

RANCANG BANGUN SISTEM DAFTAR HADIR BERBASIS FINGERPRINT DENGAN ARDUINO UNO

Kadek Laksmi Damayanthi¹, I Made Agus Widiana Putra², I Putu Buda Suyasa³

^{1,2,2}Program Studi Sistem Informasi, Fakultas Sains dan Teknologi, Universitas Tabanan
Jl. Wagimin No. 8, Kediri, Tabanan, Indonesia

e-mail: deklaksmi2016@gmail.com¹, imadeagusclass@gmail.com², budasuyasa@gmail.com³

Received : Month, Year	Accepted : Month, Year	Published : Month, Year
------------------------	------------------------	-------------------------

Abstract

Recording attendance is an important factor in the management of human resources. Information regarding attendance that can determine achievement, productivity or progress of the agency. So we need a smart attendance system that can work automatically, with a design method that makes a smart attendance system using fingerprint technology and using a control system. Data in the form of unique strokes at the fingertips of every human being is used as data. When this fingerprint is attached to the attendance device, the data is automatically entered into the Arduino. The results of this study indicate that Arduino Uno can be used as an attendance system controller using a fingerprint sensor. The FMP10 fingerprint sensor can read as many as 127 fingerprint data. This design was made to facilitate attendance automatically by attaching fingerprints to the fingerprint sensor and data will enter automatically into the Arduino without having to use stationery, pens, attendance books, and enter attendance data manually, so the impact of this design is good for use. because it is quite easy to use compared to manually.

Keywords: attendance, arduino uno, fingerprint, fingerprint sensor

Abstrak

Pencatatan absensi merupakan salah satu faktor penting dalam pengelolaan dalam sumber daya manusia. Informasi yang mengenai kehadiran yang dapat menentukan prestasi, produktivitas atau kemajuan instansi. Sehingga diperlukan sebuah sistem absensi cerdas yang dapat bekerja secara otomatis, dengan metode rancangan yang membuat sistem absensi cerdas dengan menggunakan teknologi fingerprint dan sistem pengendalinya menggunakan. Data berupa guratan-guratan unik yang ada di ujung jari setiap manusia dimanfaatkan sebagai data. Saat sidik jari ini ditempelkan pada alat pencatat kehadiran, datanya secara otomatis masuk kedalam Arduino. Hasil dari penelitian ini menunjukkan bahwa arduino uno dapat digunakan sebagai controller system absesnsi menggunakan sensor fingerprint. Sensor fingerprint FPM10 dapat membaca sebanyak 127 data sidik jari. Perancangan ini dibuat karena untuk mempermudah absensi secara otomatis dengan cara menempelkan sidik jari pada sensor fingerprint dan data akan masuk secara otomatis ke Arduino tanpa harus menggunakan alat tulis pulpen, buku absensi, dan memasukkan data absensi secara manual, sehingga dampak dari perancangan ini baik untuk digunakan karena cukup mudah cara penggunaannya dibanding secara manual.

Kata Kunci: absensi, arduino uno, fingerprint, sensor fingerprint

1. PENDAHULUAN

Absensi adalah sebuah kegiatan pengambilan data guna mengetahui jumlah kehadiran pada suatu acara. Setiap kegiatan yang membutuhkan informasi mengenai peserta tentu akan melakukan absensi. Hal ini juga terjadi pada proses belajar mengajar. Kegunaan absensi ini terjadi pada pihak pelajar dan pihak pengada proses belajar mengajar. Salah satu kegunaan absensi ini kepada pihak pelajar antara lain adalah dalam perhitungan kemungkinan pelajar untuk mengikuti ujian dan salah satu kegunaan informasi absensi ini kepada pihak pengada kegiatan belajar mengajar antara lain untuk melakukan evaluasi kepada kepuasan pelajar terhadap suatu mata pelajaran dan pembuatan tolak ukur ke depan guna pemberian ilmu yang lebih baik. Pengambilan data absensi yang dilakukan secara manual memiliki banyak kekurangan, seperti data yang tidak valid ketika data yang masuk salah, hilang atau rusaknya data yang ada, kurangnya efisiensi dan efektifitas pada pengolahan data.

