

Rancang bangun sistem robot AGV untuk penyortiran paket ekspedisi dengan fitur *anti collision*

Mohamad Nasyir Tamara¹, Eko Budi Utomo², Ni'am Tamami³, Endra Pitowarno⁴, Novian Fajar Satria⁵, Didik Setyo Purnomo⁶, Wildan Hilmi⁷, Cahyo Sugianto⁸

e-mail: nasir_meka@pens.ac.id, ekobudi_u@pens.ac.id, niam@pens.ac.id, epit@pens.ac.id, didiksp@pens.ac.id, ovinmeka@pens.ac.id, cahyocg@gmail.com, singkungblog@gmail.com

^{1,2,3,4,5,6}Program Studi Teknik Mekatronika, Politeknik Elektronika Negeri Surabaya

^{7,8}PT Nasta Mekatronika Indonesia

Informasi Artikel

Riwayat Artikel

Diterima 25 September 2022

Direvisi 20 Oktober 2022

Diterbitkan 28 Oktober 2022

Kata kunci:

AGV,
Manajemen lalu lintas,
Infrared,
Node Conflict,
Opposite conflict

Keywords:

AGV,
Traffic management,
Infrared,
Node Conflict,
Opposite conflict

ABSTRAK

Robot Automated Guided Vehicle (AGV) secara luas diterapkan di berbagai tempat seperti pabrikasi dan manufaktur yang telah menggunakan sistem otomatis, khususnya pada area pergudangan. Ferlib AGV merupakan robot pembawa barang yang didesain khusus untuk membawa dan memindah barang. Robot Ferlib AGV diciptakan untuk memenuhi kebutuhan para pekerja dalam sistem tata kelola pergudangan di industri terutama pekerjaan yang bergerak dalam bidang ekspedisi pengiriman barang. Pada saat beberapa robot berjalan di ruang atau daerah kerja yang sama, terdapat masalah yang mungkin terjadi dalam manajemen lalu lintas salah satunya adalah tabrakan. Penelitian ini bertujuan untuk membuat sebuah sistem yang dapat membuat robot menuju ke posisi yang diinginkan dan sebuah sistem yang dapat menghindari terjadinya tabrakan untuk diimplementasikan pada robot Ferlib AGV sebagai robot pembawa barang. Ada dua jenis konflik yang dapat menyebabkan terjadinya tabrakan yaitu Node Conflict dan Opposite Conflict. Robot mendeteksi konflik menggunakan sensor Position Sensitive Device (PSD) jenis inframerah. Untuk menyelesaikan konflik antar robot dimulai dari perencanaan jalur yang sesuai, dan dasar efektivitas tugas penjadwalan pergerakan robot. Hasil dari sistem ini telah mampu membuat robot Ferlib AGV sebagai robot pembawa barang yang dapat membawa barang menuju posisi koordinat berdasarkan posisi yang diberikan, serta dapat mendeteksi konflik dan melakukan penghindaran berdasarkan jenis konflik yang terjadi.

ABSTRACT

Automated Guided Vehicle System (AGVs) robots are widely applied in various places such as manufacturing and manufacturing that have used automated systems, especially in warehousing areas. Ferlib AGV is a freight robot specifically designed to carry and move goods. The Ferlib AGV robot was created to meet the needs of workers in the warehousing governance system in the industry, especially jobs engaged in the transportation and transfer of goods. When several robots run in the same work space or area, there are problems that may occur in traffic management, one of which is a collision. This study aims to create a system that can make the robot go to the desired position and a system that can avoid collisions to be implemented on the Ferlib AGV robot as a freight robot. There are two types of conflicts that can cause collisions, namely Node Conflict and Opposite Conflict. Robots detect conflicts using an infrared sensor type Position Sensitive Device (PSD). To resolve conflicts between robots starts from the appropriate path planning, and the basic effectiveness of the task of scheduling robot movements. The results of this system have been able to make Ferlib AGV robots as goods-carrying robots that can carry goods to coordinate positions based on the given position, and can detect conflicts and avoidance based on the type of conflict that occurs.

Penulis Korespondensi:

Mohamad Nasyir Tamara
Jurusan Teknik mekatronika,
Politeknik Elektronika Negeri Surabaya,
Jl. Raya ITS, Kampus PENS, Keputih, Sukolilo, surabaya. 60111.
Email: nasir_meka@pens.ac.id

1. PENDAHULUAN

Salah satu faktor yang mempunyai peranan penting dalam perkembangan teknologi saat ini adalah teknologi dalam bidang robotika. Robot dapat membantu menyelesaikan pekerjaan manusia dalam banyak hal, khususnya pada pekerjaan dengan tingkat ketelitian yang tinggi serta berisiko besar menyebabkan terjadinya kecelakaan pada tubuh manusia [1]. Robot sangat berperan untuk membantu meringankan pekerjaan manusia atau bahkan mampu untuk menggantikan pekerjaan manusia. Automated Guided Vehicle (AGV) adalah sebuah mobile robot yang mampu bergerak atau melakukan pekerjaan tertentu secara mandiri. AGV dapat diprogram sehingga bekerja secara otomatis dan digunakan sebagai perangkat manipulasi [2]. Dalam pengaplikasiannya biasanya dioperasikan di zona yang telah ditentukan seperti gudang, pabrik industri atau sejenisnya [3]. Sebuah sistem AGV umum pada dasarnya terdiri dari peripheral kendaraan pada komponen sistem control stasioner [4]. Berikut ini merupakan komponen utama pada sistem AGV. Teknik ini berdasarkan pada identifikasi dan pengenalan terhadap fitur-fitur dari suatu objek di lingkungan yang mungkin diketahui atau direkam secara dinamis (line follower robot). Robot yang beroperasi sendiri dengan mendeteksi garis yang ditentukan sebelumnya yang digambar pada lantai [4]. Untuk melacak garis robot menggunakan sensor array yang mengirimkan sinyal untuk mengendalikan sistem robot agar tetap berada pada jalur [5]. Navigasi AGV menggunakan kamera sebagai sensor eksternal untuk menangkap informasi dari ruang kerja melalui pengukuran kedekatan dan citra visual pada robot. Pada penelitian [6], AGV rentan terhadap kesalahan akibat pantulan cahaya terutama dari perubahan kondisi matahari di koridor berjendela. Telah mendiskusikan desain robot bergerak untuk lingkungan yang dinamis. Sebuah desain fuzzy brain untuk robot otomatis dan sistem sensor telah digunakan untuk mendeteksi hambatan di gudang atau daerah kerja. Untuk beroperasi secara efektif dan efisien serta meningkatkan produktivitas robot AGV, kendaraan harus dikelola dengan baik. Tugas pengiriman harus dialokasikan untuk meminimalkan waktu tunggu di tempat bongkar muat. Kontrol lalu lintas menggunakan penginderaan dari robot.

