

IMPLEMENTASI ROBOT *LINE FOLLOWER* PENYIRAM TANAMAN OTOMATIS MENGUNAKAN METODE *PROPORTIONAL-INTEGRAL-DERIVATIVE* CONTROLLER (PID)

Muhammad Yusuf^{*1}, Isnawaty², Rahmat Ramadhan³

^{*1,2,3}Jurusan Teknik Informatika, Fakultas Teknik, Universitas Halu Oleo, Kendari

e-mail: ^{*1}thotathoten@gmail.com, ²isna.1711@gmail.com, ³rahmat.ramadhan@innov-center.org

Abstrak

Penggunaan robot dapat diaplikasikan dalam penyiraman tanaman sayuran, buah dan bunga yang menggunakan sistem rumah kaca. Sistem rumah kaca tidak dipengaruhi oleh curah hujan sehingga robot dapat difungsikan untuk melakukan penyiraman tanaman yang pengaturannya menggunakan waktu sebagai perintah untuk melakukan penyiraman.

Metode yang digunakan untuk mengendalikan robot yaitu PID (dari singkatan bahasa *ProPortional - Integral - Derivative Controller*) merupakan *controller* untuk menentukan presisi suatu sistem instrumentasi dengan karakteristik adanya umpan balik pada sistem tersebut. Komponen kontrol PID ini terdiri dari tiga jenis yaitu *ProPortional*, *Integratif* dan *Derivatif*. Ketiganya dapat dipakai bersamaan maupun sendiri-sendiri tergantung dari respon yang kita inginkan terhadap suatu *plant*.

Prinsip kerja robot penyiram tanaman yaitu dilakukan dengan cara robot mendeteksi garis sebagai jalur pergerakan otomatisasi robot. Pencarian garis dilakukan dengan mendeteksi pancaran cahaya yang dipancarkan oleh LED dan dibaca oleh sensor *Photodiode*. Untuk mengikuti garis robot digerakkan oleh motor DC yang dikontrol menggunakan metode PID, robot bergerak secara otomatis menggunakan aplikasi dari motor DC. Robot menggunakan pembacaan sensor *Photodiode* untuk melakukan penyiraman, setelah robot mendeteksi perempatan pada jalur maka robot akan berhenti dan mengaktifkan pompa air untuk melakukan penyiraman dan apabila robot sudah melakukan proses penyiraman yang sesuai dengan apa yang diperintahkan oleh program maka robot akan menuju ke tempat pemberhentian yaitu dengan mendeteksi jalur pertigaan

Kata kunci— Robot, *Line Follower*, PID, Otomatisasi Penyiram Tanaman

Abstract

The use of robots can be applied in plant watering vegetables, fruits and flowers using the greenhouse system. Greenhouse system is not affected by rainfall so that the robot can be used for watering plants that perform settings using time as a command to do the watering.

The method used to control a robot that is PID (ProPortional language of abbreviations - Integral - Derivative controller) is a controller to determine the precision of the instrumentation system with the characteristics of their feedback on the proficiency level system. PID control component consists of three types: ProPortional, Integrative and Derivative. All three can be used together or individually depending on the response we wanted to a plant.

The working principle of robot watering is done by detecting robot lines as the future path of robotic automation. Line search performed by detecting the light emitted by the LED and read by the sensor Photodiode. To follow the line of the robot are driven by a DC motor which is controlled using PID method, the robot moves automatically using the application of a DC motor. Robot uses sensor readings Photodiode to do watering, after the robot detects an intersection on the path the robot will

stop and turn on the water pump to do the watering and if the robot was doing the watering according to what was ordered by the program the robot will toward the place dismissal is to detect fork in the path.

Keywords— Robot, Line Follower, PID, Automated Watering Plants

1. PENDAHULUAN

Kemajuan teknologi semakin berkembang sangat pesat pada kehidupan manusia di era modern ini, khususnya pada bidang elektronika. Hal ini ditandai dengan adanya berbagai peralatan yang diciptakan dan dapat dioperasikan serta digunakan secara otomatis. Disebabkan oleh kemajuan teknologi inilah maka berkembang suatu ilmu yang merupakan suatu pecahan dari ilmu elektronika yaitu bidang robotika.

Robot dapat diartikan suatu peralatan yang dioperasikan dengan atau tanpa bantuan manusia atau dengan kata lain bahwa robot merupakan suatu perangkat otomatis. Saat ini robot banyak digunakan dalam berbagai bidang kehidupan. Didalam bidang industri robot menjadi pekerja sehingga hasil dari industri lebih berlimpah dan waktu produksi yang dibutuhkan juga lebih cepat dibanding tenaga manusia.

Hal yang sama juga dapat diterapkan pada robot penyiram otomatis yang dapat membantu para petani dalam merawat tanaman yang dikelola. Dalam hal ini robot penyiram tanaman otomatis ini dapat diterapkan dalam pembudidayaan tanaman sayuran, buah dan bunga yang menggunakan sistem rumah kaca (*Greenhouse*). Sistem rumah kaca tidak dipengaruhi oleh curah hujan sehingga robot dapat difungsikan untuk melakukan penyiraman tanaman yang pengaturannya menggunakan waktu sebagai perintah untuk melakukan penyiraman.

Pada robot penyiram tanaman otomatis ini dapat digabungkan dengan metode *ProPortional-Integral-Derivative Controller* (PID) yang dapat mempercepat kinerja dari robot dalam menjalankan tugasnya. *ProPortional-Integral-Derivative Controller* (PID) merupakan *controller* untuk menentukan presisi suatu sistem instrumentasi dengan karakteristik adanya umpan balik pada sistem tersebut. Komponen kontrol PID ini terdiri dari tiga jenis yaitu *ProPortional*, *Integratif* dan *Derivatif*. Ketiganya dapat dipakai bersamaan maupun sendiri-sendiri tergantung dari respon

yang diinginkan terhadap suatu *plant*, sehingga robot dapat menganalisa secara otomatis terhadap *action* yang akan dilakukannya.

Berdasarkan uraian latar belakang diatas maka dari itu penulis mengangkat judul penelitian “IMPLEMENTASI ROBOT *LINE FOLLOWER* PENYIRAM TANAMAN OTOMATIS MENGGUNAKAN METODE *PROPORTIONAL-INTEGRAL-DERIVATIVE CONTROLLER* (PID)”.

2. METODE PENELITIAN

2.1 Robot

Robot adalah sebuah alat mekanik yang dapat melakukan tugas fisik. Ada yang menggunakan pengawasan dan control manusia, ataupun menggunakan program yang telah didefinisikan terlebih dulu (kecerdasan buatan). Robot klasik sudah ada sejak zaman Yunani kuno. Hingga kini robot terus dikembangkan sehingga keberadaannya sangat membantu manusia dalam mengerjakan pekerjaan rutin dan berat, atau bahkan sebagai penghibur. Elektronika merupakan bidang yang menarik untuk dipelajari oleh pelajar dan penghobi. Hal ini karena kita dapat berkreasi apa saja sesuai keinginan kita. Secara umum robot dapat didefinisikan sebagai sebuah piranti mekanik yang mampu melakukan pekerjaan manusia atau berperilaku seperti manusia [1].

