DOT MATRIX DAN EPROM

Mata Kuliah : Interface, Peripheral, dan Komunikasi

Kode Dosen : AJR

Kelas : D3TK-43-02

Anggota Kelompok :

1. Wahyu Esya Nasution (6702194052)
2. Farhan Ulil Fajri (6702190077)

PROGRAM STUDI D3 TEKNOLOGI KOMPUTER
FAKULTAS ILMU TERAPAN
UNIVERSITAS TELKOM
BANDUNG
2021

A. Tujuan

Maksud dan tujuan dari praktikum ini adalah :

1. Mahasiswa mampu menggunakan pin-pin pada mikrokontroler dalam mengendalikan EEPROM dan dot matrix
2. Mahasiswa mampu menyelesaikan kasus tertentu dengan EEPROM dan dot matrix dalam mikrokontroler.

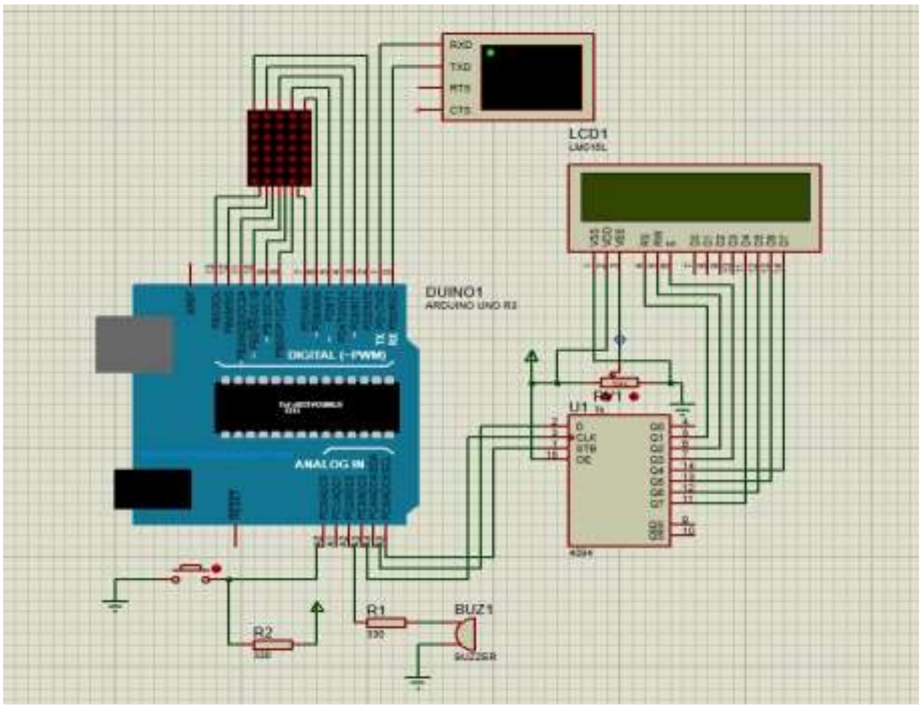
B. Alat dan Bahan

1. Arduino
2. Proteus

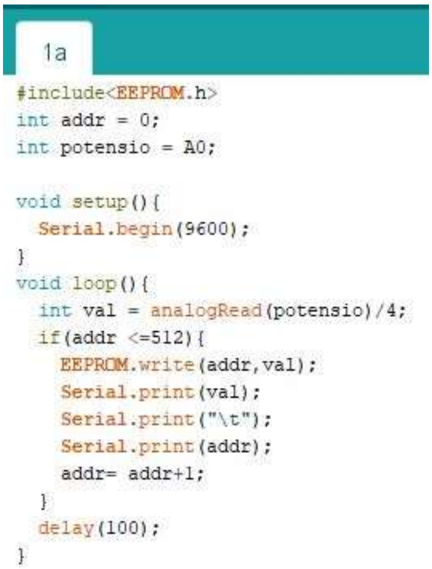
C. Teori dasar

EPROM kependekan dari Erasable Programmable Read Only Memory. EPROM berbeda dengan PROM. EPROM adalah jenis chip memori yang dapat ditulisi program secara elektris. Program atau informasi yang tersimpan di dalam EPROM dapat dihapus bila terkena sinar ultraviolet dan dapat ditulisi kembali. Kesamaannya dengan PROM adalah keduanya merupakan jenis ROM, termasuk memori nonvolatile, data yang tersimpan di dalamnya tidak bisa hilang walaupun komputer dimatikan, tidak membutuhkan daya listrik untuk mempertahankan atau menjaga informasi atau program yang tersimpan di dalamnya.

