LAPORAN PRAKTIKUM MINGGU KE-9
“I2C LCD”
INTERNET OF THINGS

Disusun oleh:

Widiareta Safitri

1941720081

TI-3H

D4 TEKNIK INFORMATIKA

TEKNOLOGI INFORMASI

POLITEKNIK NEGERI MALANG

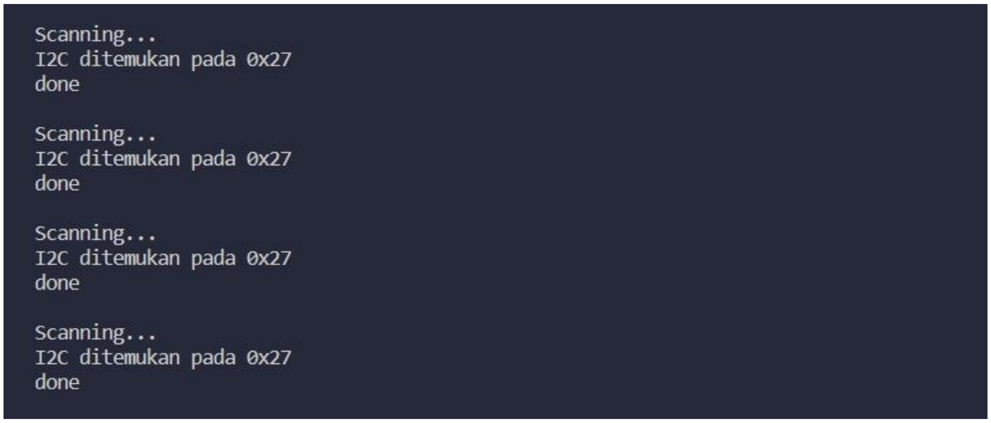
2022

PRACTICUM 1 - Searching for I2C addresses

a. Program Code

```
#include <Arduino.h>
#include <Wire.h>

void setup() {
  Wire.begin();
  Serial.begin(115200);
  Serial.println("\nI2C Scanner");
}

void loop() {
  byte error, address;
  int nDevices;
  Serial.println("Scanning...");
  nDevices = 0;
  for (address = 1; address < 127; address++)
  {
    Wire.beginTransmission(address);
    error = Wire.endTransmission();
    if (error == 0)
    {
      Serial.print("I2C ditemukan pada 0x");
      if (address < 16)
      {
        Serial.print("0");
      }
      Serial.println(address, HEX);
      nDevices++;
    }
    else if (error == 4)
    {
      Serial.print("Unknow error at address 0x");
      if (address < 16)
      {
        Serial.print("0");
      }
      Serial.println(address, HEX);
    }
  }
  if (nDevices == 0)
  {
    Serial.println("No I2C devices found\n");
  }
  else
  {
    Serial.println("done\n");
  }
  delay(3000);
}
```

b. Result



```
Scanning...
I2C ditemukan pada 0x27
done

Scanning...
I2C ditemukan pada 0x27
done

Scanning...
I2C ditemukan pada 0x27
done

Scanning...
I2C ditemukan pada 0x27
done
```

c. Explanation

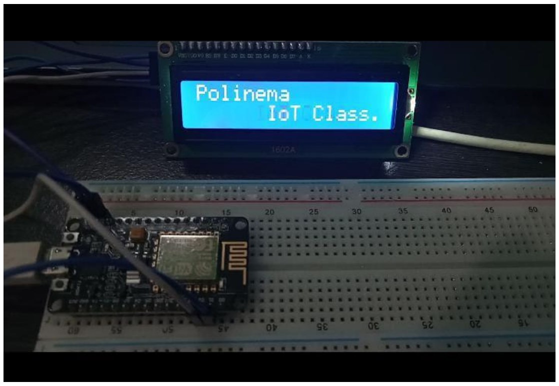
In practicum 1, the code written aims to detect the location of the I2C on the LCD where the I2C is found at 0x27. In this practicum, it does not display any writing on the LCD screen because it only finds the I2C location and there is no ldc.print command

PRACTICUM 2

a. Program Code

```
#include <Arduino.h>
#include <LiquidCrystal_I2C.h>

LiquidCrystal_I2C lcd(0x27, 16, 2);

void setup() {
  lcd.init(); // initialize the lcd
  lcd.backlight();
  lcd.clear();
  lcd.home();
}

void scrollText(int row, String message, int delayTime, int
lcdColumns)
{
  for (int i = 0; i < lcdColumns; i++)
  {
    message = " " + message;
  }
  message = message + " ";
  for (int pos = 0; pos < message.length(); pos++)
  {
    lcd.setCursor(0, row);
    lcd.print(message.substring(pos, pos + lcdColumns));
    delay(delayTime);
  }
}
 void loop() {
   lcd.home();
   lcd.print("Polinema");
   scrollText(1, "IoT Class.", 250, 16);
 }
```