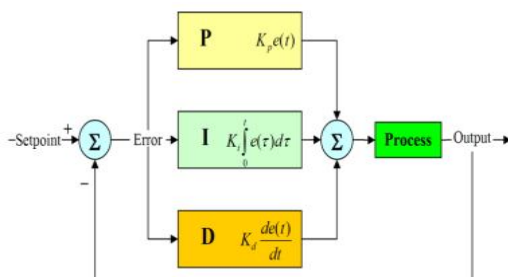
Kata robot sendiri diperkenalkan ke publik oleh Karel Capek pada saat memainkan *RUR (Rossum's Universal Robots)*, namun awal munculnya robot dapat diketahui dari bangsa Yunani kuno yang membuat patung yang dapat dipindah-pindahkan [2]. Robot adalah sebuah alat mekanik yang dapat melakukan tugas-tugas fisik, baik menggunakan pengawasan dan kontrol manusia, ataupun menggunakan program yang telah dimasukkan terlebih dulu [3].

2.2 Propotional-Integral-Derivative Controller

PID (dari singkatan bahasa *ProPortional-Integral-Derivative Controller*) merupakan *controller* untuk menentukan

presisi suatu sistem instrumentasi dengan karakteristik adanya umpan balik pada sistem tersebut. Komponen kontrol PID ini terdiri dari tiga jenis yaitu *ProPortional*, *Integratif* dan *Derivatif*. Ketiganya dapat dipakai bersamaan maupun sendiri-sendiri tergantung dari respon yang kita inginkan terhadap suatu *plant*.

Kontrol PID saat ini banyak digunakan dalam aksi-aksi di dunia industri dan juga kontrol robot. jika kita berbicara kontrol robot *line follower* dengan PID maka bukanlah kontrol PID yang sebenarnya sebab pada robot *line follower* elemen ukur (sensor) tidak terdapat pada *plant* (motor penggerak) dari robot, yang seharusnya adalah sensor terdapat di *plant* (motor penggerak), dengan contoh *tachometer* sebagai sensor pada motor, *encoder* atau yang lainnya yang terletak pada *plant*. sedangkan pada robot *line follower* sensor berupa pendeteksi garis (tidak terletak pada *plant*) dan dari hasil kondisi garis tersebut barulah dikontrol ke motor (*plant*), walaupun begitu kontrol PID masih dapat diterapkan untuk mengendalikan robot *line follower* [4]. Blok aksi *control* PID ditunjukkan oleh Gambar 1.



Gambar 1 Blok control PID

Gambar 1 menjelaskan sebagai berikut :

1. $SP = Set\ point$, *set point* adalah suatu parameter nilai acuan atau nilai yang kita inginkan.
2. $PV = Present\ Value$, *present value* adalah nilai bobot pembacaan sensor saat itu atau variabel terukur yang di umpan balikan oleh sensor (sinyal *feedback* dari sensor).
3. $Error =$ nilai kesalahan, *error* adalah Deviasi atau simpangan antar *variabel* terukur atau bobot sensor (PV) dengan

nilai acuan (SP). Persamaan (1) menunjukkan penentuan nilai *error*:

$$error = SP - PV \quad (1)$$

2.3 Microcontroller ATmega32

Microcontroller AVR (*Alf and Vegard's Risc processor*) standar memiliki arsitektur 8 bit, dimana semua instruksi dikemas dalam kode 32-bit, dan sebagian besar intruksi dieksekusi dalam 1(satu) siklus *clock*. AVR berteknologi RISC (*Reduced Instruction Set Computing*), sedangkan MCS51 berteknologi CISC (*Complex Intruction Set Computing*).

AVR dapat dikelompokkan menjadi empat kelas, yaitu keluarga *Attiny*, keluarga *AT902xx*, keluarga *Atmega*, dan keluarga *AT86RFxx*. Pada dasarnya yang membedakan masing-masing kelas adalah memori, peripheral, dan fungsinya (silahkan mengunjungi www.atmel.com untuk informasi lebih lanjut tentang berbagai variasi AVR). Untuk *microcontroller* AVR yang berukuran lebih kecil, silahkan mencoba *Atmega8*, *Attiny2313* dengan ukuran *Flash Memory* 2KB dengan dua *input* analog [5].

Microcontroller pada dasarnya diprogram dengan bahasa *assembler*, tetapi saat ini *microcontroller* dapat diprogram dengan menggunakan bahasa tingkat tinggi seperti *BASIC*, *PASCAL* atau *C*. Bahasa tingkat tinggi tersebut memiliki beberapa keuntungan dibandingkan dengan bahasa *assembler* :

1. Lebih mudah membangun program dengan menggunakan bahasa tingkat tinggi
2. Perbaikan program lebih mudah jika program dibangun menggunakan bahasa tingkat tinggi
3. *Testing* program didalam bahasa tingkat tinggi lebih mudah
4. Bahasa tingkat tinggi lebih banyak dikenal dan *error* program yang dibuat dapat dihindari
5. Mudah mendokumentasikan sebuah program tingkat tinggi

Didalam *microcontroller* *Atmega32* terdiri dari:

1. Saluran I/O ada 32 buah, yaitu *Port A*, *Port B*, *Port C*, dan *Port D*.
2. ADC (*Analog to Digital Converter*) 10 bit sebanyak 8 *channel*.
3. Tiga buah *Timer/Counter* dengan kemampuan perbandingan.
4. CPU yang terdiri dari 32 register.

5. Untuk 131 intruksi andal yang umumnya hanya membutuhkan 1 siklus *clock*.
6. *Watchdog Timer* dengan *oscilator* internal.
7. Dua buah *Timer/Counter* 8 bit.
8. Satu buah *Timer/Counter* 16 bit.
9. Tagangan operasi 2,7 V – 5,5 V pada Atmega32.
10. Internal *SRAM* sebesar 2KB.
11. *Memory Flash* sebesar 32KB dengan kemampuan *Read While Write*.
12. Unit interupsi internal dan eksternal.
13. *Port* antarmuka *SPI*.
14. *EEPROM* sebesar 1 Kb dapat diprogram saat operasi.
15. Antar muka komparator analog.
16. Empat channel *PWM*.
17. Membutuhkan 32x8 general *purpose register*.
18. Hampir mencapai 16 MIPS pada Kristal 16 MHz.
19. *Port USART* programmable untuk komunikasi serial.

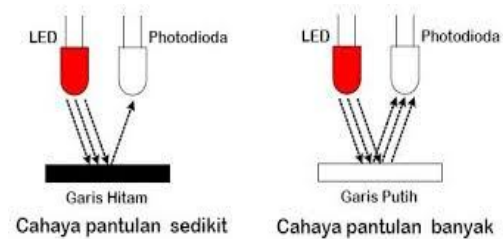
2.4 Sensor *Photodiode*

Photodiode adalah suatu jenis dioda yang resistansinya berubah-ubah. Dalam gelap nilai tahanannya sangat besar hingga praktis tidak ada arus yang mengalir. Semakin kuat cahaya yang jatuh pada dioda maka makin kecil nilai tahanannya, sehingga arus yang mengalir semakin besar. Jika *photodiode* persambungan p-n bertegangan balik disinari, maka arus akan berubah secara linier dengan kenaikan fluks cahaya yang dikenakan pada persambungan tersebut.

