



Perancangan Sistem Pengukuran Suhu Dan Kelembaban Otomatis Dengan Md_Parola Dan Sensor Dht22

Ihsanulfu'ad Suwandi

¹Jurusan Teknik Informatika-Pendidikan Teknologi Informasi, Universitas Negeri Gorontalo

ihsansuwandi@ung.ac.id

ABSTRACT

The increasing demand for environmental monitoring has led to the development of efficient and effective monitoring systems. This research presents a prototype environmental monitoring system using MD_Parola, LED Matrix, DHT22 sensor, and RTC DS1307. The system provides information on temperature, humidity, and time through an engaging visual interface. The DHT22 sensor delivers accurate measurements of temperature and humidity, while the RTC DS1307 ensures consistent time synchronization. Functional testing of this system demonstrates quick response times and optimal performance. User evaluation confirms the user interface's intuitiveness and the effectiveness of data visualization. Data analysis indicates that measurement results align with expected environmental conditions. Conclusions state that the system successfully achieves research objectives and can be widely implemented in environmental monitoring. Suggestions for further development include exploring integration with remote monitoring platforms and adding additional sensors. Thus, the system contributes to understanding and maintaining environmental quality with an innovative and integrated approach.

Keywords: DHT22 Sensor, Environmental Monitoring, MD_Parola, Matrix, RTC DS1307

ABSTRAK

Peningkatan kebutuhan akan pemantauan lingkungan mendorong pengembangan sistem monitoring yang efisien dan efektif. Penelitian ini menghasilkan *prototipe* sistem *monitoring* lingkungan yang menggunakan MD_Parola, LED Matrix, *sensor* DHT22, dan RTC DS1307. Sistem ini dapat menyajikan informasi suhu, kelembaban, dan waktu melalui antarmuka visual yang menarik. Sensor DHT22 memberikan pengukuran suhu dan kelembaban dengan tingkat akurasi yang baik, sementara RTC DS1307 menyediakan sinkronisasi waktu yang konsisten. Pengujian fungsional sistem ini memperlihatkan respons yang cepat dan performa yang optimal. Evaluasi pengguna mengonfirmasi keintuitifan antarmuka pengguna dan efektivitas visualisasi data. Analisis data menunjukkan bahwa hasil pengukuran sesuai dengan kondisi lingkungan yang diharapkan. Kesimpulan menyatakan bahwa sistem ini berhasil mencapai tujuan penelitian dan dapat diimplementasikan secara luas dalam pemantauan lingkungan. Saran untuk pengembangan lebih lanjut mencakup eksplorasi integrasi dengan platform pemantauan jarak jauh dan penambahan sensor tambahan. Dengan demikian, sistem ini memberikan kontribusi pada pemahaman dan pemeliharaan kualitas lingkungan dengan pendekatan yang inovatif dan terintegrasi.

Kata Kunci: DHT22 Sensor, Environmental Monitoring, MD_Parola, Matrix, RTC DS1307

I. PENDAHULUAN

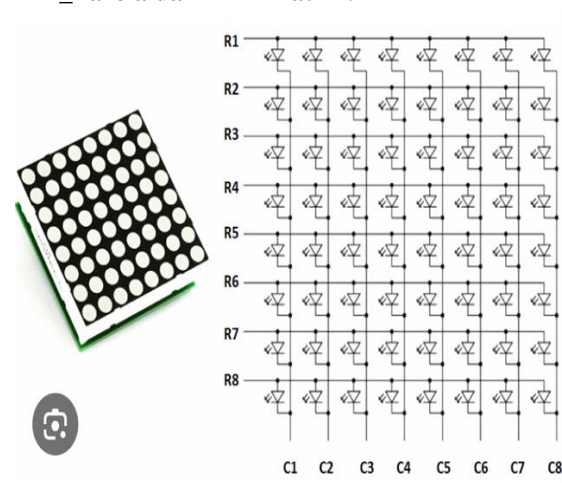
Perkembangan teknologi informasi yang semakin pesat, kebutuhan akan sistem monitoring lingkungan menjadi semakin mendesak. *Monitoring* suhu dan kelembaban merupakan aspek krusial dalam pemahaman kondisi lingkungan, baik untuk aplikasi rumah tangga, industri, maupun pertanian. *Prototyping* atau pembuatan prototipe adalah tahap pengembangan produk yang mengoptimalkan sumber daya dan biaya yang terbatas untuk menguji kinerja dan kegunaannya. Sistem monitoring yang efektif dapat memberikan informasi *real-time*, memungkinkan pengguna untuk mengambil tindakan yang tepat waktu [1].

