

# LAMPIRAN



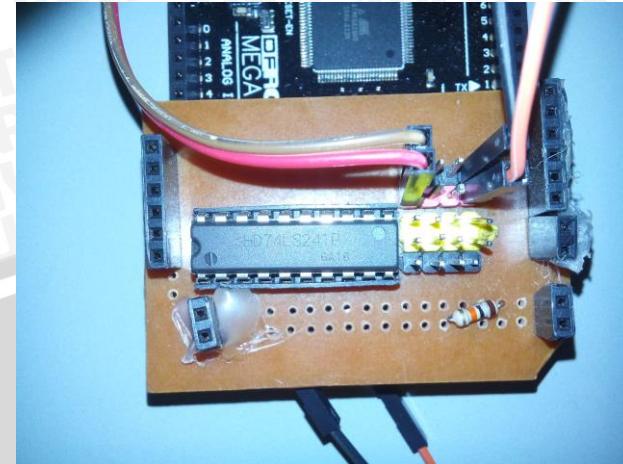
## **LAMPIRAN I**

### **FOTO ALAT**





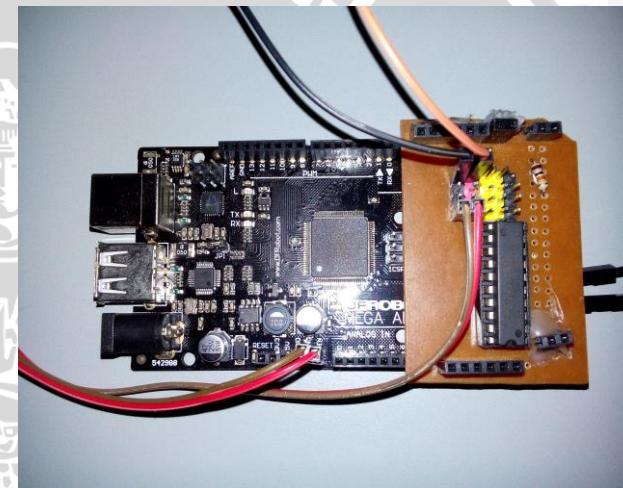
Dynamixel Digital Servo AX-12A Alarm On



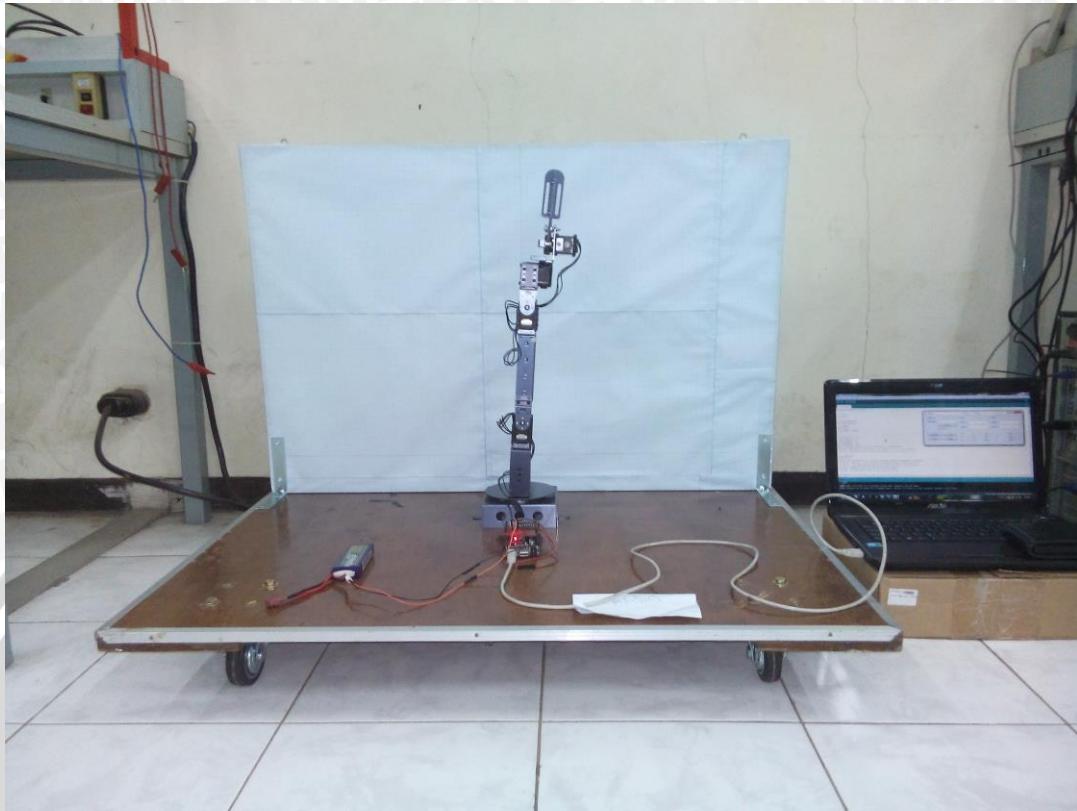
Rangkaian Buffer dengan IC 74LS241N



Smart Arm Robotic AX-12A Tanpa Medan



Arduino Mega dengan Rangkaian Buffer



Rangkaian Keseluruhan Alat

## **LAMPIRAN II**

### **LISTING PROGRAM**



### **Program Reset Resolusi**

```
#include <DynamixelSerial2.h> //library yang digunakan

int Temperature,Voltage,Position; //deklarasi variabel

void setup(){
    Serial.begin(9600);

    Dynamixel.begin(1000000,2); // setting kecepatan komunikasi untuk dynamixel dan pin 2
    sebagai pengendali aliran data

    delay(1000);}

void loop(){
    //menggerakan dynamixel

    Dynamixel.moveSpeed(7,512,200);
    delay(500);

}
```



### **Program Kalibrasi Sudut dan Pengukuran Torsi**

```
#include <DynamixelSerial2.h>

int i, data, hasil,hitung;

unsigned char data_array[6];

int val;

int currentposx = 512;

int currentposy = 512;

int currentx, currenty;

void setup()

{

    Serial.begin(115200);

    Dynamixel.begin(1000000,A8); // Initialize the servo at 1Mbps and Pin Control 2

    delay(500);

    // Dynamixel.moveSpeed(19,512,1000);

    // Dynamixel.moveSpeed(19,512,1000);

}

void loop()
```

```
{  
    if (Serial.available())  
    {  
        data = Serial.read();  
  
        if (data == 'z')  
        {  
            {  
                if (i == 0);  
  
                else if (i == 5)  
                {  
                    data_array[1] = data_array[1] - 48;  
                    data_array[2] = data_array[2] - 48;  
