

KAPITEL IV

ERGEBNISSE UND DISKUSSION

1.1 Ergebnisse der Werkzeugkonstruktion

Aus den Ergebnissen der durchgeführten Untersuchung wurde ein Motorrad-Sicherheitsschließsystem mit einem Fingerabdrucksensor und Arduino Uno hergestellt, wie auf dem folgenden Bild zu sehen ist:



Abbildung 4.1: Vorderes Sicherheitsschlosssystem



Abbildung 4.2: Rechtes Sicherheitsschlosssystem



Abbildung 4.3: Linkes Sicherheitsschloss System



Abbildung 4.4: Inneres Sicherheitsschlosssystem

Beschreibung des Bildes, wenn es von der Blackbox aus betrachtet wird:

1. Blackbox.
2. Fingerabdrucksensor
3. Relaisreiber
4. Mindestsystem *Arduino Uno*

1.2 Testwerkzeuge

Bevor ein System mit einem anderen System zusammengebaut und zu einem Teil eines organisierten Systems vereint wird, das als Motorrad-Sicherheitssystem mit dem Arduino Uno-Steuerungssystem fungiert, besteht die erste Phase darin, jedes System zu entwerfen, da das Testen ein Schritt ist, um das Ausmaß der Eignung zwischen dem Design und der Realität des hergestellten Werkzeugs herauszufinden, ob es wie erwartet ist oder nicht. Das Testen von Tools ist auch nützlich, um das Leistungsniveau des Tools herauszufinden. Nach dem Testen sollten Sie messen oder analysieren, was getestet wird, um den Erfolg der in dieser Arbeit hergestellten Werkzeuge herauszufinden. An jedem der Werkzeugblöcke wird getestet, um herauszufinden, wie das entworfene Werkzeug funktioniert. Dieser Test beinhaltet:

1. Fingerabdrucksensortest mit *Arduino Uno*
2. Relaisreibertest
3. Lautsprecher und MP3-Player testen
4. LCD-Test
5. Modultest
6. Gesamttool

1.2.1 Testen der Fingerabdrucksensorschaltung mit Arduino Uno.

Beim Testen dieses Fingerabdrucksensors verbinden sich Verfasser mit Arduino Uno, nämlich der Erstellung einer Programmauflistung und Eingabe (Upload) in das Arduino Uno-Minimumsystem. Für das Bild der Schaltung, die mit dem Infrarotsensor gemacht werden soll, ist Abbildung 4.5 unten zu sehen:





Abbildung 4.5: Testen des Fingerabdrucksensors mit Arduino Uno.

Tabelle 4.1: Messergebnisse des Fingerabdrucksensors und des Arduino Uno.

Experiment	Spannung
1	3,291 Volt
2	3,292 Volt
3	3,292 Volt
4	3,292 Volt

Formel zur Messung der Durchschnittsspannung :

$$V_{Rata-rata} = \frac{\sum V}{n} = \frac{3,291 + 3,292 + 3,292 + 3,292}{4}$$

$$\frac{13,167}{4} = 3,29175 \text{ Volt}$$

Information:

Aus der obigen Tabelle kann geschlossen werden, dass der Durchschnittswert 3,29175 Volt wert ist. Was die gemessene Spannung als Instabilität betrifft, so kommt der Faktor, der die Instabilität der Versorgungsspannung beeinflusst, von der sich ändernden PLN-Frequenz. Dies ist jedoch kein Problem, solange es den Bereich von $\pm 5\%$ nicht überschreitet.

1.2.2 Testen des Relaisreibern

Damit der Relaisreiber entsprechend den Erwartungen arbeiten kann, ist es notwendig zu testen, nämlich durch Durchführung eines Versuchs, um die Ergebnisse des Relaisreibers anzuzeigen.