Perkembangan teknologi di dunia semakin pesat, khususnya kemajuan di bidang teknologi informasi terutama pada teknologi mobile dan internet. Penggunaan aplikasi mobile dan web dikatakan lebih efektif dan efisien karena adanya kemudahan dalam pengaksesan dan pengambilan informasi. Perkembangan teknologi ini juga mempengaruhi cara input data sebagai validasi menggantikan kode password yang ada. Seperti penggunaan teknologi fingerprint atau sidik jari. Sistem input ini berkembang karena minimnya kesalahan input atau kecurangan pada input data. Tak hanya perkembangan teknologi informasi namun perkembangan aplikasi juga pesat. Salah satu perkembangan pada aplikasi teknologi informasi adalah sistem informasi.

2. METODE PENELITIAN

Metode penelitian yang digunakan ialah metode penelitian eksperimen yang dimana penelitian yang dilakukan untuk mengetahui akibat yang ditimbulkan dari suatu perlakuan yang diberikan secara sengaja oleh peneliti. Pada prosedur penelitian akan dilakukan beberapa langkah yaitu pengujian untuk mengetahui cara perancangan alat, prinsip kerja sebuah alat dan keefektifan alat tersebut.

Sistem informasi adalah sebuah sistem informasi yang mempunyai fungsi mengumpulkan, memproses, menyimpan, menganalisis, dan menyebarkan informasi untuk tujuan yang spesifik (Turban,1999).

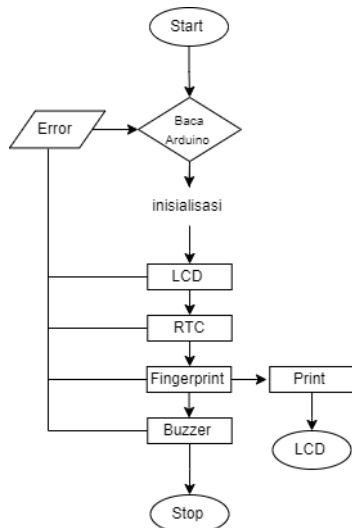
Pengertian lainnya adalah kumpulan perangkat keras dan lunak yang dirancang untuk mentransformasikan data ke dalam bentuk informasi yang berguna (HopWood). Hampir seluruh kegiatan berskala sedang hingga besar memiliki sebuah sistem informasi guna membantu pekerjaan. Sistem informasi sangat membantu dalam kegiatan hingga sering digunakan sebagai pengambilan keputusan.

Universitas Tabanan merupakan salah satu perguruan yang ada di Tabanan. Namun sangat disayangkan, di tengah perkembangan zaman yang semakin maju ini Universitas Tabanan masih memberlakukan system daftar hadir (absensi) secara manual. Dilihat dari hal tersebut sering terjadi beberapa kecurangan atau ketidakdisiplinan mahasiswa maupun dosen dalam proses pembelajaran. Seperti jam hadir yang tidak tepat waktu, mahasiswa dengan jumlah berbeda dengan yang tercatat hadir di kelas, hingga beberapa kecurangan yang lainnya.

Dalam penelitian ini telah dikembangkan suatu model absensi finger print berbasis arduino. Alat ini dibuat guna menghindari kecurangan atau manipulasi data kehadiran oleh dosen maupun mahasiswa. Hal ini dikarenakan sidik jari merupakan salah satu bagian unik dari manusia, sidik jari Dosen yang satu akan berbeda dengan sidik jari Dosen yang lainnya. Dalam rekapitulasi data absensi, tujuan utamanya adalah memperoleh hasil analisa dari sebuah alat yang dibuat Dosen, dengan begitu Dosen bisa mengetahui kehadiran secara cepat.

