

REPUBLIC INDONESIA  
KEMENTERIAN HUKUM DAN HAK ASASI MANUSIA

# SURAT PENCATATAN CIPTAAN

Dalam rangka perlindungan ciptaan di bidang ilmu pengetahuan, seni dan sastra berdasarkan Undang-Undang Nomor 28 Tahun 2014 tentang Hak Cipta, dengan ini menerangkan:

Nomor dan tanggal permohonan : EC00202233617, 2 Juni 2022

## Pencipta

Nama : **Prof. Dr. Sapto Haryoko, M.Pd. dan Dr. Hendra Jaya, M.T**  
Alamat : **Jl. Mannuruki Kompleks BTN Tabaria, Makassar, SULAWESI SELATAN, 90224**  
Kewarganegaraan : **Indonesia**

## Pemegang Hak Cipta

Nama : **Universitas Negeri Makassar**  
Alamat : **Jl. A.P Pettarani Kampus UNM Gunung Sari, Makassar, SULAWESI SELATAN, 90222**  
Kewarganegaraan : **Indonesia**

Jenis Ciptaan : **Program Komputer**  
Judul Ciptaan : **Aplikasi Textile Circuit Berbasis Sensor Cahaya, Sensor Jarak, Dan Sensor Sentuh**

Tanggal dan tempat diumumkan untuk pertama kali di wilayah Indonesia atau di luar wilayah Indonesia : **2 Juni 2022, di Makassar**

Jangka waktu perlindungan : **Berlaku selama 50 (lima puluh) tahun sejak Ciptaan tersebut pertama kali dilakukan Pengumuman.**

Nomor pencatatan : **000349216**

adalah benar berdasarkan keterangan yang diberikan oleh Pemohon.  
Surat Pencatatan Hak Cipta atau produk Hak terkait ini sesuai dengan Pasal 72 Undang-Undang Nomor 28 Tahun 2014 tentang Hak Cipta.




a.n Menteri Hukum dan Hak Asasi Manusia  
Direktur Jenderal Kekayaan Intelektual  
u.b.  
Direktur Hak Cipta dan Desain Industri

Anggoro Dasananto  
NIP.196412081991031002

Disclaimer:

Dalam hal pemohon memberikan keterangan tidak sesuai dengan surat pernyataan, Menteri berwenang untuk mencabut surat pencatatan permohonan.





# **Aplikasi Textile Circuit berbasis Sensor Cahaya, Sensor Jarak, dan Sensor sentuh**

*Oleh:*

**Prof. Dr. Spto Haryoko, M.Pd.**

**Dr. Hendra Jaya, M.T**



# PROGRAM KOMPUTER

## Aplikasi Textile Circuit berbasis Sensor Cahaya, Sensor Jarak, dan Sensor sentuh

```
#include <SPI.h>
#include <Wire.h>
#include <Adafruit_GFX.h>
#include <Adafruit_SSD1306.h>

#include <NewPing.h>

#define TRIGGER_PIN  A2  // Arduino pin tied to trigger pin on the
ultrasonic sensor.
#define ECHO_PIN     A3  // Arduino pin tied to echo pin on the
ultrasonic sensor.
#define MAX_DISTANCE 400 // Maximum distance we want to ping for (in
centimeters). Maximum sensor distance is rated at 400-500cm.

NewPing sonar(TRIGGER_PIN, ECHO_PIN, MAX_DISTANCE); // NewPing setup of
pins and maximum distance.

#define SCREEN_WIDTH 128 // OLED display width, in pixels
#define SCREEN_HEIGHT 32 // OLED display height, in pixels

// Declaration for an SSD1306 display connected to I2C (SDA, SCL pins)
#define OLED_RESET     4 // Reset pin # (or -1 if sharing Arduino reset
pin)
Adafruit_SSD1306 display(SCREEN_WIDTH, SCREEN_HEIGHT, &Wire, OLED_RESET);

#define NUMFLAKES      10 // Number of snowflakes in the animation example

#define LOGO_HEIGHT    16
#define LOGO_WIDTH     16

#define buzz          A0
#define ldr           A7
#define touch         A6
#define ledB          3
#define ledM          4
#define ledH          5
#define push          6
#define ledP          8

int sLdr, sTouch, sJarak;

const unsigned char unm[] PROGMEM = {
  // 'unm_hitam', 128x32px
  0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xf8, 0x1f, 0xff, 0xff, 0xff,
  0xff, 0xff, 0xff, 0xff,
```