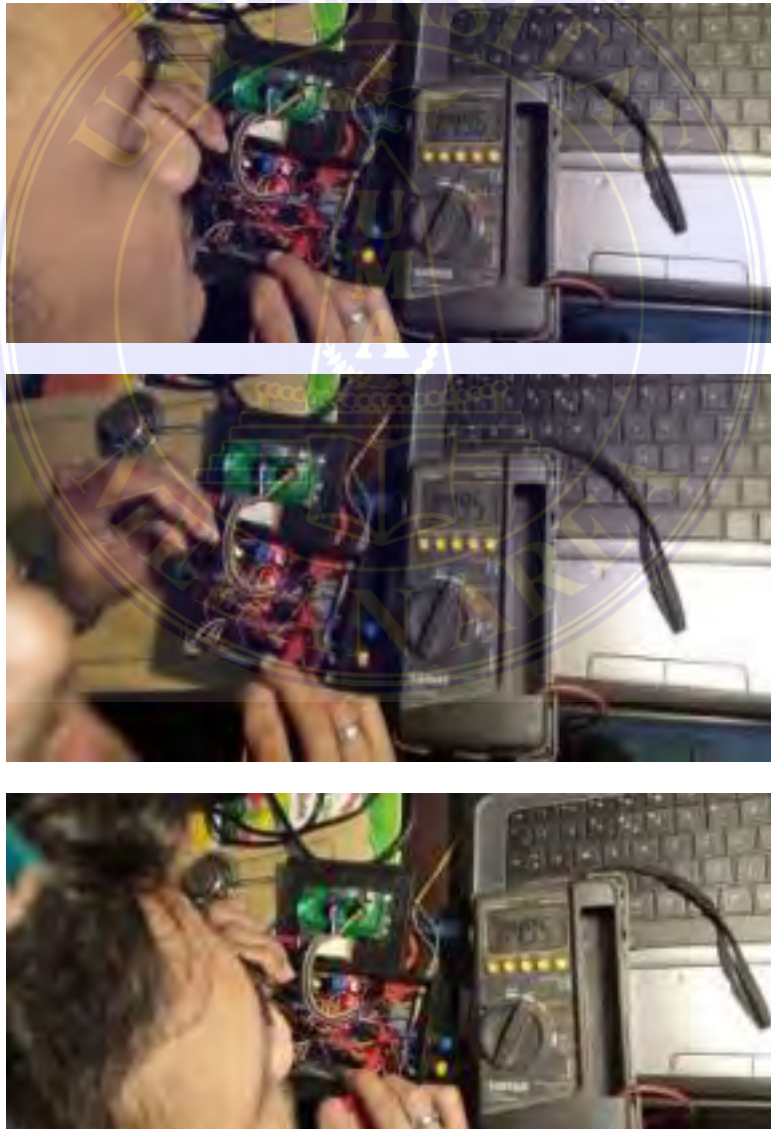


Abbildung 4.6: Prüfung Relaisreibern

Tabelle 4.2: Ergebnisse der Prüfung von Relaisrteibern

Experiment	Spannung
1	4,95 Volt
2	4,96 Volt
3	4,95 Volt
4	4,95 Volt

Formel zur Messung der Durchschnittsspannung:

$$V_{Rata-rata} = \frac{\sum V}{n} = \frac{4,95 + 4,96 + 4,95 + 4,95}{4}$$

$$= \frac{19,81}{4} = 4,9525 \text{ Volt}$$

Information:

Aus der obigen Tabelle lässt sich schließen, dass der Durchschnittswert 4,9525 Volt wert sein kann. Was die gemessene Spannung als Instabilität betrifft, so kommt der Faktor, der die Instabilität der Versorgungsspannung beeinflusst, von der sich ändernden PLN-Frequenz. Dies ist jedoch kein Problem, solange es den Bereich von +- 5% nicht überschreitet.

Nach Durchführung eines Versuchs kann die Relaisrteiberschaltung verwendet werden und nach Belieben. In der Relaisrteiberschaltung benötigt eine Arbeitsspannungsquelle von 5 VDC, die Spannung wird von der Stromversorgung erhalten.

1.2.3 Tests von Lautsprechern und MP3-Playern

Damit der Lautsprecher und der MP3-Player entsprechend den Erwartungen funktionieren, ist es notwendig zu testen, nämlich durch Durchführung eines Versuchs, um die Ergebnisse des Lautsprechers und des MP3-Players anzuzeigen..



