



**RANCANG BANGUN SISTEM PENILAIAN UJIAN PRAKTIK SIM C
BERBASIS WIRELESS MENGGUNAKAN PROTOKOL KOMUNIKASI
PESAN MQTT**



Oleh:

**RIZKI HARIS SETIAWAN
14410200031**

**FAKULTAS TEKNOLOGI DAN INFORMATIKA
INSTITUT BISNIS DAN INFORMATIKA STIKOM SURABAYA
2019**

**RANCANG BANGUN SISTEM PENILAIAN UJIAN PRAKTIK SIM C
BERBASIS WIRELESS MENGGUNAKAN PROTOKOL KOMUNIKASI
PESAN MQTT**

TUGAS AKHIR

Diajukan sebagai salah satu syarat untuk menyelesaikan

Program Sarjana Komputer



FAKULTAS TEKNOLOGI DAN INFORMATIKA

INSTITUT BISNIS DAN INFORMATIKA STIKOM SURABAYA

2019



"Bahagia adalah sebuah pilihan"

INSTITUT BISNIS
DAN INFORMATIKA

stikom
SURABAYA

TUGAS AKHIR
RANCANG BANGUN SISTEM PENILAIAN UJIAN PRAKTIK SIM C
BERBASIS WIRELESS MENGGUNAKAN PROTOKOL KOMUNIKASI

PESAN MQTT

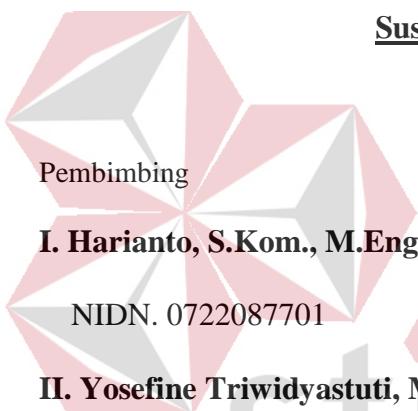
Dipersiapkan dan disusun oleh

Rizki Haris Setiawan

NIM : 14.41020.0031

Telah diperiksa, diuji dan disetujui oleh Dewan Pengaji
Pada : 18 Januari 2019

Susunan Dewan Pengaji



Pembimbing

I. Harianto, S.Kom., M.Eng.

NIDN. 0722087701

II. Yosefine Triwidayastuti, M.T.

NIDN. 0729038504

Pembahas

I. Pauladie Susanto, S.Kom., M.T.

NIDN. 0729047501



31/01/2019



31/01/2019



31/01/2019

Tugas akhir ini telah diterima sebagai salah satu persyaratan

untuk memperoleh gelar Sarjana



Dekan Fakultas Teknologi dan Informatika

INSTITUT BISNIS DAN INFORMATIKA STIKOM SURABAYA

PERSETUJUAN PUBLIKASI DAN KEASLIAN KARYA ILMIAH

Sebagai mahasiswa Institut Bisnis dan Informatika Stikom Surabaya, saya :

Nama : Rizki Haris Setiawan
NIM : 14410200031
Program Studi : S1 Sistem Komputer
Fakultas : Fakultas Teknologi dan Informatika
Jenis Karya : Tugas Akhir
Judul Karya : **RANCANG BANGUN SISTEM PENILAIAN UJIAN PRAKTIK SIM C BERBASIS WIRELESS MENGGUNAKAN PROTOKOL KOMUNIKASI PESAN MQTT**

Menyatakan dengan sesungguhnya bahwa:

1. Demi pengembangan Ilmu Pengetahuan, Teknologi dan Seni, saya menyetujui memberikan kepada Institut Bisnis dan Informatika Stikom Surabaya Hak Bebas Royalti Non-Eksklusif (*Non-Exclusive Royalty Free Right*) atas seluruh isi/ sebagian karya ilmiah saya tersebut di atas untuk disimpan, dialihmediakan dan dikelola dalam bentuk pangkalan data (*database*) untuk selanjutnya didistribusikan atau dipublikasikan demi kepentingan akademis dengan tetap mencantumkan nama saya sebagai penulis atau pencipta dan sebagai pemilik Hak Cipta
2. Karya tersebut di atas adalah karya asli saya, bukan plagiat baik sebagian maupun keseluruhan. Kutipan, karya atau pendapat orang lain yang ada dalam karya ilmiah ini adalah semata hanya rujukan yang dicantumkan dalam Daftar Pustaka saya
3. Apabila dikemudian hari ditemukan dan terbukti terdapat tindakan plagiat pada karya ilmiah ini, maka saya bersedia untuk menerima pencabutan terhadap gelar kesarjanaan yang telah diberikan kepada saya.

Demikian surat pernyataan ini saya buat dengan sebenarnya.

Surabaya, Januari 2019

Yang menyatakan



Rizki Haris Setiawan

NIM : 14410200031

ABSTRAK

Surat ijin mengemudi (SIM C) adalah salah satu tanda atau bukti yang diberikan pihak kepolisian bahwa telah layak untuk mengendarai sebuah kendaraan roda dua. Pada penelitian ini penulis ingin membuat sistem penilaian ujian praktik SIM C menggunakan Mikrokontroller. Peneliti mengembangkan penelitian sebelumnya yang pengiriman datanya menggunakan serial menjadi pengiriman data menggunakan wireless. MQTT adalah salah satu metode pengiriman data berbasis wireless, yang menggunakan server dan klien. Broker adalah program yang dijalankan pada sebuah device sebagai server. Client adalah Mikrokontroller yang sudah diprogram menggunakan library pubsubclient dari arduino ide. Aplikasi yang dibuat menggunakan bahasa pemrograman visual basic pada visual studio. Dalam penelitian ini digunakan Sensor Getar SW-420 sebagai pendekripsi untuk setiap patok dan Sensor Ultrasonik HCSR-04 sebagai penghitung kecepatan rata-rata pada kendaraan. Penelitian ini menggunakan tiga buah lintasan yaitu Keseimbangan, Zig-zag dan Putar balik. Sensor getar dapat mengirimkan data ke aplikasi ketika patok tersenggol oleh kendaraan. Sensor ultrasonik dapat merekam kecepatan kendaraan saat peserta melewati garis start dan menampilkan diaplikasi dengan jarak. Aplikasi dapat menampilkan hasil akhir dari tiga ujian yang sudah dilakukan dan disimpan sementara. Dari Pengujian yang sudah dilakukan terhadap tiga lintasan : Uji Keseimbangan, Uji Zig-Zag dan Uji Utur/Putarbalik. Didapatkan 22 komunikasi telah berhasil terkirim dari 22 Komunikasi yang berlangsung dan dengan tingkat keberhasilan 100%.

Kata Kunci : *Wireless, SIM C , MQTT, Aplikasi Sistem Penilaian*

KATA PENGANTAR

Pertama-tama penulis panjatkan puji dan syukur ke hadirat Allah SWT yang telah memberikan kekuatan, kesehatan lahir dan batin sehingga penulis dapat menyelesaikan penulisan Tugas Akhir ini dengan sebaik-baiknya. Penulis mengambil judul “*Rancang Bangun Sistem Penilaian Ujian Praktik SIM C Berbasis Wireless Menggunakan Protokol Komunikasi Pesan MQTT*” ini sebagai salah satu syarat dalam menyelesaikan Tugas Akhir di Institut Bisnis dan Informatika Stikom Surabaya

Pada kesempatan kali ini penulis juga ingin mengucapkan terima kasih kepada:

1. Bapak, Ibu tercinta yang telah memberikan dukungan dan doa selama mengerjakan Tugas Akhir ini.
2. Pimpinan Stikom Surabaya yang telah banyak memberikan motivasi serta teladan yang dapat membantu penulis selama menempuh pembelajaran hingga saat ini.
3. Bapak Pauladie Susanto, S.Kom., M.T., selaku Ketua Program Studi Sistem Komputer Stikom Surabaya yang senantiasa memberikan dukungan kepada penulis sehingga penulis dapat melaksanakan Tugas Akhir ini dengan baik.
4. Bapak Harianto, S.Kom., M.Eng. dan Ibu Yosefine Triwidayastuti, M.T., selaku dosen pembimbing satu dan dua yang telah membantu serta mendukung setiap kegiatan sehingga pelaksanaan Tugas Akhir ini dapat berjalan dengan baik.
5. Bapak Pauladie Susanto, S.Kom., M.T., selaku penguji yang senantiasa memberikan dukungan kepada penulis sehingga penulis dapat melaksanakan Tugas Akhir ini dengan baik.

6. Seluruh dosen Pengajar Program Studi S1 Sistem Komputer yang telah mendidik, memberi motivasi kepada penulis selama masa kuliah di Institut Bisnis dan Informatika Stikom Surabaya.
7. Terima kasih terhadap seluruh anggota Komunitas Robotic Stikom Surabaya, yang selalu memberikan semangat, pengalaman serta bantuannya.
8. Teman-teman dari komunitas Cyber Robotic Stikom, teman-teman seperjuangan Angkatan 2014, adik dan kakak angkatan Jurusan S1 Sistem Komputer yang mendukung dan membantu penulis selama masa dan penyusunan buku Tugas Akhir ini.
9. Seluruh pihak yang tidak dapat penulis tuliskan satu persatu yang telah membantu penulis secara langsung maupun tidak langsung.

Banyak hal dalam laporan Tugas Akhir ini yang masih perlu diperbaiki lagi. Oleh karena itu penulis mengharapkan saran dan kritik yang dapat membangun dari semua pihak agar dapat menyempurnakan penulisan ini kedepannya. Penulis juga memohon maaf yang besar jika terdapat kata-kata yang salah serta menyinggung perasaan pembaca. Akhir kata penulis ucapan banyak terima kasih yang besar kepada para pembaca, semoga tulisan ini dapat bermanfaat bagi para pembaca.

Surabaya, Desember 2018

Penulis

DAFTAR ISI

	Halaman
ABSTRAK	vi
KATA PENGANTAR	vii
DAFTAR ISI.....	ix
DAFTAR GAMBAR	xiv
DAFTAR TABEL.....	xvii
BAB I PENDAHULUAN	1
1.1 Latar Belakang	1
1.2 Perumusan Masalah.....	3
1.3 Batasan Masalah.....	4
1.4 Tujuan.....	4
1.5 Manfaat.....	5
1.6 Sistematika Penulisan	5
BAB II STUDI PUSTAKA.....	7
2.1 Tinjauan Pustaka	7
2.2 Surat Ijin Mengemudi (SIM).....	8
2.2.1 Syarat Pembuatan SIM C.....	9
2.2.2 Tatacara Pembuatan SIM C	9
2.2.3 Ujian Praktik SIM C	10
2.3 Wemos.....	14
2.3.1 Wemos D1 Mini.....	15
2.3.2 <i>Chipset</i> pada Wemos D1 Mini.....	17
2.4 Sensor Getar SW-420.....	18

2.5	Sensor Ultrasonik HC-SR04	20
2.6	Raspberry Pi 3	23
	2.6.1 Spesifikasi Raspberry Pi 3 Model B	26
2.7	MQTT (<i>Message Queuing Telemetry Transport</i>)	26
	2.7.1 <i>Publish and Subscribe</i>	27
	2.7.2 <i>Topic</i> dan <i>QOS (Quality of Service) level</i>	29
	2.7.3 Mosquitto Broker.....	30
2.8	Microsoft Visual Studio 2017	31
	2.8.1 Visual Basic .NET	32
2.9	Arduino IDE	33
BAB III METODE PENELITIAN.....		34
3.1	Metode Penelitian.....	34
3.2	Model Perancangan Keseluruhan.....	35
3.3	Perancangan Perangkat Keras	39
	3.3.1 Rangkaian Sensor Getar SW-420 pada Patok Tiang Pembatas	39
	3.3.2 Rangkaian Sensor Ultrasonik HC-SR04 pada Patok Tiang Pembatas	40
	3.3.3 Raspberry Pi 3 Sebagai Broker / Server	42
	3.3.4 Perancangan Komunikasi pada Sistem	45
3.4	Perancangan Perangkat Lunak	46
	3.4.1 Pembacaan Data Sensor Getar pada Patok Tiang Pembatas....	48
	3.4.2 Pembacaan Data Sensor Ultrasonik pada Patok Tiang Pembatas	52
	3.4.3 Perancangan Aplikasi pada Sistem	58
	3.4.4 Perhitungan Kecepatan	79

3.5	Perancangan Mekanik	81
3.5.1	Patok Tiang Pembatas Sensor Getar dan Sensor Ultrasonik	81
3.5.2	Lintasan Uji Keseimbangan.....	83
3.5.3	Lintasan Uji Zig-zag	84
3.5.4	Lintasan Uji U-turn/Putar Balik.....	85
	BAB IV HASIL DAN PEMBAHASAN	87
4.1	Pengujian Wemos D1 Mini	87
4.1.1	Tujuan	87
4.1.2	Alat yang Digunakan	87
4.1.3	Prosedur Pengujian	87
4.1.4	Hasil Pengujian	88
4.2	Pengujian Sensor Getar SW-420.....	90
4.2.1	Tujuan	90
4.2.2	Alat yang Digunakan	90
4.2.3	Prosedur Pengujian	90
4.2.4	Hasil Pengujian	91
4.3	Pengujian Sensor Ultrasonik HC-SR04	92
4.3.1	Tujuan	92
4.3.2	Alat yang Digunakan	92
4.3.3	Prosedur Pengujian	92
4.3.4	Hasil Pengujian	93
4.4	Pengujian Pengukur Kecepatan pada Lintasan Uji Keseimbangan ...	95
4.4.1	Tujuan	95
4.4.2	Alat yang Digunakan	95

4.4.3 Prosedur Pengujian	96
4.4.4 Hasil Pengujian	96
4.5 Pengujian Komunikasi Data Antara Patok Sensor Getar Lintasan dengan Broker / <i>Server</i>	98
4.5.1 Tujuan	98
4.5.2 Alat yang Digunakan	98
4.5.3 Prosedur Pengujian	99
4.5.4 Hasil Pengujian	99
4.6 Pengujian Delay Pengiriman Data <i>Client</i> ke Broker.....	102
4.6.1 Tujuan	102
4.6.2 Alat yang Digunakan	102
4.6.3 Prosedur Pengujian	103
4.6.4 Hasil Pengujian	103
4.7 Pengujian Daya Tahan Baterai pada Wemos D1 Mini saat Proses Pengiriman Data.....	104
4.7.1 Tujuan	104
4.7.2 Alat yang digunakan	104
4.7.3 Prosedur Pengujian	105
4.7.4 Hasil Pengujian	105
4.8 Pengujian Lintasan Uji pada Aplikasi Secara Keseluruhan	107
4.8.1 Tujuan	107
4.8.2 Alat yang Digunakan	108
4.8.3 Prosedur Pengujian	108
4.8.4 Hasil Pengujian	109
BAB V PENUTUP.....	116

5.1	Kesimpulan.....	116
5.2	Saran.....	117
	DAFTAR PUSTAKA	118
	LAMPIRAN	120
	Lampiran 1. Program Aplikasi Sistem Penilaian.....	120
	Form Menu Utama	120
	Form Pengaturan Broker	121
	Form Uji Keseimbangan	122
	Form Uji Zig-Zag	132
	Form Uji Putar Balik	138
	Form Hasil Akhir	144
	Lampiran 2. Program Patok Sensor Getar SW-420.....	145
	Lampiran 3. Program Patok Sensor Ultrasonik HC-SR04	147
	BIODATA.....	151

DAFTAR GAMBAR

	Halaman
Gambar 2.1 Patok Untuk Ujian Praktik Sim C	11
Gambar 2.2 Denah Lintasan Uji Keseimbangan.....	12
Gambar 2.3 Denah Lintasan Uji Membentuk Angka 8.....	13
Gambar 2.4 Denah Lintasan Uji Rem Menghindar	14
Gambar 2.5 Denah Lintasan Uji U Turn	14
Gambar 2.6 Wemos D1 Mini (Kiri) Dan Wemos D1 R1 (Kanan)	15
Gambar 2.7 Pin I/O Wemos D1 Mini	16
Gambar 2.8 Sensor Getar Sw-420.....	19
Gambar 2.9 Skematik Sensor Getar Sw-420.....	20
Gambar 2.10 Sensor Ultrasonik Hc-Sr04.....	20
Gambar 2.11 Proses Penerimaan Data Sensor Ultrasonik	22
Gambar 2.12 Raspberry Pi 3	24
Gambar 2.13 Blok Diagram Raspberry Pi 3 Model B	25
Gambar 2.14 Alur Publish Dan Subscribe Mqtt	28
Gambar 2.15 Logo Visual Studio.....	31
Gambar 3.1 Blok Diagram Struktur Sistem.....	34
Gambar 3.2 Blok Diagram Komunikasi Pada Sistem.....	36
Gambar 3.3 Rangkaian Sensor Getar Sw-402.....	39
Gambar 3.4 Rangkaian Sensor Ultrasonik Hc-Sr04	41
Gambar 3.5 Rangkaian Raspberry Pi 3	42
Gambar 3.6 Eror Pada Broker Di Raspberry Pi 3	44
Gambar 3.7 Skema Jalur Koneksi Sistem	45

Gambar 3.8 Skema Jalur Koneksi Sistem Dengan Ap Tambahan.....	46
Gambar 3.9 Flowchart Sistem Secara Keseluruhan.....	47
Gambar 3.10 Flowchart Pembacaan Data Sensor Getar	52
Gambar 3.11 Flowchart Pembacaan Data Sensor Ultrasonik	53
Gambar 3.12 Tampilan Menu Utama.....	58
Gambar 3.13 Flowchart Pada Form Main Menu	59
Gambar 3.14 Tampilan Menu Koneksi Broker.....	60
Gambar 3.15 Flowchart Pada Menu Pengaturan Broker	61
Gambar 3.16 Tampilan Menu Pengujian Uji Keseimbangan	63
Gambar 3.17 Flowchart Pada Menu Uji Keseimbangan.....	68
Gambar 3.18 Tampilan Menu Pengujian Uji Zig-Zag	69
Gambar 3.19 Flowchart Pada Menu Uji Zig-Zag	71
Gambar 3.20 Tampilan Menu Pengujian Uji U-Turn	74
Gambar 3. 21 Flowchart Menu Uji U-Turn	75
Gambar 3.22 Tampilan Hasil Akhir.....	78
Gambar 3.23 Perancangan Patok Tiang Pembatas Sensor Getar.....	81
Gambar 3.24 Desain Patok/Tiang Pembatas Sensor Ultrasonik	82
Gambar 3.25 Blok Diagram Lintasan Uji Keseimbangan	83
Gambar 3.26 Blok Diagram Lintasan Uji Zig-Zag	84
Gambar 3.27 Blok Diagram Lintasan Uji Uturn / Putar Balik.....	86
Gambar 4.1 Proses Upload Program Pada Arduino Ide.....	88
Gambar 4.2 Arduino Ide Berhasil Upload Program Ke Wemos D1 Mini	89
Gambar 4.3 Ketika Sensor Tidak Menerima Getaran.....	91
Gambar 4.4 Ketika Sensor Menerima Getaran	92

Gambar 4.5 Ouput Jarak Pada Objek Dinding.....	94
Gambar 4.6 Pengukuran Menggunakan Penggaris	95
Gambar 4.7 Percobaan Pada Aplikasi (Kiri) Dan Stopwatch Pada Handphone (Kanan).....	98
Gambar 4. 8 Percobaan Kedua Skenario Patok Tersenggol	102
Gambar 4.9 Aplikasi Untuk Menguji Baterai	107
Gambar 4.10 Uji Lintasan Keseimbangan Dengan Skenario D.....	110
Gambar 4.11 Uji Lintasan Zig-Zag Dengan Skenario D	112
Gambar 4.12 Uji Lintasan U-Turn Dengan Skenario D	113
Gambar 4.13 Hasil Akhir Pengujian Pada Aplikasi	114



DAFTAR TABEL

	Halaman
Tabel 2. 1 Spesifikasi Wemos D1 Mini	16
Tabel 2.2 Spesifikasi Sensor Getar Sw-420.....	19
Tabel 2.3 Spesifikasi Sensor Ultrasonik Hc-Sr04.....	21
Tabel 2.4 Pin Sensor Ultrasonik Hc-Sr04	21
Tabel 2.5 Kecepatan Rambat Bunyi Dalam Medium Tertentu.....	22
Tabel 2.6 Spesifikasi Raspberry Pi 3 Model B	26
Tabel 2.7 Qos Level	30
Tabel 3.1 Tabel Komponen Menu Utama.....	58
Tabel 3.2 Tabel Komponen Menu Koneksi Broker	61
Tabel 3.3 Tabel Komponen Menu Pengujian Uji Keseimbangan.....	64
Tabel 3.4 Tabel Komponen Menu Pengujian Uji Zig-Zag	69
Tabel 3. 5 Tabel Komponen Menu Pengujian Uji U-Turn	74
Tabel 3.6 Tabel Komponen Menu Hasil Akhir.....	78
Tabel 4.1 Hasil Pengujian 20 Buah Wemos D1 Mini.....	89
Tabel 4. 2 Percobaan Sensor Ultrasonik 1 Dan 2.....	93
Tabel 4.3 Percobaan Pengukur Kecepatan.....	97
Tabel 4.4 Percobaan Komunikasi Data Patok Dengan <i>Broker</i> Saat Kendaraan Melintas.....	100
Tabel 4.5 Percobaan Komunikasi Data Patok Dengan <i>Broker</i> Saat Tersenggol	100
Tabel 4.6 Hasil Pengujian Dengan Ap Andromax M2y	103
Tabel 4.7 Pengujian Baterai Pada Wemos	106
Tabel 4. 8 Uji Keseimbangan.....	109
Tabel 4.9 Uji Zig-Zag	111

Tabel 4. 10 Patok Tersenggol Dan Jatuh 113

Tabel 4. 11 Pengujian Secara Keseluruhan..... 115



BAB I

PENDAHULUAN

1.1 Latar Belakang

Indonesia merupakan negara hukum (UUD 1945 Pasal 1 Ayat 3), dimana segala sesuatu yang dilakukan memiliki hukum yang berlaku. Hukum di Indonesia didasarkan pada Undang-Undang tahun 1945 yang memuat norma-norma atau aturan-aturan yang harus ditaati dan dilaksanakan. Pada akhir-akhir ini banyak sekali warga negara Indonesia yang melanggar peraturan Undang-Undang dengan tidak memiliki Surat Ijin Mengemudi, namun tetap mengendarai kendaraan bermotor di jalan raya. Padahal UU Nomor 22 Tahun 2009 Bab 1 Pasal 1 Ayat 23 mengatakan , “Pengemudi adalah orang yang mengemudikan Kendaraan Bermotor di Jalan yang telah memiliki Surat Ijin Mengemudi”.

SIM (Surat Ijin Mengemudi) adalah bukti registrasi dan identifikasi yang diberikan oleh POLRI (Kepolisian Negara Republik Indonesia) kepada seseorang yang telah memenuhi persyaratan administrasi, sehat jasmani dan rohani, memahami peraturan lalu lintas dan terampil mengemudikan kendaraan bermotor. Berdasarkan Pasal 211(2) PP 44 / 93, golongan SIM dibedakan menjadi 6 : SIM A, SIM A Khusus, SIM B1, SIM B2, SIM C, SIM D. (<https://www.polri.go.id/layanan-sim.php>)

Untuk mendapatkan SIM, pemohon diwajibkan untuk lulus dari dua tes yang diberikan oleh POLRI (Kepolisian Negara Republik Indonesia) yaitu: tes teori dan tes praktik. Pada tes praktik, pemohon diharuskan melewati beberapa tipe rintangan yang sudah dirancang untuk kemahiran dan ketangkasaran pengendara

motor di jalan raya, diantaranya: Uji Kesimbangan, Uji Slalom (Zig Zag), Uji Membentuk Angka 8, Uji Reaksi, dan Uji Berbalik Arah (U Turn). Dalam proses penilaian petugas mengawasi dan menilai dari hasil yang dilihatnya, tidak menutup kemungkinan bahwa pendapat petugas satu dengan petugas yang lain sangat berbeda-beda dengan hasil ujian praktik sebenarnya.

Seperti penelitian yang telah dilakukan oleh Aron Octavianus Siahaan pada tahun 2017 yang berjudul Rancang Bangun Aplikasi Penilaian Ujian Praktik SIM (Surat Ijin Mengemudi) C Berbasis Arduino Mega2560, dimana sistem ujian praktik pembuatan SIM C yang sudah ter-komputerisasi dan penilaian sudah ditentukan secara otomatis berdasarkan pada kestabilan kecepatan kendaraan serta berapa kali pemohon menyentuh patok pembatas jalan yang sudah ditentukan. Sistem ini masih menggunakan pengiriman data secara serial menggunakan arduino mega.

Komunikasi wireless adalah sebuah pertukaran informasi antara dua titik tanpa hubungan langsung seperti kabel yang saling bertemu. Umumnya komunikasi wireless digunakan untuk mengirim informasi dari jarak jauh seperti monitoring dan sebagainya. MQTT (*Message Queuing Telemetry Transport*) adalah salah satu protokol komunikasi pesan yang ringan *non-wired* berbasis *publish/subscribe* yang dirancang khusus untuk komunikasi antar perangkat berdaya rendah.

Di era ini banyak sekali perangkat-perangkat yang dapat dihubungkan dengan wireless, dengan adanya wireless komunikasi antar perangkat dapat berjalan lebih mudah dibandingkan dengan non-wireless. Wireless sendiri dapat dijalankan dengan adanya koneksi internet atau local (non-internet) sebagai contoh

seperti sistem keamanan dalam rumah dimana apabila ada orang tidak dikenal masuk kedalam rumah, maka sistem akan memberi info kepada pemilik rumah melalui aplikasi wireless (Muhammad Risyat Nashrullah – 2018).

Pada sistem penilaian ujian SIM C sudah di lakukan oleh Aron, sistem masih menggunakan kabel sebagai komunikasi yang terhubung dari sensor-sensor sehingga cukup menyulitkan pada saat instalasi pemasangan sistem. Banyaknya kabel yang berserakan dilintasan ujian juga dapat menimbulkan kerusakan karena terlindas oleh kendaraan bermotor peserta ujian. Penggunaan wireless pada sistem penilaian memiliki beberapa keunggulan, diantaranya :

1. Instalasi pemasangan lebih mudah karena tidak perlu memeriksa setiap kabel yang terhubung ke sensor atau Mikrokontroller.
2. Mengurangi resiko kabel yang rusak dikarenakan terlindas kendaraan peserta ujian.
3. Mengurangi penggunaan kabel yang berlebihan.

Pada penelitian tugas akhir ini, penulis ingin mengembangkan sistem penilaian ujian praktik SIM C yang pengiriman datanya secara serial menjadi pengiriman data secara *wireless* agar dapat mengurangi penggunaan kabel yang terlalu banyak dan memudahkan proses instalasi pemasangan pada saat ujian praktik berlangsung.

1.2 Perumusan Masalah

Permasalahan yang akan dihadapi dalam pelaksanaan tugas akhir ini diantaranya adalah :

1. Bagaimana cara pengiriman data dari sensor yang terpasang pada Wemos melalui *wireless* ke sistem penilaian.
2. Bagaimana cara menerapkan protokol MQTT (*Message Queue Telemetry Transport*) pada sistem.

1.3 Batasan Masalah

Dalam perancangan dan pembuatan terdapat beberapa batasan masalah, antara lain :

1. Lintasan yang digunakan adalah lintasan uji keseimbangan, Zig Zag, dan berbalik arah (U Turn).
2. Sensor yang mendeteksi pelanggaran pada patok adalah sensor SW-420.
3. Sensor yang mengukur kecepatan kendaraan menggunakan sensor HC-SR04.
4. Mikrokontroler menggunakan Wemos D1 Mini.
5. Komunikasi data menggunakan protokol MQTT.
6. *Broker/server* menggunakan Raspberry Pi 3
7. Data diolah dan ditampilkan menggunakan aplikasi yang dibuat menggunakan aplikasi Microsoft Visual Studio 2017.

1.4 Tujuan

Berdasarkan rumusan masalah yang diuraikan diatas, maka tujuan penelitian ini :

1. Menemukan cara yang efektif dan effisien dalam pengiriman data dari Wemos D1 Mini ke Sistem Aplikasi Penilaian.

2. Menerapkan protokol MQTT pada Sistem Penilaian.

1.5 Manfaat

Manfaat penelitian ini adalah :

1. Dapat mempersingkat waktu ketika pemasangan/instalasi patok-patok pada pengujian.
2. Mengurangi penggunaan kabel yang terlalu banyak.

1.6 Sistematika Penulisan

Untuk mengetahui gambaran ringkas mengenai isi laporan tugas akhir serta untuk mempermudah pemahaman penelitian yang dilakukan, maka laporan tugas akhir ini dibagi menjadi beberapa bab antara lain.

BAB I : PENDAHULUAN

Bab ini berisi pendahuluan yang menjelaskan latar belakang permasalahan, perumusan masalah, tujuan penelitian, manfaat penelitian, batasan penelitian serta sistematika penulisan.

BAB II : STUDI PUSTAKA

Bab ini berisi tentang teori-teori yang mendukung penelitian yang dilakukan. Teori tersebut merupakan teori yang sudah ada dari penelitian sebelumnya. Pengambilan teori tersebut dapat berasal dari buku, jurnal, maupun materi dari internet.

BAB III : METODE PENELITIAN

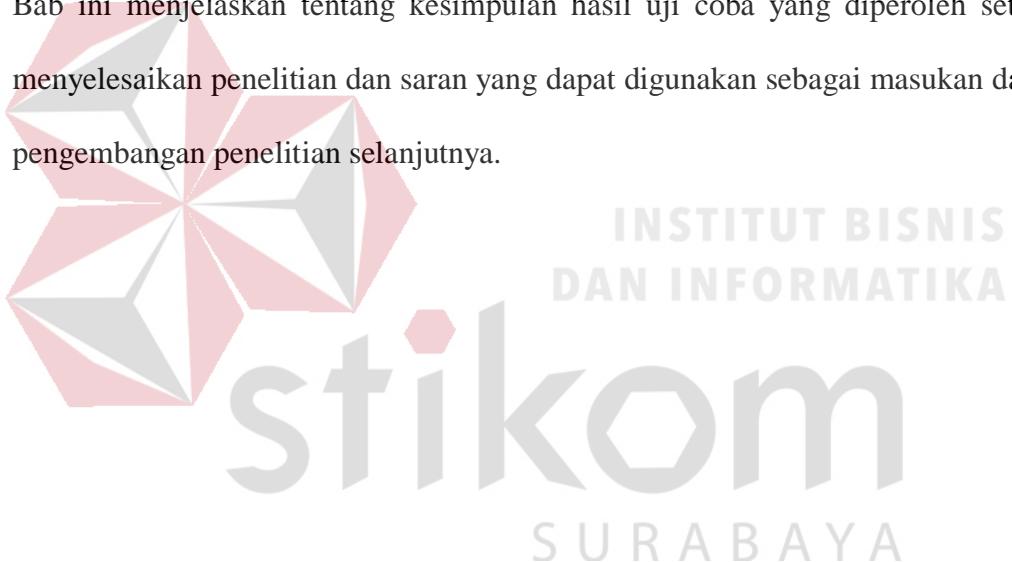
Bab ini berisi perancangan sistem tentang penelitian yang akan dibuat. Yaitu pembuatan sistem penilaian ujian praktik SIM C berbasis *Wireless* menggunakan protokol komunikasi pesan MQTT.

