

Implementasi Logika Fuzzy Pada Robot Beroda Untuk Menentukan Arah Jalan Keluar Dari Suatu Ruangan

Agus Siswanto

Program Studi Teknik Informatika, STIKOM Dinamika Bangsa Jambi

Jl. Jendral Sudirman Thehok – Jambi, telp (0741) 35095

E-mai: agussiswanto@stikom-db.ac.id

Abstract

Wheeled Robots are one example of current technological developments, especially in the field of robotics. Actuators in this robot is to use the wheel as a tool to move or move from one position to another position. One proof of the development of this wheeled robot is by the many studies that raised about wheeled robots of various uses and purposes. As for the control system in this robot many methods or algorithms that can be used one of them is fuzzy logic. The shortcomings that exist in previous research is the robot will check the existence of obstacles or not. Then the robot will follow the object of the barrier or the room, so that if the robot is inside a room the robot will check all the rooms until it finds a way out. This of course requires a longer time than if the robot immediately know where the outlet position and the robot will go straight to the way out.

Keywords: Implementation, Fuzzy Logic, Wheeled Robot, Room.

Abstrak

Robot beroda (*Wheeled Robots*) merupakan salah satu contoh dari perkembangan teknologi saat ini, khususnya di bidang robotika. Aktuator pada robot ini adalah dengan menggunakan roda sebagai alat untuk bergerak atau berpindah dari posisi satu ke posisi yang lain. Salah satu bukti dari perkembangan robot beroda ini adalah dengan banyaknya penelitian yang mengangkat tentang robot beroda dari berbagai kegunaan dan keperluan. Sedangkan untuk sistem kendali pada robot ini banyak metode atau algoritma yang dapat dipergunakan salah satunya adalah Logika *fuzzy*. Kekurangan yang ada pada penelitian sebelumnya adalah robot akan mengecek adanya objek penghalang atau tidak. Kemudian robot akan mengikuti objek penghalang atau ruangan tersebut, sehingga jika robot berada di dalam suatu ruangan robot akan mengecek semua ruangan sampai menemukan jalan keluar. Hal tersebut tentu saja membutuhkan waktu yang lebih lama dibandingkan jika robot langsung mengetahui dimana posisi jalan keluar dan robot akan langsung menuju ke arah jalan keluar tersebut.

Kata Kunci: Implementasi, Logika Fuzzy, Robot Beroda, Ruangan.

© 2018 Jurnal PROCESSOR.

1. Pendahuluan

Robot beroda (*Wheeled Robots*) merupakan salah satu contoh dari perkembangan teknologi saat ini, khususnya di bidang robotika. Aktuator pada robot ini adalah dengan menggunakan roda sebagai alat untuk bergerak atau berpindah dari posisi satu ke posisi yang lain. Salah satu bukti dari perkembangan robot beroda ini adalah dengan banyaknya penelitian yang mengangkat tentang robot beroda dari berbagai kegunaan dan keperluan. Sedangkan untuk sistem kendali pada robot ini banyak metode atau algoritma yang dapat dipergunakan, antara lain *Dead Reckoning*, *Follow Line*, *Follow Wall*, *Maze Mapping*, *Algoritma Neural Networks*, *Shortest Path*, dan *Fuzzy Logic*.

Logika fuzzy (*Fuzzy Logic*) merupakan pengembangan dari hasil yang didapat dari logika boolean. Logika boolean hanya mengenal nilai kebenaran atau tidak, sehingga kebenaran yang didapat belum tentu kebenaran yang sebenarnya. Contoh nilai yang ada pada boolean adalah 0 dan 1 atau bisa juga naik dan turun atau hitam dan putih atau contoh lainnya yang mempunyai nilai ya atau tidak. Logika fuzzy mengubah nilai kebenaran pada boolean menjadi tingkat kebenaran. Jika kita belum pernah mengetahui bagaimana logika fuzzy, mungkin akan membayangkan kerumitan dan hal-hal yang tidak menyenangkan di dalam logika fuzzy tersebut. Namun, perlu diketahui sebenarnya logika fuzzy sudah ada pada diri kita masing-masing.[1]

Penelitian tentang logika fuzzy telah dilakukan sebelumnya oleh Reza Nandika, yaitu Implementasi Sensor Ultrasonik pada Robot Pengikut Objek dengan Kontrol Logika Fuzzy. Pada penelitian ini robot dapat mempertahankan objek terhadap jarak dan mengikuti pergerakan objek yang ada di depannya dengan menggunakan *Fuzzy Logic Controller* dan membandingkannya dengan PID (Proportional Integral dan Derivative).

Penelitian yang dilakukan oleh Maulidi Rahman dan Hugo Aprilianto, yaitu menerapkan Metode *Fuzzy* Pada Robot Beroda Menggunakan *Omni-Directional Wheels*. Pada penelitian tersebut robot yang dirancang dapat bergerak bebas seperti maju, mundur, bergeser ke kanan dan ke kiri tanpa perlu berbelok karena menggunakan *Omni-Directional Wheels* sebagai penggerak robot. Sedangkan metode fuzzy dipergunakan agar robot dapat menelusuri ruangan. Sehingga dengan metode fuzzy dan *Omni-Directional Wheel* sebagai penggerak, maka pergerakan robot akan semakin cepat dalam menelusuri ruangan.[10]

Pada Penelitian lainnya yang dilakukan oleh Faikul Umam, yaitu mengembangkan sistem kendali pergerakan autonomous mobile robot untuk mendapatkan jalur bebas hambatan menggunakan fuzzy logic controller. Pada penelitian itu robot akan bergerak maju dengan bebas dan akan menghindari secara otomatis jika ada halangan yang mendekatinya.[6]

Dari ketiga penelitian sebelumnya, logika fuzzy telah digunakan untuk sistem kendali robot beroda. Kekurangan yang ada pada penelitian tersebut adalah robot akan memeriksa adanya objek penghalang atau tidak. Kemudian robot akan mengikuti objek penghalang atau ruangan tersebut, sehingga jika robot berada di dalam suatu ruangan robot akan mengecek semua ruangan sampai menemukan jalan keluar. Hal tersebut tentu saja membutuhkan waktu yang lebih lama dibandingkan jika robot langsung mengetahui dimana posisi jalan keluar dan robot akan langsung menuju ke arah jalan keluar tersebut.

Berdasarkan uraian diatas, maka penulis tertarik untuk melakukan penelitian dengan judul “Implementasi Logika Fuzzy Pada Robot Beroda Untuk Menentukan Arah Jalan Keluar Dari Suatu Ruangan”.

2. Tinjauan Pustaka

2.1 Pengertian Robot

Kata robot berasal dari *Czech, robota* yang artinya pekerja. Robot adalah perangkat atau alat yang bekerja secara otomatis yang mampu melakukan aktifitas menyerupai manusia. [2]

Sedangkan pengertian lainnya robot adalah sebuah mesin yang berbentuk seperti manusia yang bisa bergerak dan berbicara. Pendapat ini didasari oleh kemunculan film-film robot seperti *Robocop, Transformer, Astro Boy, Wall-E*, dan sebagainya. [3]

Dari ungkapan para ahli diatas dapat disimpulkan bahwa robot itu memiliki kemampuan untuk membantu pekerjaan manusia yang dapat bergerak dan berbicara dan mampu bergerak secara otomatis tanpa harus dioperasikan oleh manusia.

