**Arduino Lesson (  )**

**Use of OLED (Organic Light-Emitting Diode) Display (Advanced), display of sensor readings & mathematical functions**

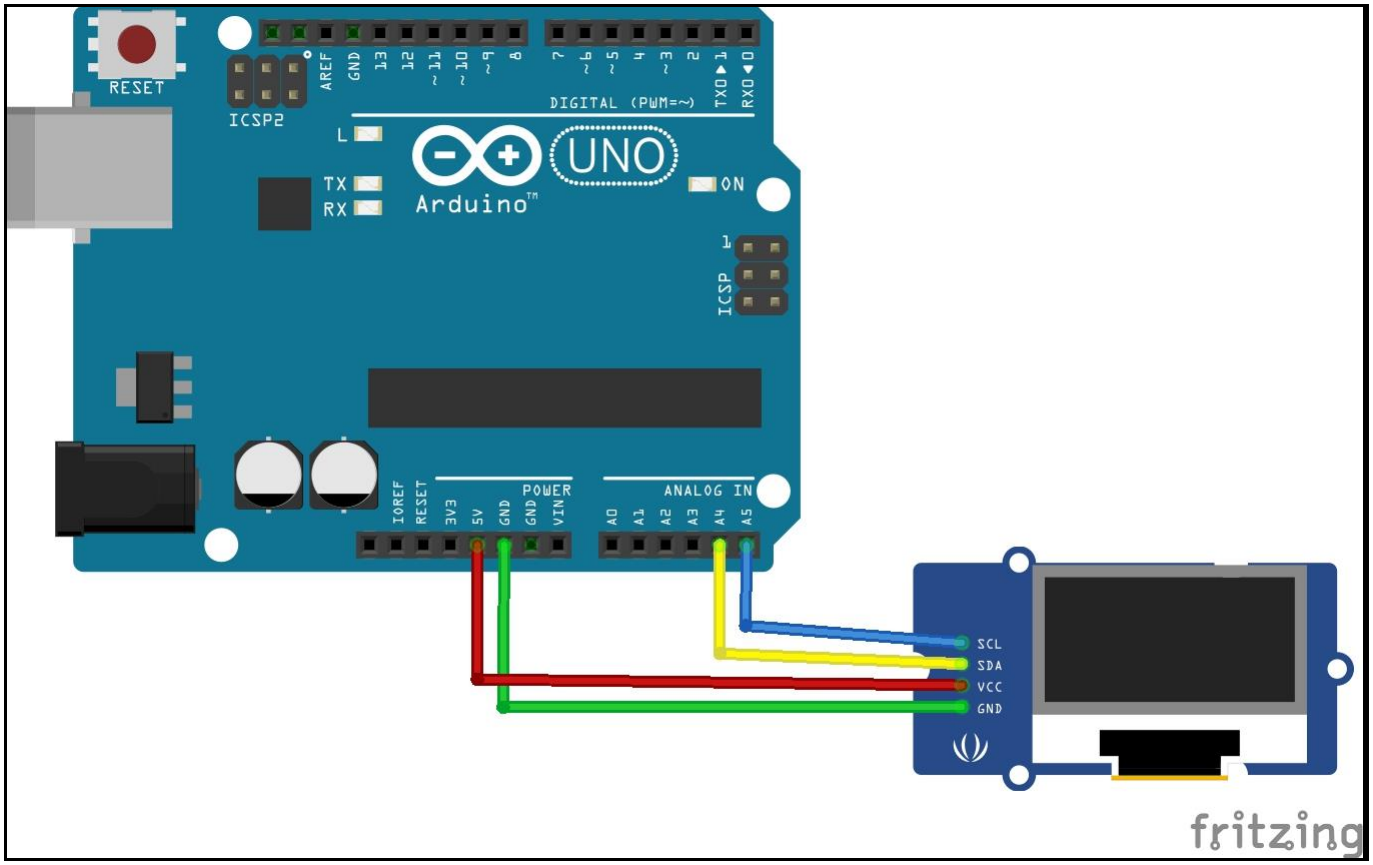
**Name: _____ (  ) Class: _____ Date: _____**

**Objectives**: At the end of this lesson, you would be able to

1. Connect an OLED to the Arduino microcontroller
2. Write sketches to display messages on the OLED

**Apparatus:**

- 01 Arduino UNO microcontroller
- 01 USB cable
- 01 OLED display (SSD1306 0.96 inch, 128x64 pixels, I2C communication protocol)
- 04 female-male jumper wires
- (optional) 01 breadboard (if used with male-male jumper wires )