2. STUDI LITERATUR

Robot pengikut garis (*line follower*) telah banyak dirancang dan digunakan baik sebagai edukasi atau diterapkan di sektor industri. Robot ini pada umumnya memiliki roda dua, dan biasanya menggunakan sensor inframerah atau *photodiode* sebagai pendeteksi garis [7]. Pada penelitian [8], masalah yang dihadapi oleh robot pengikut garis yang dilengkapi dengan warna RGB pada garis membuat sensor menjadi tidak fokus pada daerah tertentu. Kesulitan lain yang dihadapi oleh robot adalah perubahan intensitas cahaya secara kontinyu di lingkungan sekitarnya. Sensor sederhana yang dapat mendeteksi garis putih atau hitam dapat menggunakan sensor *photodiode*. Robot pengikut garis memiliki struktur sederhana dan sirkuit elektronik. Sensor dapat dibangun dengan menggunakan pasangan pemancar dan penerima, dalam hal ini dapat berupa sensor *photodiode* yang berperan sebagai *receiver* sedangkan LED sebagai pencahayaan yaitu *transmitter* [9]. Konsep robot pengikut garis cukup sederhana, jalur dirancang dengan warna gelap (hitam) sehingga akan menyerap gelombang dari IR atau gelombang cahaya. Permukaan yang mengelilingi jalan akan dibangun dengan menggunakan warna terang (putih) yang akan mencerminkan gelombang IR atau gelombang cahaya. Selanjutnya dibutuhkan dua atau tiga sensor yang digunakan sebagai aksi untuk mempermudah penentuan aksi (belok kiri, belok kanan). Output dari sensor akan diumpankan ke unit pengontrol sehingga akan diproses untuk pengambilan keputusan yang sesuai [10]. Menurut penelitian [11], menghindari hambatan (*obstacle*) secara otomatis merupakan tujuan dari pergerakan *mobile robot*. Mengetahui informasi lingkungan dengan bantuan sensor, menjadi aspek yang sangat penting. Sensor yang dapat digunakan yaitu sensor jarak. Karakteristik sensor masih rentan terhadap gangguan atau *noise*. Sensor ultrasonik merupakan salah satu sensor jarak yang sangat digemari, karena memiliki akurasi yang cukup kuat dalam mendeteksi *obstacle*, sensor ini mampu memberikan pengetahuan lingkungan yang cukup jelas pada robot [12]. Pada penelitian [13], tahap pertama pada metode *Landmarc* yaitu dilakukan pengambilan data terhadap RSSI dari tag referensi (θ_j) sebagai data yang telah diketahui posisinya (data *training*), setelah itu dilakukan pengambilan data RSSI dari tag diuji (S_j) sebagai data yang ingin diketahui posisinya (data *testing*). Kemudian dilakukan perhitungan *Euclidean distance* menggunakan persamaan (1).

$$E_i = \sqrt{\sum_{j=1}^n (\theta_j - S_j)^2} \quad (1)$$

Penerapan *manhattan distance* pada metode *Landmarc* diterapkan dengan cara melakukan penggantian pada persamaan *Euclidean distance* pada persamaan (1) menjadi persamaan *manhattan distance* yaitu persamaan (2).

$$E_i = \sum_{j=1}^n |\theta_j - S_j| \quad (2)$$

Pada penelitian [14], Prinsip algoritma A* adalah mencari jalur terpendek dari sebuah titik awal menuju titik akhir dengan memperhatikan harga F terkecil (F function heuristic). Algoritma ini memperhitungkan nilai dari *current state* ke tujuan dengan fungsi heuristik, dan juga mempertimbangkan nilai yang telah ditempuh selama ini dari *initial state* ke *current state*. Jadi jika ada jalan yang telah ditempuh sudah terlalu panjang dan ada jalan lain yang nilainya lebih kecil tetapi memberikan posisi yang sama dilihat dari goal, jalan yang lebih pendek yang akan dipilih.

$$f(x) = g(x) + h(x) \quad (3)$$

dimana:

$g(x)$ adalah jarak total dari posisi asal ke lokasi sekarang.

