

LAMPIRAN

Lampiran 1 : Spesifikasi motor PG45

No	Parameter	Nilai dan satuan
1	Diameter <i>body</i> motor	45 mm
2	Panjang <i>body</i> motor	125 mm
3	Rasio <i>gear box</i>	1 : 19
4	Diameter <i>shaft</i> motor	10 mm
5	Arus maksimal	4 ampere
6	Tegangan	24 volt DC
7	Daya	60 watt
8	Torsi konstan	25 kg/cm
9	Torsi <i>stall</i>	50 kg/cm
10	<i>Speed</i>	500 rpm
11	Berat	800 gram
12	Pulsa <i>rotary encoder</i>	7 ppr
13	<i>Chanel rotary encoder</i>	2 <i>chanel</i>

Lampiran 2 : Spesifikasi arduino mega 2560

No	Parameter	Nilai dan satuan
1	Total pin I/O digital	54 pin
2	Pin <i>pwm</i>	14 pin
3	Pin <i>input</i> analog	16 pin
4	Pin <i>UART</i>	4 pin
5	Kristal	16 MHz
6	Tegangan operasi	5 volt DC
7	Tegangan <i>input</i>	7 – 12 volt DC
8	Arus DC per <i>I / O</i> pin	40 mA
9	<i>Flash memory</i>	256 KB

Lampiran 3 : Spesifikasi *driver* motor BTS7960

No	Parameter	Nilai dan satuan
1	Pin MCU ke <i>driver</i>	4 pin (<i>gnd, vcc, input 1, input 2</i>)
2	<i>Isolation chip power supply</i>	5 volt DC
3	Dimensi	4 * 5 * 1.2 cm
4	Frekuensi <i>pwm</i> maksimal	25 KHz
5	Tegangan masukan motor	5,5 volt – 27 volt DC
6	Tegangan logika	3,3 volt – 5 volt
7	Arus untuk beban	43 Ampere
8	Perlindungan	Hubung singkat, over tegangan, tegangan bawah, <i>overheat</i> .

Lampiran 4 : Spesifikasi MPU6050

Pin	Nama Pin	Deskripsi
1	Vcc	+3V sampai +5V
2	Ground	Terhubung ke Ground sistem
3	Serial Clock (SCL)	Digunakan untuk menyediakan pulsa clock untuk Komunikasi I2C
4	Serial Data (SDA)	Digunakan untuk mentransfer data melalui komunikasi I2C
5	Auxiliary Serial Data (XDA)	Dapat digunakan untuk antarmuka modul I2C lainnya dengan MPU6050. Itu opsional
6	Auxiliary Serial Clock (XCL)	Dapat digunakan untuk antarmuka modul I2C lainnya dengan MPU6050. Itu opsional
7	AD0	Jika lebih dari satu MPU6050 digunakan satu MCU, maka pin ini dapat digunakan untuk memvariasikan alamat
8	Interrupt (INT)	Pin interupst untuk menunjukkan bahwa data tersedia untuk dibaca oleh MCU.

