

SIMULASI DESAIN SKEMA ROBOT *LINE FOLLOWER* PENYIRAM TANAMAN MENGGUNAKAN SIMULATOR SOFTWARE PROTEUS 8

Rini Nuraini

Universitas Bunda Mulia, Prodi Teknik Informatika
neneng.rininuraini@yahoo.co.id

ABSTRACT

The problem of high wages of labor today is very consideration to do things that routine work done every day, as well as watering plants, for this case, need a solution to the problem, that is by making a robot to do it. Robot Line Follower Sprinkler Plants or the next one will be shortened to LFPT Robot, the goal to do the work routinely watering the plant by way of the robot is running while watering the plants on the track. The LFPT Robot Characteristics of the robot will run following a specified black line, with the help of two IR sensors on the front of the robot for running, as well as an IR sensor on the left and right side of the LFPT Robot to detect the presence of plants. If the sensor on the left and right side of the robot detects a black line, the LFPT Robot will stop and will direct the water hose to the left and right then the robot will do the watering. This study uses a series of IR sensors to detect black lines. The LFPT Robot works will follow a black line, and will stop when the sensors on the left and right sides of the robot detect a black line. The results obtained a design of an LFPT Robot, which can be simulated directly by using Proteus 8 Software Simulator.

Keywords: *Line Follower, Penyiram Tanaman, sensor IR, Robot FLPT, Simulasi Software Proteus 8.*

ABSTRAK

Permasalahan mahalnnya upah tenaga kerja saat ini sudah sangat menjadi bahan pertimbangan untuk melakukan hal-hal pekerjaan yang rutin dikerjakan setiap hari, seperti halnya menyiram tanaman, untuk kasus ini, perlu adanya solusi atas permasalahannya, yaitu dengan membuat suatu robot untuk mengerjakan hal tersebut. Robot Line Follower Penyiram Tanaman atau yang selanjutnya akan disingkat menjadi Robot LFPT, tujuannya untuk melakukan pekerjaan rutin menyiram tanaman dengan cara robot tersebut berjalan sambil menyiram tanaman pada jalurnya. Karakteristik Robot LFPT yaitu robot akan berjalan dengan mengikuti garis hitam yang sudah ditentukan, dengan bantuan dua buah sensor IR pada bagian depan robot untuk berjalan, serta satu sensor IR pada sisi kiri dan kanan Robot LFPT untuk mendeteksi adanya tanaman. Jika sensor yang berada pada sisi kiri dan kanan robot mendeteksi adanya garis hitam, maka Robot LFPT akan berhenti dan akan mengarahkan selang air ke arah kiri dan kanan kemudian robot akan melakukan penyiraman. Penelitian ini menggunakan rangkaian sensor IR untuk mendeteksi garis hitam. Kerja Robot LFPT akan mengikuti jalur yang berwarna hitam, dan akan berhenti bila sensor pada sisi kiri dan kanan robot mendeteksi adanya garis hitam. Hasil penelitian didapatkan sebuah desain dari sebuah Robot LFPT tersebut, yang dapat langsung disimulasikan dengan menggunakan Simulator Software Proteus 8.

Kata Kunci: *Line Follower, Penyiram Tanaman, sensor IR, Robot FLPT, Simulator Software Proteus 8.*

1. Pendahuluan

1.1 Latar Belakang

Permasalahan mahalnnya upah tenaga kerja saat ini sudah sangat menjadi bahan pertimbangan untuk melakukan hal-hal pekerjaan yang rutin dikerjakan setiap hari, seperti halnya menyiram tanaman, untuk hal ini, perlu adanya solusi atas permasalahan ini, yaitu dengan membuat suatu robot untuk mengerjakan hal tersebut.

Arti robot itu sendiri berasal dari kata *robot*, yang berarti pekerja. Arti secara luas dari robot adalah sebuah alat mekanik yang dapat melakukan tugas fisik atau pekerjaan seperti halnya pekerja, baik menggunakan pengawasan dan kontrol manusia, atau menggunakan program yang telah didefinisikan atau diprogram terlebih dahulu pada IC Mikrokontroler. Sehingga pengganti pekerja rutin yang biasanya dilakukan oleh manusia dan diganti dengan sebuah robot, menurut penulis sudah

sangat tepat, atas dasar hal tersebut, penulis mengangkat topik pada penelitian ini berupa desain sebuah robot line follower penyiram tanaman, sebagai tahap awal. Line follower difungsikan, supaya robot itu dapat berjalan atau supaya tidak statis atau diam.

Robot *Line Follower* Penyiram Tanaman atau yang selanjutnya akan disingkat menjadi Robot LFPT akan berjalan dengan mengikuti garis hitam yang sudah ditentukan, dengan bantuan dua buah sensor IR pada bagian depan robot untuk berjalan, serta satu sensor IR pada sisi kiri dan kanan robot LFPT untuk mendeteksi adanya tanaman. Jika sensor yang berada pada sisi kiri dan kanan robot mendeteksi adanya garis hitam, maka robot LFPT akan berhenti dan akan mengarahkan selang air ke arah kiri dan kanan kemudian robot akan melakukan penyiraman.

1.2 Rumusan Masalah

Mengacu pada latar belakang tersebut, maka dapat di rumuskan beberapa permasalahan yang akan dibahas pada penelitian ini, yaitu: bagaimana perancangan dari desain skema serta mensimulasikan Robot LFPT tersebut dengan menggunakan simulasi *Software Proteus 8*?