$h(x)$ adalah fungsi heuristik yang digunakan untuk memperkirakan jarak dari lokasi sekarang ke lokasi tujuan

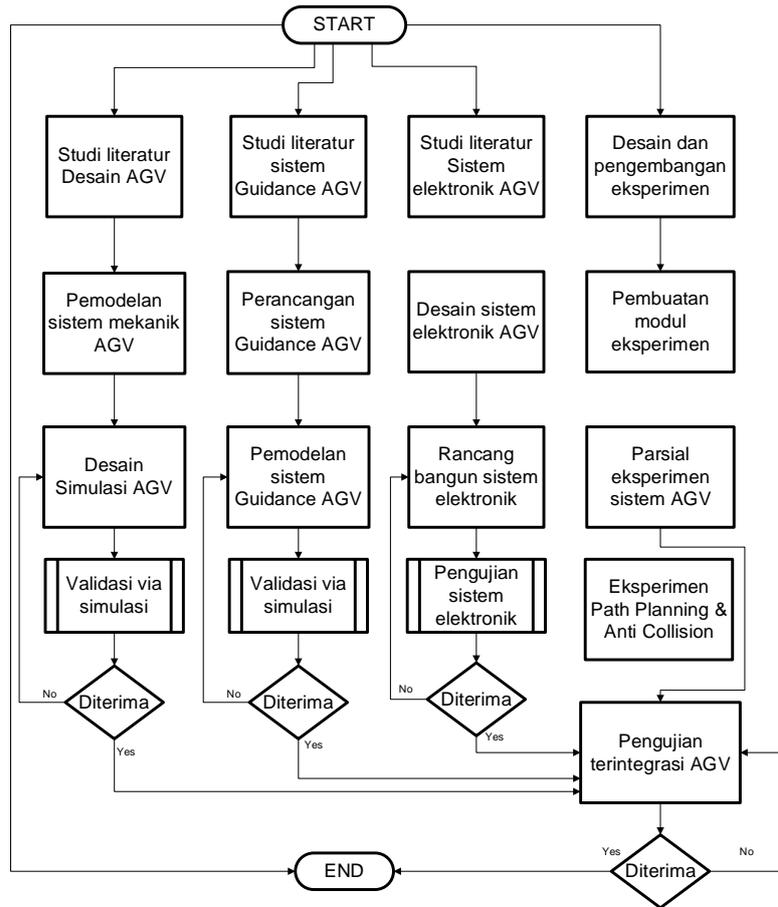
Fungsi ini jelas berbeda karena ini adalah perkiraan semata dibandingkan dengan nilai aslinya. Semakin tinggi keakuratan heuristik, semakin cepat dan bagus lokasi tujuan ditemukan dan dengan tingkat keakuratan yang lebih baik. Algoritma A* yang secara dasar digunakan adalah sebagai berikut [15]

1. Masukkan node awal ke openlist
2. Loop langkah-langkah dibawah ini:
 - a. Cari node (n) dengan nilai f(n) yang paling kecil dalam open list, dan node ini sekarang menjadi current node.
 - b. Keluarkan current node dari openlist dan masukkan ke open list
 - c. Untuk setiap tetangga dari current node lakukan berikut
 - 1) Jika tidak dapat dilalui atau sudah ada dalam close list, abaikan
 - 2) Jika belum ada di open list buat current node parent dari node tetangga ini, simpan nilai f,g, dan h dari node ini.
 - 3) Jika sudah ada di openlist cek apabila node tetangga ini lebih baik menggunakan nilai g sebagai ukuran. Jika lebih baik ganti parent dari node ini di openlist menjadi current node, lalu kalkulasikan ulang nilai g dan h dari node ini.
 - d. Hentikan looping jika:
 - 1) Node tujuan telah ditambah ke openlist yang berarti rute ditemukan
 - 2) Belum menemukan node akhir (tujuan) sementara openlist kosong atau berarti tidak ada rute.

Pada [16] mengusulkan strategi penjadwalan AGV berbasis support vector machine (SVM) yang meningkatkan efisiensi penjadwalan AGV. Skema yang dikembangkan mengoptimalkan proses pembagian area tugas untuk memberi AGV kemampuan menghindari rintangan di lingkungan dinamis yang kompleks. Secara khusus, mengingat dua kasus gerak AGV, yaitu menuju satu titik target dan beberapa titik target, jalur optimal ditentukan dengan menggunakan metode exhaustive dan Q-learning, sementara optimasi jalur diwujudkan dengan memanfaatkan skema yang berbeda. Berdasarkan jalur terpendek yang diperoleh, model pemrograman nonlinier dengan waktu terpendek sebagai tujuan dibangun, dan jalur balik AGV terbukti optimal dengan non-dominated sorting genetic algorithm (NSGA-II).

3. METODOLOGI PENELITIAN

Robot AGV dirancang untuk memiliki kemampuan untuk bergerak secara otomatis. Dengan mempunyai fungsi utama yaitu untuk mengangkut dan memindahkan barang dari satu lokasi ke lokasi lainnya, yang beroperasi pada kondisi permukaan yang datar dan tidak bergelombang seperti beton/lantai. Robot AGV menggunakan garis sebagai pemandu dalam bergerak pada saat memindahkan barang, garis atau jalur robot didesain khusus sesuai dengan kebutuhan robot dalam industri tersebut.



Gambar 1. Metodologi Penelitian

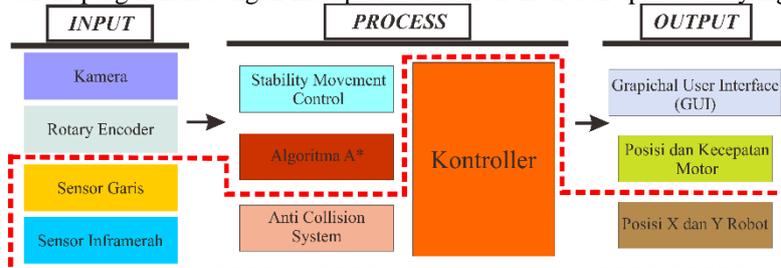
Sebagaimana ditunjukkan pada Gambar 1 dalam penelitian ini perancangan robot AGV anti collision meliputi perancangan bodi robot, mekanisme pergerakan robot dan juga perancangan sistem agar robot dapat melakukan metode yang di inginkan, yaitu robot dapat berjalan sesuai dengan alur tanpa adanya tabrakan dengan obstacle maupun dengan robot AGV lain.

