# LAMPIRAN

Lampiran Alat 1. Gambar Alat



Monitoring Progam Pzem

```
#include <PZEM004Tv30.h>

#include <Wire.h> // i2C Conection Library

#include <LiquidCrystal_I2C.h> //i2C LCD Library

#define r1 5

#define r2 6

#define r3 7

PZEM004Tv30 pzem(11, 12);

LiquidCrystal_I2C lcd(0x27, 16, 2);

int bton = 8;

int btoff = 9;
```

```cpp
int btonx = 0;

int btoffx = 0;

int ax;

int data1 = 1;

int data2 = 2;

float a [12];

float target1 = 0.80 ;

float target2 = 0.95 ;

float current;

float pf;

float voltage;

float power;

float A , B, V, p;

int op;

void setup() {

 Serial.begin(9600);

 lcd.begin();

 lcd.clear();

 lcd.noCursor();

 pinMode(r1, OUTPUT);

 pinMode(r2, OUTPUT);

 pinMode(r3, OUTPUT);

 pinMode(bton,  INPUT_PULLUP);

 pinMode(btoff, INPUT_PULLUP);

 STOP();
```

```
  op = 0;

 }

 void loop()

 { PZEM();

  scan();
```

Progam Scan Relay

```
#include <PZEM004Tv30.h>

#include <Wire.h> // i2C Conection Library

#include <LiquidCrystal_I2C.h> //i2C LCD Library

#define r1 5

#define r2 6

#define r3 7

PZEM004Tv30 pzem(11, 12);

LiquidCrystal_I2C lcd(0x27, 16, 2);

int bton = 8;

int btoff = 9;

int btonx = 0;

int btoffx = 0;

int ax;

int data1 = 1;

int data2 = 2;

float a [12];

float target1 = 0.80 ;
```

```
float target2 = 0.95 ;

float current;

float pf;

float voltage;

float power;

float A , B, V, p;

int op;

void setup() {

  Serial.begin(9600);

  lcd.begin();

  lcd.clear();

  lcd.noCursor();

  pinMode(r1, OUTPUT);

  pinMode(r2, OUTPUT);

  pinMode(r3, OUTPUT);

  pinMode(bton,  INPUT_PULLUP);

  pinMode(btoff, INPUT_PULLUP);

  STOP();

  op = 0;

}

void loop() {

  PZEM();

  scan();

}
```

## Progam Sensor Pzem 004t

```
float a [12];

float target1 = 0.80 ;

float target2 = 0.95 ;

float current;

float pf;

float voltage;

float power;

float A , B, V, p;

int op;

void setup() {

  Serial.begin(9600);

  lcd.begin();

  lcd.clear();

  lcd.noCursor();

  pinMode(r1, OUTPUT);

  pinMode(r2, OUTPUT);

  pinMode(r3, OUTPUT);

  pinMode(bton,  INPUT_PULLUP);

  pinMode(btoff, INPUT_PULLUP);

  STOP();

  op = 0; }

void loop() {

PZEM();

  scan();
```

Lampiran. Gambar Arduino uno

(Sumber ; Iot-Spot.com)

Data Shet Arduino Uno

| Microcontroller | ATmega328 |
|---|---|
| Operating Voltage | 5V |
| Input Voltage (recommended) | 7-12V |
| Input Voltage (limits) | 6-20V |
| Digital I/O pins | 14(of which 6 provide PWM output) |
| Analog Input pins | 6 |
| Dc Curren per I/O pin | 40mA |
| Dc Curren for 3,3 v pin | 50mA |
| Flash Memory | 32KB (ATmega328) of which 0.5KB used by bootloader |
| SRAM | 2KB (ATmega328) |
| EEPROM | 1KB (ATmega328) |
| Clock Speed | 16 MHz |

| | |
|---|---|
| Length | 68.6 mm |
| Width | 53.4 mm |
| Weight | 25 g |



Lampiran . Gambar Sensor Pzem 004t
(Sumber:nn.digital.com)

DataShet Pzem 004t

1. Voltage

Measuring

range:80～260V

Resolution: 0.1V

Measurement accuracy: 0.5% 1.2 Current

Measuring range: 0～10A(PZEM-004T-10A); 0～100A(PZEM-004T-100A)

Starting measure current: 0.01A(PZEM-004T-10A); 0.02A(PZEM-

004T-100A) Resolution: 0.001A 1.2.4 Measurement accuracy: 0.5%

1. Active power Measuring range: 0～2.3kW(PZEM-004T-10A);

0～23kW(PZEM-004T-100A)

Starting measure power: 0.4W

Resolution: 0.1W

Display format: ＜1000W, it display one decimal, such as: 999.9W

≥1000W, it display only integer, such as: 1000W

Measurement accuracy: 0.5%

2. Power factor

Measuring range:

0.00～1.00 Resolution:

0.01

Measurement accuracy: 1%

3. Frequency

Measuring range:

45Hz～65Hz Resolution:

0.1Hz

Measurement accuracy: 0.5% 1.6

4. Active energy

Measuring range: 0～9999.99kWh

Resolution: 1Wh 1.6.3 Measurement accuracy: 0.5%

Display format: ＜10kWh, the display unit is Wh(1kWh=1000Wh), such as: 9999Wh

≥10kWh, the display unit is kWh, such as: 9999.99kWh

Reset energy: use software to reset.

5. Over power alarm Active power threshold can be set, when the measured active power exceeds the threshold, it can alarm

6. Communication interface RS485 interface。