1.3 Batasan Masalah

Untuk membatasi cakupan penelitian supaya tidak terlalu luas, maka dapat dibuat ruang lingkupnya sebagai berikut: menggunakan simulasi *Software Proteus 8* dengan bahasa pemrograman C.

2. Landasan Teori

Robot Line Follower Berbasis Mikrokontroler Atmega 16 dengan Menampilkan Status Gerak Pada LCD. Penelitian ini dilakukan oleh Achmad Zakki Falani dan Setyawan Budy pada tahun 2015. Penelitian ini bertujuan untuk membuat robot *line follower* yang dapat meringankan pekerjaan manusia. Kesimpulan yang didapat dari penelitian tersebut yaitu:

- a. *Software Proteus* sangat bermanfaat bagi para pemula yang ingin mempelajari tentang komponen elektronika.
- b. Mikrokontroler ATmega32A dapat digunakan untuk membuat rangkaian sistem minimum karena fitur dan jumlah *memory* yang dimiliki ATmega32A.

Sistem Pengaturan Kecepatan Motor DC Pada Alat Penyiram Tanaman Menggunakan Kontroler PID. Penelitian yang dilakukan oleh Ahmad Akhyar pada tahun 2014, bertujuan untuk membuat robot penyiram tanaman yang dapat berjalan. Kesimpulan dari penelitian tersebut yaitu: penelitian tersebut menggunakan sensor kelembaban yang dapat mengurangi jangkauan jarak karena terkendala kabel yang terhubung dari tanah ke robot.

Design and Implementation of IR based Line Follower Robot for Cooperative Task Sharing. Penelitian ini dilakukan oleh Pintu Dubey dan Alka Dubey pada tahun 2013. Tujuan dari penelitian ini adalah untuk mengembangkan robot yang dapat membagi tugasnya dengan bekerja sama menggunakan sensor IR, pada penelitian ini tugas tersebut ada balapan relay. Kesimpulan yang didapat adalah, robot yang didesain oleh Pintu dan Alka menunjukkan bahwa robot dapat bekerjasama untuk menyelesaikan pekerjaannya.

2.1 Definisi Robot

Kata robot berasal dari bahasa Cek yaitu *robota*, yang berarti pekerja. Menurut arti bahasa, robot adalah sebuah alat mekanik yang dapat melakukan tugas fisik, baik menggunakan pengawasan dan kontrol manusia, atau menggunakan program yang telah didefinisikan terlebih dahulu. Sedangkan menurut Kamus Besar Bahasa Indonesia, robot adalah alat berupa orang-orang dan sebagainya yang dapat bergerak atau berbuat seperti manusia, dan dikendalikan oleh mesin. (Winarno, et al, 2011, p2)

Berdasarkan Cangelosi dan Schelsinger (2015, p19), Secara historis, Etimologi kata robot berasal dari kata slavia yaitu *robota*, yang digunakan untuk merujuk kepada budak atau pekerja paksa. Kata ini pertama kali muncul dalam drama R.U.R (*Rossum Universal Robots*), yang ditulis oleh Karel Capek. Etimologi ini menunjukkan kalau robot di buat untuk membantu manusia dalam tugasnya sehari-hari, dan pada kasus tertentu untuk menggantikan manusia dengan melakukan tugas mereka pada industri robotika.

Kamus bahasa inggris Oxford menyatakan robot sebagai mesin yang mampu menjalankan aksi yang kompleks secara otomatis, terutama yang dapat di program menggunakan komputer. Definisi ini mengandung 4 konsep utama yang penting dalam mengembangkan robot, yaitu (Cangelosi, Schlesinger 2015, p19):

1. Mesin
2. Kompleks ... Aksi
3. Otomatis
4. Dapat di program dengan komputer

Berdasarkan Cook (2015, p2), Berdasarkan perspektif anatomi, bagian-bagian robot dapat di masukan kedalam beberapa kategori:

1. Otak
Robot bisa juga berfungsi tanpa menggunakan sebuah otak, contohnya adalah robot yang di kendalikan oleh manusia, sedangkan ada dua tipe untuk robot yang menggunakan otak, yaitu:

- a. Mikroprosesor

Mikroprosesor adalah sirkuit terintegrasi yang didalamnya hanya terdapat *CPU*. Mikroprosesor ini tidak memiliki *RAM*, *ROM* dan beberapa perangkat lainnya, sehingga seorang perancang sistem harus menambahkannya secara eksternal agar dapat bekerja dengan baik. Contohnya adalah chip prosesor buatan Intel dan AMD. Mikroprosesor ini melakukan pekerjaan yang tidak spesifik seperti bermain *game*, mengembangkan perangkat lunak, *browsing*, dan lain-lain.

- b. Mikrokontroler

Mikrokontroler adalah sirkuit terintegrasi yang memiliki *CPU* dan memiliki *RAM*, *ROM*, dan perangkat lainnya didalam sebuah chip. Mikrokontroler biasanya digunakan untuk melakukan pekerjaan yang spesifik, dimana pekerjaan yang akan dilakukan memiliki input dan output yang sudah ditentukan. Mikrokontroler bekerja pada kecepatan yang kecil yaitu 30 *MHz* sampai dengan 50 *MHz*.

