[36] Thomas Publishing Company, "Industrial Limit Switch Theory and Applications." [Online]. Available: https://www.thomasnet.com/

[37] F. H. Nasution and M. M. Siregar, "Perancangan Sistem Pengamanan Pintu Otomatis Menggunakan Mikrokontroler," Jurnal Infra, vol. 8, no. 1, pp. 55–60, 2020.

[38] D. Fitriyani and R. Ridwan, "Pengendalian Kunci Pintu Otomatis Menggunakan Mikrokontroler Arduino dan Modul RFID," Jurnal Teknologi dan Sistem Komputer, vol. 10, no. 1, pp. 22–27, 2022.

[39] C. E. P. Wijayanti and B. H. Prasetio, "Alat Ukur Berat dan Volume Buah Menggunakan Multi Sensor Ultrasonic dan Metode Regresi Linier Berganda Berbasis Arduino Uno," vol. 1, no. 1, pp. 1–9, 2024.

[40] F. R. Utami, M. A. Riyadi, and Y. Christyono, "Perancangan Catu Daya Arus Searah Keluaran Ganda sebagai Penggerak Robot Lengan Artikulasi," Transient, vol. 9, no. 3, 2020.

[41] Datasheet resmi XL4015, XLsemi Semiconductor Co. Ltd, 2018.

[42] M. Hasanuzzaman et al., "Modern Electrical Control System Applications in Smart Homes and Automation," Journal of Electrical Engineering, 2019.

[43] J. Kaur and G. Kaur, "Smart Control of Home Appliances Using IoT-Based Relay Modules," International Journal of Scientific Research in Engineering and Management, 2021.

[44] A. Novriadi, "Perancangan Pengontrolan Overhead Crane Menggunakan Kabel dan Nirkabel Berbasis Arduino," JTT (Jurnal Teknologi Terpadu), vol. 7, no. 2, pp. 76–84, 2019.

[45] S. Purwanto, "Rancang Bangun Electric Power Converter (Catu Daya) untuk Alat Anodizing Portable," Energi & Kelistrikan, vol. 13, no. 2, pp. 86–94, 2021.

[46] N. Khaerudin and M. Yusuf, "Analisis Konsumsi Daya Listrik pada Sistem Instalasi Rumah Tangga," *Jurnal Teknik Elektro*, vol. 11, no. 2, pp. 85–92, 2022.