Penyampaian informasi ini dapat dirancang menjadi alat pengontrolan *running text* untuk meningkatkan inovasi yang ada dengan menggunakan suara dan arduino uno dengan memanfaatkan smartphone android sebagai sarana penunjang dalam penyampaian informasi, banyak sistem *monitoring* saat ini masih kurang memanfaatkan potensi teknologi visualisasi informasi [2]. Penyajian data dalam bentuk yang menarik dan mudah dimengerti dapat meningkatkan pemahaman pengguna terhadap kondisi lingkungan yang diukur. Oleh karena itu, penelitian ini akan fokus pada perancangan dan implementasi sistem pengukuran suhu dan kelembaban otomatis yang mengintegrasikan modul LED Matrix (MD_Parola), sensor DHT22, dan RTC DS1307.

Penggunaan MD_Parola sebagai media penyampaian informasi, [3] Matriks LED P10 merupakan rangkaian LED berukuran 16 x 32 cm yang dapat digunakan untuk menampilkan teks matriks adalah rangkaian LED yang membentuk sejumlah kolom dan baris tertentu, memberikan kemungkinan untuk menampilkan data suhu, kelembaban, dan waktu dalam bentuk visual yang lebih menarik. Dengan demikian, penelitian ini bertujuan untuk menggali potensi efektivitas visualisasi data pada modul LED Matrix untuk meningkatkan pemahaman dan respons pengguna terhadap kondisi lingkungan.

Selain itu, penggunaan sensor DHT22 dan RTC DS1307 diharapkan dapat memberikan akurasi pengukuran dan sinkronisasi waktu yang tinggi. [4] Dalam pengujian ditemukan kesalahan rata-rata 0,34% untuk suhu dan 0,75% untuk kelembaban dibandingkan dengan thermohygrometer merek Krisbow KW06 561 dengan sensor DHT 22. Hal ini penting untuk memastikan bahwa data yang ditampilkan adalah representasi yang akurat dari kondisi lingkungan pada saat tertentu.

Beberapa masalah utama yang dapat dijelajahi terkait Efektivitas Tampilan Informasi (1) Bagaimana penggunaan modul LED Matrix (MD_Parola) dapat meningkatkan efektivitas penyampaian informasi suhu, kelembaban, dan waktu? (2) Apakah efek animasi dan teks yang digunakan pada modul LED Matrix dapat meningkatkan pemahaman dan perhatian pengguna terhadap data yang ditampilkan? **MD_Parola dan LED Matrix:**



GAMBAR 1.

LED Matrix (Sumber: Google.com)

MD_Parola adalah sebuah perpustakaan (library) yang digunakan untuk mengontrol dan menampilkan teks pada matriks LED. Perpustakaan ini dirancang khusus untuk digunakan dengan berbagai jenis matriks LED, termasuk matriks LED dot-matrix dan modul-modul LED lainnya. MD_Parola menyediakan berbagai efek animasi, cara mengontrol kecepatan animasi, dan dukungan untuk beberapa zona teks pada satu waktu.

Beberapa fitur MD_Parola meliputi:

1. Animasi: Mendukung berbagai efek animasi seperti bergulir, muncul, dan lainnya.
2. Zona Teks: Memungkinkan pengguna untuk mendefinisikan beberapa zona teks yang dapat ditampilkan secara bersamaan atau bergantian.
3. Kustomisasi Efek: Memungkinkan kontrol yang baik terhadap kecepatan dan efek animasi.

