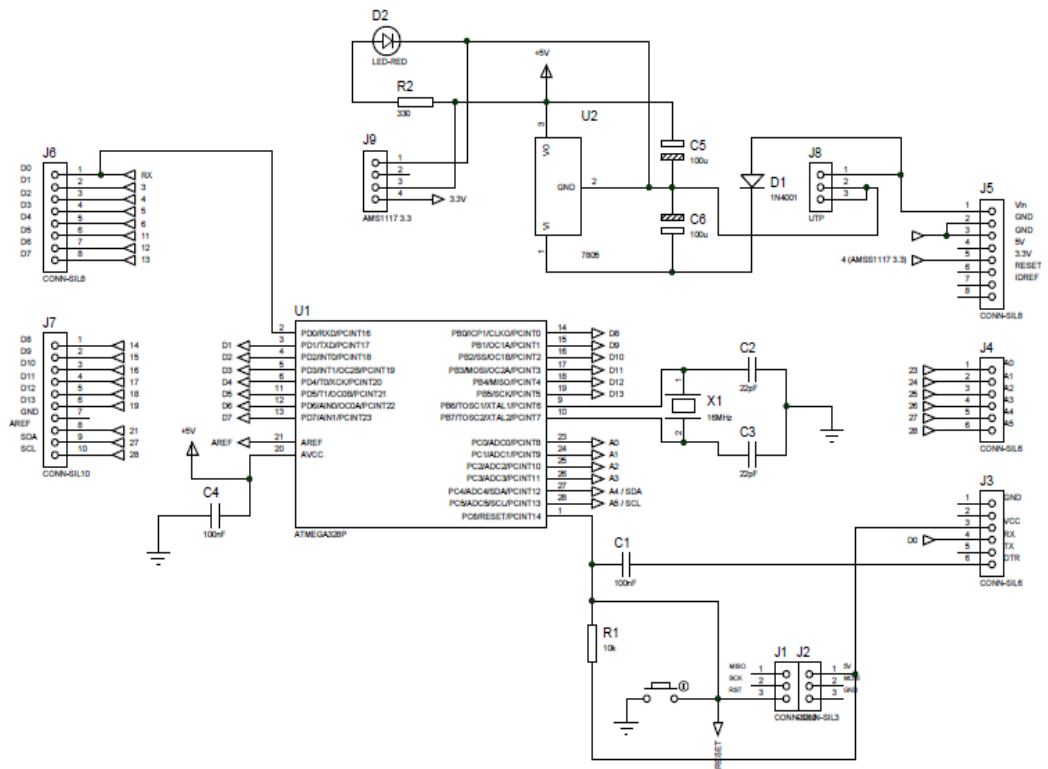
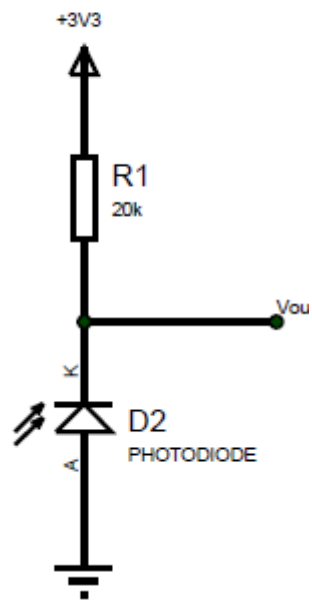


