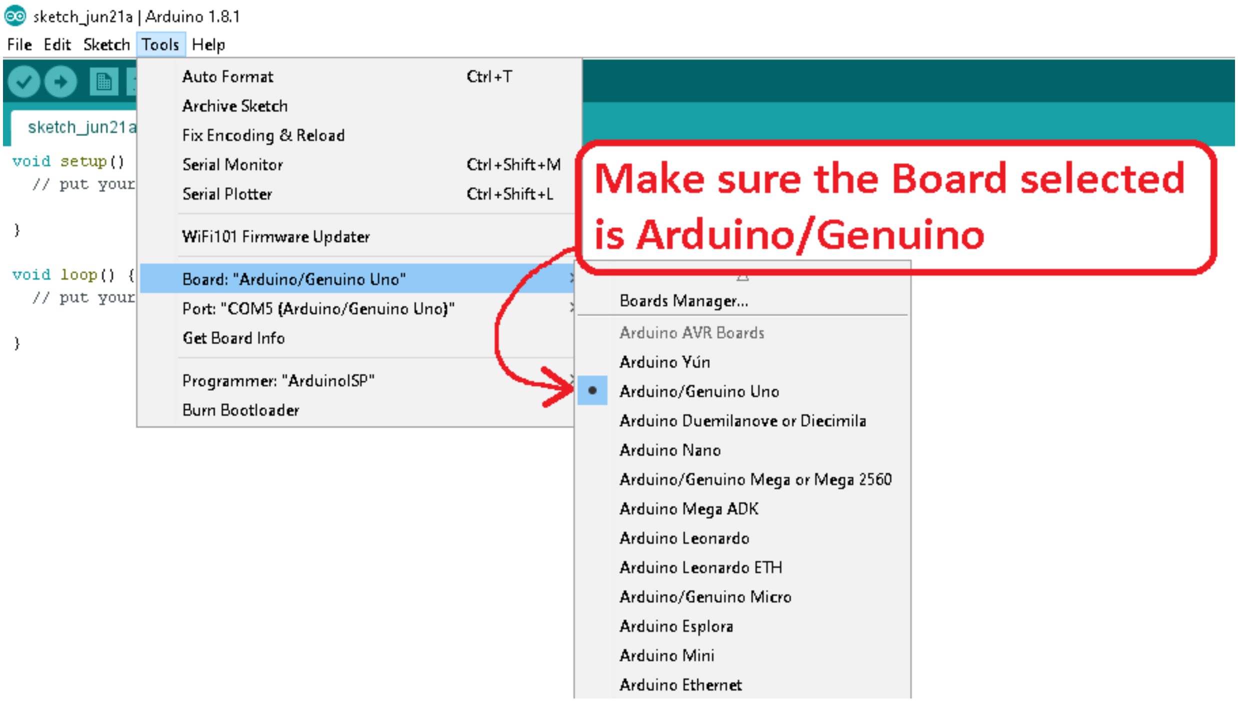
Using 2 Pin wire connector	
Sensor Board	**LCD Shield on ARDUINO**
+ (Vcc)	**+ (Vcc)**
- (GND)	**- (GND)**

- Connect **T** on sensor board to **A2** pin of the LCD Shield also connect **P** to pin **6**.
- Connect Power supply to sensor Board using a **2 pin wire** connector from the LCD Shield connected to Arduino UNO board as shown in picture.
- Connect **USB cable** from Arduino UNO board to PC.

To perform the Practical follow the steps given below.

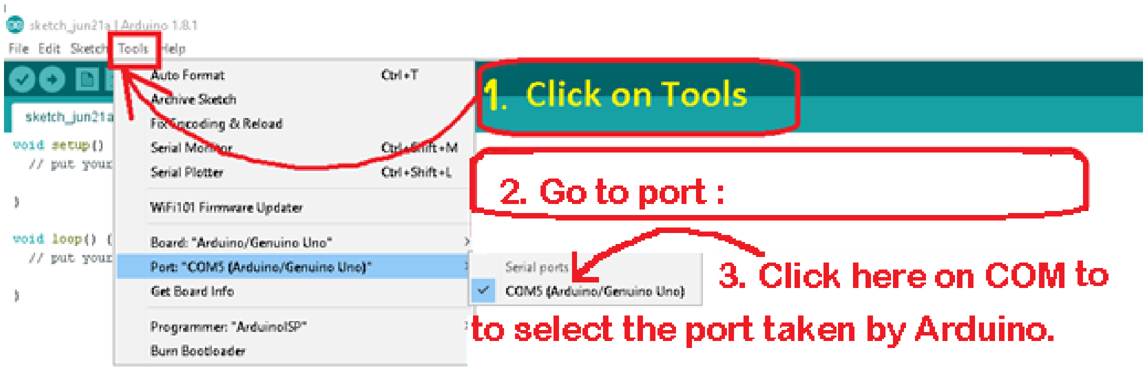
Now connect your Arduino Board to PC via USB Cable given to you in the kit.