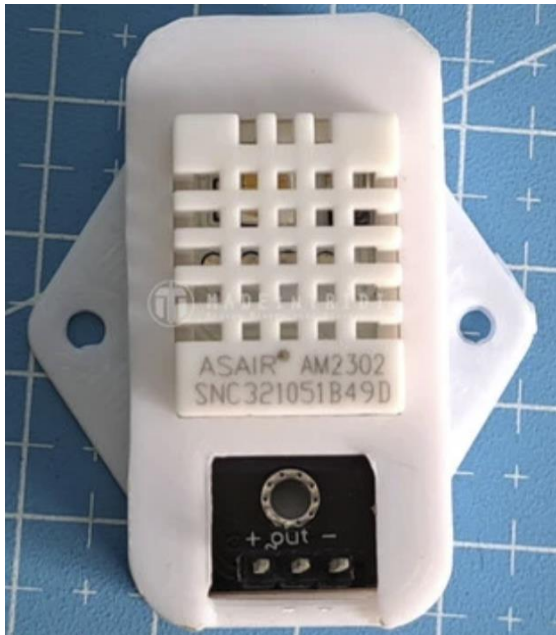
LED Matrix:

LED matrix adalah tata letak dari sejumlah lampu LED yang membentuk matriks, biasanya dalam bentuk persegi atau persegi panjang. Setiap lampu LED diatur dalam baris dan kolom, dan dengan mengontrol baris dan kolom ini, kita dapat menyalakan atau mematikan setiap lampu LED individual untuk membuat tampilan atau teks yang diinginkan.

Kaitan MD_Parola dan LED Matrix, MD_Parola digunakan sebagai antarmuka perangkat lunak untuk mengontrol dan menampilkan teks atau animasi pada LED matrix. Ini menyediakan lapisan abstraksi yang mempermudah pengembangan perangkat lunak untuk menangani tampilan pada matriks LED. Dengan menggunakan MD_Parola, pengguna dapat dengan mudah mengatur teks, efek animasi, dan zona-zona khusus pada LED matrix tanpa harus menangani detail teknis pengendalian matriks LED secara langsung.

Dengan mengintegrasikan MD_Parola dan LED Matrix, pengembang dapat membuat proyek-proyek tampilan yang dinamis dan menarik dengan lebih mudah dan efisien. [5] Monitoring konsumsi daya yang digunakan pada panel LED Running Text P10 pada kondisi semua LED menyala, kondisi statis, kondisi dinamis dan saat dikurangi 1-2 blok sebesar arus sehingga bloknya mati. Hal itu bertujuan untuk mengetahui berapa besar konsumsi daya yang terpakai saat LED menyala. Proses ini diterapkan untuk mengetahui kondisi yang menampilkan informasi dengan konsumsi daya rendah dan ideal

Sensor DHT22:



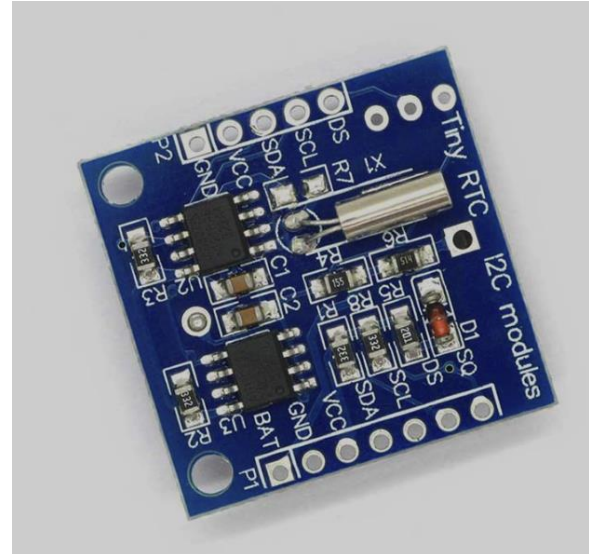
GAMBAR 2.

DHT 22 (Sumber: Google.com)

Sensor ini dapat memberikan pembacaan suhu dan kelembaban dengan presisi tinggi dan komunikasi sederhana melalui satu saluran data. [6] Sensor DHT 22 beroperasi pada suhu kisaran antara -40°C sampai 80°C , memiliki akurasi $\pm 0,5^{\circ}\text{C}$ dan kelembaban antara $0-100\%$.

Sensor DHT22 sering digunakan dalam proyek-proyek yang memerlukan pemantauan kondisi lingkungan, kendali iklim, atau sistem otomatisasi rumah pintar. Dengan akurasi tinggi dan kemampuan komunikasi yang sederhana, DHT22 menjadi pilihan yang baik.

RTC DS1307



GAMBAR 3.

RTC DS1307 (Sumber: Google.com)

DS1307 adalah sebuah chip Real-Time Clock (RTC) yang umum digunakan dalam proyek-proyek berbasis mikrokontroler untuk menyediakan informasi waktu yang akurat. RTC berfungsi untuk melacak waktu yang terus berjalan bahkan ketika daya pada mikrokontroler atau sistem mati. DS1307 biasanya digunakan bersama dengan mikrokontroler seperti Arduino untuk menyediakan fungsi waktu dalam aplikasi.