LAMPIRAN

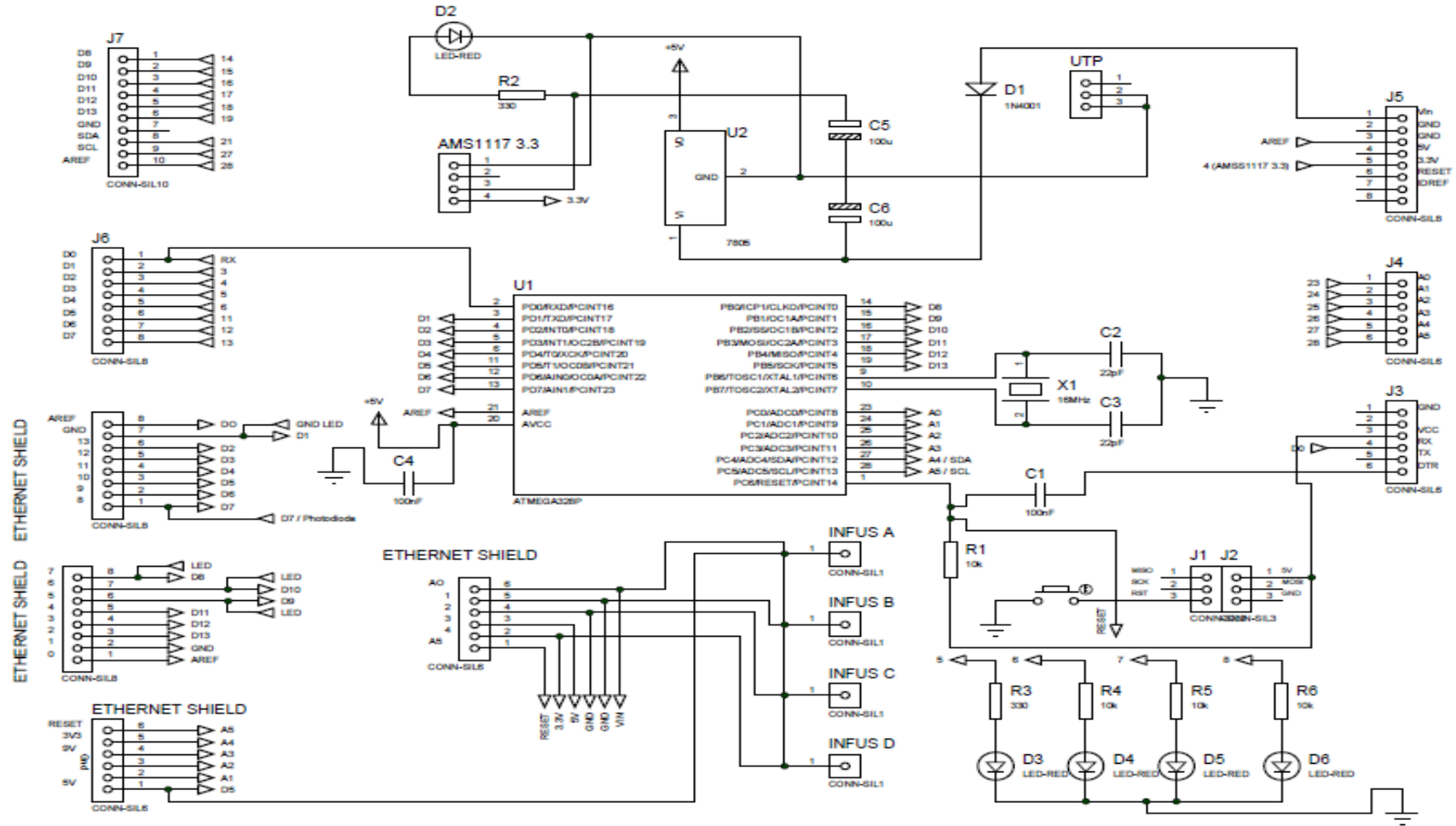
Rangkaian Skematik Sistem Minimum



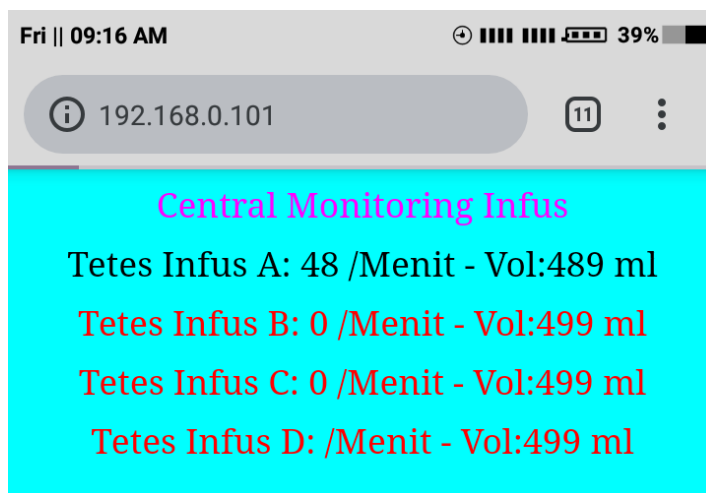
Rangkaian Skematik Sensor

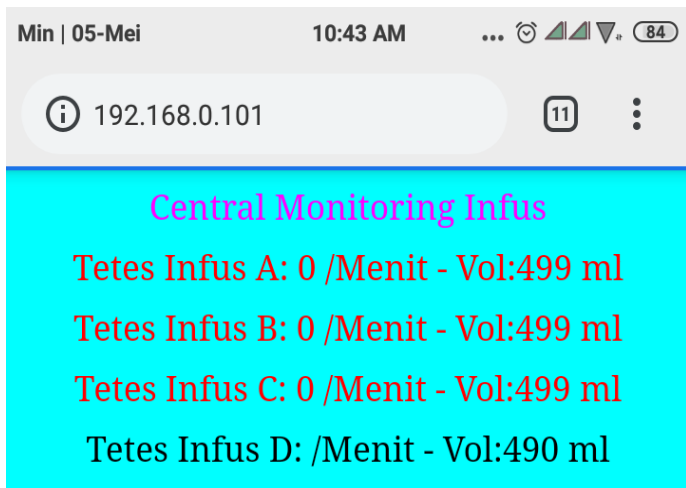


Rangkaian Keseluruhan



1. Gambar tampilan di gelas ukur, infus pump dan tampilan web server pada infus A, B, C, dan D saat kondisi cairan 10 ml.





2. Gambar tampilan di digelas ukur, infus pump dan tampilan web server pada infus A, B, C, dan D saat kondisi cairan 50 ml.



Fri || 09:46 AM 📶 35%

192.168.0.101 11

Central Monitoring Infus

Tetes Infus A: 49 /Menit - Vol:449 ml
