

**LAPORAN PROGRES MINGGUAN
FINAL PROJECT SISTEM TERTANAM**



Disusun oleh:

Iqbal Muchlis 5024201073

Dosen Pembimbing:

Eko Pramunanto, S.T., M.T.

**FAKULTAS TEKNOLOGI ELEKTRO DAN INFORMATIKA CERDAS
TEKNIK KOMPUTER
INSTITUT TEKNOLOGI SEPULUH NOPEMBER
SURABAYA
2023**

I. PENJELASAN TUGAS

Rancanglah sistem untuk menampilkan jam, kalender, suhu dan alarm dengan karakter pada dot matrix LED 8 x 32 (4 buah matrix LED 8x8), dengan input keyboard USB. Suhu diukur dengan menggunakan sensor suhu analog (seperti LM35 atau yang sejenisnya).

Spesifikasi fitur:

- a. mode : run, set jam, set tanggal, set alarm.
- b. select set : jam, menit, detik / tgl, bln, thn,
- c. 3 waktu alarm dengan text (wajib) + buzzer (optional nilai plus):

- alarm 1: Display NRP
- alarm 2: Display NRP + Nama
- alarm 3: Display text yang diinputkan

Waktu aktif untuk alarm 1, 2 dan 3 bisa diset dengan tanggal, jam, menit dan durasinya dalam detik.

d. Tampilan:

- Kecerahan diatur sesuai kecerahan lingkungan dengan sensor cahaya.
- Jam, menit, detik
- Pada setiap detik ke 10 dan 40 tampilkan tgl-bln-thn selama 3 detik
- Pada setiap detik ke 13 dan 43 tampilkan suhu dengan keterangan °C (derajat Celcius)
- Pada saat waktu tepat sama dengan waktu seting alarm tampilkan text alarm sesuai durasi setingnya.
- Kecepatan geser tampilan text panjang (running text) adalah 0.5 detik per kolom matrix LED.
- Sebagai tambahan nilai : tampilan berkedip saat berada pada mode seting (input dari keyboard).

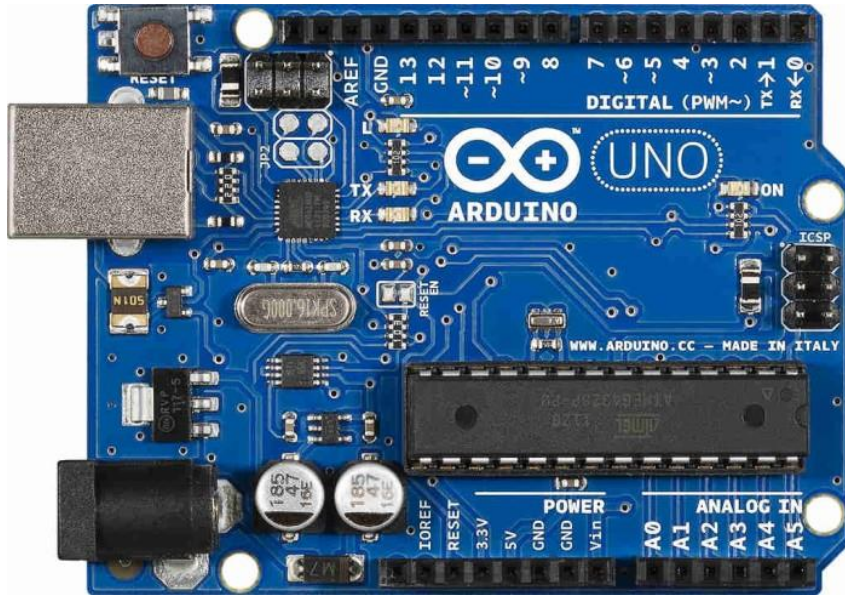
Output:

- 8x32 dot matrix LED

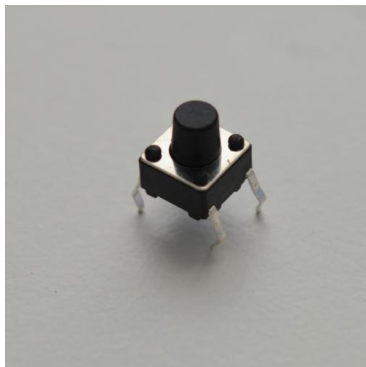
II. ALAT DAN BAHAN

Alat dan bahan yang saya pakai untuk final project ini adalah:

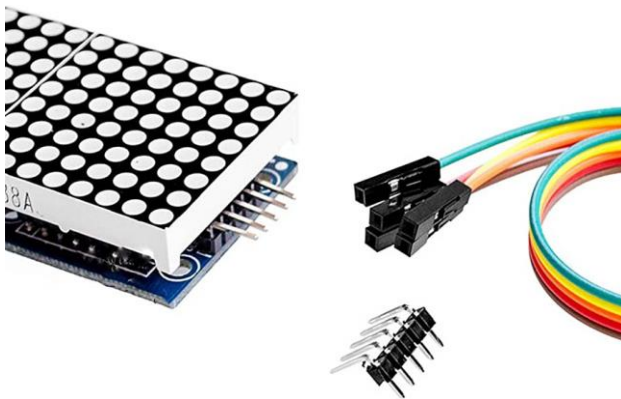
- Arduino UNO



- Button



- MAX7219 LED Dot Matrix Module 4-IN-1 32x8



- LM35 (analog temperature sensor)

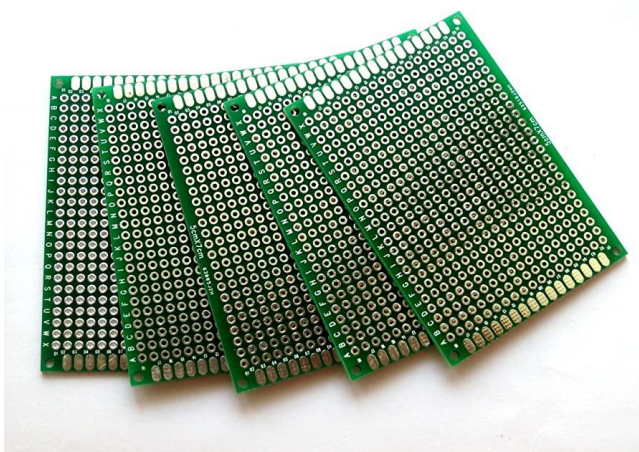


-
- A blue printed circuit board (PCB) module, likely a sensor or communication module, is shown. It features a black integrated circuit (IC) in the center, a small silver coin cell battery, and several pins labeled: GND, VCC, SDA, SCL, and a header with pins labeled 12K, 10W, 3CL, 3DA, VCC, and GND. The module is connected to a black cable.