LAMPIRAN

Lampiran 1. Code Program Arduino IDE

```
#include <WiFi.h>
#include <HTTPClient.h>
#include <HardwareSerial.h>

#define QRIS_PIN 2

// ===== PILOT LAMP =====
#define LAMP_S 5
#define LAMP_M 18
#define LAMP_L 19

// ===== LOKER S =====
#define PINTU_S 12
#define TRIG_S 26
#define ECHO_S 32
#define LIMIT_S 21

// ===== LOKER M =====
#define PINTU_M 27
#define TRIG_M 25
#define ECHO_M 35
#define LIMIT_M 22

// ===== LOKER L =====
#define PINTU_L 14
#define TRIG_L 33
#define ECHO_L 34
#define LIMIT_L 23

HardwareSerial nextion(2);
```

```
const char* ssid = "ode";
const char* password = "ode170502";
const char* scriptURL =
"https://script.google.com/macros/s/AKfycbwQCvo_PSwkAnVBvcOBjYmx2gZ4
75gKOZ5p7WLRYPD7x00_y3NfYit_ZNPBO5wy-FQQhA/exec";

/* ================= STRUCT ================= */
struct Loker {

  String id;
  int pintu, trig, echo, limitSwitch;
  int pilotLamp;

  bool tungguQRIS = false;
  bool menungguTutup = false;
  bool errorTutup = false;
  bool sudahKirimBarang = false;

  unsigned long waktuMulaiTutup = 0;

  // ===== DEBOUNCE LIMIT SWITCH =====
  int lastLimitState = HIGH;
  int limitState = HIGH;
  unsigned long lastDebounceTime = 0;
  const int debounceDelay = 50;

  Loker(String i, int p, int t, int e, int ls, int lamp)
    : id(i), pintu(p), trig(t), echo(e), limitSwitch(ls), pilotLamp(lamp) {}
};

Loker lokerS("S", PINTU_S, TRIG_S, ECHO_S, LIMIT_S, LAMP_S);
```

```
Loker lokerM("M", PINTU_M, TRIG_M, ECHO_M, LIMIT_M, LAMP_M);
Loker lokerL("L", PINTU_L, TRIG_L, ECHO_L, LIMIT_L, LAMP_L);

bool globalModePut = false;
bool globalModeTake = false;

String inputKode = "";
Loker* lokerDipilih = nullptr;
int hargaTerakhir = 0;

const String kodeS = "1234";
const String kodeM = "5678";
const String kodeL = "9012";

/* ================= LOGIN ADMIN ================= */

const String ADMIN_PIN = "1705";

bool modeLogin = true;

/* ================= SETUP ================= */
void setup() {

  Serial.begin(115200);
  nextion.begin(9600);

pinMode(QRIS_PIN, INPUT_PULLDOWN);

  for (Loker* l : {&lokerS, &lokerM, &lokerL}) {

    pinMode(l->pintu, OUTPUT);
    digitalWrite(l->pintu, LOW);
```

```
    pinMode(l->pilotLamp, OUTPUT);
    digitalWrite(l->pilotLamp, LOW);

    pinMode(l->trig, OUTPUT);
    pinMode(l->echo, INPUT);

    pinMode(l->limitSwitch, INPUT_PULLUP);
   }

   WiFi.begin(ssid, password);

   while (WiFi.status() != WL_CONNECTED) {
    delay(500);
    Serial.print(".");
   }

   Serial.println("\nWiFi Connected");

   sendToNextion("page CODE");

modeLogin = true;

inputKode = "";
}

/* ================== LOOP ================== */
void loop() {

  bacaNextion();

  handleLoker(lokerS);
```

```
  handleLoker(lokerM);
  handleLoker(lokerL);

  delay(30);
}

/* ================== HANDLE LOKER ================== */
void handleLoker(Loker& l) {

  long jarak = readUltrasonic(l.trig, l.echo);

  // ===== PILOT LAMP =====
bool kosong;

if (l.id == "L")
    kosong = (getUkuranLokerL(jarak) == 0);
else
    kosong = (jarak >= 25);

digitalWrite(l.pilotLamp, kosong ? HIGH : LOW);

// ===== MODE PUT =====
if (globalModePut) {

    bool kosong;
    if(l.id=="L") kosong=(getUkuranLokerL(jarak)==0);
    else kosong=(jarak>=25);

    nextion.print("bt" + l.id + ".en=" + String(kosong ? 1 : 0));
    sendEnd();

    uint16_t warna = kosong ? 2016 : 63488; // hijau : merah
```

```
    nextion.print("bt" + l.id + ".bco=" + String(warna));
    sendEnd();

    nextion.print("bt" + l.id + ".bco2=" + String(warna));
    sendEnd();

    nextion.print("t" + l.id + ".txt=\"" +
                String(kosong ? "Kosong" : "Ada Barang") + "\"");
    sendEnd();
}

// ===== MODE TAKE =====
if (globalModeTake) {

    bool adaBarang;
    if(l.id=="L") adaBarang=(getUkuranLokerL(jarak)!=0);
    else adaBarang=(jarak<25);

    nextion.print("bt" + l.id + ".en=" + String(adaBarang ? 1 : 0));
    sendEnd();

    uint16_t warna = adaBarang ? 63488 : 2016; // merah : hijau

    nextion.print("bt" + l.id + ".bco=" + String(warna));
    sendEnd();

    nextion.print("bt" + l.id + ".bco2=" + String(warna));
    sendEnd();

    if (adaBarang) {
```

```
      unsigned long durasiMs = millis() - l.waktuMulaiTutup;

      int jam = max(1, (int)(durasiMs / 3600000UL));

      int hargaAwal;
      if(l.id=="L"){
        int ukuran=getUkuranLokerL(jarak);
        if(ukuran==1) hargaAwal=1000;
        else if(ukuran==2) hargaAwal=2000;
        else hargaAwal=3000;
      } else {
        hargaAwal=(l.id=="S")?1000:2000;
      }

      int harga = hargaAwal + (jam - 1) * 1000;

      nextion.print("t" + l.id + ".txt=\"Rp" + String(harga) + "\"");
      sendEnd();
    }
    else {

      nextion.print("t" + l.id + ".txt=\"Tidak Ada\"");