Abbildung 4.7: Testen von Lautsprechern und MP3-Playern

Tabelle 4.3: Messergebnisse der Lautsprecher und MP3-Player

Experiment	Spannung
1	4,96 Volt
2	4,96 Volt
3	4,96 Volt
4	4,97 Volt

Formel zur Messung der Durchschnittsspannung:

$$V_{Rata-rata} = \frac{\sum V}{n} = \frac{4,96 + 4,96 + 4,96 + 4,97}{4}$$

$$= \frac{19,85}{4} = 4,9625 \text{ Volt}$$

Information:

Aus der obigen Tabelle kann geschlossen werden, dass der Durchschnittswert bekanntermaßen 4,9625 Volt beträgt. Was die gemessene Instabilität der gemessenen Spannung betrifft, so kommt der Faktor, der die Instabilität der Stromversorgungsspannung beeinflusst, von der sich ändernden PLN-Frequenz. Dies ist jedoch kein Problem, solange es den Bereich von +/- 5% nicht überschreitet.

Nach dem Testen . Der Lautsprecher und der MP3-Player funktionieren gut bei einer Spannung von 5 VDC.

1.2.4 LCD-Tests

Damit das LCD entsprechend den Erwartungen funktioniert, ist es notwendig zu testen, nämlich durch die Durchführung eines Versuchs, um die Ergebnisse des LCD anzuzeigen.



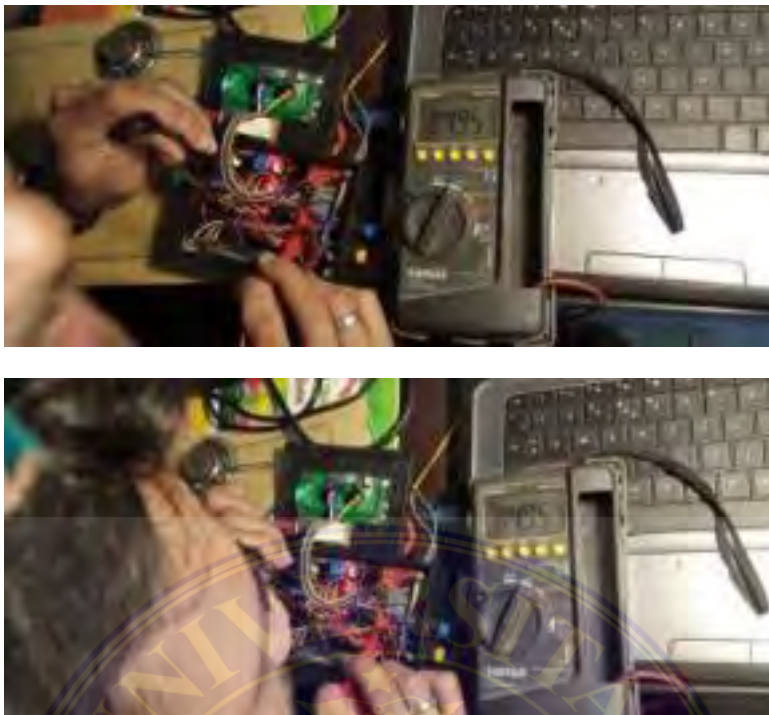


Abbildung 4.8: LCD-Tests

Tabelle 4.4: LCD-Messergebnisse

Experiment	Spannung
1	4,95 Volt
2	4,96 Volt
3	4,95 Volt
4	4,95 Volt

Formel zur Messung der Durchschnittsspannung:

$$\begin{aligned}
 V_{Rata-rata} &= \frac{\sum V}{n} = \frac{4,96 + 4,95 + 4,95 + 4,95}{4} \\
 &= \frac{19,81}{4} = 4,9525 \text{ Volt}
 \end{aligned}$$

Information:

Aus der obigen Tabelle lässt sich schließen, dass der Durchschnittswert 4,9525 Volt wert sein kann. Was die gemessene Spannung als Instabilität betrifft, so kommt der Faktor, der die Instabilität der Versorgungsspannung beeinflusst, von der sich ändernden PLN-Frequenz. Dies ist jedoch kein Problem, solange es den Bereich von +/- 5% nicht überschreitet. Nach dem Testen kann das LCD je nach Bedarf ordnungsgemäß funktionieren.

1.2.5 Die Modulprüfung

Dieser Test wird durchgeführt, indem das Arduino Uno-Modul mit dem Fingerabdrucksensor verbunden wird, der durch die Beine der jeweiligen Sensorpins montiert wird, die mit den Arduino Uno-Pins verbunden sind. Mit diesem Test soll überprüft werden, ob die Eingabedaten (Ein- und Ausgang (Ausgabe) entsprechend der Arbeitsbeschreibung des Systems funktionieren können. Hier ist Abbildung 4.9, die das Arduino uno-Programm mit Fingerabdrucksensor auflistet.