-

-

- PCB



- Kabel
 - Male to female untuk dot matrix



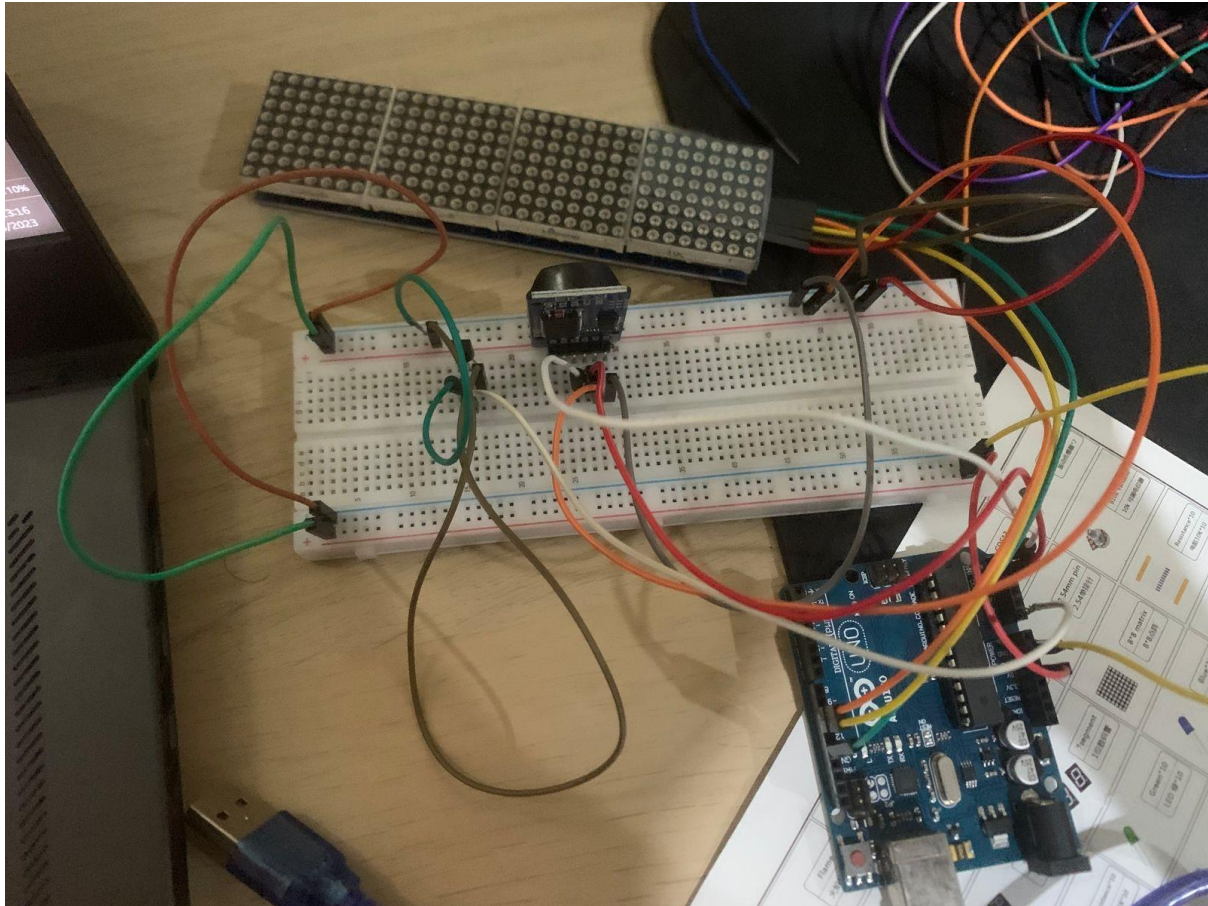
- Male to male untuk selain dot matrix



*detail berapa banyak kabel/button/dll akan saya keluarkan pada laporan terakhir.

III. RANGKAIAN SEMENTARA

Berikut merupakan rangkaian semetara pada breadboard yang telah saya rancang:



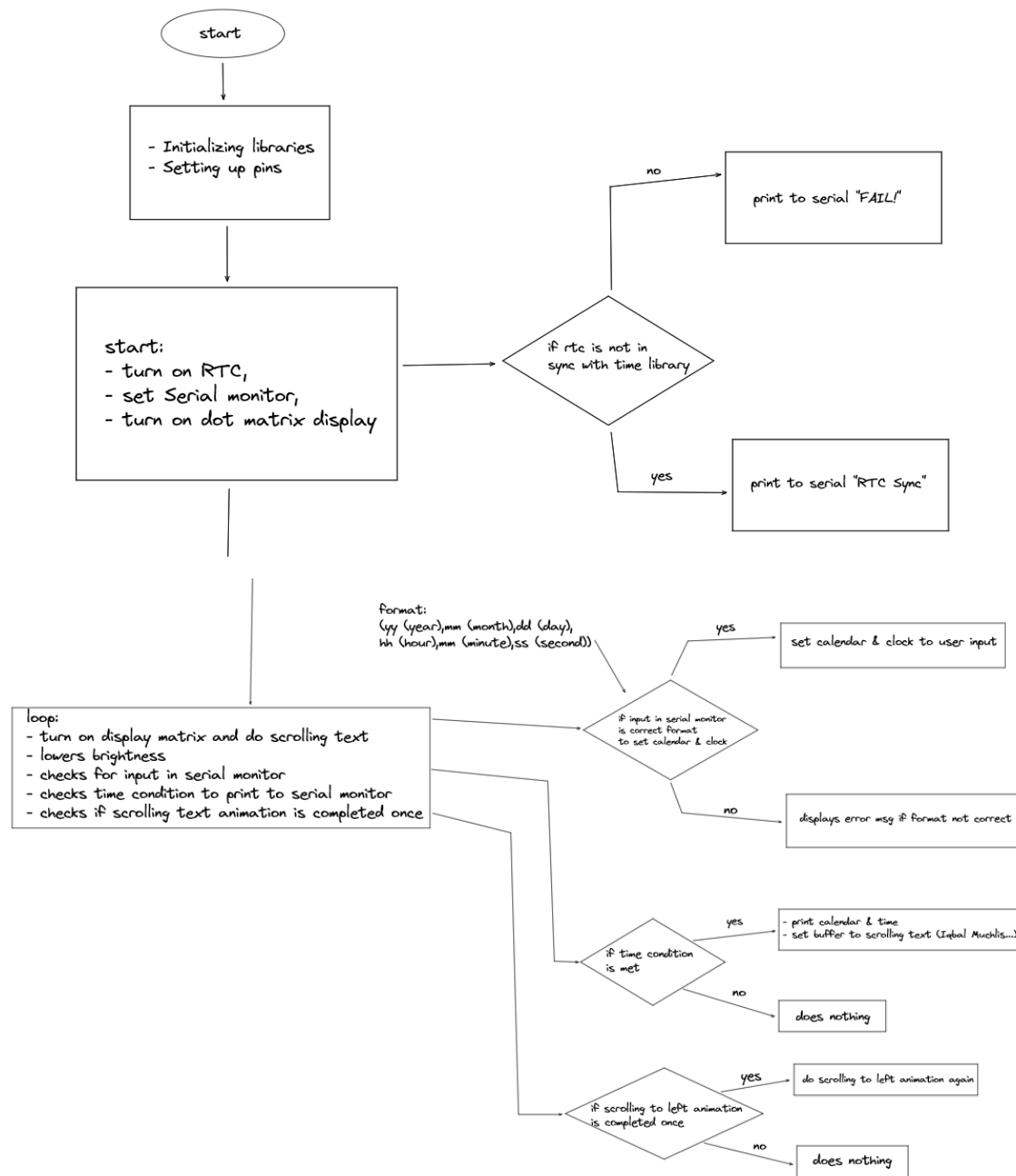
Detail rangkaian:

- Dot matrix
 - VCC disambungkan pada 5v di arduino
 - GND disambungkan pada GND di arduino
 - DIN disambungkan pada pin 11 di arduino
 - CS disambungkan pada pin 10 di arduino
 - CLK disambungkan pada pin 13 di arduino
- RTC
 - VCC disambungkan pada 5v di arduino
 - GND disambungkan pada GND di arduino
 - SDA disambungkan pada A4 di arduino
 - SCL disambungkan pada A5 di arduino
- LM35

- 4-20V disambungkan pada 5v di arduino
- GND disambungkan pada GND di arduino
- OUT disambungkan pada A0 di arduino

IV. PROGRESS

- Diagram blok:



- Dengan menggunakan library:

- MD_MAX72XX (https://github.com/MajicDesigns/MD_MAX72XX)
 - untuk mengatur display dot matrix
 - kemungkinan akan saya hapus karena bisa digunakan dengan MD_Parola saja
- MD_Parola (https://github.com/MajicDesigns/MD_Parola)
 - untuk mengatur, memunculkan tampilan, scrolling text, dll pada display dot matrix

- Streaming (<https://github.com/janelia-arduino/Streaming>)
 - sehingga dapat melakukan cout << variable << endl seperti pada c++
 - dilakukan dengan cara Serial << variable << endl
- TimeLib (<https://github.com/PaulStoffregen/Time>)
 - untuk mendapatkan time
- DS3232RTC (<https://github.com/JChristensen/DS3232RTC>)
 - untuk mengatur real time clock
- LM35 (<https://github.com/wilmouths/LM35>)
 - untuk mengatur sensor temperature agar lebih mudah
- Hasil:
 - dapat menyalakan display dot matrix
 - dapat mengatur brightness dot matrix
 - dapat menampilkan text/angka pada display dot_matrix
 - dapat melakukan scrolling text
 - dapat melakukan set kalender dan jam dengan keyboard
 - dapat menampilkan real time untuk:
 - jam
 - kalender
 - suhu

- Code:

```
// TODO:
// - check if temp is accurate or not
// - make alarm
// - implement light sensor to check brightness
// - if can fix high memory usage
// - problem is at library itself "MD_Parola P =
MD_Parola(HARDWARE_TYPE, CS_PIN, MAX_DEVICES);" causes like 70%
memory usage
// //- display clock, calendar properly
// //- make text scroll

// dot matrix
#include <MD_Parola.h> // https://github.com/MajicDesigns/MD_Parola
// #include <MD_MAX72xx.h> //
https://github.com/MajicDesigns/MD_MAX72XX)
#include <SPI.h>
#define HARDWARE_TYPE MD_MAX72XX::FC16_HW
#define MAX_DEVICES 4
```

```

#define CLK_PIN 13
#define DATA_PIN 11
#define CS_PIN 10
MD_Parola P = MD_Parola(HARDWARE_TYPE, CS_PIN, MAX_DEVICES);
// real time clock
#include <Streaming.h> // https://github.com/janelia-
arduino/Streaming
#include <TimeLib.h> // https://github.com/PaulStoffregen/Time
#include <DS3232RTC.h> // https://github.com/JChristensen/DS3232RTC
DS3232RTC myRTC;
// temp sensor
#include <LM35.h> // https://github.com/wilmouths/LM35
LM35 lm35(A0);

// for snprintf
char buffer[60];

void setup() {
    Serial.begin(115200);
    myRTC.begin();
    P.begin();

    // setSyncProvider() causes the Time library to synchronize with
the
    // external RTC by calling RTC.get() every five minutes by
default.
    setSyncProvider(myRTC.get);
    Serial << F("RTC Sync");
    if (timeStatus() != timeSet) Serial << F(" FAIL!");
    Serial << endl;

    P.displayClear();
}

void loop() {
    P.setIntensity(0);
    // change time_t (uint32_t) to uint16_t if need to free more
memory
    static time_t tLast;
    time_t t;
    tmElements_t tm;

    // check for input to set the RTC, minimum length is 12, i.e.
yy,m,d,h,m,s
    if (Serial.available() >= 12) {
        // note that the tmElements_t Year member is an offset from
1970,
        // but the RTC wants the last two digits of the calendar year.