2.1.1 Macam-Macam Bentuk Robot

Adapun macam-macam bentuk robot berdasarkan bentuknya antara lain [4]:

1. *Mobile* Robot atau robot yang bisa berpindah-pindah.

Mobile Robot adalah konstruksi robot yang ciri khasnya adalah mempunyai aktuator berupa roda untuk menggerakkan keseluruhan badan robot tersebut, sehingga robot tersebut dapat melakukan perpindahan posisi dari satu titik ke titik yang lain.

2. Robot tangan (*robot Manipulator*)
Robot ini hanya memiliki satu tangan seperti tangan manusia yang fungsinya untuk memegang atau memindahkan barang, contoh robot ini adalah robot las di industri mobil, robot merakit elektronik.
3. Robot *Humanoid*
Robot *Humanoid* yaitu robot yang memiliki kemampuan menyerupai manusia, baik fungsi maupun cara bertindak, contoh robot ini adalah Ashimo yang dikembangkan oleh Honda.
4. Robot Berkaki
Robot ini memiliki kaki seperti hewan atau manusia, yang mampu melangkah, seperti robot serangga, robot kepiting.
5. Robot Terbang (*Flying Robot*)
Robot terbang (*Flying Robot*) yaitu robot yang mampu terbang, robot ini menyerupai pesawat yang di program khusus untuk memonitor keadaan tanah dari atas, dan juga untuk meneruskan komunikasi.
6. Robot dalam air (*Under Water Robot*)
Robot ini digunakan di bawah laut untuk memonitor kondisi bawah laut dan juga untuk mengambil sesuatu di bawah laut.

2.1.2 Jenis-Jenis Robot

Secara umum, ada dua jenis robot yaitu robot terkontrol (*controlled robot*) dan robot otomatis (*autonomous robot*). Adapun penjelasan dari dua jenis robot tersebut adalah sebagai berikut [4]:

1. *Aautonomous Robot*
Robot jenis ini bergerak berdasarkan input dari perintah-perintah yang dikirimkan secara manual baik melalui *remote control*, CPU, atau *joystick*.
2. *Automatic Robot*
Automatic Robot bergerak berdasarkan perintah-perintah yang telah di input program sebelumnya atau berdasarkan masukan dari sensor-sensor yang ada pada robot.

2.2 Logika Fuzzy

Logika Fuzzy merupakan kecerdasan buatan yang pertama kali dipublikasikan oleh Prof. Dr. Lotfi Zadeh yang berasal dari Pakistan. Melalui logika fuzzy ini sistem dapat membuat keputusan sendiri dan terkesan seperti memiliki perasaan, karena memiliki keputusan lain selain iya dan tidak. Oleh karena itu fuzzy logic sangat berbeda jauh dari alur algoritma pemrograman.[5]

Fuzzy dapat diartikan kabur atau samar. Di dalam fuzzy suatu nilai dapat berarti benar atau salah secara bersamaan. Keanggotaan dari fuzzy adalah rentang nilai 0 sampai dengan 1, bukan bernilai 0 dan 1 atau ya dan tidak. Namun berapa besar rentang nilai tergantung dari bobot keanggotaan yang dimilikinya.[6]

Logika Fuzzy dapat dipergunakan untuk menerjemahkan suatu besaran yang dinyatakan menggunakan *linguistic* (ilmu bahasa). Hal tersebut dapat dicontohkan misalkan besaran kecepatan laju suatu kendaraan yang dinyatakan dengan pelan, agak cepat, cepat, dan sangat cepat. Dalam hal ini logika fuzzy menunjukkan sejauh mana suatu nilai dapat dikatakan benar dan sejauh mana nilai dikatakan salah. Tidak seperti logika klasik atau tegas yang hanya mempunyai dua kemungkinan yaitu cepat atau lambat. Derajat keanggotaan lambat artinya bukan merupakan anggota himpunan dan cepat berarti adalah anggota himpunan. Oleh sebab itu dengan fuzzy sesuatu dapat dikatakan sebagian benar dan sebagian salah pada waktu yang sama.

Contoh penggunaan Logika Fuzzy kedalam sistem informasi dan rekayasa proses adalah menghasilkan aplikasi seperti sistem kontrol, alat-alat rumah tangga, dan sistem pengambil keputusan (*decision support system*). Dalam hal ini dapat dikatakan bahwa logika fuzzy memimpin dalam pengembangan kecerdasan mesin yang lebih tinggi (*machine Intelligency Quotient/MIQ*). Contoh produk-produk yang telah menggunakan logika fuzzy adalah mesin cuci, video dan kamera refleksi lensa tunggal, pendingin ruangan, *oven microwave*, dan banyak sistem diagnosa mandiri.

Di dalam diri setiap manusia sebenarnya logika fuzzy sudah ada. Konsep logika fuzzy dapat dengan mudah ditemukan pada perilaku manusia dalam sehari-hari antara lain:

1. Kecepatan kendaraan sesuai dengan seberapa besar kita menekan gas pada kendaraan tersebut. Berapa besar tekanan yang kita berikan?

2. Jika kita memberikan hadiah kepada orang lain, maka besar hadiah tersebut tergantung dari sikap orang tersebut terhadap kita. (Berapa besar nilai hadiah yang akan kita berikan?)
3. Setiap orang akan marah jika merasa dirugikan oleh orang lain. (Berapa besar kemarahan kita?)

Pertanyaan-pertanyaan tersebut tidak bisa kita jawab dengan pasti. Untuk contoh yang bisa dijawab atau dijelaskan dengan pasti menggunakan konsep logika fuzzy adalah sebuah robot yang menggunakan fuzzy logic dapat memprediksikan kapan ia harus bertindak atau menghindar saat ada halangan di depannya. Sedangkan robot yang menggunakan algoritma pemrograman konvensional tidak akan dapat memutuskan sendiri untuk menghindar dari halangan yang ada di depannya.[1]

2.3 Mikrokontroler

Mikrokontroler, sebagai suatu terobosan teknologi mikroprosesor dan mikrokomputer, hadir memenuhi kebutuhan pasar (*market need*) dan teknologi baru. Sebagai teknologi baru, yaitu teknologi dengan kandungan transistor yang lebih banyak namun hanya membutuhkan ruang yang kecil serta dapat diproduksi secara massal (dalam jumlah banyak) membuat harganya menjadi lebih murah (dibandingkan mikroprosesor). Atmel sebagai salah satu vendor yang mengembangkan dan memasarkan produk mikroelektronika telah menjadi suatu teknologi standar bagi para desainer AVR (*Atmega*), para desainer sistem elektronika telah diberi suatu teknologi yang memiliki kapasitas yang amat maju, tetapi dengan biaya ekonomis yang cukup minimal.