Lampiran 5: Program

```
#include <MsTimer2.h>

#include <LiquidCrystal.h>

LiquidCrystal lcd(34, 36, 38, 40, 42, 44);

#define sw6 41

#define sw5 33

#define sw4 35

#define sw3 37

#define sw2 39

#define sw1 31

#define pwmL1 11

#define pwmL2 10

#define pwmR1 5

#define pwmR2 4

#define pwmB1 7

#define pwmB2 6

#define pwmD1 13

#define pwmD2 12

#define intPinL 3

#define intPinR 20

#define intPinB 21

#define digPinL 30

#define digPinR 26

#define digPinB 28

float const maxRPM = 530;

float const rRobot = 0.185; //18.5 cm

float const dRobot = 0.37; //37 cm

float const perRotasi1 = 800; //270 pulsa

float const dRoda = 10; //10 cm
```

```

float const dRoda1 = 6.5; //6.5 cm

float const kelRoda = (dRoda * 22 / 7);

float const kelRoda1 = (dRoda1 * 22 / 7);

float const jarPerPulsal = kelRoda1 / perRotasil;

float rpmL, rpmR, rpmB, pwmL, pwmR, pwmB, jarL, jarR, jarB, jarL1,
jarR1, jarB1, kecL, kecR, kecB, errorL, errorR, errorB, Theta,
rTheta, T, rotpsL, rotpsR, dataGyro,error, errorl, lastError,
lastErrorl, PID, mmplsL, mmplsR, mmplsB, errorL1, errorR1, errorB1,
errorIL, errorIR, errorIB, errorIB1, radpsL, radpsR,vL, vR, vB, vX,
vY, vT, iX, iY, iT, derajat, errorX, errorY, errorX1, errorY1,
errorT, xTarget, yTarget, tTarget;

float jarX = 0, jarY = 0;

String inString = "";

int m, titikTuju;

char data;

long count, L, R, B, intL, intR, intB, frekL, frekR, frekB;

int posisiBola, detekBola = 0, detekBola360 = 0, countRun,
countTendang, countManual, detikManual, detikRun, detikRunBertahan,
detikRunSerang, countDetek, countGaDetek;

void setup() {

    pinMode(pwmL1, OUTPUT);

    pinMode(pwmL2, OUTPUT);

    pinMode(pwmR1, OUTPUT);

    pinMode(pwmR2, OUTPUT);

    pinMode(pwmB1, OUTPUT);

    pinMode(pwmB2, OUTPUT);

    pinMode(9, OUTPUT);

    pinMode(8, OUTPUT);

    digitalWrite(9, 1);

    digitalWrite(10, 1);

    MsTimer2::set(100, interrupt0);

    MsTimer2::start();

```

```

    attachInterrupt(digitalPinToInterrupt(intPinL), interruptL,
CHANGE);

    attachInterrupt(digitalPinToInterrupt(intPinR), interruptR,
CHANGE);

    lcd.begin(20, 4);

    Serial.begin(9600);

    Serial1.begin(9600);

    Serial2.begin(9600);

    Serial3.begin(38400);

    pinMode(sw1, INPUT_PULLUP);
    pinMode(sw2, INPUT_PULLUP);
    pinMode(sw3, INPUT_PULLUP);
    pinMode(sw4, INPUT_PULLUP);
    pinMode(sw5, INPUT_PULLUP);
    pinMode(sw6, INPUT_PULLUP);

    pinMode(intPinL, INPUT_PULLUP);
    pinMode(intPinR, INPUT_PULLUP);
    pinMode(digPinL, INPUT_PULLUP);
    pinMode(digPinR, INPUT_PULLUP);

    motor(0, 0, 0);
}

byte baruReset = 'Y';

void loop() {
    if(Theta < 0){rTheta = 360 + Theta;}
    else{rTheta = Theta;}

    lcd.setCursor(0, m); lcd.print(">");

    lcd.setCursor(1, 0); lcd.print("Kinematik ");

    lcd.setCursor(1, 1); lcd.print("Gyro : "); lcd.print(Theta);
    lcd.print(" "); lcd.print((int)rTheta);    lcd.print(" ");

    lcd.setCursor(1, 2);

```

```

    lcd.print("x"); lcd.print(jarX); lcd.print(" y");
    lcd.print(jarY); lcd.print(" ");

    lcd.setCursor(1, 3);

    lcd.print("L"); lcd.print(intL); lcd.print(" R");
    lcd.print(intR); lcd.print(" B"); lcd.print(intB); lcd.print("
");

    if (O() && m == 0) { testRunKinematik(); }

    if (U()) {lcd.setCursor(0, m); lcd.print(" "); m--; delay(150);
}

    if (D()) {lcd.setCursor(0, m); lcd.print(" "); m++; delay(150);
}

    if (m < 0) { m = 1; }

    if (m > 1) { m = 0; }

    if (C() && m == 1) { Serial3.println("0#"); intL = intR = jarX
= jarY = jarL = jarR = jarB = 0; }

    else if (C()) { motor(0, 0, 0); }

}

void interrupt0() {

    jarL = jarPerPulsal * frekL;

    jarR = jarPerPulsal * frekR;

    updateJarak();

    frekL = frekR = 0;

}

void interruptL() {

    if (digitalRead(intPinL) != digitalRead(digPinL)) {

        intL++;

        frekL++;

    }

    else {

        intL--;

        frekL--;

    }

}
}