2.1. Rancangan Penelitian

Rancangan penelitian Perancangan Sistem daftar hadir berbasis fingerprint dengan Arduino Uno dapat digambarkan dengan sebagai berikut:



Gambar 1. Skema perancangan sistem daftar hadir berbasis fingerprint dengan Arduino Uno

2.2. Prosedur Penelitian

Adapun langkah-langkah alat ini bekerja pada sensor fingerprint adalah :

1. Arduino berfungsi untuk memerintahkan komponen-komponen mikrokontroler
2. Sensor *fingerprint* berfungsi untuk mendeteksi sidik jari
3. Setelah sidik jari ditempelkan pada sensor fingerprint, maka selanjutnya sensor akan mengirim data ke LCD untuk menampilkan inisial mahasiswa dan waktu absensi
4. RTC berfungsi sebagai pengatur waktu pada system
5. Buzzer berfungsi sebagai pemberi sinyal berbentuk suara berbarengan dengan hasil tampilan data di LCD
6. Selesai

Gambaran umum langkah-langkah kerja dalam penelitian ini dapat dilihat dalam diagram alur perancangan pada gambar dibawah:

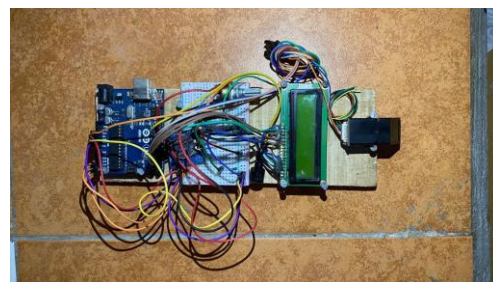


Gambar 2. Diagram alur perancangan penelitian

3. HASIL DAN PEMBAHASAN

3.1 Deskripsi Data

Data yang dihasilkan dari pengujian ini, Merancang system daftar hadir berbasis fingerprint dengan Arduino Uno dapat dilihat pada gambar di bawah ini:



Gambar 3. Rangkaian fingerprint arduino uno

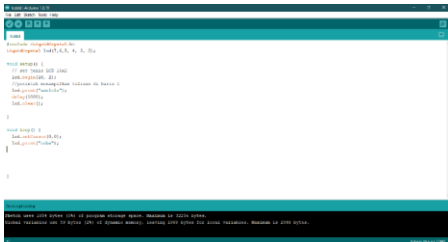
3.2 Pembahasan

Pengujian dan Analisa Perangkat Keras
 Pengujian dan analisa perangkat keras bertujuan untuk menguji dan menganalisa fungsi dari perangkat-perangkat yang digunakan oleh sistem serta memastikan semua perangkat yang akan digunakan telah siap beroperasi. Pengujian dan analisa perangkat keras ini terdiri dari :

1. Pengujian LCD
2. Pengujian RTC
3. Pengujian sensor Fingerprint

3.3.1. Pengujian LCD

Tahap awal pengujian ini adalah dengan melakukan proses sketch program pada jendela koding program Arduino seperti pada gambar di bawah:



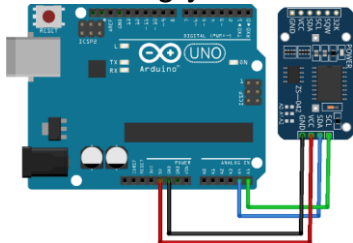
Gambar 4. Tampilan sketch LCD

Setelah proses koding berhasil dan tidak ada error maka pada lcd akan ditampilkan hasil seperti dibawah:



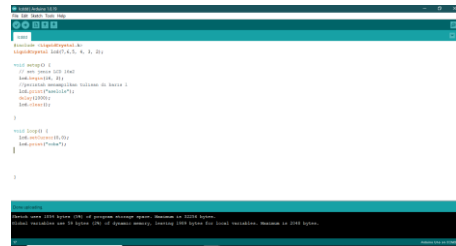
Gambar 5. Tampilan hasil test LCD

3.2.2. Pengujian RTC

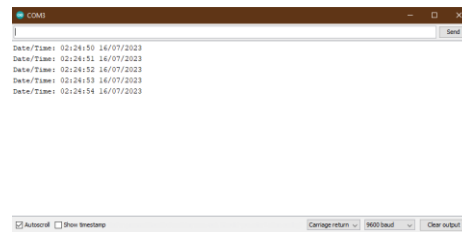


Gambar 6. Tampilan skema pemasangan RTC dengan arduino

Pada pengujian chip RTC, uji coba dilakukan dengan metode yang sama dengan uji coba pada LCD . sketch koding dimasukan pada jendela koding program seperti pada gambar dibawah :