2. Tenaga Listrik

Pasokan tenaga listrik ini terdiri dari:

- a. Sumber Tenaga

Robot secara umum paling banyak menggunakan sumber daya dari baterai. Tenaga matahari juga bisa menjadi pilihan untuk memasok daya ke robot. Pada robot yang canggih, tenaga matahari juga digunakan mengisi daya baterai pada kondisi cahaya yang optimal, dimana pada malam harinya robot dapat tetap bekerja dengan menggunakan baterai.

b. Regulasi Listrik

Bagian kecil dari robot yang di dedikasikan secara khusus untuk meningkatkan kestabilan kekuatan daya yang tersedia untuk perangkat elektronik. Alasan lainnya regulasi ini adalah karena beberapa perangkat robot membutuhkan lebih banyak tenaga dibandingkan perangkat yang lain.

c. Saklar On/Off

Saklar ini berfungsi untuk mematikan dan menghidupkan robot, sehingga robot dapat digunakan dan dimatikan jika tidak sedang diperlukan.

3. Sensor

Berikut adalah beberapa jenis sensor, yaitu:

- a. Tombol tekan,
- b. Sensor cahaya,
- c. Sensor *infrared*,
- d. Baterai *tester*,
- e. Saklar kemiringan,
- f. Sensor suhu.

4. Aksi dan Tanggapan

Robot melakukan aksi dari hasil yang didapatkan oleh sensor. Contohnya adalah:

- a. Gerakan,
- b. Suara,
- c. Tampilan,
- d. Indikasi Cahaya.

5. Tubuh

Semua bagian dari robot harus dihubungkan ke kerangka utama dari Robot. Kerangka utama inilah yang dapat melindungi bagian robot tersebut dari kerusakan.

a. Estetis

Aspek yang penting bagi tubuh adalah daya tarik. Tidak peduli seberapa bagus robot jika dilihat secara teknis, penampilan ini akan memberikan dampak yang besar terhadap bagaimana orang melihat robot tersebut.

2.2 Robot Line Follower

Sesuai dengan namanya, line follower (LF) adalah robot yang dapat bergerak tanpa bantuan manusia, tetapi menggunakan panduan berupa garis. Prinsip dasar robot line follower adalah membedakan warna gelap dan terang pada lantai/dasar lapangan. Warna gelap dapat diartikan warna yang tidak dapat memantulkan cahaya, contohnya hitam. Sedangkan warna terang diartikan sebagai warna yang dapat memantulkan cahaya, contohnya putih. Perbedaan warna pada dasar permukaan tersebut menjadi acuan sebuah robot line follower untuk bergerak mengikuti garis yang telah ditentukan. Robot jenis ini memiliki dua kategori, yaitu line follower analog dan digital. Line follower analog adalah jenis robot pengikut garis yang tidak memerlukan pemrograman robot yang melibatkan software sedangkan line follower digital melibatkan software. (Winarno, 2011, p14).

2.3 Sistem Dasar Robot

Berdasarkan Winarno (2011, p14), untuk membuat sebuah robot, kita harus memahami sistem dasar yang bekerja pada robot tersebut. Sistem dasar ini dibuat dengan perencanaan yang tepat, agar robot yang dirancang dapat berfungsi dengan baik.

a. Sistem Mekanik

Sistem mekanik merupakan salah satu bagian penting yang digunakan untuk menyusun sebuah robot. Sistem mekanik meliputi bentuk dan desain robot, material penyusun robot, serta sistem penunjang penggerak robot. Beberapa material yang dapat digunakan sebagai penyusun robot antara lain acrylic, kayu triplek, PVC, dan plastik mainan anak-anak. Material penyusun robot merupakan unsur penting dalam menentukan bobot robot yang mempengaruhi kelincahan robot tersebut.

b. Sistem Elektronik

Sistem elektronik merupakan faktor utama sebuah robot. Sistem elektronik digunakan untuk menggerakkan, mengendalikan, dan menstabilkan robot. Secara umum, yang tergolong dalam sistem elektronik robot adalah rangkaian sensor, rangkaian catu daya, rangkaian pengendali (driver), rangkaian kontrol, dan rangkaian penggerak (aktuator).

Rangkaian sensor dapat dikatakan sebagai 'mata' sebuah robot untuk mengenali lingkungan sekitarnya. Rangkaian catu daya adalah rangkaian pembangkit listrik sebuah robot. Rangkaian ini berfungsi untuk memberi, mengatur, dan membagi daya listrik pada robot.

Rangkaian oengendali (driver) merupakan rangkaian yang berguna untuk mengendalikan beberapa rangkaian yang terdapat pada robot. Rangkaian kontrol adalah sistem otak dari sebuah robot. Rangkaian ini berfungsi untuk mengontrol semua komponen yang ada pada sebuah robot.

Rangkaian penggerak biasa disebut dengan aktuator berfungsi untuk menggerakkan robot sesuai dengan perintah yang diberikan. Penggerak robot yang utama adalah motor. Jenis-jenis motor yang digunakan dapat disesuaikan dengan jenis robot yang ingin dibuat. Misalnya untuk lengan robot dapat menggunakan motor servo, dan untuk line follower dapat menggunakan motor DC.

c. Sistem Pemrograman

Sistem pemrograman digunakan pada pembuatan robot yang bersifat programmable, atau robot yang menggunakan IC mikrokontroler atau sistem kontrol lain sebagai otaknya. Robot yang menggunakan mikrokontroler memiliki rangkaian elektronik yang relatif lebih sederhana., karena sebagian elektronik telah digantikan oleh mikrokontroler itu sendiri. Gerak robot bergantung pada program yang disimpan dalam IC mikrokontroler. Sistem pemrograman yang biasa digunakan antara lain Bahasa C, Assembly, Basic, dan Pascal.