D. Hasil Percobaan

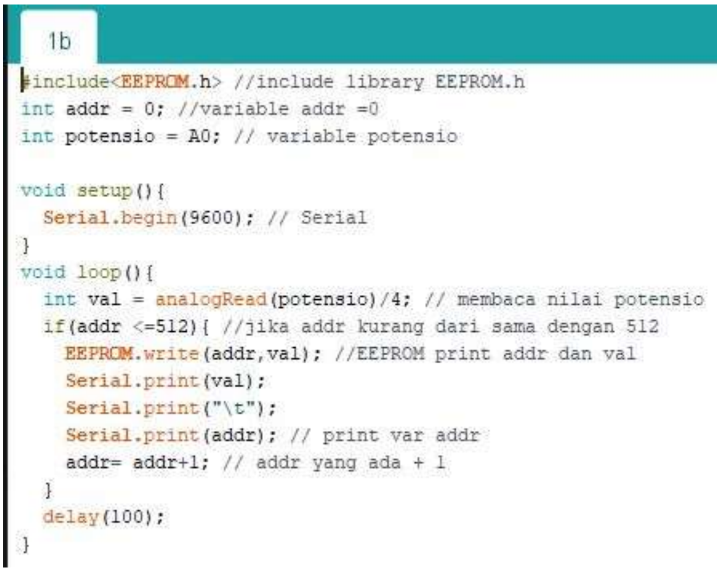


1a.



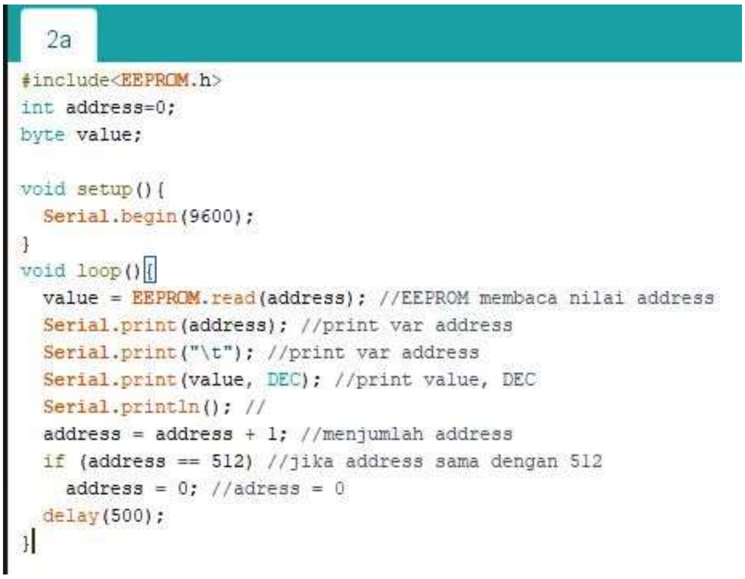
```
#include<EEPROM.h>
int addr = 0;
int potensio = A0;

void setup(){
  Serial.begin(9600);
}
void loop(){
  int val = analogRead(potensio)/4;
  if(addr <=512){
    EEPROM.write(addr,val);
    Serial.print(val);
    Serial.print("\t");
    Serial.print(addr);
    addr= addr+1;
  }
  delay(100);
}
```

1b.



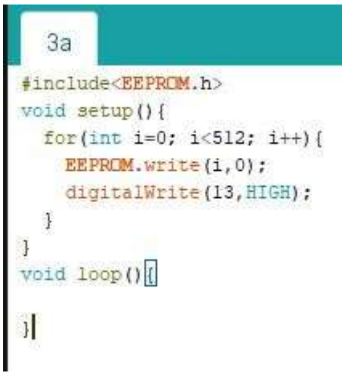
```
#include<EEPROM.h> //include library EEPROM.h
int addr = 0; //variable addr =0
int potensio = A0; // variable potensio

void setup(){
  Serial.begin(9600); // Serial
}
void loop(){
  int val = analogRead(potensio)/4; // membaca nilai potensio
  if(addr <=512){ //jika addr kurang dari sama dengan 512
    EEPROM.write(addr,val); //EEPROM print addr dan val
    Serial.print(val);
    Serial.print("\t");
    Serial.print(addr); // print var addr
    addr= addr+1; // addr yang ada + 1
  }
  delay(100);
}
```

2a.



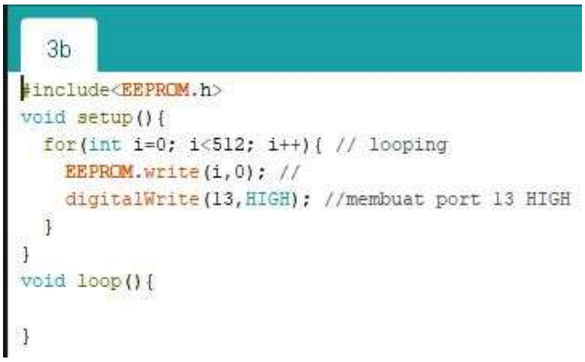
```
#include<EEPROM.h>
int address=0;
byte value;

void setup(){
  Serial.begin(9600);
}
void loop(){
  value = EEPROM.read(address); //EEPROM membaca nilai address
  Serial.print(address); //print var address
  Serial.print("\t"); //print var address
  Serial.print(value, DEC); //print value, DEC
  Serial.println(); //
  address = address + 1; //menjumlah address
  if (address == 512) //jika address sama dengan 512
    address = 0; //adress = 0
  delay(500);
}
```

3a.