Testschritte des Arduino Uno-Moduls:

1. Man Öffnet die arduino IDE Software.
2. Als nächstes erscheint automatisch die anfängliche Sketchanzeige.
3. Man beginnt mit dem Schreiben eines Programms, um Sensorlesedaten wie folgt zu verarbeiten.

```
#include <LiquidCrystal.h>
#include <SoftwareSerial.h>
#include <EEPROM.h>
#include <Adafruit_Fingerprint.h>
#include "DFRobotDFPlayerMini.h"
LiquidCrystal lcd(13,12,11,10,9,8); //RS,E,D4,D5,D6,D7
SoftwareSerial mySerial(A3,A2);
Adafruit_Fingerprint finger = Adafruit_Fingerprint(&mySerial);
DFRobotDFPlayerMini Mp3Player;
```

```
#define relay_1 A0
#define relay_2 A1
#define tombol A4
```

```
int getFingerprintIDez();
```

```
uint8_t id;
uint8_t getFingerprintEnroll();
byte state,salah;

void setup() {
  lcd.begin(16,2);
  Serial.begin(9600);
  mySerial.begin(57600);
  pinMode(relay_1,OUTPUT);
  pinMode(relay_2,OUTPUT);
  digitalWrite(relay_1,HIGH); //relay 1 gelöscht
  digitalWrite(relay_2,HIGH); //relay 2 gelöscht
  pinMode(tombol,INPUT_PULLUP);

if(finger.verifyPassword()) {
  lcd.setCursor(0,0); lcd.print("FingerPrint OKE!");
  delay(2000);
}
else {
  lcd.setCursor(0,0); lcd.print("FingerPrnt ERROR");
  while (1);
}
if(!Mp3Player.begin(Serial)) { //Use softwareSerial to communicate with mp3.
  lcd.setCursor(0,1); lcd.print("DFPlayer ERROR..");
  while(true);
}
lcd.setCursor(0,1); lcd.print("DFPlayer online.");
Mp3Player.volume(30); //Set volume value (0~30).
Mp3Player.EQ(DFPLAYER_EQ_NORMAL);
Mp3Player.outputDevice(DFPLAYER_DEVICE_SD);
Mp3Player.enableDAC(); //Enable On-chip DAC
//Mp3Player.play(1); //Sidik Jari Tidak Cocok
//Mp3Player.play(2); //Coba Lagi Boss...
//Mp3Player.play(3); //Halo Bos...
//Mp3Player.play(4); //Hidupkan Mesin
id=EEPROM.read(1);
delay(1000);
}

void loop() {
  lcd.setCursor(0,0); lcd.print("=SECURITi MOTOR=");
  lcd.setCursor(0,1); lcd.print("Scan Your Finger");
```

```
if(digitalRead(tombol)==0) {
  while(1) {
    lcd.clear();
    lcd.setCursor(0,0); lcd.print("REGISTRASI SIDIK");
    lcd.setCursor(0,1); lcd.print("JARI : "); lcd.print(id);
    delay(2000); lcd.clear();
    lcd.setCursor(0,0); lcd.print("Registrsi ID="); lcd.print(id);
    while (! getFingerprintEnroll() );
  }
}
getFingerprintIDez();
delay(50);      //don't ned to run this at full speed.
}

uint8_t getFingerprintID() {
  uint8_t p = finger.getImage();
  switch (p) {
    case FINGERPRINT_OK:          break;
    case FINGERPRINT_NOFINGER:    return p;
    case FINGERPRINT_PACKETRECI VEERR: return p;
    case FINGERPRINT_IMAGEFAIL:  return p;
    default:                      return p;
  }
  // OK success!
  p = finger.image2Tz();
  switch (p) {
    case FINGERPRINT_OK:          break;
    case FINGERPRINT_IMAGEMESS:   return p;
    case FINGERPRINT_PACKETRECI VEERR: return p;
    case FINGERPRINT_FEATUREFAIL: return p;
    case FINGERPRINT_INVALIDIMAGE: return p;
    default:                      return p;
  }
  // OK converted!
  p = finger.fingerFastSearch();
  if (p == FINGERPRINT_OK) {}
  else if (p == FINGERPRINT_PACKETRECI VEERR) return p;
  else if (p == FINGERPRINT_NOTFOUND) return p;
  else return p;
  // found a match!
  //Serial.print("ID-"); Serial.println(finger.fingerID);
}
```

```
// returns -1 if failed, otherwise returns ID #
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_OK) {
        //Serial.println("#error");
        lcd.setCursor(0,0); lcd.print("---SIDIK JARI---");
        lcd.setCursor(0,1); lcd.print("--TIDAK SESUAI--");
        Mp3Player.play(2); //1
        delay(3000);
        salah = salah + 1;
        if(salah>3) {
            digitalWrite(relay_1,HIGH);
            Mp3Player.play(1);
        }
        return -1;
    }

    match: // SIDIK JARI COCOK
    //Serial.print("ID-"); Serial.println(finger.fingerID);
    lcd.setCursor(0,0); lcd.print("SIDIK JARI COCOK");
    lcd.setCursor(0,1);
    if(state==0) {
        lcd.print("-KUNCI TERBUKA!-");
        Mp3Player.play(3);
        digitalWrite(relay_1,LOW); // Relay CDI Aktif.
        state=1;
    }
    else if(state==1) {
        lcd.print("-STARTER MOTOR!-");
        Mp3Player.play(4);
        digitalWrite(relay_2,LOW); delay(2000); // Relay Starter Aktif 2 detik.
        digitalWrite(relay_2,HIGH);