Saat *photodiode* terkena cahaya, maka akan bersifat sebagai sumber tegangan dan nilai resistansinya akan menjadi kecil. Saat *photodiode* tidak terkena cahaya, maka nilai resistansinya akan besar atau dapat diasumsikan tak hingga. Besarnya tegangan atau arus listrik yang dihasilkan oleh *photodiode* tergantung besar kecilnya radiasi yang dipancarkan oleh infrared. *Photodiode* digunakan sebagai penangkap gelombang cahaya yang dipancarkan oleh infrared. Besarnya tegangan atau arus listrik yang dihasilkan oleh *Photodiode* tergantung besar kecilnya radiasi yang dipancarkan oleh infrared [6].

LED *superbright* berfungsi sebagai pengirim cahaya ke garis untuk dipantulkan lalu dibaca oleh sensor *photodiode*. Sifat dari

warna putih (permukaan terang) yang memantulkan cahaya dan warna hitam (permukaan gelap) yang tidak memantulkan cahaya digunakan dalam aplikasi ini. Gambar 2 adalah ilustrasi mekanisme sensor garis.



Gambar 2 Blok sensor *photodiode*

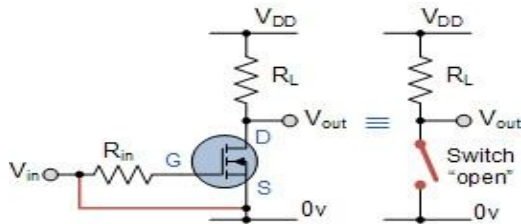
Sensor garis adalah suatu perangkat/alat yang digunakan untuk mendeteksi adanya sebuah garis atau tidak. Garis yang dimaksud adalah garis berwarna hitam diatas permukaan berwarna putih. Alat ini menggunakan teknik pantulan cahaya inframerah yang ditangkap oleh *photodiode* dari sebuah LED seperti layaknya manusia, bagaimana manusia dapat berjalan pada mengikuti jalan yang ada tanpa menabrak dan sebagainya, tentunya karena manusia memiliki “mata” sebagai penginderanya.

2.5 MOSFET

MOSFET (*Metal Oxide Semiconductor Field Effect Transistor*) merupakan salah satu jenis transistor yang memiliki impedansi masukan (*gate*) sangat tinggi (hampir tak berhingga) sehingga dengan menggunakan MOSFET sebagai saklar elektronik, memungkinkan untuk menghubungkannya dengan semua jenis gerbang logika. Dengan menjadikan MOSFET sebagai saklar, maka dapat digunakan untuk mengendalikan beban dengan arus yang tinggi dan biaya yang lebih murah daripada menggunakan transistor bipolar. Untuk membuat MOSFET sebagai saklar maka hanya menggunakan MOSFET pada kondisi saturasi (*ON*) dan kondisi *Cut-Off* (*OFF*).

Wilayah *Cut-Off* (MOSFET *OFF*) Pada daerah *Cut-Off* MOSFET tidak mendapatkan tegangan *input* ($V_{in} = 0V$) sehingga tidak ada arus *drain Id* yang mengalir. Kondisi ini akan membuat tegangan $V_{ds} = V_{dd}$. Dengan beberapa kondisi diatas maka pada daerah *Cut-Off* ini MOSFET dikatakan *OFF* (*Full-Off*). Kondisi *Cut-Off* ini dapat diperoleh dengan

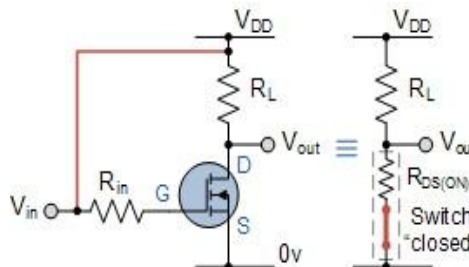
menghubungkan jalur *input (gate)* ke *ground*, sehingga tidak ada tegangan *input* yang masuk ke rangkaian saklar MOSFET. Gambar 3 menunjukkan kondisi MOSFET *OFF*.



Gambar 3 Kondisi MOSFET *OFF*

Input gate tidak mendapat tegangan bias karena terhubung ke *ground (0V)* Tegangan *gate* lebih rendah dari tegangan *threshold* ($V_{gs} < V_{th}$) MOSFET *OFF (Fully-Off)* pada daerah *Cut-Off* ini. Tidak arus *drain* yang mengalir pada MOSFET Tegangan *output* $V_{out} = V_{ds} = V_{dd}$ Pada daerah *Cut-Off* MOSFET dalam kondisi *open circuit*. Dengan beberapa karakteristik diatas maka dapat dikatakan bahawa MOSFET pada daerah *Cut-Off* merupakan saklar terbuka dengan arus *drain* $I_d = 0$ Ampere. Untuk mendapatkan kondisi MOSFET dalam keadaan open maka tegangan *gate* V_{gs} harus lebih rendah dari tegangan *threshold* V_{th} dengan cara menghubungkan terminal *input (gate)* ke *ground*.

Wilayah Saturasi (MOSFET *ON*) Pada daerah saturasi MOSFET mendapatkan bias *input* (V_{gs}) secara maksimum sehingga arus *drain* pada MOSFET juga akan maksimum dan membuat tegangan $V_{ds} = 0V$. Pada kondisi saturasi ini MOSFET dapat dikatakan dalam kondisi *ON* secara penuh (*Fully-ON*). Gambar 4 menunjukkan kondisi MOSFET *ON*.



Gambar 4 Kondisi MOSFET *ON*

Tegangan *input gate* (V_{gs}) tinggi Tegangan *input gate* (V_{gs}) lebih tinggi dari tegangan *threshold* ($V_{gs} > V_{th}$) MOSFET *ON (Fully-ON)* pada daerah Saturasi Tegangan *drain* dan *source* ideal (V_{ds}) pada daerah

saturasi adalah $0V$ ($V_{ds} = 0V$) Resistansi *drain* dan *source* sangat rendah ($R_{ds} < 0,1 \text{ Ohm}$) Tegangan *output* $V_{out} = V_{ds} = 0,2V$ ($R_{ds} \cdot I_d$) MOSFET dianalogikan sebagai saklar kondisi tertutup Kondisi saturasi MOSFET dapat diperoleh dengan memberikan tegangan *input gate* yang lebih tinggi dari tegangan *threshold* dengan cara menghubungkan terminal *input* ke V_{dd} . Sehingga MOSFET menjadi saturasi dan dapat dianalogikan sebagai saklar pada kondisi tertutup[7].

2.6 LCD (Liquid Crystal Display)

Liquid Crystal Display (LCD) merupakan perangkat (devais) yang sering digunakan untuk menampilkan data selain menggunakan *seven segment*. LCD berfungsi sebagai salah satu alat komunikasi dengan manusia dalam bentuk tulisan/gambar. Untuk menghubungkan *microcontroller* dengan LCD dibutuhkan konfigurasi antara pin-pin yang ada di LCD dengan *Port* yang ada di *microcontroller* [5]. Gambar 5 menunjukkan LCD 16 X 2.



