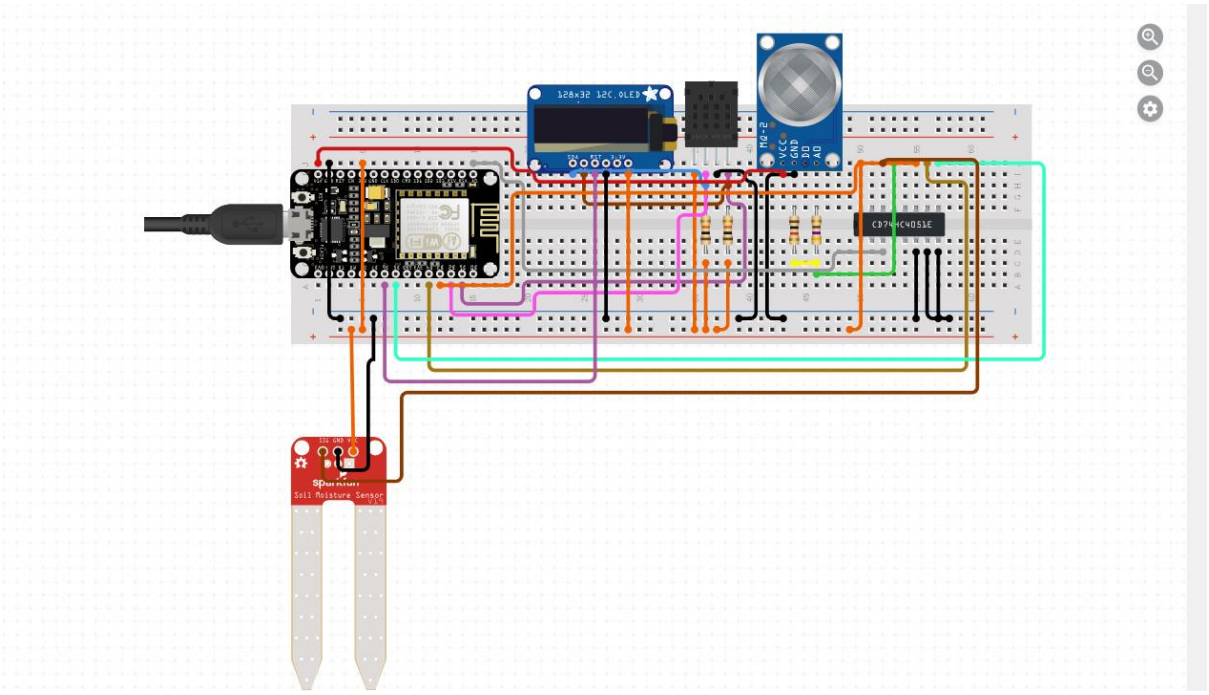
IBMNM PHASE-3

CIRCUIT DIAGRAM



CODE

```
// Include Libraries
#include "Arduino.h"
#include "Wire.h"
#include "SPI.h"
#include "Adafruit_SSD1306.h"
#include "Adafruit_GFX.h"
#include "SoilMoisture.h"


// Pin Definitions
#define HC744051_PIN_S0        0
#define HC744051_PIN_S1        2
#define HC744051_PIN_S2        14
#define HC744051_PIN_A         A0
#define MQ2_3V3_PIN_AOUT    0
#define OLED128X32_PIN_RST   12
#define SOILMOISTURE_3V3_PIN_SIG     1



// Global variables and defines

// object initialization
#define SSD1306_LCDHEIGHT 32
Adafruit_SSD1306 oLed128x32(OLED128X32_PIN_RST);
SoilMoisture soilMoisture_3v3(SOILMOISTURE_3V3_PIN_SIG);


// define vars for testing menu
```

```
const int timeout = 10000;       //define timeout of 10 sec

char menuOption = 0;

long time0;


// Setup the essentials for your circuit to work. It runs first every time your circuit is powered with electricity.

