



Implementasi dan Analisis Purwarupa Sistem Collision Avoidance pada Mobil Pintar Berbasis Jaringan Sensor Nirkabel

Implementation and Analysis Prototype of Collision Avoidance System in Smart Car Based on Wireless Sensor Network (WSN)

Dwi Kresna Wijaya¹, Doan Perdana², Y. Gustommy Bisono³

^{1,2,3} Fakultas Teknik Elektro, Telkom University

^{1,2,3} Jl. Telekomunikasi No1, Terusan Buah Batu, Bandung, 40257, Indonesia

email: ¹dkresnawijaya@gmail.com, ²doanperdana@telkomuniversity.ac.id, ³tommy.bisono@gmail.com

INFORMASI ARTIKEL

Naskah diterima 26 Juli 2017

Direvisi 30 Agustus 2017

Disetujui 31 Oktober 2017

Keywords:

Collision Avoidance

Wireless Sensor Network

Pulse Width Modulation

Kata kunci :

Collision Avoidance

Wireless Sensor Network

Pulse Width Modulation

ABSTRACT

Accidents are the third highest cause of death in the world. Through the collision avoidance system applied to smart car, it is expected that accidents on the car can be avoided and the number of accidents can be reduced. This system applies the concept Wireless Sensor Network (WSN), the type of network that can pass through the process of sensing, transmitting data, and monitoring through internet connection.

The design and implementation of collision avoidance prototype in this smart car works by means of the distance between the cars obtained from the Ultrasonic HC-SR04 sensor which then becomes the Arduino Uno input to regulate the speed of the car in order to avoid a collision. Communication between cars using Xbee S2 is connected to Raspberry Pi to connect to the web server. Information gained distance and speed of the car will occur on websites that can be accessed by the user.

Maximum Xbee S2 test result are 89 meters. The delay value from the sensor to the monitoring system is 0.411 second, while the average throughput value is 641.73 bytes/s.

ABSTRAK

Kecelakaan menjadi penyebab tertinggi ketiga kematian di dunia. Melalui sistem *collision avoidance* yang diterapkan pada mobil pintar, diharapkan agar kecelakaan pada mobil dapat dihindarkan dan angka kecelakaan dapat berkurang. Sistem ini menerapkan konsep *Wireless Sensor Network* (WSN), jenis jaringan yang dapat melakukan proses *sensing*, pengiriman data, serta monitoring melalui koneksi internet.

Perancangan dan implementasi purwarupa *collision avoidance* pada mobil pintar ini bekerja dengan cara mendapatkan jarak antar mobil yang didapat dari sensor Ultrasonik HC-SR04 yang kemudian menjadi *input* Arduino untuk mengatur kecepatan optimal mobil agar tidak terjadi tabrakan. Komunikasi antar mobil menggunakan Xbee S2 yang dihubungkan ke Raspberry Pi untuk dikoneksikan ke *web server*. Informasi yang didapat berupa jarak dan kecepatan mobil akan ditampilkan pada *website* yang dapat diakses oleh pengguna.

Hasil pengujian didapatkan jangkauan maksimal Xbee S2 adalah 89 meter. Nilai *delay* dari sensor hingga sistem monitoring didapatkan rata-rata 0,411 detik, sedangkan nilai *throughput* rata-rata adalah 641,73 bytes/s.

1. Pendahuluan

Teknologi mobil pintar mulai dikembangkan oleh produsen mobil. Pengembangan ini menjadi topik hangat pada *Intelligent Transportation System* (ITS) karena berdasarkan data dari *World Health Organization* (WHO), kecelakaan kendaraan merupakan salah satu penyebab kematian yang tertinggi di dunia. *Collision avoidance* pada *smart car* merupakan suatu sistem pengendalian pada mobil yang bertujuan menghindari tabrakan dengan cara mengatur kecepatan mobil berdasarkan jarak. Pengendalian yang dilakukan oleh mobil menerapkan *self-organizing network*. Dengan adanya sistem ini, diharapkan agar tabrakan dapat dihindari. Implementasi teknologi ini menerapkan konsep *Wireless Sensor Network* (WSN), jenis jaringan nirkabel terdistribusi yang memanfaatkan teknologi *embedded system* yang

menggunakan sejumlah *node* sensor untuk melakukan proses *sensing*, pengiriman data, dan monitoring melalui koneksi internet (Pratama & Suakanto, 2015).

Sistem ini menggunakan sensor ultrasonik yang akan membaca jarak kemudian diteruskan ke mikrokontroler untuk dilakukan pengolahan menjalankan algoritma *collision avoidance* yang kemudian data-data tersebut akan dikirimkan pada *coordinator node* sebagai nilai pada sistem monitoring. Pada sistem ini terdapat sistem monitoring untuk memantau jarak terhadap objek dan kecepatan yang digunakan oleh mobil.

Tujuan dari penelitian ini adalah untuk menciptakan suatu sistem yang akan diterapkan pada mobil sebagai solusi untuk mengurangi tingginya tingkat kecelakaan. Sistem *collision avoidance* ini akan digunakan pada mobil pintar yang akan secara otomatis menghindari tabrakan melalui pengaturan kecepatan yang telah dilakukan perhitungan menggunakan *braking distance*.

Penelitian ini akan diterapkan pada *prototype* mobil pintar yang terdiri dari komponen motor dc, motor *driver* yang akan terhubung dengan Arduino sebagai mikrokontroler. Sebagai pengembangan dari penelitian sebelumnya, penelitian ini akan ditambahkan berupa sistem monitoring yang dapat di akses oleh pengguna menggunakan metode *wireless sensor network* (WSN).