BAB IV : HASIL DAN PEMBAHASAN

Bab ini berisi tentang analisa sistem baik dari segi kebutuhan hardware dan software. Serta implementasi prototype dan pengujian prototype.

BAB V : PENUTUP

Bab ini menjelaskan tentang kesimpulan hasil uji coba yang diperoleh setelah menyelesaikan penelitian dan saran yang dapat digunakan sebagai masukan dalam pengembangan penelitian selanjutnya.



BAB II

STUDI PUSTAKA

2.1 Tinjauan Pustaka

Berikut ini adalah penelitian-penelitian yang berhubungan dengan Rancang Bangun Sistem Penilaian Ujian Praktik SIM C berbasis *Wireless* Menggunakan Protokol Komunikasi Pesan MQTT, penelitian tersebut diantaranya,

1. Siahaan, Aron Octavianus dengan judul “Rancang Bangun Aplikasi Penilaian Ujian Praktik SIM (Surat Izin Mengemudi) C Berbasis Arduino Mega2560”, STIKOM Surabaya 2017. Pada jurnal ini dibuat rancang bangun aplikasi sistem penilaian pada ujian praktik SIM C yang dimana menggunakan Arduino Mega2560 dan Visual basic. Pada aplikasi ini Arduino Mega2560 berkomunikasi secara serial dengan sensor getar sw-402 dan sensor ultrasonik hc-sr04. Aplikasi juga dapat menampilkan pelanggaran pada patok/tiang pembatas lintasan yang ditabrak/disentuh oleh peserta ujian. Aplikasi juga menghitung kecepatan kendaraan dan menampilkannya.
2. Hudan Abdur Rochman, Rakhmadhany Primananda, Heru Nurwasito dengan judul “Sistem Kendali Berbasis Mikrokontoler Menggunakan Protokol *MQTT*”, Universitas Brawijaya 2017. Pada jurnal ini dibuat sebuah sistem kendali berbasis mikrokontroler yang menggunakan protokol *MQTT* yang digunakan untuk monitoring suatu tempat tertentu seperti rumah, yang biasa disebut dengan *smarthome*. Perangkat-perangkat yang

3. dibutuhkan seperti sensor dht11 yang digunakan untuk membaca kondisi suhu, sensor ldr untuk intensitas cahaya dan led yang akan merepresentasikan device-device yang akan dikontrol pada *smarthome*.

2.2 Surat Ijin Mengemudi (SIM)

Menurut hukum dan undang-undang yang ada di Indonesia, setiap pengemudi kendaraan bermotor diwajibkan memiliki SIM (Surat Ijin Mengemudi). Dan pihak kepolisian sudah menfasilitasi dengan pembuatan SIM dengan syarat yang telah ditentukan. Terdapat fungsi dan peranan sebagai berikut :

1. Sebagai sarana identifikasi/jati diri seseorang.
2. Sebagai alat bukti.
3. Sebagai sarana upaya paksa.
4. Sebagai sarana pelayanan masyarakat.

SIM juga digolongkan menjadi beberapa bagian :

1. Golongan SIM A

SIM untuk kendaraan bermotor roda 4 dengan berat tidak lebih dari 3.500 kg.

2. Golongan SIM A Khusus

SIM untuk kendaraan bermotor roda 3 dengan karoseri mobil yang digunakan untuk angkutan orang/barang.

3. Golongan SIM B1

SIM untuk kendaraan bermotor dengan berat lebih dari 1000 kg.

4. Golongan SIM B2

SIM untuk kendaraan bermotor yang menggunakan kereta tempelan dengan berat lebih dari 1000 kg.

5. Golongan SIM C

SIM untuk kendaraan bermotor roda 2 dengan kecepatan lebih dari 40 km/jam

6. Golongan SIM D

SIM khusus untuk pengemudi yang menyandang disabilitas.

2.2.1 Syarat Pembuatan SIM C

Berikut adalah syarat-syarat yang harus dipenuhi untuk pembuatan SIM (Surat Ijin Mengemudi) :

1. Minimal berusia 17 tahun.
2. Sehat jasmani dan rohani.
3. Sudah memiliki KTP (Kartu Tanda Penduduk).
4. Mengisi formulir permohonan.
5. Mengikuti ujian teori dan ujian praktik.
6. Membayar biaya administrasi.

2.2.2 Tatacara Pembuatan SIM C

Berikut alur dan tatacara pembuatan SIM (Surat Ijin Mengemudi) C :

1. Pemeriksaan kesehatan.
2. Membayar biaya administrasi.
3. Mengisi formulir pendaftaran.
4. Ujian teori.
5. Ujian praktik.

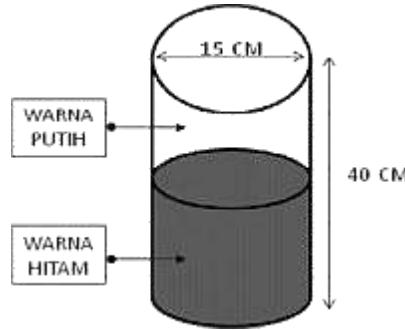
2.2.3 Ujian Praktik SIM C

Sebelum pengujian, ada beberapa syarat yang harus dipenuhi agar pengujian berjalan dengan baik dan benar. Berikut standar prasarana lapangan untuk pengujian :

1. Penguji terdiri dari dua orang atau lebih untuk seorang peserta uji sepeda motor di masing-masing materi uji.
2. Permukaan lapangan aspal atau beton tidak bergelombang, tidak licin dan tidak berpasir.
3. Pelaksanaan ujian praktik di lapangan praktik dan jalanan umum dalam satu wilayah hukum.
4. Sebelum melakukan pengujian disarankan kepada peserta tes kendaraan yang akan digunakan uji praktik.
5. Standar pengukuran sepeda motor bagian fisik ban depan dan belakang terluar pada kendaraan (kecuali) stang stir serta kaca spion.
6. Lebar lapangan ujian minimum 50 m.
7. Panjang lintasan jalan/lapangan ujian minimum 100 m.
8. Ruang tunggu peserta uji SIM.

Ada beberapa sarana untuk ujian praktik, diantaranya sebagai berikut:

1. Patok uji dengan ukuran
 - a. Tinggi patok 40 cm.
 - b. Diameter 10 s.d 15 cm.
 - c. Warna orange/hitam dilengkapi *scotch light*.



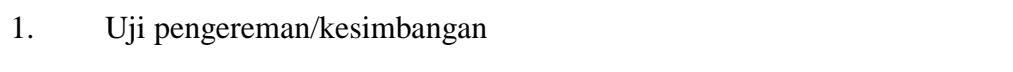
Gambar 2.1 Patok Untuk Ujian Praktik SIM C

(Sumber : <http://ngada.org/bn279-2012lmp.htm>)

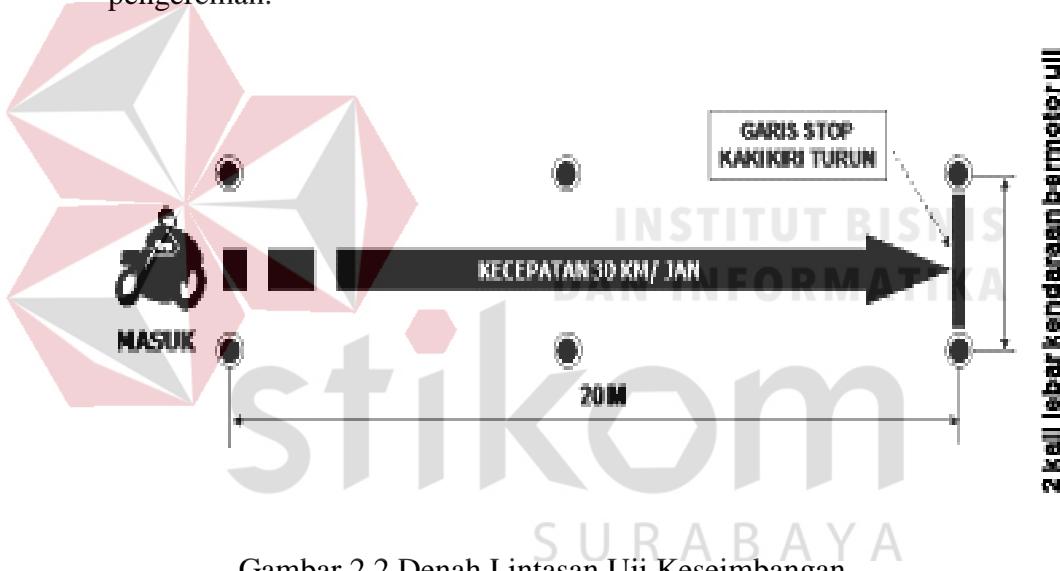
2. Garis-garis lapangan terbuat dari cat berwarna putih, khusus untuk ujian angka delapan di atas garis agar ditempatkan patok dengan jarak antar patok masing-masing 1,5 m kali panjang kendaraan bermotor uji.
3. Meja pengujji.
4. Nomor peserta uji.
5. Rompi peserta uji.
6. Helm.
7. Peluit.
8. Jas hujan.
9. Sepeda motor uji.
10. Pengeras suara.
11. Kaca mata teduh (hitam).
12. Stop watch.
13. Belangko pengujian.
14. Alat tulis (pulpen/spidol).

Materi ujian praktik SIM C dibagi menjadi 5, yaitu:

1. Uji pengerman/kesimbangan



- a. Menjalankan sepeda motor dengan kecepatan stabil 30 km/jam, dengan perseneling 2 berhenti pada garis *stop* dengan teknik penggereman kombinasi yang lebih dominan rem tangan bersamaan dengan rem belakang (kaki) untuk mengimbangi rem depan, kaki kiri turun dan palingkan kepala ke kanan belakang konfirmasi keselamatan.
- b. Jarak dari start sampai finish adalah 9 buah patok dari ukuran panjang kendaraan uji tambah $\frac{1}{2}$ panjang kendaraan uji (1,5 m) sedang lebar patok yang dilintasi adalah 2 x lebar kendaraan bermotor uji untuk lebar lintasan penggereman.



Gambar 2.2 Denah Lintasan Uji Keseimbangan

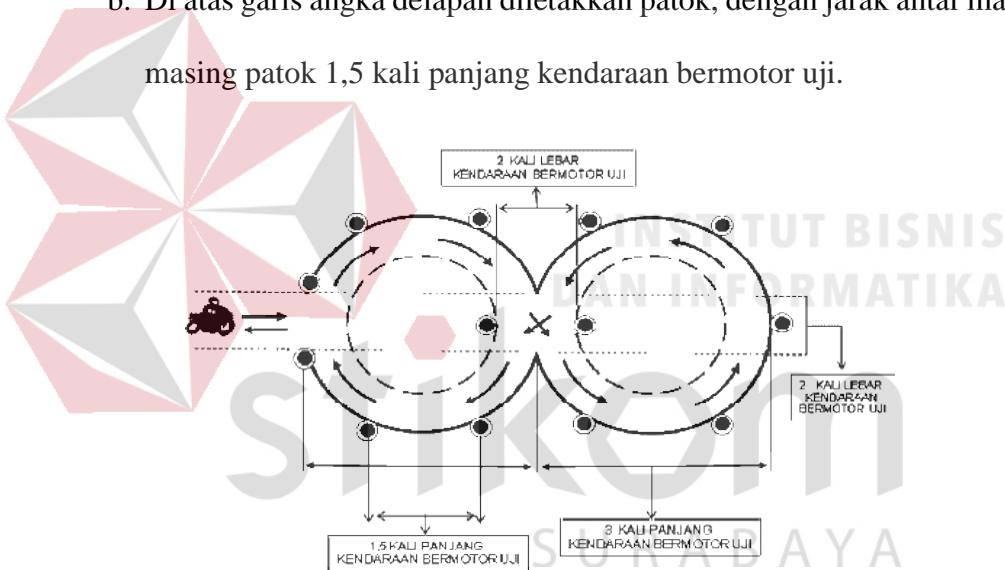
(Sumber : <http://ngada.org/bn279-2012lmp.htm>)

2. Uji slalom/Zig-zag
 - a. Menjalankan sepeda motor slalom/zig-zag melintasi patok (kerucut) dengan kecepatan 10 km/jam, jarak antar patok 1,5 kali panjang kendaraan bermotor uji dan jari-jari tangan tidak menekan tangki kopling/pengereman sebelum titik berhenti yang ditentukan.
 - b. Kemudian dilanjutkan slalom/zig-zag dengan kecepatan stabil, jarak patok satu dengan yang satu 3 kali panjang kendaraan bermotor uji dan

berhenti pada garis stop, dengan teknik penggereman kombinasi rem depan lebih dominan dan rem belakang mengimbangi asumsi (70%/30 persen), kaki kiri menapak di jalan, Kepala memalingkan ke kanan belakang konfirmasi keselamatan.

3. Uji membentuk angka delapan

- menjalankan sepeda motor di dalam lingkaran 3 kali membentuk angka 8 (delapan), mengikuti petunjuk arah, tidak berhenti dan kaki tidak menginjak lapangan serta jari-jari tangan tidak menarik kopling/rem: dan
- Di atas garis angka delapan diletakkan patok, dengan jarak antar masing-masing patok 1,5 kali panjang kendaraan bermotor uji.



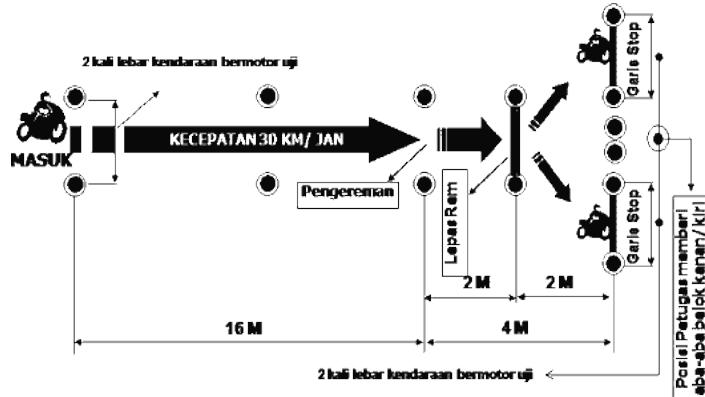
Gambar 2.3 Denah Lintasan Uji Membentuk Angka 8

(Sumber : <http://ngada.org/bn279-2012lmp.htm>)

4. Uji reaksi rem menghindar

Konfirmasi keselamatan pada saat menjalankan sepeda motor dengan kecepatan stabil perseneleng 2 atau 3, kemudian melakukan penggereman pada Garis Kuning atau patok, lepas rem pada patok atau Garis Hijau, lalu membelok sesuai petunjuk dari petugas, serta berhenti pada garis stop dengan teknik penggereman kombinasi untuk rem belakang

mengimbangi dan untuk rem depan dominan, kaki kiri turun dan palingkan kepala ke kanan belakang.

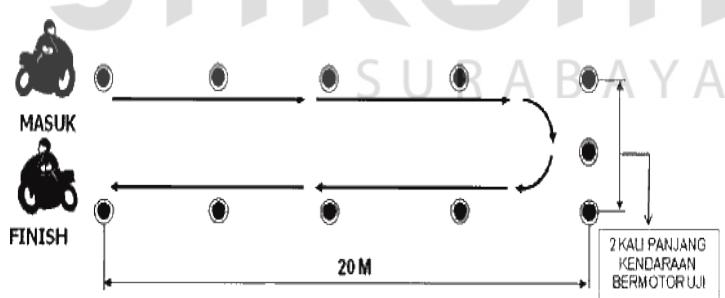


Gambar 2.4 Denah Lintasan Uji Rem Menghindar

(Sumber : <http://ngada.org/bn279-2012lmp.htm>)

5. Uji berbalik arah membentuk huruf U (Turn)

Konfirmasi keselamatan pada saat akan menjalankan sepeda motor memutar dengan membentuk huruf U di jalan sempit yang lebarnya 2 kali panjang kendaraan bermotor uji, tanpa menginjakkan kaki ke lapangan dan pandangan tertuju ke arah yang akan dituju.

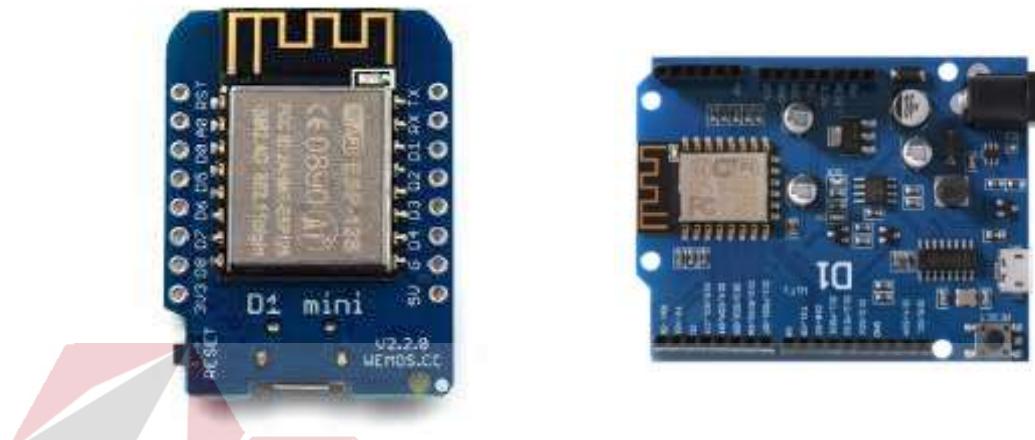


Gambar 2.5 Denah Lintasan Uji U Turn

2.3 Wemos

Wemos adalah sebuah *microcontroller* yang dikembangkan berbasis modul ESP-8266. Wemos sendiri dibuat untuk mengatasi solusi dari mahalnya sebuah sistem *wireless* berbasis *microcontroller* lainnya. Dengan menggunakan Wemos

biaya yang dikeluarkan untuk membangun system *microcontroller* berbasis WiFi lebih mudah dan murah dibandingkan dengan membangun sistem WiFi menggunakan *microcontroller* Arduino Uno dan WiFi Shield.



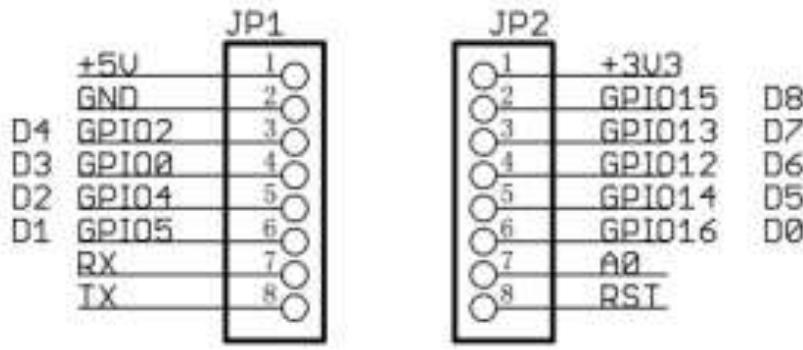
Gambar 2.6 Wemos D1 Mini (kiri) dan Wemos D1 R1 (kanan)

(Sumber : <https://www.fabtolab.com/wemos-d1-mini-V2.2.0>)

2.3.1 Wemos D1 Mini

Wemos D1 Mini merupakan *microcontroller* yang sudah *built-in* dengan modul ESP-8266. Wemos D1 Mini memiliki 16 Pin, 3 pin untuk daya/power 3,3V, 5V dan 0V, 10 pin untuk *input* dan *output*, dan 3 pin terakhir untuk rx, tx dan rst. Lalu juga terdapat tombol kecil yang digunakan untuk mereset Wemos D1 Mini.

Berikut ini adalah spesifikasi dan Pin I/O dari Wemos D1 Mini :



Gambar 2.7 Pin I/O Wemos D1 Mini

Tabel 2. 1 Spesifikasi Wemos D1 Mini

Microcontroller	ESP-8266EX
Operating Voltage	3.3V
Digital I/O Pins	11
Analog Input Pins	1 (Max input: 3.2 V)
Clock Speed	80MHz/160MHz
Flash	4M bytes
Length	34.2mm
Width	25.6mm
Weight	3g

Dapat dilihat dari tabel diatas bahwa Wemos dapat diaktifkan melalui koneksi USB atau dengan daya *external*. External(non-USB) daya dapat berasal dari adaptor DC atau baterai. Adaptor ini nantinya akan ditancapkan ke *pin 5V* pada Wemos D1 Mini.

Board Wemos D1 Mini dapat beroperasi pada rentang tegangan 3,3V – 7V.

Jika tegangan kurang dari 3,3V, maka Wemos tidak dapat menyala ataupun beroperasi dengan baik dikarenakan kurangnya tegangan yang diterima. Dan sebaliknya jika Wemos menerima tegangan lebih dari 5V, maka Wemos akan panas dan terbakar yang mengakibatkan kerusakan parah pada *board*.

Pin listrik pada Wemos D1 Mini :

5V: Dapat digunakan sebagai *input* atau *output* daya sebesar 5V. Jika Wemos sudah menggunakan USB sebagai *input* tegangan maka *pin* ini berfungsi sebagai *output* tegangan, bila daya yang digunakan bukan dari USB maka *pin* ini dapat digunakan sebagai *input* tegangan.

3,3V: Dapat digunakan sebagai *input* atau *output* daya sebesar 3,3V. Jika Wemos sudah menggunakan USB sebagai *input* tegangan maka *pin* ini berfungsi sebagai *output* tegangan, bila daya yang digunakan bukan dari USB maka *pin* ini dapat digunakan sebagai *input* tegangan.

GND: Pin untuk menetralkan *noise* atau juga berperan sebagai 0V.

Wemos D1 Mini memiliki Clock speed 160Mhz, koneksi WiFi, dan memori *flash* sebesar 4 MB. Dalam operasi kerjanya Wemos ini dapat bekerja direntang suhu antara 40 °C - 125 °C. Walaupun modul ini bukan Arduino, namun modul ini mendukung pemrograman menggunakan Arduino IDE, beserta library dan fungsi-fungsinya yang lain. Untuk koneksi ke laptop atau ke satu daya microcontroller Wemos ini menggunakan konektor micro USB yang umum digunakan untuk kabel data smartphone Android.

2.3.2 *Chipset* pada Wemos D1 Mini

1. **Chipset Esp 8266**

Chipset Esp-8266 adalah *microchip WiFi* dengan harga yang murah lengkap dengan pengaturan TCP/IP dan di produksi di Shanghai-China. Chip ini memungkinkan microcontroller untuk terhubung ke jaringan WiFi pada frekuensi 2.4GHz dan membuat koneksi TCP/IP hanya dengan command yang sederhana seperti gaya hayes. Dengan clock 80 MHz chip ini dibekali dengan 4MB external RAM, mendukung format IEEE 802.11 b/g/n sehingga tidak menyebabkan interferensi bagi yang lain. Dari sisi keamanan chip ini sudah cukup aman digunakan karena mendukung enkripsi WEP dan WPA. Chip ini mempunyai 16 GPIO pin yang bekerja pada 3.3 Volt, dan 1 pin ADC dengan resolusi 10 bit.

2. **Chipset CH340**

Chipset CH340 adalah sebuah chip yang berfungsi untuk mengubah USB menjadi serial interface. Sebagai contohnya adalah aplikasi USB converter IrDA atau USB converter to printer. Dalam mode serial interface, chip ini digunakan untuk memperbesar sinyal asynchronous serial interface komputer atau mengubah perangkat serial interface umum untuk berhubungan dengan bus USB langsung.

2.4 **Sensor Getar SW-420**

Sensor getar adalah suatu alat yang berfungsi untuk mendeteksi adanya getaran dan mengubahnya kedalam sinyal listrik. Sensor getar SW-420 merupakan sensor getar digital yang akan menghasilkan keluaran logika *HIGH* pada saat mendeteksi getaran.



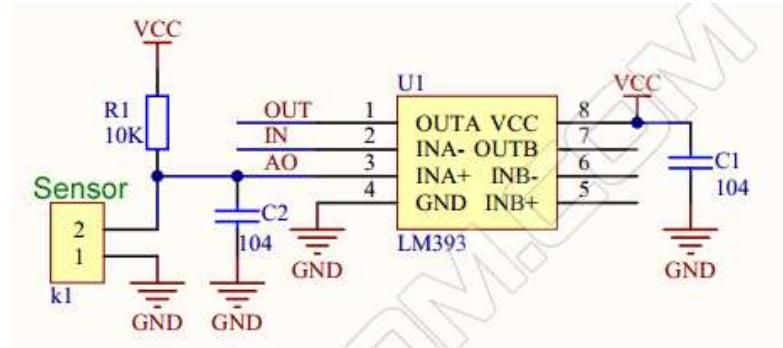
Gambar 2.8 Sensor Getar SW-420

(Sumber : <https://www.gadgetkudus.com/sensor-getar-sw-420/>)

Inti dari sensor ini adalah komponen pendeksi getar SW-420 yang bereaksi terhadap getaran dari berbagai sudut. Pada kondisi statis/tanpa getaran, sensor ini berfungsi seperti saklar yang berada pada kondisi menutup (*normally closed*) dan bersifat konduktif, sebaliknya pada kondisi terpapar getaran, saklar akan membuka/menutup dengan kecepatan pengalihan (*switching frequency*) proporsional dengan kekerapan getaran. Tingkat sensitifitas pendeksi dapat diatur dengan memutar potensiometer (*variable resistor*) yang terpasang di sensor ini. Berikut tabel spesifikasi dan skematik dari sensor getar SW-420 :

Tabel 2.2 Spesifikasi Sensor getar SW-420

Chipset	LM393
Jenis	SW-420
Tegangan	3.3 – 5V DC
Ukuran	3.2 x 1.4 mm
Output	Digital (0 atau 1)



Gambar 2.9 Skematik sensor getar SW-420

2.5 Sensor Ultrasonik HC-SR04

Sensor ultrasonik HC-SR04 adalah salah satu jenis sensor yang berfungsi sebagai pengirim, penerima, dan pengontrol gelombang ultrasonik. Sensor ini bisa digunakan untuk mengukur jarak benda dari 2cm - 4m dengan akurasi 3mm. Sensor ini memiliki 4 pin, pin Vcc, Gnd, Trigger, dan Echo. Pin Vcc untuk listrik positif dan Gnd untuk 0V-nya. Pin Trigger untuk trigger keluarnya sinyal dari sensor dan pin Echo untuk menangkap sinyal pantul dari benda.



Gambar 2.10 Sensor Ultrasonik HC-SR04

(Sumber : <https://www.makerlab-electronics.com/product/ultrasonic-sensor-hc-sr04/>)

Sinyal yang dipancarkan oleh pemancar ultrasonik dengan frekuensi tertentu dan dengan durasi waktu tertentu akan merambat sebagai gelombang bunyi dengan kecepatan sekitar 340m/s. Ketika sinyal memantul suatu benda, pantulan akan kembali ke alat penerima dan akan diproses untuk menghitung jarak benda tersebut. Berikut tabel spesifikasi dan pin dari sensor ultrasonik HC-SR04 :

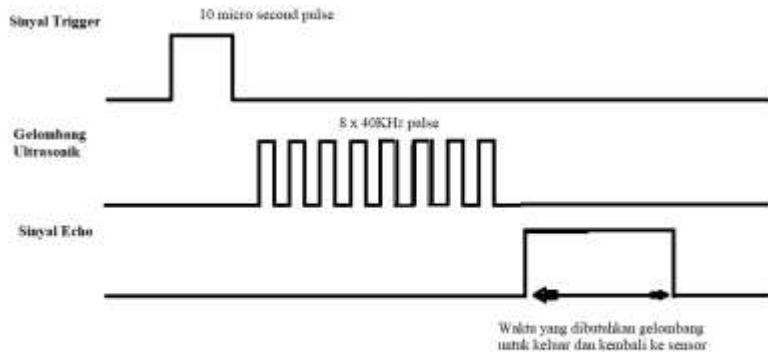
Tabel 2.3 Spesifikasi sensor ultrasonik HC-SR04

Sensor	Ultrasonik HC-SR04
Tegangan	5V
Arus	15 mA
Frekuensi	40 Hz
Jarak minimal	2 cm
Jarak maksimal	4 m
Sudut efektif	15 derajat
Triger sinyal input	10µS TTL pulse
Echo sinyal output	Triger sinyal input dan jarak yang tepat
Ukuran	45 x 20 x 15 mm

Tabel 2.4 Pin Sensor ultrasonik HC-SR04

Vcc	Tegangan 5V
Trig	Triger pin
Echo	Echo pin
GND	0V atau ground pin

Berikut adalah cara kerja pengambilan data dari sensor Ultrasonik Hc-SR04, seperti pada gambar 2.11 :



Gambar 2.11 Proses Penerimaan Data Sensor Ultrasonik

1. Ketika Pin Triger pada sensor ultrasonik menerima tegangan positif $\geq 10 \mu\text{s}$, Sensor akan mengeluarkan gelombang ultrasonik $8 \times 40\text{KHz}$. Gelombang merambat dengan kecepatan 344m/s , seperti pada tabel 2.5 berikut :

Tabel 2.5 Kecepatan Rambat Bunyi Dalam Medium Tertentu

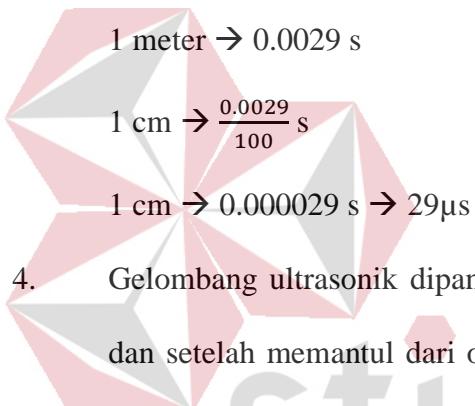
Medium	Ultrasonik HC-SR04
Udara pada temperatur -20°C	319.3 meter/detik
Udara pada temperatur 0°C	331.8 meter/detik
Udara pada temperatur 10°C	337.4 meter/detik
Udara pada temperatur 20°C	343.8 meter/detik
Udara pada temperatur 30°C	349.6 meter/detik
Gas hidrogen	1284 meter/detik
Gas O ₂	316 meter/detik

2. Pin echo akan bernilai HIGH ketika gelombang ultrasonik mulai dipancarkan dan merambat. Ketika gelombang memantul dan kembali dari objek ke receiver, pin echo akan bernilai LOW.
3. Rumus mencari jarak adalah $s = v * t$. Karena $v = 344 \text{ m/s}$ yang artinya dalam 1 detik dapat menempuh 344 meter, maka untuk mencari 1 meter :

$$s = v * t$$

$$t = \frac{s}{v} = \frac{1 \text{ m}}{344 \text{ m/s}} = \frac{1}{344} \text{ s}$$

$$t = \frac{1}{344} \text{ s} = 0.0029 \text{ s}$$



4. Gelombang ultrasonik dipancarkan dua kali, Ketika pertama dikeluarkan dan setelah memantul dari objek di depannya. Maka waktu tempuh untuk jarak 1 cm adalah $29\mu\text{s} * 2 = 58 \mu\text{s}$
5. Jadi dapat disimpulkan untuk mencari jarak (cm) = waktu / 58

2.6 Raspberry Pi 3

Raspberry Pi 3 adalah komputer *Single Board Circuit* (SBC) yang memiliki ukuran kecil setara dengan ukuran ATM, KTP atau Kartu Kredit. Perangkat ini dapat digunakan untuk menjalankan program perkantoran, permainan, pemutar video beresolusi tinggi dan lain sebagainya. Raspberry Pi 3 ini sangat cocok untuk developer karena ukurannya yang kecil tetapi memiliki kinerja yang baik. Selain itu kelebihan Raspberry Pi 3 dibanding dengan mikrokontroler lain yaitu mempunyai Port I/O, display HDMI, koneksi USB dan banyak kelebihan lainnya. Raspberry Pi 3 dikembangkan oleh yayasan

nirlaba, Raspberry Pi Foundation, yang anggotanya merupakan ahli komputer dari Universitas Cambridge, Inggris. Raspberry Pi 3 *Board* ditunjukkan pada.