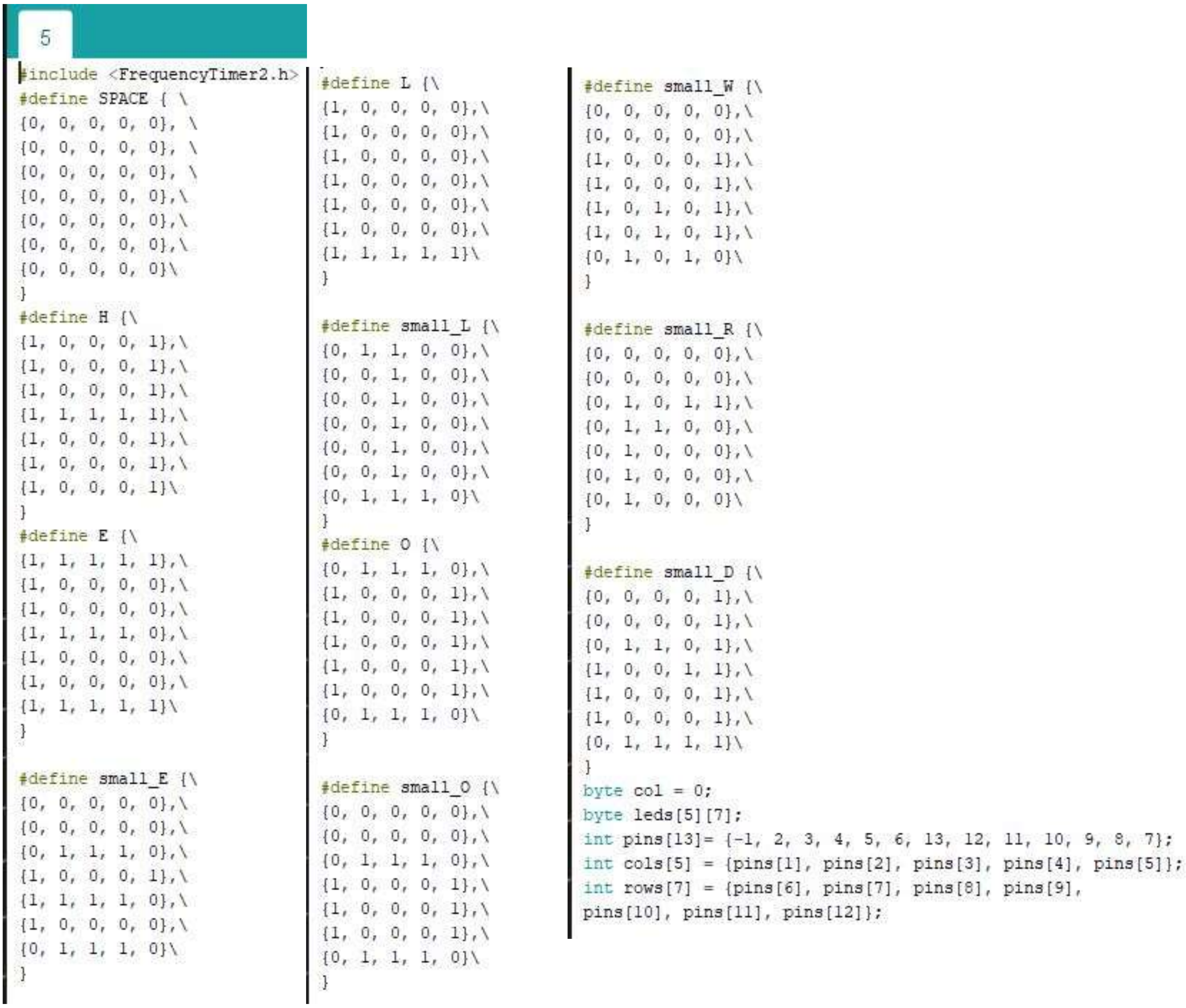
```
#include<EEPROM.h>
void setup(){
  for(int i=0; i<512; i++){
    EEPROM.write(i,0);
    digitalWrite(13,HIGH);
  }
}
void loop(){

}
```

3b.



```
#include<EEPROM.h>
void setup(){
  for(int i=0; i<512; i++){ // looping
    EEPROM.write(i,0); //
    digitalWrite(13,HIGH); //membuat port 13 HIGH
  }
}
void loop(){

}
```