Beberapa fitur utama dari RTC DS1307 meliputi:

1. Penyimpanan Informasi Waktu, DS1307 dapat menyimpan informasi waktu saat ini, termasuk jam, menit, detik, hari, tanggal, bulan, dan tahun.
2. Presisi Waktu yang Tinggi, RTC ini menyediakan presisi waktu yang tinggi, yang penting untuk aplikasi yang memerlukan pemantauan waktu yang akurat.
3. Baterai Cadangan, DS1307 sering dilengkapi dengan baterai cadangan kecil (biasanya baterai lithium) sehingga dapat terus menyimpan informasi waktu bahkan ketika daya utama mati.
4. Komunikasi I2C, DS1307 berkomunikasi dengan mikrokontroler melalui antarmuka I2C



(Inter-Integrated Circuit), yang memudahkan integrasinya dengan mikrokontroler.

5. Aplikasi Umum, RTC DS1307 digunakan dalam berbagai aplikasi seperti jam digital, logger data waktu, kontrol waktu pada sistem otomatisasi, dan proyek-proyek IoT yang memerlukan sinkronisasi waktu.
6. Modul yang Tersedia: DS1307 sering dijual dalam bentuk modul yang memudahkan penggunaan dalam proyek-proyek DIY. Modul ini biasanya mencakup kristal osilator dan baterai cadangan.

[7] Jam Real time berbasis Arduino ini merupakan jam digital untuk menampilkan waktu nyata dengan menggunakan IC RTC DS1307 yang bekerja pada protokol I2C. Jam *Real Time* berarti jam tetap berjalan bahkan setelah listrik mati. Saat daya disambungkan kembali, ini akan menampilkan waktu nyata terlepas dari waktu dan durasi daya dalam keadaan mati.

Sebelumnya, telah dilakukan sejumlah penelitian terkait pemantauan lingkungan dan pengembangan sistem berbasis sensor dengan judul [8] *The High Accurate Automatic School Bell Controller Based On Arduino Uno DS1307 I2C Real-time Clock* yang mendemonstrasikan sistem bel sekolah otomatis yang dapat secara otomatis menghasilkan suara notifikasi selama kegiatan belajar mengajar, mengoptimalkan konsumsi daya, dan menghasilkan frekuensi suara yang akurat. Tegangan dan arus keluaran berbagai bagian sistem bel sekolah otomatis masing-masing adalah 4,17 hingga 5,90 volt dan 488 hingga 602 mA. Sedangkan konsumsi dayanya sekitar 2,07-3,38 watt. Sistem bel sekolah otomatis terintegrasi dengan lancar ke dalam jadwal sekolah (*software*).

Melalui penelitian ini, diharapkan dapat diperoleh wawasan yang mendalam tentang integrasi teknologi visualisasi pada sistem monitoring lingkungan, dengan fokus pada aspek akurasi, efektivitas, dan interaksi pengguna. Hasil penelitian ini dapat memberikan kontribusi signifikan terhadap pengembangan sistem monitoring lingkungan yang lebih canggih dan dapat diadopsi secara luas untuk keberlanjutan dan efisiensi

II. METODE PENELITIAN

Metode penelitian ini juga dapat disebut sebagai "Penelitian Perancangan" atau "Design Research." Metode perancangan fokus pada tahap perancangan dan pengembangan suatu produk atau sistem, dengan penekanan pada proses desain, pembuatan prototipe, dan evaluasi. Metode ini sering digunakan dalam

konteks pengembangan teknologi, desain produk, atau pengembangan sistem baru. Pendekatan ini memandang desain sebagai suatu proses eksplorasi dan eksperimen untuk mencapai solusi yang optimal.

1. Studi Literatur:

- a. Melakukan studi literatur mendalam tentang teknologi MD_Parola, LED Matrix, sensor DHT22, dan RTC DS1307.
- b. Menelaah penelitian-penelitian terkait untuk memahami penggunaan dan keunggulan masing-masing elemen teknologi.

