**IOT\_PHASE3:** **PUBLIC TRANSPORT OPTIMIZATION**

**TEAM MEMBER:** MOHAMMED JASIR ALI S A (822721106305)

**Project Title:** Public Transport Optimization

**Project Description:** This project involves integrating IOT Sensors into public transportation vehicles to monitor ridership, track locations, and predict perfect arrival times. The goal is to provide real -time transit information to the public through a public platform, enhancing the efficiency and quality of public transportation services. This project includes defining Objectives, designing the IOT sensor system, developing the real-time transit information platform, and integrating them using IOT technology and python.

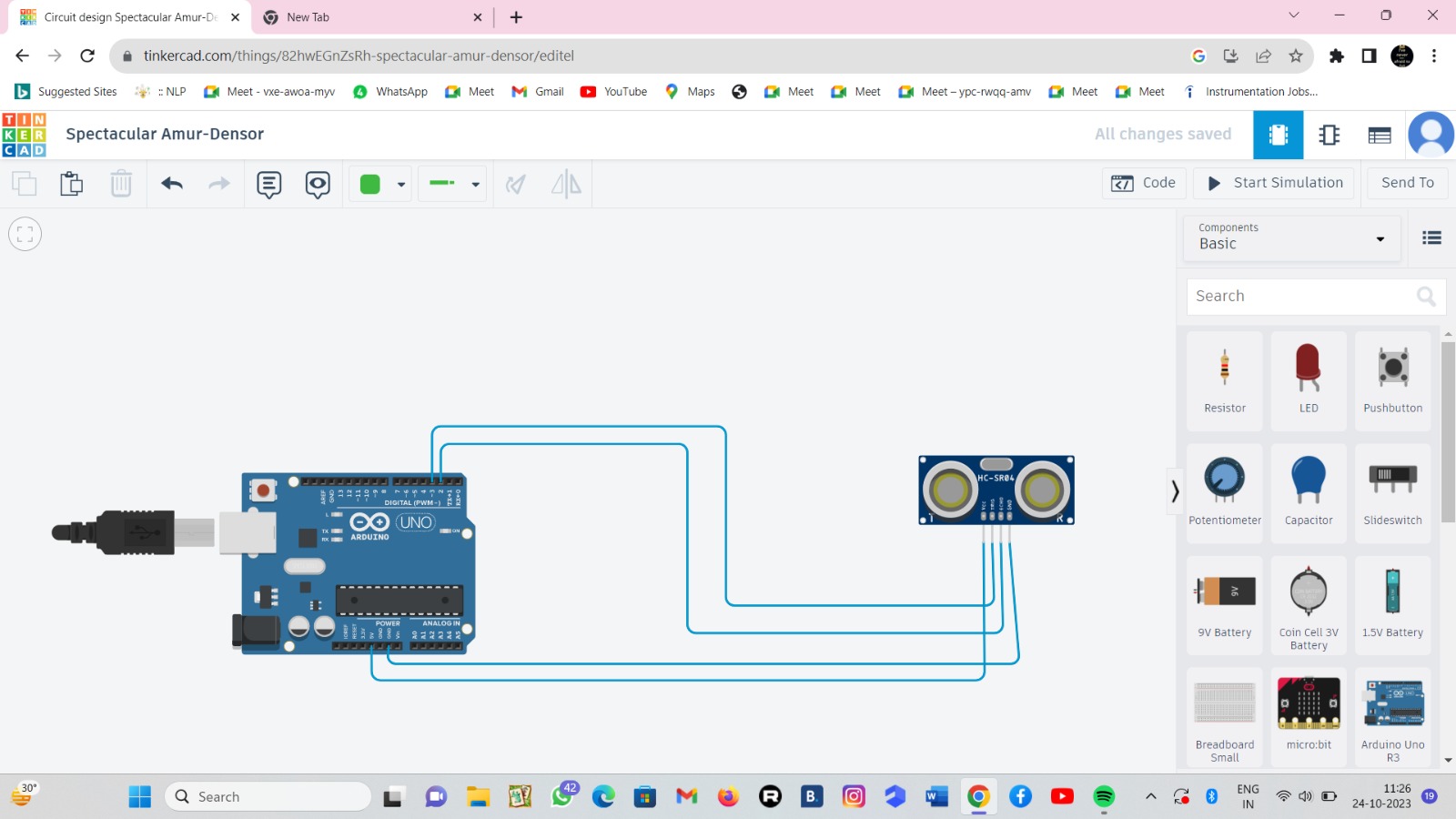
**SOFTWARE REQUIREMENT:**

* TinkerCad used for Simulation Process

**CONNECTION DETAILS:**

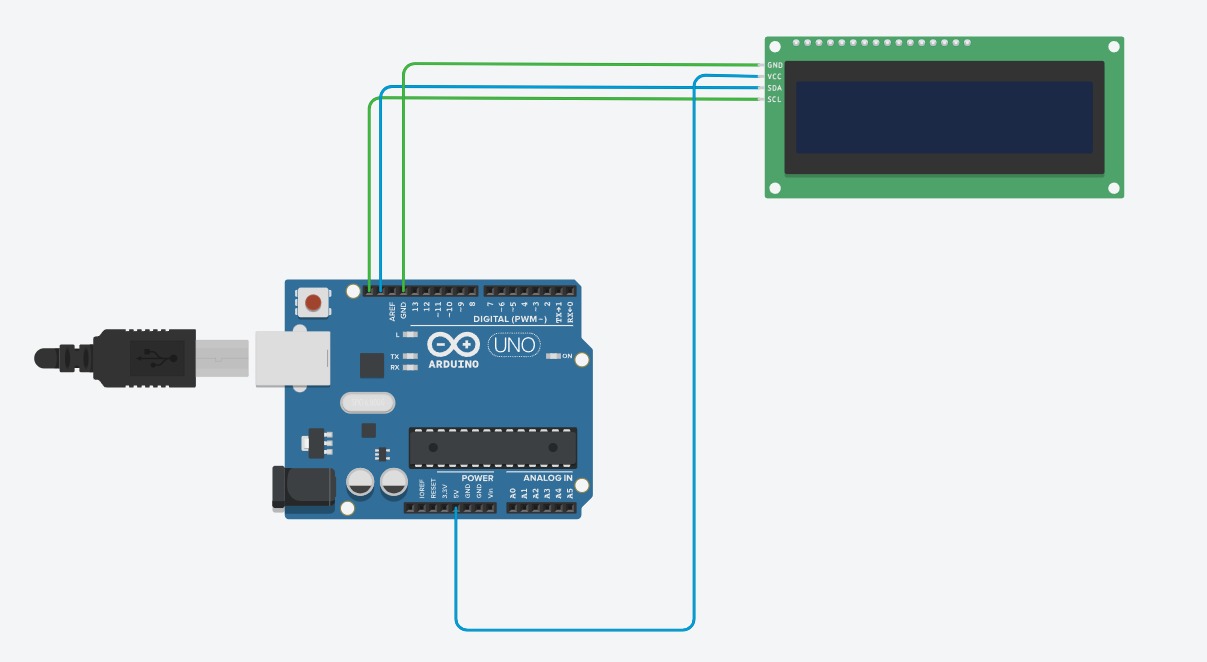
**BETWEEN ARDUINO UNO & ULTRASONIC SENSOR:**

|  |  |  |
| --- | --- | --- |
| **S.NO** | **ARDUINO UNO** | **ULTRASONIC SENSOR** |
| 1 | 5V | VCC |
| 2 | DIGITAL PIN 3 | TRIGGER |
| 3 | DIGITAL PIN 2 | ECHO PIN |
| 4 | GND | GND |