2.4 Software Proteus 8

Proteus adalah sebuah *software* untuk mendesain PCB yang juga dilengkapi dengan

simulasi pspice pada level skematik sebelum rangkaian skematik diupgrade ke PCB (Falany dan Budi, 2015).

Proteus merupakan *software* yang dibuat oleh Labcenter Electronics, yang dapat digunakan untuk melakukan simulasi VSM, dan PCB. Pada saat ini proteus menyediakan hampir 800 variasi mikrokontroler yang langsung dapat di gunakan untuk simulasi. Beberapa fitur proteus, yaitu:

1. Simulasi VSM
2. Membuat desain PCB
3. Visual Desainer

3. Metodologi Penelitian

Metodologi pada penelitian ini, terdiri dari tahapan mendesain diagram blok, perencanaan, pembuatan, dan uji coba.

3.1 Diagram Blok Pembuatan Robot

Tahap awal membuat sebuah robot adalah memdesain diagram blok, sehingga proses pengerjaan robot dapat diselesaikan dengan cepat, karena tahapan pembuatan robot sudah diketahui di awal dengan jelas.



Sumber: Winarno (2011)

Gambar 1. Diagram Blok Proses Pembuatan Robot LFLPT

3.2 Perencanaan Robot FLPT

Tahap perencanaan meliputi hal-hal berikut, yaitu: pemilihan komponen elektronik dan mekanik, termasuk di dalamnya menentukan komponen sensor, komparator, driver motor, dan motor yang digunakan dalam membuat robot. Terkait dengan mendesain skema Robot LFPT menggunakan *Software* Proteus 8, pemilihan komponen ini dapat dikerjakan secara langsung pada *software* tersebut.

3.3 Mendesain Robot FLPT

Mendesain adalah tahap setelah memilih komponen elektronik yang dibutuhkan untuk membuat robot. Setelah memilih komponen, selanjutnya mendesain komponen atau merangkai atau merakit komponen-komponen tersebut pada lembar kerja dari *Software* Proteus 8 tersebut, seperti halnya merangkai komponen pada papan PCB (*Printed Circuit Board*).

3.4 Uji Coba Robot FLPT

Pada penelitian ini, uji coba dapat langsung dikerjakan pada *Software* Proteus 8 tersebut, setelah sebelumnya mengerjakan kompilasi dan

tahapan lainnya, dengan cara menekan tombol icon Run. Jika rangkaian tersebut benar, maka hasil dari desain tersebut ada output atau keluaran, seperti roda berputar, lampu led menyala, *buzzer* bunyi, dan lain-lainnya.

4. Hasil dan Analisa

Untuk melihat hasil dan analisa dari penelitian ini, penulis menjelaskannya dalam sub bab Hasil dan Analisa, sebagai berikut ini:

4.1 Hasil

Untuk menghasil sebuah desain dari Robot FLPT, penulis menghasil dua hal berikut, yaitu: diagram blok Robot FLPT dan Desain Skema Robot FLPT, dapat dijelaskan secara detail sebagai berikut:

a. Diagram Blok Robot FLPT

Tahap awal dalam membuat sebuah robot diawali dengan pembuatan diagram blok dari Robot FLPT tersebut, secara global dapat digambarkan sebagai berikut:



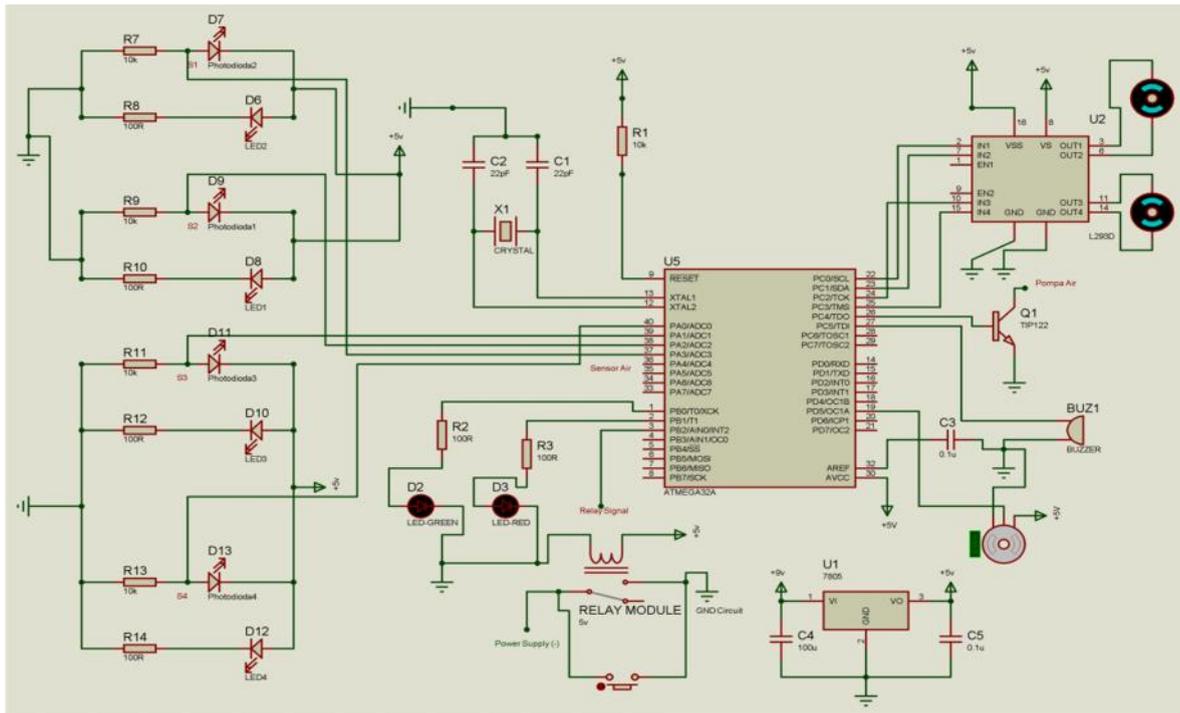
Sumber: Hasil Penelitian (2017)