1. Open Arduino IDE Go to the **Tools > Board**.



Make sure the Board selected is Arduino/Genuino

2. Go to **Tools > Port** do the following.



1. Click on Tools

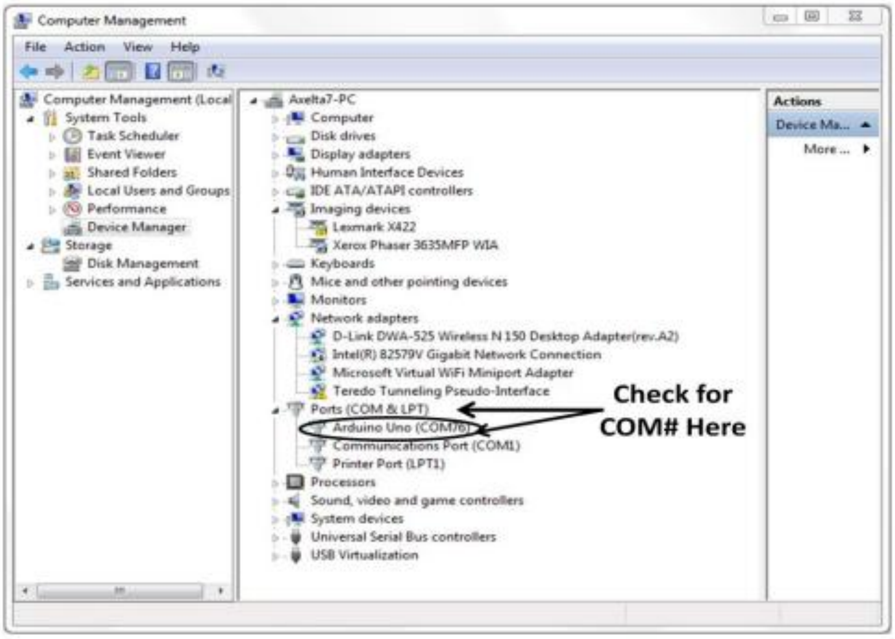
2. Go to port :

3. Click here on COM to to select the port taken by Arduino.

4. Please make sure to select the COM Port where you will find " (Arduino/Genuino Uno) "

Optional : You can verify it in My Computer >Properties >Device manager > COM & LPT

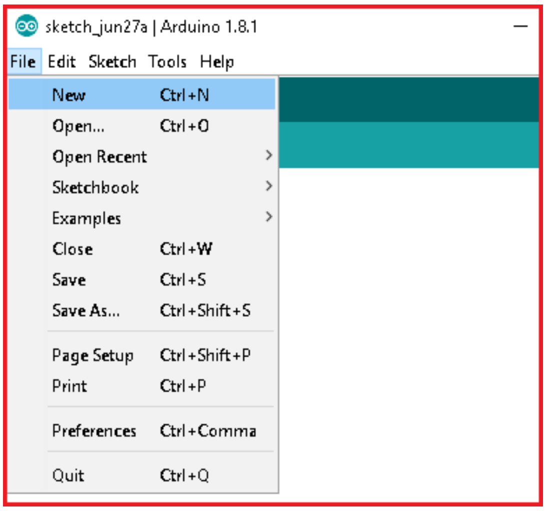
Open the “**Device Manager**” (Right click on **My Computer > Manage > DeviceManager**) and check **“Ports (COM & LPT)”** where you can find “Arduino Uno(COM#)” with a port/COM number. This COM should be selected in the Arduino IDE in ports.

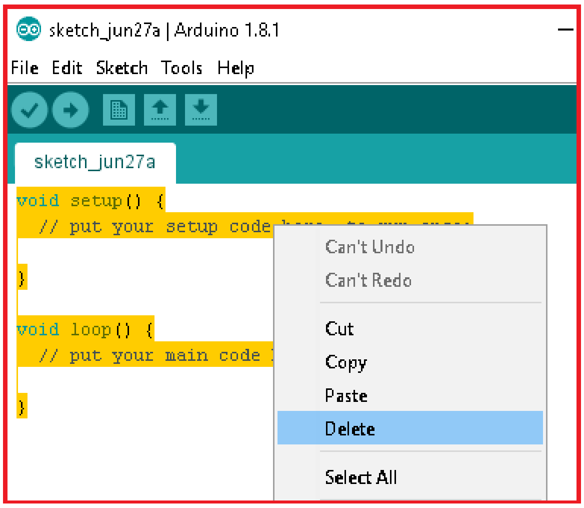


Check for COM# Here

Programming:

When you will open **Arduino IDE ,** You may see the following code or any other code already written. **First delete this entire code** & then write your own program or do the following :

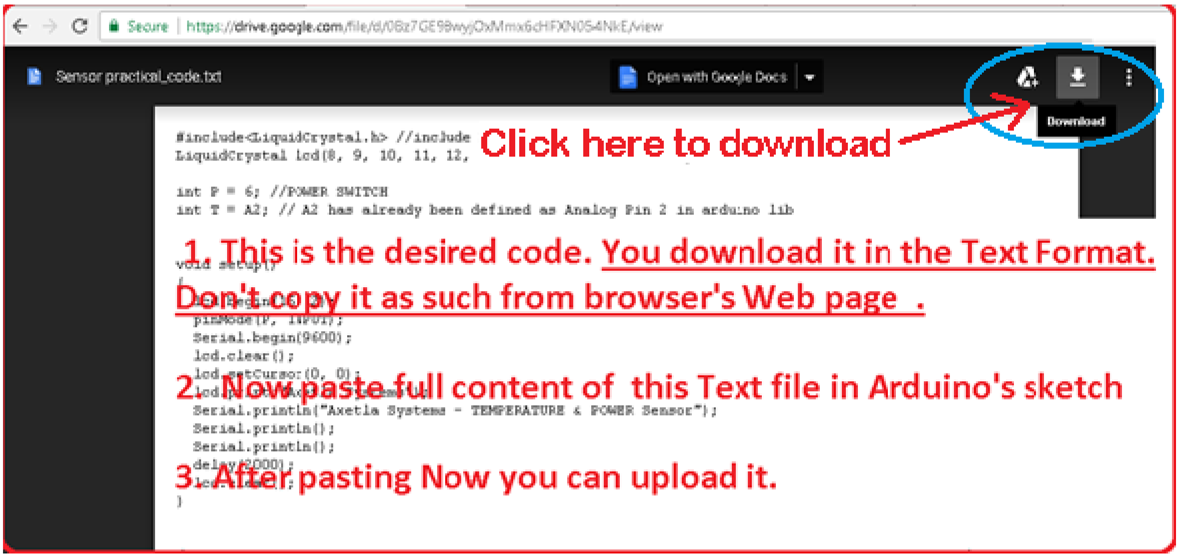




```
void setup() {
  // put your setup code

}

void loop() {
  // put your main code

}
```