*Above diagram created with fritzing.org software*

**References:**

- https://randomnerdtutorials.com/guide-for-oled-display-with-arduino/
- https://www.instructables.com/id/OLED-I2C-DISPLAY-WITH-ARDUINO-Tutorial/
- https://lastminuteengineers.com/ds18b20-arduino-tutorial/

**Assignment**: Set up an OLED to display messages

1. Connect the OLED to the Arduino microcontroller using 4 female to male jumper wires in the following way:

| | | | | |
|---|---|---|---|---|
| **OLED** | GND | → | GND | **Arduino** |
| | VCC | → | 5V | |
| | SDA (data) | → | A4 (Analog pin 4) | |
| | SCL (clock) | → | A5 (Analog pin 5) | |

2. Connect the Arduino microcontroller to the laptop and check for connectivity:

Select **Tools** → **Board**: "Arduino/Genuino Uno" and **Port**: "COM#"

3. [Optional step] Select **File** → **New**: Copy & paste, verify and upload **Sketch A** to find the I2C address for the display of the OLED.

4. Activate the **serial monitor** (top right corner) to locate the I2C address in the line "Founder: 60 (0x3c)"

## [Optional] Sketch A: Arduino I2C Address Locator

```
/* I2C Address Finder
*/
#include <Wire.h>

void setup() {
  Serial.begin (115200);
while (!Serial)
    {
    }

  Serial.println ();
  Serial.println ("I2C scanner. Scanning ...");
  byte count = 0;
  pinMode(13,OUTPUT);
  digitalWrite(13,HIGH);
  Wire.begin();
  for (byte i = 1; i < 120; i++)
  {
    Wire.beginTransmission (i);
    if (Wire.endTransmission () == 0)
      {
      Serial.print ("Found address: ");
      Serial.print (i, DEC);
      Serial.print (" (0x");
      Serial.print (i, HEX);
      Serial.println (")");
      count++;
      delay (1);
      }
  }
  Serial.println ("Done.");
  Serial.print ("Found ");
  Serial.print (count, DEC);
  Serial.println (" device(s).");
}

void loop() {}
```

5. Install **two libraries** into Arduino IDE (Integrated Development Environment):

*Sketch → Include Library → Manage Libraries*

- Type "**SSD1306**" in the search box and install the **SSD1306 library** from Adafruit.
- Search for "**GFX**" in the search box and install the **adafruit_GFX.h** libraries.

6. [Optional step] Select **File → Examples → Adafruit SSD1306 → ssd1306_128x64_i2c**

This will load **Sketch B**. Verify and upload this sketch to test the OLED display.

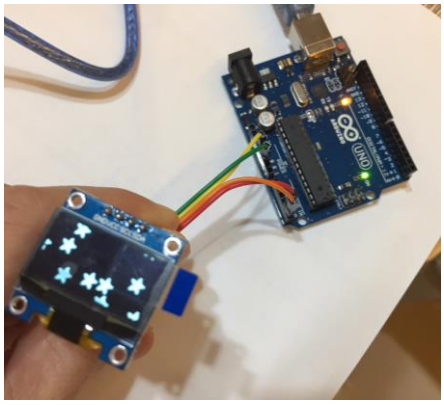
7. If you see a series of different animations on the OLED screen, it means you have successfully set up the OLED.

## [Optional] Sketch B: Testing the OLED Display (ssd1306_128x64_i2c) (extract shown below)

```
/*********
  Complete project details at https://randomnerdtutorials.com
  This is an example for our Monochrome OLEDs based on SSD1306 drivers.
*********/

#include <SPI.h>
#include <Wire.h>
#include <Adafruit_GFX.h>
#include <Adafruit_SSD1306.h>

#define SCREEN_WIDTH 128          // OLED display width, in pixels
#define SCREEN_HEIGHT 64          // OLED display height, in pixels

// Declaration for an SSD1306 display connected to I2C (SDA, SCL pins)
#define OLED_RESET -1     // Reset pin # (or -1 if sharing Arduino reset pin)
Adafruit_SSD1306 display(SCREEN_WIDTH, SCREEN_HEIGHT, &Wire, OLED_RESET);

#define NUMFLAKES 10      // Number of snowflakes in the animation example
```

***… continue …***

*From: https://raw.githubusercontent.com/RuiSantosdotme/Random-Nerd-Tutorials/master/Projects/OLED/oled_adafruit_demo.ino*



***Trouble-shooting** - If your OLED display is not showing anything:*

- *Check that the OLED display is properly wired to the Arduino*
- *Double-check the OLED display I2C address using Sketch A*