Gambar 2. Diagram Blok Robot FLPT

Berdasarkan Gambar 2., maka hasil pembacaan dari sensor IR 1, IR 2, IR 3, IR 4, dan Sensor Air akan di proses pada mikrokontroler ATmega32A, lalu memberikan perintah ke Driver L293D untuk menggerakkan Motor DC, memberikan perintah ke *relay* jika mendeteksi adanya garis pada sisi kiri dan kanan, kemudian robot akan menggerakkan servo ke arah garis yang terdeteksi dan menyalakan pompa air. Ketika tangki air habis, maka mikrokontroler akan memberikan perintah untuk menyalakan *buzzer*.

b. Desain Skema Robot FLPT Dengan Simulator Software Proteus 8

Desain Robot FLPT dibuat dengan menggunakan *software* Proteus 8. Pada perancangan ini dilakukan simulasi pemasangan alat pada mikrokontroler ATmega32A. Berikut adalah gambar rangkaian Robot LFPT.



Sumber: Hasil Penelitian (2017)

Gambar 3. Desain Skema Robot LFPT

c. Coding C Robot FLPT

```
#ifndef F_CPU
#define F_CPU 16000000UL
#endif
#include <avr/io.h>
#include <util/delay.h>
void kanan();
void kiri();
void berhenti();
void servo_kekanan();
void servo_kekiri();
float sensorpertama[2];
float sensorkedua[2];
float sensorketiga[2];
float sensorkeempat[2];
float rule[16];
float jb1, jb2, jb3, jb4, jb5, jb6, jb7, jb8, jb9, jb10,
jb11, jb12, jberhenti, jkanan, jkiri, jlurus;
float jkan1, jkan2, jkir1, jkir2, jl1, jl2, js1, js2, mj1,
mj2, mj3, mj4, mj5, mj6, mj7, mj8, mj9, mj10, mj11,
mj12, mjttotal;
float luasjalan, lj1, lj2, lj3, lj4, lj5, lj6, lj7, lj8, lj9, lj10,
lj11, lj12, zjalan;
float akiri1, akiri2, akankir1, akankir2, akanan1,
akanan2, atidak1, atidak2, ma1, ma2, ma3, ma4,
ma5, ma6, ma7, ma8, ma9, ma10, ma11, ma12,
matotal;
float luasarah, la1, la2, la3, la4, la5, la6, la7, la8,
la9, la10, la11, la12, zarah;
float akk1, akk2, akn1, akn2, akr1, akr2, at1, at2,
at3, at4, at5, at6, at7, akk, akn, akr, at;
void fuzzifikasi(float s1, float s2, float s3, float s4)
{
    if(s1 <= 300)
    {
        sensorpertama[0] = 1;
    }
    else if(s1 >= 300 && s1 <= 700)
```

```
{
    sensorpertama[0] = (700 - s1) /
(700 - 300);
}
else if (s1 >= 700)
{
    sensorpertama[0] = 0;
}
if(s1 >= 700)
{
    sensorpertama[1] = 1;
}
else if(s1 >= 300 && s1 <= 700)
{
    sensorpertama[1] = (s1 - 300) /
(700 - 300);
}
else if (s1 <= 300)
{
    sensorpertama[1] = 0;
}
if(s2 <= 300)
{
    sensorkedua[0] = 1;
}
else if(s2 >= 300 && s2 <= 700)
{
    sensorkedua[0] = (700 - s2) / (700
- 300);
}
else if (s2 >= 700)
{
    sensorkedua[0] = 0;
}
if(s3 <= 300)
{
    sensorketiga[0] = 1;
}
else if(s3 >= 300 && s3 <= 700)
{
    sensorketiga[0] = (700 - s3) / (700
- 300);
}
else if (s3 >= 700)
{
    sensorketiga[0] = 0;
}
if(s3 >= 700)
{
    sensorketiga[1] = 1;
}
else if(s3 >= 300 && s3 <= 700)
{
    sensorketiga[1] = (s3 - 300) / (700
- 300);
}
else if (s3 <= 300)
{
    sensorketiga[1] = 0;
}
if(s4 <= 300)
{
    sensorkeempat[0] = 1;
}
else if(s4 >= 300 && s4 <= 700)
```

```

        {
            sensorkeempat[0] = (700 - s4) /
(700 - 300); }
        else if (s4 >= 700)
        {
            sensorkeempat[0] = 0; }
        else if (s4 <= 300)
        {
            sensorkeempat[1] = 0; } }
void implikasi(void)
{
    int i,j,k,l, a;
    a = 0;
    for(i=0;i<=1;i++)
    {
        for(j=0;j<=1;j++)
        {
            for(k=0;k<=1;k++)
            {
                rule[a] =
fmin(sensorpertama[i],sensorkedua[j]);
                rule[a] =
fmin(rule[a], sensorketiga[k]);
                rule[a] =
fmin(rule[a], sensorkeempat[l]);
                a = a + 1; } } } } }
void komposisi_aturan(void)
{
    jb1 = fmax(rule[0], rule[1]);
    jb2 = fmax(jb1, rule[2]);
    jb3 = fmax(jb2, rule[4]);
    jb4 = fmax(jb3, rule[5]);
    jb5 = fmax(jb4, rule[6]);
    jb6 = fmax(jb5, rule[8]);
    jb7 = fmax(jb6, rule[9]);
    jb8 = fmax(jb7, rule[10]);
    jb9 = fmax(jb8, rule[12]);
    jb10 = fmax(jb9, rule[13]);
    jlurus = rule[15];
    jkanan = rule[11];
    jkiri = rule[7];
    akk1 = fmax(rule[4], rule[8]);
    akk2 = fmax(akk1, rule[12]);
    akk = akk2;
    akn1 = fmax(rule[6], rule[10]);
    akn2 = fmax(akn1, rule[14]);
    akn = akn2;
    akr1 = fmax(rule[5], rule[9]);
    akr2 = fmax(akr1, rule[13]);
    akr = akr2;
    at1 = fmax(rule[0], rule[1]);
    at2 = fmax(at1, rule[2]);
    at3 = fmax(at2, rule[3]);
    at4 = fmax(at3, rule[7]);
    at5 = fmax(at4, rule[11]);
    at6 = fmax(at5, rule[15]);
    at = at6;
    jkan1 = (5 * jkanan);
    jkan2 = 10 - (5 * jkanan);
    jl1 = (5 * jlurus) + 10 ;
    jl2 = 20 - (5 * jlurus);
    jkir1 = (5 * jkiri) + 20 ;
    jkir2 = 30 - ( 5 * jkiri) ;
    akiri1 = (5 * akr);
    akiri2 = 10 - ( 5 * akr);
    akankir1 = (5 * akk) + 10 ;
    akankir2 = 20 - (5 * akk);
    akanan1 = (5 * akn) + 20;
    akanan2 = 30 - (5 * akn);
void defuzzyfikasi(void)
{
    if(jkanan >= jkiri && jkanan >= jlurus &&
jkanan >= jberhenti)
    {
        zjalan = (jkan1 + jkan2) / 2;}