      sendEnd();
    }
}

  // ===== QRIS SUCCESS =====

  if (l.tungguQRIS && digitalRead(QRIS_PIN) == HIGH) {

    sendToNextion("page SUCCESS");
long j = readUltrasonic(l.trig, l.echo);
```

```
String ukuran = getUkuranString(l, j);

sendToSheet(
   "TAKE",
   l.id,
   ukuran,
   "Pembayaran QRIS Berhasil"
);

   digitalWrite(l.pintu, HIGH);
   delay(3000);
   digitalWrite(l.pintu, LOW);

   sendToNextion("page MENU");

   l.tungguQRIS = false;
   l.waktuMulaiTutup = 0;
  }

  // ===== DEBOUNCE LIMIT SWITCH =====

  int reading = digitalRead(l.limitSwitch);

  if (reading != l.lastLimitState) {
   l.lastDebounceTime = millis();
  }

  if ((millis() - l.lastDebounceTime) > l.debounceDelay) {
   l.limitState = reading;
  }
```

```
  l.lastLimitState = reading;

  // ===== MENUNGGU PINTU DITUTUP =====
  if (l.menungguTutup) {

    if (l.limitState == LOW) {

      digitalWrite(l.pintu, LOW);

      sendText("TERKUNCI");

long j = readUltrasonic(l.trig, l.echo);

bool adaBarang =
    (l.id=="L") ? (getUkuranLokerL(j)!=0) : (j<25);

if (adaBarang && !l.sudahKirimBarang) {

    String ukuran = getUkuranString(l, j);

    sendToSheet(
        "PUT",
        l.id,
        ukuran,
        "Barang Terdeteksi"
    );

    l.sudahKirimBarang = true;
}

      sendToNextion("page MENU");
```

```
      l.menungguTutup = false;
    }

    else if (millis() - l.waktuMulaiTutup > 10000 && !l.errorTutup) {

      digitalWrite(l.pintu, LOW);

      sendText("ERROR");

long j = readUltrasonic(l.trig, l.echo);

String ukuran = getUkuranString(l, j);

sendToSheet(
    "PUT",
    l.id,
    ukuran,
    "Pintu Tidak Tertutup!"
);

l.errorTutup = true;
    }
else if (l.errorTutup && l.limitState == LOW) {

    sendToNextion("page MENU");

    long j = readUltrasonic(l.trig, l.echo);

    String ukuran = getUkuranString(l, j);

    sendToSheet(
        "PUT",
```

```
        l.id,
        ukuran,
        "Reset Setelah Error"
    );

    l.errorTutup = false;
    l.menungguTutup = false;
    }
  }
}

/* ================== NEXTION ================== */
void bacaNextion() {

  static String input = "";

  while (nextion.available()) {

    input += char(nextion.read());

if (input.endsWith("b0\xFF\xFF\xFF")) {

    if(modeLogin) return;

    globalModePut = true;
    globalModeTake = false;

    sendToNextion("page PUT");
}

else if (input.endsWith("b1\xFF\xFF\xFF")) {
```

```
    if(modeLogin) return;

    globalModePut = false;
    globalModeTake = true;

    sendToNextion("page TAKE");
}

    else if (input.endsWith("btS\xFF\xFF\xFF")) tekanLoker(lokerS);
    else if (input.endsWith("btM\xFF\xFF\xFF")) tekanLoker(lokerM);
    else if (input.endsWith("btL\xFF\xFF\xFF")) tekanLoker(lokerL);

    else if (input.endsWith("bt1\xFF\xFF\xFF")) tambahDigit('1');
    else if (input.endsWith("bt2\xFF\xFF\xFF")) tambahDigit('2');
    else if (input.endsWith("bt3\xFF\xFF\xFF")) tambahDigit('3');
    else if (input.endsWith("bt4\xFF\xFF\xFF")) tambahDigit('4');
    else if (input.endsWith("bt5\xFF\xFF\xFF")) tambahDigit('5');
    else if (input.endsWith("bt6\xFF\xFF\xFF")) tambahDigit('6');
    else if (input.endsWith("bt7\xFF\xFF\xFF")) tambahDigit('7');
    else if (input.endsWith("bt8\xFF\xFF\xFF")) tambahDigit('8');
    else if (input.endsWith("bt9\xFF\xFF\xFF")) tambahDigit('9');
    else if (input.endsWith("bt0\xFF\xFF\xFF")) tambahDigit('0');
    //================================================
// BUTTON BACK / CANCEL (Page CODE)
// Hex : 65 07 0D 01 FF FF FF
//================================================
else if (input.endsWith("\x65\x07\x0D\x01\xFF\xFF\xFF")) {

    // Hapus semua input PIN
    inputKode = "";
    sendText("tn", "");
```

```
    // Jika masih di login admin
    if (modeLogin) {

      // Tetap di halaman CODE, hanya mengosongkan PIN
      sendToNextion("page CODE");
    }

    // Jika sedang memasukkan PIN loker
    else {

      lokerDipilih = nullptr;

      globalModePut = false;
      globalModeTake = true;

      sendToNextion("page TAKE");
    }

    input = "";
    return;
}

    else if (input.endsWith("btD\xFF\xFF\xFF")) {

    if (inputKode.length() > 0) {

      inputKode.remove(inputKode.length() - 1);
    }

    sendText("tn", inputKode);
    }
```

```
    else if (input.endsWith("btE\xFF\xFF\xFF")) {

//================================================
// LOGIN ADMIN
//================================================

 if(modeLogin)
{
   if(inputKode == ADMIN_PIN)
   {
      modeLogin = false;

      // Reset kondisi sistem
      globalModePut = false;
      globalModeTake = false;

      inputKode = "";

      sendText("tn","");

      sendToNextion("page MENU");
   }
   else
   {