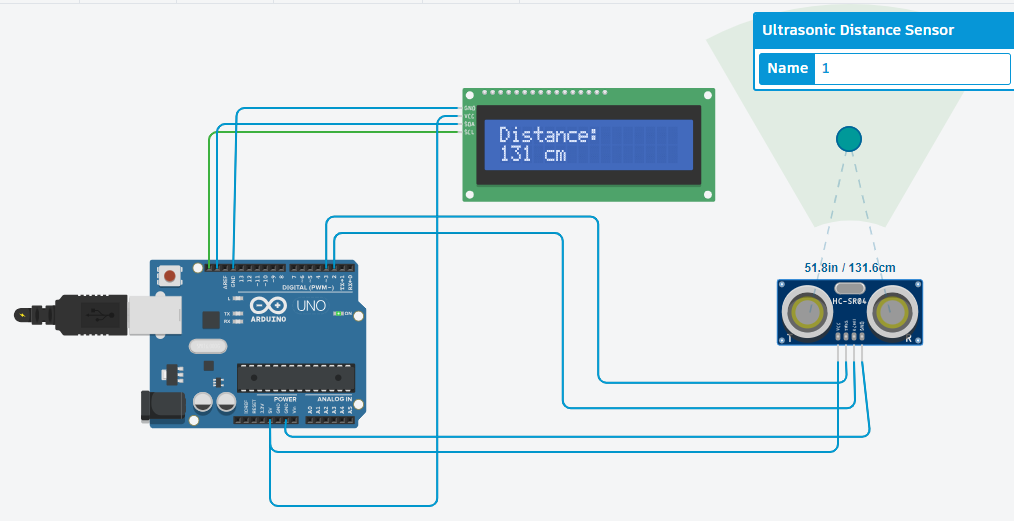
****

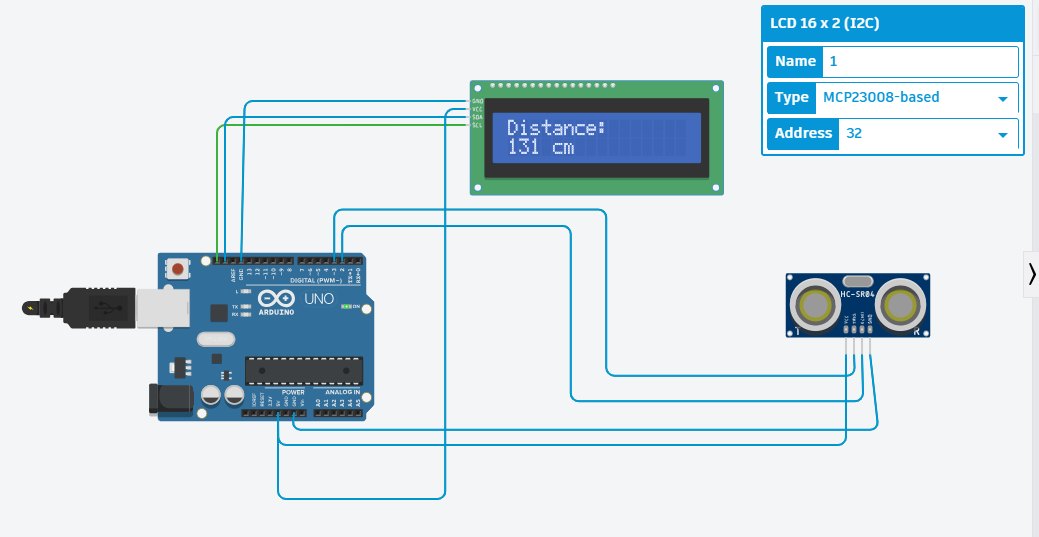
**BETWEEN ARDUINO UNO & LED:**

|  |  |  |
| --- | --- | --- |
| **S.NO** | **ARDUINO UNO** | **LCD** |
| 1 | 5V | VCC |
| 2 | FDA PIN | SBA PIN |
| 3 | SCL PIN | SCL PIN |
| 4 | GND | GND |

****

**AFTER SIMULATION PROCESS:**

****

****

**C++ CODE FOR SIMULATION PROCESS:**

#include <LiquidCrystal\_I2C.h>

LiquidCrystal\_I2C lcd(0x20, 16, 2); // Format -> (Address,Width,Height )

#define echoPin 2 // attach pin D2 Arduino to Echo pin of Sensor module

#define trigPin 3 // attach pin D3 Arduino to Trig pin of Sensor module

long duration; // Declare variable to store echo time duration

int distance; // Declare variable to store the result (distance)

void setup()

{

lcd.init(); // initialize the lcd

lcd.backlight(); // Turn on the Backlight

pinMode(trigPin,OUTPUT); // Sets the trigPin as an OUTPUT

pinMode(echoPin, INPUT); // Sets the echoPin as an INPUT

// Serial Communication is starting with 9600 of baudrate speed

Serial.begin(9600);

// The text to be printed in serial monitor Serial.println("Distance measurement using Arduino Uno");

delay(500);

}

void loop()

{

digitalWrite(trigPin, LOW);

delayMicroseconds(2);

digitalWrite(trigPin, HIGH);

delayMicroseconds(10);

digitalWrite(trigPin, LOW);

duration = pulseIn(echoPin, HIGH);

distance = duration \* 0.0344 / 2;

Serial.print("Distance: ");

Serial.print(distance);

Serial.println(" cm");

lcd.clear(); // Clear the display buffer

lcd.setCursor(0, 0); // Set cursor for "Distance:" (Column, Row)

lcd.print("Distance:"); // print "Distance:" at (0, 0)

lcd.setCursor(0,1); // Set cursor for output value (0, 1)

lcd.print(distance); // print Output in cm at (0, 1)

lcd.setCursor(4, 1); // move cursor to (4, 1)

lcd.print("cm"); // print "cm" at (4, 1)

delay(100);

}