Gambar 5 LCD 16 X 2

2.7 Push-Button

Push yang berarti tekan dan button yang artinya tombol adalah salah satu komponen yang dikendalikan secara manual oleh manusia. *Push button* biasanya digunakan dalam panel listrik yang berfungsi untuk menghubungkan dan memutuskan aliran listrik ke beban [8]. Gambar 6 menunjukkan *push-button*.



Gambar 6 *Push-Button*

Push button terbagi menjadi 3 jenis.

1. *Push Button NC (Normally Open)*
2. *Push Button NO (Normally Close)*
3. *Push Button NO/NC*

Push button NC dan *Push Button NO* saling berpasangan dan harus selalu sepasang.

Sebab akan kesulitan untuk mengoperasikan beban jika salah satunya tidak ada.

2.8 Relay

Relay adalah Saklar (*Switch*) yang dioperasikan secara listrik dan merupakan komponen *Electromechanical* (Elektromekanikal) yang terdiri dari 2 bagian utama yakni Elektromagnet (*Coil*) dan Mekanikal (seperangkat Kontak Saklar/*Switch*). *Relay* menggunakan Prinsip Elektromagnetik untuk menggerakkan Kontak Saklar sehingga dengan arus listrik yang kecil (*low power*) dapat menghantarkan listrik yang bertegangan lebih tinggi. Sebagai contoh, dengan *Relay* yang menggunakan Elektromagnet 5V dan 50 mA mampu menggerakkan Armature *Relay* (yang berfungsi sebagai saklarnya) untuk menghantarkan listrik 220V 2A [9]. Gambar 7 menunjukkan *relay*.



Gambar 7 *Relay*

Prinsip Kerja *Relay* pada dasarnya, *Relay* terdiri dari 4 komponen dasar yaitu :

1. Electromagnet (*Coil*)
2. Armature
3. *Switch Contact Point* (Saklar)
4. *Spring*

Poin Kontak (*Contact Point*) *Relay* terdiri dari 2 jenis yaitu :

1. *Normally Close* (NC) yaitu kondisi awal sebelum diaktifkan akan selalu berada di posisi *CLOSE* (tertutup)
2. *Normally Open* (NO) yaitu kondisi awal sebelum diaktifkan akan selalu berada di posisi *OPEN* (terbuka)

2.9 Motor DC

Motor DC adalah motor yang non-terpolarisasi -yang berarti bahwa tegangan dapat di balikkan maka putarannya pun akan berubah. Tegangan yang digunakan untuk memutar motor DC sekitar 6V-24V atau lebih. Motor DC yang digunakan pada robot sekitar motor DC 6V-12V, tegangan secara langsung berkaitan dengan *torsi* dari sebuah

motor. Lebih besar tegangan, maka lebih besar *torsi* yang dihasilkan. Tetapi dalam pemberian tegangan tidak boleh melebihi dari tegangan yang dibutuhkan. Misalkan pemberian tegangan hingga 100V, itu menyebabkan motor tidak akan lagi berputar(rusak).

Hal itu menyebabkan motor menjadi terlalu panas dan kumparan akan meleleh. Meskipun motor 24V mungkin lebih kuat, apakah benar-robot harus membawa baterai 24V (yang lebih berat dan lebih besar,kecuali jika benar-benar membutuhkan sebuah *torsi* pada motor. Ada beberapa macam *driver* motor DC yang biasa kita pakai seperti menggunakan *relay* yang diaktifkan dengan transistor sebagai saklar, namun yang demikian dianggap tidak efisien dalam pengerjaan *hardware*-nya. Dengan berkembangnya dunia IC, sekarang sudah ada H *Bridge* yang dikemas dalam satu IC dimana memudahkan kita dalam pelaksanaan *hardware* dan kendalinya menggunakan *microcontroller*, berikut IC yang familiar dalam dunia robotika seperti IC L298 dan L293, kedua IC ini memiliki kelebihan dan kekurangan masing-masing. Modul yang menggunakan IC *driver* L298 yang memiliki kemampuan menggerakkan motor DC sampai arus 4A dan tegangan maksimum 46 VoltDC untuk satu kanalnya [5]. Gambar 8 menunjukkan Motor DC.



Gambar 8 Motor DC

2.10 Pompa Air

Pompa Air adalah suatu alat atau mesin yang digunakan untuk memindahkan cairan dari suatu tempat ke tempat yang lain melalui suatu media perpipaan dengan cara menambahkan energi pada cairan yang dipindahkan dan berlangsung secara terus menerus.

Pompa beroperasi dengan prinsip membuat perbedaan tekanan antara bagian masuk (*suction*) dengan bagian keluar (*discharge*). Dengan kata lain, pompa berfungsi mengubah tenaga mekanis dari suatu

sumber tenaga (penggerak) menjadi tenaga kinetis (kecepatan), dimana tenaga ini berguna untuk mengalirkan cairan dan mengatasi hambatan yang ada sepanjang pengaliran. Pompa Air ditunjukkan oleh Gambar 9.



Gambar 9 Pompa Air

2.10 Flowchart

Flowchart adalah penggambaran secara grafik dari langkah-langkah dan urutan prosedur dari suatu program. Flowchart menolong analis dan programmer untuk memecahkan masalah kedalam segmen-segmen yang lebih kecil dan menolong dalam menganalisis alternatif-alternatif lain dalam pengoperasian.

Flowchart biasanya mempermudah penyelesaian suatu masalah khususnya masalah yang perlu dipelajari dan dievaluasi lebih lanjut. Tujuan Membuat flowchart:

- a. Menggambarkan suatu tahapan penyelesaian masalah
- b. Secara sederhana, terurai, rapi dan jelas
- c. Menggunakan simbol-simbol standar

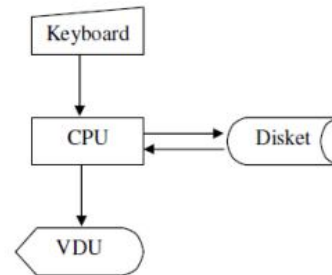
Dalam penulisan flowchart dikenal dua model, yaitu Sistem flowchart dan Program flowchart.

1. Sistem Flowchart

Sistem flowchart yaitu bagan yang memperlihatkan urutan prosedur dan proses dari beberapa file di dalam media tertentu. Melalui flowchart ini terlihat jenis media penyimpanan yang dipakai dalam pengolahan data.

- a. Selain itu juga menggambarkan file yang dipakai sebagai input dan output.
- b. Tidak digunakan untuk menggambarkan urutan langkah untuk memecahkan masalah
- c. Hanya untuk menggambarkan prosedur dalam sistem yang dibentuk

Gambar 10 menunjukkan Sistem Flowchart.



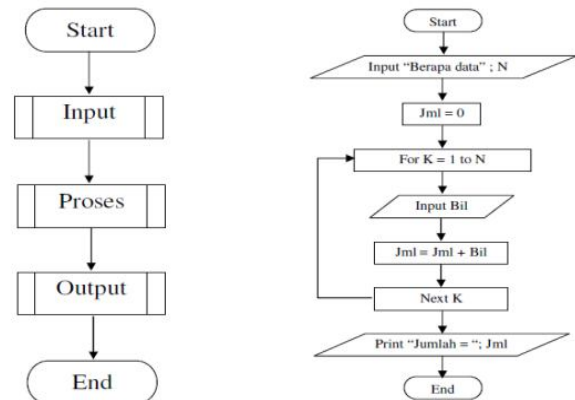
Gambar 10 System Flowchart

2. Program Flowchart

Program flowchart yaitu bagan yang memperlihatkan urutan dan hubungan proses dalam suatu program. Dua jenis metode penggambaran program flowchart.

