

BAB V

KESIMPULAN DAN SARAN

5.1 Kesimpulan

Dari perancangan dan pengujian alat diperoleh kesimpulan penelitian skripsi ini adalah sebagai berikut :

1. Perancangan alat monitoring ini memungkinkan untuk pemantauan secara langsung dan real-time terhadap tetesan infus, memungkinkan deteksi cepat terhadap perubahan kondisi yang mungkin terjadi.
2. Setelah dilakukan pengujian pada alat monitoring tetes infus berbasis mikrokontroler maka didapat kesimpulan ketika terjadi penyumbatan atau kehabisan cairan infus maka dapat diketahui melalui alarm pada alat yang dirancang dan dimonitoring melalui aplikasi remote xy.
3. Dengan penggunaan teknologi IoT, sistem monitoring dapat menghemat waktu dan tenaga dibandingkan dengan metode manual tradisional.

5.2 Saran

Dari penelitian Tugas Akhir ini ada beberapa saran yang diajukan penulis untuk pengembangan selanjutnya antara lain sebagai berikut :

1. Sebaiknya alat monitoring tetes infus berbasis mikrokontroler bisa dijalankan dengan metode otomatis.
2. Lakukan pemantauan rutin terhadap sistem dan sensor untuk memastikan kinerjanya tetap optimal dan melakukan perbaikan atau penggantian jika diperlukan.
3. Berikan pelatihan dan edukasi kepada pengguna tentang cara menggunakan sistem ini dengan baik, serta pentingnya interpretasi data yang diberikan oleh sistem.
4. Pastikan sensor yang digunakan handal dan tahan terhadap berbagai kondisi lingkungan untuk memastikan data yang dihasilkan akurat.
5. Teliti aspek keamanan dan keandalan dari alat monitoring tetes infus yang dibuat. Pastikan bahwa sistem mampu mengatasi situasi darurat, seperti deteksi air yang terhenti atau pemadaman daya.
6. Bandingkan kinerja alat monitoring tetes infus berbasis mikrokontroler dengan sistem infus konvensional dalam hal keamanan, efisiensi, dan kemudahan penggunaan.

7. Pastikan selang infus yang digunakan tidak terjepit ketika melakukan pengujian, sehingga laju tetes infus berjalan lancar dan indikator pada *LCD* atau *Oled* bisa stabil dan dapat dengan mudah untuk mengatur jumlah tetesan per menit.

DAFTAR PUSTAKA

- 1) Achmad, S. (2019). Rancang Bangun Alat Monitoring Cairan Infus Berdasarkan Laju Tetes Dan Volume Cairan Infus Berbasis Nodemcu Esp 8266.
- 2) Astuti, S., Nursaputro, S. T., Utomo, I. J. D., & Oktaviana, N. A. (2023). Alat Kendali Dan Monitoring Volume Serta Laju Tetes Infus Berbasis Internet Of Things. *Orbith: Majalah Ilmiah Pengembangan Rekayasa Dan Sosial*, 18(3), 237-249.
- 3) Bagaskara, M., Dwiono, W., & Hayat, L. (2020). Rancang Bangun Pengatur Infus Berbasis Mikrokontroller. *Jurnal Riset Rekayasa Elektro*, 2(2), 85-93.
- 4) Firdaus, M., Al Banna, A. S., Saputra, A. T., & Nurmahaludin, N. (2020). Sistem Kontrol Dan Monitoring Infus Berbasis Nodemcu.
- 5) Hadhara, F., Munadi, R., & Muayyadi, A. (2023). Alat Monitoring Infus Pasien Berbasis Internet Of Things. *Eproceedings Of Engineering*,
- 6) Harlin, Y. E. K. (2019). Rancang Bangun Emergency Respons Cairan Infus Berbasis Internet Of Things Dan Tetes Infus Otomatis.
- 7) Kusuma, T., & Mulis, M. T. (2018). Perancangan Sistem Monitoring Infus Berbasis Mikrokontroler Wemos D1 R2. *Konferensi Nasional Sistem Informasi (Knsi) 2018*.
- 8) Maulana, M. S. R., Rohana, T., & Al Mudzakir, T. (2023). Implementasi Fuzzy Logic Dalam Monitoring Infus Berbasis Internet Of Things (Iot). *J-Sakti (Jurnal Sains Komputer Dan Informatika)*, 7(2), 957-967.
- 9) Priyandoko, G. (2021). Rancang Bangun Sistem Portable Monitoring Infus Berbasis Internet Of Things. *Jambura Journal Of Electrical And Electronics Engineering*, 3(2), 56-61.
- 10) Raja Hafizh, S. (2023). Pengendali Laju Tetesan Cairan Infus Berdasarkan Detak Jantung Dengan Sistem Pemantauan Menggunakan Website Thingspeak.
- 11) Ramadhan, M. P. B. C., & Saputro, U. A. (2023). Perancangan Sistem Monitoring Tetes Dan Kapasitas Cairan Infus Kamar Pasien Menggunakan Esp8266 Berbasis Website Dengan Framework Laravel. *Jurikom (Jurnal Riset Komputer)*, 10(2), 576-589.
- 12) Retno, D., Sari, M. W., & Ciptadi, P. W. (2021, May). Pengembangan Sistem Kontrol Dan Monitoring Jumlah Tetesan Infus Pada Pasien Menggunakan Android. In *Seri Prosiding Seminar Nasional Dinamika Informatika* (Vol. 5, No. 1).
- 13) Rohman, A., Makruf, M., & Umam, B. A. (2023). Rancang Bangun Monitoring Tetesan Infus Macet Di Nurses Station Berbasis Mikrokontroler Puskesmas

Kamoning. *Jurnal Tiarsie*, 20(4), 97-106.