```

```

        else if(jkiri >= jkanan && jkiri >= jlurus &&
jkiri >= jberhenti)
        {
            zjalan = (jkir1 + jkir2) / 2;}
        else if(jlurus >= jkanan && jlurus >= jkiri &&
jlurus >= jberhenti)
        {
            zjalan = (jl1 + jl2) / 2;}
        else if(jberhenti >= jkanan && jberhenti >=
jlurus && jberhenti >= jkiri)
        {
            zjalan = (js1 + js2) / 2;}
        else if(akr >= akn && akr >= akk && akr >=
at)
        {
            zarah = (akiri1 + akiri2) / 2;}
        else if(at >= akn && at >= akr && at >=
akk)
        {
            zarah = (atidak1 + atidak2) / 2;}}
int baca_adc(int port)
{
    ADMUX = (ADMUX & 0xF8) | port;
    ADCSRA |= (1<<ADSC);
    while(ADCSRA & (1<<ADSC));
    return (ADC);}
int main(void)
{
    DDRC = 0xFF;
    DDRD = 0xFF;
    float sensor1, sensor2, sensor3, sensor4;
    int sensorair;
    set_sleep_mode(SLEEP_MODE_PWR_D
OWN);
    sleep_disable();
    TCCR1B = 1<<WGM13 | 1<<WGM12 |
1<<CS11 | 1<<CS10;
    ICR1 = 4999;
    ADCSRA = 1<<ADEN | 1<<ADPS2 |
1<<ADPS1 | 1<<ADPS0; PORTB = (1<<0) |
(0<<1) | (1<<2);
    lurus();
    _delay_ms(250);
    while (1)
    {
        sensor1 = baca_adc(0);
        sensor2 = baca_adc(1);
        sensor3 = baca_adc(2);
        sensor4 = baca_adc(3);
        sensorair = baca_adc(4);
        fuzzifikasi(sensor1, sensor2,
sensor3, sensor4);
        implikasi();
        if(sensorair <= 100)
        {
            PORTC = (1<<5);
            PORTB = (1<<0) |
(1<<1) | (1<<2);
            _delay_ms(1000);
            PORTC = (1<<5);
            _delay_ms(1000);}
        else
        {
            PORTC = (0<<5);
            if(zjalan >= 0 && zjalan
<= 10)
            {
                kanan();
            }
            else if(zjalan > 10 &&
zjalan <=20)
            {
                lurus(); }
            else if(zjalan > 30 &&
zjalan <=40)
            {
                berhenti();}
            if(zarah >= 0 && zarah
<= 10)