Gambar 2.12 Raspberry Pi 3

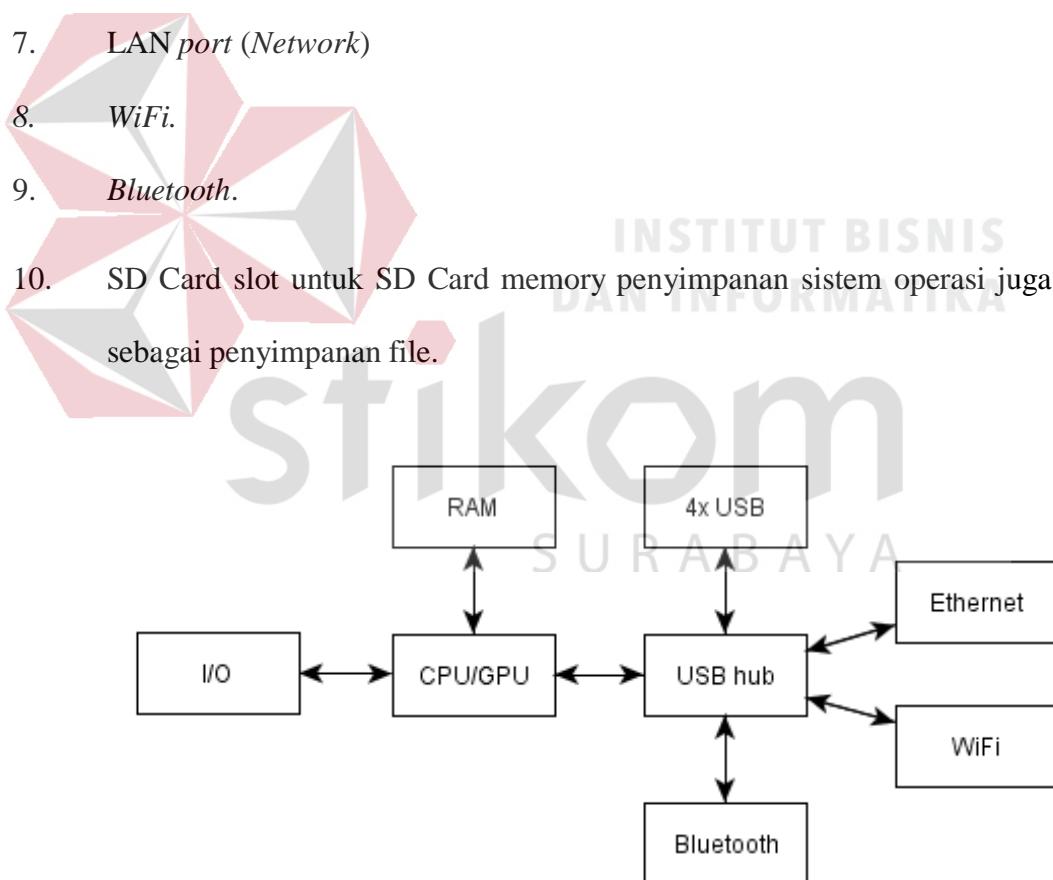
(Sumber : <https://www.raspberrypi.org/products/raspberry-pi-3-model-b/>)

Terdapat 2 versi dari Raspberry yaitu model A dan model B. Perbedaan dari model A dan model B adalah kapasitas penyimpanannya, model B menggunakan penyimpanan yang lebih besar dari pada model A. Selain itu model B juga sudah menggunakan Ethernet, Bluetooth dan Wi-Fi sebagai sarana komunikasi. Penyimpanan data tidak didesain untuk menggunakan cakram keras atau *solid-state drive*, melainkan menggunakan mikro SD untuk menjalankan sistem dan media penyimpanan dalam jangka panjang. Pengguna dianjurkan menggunakan Flashdisk atau penyimpanan lain untuk menyimpan file yang berukuran besar. Perangkat Raspberry Pi 3 ini adalah hasil perkembangan dari perangkat yang dahulu yaitu Raspberry Pi 2. Banyak sekali perubahan yang terjadi mulai dari RAM, processor dan lain sebagainya.

Blok Diagram Raspberry Pi 3 Model B ditunjukkan pada Gambar 2.12, mempunya *input* dan *output* antara lain:

1. HDMI, sebagai penghubung ke LCD Monitor yang mempunyai *port* HDMI atau dengan converter HDMI ke VGA untuk bisa digunakan pada

- LCD Monitor yang memiliki *port VGA*.
2. Audio *output* menggunakan jack 3,5 mm, sebagai output untuk speaker atau sejenisnya yang menggunkan jack 3,5 mm.
 3. 4 buah port USB.
 4. 40 *pin I/O*.
 5. CSI (*Camera Serial Interface*), digunakan khusus untuk kamera yang menggunakan CSI sebagai antarmuka.
 6. DSI (*Display Serial Interface*)
 7. LAN *port (Network)*
 8. WiFi.
 9. Bluetooth.
 10. SD Card slot untuk SD Card memory penyimpanan sistem operasi juga sebagai penyimpanan file.



Gambar 2.13 Blok diagram Raspberry Pi 3 Model B

2.6.1 Spesifikasi Raspberry Pi 3 Model B

Beberapa spesifikasi Raspberry Pi 3 Model B dapat dilihat pada Tabel 5.

Tabel 2.6 Spesifikasi Raspberry Pi 3 Model B

<i>Processor</i>	Broadcom BCM2387 <i>chipset</i> 1.2Ghz <i>Quad-Core</i> ARM Cortex-A53 802.11 b/g/n Wireless LAN and <i>Bluetooth</i> 4.1 (<i>Bluetooth</i> Classic abd LE)
<i>GPU</i>	<i>Dual Core VideoCore IV® Multimedia Co-Processor. Provides Open GL ES 2.0, hardware- accelerated OpenVG, and 1080p30 H.264 high- profile decode. Capable of 1Gpixel/s, 1.5Gtexel/s or 24GFLOPs with texture filtering and DMA infrastructure</i>
<i>Memory</i>	1GB LPDDR2
<i>Operating System</i>	Boots from Micro SD card, <ul style="list-style-type: none"> • Raspbian • Ubuntu MATE • Snappy Ubuntu Core • Windows 10 IoT Core • RISC OS • Debian • Arch Linux ARM • Android
Dimensions	85 x 56 x 17mm
Power	Micro USB socket 5 Volt 2.5A

2.7 MQTT (*Message Queuing Telemetry Transport*)

Message Queue Telemetry Transport (MQTT) adalah sebuah protokol yang digunakan untuk berkomunikasi *machine to machine (M2M)* yang berada pada

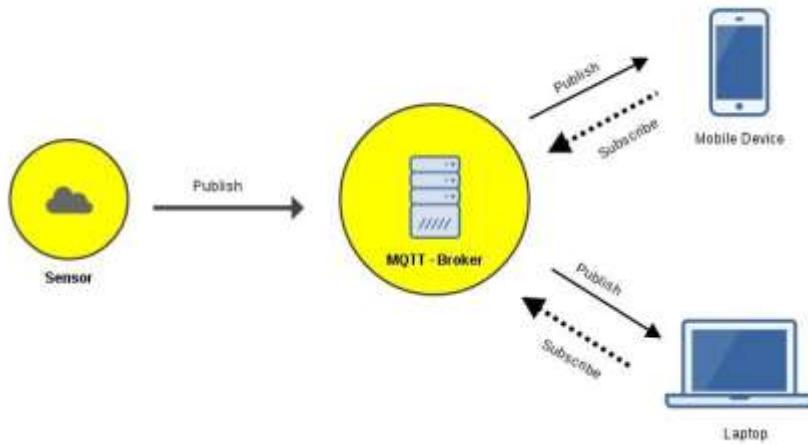
layer aplikasi. *MQTT* bersifat *lightweight* yang artinya dapat berkomunikasi mengirimkan pesan dengan ukuran header yang kecil (minimum 2 bytes) untuk setiap jenis data, hal ini memungkinkan dapat bekerja di lingkungan yang terbatasnya sumber daya listrik dan kecilnya *bandwidth*. Protokol ini adalah jenis protokol data-agnostic yang artinya bisa mengirimkan data apapun seperti data binary, text bahkan XML ataupun JSON.

2.7.1 *Publish and Subscribe*

MQTT mengimplementasikan *publish/subscribe* untuk berkomunikasi. *Publish/subscribe* menggunakan dua *client*, *client* pertama sebagai (“*Publisher*”) dan *client* kedua sebagai (“*Subscriber*”). *Publisher* adalah *client* yang bertugas mengirimkan pesan ke *client* lainnya, sedangkan *subscriber* adalah *client* yang bertugas menerima pesan. *Publisher* dan *subscriber* hanya mengetahui bahwa pesan yang dikirim/diterima sudah difilter dengan *topic* yang sesuai oleh broker/server.

Tiga keuntungan dari publish/subscribe, sebagai berikut :

1. *Publisher* dan *subscriber* tidak mengetahui keadaan satu sama lain, seperti *IP Adress* dan *port* karena ada broker diantara mereka (*Space decoupling*).
2. *Publisher* dan *subscriber* tidak diharuskan untuk tersambung pada saat yang bersamaan (*Time decoupling*) dan,
3. Proses dari kedua client tidak akan berhenti pada saat mengirim dan menerima pesan (*Synchronization decoupling*).



Gambar 2.14 Alur *Publish* dan *Subscribe* MQTT

(Sumber : <https://medium.com/pemrograman/mengenal-mqtt-998b6271f585>)

Sistem umum MQTT seperti pada Gambar 2.13 membutuhkan dua komponen perangkat lunak utama yaitu:

- MQTT Client**, yang nantinya berperan sebagai *client* yang bisa diperintahkan untuk *subscribe* ataupun *publish*. Untuk Arduino bisa memakai library *pubsubclient*, sedangkan untuk Raspberry pi bisa menggunakan library *paho mqtt* yang berjalan menggunakan Bahasa pemrograman python.
- MQTT Broker**, yang berfungsi untuk menangani publish dan subscribe data dengan kata lain sebagai server atau penghubung dari 2 buah *client* yang saling berinteraksi. Untuk platform Node.js anda bisa memakai broker mosca sedangkan untuk platform yg lain banyak broker tersedia seperti mosquitto, HiveMQ dll.

MQTT mempunyai 14 tipe sinyal kontrol seperti berikut:

- a. Connect — Client menyambungkan ke *server/broker*.

- b. *Connack* — Koneksi dari *Acknowledgement*
- c. *Publish* — Mempublikasikan pesan
- d. *Puback* — *QoS* 1 merespon ke pesan yang dipublikasikan
- e. *Pubrec* — Bagian pertama dari alur pengiriman pesan *QoS* 2
- f. *Pubrel* — Bagian kedua dari alur pengiriman pesan *QoS* 2
- g. *Pubcomp* — Bagian terakhir dari alur pengiriman pesan *QoS* 2
- h. *Subscribe* — Sebuah pesan yang digunakan *client* untuk mengsubscribe topik
- i. *Suback* — *Acknowledgement* dari pesan yang di subscribe
- j. *Unsubscribe* — Sebuah pesan yang digunakan *client* untuk mengunsubscribe topik
- k. *Unsuback* — *Acknowledgement* dari pesan yang di unsubscribe
- l. *Pingreq* — Meminta ping dari server.
- m. *Pingresp* — Merespon ping dari server
- n. *Disconnect* — Memutuskan koneksi dari server.

Dari sinyal-sinyal tersebut hanya 4 sinyal utama yang dipakai langsung oleh *client* seperti *Publish*, *Subscribe*, *Unsubscribe*, *Connect* dan sinyal lainnya merupakan bagian dari mekanisme kerja publish/subscribe.

2.7.2 Topic dan QOS (Quality of Service) level

Topic adalah UTF-8 dalam bentuk string yang digunakan broker/server untuk memfilter pesan untuk setiap client yang terhubung. Sebuah topic terdiri satu atau lebih level topik, setiap level topic dipisahkan oleh *forward slash* ('\'').

Contoh topic MQTT yang benar :

- rumah/sensor/getar
- rumah/sensor
- rumah

Setiap *client* MQTT mengirim pesan dengan salah satu dari tiga QOS level.

Ketiga level ini yang menjamin apakah pesan terkirim atau tidak. *Client* dan *broker* menyediakan proses pengiriman ulang apabila terjadi kegagalan jaringan. Berikut adalah tabel tiga level QOS.

Tabel 2.7 QOS level

QOS level	Keterangan
0	Hanya satu kali pengiriman. Pengirim akan mencoba upaya terbaik untuk mengirim pesan dan bergantung pada keandalan TCP.
1	Setidaknya dikirim satu kali. Penerima akan mendapatkan pesan setidaknya sekali. Apabila penerima tidak menerima <i>acknowledge</i> pesan atau <i>acknowledge</i> tidak sampai tujuan, maka pesan akan dikirim sampai penerima mendapatkan <i>acknowledge</i>
2	Tepat hanya satu kali dikirim. Protokol memastikan bahwa pesan akan tiba tepat satu kali di penerima dalam satu kali pengiriman. Ini dapat meningkatkan <i>communication overhead</i> , tetapi ini merupakan pilihan terbaik

2.7.3 Mosquitto Broker

Mosquitto broker adalah salah satu jenis opensource broker pesan yang sudah mengimplementasikan protokol *MQTT*. Broker ini juga mendukung pengaplikasian lightweight server dari *MQTT*. Ditulis dalam bahasa pemrograman C dengan alasan agar dapat tetap berkomunikasi dengan mesin yang belum

mendukung JVM. Dari hasil pengujian yang telah dilakukan sebelumnya broker mosquito dapat menampung 100.000 client secara bersamaan.(

<https://mosquitto.org/>

2.8 Microsoft Visual Studio 2017

Microsoft Visual Studio merupakan sebuah perangkat lunak lengkap (*suite*) yang dapat digunakan untuk melakukan pengembangan aplikasi, baik itu aplikasi bisnis, aplikasi personal, ataupun komponen aplikasinya, dalam bentuk aplikasi console, aplikasi Windows, ataupun aplikasi Web. Visual Studio mencakup kompiler, **SDK**, *Integrated Development Environment* (IDE), dan dokumentasi (umumnya berupa MSDN Library). Kompile yang dimasukkan ke dalam paket Visual Studio antara lain Visual C++, Visual C#, Visual Basic, Visual Basic .NET, Visual InterDev, Visual J++, Visual J#, Visual FoxPro, dan Visual SourceSafe.



Gambar 2.15 Logo Visual Studio

Microsoft Visual Studio dapat digunakan untuk mengembangkan aplikasi dalam native code (dalam bentuk bahasa mesin yang berjalan di atas Windows) ataupun managed code (dalam bentuk Microsoft Intermediate Language di atas .NET Framework). Selain itu, Visual Studio juga dapat digunakan untuk mengembangkan aplikasi Silverlight, aplikasi Windows *Mobile* (yang berjalan di atas .NET Compact Framework).

2.8.1 Visual Basic .NET

Visual Basic .NET merupakan bahasa pemrograman komputer berorientasi objek yang bergerak di bidang *.NET Framework*, yang telah diadaptasi dan diuptade dari pendahulunya yaitu Visual Basic. Memiliki kemampuan yang sama dengan pendahulunya yaitu dapat digunakan untuk pembuatan aplikasi *Graphical User Interface*(GUI) atau pemrograman yang menggunakan tampilan grafis untuk berinteraksi dengan pemakainya.

Ada tiga hal utama yang harus diperhatikan dalam mempelajari pemrograman berorientasi objek atau GUI, yaitu :

1. Properti atau Atribut

Merupakan karakteristik atau sifat dari suatu benda. Sebagai contoh seperti warna, ukuran, posisi, dan sebagainya.

2. Event

Segala sesuatu yang akan ingin dilakukan pengguna ke sebuah objek dan objek tersebut akan bereaksi dengan tindakan pengguna. Sebagai contoh menekan sebuah tombol dan muncul pop up, menekan adalah *event* dari sebuah objek yang biasa dinamakan *On_press*.

3. Metode

Serangkaian prosedur atau function yang akan dijalankan ketika dipanggil.

Sebagai contoh ketika menutup sebuah aplikasi, kita bisa menggunakan *form.close()*.

2.9 Arduino IDE

Arduino IDE merupakan aplikasi/software *Integrated Development Environment* (IDE) yang digunakan untuk memprogram sebuah *microcontroller*. Menggunakan bahasa pemrograman C yang sedikit dimodifikasi untuk keperluan program di *microcontroller* itu sendiri. Mempunyai banyak *library* yang dapat di akses dan di download melalui *board* manager. Proses upload program menggunakan port COM karena lebih mudah untuk dipahami oleh *microcontroller*.



BAB III

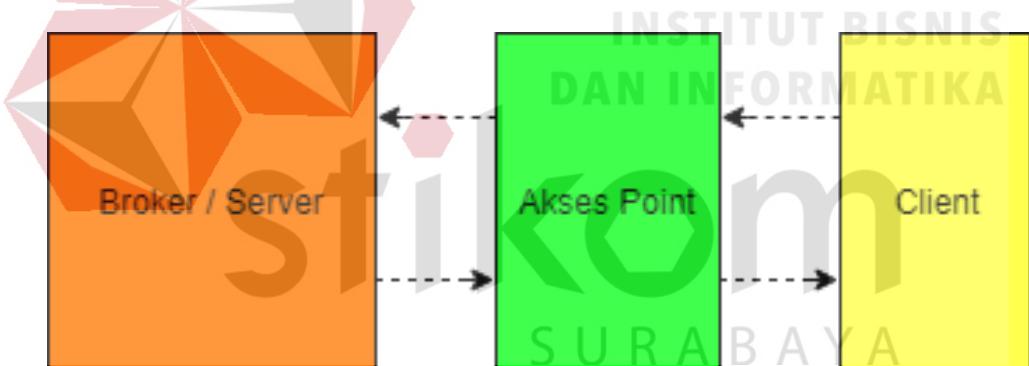
METODE PENELITIAN

3.1 Metode Penelitian

Dalam proses penggerjaan Tugas Akhir ini penulis menggunakan metode penelitian sebagai berikut :

1. Studi Literatur tentang komponen dan metode yang akan digunakan.
2. Perancangan Perangkat Keras.
3. Pembuatan software / program.

Pada perancangan sistem ini terdapat dua bagian yang berperan penting agar sistem dapat berjalan sesuai dengan tujuan. Seperti pada gambar 3.1 .



Gambar 3.1 Blok Diagram Struktur Sistem

Tiap bagian dari blok diagram sistem pada gambar diatas dapat dijelaskan sebagai berikut :

1. Akses Point

Sebagai jalur komunikasi antara broker dengan client. Untuk dapat berkomunikasi satu sama lain, broker dan client harus terhubung dengan Akses Point yang sama

2. Broker / Server

Merupakan sebuah perangkat keras yang digunakan untuk menjalankan sebuah program yang nantinya akan berperan sebagai broker/server.

- a. Device : Raspberry Pi 3.
- b. Input : Data dari client yang sudah dipublish.
- c. Output : Data dari client yang sudah disubscribe.

3. Client

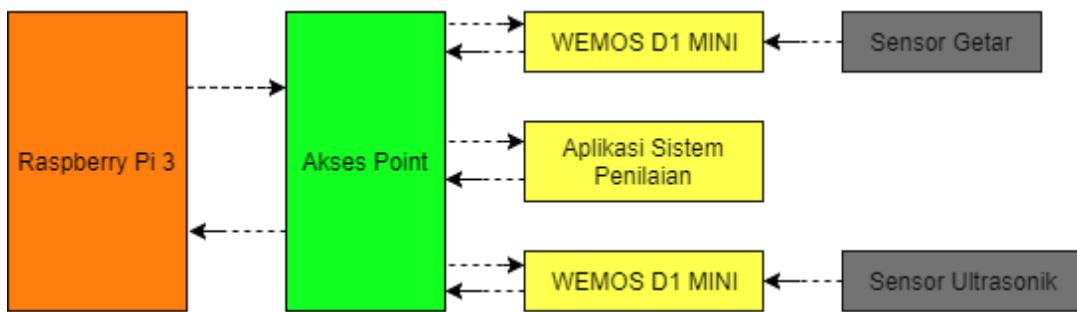
Perangkat keras yang berperan sebagai pengirim data dan penerima data.

- a. Device : Wemos D1 Mini dan Aplikasi Sistem.
- b. Input : Data dari sensor dan data dari broker/server.
- c. Output : Data dari sensor.

3.2 Model Perancangan Keseluruhan

Pada perancangan secara keseluruhan, broker/server menggunakan Raspberry Pi 3 sedangkan untuk client menggunakan Wemos D1 Mini dan Aplikasi Sistem. Wemos D1 Mini mengirim data yang diambil dari sensor getar sw-420 dan sensor ultrasonik hc-sr04 ke broker menggunakan metode komunikasi pesan MQTT, kemudian broker meneruskan data ke Aplikasi Sistem dan kemudian aplikasi menampilkan data. Dalam sistem ini Wemos D1 Mini berperan sebagai *publisher* (Pengirim data), dan Aplikasi sistem sebagai *subscriber* (Penerima data).

Perancangan secara keseluruhan sistem dapat dilihat pada gambar 3.2 :



Gambar 3.2 Blok Diagram Komunikasi pada Sistem

Berikut penjelasan dari setiap blok dan komponen yang digunakan pada sistem :

1. Raspberry Pi 3

- a. Peran : Broker.
- b. Input : Data dari Wemos D1 Mini.
- c. Output : Data menuju ke Aplikasi Sistem.
- d. Fungsi : Sebagai pusat komunikasi antar client yang terhubung dengan akses point yang sama. Tanpa broker, komunikasi tidak dapat berjalan.

2. Wemos D1 Mini

- a. Peran : Client (*publisher*).
- b. Input : Data yang diambil dari Sensor Getar SW-402 dan Sensor Ultrasonik HC-SR04.
- c. Output : dan jarak untuk Sensor Ultrasonik .Data yang diterima dari Sensor getar SW-420 dan Sensor Ultrasonik HC-SR04
- d. Fungsi : Wemos dengan sensor getar sebagai pendekripsi ketika patok tiang pembatas tersenggol/terjatuh, sedangkan Wemos dengan sensor ultrasonik sebagai pendekripsi kecepatan kendaraan peserta.

3. Aplikasi Sistem Penilaian

- a. Peran : Client (*subscriber*).
- b. Input : Data dari Wemos D1 Mini.
- c. Output : Tampilan pada aplikasi yang menunjukkan jumlah patok tersenggol/terjatuh dan kecepatan kendaraan peserta.
- d. Fungsi : Sebagai alat untuk memonitoring patok/tiang pembatas dan kecepatan kendaraan peserta.

4. Sensor Getar SW-420.

- a. Peran : Komponen Sensor.
- b. Input : Getaran pada patok.
- c. Output : Nilai ‘1’ ketika patok tersenggol/terjatuh.
- d. Fungsi : Sebagai pengukur apakah patok tersenggol/terjatuh.

5. Sensor Ultrasonik HC-SR402

- a. Peran : Komponen Sensor
- b. Input : Benda yang melewati sensor.
- c. Output : Jarak benda asing ke sensor.
- d. Fungsi : Sebagai penghitung kecepatan kendaraan peserta ujian.

Berikut adalah alur dari blok diagram diatas :

1. Sensor Getar SW-402 yang tersambung pada Mikrokontroller Wemos D1 Mini berfungsi sebagai pendeksi apakah tiang pembatas/patok terjatuh atau tersenggol. Apabila tiang pembatas terjatuh/tersenggol, Sensor mengirim data ke Mikrokontroller Wemos D1 Mini berupa nilai 1 kemudian Mikrokontroller Wemos D1 Mini akan mengirim pesan ke Aplikasi Sistem Penilaian berupa pesan “ON” yang nantinya akan diproses oleh Aplikasi

- Sistem agar dapat menampilkan visualisasi dari jatuhnya atau tersenggolnya patok tersebut.
2. Sensor Ultrasonik HC-SR04 yang tersambung ke Mikrokontroller Wemos D1 Mini berfungsi sebagai penghitung kecepatan pada Sistem Penilaian. Sensor pertama diletakkan pada garis start lintasan, dan Sensor kedua diletakkan di garis finish lintasan. Cara kerja dari kedua Sensor ini adalah ketika peserta ujian melewati garis start, Sensor pertama akan mendeteksi adanya pergerakan dan mengirim data ke Aplikasi, kemudian ketika peserta ujian melewati garis finish, Sensor kedua juga mengirim data ke Aplikasi dan Aplikasi akan memvisualkan berapa kecepatan rata-rata peserta saat ujian pada lintasan tersebut.
 3. Raspberry Pi 3 sebagai broker/server yang bertugas menerima data dari Wemos dan mengirim data ke aplikasi. Raspberry Pi 3 tersambung ke akses point yang sama dengan Mikrokontroller dan aplikasi agar komunikasi MQTT dapat berlangsung.
 4. Wemos D1 Mini merupakan Mikrokontroller utama yang berfungsi menerima data dari sensor getar ataupun sensor ultrasonik yang kemudian dikirim ke broker/server Raspberry Pi 3. Program pada Wemos menggunakan bahasa C++ yang dicompile menggunakan aplikasi Arduino IDE. Wemos juga tersambung ke akses point yang sama dengan broker/server agar komunikasi dapat berlangsung.
 5. Sistem penilaian merupakan program aplikasi yang dibuat di Microsoft Visual Studio 2017 menggunakan bahasa Visual Basic .NET. Aplikasi dapat menampilkan hasil visual dari tiang-tiang yang tersenggol atau terjatuh pada

saat pengujian dan menampilkan berapa kecepatan kendaraan peserta saat ujian berlangsung. Sistem juga tersambung dengan akses point yang sama dengan broker/server.

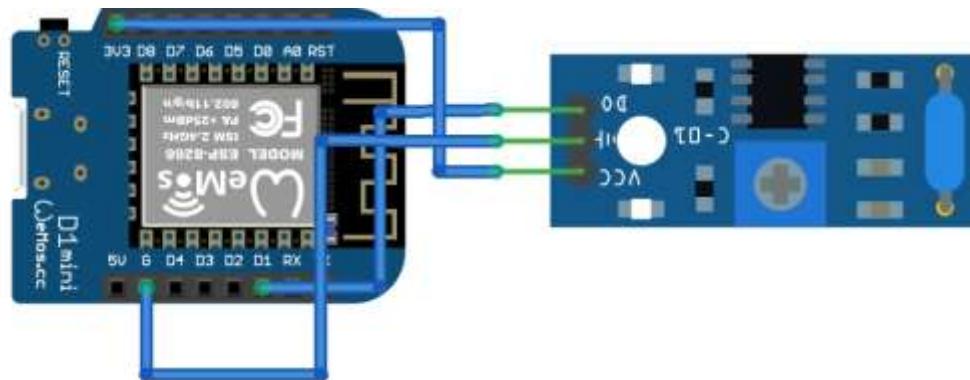
3.3 Perancangan Perangkat Keras

Pada pelaksanaan Tugas Akhir dengan judul Rancang Bangun Sistem Penilaian Ujian Praktik SIM C Berbasis Wireless Menggunakan Protokol Komunikasi Pesan MQTT, Penulis menggunakan tiga lintasan sebagai berikut :

1. Lintasan Uji Keseimbangan.
2. Lintasan Uji Zig-zag.
3. Lintasan Uji U-turn/Putar balik.

Mikrokontroller yang digunakan adalah Wemos D1 Mini, dengan sensor getar SW-420 sebagai pendekripsi getaran pada patok tiang pembatas dan sensor ultrasonik HC-SR04 sebagai penghitung kecepatan kendaraan peserta ujian. Raspberry Pi 3 sebagai broker/server yang nantinya menjadi pusat dari komunikasi antara Mikrokontroller dan Aplikasi sistem.