2. Tinjauan Pustaka

2.1 Sistem *Collision Avoidance* pada Mobil Pintar

Sistem *Collision Avoidance* yang bekerja pada mobil pintar ini bekerja dengan cara mengubah kecepatan berdasarkan jarak objek di depannya. Jarak yang didapat merupakan hasil gelombang pantul dari objek yang ditangkap oleh *receiver* sensor ultrasonik. Setelah itu, dengan menggunakan nilai jarak terhadap objek, akan dapat ditentukan kecepatan optimum yang dapat digunakan oleh mobil. Dengan menggunakan sistem ini, potensi terjadinya kecelakaan yang dilakukan antar mobil dapat dikenali sehingga dapat dihindari. Perhitungan jarak aman antar mobil dapat dihitung menggunakan rumus *braking distance*.

$$d_b = \frac{\Delta V^2}{2\mu g} \dots\dots\dots 1)$$

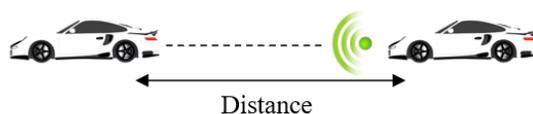
Dimana :

d_b = *braking distance* (m)

V = Kecepatan kendaraan (km/h)

μ = Koefisien friksi

g = Percepatan gravitasi (m/sec²)

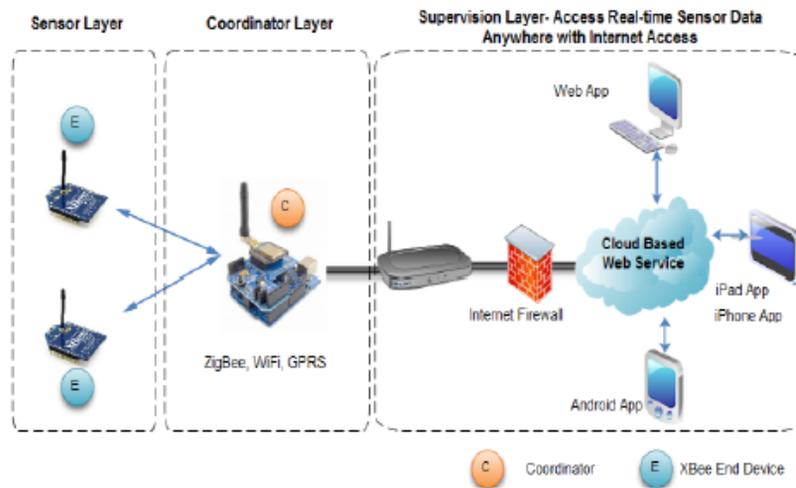


Gambar 1. Car Braking Distance

2.2 *Wireless Sensor Network* (WSN)

Wireless Sensor Network (WSN) adalah jenis jaringan nirkabel terdistribusi yang menerapkan teknologi *embedded system* yang menggunakan sejumlah *node* sensor untuk melakukan proses *sensing*, pengiriman data, dan monitoring melalui koneksi sensor meliputi banyak jenis di antaranya yaitu jarak, cahaya, temperatur, tekanan, dan lain-lain (Pratama & Suakanto, 2015).

WSN dapat diintegrasikan dengan *public network* seperti internet. Integrasi dengan jaringan internet dapat memberikan keuntungan berupa *remote* oleh pengguna dalam melakukan kontrol ataupun monitoring. Dalam mengintegrasikan dengan jaringan internet, WSN harus melalui lapisan *server* sebagai penyimpanan data hasil penginderaan sensor *node* serta pengolahannya yang kemudian akan diteruskan pada lapisan *client*.



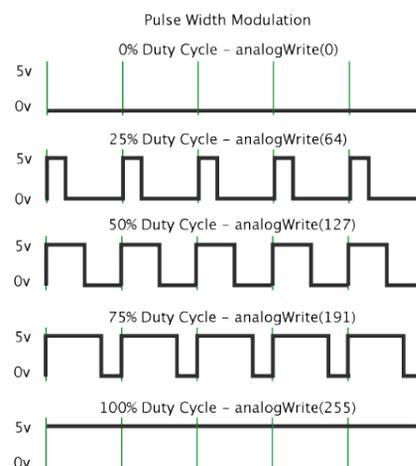
Gambar 2. Integrasi WSN dengan Teknologi Cloud (Yasodha & Babu, 2016)

Komunikasi dalam WSN membuhkan sensor *node* yang terhubung dengan *coordinator node*. Fungsi dari sensor *node* adalah melakukan pengamatan terhadap lingkungannya kemudian mengambil data yang dikirimkan pada mikrokontroler melalui modul *wireless* untuk dilakukan suatu pemrosesan. Data dari sensor *node* yang dikirimkan ke *coordinator node* itulah yang akan ditampilkan sebagai informasi.

2.3 Pulse Width Modulation (PWM)

Pulse Width Modulation (PWM) merupakan metode untuk mengatur lebar pulsa dalam suatu periode. Dengan kata lain PWM digunakan untuk mengatur kecepatan pada motor DC. Pengaplikasian PWM banyak diterapkan pada kontrol yang terintegrasi dengan mikrokontroler seperti pengaturan kecepatan motor DC. Dalam pengaturan kecepatan motor DC, ketika tegangan yang diberikan besar maka kecepatan yang dihasilkan akan tinggi. Pengaturan kontrol kecepatan motor DC dilakukan dengan memperhatikan suatu kondisi tertentu seperti pengaturan kecepatan motor DC berdasarkan suhu ruangan. (Marzuki, n.d.)