Gambar 7. Tampilan sketch RTC



Gambar 8. Tampilan serial code pada arduino

Lalu ketika sudah tidak ada error, waktu pada RTC bisa disetting melalui serial monitor pada Arduino dengan mengisi format waktu yang dibutuhkan. Jika tampilan RTC sudah tepat maka RTC sudah berhasil dalam tahap uji coba

3.2.3. Pengujian Fingerprint

Pengujian ini dilakukan dengan tujuan untuk mengetahui kemampuan modul fingerprint dalam scanning sidik jari. Fingerprint harus bisa membedakan sidik jari yang sudah terdaftar di ENROLL serial monitor dan sidik jari yang belum terdaftar.

Table 1: Tabel sketch/source code ENROLL

#include "Adafruit_Fingerprint.h"
#include "SoftwareSerial.h"
SoftwareSerial mySerial(8, 9);
Adafruit_Fingerprint finger = Adafruit_Fingerprint(&mySerial);
uint8_t id;
void setup() {
Serial.begin(9600);
finger.begin(57600);
if (finger.verifyPassword()) {
Serial.println("FingerPrint Sensor Ditemukan!");
} else {
Serial.println("FingerPrint Sensor Tidak Ditemukan! :(");

```

while (1) {
    delay(1);
}
}

void loop() {
    ENROLL();
}

//ENROLL//
void ENROLL() {
    Serial.println("Ready to enroll a
fingerprint!");
    Serial.println("Please type in the
ID # (from 1 to 127) you want to
save this finger as...");
    id = readnumber();

    if (id == 0) { // ID #0 not
allowed, try again!
        return;
    }

    Serial.print("Enrolling ID #");
    Serial.println(id);

    while (! getFingerprintEnroll()
);
}

//PROSES ENROLL//

uint8_t readnumber(void) {
    uint8_t num = 0;

    while (num == 0) {
        while (! Serial.available());
        num = Serial.parseInt();
    }

    return num;
}

//RETURN NUM//

```

```

uint8_t getFingerprintEnroll() {
    int p = -1;

    Serial.print("Waiting for valid
finger to enroll as #");
    Serial.println(id);
    while (p != FINGERPRINT_OK) {
        p = finger.getImage();

        switch (p) {
            case FINGERPRINT_OK:
                Serial.println("Image
taken");
                break;

            case FINGERPRINT_NOFINGER:
                Serial.println(".");
                break;

            case
FINGERPRINT_PACKETRECIEVEERR:
                Serial.println("Communication
error");
                break;

            case FINGERPRINT_IMAGEFAIL:
                Serial.println("Imaging
error");
                break;

            default:
                Serial.println("Unknown
error");
                break;
        }
    }

    // OK success!

    p = finger.image2Tz(1);
    switch (p) {
        case FINGERPRINT_OK:
            Serial.println("Image
converted");
            break;

        case FINGERPRINT_IMAGEMESS:
            Serial.println("Image too
messy");
            return p;

        case
FINGERPRINT_PACKETRECIEVEERR:
            Serial.println("Communication
error");
            return p;
    }
}

```

```

case FINGERPRINT_FEATUREFAIL:
    Serial.println("Could not find
fingerprint features");
    return p;
case FINGERPRINT_INVALIDIMAGE:
    Serial.println("Could not find
fingerprint features");
    return p;
default:
    Serial.println("Unknown
error");
    return p;
}

Serial.println("Remove finger");
delay(2000);
p = 0;
while (p != FINGERPRINT_NOFINGER)
{
    p = finger.getImage();
}

Serial.print("ID ");
Serial.println(id);
p = -1;
Serial.println("Place same finger
again");
while (p != FINGERPRINT_OK) {
    p = finger.getImage();
    switch (p) {
        case FINGERPRINT_OK:
            Serial.println("Image
taken");
            break;
        case FINGERPRINT_NOFINGER:
            Serial.print(".");
            break;
        case
FINGERPRINT_PACKETRECIEVEERR:
            Serial.println("Communication
error");
            break;
        case FINGERPRINT_IMAGEFAIL:
            Serial.println("Imaging
error");
            break;
        default:
            Serial.println("Unknown
error");