Mikrokontroler dapat didefinisikan sebagai sebuah sistem *microprocessor* dimana didalamnya sudah terdapat CPU, ROM, RAM, I/O, *Clock* dan Peralatan lainnya yang sudah saling terhubung dan terorganisir dengan baik oleh pabrik pembuatnya dan dikemas dalam satu *chip* yang siap pakai.[7]

Selain definisi diatas, mikrokontroler juga dapat didefinisikan sebagai sebuah chip terintegrasi yang biasanya menjadi bagian dari sebuah embedded sistem (sistem yang didesain untuk melakukan satu atau lebih fungsi khusus yang real time). Mikrokontroler terdiri dari CPU, Memory, I/O port dan timer seperti sebuah komputer standar, tetapi karena didesain hanya untuk menjalankan satu fungsi yang spesifik dalam mengatur sebuah sistem.[8]

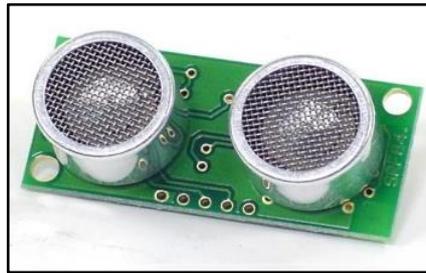
Dari beberapa definisi diatas dapat disimpulkan bahwa mikrokontroler adalah sebuah chip yang memiliki CPU, ROM, RAM, I/O, *Clock* dan peralatan lainnya yang memiliki kemampuan mengendalikan sistem-sistem secara otomatis dan berdiri sendiri. Oleh karena itulah mikrokontroler ini sangat praktis untuk digunakan dalam berbagai aplikasi karena menghemat ruang dan waktu dalam perakitan aplikasinya.

Mikrokontroler terdiri atas beberapa bagian yang saling terhubung sehingga mikrokontroler dapat melakukan tugas sesuai dengan program yang ada didalamnya. Bagian-bagian penyusun mikrokontroler standar adalah:

- a. Unit Memori
- b. CPU (*Central Processing Unit*)
- c. Bus
- d. Unit Input/Output
- e. Pembangkit *Clock Oscillator*
- f. Unit *Timer/Counter*
- g. Komponen Tambahan
- h. Program

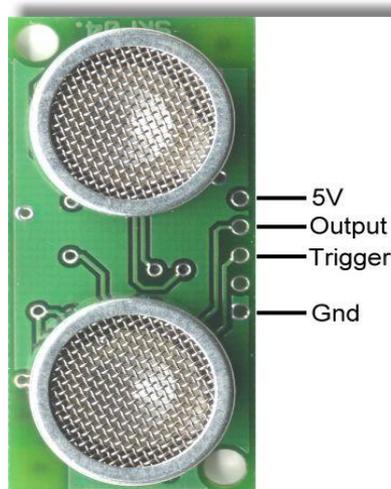
2.4 Devantech SRF04 Ultrasonic Range Finder

Sensor jarak merupakan sensor yang wajib ada pada robot terkini. *Devantech* ialah salah satu sensor jarak yang digunakan paling banyak pada Kontes Robot di Indonesia. *Devantech SRF04 Ultrasonic Range Finder* memberikan informasi jarak kisaran 3cm sampai dengan 3m. Bentuk dari sensor *Devantech SRF04 Ultrasonic Range Finder* dapat dilihat pada gambar berikut [9]:



Gambar 1. *Devantech SRF-04 Ultrasonic Range Finder [9]*

Kit dari sensor *Devantech SRF04 Ultrasonic Range Finder* ini sangat mudah untuk dirangkai dan membutuhkan sumber daya yang kecil sekali. Sangat ideal untuk aplikasi mobile robot. Susunan dari kakinya adalah sebagai berikut [9]:



Gambar 2. *Susunan Kaki SRF-04 [9]*

Sensor SRF04 ini bekerja dengan cara memancarkan pulsa suara dengan kecepatan suara (0.9 ft/milidetik). Karakteristik dari sensor ini dapat dilihat pada tabel berikut [9]:

Tabel 1. *Karakteristik SRF-04 [9]*

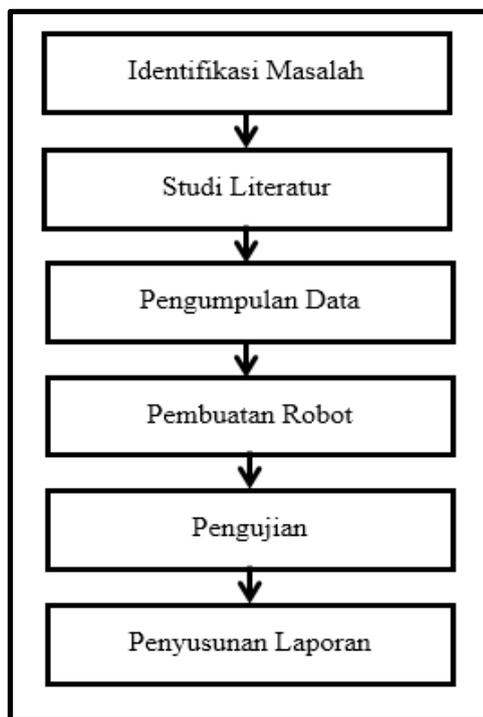
Tegangan	5v
Arus	30mA Typ. 50mA Max
Frekwensi	40 KHz
Maximum Range	3 Meter
Minimum Range	3 CentiMeter
Sensitifitas	Detect a 3cm diameter stick at > 2m
Trigger Input	10uS Min. TTL level pulse
Pulsa Echo	Positive TTL level signal, width proportional to range

3. Metodologi

3.1 Kerangka Kerja Penelitian

Pada suatu penelitian kita perlu membuat suatu kerangka kerja penelitian agar apa saja yang kita butuhkan untuk merancang suatu sistem dapat tergambar dengan jelas dan dapat diperoleh suatu logika, baik didalam melakukan pengujian maupun dalam membuat kesimpulan. Kerangka kerja (*framework*) penelitian merupakan sebuah bagan yang terdiri atas tahapan-tahapan yang tersusun secara sistematis

yang kemudian akan digunakan dalam proses penyelesaian penelitian. Adapun kerangka kerja penelitian yang akan digunakan adalah sebagai berikut:



Gambar 3. Kerangka Kerja Penelitian

Berdasarkan kerangka kerja penelitian pada gambar 3, maka dapat diuraikan pembahasan dari masing-masing tahapan dalam penelitian ini adalah sebagai berikut:

1. Identifikasi Masalah

Tahap ini adalah proses dan hasil pengenalan masalah. Dengan kata lain, identifikasi masalah adalah salah satu proses penelitian yang boleh dikatakan paling penting di antara proses lain. Masalah penelitian akan menentukan kualitas suatu penelitian, bahkan itu juga menentukan apakah sebuah kegiatan bisa disebut penelitian atau tidak.

2. Studi Literatur

Pada tahapan pertama ini penulis menghimpun data-data atau sumber-sumber yang berhubungan dengan topik penelitian. Adapun data atau sumber yang dimaksud adalah seperti jurnal, buku, dan internet. Data yang didapat dari studi literatur ini akan digunakan sebagai acuan untuk merancang dan membuat robot di dalam penelitian ini.

3. Pengumpulan Data

Pada tahapan proses ini, dilakukan pengumpulan data yaitu penulis melakukan pengumpulan data berupa data-data pustaka antara lain: buku teori tentang robot, buku mengenai cara mengoperasikan mikrokontroler, buku elektronika dasar, buku tentang aplikasi *CodeVision AVR*, buku tentang algoritma robot yang berhubungan dengan metode yang digunakan dan bentuk ruangan yang ada jalan keluarnya. Adapun tujuan dari pengumpulan data pustaka ini adalah agar penulis dapat memahami teori dan konsep dari metode yang dipakai serta robot yang akan dirancang.