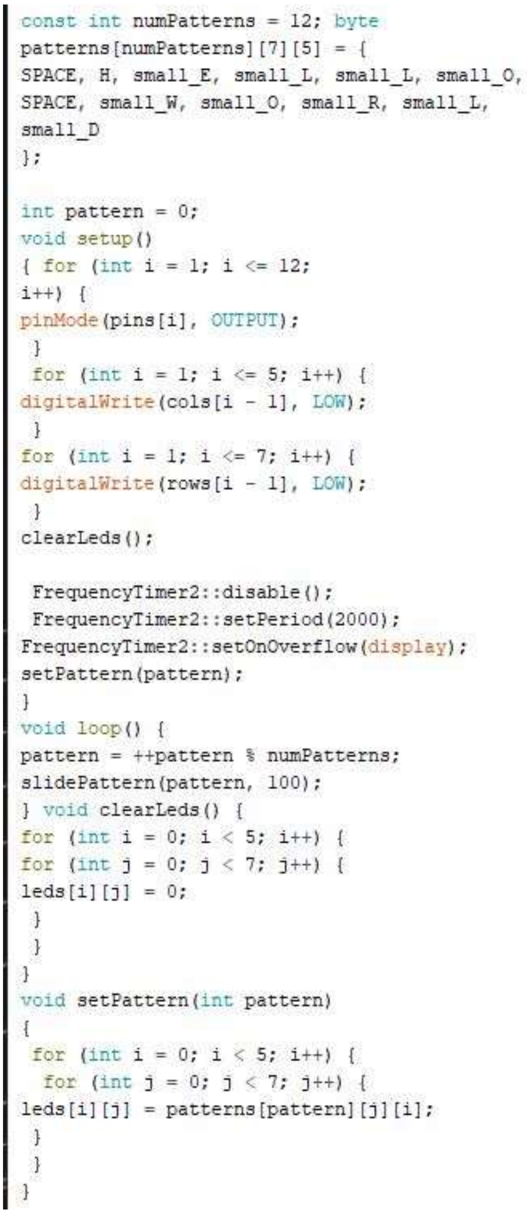
5.