3.3.1 Rangkaian Sensor Getar SW-400 pada Patok Tiang Pembatas



Gambar 3.3 Rangkaian Sensor Getar SW-400

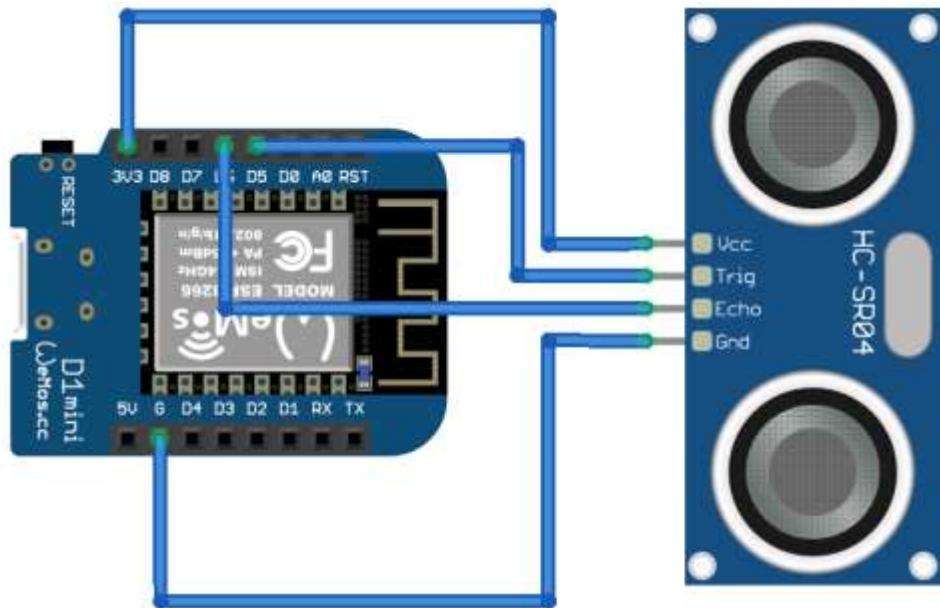
Pada Gambar 3.3 adalah rangkaian Sesor Getar yang akan dipasang di tiang pembatas/patok untuk sistem penilaian ujian praktik SIM C. Mikrokontroller Wemos D1 Mini sebagai pusat kendali yang mengatur pengiriman data dari Sensor ke broker/server. Sensor Getar SW-402 berperan sebagai input yang digunakan untuk mengetahui apakah tiang pembatas/patok tersenggol atau terjatuh, dimana nanti akan mengirim nilai logika HIGH ke Wemos apabila Sensor menerima getaran pada tiang pembatas/patok.

Keterangan :

- A. Rangkaian ditempel di bagian dalam tiang pembatas/patok.
- B. Daya menggunakan baterai 1.5V x 3
- C. Pin yang terhubung antar rangkaian adalah
 - 1. Pin 5V dan GND pada Wemos D1 Mini dihubungkan ke daya baterai.
 - 2. Pin VCC Sensor Getar SW-402 terhubung ke pin 3.3v Wemos D1 Mini
 - 3. Pin 0v Sensor Getar SW-402 terhubung ke pin GND Wemos D1 Mini.
 - 4. Pin D0 Sensor Getar SW-402 terhubung ke pin D1 Wemos D1 Mini

3.3.2 Rangkaian Sensor Ultrasonik HC-SR04 pada Patok Tiang Pembatas

Untuk mendeteksi kecepatan kendaraan peserta ujian menggunakan rangkaian Sensor Ultrasonik HC-SR04, Dimana nanti Sensor akan ditempatkan di garis start dan garis finish pada Uji Keseimbangan dan Uji Zig-zag seperti pada gambar 3.4 . Sensor pertama digunakan sebagai tanda bahwa peserta telah melewati garis start atau sudah memulai ujian praktik dan sesor kedua sebagai tanda bahwa peserta telah mencapai garis finish atau selesai menyelesaikan ujian praktik.



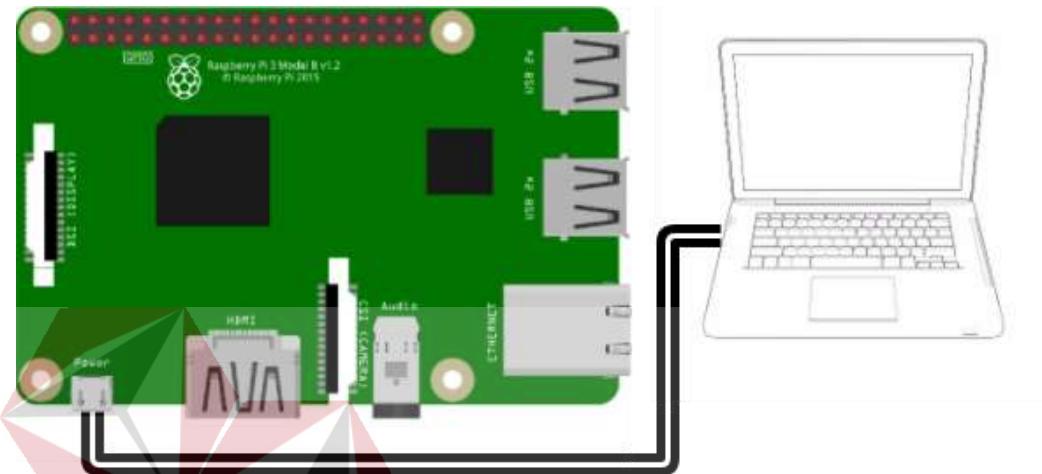
Gambar 3.4 Rangkaian Sensor Ultrasonik HC-SR04

Keterangan :

- A. Pada sistem ini menggunakan 2 buah Sensor yang diletakkan di garis start dan garis finish.
- B. Rangkaian ditempel pada patok yang berbeda dengan patok sensor getar, dan penempatan lebih tinggi agar sensor dapat bekerja lebih maksimal.
- C. Daya menggunakan baterai 1.5V x 3
- D. Pin yang terhubung pada rangkaian adalah.
 1. Pin VCC Sensor Ultrasonik HC-SR04 terhubung ke pin 3.3v Wemos D1 Mini.
 2. Pin GND Sensor Ultrasonik HC-SR04 terhubung ke pin GND Wemos D1 Mini.
 3. Pin TRIG Sensor Ultrasonik HC-SR04 terhubung ke pin D5 Wemos D1 Mini.

4. Pin ECHO Sensor Ultrasonik HC-SR04 terhubung ke pin D6 Wemos D1 Mini.
5. Pin 5V dan GND pada Wemos D1 Mini dihubungkan ke daya baterai.

3.3.3 Raspberry Pi 3 Sebagai Broker / Server



Gambar 3.5 Rangkaian Raspberry Pi 3

Pada Tugas Akhir ini Rapsberry pi 3 sebagai perangkat keras digunakan sebagai broker / server, dimana Raspberry menjalankan program *mosquitto-broker* agar Wemos dan Aplikasi Sistem dapat terhubung dan berkomunikasi. Berikut penjelasan pada rangkaian gambar 3.5 .

Keterangan :

- A. Raspberry Pi 3 menggunakan Usb type micro B sebagai daya.
- B. Raspberry terhubung ke Akses Point yang sama dengan Mikrokontroller dan Aplikasi Sistem.
- C. Pin yang terhubung pada rangkaian adalah.

1. POWER pada Raspberry terhubung ke Laptop / PC menggunakan kabel Usb type micro B, juga dapat menggunakan charger handphone android (bukan iphone).

Sebelum menjalankan broker pada Raspberry Pi ada beberapa tahapan yang mungkin perlu dilakukan agar saat instalasi broker pada Raspi dapat berjalan dengan baik, Diantaranya adalah :

1. Instalasi Operating Sistem Raspbian pada Raspberry Pi 3
2. Menyambungkan ke Akses Point yang memiliki akses internet.

Broker pada Raspberry Pi merupakan program yang diunduh dari internet yang kemudian di jalankan secara otomatis ketika Raspberry pi dinyalakan. Berikut *Source Code* yang digunakan melalui ssh putty.

```
pi@raspberrypi :~ $ sudo apt-get update  
pi@raspberrypi :~ $ sudo apt install -y mosquitto mosquitto-clients
```

Setelah proses pengunduhan selesai, broker dapat dicoba dengan mengetik :

```
pi@raspberrypi :~ $ mosquitto -v
```

Jika muncul error seperti pada gambar 3.6, Maka itu adalah tanda bahwa broker pada Raspberry Pi telah berhasil dijalankan. Setelah broker menyala maka akses point dapat dirubah kembali ke akses point yang sedang tersambung dengan patok-patok dan laptop petugas agar wemos dapat berkomunikasi ke broker. Pada proses menjalankan broker pada Raspberry, Raspberry menjalankan broker dengan *default config*.

```
pi@raspberrypi:~ $ mosquitto -v
1544767991: mosquitto version 1.4.10 (build date Wed, 17 Oct 2018 19:03:03 +0200)
) starting
1544767991: Using default config.
1544767991: Opening ipv4 listen socket on port 1883.
1544767991: Error: Address already in use
...  
[REDACTED]
```

Gambar 3.6 Eror pada Broker di Raspberry Pi 3

Terdapat beberapa parameter dalam proses menjalankan *mosquitto* pada Raspberry Pi, Seperti :

1. **-c** : Menjalankan broker dengan file *config*.
2. **-d** : Menjalankan broker pada background aplikasi setelah dijalankan.
3. **-p** : Menjalankan broker dengan *port* tertentu. Tidak disarankan menjalankan ini dengan **-c**.
4. **-v** : *verbose mode*. Mengaktifkan semua tipe *log*.

Pada file konfigurasi *mosquitto* juga terdapat beberapa pengaturan yang dapat dirubah sesuai kebutuhan, diantaranya :

1. **retry_interval 20**

Waktu dalam detik untuk menunggu sebelum mengirimkan kembali pesan,
Berlaku untuk QoS=1 atau Qos=2.

2. **port 1883**

Sebagai *default port* pada saat broker dinyalakan.

3. **max_connections -1**

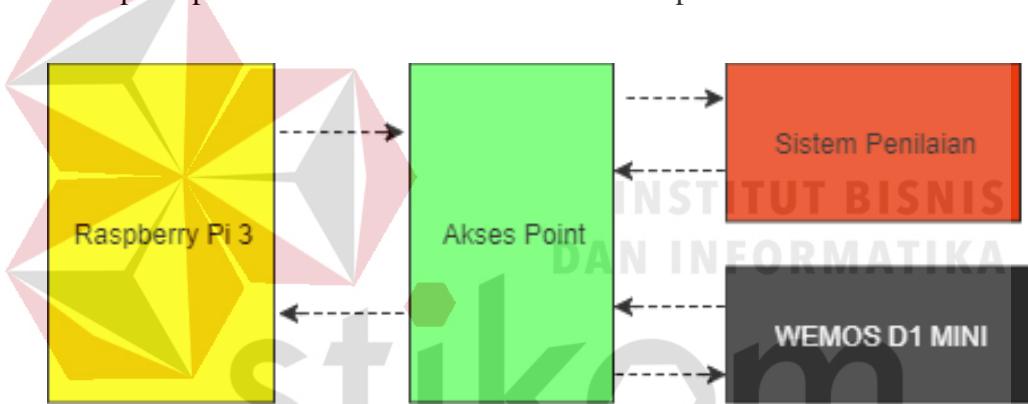
Jumlah maksimum dari klien yang dapat tersambung kebroker, -1 untuk jumlah paling maksimal yaitu 1024.

4. **connection_messages true**

Digunakan untuk pemberitauan ketika ada klien yang tersambung atau terputus dari broker.

3.3.4 Perancangan Komunikasi pada Sistem

Mikrokontroller yang digunakan pada sistem adalah Wemos D1 Mini, dan *broker/server* menggunakan Raspberry Pi 3. Kedua device ini diharuskan tersambung pada satu jaringan yang sama agar dapat berkomunikasi seperti gambar 3.7 . Pada system ini menggunakan protocol MQTT untuk proses pengiriman dan penerimaan data. Data dari sensor akan dikirim ke system melalui perantara broker/server, proses pengiriman dan penerimaan ini bergantung pada koneksi TCP/IP itu sendiri. Jadi apabila kedua device ini tidak berada pada satu jaringan, maka dapat dipastikan bahwa kedua device tidak dapat berkomunikasi.



Gambar 3.7 Skema Jalur Koneksi Sistem

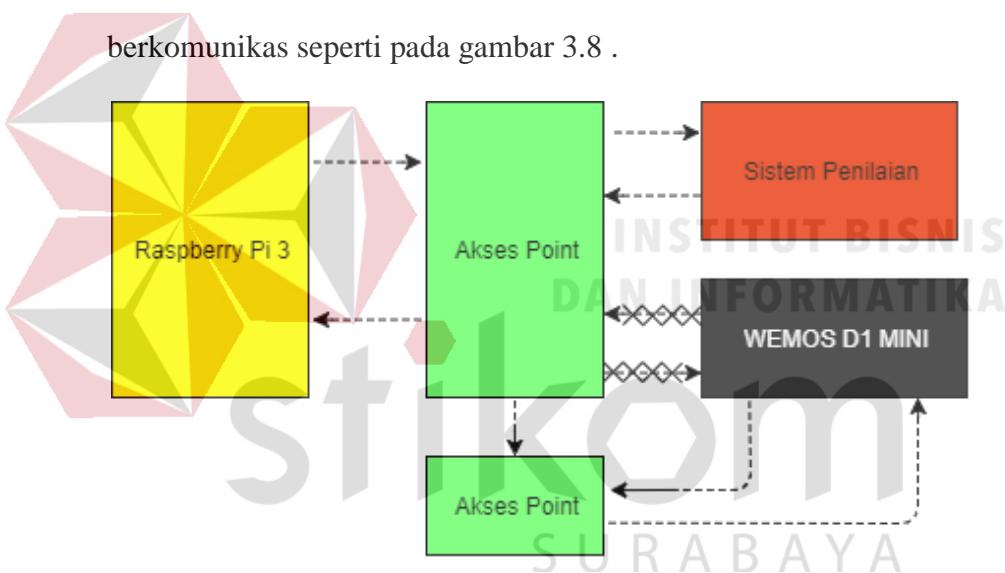
Berikut adalah alur dari blok diagram gambar 3.7 :

1. Wemos, Raspberry Pi 3 dan Aplikasi sistem tersambung pada akses point yang sama.
2. Wemos dan Aplikasi sistem sebagai *client* tersambung ke broker yang ada pada Raspberry Pi 3.
3. Data dari sensor diterima oleh Wemos, kemudian dikirm ke broker Raspberry dengan topic yang sudah ditentukan (*publish*).

4. Aplikasi sistem menerima data dengan dari broker Raspberry pi dengan topic yang sama dengan Wemos (*subscribe*).

Apabila Wemos tidak dapat menjangkau akses point dikarenakan terlalu jauh dari radius jaringan, maka terdapat beberapa solusi yang penulis dapat lakukan diantaranya.

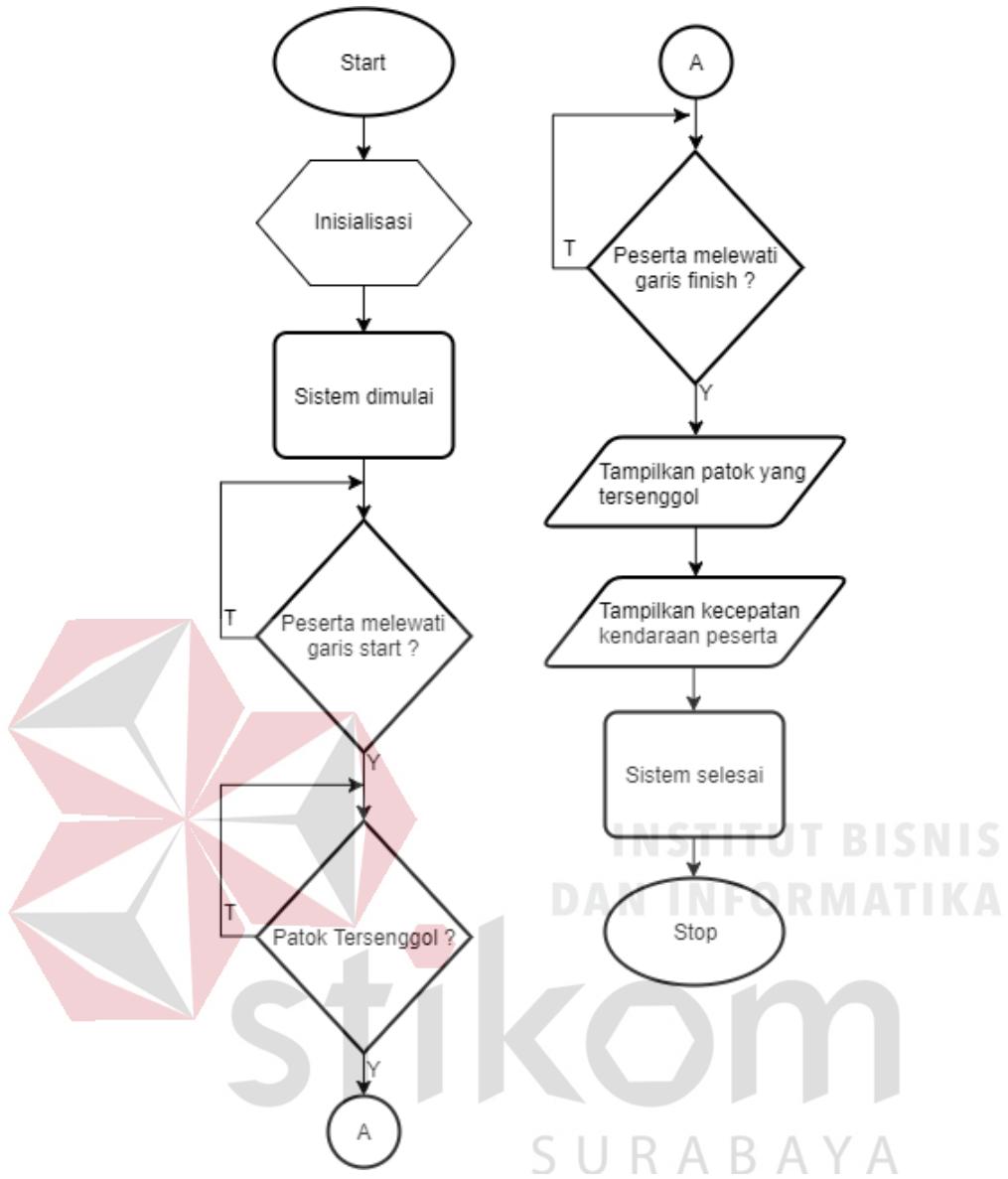
1. Menambahkan antena eksternal.
2. Menambahkan akses point lagi dari router utama. Hal ini dilakukan agar Wemos dan sistem tetap dapat tersambung pada satu jaringan dan dapat berkomunikasi seperti pada gambar 3.8 .



Gambar 3.8 Skema Jalur Koneksi Sistem Dengan AP Tambahan

3.4 Perancangan Perangkat Lunak

Pada perancangan sistem ini menggunakan bahasa pemrograman Visual basic .Net untuk pembuatan aplikasi pada laptop / pc, sedangkan untuk pembuatan program pada pendekripsi getaran dan pendekripsi kecepatan kendaraan peserta pada patok tiang pembatas menggunakan Arduino Ide. Aplikasi pada laptop hanya menampilkan hasil dari data yang diterima dari Wemos D1 mini.



Gambar 3.9 Flowchart Sistem Secara Keseluruhan

Proses dari flowchart gambar 3.9 adalah sebagai berikut :

1. Insialisasi diatas adalah menyambungkan ke Akses Point dan menyambungkan ke broker / server pada Raspberry Pi 3.
2. Sistem memulai dengan men-subscribe / mengpublish ke topic yang telah ditentukan.

3. Setelah itu sensor ultrasonik mendeteksi adanya pergerakan di garis start. Ketika mendeteksi adanya pergerakan sistem mencatat waktu (jam, menit dan detik).
4. Ketika patok tiang pembatas tersenggol, Aplikasi menerima data dari Wemos dan menampilkannya.
5. Sensor ultrasonik kedua yang berada di garis finish dan mendeteksi adanya gerakan ketika pengendara melewati garis. Kemudian mencatat waktu (jam, menit dan detik).
6. Data di outputkan melalui tampilan pada aplikasi berupa kecepatan kendaraan dan patok yang tersenggol sebelumnya.
7. Sistem selesai dengan meng-unsubscribe topic-topic yang sama seperti sebelumnya, agar tidak terjadi kekacuan pada pengiriman data.

3.4.1 Pembacaan Data Sensor Getar pada Patok Tiang Pembatas

Proses dari algoritma flowchart gambar 3.10 sebagai berikut :

1. Inisialisasi diatas adalah menyiapkan nama ssid, password, alamat broker, nama client dan pin yang digunakan untuk Sensor Getar. Untuk lebih jelasnya dapat dilihat pada source berikut.

Source Code :

```
#include <ESP8266WiFi.h>
#include <PubSubClient.h>
const char* ssid = "tugasakhirkku";
const char* password = "qwertyuiop";
const char* mqttServer = "192.168.1.111";
const char* clientId = "Getar-1";
int getar = D1;
```

```

int state;
WiFiClient espClient;
PubSubClient client(espClient);

```

2. Setelah proses inisialisasi, Wemos menjalankan *void setup()* yang dimana dijalankan tepat satu kali saat pertama kali Wemos dinyalakan.

Source Code :

```

void setup() {
    // put your setup code here, to run once:
    Serial.begin(115200);
    wifiConnect();
    client.setServer(mqttServer, 1883);
    pinMode(getar, INPUT);
}

```

3. Berikut adalah proses untuk menghubungkan ke Akses Point dengan ssid dan password yang sudah disiapkan.

Source Code :

```

void wifiConnect()
{
    delay(10);

    WiFi.mode(WIFI_STA);

    WiFi.hostname("wemos_Getar1");

    Serial.println("Connecting to : ");
    Serial.print(ssid); Serial.println(" ...");

    WiFi.begin(ssid, password);

    int i = 0;

    while (WiFi.status() != WL_CONNECTED)
    {
        delay(1000);

        Serial.print(++i); Serial.print(' ');
    }
}

```

```

    }

    Serial.println('\n');

    Serial.println("Connection established!");

    Serial.print("IP address:\t");

    Serial.println(WiFi.localIP());

}

```

4. Ketika Wemos sudah tersambung ke Akses Point, Wemos akan menjalankan proses loop utama, pembacaan sensor getar menggunakan digitalRead dimana nilai yang dihasilkan berupa 1 jika menerima data dan 0 jika tidak menerima data. Setelah itu masuk ke function reconnect untuk menyambungkan ke broker. Setelah tersambung ke broker, Wemos akan memulai pembacaan sensor getar. Apabila sensor getar terdeteksi maka Wemos akan mempublish ke topic “getar/1” dengan pesan “on”.

Source Code :

```

void loop() {
    state = digitalRead(getar);

    if (!client.connected())
    {
        reconnect();
    }

    delay(100);

    Serial.println(state);

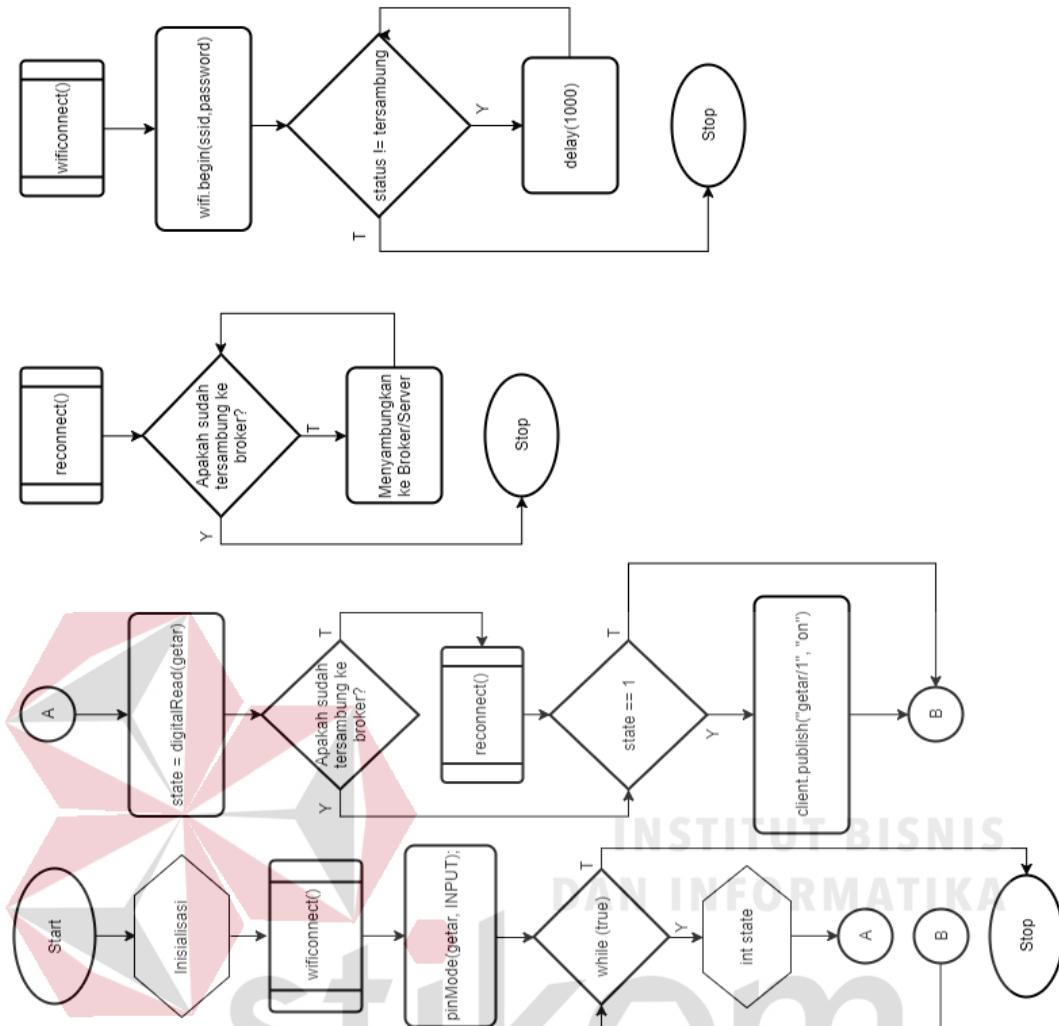
    if (state == 1)
    {
        client.publish("getar/1", "on");
        delay(1000);
    }
}

```

5. Berikut adalah isi dari function reconnect, yang digunakan untuk menyambungkan Wemos ke broker/server. Function ini akan terus dijalankan hingga Wemos tersambung ke broker/server.

Source Code :

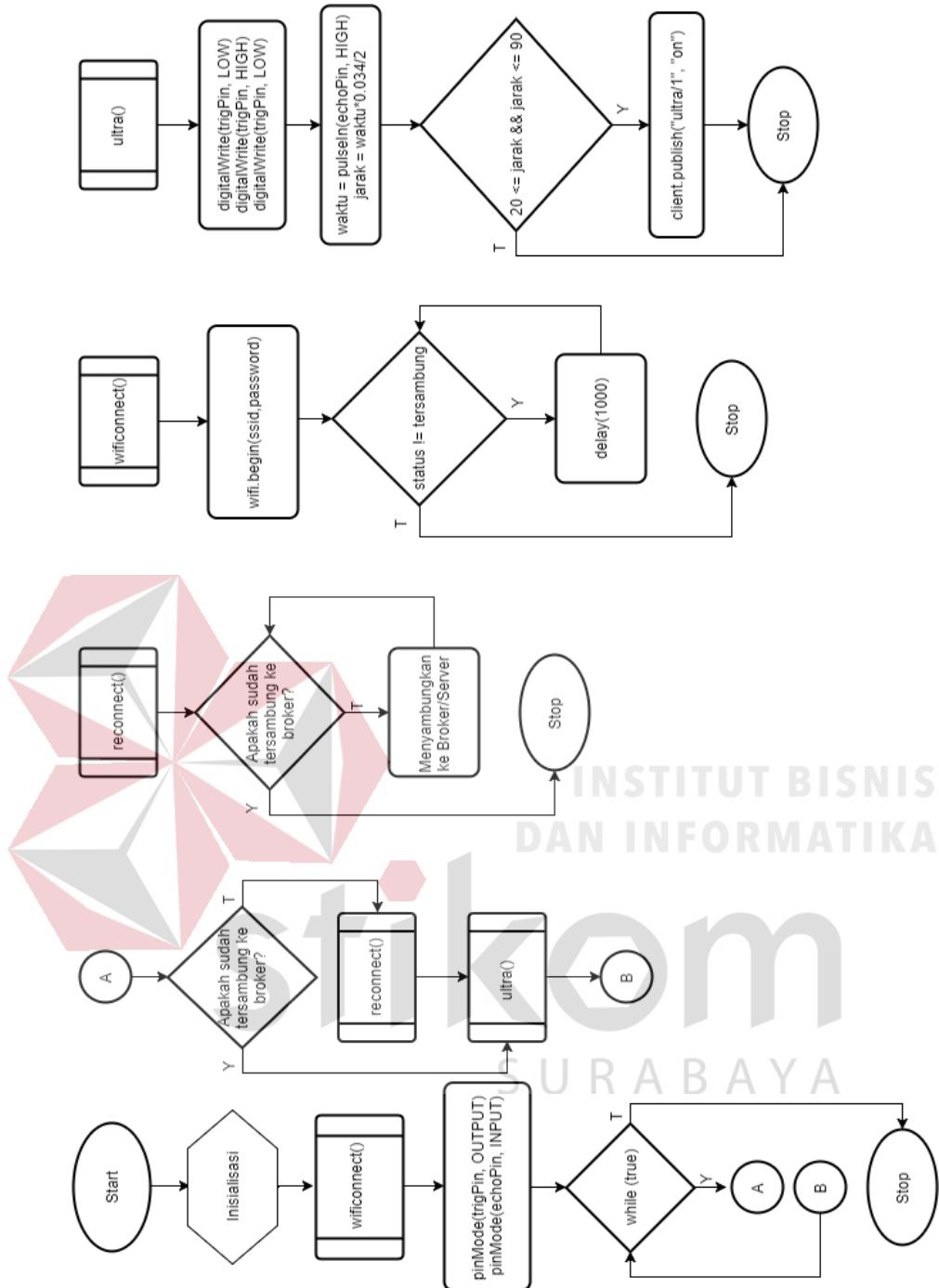
```
void reconnect() {
    while (!client.connected())
    {
        Serial.println('\n');
        Serial.print("Menghubungkan ke broker");
        Serial.print(mqttServer);
        if (client.connect(clientId))
        {
            Serial.println("connected");
        }
        else
        {
            Serial.print("failed, rc=");
            Serial.print(client.state());
            Serial.println(" try again in 5 seconds");
            delay(6000);
        }
    }
} //end reconnect
```



Gambar 3.10 Flowchart Pembacaan Data Sensor Getar

3.4.2 Pembacaan Data Sensor Ultrasonik pada Patok Tiang Pembatas

Sensor Ultrasonik digunakan untuk mengetahui ketika peserta ujian sudah melewati garis start dan garis finish. Ketika peserta melewati garis start, Sensor akan mengirim data dan ketika peserta melewati garis finish, Sensor akan kembali mengirim data ke broker/server. Lintasan yang menggunakan Sensor Ultrasonik adalah lintasan Uji Keseimbangan. Berikut adalah algoritma dari Sensor Ultrasonik.