Create a new sketch [program]. You can find the “**Sensor practical_code**” code by the link:

https://drive.google.com/open?id=0Bz7GE98wyjOxMmx6cHFXN054NkE


Secure | https://drive.google.com/file/d/0Bz7GE9BwyjOxMmx6cHFXN054NkE/view
Sensor practical_code.txt
Open with Google Docs
Download
Click here to download

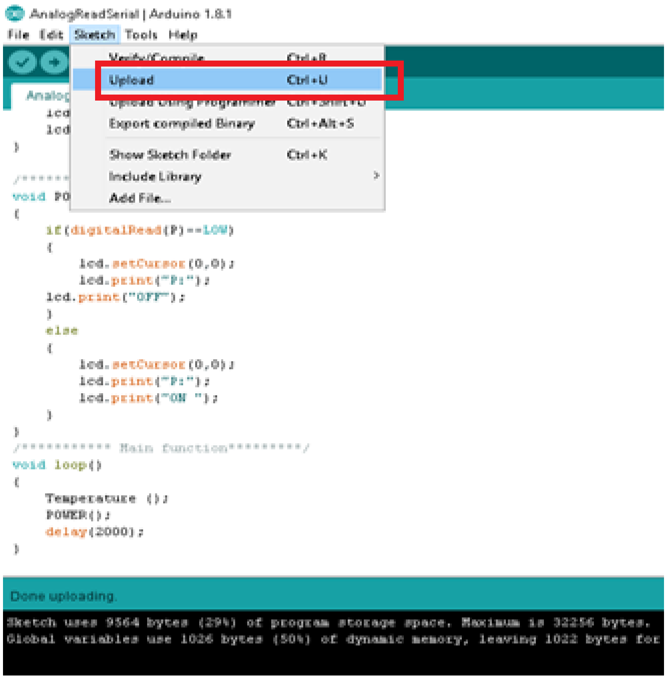
```
#include<LiquidCrystal.h> //include
LiquidCrystal lcd(8, 9, 10, 11, 12,

int P = 6; //POWER SWITCH
int T = A2; // A2 has already been defined as Analog Pin 2 in arduino lib

void setup()
{
  lcd.begin(16, 2);
  pinMode(P, INPUT);
  Serial.begin(9600);
  lcd.clear();
  lcd.setCursor(0, 0);
  lcd.[illegible]
  Serial.println("Axetla Systems - TEMPERATURE & POWER Sensor");
  Serial.println();
  Serial.println();
  delay(2000);
  lcd.clear();
}
```

1. This is the desired code. You download it in the Text Format. Don't copy it as such from browser's Web page .

2. Now paste full content of this Text file in Arduino's sketch

3. After pasting Now you can upload it.

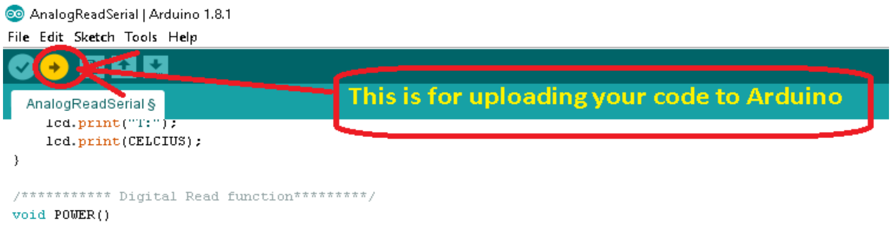
For uploading the code by Click on sketch and select “Upload” or Press <Control +U>


AnalogReadSerial | Arduino 1.8.1
File Edit Sketch Tools Help
Verify/Compile Ctrl+R
Upload Ctrl+U
Upload Using Programmer Ctrl+Shift+U
Export compiled Binary Ctrl+Alt+S
Show Sketch Folder Ctrl+K
Include Library
Add File...
Analog

```
    lcd
    lcd
}

/******
void PO
{
    if(digitalRead(P)==LOW)
    {
        lcd.setCursor(0,0);
        lcd.print("P:");
    lcd.print("OFF");
    }
    else
    {
        lcd.setCursor(0,0);
        lcd.print("P:");
        lcd.print("ON ");
    }
}
/*********** Main function*********/
void loop()
{
    Temperature ();
    POWER();
    delay(2000);
}
```

Done uploading.

```
Sketch uses 9564 bytes (29%) of program storage space. Maximum is 32256 bytes.
Global variables use 1026 bytes (50%) of dynamic memory, leaving 1022 bytes for
```