                    data_array[3] = data_array[3] - 48;  
                    data_array[4] = data_array[4] - 48;  
  
                    hasil = (data_array[1] * 1000) + (data_array[2] * 100) + (data_array[3] * 10) +  
                    (data_array[4]);  
  
                    i = 0;  
                }  
            }  
        }  
    }  
}
```

```
else if (i == 4)
{
    data_array[1] = data_array[1] - 48;
    data_array[2] = data_array[2] - 48;
    data_array[3] = data_array[3] - 48;
    hasil = (data_array[1] * 100) + (data_array[2] * 10) + (data_array[3]);
    i = 0;
}

else if (i == 3)
{
    data_array[1] = data_array[1] - 48;
    data_array[2] = data_array[2] - 48;
    hasil = (data_array[1] * 10) + (data_array[2]);
    i = 0;
}

else if (i == 2)
{
    data_array[1] = data_array[1] - 48;
    hasil = (data_array[1]);
```

```
i = 0;  
}  
  
else i=0;  
}  
  
switch(data_array[0])  
{  
case 'a':  
    Dynamixel.moveSpeed(1,hasil,1000);  
    break;  
  
case 'b':  
    Dynamixel.moveSpeed(2,hasil,1000);  
    break;  
  
case 'c':  
    Dynamixel.moveSpeed(3,hasil,1000);  
    break;  
  
case 'd':  
    Dynamixel.moveSpeed(4,hasil,1000);  
    break;  
  
case 'e':
```



```
Dynamixel.moveSpeed(5,hasil,1000);
```

```
break;
```

```
case 'f':
```

```
Dynamixel.moveSpeed(6,hasil,1000);
```

```
break;
```

```
case 'g':
```

```
Dynamixel.moveSpeed(7,hasil,1000);
```

```
break;
```

```
case 'h':
```

```
Dynamixel.moveSpeed(8,hasil,1000);
```

```
break;
```

```
case 'i':
```

```
Dynamixel.moveSpeed(9,hasil,1000);
```

```
break;
```

```
case 'j':
```

```
Dynamixel.moveSpeed(10,hasil,1000);
```