Sinyal PWM dapat dibangkitkan menggunakan dua cara yaitu metode analog dan metode digital. Metode analog merupakan metode pembangkitan sinyal PWM yang paling sederhana dimana cara kerjanya yaitu membandingkan tegangan *carrier* dengan tegangan referensi. Sedangkan pada metode *digital*, perubahan nilai PWM berdasarkan dengan resolusi yang dimiliki oleh PWM tersebut. Resolusi merupakan banyaknya variasi perubahan nilai pada PWM. Misalkan dalam PWM digital memiliki resolusi 8 bit, artinya PWM tersebut mempunyai variasi nilai sebanyak $2^8 = 256$ yaitu pada rentang nilai PWM 0 – 255. Dengan kata lain, dapat dilakukan perubahan nilai PWM pada *duty cycle* 0 - 100% dari *output* PWM. (Arduino.cc, n.d.)(Raza, Kamil, & Kumar, 2016)



Gambar 3. Pulse Width Modulation (Arduino.cc, n.d.)

2.4 Vehicular Ad Hoc Network (VANET)

Sekarang ini, *Vehicular Ad Hoc Network* (VANET) menjadi penelitian yang terus dikembangkan pada topik *Intelligent Transportation System* (ITS). VANET adalah *self-organizing network* yang diterapkan dalam kendaraan bergerak atau dengan mobilitas yang tinggi. Tujuan dari teknologi ini adalah untuk mengurangi permasalahan pada kendaraan seperti kemacetan, kecelakaan, dan lain sebagainya. Sebagai pengembangan, pada setiap mobil akan dapat bertukar informasi dengan cara pada setiap mobil akan dijadikan sebagai *node* dan sekaligus *router node* di sepanjang jalan menggunakan perangkat komunikasi *wireless*. Sehingga setiap *node* selalu bertukar informasi dengan *node* lainnya seperti pengiriman informasi trafik lalu lintas. Penerapan VANET dalam bertukar informasi dapat dijalankan melalui *Vehicle-to-Vehicle* (V2V) atau *Infrastructure-to-Vehicle*. (Muhtadi, A; Perdana, Doan; Munadi, 2015) (Perdana, Doan; Fitri Sari, 2013)

2.5 Parameter Pengujian

Parameter yang digunakan dalam pengujian pada sistem ini sebagai berikut (Sugeng, Istiyanto, Mustofa, & Ashari, 2015):

- *Delay*, merupakan waktu yang diperlukan untuk pengiriman sebuah paket dari *transmitter* ke *receiver*.

$$Delay = \frac{\text{Panjang paket}}{\text{link bandwidth}} \dots\dots\dots 2)$$

- *Throughput*, merupakan banyaknya paket yang diterima dibagi dengan interval waktu pengamatan.

$$Throughput = \frac{\text{Jumlah data yang diterima}}{\text{waktu pengiriman data}} \dots\dots\dots 3)$$

- *Availibilitas*, merupakan kemampuan suatu sistem dalam hal kesiapan pada saat dibutuhkan dengan kata lain adalah ketersediaan.

$$Availibilitas = \frac{(\text{uptime}-\text{downtime})}{\text{waktu total}} \times 100\% \dots\dots\dots 4)$$

- *Realibilitas*, merupakan kemampuan sistem dalam menjalankan fungsinya dalam suatu waktu tertentu dengan kata lain adalah kehandalan.

$$Realibilitas = \frac{(\text{uptime}-\text{downtime})}{\text{uptime}} \times 100\% \dots\dots\dots 5)$$

3. Metode Penelitian

Metode yang digunakan untuk menyelesaikan penelitian ini adalah :

1. Teknik Penelitian

Penelitian dilakukan untuk mencari metode yang akan diterapkan pada *prototype* mobil pintar menggunakan sistem *collision avoidance*. Metode yang diterapkan yaitu pengaturan kecepatan berdasarkan jarak menggunakan perhitungan *braking distance*.

Penelitian yang dilakukan oleh A.H. Ingle dan Rajesh Kumar Bambal (Ingle, Bambal, & Shobhane, 2017) yang berjudul *Intelligent Braking System* serta oleh S. Kalpana, J. Jerald Roiston, dan Sinto P Davis (Davis, 2016) yang berjudul *Intelligent Collision Preventive System Using Arduino Microcontroller* dijadikan referensi oleh penulis dalam menyusun penelitian ini. Penulis menggunakan metode yang sama yaitu *braking distance* yang digunakan dalam menyusun sistem *collision avoidance* ini. Kemudian komponen yang digunakan juga sama seperti Arduino Uno, sensor ultrasonik HC-SR04, motor DC, dan motor driver. Yang membedakan adalah adanya sistem monitoring yang dirancang oleh penulis berupa website sebagai *displaynya*. Penelitian ini juga dilakukan oleh Jordan Lewis dan tim yang berjudul *Fabrication of an Automated Collision Avoidance System Using Ultrasonic Sensor* (Lewis, Karthik, Lobo, Valder, & Rijesh, 2016).

Referensi pada penelitian selanjutnya berjudul *Sensor Based Accident Prevention System* oleh B. Aravinda dkk (Aravinda, Chaithralakshmi, & Ashutha, 2016) dan pada jurnal yang berjudul *Collision*

Avoidance System for Vehicle Safety oleh Shirish Srivatava (Srivastava, Kumar, & Singh, 2015). Pada kedua penelitian ini bertujuan untuk memberikan pencegahan dini kecelakaan melalui peringatan yang dihasilkan oleh perangkat sensor, akan tetapi sistem yang dijalankan masih menggunakan pengaturan kecepatan secara manual oleh pengemudi sehingga masih perlu untuk dikembangkan penelitian mengenai sistem *collision avoidance*.