- 1) Conceptual flowchart, menggambarkan alur pemecahan masalah secara global
- 2) Detail flowchart, menggambarkan alur pemecahan masalah secara rinci.

Gambar 11 menunjukkan Conceptual Flowchart dan Detail Flowchart



Gambar 11 Conceptual Flowchart dan Detail Flowchart

2.11 Metode Pengumpulan Data

Guna mendapatkan hasil akhir yang maksimal dan sesuai dengan harapan, maka perlu adanya metode-metode yang membantu dan mendukung tercapainya tujuan penelitian. Metode tersebut adalah sebagai berikut :

1. Pengamatan

Pada penelitian ini, pengamatan difokuskan pada komponen elektronika berupa ATmega32, Sensor Photodiode, dan skema rangkaian yang akan digunakan dikarenakan penggunaan komponen tersebut terbilang sangat baru dalam dunia informatika.

Pengamatan dilakukan untuk mengetahui struktur dalam komponen yang akan digunakan, prinsip kerja serta perangkaiannya.

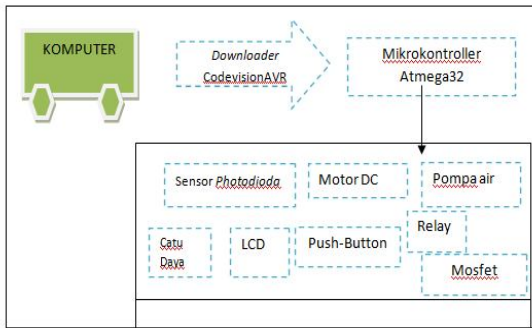
2. Dokumentasi

Dalam penelitian ini, dokumentasi dilakukan dengan membaca literatur – literatur yang berhubungan dengan pemrograman Arduino, pemanfaatan dan prinsip kerja ATmega32, Sensor *Photodiode*, pemahaman metode PID, dan literatur – literatur lain yang dinilai berhubungan dengan penelitian ini.

2.12 Perancangan Sistem

1. Gambaran Umum Sistem

Secara umum sistem dibuat agar dapat membaca perintah dari *software* CodevisionAVR berupa bahasa pemrograman ke *microcontroller* Atmega32, kemudian dari Atmega32 melakukan kontrol ke sensor-sensor yang ada di robot. Gambar 13 adalah gambaran umum sistem kendali robot *Line follower* Penyiram Tanaman. Gambar 12 menunjukkan Gambaran Umum Sistem.



Gambar 12 Gambaran Umum Sistem

Komputer berfungsi untuk membuat program, sebelumnya komputer telah di instal dengan *software* CodevisionAVR untuk membuat program, setelah itu program/coding di kirim menggunakan *Usb Downloader* ke dalam mikrokontroller Atmega32 sebagai inti/otak dari robot.

Microcontroller sebagai pengendali semua sensor yang digunakan. Sensor *Photodiode* berfungsi untuk mendeteksi garis telah disediakan sebagai acuan robot untuk menuju ke tanaman lalu melakukan proses penyiraman, motor DC melalui *MOSFET* berguna untuk menggerakkan roda robot apakah harus maju, ke kiri ataupun ke kanan dan berhenti. Pompa air berguna untuk melakukan penyiraman, *push-button* berfungsi untuk melakukan *setpoint* dan LCD berfungsi

untuk menampilkan setiap *input*-an yang terdeteksi oleh sensor dalam bekerja.

2. Analisis Kebutuhan Sistem

Dalam perancangan robot pemadam api, diperlukan *software* CodevisionAVR untuk merancang semua jenis *input*-an terhadap robot yang akan digunakan. *Software* CodevisionAVR diinstal pada laptop Asus yang mempunyai *processor* intel core i3, RAM 4 GB, dan memori laptop 458 GB. Adapun kebutuhan lainnya ditunjukkan oleh Tabel 1.

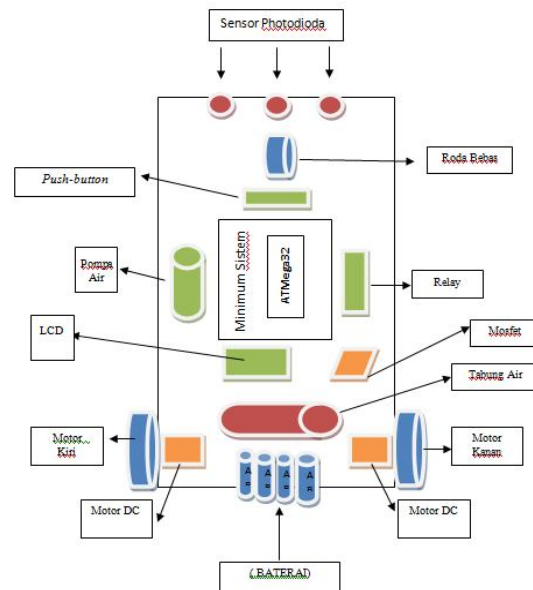
Tabel 1 Analisa Kebutuhan Sistem

NO	ALAT	JUMLAH
1	<i>Photodiode</i>	8
2	<i>Pompa air</i>	1
3	Roda Robot	2
4	Motor DC	2
5	<i>Mosfet</i> 2 pin	2
6	<i>Relay</i>	1
7	LCD	1
8	Catu Daya	10
9	Papan Pcb	3
10	<i>Push-Button</i>	1
11	Body Robot	1
12	Tabung Air	1

3. Perancangan *Hardware*

a. Perancangan Mekanik

Perancangan Robot *Line Follower* Penyiraman Tanaman ditunjukkan oleh Gambar 13.

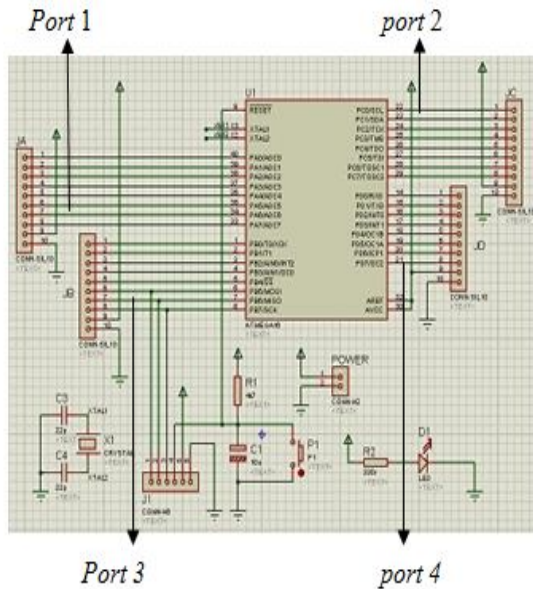


Gambar 13 Perancangan robot tampak atas

Gambar 13 menunjukkan bahwa robot menggunakan 3 roda, 2 roda sebagai pengontrol yang disimpan di kiri dan di kanan samping robot, dan 1 roda disimpan di depan robot, sedangkan sensor *Photodiode* disimpan didepan, Sismis, pompa, *push-button* dan LCD disimpan dibagian atas robot.

b. Perancangan Elektronik

Pada desain robot pemadam api dibuat skematik rangkaian berupa hubungan antara *microcontroller* dengan komponen-komponen pendukung lainnya, selanjutnya data diproses oleh *microcontroller* melalui pin-pin yang ada, LED ditempatkan pada *Port 1*, sensor *Photodiode* ditempatkan pada *Port 2*, mosfet, motor DC, LCD, *relay* berada di *Port 3* dan *push-button* ditempatkan pada *Port 4*. Adapun Komponen-komponen utama dari rangkaian robot pemadam api ini dapat dilihat didalam Skematik rangkaian keseluruhan ditunjukkan oleh Gambar 14.