```



```
void interruptR() {
    if (digitalRead(intPinR) != digitalRead(digPinR)) {
        intR++;
        frekR++;
    }
    else {
        intR--;
        frekR--;
    }
}

bool O() {
    if (digitalRead(sw1) == 0) {
        return true;
    }
    else {
        return false;
    }
}

bool D() {
    if (digitalRead(sw2) == 0) {
        return true;
    }
    else {
        return false;
    }
}

bool U() {
    if (digitalRead(sw3) == 0) {
        return true;
    }
}
```

```
        else {
            return false;
        }
    }
bool P() {
    if (digitalRead(sw4) == 0) {
        return true;
    }
    else {
        return false;
    }
}
bool M() {
    if (digitalRead(sw5) == 0) {
        return true;
    }
    else {
        return false;
    }
}
bool C() {
    if (digitalRead(sw6) == 0) {
        return true;
    }
    else {
        return false;
    }
}
void motor(int L, int R, int B) {
    if (L > 0) {
```

```
    analogWrite(pwmL1, L);
    digitalWrite(pwmL2, 0);
} else if (L < 0) {
    analogWrite(pwmL1, 255 + L);
    digitalWrite(pwmL2, 1);
} else {
    digitalWrite(pwmL1, 0);
    digitalWrite(pwmL2, 0);
}
if (R > 0) {
    analogWrite(pwmR1, R);
    digitalWrite(pwmR2, 0);
} else if (R < 0) {
    analogWrite(pwmR1, 255 + R);
    digitalWrite(pwmR2, 1);
} else {
    digitalWrite(pwmR1, 0);
    digitalWrite(pwmR2, 0);
}
if (B > 0) {
    analogWrite(pwmB1, B);
    digitalWrite(pwmB2, 0);
} else if (B < 0) {
    analogWrite(pwmB1, 255 + B);
    digitalWrite(pwmB2, 1);
} else {
    digitalWrite(pwmB1, 0);
    digitalWrite(pwmB2, 0);
}
pwmL = L;
```

```

    pwmR = R;
    pwmB = B;
}
void maju(int L, int R) {
    motor(-L, R, 0);
}
void mundur(int L, int R) {
    motor(L, -R, 0);
}
void geserKiri(int L) {
    motor(L / 2, L / 2, -L);
}
void geserKanan(int R) {
    motor(-R / 2, -R / 2, R);
}
void berhenti() {
    motor(0, 0, 0);
}
void parseGyro() {
    int dataG;
    while (Serial3.available() > 0) {
        int inChar = Serial3.read();

        if (inChar != '\n') {
            inString += (char)inChar;
        }
        else {
            dataGyro = inString.toFloat();
            inString = "";
        }
    }
}

```

```

    }
}

void updateJarak() {
    float vX, vY, w;
    float offsetX = 1.01, offsetY = 1.05;
    vX = (-cos(radians(45)) * jarR - cos(radians(45)) * jarL);
    vY = (sin(radians(45)) * jarR - sin(radians(45)) * jarL);
    jarX += (cos(radians(Theta)) * vX + sin(radians(Theta)) * vY) *
offsetX;
    jarY += (-sin(radians(Theta)) * vX + cos(radians(Theta)) * vY) *
offsetY;
    parseGyro(); //Theta=dataGyro;
    if (dataGyro < 0) {
        Theta = 360 + dataGyro;
    } else {
        Theta = dataGyro;
    }
    if (Theta > 180) {
        Theta = Theta - 360;
    }
}

void displayJarak() {
    lcd.setCursor(0, 1);
    lcd.print("X"); lcd.print((int)jarX);
    lcd.print(" Y"); lcd.print((int)jarY);
    lcd.print(" T"); lcd.print((int)Theta); lcd.print("  ");
    lcd.setCursor(0, 2);
    lcd.print("L"); lcd.print((int)intL);
    lcd.print(" R"); lcd.print((int)intR);
    lcd.print(" B"); lcd.print((int)intB); lcd.print("  ");
}