4. HASIL DAN PEMBAHASAN

Ferlib AGV (*Automated Guided Vehicle*) merupakan robot yang dapat memindahkan barang dari satu lokasi ke lokasi yang lain. Nama Ferlib sendiri diambil dari penggabungan kata dari *Differential*, *Lifter* dan *Barcode* yang artinya robot memiliki pergerakan motor secara *differential drive*, dan memiliki mekanisme *lifter* sebagai media untuk membawa dan menaruh barang, serta dapat membaca *barcode* yang terdapat di lintasan.

4.1. Desain sistem robot

Perancangan sistem dalam pembuatan Ferlib AGV memperhatikan berbagai macam aspek dalam referensi teori yang digunakan dan pengetahuan yang mendasari bagaimana sistem dari Ferlib AGV. Sistem yang dirancang terdiri dari sistem pergerakan, sensor-sensor, dan komunikasi data. Kontrol pergerakan robot sangat diperlukan dalam pergerakan dengan memperhatikan fitur-fitur dan spesifikasi yang diberikan.

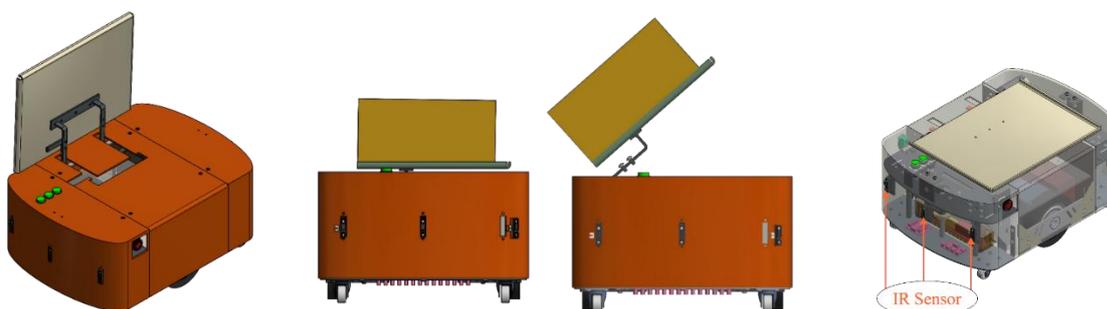


Gambar 2. Diagram perencanaan sistem pada robot

Gambar 2 menggambarkan tentang perencanaan perangkat keras dan sistem yang digunakan pada robot. Dilihat dari diagram sistem pada robot bahwa terdapat tiga kombinasi sistem yang akan dipadukan agar robot dapat bekerja dengan maksimal diantaranya sistem kontrol kestabilan dalam pergerakan robot yaitu sistem kontrol yang membuat robot tetap stabil ketika membawa ataupun tidak membawa barang, kemudian algoritma A* (*A Star*) digunakan agar robot dapat memiliki kemampuan untuk mencari jalur tercepat ketika robot akan menuju ke posisi tujuan, dan sistem *anti collision* atau sistem untuk menghindari tabrakan yang digunakan pada saat robot sedang bergerak menuju posisi tujuan. Berdasarkan Gambar 2, penelitian ini berfokus kepada sistem agar robot dapat menghindari terjadinya tabrakan yang tidak terprediksi ketika robot sedang menuju ke posisi tujuan yaitu koordinat x dan y.

4.2. Desain mekanik dan elektronik

Desain mekanik dari FERLIB AGV dibuat berdasarkan referensi AGV pada umumnya yang digunakan di industri *warehouse e-commerce*. Telah dilakukan penyesuaian tersendiri dengan memperhatikan bentuk yang diinginkan, komponen elektronik yang akan digunakan, proses pembuatan, bagaimana metode yang akan digunakan dalam pembuatan, menggunakan las dan metode manufaktur lainnya seperti mesin yang akan digunakan dengan mesin bubut, dan *milling*. Terdapat mekanisme mekanik dalam pekerjaan yang diberikan pada fitur-fitur robotnya seperti *Flipping mechanism* untuk menjatuhkan barang yang telah sampai pada titik tujuan. Dan mekanisme pengatur ketinggian roda kaster untuk menghindari terjadinya selip roda drive dengan lantai. Desain dibuat dengan aplikasi *Autodesk Inventor*. Desain rangka robot terbuat dari beberapa *part* yang terbuat dari plat 3mm dan 4mm untuk *main base* bawah. Dengan membuat *part* berukuran kecil-kecil yang perlu dilas hingga menjadi satu kesatuan rangka robot untuk menopang semua komponen mekanik lainnya dan elektronik. *Sorting AGV* pada umumnya juga memiliki 4 roda *caster* untuk menahan keseimbangan robot dan 2 roda drive sebagai roda penggerak utama dari motor. Sistem keseimbangan pada robot tersebut diberikan spring untuk memberikan suspensi pada pergerakan lantai robot yang tidak rata. Sedangkan robot sendiri didesain untuk penggunaan *indoor* dengan permukaan lantai yang halus dan rata tanpa ada kemiringan sudut lantai.