      inputKode = "";

      sendText("tn","SALAH");

      delay(1500);

      inputKode = "";
```

```
        sendText("tn","");
        sendToNextion("page CODE");
    }

    return;
}

//=================================================
// PIN LOKER
//=================================================

    bool valid = false;

    if (lokerDipilih == &lokerS && inputKode == kodeS) valid = true;
    if (lokerDipilih == &lokerM && inputKode == kodeM) valid = true;
    if (lokerDipilih == &lokerL && inputKode == kodeL) valid = true;

    if (valid) {

        sendText("tS", "");
        sendText("tM", "");
        sendText("tL", "");

        if(lokerDipilih == &lokerS){

        sendToNextion("page S");

        delay(300);

        sendText("tS","Rp" + String(hargaTerakhir));

        }
```

```
    if(lokerDipilih == &lokerM){
    sendToNextion("page M");
    delay(300);
    sendText("tM","Rp" + String(hargaTerakhir));
    }
    if(lokerDipilih == &lokerL){
    sendToNextion("page L");
    delay(300);
    sendText("tL","Rp" + String(hargaTerakhir));
    }
    lokerDipilih->tungguQRIS = true;
    inputKode = "";
}
else {
    sendText("tn", "SALAH");
    delay(1000);
    inputKode = "";
```

```
      sendText("tn", "");
    }
   }

   if (input.length() > 40) input = "";
  }
}

/* ================== TEKAN LOKER ================== */
void tekanLoker(Loker& l) {

  long jarak = readUltrasonic(l.trig, l.echo);

bool kosong;

if(l.id=="L")
   kosong=(getUkuranLokerL(jarak)==0);
else
   kosong=(jarak>=25);

if(globalModePut && kosong)
{

   digitalWrite(l.pintu, HIGH);

   sendText("TERBUKA");

String ukuran=getUkuranString(l,jarak);

sendToSheet("PUT", l.id, ukuran, "Loker Dibuka");

   l.waktuMulaiTutup = millis();
```

```
    l.menungguTutup = true;
    l.errorTutup = false;
    l.sudahKirimBarang = false;
  }

bool adaBarang;

if(l.id=="L")
    adaBarang=(getUkuranLokerL(jarak)!=0);
else
    adaBarang=(jarak<25);

if(globalModeTake && adaBarang) {

    unsigned long durasiMs = millis() - l.waktuMulaiTutup;

    int jam = max(1, (int)(durasiMs / 3600000UL));

    int hargaAwal;
    if(l.id=="L"){
      int ukuran=getUkuranLokerL(jarak);
      if(ukuran==1) hargaAwal=1000;
      else if(ukuran==2) hargaAwal=2000;
      else hargaAwal=3000;
    } else {
      hargaAwal=(l.id=="S")?1000:2000;
    }

    hargaTerakhir = hargaAwal + (jam - 1) * 1000;

    lokerDipilih = &l;
```

```
    inputKode = "";

    sendText("tn", "");

// ================== PIN LOCKER ==================
    sendToNextion("page CODE");
  }
}

void tambahDigit(char digit) {

  if (inputKode.length() < 4) {

    inputKode += digit;

    sendText("tn", inputKode);
  }
}
/* ================== UTIL ================== */

long readUltrasonic(int trig, int echo) {

  digitalWrite(trig, LOW);
  delayMicroseconds(2);

  digitalWrite(trig, HIGH);
  delayMicroseconds(10);

  digitalWrite(trig, LOW);

  long duration = pulseIn(echo, HIGH, 30000);
```

```
  return duration * 0.034 / 2;
}

//=======================================================
// DETEKSI UKURAN KOPER DI LOKER L
//=======================================================
int getUkuranLokerL(long jarak)
{
   if (jarak >= 27) return 1;
   else if (jarak >= 22) return 2;
   else if (jarak >= 18) return 3;
   else return 0;
}

String getUkuranString(Loker &l, long jarak)
{
   if (l.id == "S")
      return "S";

   if (l.id == "M")
      return "M";

   int ukuran = getUkuranLokerL(jarak);

   if (ukuran == 1)
      return "S";

   if (ukuran == 2)
      return "M";

   return "L";
```

```
}

void sendToNextion(String cmd) {

  nextion.print(cmd);

  sendEnd();
}

void sendText(String obj, String teks) {

  nextion.print(obj + ".txt=\"" + teks + "\"");

  sendEnd();
}

void sendText(String teks) {

  nextion.print("tS.txt=\"" + teks + "\"");

  sendEnd();
}

void sendEnd() {

  nextion.write(0xFF);
  nextion.write(0xFF);
  nextion.write(0xFF);
}

void setButtonState(String id, bool enable, String teks) {
```

```
  nextion.print("bt" + id + ".en=" + (enable ? "1" : "0"));
  sendEnd();

  uint16_t color = enable ? 2016 : 63488;

  nextion.print("bt" + id + ".bco=" + color);
  sendEnd();

  nextion.print("bt" + id + ".bco2=" + color);
  sendEnd();

  if (teks != "") {

    nextion.print("t" + id + ".txt=\"" + teks + "\"");

    sendEnd();
  }
}

void sendToSheet(String mode, String loker, String ukuran, String status) {

  HTTPClient http;

  http.begin(scriptURL);

  http.addHeader("Content-Type", "application/x-www-form-urlencoded");

  String postData =
  "mode=" + mode +
  "&loker=" + loker +
  "&ukuran=" + ukuran +
  "&status=" + status;
```

```
  http.POST(postData);

  http.end();
}
```