```

```

        break;
    }
}

// OK success!

p = finger.image2Tz(2);
switch (p) {
    case FINGERPRINT_OK:
        Serial.println("Image
converted");
        break;
    case FINGERPRINT_IMAGEMESS:
        Serial.println("Image too
messy");
        return p;
    case
FINGERPRINT_PACKETRECIEVEERR:
        Serial.println("Communication
error");
        return p;
    case FINGERPRINT_FEATUREFAIL:
        Serial.println("Could not find
fingerprint features");
        return p;
    case FINGERPRINT_INVALIDIMAGE:
        Serial.println("Could not find
fingerprint features");
        return p;
    default:
        Serial.println("Unknown
error");
        return p;
}

// OK converted!
Serial.print("Creating model for
#"); Serial.println(id);

p = finger.createModel();
if (p == FINGERPRINT_OK) {
    Serial.println("Prints
matched!");
} else if (p ==
FINGERPRINT_PACKETRECIEVEERR) {
    Serial.println("Communication
error");
    return p;
} else if (p ==

```

```

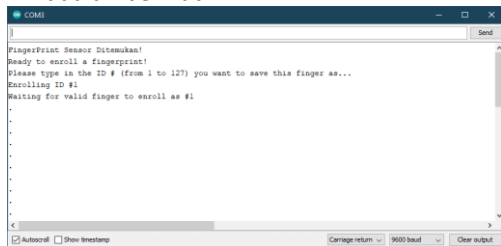
FINGERPRINT_ENROLLMISMATCH) {
  Serial.println("Fingerprints did not match");
  return p;
} else {
  Serial.println("Unkown error");
  return p;
}

Serial.print("ID ");
Serial.println(id);
p = finger.storeModel(id);
if (p == FINGERPRINT_OK) {
  Serial.println("Stored!");
} else if (p == FINGERPRINT_PACKETRECEIVEERR) {
  Serial.println("Communication error");
  return p;
} else if (p == FINGERPRINT_BADLOCATION) {
  Serial.println("Could not store in that location");
  return p;
} else if (p == FINGERPRINT_FLASHERR) {
  Serial.println("Error writing to flash");
  return p;
} else {
  Serial.println("Unkown error");
  return p;
}
}

//ENROLL END//

```

Maka tampilan pada serial monitor adalah berikut :



Gambar 9. Tampilan serial monitor pada program ENROLL

Dalam pengujian ini sidik jari yang terdaftar adalah sepuluh jari tangan sebagai ID #1 sampai ID #10, sedangkan untuk pembandingnya adalah sidik jari penguji lain yang mana belum terdaftar.

Setelah proses pendaftaran sidik jari selesai, mulai memasuki program utama yaitu program absensi dengan sidik jari Sketch/source code program utama :

Table 2: Tabel sketch/source code program utama

```

#include <Adafruit_Fingerprint.h>
#include <SoftwareSerial.h>
#include <LiquidCrystal.h>
#include <EEPROM.h>
#include <Wire.h>
#include <SPI.h>
#define DS1307_ADDRESS 0x68
byte zero = 0x00;

LiquidCrystal lcd(7,6,5, 4, 3, 2);

byte second ,minute,hour, weekDay;
byte monthDay,month,year;

uint8_t id;
uint8_t getFingerprintEnroll();

// pin #8 is IN from sensor
// pin #9 is OUT from arduino
SoftwareSerial mySerial(8, 9);
Adafruit_Fingerprint finger =
Adafruit_Fingerprint(&mySerial);

void setup() {
  pinMode(10, OUTPUT);
  digitalWrite(10, LOW);
  delay(500);

  Wire.begin();
  /*
  // Seting RTC
  second=0;
  minute=50;
  hour=12;