2. Perancangan dan Realisasi

Perancangan *prototype smart car* menggunakan komponen Arduino uno, motor dc, dan motor *driver* yang diterapkan dengan algoritma sistem *collision avoidance*. *Prototype* ini akan ditambahkan sistem monitoring yang dapat diakses oleh *user* mengenai jarak dan kecepatan maksimum mobil.

3. Pengujian dan Analisis Data

Pengujian sistem *collision avoidance* yang telah diterapkan pada *prototype smart car* yaitu pengujian gerak dan pengukuran parameter-parameter QoS pada saat pengiriman informasi sampai dengan web monitoring.

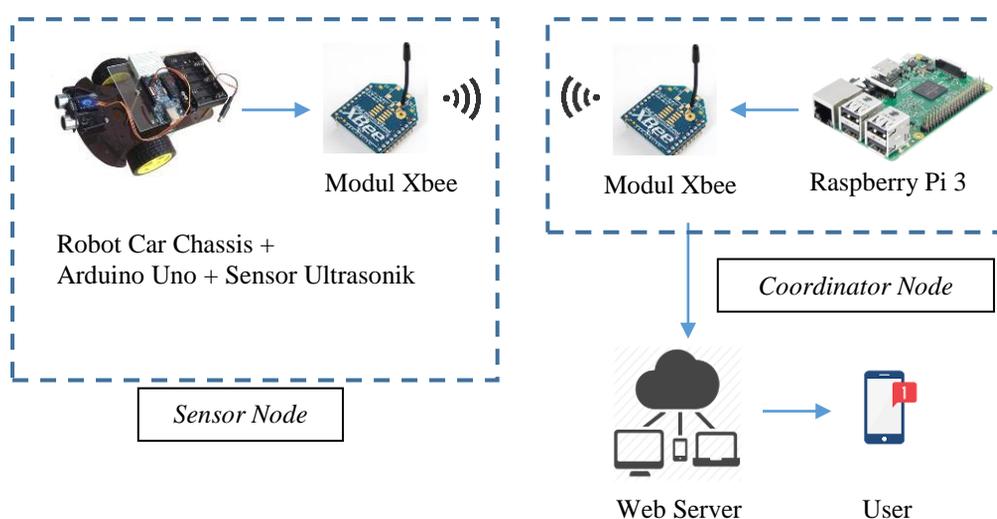
4. Hasil Penelitian dan Pembahasan

4.1. Diagram Blok Sistem

Prototype collision avoidance pada mobil pintar ini menerapkan sistem *Wireless Sensor Network* (WSN). Pada sistem ini memerlukan beberapa komponen yaitu, sensor ultrasonik HC-SR04, motor DC, dan motor *driver* yang diintegrasikan dengan Arduino Uno yang di rancang pada *robot car chassis*. Bagian ini disebut sebagai *sensor node*. Pada simulasi *collision avoidance* ini digunakan tembok sebagai objek yang akan dikenali oleh sensor ultrasonik. Mobil yang diberi sensor ultrasonik HC-SR04 ini akan melakukan *sensing* pada objek di depannya dan akan menerima pantulan gelombang yang kemudian akan diolah datanya untuk mendapatkan jarak antar mobil. Sensor tersebut terhubung pada mikrokontroler yang terdapat perangkat komunikasi *wireless* yaitu Xbee S2.

Setelah mendapatkan data jarak, pada mikrokontroler Arduino Uno diatur sebuah algoritma untuk instruksi agar terjadi *collision avoidance* seperti yang ingin disimulasikan. Kemudian data tersebut dikirimkan menggunakan modul RF Xbee Series 2 menuju Xbee yang terletak pada *coordinator node* untuk diteruskan ke *web server* hingga data dapat ditampilkan ke *website*. Dalam menghubungkan data sensor ke *web server* menggunakan koneksi selular LTE. Saat ini LTE menjadi rilis terbaik yang dikeluarkan oleh 3rd *Generation Partnership Project* (3GPP) dimana pada LTE kecepatan *downlink* mampu mencapai 100 Mbps pada mobilitas tinggi dan 1 Gbps pada mobilitas rendah serta *uplink* mencapai 50 Mbps (Ariyanti & Perdana, 2015; Perdana, Muayyadi, Mufti, & Chumaidiyah, 2014).

Pada *display* pengemudi nantinya akan terdapat jarak dan juga kecepatan pada mobil yang dikendarai. *Display* berupa *website* untuk monitoring yang dapat diakses oleh *user*. Diagram blok dari sistem *collision*