Tetes Infus B: 0 /Menit - Vol:499 ml
Tetes Infus C: 0 /Menit - Vol:499 ml
Tetes Infus D: /Menit - Vol:499 ml

192.168.0.101 11

Central Monitoring Infus

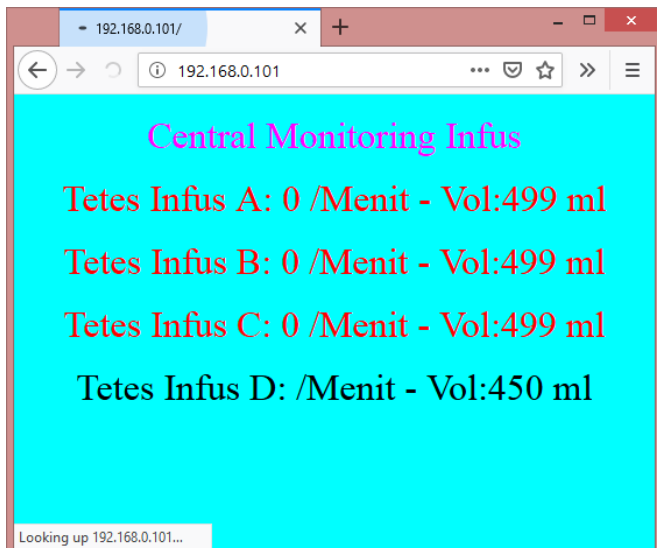
Tetes Infus A: 0 /Menit - Vol:499 ml
Tetes Infus B: 34 /Menit - Vol:449 ml
Tetes Infus C: 0 /Menit - Vol:499 ml
Tetes Infus D: /Menit - Vol:499 ml

Sab | 04-Mei 11:46 PM 📶 15

192.168.0.101 11

Central Monitoring Infus

Tetes Infus A: 0 /Menit - Vol:499 ml
Tetes Infus B: 0 /Menit - Vol:499 ml
Tetes Infus C: 51 /Menit - Vol:449 ml
Tetes Infus D: /Menit - Vol:499 ml