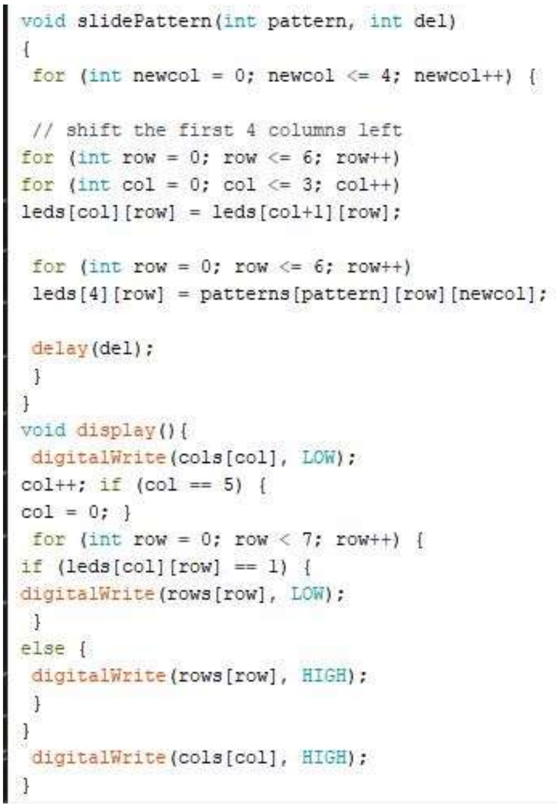
```
#include <FrequencyTimer2.h>
#define SPACE { \
{0, 0, 0, 0, 0}, \
{0, 0, 0, 0, 0}, \
{0, 0, 0, 0, 0}, \
{0, 0, 0, 0, 0},\
{0, 0, 0, 0, 0},\
{0, 0, 0, 0, 0},\
{0, 0, 0, 0, 0}\
}
#define H {\
{1, 0, 0, 0, 1},\
{1, 0, 0, 0, 1},\
{1, 0, 0, 0, 1},\
{1, 1, 1, 1, 1},\
{1, 0, 0, 0, 1},\
{1, 0, 0, 0, 1},\
{1, 0, 0, 0, 1}\
}
#define E {\
{1, 1, 1, 1, 1},\
{1, 0, 0, 0, 0},\
{1, 0, 0, 0, 0},\
{1, 1, 1, 1, 0},\
{1, 0, 0, 0, 0},\
{1, 0, 0, 0, 0},\
{1, 1, 1, 1, 1}\
}

#define small_E {\
{0, 0, 0, 0, 0},\
{0, 0, 0, 0, 0},\
{0, 1, 1, 1, 0},\
{1, 0, 0, 0, 1},\
{1, 1, 1, 1, 0},\
{1, 0, 0, 0, 0},\
{0, 1, 1, 1, 0}\
}
```

```
#define L {\
{1, 0, 0, 0, 0},\
{1, 0, 0, 0, 0},\
{1, 0, 0, 0, 0},\
{1, 0, 0, 0, 0},\
{1, 0, 0, 0, 0},\
{1, 0, 0, 0, 0},\
{1, 1, 1, 1, 1}\
}

#define small_L {\
{0, 1, 1, 0, 0},\
{0, 0, 1, 0, 0},\
{0, 0, 1, 0, 0},\
{0, 0, 1, 0, 0},\
{0, 0, 1, 0, 0},\
{0, 0, 1, 0, 0},\
{0, 1, 1, 1, 0}\
}
#define O {\
{0, 1, 1, 1, 0},\
{1, 0, 0, 0, 1},\
{1, 0, 0, 0, 1},\
{1, 0, 0, 0, 1},\
{1, 0, 0, 0, 1},\
{1, 0, 0, 0, 1},\
{0, 1, 1, 1, 0}\
}

#define small_O {\
{0, 0, 0, 0, 0},\
{0, 0, 0, 0, 0},\
{0, 1, 1, 1, 0},\
{1, 0, 0, 0, 1},\
{1, 0, 0, 0, 1},\
{1, 0, 0, 0, 1},\
{0, 1, 1, 1, 0}\
}
```

```
#define small_W {\
{0, 0, 0, 0, 0},\
{0, 0, 0, 0, 0},\
{1, 0, 0, 0, 1},\
{1, 0, 0, 0, 1},\
{1, 0, 1, 0, 1},\
{1, 0, 1, 0, 1},\
{0, 1, 0, 1, 0}\
}

#define small_R {\
{0, 0, 0, 0, 0},\
{0, 0, 0, 0, 0},\
{0, 1, 0, 1, 1},\
{0, 1, 1, 0, 0},\
{0, 1, 0, 0, 0},\
{0, 1, 0, 0, 0},\
{0, 1, 0, 0, 0}\
}

#define small_D {\
{0, 0, 0, 0, 1},\
{0, 0, 0, 0, 1},\
{0, 1, 1, 0, 1},\
{1, 0, 0, 1, 1},\
{1, 0, 0, 0, 1},\
{1, 0, 0, 0, 1},\
{0, 1, 1, 1, 1}\
}
byte col = 0;
byte leds[5][7];
int pins[13]= {-1, 2, 3, 4, 5, 6, 13, 12, 11, 10, 9, 8, 7};
int cols[5] = {pins[1], pins[2], pins[3], pins[4], pins[5]};
int rows[7] = {pins[6], pins[7], pins[8], pins[9],
pins[10], pins[11], pins[12]};
```