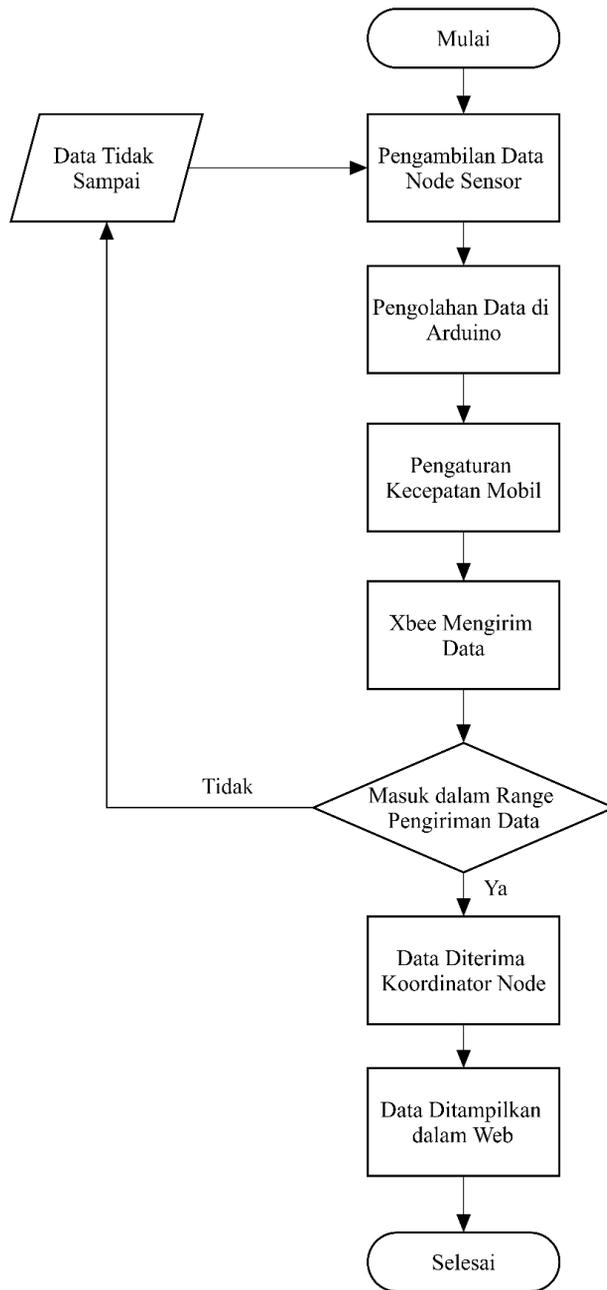


avoidance pada mobil pintar adalah seperti pada gambar berikut.

Gambar 4. Diagram Blok Sistem *Collision Avoidance*

4.2. Flowchart Sistem

Berikut ini merupakan alur kerja sistem *collision avoidance* beserta monitoring jarak dan kecepatan dari pengamatan sensor hingga sampai dapat diakses oleh pengguna.



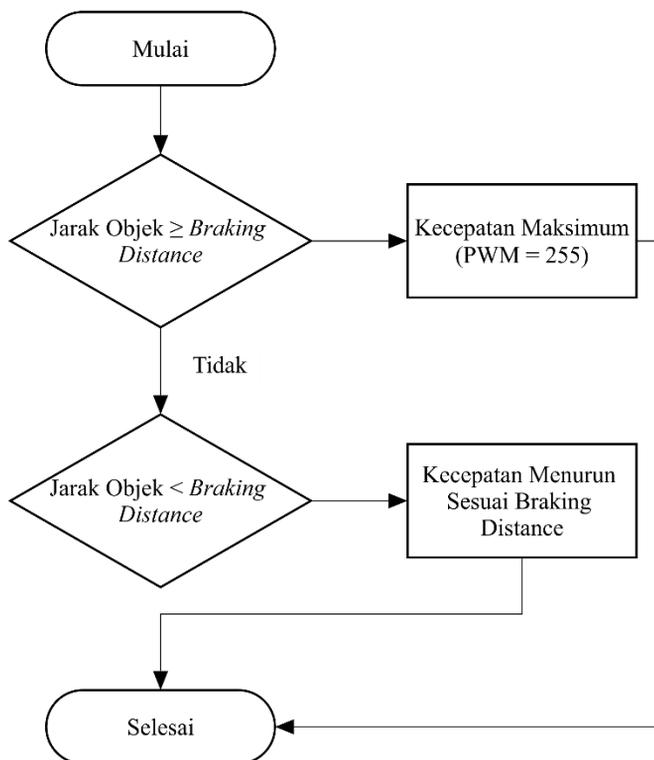
Gambar 5. Flow Chart Sistem Keseluruhan

4.3. Sistem Collision Avoidance

Sistem *collision avoidance* pada tugas akhir ini bekerja dengan cara mengatur kecepatan mobil berdasarkan jarak objek di depannya. Jarak objek didapat melalui sensor ultrasonik HC-SR04 yang memantulkan gelombang ke objek hingga diterima di *receiver* sensor ultrasonik. Hasil jarak yang diterima menjadi masukan di Arduino Uno untuk mengatur kecepatan yang akan dijalankan pada mobil untuk menjalankan sistem *collision avoidance*.



Gambar 6. Alur Pendeteksian Objek



Gambar 7. Flow Chart Pengaturan Kecepatan

Pengaturan kecepatan pada mobil dilakukan dengan cara mengatur kecepatan pada motor DC menggunakan metode *Pulse Width Modulation* (PWM) yaitu dengan rentang 0-255. Melalui PWM dapat diatur lebar pulsa yang telah dihasilkan oleh mikrokontroler sehingga dapat digunakan untuk mengatur kecepatan motor DC yang terhubung dengan motor *driver*. Berikut merupakan *flowchart* pengaturan kecepatan pada sistem *collision avoidance*.

Apabila jarak pengereman (d_b) lebih dari jarak yang didapat oleh sensor ultrasonik HC-SR04 maka kecepatan akan otomatis menurun. Kemudian apabila terdapat kecepatan yang sama dengan objek atau mobil yang berada di depannya maka kecepatan akan tetap sesuai kecepatan pada saat itu. Sehingga dengan menggunakan sistem pengaturan kecepatan berdasarkan jarak maka kecelakaan dapat dihindari.

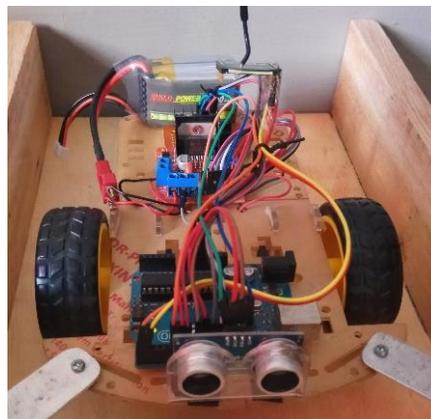
4.4. Kebutuhan *Hardware*

Pada tugas akhir ini, untuk membangun suatu sistem *collision avoidance* pada mobil pintar dibutuhkan beberapa komponen sebagai berikut.