2. Perancangan Sistem:

- a. Merancang sistem monitoring lingkungan dengan menggunakan integrasi MD_Parola, LED Matrix, sensor DHT22, dan RTC DS1307.
- b. Menentukan spesifikasi perangkat keras dan perangkat lunak yang diperlukan untuk mencapai tujuan penelitian.

3. Pengembangan Prototipe:

- a. Membangun prototipe sistem berbasis perancangan yang telah dibuat.
- b. Mengintegrasikan perangkat keras dan perangkat lunak untuk menghasilkan sistem yang berfungsi secara menyeluruh.

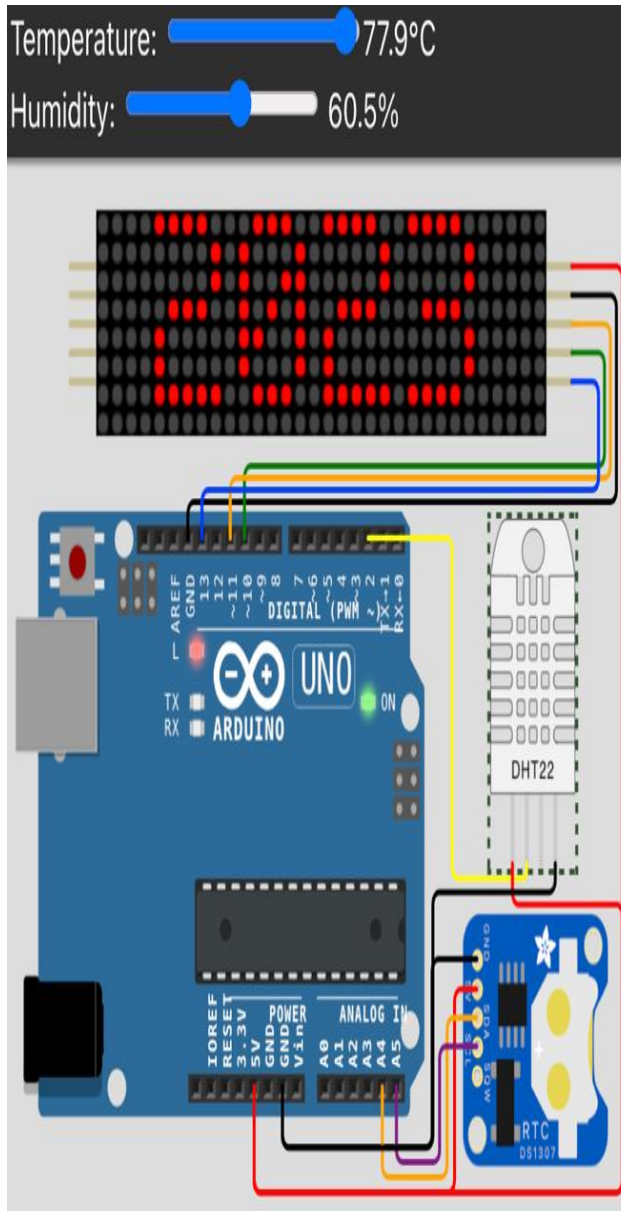
4. Kalibrasi Sensor:

- a. Melakukan kalibrasi sensor DHT22 untuk memastikan akurasi pengukuran suhu dan kelembaban.
- b. Mengevaluasi respons sensor terhadap variasi kondisi lingkungan.

5. Pengujian Fungsional:

- a. Melakukan pengujian fungsional terhadap sistem secara menyeluruh.
- b. Memastikan bahwa MD_Parola dan LED Matrix dapat menampilkan data secara efektif, sensor DHT22 mengukur suhu dan kelembaban dengan akurat, dan RTC DS1307 menyediakan waktu yang sinkron.