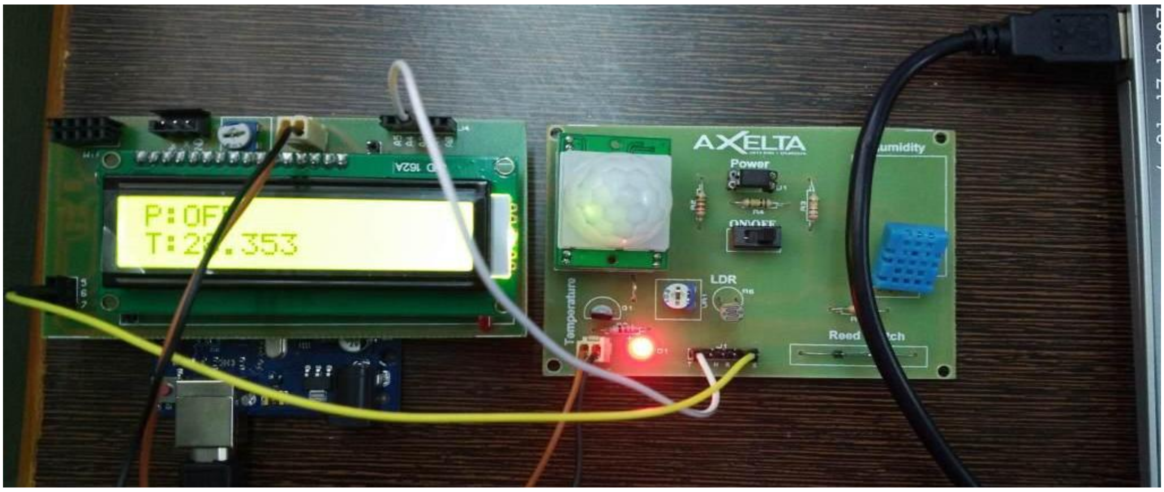
This is for uploading your code to Arduino

```
    lcd.print("T:");
    lcd.print(CELCIUS);
}

/*********** Digital Read function*********/
void POWER()
```

If it shows “Done Uploading” message. It means the code is compiled and loaded in to Arduino UNO board successfully and output data from sensors will be displayed on the LCD

OUTPUT :

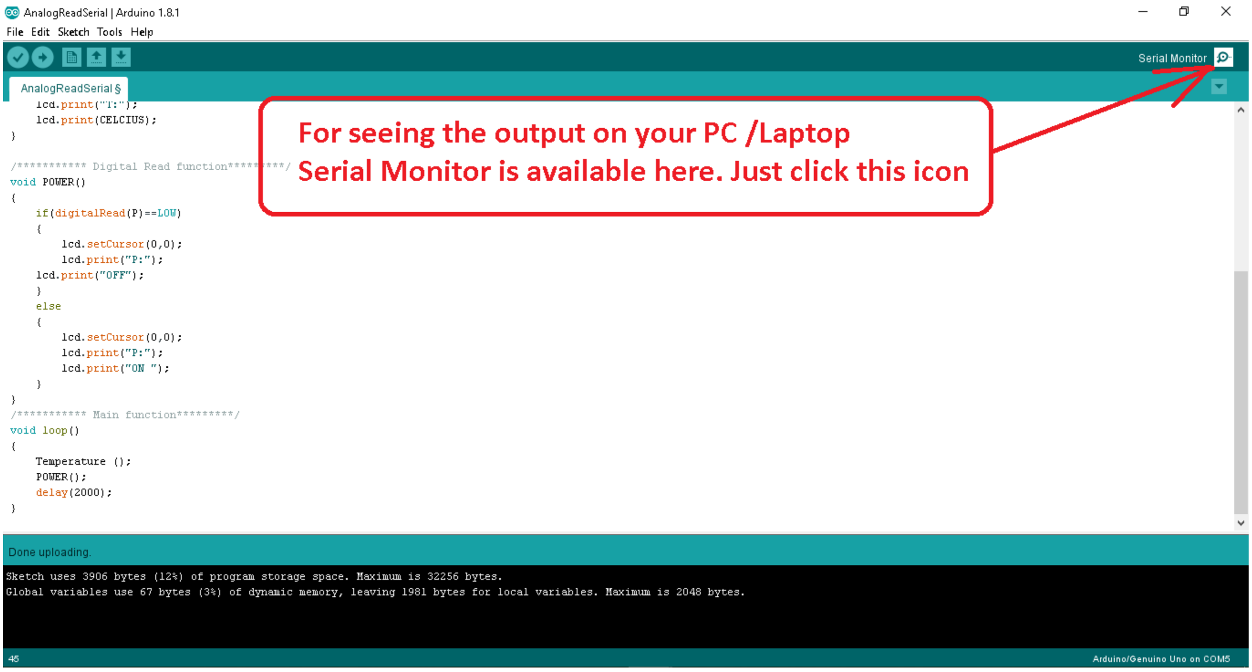
Sensor Board	LCD Display
P	OFF or ON
T	current temp in °C



You can see the same output on Serial monitor.

Follow the steps given here to configure it .