```

```

    // use the convenience macros from the Time Library to do the
    conversions.

    int y = Serial.parseInt();
    if (y >= 100 && y < 1000)
        Serial << F("Error: Year must be two digits or four digits!")
<< endl;
    else {
        if (y >= 1000)
            tm.Year = CalendarYrToTm(y);
        else // (y < 100)
            tm.Year = y2kYearToTm(y);
        tm.Month = Serial.parseInt();
        tm.Day = Serial.parseInt();
        tm.Hour = Serial.parseInt();
        tm.Minute = Serial.parseInt();
        tm.Second = Serial.parseInt();
        t = makeTime(tm);
        myRTC.set(t); // use the time_t value to ensure correct
weekday is set
        setTime(t);
        Serial << F("RTC set to: ");
        printDateTime(t);

        // sprintf(buffer, "%d %s %d %d:%d:%d", day(t),
monthShortStr(month(t)), _DEC(year(t)), hour(t), minute(t),
second(t));
        // P.displayText(buffer, PA_CENTER, 25, 25, PA_SCROLL_LEFT,
PA_SCROLL_LEFT);
        // P.displayReset();

        Serial << endl;
        // dump any extraneous input
        while (Serial.available() > 0) Serial.read();
    }
}

t = now();
// sprintf(buffer, "%d.%d.%d > %d %s %d", hour(t), minute(t),
second(t), day(t), monthShortStr(month(t)), _DEC(year(t)));
if (t != tLast) {
    tLast = t;
    // prints calendar and clock
    printDateTime(t);

    // // prints celcius
    // uint8_t c = myRTC.temperature() / 4.;
    // Serial << c << F(" C ");

```

```

    // from LM35 (celcius)
    // int celcius = analogRead(LM_35);
    // int displayCelcius = celcius * (5.0/1023.0) * 100;
    // Serial << displayCelcius << F(" C");
    // displays to dot matrix
    // snprintf(buffer, sizeof(buffer), "Iqbal Muchlis 5024201073 ->
%d.%d.%d >> %d %s %d >> %d C", hour(t), minute(t), second(t),
day(t), monthShortStr(month(t)), _DEC(year(t)), displayCelcius);
    Serial << lm35.getTemp(CELCIUS) << " " << myRTC.temperature() /
4.0;
    int celciusTemp = lm35.getTemp(CELCIUS);
    snprintf(buffer, sizeof(buffer), "Iqbal Muchlis 5024201073 -> %d
C >> %d.%d.%d >> %d %s %d", celciusTemp, hour(t), minute(t),
second(t), day(t), monthShortStr(month(t)), _DEC(year(t)));
    // snprintf(buffer, sizeof(buffer), "%d C", celciusTemp);

    // float voltage = analogRead(LM_35) * (5.0 / 1023.0);
    // float celcius = (voltage - 0.5) * 100.0;
    // Serial << celcius << F(" C");

    // snprintf(buffer, sizeof(buffer), "Iqbal Muchlis 5024201073 ->
%d.%d.%d >> %d %s %d >> %0.2f C", hour(t), minute(t), second(t),
day(t), monthShortStr(month(t)), _DEC(year(t)), celcius);

    // THIS FLOAT BELOW IS EATING TOO MUCH MEMORY DONT USE IT

    // (this was once a frightening float) uint8_t c =
myRTC.temperature() / 4.;
    // (this was once a frightening float) uint8_t f = c * 9. / 5. +
32.;
    // Serial << F(" ") << c << F(" C ") << f << F(" F");

    Serial << endl;
}
// sprintf(buffer, "%d %s %d %d:%d:%d", day(t),
monthShortStr(month(t)), _DEC(year(t)), hour(t), minute(t),
second(t));
if (P.displayAnimate()) {
    P.displayText(buffer, PA_CENTER, 25, 25, PA_SCROLL_LEFT,
PA_SCROLL_LEFT);
    P.displayReset();
}
}

// print date and time to Serial