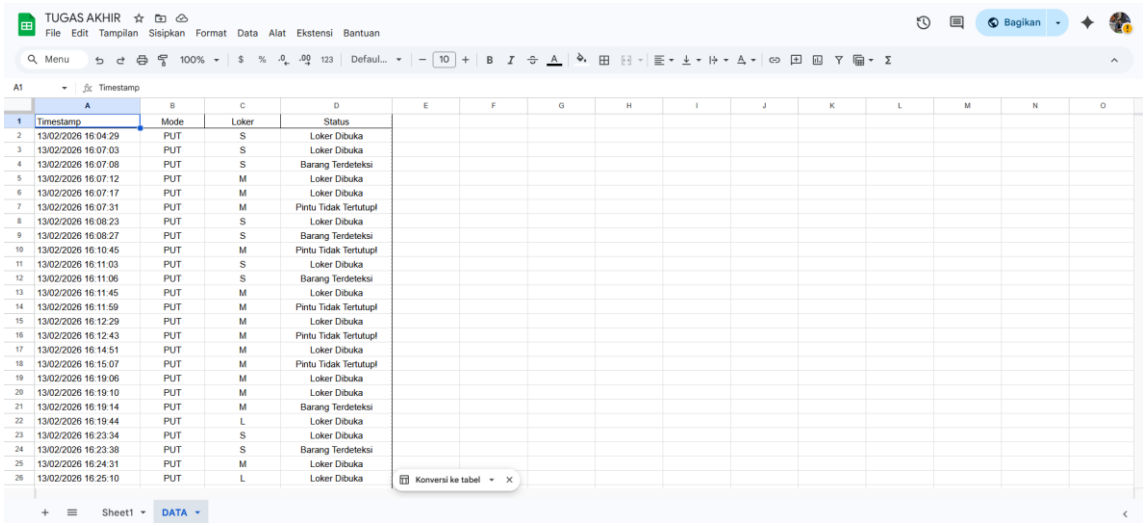
Lampiran 2. Code Program Apps Script

```
function doPost(e) {
  try {
    if (!e || !e.parameter) {
      return outputJSON("error", "No data received");
    }

    const mode   = (e.parameter.mode   || "").toUpperCase().trim();
    const loker  = (e.parameter.loker  || "").toUpperCase().trim();
    const ukuran = (e.parameter.ukuran || "").toUpperCase().trim();
    const status = (e.parameter.status || "").trim();

    if (!mode || !loker || !ukuran || !status) {
      return outputJSON("error", "Incomplete data");
    }

    const ss = SpreadsheetApp.getActiveSpreadsheet();
    let sheet = ss.getSheetByName("DATA");

    // Jika sheet belum ada → buat
    if (!sheet) {
      sheet = ss.insertSheet("DATA");
    }

    // Jika header belum ada → buat header
```

```
    if (sheet.getLastRow() === 0) {
      sheet.appendRow([
        "Timestamp",
        "Mode",
        "Loker",
        "Ukuran",
        "Status"
      ]);
    }

    // Tambahkan data baru
    sheet.appendRow([
      new Date(),
      mode,
      loker,
      ukuran,
      status
    ]);

    return outputJSON("success", "Data saved");

  } catch (err) {
    return outputJSON("error", err.toString());
  }
}

/* ================== JSON RESPONSE ================== */
function outputJSON(status, message) {
  return ContentService
    .createTextOutput(
      JSON.stringify({
        result: status,
```

```
      message: message
    })
  )
  .setMimeType(ContentService.MimeType.JSON);
}
```