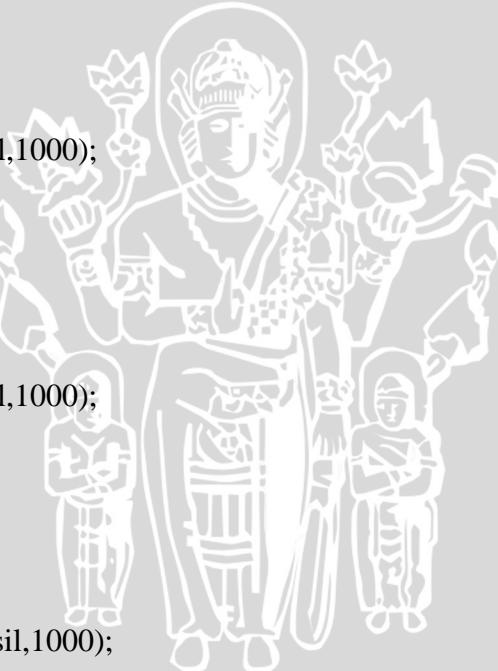
```
break;
```

```
case 'k':
```

```
Dynamixel.moveSpeed(11,hasil,1000);
```

```
break;
```

```
case 'l':
```



```
Dynamixel.moveSpeed(12,hasil,1000);
break;

case 'm':
Dynamixel.moveSpeed(13,hasil,1000);
break;

case 'n':
Dynamixel.moveSpeed(14,hasil,1000);
break;

case 'o':
Dynamixel.moveSpeed(15,hasil,1000);
break;

case 'p':
Dynamixel.moveSpeed(16,hasil,1000);
break;

case 'q':
Dynamixel.moveSpeed(17,hasil,1000);
break;

case 'r':
Dynamixel.moveSpeed(18,hasil,1000);
break;

case 's':
```



```
Dynamixel.moveSpeed(19,hasil,1000);
```

```
break;
```

```
case 't':
```

```
Dynamixel.moveSpeed(20,hasil,1000);
```

```
break;
```

```
case 'u':
```

```
currentposx = currentposx-10;
```

```
Dynamixel.moveSpeed(19,currentposx,1000);
```

```
break;
```

```
case 'v':
```

```
currentposx = currentposx+10;
```

```
Dynamixel.moveSpeed(19,currentposx,1000);
```

```
break;
```

```
case 'w':
```

```
currentposy = currentposy-10;
```

```
Dynamixel.moveSpeed(20,currentposy,1000);
```

```
break;
```

```
case 'x':
```

```
currentposy=currentposy+10;
```

```
Dynamixel.moveSpeed(20,currentposy,1000);
```

```
break;
```

UNIVERSITAS BRAWIJAYA



```
// case '0':  
  
// for(hitung=1;hitung<21;hitung++)  
  
// {  
  
//     Dynamixel.torqueStatus(hitung,OFF);  
  
// }  
  
// break;  
  
//  
  
// case '1':  
  
// for(hitung=1;hitung<21;hitung++)  
  
// {  
  
//     Dynamixel.torqueStatus(hitung,ON);  
  
// }  
  
// break;  
  
}  
  
if(currentposy>1000)currentposy=currentposy-10;  
  
if(currentposy<23)currentposy=currentposy+10;  
  
if(currentposx>1000)currentposx=currentposx-10;  
  
if(currentposx<23)currentposx=currentposx+10;  
}
```

```
    else
    {
        data_array[i] = data;
        i++;
    }
}
```