Gambar 14 Skematik Robot *Line Follower* Penyiram

4. Perancangan *Software*

a. Flowchart Secara Umum

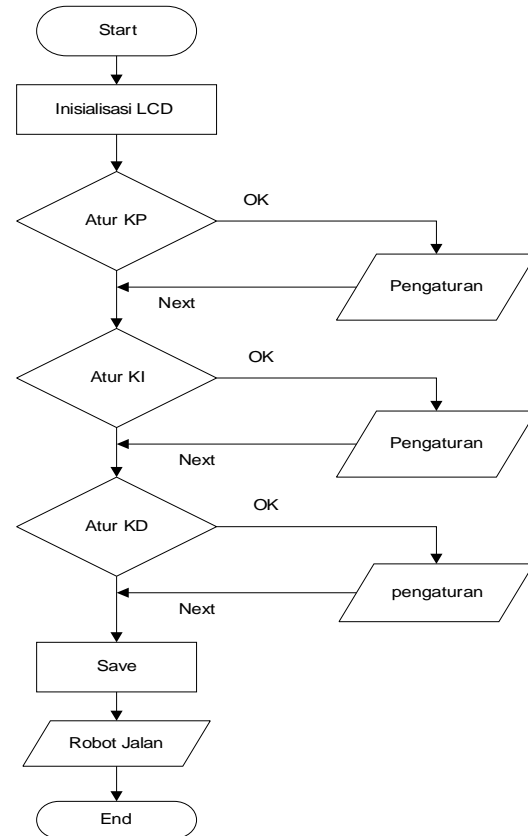
Flowchart sistem secara umum Robot *Line Follower* Penyiraman Tanaman ditunjukkan oleh Gambar 15.

Flowchart secara umum dapat dijelaskan, saat *sensor Photodiode* robot mendeteksi garis hitam maka robot akan mengikuti jalur tersebut dan melakukan proses PID sebagai pengatur pergerakan robot terhadap *error* yang didapat,

robot akan berhenti untuk melakukan penyiraman selama 5 detik ketika robot mendeteksi adanya perempatan pada jalur dan akan berhenti/*finish* ketika mendeksi jalur pertigaan.

b. *Flowchart* Kontrol PID

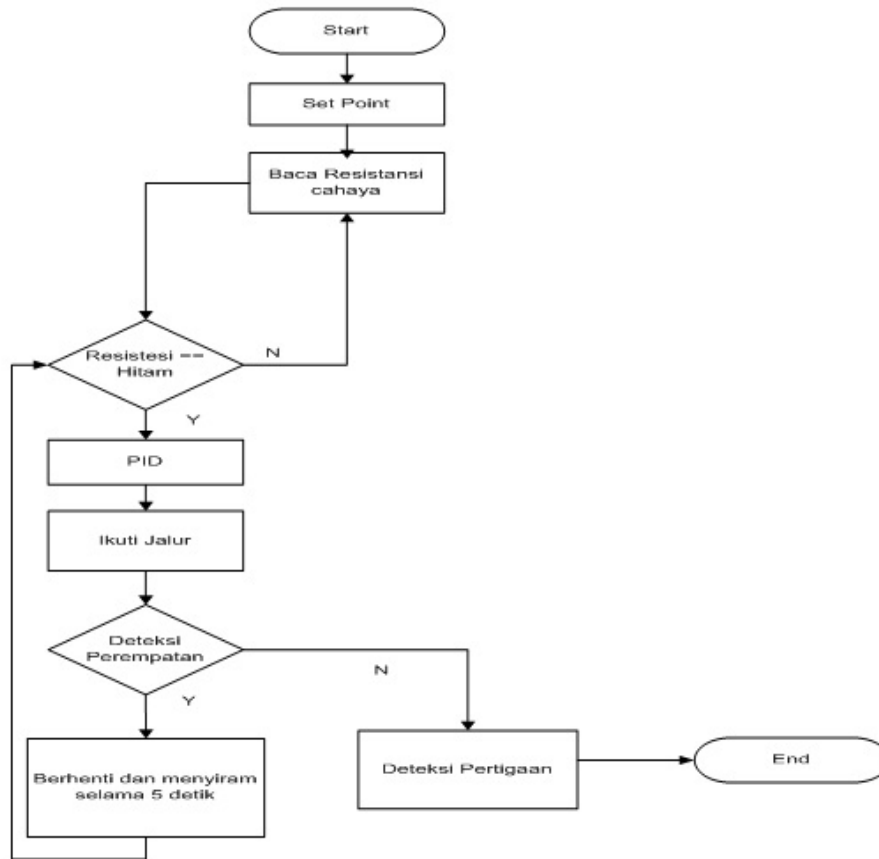
Flowchart Setting Kontrol PID pada Robot *Line Follower* Penyiraman Tanaman ditunjukkan oleh Gambar 16.



Gambar 16 *Flowchart* Kontrol PID

Pada saat robot dinyalakan, robot dapat di-*setting* melalui menu atau langsung di-*start*. Jika robot di-*setting* maka akan masuk ke menu atur Kp, ketika Kp dianggap perlu dirubah maka dapat di *setting* dan ketika dirasa cukup maka dapat *next*. Setelah *pengaturan* Kp maka akan masuk ke menu atur KD, ketika KD dianggap perlu dirubah maka dapat di *setting* dan ketika dirasa cukup maka dapat *next*. Setelah *pengaturan* KD maka akan masuk ke menu atur KI, ketika KI dianggap perlu dirubah maka dapat di *setting* dan ketika dirasa cukup maka dapat *next*. Kemudian ketika semua sudah selesai maka data akan tersimpan

di memori *microcontroller* ATmega32. Robot dapat dijalankan dengan menekan tombol ke 4.