b. Result



```
Polinema
     IoT Class.
```

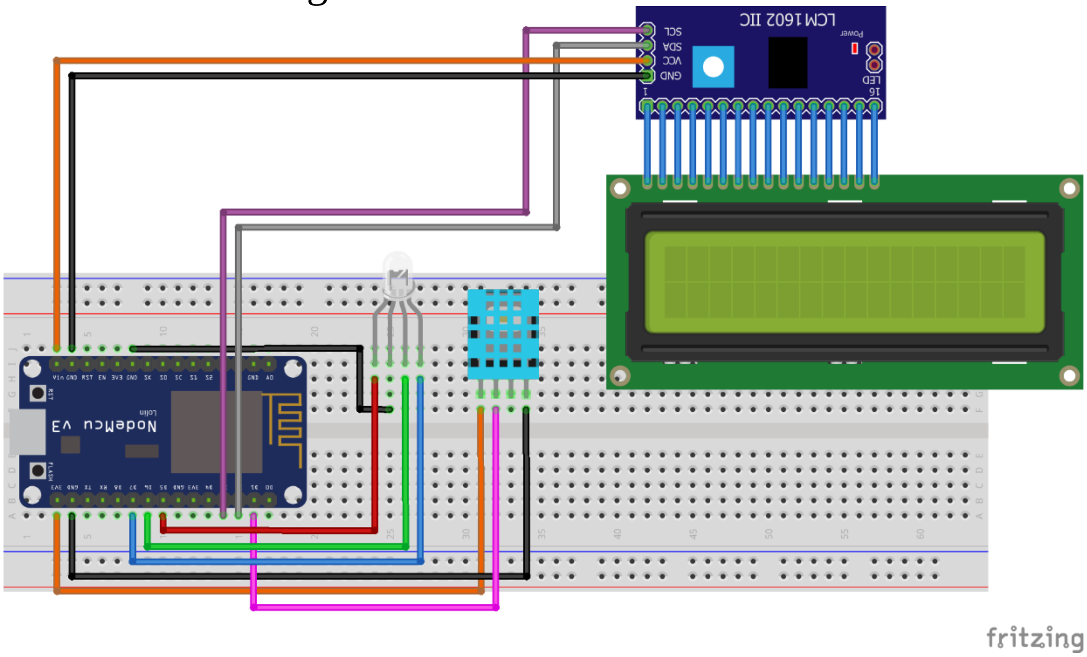
Video : [Link]

c. Explanation

In practicum 2, the running text will be displayed on the LCD screen. The library used is LiquidCrystal_I2C so that it can print text with the lcd.print command. On the LCD screen there are 2 rows and 16 columns where to write the text in the first row use the command setCursor(0,0) while in the second row use the command setCursor(0,1)