8. Select **File → New**: Copy & paste, verify and upload **Sketch C** to display the “Hello world! ” message.

**Sketch C: Displaying “Hello world!” message**

```
/* Starting with Arduino OLED coding
 *  for " arduino oled i2c tutorial : 0.96" 128 X 32 for beginners"
 *  subscribe for more arduino Tuorials and Projects
https://www.youtube.com/channel/UCM6rbuieQBBLFsxszWA85AQ?sub_confirmation=1
 */

#include <SPI.h>
#include <Wire.h>
#include <Adafruit_GFX.h>
#include <Adafruit_SSD1306.h>

#define OLED_RESET 4
Adafruit_SSD1306 display(OLED_RESET);

void setup()
{
  display.begin(SSD1306_SWITCHCAPVCC, 0x3C);
  display.clearDisplay();
}

void loop()
{
  display.clearDisplay();
  display.setTextSize(1);
  display.setTextColor(WHITE);
  display.setCursor(0,0);
  display.println("Hello world!");      // can replace with text or reading from sensor
                                       // or value from math function
  display.display();
}
```

**Reference:**

- *Mr Ang: http://johnlittle.pbworks.com/w/page/130434072/basics*

## Tips for writing text using the installed libraries

Some functions to use the OLED display library to write text or draw simple graphics:

- display.clearDisplay() – all pixels are off
- display.drawPixel(x,y, color) – plot a pixel in the x,y coordinates
- display.setTextSize(n) – set the font size, supports sizes from 1 to 8
- display.setCursor(x,y) – set the coordinates to start writing text
- display.print("message") – print the characters at location x,y
- display.display() – call this method for the changes to make effect

## 1. OLED Display features

Explore how to show other items on OLED display at
*https://randomnerdtutorials.com/guide-for-oled-display-with-arduino/*

- Scrolling text
- Using other fonts
- Draw shapes: pixel, line, rectangle, circle, triangle, invert colours,

*Sketch at: https://raw.githubusercontent.com/RuiSantosdotme/Random-Nerd-Tutorials/master/Projects/OLED/oled_draw_shapes.ino*

- Display bitmap image

## 2. Displaying Sensor Readings

### 2.1 DS18B20 Temperature Sensor Readings

```
float T = sensors.getTempCByIndex(0);  // let T be temperature in degC from sensor
                                       // floating-point number, with a decimal point
display.clearDisplay();
display.setTextSize(1);
display.setTextColor(WHITE);
display.setCursor(0,0);
display.println("Temperature: ");   // display temperature in deg C
display.print(T);
display.print(" ");
display.cp437(true);          // code page 437
display.write(167);           // character 167 is degree
display.println("C");
display.display();            // for the changes to make effect
```

**Note**: Other special symbols (characters) available are shown on https://www.ascii-codes.com/

**Rate of temperature change**

- Obtain T1 and T2 (at time interval of t later)
- Rate of change of temperature = dT/dt = (T2 – T1)/ t

## 2.2 LDR Circuit (light intensity) Readings

```
float L = analogRead(A0);          // let T be temperature in degC from sensor
                                   // floating-point number, with a decimal point
```

## 3. Displaying Mathematical functions

```
/* Arduino OLED coding
 *  Ang JL - display Mathematical functions for X, Y and Z
 */

#include <SPI.h>
#include <Wire.h>
#include <Adafruit_GFX.h>
#include <Adafruit_SSD1306.h>

#define OLED_RESET 4
Adafruit_SSD1306 display(OLED_RESET);

float A = 5.0;
float B = 8.0;
float X = 4.5 * log10 (A);                  // logarithm of A to base 10
float Y = 8.4 * pow(B, 1.5 );        // B to the power of 1.5
float Z = X + Y;

void setup()
{
  display.begin(SSD1306_SWITCHCAPVCC, 0x3C);
  display.clearDisplay();
}

void loop()
{
  display.clearDisplay();
  display.setTextSize(1);
  display.setTextColor(WHITE);
  display.setCursor(0,0);

  display.print("X = 4.5log10(");   // display text
  display.print(A);                 // display value of A
  display.println(")");             // display text followed by new line
  display.print("= ");              // display text
  display.println(X);               // display final value of X

  display.print("Y = 8.4(");        // display text
  display.print(B);                 // display value of A
  display.println("^1.5) ");        // display "B to the power of 1.5" followed by new line
  display.print("= ");              // display text on new line
  display.println(Y);               // display final value of Y
  display.display();                // display all
  delay (3000);                     // for 3 seconds

  display.clearDisplay();           // clear display and start again
  display.setTextSize(1);
  display.setTextColor(WHITE);
  display.setCursor(0,0);
  display.print("X + Y = ");        // display text (2nd page)
  display.print(Z);                 // display final value of Z
  display.display();                // display all
  delay (3000);                     // for 3 seconds
}
```