4. Pembuatan Robot

Pada tahap ini, pertama kali akan dilakukan analisa kebutuhan yaitu untuk mengetahui apa-apa saja yang dibutuhkan dalam membangun robot beroda ini. Dalam merancang sebuah robot yang dikendalikan oleh mikrokontroler, hal yang pertama dilakukan yaitu melakukan perancangan bentuk mekatronika. Setelah didapat bentuk fisik, selanjutnya dilakukan perancangan rangkaian elektronika. Dalam merancang rangkaian elektronika dibagi menjadi beberapa bagian antara lain yaitu rangkaian motor dc, sensor dan rangkaian mikrokontroler yang terhubung ke masing-masing rangkaian. Selanjutnya sebelum melakukan penulisan kode program terlebih dahulu dilakukan perancangan algoritma. Kemudian baru dilanjutkan dengan proses penulisan kode (*coding*) program dan dilanjutkan dengan pengujian. Dalam perancangan algoritmanya, menggunakan *tool* pengembangan

sistem yaitu *flowchart*. Adapun *flowchart* yang dipakai adalah *flowchart* program atau sering juga disebut program *flowchart*.

5. Pengujian

Pada tahapan proses ini, dilakukan proses pengujian. Sebelum melakukan pengujian terlebih dahulu robot harus sudah dalam keadaan selesai dirancang baik dari fisik maupun rangkaiannya. Pengujian alat ini sangat penting karena dengan pengujian inilah dapat diketahui apakah alat yang dibuat dapat berjalan sesuai dengan perencanaan yang diharapkan. Selain itu juga apakah metode yang dipakai yaitu fuzzy dapat diterapkan pada robot ini.

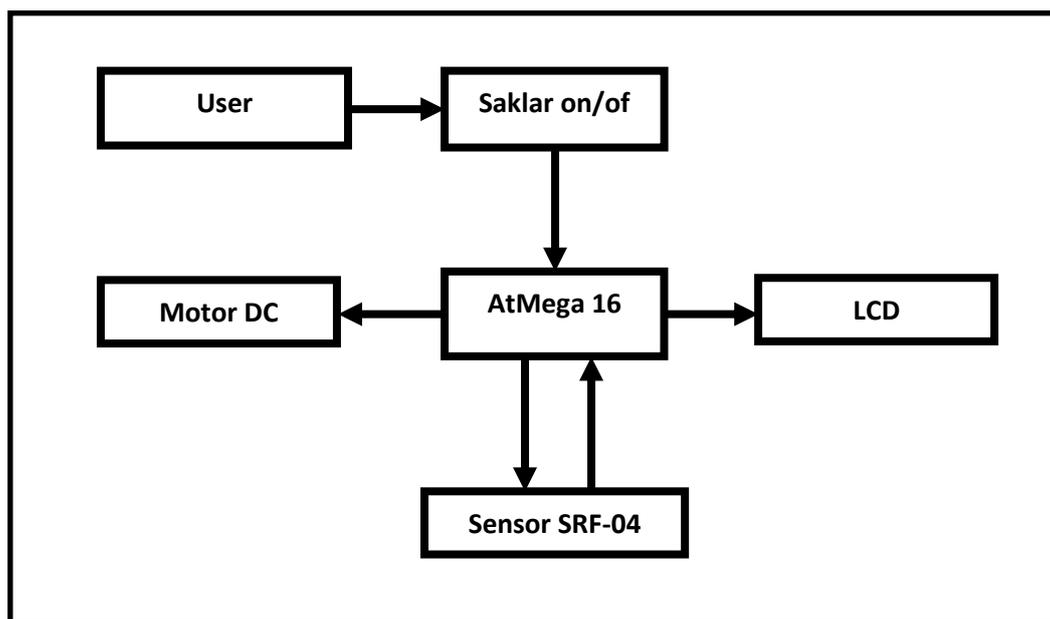
6. Penyusunan Laporan

Pada tahapan proses ini, dilakukan proses penyusunan atau pembuatan laporan yang diperoleh dari hasil penelitian yang dilakukan oleh penulis. Tujuan dari tahap ini adalah agar penelitian ini dapat dibaca sehingga dapat diperoleh kritik maupun saran dari para pembaca. Serta dapat juga dijadikan sebagai bahan acuan dan referensi bagi pengembangan penelitian yang selanjutnya.

4. Hasil dan Pembahasan

4.1 Blok Diagram Sistem Robot Beroda

Blok diagram merupakan sistem yang terintegrasi, karena sistem tersebut tidak dapat bekerja apabila salah satu perangkat tidak ada. Isi dari sistem ini adalah komputer sebagai pusat pengendali utama dengan perangkat lunak (*software*) sebagai instruksi yang dilakukan oleh rangkaian (*input*), USB kita gunakan untuk memasukkan program ke dalam mikrokontroler, dengan mikrokontroler ini akan mengendalikan motor DC untuk menggerakkan robot jika mendapat perintah dari *input* 1 dari program. Block diagram dapat dilihat pada gambar 4:



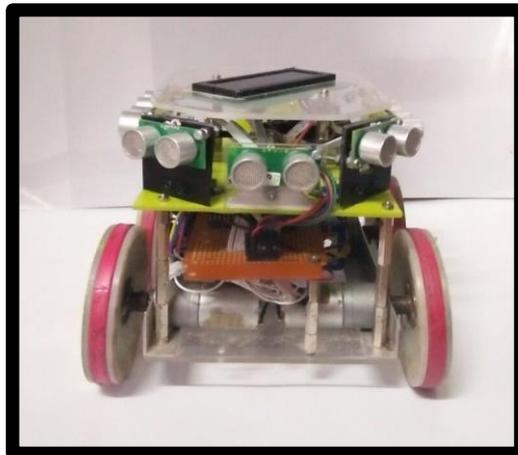
Gambar 4. Blok Diagram Rancangan Robot

Dari gambar 4 dapat dilihat bahwa user yang menghidupkan dan mematikan robot, pengontrolan dilakukan oleh atmega yang telah diprogram. SRF-04 yang menjadi input dan motor dc yang menjadi penggerak robot.

4.2 Bentuk Fisik Robot Beroda

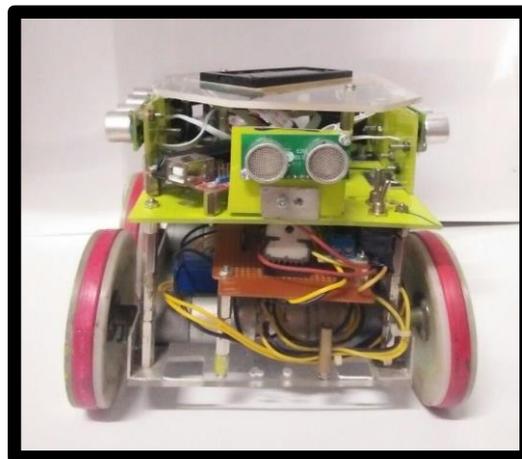
Hasil fisik robot merupakan kegiatan merubah dari rancangan desain robot menjadi bentuk fisik robot yang dapat dieksekusi. Kegiatan ini memadukan rancangan fisik antar muka dan algoritma menjadi satu kesatuan. Berikut ini adalah hasil dari bentuk fisik robot.