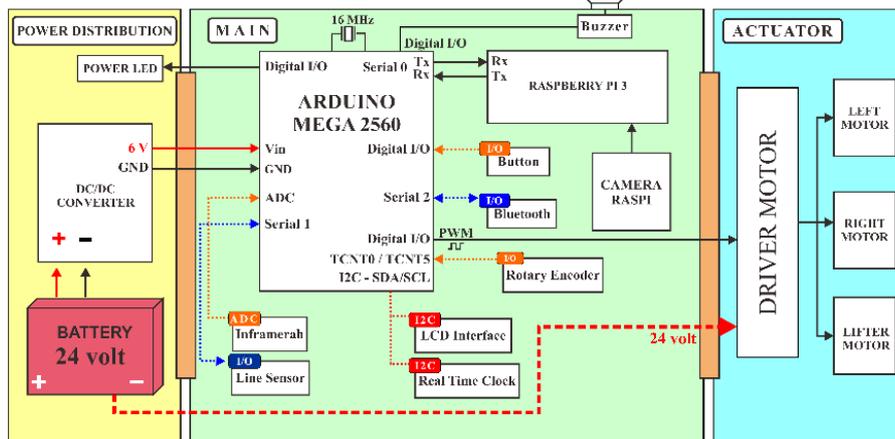


Gambar 3. Desain Mekanik Ferlib AGV dan sensor Anti Collision

FERLIB AGV menerapkan sistem pergerakan *differential drive* dimana 2 roda pada posisi kanan dan kiri yang akan menentukan pergerakannya. Roda berdiameter 109 mm Kecepatan dan arah putar motor yang akan menentukan untuk berbelok kekanan, ke kiri ataupun berputar. Ketika berbelok FERLIB AGV akan berhenti pada barcode lantai tempat berbelok kemudian berputar sesuai jalurnya untuk berbelok. Tidak hanya roda drive namun juga terdapat 4 roda bebas yang diletakkan 2 depan dan 2 belakang yang berfungsi untuk menjaga keseimbangan FERLIB AGV serta membantu dalam bergerak dan berbelok. Ditengah tampak atas terlihat berbentuk persegi panjang sebagai alas sistem *lifter* sebagai lokasi penerima beban utama muatan rak yang diangkat terbuat dari akrilik. *Cover robot* akan dibuat dari bahan *fiberglass* dan resin akan tampak seperti pada gambar di atas. Lengkungan pada bentuk depan robot harus pasti dikarenakan posisi sudut sensor yang akan diterapkan pada sistem anti tabrakan harus tepat. Karena robot akan mendeteksi jika terjadi halangan di depan. Dan akan bertindak sesuai dengan algoritma yang diberikan (tergantung kondisi). Dan juga tata letak komponen seperti *emergency button*, *pushbutton* dan *LCD text*.

Dalam perancangan sistem elektronik secara umum digunakan kontroler yang bersifat sebagai *master* dan beberapa kontroler berperan sebagai *slave*. Kontroler yang berperan sebagai *master* berfungsi menjalankan seluruh sistem kontrol dan proses pengambilan keputusan, sedangkan kontroler sebagai *slave* bertujuan untuk menjangkau aktuator dan mengambil data dari sensor-sensor yang digunakan. Untuk memproses data dari kedua peran kontroler tersebut dibutuhkan proses komunikasi antara *master* dan *slave*, metode yang digunakan dalam hal ini adalah metode *I2C (Inter-Integrated Circuit)* yang secara umum dikenal sebagai pengiriman data

secara paralel, dan metode serial dengan menggunakan pin *Rx* dan *Tx* dari masing-masing kontroler *master* dan *slave*. Blok diagram sistem elektronik Ferlib AGV dapat dilihat pada Gambar 5.

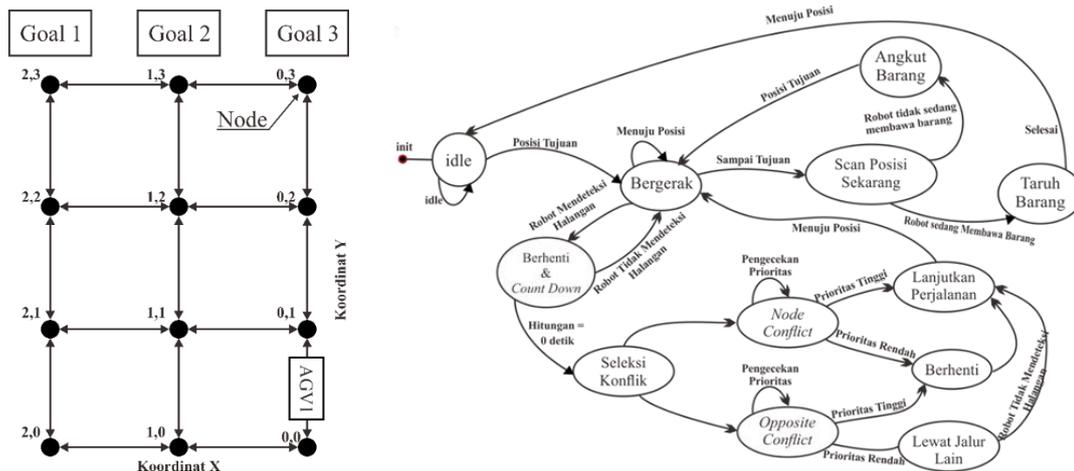


Gambar 4. Desain Sistem Elektronik Ferlib AGV

Tampak pada Gambar 4, terlihat hubungan antara perangkat keras yang digunakan baik yang berperan sebagai *master* maupun yang berperan sebagai *slave*. Berdasarkan gambar di atas dapat dilihat bahwa yang bertindak atau berperan sebagai *master* pada sistem elektronik robot Ferlib adalah Arduino Mega 2560, sedangkan yang bertindak sebagai *slave* adalah kontroler Raspberry Pi 3 Model B. Untuk pengiriman data menggunakan komunikasi via serial. Raspberry Pi 3 model B terhubung dengan koneksi pada *server* dan kamera yang digunakan untuk membaca *barcode*. Terdapat juga *slave* lainnya seperti LCD sebagai antarmuka operator dengan robot, *Bluetooth* sebagai media koneksi untuk menghubungkan antara *remote* kontrol dengan robot, dan *Real Time Clock* digunakan untuk memberikan informasi waktu.