Lampiran 3. Tampilan Spreadsheets



TUGAS AKHIR

File Edit Tampilan Sisipkan Format Data Alat Ekstensi Bantuan

Bagikan

A1 | Timestamp

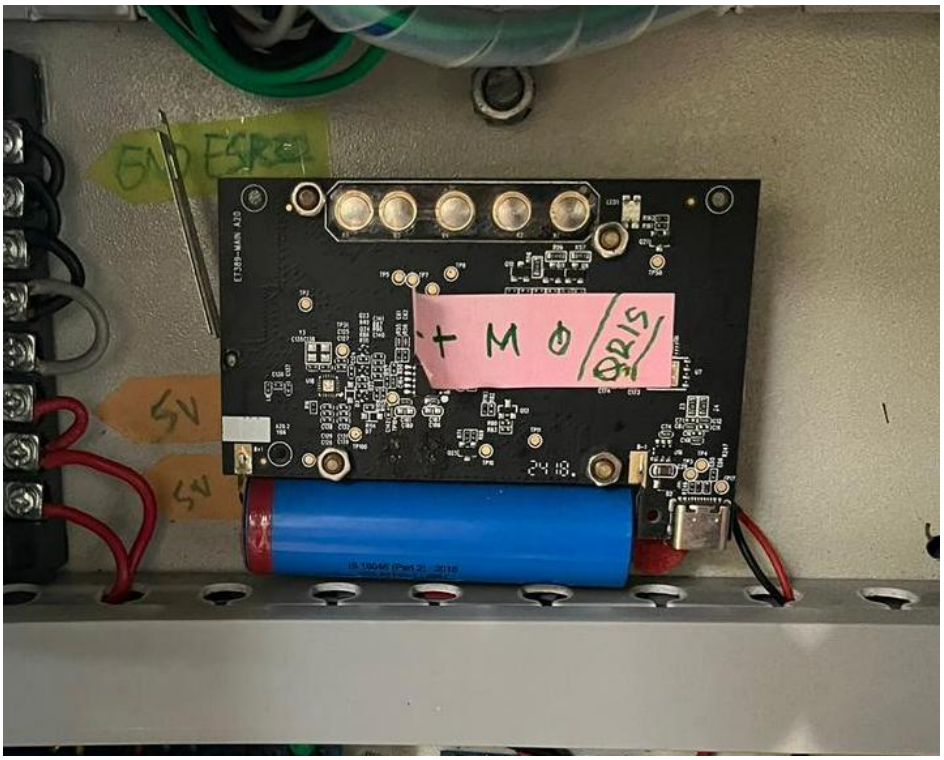
	A	B	C	D
1	Timestamp	Mode	Loker	Status
2	13/02/2026 16:04:29	PUT	S	Loker Dibuka
3	13/02/2026 16:07:03	PUT	S	Loker Dibuka
4	13/02/2026 16:07:08	PUT	S	Barang Terdeteksi
5	13/02/2026 16:07:12	PUT	M	Loker Dibuka
6	13/02/2026 16:07:17	PUT	M	Loker Dibuka
7	13/02/2026 16:07:31	PUT	M	Pintu Tidak Tertutup!
8	13/02/2026 16:08:23	PUT	S	Loker Dibuka
9	13/02/2026 16:08:27	PUT	S	Barang Terdeteksi
10	13/02/2026 16:10:45	PUT	M	Pintu Tidak Tertutup!
11	13/02/2026 16:11:03	PUT	S	Loker Dibuka
12	13/02/2026 16:11:06	PUT	S	Barang Terdeteksi
13	13/02/2026 16:11:45	PUT	M	Loker Dibuka
14	13/02/2026 16:11:59	PUT	M	Pintu Tidak Tertutup!
15	13/02/2026 16:12:29	PUT	M	Loker Dibuka
16	13/02/2026 16:12:43	PUT	M	Pintu Tidak Tertutup!
17	13/02/2026 16:14:51	PUT	M	Loker Dibuka
18	13/02/2026 16:15:07	PUT	M	Pintu Tidak Tertutup!
19	13/02/2026 16:19:06	PUT	M	Loker Dibuka
20	13/02/2026 16:19:10	PUT	M	Loker Dibuka
21	13/02/2026 16:19:14	PUT	M	Barang Terdeteksi
22	13/02/2026 16:19:44	PUT	L	Loker Dibuka
23	13/02/2026 16:23:34	PUT	S	Loker Dibuka
24	13/02/2026 16:23:38	PUT	S	Barang Terdeteksi
25	13/02/2026 16:24:31	PUT	M	Loker Dibuka
26	13/02/2026 16:25:10	PUT	L	Loker Dibuka

