

LAMPIRAN

1. Perhitungan Rata-Rata

$$\text{Rata - Rata } (\bar{X}) = \frac{\sum Xi}{n}$$

Dimana : \bar{X} = rata-rata

$\sum Xi$ = jumlah nilai data

n = Banyak data (1,2,3,...,n)

a. Perhitungan rata-rata parameter NIBP

$$\text{Rata - Rata } (\bar{X}) \text{ sistol} = \frac{\sum Xi}{n}$$

$$\begin{aligned} & 118+117+122+118+121+124+117+118+119+ \\ & 118+119+123+121+122+121+124+123+123+ \\ & 122+124 \\ & = \frac{ }{20} \\ & = \frac{2414}{20} \\ & = 120,7 \end{aligned}$$

$$\text{Rata - Rata } (\bar{X}) \text{ diastol} = \frac{\sum Xi}{n}$$

$$\begin{aligned} & 82+78+79+83+77+78+81+82+81+80+82+81+ \\ & 79+79+81+84+78+83+77+81 \\ & = \frac{ }{20} \\ & = \frac{1606}{20} \\ & = 80,3 \end{aligned}$$

b. Perhitungan rata-rata parameter BPM

$$\text{Rata - Rata } (\bar{X}) \text{ 60} = \frac{\sum Xi}{n}$$

$$= \frac{59+60+58+59+62}{5}$$

$$= \frac{298}{5}$$

$$= 59,6$$

$$\text{Rata - Rata } (\bar{X})_{80} = \frac{\sum X_i}{n}$$

$$= \frac{81+78+82+80+77}{5}$$

$$= \frac{398}{5}$$

$$= 79,6$$

$$\text{Rata - Rata } (\bar{X})_{100} = \frac{\sum X_i}{n}$$

$$= \frac{99+97+98+99+101}{5}$$

$$= \frac{494}{5}$$

$$= 98,8$$

c. Perhitungan rata-rata parameter SPO2

$$\text{Rata - Rata } (\bar{X})_{90} = \frac{\sum X_i}{n}$$

$$= \frac{88+87+92+93+87}{5}$$

$$= \frac{447}{5}$$

$$= 89,4$$

$$\text{Rata - Rata } (\bar{X})_{95} = \frac{\sum X_i}{n}$$

$$= \frac{97+96+97+93+98}{5}$$

$$= \frac{481}{5}$$

$$= 96,2$$

$$\text{Rata - Rata } (\bar{X})_{100} = \frac{\sum X_i}{n}$$

$$= \frac{98+97+98+99+102}{5}$$

$$= \frac{494}{5}$$

$$= 98,8$$

2. Perhitungan *Error*

$$\%Error = \frac{X_n - Y_n}{X_n} \times 100\%$$

Dimana : X_n = rata-rata data kalibrator

Y_n = rata-rata data alat

a. Perhitungan *error* parameter NIBP

$$\%Error \text{ Sistol} = \frac{X_n - Y_n}{X_n} \times 100\%$$

$$= \frac{120 - 120,7}{120} \times 100\%$$

$$= \frac{0,7}{120} \times 100\%$$

$$= 0,0058 \%$$

$$\begin{aligned} \%Error \text{ Diastol} &= \frac{Xn - Yn}{Xn} \times 100\% \\ &= \frac{80 - 80,3}{80} \times 100\% \\ &= \frac{0,3}{80} \times 100\% \\ &= 0,00375 \% \end{aligned}$$

b. Perhitungan *error* parameter BPM

$$\begin{aligned} \%Error \text{ 60} &= \frac{Xn - Yn}{Xn} \times 100\% \\ &= \frac{60 - 59,6}{60} \times 100\% \\ &= \frac{0,4}{60} \times 100\% \\ &= 0,0067 \% \end{aligned}$$