```

```

        {
        _delay_ms(1000);
            servo_kekiri();
        _delay_ms(1000);
            pump();
        _delay_ms(250); }
        else if(zarah > 10 &&
        zarah <=20)
        {
        _delay_ms(1000);
            servo_kekiri();
        _delay_ms(1000);
            pump();
        _delay_ms(1000);
            pump();
        _delay_ms(1000);
        else if(zarah > 20 &&
        zarah <=30)
        {
        _delay_ms(1000);
            servo_kekanan();
        _delay_ms(250); }
        if(zjalan > 30 && zjalan
        <=40 && zarah >30 && zarah <= 40)
        {
            lurus();
        _delay_ms(250);
            sleep_enable();
        for(int
        i=1;i<=3;i++)
        {
            PORTB = (1<<0) | (1<<2);
            PORTC = (1<<5);
            PORTB = 0x00;
            sleep_cpu(); } }
    } }
void lurus()
{
    PORTC = (1<<0) | (1<<2); }
void kanan()
{
    PORTC = (1<<0) | (0<<2); }
void berhenti()
{
    PORTC = (0<<0) | (0<<2); }

```

4.2 Analisa

Untuk analisa dari penulis pada desain skema Robot FLPT dapat disampaikan hal-hal berikut, yaitu analisa dari komponen yang dibutuhkan untuk membuat Robot FLPT dan logika dari cara kerja Robot FLPT, yang dapat dijelaskan sebagai berikut:

a. Analisa Komponen

Desain Robot LFPT tersebut dapat berjalan dengan baik pada simulator *Software Proteus 8*, jika demikian berarti selanjutnya dapat dibuat sebuah Robot LFPT, dengan komponen-komponen utama sesuai dengan desain pada skema Robot FLPT.

Berikut adalah hasil analisa dari komponen-komponen tersebut adalah:

- 1) Sensor IR, terdiri dari sensor IR 1, sensor IR 2, sensor IR 3, dan sensor IR 4
- 2) Sensor Air
- 3) Atmega32A
- 4) Driver L293D
- 5) Relay
- 6) Servo
- 7) Buzzer
- 8) Motor DC
- 9) Pompa Air

Komponen-komponen detail lainnya yang berperan beserta fungsinya untuk membuat sebuah Robot LFPT dapat dijelaskan pada Tabel 1., sebagai berikut:

Tabel 1. Komponen Robot FLPT

No	Komponen	Jumlah	Fungsi
1	Mikrokontroler ATmega32A	1	Sebagai otak dari robot
2	Relay	1	Sebagai switch untuk mematikan power supply
3	Servo	1	Mengarahkan selang air
4	Pompa Air	1	Menyalurkan air
5	IR LED	4	Sensor untuk memberikan cahaya IR
6	Photodiode	4	Sensor untuk menerima cahaya IR
7	Kristal Osilator	1	Meningkatkan kecepatan untuk proses program
8	LED	2	Lampu untuk status robot
9	Motor Driver L293D	1	Untuk mengendalikan motor dc
10	Gearbox Motor DC + Roda	2	Menggerakkan robot
11	Push button	1	Tombol power robot
12	Resistor	11	Untuk IR LED, Photodiode, dan Pin Reset ATmega32A
13	Kapasitor	10	Untuk power supply, dan mengurangi noise
14	TIP 122	1	Switch untuk menyalakan pompa air
15	Box Baterai AA 8 pcs	1	Tempat baterai untuk sirkuit utama
16	Box Baterai AA 4pcs	1	Tempat baterai untuk pompa air

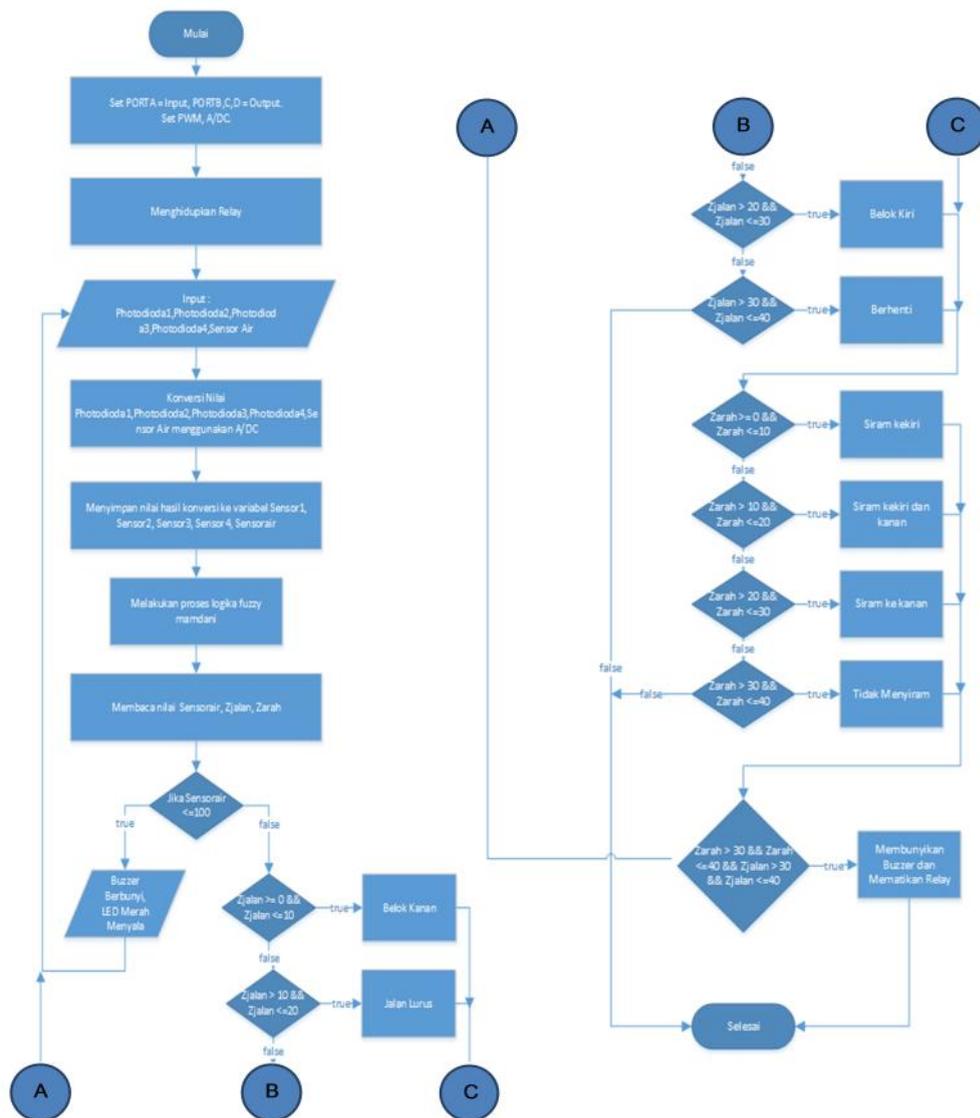
17	Baterai AA	10	Tenaga untuk robot
18	Papan PCB IC	1	Media untuk solder IR Led dan Photodiode
19	Breadboard	1	Pengganti papan PCB untuk semua sirkuit
20	Papan Acrylic 2mm ukuran A4	2	Badan robot
21	Kabel Jumper 20pcs	2	Kabel
22	7805	1	Menurunkan voltage ke 5v