- 14) Sucipta, I., Simatupang, J. W., Kaswandi, C., & Purnama, I. (2021). Prototipe Pemantauan Tetes Cairan Infus Berbasis Iot Terkoneksi Perangkat Android.

LAMPIRAN

```
#include <SPI.h>
#include <Wire.h>
#include <EEPROM.h>
#include <Adafruit_GFX.h>
#include <Adafruit_SSD1306.h>
#include <Servo.h>
Servo myservo;

#define hitam SSD1306_BLACK
#define putih SSD1306_WHITE
#define ledPin D3
#define buzzerPin D8
#define bt A0
```

```

#define SCREEN_ADDRESS 0x3C

Adafruit_SSD1306 Oled(128, 64, &Wire, -1);

int fMenit;
int fDetik;
int fJam;
byte delaySave = 3;
unsigned long jumlahTetes = 0, counter = 0;
byte targetDrops = 50;
float persentase = 100;
float TPM = 0;
float tps;
int fJ;

byte runSt = 0;
byte setSt = 0;
byte tundaMenu = 0;
long jumlahInfus = 8400;

double tOld, timee;
long countPing = 0;
byte ttsSt = 0;

int pos = 90;

void ICACHE_RAM_ATTR sens() {
    jumlahTetes++;
    jumlahInfus--;
    countPing = 0;
    ttsSt = 1;
    timee = millis() - tOld;
}

#define EEPROM_SIZE 128
void setup() {
    setupOled();
    EEPROM.begin(EEPROM_SIZE);

```

```

EEPROM.write(2, targetDrops); EEPROM.commit();
targetDrops = EEPROM.read(2), DEC; EEPROM.end();
pinMode(ledPin, OUTPUT);
digitalWrite(ledPin, LOW);
pinMode(bt, INPUT);
myservo.attach(D5);
myservo.write(pos);
attachInterrupt(digitalPinToInterrupt(12), sens, RISING);
frame1(TPM, runSt);
beep(); beep();
setupXY();
}

void loop() {
  loopXY();
  tombol();
  timer();
  hitungTPM();
}

void hitungTPM() {
  if (ttsSt == 1) {
    digitalWrite(ledPin, HIGH);
    if (timee > 250) {
      tps = timee;
      if (tps != 0) {
        TPM = 1000 / tps * 60;
      }
      fJ = jumlahInfus * 60 / TPM;
      formatJam(fJ);
      frame1(TPM, runSt);
      ttsSt = 0;
      delay(10);
      tOld = millis();
    }
  }
  digitalWrite(ledPin, LOW);
}

```

```

unsigned long oldMillis = 0, counterr = 0;
void timer() {
  if (millis() - oldMillis >= 1000) {
    oldMillis = millis();
    countPing++;
    Serial.print("time ");
    Serial.println(timee);
    Serial.print("TPM ");
    Serial.println(TPM);
    Serial.println();
    persentase = mapFloat(jumlahInfus, 0, 8400, 0, 100);
    //Serial.println(analogRead(bt));
    if (delaySave > 0) delaySave--; // save setting
    frame1(TPM, runSt);
    if (countPing > 15) {
      if (countPing % 5 == 0) beep();
    }
    if (runSt == 1) {
      if (countPing < 15) servoControl();
    } else {
      countPing = 0;
    }
  }
}

void servoControl() {
  if ((TPM != 0) and (countPing < 15)) {
    if (TPM < targetDrops - 1) {
      if (pos < 170) pos++;
      myservo.write(pos);
    }
    if (TPM > targetDrops + 1) {
      if (pos > 20) pos--;
      myservo.write(pos);
    }
  }
}
}

```

```

void beep() {
    pinMode(buzzerPin, OUTPUT);
    digitalWrite(buzzerPin, HIGH);
    delay(100);
    digitalWrite(buzzerPin, LOW);
    delay(100);
}

void formatJam(unsigned int t) {
    fJam = t / 3600;
    if (t < 3600) fMenit = t / 60;
    else fMenit = (t - (fJam * 3600)) / 60;
    if (fMenit >= 60) fMenit = fMenit - 60;
    fDetik = t % 60;
}

float mapFloat(float x, float fromLow, float fromHigh, float toLow, float
toHigh) {
    return (x - fromLow) * (toHigh - toLow) / (fromHigh - fromLow) + toLow;
}

```








```
Infus_Oled_Wemos_XY_2024 | Arduino IDE 2.3.2
File Edit Sketch Tools Help
Select Board
Infus_Oled_Wemos_XY_2024.ino
1 #include <SPI.h>
2 #include <Wire.h>
3 #include <EEPROM.h>
4 #include <Adafruit_GFX.h>
5 #include <Adafruit_SSD1306.h>
6 #include <Servo.h>
7 Servo myservo;
8
9
10 #define hitam SSD1306_BLACK
11 #define putih SSD1306_WHITE
12 #define ledPin D3
13 #define buzzerPin D8
14 #define bt A0
15 #define SCREEN_ADDRESS 0x3C
16
17 Adafruit_SSD1306 Oled(128, 64, &Wire, -1);
18
19 int fMenit;
20 int fDetik;
21 int fJam;
22 byte delaySave = 3;
23 unsigned long jumlahTetes = 0, counter = 0;
24 byte targetDrops = 50;
25 float persentase = 100;
26 float TPM = 0;
27 float tps;
28 int f3;
```