        state=2;
        salah=0;
    }
    else {
        lcd.print("MOTOR SDH AKTIF!");
    }
    delay(3000);
}
```

```
return finger.fingerID;

//=====FINGERABDRUCK-REGISTRIERUNG =====//
uint8_t getFingerprintEnroll() {
    int p = -1;
    lcd.setCursor(0,1); lcd.print("Letakkan Jari...");
    while (p != FINGERPRINT_OK) {
        p = finger.getImage();
        switch (p) {
            case FINGERPRINT_OK:          lcd.setCursor(0,1); lcd.print("Image
Taken... "); break;
            case FINGERPRINT_NOFINGER:
break;
            case FINGERPRINT_PACKETRECIIEVEERR: lcd.setCursor(0,1);
lcd.print("Komunikasi Error"); break;
            case FINGERPRINT_IMAGEFAIL:      lcd.setCursor(0,1);
lcd.print("Imaging Error.. "); break;
            default:                        lcd.setCursor(0,1); lcd.print("Unknown Error.. ");
break;
        }
    }
    // OK success!
    p = finger.image2Tz(1);
    switch (p) {
        case FINGERPRINT_OK:          lcd.setCursor(0,1); lcd.print("Image
Converted."); break;
        case FINGERPRINT_IMAGEMESS:      lcd.setCursor(0,1);
lcd.print("Image too Messy."); return p;
        case FINGERPRINT_PACKETRECIIEVEERR: lcd.setCursor(0,1);
lcd.print("Komunikasi Error"); return p;
        case FINGERPRINT_FEATUREFAIL:      lcd.setCursor(0,1); lcd.print("No
Feature Found"); return p;
        case FINGERPRINT_INVALIDIMAGE:      lcd.setCursor(0,1); lcd.print("No
Feature Found"); return p;
        default:                        lcd.setCursor(0,1); lcd.print("Unknown Error...");
return p;
    }

    lcd.setCursor(0,1); lcd.print("Angkat Jarinya.."); delay(2000);
    p = 0;
    while (p != FINGERPRINT_NOFINGER) { p = finger.getImage(); }
    //Serial.print("ID "); Serial.println(id);
    p = -1;
}
```

```
lcd.setCursor(0,1); lcd.print("Letakkan Jari lg");
while (p != FINGERPRINT_OK) {
  p = finger.getImage();
  switch (p) {
    case FINGERPRINT_OK:          lcd.setCursor(0,1); lcd.print("Image
Taken.... "); break;
    case FINGERPRINT_NOFINGER:
break;
    case FINGERPRINT_PACKETRECIIEVEERR: lcd.setCursor(0,1);
lcd.print("Komunikasi Error"); break;
    case FINGERPRINT_IMAGEFAIL:      lcd.setCursor(0,1);
lcd.print("Imaging Error.. "); break;
    default:                        lcd.setCursor(0,1); lcd.print("Unknown Error.. ");
break;
  }
}
delay(2000);
// OK success!
p = finger.image2Tz(2);
switch (p) {
  case FINGERPRINT_OK:          lcd.setCursor(0,1); lcd.print("Image
Converted."); break;
  case FINGERPRINT_IMAGEMESS:   lcd.setCursor(0,1);
lcd.print("Image too Messy."); return p;
  case FINGERPRINT_PACKETRECIIEVEERR: lcd.setCursor(0,1);
lcd.print("Komunikasi Error"); return p;
  case FINGERPRINT_FEATUREFAIL:  lcd.setCursor(0,1); lcd.print("No
Feature Found"); return p;
  case FINGERPRINT_INVALIDIMAGE: lcd.setCursor(0,1); lcd.print("No
Feature Found"); return p;
  default:                      lcd.setCursor(0,1); lcd.print("Unknown Error...");
return p;
}

// OK converted!
delay(2000); lcd.clear();
lcd.setCursor(0,1); lcd.print("Creating Model..");
p = finger.createModel();
if (p == FINGERPRINT_OK)          { lcd.setCursor(0,1); lcd.print("Prints
Matched!!"); }
```

```
else if (p == FINGERPRINT_PACKETRECIEVEERR) { lcd.setCursor(0,1);  
lcd.print("Komunikasi Error"); return p; }  
else if (p == FINGERPRINT_ENROLLMISMATCH) { lcd.setCursor(0,1);  
lcd.print("Finger Tdk Cocok"); return p; }  
else { lcd.setCursor(0,1); lcd.print("Unknown Error...");  
return p; }  
delay(2000); lcd.clear();  
lcd.setCursor(0,0); lcd.print("ID #"); lcd.print(id); p =  
finger.storeModel(id);  
if (p == FINGERPRINT_OK) { lcd.setCursor(0,1);  
lcd.print("Success Stored!!"); }  
else if (p == FINGERPRINT_PACKETRECIEVEERR) { lcd.setCursor(0,1);  
lcd.print("Komunikasi Error"); return p; }  
else if (p == FINGERPRINT_BADLOCATION) { lcd.setCursor(0,1);  
lcd.print("Error Stored Loc"); return p; }  
else if (p == FINGERPRINT_FLASHERR) { lcd.setCursor(0,1);  
lcd.print("Error Write Flas"); return p; }  
else { lcd.setCursor(0,1); lcd.print("Unknown Error...");  
return p; }  
delay(2000);  
lcd.setCursor(0,0); lcd.print("REGISTRASI SIDIK");  
lcd.setCursor(0,1); lcd.print("=JARI BERHASIL!=");  
EEPROM.write(1,id+1);  
while(1) {}  
}
```