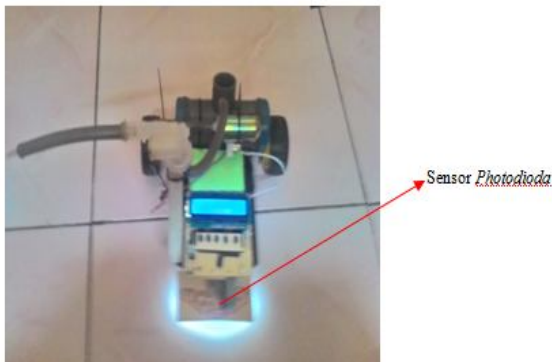


Gambar 15 Flowchart Secara Umum

3. HASIL DAN PEMBAHASAN

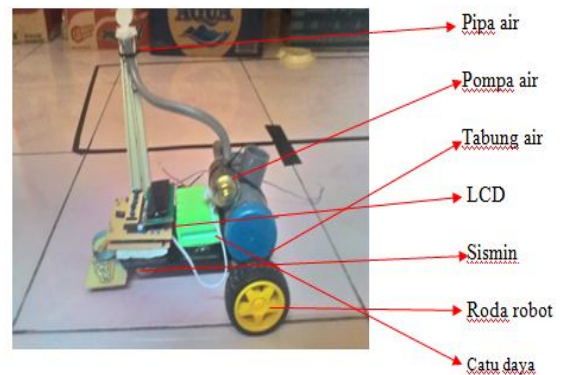
3.1 Implementasi Sistem

Pada implementasi ini akan dijelaskan masing-masing kegunaan komponen utama pada robot, Gambar 17 menunjukkan tampilan robot tampak depan.



Gambar 17 Robot Tampak Depan

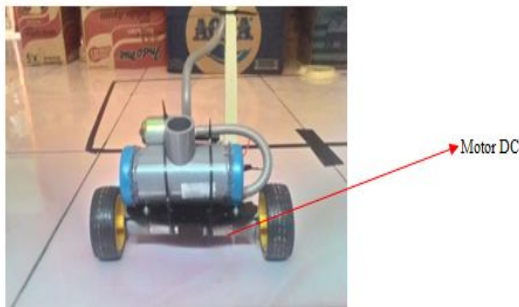
Sensor *Photodiode* ada 8 buah disimpan di depan robot karena saat robot berjalan sensor *Photodiode* yang mendeteksi jalur akan dilalui robot untuk melakukan proses penyiraman tanaman. Gambar 18 menunjukkan tampilan robot tampak samping.



Gambar 18 Robot Tampak Samping Kiri

Pipa air disimpan dibagian kanan atas agar dapat menjangkau pot tanaman yang akan

disiram, LCD robot disimpan di bagian atas dari sismin agar saat LCD menampilkan nilai *range* sensor, nilai-nilai yang dikeluarkan dapat dilihat dengan jelas oleh pengguna robot, Lalu *sismin* tepat dibagian bawah LCD dimana komponen *sismin* terdiri dari Atmega32, mosfet, *relay* dan *push-button*, agar mempermudah pemasangan kabel yang mengarah ke Atmega32 dan tidak sulit untuk dilepas maka *sismin* diletakkan agak depan, sedangkan tabung dan pompa air diletak dibelakang sejajar dengan roda robot karena titik terberat harus diletakkan sejajar dengan roda agar dapat menjaga keseimbangannya, catu daya diletakkan ditengah badan robot. Gambar 19 menunjukkan tampilan tampak robot belakang.



Gambar 19 Robot tampak belakang

Motor DC diletakan di bagian belakang robot karena posisi ban motor berada dibelakang, motor DC merupakan pemutar roda robot yang terhubung dengan *MOSFET* sebagai pengatur arus pergerakan roda kanan dan kiri, yang terhubung dalam pin-pin Atmega32.

3.2 Pengujian Robot *Line Follower* Penyiram Tanaman

Pengujian sistem robot *line follower* penyiram tanaman terdiri atas 2 jenis pengujian sistem kendali yaitu:

1. Pengujian Sensor *Photodiode* terhadap pembacaan garis sebagai jalur pergerakan robot.
2. Pengujian keberhasilan robot berhenti di garis perempatan untuk melakukan penyiraman dan berhenti atau *finish* di garis pertigaan pada jalur.
 - a. Pengujian Robot Mendeteksi Garis
 Pengujian garis dilakukan dengan membuat jalur lurus dan terdapat 3 garis perempatan dan 1 garis pertigaan

menggunakan solasi atau lakban hitam yang berukuran 2.5 cm. Gambar 20 menunjukkan jalur pergerakan robot *line follower* penyiram tanaman.



Gambar 20 Jalur Robot

Pengujian kondisi robot dengan garis ditunjukkan oleh Tabel 2.

Tabel 2 Pengujian *photodiode* mendeteksi garis

No	Sensor <i>Photodiode</i>	Bacaan LCD jalur/garis hitam	Ket
1	1	1	DETEKSI
2	2	1	DETEKSI
3	3	1	DETEKSI
4	4	1	DETEKSI
5	5	1	DETEKSI
6	6	1	DETEKSI
7	7	1	DETEKSI
8	8	1	DETEKSI

Dari pengujian sensor *Photodiode* yang diuji didapatkan bahwa 8 sensor *Photodiode* dapat membaca garis hitam sebagai jalur yang digunakan robot bergerak untuk melakukan proses penyiraman tanaman, dimana pembacaan LCD menunjukan sensor bernilai 1 sebagai nilai ketetapan ketika mendeteksi jalur/garis hitam.

3.3 Pengujian Robot Melakukan Penyiraman

Pengujian dilakukan untuk mengetahui presentasi keberhasilan robot berhenti diperempatan melakukan penyiraman dan *finish* dipertigaan pada jalur. Pengujian kendali robot dilakukan sampai 15 kali pengujian untuk jalur perempatan dan jalur pertigaan, yang ditunjukkan oleh Tabel 3.

Diketahui bahwa saat robot berjalan dan mendeteksi jalur perempatan robot akan berhenti selama 5 detik untuk melakukan penyiraman, jumlah keberhasilan robot

mencapai 85%, 15% ketidak keberhasilan dipengaruhi catu daya (Baterai) yang tidak memadai dan pembacaan sensor yang kurang akurat.

Tabel 3 Pengujian perintah jenis kendali robot

No	Perintah	Jumlah keberhasilan perintah	Presentasi keberhasilan
1	Perempatan	13	85 %
2	Pertigaan	15	100 %

3.4 Pengujian Robot Melakukan Penyiraman Jalur Berliku

Pada pengujian ini arena robot didesain menggunakan jalur yang berliku dimana robot akan melewati jalur tersebut untuk melakukan penyiraman. Dalam pengujian ini robot dapat berhasil melewati jalur tersebut namun untuk berhenti melakukan penyiraman, dalam beberapa kali pengujian robot tidak berhenti disatu titik tempat untuk melakukan penyiraman yang dikarenakan pembacaan sensor yang tidak akurat. Pengujian robot ditunjukkan oleh Tabel 4.