### Program Pengukuran Temperatur, Tegangan, dan Resolusi

```
#include <DynamixelSerial2.h> //library yang digunakan
```

```
int Temperature,Voltage,Position; //deklarasi variabel
```

```
void setup(){
```

```
Serial.begin(9600);
```

```
Dynamixel.begin(1000000,2); // setting kecepatan komunikasi untuk dynamixel dan pin 2 sebagai pengendali aliran data
```

```
delay(1000);}
```

```
void loop(){
```

```
//menggerakan dynamixel
```

```
if (Position==0){  
  
Dynamixel.moveSpeed(7,1023,200);  
  
delay(500);}  
  
if (Position==1023){  
  
Dynamixel.moveSpeed(7,0,200);  
  
delay(500);}  
  
//membaca kondisi dynamixel  
  
Temperature = Dynamixel.readTemperature(7); //membaca temperatur dynamixel  
  
Voltage = Dynamixel.readVoltage(7); //membaca voltage dynamixel  
  
Position = Dynamixel.readPosition(7); //membaca posisi dynamixel  
  
//menampilkan ke serial monitor  
  
Serial.print("****Temperature:");  
  
Serial.print(Temperature);  
  
Serial.print("Celcius Voltage:");  
  
Serial.print(Voltage);  
  
Serial.print("Volts Position:");  
  
Serial.print(Position);  
  
Serial.println("of 1023 resolution");  
  
delay(100);
```

}

**Program Forward Kinematic**

#include &lt;DynamixelSerial2.h&gt;

int tetha1deg = 15;

int tetha2deg = 24;

int a = 17;

int b = 23;

int currentpos = 512;

int currentposg = 600;

float tetha1rad,tetha2rad, Px, Py, currenttetha1, currenttetha2;

float K1 = (300/1023); //Faktor Konversi dari Resolusi (bit) ke Sudut (derajat)

float K2 = (1023/300); //Faktor Konversi dari Sudut (derajat) ke Resolusi (bit)

void setup()

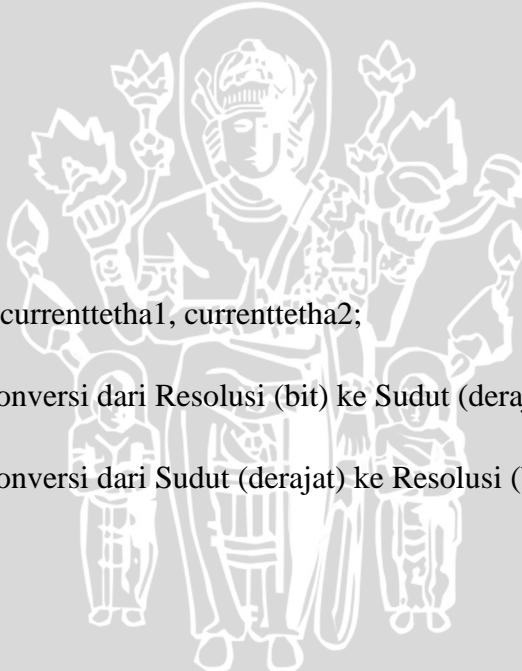
{

Serial.begin(115200);

Dynamixel.begin(1000000,A8); // Initialize the servo at 1Mbps and Pin Control 2

delay(500);

// Dynamixel.moveSpeed(19,512,1000);



```
// put your setup code here, to run once:  
  
}  
  
void loop() {  
  
// put your main code here, to run repeatedly:  
  
tetha1rad = tetha1deg*(PI/180);  
  
tetha2rad = tetha2deg*(PI/180);  
  
Px=a*(cos(tetha1rad))+(b*(cos((tetha1deg+tetha2deg)*(PI/180))));  
  
Py=a*(sin(tetha1rad))+(b*(sin((tetha1deg+tetha2deg)*(PI/180))));  
  
currenttetha1 = currentpos-(tetha1deg*K2);//Posisi Servo ID 2  
  
currenttetha2 = currentpos-(tetha2deg*K2);//Posisi Servo ID 4  
  
//menampilkan hasil perhitungan ke serial Monitor:  
  
Serial.print("Px = ");  
  
Serial.print(Px);  
  
Serial.print("cm Py = ");  
  
Serial.print(Py);  
  
Serial.println("cm");  
  
delay(200);  
  
//Pergerakan Servo
```

```
if(currentttheta1)
    Dynamixel.moveSpeed(2,461,1000);
    Dynamixel.moveSpeed(3,563,1000);
if(currentttheta2)
    Dynamixel.moveSpeed(4,430,1000);
    Dynamixel.moveSpeed(5,594,1000);
Dynamixel.moveSpeed(7,currentposg,1000);
}
```