```

```

monthDay=15;
month=10;
year=16;
setingRTC();
delay(100);
*/
Serial.begin(9600);

SPI.begin();
Serial.println("CLEARSHEET");
Serial.println("LABEL,Data");

lcd.begin(16, 2);
lcd.clear();
lcd.print("Absen Mahasiswa");
lcd.setCursor(0,3);
lcd.print(" Sidik Jari");
delay(2000);
lcd.clear();
lcd.print("Test Koneksi");

// set the data rate for the
sensor serial port
finger.begin(57600);

if (finger.verifyPassword()) {
  lcd.setCursor(0,1);
  lcd.print("OK !");
} else {
  lcd.setCursor(0,1);
  lcd.print("ERROR !");
  while (1);
}
delay(2000);
}

//program utama
void loop(){
// bacaRTC();

Serial.print("DATA");
Serial.print("absensi");

```

```

Serial.println(" ");
delay(1000);

lcd.setCursor(0, 0);
lcd.print("Absen Mahasiswa");
lcd.setCursor(0, 1);
lcd.print("Letakkan Jari!");
getFingerprintIDez();
 kirimSerial();
 delay(50);
}

int getFingerprintIDez() {
  uint8_t p = finger.getImage();
  if (p != FINGERPRINT_OK) return -1;

  p = finger.image2Tz();
  if (p != FINGERPRINT_OK) return -1;

  p = finger.fingerFastSearch();
  if (p == FINGERPRINT_NOTFOUND){
    lcd.clear();
    lcd.print(" Akses");
    lcd.setCursor(0,1);
    lcd.print(" Tidak Diterima");
    digitalWrite(10, HIGH);
    delay(500);
    digitalWrite(10, LOW);
    delay(50);

    return p;
  }

  if (p != FINGERPRINT_OK) return -1;
  // found a match!
  digitalWrite(10, HIGH);
  delay(50);
  digitalWrite(10, LOW);
  delay(50);

  bacaRTC();
  lcd.clear();
  lcd.print("Mahasiswa ");

```



```

lcd.print(finger.fingerID);
lcd.setCursor(0,1);
lcd.print(hour);
lcd.print(":");
lcd.print(minute);
lcd.print(":");
lcd.print(second);
lcd.print(" ");
lcd.print(monthDay);
lcd.print("/");
lcd.print(month);
lcd.print(" ");

//-----simpan data
id=finger.fingerID;
byte addr = (monthDay*10)+id;
EEPROM.write(addr, 0);
delay(4000);
lcd.clear();
return finger.fingerID;
}

/*****
****
* peta penyimpanan di ENROLL
* tanggal disimpan pada alamat 0
* -----
* tgl | Alamat ENROLL
mahasiswa
* | 1 2 3 4 5 6 7
8 9 10
* -----
* 1 | 11 12 13 14 15 16 17
18 19 20
* 2 | 21 22 23 24 25 26 27
28 29 30
* 3 | 31 32 33 34 35 36 37
38 39 40
*
* 30 | 311 312 313 314 315 316
317 318 319 320
* -----
* Rumus Penyimpanan ENROLL =
* alamatSimpan = (tgl * 10)+id
*
* kode absensi:

```

```

* * 0 = Hadir
* * 255 = Alpha
*/

//-----RTC
Area
byte decToBcd(byte val){
    return ( (val/10*16) + (val%10) );
}

byte bcdToDec(byte val) {
    return ( (val/16*10) + (val%16) );
}

void bacaRTC(){

Wire.beginTransmission(DS1307_ADDRESS);
Wire.write(zero);
Wire.endTransmission();
Wire.requestFrom(DS1307_ADDRESS,
7);
second = bcdToDec(Wire.read());
minute = bcdToDec(Wire.read());
hour = bcdToDec(Wire.read() &
0b111111);
weekDay = bcdToDec(Wire.read());
monthDay = bcdToDec(Wire.read());
month = bcdToDec(Wire.read());
year = bcdToDec(Wire.read());
}

void setingRTC(){

Wire.beginTransmission(DS1307_ADDRESS);
Wire.write(zero); //stop RTC
Wire.write(decToBcd(second));
Wire.write(decToBcd(minute));
Wire.write(decToBcd(hour));
Wire.write(decToBcd(weekDay));
Wire.write(decToBcd(monthDay));
Wire.write(decToBcd(month));
Wire.write(decToBcd(year));
Wire.write(zero); //start
Wire.endTransmission();