```

```

void runKinematik(int x, int y, int t, int maxPWM) {
    float kpXY = 3.3, kiXY = 0.095, kdXY = 0;
    float kpT = 2.8, kiT = 0.07, kdT = 0, maxRPM = maxPWM;
    float eTheta, pTheta, mTheta, batasPlus, batasMin;
    errorX = x - jarX;
    errorY = y - jarY;
    if(Theta < 0){pTheta = 360 + Theta;}
    else{pTheta = Theta;}
    mTheta = Theta;
    if(t<0){ eTheta = 360 + t; }
    else{ eTheta = t; }
    if(pTheta > eTheta){errorT = pTheta - eTheta;}
    else {errorT = Theta - t;}
    if(errorT>180){errorT =(int) (errorT - 360);}
    long r = sqrt(((errorX / 10) * (errorX / 10)) + ((errorY / 10) *
(errorY / 10)));
    r *= 10;
    if (r < 15) {
        iX += errorX;
        iY += errorY;
        if (errorT < 15 && errorT > -15) { iT += errorT; }
    }else{
        iX=iY=iT=0;
    }
    vX = errorX * kpXY + (iX * kiXY);
    vY = errorY * kpXY + (iY * kiXY);
    vT = errorT * kpT + (iT * kiT);
    if(vX>0 && vY>0){ //Kuadran I
        if(vX>maxPWM && vX>vY){
            vY=maxPWM*vY/vX;
            vX=maxPWM;

```

```

    }else if(vY>maxPWM && vY>vX){
        vX=maxPWM*vX/vY;
        vY=maxPWM;
    }
}else if(vX<0 && vY>0){ //Kuadran II
    if(-vX>maxPWM && -vX>vY){
        vY=maxPWM*vY/-vX;
        vX=-maxPWM;
    }else if(vY>maxPWM && vY>-vX){
        vX=maxPWM*vX/vY;
        vY=maxPWM;
    }
}else if(vX>0 && vY<0){ //Kuadran III
    if(vX>maxPWM && vX>-vY){
        vY=maxPWM*vY/vX;
        vX=maxPWM;
    }else if(-vY>maxPWM && -vY>vX){
        vX=maxPWM*vX/-vY;
        vY=-maxPWM;
    }
}else if(vX<0 && vY<0){ //Kuadran IV
    if(-vX>maxPWM && -vX>-vY){
        vY=maxPWM*vY/-vX;
        vX=-maxPWM;
    }else if(-vX>maxPWM && -vY>-vX){
        vX=maxPWM*vX/-vY;
        vY=-maxPWM;
    }
}

vB = ((vY) * (-sin(radians(Theta)))) + ((vX) *
(cos(radians(Theta)))) + ((vT / 3));

```

```

    vL = ((vY) * (-sin(radians(60 - Theta)))) + ((vX) * (-
cos(radians(60 - Theta)))) + ((vT / 3));

    vR = ((vY) * ( sin(radians(60 + Theta)))) + ((vX) * (-
cos(radians(60 + Theta)))) + ((vT / 3));

    if (vL > maxRPM) {
        vL = maxRPM;
    }

    if (vR > maxRPM) {
        vR = maxRPM;
    }

    if (vB > maxRPM) {
        vB = maxRPM;
    }

    if (vL < -maxRPM) {
        vL = -maxRPM;
    }

    if (vR < -maxRPM) {
        vR = -maxRPM;
    }

    if (vB < -maxRPM) {
        vB = -maxRPM;
    }

    if ((errorX < 4 && errorX > -4) && (errorY < 4 && errorY > -4)
&& (errorT < 2 && errorT > -2)) {
        motor(0, 0, 0);
        iX = iY = iT = 0;
        titikTuju = 1;
    }

    else {
        motor(vL, vR, vB);
        titikTuju = 0;
    }
}