III. HASIL DAN PEMBAHASAN



GAMBAR 4.
RANCANGAN PENGUKURAN SUHU DAN
KELEMBABAN OTOMATIS

```
//Ihsanulfu'ad Suwandi
#include <MD_Parola.h>
#include <MD_MAX72xx.h>
#include <DHT.h>
#include <SPI.h>
#include <Wire.h>
#include "Font7Seg.h"
#define HARDWARE_TYPE
MD_MAX72XX::PAROLA_HW
```

```
#define MAX_DEVICES 4
#define CLK_PIN 13
#define DATA_PIN 11
#define CS_PIN 10
#define SPEED_TIME 75
#define PAUSE_TIME 0
#define MAX_MESG 20
#define DS1307_ADDRESS 0x68
#define DHTPIN 2
#define DHTTYPE DHT22
#define TIMEDHT 1000
uint8_t wday, mday, month, year;
uint8_t hours, minutes, seconds;
char szTime[9];
char szMesg[MAX_MESG + 1] = "";
float humidity, celsius, fahrenheit;
uint8_t degC[] = { 6, 3, 3, 56, 68, 68, 68 };
uint8_t degF[] = { 6, 3, 3, 124, 20, 20, 4 };
uint8_t clear = 0x00;
uint32_t timerDHT = TIMEDHT;
DHT dht(DHTPIN, DHTTYPE);
MD_Parola P = MD_Parola(HARDWARE_TYPE,
CS_PIN, MAX_DEVICES);
void beginDS1307()
{
  Wire.beginTransmission(DS1307_ADDRESS);
  Wire.write(clear);
  Wire.endTransmission();
  Wire.requestFrom(DS1307_ADDRESS, 0x07);
  seconds = bcdToDec(Wire.read());
  minutes = bcdToDec(Wire.read());
  hours = bcdToDec(Wire.read() & 0xff);
  wday = bcdToDec(Wire.read());
  mday = bcdToDec(Wire.read());
  month = bcdToDec(Wire.read());
  year = bcdToDec(Wire.read());
}
uint8_t decToBcd(uint8_t value)
{ return ((value / 10 * 16) + (value % 10)); }
uint8_t bcdToDec(uint8_t value)
{ return ((value / 16 * 10) + (value % 16)); }
void getTime(char *psz, bool f = true)
{ sprintf(psz, "%02d%c%02d", hours, (f ? ':' : '.'),
minutes); }
void getDate(char *psz)
{ char szBuf[10];
  sprintf(psz, "%d %s %04d", mday, mon2str(month),
szBuf, sizeof(szBuf) - 1, (year + 2000)); }
void getTemperature()
{
  if ((millis() - timerDHT) > TIMEDHT) {

    timerDHT = millis();
    humidity = dht.readHumidity();
    celsius = dht.readTemperature();
    fahrenheit = dht.readTemperature(true);
    if (isnan(humidity) || isnan(celsius) ||
    isnan(fahrenheit)) {
```



```

        Serial.println("Failed to read from DHT sensor!");
        return;
    }
}
}
char *mon2str(uint8_t mon, char *psz, uint8_t len)
{
    static const __FlashStringHelper* str[] =
    {
        F("Jan"), F("Feb"), F("Mar"), F("Apr"),
        F("May"), F("Jun"), F("Jul"), F("Aug"),
        F("Sep"), F("Oct"), F("Nov"), F("Dec")
    };
    strncpy_P(psz, (const char PROGMEM *)str[mon -
1], len);
    psz[len] = '\0';
    return (psz);
}
char *dow2str(uint8_t code, char *psz, uint8_t len)
{
    static const __FlashStringHelper* str[] =
    {
        F("Sunday"), F("Monday"), F("Tuesday"),
        F("Wednesday"), F("Thursday"), F("Friday"),
        F("Saturday")
    };
    strncpy_P(psz, (const char PROGMEM *)str[code -
1], len);
    psz[len] = '\0';
    return (psz);
}
void setup(void)
{
    Wire.begin();
    P.begin(2);
    P.setInvert(false);
    P.setZone(0, MAX_DEVICES - 4, MAX_DEVICES -
1);
    P.setZone(1, MAX_DEVICES - 4, MAX_DEVICES -
1);
    P.displayZoneText(1, szTime, PA_CENTER,
SPEED_TIME, PAUSE_TIME, PA_PRINT,
PA_NO_EFFECT);
    P.displayZoneText(0, szMesg, PA_CENTER,
SPEED_TIME, 0, PA_PRINT, PA_NO_EFFECT);
    P.addChar('$', degC);
    P.addChar('&', degF);
    dht.begin();
}
void loop(void)
{
    static uint32_t lastTime = 0;
    static uint8_t display = 0;
    static bool flasher = false;
    beginDS1307();
    getTemperature();
    P.displayAnimate();
    if (P.getZoneStatus(0))