PROGRAM KESELURUHAN

```
//referensi
https://playground.arduino.cc/Main/PinChangeInterrupt

#include <TimerOne.h> // library timer1

// spi library

#include <SPI.h>

// ethernet library

#include <Ethernet.h>

// parameter

byte mac[] = { 0xDE, 0xAD, 0xBE, 0xEF, 0xFE, 0xED }; //
physical mac address

byte ip[] = { 192, 168, 0, 101 }; // ip address

byte gateway[] = { 192, 168, 0, 254 }; // ip router

byte subnet[] = { 255, 255, 255, 0 }; // subnet mask

EthernetServer server(80); // server port

// pin sensor

#define sensor1 A0

#define sensor2 A1

#define sensor3 A2

#define sensor4 A3

// pin tombol

#define s1 2

#define s2 3

// pin led
```

```

#define led1 5

#define led2 6

#define led3 7

#define led4 8

bool lock1=0,lock2=0,lock3=0,lock4=0;// variable pengunci,
untuk memodifikasi dari mode CHANGE ke FALLING

int tandal=0,tanda2=0,tanda3=0,tanda4=0;

int time1=0,time2=0,time3=0,time4=0;

int timeok1,timeok2,timeok3,timeok4;

int tetes1,tetes2,tetes3,tetes4;

long int ml1,ml2,ml3,ml4;

long int counter1,counter2,counter3,counter4;

// mengaktifkan mode interrupt external pada masing2 port
secara independen

void pciSetup(byte pin) {

    *digitalPinToPCMSK(pin) |=bit(digitalPinToPCMSKbit(pin));
    // enable pin

    PCIFR  |= bit (digitalPinToPCICRbit(pin)); // clear
any outstanding interrupt

    PCICR  |= bit (digitalPinToPCICRbit(pin)); // enable
interrupt for the group
}

// handle pin change interrupt for A0 to A5

ISR (PCINT1_vect) {

    // mofikasi ke mode falling

```

```

if(digitalRead(sensor1)==HIGH&&lock1==0){

    counter1++;

    digitalWrite(led1,1);

    lock1=1;    // mengunci

    tandal1++; // tanda mencacah

    if(tandal1==2){ // jika tanda bernilai 2 kemudian
time1>100 simpan hasil dari pencacah timer

if(time1>=100){timeok1=time1;tetes1=((float)1000/timeok1)
*60;}// ketelitian baca 0.1 detik dan rubah ke tetes per
menit

        time1=0; // reset timer

        tandal1=1; // tanda bernilai 1 untuk mengjinkan timer
mencacah

    }

}

if(digitalRead(sensor2)==HIGH&&lock2==0){

    counter2++;

    digitalWrite(led2,1);

    lock2=1;

    tanda2++;

    if(tanda2==2){

if(time2>=100){timeok2=time2;tetes2=((float)1000/timeok2)
*60;}

        time2=0;

        tanda2=1;

    }

}

```

```
if (digitalRead(sensor3)==HIGH&&lock3==0) {  
  
    counter3++;  
  
    digitalWrite(led3,1);  
  
    lock3=1;  
  
    tanda3++;  
  
    if(tanda3==2){  
if (time3>=100) {timeok3=time3;tetes3=((float)1000/timeok3)  
*60;}  
  
    time3=0;  
  
    tanda3=1;  
  
    }  
  
    }  
  
if (digitalRead(sensor4)==HIGH&&lock4==0) {  
  
    counter4++;  
  
    digitalWrite(led4,1);  
  
    lock4=1;  
  
    tanda4++;  
  
    if(tanda4==2){  
if (time4>=100) {timeok4=time4;tetes4=((float)1000/timeok4)  
*60;}  
  
    time4=0;  
  
    tanda4=1;  
  
    }  
  
    }  
  
}
```

```

// timer1 interrupt di jalankan

void timerIsr() {

    if(tanda1==1)time1++; // jika tanda bernilai 1 timer
mencacah

    if(tanda2==1)time2++;

    if(tanda3==1)time3++;

    if(tanda4==1)time4++;

}

void reset_data(){

    if(digitalRead(sensor1)==LOW) lock1=0; // reset pengunci
lock = 0 berarti unlock

    if(digitalRead(sensor2)==LOW) lock2=0;

    if(digitalRead(sensor3)==LOW) lock3=0;

    if(digitalRead(sensor4)==LOW) lock4=0;

    if(time1>3000){ // time out 3000detik jika tidak baca

// reset semua variavle pemroses

time1=0;

tanda1=0;

timeok1=0;

tetes1=0;

    }

if(time2>3000){

// reset semua variavle pemroses

time2=0;

tanda2=0;

timeok2=0;

tetes2=0; }