0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0x88, 0x19, 0xff, 0xff, 0xff,  
0xff, 0xff, 0xff, 0xff,  
0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0x08, 0x10, 0x7f, 0xff, 0xff,  
0xff, 0xff, 0xff, 0xff,  
0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xfe, 0x08, 0x10, 0x7f, 0xff, 0xff,  
0xff, 0xff, 0xff, 0xff,  
0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xf9, 0x08, 0x10, 0xdf, 0xff, 0xff,  
0xff, 0xff, 0xff, 0xff,  
0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xf0, 0x84, 0x01, 0x8f, 0xff, 0xff,  
0xff, 0xff, 0xff, 0xff,  
0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xe0, 0x80, 0x01, 0x07, 0xff, 0xff,  
0xff, 0xff, 0xff, 0xff,  
0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xe0, 0x00, 0x00, 0x07, 0xff, 0xff,  
0xff, 0xff, 0xff, 0xff,  
0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xf0, 0x00, 0x00, 0x07, 0xff, 0xff,  
0xff, 0xff, 0xff, 0xff,  
0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xc4, 0x0f, 0xe0, 0x13, 0xff, 0xff,  
0xff, 0xff, 0xff, 0xff,  
0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0x82, 0x1f, 0xf8, 0x61, 0xff, 0xff,  
0xff, 0xff, 0xff, 0xff,  
0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0x80, 0x3e, 0xfc, 0x01, 0xff, 0xff,  
0xff, 0xff, 0xff, 0xff,  
0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0x00, 0x7d, 0xfc, 0x00, 0xff, 0xff,  
0xff, 0xff, 0xff, 0xff,  
0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0x00, 0x79, 0xfe, 0x00, 0xff, 0xff,  
0xff, 0xff, 0xff, 0xff,  
0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xf1, 0x7f, 0xff, 0xff, 0xff,  
0xff, 0xff, 0xff, 0xff,  
0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xb6, 0xe0, 0xff, 0x6d, 0xff, 0xff,  
0xff, 0xff, 0xff, 0xff,  
0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0x00, 0x60, 0xbe, 0x00, 0xff, 0xff,  
0xff, 0xff, 0xff, 0xff,  
0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0x00, 0x44, 0xbe, 0x00, 0xff, 0xff,  
0xff, 0xff, 0xff, 0xff,  
0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0x00, 0x30, 0x7e, 0x00, 0xff, 0xff,  
0xff, 0xff, 0xff, 0xff,  
0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0x1c, 0x30, 0x7c, 0x3c, 0xff, 0xff,  
0xff, 0xff, 0xff, 0xff,  
0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0x38, 0x7c, 0xff, 0xff, 0xff,  
0xff, 0xff, 0xff, 0xff,  
0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0x9b, 0xf9, 0xff, 0xff, 0xff,  
0xff, 0xff, 0xff, 0xff,  
0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xe0, 0xc7, 0xf3, 0x03, 0xff, 0xff,  
0xff, 0xff, 0xff, 0xff,  
0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xc0, 0x30, 0x04, 0x03, 0xff, 0xff,  
0xff, 0xff, 0xff, 0xff,  
0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xc0, 0x0c, 0x38, 0x83, 0xff, 0xff,  
0xff, 0xff, 0xff, 0xff,  
0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xe7, 0xf8, 0x0f, 0xe7, 0xff, 0xff,  
0xff, 0xff, 0xff, 0xff,  
0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0x0f, 0xf0, 0x7f, 0xff, 0xff,  
0xff, 0xff, 0xff, 0xff,  
0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xfe, 0x00, 0x00, 0x3f, 0xff, 0xff,  
0xff, 0xff, 0xff, 0xff,  
0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xfc, 0x1c, 0x38, 0x3f, 0xff, 0xff,  
0xff, 0xff, 0xff, 0xff,

```

    0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xfe, 0x38, 0x1c, 0x7f, 0xff, 0xff,
    0xff, 0xff, 0xff, 0xff,
    0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xf8, 0x0f, 0xff, 0xff, 0xff,
    0xff, 0xff, 0xff, 0xff,
    0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xf8, 0x1f, 0xff, 0xff, 0xff,
    0xff, 0xff, 0xff, 0xff,
};

void setup() {
  Serial.begin(9600);
  pinMode(buzz, OUTPUT);
  pinMode(ldr, INPUT);
  pinMode(push, INPUT);
  pinMode(ledH, OUTPUT);
  pinMode(ledM, OUTPUT);
  pinMode(touch, INPUT);
  pinMode(ledB, OUTPUT);
  pinMode(ledP, OUTPUT);
  // SSD1306_SWITCHCAPVCC = generate display voltage from 3.3V internally
  if(!display.begin(SSD1306_SWITCHCAPVCC, 0x3C)) { // Address 0x3C for
128x32
    Serial.println(F("SSD1306 allocation failed"));
    for(;;); // Don't proceed, loop forever
  }
  display.clearDisplay();
  display.drawPixel(10, 10, WHITE);
  display.display();

  display.drawBitmap(0, 0, unm, 128, 32, WHITE);
  display.display();
  delay(1000);
}

void loop() {
  sLdr = analogRead(ldr);
  sTouch = analogRead(touch);
  sJarak = sonar.ping_cm();
  display.clearDisplay();

  display.setTextSize(1);
  display.setTextColor(WHITE);
  display.setCursor(0,0);
  display.println(F("Cahaya"));
  display.setCursor(50,0);
  display.println(sLdr);
  display.setCursor(0,10);
  display.println(F("Touch"));
  display.setCursor(50,10);
  display.println(sTouch);
  display.setCursor(0,20);
  display.println(F("Jarak"));
  display.setCursor(50,20);
  display.println(sJarak);
  display.display();
  delay(50);
}

```

```
    if (sJarak < 100) { digitalWrite(ledP, HIGH); }  
    else { digitalWrite(ledP, LOW); }  
  
    if (sLdr > 100) { digitalWrite(ledH, HIGH); }  
    else { digitalWrite(ledH, LOW); }  
  
    if (sTouch > 400) { digitalWrite(ledM, HIGH); }  
    else { digitalWrite(ledM, LOW); }  
  
}
```