Gambar 3.11 Flowchart Pembacaan Data Sensor Ultrasonik

Proses dari flowchart gambar 3.11 diatas sebagai berikut :

1. Inisialisasi diatas adalah menyiapkan nama ssid, password, alamat broker, pin dan lain lain yang digunakan untuk Sensor Ultrasonik. Untuk lebih jelasnya dapat dilihat pada *Source Code* dibawah ini.

Source Code :

```
#include <ESP8266WiFi.h>
#include <PubSubClient.h>

const char* ssid = "tugasakhirku";
const char* password = "qwertyuiop";
const char* mqttServer = "192.168.1.111";
const char* clientId = "Ultra-1";
const int trigPin = D5;
const int echoPin = D6;
long waktu;
int jarak;
double state;
WiFiClient espClient;
PubSubClient client(espClient);
```

2. Setelah proses inisialisasi, Wemos menjalankan *void setup()* yang dimana dijalankan tepat satu kali saat pertama kali Wemos dinyalakan.

Source Code :

```
void setup() {
    // put your setup code here, to run once:
    Serial.begin(115200);
    wifiConnect();
    client.setServer(mqttServer, 1883);
    pinMode(trigPin, OUTPUT); // Sets the trigPin as an Output
    pinMode(echoPin, INPUT); // Sets the echoPin as an Input
}
```

3. Berikut adalah proses untuk menghubungkan ke Akses Point dengan ssid dan password yang sudah disiapkan.

Source Code :

```

void wifiConnect()
{
    delay(10);

    WiFi.mode(WIFI_STA);
    WiFi.hostname("wemos_Ultra1");
    Serial.println("Connecting to : ");
    Serial.print(ssid); Serial.println(" ...");
    WiFi.begin(ssid, password);

    int i = 0;
    while (WiFi.status() != WL_CONNECTED)
    {
        delay(1000);
        Serial.print(++i); Serial.print(' ');
    }
    Serial.println('\n');
    Serial.println("Connection established!");
    Serial.print("IP address:\t");
    Serial.println(WiFi.localIP());
}

```

4. Ketika Wemos sudah tersambung ke Akses Point, Wemos akan menjalankan proses loop utama, Dimana terdapat pengecekan apakah Wemos sudah tersambung ke broker. Apabila belum tersambung maka function reconnect akan dijalankan. Ketika sudah tersambung, Wemos akan menjalankan function untuk sensor ultrasonik.

Source Code :

```

void loop() {
    // put your main code here, to run repeatedly:
    if (!client.connected())
    {
        reconnect();
    }
}

```

```

delay(100);
ultra();
}

```

5. Pada function reconnect terdapat perulangan while dengan kondisi sudah tersambung ke broker atau belum.

Source Code :

```

void reconnect(){
    while (!client.connected())
    {
        Serial.println('\n');
        Serial.print("Menghubungkan ke broker ");
        Serial.print(mqttServer);
        if (client.connect(clientId))
        {
            Serial.println("connected");
        }
        else
        {
            Serial.print("failed, rc=");
            Serial.print(client.state());
            Serial.println(" try again in 5 seconds");
            delay(6000);
        }
    }
} //end reconnect

```

6. Setelah Akses Point dan broker/server tersambung, Wemos akan menunggu data dari Sensor Ultrasonik. Data dari Sensor Ultrasonik diolah agar dapat mendeteksi jarak, Seperti pada *Source Code* berikut.

Source Code :

```
void ultra()
{
    digitalWrite(trigPin, LOW);
    delayMicroseconds(2);

    digitalWrite(trigPin, HIGH);
    delayMicroseconds(10);

    digitalWrite(trigPin, LOW);

    waktu = pulseIn(echoPin, HIGH);

    //jarak = waktu*0.034/2;

    jarak = waktu/58;

    if (20 <= jarak && jarak <= 50) {
        client.publish("ultra/1", "on");
        delay(1000);
    }

    Serial.print("Jarak: ");
    Serial.println(jarak);
}
```

Menggunakan *pulsein* lalu variabel waktu dibagi 58 dan akan menghasilkan jarak. Setelah itu jarak akan dibandingkan lebih besar sama dengan dari 20 dan jarak lebih kecil samadengan 90. Apabila bernilai TRUE, Wemos akan mengirim pesan “on” ke topic “ultra/1” di broker/server.

3.4.3 Perancangan Aplikasi pada Sistem

3.3.4.1 Perancangan Form Menu Utama



Gambar 3.12 Tampilan Menu Utama

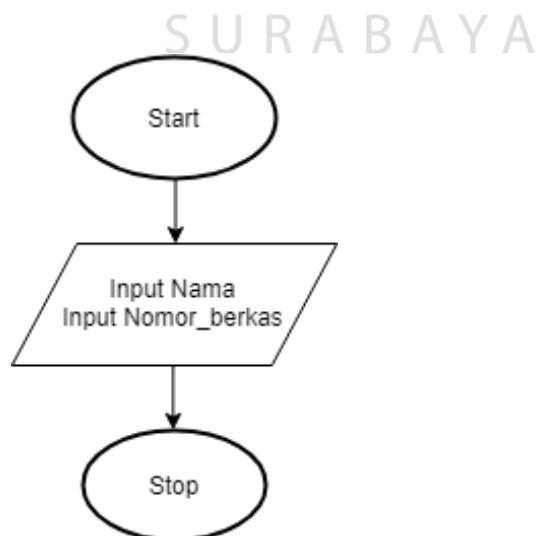
Pada gambar 3.12 form main menu terdapat beberapa komponen yang digunakan, seperti pada tabel 3.1 berikut :

Tabel 3.1 Tabel Komponen Menu Utama

Nama Komponen	Keterangan	Fungsi	Jumlah
Picture Box	-	Logo Kepolisian	3
Button	Text = "Edit"	Untuk mengedit nama dan nomor berkas.	1

Nama Komponen	Keterangan	Fungsi	Jumlah
Button	Text = “Pengaturan Broker”	Untuk masuk ke form/menu pengaturan broker.	1
Button	Text = “Uji Keseimbangan”	Untuk masuk ke form/menu pengujian lintasan keseimbangan.	1
Button	Text = “Uji Zig-zag”	Untuk masuk ke form/menu pengujian lintasan zig-zag.	1
Button	Text = “Uji Putar balik”	Untuk masuk ke form/menu pengujian lintasan putar balik.	1
Button	Text = “Lihat Hasil Akhir”	Untuk masuk ke form/menu hasil penilaian semua lintasan.	1
Text Box	-	Tempat mengisi nomor berkas dan nama peserta.	2

Berikut adalah flowchart dari form main menu :



Gambar 3.13 Flowchart pada Form Main Menu

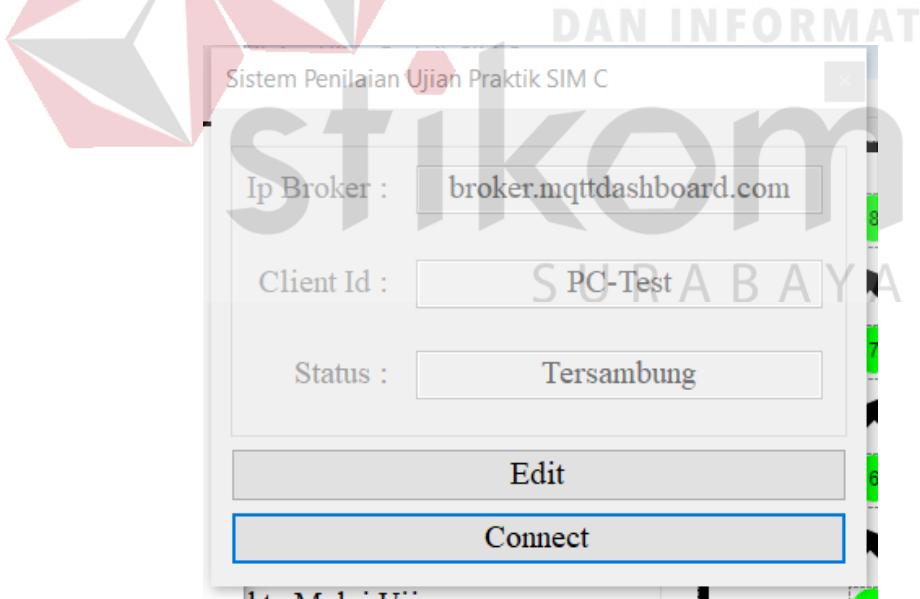
Proses dari algoritma flowchart gambar 3.13 diatas sebagai berikut :

1. Petugas mengisikan nomor berkas dan nama peserta ujian, kemudian disimpan.
2. Data yang telah diinputkan akan disimpan ke variabel global yang nantinya dapat diakses oleh seluruh aplikasi.Seperti pada *Source Code* berikut,

Source Code :

```
Private Sub btn_savemenu_Click(sender As Object, e As EventArgs) Handles btn_savemenu.Click
    GroupBox2.Enabled = False
    GroupBox1.Enabled = True
    namaPeserta = txt_inputnama.Text
    noBerkas = txt_inputnomorberkas.Text
End Sub
```

3.3.4.2 Perancangan Form Pengaturan Broker



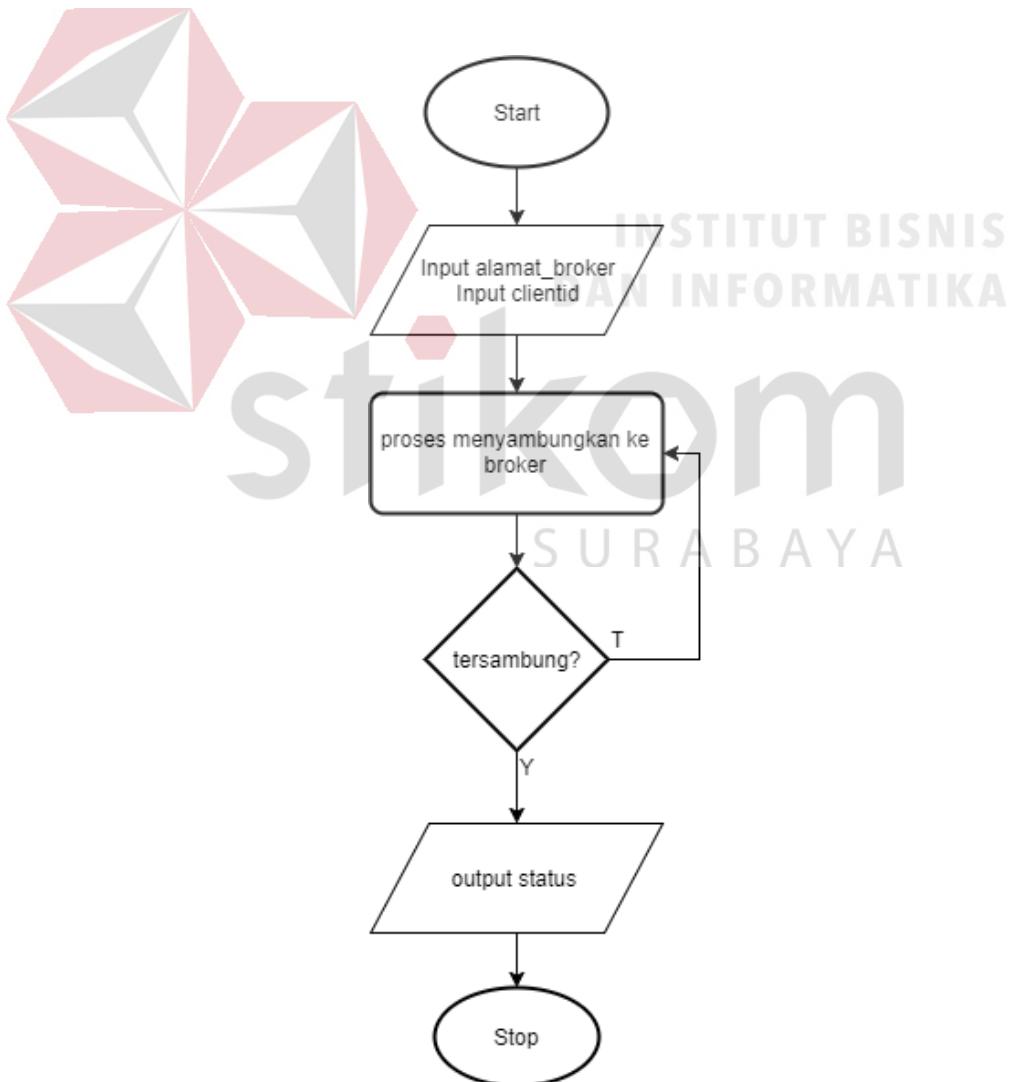
Gambar 3.14 Tampilan Menu Koneksi Broker

Pada gambar 3.14 form pengaturan broker terdapat beberapa komponen yang digunakan, seperti pada tabel 3.2 berikut :

Tabel 3.2 Tabel Komponen Menu Koneksi Broker

Nama Komponen	Keterangan	Fungsi	Jumlah
Text Box	-	Tempat mengisi alamat broker, client id dan status koneksi broker	3
Button	Text = “Edit”	Untuk mengedit alamat broker dan client id.	1
Button	Text = “Connect”	Untuk memulai proses penyambungan ke broker	1

Berikut adalah flowchart dari menu pengaturan broker :



Gambar 3.15 Flowchart pada Menu Pengaturan Broker

Proses dari algoritma flowchart gambar 3.15 diatas sebagai berikut :

1. Setelah petugas menginputkan data peserta, petugas dapat mengisi alamat broker/server dan nama dari clientid aplikasi, dalam penelitian ini alamat server nantinya diisikan dengan ip address dari Raspberry Pi 3 yang berperan sebagai broker. Client id dapat berupa huruf, simbol maupun angka, hanya sebagai pengenal pada broker/server nantinya.
2. Apabila ip address yang dimasukan salah atau terjadi kendala pada internet, akan muncul notifikasi bahwa error telah terjadi.
3. Ketika berhasil tersambung, akan muncul tulisan “Tersambung” pada aplikasi. Baru setelah berhasil tersambung, petugas dapat melanjutkan ke proses pengujian. Seperti pada *Source Code* berikut.

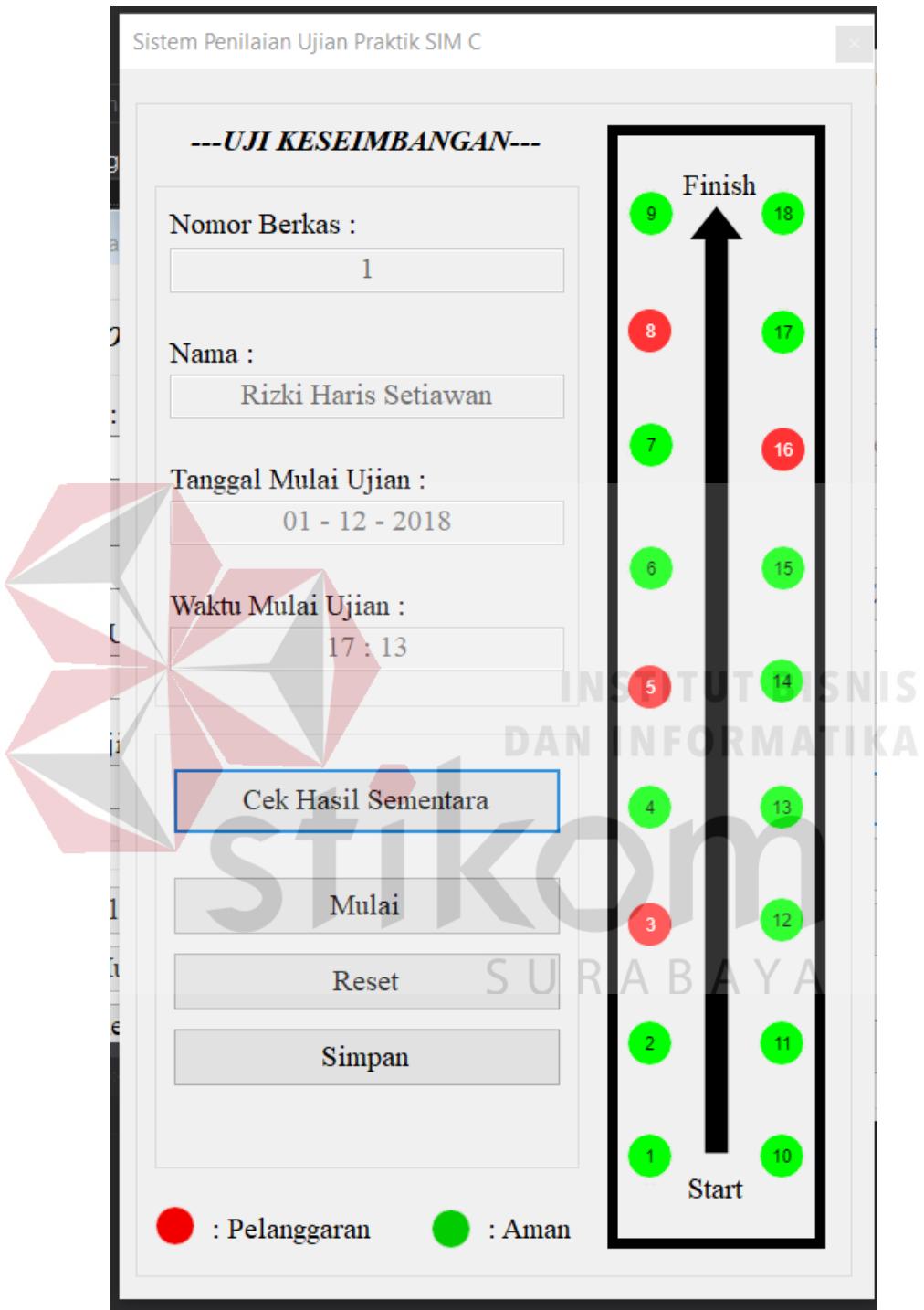
Source Code :

```

Private Sub btn_connect_Click(sender As Object, e As EventArgs) Handles btn_connect.Click
    ipBroker = txt_ipbroker.Text
    clientId = txt_client.Text
    mqttClient = New MqttClient(ipBroker)
    Try
        mqttClient.Connect(clientId)
        GroupBox1.Enabled = False
    Catch ex As Exception
        MsgBox("Tidak dapat tersambung." + vbCrLf +
               "Mohon dicek kembali IP dari broker / server.")
        GroupBox1.Enabled = True
    End Try
    If mqttClient.IsConnected = True Then
        txt_status.Text = "Tersambung"
    End If
End Sub

```

3.3.4.3 Perancangan Form Uji Keseimbangan



Gambar 3.16 Tampilan Menu Pengujian Uji Keseimbangan

Pada gambar 3.16 form uji keseimbangan terdapat beberapa komponen yang digunakan, seperti pada tabel 3.3 berikut :

Tabel 3.3 Tabel Komponen Menu Pengujian Uji Keseimbangan

Nama Komponen	Keterangan	Fungsi	Jumlah
Picture Box	Colour = “green”	Sebagai indikator apakah patok pada lintasan tesenggol atau tersenggol	18
Button	Text = “Cek hasil sementara”	Untuk melihat berapa patok yang tersenggol/terjatuh, kecepatan pengendara dan estimasi dari garis start ke garis finish	1
Button	Text = “Mulai”	Untuk memulai pengujian.	1
Button	Text = “Reset”	Untuk menghapus seluruh data.	1
Button	Text = “Simpan”	Untuk menyimpan data pada “cek hasil sementara”	1
Text Box	-	Untuk menampilkan nomor berkas, nama peserta, tanggal dan waktu mulai ujian.	1

Pada gambar 3.17 adalah flowchart menu pengujian uji keseimbangan :

Proses dari algoritma flowchart gambar 3.17 sebagai berikut :

1. Pada saat tombol mulai ditekan, aplikasi akan men-subscribe topic yang sama dengan sensor getar dan sensor ultrasonik. Dan pengujian dapat dimulai saat itu juga.

Source Code :

```
Private Sub btn_mulai_keseim_Click(sender As Object, e As EventArgs) Handles btn_mulai_keseim.Click
    TextBox4.Text = Date.Now.ToString(" HH : mm ")
    Try
```

```

CheckForIllegalCrossThreadCalls = False

AddHandler mqttClient.MqttMsgPublishReceived,
AddressOf terimaPesanUjiKeseimbangan

For i As Integer = 1 To 18

    mqttClient.Subscribe(New String() {"getar/" & i}, New Byte()
    {MqttMsgBase.QOS_LEVEL_AT_MOST_ONCE})

Next

For m As Integer = 1 To 2

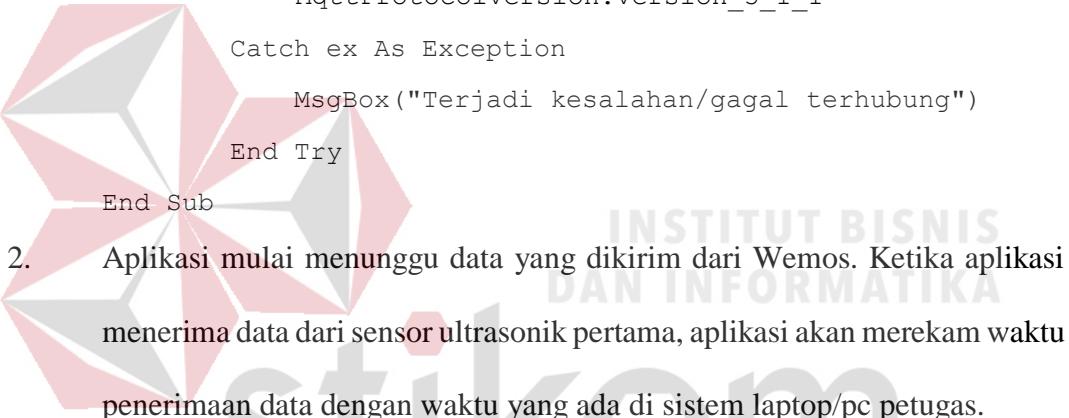
    mqttClient.Subscribe(New String() {"ultra/" & m}, New Byte()
    {MqttMsgBase.QOS_LEVEL_AT_MOST_ONCE})

Next

mqttClient.ProtocolVersion =
MqttProtocolVersion.Version_3_1_1

Catch ex As Exception
MsgBox("Terjadi kesalahan/gagal terhubung")

```



Source Code :

```

Public Sub terimaPesanUjiKeseimbangan(sender As Object, k As
MqttMsgPublishEventArgs)

    Dim pesan As String
    pesan = Encoding.UTF8.GetString(k.Message())
    Select Case k.Topic
        Case "ultra/1"
            If pesan = "on" Then
                If speedReset = 1 Then
                    datetimel = Date.Now
                    btn_mulai_keseim.Enabled = False
                    speedReset = 0
                End If
            End If
    End Select

```

```

    End If
    Exit Select
End Select
End Sub

```

3. Apabila patok tiang pembatas tersenggol atau tersenggol, Wemos akan mengirim data ke aplikasi dan aplikasi akan mengvisualkannya dengan simbol lingkaran yang berubah warna menjadi merah.

Source Code :

```

Public Sub terimaPesanUjiKeseimbangan(sender As Object, k As
MqttMsgPublishEventArgs)
    Dim pesan As String
    pesan = Encoding.UTF8.GetString(k.Message())
    Select Case k.Topic
        Case "getar/1"
            If pesan = "on" Then
                If getar1 = 1 Then
                    k_1.Visible = False
                    patok = patok + 1
                    getar1 = 0
                End If
            End If
            Exit Select
        Case "getar/2"
            If pesan = "on" Then
                If getar2 = 1 Then
                    k_2.Visible = False
                    patok = patok + 1
                    getar2 = 0
                End If
            End If
            Exit Select
        ...
        Case "getar/18"
    End Select

```

```

    If pesan = "on" Then
        If getar18 = 1 Then
            k_18.Visible = False
            patok = patok + 1
            getar18 = 0
        End If
    End If
    Exit Select
End Select
End Sub

```

4. Ketika sensor ultrasonik kedua terlewati (garis finish), aplikasi kembali merekam waktu yang ada di sistem pc/laptop petugas.

Source Code :

```

Public Sub terimaPesanUjiKeseimbangan(sender As Object, k As
MqttMsgPublishEventArgs)
    Dim pesan As String
    pesan = Encoding.UTF8.GetString(k.Message())
    Select Case k.Topic
        Case "ultra/2"
            If pesan = "on" Then
                datetimenow = Date.Now
                secondsDiff =
                    DateDiff(DateInterval.Second,
                    datetimenow, datetimenow)
                kecepatanKeseimbangan()
                btn_mulai_keseim.Enabled = True
                btn_reset_keseim.Enabled = True
            End If
        Exit Select
    End Select
End Sub

```

5. Ketika data waktu pertama dan waktu kedua terkumpul, maka dapat dilakukan perhitungan untuk mencari kecepatan kendaraan peserta ujian.

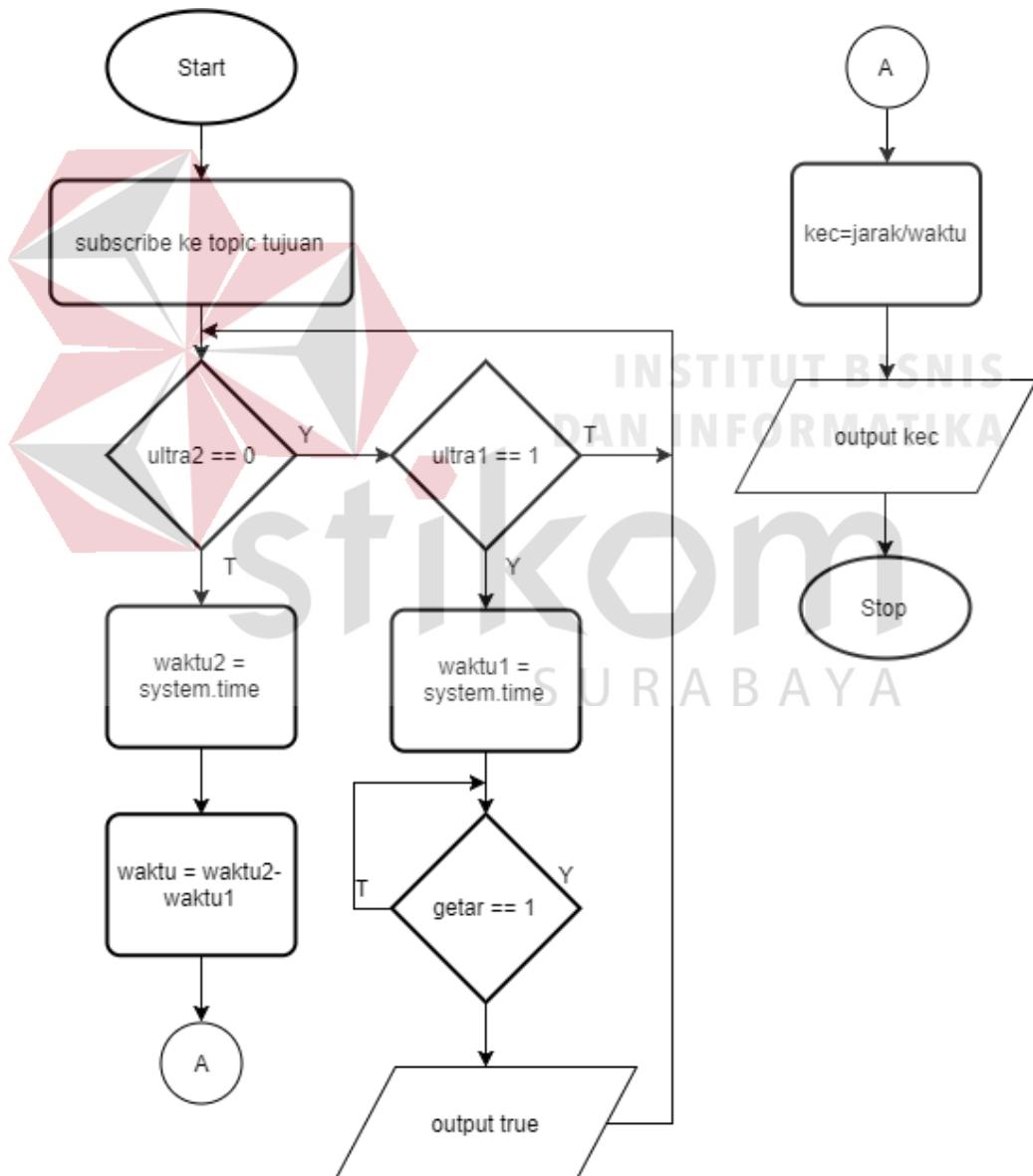
`secondsDiff = DateDiff(DateInterval.Second, datetime1, datetime2)`