4.3. Jalur robot dan alur system kerja/pergerakan robot

Jalur robot dirancang sesuai dengan kebutuhan robot. Jalur yang dirancang untuk robot berbentuk kotak persegi dan memiliki perempatan. Jalur dibuat berbentuk persegi dengan tujuan agar robot dapat mencari jalan tercepatnya dalam memindah barang. Berikut ini merupakan bentuk dari jalur yang digunakan pada robot dan metode mencari jalur tercepat menggunakan metode *manhattan distance*. Berikut merupakan perencanaan jalur robot yang ditunjukkan pada Gambar 5.



Gambar 5. Perencanaan lintasan robot dan *Finite State Machine* sistem kerja robot

Robot akan bergerak menuju ke arah tujuan tertentu hingga robot berada pada posisi tujuan. Robot akan mendeteksi apabila ada halangan ketika bergerak menuju posisi tujuan dan robot akan bertransisi ke *state* berhenti dan menghitung mundur dengan menyesuaikan lama robot mendeteksi halangan (berhenti). Robot akan kembali bergerak ketika robot tidak mendeteksi halangan lagi dengan syarat hitungan mundur yang dilakukan oleh robot belum selesai. Ketika hitungan mundur selesai dan robot masih mendeteksi halangan yang

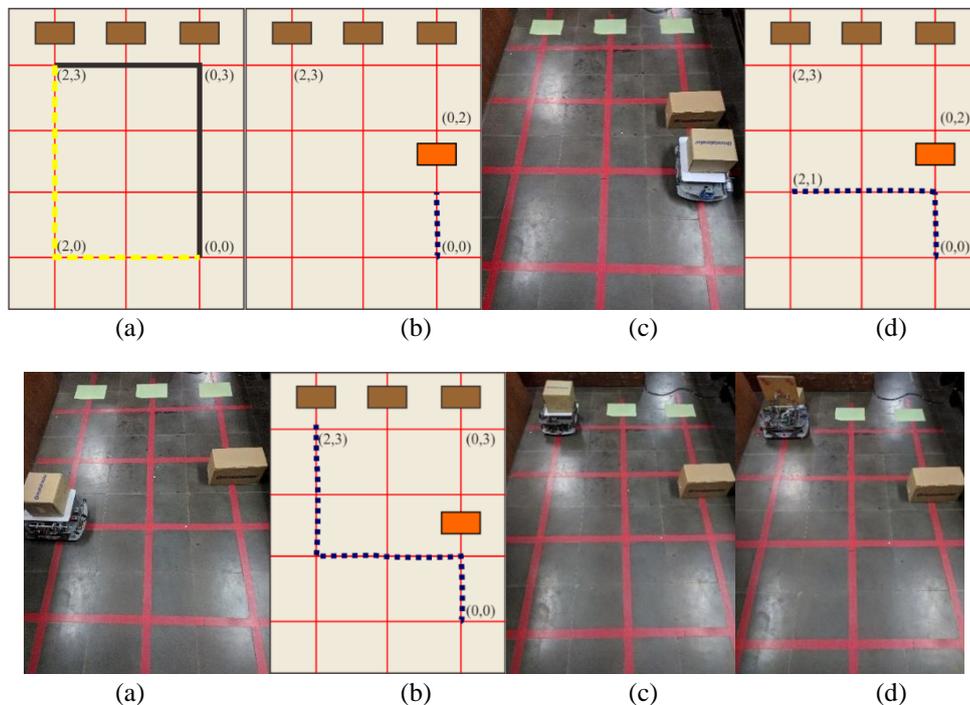
ada di depannya, maka robot akan berpindah ke keadaan berikutnya yaitu *state* lewat jalur lain yang menyesuaikan dengan jalur yang telah diberikan oleh GUI sebelumnya.

Ketika robot telah sampai pada posisi tujuan, maka robot akan bertransisi menuju *state* scan posisi. Apabila robot tidak sedang membawa barang maka robot akan bertransisi menuju *state* angkut barang dan robot akan mengangkut barang kemudian menuju *state* bergerak yaitu menuju lokasi atau posisi tujuan, apabila telah sampai pada posisi tujuan maka robot akan menuju *state* taruh barang. Pada *state* ini robot akan meletakkan barang yang telah diangkut kemudian robot akan kembali ke posisi awal (*start position*) dan bersiap untuk diinisialisasi.

4.4. Pengujian sistem anti collision

Pengujian konflik berhadapan atau *opposite conflict* dilakukan dengan memberi objek statis pada area kerja robot. Pengujian dimulai dengan memberi perintah pada robot untuk menuju posisi tujuan yaitu koordinat 2,3 lalu menaruh barang (Gambar 6.a). Robot mengambil jalur dengan cara *manhattan distance* yaitu robot bergerak dengan 2 tahap. Tahap pertama robot bergerak pada sumbu y yaitu menuju pada titik y3, kemudian bergerak lagi pada sumbu x menuju titik x2.