$$\begin{aligned} \%Error \text{ 80} &= \frac{Xn - Yn}{Xn} \times 100\% \\ &= \frac{80 - 79,6}{80} \times 100\% \\ &= \frac{0,4}{80} \times 100\% \\ &= 0,005 \% \end{aligned}$$

$$\begin{aligned} \%Error \text{ 100} &= \frac{Xn - Yn}{Xn} \times 100\% \\ &= \frac{100 - 98,8}{100} \times 100\% \\ &= \frac{1,2}{100} \times 100\% \\ &= 0,012 \% \end{aligned}$$

c. Perhitungan *error* parameter SPO2

$$\begin{aligned} \%Error\ 90 &= \frac{Xn - Yn}{Xn} \times 100\% \\ &= \frac{90 - 89,4}{90} \times 100\% \\ &= \frac{0,6}{90} \times 100\% \\ &= 0,0067\ \% \end{aligned}$$

$$\begin{aligned} \%Error\ 95 &= \frac{Xn - Yn}{Xn} \times 100\% \\ &= \frac{95 - 96,2}{95} \times 100\% \\ &= \frac{1,2}{95} \times 100\% \\ &= 0,0126\ \% \end{aligned}$$

$$\begin{aligned} \%Error\ 100 &= \frac{Xn - Yn}{Xn} \times 100\% \\ &= \frac{100 - 98,8}{100} \times 100\% \\ &= \frac{1,2}{100} \times 100\% \\ &= 0,012\ \% \end{aligned}$$

3. Perhitungan Simpangan

$$\text{Simpangan} = Xn - Yn$$

Dimana :