```



```
}  
void testRunKinematik() {  
    while (!C()) {  
        lcd.clear();  
        while (!C() && titikTuju != 1) {  
            runKinematik(0, 210, 0, 50);  
            displayJarak();  
        } titikTuju = 1; displayJarak();  
        while (!C() && titikTuju != 1) {  
            runKinematik(0, 0, 0, 50);  
            displayJarak();  
        } //titikTuju = 1;  
    }  
    motor(0, 0, 0); delay(300); lcd.clear();  
}
```



LEMBAR REVISI dan TUGAS UJIAN SARJANA

Berdasarkan Rapat Tim Penguji Ujian Sarjana

Hari : Senin
Tanggal : 23 September 2019
Tempat : R. Lab TE

Memutuskan bahwa mahasiswa :

Nama : Eka Kulana Pambudi
NIM : 30601501708
Judul TA : Implementasi Trajektori dengan Sensor MPU 6050
sebagai Pembacaan Arah pada Robot Sepak Bola Beroda
untuk Menentukan Posisi Robot

wajib melakukan perbaikan dan membuat tugas seperti tercantum dibawah ini:

NO	REVISI	BATAS REVISI
	<i>tambahan desain pengujian → bab 3 Hasil → bab 4.</i>	<i>3/4.</i> <i>OK</i>

NO	TUGAS

Mengetahui,
Ketua Tim Penguji

Muhammad Khosyil'in, ST, MT
NIDN. 0625077901

Semarang, 23 September 2019
Penguji, I

Muhammad Khosyil'in, ST, MT
NIDN. 0625077901



LEMBAR REVISI dan TUGAS UJIAN SARJANA

Berdasarkan Rapat Tim Penguji Ujian Sarjana

Hari : Senin
 Tanggal : 23 September 2019
 Tempat : R. Lab TE

Memutuskan bahwa mahasiswa :

Nama : Eka Kulana Pambudi
 NIM : 30601501708
 Judul TA : Implementasi Trajektori dengan Sensor MPU 6050 sebagai Pembacaan Arah pada Robot Sepak Bola Beroda untuk Menentukan Posisi Robot

wajib melakukan perbaikan dan membuat tugas seperti tercantum dibawah ini:

NO	REVISI	BATAS REVISI
1	Space put dfttr gbr, dfttr tabel, Header tabel	
2	Urutan penomoran, huruf awalan "Italic"	
3	Teori ttr arduino dilengkap.	
4	Rumus 2.1 didapat dr mana? di citaci	
5	Rumus duty cycle blm ada	
6	Teori cara kerja LCD blm ada	
7	Teori cara kerja diagram dr driver motor blm ada	
8	Gbr flowchart 3.1	
9	Rumus 3.1 mpla ke bab 2	

Acc

 Budi P. J.
 1/10/19

NO	TUGAS
1	Cara kerja Gyroscopy (bab 2)

Mengetahui,
 Ketua Tim Penguji

Muhammad Khosyil'in, ST, MT
 NIDN. 0625077901

Semarang, 23 September 2019
 Penguji, II

Ir. Budi Pramono Jati, MM, MT
 NIDN. 0623126501

10. Antena 1.2 1.4 5.1 fsk simbol



LEMBAR REVISI dan TUGAS UJIAN SARJANA

Berdasarkan Rapat Tim Penguji Ujian Sarjana

Hari : Senin
Tanggal : 23 September 2019
Tempat : R. Lab TE

Memutuskan bahwa mahasiswa :

Nama : Eka Kulana Pambudi
NIM : 30601501708
Judul TA : Implementasi Trajektori dengan Sensor MPU 6050
sebagai Pembacaan Arah pada Robot Sepak Bola Beroda
untuk Menentukan Posisi Robot

wajib melakukan perbaikan dan membuat tugas seperti tercantum dibawah ini:

NO	REVISI	BATAS REVISI
	<i>Cek label rps - rps A64</i>	<i>[Signature]</i>

NO	TUGAS

Mengetahui,
Ketua Tim Penguji

Muhammad Khosyi'in, ST, MT
NIDN. 0625077901

Semarang, 23 September 2019
Penguji, III

[Signature]
Ir.H. Sukarno Budi Utomo, MT
NIDN. 0619076401

IMPLEMENTASI TRAJEKTORI DENGAN SENSOR MPU 6050 SEBAGAI PEMBACAAN ARAH PADA ROBOT SEPAK BOLA BERODA UNTUK MENENTUKAN POSISI ROBOT

by eka pambudi

Submission date: 16-Sep-2019 04:28PM (UTC+0800)