23	Sensor Air	1	Mendeteksi air
24	Buzzer	1	Alarm

Sumber: Hasil Penelitian (2017)

b. Analisa Logika Robot FLPT

Logika dari cara kerja Robot FLPT dapat dijelaskan dengan menggunakan simbol *flowchart* sebagai berikut:



Sumber: Hasil Penelitian (2017)

Gambar 4. Flowchart Logika Robot FLPT

Penjelasan Gambar 4, yaitu: nilai dari *output* sensor Photodiode, dan sensor air akan digunakan sebagai nilai input bagi Robot LFPT. Kemudian akan dilakukan proses konversi nilai tersebut ke sinyal *digital*. Hasil konversi tersebut memiliki nilai

range 0 sampai 1023, dimana nilai 0 volt pada sensor akan menghasilkan nilai 0 pada hasil konversi, dan jika nilai sebesar 5 volt pada sensor maka hasil konversinya adalah 1023. Setelah mendapatkan hasil konversi dari setiap sensor,

maka akan dilakukan proses logika *fuzzy* menggunakan metode Mamdani. Hasil dari proses logika *fuzzy* ini adalah nilai yang akan digunakan untuk menentukan arah jalan dan arah siram robot LFPT. Tetapi jika nilai hasil konversi dari sensor air maka buzzer akan berbunyi dan robot akan berhenti.

5. Kesimpulan dan Saran

Penulis dapat menyimpulkan dan memberikan saran atas penelitian ini sebagai berikut:

5.1 Kesimpulan

Simulator *Software* Proteus 8 dapat dimanfaatkan sebelum pembuatan robot secara *hardware*, sehingga kebutuhan dan pengujian komponen dapat dilakukan sebelum pembuatan robot real, sehingga menjadi lebih efektif dan efisien dari segi waktu, tenaga, dan biaya. Hal ini, karena sebelum pemilihan dan pembelian komponen sudah di uji coba terlebih dahulu dengan menggunakan simulator tersebut. Kesalahan pemilihan dan pembelian komponen dapat dihindari atau teratasi sebelumnya.

5.2 Saran

Untuk pembuatan robot secara real, sebaiknya untuk melakukan desainnya dengan menggunakan alat bantu berupa simulator *Software* Proteus 8, untuk lebih efektif dan efisien dari segi waktu, tenaga, dan biaya serta lebih kreatif lagi dalam mendesain dan pemilihan komponen-komponennya. Jenis-jenis robot lainnya seperti Robot Pemindah Barang, Robot Pakan Ternak Otomatis, dan lain-lain dapat di simulasikan dengan menggunakan simulasi *Software* Proteus 8. Selamat berkreasi.

Hasil desain dari skema Robot LFPT tersebut, adalah hanya salah satu dari sekian desain skema robot yang dapat dibuat, desain lain sangat mungkin untuk dibuat dan dengan logika yang berbeda juga.

DAFTAR PUSTAKA

- Achmad Zakki Falani dan Setyawan Budi. (2015). Robot Line Follower Berbasis Mikrokontroler Atmega 16 dengan Menampilkan Status Gerak Pada LCD. *Narodroid*: Vol.1. No.1.
- Akhyar, A. (2014). Sistem Pengaturan Kecepatan Motor DC Pada Alat Penyiraman Tanaman Menggunakan Kontroler PID, Teknik Elektro Universitas Brawijaya, Vol.2. No.4.
- Cangelosi, A. dan Schlesinger Matthew. (2015). *Developmental Robotics: From Babies to Robots*. The MIT Press, London.
- Cook, David. (2015). *Robot Building for Beginners: Third Edition*. Apress. California.
- Pintu Dubey dan Alka Dubey. (2013). Design and Implementation of IR based Line Follower for Cooperative Task Sharing. *E-Journal of Science & Technology*. Vol.8. No.3.
- Winarno dan Arifianto Deni. (2011). *BIKIN ROBOT ITU GAMPANG*. Jakarta: Penerbit PT Kawan Pustaka.