`kecRataRata = 15 / secondsDiff`

`kecRataRata = kecRataRata * 0.001 / 0.000277778`

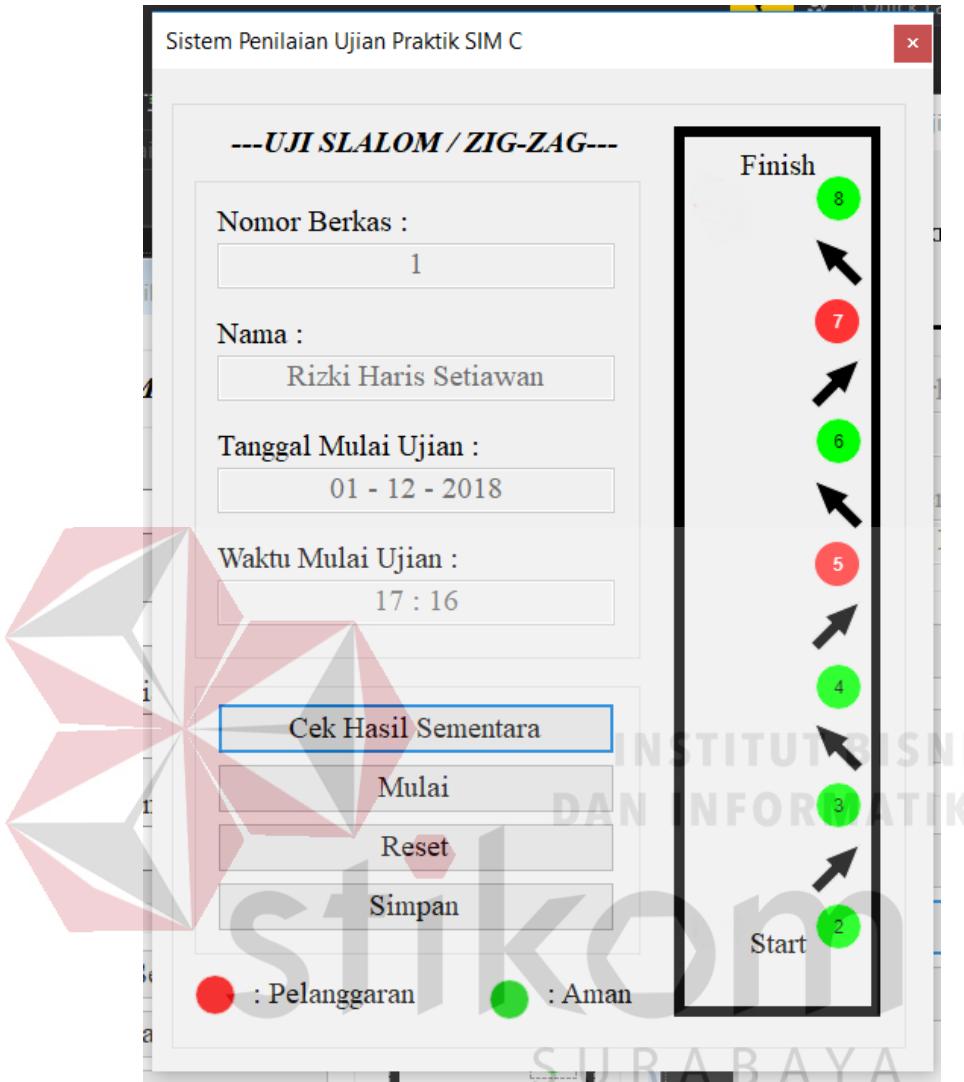
`kecRataRata = Round(kecRataRata, 2)`

6. Data yang sudah ada dapat disimpan / atau di reset jika ingin ujian ulang.



Gambar 3.17 Flowchart pada Menu Uji Keseimbangan

3.3.4.4 Perancangan Form Uji Zig-Zag



Gambar 3.18 Tampilan Menu Pengujian Uji Zig-zag

Pada gambar 3.18 form uji zig-zag terdapat beberapa komponen yang digunakan, seperti pada tabel 3.4 berikut :

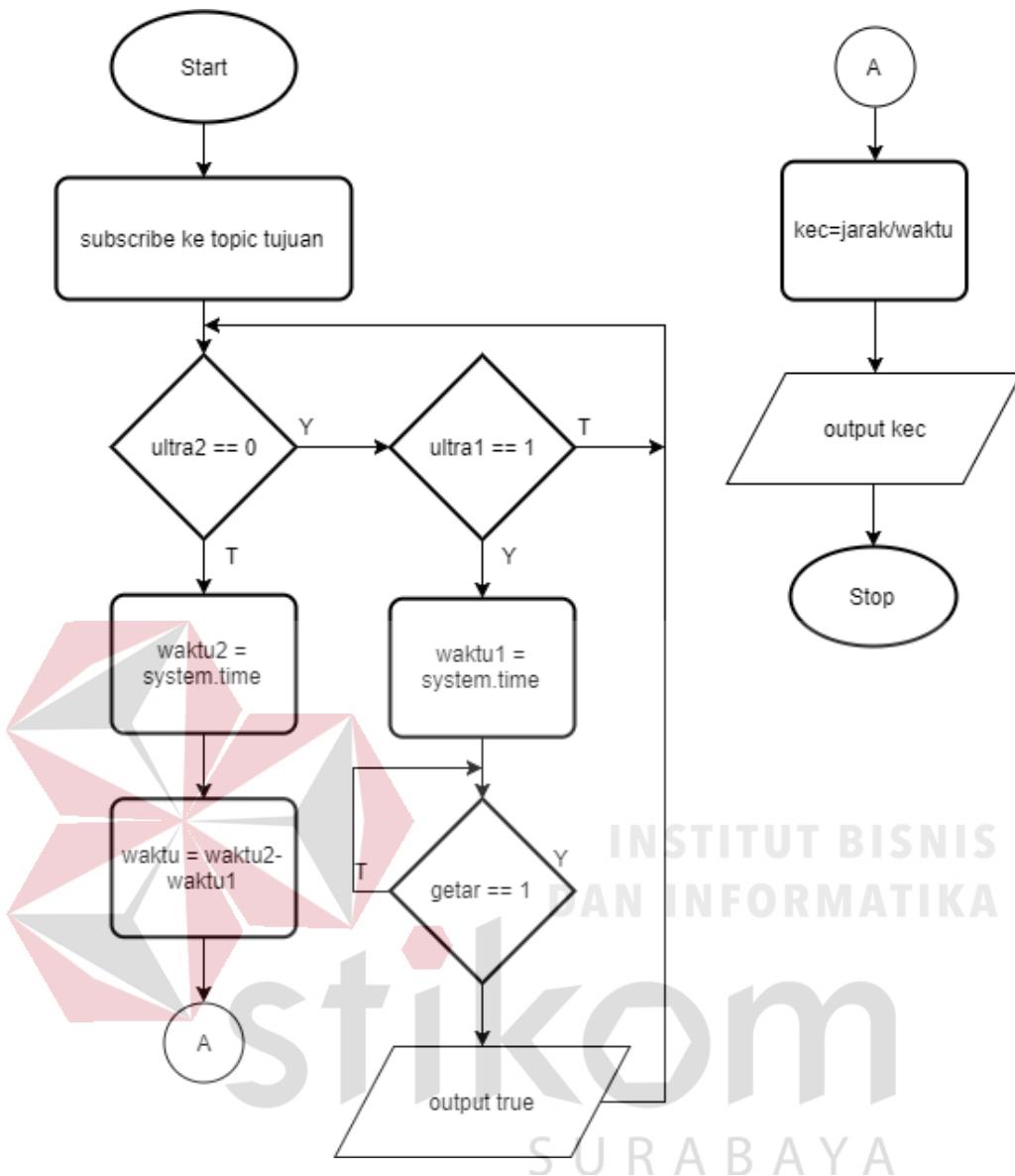
Tabel 3.4 Tabel Komponen Menu Pengujian Uji Zig-zag

Nama Komponen	Keterangan	Fungsi	Jumlah
Picture Box	Colour = “green”	Sebagai indikator apakah patok pada lintasan tesenggol atau tersenggol	7

Nama Komponen	Keterangan	Fungsi	Jumlah
Button	Text = “Cek hasil sementara”	Untuk melihat berapa patok yang tersenggol/terjatuh.	1
Button	Text = “Mulai”	Untuk memulai pengujian.	1
Button	Text = “Reset”	Untuk menghapus seluruh data.	1
Button	Text = “Simpan”	Untuk menyimpan data pada “cek hasil sementara”	1
Text Box	-	Untuk menampilkan nomor berkas, nama peserta, tanggal dan waktu mulai ujian.	1

Berikut adalah flowchart dari menu Uji Zig-zag :





Gambar 3.19 Flowchart pada Menu Uji Zig-zag

Proses dari algoritma flowchart gambar 3.19 diatas sebagai berikut :

1. Pada saat tombol mulai ditekan, aplikasi akan men-subscribe topic yang sama dengan sensor getar. Dan pengujian dapat dimulai saat itu juga.

Source code :

```

Private Sub btn_mulai_zigzag_Click(sender As Object, e As EventArgs) Handles btn_mulai_zigzag.Click
    TextBox4.Text = Date.Now.ToString(" HH : mm ")
    Try
  
```

```

CheckForIllegalCrossThreadCalls = False

AddHandler mqttClient.MqttMsgPublishReceived,
AddressOf terimaPesanUjiZigZag

For i As Integer = 2 To 8

    mqttClient.Subscribe(New String() {"getar/" &
    i}, New Byte()
    {MqttMsgBase.QOS_LEVEL_AT_MOST_ONCE})

Next

For i As Integer = 1 To 2

    mqttClient.Subscribe(New String() {"ultra/" &
    i}, New Byte()
    {MqttMsgBase.QOS_LEVEL_AT_MOST_ONCE})

Next

mqttClient.ProtocolVersion =
MqttProtocolVersion.Version_3_1_1

Catch ex As Exception
    MsgBox("Terjadi kesalahan/gagal terhubung")
End Try
End Sub

```

2. Aplikasi mulai menunggu data yang dikirim dari Wemos.
3. Apabila patok tiang pembatas tersenggol atau tersenggol, Wemos akan mengirim data ke aplikasi dan aplikasi akan mengvisualkannya dengan simbol lingkaran yang berubah warna menjadi merah.

Source Code :

```

Public Sub terimaPesanUjiZigZag(sender As Object, l As
MqttMsgPublishEventArgs)

Dim pesan As String
pesan = Encoding.UTF8.GetString(l.Message())
Select Case l.Topic
Case "getar/2"
If pesan = "on" Then
If getar2 = 1 Then
z_2.Visible = False
patok = patok + 1
getar2 = 0
End If
End If
End Sub

```

```

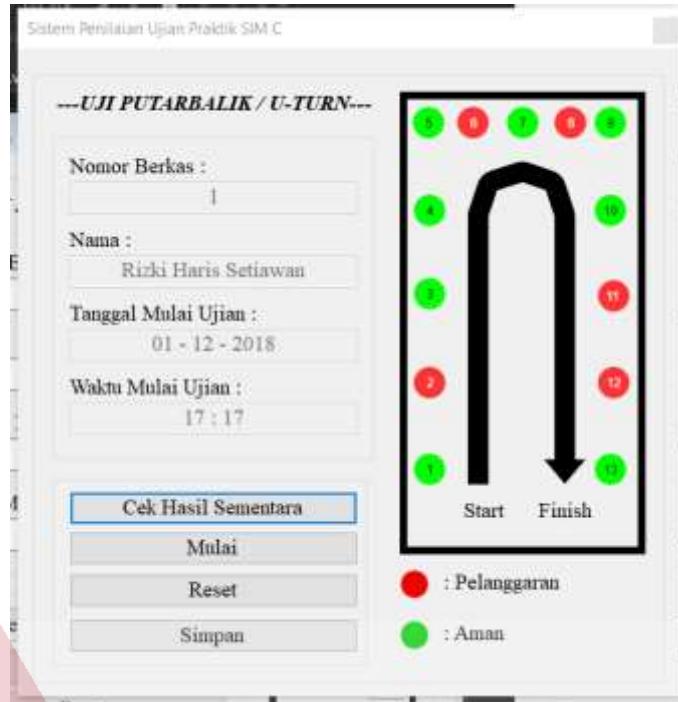
        End If
    End If
    Exit Select
Case "getar/3"
If pesan = "on" Then
    If getar3 = 1 Then
        z_3.Visible = False
        patok = patok + 1
        getar3 = 0
    End If
End If
Exit Select
...
Case "getar/8"
If pesan = "on" Then
    If getar8 = 1 Then
        z_8.Visible = False
        patok = patok + 1
        getar8 = 0
    End If
End If
Exit Select
End Select
End Sub

```

4. Data yang sudah ada dapat disimpan / atau di reset jika ingin ujian ulang.

3.3.4.5 Perancangan Form Uji U-turn / Putar balik

Pada gambar 3.20 form uji keseimbangan terdapat beberapa komponen yang digunakan, seperti pada tabel 3.5 :

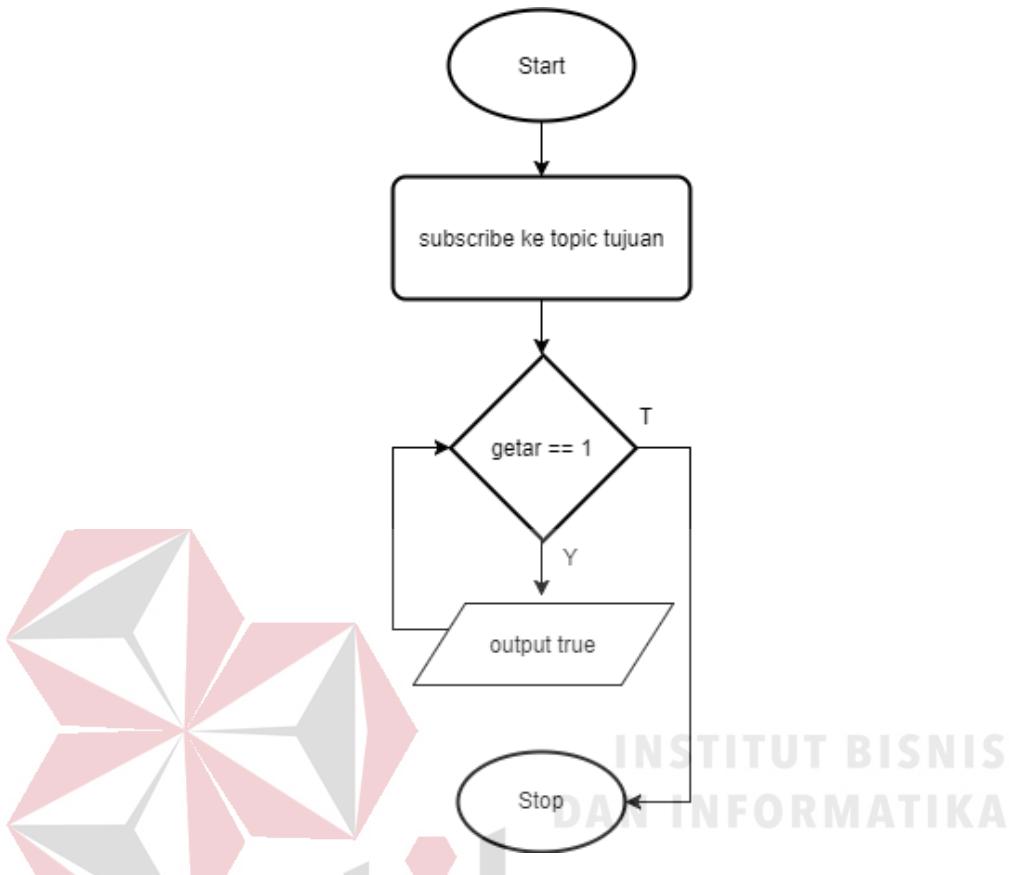


Gambar 3.20 Tampilan Menu Pengujian Uji U-turn

Tabel 3. 5 Tabel Komponen Menu Pengujian Uji U-Turn

Nama Komponen	Keterangan	Fungsi	Jumlah
Picture Box	Colour = “green”	Sebagai indikator apakah patok pada lintasan tesenggol atau tersenggol	13
Button	Text = “Cek hasil sementara”	Untuk melihat berapa patok yang tersenggol/terjatuh.	1
Button	Text = “Mulai”	Untuk memulai pengujian.	1
Button	Text = “Reset”	Untuk menghapus seluruh data.	1
Button	Text = “Simpan”	Untuk menyimpan data pada “cek hasil sementara”	1
Text Box	-	Untuk menampilkan nomor berkas, nama peserta, tanggal dan waktu mulai ujian.	1

Berikut adalah flowchart dari menu Uji U-turn :



Gambar 3. 21 Flowchart menu uji U-turn

Proses dari algoritma flowchart gambar 3.21 diatas sebagai berikut :

1. Pada saat tombol mulai ditekan, aplikasi akan men-subscribe topic yang sama dengan sensor getar. Dan pengujian dapat dimulai saat itu juga.

Source Code :

```

Private Sub btn_mulai_uturn_Click(sender As Object, e As EventArgs) Handles btn_mulai_uturn.Click
    Try
        CheckForIllegalCrossThreadCalls = False
        AddHandler mqttClient.MqttMsgPublishReceived,
        AddressOf terimaPesanUjiUturn
        For i As Integer = 1 To 13
    End Sub
  
```

```

        mqttClient.Subscribe(New String() {"getar/" &
        i}, New Byte()
        {MqttMsgBase.QOS_LEVEL_AT_MOST_ONCE})

    Next

    mqttClient.ProtocolVersion =
        MqttProtocolVersion.Version_3_1_1

    TextBox4.Text = Date.Now.ToString(" HH : mm ")

    Catch ex As Exception

        MsgBox("Terjadi kesalahan/gagal terhubung")

    End Try

End Sub

```

2. Aplikasi mulai menunggu data yang dikirim dari Wemos.
3. Apabila patok tiang pembatas tersenggol atau tersenggol, Wemos akan mengirim data ke aplikasi dan aplikasi akan mengvisualkannya dengan simbol lingkaran yang berubah warna menjadi merah.

Source Code :

```

Public Sub terimaPesanUjiUturn(sender As Object, l As
MqttMsgPublishEventArgs)

    Dim pesan As String
    pesan = Encoding.UTF8.GetString(l.Message())
    Select Case l.Topic

        Case "getar/1"
            If pesan = "on" Then
                If getar1 = 1 Then
                    u_1.Visible = False
                    patok = patok + 1
                    getar1 = 0
                End If
            End If
            Exit Select
        Case "getar/2"
            If pesan = "on" Then
                If getar2 = 1 Then
                    u_2.Visible = False
                End If
            End If
    End Select

```

```

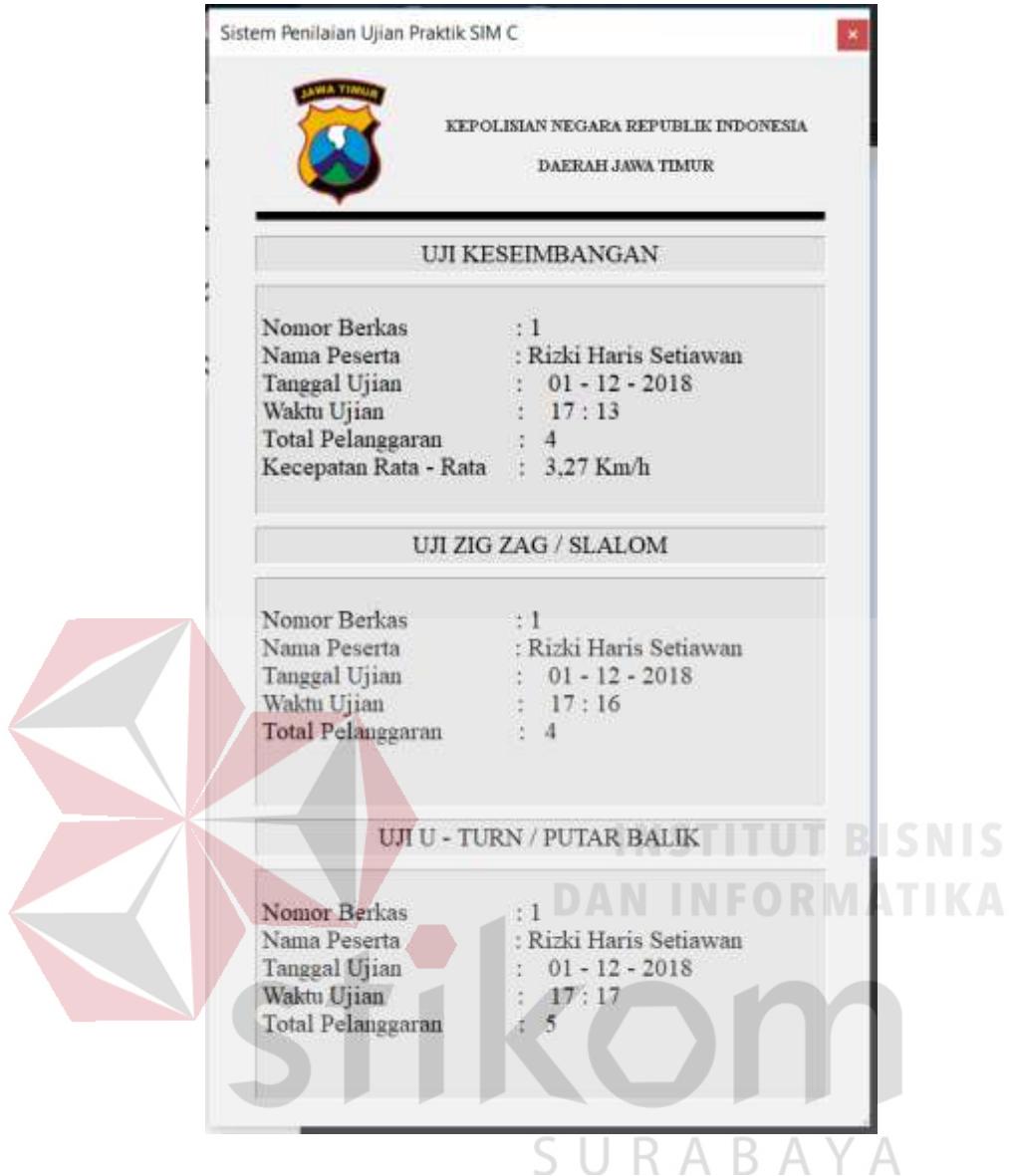
        patok = patok + 1
        getar2 = 0
    End If
End If
Exit Select
...
Case "getar/13"
If pesan = "on" Then
    If getar13 = 1 Then
        u_13.Visible = False
        patok = patok + 1
        getar13 = 0
    End If
End If
Exit Select
End Select
End Sub

```

4. Data yang sudah ada dapat disimpan / atau di reset jika ingin ujian ulang.

3.3.4.6 Perancangan Form Hasil Akhir

Data dari seluruh hasil pengujian akan disimpan kedalam *variabel global* yang kemudian akan ditampilkan pada Form Hasil Akhir seperti pada gambar 3.22. Apabila aplikasi dikeluarkan data hasil pengujian tidak tersimpan kedatabase atau semacamnya, karena data hasil pengujian pada aplikasi hanya disimpan sementara (*temporary*). Pada gambar 3.22 form uji keseimbangan terdapat beberapa komponen yang digunakan, seperti pada tabel 3.6 :



Gambar 3.22 Tampilan Hasil Akhir

Tabel 3.6 Tabel Komponen Menu Hasil Akhir

Nama Komponen	Keterangan	Fungsi	Jumlah
Label	Text = “Uji Keseimbangan”	-	1
Label	Text = “UJI Zig-zag”	-	1
Label	Text = “UJI U-Turn/Putar Balik”	-	1

Nama Komponen	Keterangan	Fungsi	Jumlah
Label	Text = variabel_Keseimbangan	Menampilkan data yang diperoleh dari hasil ujian (patok,tanggal,jam dan kecepatan) uji keseimbangan	1
Label	Text = variabel_Zig_zag	Menampilkan data yang diperoleh dari hasil ujian (patok,tanggal dan jam) uji zig-zag	1
Label	Text = variabel_Uturn	Menampilkan data yang diperoleh dari hasil ujian (patok,tanggal dan jam) uji uturn/putar balik	1

3.4.4 Perhitungan Kecepatan

Perhitungan kecepatan kendaraan peserta ujian hanya terdapat pada form uji keseimbangan ,karena hanya pengujian tersebut yang membutuhkan kecepatan dari kendaraan peserta. Pada lintasan uji keseimbangan, ketika peserta melewati garis awal(start) aplikasi akan mendeteksi dan program akan mengambil waktu (jam,menit dan detik) dari laptop/komputer pada aplikasi berjalan. Setelah peserta melewati garis akhir(finish) aplikasi akan kembali mengambil data waktu(jam, menit dan detik) dari laptop/komputer yang digunakan. Setelah data awal dan data akhir terkumpul, program akan mencari selisih dari waktu awal dan waktu akhir seperti contoh berikut :

$$\text{Waktu1} = 08:30:50$$

$$\text{Waktu2} = 08:31:00$$

$$\text{Waktu_interval} = \text{waktu2} - \text{waktu1}$$

$$= 08:31:00 - 08:30:50$$

$$= 10 \text{ detik}$$

Pada perhitungan diatas dalam pencarian waktu_interval tidak dihitung manual, melainkan menggunakan built-in function yang sudah tersedia pada pemrograman Visual Basic .Net . Hasil dari built-in function diatas langsung menjadi detik, sehingga penulis hanya perlu meneruskan ke langkah berikutnya untuk mencari kecepatan.

Pada lintasan Uji keseimbangan panjang lintasan adalah 12 m jadi dapat ditulis rumus mencari kecepatan :

$$\begin{aligned} &\text{Jarak lintasan} = 12 \text{ meter} \\ &\text{Waktu tempuh} = 10 \text{ detik} \\ &\text{Kecepatan} = \frac{\text{Jarak lintasan}}{\text{Waktu tempuh}} \\ &= \frac{12 \text{ meter}}{10 \text{ detik}} \\ &\text{Kecepatan} = 1.2 \text{ m / s} \end{aligned}$$

Hasil dari perhitungan diatas masih menggunakan meter per detik, maka selanjutnya diubah menjadi kilometer per jam seperti berikut :

$$1 \text{ meter} = 0.001 \text{ km}$$

$$1 \text{ detik} = 0.0002777777777777778 \text{ jam}$$

Maka diperoleh :

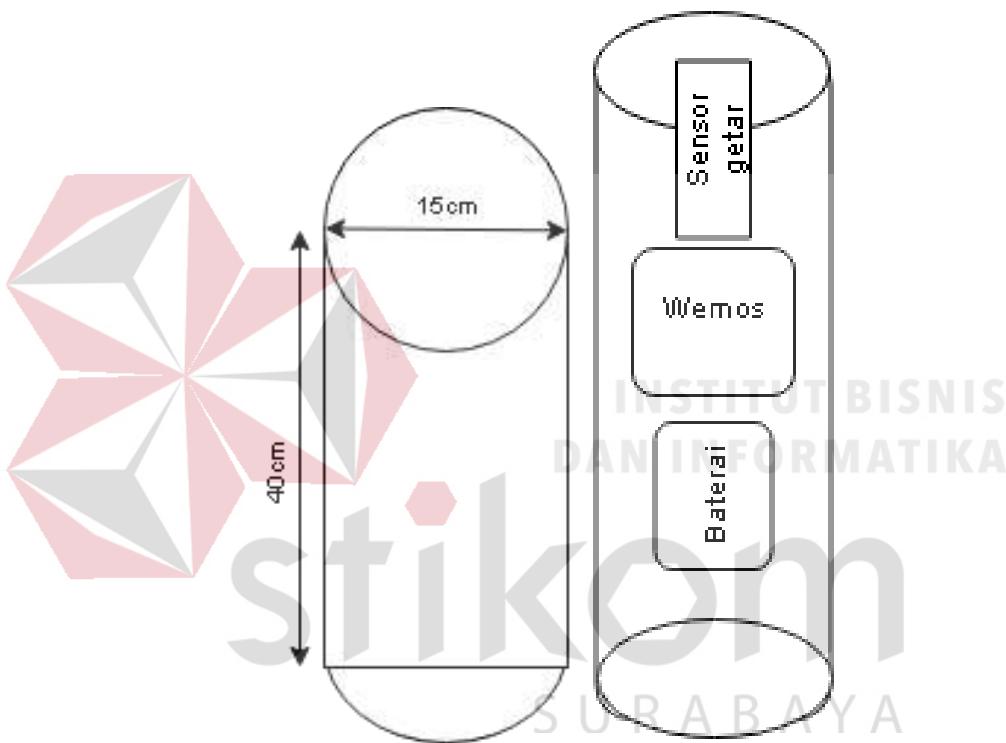
$$\text{Kecepatan} = 1.2 \times \frac{0.001}{0.0002777777777777778}$$

$$\text{Kecepatan} = 4.33 \text{ km/jam}$$

3.5 Perancangan Mekanik

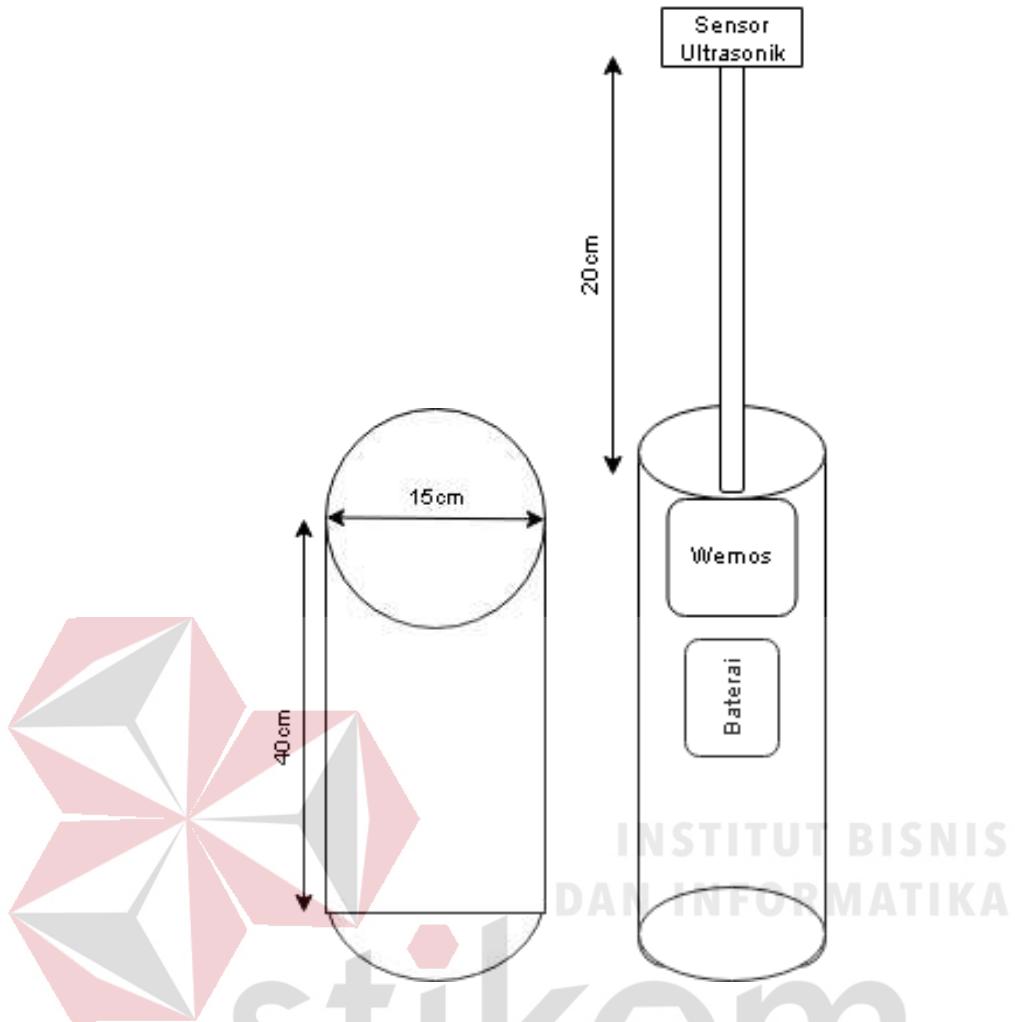
3.5.1 Patok Tiang Pembatas Sensor Getar dan Sensor Ultrasonik

Bersasarkan syarat sarana patok untuk ujian praktik, maka pembatas/patok di desain sebagai berikut.