```

```
if(time3>3000){
    // reset semua variabel pemroses
    time3=0;
    tanda3=0;
    timeok3=0;
    tetes3=0;
}
if(time4>3000){
    // reset semua variabel pemroses
    time4=0;
    tanda4=0;
    timeok4=0;
    tetes4=0;
}
delay(1); // jeda 1 milidetik
}
// pengaturan i/o
void setup() {
    // enable serial 9600bps
    Serial.begin(9600);
    pinMode(led1,OUTPUT);
    pinMode(led2,OUTPUT);
    pinMode(led3,OUTPUT);
    pinMode(led4,OUTPUT);
    // set pin input
    pinMode(sensor1,INPUT);
```

```
pinMode(sensor2, INPUT);  
pinMode(sensor3, INPUT);  
pinMode(sensor4, INPUT);  
  
// set pin interrupt  
pciSetup(sensor1);  
pciSetup(sensor2);  
pciSetup(sensor3);  
pciSetup(sensor4);  
  
Timer1.initialize(1000); // set timer1 1000us = 1ms  
Timer1.attachInterrupt( timerIsr ); // enable timer1  
  
Serial.println("Inisialisasi");  
  
// start the Ethernet connection and the server:  
Ethernet.begin(mac, ip, gateway, subnet);  
server.begin(); // inisialisasi server  
Serial.println("ready");  
for(int i=0;i<5;i++){  
    digitalWrite(led1,1);  
    digitalWrite(led2,1);  
    digitalWrite(led3,1);  
    digitalWrite(led4,1);  
    delay(50);  
    digitalWrite(led1,0);
```

```
digitalWrite(led2,0);

digitalWrite(led3,0);

digitalWrite(led4,0);

delay(200);

}

}

// program utama

void loop() {

    // panggil reset data

    reset_data();

    m1=500-((float)counter1/20); // 20 tetes per mili

    m2=500-((float)counter2/20);

    m3=500-((float)counter3/20);

    m4=500-((float)counter4/20);

    // terima data dari ethernet

    EthernetClient client = server.available();

// program html

    client.println("HTTP/1.1 200 OK");

    client.println("Content-Type: text/html");

    client.println("Connection: close");//the connection

will be closed after completion of the response

    client.println("Refresh: 0.5");// refresh otomatis tiap

0.5 detik

    client.println();
```



```
client.println("<!DOCTYPE HTML>");
client.println("<html>");

client.println("<body bgcolor='cyan'>");// warna begron
client.println("<p align='center'>"); // posisi tulisan
client.print("<font color='magenta' size='6' >"); //
warna dan ukuran tulisan

client.print("Central Monitoring Infus");

if(tetes1<=0){ // warna dan ukuran tulisan
client.print("<font color='red' size='6' >");
}
else {client.print("<font color='black' size='6' >"); //
warna dan ukuran tulisan
client.println("<p align='center'>"); // posisi tulisan

client.print("Tetes Infus A: ");
client.print(tetes1);
client.print(" /Menit ");
client.print("- Vol:");
client.print(ml1);
client.print(" ml");
(tetes2<=0)client.print("<font color='red' size='6' >");
else client.print("<font color='black' size='6' >");

client.println("<p align='center'>");
client.print("Tetes Infus B: ");
client.print(tetes2);
```

```
        client.print(" /Menit ");
        client.print("- Vol:");
        client.print(ml2);
        client.print(" ml");

if(tetes3<=0)client.print("<font color='red' size='6' >");
else client.print("<font color='black' size='6' >");
client.println("<p align='center'>");
client.print("Tetes Infus C: ");
        client.print(tetes3);
        client.print(" /Menit ");
        client.print("- Vol:");
        client.print(ml3);
        client.print(" ml");

if(tetes4<=0)client.print("<font color='red' size='6' >");
else client.print("<font color='black' size='6' >");
client.println("<p align='center'>");
        client.print("Tetes Infus D: ");
        client.print(" /Menit ");
        client.print("- Vol:");
        client.print(ml4);
        client.print(" ml");
client.stop(); // stop perintah
        digitalWrite(led1,0);
        digitalWrite(led2,0);
```

```
digitalWrite(led3,0);
```

```
digitalWrite(led4,0);
```

```
}
```