```

```

{
    switch (display)
    {
        case 0:
            P.setPause(0, 1000);
            P.setTextEffect(0, PA_SCROLL_LEFT,
PA_SCROLL_UP);
            display++;
            dtostrf(celsius, 3, 1, szMesg);
            strcat(szMesg, "$");
            break;
        case 1:
            P.setTextEffect(0, PA_SCROLL_UP,
PA_SCROLL_DOWN);
            display++;
            dtostrf(fahrenheit, 3, 1, szMesg);
            strcat(szMesg, "&");
            break;
        case 2:
            P.setTextEffect(0, PA_SCROLL_DOWN,
PA_SCROLL_LEFT);
            display++;
            dtostrf(humidity, 3, 0, szMesg);
            strcat(szMesg, "%UR");
            break;
        case 3:
            P.setFont(0, numeric7Seg);
            P.setTextEffect(0, PA_PRINT, PA_NO_EFFECT);
            P.setPause(0, 0);
            if ((millis() - lastTime) >= 1000)
            {
                lastTime = millis();
                getTime(szMesg, flasher);
                flasher = !flasher;
            }
            if ((seconds == 00) && (seconds <= 30)) {
                display++;
                P.setTextEffect(0, PA_PRINT,
PA_WIPE_CURSOR);
            }
            break;
        case 4:
            P.setFont(0, nullptr);
            P.setTextEffect(0, PA_SCROLL_LEFT,
PA_SCROLL_LEFT);
            display++;
            dow2str(wday, szMesg, MAX_MESG);
            break;
        default:
            P.setTextEffect(0, PA_SCROLL_LEFT,
PA_SCROLL_LEFT);
            display = 0;
            getDate(szMesg);
            break;
    }
    P.displayReset(0);
}

```



GAMBAR 5.
PROGRAM ARDUINO IDE

Berikut tabel hasil simulasi:

TABEL 1.
Hasil Simulasi *Real Time*

No	Ujicoba	Suhu (°C)	Kelembaban (%)	Waktu
1.	Tampilan awal	25.5	60.5	08:45:00 AM
2.	Scroll suhu Celsius	25.5	60.5	08:45:15 AM
3.	Scroll suhu Fahrenheit	77.9	60.5	08:45:30 AM
4.	Scroll kelembaban	77.9	60.5	08:45:45 AM
5.	Tampilan jam digital	77.9	60.5	08:46:00 AM
6.	Scroll hari	77.9	60.5	08:46:15 AM
7.	Kembali ke tampilan tanggal	77.9	60.5	08:46:30 AM

1. Prototipe sistem monitoring lingkungan berbasis MD_Parola, LED Matrix, sensor DHT22, dan RTC DS1307 telah berhasil dikembangkan. Dalam pengujian fungsional, sistem mampu menampilkan data suhu, kelembaban, dan waktu dengan jelas dan efektif melalui antarmuka MD_Parola dan LED Matrix. Integrasi sensor DHT22 memberikan hasil pengukuran yang akurat, dengan respons yang cepat terhadap perubahan kondisi lingkungan.
2. Hasil kalibrasi sensor DHT22 menunjukkan tingkat akurasi yang tinggi dalam mengukur suhu dan kelembaban. Sensor memberikan respons yang cepat dan stabil, mendukung ketepatan pengukuran dalam berbagai kondisi lingkungan.
3. RTC DS1307 memberikan waktu yang sinkron dengan baik, memastikan data waktu yang ditampilkan pada LED Matrix tetap akurat dan sesuai dengan waktu aktual.
4. Seluruh fungsi sistem, termasuk visualisasi data dan pengukuran suhu-kelembaban, berjalan sesuai dengan spesifikasi yang telah dirancang. Pengujian performa menunjukkan respons waktu yang cepat dan efisiensi energi yang optimal, tanpa adanya keterlambatan signifikan dalam penyajian data visual.
5. Hasil dari evaluasi pengguna menunjukkan

bahwa antarmuka pengguna sistem dianggap intuitif dan mudah digunakan. Pengguna mengakui efektivitas visualisasi data yang membantu dalam pemahaman informasi mengenai kondisi lingkungan.