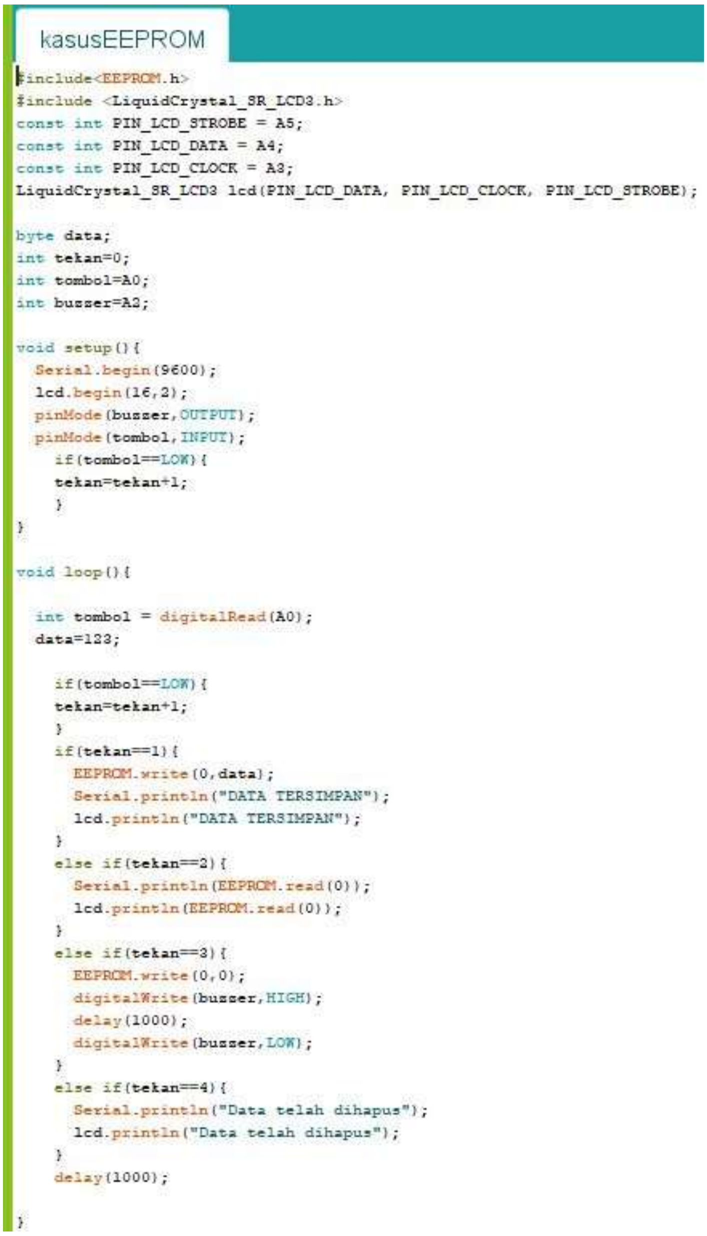


```
const int numPatterns = 12; byte
patterns[numPatterns][7][5] = {
SPACE, H, small_E, small_L, small_L, small_O,
SPACE, small_W, small_O, small_R, small_L,
small_D
};

int pattern = 0;
void setup()
{ for (int i = 1; i <= 12;
i++) {
pinMode(pins[i], OUTPUT);
 }
 for (int i = 1; i <= 5; i++) {
digitalWrite(cols[i - 1], LOW);
 }
for (int i = 1; i <= 7; i++) {
digitalWrite(rows[i - 1], LOW);
 }
clearLeds();

 FrequencyTimer2::disable();
 FrequencyTimer2::setPeriod(2000);
FrequencyTimer2::setOnOverflow(display);
setPattern(pattern);
}
void loop() {
pattern = ++pattern % numPatterns;
slidePattern(pattern, 100);
} void clearLeds() {
for (int i = 0; i < 5; i++) {
for (int j = 0; j < 7; j++) {
leds[i][j] = 0;
 }
 }
}
void setPattern(int pattern)
{
 for (int i = 0; i < 5; i++) {
  for (int j = 0; j < 7; j++) {
leds[i][j] = patterns[pattern][j][i];
 }
 }
}
```

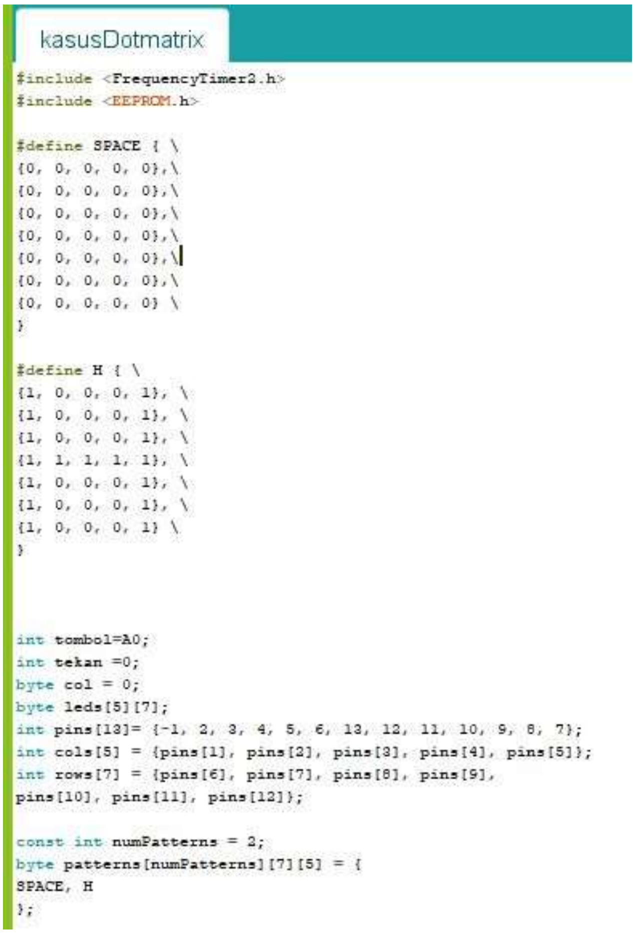


```
void slidePattern(int pattern, int del)
{
 for (int newcol = 0; newcol <= 4; newcol++) {

 // shift the first 4 columns left
for (int row = 0; row <= 6; row++)
for (int col = 0; col <= 3; col++)
leds[col][row] = leds[col+1][row];

 for (int row = 0; row <= 6; row++)
 leds[4][row] = patterns[pattern][row][newcol];

 delay(del);
 }
}
void display(){
 digitalWrite(cols[col], LOW);
col++; if (col == 5) {
col = 0; }
 for (int row = 0; row < 7; row++) {
if (leds[col][row] == 1) {
digitalWrite(rows[row], LOW);
 }
else {
 digitalWrite(rows[row], HIGH);
 }
}
 digitalWrite(cols[col], HIGH);
}
```