void setup()

{

  // Setup Serial which is useful for debugging

  // Use the Serial Monitor to view printed messages

  Serial.begin(9600);

  while (!Serial) ; // wait for serial port to connect. Needed for native USB

  Serial.println("start");


  oLed128x32.begin(SSD1306_SWITCHCAPVCC, 0x3C);  // initialize with the I2C addr 0x3C (for the 128x32)

  oLed128x32.clearDisplay(); // Clear the buffer.

  oLed128x32.display();

  menuOption = menu();


}


// Main logic of your circuit. It defines the interaction between the components you selected. After setup, it runs over and over again, in an eternal loop.

void loop()

{



  if(menuOption == '1')

  {

  // Disclaimer: The 74HC4051 - Analog Multiplexer and Demultiplexer is in testing and/or doesn't have code, therefore it may be buggy. Please be kind and report any bugs you may find.
```

```
  }
  else if(menuOption == '2')
  {
  // Disclaimer: The AM2320 Digital Temperature and Humidity Sensor is in testing and/or doesn't have code, therefore it may be buggy. Please be kind and report any bugs you may find.
  }
  else if(menuOption == '3')
  {
  // Disclaimer: The Methane, Butane, LPG and Smoke Gas Sensor - MQ-2 is in testing and/or doesn't have code, therefore it may be buggy. Please be kind and report any bugs you may find.
  }
  else if(menuOption == '4') {
  // Monochrome 128x32 I2C OLED graphic display - Test Code
  oLed128x32.setTextSize(1);
  oLed128x32.setTextColor(WHITE);
  oLed128x32.setCursor(0, 10);
  oLed128x32.clearDisplay();
  oLed128x32.println("Circuito.io Rocks!");
  oLed128x32.display();
  delay(1);
  oLed128x32.startscrollright(0x00, 0x0F);
  delay(2000);
  oLed128x32.stopscroll();
  delay(1000);
  oLed128x32.startscrollleft(0x00, 0x0F);
  delay(2000);
  oLed128x32.stopscroll();
  }
  else if(menuOption == '5') {
  // Soil Moisture Sensor - Test Code
  int soilMoisture_3v3Val = soilMoisture_3v3.read();
  Serial.print(F("Val: ")); Serial.println(soilMoisture_3v3Val);
```

```
    }

    if (millis() - time0 > timeout)
    {
        menuOption = menu();
    }

}



// Menu function for selecting the components to be tested
// Follow serial monitor for instrcutions
char menu()
{

    Serial.println(F("\nWhich component would you like to test?"));
    Serial.println(F("(1) 74HC4051 - Analog Multiplexer and Demultiplexer"));
    Serial.println(F("(2) AM2320 Digital Temperature and Humidity Sensor"));
    Serial.println(F("(3) Methane, Butane, LPG and Smoke Gas Sensor - MQ-2"));
    Serial.println(F("(4) Monochrome 128x32 I2C OLED graphic display"));
    Serial.println(F("(5) Soil Moisture Sensor"));
    Serial.println(F("(menu) send anything else or press on board reset button\n"));
    while (!Serial.available());

    // Read data from serial monitor if received
    while (Serial.available())
    {
        char c = Serial.read();
        if (isAlphaNumeric(c))
```

```
      {


        if(c == '1')

                        Serial.println(F("Now Testing 74HC4051 - Analog Multiplexer and
Demultiplexer - note that this component doesn't have a test code"));

                else if(c == '2')

                        Serial.println(F("Now Testing AM2320 Digital Temperature and Humidity
Sensor - note that this component doesn't have a test code"));

                else if(c == '3')

                        Serial.println(F("Now Testing Methane, Butane, LPG and Smoke Gas Sensor -
MQ-2 - note that this component doesn't have a test code"));

                else if(c == '4')

                        Serial.println(F("Now Testing Monochrome 128x32 I2C OLED graphic
display"));

                else if(c == '5')

                        Serial.println(F("Now Testing Soil Moisture Sensor"));

        else

        {

           Serial.println(F("illegal input!"));

           return 0;

        }

        time0 = millis();

        return c;

      }

   }

}
```