```

```

void printDateTime(time_t t) {
    printDate(t);
    Serial << ' ';
    printTime(t);

    // sprintf(buffer, "%d %s %d %d:%d:%d", day(t),
monthShortStr(month(t)), _DEC(year(t)), hour(t), minute(t),
second(t));
    // 23,05,09,16,53,10
    // sprintf(buffer, "%d.%d.%d > %d %s %d", hour(t), minute(t),
second(t), day(t), monthShortStr(month(t)), _DEC(year(t)));
    // sprintf(buffer, "%d %s %d", day(t), monthShortStr(month(t)),
_DEC(year(t)));
    // if (P.displayAnimate()) {
    //     P.displayText(buffer, PA_CENTER, 0, 0, PA_SCROLL_LEFT,
PA_SCROLL_LEFT);
    //     P.displayReset();
    // }
}

// print time to Serial
void printTime(time_t t) {
    printI00(hour(t), ':');
    printI00(minute(t), ':');
    printI00(second(t), ' ');
}

// print date to Serial
void printDate(time_t t) {
    printI00(day(t), 0);
    Serial << "-" << monthShortStr(month(t)) << "-" << _DEC(year(t));
}

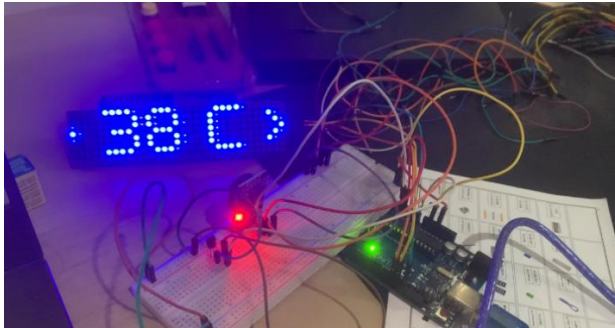
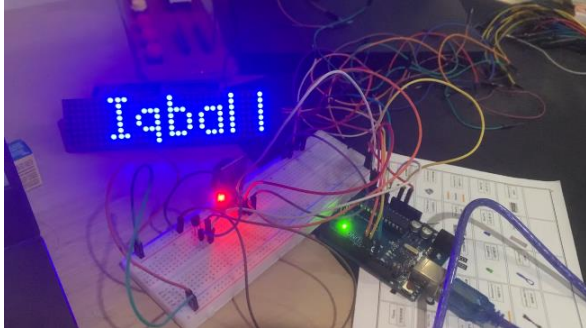
// Print to serial an integer in "00" format (with leading zero),
// followed by a delimiter character.
// Input value assumed to be between 0 and 99.
void printI00(int val, char delim) {
    if (val < 10) Serial << '0';
    Serial << _DEC(val);
    if (delim > 0) Serial << delim;
    return;
}

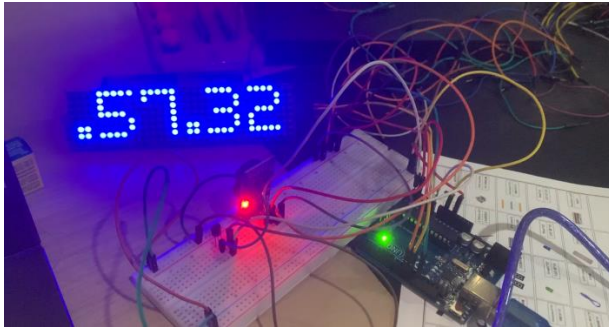
```

- Hasil display

dibawah ini merupakan scrolling text dari variable buffer yang berisi:

“Iqbal Muchlis 5024201073 -> *suhu* >> *clock* >> *calendar*”





V. KESIMPULAN

Dengan selesainya implementasi:

- input keyboard untuk setting jam dan kalender (pada example dalam library RTC yang saya temukan saya dapat langsung melakukan setting serial input keyboard terhadap jam dan kalender)
- kalender
- jam
- suhu
- scrolling text

Saya bisa fokus untuk melanjutkan implelementasi terhadap:

- pembuatan alarm, bisa menggunakan library dari RTC
- input keyboard untuk alarm
- sensor cahaya yang mengatur brightness
- spesifikasi fitur sesuai dengan penjelasan tugas (mode/select set/tampilan yang sesuai)
- improve memory usage karena sekarang code saya memakai 92% storage space