### **Program Invers Kinematic**

```
#include <math.h>
```

```
#include <DynamixelSerial2.h>
```

```
#define PI 3.14159265
```

```
#define a 17
```

```
#define b 23
```

```
int currentposg = 512;
```

```
double Y0 = 34.2;
```

```
double X0 = 18.2;
```



```
double c, alpha1rad, alpha1deg, phi1rad, phi1deg, tetha1rad, tetha1deg, alpha2rad, alpha2deg,  
tetha2rad, tetha2deg;
```

```
float K1 = (300/1023);//Faktor Konversi dari Resolusi (bit) ke Sudut (derajat)
```

```
float K2 = (1023/300);//Faktor Konversi dari Sudut (derajat) ke Resolusi (bit)
```

```
float K3 = (180/PI);//Faktor Konversi dari Rad ke Derajat
```

```
float K4 = (PI/180);//Faktor Konversi dari Derajat ke Radian
```

```
void setup()
```

```
{
```

```
Serial.begin(115200);
```

```
Dynamixel.begin(1000000,A8); // Initialize the servo at 1Mbps and Pin Control 2
```

```
delay(500);
```

```
// Dynamixel.moveSpeed(19,512,1000);
```

```
}
```

```
void loop() {
```

```
// put your main code here, to run repeatedly:
```

```
c = sqrt(pow(Y0,2)+pow(X0,2));
```

```
alpha1rad = atan2(Y0,X0);
```

```
alpha1deg = alpha1rad*K3;
```

```
phi1rad = acos (pow(a,2)+pow(c,2)- pow(b,2))/(2*a*c);
```

```
phi1deg = phi1rad*K3;
```

```
if (alpha1deg >=0)
```

```
{
```

```
    ttheta1deg = alpha1deg-phi1deg;
```

```
} //Solusi servo link bawah jika alpha1deg positif
```

```
else
```

```
{
```

```
    ttheta1deg = alpha1deg+phi1deg;
```

```
//Solusi servo link bawah jika alpha1deg negatif
```

```
alpha2rad = acos (pow(a,2)+pow(b,2)-pow(c,2))/(2*a*b);
```

```
alpha2deg = alpha1rad*K3;
```

```
if (alpha1deg >=0)
```

```
{
```

```
    ttheta2deg = 180-alpha2deg;
```

```
} //Solusi servo link atas jika alpha1deg positif
```

```
else
```

```
{
```

```
    ttheta2deg = -1*(180-alpha2deg);
```

```
//Solusi servo link atas jika alpha1deg negatif
```

//Menampilkan hasil perhitungan ke serial monitor:

```
Serial.print("tetha1deg = ");
```

```
Serial.print(tetha1deg);
```

```
Serial.print("tetha1deg = ");
```

```
Serial.println(tetha2deg);
```

```
delay(200);
```

//Pergerakan Servo

```
if(tetha1deg)
```

```
    Dynamixel.moveSpeed(2,461,1000);
```

```
    Dynamixel.moveSpeed(3,563,1000);
```

```
if(tetha2deg)
```

```
    Dynamixel.moveSpeed(4,430,1000);
```

```
    Dynamixel.moveSpeed(5,594,1000);
```

```
    Dynamixel.moveSpeed(7,currentposg,1000);
```

```
}
```

### Program Antarmuka Kinematic Software (Control)

#### Mainwindow

```
#include "mainwindow.h"
```

```
#include "ui_mainwindow.h"
```



```
MainWindow::MainWindow(QWidget *parent) :  
    QMainWindow(parent),  
    ui(new Ui::MainWindow)  
{  
    ui->setupUi(this);  
}
```

```
MainWindow::~MainWindow()  
{  
    delete ui;
```