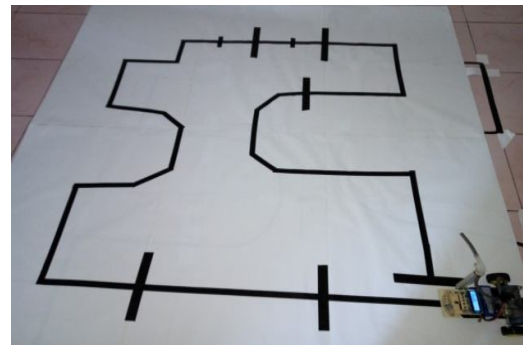
Tabel 4 Pengujian Penyiraman Jalur berliku

NO	POT 1	POT 2	POT 3	POT 4	KEBERHASILAN
1	Menyiram	Menyiram	Tdk Menyiram	Tdk Menyiram	50%
2	Menyiram	Menyiram	Menyiram	Menyiram	100%
3	Tdk Menyiram	Menyiram	Menyiram	Tdk Menyiram	50%
4	Tdk Menyiram	Tdk Menyiram	Tdk Menyiram	Menyiram	25%
5	Menyiram	Menyiram	Tdk Menyiram	Menyiram	75%
6	Menyiram	Menyiram	Menyiram	Menyiram	100%
7	Tdk Menyiram	Menyiram	Menyiram	Menyiram	75%
8	Menyiram	Menyiram	Tdk Menyiram	Menyiram	75%
9	Tdk Menyiram	Menyiram	Menyiram	Menyiram	75%
10	Menyiram	Tdk Menyiram	Menyiram	Menyiram	75%
11	Tdk Menyiram	Menyiram	Menyiram	Menyiram	75%
12	Menyiram	Menyiram	Menyiram	Menyiram	100%
13	Menyiram	Menyiram	Menyiram	Menyiram	100%
14	Menyiram	Menyiram	Menyiram	Menyiram	100%
15	Menyiram	Menyiram	Menyiram	Menyiram	100%

Tabel 4 menunjukkan bahwa saat pengujian, robot terkadang melewati beberapa pot untuk melakukan penyiraman, dimana presisi pembacaan sensor yang tidak akurat pembacaannya pada saat melintasi jalur perempatan untuk berhenti dan melakukan penyiraman. Gambar 21 menunjukkan pengujian robot untuk jalur berliku.

3.5 Pengujian Jenis Jalur Robot

Pada pengujian ini robot akan melewati berbagai jenis jalur. pengujian robot untuk berbagai jalur ditunjukkan oleh Tabel 5.



Gambar 21 Pengujian Jalur Berliku

Tabel 5 Pengujian jenis jalur

NO	Jenis Jalur	Keterangan
1	Belokan kiri dan kanan	Berhasil
2	Belokan Zigzag	Berhasil
3	Jalur Melingkar	Berhasil
4	Jalur putus-putus pendek	Berhasil
5	Jalur perempatan	Berhasil
6	Jalur pertigaan	Berhasil

Tabel 5 menunjukkan bahwa saat pengujian robot, robot dapat berhasil membaca berbagai jenis jalur diantaranya belokan kiri dan kanan, belokan zigzag, jalur melingkar, jalur putus-putus pendek, jalur perempatan dan jalur pertigaan yang ada pada lintasan robot melakukan penyiraman.

3.6 Pengujian Kontrol PID

Pada pengujian kontrol PID akan dilihat bagaimana pengaruh bobot nilai yang diberikan disetiap masing-masing kontrol yaitu *proPortional*, *integral* dan *derivative*. Tabel 6 menunjukkan pengujian kontrol PID robot.

Tabel 6 Pengujian Kontrol PID

NO	<i>ProPortional</i>	<i>Integral</i>	<i>Derivative</i>	Pengaruh
1	21	9	150	Robot sangat bergetar
2	18	7	100	Robot agak bergetar
3	16	6	70	Robot agak bergetar
4	16	6	0	Robot kurang bergetar
5	12	4	2	Robot lambat merespon

Tabel 6 menunjukkan bahwa pengujian terlihat ketika semakin besar bobot nilai PID yang diberikan maka robot semakin bergetar dan tidak stabil dalam berjalan, sedangkan semakin kecil bobot nilai PID yang diberikan maka robot semakin lambat dalam berjalan dan merespon pembacaan jalur, jadi dari

pengujian pemberian bobot nilai PID yang stabil digunakan yaitu pada pengujian poin 4 dimana nilai *proportional* = 16, *integral* = 6 dan *derivative* = 0.

4. KESIMPULAN

Dari hasil penelitian dan implementasi sistem robot *line follower* penyiram tanaman yang telah diuraikan pada bab sebelumnya, maka dapat ditarik kesimpulan sebagai berikut:

1. Dalam tahap pengimplementasian robot *line follower* penyiram tanaman, aplikasi *CodevisionAVR* berhasil melakukan pengendalian terhadap robot untuk mengimplementasikan kinerja robot melalui perintah *coding* yang ditulis, sehingga robot dapat menjalankan segala perintah yang telah ada di program.
2. Metode PID berhasil diimplementasikan ke robot dengan memproses nilai *error* yang didapat robot sehingga robot dapat berjalan dengan baik.
3. Fitur sensor *Photodiode* berhasil diimplementasikan sebagai pendeteksi garis, sehingga saat beroperasi robot dapat mendeteksi jalur yang akan dilalui untuk melakukan proses penyiraman dan berhenti.
4. Didapatkan saat pengujian robot, tingkat keberhasilan robot *line follower* penyiram tanaman untuk melakukan proses penyiraman mencapai 85 % dengan pengujian yang dilakukan 15 kali.

5. SARAN

Adapun saran penulis untuk penelitian selanjutnya yaitu:

1. Menambahkan fitur pendeteksi kelembaban pada tanaman sehingga robot dapat lebih cerdas dalam melakukan proses penyiraman.
2. Menambahkan sensor *photo* sehingga robot lebih akurat dalam pembacaan garis.
3. Pada pengujian, robot hanya mampu membaca garis hitam dan hanya disetting untuk melakukan penyiraman selama 5 detik saja, oleh karena itu perlu ditambahkan sensor kelembaban agar robot dapat mengatur sendiri lama waktu robot melakukan proses penyiraman.

4. Pada pengembangan selanjutnya penggunaan mosfet dapat diganti menggunakan resistor agar rangkaian alat lebih sederhana. [relay-fungsi-relay/](#), diakses pada 11 Maret 2015.

DAFTAR PUSTAKA

- [1] McComb, G., 2001, *The Robot Builder's Bonanza Second Edition*, McGraw-Hill, United States of America.
- [2] Budiharto, W., 2009, *Robotika teori dan implementasi*, Andi Publisher Jakarta.
- [3] Artanto, D., 2012, *Yuk Membuat Robot*, Gramedia Widiasarana Indonesia. Jakarta.
- [4] Bolton, W., 2006, *Sistem Instrumentasi dan system Kontrol*, Penerbit Erlangga, Jakarta.
- [5] Saputro, G., 2012, Sistem Kontrol Motor Robot Line Follower Berbasis Microcontroller ATmega32 Menggunakan Algoritma PID (Proporsional Integral Derivatif), http://repository.amikom.ac.id/files/publikasi_11.21.0565.PDF, diakses pada 11 Februari 2015.
- [6] Irma, 2013, Sensor Photodiode, http://irmatrianjaswati-fst11.web.unair.ac.id/artikel_detail-84996-Sensor-sensor%20Photodiode.html, diakses pada 13 Februari 2015.
- [7] Elektronika, 2012, MOSFET sebagai saklar, <http://elektronika-dasar.web.id/teori-elektronika/mosfet-sebagai-saklar/>, diakses pada 11 Februari 2015.
- [8] Yusuf, T., 2014, Penjelasan tentang push-button, <http://belajar-pik.heck.in/penjelasan-tentang-push-button.xhtml>, diakses pada 15 Februari 2015.
- [9] Teknikelektronika, 2012, Pengertian dan fungsi relay, <http://teknikelektronika.com/pengertian->