Konversi ke tabel

Sheet1 DATA

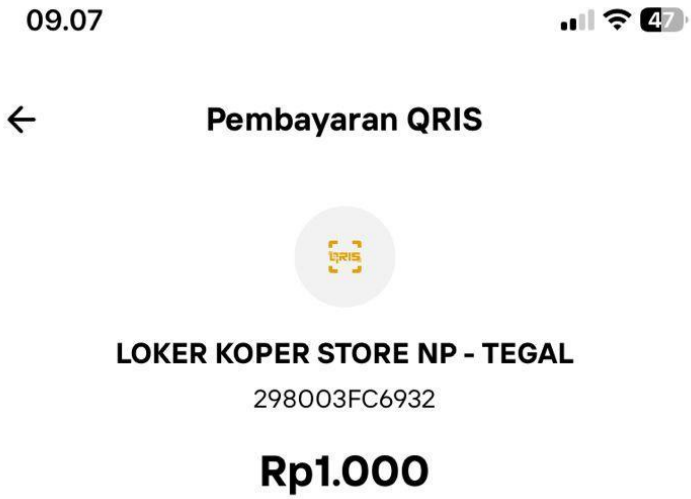
Lampiran 4. Tampilan HMI

Lampiran 5. Gambar Perangkat Loker Koper Pembayaran QRIS



Lampiran 6. Dokumentasi Pengambilan Data


09.05
Pembayaran QRIS
LOKER KOPER STORE NP - TEGAL
293403F31122
Rp1.000
Detail transaksi
Tanggal dan waktu
11 Mar 2026 • 16:24:34
Tipe transaksi
Pengeluaran


09.07
Pembayaran QRIS
LOKER KOPER STORE NP - TEGAL
298003FC6932
Rp1.000
Detail transaksi
Tanggal dan waktu
10 Mar 2026 • 17:58:05
Tipe transaksi
Pengeluaran

09.06

47

← Pembayaran QRIS

LOKER KOPER STORE NP - TEGAL

290203EE1580

Rp1.000

Detail transaksi

Tanggal dan waktu	11 Mar 2026 • 16:22:18
Tipe transaksi	Pengeluaran

Lampiran 7. Datasheet ESP32

Zerynth
PINOUT DIAGRAM

DOIT ESP32 DevKit v1

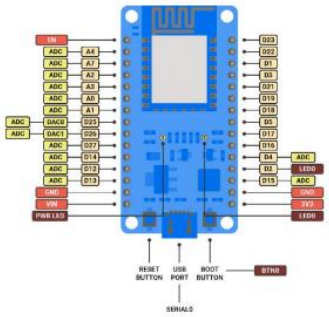
Zerynth

DO NOT USE D6 TO D11

PWM IS ENABLED ON EVERY DIGITAL PIN

ICU NOT SUPPORTED

ADC ON PINS D4, D12, D13, D14, D15, D25, D26, D27
CAN BE READ ONLY WITH WI-FI NOT STARTED



More info about DOIT Esp32 DevKit v1 can be found here.

Flash Layout

The internal flash of the ESP32 module is organized in a single flash area with pages of 4096 bytes each. The flash starts at address 0x00000, but many areas are reserved for Esp32 IDF SDK and Zerynth VM. There exist two different layouts based on the presence of BLE support.

In particular, for non-BLE VMs:

Start address	Size	Content
0x00009000	16Kb	Esp32 NVS area
0x0000D000	8Kb	Esp32 OTA data
0x0000F000	4Kb	Esp32 PHY data
0x00010000	1Mb	Zerynth VM
0x00110000	1Mb	Zerynth VM (FOTA)
0x00210000	512Kb	Zerynth Bytecode
0x00290000	512Kb	Zerynth Bytecode (FOTA)
0x00310000	512Kb	Free for user storage
0x00390000	448Kb	Reserved

Lampiran 8. Datasheet Sensor Ultrasonik HC-SR04

www.elecfreaks.com

2.2 Product Image

HC-SR04

2.3、Module pin definitions

Types	Pin Symbol	Pin Function Description
HC-SR04	VCC	5V power supply
	Trig	Trigger pin
	Echo	Receive pin
	GND	Power ground

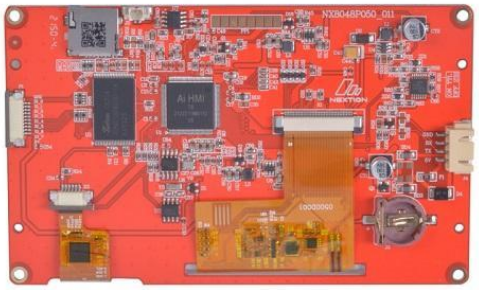
2.4、Electrical parameters

Electrical Parameters	HC-SR04 Ultrasonic Module
Operating Voltage	DC-5V
Operating Current	15mA
Operating Frequency	40KHZ
Farthest Range	4m
Nearest Range	2cm
Measuring Angle	15 Degree
Input Trigger Signal	10us TTL pulse
Output Echo Signal	Output TTL level signal, proportional with range
Dimensions	45*20*15mm

Lampiran 9. Datasheet HMI Nextion NX8048P050-011

Nextion 5.0” Intelligent Series HMI Touch Display

Nextion is a Human Machine Interface （HMI）solution combining an onboard processor and memory touch display with Nextion Editor software for HMI GUI project development.