```

```

}

void kirimSerial(){
  if(Serial.available()){
    char str;
    str=Serial.read();
    if (str=='R'){
      /*
      * -----
      -----
      * tgl |          Absensi
      Mahasiswa
      *      | 1  2  3  4  5
      6  7  8  9  10
      * -----
      -----
      * 1 | H  H  -  -  H
      H  H  H  H  H
      * 2 | H  H  H  H  -
      H  H  H  H  H
      * 3 |
      *
      * 30 |
      * -----
      -----
      * H = HADIR
      * - = ALPHA
      */

      for(byte i=0; i<60;i++){
        Serial.print('-');
      }
      Serial.println('-');

      Serial.println(" tgl |
      Absens Mahasiswa");
      Serial.print ("      | ");

      for(byte i=1; i<11;i++){
        Serial.print(i);
        Serial.print(" ");
      }
      Serial.println(" ");

      for(byte i=0; i<60;i++){
        Serial.print('-');
      }
      Serial.println('-');

```

```

byte value;
byte address;
byte j=0;
String absensi="";
for(byte tgl=1; tgl<31;
tgl++){
  Serial.print(tgl);
  Serial.print(" |");
  j=tgl*10;
  for(byte i=1; i<11; i++){
    address=i+j;
    value =
    EEPROM.read(address);
    if (value==0) absensi="H";
    else absensi="-";
    Serial.print(" "+absensi+"
");
  }
  Serial.println(" ");
}
str=' ';
}
else if(str=='D'){
  Serial.println("Hapus Data");
  for(word i=0; i<310; i++){
    EEPROM.write(i, 255);
    Serial.println(".");
    delay(5);
  }
  Serial.println("Hapus Data
Sukses!");
}
}
}

```

Jika program sudah berjalan maka tampilan pertama pada LCD adalah :



Gambar 10. Tampilan awal pada LCD

Selanjutnya tes koneksi antara sensor fingerprint dengan Arduino , jika sukses maka tampilannya seperti di bawah :



Gambar 11. Tampilan test koneksi sukses pada LCD

Letakan sidik jari pada sensor finger print, jika sidik jari sudah terdaftar maka tampilannya akan seperti di bawah :



Gambar 12. Tampilan sukses pada LCD

Pada tampilan ini baris pertama menampilkan data mahasiswa , lalu baris kedua menampilkan waktu absensi

Jika sidik jari yang di tempelkan belum atau tidak sesuai dengan yang di daftarkan pada program sebelumnya maka tampilannya akan seperti di bawah ini :



Gambar 13. Tampilan akses ditolak pada LCD

Selanjutnya tampilan lcd kembali ke tampilan awal.

Jika ingin melihat daftar mahasiswa, buka serial monitor dengan baudrate 9600 + carriage return, lalu ketikan R kemudian kirim. Lalu serial monitor akan menampilkan data absensi, H menyatakan hadir, sedangkan (-) menyatakan ketidakhadiran.

3.3. Data pengujian

Hasil dari pengujian rangkaian system daftar hadir berbasis fingerprint dengan Arduino Uno dapat dilihat di table di bawah ini :

Table 3: Tabel data sidik jari mahasiswa

tg	alamat eprom mahasiswa									
	1	2	3	4	5	6	7	8	9	10
1	11	12	13	14	15	16	17	18	19	20
2	21	22	23	24	25	26	27	28	29	30
3	31	32	33	34	35	36	37	38	39	40

Table 4: Tabel absensi mahasiswa

tgl	absensi mahasiswa									
	1	2	3	4	5	6	7	8	9	10

1	H	H	-	-	H	H	H	H	H	H
2	H	H	H	H	-	H	H	H	H	H
3	-	-	H	-	H	-	H	H	H	H

4. KESIMPULAN

Dalam perancangan dan pembuatan system daftar hadir berbasis fingerprint dengan Arduino Uno bekerja dengan baik. Adapun cara alat ini bekerja adalah, ketika pertama sidik jari di daftarkan di system enrol yaitu sisem yang berfungsi untuk menambah atau mendaftarkan sidik jari pengguna (mahasiswa) setelah sudah didaftarkan lalu pada system utama sidik jari di tempelkan sebagai input pada sensor fingerprint, lalu oleh Arduino data akan diteruskan lalu Arduino akan memberikan output ke LCD dan Buzzer