Submission ID: 1173596006

File name: Laporan_TA.pdf (813.15K)

Word count: 10206

Character count: 59611

IMPLEMENTASI TRAJEKTORI DENGAN SENSOR MPU 6050 SEBAGAI PEMBACAAN ARAH PADA ROBOT SEPAK BOLA BERODA UNTUK MENENTUKAN POSISI ROBOT

ORIGINALITY REPORT

21%	16%	1%	19%
SIMILARITY INDEX	INTERNET SOURCES	PUBLICATIONS	STUDENT PAPERS

PRIMARY SOURCES

1	Submitted to Sultan Agung Islamic University Student Paper	5%
2	Submitted to Universitas Brawijaya Student Paper	2%
3	repository.its.ac.id Internet Source	2%
4	Submitted to Universitas Muria Kudus Student Paper	1%
5	www.scribd.com Internet Source	1%
6	Submitted to Universitas Andalas Student Paper	1%
7	kontesrobotindonesia.id Internet Source	1%
8	yuhardiansyahblog.wordpress.com Internet Source	1%

9	Submitted to Politeknik Negeri Bandung Student Paper	1%
10	Submitted to Universitas Negeri Jakarta Student Paper	1%
11	docplayer.info Internet Source	<1%
12	repository.unhas.ac.id Internet Source	<1%
13	Submitted to Sriwijaya University Student Paper	<1%
14	digilib.unila.ac.id Internet Source	<1%
15	eprints.umm.ac.id Internet Source	<1%
16	Submitted to Universitas Jember Student Paper	<1%
17	media.neliti.com Internet Source	<1%
18	eprints.uny.ac.id Internet Source	<1%
19	teknikelektronika.com Internet Source	<1%
20	mission-ug.blogspot.com Internet Source	

<1%

21 aimos.ugm.ac.id
Internet Source

<1%

22 www.android-1.xyz
Internet Source

<1%

23 digilib.unimed.ac.id
Internet Source

<1%

24 etheses.uin-malang.ac.id
Internet Source

<1%

25 Submitted to Universitas International Batam
Student Paper

<1%

26 www.pens.ac.id
Internet Source

<1%

27 Submitted to STIKOM Surabaya
Student Paper

<1%

28 eprints.uns.ac.id
Internet Source

<1%

29 repository.usu.ac.id
Internet Source

<1%

30 repo.iain-tulungagung.ac.id
Internet Source

<1%

31 charisincorporation.blogspot.com
Internet Source

<1%

32	pt.scribd.com Internet Source	<1%
33	www.kepogaul.com Internet Source	<1%
34	alwasathiyah.com Internet Source	<1%
35	Submitted to Universitas Islam Indonesia Student Paper	<1%
36	text-id.123dok.com Internet Source	<1%
37	repository.uksw.edu Internet Source	<1%
38	ejournal.undip.ac.id Internet Source	<1%
39	Submitted to Universitas Dian Nuswantoro Student Paper	<1%
40	Submitted to Universitas Muhammadiyah Sidoarjo Student Paper	<1%
41	www.physics.leidenuniv.nl Internet Source	<1%
42	sir.stikom.edu Internet Source	<1%

43

Submitted to President University

Student Paper

<1%

44

Submitted to Universitas Jenderal Achmad Yani

Student Paper

<1%

45

ojs.uniska-bjm.ac.id

Internet Source

<1%

46

repository.upi.edu

Internet Source

<1%

47

id.123dok.com

Internet Source

<1%

48

eprints.akakom.ac.id

Internet Source

<1%

49

Submitted to Udayana University

Student Paper

<1%

50

Submitted to Universitas Pancasila

Student Paper

<1%

51

Submitted to Universitas Putera Batam

Student Paper

<1%

52

Submitted to Universitas Kristen Satya Wacana

Student Paper

<1%

Exclude quotes

On

Exclude matches

Off