## Main

```
#include <QApplication>  
#include <QLabel>  
#include <QPushButton>  
#include <QLineEdit>  
#include <QHBoxLayout>  
#include <QGridLayout>  
#include <QGroupBox>  
#include "file_control.h"  
  
int main(int argc, char *argv[])  
{  
    QApplication a(argc, argv);
```

```
    QWidget *w = new QWidget();
```

```
    file_control *control = new file_control();
```

```
    QGridLayout *mainLayout = new QGridLayout();
```



```
QGridLayout *layout1 = new QGridLayout();
QGridLayout *layout2 = new QGridLayout();
QGridLayout *layout3 = new QGridLayout();
QGridLayout *layout4 = new QGridLayout();
QGridLayout *layout5 = new QGridLayout();

QGroupBox *groupPort = new QGroupBox("Setting Port");
QGroupBox *groupTheta = new QGroupBox("Theta");
QGroupBox *groupPosition = new QGroupBox("Position");
QGroupBox *groupBut = new QGroupBox();
QGroupBox *groupLabel = new QGroupBox();

QLabel *labelPortName = new QLabel("PORT NAME :");
QLabel *labelConnect = new QLabel("<font color = red>NotConnect</font>");
QLabel *labelTheta1 = new QLabel("Thetha 1 :");
QLabel *labelTheta2 = new QLabel("Thetha 2 :");
QLabel *labelPosition1 = new QLabel("Position 1 :");
QLabel *labelPosition2 = new QLabel("Position 2 :");
QLabel *labelLink = new QLabel("Link");
QLabel *labelLink1 = new QLabel("I");
QLabel *labelLink2 = new QLabel("II");
QLabel *labelAi = new QLabel("ai");
QLabel *labelAi1 = new QLabel("17");
QLabel *labelAi2 = new QLabel("23");
QLabel *labelAlfa = new QLabel("alpha i");
QLabel *labelAlfa1 = new QLabel("alpha 1");
QLabel *labelAlfa2 = new QLabel("alpha 2");
QLabel *labelDi = new QLabel("di");
QLabel *labelDi1 = new QLabel("0");
QLabel *labelDi2 = new QLabel("0");
QLabel *labelTheta = new QLabel("Theta");
```

```
QLabel *labelThetaA = new QLabel("Thetha 1");
QLabel *labelThetaB = new QLabel("Thetha 2");

QLineEdit *linePortName = new QLineEdit();
QLineEdit *lineTheta1 = new QLineEdit();
QLineEdit *lineTheta2 = new QLineEdit();
QLineEdit *linePosition1 = new QLineEdit();
QLineEdit *linePosition2 = new QLineEdit();

QPushButton *butConnect = new QPushButton("Connect");
QPushButton *butTheta = new QPushButton("Forward");
QPushButton *butPosition = new QPushButton("Inverse");
QPushButton *butAuto = new QPushButton("Auto");
QPushButton *butReset = new QPushButton("Reset");

layout2->addWidget(labelTheta1,0,0,1,1);
layout2->addWidget(lineTheta1,0,1,1,1);
layout2->addWidget(labelTheta2,1,0,1,1);
layout2->addWidget(lineTheta2,1,1,1,1);

layout3->addWidget(labelPosition1,0,0,1,1);
layout3->addWidget(linePosition1,0,1,1,1);
layout3->addWidget(labelPosition2,1,0,1,1);
layout3->addWidget(linePosition2,1,1,1,1);

layout1->addWidget(labelPortName,0,0,1,1);
layout1->addWidget(linePortName,0,1,1,1);
layout1->addWidget(butConnect,1,1,1,1);
layout1->addWidget(labelConnect,2,1,1,1);

layout4->addWidget(butTheta,0,0,1,1);
```

```
layout4->addWidget(buttonPosition,0,1,1,1);
layout4->addWidget(buttonAuto,1,0,1,1);
layout4->addWidget(buttonReset,1,1,1,1);

layout5->addWidget(labelLink,0,0,1,1);
layout5->addWidget(labelLink1,1,0,1,1);
layout5->addWidget(labelLink2,2,0,1,1);
layout5->addWidget(labelAi,0,1,1,1);
layout5->addWidget(labelAi1,1,1,1,1);
layout5->addWidget(labelAi2,2,1,1,1);
layout5->addWidget(labelAlfa,0,2,1,1);
layout5->addWidget(labelAlfa1,1,2,1,1);
layout5->addWidget(labelAlfa2,2,2,1,1);
layout5->addWidget(labelDi,0,3,1,1);
layout5->addWidget(labelDi1,1,3,1,1);
layout5->addWidget(labelDi2,2,3,1,1);
layout5->addWidget(labelTheta,0,4,1,1);
layout5->addWidget(labelThetaA,1,4,1,1);
layout5->addWidget(labelThetaB,2,4,1,1);

groupPort->setLayout(layout1);
groupTheta->setLayout(layout2);
groupPosition->setLayout(layout3);
groupBut->setLayout(layout4);
groupLabel->setLayout(layout5);

mainLayout->addWidget(groupPort,0,0,1,1);
mainLayout->addWidget(groupTheta,0,1,1,1);
mainLayout->addWidget(groupPosition,0,2,1,1);
mainLayout->addWidget(groupBut,1,0,1,1);
mainLayout->addWidget(groupLabel,1,1,2,2);
```

```
w->setLayout(mainLayout);
w->show();

QObject::connect(linePortName,SIGNAL(textChanged(QString)),control,SLOT(setPortName(QString)));
QObject::connect(linePosition1,SIGNAL(textChanged(QString)),control,SLOT(getPosition1(QString)));
QObject::connect(linePosition2,SIGNAL(textChanged(QString)),control,SLOT(getPosition2(QString)));
QObject::connect(lineTheta1,SIGNAL(textChanged(QString)),control,SLOT(getTheta1(QString)));
QObject::connect(lineTheta2,SIGNAL(textChanged(QString)),control,SLOT(getTheta2(QString)));
QObject::connect(butConnect,SIGNAL(clicked()),control,SLOT(setPort()));
QObject::connect(butTheta,SIGNAL(clicked()),control,SLOT(sendTheta()));
QObject::connect(butPosition,SIGNAL(clicked()),control,SLOT(sendPosition()));
QObject::connect(butAuto,SIGNAL(clicked()),control,SLOT(sendAuto()));
QObject::connect(butReset,SIGNAL(clicked()),control,SLOT(sendReset()));

QObject::connect(control,SIGNAL(stringConnect(QString)),labelConnect,SLOT(setText(QString)));
return a.exec();
}
```