TASK

a. Schema in Fritzing

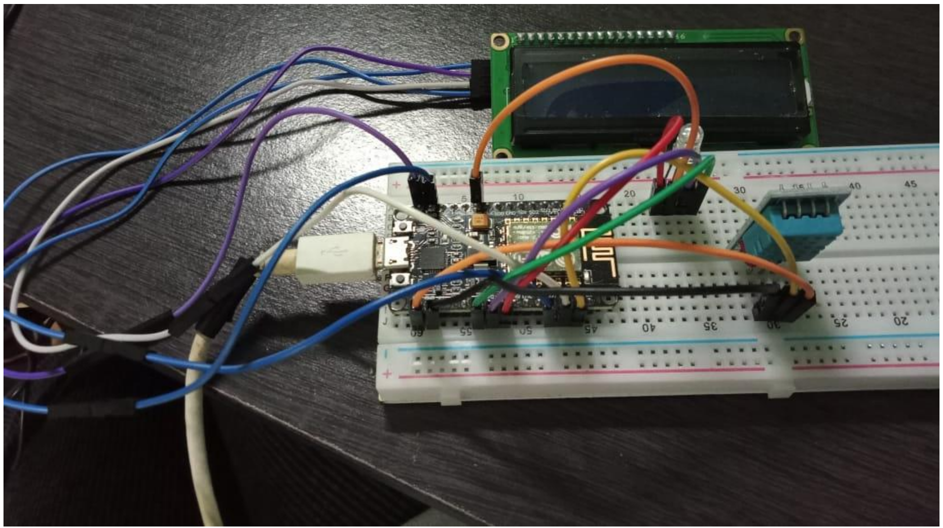


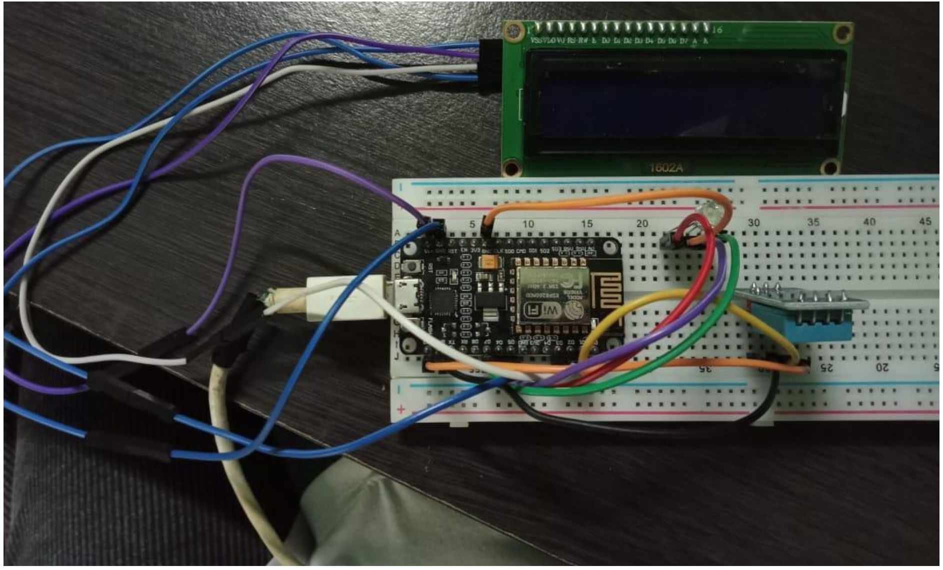
b. Program Code

```
#include <Arduino.h>
#include <Wire.h>
#include <LiquidCrystal_I2C.h>
#include <ESP8266WiFi.h>
#include <NTPClient.h>
#include <WiFiUdp.h>
// #include <SimpleDHT.h>
#include <Adafruit_Sensor.h>
#include <DHT.h>

// Network credentials
const char *ssid = "TP-Link_9144";
const char *password = "Tole1234";

// Define NTP Client to get time
WiFiUDP ntpUDP;
NTPClient timeClient(ntpUDP, "pool.ntp.org");

// Week Days
String weekDays[7] = {"Sunday", "Monday", "Tuesday", "Wednesday",
"Thursday", "Friday", "Saturday"};
// Month names
String months[12] = {"January", "February", "March", "April",
"May", "June", "July", "August", "September", "October",
"November", "December"};

int lcdColumns = 16;
int lcdRows = 2;

LiquidCrystal_I2C lcd(0x27, lcdColumns, lcdRows);


#define DHTTYPE DHT11
DHT dht(D1, DHTTYPE); // instan sensor dht11

#define RED_LED D5
#define GREEN_LED D6
#define BLUE_LED D7


String tempNow = "temperature";
String timeNow = "time";

void setup() {
  Serial.begin(9600);
  Serial.print("Connecting to ");
  Serial.println(ssid);
  WiFi.begin(ssid, password);
  while (WiFi.status() != WL_CONNECTED)
  {
    delay(500);
    Serial.print(".");
  }
  timeClient.begin();
  timeClient.setTimeOffset(25200);

  pinMode(RED_LED, OUTPUT);
  pinMode(GREEN_LED, OUTPUT);
  pinMode(BLUE_LED, OUTPUT);

  Wire.begin(D2, D3);
  lcd.init();
  lcd.backlight();
}


void getTemp(){

  float c = dht.readTemperature();
  float f = dht.readTemperature(true);
  if (isnan(c) || isnan(f))
  {
    lcd.setCursor(0, 0);
    lcd.print("Err Temp");
    Serial.println("Failed to read from DHT sensor!");
    return;
  }
  if(c<20){

    digitalWrite(RED_LED, LOW);
    digitalWrite(GREEN_LED, HIGH);
    digitalWrite(BLUE_LED, LOW);
  }

  else if(c<28){
    digitalWrite(RED_LED, LOW);
    digitalWrite(GREEN_LED, LOW);
    digitalWrite(BLUE_LED, HIGH);
  }

  else if(c>=28){
    digitalWrite(RED_LED, HIGH);
    digitalWrite(GREEN_LED, LOW);
    digitalWrite(BLUE_LED, LOW);
  }

  tempNow = String(c) + char(0b11011111) + "C " + String(f) +
char(0b11011111) + "F";
  Serial.println();
  Serial.print("Celcius Temp      : ");
  Serial.print(c);
  Serial.print(" *C, ");
  Serial.println();
  Serial.print("Fahrenheit Temp   : ");
  Serial.print(f);
  Serial.print(" *F, ");
  Serial.println();
  lcd.setCursor(0, 0);
  lcd.print(tempNow);
}

void getTime(){
  timeClient.update();
  time_t epochTime = timeClient.getEpochTime();
  String formattedTime = timeClient.getFormattedTime();
  int currentHour = timeClient.getHours();
  int currentMinute = timeClient.getMinutes();
  int currentSecond = timeClient.getSeconds();
  String weekDay = weekDays[timeClient.getDay()];
  struct tm *ptm = gmtime((time_t *)&epochTime);
  int monthDay = ptm->tm_mday;
  int currentMonth = ptm->tm_mon + 1;
  String currentMonthName = months[currentMonth - 1];
  int currentYear = ptm->tm_year + 1900;
  String currentDate = String(monthDay) + "-" +
String(currentMonth) + "-" + String(currentYear);
```