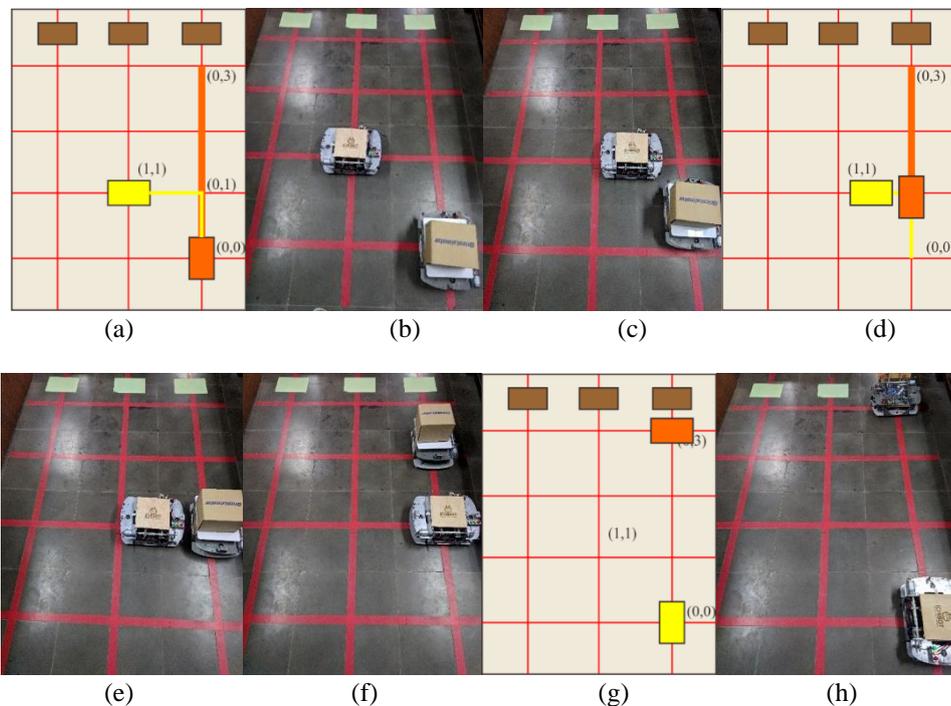
Berdasarkan Gambar 6.a, bahwa robot diperintah untuk menuju posisi koordinat 2,3 dan kemudian menaruh barang. Akan tetapi pada saat robot bergerak, diberikan halangan statis yang berupa objek (kardus) pada titik y2 (Gambar 6.b). Berdasarkan pada perancangan untuk menghindari tabrakan yaitu ketika robot mendeteksi sebuah halangan objek statis, maka robot akan menghitung mundur selama 5 detik (Gambar 6.c). Apabila setelah 5 detik halangan masih terdeteksi oleh robot, maka robot akan menggunakan jalur lain untuk menuju ke posisi tujuan yaitu pada koordinat 2,3. Gambar 6.d menunjukkan bahwa robot mendeteksi halangan statis (kardus). Setelah robot mengetahui jenis objek atau obstacle yang terdeteksi, maka robot akan menghitung selama 5 detik. Setelah hitungan selesai, Robot mencari jalur lain ketika halangan masih terdeteksi. Sebelum itu, robot terlebih dahulu berputar ke kiri 90 derajat kemudian robot menuju koordinat tujuan dengan menggunakan jalur alternatif karena jalur utama dihalangi oleh objek (Gambar 6.d dan 6.e). Setelah robot berada pada koordinat 2,1, robot melakukan rotasi 90 derajat ke kanan agar dapat bergerak menuju koordinat 2,3 (Gambar 6.f dan 6.g). Berdasarkan Gambar 6.h, robot telah sampai pada koordinat tujuan dan menaruh barang. Berdasarkan hasil percobaan untuk konflik berhadapan (*opposite conflict*) dalam menghindari halangan agar tidak terjadi tabrakan, dapat disimpulkan bahwa walaupun robot bergerak dengan tidak menggunakan jalur utama tetapi robot akan selalu bergerak hingga robot sampai pada posisi tujuan dan menaruh barang.



Gambar 6. Pengujian Sistem anti collision 1

Pengujian robot mendeteksi konflik titik atau node conflict dilakukan dengan menggunakan dua buah robot yang nantinya akan mendeteksi satu sama lain dan berjalan pada arah yang sama. Pengujian dimulai

dengan memberi perintah kepada kedua robot untuk menuju posisi tujuan. Untuk robot pertama, diperintah menuju ke tujuan pada koordinat 2,3 sedangkan untuk robot kedua diperintahkan untuk menuju ke tujuan pada koordinat posisi awal yaitu 0,0. Posisi awal dari masing-masing robot ditunjukkan pada Gambar 7.a. Terdapat dua robot yang ditunjukkan pada 7.a, robot pertama ditandai dengan kotak berwarna orange (robot 1) dan robot kedua ditandai dengan kotak berwarna kuning (robot 2). Pada pengujian konflik titik ini, masing-masing robot ditempatkan pada koordinat yang berbeda. Robot 1 ditempatkan pada koordinat 0,0 dan robot 2 ditempatkan pada koordinat 1,1. Kedua robot ditempatkan pada posisi tersebut untuk memungkinkan terjadinya tabrakan pada koordinat 0,1. Posisi awal untuk masing-masing robot dapat dilihat pada Gambar 7.b. Robot 1 sedang membawa barang untuk diletakkan pada posisi tujuan yaitu koordinat 0,3 sedangkan robot 2 pada kondisi ini tidak sedang membawa barang dan memiliki posisi tujuan pada koordinat 0,0 (posisi awal). Posisi robot ketika mendeteksi konflik di titik persimpangan ditunjukkan pada Gambar 7.c. Masing-masing robot akan berhenti ketika mendeteksi objek apapun yang menghalangi robot bergerak. Kondisi robot Gambar 7.c merupakan kondisi ketika robot mendeteksi objek (dalam hal ini robot lain). Pengecekan prioritas terjadi ketika kedua robot telah berhenti karena masing-masing dari robot telah mendeteksi satu sama lain. Berdasarkan perancangan robot yang sedang membawa barang memiliki prioritas paling tinggi dari robot yang sedang tidak membawa barang. Kondisi atau *state* yang akan dilakukan masing-masing robot dapat dilihat pada Gambar 7.d dan 7.e. Telah diketahui sebelumnya bahwa robot 1 sedang membawa barang dan robot 2 tidak sedang membawa barang. Oleh karena itu, robot 2 berhenti sedangkan robot 1 melanjutkan perjalanan menuju posisi tujuan dan tabrakan tidak akan terjadi. Pergerakan dari robot selanjutnya ditunjukkan pada Gambar 7.f. Setelah robot 1 melewati robot 2, maka kedua robot akan melanjutkan pergerakan sampai pada posisi koordinat tujuan masing-masing. Robot 2 akan melanjutkan pergerakan setelah tidak lagi mendeteksi adanya objek (robot 1). Sehingga pada saat robot 1 sedang menuju ke posisi tujuan, maka robot 2 dengan otomatis juga melanjutkan pergerakan menuju posisi awal pada koordinat 0,0. Tampak pada Gambar 7.g dan 7.h, bahwa masing-masing robot telah sampai pada posisi tujuan, yaitu robot 1 telah sampa pada koordinat 0,3 dan juga robot 2 telah sampai pada koordinat 0,0. Setelah robot 1 sampai pada posisi koordinat 0,3, robot melakukan pergerakan rotasi -90 derajat hadap kiri untuk meletakkan muatan. Berdasarkan kondisi robot, bahwa kondisi robot 1 telah sampai pada posisi tujuan dan sedang menaruh barang. Sedangkan kondisi robot 2 telah sampai pada koordinat 0,0. Berdasarkan pengujian konflik titik yang telah dilakukan, kedua robot dapat sampai pada posisi tujuan dengan selamat dan terhindar dari tabrakan yang hampir terjadi di node atau titik pada koordinat 0,2.