Xn = rata-rata data kalibrator

Yn = rata-rata data alat

a. Perhitungan simpangan parameter NIBP

$$\text{Simpangan Sistol} = Xn - Yn$$

$$= 120 - 120,7$$

$$= 0,7$$

$$\text{Simpangan Diastol} = Xn - Yn$$

$$= 80 - 80,3$$

$$= 0,3$$

b. Perhitungan simpangan parameter BPM

$$\text{Simpangan 60} = Xn - Yn$$

$$= 60 - 59,6$$

$$= 0,4$$

$$\text{Simpangan 80} = Xn - Yn$$

$$= 80 - 79,6$$

$$= 0,4$$

$$\text{Simpangan 100} = Xn - Yn$$

$$= 100 - 98,8$$

$$= 1,2$$

c. Perhitungan simpangan parameter SPO2

$$\text{Simpangan 90} = Xn - Yn$$

$$= 90 - 89,4$$

$$= 0,6$$

$$\text{Simpangan 95} = Xn - Yn$$

$$= 95 - 96,2$$

$$= 1,2$$

$$\text{Simpangan 100} = Xn - Yn$$

$$= 100 - 98,8$$

= 1,2

4. Program alat

a. Program Pembacaan Tensimeter

Program ini merupakan program pembacaan tensimeter menggunakan arduino Mega.

```
int bacadenyut=0;
int flag=0;
int rcount=0;
float sistol=0;
float diastol=0;
int last_mmHg;
int mmHg;
float x,r;
int motor = 12;
int selenoid = 13;
int Start = 2; // tombol untuk start
int Stop = 3; // tombol untuk stop
int Reset=14; //tombol untuk reset
void setup()
{
  Serial.begin(115200);
  // inisialisasi serial komunikasi pada baudrate
  115200 bps
  pinMode(motor, OUTPUT);
  pinMode(selenoid, OUTPUT);
  pinMode(Start, INPUT);
  pinMode(Stop, INPUT);
  // pengaturan mode pin sebagai input/output
```

```
digitalWrite(Reset,HIGH);
pinMode(Reset,OUTPUT);
// pengaturan pada tombol reset untuk memberikan
nilai HIGH pada pin digital
}
void loop()
{
  if (digitalRead (Start)==HIGH)
  {
    digitalWrite(motor, HIGH);
    digitalWrite(solenoid, HIGH);
    bacadenyut==0;
  }
// program ini berfungsi ketika tombol start dalam
keadaan HIGH maka nilai pin digital untuk motor dan
solenoid HIGH
if (digitalRead (Stop)==HIGH)
{
  digitalWrite(motor, LOW);
  digitalWrite(solenoid, LOW);
  bacadenyut==0;
}
// program ini berfungsi ketika tombol STOP dalam
keadaan HIGH maka nilai pin digital untuk motor dan
solenoid LOW
```



```

digitalWrite(Reset,LOW);

//pengaturan pada tombol reset untuk memberikan nilai
LOW pada pin digital

    r=0;
    for(int i=0;i<100;i++)
    {
        x=((analogRead(A0)*(5.0/1023))-0.17);
        r=r+x;
    }
// program ini berfungsi mengubah nilai ADC ke
tegangan dan dikurangi nilai offset sebesar 0.17 V

    r=r/100;
    float pressure=
((r-0.04*4.75)/(0.009*4.75)+(2.5*1*0.009*4.75))+4.33;
//Mengkonversi tegangan menjadi nilai ke kPa
    pressure = (pressure*7.5);
//Mengkonversi nilai kPa menjadi nilai mmHg
    int mmHg=(int)pressure; if(mmHg>180)
// Berhenti memompa ketika tekanan melebihi 180 mmHg
    {
        digitalWrite(motor, LOW);
//motor off
        if (bacadenyut==0)last_mmHg=mmHg;
        bacadenyut=1;
//tanda baca denyut dimulai
    }

```

```
    if (bacadenyut==1)
// Mulai membaca denyut ketika motor off
    {
        if(mmHg>last_mmHg&&rcount>=3)