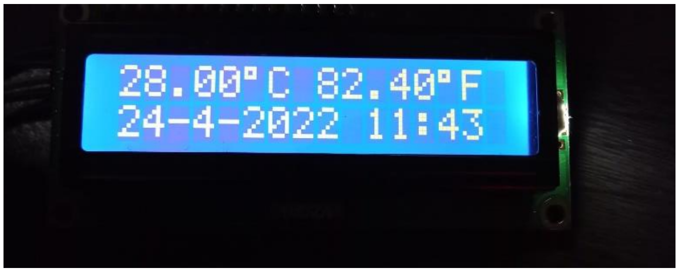
```
  timeNow = currentDate + " " + String(currentHour) + ":" +
String(currentMinute);
  Serial.println(timeNow);
  Serial.println("-----------------------------------");
  lcd.setCursor(0, 1);
  lcd.print(timeNow);
}

void loop() {
  getTemp();
  getTime();
  delay(3000);
}
```

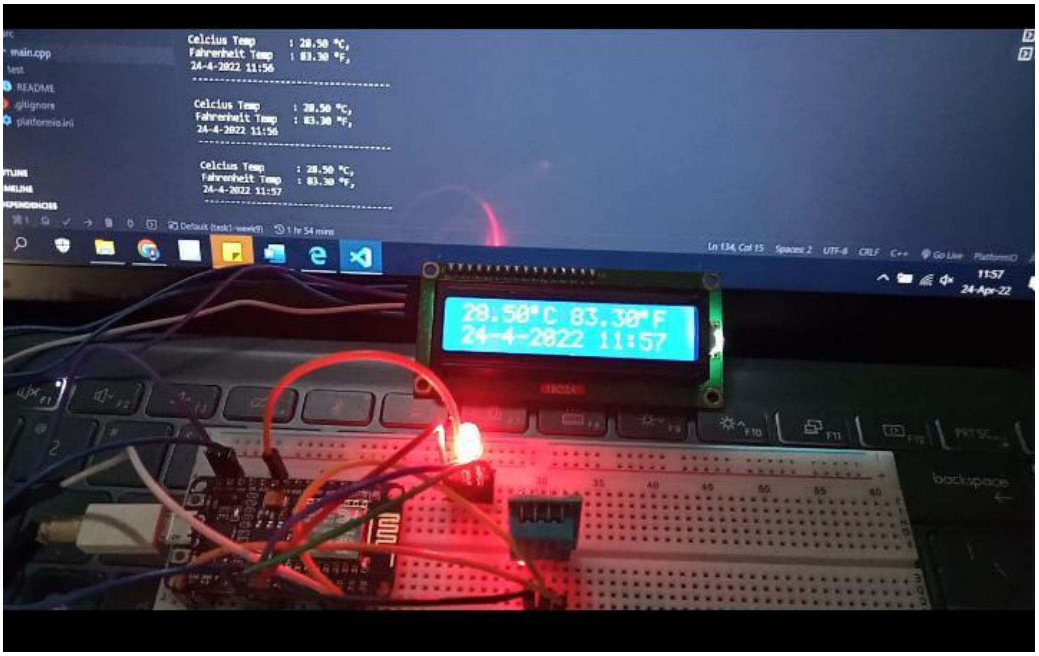
c. Schema





d. Result




```
Celcius Temp      : 28.50 °C,
Fahrenheit Temp   : 83.30 °F,
24-4-2022 11:56
-----------------------------------

Celcius Temp      : 28.50 °C,
Fahrenheit Temp   : 83.30 °F,
24-4-2022 11:56
-----------------------------------

Celcius Temp      : 28.50 °C,
Fahrenheit Temp   : 83.30 °F,
24-4-2022 11:57
-----------------------------------
```

Video : [Link]

e. Explanation

In the given task, which is to display the real time on the LCD and also display the temperature, I encountered some problems. Among other things: the DHT sensor cannot work if the data pin is connected to the d8 pin, so it must be replaced on the d1 pin and use the adafruit sensor. Then to display the time on the LCD, the problem is that the wifi on the laptop is not connected to esp so it takes some time to fix it.