## Control

```
#include "file_control.h"

file_control::file_control(QObject *parent) :
    QThread(parent)
{
    portme = new QSerialPort(this);
}

void file_control::setPort()
{
    portme->setPortName(portName);
    portme->setBaudRate(QSerialPort::Baud9600);
    portme->setDataBits(QSerialPort::Data8);
    portme->setFlowControl(QSerialPort::NoFlowControl);
    portme->setParity(QSerialPort::NoParity);
    portme->setStopBits(QSerialPort::OneStop);
    portme->open(QIODevice::WriteOnly);
    connect = isConnect();
    if(connect == true)
        stringConnect("<font color=green>Connect</font>");
    else
        stringConnect("<font color=red>Disconnect</font>");
}

void file_control::setPortName(QString x)
{
    portName = x;
}

bool file_control::isConnect()
{
    if(portme->isOpen()){
        return true;
        qDebug()<<"true";
    }
}
```

```
    else{
        qDebug()<<"fail";
        return false;
    }

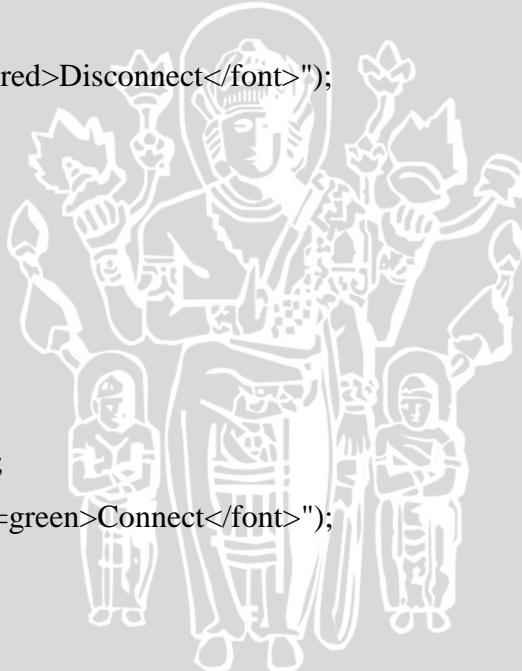
void file_control::sendAuto()
{
    connect = isConnected();
    if(connect == true)
    {
        portme->write("c");
    }
    else
    {
        stringConnect("<font color=red>Disconnect</font>");
    }
}

void file_control::sendReset()
{
    if(connect == true)
    {
        portme->write("d");
    }
    else
    {
        stringConnect("<font color=red>Disconnect</font>");
    }
}

void file_control::sendPosition()
{
    connect = isConnected();
    if(connect == true){
```



```
stringConnect("<font color=green>Connect</font>");  
qDebug()<<position1<<position2;  
portme->write("a");  
portme->write(position1);  
portme->write(position2);  
portme->write(" ");  
portme->write(sign1,1);  
portme->write(" ");  
portme->write(sign2,1);  
}  
else  
{  
    stringConnect("<font color=red>Disconnect</font>");  
}  
}  
}  
void file_control::sendTheta()  
{  
    connect = isConnected();  
    if(connect == true){  
        qDebug()<<theta1<<theta2;  
        stringConnect("<font color=green>Connect</font>");  
        portme->write("b");  
        portme->write(theta1);  
        portme->write(theta2);  
        portme->write(" ");  
        portme->write(sign1,1);  
        portme->write(" ");  
        portme->write(sign2,1);  
    }  
    else  
{
```



```
        stringConnect("<font color=red>Disconnect</font>");  
    }  
}  
  
void file_control::getPosition1(QString x)  
{  
    posiFloat1 = x.toFloat();  
    if(posiFloat1 > 0){  
        position1 = x.toLatin1();  
        sign1[0] = '1';  
    }  
    else{  
        posiFloat1 = posiFloat1*-1.00;  
        position1.setNum(posiFloat1);  
        sign1[0] = '0';  
    }  
}  
  
void file_control::getPosition2(QString x)  
{  
    posiFloat2 = x.toFloat();  
    if(posiFloat2 > 0){  
        position2 = x.toLatin1();  
        sign2[0] = '1';  
    }  
    else{  
        posiFloat2 = posiFloat2*-1.00;  
        position2.setNum(posiFloat2);  
        sign2[0] = '0';  
    }  
}  
  
void file_control::getTheta1(QString x)  
{
```



```
thethaFloat1 = x.toFloat();
if(thethaFloat1 > 0){
    theta1 = x.toLatin1();
    sign1[0] = '1';
}
else{
    thethaFloat1 = thethaFloat1*-1.00;
    theta1.setNum(thethaFloat1);
    sign1[0] = '0';
}

void file_control::getTheta2(QString x)
{
    thethaFloat2 = x.toFloat();
    if(thethaFloat2 > 0){
        theta2 = x.toLatin1();
        sign2[0] = '1';
    }
    else{
        thethaFloat2 = thethaFloat2*-1.00;
        theta2.setNum(thethaFloat2);
        sign2[0] = '0';
    }
}
```