Tabel 1. Kebutuhan Hardware

No.	Nama Komponen	Jumlah
1.	Robot Car Chassis	1
2.	Arduino Uno	1
3.	Sensor Ultrasonik	1
4.	Driver Motor	1
5.	Motor DC	2
6.	Modul Xbee Series 2	2
7.	Raspberry Pi 3	1
8.	Baterai Li-Po 11.1 Volt	1

Berikut merupakan hasil perancangan dari *prototype* mobil pintar dengan sistem *collision avoidance*.



Gambar 8. Prototype mobil pintar

4.5. Pengujian Sistem Gerak pada *Prototype* Mobil Pintar

Implementasi sistem *collision avoidance* pada tugas akhir ini memanfaatkan pengaturan pengendalian kecepatan pada motor DC menggunakan metode *Pulse Width Modulation* (PWM). Melalui PWM dapat diatur lebar pulsa yang telah dihasilkan oleh mikrokontroler sehingga dapat digunakan untuk mengatur kecepatan motor DC yang terhubung dengan *motor driver*. Berikut ini merupakan nilai besaran PWM yang akan diterapkan pada sistem *collision avoidance*.

Tabel 2. Rentang Nilai PWM

PWM	Keterangan
0	Diam
105	Sangat Lambat
155	Lambat
205	Sedang
255	Cepat

Berdasarkan tabel nilai PWM di atas, kemudian di terapkan pada *prototype* mobil untuk menguji kecepatan. Nilai kecepatan didapatkan dari pengujian yang dijalankan pada *prototype* mobil sepanjang satu meter yang kemudian didapat nilai waktu. Berdasarkan rumus $v = \frac{s}{t}$ maka dihasilkan kecepatan sebagai berikut.

Tabel 3. Pengujian Kecepatan Mobil

PWM	Kecepatan (m/s)	Kecepatan (km/jam)
0	0	0
105	0.42	1.51
155	0.62	2.32
205	0.85	3.06
255	1.54	5.54

Pada pengujian di atas, didapatkan hasil kecepatan yang kecil dibandingkan dengan keadaan sebenarnya pada mobil. Hal ini dikarenakan pada *prototype* sistem *collision avoidance* ini menggunakan komponen dengan spesifikasi rendah yang biasanya digunakan pada robot. Komponen tersebut antara lain yaitu motor DC yang mempunyai rentang 3V-6V, *motor driver* dengan rentang 7V - 12V, dan catu daya baterai Li-Po 11,7 Volt.

4.6. Braking Distance

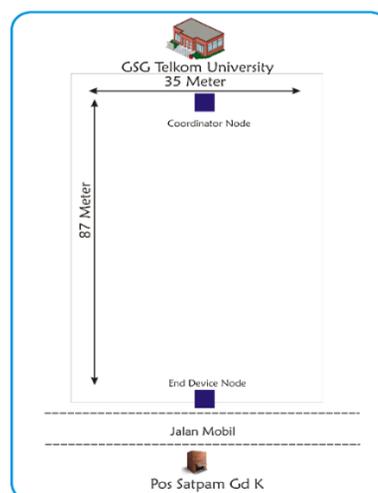
Braking distance dalam hal berkendara merupakan hal penting untuk menghindari kecelakaan. Jarak pengereman merupakan jarak yang akan ditempuh kendaraan saat terjadi perlambatan hingga kendaraan berhenti. Perhitungan *braking distance* menggunakan Persamaan 1.

Tabel 4. Perhitungan *Braking Distance*

Kecepatan (km/jam)	<i>Braking Distance</i> (m)
5	0,12
10	0,49
20	1,96
30	4,42
40	7,78

Dengan menggunakan rumus di atas, akan didapatkan jarak pengereman seperti pada Tabel 4. Dalam sistem *collision avoidance*, pengaturan kecepatan akan dilakukan berdasarkan jarak pengereman. Apabila jarak pengereman (d_b) lebih dari jarak yang didapat oleh sensor ultrasonik HC-SR04 maka kecepatan akan otomatis menurun. Kemudian apabila terdapat kecepatan yang sama dengan objek atau mobil yang berada di depannya maka kecepatan akan tetap sesuai kecepatan pada saat itu. Sehingga dengan menggunakan sistem pengaturan kecepatan berdasarkan jarak maka kecelakaan dapat terhindari.

4.7. Pengujian Performansi Xbee S2

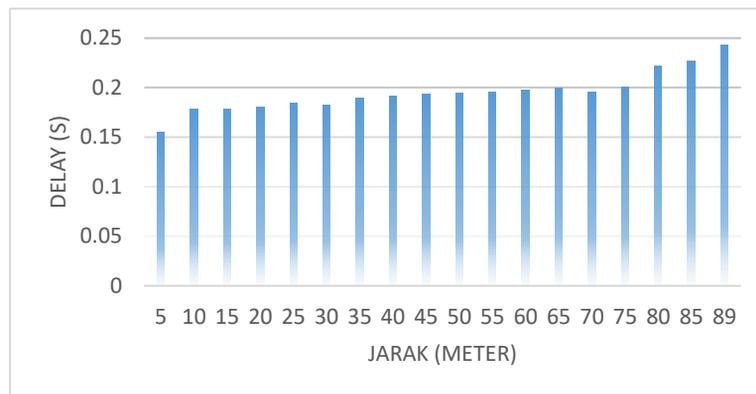


Gambar 9. Denah Pengujian Halaman GSG Telkom University

Pengujian jangkauan Xbee S2 dilakukan untuk mengetahui jarak maksimal yang dapat dijangkau antar modul Xbee S2 dalam pengiriman data. Dalam pengujian ini, penulis melakukan pengujian di ruang terbuka yang terletak di halaman Gedung Serba Guna (GSG) Telkom University. Pengujian ini dilakukan dengan cara memasang modul Xbee S2 sebagai *end device node* dan *coordinator node*.