        {
            if (flag==2&&mmHg<=100)
                {(diastol=last_mmHg-10);flag++;}
//denyutan kedua diambil sebagai nilai diastol
            if(flag==1&&mmHg<=100) flag++;
            if(flag==0&&mmHg>=100) {sistol=last_mmHg;flag++;}
//denyutan pertama diambil sebagai nilai sistol
        }
        if (rcount<3)rcount++;
        else last_mmHg=mmHg;
    }
    Serial.print("C ");
    Serial.print(sistol,0);

    Serial.print("D ");
    Serial.print(diastol,0);
    delay(100);
//pada Serial.print("C ") dan ("D ") digunakan untuk
proses parsing data sebagai transmitter
```

```
    if (bacadenyut==1)
    {
        if (rcount<3)rcount++;
        {
            if (mmHg<60)
            {
                digitalWrite(solenoid, LOW);
            }
        }
    }
    // program ketika nilai sistol dan diastol telah
    terbaca maka solenoid akan off ketika tekanan dibawah
    60 mmHg
}
```

b. Program Pembacaan BPM

Program ini merupakan program pembacaan BPM menggunakan arduino Mega.

```
#include <Wire.h>
#include "MAX30105.h"
#include "heartRate.h"
MAX30105 particleSensor;
const byte RATE_SIZE = 7;
byte rates[RATE_SIZE];
byte rateSpot = 0;
long lastBeat = 0;
float beatsPerMinute;
int beatAvg;
// program ini digunakan sebagai inisialisasi
variabel sensor

int Reset=14;//variabel untuk tombol reset
void setup()
{
    Serial.begin(115200);
    // inisialisasi serial komunikasi pada baudrate
    115200 bps

    digitalWrite(Reset,HIGH);
    pinMode(Reset,OUTPUT);
    // Pengaturan pada tombol reset untuk memberikan
    nilai HIGH pada pin digital
```

```
// inisialisasi sensor
if (!particleSensor.begin(Wire, I2C_SPEED_FAST))
{
    Serial.println("MAX30105 was not found. Please
    check wiring/power. ");

    while (1);
}

Serial.println("Place your index finger on the
sensor with steady pressure.");

particleSensor.setup();
particleSensor.setPulseAmplitudeRed(0x0A);
particleSensor.setPulseAmplitudeGreen(0);
}

void loop()
{
    digitalWrite(Reset, LOW);
    // Pengaturan pada tombol reset untuk memberikan
    nilai LOW pada pin digital

    {

        long irValue = particleSensor.getIR();

        if (checkForBeat(irValue) == true)
        {
            long delta = millis() - lastBeat;
            lastBeat = millis();
            beatsPerMinute = 60 / (delta / 1000.0);
```

```
if (beatsPerMinute < 255 && beatsPerMinute > 20)
{
    rates[ratesSpot++] = (byte)beatsPerMinute;
    ratesSpot %= RATE_SIZE;
    beatAvg = 0;
    for (byte x = 0 ; x < RATE_SIZE ; x++)
        beatAvg += rates[x];
    beatAvg /= RATE_SIZE;
}
}

// Program ini digunakan untuk proses perhitungan BPM
Dan proses penerimaan besar nilai IR yang diterima

Serial.print("E ");
Serial.println(beatAvg);

//pada Serial.print("E ")digunakan untuk proses
parsing data sebagai transmitter

}
}
```