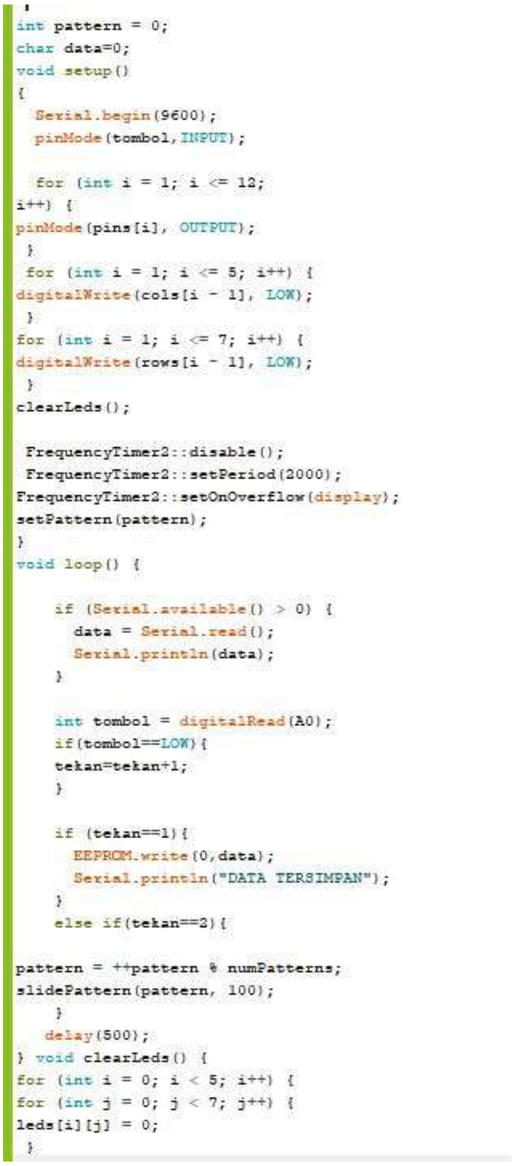
Kasus dot dan eprom



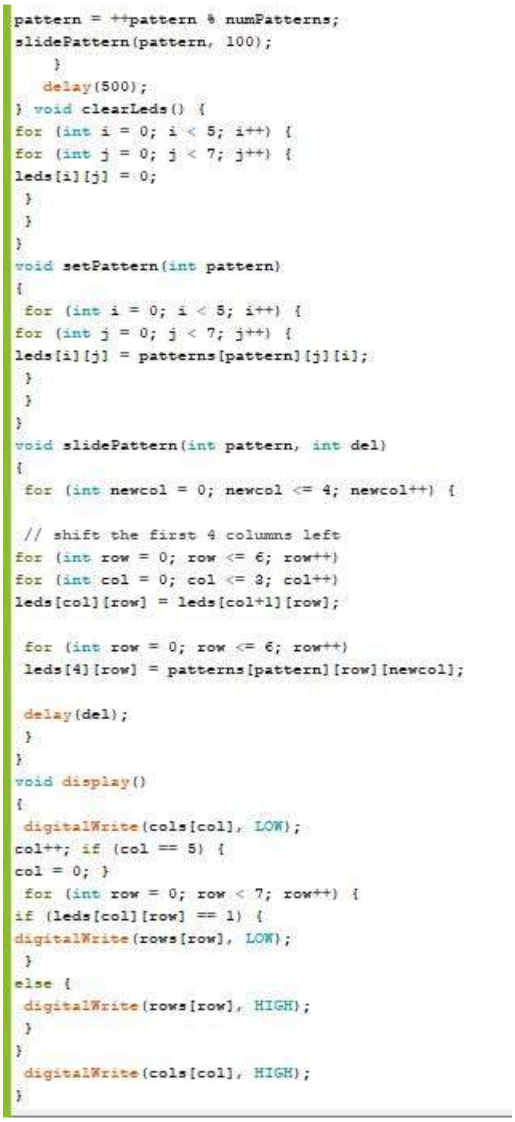
```
#include<EEPROM.h>
#include <LiquidCrystal_SR_LCD3.h>
const int PIN_LCD_STROBE = A5;
const int PIN_LCD_DATA = A4;
const int PIN_LCD_CLOCK = A3;
LiquidCrystal_SR_LCD3 lcd(PIN_LCD_DATA, PIN_LCD_CLOCK, PIN_LCD_STROBE);

byte data;
int tekan=0;
int tombol=A0;
int busser=A2;

void setup(){
  Serial.begin(9600);
  lcd.begin(16,2);
  pinMode(busser,OUTPUT);
  pinMode(tombol,INPUT);
    if(tombol==LOW){
    tekan=tekan+1;
    }
}

void loop(){

  int tombol = digitalRead(A0);
  data=123;

    if(tombol==LOW){
    tekan=tekan+1;
    }
    if(tekan==1){
      EEPROM.write(0,data);
      Serial.println("DATA TERSIMPAN");
      lcd.println("DATA TERSIMPAN");
    }
    else if(tekan==2){
      Serial.println(EEPROM.read(0));
      lcd.println(EEPROM.read(0));
    }
    else if(tekan==3){
      EEPROM.write(0,0);
      digitalWrite(busser,HIGH);
      delay(1000);
      digitalWrite(busser,LOW);
    }
    else if(tekan==4){
      Serial.println("Data telah dihapus");
      lcd.println("Data telah dihapus");
    }
    delay(1000);

}
```



```
#include <FrequencyTimer2.h>
#include <EEPROM.h>

#define SPACE { \
{0, 0, 0, 0, 0},\
{0, 0, 0, 0, 0},\
{0, 0, 0, 0, 0},\
{0, 0, 0, 0, 0},\
{0, 0, 0, 0, 0},\
{0, 0, 0, 0, 0},\
{0, 0, 0, 0, 0} \
}

#define H { \
{1, 0, 0, 0, 1}, \
{1, 0, 0, 0, 1}, \
{1, 0, 0, 0, 1}, \
{1, 1, 1, 1, 1}, \
{1, 0, 0, 0, 1}, \
{1, 0, 0, 0, 1}, \
{1, 0, 0, 0, 1} \
}



int tombol=A0;
int tekan =0;
byte col = 0;
byte leds[5][7];
int pins[13]= {-1, 2, 3, 4, 5, 6, 13, 12, 11, 10, 9, 8; 7};
int cols[5] = {pins[1], pins[2], pins[3], pins[4], pins[5]};
int rows[7] = {pins[6], pins[7], pins[8], pins[9],
pins[10], pins[11], pins[12]};

const int numPatterns = 2;
byte patterns[numPatterns][7][5] = {
SPACE, H
};
```



```
int pattern = 0;
char data=0;
void setup()
{
  Serial.begin(9600);
  pinMode(tombol,INPUT);

  for (int i = 1; i <= 12;
i++) {