Gambar 3.23 Perancangan Patok Tiang Pembatas Sensor Getar

Seperti pada Gambar 3.24, tiang pembatas sensor getar memiliki tinggi 40 cm dan diameter antara 10-15 cm. Didalam patok terdapat Wemos, Sensor getar, dan Baterai. Sensor terhubung ke Wemos dan mendapat daya dari baterai kurang lebih 5V. Peletakan patok/pembatas di arahkan ke lintasan tepat pada sisi yang tertempel sensor getar, hal ini dilakukan agar pembacaan sensor dapat lebih maksimal.



Gambar 3.24 Desain patok/tiang pembatas sensor ultrasonik

Pada gambar 3.25 tiang pembatas/patok yang digunakan untuk sensor ultrasonik memiliki dimeter 15cm dan tinggi 40cm sama seperti patok/pembatas sensor getar. Pada patok sensor ultrasonik ditambahkan tinggi ~20cm agar sensor ultrasonik dapat mendeteksi kendaraan saat melewati garis start. Penambahan ini dilakukan karena peletakan patok sensor ultrasonik berada dibelakang patok sensor getar, apabila tidak ditambahkan tinggi ~20cm sensor ultrasonik akan terhalang oleh patok sensor getar. Pada lintasan ditempatkan dua buah sensor ultrasonik di awal lintasan dan akhir lintasan. Berikut adalah gambar dari patok/tiang pembatas sensor ultrasonik.

3.5.2 Lintasan Uji Keseimbangan



Gambar 3.25 Blok Diagram Lintasan Uji Keseimbangan

Pada blok diagram gambar3.26 diatas menjelaskan bahwa :

1. Lintasan uji keseimbangan menggunakan 20 unit Wemos D1 Mini, 18 buah sensor getar sw-420 dan dua buah sensor ultrasonik hc-sr04.
2. Satu patok tiang pembatas berisikan satu buah Wemos dan satu buah sensor getar/sensor ultrasonik.
3. Raspberry, Wemos, dan aplikasi tersambung ke akses point yang sama.

3.5.3 Lintasan Uji Zig-zag



Gambar 3.26 Blok Diagram Lintasan Uji Zig-zag

Pada blok diagram gambar 3.27 diatas menjelaskan bahwa :

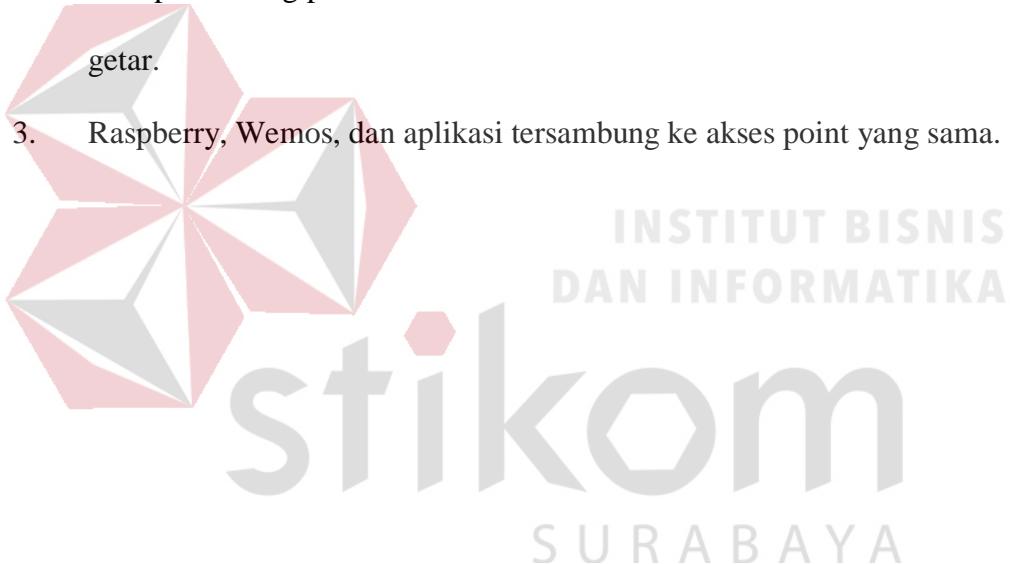
1. Lintasan uji keseimbangan menggunakan 9 unit Wemos D1 Mini dan 7 buah sensor getar sw-420 dan 2 buah sensor ultrasonik hc-sr04

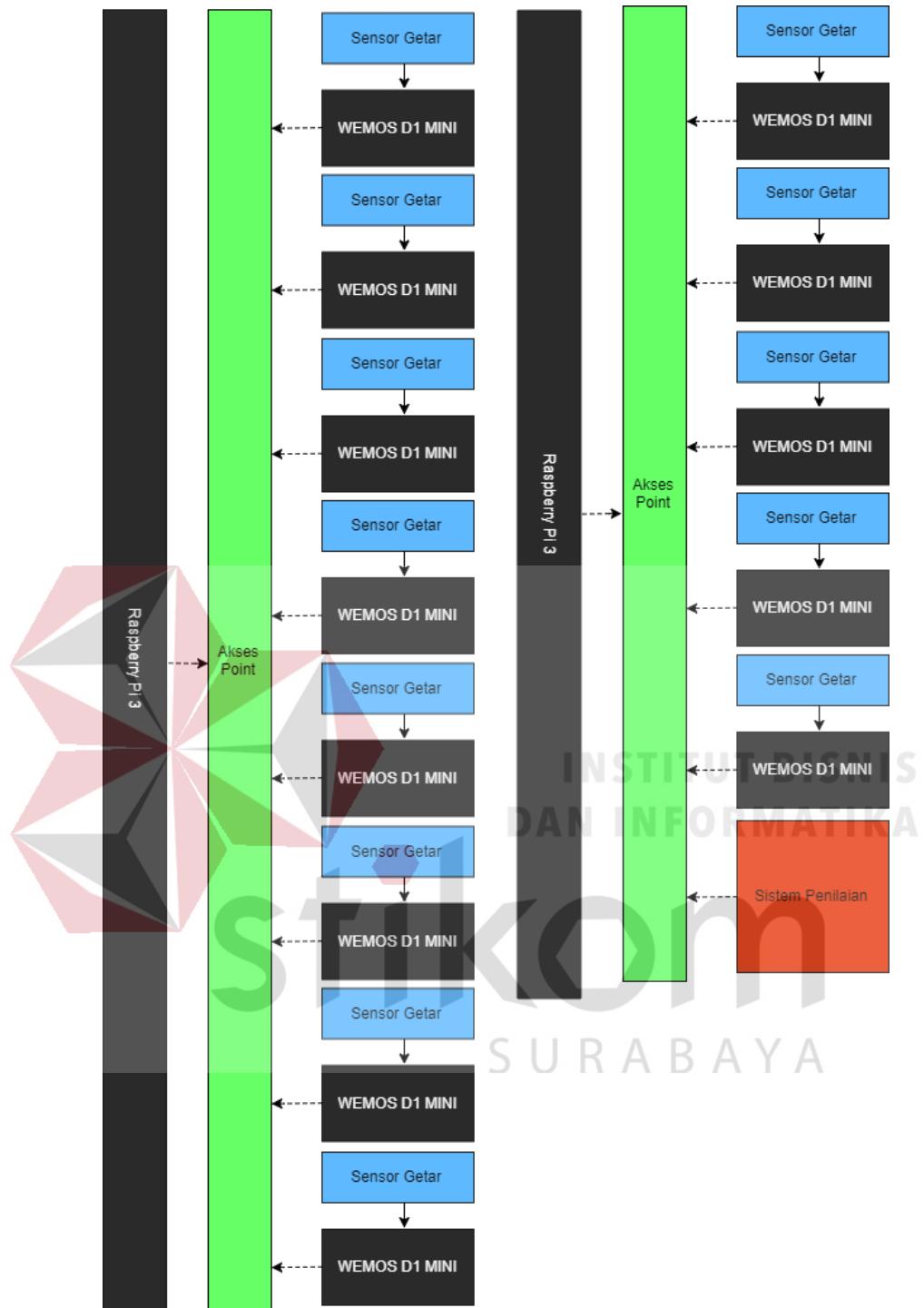
2. Satu patok tiang pembatas berisikan satu buah Wemos dan satu buah sensor getar.
3. Raspberry, Wemos, dan aplikasi tersambung ke akses point yang sama.

3.5.4 Lintasan Uji U-turn/Putar Balik

Pada blok diagram gambar 3.28 menjelaskan bahwa :

1. Lintasan uji keseimbangan menggunakan 13 unit Wemos D1 Mini dan 13 buah sensor getar sw-420.
2. Satu patok tiang pembatas berisikan satu buah Wemos dan satu buah sensor getar.
3. Raspberry, Wemos, dan aplikasi tersambung ke akses point yang sama.





Gambar 3.27 Blok Diagram Lintasan Uji Uturn / Putar balik

BAB IV

HASIL DAN PEMBAHASAN

Bab ini akan membahas pengujian yang telah dilakukan penulis terhadap perangkat keras serta perangkat lunak dan sistem secara keseluruhan yang telah selesai dibuat untuk mengetahui kerja dari sistem berjalan dengan baik atau tidak.

4.1 Pengujian Wemos D1 Mini

4.1.1 Tujuan

Tujuan dari pengujian ini adalah untuk memastikan apakah Wemos D1 Mini dapat menjalankan program dari Arduino IDE dengan baik. Indikator keberhasilannya adalah program dapat diupload dengan baik tanpa ada error sama sekali.

4.1.2 Alat yang Digunakan

Peralatan yang dibutuhkan untuk pengujian alat ini adalah sebagai berikut :

1. Wemos D1 Mini
2. Komputer / Laptop
3. Kabel USB
4. Software Arduino IDE

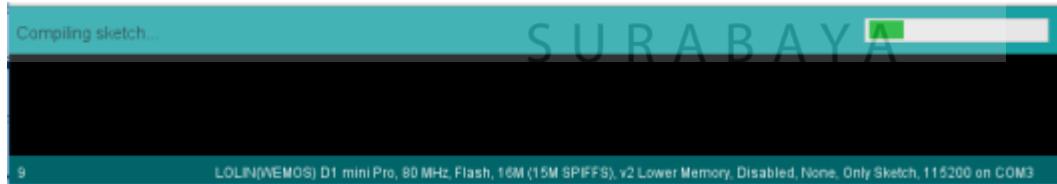
4.1.3 Prosedur Pengujian

Langkah – langkah yang dilakukan untuk melakukan pengujian sistem adalah seperti berikut :

1. Menghidupkan Komputer / Laptop.
2. Menyambungkan Komputer / Laptop pada Wemos D1 Mini dengan menggunakan kabel USB.
3. Membuka program Arduino IDE pada Komputer / Laptop.
4. Konfigurasi pada menu tools, seperti *Board : LOLINWEMOSD1MINIPRO;*
Port : COM3 (tergantung komputer masing-masing).
5. Klik *verify* untuk mengecek seluruh *Source Code* yang akan diupload ke Wemos, dan *upload* untuk memulai proses.
6. Tunggu dan amati log dari Arduino IDE.

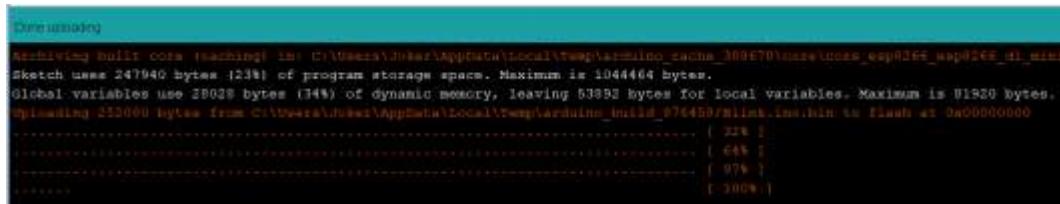
4.1.4 Hasil Pengujian

Pada gambar 4.1 merupakan proses ketika program sedang di upload ke Wemos D1 Mini , terdapat tulisan “*Compiling sketch*” saat proses berlangsung.



Gambar 4.1 Proses Upload Program pada Arduino IDE

Gambar 4.2 adalah tampilan pada log Arduino IDE setelah selesai upload program ke Wemos D1 Mini. Tulisan “*Done compiling*” menandakan bahwa program berhasil diupload ke Wemos D1 Mini



Gambar 4.2 Arduino IDE Berhasil Upload Program ke Wemos D1 Mini

Tabel 4.1 memperlihatkan data hasil pengujian 20 buah Wemos D1 Mini. Masing-masing Wemos D1 Mini dilakukan percobaan untuk mengetahui apakah Wemos dapat digunakan atau tidak.

Tabel 4.1 Hasil Pengujian 20 buah Wemos D1 Mini

Wemos ke-	Hasil	Wemos ke-	Hasil
1	Berhasil ter-upload	11	Berhasil ter-upload
2	Berhasil ter-upload	12	Berhasil ter-upload
3	Berhasil ter-upload	13	Berhasil ter-upload
4	Berhasil ter-upload	14	Berhasil ter-upload
5	Berhasil ter-upload	15	Berhasil ter-upload
6	Berhasil ter-upload	16	Berhasil ter-upload
7	Berhasil ter-upload	17	Berhasil ter-upload
8	Berhasil ter-upload	18	Berhasil ter-upload
9	Berhasil ter-upload	19	Berhasil ter-upload
10	Berhasil ter-upload	20	Berhasil ter-upload

4.2 Pengujian Sensor Getar SW-420

4.2.1 Tujuan

Tujuan dari pengujian ini adalah untuk mengatur sensitifitas dari sensor getar dan mengetahui apakah sensor bekerja dengan baik atau tidak. Indikator keberhasilannya adalah dapat mengoutputkan “1” ketika tersenggol dan “0” ketika tidak tersenggol.

4.2.2 Alat yang Digunakan

Peralatan yang dibutuhkan untuk pengujian alat ini adalah sebagai berikut :

1. Wemos D1 Mini
2. Komputer / Laptop
3. Kabel USB
4. Kabel Jumper
5. Sensor Getar SW-420

4.2.3 Prosedur Pengujian

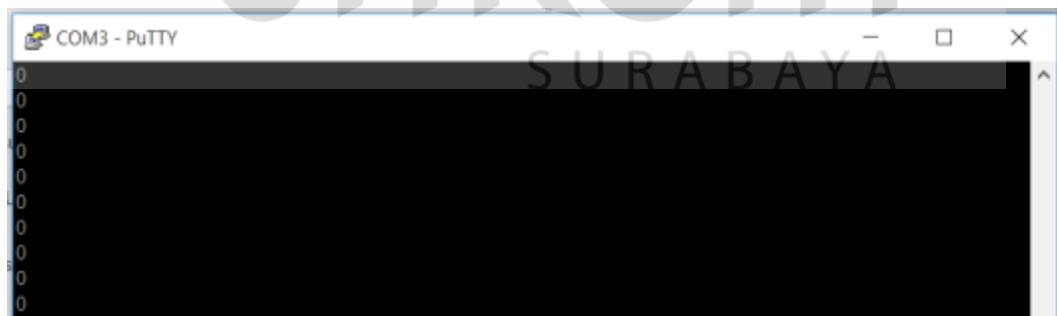
Langkah – langkah yang dilakukan untuk melakukan pengujian sistem adalah seperti berikut :

1. Menghubungkan pin VCC dan GND yang ada di Sensor Getar SW-420 dan Wemos D1 Mini menggunakan kabel jumper.
2. Menghubungkan pin D0 yang ada di Sensor Getar SW-420 ke pin D3 Wemos D1 Mini menggunakan kabel jumper.
3. Menghidupkan Komputer / Laptop.

4. Menyambungkan Komputer / Laptop pada Wemos D1 Mini dengan menggunakan kabel USB.
5. Membuka program Arduino IDE pada Komputer / Laptop.
6. Upload program sensor_getar.
7. Buka serial monitor untuk melihat output dari Sensor Getar SW-420.

4.2.4 Hasil Pengujian

Pada Sensor Getar SW-420 terdapat variabel resistor yang dapat diatur untuk mengatur sensitivitas dari sensor getar. Putaran berlawanan dengan arah jarum jam dapat membuat Sensor menjadi semakin sensitif dan putaran dengan arah jarum jam dapat membuat Sensor menjadi semakin tidak sensitif. Dapat dilihat pada gambar 4.3 , Ketika Sensor tidak menerima getaran maka akan memberi nilai “0” sedangkan ketika Sensor menerima getaran maka akan memberi nilai “1” seperti pada gambar 4.4. Output dari Sensor Getar SW-420 merupakan data Digital maka hanya dapat mengeluarkan “0” atau “1”.



Gambar 4.3 Ketika Sensor Tidak Menerima Getaran



Gambar 4.4 Ketika Sensor Menerima Getaran

4.3 Pengujian Sensor Ultrasonik HC-SR04

4.3.1 Tujuan

Tujuan dari pengujian ini adalah untuk melihat data output pada Sensor Ultrasonik HC-SR04. Indikator keberhasilannya adalah dapat memperlihatkan jarak dari sensor ultrasonik ke suatu objek dalam cm.

4.3.2 Alat yang Digunakan

Peralatan yang dibutuhkan untuk pengujian alat ini adalah sebagai berikut :

1. Wemos D1 Mini
2. Komputer / Laptop
3. Kabel USB
4. Kabel Jumper
5. Patok Sensor Ultrasonik HC-SR04

4.3.3 Prosedur Pengujian

Langkah – langkah yang dilakukan untuk melakukan pengujian sistem adalah seperti berikut :

1. Menghubungkan pin VCC dan GND yang ada di Sensor Ultrasonik HC-SR04 dan Wemos D1 Mini menggunakan kabel jumper.

2. Menghubungkan pin Trig yang ada di Sensor Ultrasonik HC-SR04 ke pin D5 Wemos D1 Mini menggunakan kabel jumper.
3. Menghubungkan pin Echo yang ada di Sensor Ultrasonik HC-SR04 ke pin D6 Wemos D1 Mini menggunakan kabel jumper.
4. Menghidupkan Komputer / Laptop.
5. Menyambungkan Komputer / Laptop pada Wemos D1 Mini dengan menggunakan kabel USB.
6. Membuka program Arduino IDE pada Komputer / Laptop.
7. Upload program sensor_ultrasonik.
8. Buka serial monitor untuk melihat output dari Sensor Ultrasonik HC-SR04.

4.3.4 Hasil Pengujian

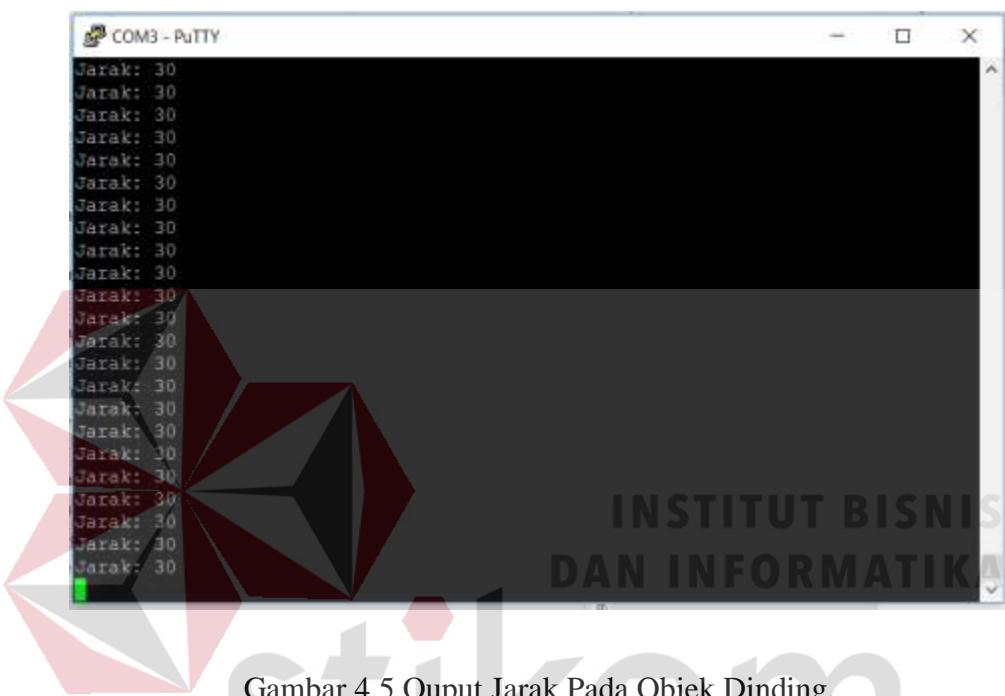
Berikut adalah tabel 4.2 hasil dari pengujian yang sudah dilakukan sebanyak 10 kali dengan jarak yang berbeda-beda.

Tabel 4. 2 Percobaan Sensor Ultrasonik 1 dan 2

No.	Sensor Ultrasonik 1	Sensor Ultrasonik 2	Jarak benda
1.	54 Cm	54 Cm	54 Cm
2.	42 Cm	42 Cm	42 Cm
3.	33 Cm	33 Cm	33 Cm
4.	22 Cm	22 Cm	22 Cm
5.	15 Cm	15 Cm	15 Cm
6.	17 Cm	17 Cm	17 Cm
7.	5 Cm	5 Cm	5 Cm
8.	12 Cm	12 Cm	12 Cm

No.	Sensor Ultrasonik 1	Sensor Ultrasonik 2	Jarak benda
9.	35 Cm	35 Cm	35 Cm
10.	24 Cm	24 Cm	24 Cm

Dapat dilihat pada gambar 4.5 dimana nilai jarak dari suatu objek terlihat stabil



Gambar 4.5 Ouput Jarak Pada Objek Dinding.

Untuk menguatkan bukti pengukuran, dilakukan pengukuran dengan penggaris dengan objek dinding seperti pada gambar 4.6 . Data pada serial monitor terlihat sama dengan pengukuran menggunakan penggaris.



Gambar 4.6 Pengukuran Menggunakan Penggaris

4.4 Pengujian Pengukur Kecepatan pada Lintasan Uji Keseimbangan

4.4.1 Tujuan

Pengujian dari proses ini untuk mengetahui apakah data dari sensor Ultrasonik HC-SR04 telah berhasil dikirim ke broker / server. Indikator keberhasilannya adalah Aplikasi Sistem dapat menjalankan proses perhitungan kecepatan kendaraan peserta ujian dan menampilkannya.

4.4.2 Alat yang Digunakan

Peralatan yang dibutuhkan untuk pengujian alat ini adalah sebagai berikut :

1. Raspberry Pi 3
2. Komputer / Laptop
3. Andromax MiFi
4. Patok Sensor Ultrasonik

5. Kendaraan Roda Dua
6. Aplikasi Penilaian Ujian Praktik SIM C

4.4.3 Prosedur Pengujian

Langkah – langkah yang dilakukan untuk melakukan pengujian sistem adalah seperti berikut :

1. Hidupkan Raspberry Pi 3, Sambungkan ke Akses Point Andromax MiFi.
2. Upload program patok sensor ultrasonik ke patok.
3. Hidupkan patok dengan tiga buah baterai 1.5V sebagai daya.
4. Letakan dua patok sensor ultrasonik dengan jarak 15m satu sama lain.
5. Hidupkan Laptop dan sambungkan ke Akses Point Andromax MiFi.
6. Buka Aplikasi Sistem dan masukan biodata.
7. Masuk kemenu Broker dan masukan IP : 192.168.1.111 lalu sambungkan.
8. Masuk kemenu pengujian Keseimbangan
9. Klik mulai dan jalankan pengujian.
10. Pilih “Cek Hasil Sementara” untuk melihat kecepatan.

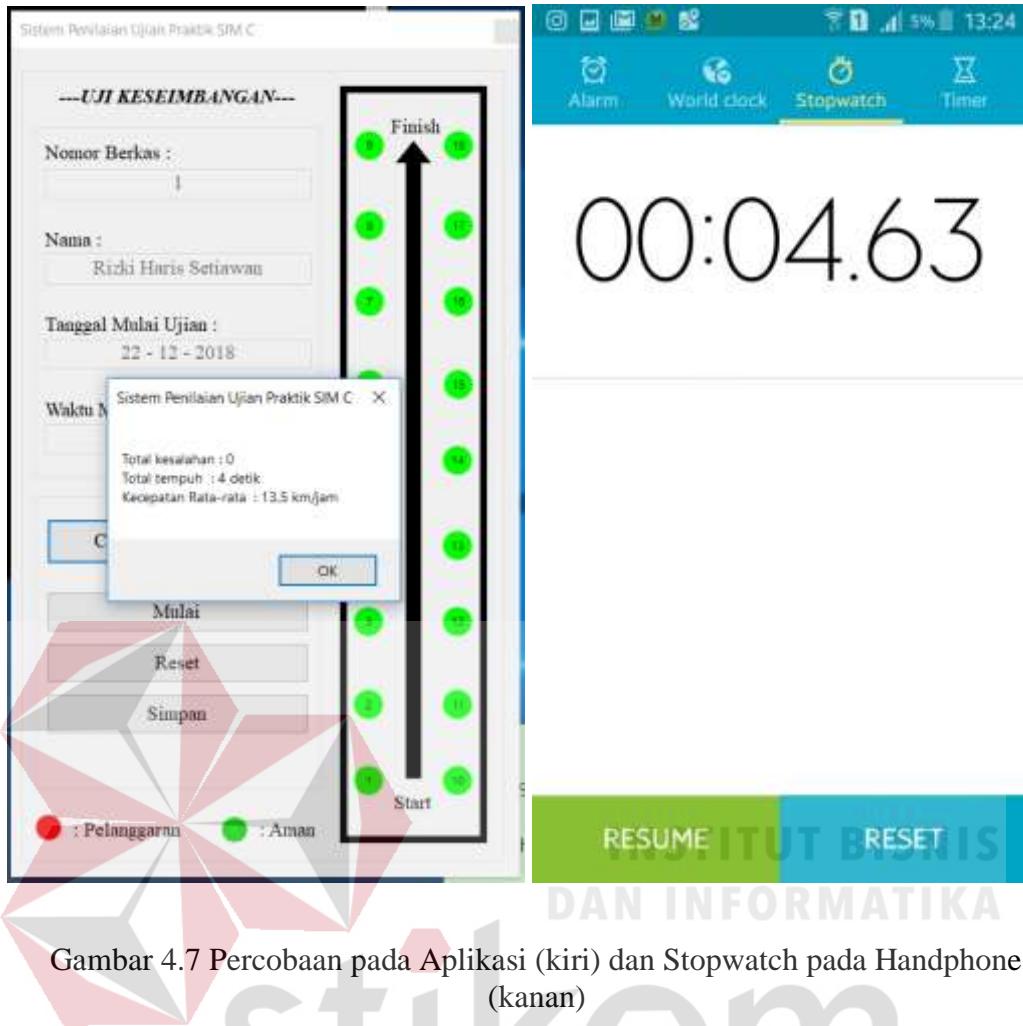
4.4.4 Hasil Pengujian

Pengujian dilakukan sebanyak 10 kali pada lintasan Uji Keseimbangan dengan menggunakan motor Yamaha Jupiter MX sebagai kendaraan roda dua. Berikut adalah hasil dari pengujian yang sudah dilakukan :

Tabel 4.3 Percobaan Pengukur Kecepatan

No	Sensor Ultrasonik		Selisih (Ultra2-Ultra1)	Stopwatch (menit : detik : milidetik)	Kecepatan (jarak/waktu)
	Ultra1 (jam : menit : detik)	Ultra2 (jam : menit : detik)			
1.	13 : 09 : 40	13 : 09 : 45	5 Detik	00 : 05 : 70	10.8 Km/h
2.	13 : 12 : 20	13 : 12 : 25	5 Detik	00 : 05 : 25	10.8 Km/h
3.	13 : 14 : 20	13 : 14 : 29	9 Detik	00 : 09 : 41	6 Km/h
4.	13 : 16 : 34	13 : 16 : 38	4 Detik	00 : 04 : 00	13.5 Km/h
5.	13 : 20 : 31	13 : 20 : 34	3 Detik	00 : 03 : 17	18 Km/h
6.	13 : 23 : 19	13 : 23 : 23	4 Detik	00 : 04 : 63	13.5 Km/h
7.	13 : 25 : 30	13 : 25 : 34	4 Detik	00 : 04 : 48	13.5 Km/h
8.	13 : 27 : 07	13 : 27 : 13	6 Detik	00 : 06 : 13	9 Km/h
9.	13 : 28 : 34	13 : 28 : 40	6 Detik	00 : 06 : 43	9 Km/h
10.	13 : 29 : 52	13 : 29 : 01	9 Detik	00 : 09: 47	6 Km/h

Pada saat melakukan pengujian, digunakan stopwatch sebagai pembanding pada selisih waktu pada Aplikasi Sistem agar dapat mengetahui tingkat ketepatan saat sensor Ultrasonik terlewati. Berikut adalah gambar perbandingan Aplikasi Sistem dengan stopwatch yang digunakan.



Gambar 4.7 Percobaan pada Aplikasi (kiri) dan Stopwatch pada Handphone (kanan)

4.5 Pengujian Komunikasi Data Antara Patok Sensor Getar Lintasan dengan Broker / Server

4.5.1 Tujuan

Pengujian dari proses ini untuk mengetahui apakah data dari sensor Getar SW-420 telah berhasil dikirim ke broker / *server*. Indikator keberhasilannya adalah Aplikasi Sistem dapat mengetahui apakah patok / tiang pembatas tersenggol atau tidak dan menampilkannya pada Aplikasi.

4.5.2 Alat yang Digunakan

Peralatan yang dibutuhkan untuk pengujian alat ini adalah sebagai berikut :

1. Raspberry Pi 3
2. Komputer / Laptop
3. Andromax MiFi
4. Patok Sensor Getar
5. Kendaraan Roda Dua
6. Aplikasi Penilaian Ujian Praktik SIM C

4.5.3 Prosedur Pengujian

Langkah – langkah yang dilakukan untuk melakukan pengujian sistem adalah seperti berikut :

1. Hidupkan Raspberry Pi 3, Sambungkan ke Akses Point Andromax MiFi.
2. Upload program patok sensor getar ke patok.
3. Hidupkan patok dengan tiga buah baterai 1.5V sebagai daya.
4. Letakan 18 patok sensor getar sesuai dengan lintasan uji Keseimbangan.
5. Hidupkan Laptop dan sambungkan ke Akses Point Andromax MiFi.
6. Buka Aplikasi Sistem dan masukan biodata.
7. Masuk kemenu Broker dan masukan IP : 192.168.1.111 lalu sambungkan.
8. Masuk kemenu pengujian Keseimbangan
9. Klik mulai dan jalankan pengujian.