c. Program Pembacaan SPO2

Program ini merupakan program pembacaan SPO2 menggunakan arduino

Mega.

```
#include <Wire.h>
#include "MAX30105.h"
#include "spo2_algorithm.h"

MAX30105 particleSensor;

#define MAX_BRIGHTNESS 255

#if defined(__AVR_ATmega328P__) ||
defined(__AVR_ATmega168__)
uint16_t irBuffer[100];
uint16_t redBuffer[100];
#else
uint32_t irBuffer[100];
uint32_t redBuffer[100];
#endif

int32_t bufferLength;
int32_t spo2;
int8_t validSPO2;
int32_t heartRate;
int8_t validHeartRate;

// Program ini digunakan sebagai inisialisasi pada
infrared LED dan red LED pada sensor. Pada bagian
int32_t bufferLength untuk mengetahui banyak data
yang diterima sedangkan int32_t spo2 dan int8_t
validSPO2 untuk menandakan kalkulasi SPO2 siap
dilakukan
```

```
int Reset=14;//variabel untuk tombol reset
void setup()
{
  Serial.begin(115200);
  // inisialisasi serial komunikasi pada baudrate
  115200 bps

  digitalWrite(Reset,HIGH);
  pinMode(Reset,OUTPUT);
  // Pengaturan pada tombol reset untuk memberikan HIGH
  pada pin digital
  // inisialisasi sensor
  if (!particleSensor.begin(Wire, I2C_SPEED_FAST))
  {
    Serial.println(F("MAX30105 was not found. Please
    check wiring/power."));

    while (1);
  }
  Serial.read();
  byte ledBrightness = 60; //Ops: 0=Off to 255=50mA
  byte sampleAverage = 4; //Ops: 1, 2, 4, 8, 16, 32
  byte ledMode = 2;
  //Ops: 1 = Red only, 2 = Red + IR, 3 = Red+IR+Green
  byte sampleRate = 100;
  //Ops: 50, 100, 200, 400, 800, 1000, 1600, 3200
  int pulseWidth = 411; //Ops: 69, 118, 215, 411
  int adcRange = 4096;
  //Ops: 2048, 4096, 8192, 16384
```



```
particleSensor.setup(ledBrightness, sampleAverage,
ledMode, sampleRate, pulseWidth, adcRange);

// Pada bagian ini berfungsi sebagai pengaturan
konfigurasi pada sensor yang digunakan untuk
mendapatkan nilai yang sesuai pada sensor
}

void loop()

{

    digitalWrite(Reset,LOW);

//Pengaturan pada tombol reset untuk memberikan nilai
LOW pada pin digital

{

    bufferLength = 100;

    for (byte i = 0 ; i < bufferLength ; i++)

    {

        while (particleSensor.available() == false)

        particleSensor.check();

        redBuffer[i] = particleSensor.getRed();

        irBuffer[i] = particleSensor.getIR();

        particleSensor.nextSample();

        Serial.print(F(", SPO2="));

        Serial.println(spo2, DEC);

    }

}
```

```

maxim_heart_rate_and_oxygen_saturation(irBuffer,
bufferLength, redBuffer, &spo2, &validSPO2,
&heartRate, &validHeartRate);

while (1){
    for (byte i = 25; i < 100; i++)
    {
        redBuffer[i - 25] = redBuffer[i];
        irBuffer[i - 25] = irBuffer[i];}
    for (byte i = 75; i < 100; i++)
    {
        while (particleSensor.available() == false)
        particleSensor.check();
        redBuffer[i] = particleSensor.getRed();
        irBuffer[i] = particleSensor.getIR();
        particleSensor.nextSample();
        Serial.print(F("F "));
        Serial.println(spo2, DEC);
    }
    maxim_heart_rate_and_oxygen_saturation(irBuffer,
bufferLength, redBuffer, &spo2, &validSPO2,
&heartRate, &validHeartRate);

    //proses penghitungan setiap data yang masuk pada
sensor setiap detik dan pada Serial.print(F("F
"))digunakan untuk proses parsing data sebagai
transmitter

}}}
```