pinMode(pins[i], OUTPUT);
 }
 for (int i = 1; i <= 5; i++) {
digitalWrite(cols[i - 1], LOW);
 }
for (int i = 1; i <= 7; i++) {
digitalWrite(rows[i - 1], LOW);
 }
clearLeds();

 FrequencyTimer2::disable();
 FrequencyTimer2::setPeriod(2000);
FrequencyTimer2::setOnOverflow(display);
setPattern(pattern);
}
void loop() {

    if (Serial.available() > 0) {
      data = Serial.read();
      Serial.println(data);
    }

    int tombol = digitalRead(A0);
    if(tombol==LOW){
    tekan=tekan+1;
    }

    if (tekan==1){
      EEPROM.write(0,data);
      Serial.println("DATA TERSIMPAN");
    }
    else if(tekan==2){

pattern = ++pattern % numPatterns;
slidePattern(pattern, 100);
    }
   delay(500);
} void clearLeds() {
for (int i = 0; i < 5; i++) {
for (int j = 0; j < 7; j++) {
leds[i][j] = 0;
 }
```



```
pattern = ++pattern % numPatterns;
slidePattern(pattern, 100);
    }
   delay(500);
} void clearLeds() {
for (int i = 0; i < 5; i++) {
for (int j = 0; j < 7; j++) {
leds[i][j] = 0;
 }
 }
}
void setPattern(int pattern)
{
 for (int i = 0; i < 5; i++) {
for (int j = 0; j < 7; j++) {
leds[i][j] = patterns[pattern][j][i];
 }
 }
}
void slidePattern(int pattern, int del)
{
 for (int newcol = 0; newcol <= 4; newcol++) {

 // shift the first 4 columns left
for (int row = 0; row <= 6; row++)
for (int col = 0; col <= 3; col++)
leds[col][row] = leds[col+1][row];

 for (int row = 0; row <= 6; row++)
 leds[4][row] = patterns[pattern][row][newcol];

 delay(del);
 }
}
void display()
{
 digitalWrite(cols[col], LOW);
col++; if (col == 5) {
col = 0; }
 for (int row = 0; row < 7; row++) {
if (leds[col][row] == 1) {
digitalWrite(rows[row], LOW);
 }
else {
 digitalWrite(rows[row], HIGH);
 }
}
 digitalWrite(cols[col], HIGH);
}
```

E. Kesimpulan

Kesimpulan praktikum kali ini adalah eprom lebih mirip seperti memori pada komputer dan pada kasus ini simpanan data akan di simpan dan akan dimunculkan pada LCD display

F. Link Video Praktikum

https://www.youtube.com/watch?v=JKcuz78bJ58