PERNYATAAN PENGHARGAAN

Penulis mengucapkan terimakasih sebesar-besarnya kepada para dosen pembimbing dan orang tua penulis atas segala bimbingannya dalam proses penelitian ini, sehingga penelitian yang berjudul “ Merancang Sistem Daftar Hadir Berbasis Fingerprint Dengan Arduino Uno” ini dapat terselesaikan dengan baik.

DAFTAR PUSTAKA

Arfah, A. A., & Suwardoyo, U. (2022). Aplikasi absensi karyawan menggunakan geolocation dan finger print berbasis android. *Jurnal Sintaks Logika*, 2(2), 1-8.

Fahmizal, S. T., Mayub, A., Kom, M., Arrofiq, I. M., & Ruciyanti, F. (2022). Mudah Belajar Arduino dengan Pendekatan berbasis Fritzing, Tinkercad dan Proteus. Deepublish.

Firliana, R., & Rhozman, F. (2019). Aplikasi Sistem Informasi Absensi Mahasiswa dan Dosen. *DoubleClick: Journal of Computer and Information Technology*, 2(2), 70-74.

Gandhi, M. A. (2017). Penerapan Absensi Finger Print Dalam Mendisiplinkan Kerja Pegawai Di Sekolah Menengah Kejuruan (Smk) Sekolah Menengah Tekhnik Industri (Smti) Bandar Lampung (Doctoral dissertation, UIN Raden Intan Lampung).

Hardiyansyah, M. V., Kabib, M., & Hudaya, A. Z. (2021). Rancang Bangun Sistem Kontrol Suhu Pada Mesin Oven

Kadir, A. (2017). Pemrograman arduino dan processing. Elex Media Komputindo.

Kamal, F., Winarso, W., & Hidayat, W. W. (2020). Pengaruh Absensi Fingerprint Terhadap Peningkatan Disiplin Kerja

Pegawai Pada Kementerian Komunikasi Dan Informatika. *Jurnal Ilmiah Akuntansi dan Manajemen*, 16(2).

Pratiwi, K. Y., Suprihatin, S., & Atmoko, P. W. (2020). Pengembangan Sistem RFID dan Fingerprint Terintegrasi dengan Sistem Otomasi Layanan di Perpustakaan Universitas Brawijaya. *Jurnal Pustaka Ilmiah*, 6(1), 963-978.

Putra, J., HD, M. A., & Pamungkas, W. W. (2022). SISTEM PENGAMAN PINTU RUMAH MENGGUNAKAN SENSOR RFID RC522 DAN FINGERPRINT BERBASIS INTERNET OF THINGS. *JUPITER: Jurnal Penerapan Ilmu-ilmu Komputer*, 8(2), 14-21.

Safi'i, M. (2018). Perancangan sistem monitoring tegangan output genset menggunakan ethernet shield & sms gateway berbasis arduino uno. *Metik Jurnal*, 2(1), 46-52.

Safii, M., & Vidy, V. (2022). IMPLEMENTASI BILIK RUANGAN DISINFEKTAN MENGGUNAKAN SENSOR ULTRASONIK BERBASIS MIKROKONTROLLER. *Jurnal Informatika Wicida*, 11(1), 21-27.

Wasista, S., & Saraswati, D. A. (2019). 19 Jam Belajar Cepat Arduino: Edisi Revisi. Bumi Aksara.

Yani, A., & Rosyida, S. (2022). Penerapan Sistem Informasi Absensi Karyawan Pada CV. Bintang Bangun Persada Bekasi. *Jurnal Khatulistiwa Informatika*, 10(1), 1-7.

ZAINUDDIN, N. Q. (2020). PENERAPAN MODEL ABSENSI FINGER PRINT BERBASIS ARDUINO. 22.