d. Program LCD TFT

Program ini merupakan program LCD TFT menggunakan arduino Mega.

```
#include <Messenger.h>
#include "Adafruit_GFX.h"// Hardware-specific library
#include <MCUFRIEND_kbv.h>
MCUFRIEND_kbv tft;

#define BLACK 0x0000
#define BLUE 0x001F
#define RED 0xF800
#define GREEN 0x07E0
#define CYAN 0x07FF
#define MAGENTA 0xF81F
#define YELLOW 0xFFE0
#define WHITE 0xFFFF
// program ini berfungsi sebagai library dari LCD TFT

Messenger message = Messenger();
Messenger message1 = Messenger();
Messenger message2 = Messenger();
Messenger message3 = Messenger();

// pada program ini berfungsi sebagai parcing data
receiver

int dataIn, dataIn2,dataIn3,dataIn4,dataIn5,dataIn6;
```

```
void message1Completed() {
    if ( message1.checkString("C") {
        dataIn = message1.readInt();
        tft.setTextColor(WHITE);
        tft.setTextSize(3);
        tft.setCursor(360, 15);
        tft.print(dataIn)

        dataIn2 = message1.readInt();
        tft.setTextColor(WHITE);
        tft.setTextSize(3);
        tft.setCursor(360, 60);
        tft.print(dataIn2);
    }
}

// program ini berfungsi menerima data/receiver pada
pin RX1 (D19)

void message2Completed() {
    if ( message2.checkString("E") )
    {
        dataIn3 = message2.readInt();
        tft.setTextColor(WHITE);
        tft.setTextSize(3);
        tft.setCursor(120, 155);
        tft.print(dataIn3);
    }
}
```

```
// program ini berfungsi menerima data/receiver pada
pin RX2(D17)

void message3Completed()
{
    if ( message3.checkString("F") )
    {
        dataIn4 = message3.readInt();
        tft.setTextColor(WHITE);
        tft.setTextSize(3);
        tft.setCursor(360, 155);
        tft.print(dataIn4);
    }
}

// program ini berfungsi menerima data/receiver pada
pin RX3(D15)

void messageCompleted()
{
    if ( message.checkString("A") )
    {
        dataIn5 = message.readInt();
        tft.setTextColor(WHITE);
        tft.setTextSize(3);
        tft.setCursor(120, 265);
        tft.print(dataIn5);
    }
}
```

```
dataIn6 = message.readInt();

    tft.setTextColor(WHITE);

    tft.setTextSize(3);

    tft.setCursor(360, 265);

    tft.print(dataIn6);

    }

}

// program ini berfungsi menerima data/receiver pada
pin RX0 (D0)

void setup() {

    Serial.begin(115200);

    Serial1.begin(115200);

    Serial2.begin(115200);

    Serial3.begin(115200);

// inisialisasi serial komunikasi pada baudrate
115200 bps

    message1.attach(message1Completed);

    message2.attach(message2Completed);

    message3.attach(message3Completed);

    message.attach(messageCompleted);

// Melampirkan data yang diterima dari setiap pin RX

    uint16_t ID = tft.readID();

    tft.begin(ID);

    tft.setRotation(3);

    tft.fillScreen(BLACK);

    tft.setTextColor(WHITE);

    tft.setTextSize(4);
```

```
tft.setCursor(60, 140);
tft.print("ASSALAMUALAIKUM");
delay(2000);
tft.fillScreen(BLACK);
tft.setTextColor(WHITE);
    tft.setTextSize(4);
    tft.setCursor(110, 70);
    tft.print("VITAL SIGNS");
    tft.setTextColor(WHITE);
    tft.setTextSize(3);
    tft.setCursor(70, 200);
    tft.print("ARIF.R.");
    tft.setTextColor(WHITE);
    tft.setTextSize(2);
    tft.setCursor(55, 250);
    tft.print("(20163010003)");
    tft.setTextColor(WHITE);
    tft.setTextSize(3);
    tft.setCursor(250, 200);
    tft.print("FERIYAN.A.");
    tft.setTextColor(WHITE);
    tft.setTextSize(2);
    tft.setCursor(260, 250);
    tft.print("(20163010041)");
    delay(5000);
    tft.fillScreen(BLACK);
}
// program ini berfungsi sebagai tampilan awal pada
LCD TFT
```

```
void loop()
{
    tft.fillRect(350,5,100,40,BLACK);//data C
    tft.fillRect(350,50,100,40,BLACK);//data D
    tft.fillRect(110,145,100,40,BLACK);//data E
    tft.fillRect(350,145,100,40,BLACK);//data F
    tft.fillRect(110, 255,100,40,BLACK);//data A
    tft.fillRect(350, 255,100,40,BLACK);//data B
    // Program ini berfungsi untuk menerima dan menghapus
    data yang masuk agar tidak terjadi penumpukan data
    pada LCD TFT

    while ( Serial1.available() )
    {
        message1.process( Serial1.read() );
        if ( Serial2.available() )
        {
            message2.process( Serial2.read() );
        }
        if ( Serial3.available() )
        {
            message3.process( Serial3.read() );
        }
        if ( Serial.available() )
        {
            message.process( Serial.read() );
        }
    }
}
```



```
//program ini berfungsi untuk menandakan bahwa proses
penerimaan data pada setiap pin RX dilakukan dan
diterima
```

```
tft.setTextColor(BLUE);
tft.setTextSize(3);
tft.setCursor(10, 40);
tft.print(" VITAL SIGNS ");
```

```
tft.setTextColor(BLUE);
tft.setTextSize(3);
tft.setCursor(250, 10);
tft.print("SIS :");
```

```
tft.setTextColor(BLUE);
tft.setTextSize(3);
tft.setCursor(250, 60);
tft.print("DIA :");
```

```
tft.setTextColor(BLUE);
tft.setTextSize(3);
tft.setCursor(10, 155);
tft.print("BPM :");
```

```
tft.setTextColor(BLUE);
tft.setTextSize(3);
tft.setCursor(250, 155);
tft.print("SPO2 :");
```

```
tft.setTextColor(BLUE);  
    tft.setTextSize(3);  
    tft.setCursor(10, 260);  
    tft.print("RESP :");  
  
    tft.setTextColor(BLUE);  
    tft.setTextSize(3);  
    tft.setCursor(250, 260);  
    tft.print("SUHU :");  
  
    tft.drawLine(0,110,479,110,WHITE);  
    tft.drawLine(0,220,479,220,WHITE);  
    tft.drawLine(240,0,240,319,WHITE);  
  
//tampilan LCD TFT untuk setiap parameter  
}
```