NX8048P050-011R/C Hardware Feature:

- 5.0” LCD-TFT Capacitive/Resistive HMI touch display module
- 800 x 480 screen resolution
- RGB 65K true-to-life colours
- Onboard 200MHz MCU
- 512 KB SRAM
- 128 MB Flash memory
- XH2.54 4 Pins (+5V, TX, RX, GND) TTL serial interface
- Built-in 1024 Byte EEPROM， RTC and 8 GPIOs

Lampiran 10. Datasheet Solenoid Door Lock 12 VDC

Solenoid Lock

multicomp PRO

RoHS Compliant

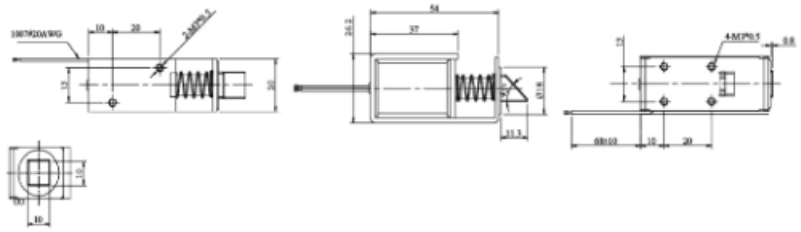
Specifications

Operating Temperature / Humidity	: -20°C to +45°C / 5% to 95% RH
Store Temperature / Humidity	: -20°C to +65°C / 5% to 60% RH
Operating Voltage	: 12V DC ±10%

Electrical Specifications

Insulation Resistance	: 500V DC, ≥50MΩ
Dielectric Strength	: 700V AC 50/60Hz
Insulation Level	: Class B (130°C)
Wattage	: 9W (12V DC, R=16Ω ±10%)
Stroke-Force	: 6mm thrust: ≥50gf (12V DC)
Work Cycle	: Pass 0.05 seconds, break 0.05 seconds, max. power-on time, 10 seconds (ED 50%)
Temperature Rise	: ≤80°C (12V DC, 0.05 seconds off for 0.05 seconds, no load)
Response Time	: ≥50mS (12V DC, S=10.5mm, no load)
Leading strength	: 1Kgf-30 seconds
Life	: ≥500,000 times (12V DC, pass for 0.05 seconds, break 0.05 seconds for one time, load (institution)

Diagram



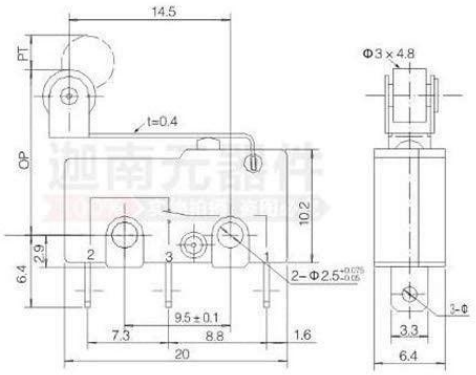
Dimensions : Millimetres

Part Number Table

Description	Part Number
Solenoid Lock, 12V DC, 9W, 50gf	MP001162

Lampiran 11. Datasheet Limit Switch KW12-3

动作力	OF max.	70g(0.7N)
回复力	RF min.	10g(0.1N)
预行程	PT max.	2.6mm
超行程	OT min.	0.8mm
差行程	MD max.	0.8mm
动作位置	OP	14.5±0.5mm



□ 性能参数 Specification

项　目 Item		参数值 Data
额定负荷 Rating load		5A/125VAC,3A/250VAC
操作速度 Operation speed		0.1mm~1m/s
操作频率 Operation frequency	机　械 Mechanical	400次/分 400times/min
	电　气 Electrical	30次/分 30times/min
绝缘电阻 Insulation resistance		100mΩ以上(Above) DC500V
接触电阻 Contact resistance		30mΩ初值 initial value)
介质耐压 Withstand voltage	非 连 线 Non-connection wire	1000VAC
	各端子间 Every terminal	1500VAC
振　动 Vibration	误 动 作 Misoperation	10~55Hz 复振幅 Amplitude 1.5mm
冲　击 Impact	耐　　久 Durable	1000m/s²
	误 动 作 Misoperation	300m/s²
寿　命 Life	电　　气 Electrical	100,000次以上 above
	机　　械 Mechanical	1,000,000次以上 above
防护等级 Protection degree		IP40
使用温度 Operating temperature		-25~+80℃
使用湿度 Operating humidity		<85%