Gambar 7. Pengujian Sistem anti collision 2

5. KESIMPULAN

Dari hasil percobaan yang telah dilakukan dapat disimpulkan bahwa

1. Robot dapat mendeteksi 2 jenis konflik yang akan menyebabkan terjadinya tabrakan yaitu: *Node Conflict* dan *Opposite Conflict*.
2. Pada *node conflict* atau konflik titik solusi menghindari terjadinya tabrakan yaitu salah satu robot harus berhenti dan memberikan akses kepada robot lain untuk melanjutkan pergerakan.
3. Robot tetap dapat sampai pada posisi tujuan walaupun robot bergerak tidak dengan menggunakan jalur utama (*manhattan distance*).

6. DAFTAR PUSTAKA

- [1] Pitowarno.E., "ROBOTIKA: Desain, Kontrol, dan Kecerdasan Buatan." Yogyakarta. Penerbit ANDI, 2006.
- [2] Arizal. R., "Rancang Bangun Sistem Navigasi Automatic Guided Vehicle (AGV) Menggunakan Sensor Garis Berbentuk Lingkaran dan Fuzzy Logic." Bandung: Universitas Telkom, 2014.
- [3] Eberhardt, N., Bethlehem, P., "Automatic Guided Vehicle". United State Patent, 1989.
- [4] Suman. D.K, Pasan, M.K., "Design and Methodology Of Automated Guided Vehicle-A Review". 2016.
- [5] Nugroho, A.B., Haffif, L.K "Rancang Bangun Robot Pemindah Barang Berdasarkan Warna Berbasis Mikrokontroler Parallax BS2P40, UMJ, 2018.
- [6] Henry, F.T., "Reactive Automated Guided Vehicle Vision Guidance System". Cycle time cooperation, 2006.
- [7] Rashid, M.Z.A., "Metal Line Detection: A New Sensory System for Line Following Mobile Robot." Proceeding of ICCP2912 IEEE, 2016.
- [8] Michail, M., Alexandros, N., Evangelos, P., "Semi-autonomous Color Line-Following Educational Robots: Design and Implementation". 2011 IEEE/ASME International Conference on Advanced Intelligent Mechatronics (AIM2011) Budapest, Hungary, 2011.
- [9] Mehran, P.M., Sanaatiyan, M., Ghahroudi, M.R., "A Line Follower Robot from design to Implementation: Technical issues and problems", IEEE, 2011.
- [10] Barayyan, T.O., "Enhanced Line Follower Robot Using IR coding as an Artificial Landmark". Yanbu University College, 2013.
- [11] Abdul Ghani. M.F., Jati.A.N., Susanto.E., "Desain dan Realisasi Mobile Robot Menggunakan Sensor Ultrasonic untuk Menghindari Obstacle". e-Proceeding of Engineering, 2015.
- [12] Sonali K.K., Dhamesh H.S., Nishant, M.R., "Obstacle Avoidance for a Mobile Robot using a single ultrasonic range sensor". Emerging Trends in Robotics and Communication Technologies IEEE.2010:8-11, 2010.
- [13] Ahmad Fali, Mirani Wulandari, "Penerapan Metode Landmarc Menggunakan Manhattan Distance untuk Penentuan Lokasi RFID Tag pada Area Parkir Kendaraan Roda Dua", Prosiding Annual Research Seminar. Computer Science and ICT, 2017.
- [14] Yenie Syukriyah, Falahah, Hermi Solihin. "Penerapan algoritma A* (Star) untuk Mencari Rute Tercepat dengan Hambatan". Seminar nasional telekomunikasi dan informatik (SELISIK). Bandung. 2016.
- [15] Robby Rizky. "Pencarian Jalur Terdekat dengan Metode A*(Star) Studi Kasus Serang Labuan Provinsi Banten". Seminar Nasional Rekayasa Teknologi Informasi (SNARTISI). 2018
- [16] Ruixi Liu. Research on Optimization of the AGV Shortest-Path Model and Obstacle Avoidance Planning in Dynamic Environments. Mathematical Problems in Engineering Volume 2022, Article ID 2239342, 14 pages <https://doi.org/10.1155/2022/2239342>.