## **Program Perhitungan Manual Dengan MATLAB**

*Forward Kinrmatic :*

$\text{tetha1}=15, \text{tetha2}=24, a=(\pi/180), b=17, c=23,$

$$Px = ((b * \cos(\text{tetha1} * a)) + (c * \cos((\text{tetha1} + \text{tetha2}) * a)))$$

$$Py = ((b * \sin(\text{tetha1} * a)) + (c * \sin((\text{tetha1} + \text{tetha2}) * a)))$$

*Invers Kinrmatic :*

$$Y0=18.87435$$

X0=34.2951

a=17

b=23

c=sqrt((Y0^2)+(X0^2))

d=(180/pi)

e= (Y0/X0)

f=((a^2)+(c^2)-(b^2))/(2\*a\*c)

g=atan (e)

h=acos(f)

alpha1= g\*d

phi1=h\*d

if alpha1 >=0

    tetha1=(alpha1)-(phi1)%Solusi Motor Servo Link 1 jika alpha1 positif

else

    tetha1=(alpha1)+(phi1)%Solusi Motor Servo Link 1 jika alpha 1 negatif

end

i=((a^2)+(b^2)-(c^2))/(2\*a\*b)

j=acos(i)

alpha2= j\*d

if alpha1 >=0

    tetha2=(180-alpha2)%Solusi Motor Servo Link 2 jika alpha1 positif

else

    tetha1= -(180-alpha2)%Solusi Motor Servo Link 2 jika alpha 1 negatif

end

## **LAMPIRAN III**

### ***DATA SHEET***