1. Fisik Robot Beroda Dari Depan



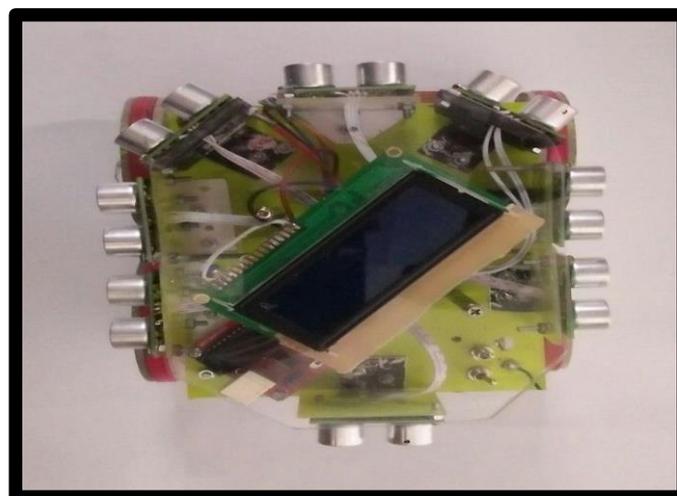
Gambar 5. Fisik Robot Beroda Dari Depan

2. Fisik Robot Beroda Dari Belakang



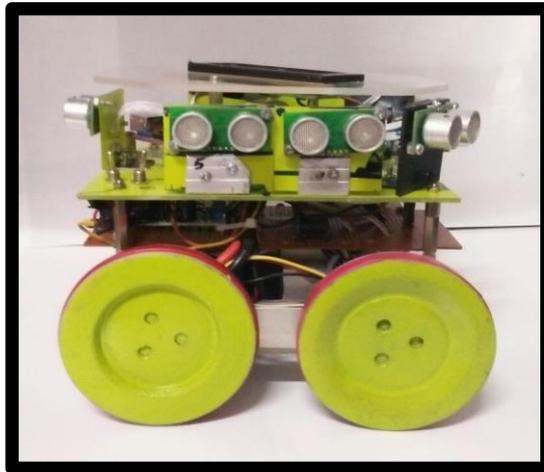
Gambar 6. Fisik Robot Beroda Dari Belakang

3. Fisik Robot Beroda Dari Atas



Gambar 7. Fisik Robot Beroda Dari Atas

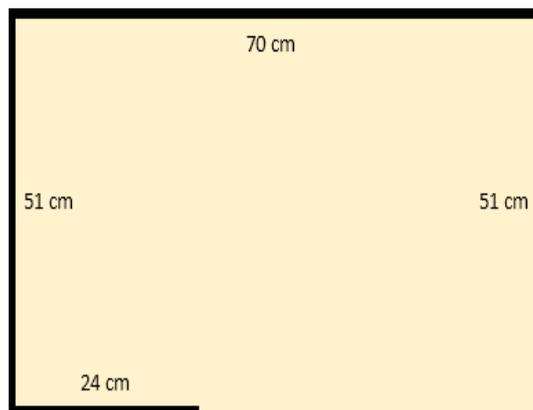
4. Fisik Robot Beroda Dari Samping



Gambar 8. Fisik Robot Beroda Dari Samping

4.3 Bentuk dan Ukuran Ruang Robot

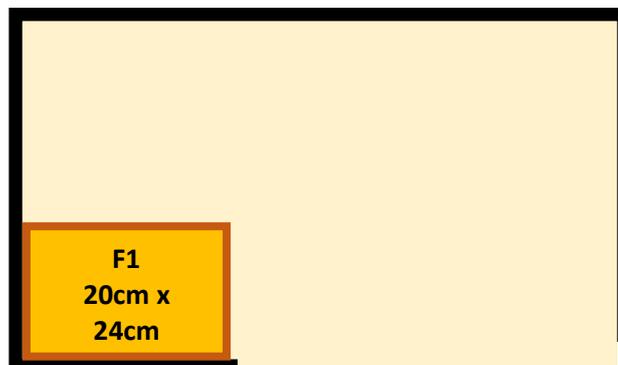
Setelah menentukan ukuran dari fisik robot selanjutnya adalah menentukan bentuk dan ukuran dari ruangan. Adapun bentuk dan ukuran ruangan yang akan dijadikan sebagai tempat pengujian robot adalah sebagai berikut:



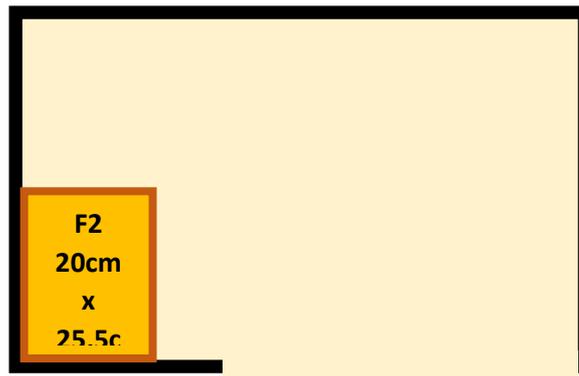
Gambar 9. Bentuk dan Ukuran Ruang Robot

4.4 Posisi Fuzzy Untuk Robot Dalam Ruang

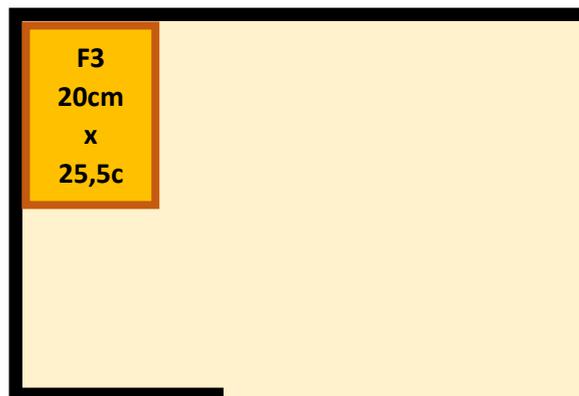
Di dalam ruangan yang sudah ditentukan ukurannya, penulis membagi posisi fuzzy untuk robot menjadi 8 bagian yaitu F1, F2, F3, F4, F5, F6, F7 dan F8. Adapun pembagiannya dapat dilihat sebagai berikut:



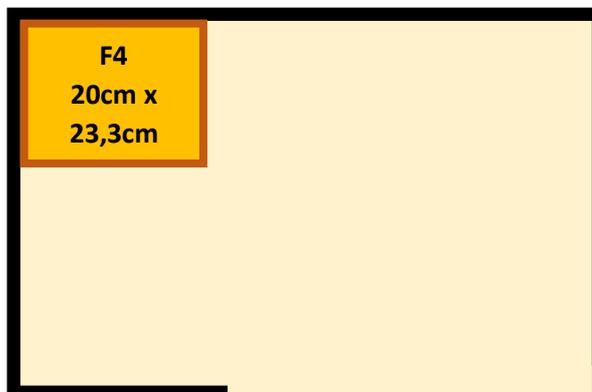
Gambar 10. Posisi Fuzzy F1



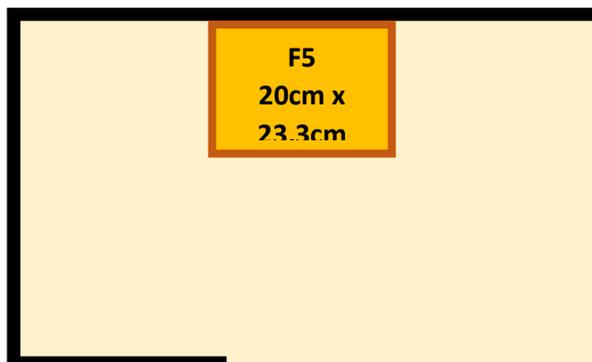
Gambar 11. *Posisi Fuzzy F2*



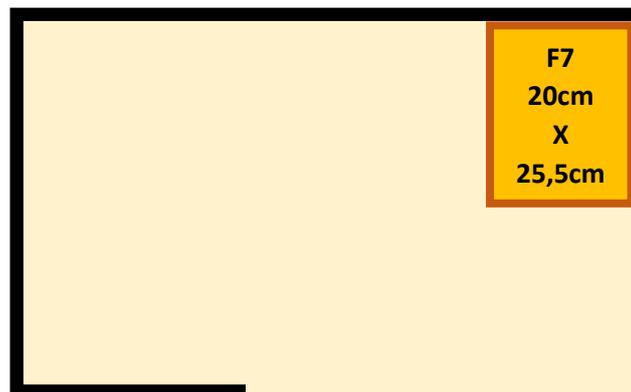
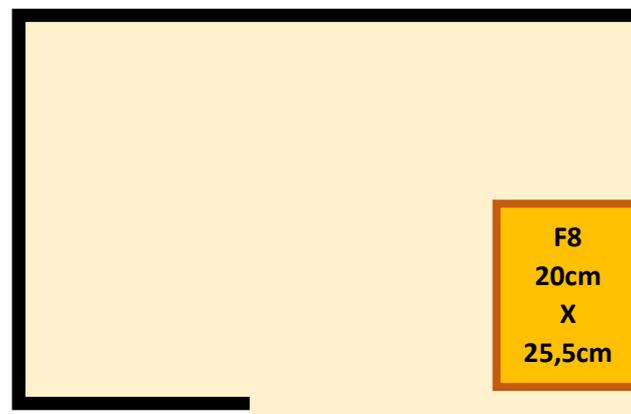
Gambar 12. *Posisi Fuzzy F3*



Gambar 13. *Posisi Fuzzy F4*



Gambar 14. *Posisi Fuzzy F5*

Gambar 15. *Posisi Fuzzy F6*Gambar 16. *Posisi Fuzzy F7*Gambar 17. *Posisi Fuzzy F8*

4.5 Pengujian Gerak Robot

Untuk menguji gerak robot, hal pertama yang dilakukan adalah pembuatan *listing program* dan menentukan berapa deklarasi motor dc yang dibutuhkan untuk menggerakkan motor dc agar robot bisa bergerak maju, berbelok, berhenti, dan mundur.

Tabel 2. *Pengujian Gerak Motor Pada Robot*

Input Port // Keterangan	Tegangan	Gerak Robot
PORTD.0=0; //kn_dpn mundur PORTD.1=1; //kr_dpn maju	12 V DC	maju

PORTD.2=0; //kr_dpn mundur PORTD.3=1; //kn_dpn maju PORTD.4=1; //kr_blk maju PORTD.5=1; //kn_blk maju PORTD.6=0; //kn_blk mundur PORTD.7=0; //kr_blk mundur delay_ms(5);		
PORTD.0=0; //kn_dpn mundur PORTD.1=1; //kr_dpn maju PORTD.2=0; //kr_dpn mundur PORTD.3=1; //kn_dpn maju PORTD.4=1; //kr_blk maju PORTD.5=1; //kn_blk maju PORTD.6=0; //kn_blk mundur PORTD.7=0; //kr_blk mundur delay_ms(5); PORTD.0=1; //kn_dpn mundur PORTD.1=1; //kr_dpn maju PORTD.2=1; //kr_dpn mundur PORTD.3=1; //kn_dpn maju PORTD.4=1; //kr_blk maju PORTD.5=1; //kn_blk maju PORTD.6=1; //kn_blk mundur PORTD.7=1; //kr_blk mundur delay_ms(10);	12 V DC	majuCTFzy
PORTD.0=1; //kn_dpn mundur PORTD.1=1; //kr_dpn maju PORTD.2=1; //kr_dpn mundur PORTD.3=1; //kn_dpn maju PORTD.4=1; //kr_blk maju PORTD.5=1; //kn_blk maju PORTD.6=1; //kn_blk mundur PORTD.7=1; //kr_blk mundur delay_ms(300);	12 V DC	berhenti
PORTD.0=1; //kn_dpn mundur PORTD.1=0; //kr_dpn maju PORTD.2=1; //kr_dpn mundur PORTD.3=0; //kn_dpn maju PORTD.4=0; //kr_blk maju PORTD.5=0; //kn_blk maju PORTD.6=1; //kn_blk mundur PORTD.7=1; //kr_blk mundur delay_ms(5);	12 V DC	mundur
PORTD.0=1; //kn_dpn mundur PORTD.1=1; //kr_dpn maju PORTD.2=0; //kr_dpn mundur PORTD.3=0; //kn_dpn maju PORTD.4=1; //kr_blk maju PORTD.5=0; //kn_blk maju PORTD.6=1; //kn_blk mundur PORTD.7=0; //kr_blk mundur delay_ms(5);	12 V DC	belok_kanan
PORTD.0=0; //kn_dpn mundur PORTD.1=1; //kr_dpn maju PORTD.2=0; //kr_dpn mundur PORTD.3=0; //kn_dpn maju PORTD.4=1; //kr_blk maju	12 V DC	srg_kanan

PORTD.5=0;//kn_blk maju PORTD.6=0;//kn_blk mundur PORTD.7=0;//kr_blk mundur delay_ms(5);		
PORTD.0=0; //kn_dpn mundur PORTD.1=0; //kr_dpn maju PORTD.2=1; //kr_dpn mundur PORTD.3=1; //kn_dpn maju PORTD.4=0; //kr_blk maju PORTD.5=1;//kn_blk maju PORTD.6=0;//kn_blk mundur PORTD.7=1;//kr_blk mundur delay_ms(5);	12 V DC	belok_kiri
PORTD.0=0; //kn_dpn mundur PORTD.1=0; //kr_dpn maju PORTD.2=0; //kr_dpn mundur PORTD.3=1; //kn_dpn maju PORTD.4=0; //kr_blk maju PORTD.5=1;//kn_blk maju PORTD.6=0;//kn_blk mundur PORTD.7=0;//kr_blk mundur delay_ms(5);	12 V DC	srg_kiri

4.6 Pengujian Sensor SRF-04

Untuk menguji sensor srf-04, hal pertama yang dilakukan adalah pembuatan deklarasi pembacaan sensor tersebut. Adapun hasil pengujian dari sensor srf-04 dapat dilihat pada tabel 3:

Tabel 3. Pengujian Sensor SRF-04

Sensor yang diuji	Prosedur pengujian	Masukan	Keluaran yang diharapkan	Hasil yang didapat	Kesimpulan
Sensor Depan	- Aktifkan Mikrokontroler - Aktifkan deklarasi sensor depan (s_depan)	- Berikan penghalang pada sensor	Tampilan jarak antara sensor dan penghalang	Tampilan jarak antara sensor dan penghalang	Baik
Sensor Serong kiri	- Aktifkan Mikrokontroler - Aktifkan deklarasi sensor serong kiri (s_skiri)	- Berikan penghalang pada sensor	Tampilan jarak antara sensor dan penghalang	Tampilan jarak antara sensor dan penghalang	Baik
Sensor Kiri1	- Aktifkan Mikrokontroler - Aktifkan deklarasi sensor kiri1 (s_kiri1)	- Berikan penghalang pada sensor	Tampilan jarak antara sensor dan penghalang	Tampilan jarak antara sensor dan penghalang	Baik
Sensor Kiri2	- Aktifkan Mikrokontroler - Aktifkan deklarasi sensor kiri2 (s_kiri2)	- Berikan penghalang pada sensor	Tampilan jarak antara sensor dan penghalang	Tampilan jarak antara sensor dan penghalang	Baik
Sensor belakang	- Aktifkan Mikrokontroler - Aktifkan deklarasi	- Berikan penghalang pada sensor	Tampilan jarak antara sensor dan	Tampilan jarak antara sensor dan penghalang	Baik

	sensor belakang (s_blk)		penghalang		
Sensor Serong kanan	- Aktifkan Mikrokontroler - Aktifkan deklarasi sensor serong kanan (s_skanan)	- Berikan penghalang pada sensor	Tampilan jarak antara sensor dan penghalang	Tampilan jarak antara sensor dan penghalang	Baik
Sensor Kanan1	- Aktifkan Mikrokontroler - Aktifkan deklarasi sensor kanan1 (s_kanan1)	- Berikan penghalang pada sensor	Tampilan jarak antara sensor dan penghalang	Tampilan jarak antara sensor dan penghalang	Baik
Sensor Kanan2	- Aktifkan Mikrokontroler - Aktifkan deklarasi sensor kanan2 (s_kanan2)	- Berikan penghalang pada sensor	Tampilan jarak antara sensor dan penghalang	Tampilan jarak antara sensor dan penghalang	Baik

4.7 Nilai Input Output Fuzzy Kontrol Pada Robot

Setelah diketahui posisi-posisi fuzzy dalam ruangan dan sensor sudah dapat membaca jarak dengan baik. Selanjutnya adalah menentukan nilai input yang didapat dari masing-masing posisi tersebut. Adapun nilai input yang ditentukan dari masing-masing posisi fuzzy dapat dilihat pada tabel 4:

Tabel 4. Nilai Input Fuzzy Pada Robot

Posisi Fuzzy	Input							
	Depan	Srg kanan	Srg kiri	kanan 1	kanan 2	Kiri 1	Kiri 2	Belakang
F1	< 8 && > 2	< 13 && > 2	< 13 && > 2	< 36 && > 30	< 36 && > 30	< 6 && > 2	< 6 && > 2	< 51 && > 41
F2	< 32 && > 23	< 37 && > 26	< 9 && > 3	< 54 && > 48	< 54 && > 48	< 6 && > 2	< 6 && > 2	< 11 && > 2
F3	< 10 && > 3	< 12 && > 2	< 8 && > 2	< 52 && > 49	< 52 && > 49	< 5 && > 2	< 5 && > 2	< 32 && > 22
F4	< 51 && > 45	< 54 && > 46	< 9 && > 3	< 35 && > 29	< 35 && > 29	< 7 && > 2	< 7 && > 2	< 8 && > 2
F5	< 34 && > 28	< 38 && > 26	< 9 && > 3	> 70	> 70	< 5 && > 2	< 5 && > 2	< 30 && > 17
F6	< 8 && > 2	< 11 && > 8	< 8 && > 3	> 70	> 70	< 4 && > 2	< 4 && > 2	< 51 && > 43
F7	> 70	> 70	< 9 && > 3	< 54 && > 49	< 54 && > 49	< 5 && > 2	< 5 && > 2	< 8 && > 2
F8	> 70	> 70	< 8 && > 3	< 52 && > 49	< 52 && > 49	< 4 && > 2	< 4 && > 2	< 32 && > 21

Dari tabel 5.4 dapat dilihat bahwa posisi F5,F6,F7, dan F8 robot sudah mengetahui posisi atau arah untuk jalan keluar dari ruangan. Setelah mengetahui nilai input fuzzy, maka selanjutnya menentukan output yang akan dilakukan oleh robot. Adapun output kontrol dapat dilihat pada tabel 5:

Tabel 5. *Ouput Fuzzy Kontrol Pada Robot*

Posisi Fuzzy	Output (Memanggil Fungsi)	Keterangan	Waktu
F1	kanan180(); berhenti(); maju(); delay_ms(700); berhenti(); kanan90(); berhenti(); goto Selesai;	Robot berbelok 180 derajat ke kanan, kemudian maju sampai di depan jalan keluar ruangan, kemudian berbelok 90 derajat ke kanan lalu maju untuk keluar ruangan.	4,5 Detik 4,1 Detik 4,2 Detik
F2	kanan90(); berhenti(); maju(); delay_ms(900); berhenti(); kanan90(); berhenti(); goto Selesai;	Robot berbelok 90 derajat ke kanan, kemudian maju sampai di depan jalan keluar ruangan, kemudian berbelok lagi 90 derajat ke kanan lalu maju untuk keluar ruangan.	3,7 Detik 3,7 Detik 3,7 Detik
F3	kanan90(); berhenti(); maju(); delay_ms(900); berhenti(); kanan90(); berhenti(); goto Selesai;	Robot berbelok 90 derajat ke kanan, kemudian maju sampai di depan jalan keluar ruangan, kemudian berbelok lagi 90 derajat ke kanan lalu maju untuk keluar ruangan.	4,3 Detik 4,3 Detik 4,3 Detik
F4	maju(); delay_ms(500); berhenti(); goto F5;	Robot maju sampai di posisi F5.	0,6 Detik 0,5 Detik 0,5 Detik
F5	maju(); delay_ms(300); berhenti(); kanan90(); berhenti(); goto Selesai;	Robot maju sampai di depan jalan keluar ruangan, kemudian berbelok 90 derajat ke kanan lalu maju untuk keluar ruangan.	3,5 Detik 3,3 Detik 3,2 Detik
F6	kanan90(); berhenti(); goto Selesai;	Robot langsung berbelok 90 derajat ke kanan lalu maju untuk keluar ruangan.	2,2 Detik 2,3 Detik 2,2 Detik
F7	goto Selesai;	Robot langsung maju untuk keluar ruangan.	1,0 Detik 1,1 Detik 0,9 Detik
F8	goto Selesai;	Robot langsung maju untuk keluar ruangan.	0,4 Detik 0,3 Detik 0,4 Detik

4.8 Fuzzy If-Then

Aturan Fuzzy If-Then atau disebut juga aturan fuzzy atau pernyataan kondisional Fuzzy adalah aturan yang digunakan untuk merumuskan relasi kondisi antara dua atau lebih himpunan fuzzy. Adapun Fuzzy If-Then pada penelitian ini dapat dilihat pada tabel 6:

Tabel 6. Fuzzy If-Then

No	Aturan Fuzzy If-Then	Posisi Fuzzy
1	if (jrk_dpn1>2 && jrk_dpn1<8 && jrk_skanan>2 && jrk_skanan<13 && jrk_skiri>2 && jrk_skiri<13 && jrk_kn1>30 && jrk_kn1<36 && jrk_kn2>30 && jrk_kn2<36 && jrk_kiri1>2 && jrk_kiri1<6 && jrk_kiri2>2 && jrk_kiri2<6 && jrk_blk>41 && jrk_blk<51)	F1
2	if (jrk_dpn1>23 && jrk_dpn1<32 && jrk_skanan>26 && jrk_skanan<37 && jrk_skiri>3 && jrk_skiri<9 && jrk_kn1>48 && jrk_kn1<54 && jrk_kn2>48 && jrk_kn2<54 && jrk_kiri1>2 && jrk_kiri1<6 && jrk_kiri2>2 && jrk_kiri2<6 && jrk_blk>2 && jrk_blk<11)	F2
3	if (jrk_dpn1>3 && jrk_dpn1<10 && jrk_skanan>2 && jrk_skanan<12 && jrk_skiri>2 && jrk_skiri<8 && jrk_kn1>49 && jrk_kn1<52 && jrk_kn2>49 && jrk_kn2<52 && jrk_kiri1>2 && jrk_kiri1<5 && jrk_kiri2>2 && jrk_kiri2<5 && jrk_blk>22 && jrk_blk<32)	F3
4	if (jrk_dpn1>45 && jrk_dpn1<51 && jrk_skanan>46 && jrk_skanan<54 && jrk_skiri>3 && jrk_skiri<9 && jrk_kn1>29 && jrk_kn1<35 && jrk_kn2>29 && jrk_kn2<35 && jrk_kiri1>2 && jrk_kiri1<7 && jrk_kiri2>2 && jrk_kiri2<7 && jrk_blk>2 && jrk_blk<8)	F4
5	if (jrk_dpn1>28 && jrk_dpn1<34 && jrk_skanan>26 && jrk_skanan<38 && jrk_skiri>3 && jrk_skiri<9 && jrk_kn1>70 && jrk_kn2>70 && jrk_kiri1>2 && jrk_kiri1<5 && jrk_kiri2>2 && jrk_kiri2<5 && jrk_blk>17 && jrk_blk<30)	F5
6	if (jrk_dpn1>2 && jrk_dpn1<8 && jrk_skanan>8 && jrk_skanan<11 && jrk_skiri>3 && jrk_skiri<8 && jrk_kn1>70 && jrk_kn2>70 && jrk_kiri1>2 && jrk_kiri1<4 && jrk_kiri2>2 && jrk_kiri2<4 && jrk_blk>43 && jrk_blk<51)	F6
7	if (jrk_dpn1>70 && jrk_skanan>70 && jrk_skiri>3 && jrk_skiri<9 && jrk_kn1>49 && jrk_kn1<54 && jrk_kn2>49 && jrk_kn2<54 && jrk_kiri1>2 && jrk_kiri1<5 && jrk_kiri2>2 && jrk_kiri2<5 && jrk_blk>2 && jrk_blk<8)	F7
8	if (jrk_dpn1>70 && jrk_skanan>70 && jrk_skiri>3 && jrk_skiri<8 && jrk_kn1>49 && jrk_kn1<52 && jrk_kn2>49 && jrk_kn2<52 && jrk_kiri1>2 && jrk_kiri1<4 && jrk_kiri2>2 && jrk_kiri2<4 && jrk_blk>21 && jrk_blk<32)	F8

4.9 Analisis Hasil Pencapaian Sistem Pada Robot

Setelah selesai melakukan pengujian, adapun hasil analisa yang dicapai sistem ini adalah sebagai berikut:

1. Beberapa gerakan yang dapat dilakukan oleh robot adalah gerak maju, gerak mundur, gerak berbelok kekanan maupun kekiri, dan berhenti.
2. Robot yang dibuat mempunyai delapan sensor untuk mendeteksi penghalang dan posisi sensor diletakkan di sekeliling sisi robot.
3. Robot yang dibuat memiliki sistem yang baik dan dapat mengikuti instruksi sesuai dengan yang telah diprogram.
4. Robot ini dapat mengetahui dengan cepat dimana posisi jalan keluar dari sebuah ruangan yang telah ditentukan.

5. Penutup

5.1 Kesimpulan

1. Dari analisis yang dilakukan pada penelitian sebelumnya, diketahui permasalahan yang ada adalah robot akan memeriksa semua ruangan sampai menemukan jalan keluar. Hal ini membutuhkan waktu yang lebih lama dibandingkan jika robot langsung mengetahui dimana posisi jalan keluar dan langsung menuju ke arah jalan keluar tersebut.
2. Penelitian ini menghasilkan sebuah robot yang menggunakan 4 buah roda sebagai penggerak dan menggunakan 8 buah sensor jarak untuk mendeteksi penghalang. Semua sistem dapat bekerja sesuai dengan perintah atau instruksi dari mikrokontroler.
3. Penelitian ini menghasilkan sistem yang baik menggunakan logika fuzzy, sehingga robot dapat langsung mengetahui posisi arah jalan keluar dari suatu ruangan.

5.2 Saran

1. Diharapkan agar posisi fuzzy dapat ditambah lebih banyak lagi sehingga robot lebih cepat mendapatkan titik keputusan untuk keluar dari ruangan.
2. Diharapkan agar logika fuzzy ini dapat diimplementasikan dengan baik pada robot pada saat Kontes Robot Indonesia khususnya kategori Kontes Robot Pemadam Api Indonesia (KRPAI).

6. Daftar Pustaka

- [1] Helfi Nasution. (2012). Implementasi Logika Fuzzy pada Sistem Kecerdasan Buatan. *Jurnal ELKHA Vol.4*, 4-7.
- [2] Fadlisyah M, Sayuti. (2009). *Robot Visi*. Yogyakarta: Graha Ilmu.
- [3] Taufiq Dwi Septian Suyadhi. (2010). *Buku Pintar Robotika Bagaimana Merancang dan Membuat Robot Sendiri*. Yogyakarta: ANDI Yogyakarta.
- [4] Winarno, Deni Arifianto. (2011). *Bikin Robot Itu Gampang*. Surabaya: PT. Kawan Pustaka.
- [5] Gilang Nugraha Putu Pratama, Andi Dharmawan, Catur Atmaji. (2014). Implementasi Kendali Logika Fuzzy pada Robot Line Follower. *IJEIS, Vol.4, No.1*, 45-56.
- [6] Faikul Umam. (2013). Pengembangan Sistem Kendali Pergerakan Autonomous Mobile Robot Untuk Mendapatkan Jalur Bebas Hambatan Menggunakan Fuzzy Logic Controller. *Jurnal Ilmiah Mikrotek Vol.1, No.1*, 35-42.
- [7] Ardi Winoto. (2008). *Mikrokontroler AVR Atmega8/16/8535 dan Pemrogramannya dengan Bahasa C pada WinAVR*. Bandung: Informatika.
- [8] Hendriono. (2011). *Pengenalan Mikroprosesor dan Bahasa C (Codevision AVR)*. Yogyakarta: Graha Ilmu.
- [9] Widodo Budiharto. (2010). *Belajar Sendiri Membuat Robot Cerdas*. Jakarta: Elex Media Komputindo.
- [10] Maulidi Rahman, Hugo Aprilianto. (2016). Penerapan Metode Fuzzy Pada Robot Beroda Menggunakan Omni-Directional Wheels. *Jutisi Vol 5 (ISSN: 2089-3787)*, 1075 –1082.