1. Click on the Serial monitor icon on the rightmost corner.


AnalogReadSerial | Arduino 1.8.1
File Edit Sketch Tools Help
Serial Monitor
AnalogReadSerial §
45
Arduino/Genuino Uno on COM5

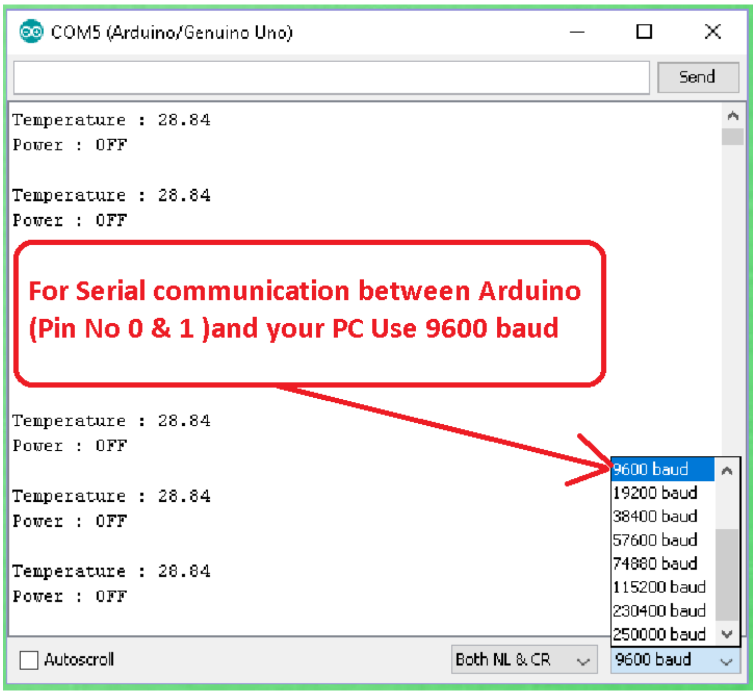
```
    lcd.print("T:");
    lcd.print(CELCIUS);
}

/*********** Digital Read function**********/
void POWER()
{
    if(digitalRead(P)==LOW)
    {
        lcd.setCursor(0,0);
        lcd.print("P:");
    lcd.print("OFF");
    }
    else
    {
        lcd.setCursor(0,0);
        lcd.print("P:");
        lcd.print("ON ");
    }
}
/*********** Main function*********/
void loop()
{
    Temperature ();
    POWER();
    delay(2000);
}
```

For seeing the output on your PC /Laptop
Serial Monitor is available here. Just click this icon

Done uploading.

```
Sketch uses 3906 bytes (12%) of program storage space. Maximum is 32256 bytes.
Global variables use 67 bytes (3%) of dynamic memory, leaving 1981 bytes for local variables. Maximum is 2048 bytes.
```

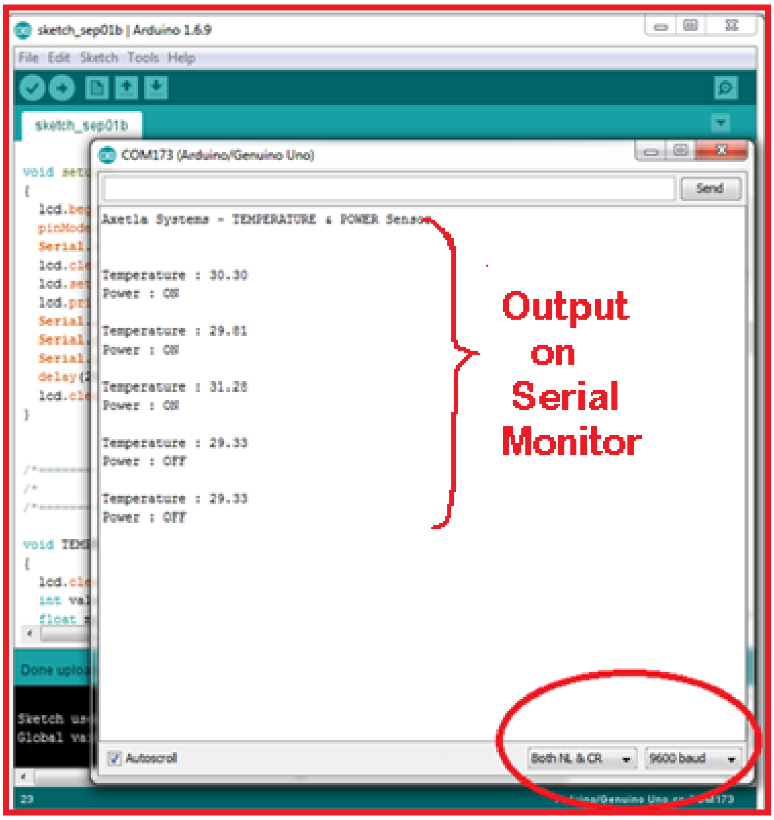
2. **Select the desired Baud rate and NL & CR options as shown below :**



```
Temperature : 28.84
Power : OFF

Temperature : 28.84
Power : OFF

Temperature : 28.84
Power : OFF

Temperature : 28.84
Power : OFF

Temperature : 28.84
Power : OFF
```

For Serial communication between Arduino (Pin No 0 & 1)and your PC Use 9600 baud

□ Autoscroll   Both NL & CR   9600 baud


sketch_sep01b | Arduino 1.6.9
File Edit Sketch Tools Help
sketch_sep01b
COM173 (Arduino/Genuino Uno)
Send

```
Axetla Systems - TEMPERATURE & POWER Sensor


Temperature : 30.30
Power : ON

Temperature : 29.81
Power : ON

Temperature : 31.28
Power : ON

Temperature : 29.33
Power : OFF

Temperature : 29.33
Power : OFF
```

Output on Serial Monitor

☑ Autoscroll   Both NL & CR   9600 baud

If you get any issues with the uploading there might be a problem in the selection of COM port or with the drivers. Refer to the Ardunio environment set up manual to make sure that there are no issues.

Description of Arduino Code.