4.5.4 Hasil Pengujian

Pengujian dilakukan dengan dua skenario, pertama kendaraan hanya melintas dan kedua kendaraan menyenggol patok. Setiap skenario dilakukan lima kali pengujian.

Tabel 4.4 Percobaan Komunikasi Data Patok Dengan *Broker* Saat Kendaraan Melintas

Patok Getar	Percobaan ke					Patok Getar	Percobaan ke				
	1	2	3	4	5		1	2	3	4	5
Patok Getar 1	0	0	0	0	0	Patok Getar 11	0	0	0	0	0
Patok Getar 2	0	0	0	0	0	Patok Getar 12	0	0	0	0	0
Patok Getar 3	0	0	0	0	0	Patok Getar 13	0	0	0	0	0
Patok Getar 4	0	0	0	0	0	Patok Getar 14	0	0	0	0	0
Patok Getar 5	0	0	0	0	0	Patok Getar 15	0	0	0	0	0
Patok Getar 6	0	0	0	0	0	Patok Getar 16	0	0	0	0	0
Patok Getar 7	0	0	0	0	0	Patok Getar 17	0	0	0	0	0
Patok Getar 8	0	0	0	0	0	Patok Getar 18	0	0	0	0	0
Patok Getar 9	0	0	0	0	0						
Patok Getar 10	0	0	0	0	0						

Tabel 4.5 Percobaan Komunikasi Data Patok dengan *Broker* saat Tersenggol

Patok Getar	Percobaan ke					Patok Getar	Percobaan ke				
	1	2	3	4	5		1	2	3	4	5
Patok Getar 1	0	0	1	1	1	Patok Getar 11	1	1	1	1	1
Patok Getar 2	1	1	1	1	1	Patok Getar 12	1	1	1	1	1
Patok Getar 3	1	1	1	1	1	Patok Getar 13	1	1	1	1	1
Patok Getar 4	1	1	1	1	1	Patok Getar 14	1	1	1	1	1
Patok Getar 5	1	1	1	1	1	Patok Getar 15	0	0	1	1	1

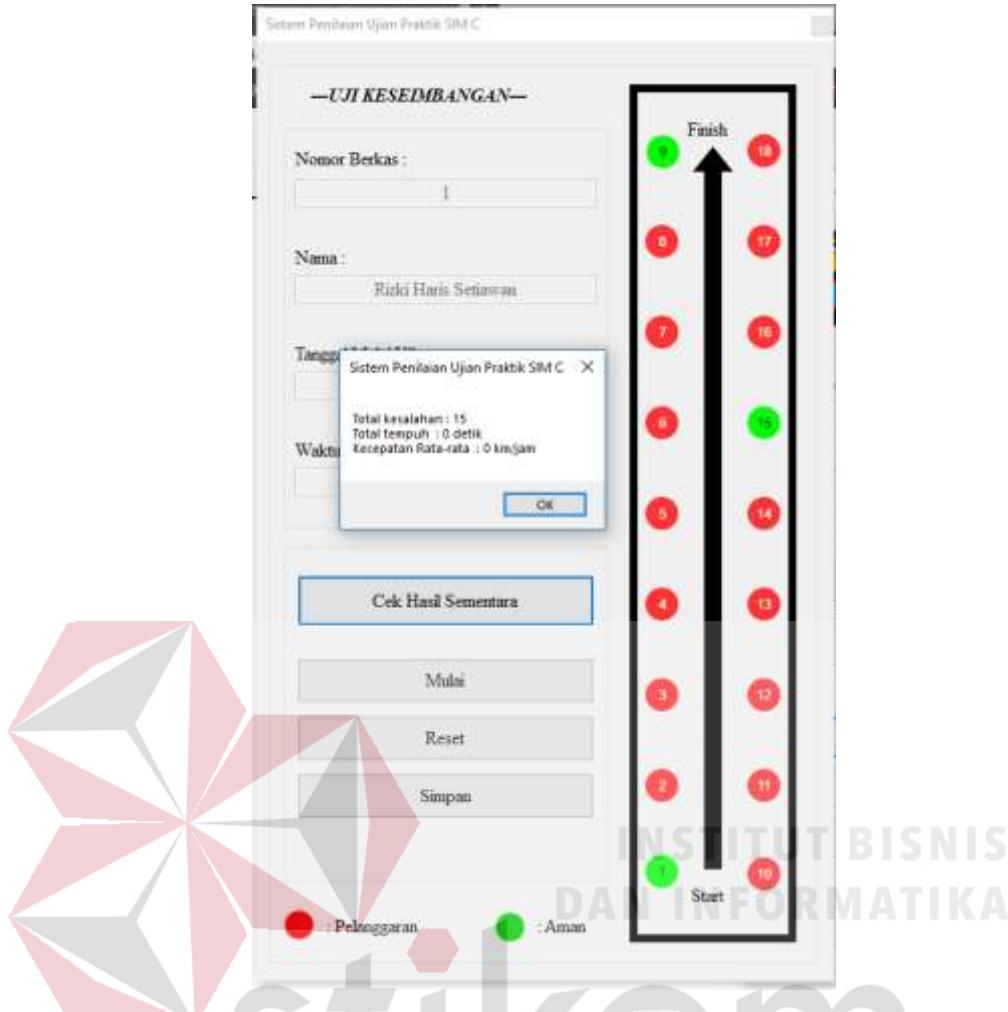
Patok Getar	Percobaan ke					Patok Getar	Percobaan ke				
	1	2	3	4	5		1	2	3	4	5
Patok Getar 6	1	1	1	1	1	Patok Getar 16	1	1	1	1	1
Patok Getar 7	1	1	1	1	1	Patok Getar 17	1	1	1	1	1
Patok Getar 8	1	1	1	1	1	Patok Getar 18	1	1	1	1	1
Patok Getar 9	0	0	1	1	1						
Patok Getar 10	1	1	1	1	1						

Keterangan :

“1” = Berhasil diterima

“0” = Tidak berhasil diterima

Pada tabel 4.5 skenario saat patok tersenggol, Percobaan ke satu dan dua pada patok 1, 9 dan 15 Broker tidak menerima data karena patok sensor getar tidak mendeteksi getaran pada saat tersenggol. Hal ini disebabkan oleh beberapa faktor diantaranya sensitivitas sensor getar yang terlalu kecil sehingga tidak mendeteksi adanya getaran saat tersenggol atau daya pada patok yang tinggal sedikit. Berikut adalah tampilan pada Aplikasi Sistem pada percobaan kedua dengan skenario Patok tersenggol.



Gambar 4. 8 Percobaan kedua Skenario Patok tersenggol

4.6 Pengujian Delay Pengiriman Data *Client* ke Broker

4.6.1 Tujuan

Tujuan dari pengujian ini adalah menemukan jarak yang efektif dalam pengiriman data dari Wemos ke broker. Indikator keberhasilannya adalah menemukan jarak yang efektif agar dapat diterima oleh Broker.

4.6.2 Alat yang Digunakan

Peralatan yang dibutuhkan untuk pengujian alat ini adalah sebagai berikut :

1. Raspberry Pi 3

2. Laptop / Komputer
3. Andromax MiFi
4. Patok Sensor Ultrasonik.

4.6.3 Prosedur Pengujian

Langkah – langkah yang dilakukan untuk melakukan pengujian sistem adalah seperti berikut :

1. Hidupkan Raspberry Pi 3, Sambungkan ke Akses Point Andromax MiFi.
2. Upload program patok sensor ultrasonik ke patok.
3. Hidupkan patok dengan tiga buah baterai 1.5V sebagai daya.
4. Letakan dua patok sensor ultrasonik dengan jarak 15m satu sama lain.
5. Hidupkan Laptop dan sambungkan ke Akses Point Andromax MiFi.
6. Buka konsole cmd dan ketikan “mosquitto_sub -h 192.168.1.111 -t “ultra/1””
7. Buka konsole cmd dan ketikan “mosquitto_sub -h 192.168.1.111 -t “ultra/2””
8. Meletakan sebuah objek didepan kedua patok Ultrasonik.
9. Pengujian dapat dimulai.

4.6.4 Hasil Pengujian

Akses point diletakkan pada jarak-jarak tertentu seperti pada tabel 4.6, Kemudian sebuah objek diletakkan tepat didepan masing-masing Patok Ultrasonik. Dengan demikian konsole cmd pada setiap patok akan berulang-ulang mengirimkan pesan yang menandakan bahwa sebuah objek terdeteksi.

Berikut adalah hasil dari pengujian yang sudah dilakukan :

Tabel 4.6 Hasil Pengujian dengan AP Andromax M2y

No.	Jarak	Sensor	
		Ultra1	Ultra2
1.	2 m	0s	0s
2.	4 m	0s	0s
3.	6 m	0s	0s
4.	8 m	0s	0s
5.	10 m	0s	0s
6.	12 m	1s	1s
7.	14 m	Tidak Terjangkau	Tidak Terjangkau
8.	16 m	Tidak Terjangkau	Tidak Terjangkau
9.	18 m	Tidak Terjangkau	Tidak Terjangkau

Dengan membandingkan data yang masuk dengan data yang akan diterima, maka dapat diketahui berapa detik delay dari pengiriman data Sensor ke Broker.

4.7 Pengujian Daya Tahan Baterai pada Wemos D1 Mini saat Proses

Pengiriman Data

4.7.1 Tujuan

Tujuan dari pengujian ini adalah untuk mengetahui berapa lama waktu hidup dari Wemos D1 Mini dengan Sensor Ultrasonik maupun Sensor Getar.

4.7.2 Alat yang digunakan

Peralatan yang dibutuhkan untuk pengujian ini adalah sebagai berikut :

1. Raspberry Pi 3
2. Laptop / Komputer

3. Aplikasi Sistem Penilaian
4. Andromax MiFi
5. Patok Sensor Getar
6. Patok Sensor Ultrasonik

4.7.3 Prosedur Pengujian

Langkah – langkah yang dilakukan untuk melakukan pengujian sistem adalah seperti berikut :

1. Hidupkan Raspberry Pi 3, Sambungkan ke Akses Point Andromax MiFi.
2. Buka Terminal pada Raspberry Pi 3 dan Ketik :


```
sudo pkill mosquitto
sudo mosquitto
```
3. Buka Aplikasi, Isi Biodata dan Sambungkan ke Broker Raspberry
4. Buka Pengujian Uji Keseimbangan, dan Mulai.
5. Nyalakan Patok Sensor Getar dan Patok Sensor Ultrasonik.
6. Lihat, Amati Komunikasi Data yang Berlangsung dan Gunakan Stopwatch untuk Mengetahui Daya Tahan Baterai.

4.7.4 Hasil Pengujian

Pada Patok digunakan tiga buah baterai dengan tegangan 1.5V dengan kapasitas 1000mAH untuk setiap satu buah baterai. Baterai disusun menggunakan Rangkaian Seri agar dapat mensuplai Wemos dengan tegangan minimal 3.3V-5V, maka dapat dihitung $4.5 \text{ V} \times 1 \text{ AH} = 4.5 \text{ VAH}$.

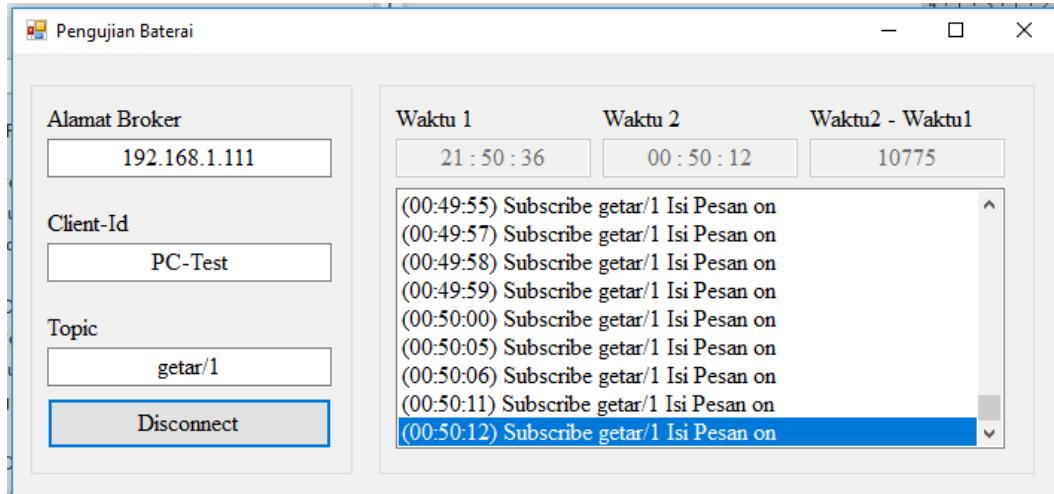
Pengujian secara langsung dilakukan dengan pengiriman data *publisher* setiap satu detik ke aplikasi *subscriber* dengan topik yang sudah ditentukan hingga

Wemos tidak dapat mengirim atau jeda pengiriman semakin lama. Data hasil pengujian dapat dilihat pada tabel 4.7.

Tabel 4.7 Pengujian Baterai pada Wemos

Waktu	Data
21.50	Terkirim
22.05	Terkirim
22.20	Terkirim
22.35	Terkirim
22.50	Terkirim
23.05	Terkirim
23.20	Terkirim
23.35	Terkirim
23.50	Terkirim
24.05	Terkirim
24.20	Terkirim
24.35	Terkirim
24.50	Tidak Terkirim

Pada tabel 4.7, data diambil setiap 15 menit dan menunjukkan bahwa Wemos masih dapat mengirim data dengan jeda kurang lebih sama dengan 2 detik. Pada pukul 24.50 data tidak dapat diterima oleh Wemos karena daya pada baterai sudah berkurang.



Gambar 4.9 Aplikasi untuk Menguji Baterai

Pada gambar 4.9 dapat dilihat bahwa waktu1 merupakan waktu dimana data pertama kali dikirim oleh Wemos dan waktu2 merupakan waktu terakhir Wemos mengirim data ke Aplikasi. Selisih dari waktu2 dan waktu1 adalah 10775 detik, sehingga dapat disimpulkan :

$$\begin{aligned} \text{Waktu2} - \text{waktu1} &= 00.50 - 21.50 \\ &= 10775 \text{ detik} \end{aligned}$$

$$\frac{10775}{3600} = 2.99 \text{ Jam}$$

4.8 Pengujian Lintasan Uji pada Aplikasi Secara Keseluruhan

4.8.1 Tujuan

Tujuan dari pengujian ini adalah untuk menguji apakah 18 buah Patok Sensor Getar dan dua buah Patok Sensor Ultrasonik dapat terhubung dengan Aplikasi Sistem dan mengirimkan data dengan baik.

4.8.2 Alat yang Digunakan

Peralatan yang dibutuhkan untuk pengujian alat ini adalah sebagai berikut :

1. Raspberry Pi 3
2. Laptop / Komputer
3. Andromax MiFi
4. Patok Sensor Getar
5. Patok Sensor Ultrasonik
6. Kendaraan Roda Dua
7. Aplikasi Sistem Penilaian

4.8.3 Prosedur Pengujian

Langkah – langkah yang dilakukan untuk melakukan pengujian sistem adalah seperti berikut :

1. Hidupkan Raspberry Pi 3, Sambungkan ke Akses Point Andromax MiFi.
2. Hidupkan patok dengan tiga buah baterai 1.5V sebagai daya.
3. Hidupkan Laptop dan sambungkan ke Akses Point Andromax MiFi.
4. Buka Aplikasi Sistem. Isi biodata, Alamat Broker : 192.168.1.111 dan sambungkan.
5. Buka menu pengujian Lintasan Keseimbangan.
6. Letakan 18 Patok Sensor Getar dan dua Patok Sensor Ultrasonik sesuai dengan lintasan yang akan diuji.
7. Klik mulai dan jalankan pengujian sesuai dengan Skenario.
8. Ulangi proses nomor 6 – 7 dengan Lintasan Zig-zag dan Putar balik.
9. Cek hasil akhir dengan menu “Lihat Hasil Akhir”.

4.8.4 Hasil Pengujian

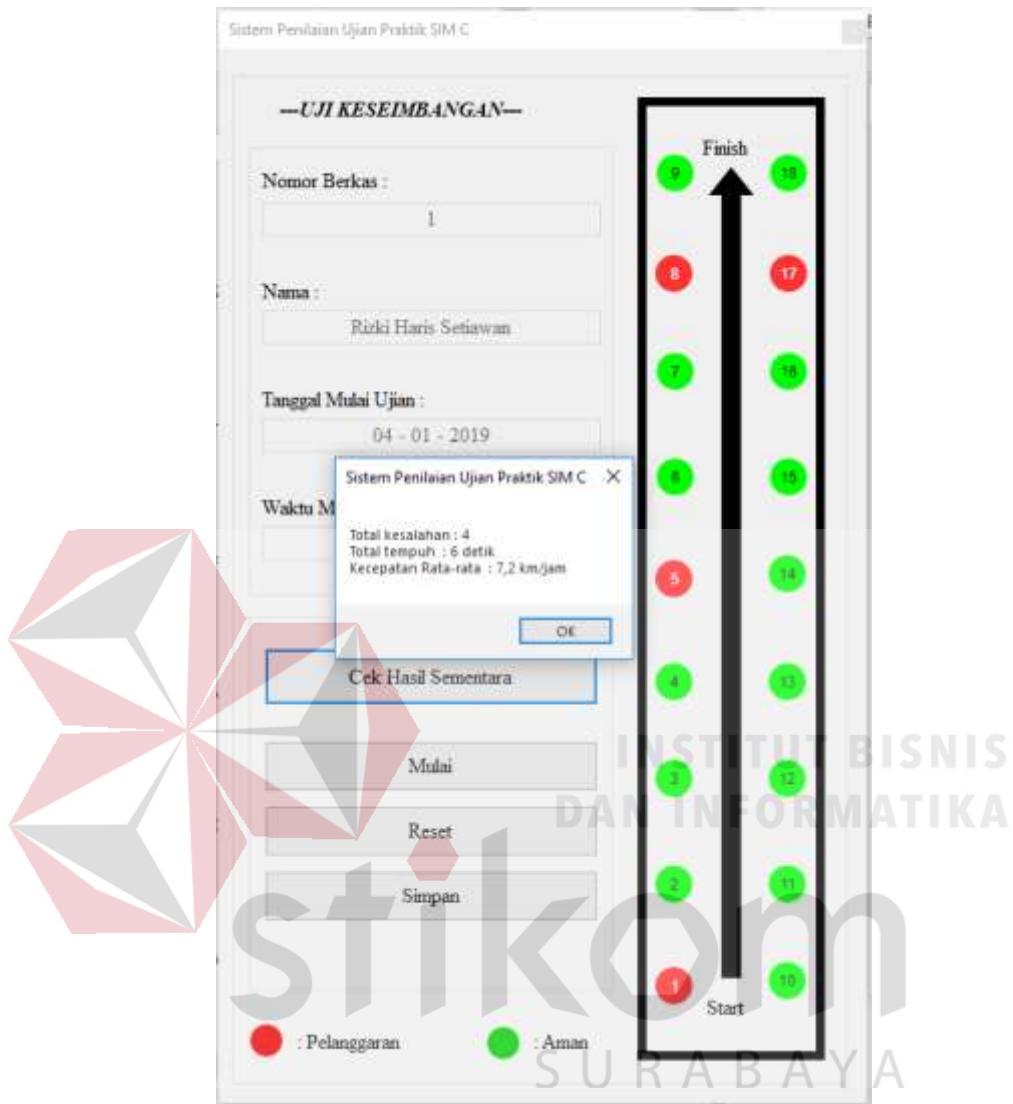
Dari percobaan uji keseimbangan dengan beberapa skenario maka dapat dilihat pada tabel 4.8 :

- Kendaraan jalan lurus tanpa menabrak patok / tiang pembatas.
- Kendaraan jalan lurus tanpa menabrak patok dekat dengan patok sebelah kiri dengan jarak 10cm.
- Kendaraan jalan lurus tanpa menabrak patok dekat dengan patok sebelah kanan dengan jarak 10cm.
- Kendaraan jalan lurus dan menyenggol patok namun tidak terjatuh.
- Kendaraan jalan lurus dan menyenggol patok hingga jatuh.

Hasil dari aplikasi dapat dilihat pada gambar 4.10 , maka dapat disimpulkan tingkat keberhasilan dari lima percobaan adalah 80%.

Tabel 4. 8 Uji keseimbangan

Skenario	Kecepatan	Target	Hasil
a	6 Km/h	Tidak ada pelanggaran	Tidak ada pelanggaran.
b	10.8 Km/h	Tidak ada pelanggaran	Tidak ada pelanggaran.
c	9 Km/h	Tidak ada pelanggaran	Tidak ada pelanggaran.
d	7.2 Km/h	patok 1, 5, 8 dan 17 yang menyala dan dapat mengirim ke aplikasi	patok 1, 5, 8 dan 17 tersenggol dan menyala
e	9 Km/h	Hanya patok 1, 5, 8 dan 17 yang menyala dan dapat mengirim ke aplikasi	patok 1, 5, 8 dan 17 tersenggol dan jatuh namun tidak mengirim karena wemos mati.



Gambar 4.10 Uji Lintasan Keseimbangan dengan skenario d

Dari percobaan uji zig-zag dengan beberapa skenario maka dapat dilihat pada tabel

4.9 :

- Kendaraan jalan tanpa menabrak patok / tiang pembatas.
- Kendaraan jalan tanpa menabrak patok dekat dengan patok sebelah kiri dengan jarak 10cm.

- c. Kendaraan jalan tanpa menabrak patok dekat dengan patok sebelah kanan dengan jarak 10cm.
- d. Kendaraan jalan dan menyenggol patok namun tidak terjatuh.
- e. Kendaraan jalan dan menyenggol patok hingga jatuh.

Hasil dari aplikasi dapat dilihat pada gambar 4.11 , maka dapat disimpulkan tingkat keberhasilan dari 5 percobaan diatas adalah 80%.

Tabel 4.9 Uji zig-zag

Skenario	Kecepatan	Target	Hasil
a	1 Km/h	Tidak ada pelanggaran	Tidak ada pelanggaran
b	2.4 Km/h	Tidak ada pelanggaran	Tidak ada pelanggaran.
c	1.2 Km/h	Tidak ada pelanggaran	Tidak ada pelanggaran.
d	1.41 Km/h	Hanya patok 4, 6 dan 8 yang menyala dan dapat mengirim ke aplikasi	Patok 4, 6 dan 8 tersenggol dan menyala.
e	2 Km/h	Hanya patok 4, 6 dan 8 yang menyala dan dapat mengirim ke aplikasi	Patok 4, 6 dan 8 tersenggol terjatuh dan tidak terkirim karena wemos mati

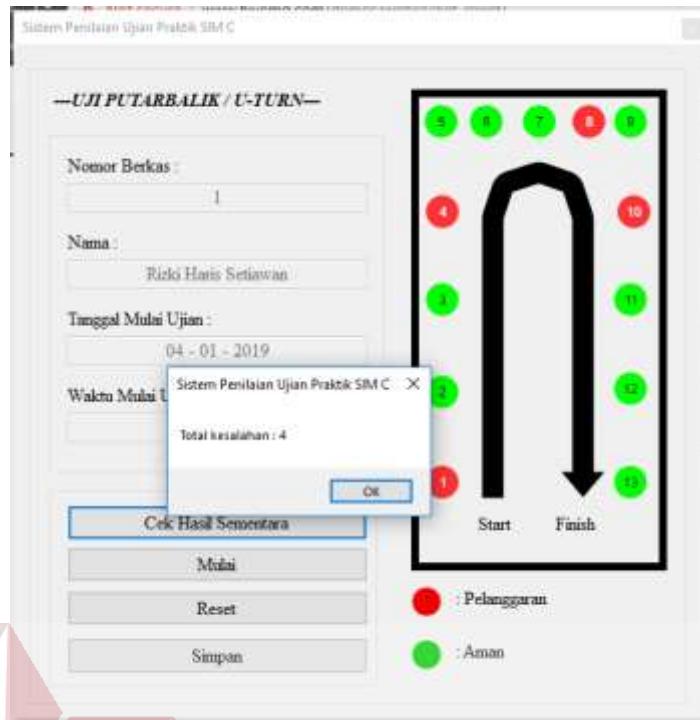


Gambar 4.11 Uji Lintasan Zig-Zag dengan skenario d

Dari percobaan uji putar balik dengan beberapa skenario maka dapat dilihat pada tabel 4.10 :

- Kendaraan jalan tanpa menabrak patok / tiang pembatas.
- Kendaraan jalan tanpa menabrak patok dekat dengan patok sebelah kiri dengan jarak 10cm.
- Kendaraan jalan tanpa menabrak patok dekat dengan patok sebelah kanan dengan jarak 10km.
- Kendaraan jalan dan menyenggol patok namun tidak terjatuh.
- Kendaraan jalan dan menyenggol patok hingga jatuh.

Hasil dari aplikasi dapat dilihat pada gambar 4.12 , maka dapat disimpulkan tingkat keberhasilan dari lima percobaan adalah 80%.



Gambar 4.12 Uji Lintasan U-Turn dengan skenario d

Tabel 4. 10 Patok tersenggol dan jatuh

Skenario	Target	Hasil
a	Tidak ada pelanggaran	Tidak ada pelanggaran
b	Tidak ada pelanggaran	Tidak ada pelanggaran
c	Tidak ada pelanggaran	Tidak ada pelanggaran
d	Hanya patok 1, 4, 8, dan 10 yang menyala dan dapat mengirim ke aplikasi	patok 1, 4, 8 dan 10 tersenggol dan menyala.
e	Hanya patok 1, 4, 8, dan 10 yang menyala dan dapat mengirim ke aplikasi	patok 1, 4, 8 dan 10 tersenggol dan jatuh namun tidak mengirim karena wemos mati.

Hasil akhir dari aplikasi dapat dilihat pada gambar 4.13 :



Gambar 4.13 Hasil Akhir Pengujian pada Aplikasi

Pada gambar 4.13 , dapat diketahui bahwa Patok Sensor Getar dan Patok Sensor Ultrasonik dapat berkomunikasi dengan Aplikasi. Aplikasi dapat menampilkan Total pelanggaran dan Kecepatan rata-rata dari kendaraan peserta pada Lintasan Keseimbangan dan Zig-zag.

Tabel 4. 11 Pengujian Secara Keseluruhan

No	Jenis Pengujian	Jumlah Pengujian	Komunikasi Berlangsung	Komunikasi Berhasil
1	Uji Keseimbangan	5	8	8
2	Uji Zig - Zag	5	6	6
3	Uji Uturn / Putar balik	5	8	8
Total		15	22	22

Pada tabel 4.11 telah dilakukan 15 pengujian dengan lima kali percobaan pada Uji Keseimbangan, lima kali Uji Zig-zag dan lima kali Uji Uturn/Putarbalik. Pada Uji Keseimbangan dan Uji Uturn/Putarbalik terdapat delapan kali komunikasi dan seluruhnya berhasil, namun karena Wemos mati saat terjatuh maka Wemos tidak dapat mengirim. Pada Uji Zig-Zag terdapat delapan kali komunikasi dan seluruhnya berhasil, namun karena Wemos mati saat terjatuh maka Wemos tidak dapat mengirim.

$$Keberhasilan = \frac{22}{22} \times 100\% = 100\%$$

BAB V

PENUTUP

5.1 Kesimpulan

Berdasarkan hasil dari percobaan yang sudah dilakukan, maka dapat disimpulkan bahwa :

1. Aplikasi penilaian ujian praktik SIM C dapat menampilkan pelanggaran pada patok/tiang pembatas yang terjatuh/tersenggol.
2. Aplikasi penilaian ujian praktik SIM C dapat menghitung dan menampilkan kecepatan kendaraan peserta pada uji keseimbangan.
3. Pengiriman setiap data pada patok bergantung pada jarak antara patok dengan akses point, dimana pada jarak $\geq 14m$ patok sudah tidak dapat mengirimkan data dengan normal.
4. Dari Pengujian yang sudah dilakukan terhadap tiga lintasan : Uji Keseimbangan, Uji Zig-Zag dan Uji Uturn/Putarbalik. Didapatkan 22 komunikasi telah berhasil terkirim dari 22 Komunikasi yang berlangsung dan dengan tingkat keberhasilan 100%.
5. Tingkat sensitivitas pada sensor getar SW-420 :
 - a. Jika variabel resistor di setting untuk lebih sensitif, maka ketika kendaraan hanya melewati lintasan akan langsung terdeteksi getaran.
 - b. Jika variabel resistor di setting untuk tidak terlalu sensitif, maka ketika kendaraan hanya melintas tidak akan terdeteksi namun patok harus benar-benar bergoyang agar dapat terdeteksi.

5.2 Saran

Dalam pembuatan Sistem Penilaian ini terdapat beberapa saran untuk perbaikan agar dapat lebih baik ke depannya, sebagai berikut :

1. Setiap patok diberi pelindung agar ketika patok tersenggol dan jatuh tidak merusak komponen yang ada didalamnya.
2. Memanajemen penggunaan daya pada patok-patok sistem penilaian ujian praktik SIM C.



DAFTAR PUSTAKA

- Aroon Octaviannus, H. P. (2017). Rancang Bangun Aplikasi Penilaian Ujian Praktik SIM C Berbasis ArduinoMega2560. *Journal of Control and Network System*, 6(2), 10-20.
- Baldengineer. (2018, Oktober 31). *MQTT Tutorial for Raspberry Pi, Arduino, and ESP8266*. Retrieved from <https://www.baldengineer.com: https://www.baldengineer.com/mqtt-tutorial.html>
- Dejan. (2018, November 14). Retrieved from Ultrasonik Sensor HC-SR04 and Arduino Tutorial: <https://howtomechatronics.com/tutorials/arduino/ultrasonic-sensor-hc-sr04/>
- Hudan Abdur Rochman, R. P. (2017, Juni). Sistem Kendali Berbasis Mikrokontroler Menggunakan Protokol MQTT pada Smarthome. *Jurnal Pengembangan Teknologi Informasi dan Ilmu Komputer*, 1(6), 445-455.
- Jainrk. (2018, Oktober 8). *Programming the ESP8266 WeMos-DIR2 Using Arduino Software/IDE*. Retrieved from [https://www.instructables.com:id/Programming-the-WeMos-Using-Arduino-SoftwareIDE/](https://www.instructables.com: https://www.instructables.com/id/Programming-the-WeMos-Using-Arduino-SoftwareIDE/)
- Oasis. (2018, Oktober 24). *MQTT Version 3.1.1*. Retrieved from <http://docs.oasis-open.org/mqtt/mqtt/v3.1.1/errata01/os/mqtt-v3.1.1-errata01-os-complete.doc>