Berdasarkan pengujian, didapatkan hasil jangkauan maksimal antara *end device node* dengan *coordinator node* dalam keadaan LOS mencapai jarak 89 meter. Hasil pengujian yang didapatkan lebih rendah dibandingkan dengan jangkauan maksimum menurut datasheet Xbee S2 yaitu mencapai 120 meter. Hal tersebut dapat dikarenakan karena fabrikasi Xbee yang kurang bagus sehingga mempengaruhi kinerja optimal dari Xbee itu sendiri.

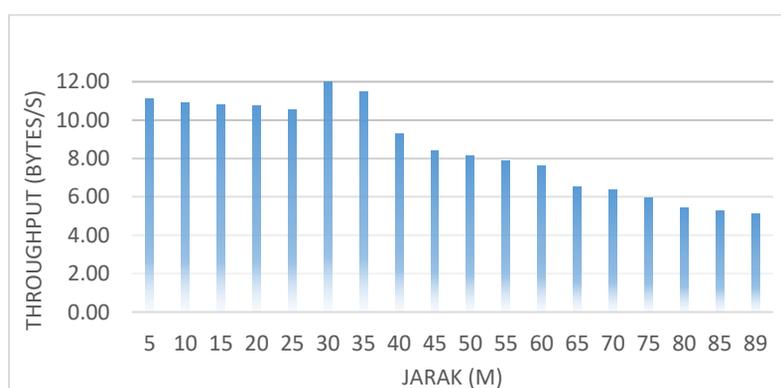
4.7.1. Pengujian Delay



Gambar 10. Grafik Pengujian Delay pada Xbee S2

Pengujian dilakukan pada jarak 5 meter hingga 90 meter dengan kelipatan 5 meter. Didapatkan hasil bahwa semakin jauh jangkauan antara *coordinator node* dengan *end device node* maka *delay* yang dihasilkan semakin besar. Hal ini dikarenakan ketika jarak antar *node* semakin jauh maka proses pengiriman data akan membutuhkan waktu yang semakin lama di samping interferensi dari luar sistem yang dapat mempengaruhi nilai *delay* yang semakin besar.

4.7.2. Pengujian Throughput



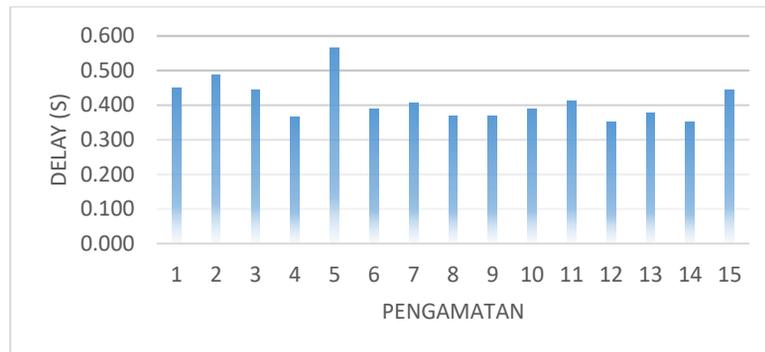
Gambar 11. Pengujian Throughput pada Xbee S2

Pengujian dilakukan pada jarak 5 meter hingga 90 meter dengan kelipatan 5 meter. Berdasarkan hasil pengujian didapatkan nilai bahwa semakin jauh jangkauan Xbee maka nilai *throughput* semakin turun. Hal ini berbanding terbalik dengan nilai *delay* dimana jangkauan semakin jauh nilai *delay* semakin tinggi. Terdapat nilai optimum *throughput* pada jarak 30 meter dimana nilainya yaitu 12 bytes/s. Penggunaan modul Xbee S2 mempunyai jarak kerja optimal yaitu pada jarak 30-35 meter sehingga terdapat anomali pada grafik. (Mukherji & Eba, 2016)

4.8. Pengujian Performansi Sistem Monitoring

4.8.1. Pengujian Delay

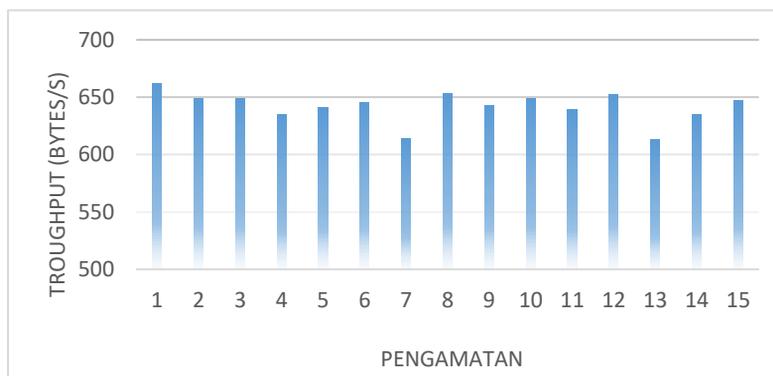
Pengujian *delay* dilakukan dengan cara melakukan pengiriman data dari *end node device* ke *coordinator node* selama dua menit. Kemudian secara bersamaan dilakukan *capture* data pada Raspberry Pi menggunakan *software* tshark. Hasil *capture* tersebut akan didapat setelah waktu pegujian selesai. Analisa hasil pengujian menggunakan *software* wireshark. Berikut merupakan hasil pengujian *delay* pada sistem monitoring



Gambar 12. Grafik Pengujian *Delay* pada Sistem Monitoring

Berdasarkan data yang didapat, dihasilkan rata-rata nilai *delay* yaitu 0,411 detik. Hasil ini merupakan *delay end-to-end* yang dihasilkan oleh sistem.