6. Data hasil pengukuran suhu dan kelembaban tampil dengan akurat dan sesuai dengan kondisi lingkungan yang diharapkan.

IV. KESIMPULAN

Berdasarkan uji coba dan pengujian sistem monitoring lingkungan yang menggunakan MD_Parola, LED Matrix, sensor DHT22, dan RTC DS1307, diperoleh kesimpulan sebagai berikut:

1. Prototipe sistem berhasil dikembangkan dengan sukses, menunjukkan fungsi visualisasi data suhu, kelembaban, dan waktu dengan baik melalui antarmuka MD_Parola dan LED Matrix.
2. Sensor DHT22 menunjukkan kinerja yang baik dalam mengukur suhu dan kelembaban. Respons yang cepat dan akurasi tinggi mendukung ketepatan pengukuran dalam berbagai kondisi lingkungan.
3. RTC DS1307 menyediakan waktu yang akurat, memastikan tampilan waktu pada LED Matrix selalu sesuai dengan waktu aktual.
4. Seluruh fungsi sistem berjalan sesuai dengan spesifikasi yang dirancang. Pengujian performa menunjukkan respons waktu yang cepat dan efisiensi energi yang baik.
5. Data hasil pengukuran suhu dan kelembaban tampil dengan akurat dan sesuai dengan kondisi lingkungan yang diharapkan. Korelasi positif antara data sensor dan kondisi aktual terkonfirmasi melalui analisis statistik.

Sistem monitoring lingkungan ini memenuhi tujuan penelitian dan spesifikasi yang telah ditetapkan. Keberhasilan pengembangan menunjukkan potensi implementasi sistem serupa dalam pemantauan lingkungan. Dengan kesimpulan ini, sistem monitoring lingkungan berbasis MD_Parola, LED Matrix, sensor DHT22, dan RTC DS1307 dapat dianggap sebagai solusi yang efektif dalam memantau dan menyajikan informasi mengenai kondisi lingkungan dengan baik.



V. SARAN

Untuk pengembangan lebih lanjut, disarankan untuk:

1. Mengeksplorasi integrasi dengan platform pengumpulan dan analisis data untuk pemantauan jarak jauh.
2. Menambahkan sensor tambahan untuk mengukur parameter lingkungan lainnya, seperti kualitas udara atau kebisingan.

I2C Real-time Clock,” *J. Tek. Mesin Mech. Xplore*, vol. 4, no. 1, pp. 17–26, 2023.

DAFTAR PUSTAKA

- [1] R. Nazzla, T. Hestirianoto, and S. Pujiyati, “Prototype Testing of Automatic Vessel Tracking System using Web-Based Visualization,” *PELAGICUS*, vol. 3, no. 2, pp. 61–75, 2023.
- [2] F. Sudarto and M. Fazri, “Prototipe Pengontrolan Running Text Menggunakan Voice dan Arduino Uno Via Smartphone Android,” *Creat. Commun. Innov. Technol. J.*, vol. 10, no. 1, pp. 75–82.
- [3] R. Ananda, M. Amin, and N. Manurung, “WORKSHOP PELATIHAN PEMBUATAN TULISAN BERJALAN DI LED P10 DENGAN MENGGUNAKAN HD-W00 BAGI SISWA/I JURUSAN REKAYASA PERANGKAT LUNAK DI SMK KARYA UTAMA,” *J. Pemberdaya. Sos. dan Teknol. Masy.*, vol. 3, no. 1, pp. 31–35, 2023.
- [4] Z. Akbar, S. W. Sidehabi, and T. Aulani, “Temperature and Humidity Monitoring System on Bread Poofers Using Arduino IoT Cloud,” *JEAT J. Electr. Autom. Technol.*, vol. 2, no. 1, pp. 26–32, 2023.
- [5] P. Yuliantoro, S. Romadhona, and D. O. Ayuninda, “Monitoring Power Consumption On LED Using LED Matrix P10 Panel,” *Power (W)*, vol. 41, p. 40, 2023.
- [6] S. Nurrahmi, N. Miseldi, and S. H. Syamsu, “RANCANG BANGUN SISTEM PENYIRAMAN OTOMATIS PADA GREEN HOUSE TANAMAN ANGGREK MENGGUNAKAN SENSOR DHT22,” *JPF (Jurnal Pendidik. Fis. Univ. Islam Negeri Alauddin Makassar)*, vol. 11, no. 1, pp. 33–43, 2023.
- [7] A. H. B. I. N. KAHARUDDIN, “SMART HEALTHY CHAIR WITH IOT,” 2023.
- [8] R. A. Dinda, S. Sadrina, and M. Mursyidin, “The High Accurate Automatic School Bell Controller Based On Arduino Uno DS1307