4. Klickt man auf Sketch → Verify. Ein Dialogfeld zum Speichern des Programms erscheint
5. Ladet man das erstellte Programm über ein USB-Kabel in Arduino Uno hoch, indem Sie auf Sketch→Upload

1.2.6 Gesamtbasistest

In dieser Phase führte die Studie einen Gesamtversuch des Werkzeugs durch, der alle Werkzeuge umfasste, die ein Motorradsicherheitssystem mit einem Fingerabdrucksensor bildeten. Das Testmuster besteht darin, auf den Fingerabdrucksensor zu achten, der bei Eingabe ein- oder ausgeschaltet ist Das vom Fingerabdrucksensor erkannte Fingerabdruckmuster.



Abbildung 4.9: Fingerabdruck zur Stromversorgung des CDI



Abbildung 4.10: Fingerabdruck für Motorstarter

Um die Analyse der Diskussion über das Testen des Tools als Ganzes zu erleichtern, finden Sie im Folgenden Tabelle 4.5, in der die Gesamttestergebnisse angezeigt werden:

Motorrad Zustand	Zustand des Fingerabdruckensors	Spannung	Relais-Treiber	Information
Motorrad startet	geeignet	5 vdc	Startet	Aktiv
Tote Motorrad	Ungeeignet	0 vdc	Tote	Inaktiv

Information:

Wenn der Fingerabdruckeingang übereinstimmt, beträgt die eingehende Eingangsspannung 5 Volt und der Relaisreiber schaltet sich ein, damit das Motorrad starten kann. Umgekehrt Wenn der Fingerabdruckeingang nicht übereinstimmt, beträgt die eingehende Eingangsspannung 0 Volt und der Relaisreiber schaltet sich nicht ein, so dass das Motorrad nicht starten kann.