4.8.2. Pengujian Throughput



Gambar 13. Grafik Pengujian *Throughput* pada Sistem Monitoring

Berdasarkan data yang didapat, dihasilkan rata-rata nilai *throughput* yaitu 641,73 bytes/s. Hasil ini merupakan nilai *throughput* yang dihasilkan oleh sistem.

5. Simpulan dan Saran

5.1 Kesimpulan

Penerapan sistem *collision avoidance* pada penelitian ini dapat menjalankan sesuai fungsinya yaitu menghindari tabrakan secara otomatis. Sistem *collision avoidance* yang diterapkan pada *prototype* mobil pintar ini dirancang menggunakan sensor ultrasonic HC-SR04, motor DC, motor *driver* dan Arduino Uno sebagai mikrokontrolernya. Sistem *collision avoidance* ini menerapkan pengaturan kecepatan berdasarkan jarak untuk menghindari tabrakan dengan menggunakan perhitungan *braking distance*. Xbee S2 yang digunakan sebagai modul komunikasi *wireless*-nya dapat bekerja hingga jarak 89 meter dengan *delay* terbesar yaitu 0.2424 detik pada jarak 89 meter. Besarnya jarak antara *node device* dengan *coordinator node* berpengaruh pada kualitas jaringan yaitu *delay* dan *throughput* dalam melakukan pengiriman data. Semakin jauh jarak antara *node device* dengan *coordinator node* maka kualitas jaringan akan semakin menurun.

5.2 Saran

Sistem *collision avoidance* dapat dikembangkan untuk lintasan berkelok bukan hanya lintasan lurus saja yang dapat dilakukan dengan penambahan komponen sensor. Kemudian dapat dilakukan dengan penggunaan metode lain dalam pengaturan sistem *collision avoidance* yang lebih akurat dalam perhitungannya. Pada sistem monitoring dapat dikembangkan untuk penambahan *buzzer* sebagai pengingat tanda bahaya.

Daftar Pustaka

- Aravinda, B., Chaithralakshmi, C., & Ashutha, K. (2016). Sensor Based Accident Prevention System. *IJIREEICE*, 4(6), 14–17.
- Arduino.cc. (n.d.). Pulse Width Modulation (PWM). Retrieved April 6, 2017, from <https://www.arduino.cc/en/Tutorial/PWM>
- Ariyanti, S., & Perdana, D. (2015). Feasibility Analysis of LTE 1 . 8 GHz for Mobile Operators in Indonesia Analisis Kelayakan Implementasi LTE 1 . 8 GHz Bagi Operator Seluler di Indonesia. *Buletin Pos Dan Telekomunikasi*, 13(1), 61–78.
- Davis, S. P. (2016). INTELLIGENT COLLISION PREVENTIVE SYSTEM USING ARDUINO MICROCONTROLLER Motor drive Bluetooth module Ultrasonic sensor, 78–81.
- Ingle, A. H., Bambal, R. K., & Shobhane, S. (2017). Intelligent Braking System. *International Journal of Research In Science & Engineering*, 3(April), 39–47.
- Lewis, J., Karthik, B. M., Lobo, J. M., Valder, J., & Rijesh, M. (2016). Fabrication of an Automated Collision Avoidance System Using Ultrasonic Sensor. *Journal of Mechanical Engineering and Automation*, 6, 97–101.
- Marzuki, A. (n.d.). Pulse Width Modulation (PWM). Retrieved June 28, 2017, from http://andri_mz.staff.ipb.ac.id/pulse-width-modulation-pwm/%0APulse
- Muhtadi, A; Perdana, Doan; Munadi, R. (2015). Performance Evaluation of AODV, DSDV, and ZRP Using Vehicular Traffic Load Balancing Scheme on VANETs. *International Journal of Simulation System, Science and Technology*.
- Mukherji, A., & Eba, R. (2016). ZigBee Performance Analysis. *IEEE WiSPNET*.
- Perdana, Doan; Fitri Sari, R. (2013). Performance Evaluation of Multi-channel Operation IEEE. *International Journal of Computer Science and Network Security*, 13(3), 42–44.
- Perdana, D., Muayyadi, A. A., Mufti, N., & Chumaidiyah, E. (2014). Analisa Kelayakan Refarming Frekuensi 2100 MHz dengan Analisis Prediksi Cakupan Feasibility Analysis Refarming Frequency 2100 MHz with Coverage Estimation Analysis, 183–196.
- Pratama, I. P. A. E., & Suakanto, S. (2015). *Wireless Sensor Network* (1st ed.). Bandung: Informatika.
- Raza, K. M., Kamil, M., & Kumar, P. (2016). “ Speed Control of DC Motor by using PWM ,” 5(4), 307–309.
- Srivastava, S., Kumar, R., & Singh, K. S. K. (2015). Collision Avoidance System for Vehicle Safety. *IJSRD*, 3(4), 1730–1732.
- Sugeng, W., Istiyanto, J. E., Mustofa, K., & Ashari, A. (2015). The Impact of QoS Changes towards Network Performance. *International Jurnal of Computer Networks and Communications Security*, 3(2), 48–53.
- Yasodha, R., & Babu, R. L. (2016). Integrating Wireless Sensor Network into Cloud Services for Dynamic Environment Monitoring. *South Asian Journal of Engineering and Technology*, 2(17), 294–298.