Step1: Create a new sketch [program]

Step2: Include the header file “LiquidCrystal.h” at the start of the program, to use the functions related to LCD

```
#include<LiquidCrystal.h>
```

Step3: Make the global declaration of the LCD pins with the microcontroller pins of the Arduino board, i.e., RS, E, D4, D5, D6, D7, with pins 8, 9, 10, 11, 12, 13 using “LiquidCrystal”.

```
LiquidCrystal lcd(8,9,10,11,12,13);
```

Step4: Initialize a variable as integer with the pin number to which the digital sensor is connected.

```
int P=6;          //POWER SWITCH
```

Step5: Initialize a variable as integer with the Analog pin number to which the analog sensor is connected.

```
int T= A2;        // A2 has already been defined as Analog Pin 2 in arduino lib
```

Step6: Write a “setup” function and initialize the LCD as 16 columns with 2 rows with the help of “lcd.begin” function.

```
void setup()
{
    lcd.begin(16,2);
    pinMode(P, INPUT);

}
```

/*********** Analog Read function*********/

Step7: Capture Temperature and display

7.1 Start a void “TEMPERATURE” function and with the help of “analogRead” function to read the value from the Analog variable and store it to an integer variable. This gets the Temperature from the sensor and prints the actual value by converting analog voltage to the temp. Refer to data sheet for temp in manual

7.2 Divide the obtained value with the resolution of the ADC (i.e., 1023.0) and multiply the result with reference milli volts. (5V = 5000mV) and save it to a float variable.

7.3 Divide the above result by 10 because there will be 1°C change for every 10mV of output and save it to another float variable.

7.4 Set the cursor position by column and row numbers using “lcd.setCursor” function.

7.5 Display the value on the LCD using “lcd.print” function.

```
void Temperature()

{           int value_temp = analogRead(T);   // reading from A2 pin
            float millivolts_temp= ((value_temp/1023.0)*5000);
            float CELSIUS = millivolts_temp/10;
            lcd.setCursor(0,1);  // column, row
            lcd.print("T:");
            lcd.print(CELSIUS);
}
```

```
/*********** Digital Read function*********/
```

Step8: Capture Power Status and display

8.1 Start a void “POWER” function and check the concerned digital pin/variable is LOW.

8.2 If it is LOW, display the status on the LCD with the help of “lcd.print” function and “lcd.setCursor” functions respectively.

8.3 Else, display its status.

```
void POWER()

{

            if(digitalRead(P)==LOW)
            {
                        lcd.setCursor(0,0); // column, row
                        lcd.print("P:");
                        lcd.print("OFF");
            }
            else
            {
                        lcd.setCursor(0,0); // column, row
                        lcd.print("P:");
                        lcd.print("ON ");
            }
}
```

/************ Main function**********/

Step9:

9.1 Start a function with the name “loop”. This is the Main loop/function of the whole code/program. As the name indicates, this function is called in a continuous loop by Arduino library.

9.2 Call the above Analog and Digital functions simultaneously with sufficient delays, using “delay” function in milliseconds.

```
void loop()

{
        Temperature ();
        POWER();
        delay(2000);
}
```

Step10:

After uploading, data from two sensors will be displayed on the LCD display.

If you get any issues with the uploading there might be a problem in the selection of COM port or with the drivers. Refer to the Ardunio environment set up manual to make sure that there are no issues.

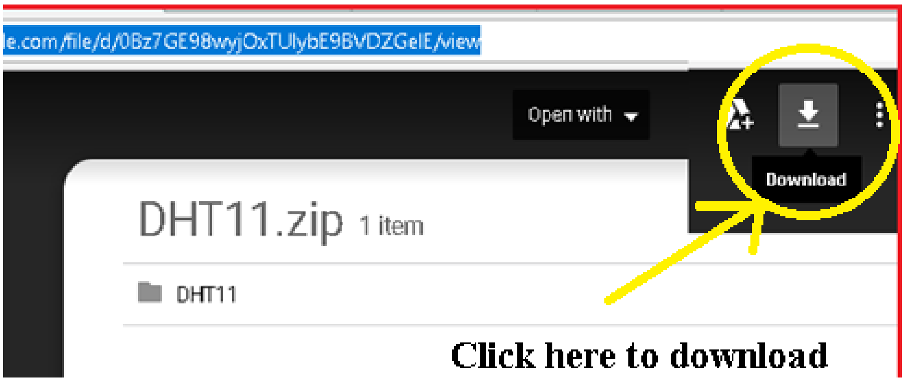
****

All Sensors Practical :

Steps to add **DHT11 Library -**

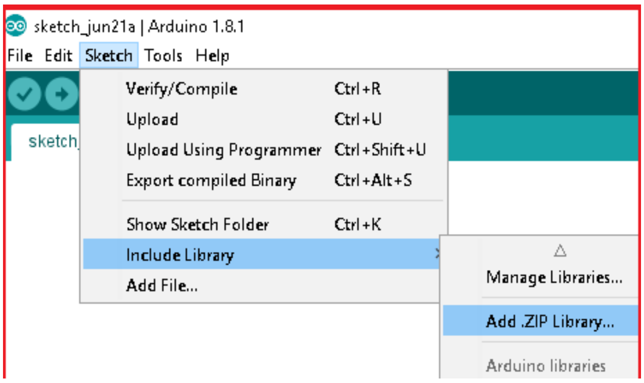
For Interfacing the"**DHT11**(**Humidity sensor**)" Click on the following link & download it.

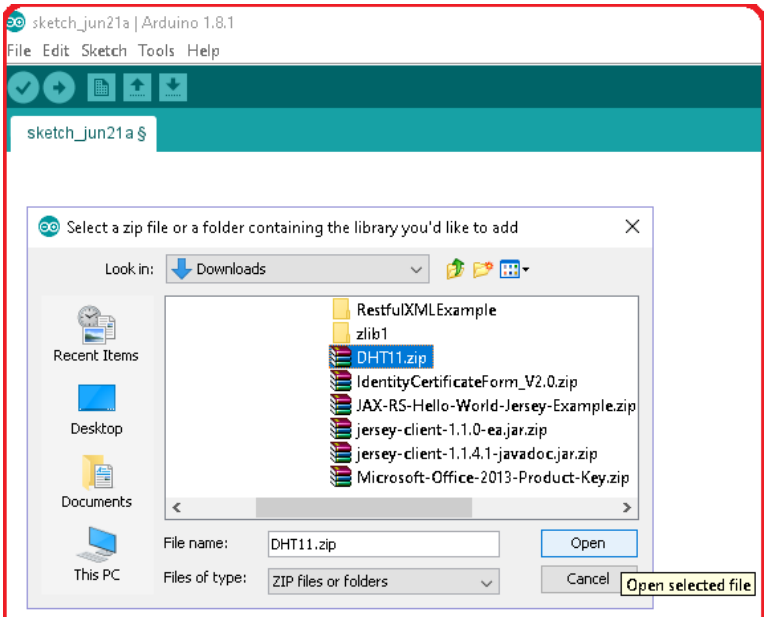
https://drive.google.com/file/d/0Bz7GE98wyjOxTUlybE9BVDZGelE/view



Click here to download

To add DHT11 library to Arduino do the following :


sketch_jun21a | Arduino 1.8.1
File Edit Sketch Tools Help
sketch
Verify/Compile Ctrl+R
Upload Ctrl+U
Upload Using Programmer Ctrl+Shift+U
Export compiled Binary Ctrl+Alt+S
Show Sketch Folder Ctrl+K
Include Library
Add File...
Manage Libraries...
Add .ZIP Library...
Arduino libraries


sketch_jun21a | Arduino 1.8.1
File Edit Sketch Tools Help
sketch_jun21a §

Select a zip file or a folder containing the library you'd like to add

Look in: Downloads

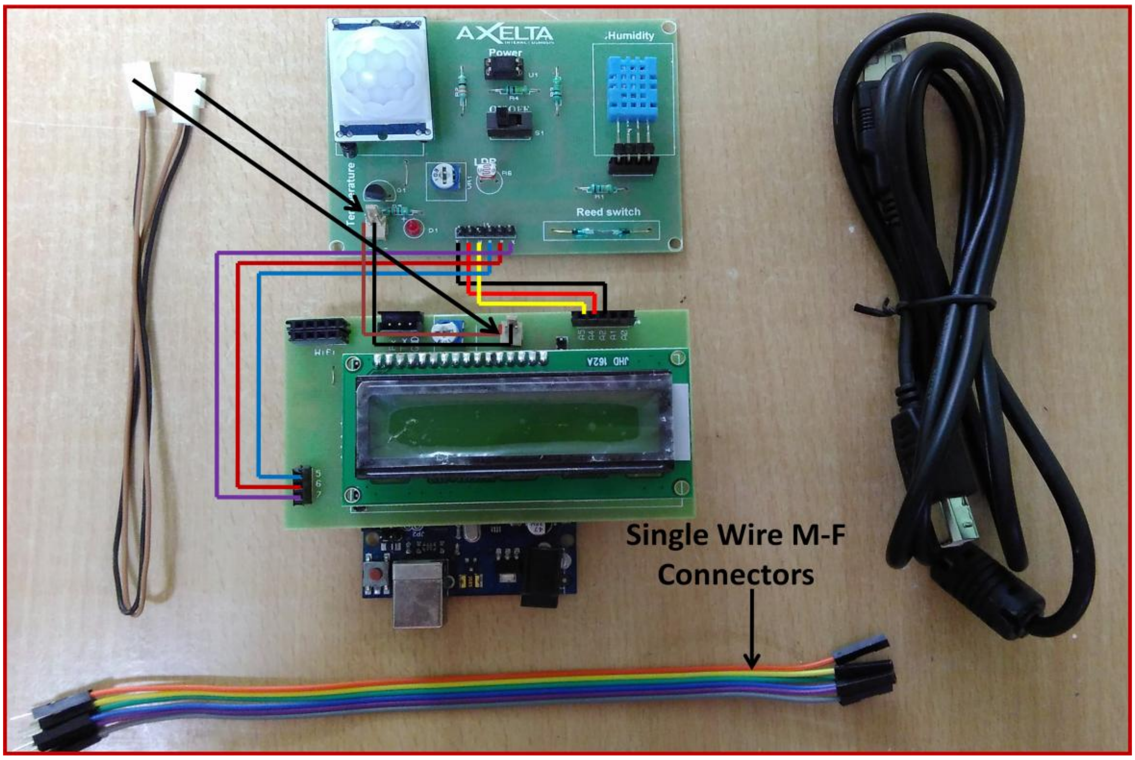
Recent Items
Desktop
Documents
This PC
RestfulXMLExample
zlib1
DHT11.zip
IdentityCertificateForm_V2.0.zip
JAX-RS-Hello-World-Jersey-Example.zip
jersey-client-1.1.0-ea.jar.zip
jersey-client-1.1.4.1-javadoc.jar.zip
Microsoft-Office-2013-Product-Key.zip
Open
Cancel
Open selected file

File name: DHT11.zip

Files of type: ZIP files or folders

Arduino UNO Board Connections for All sensors:

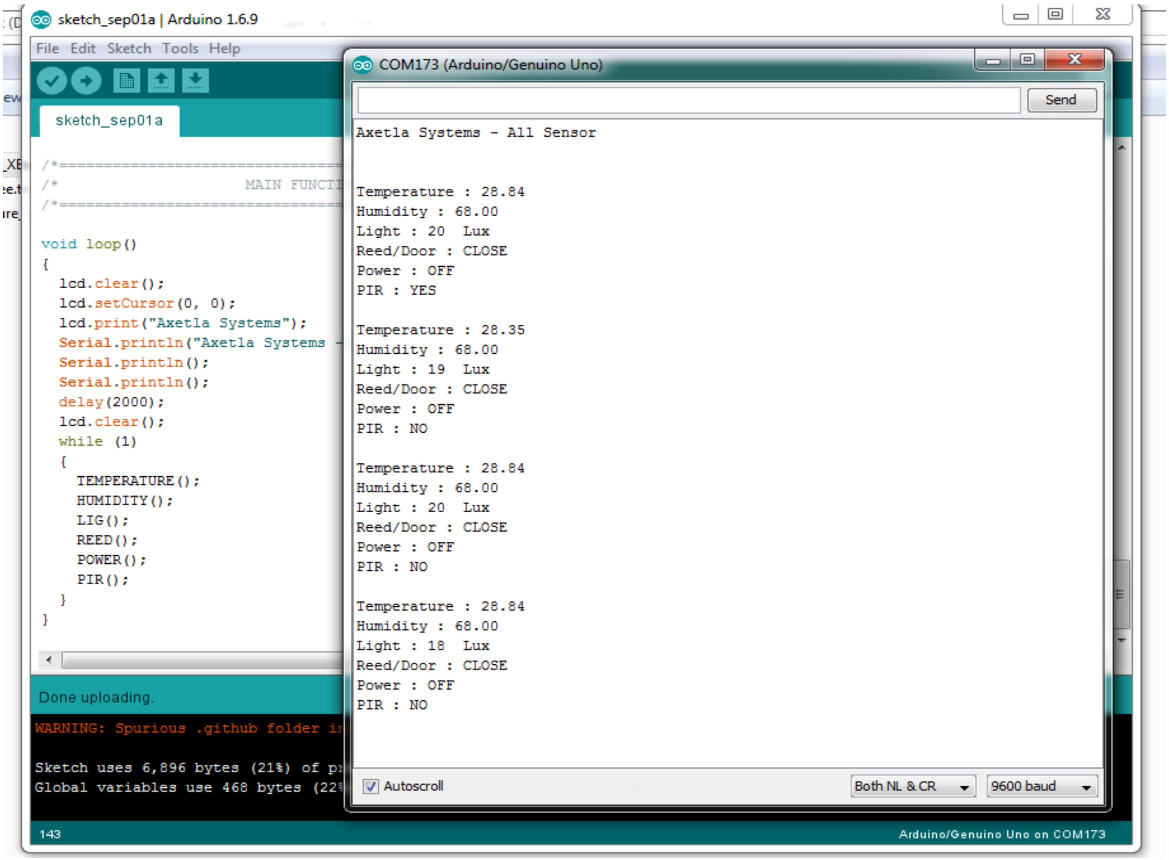
Sensor Board	LCD Shield on ARDUINO
T (Temp sensor)	**A2** (Analog i/p pin)
L (Light sensor (LDR))	**A4** (Analog i/p pin)
H (Humidity sensor)	**A5** (Analog i/p pin)
R (Reed switch)	**5** (Digital input pin)
P (Power switch)	**6** (Digital input pin)
X (PIR sensor)	**7** (Digital input pin)



Click https://drive.google.com/open?id=0Bz7GE98wyjOxWGlBUG1yTzdaQ00 for All sensors.

Paste the **"All Sensor code.text".** Compile & Upload it. You can see the output data from all the sensors on LCD Display /Serial Monitor.

All sensor output on Serial Monitor is shown below :


sketch_sep01a | Arduino 1.6.9
File Edit Sketch Tools Help
sketch_sep01a
COM173 (Arduino/Genuino Uno)
Send
☑ Autoscroll
Both NL & CR
9600 baud
143
Arduino/Genuino Uno on COM173

```
/*==================================
/*                         MAIN FUNCTI
/*==================================

void loop()
{
  lcd.clear();
  lcd.setCursor(0, 0);
  lcd.print("Axetla Systems");
  Serial.println("Axetla Systems -
  Serial.println();
  Serial.println();
  delay(2000);
  lcd.clear();
  while (1)
  {
    TEMPERATURE();
    HUMIDITY();
    LIG();
    REED();
    POWER();
    PIR();
  }
}
```

Done uploading.

```
WARNING: Spurious .github folder in

Sketch uses 6,896 bytes (21%) of pr
Global variables use 468 bytes (22%
```

```
Axetla Systems - All Sensor


Temperature : 28.84
Humidity : 68.00
Light : 20  Lux
Reed/Door : CLOSE
Power : OFF
PIR : YES

Temperature : 28.35
Humidity : 68.00
Light : 19  Lux
Reed/Door : CLOSE
Power : OFF
PIR : NO

Temperature : 28.84
Humidity : 68.00
Light : 20  Lux
Reed/Door : CLOSE
Power : OFF
PIR : NO

Temperature : 28.84
Humidity : 68.00
Light : 18  Lux
Reed/Door : CLOSE
Power : OFF
PIR : NO
```