

SISTEM MONITORING JARAK JAUH BERBASIS RASPBERRY PI PADA SIMULATOR LIFT YANG DIKENDALIKAN PLC

Lido Sanmartya, Handy Wicaksono
Program Studi Teknik Elektro, Universitas Kristen Petra
Jl. Siwalankerto No. 121-131, Surabaya, 60236, Indonesia
E-Mail: sanmrtya@gmail.com, handy@petra.ac.id

Abstrak – Pada penelitian ini memiliki tujuan untuk dapat membuat sebuah sistem monitoring simulator lift jarak jauh yang dapat diakses melalui internet. Lift terdiri dari dua lift satu sangkar yang masing-masing 6 lantai dan dikendalikan oleh dua PLC M221. PLC akan berkomunikasi dengan Raspberry Pi dengan protokol Modbus TCP yang kemudian akan diubah menjadi protokol MQTT dengan memanfaatkan NodeRed. Supaya PLC tersebut dapat terhubung dengan internet, RaspberryPi akan terhubung dengan MQTT broker. Pada penelitian ini MQTT broker di letakkan di server cloud. Berdasarkan hasil pengujian pada penelitian ini, sistem telah berhasil berkomunikasi antara PLC dengan Raspberry Pi menggunakan Modbus TCP dan dashboard untuk monitoring lift dapat ditampilkan melalui web browser menggunakan MQTT. Waktu yang diperlukan untuk komunikasi Modbus dan MQTT relatif cepat dengan rata-rata 0,84 detik.

Kata Kunci – Sistem monitoring, Simulator lift, NodeRed, PLC, Raspberry Pi

I. PENDAHULUAN

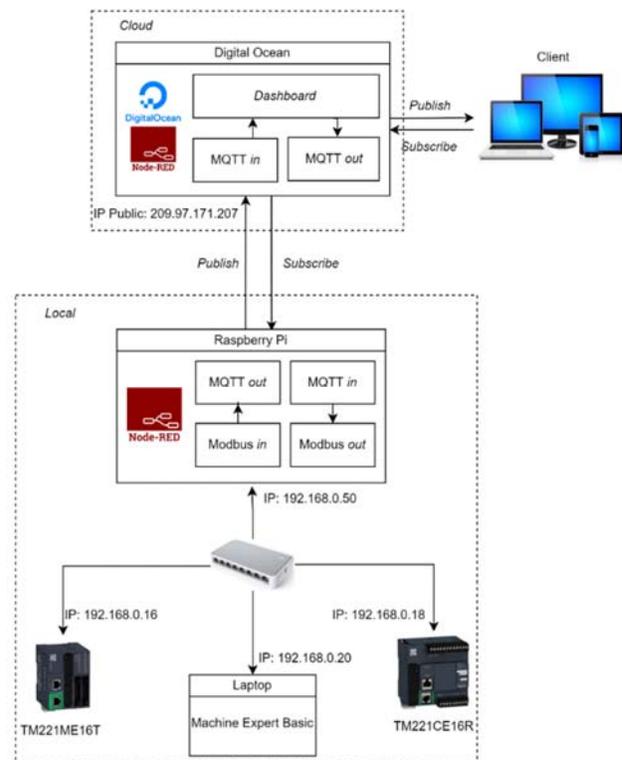
Pada zaman yang serba modern ini semua akan dapat terhubung dengan internet. Pada dunia automasi juga perlahan akan menuju ke sistem yang terhubung dengan IOT yang biasa disebut dengan industry IOT. Pada penelitian ini akan membuat sebuah sistem monitoring jarak jauh untuk memantau status dari sebuah lift. Simulasi akan dilakukan dengan cara menggunakan 2 buah PLC M221 dengan memanfaatkan input diskrit dan juga input analog dari PLC tersebut untuk menggantikan sensor yang sebenarnya. Tujuan dari penelitian ini untuk dapat melakukan sistem monitoring yang dapat di akses menggunakan internet.

Studi literatur yang dilakukan dalam penelitian ini yaitu pertama akan mempelajari komponen-komponen PLC yang akan digunakan dalam penelitian ini. Berikutnya mempelajari penggunaan dari Raspberry Pi, mempelajari cara menggunakan NodeRed secara lokal maupun dari server cloud. Selain itu juga mempelajari cara menampilkan user interface pada internet menggunakan dashboard NodeRed.

Sebelumnya telah dilakukan penelitian terkait komunikasi antara PLC Siemens S7 1200 dengan RaspberryPi memanfaatkan protokol Modbus RTU [1]. Pada penelitian tersebut, Antares digunakan untuk menampilkan status peralatan yang terhubung dengan PLC dan RaspberryPi tersebut. Peneliti lain juga mengembangkan IoT gateway berbasis Raspberry Pi Zero W untuk merubah protokol Modbus RTU ke MQTT [2]. Pada penelitian ini, beberapa PLC akan dihubungkan dengan protokol Modbus TCP akan digunakan untuk mengawasi peralatan dan proses pada sistem lift.

II. PERENCANAAN DAN IMPLEMENTASI SISTEM

Sistem yang akan dibuat merupakan sistem monitoring dari status lift. Pada penelitian ini akan menggunakan 2 lift 1 sangkar yang masing-masing 6 lantai. Pada Gambar 1 merupakan blok diagram sistem dari penelitian ini secara keseluruhan. Proses kerja sistem ini yaitu pertama dari PLC. Pada penelitian ini peneliti menggunakan 2 buah PLC yaitu PLC TM221ME16T dan juga TM221CE16R. peneliti akan memanfaatkan input diskrit, input analog dan juga output diskrit sebagai pengganti dari sensor asli. Setelah dari PLC akan terhubung dengan Raspberry Pi dengan menggunakan ethernet yang dihubungkan terlebih dahulu kedalam sebuah switch.



Gambar 1. Blok Diagram Sistem Keseluruhan

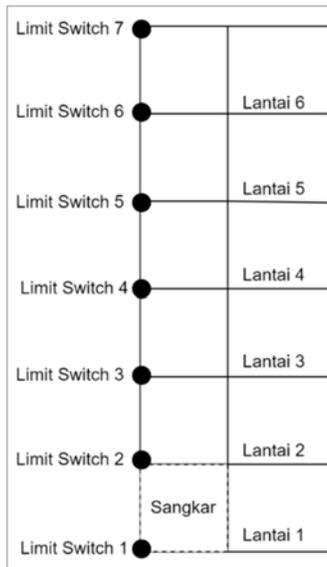
Pada bagian Raspberry Pi akan digunakan untuk menjalankan NodeRed secara lokal. NodeRed tersebut akan berfungsi untuk menerima data sensor dari PLC dengan memanfaatkan Modbus TCP. Selain itu NodeRed lokal juga berfungsi meneruskan

data tersebut ke server cloud dengan menggunakan MQTT out. MQTT out disini berfungsi untuk dapat berkomunikasi dengan server cloud. Server cloud dalam penelitian ini menggunakan platform Digital Ocean.

Pada Digital Ocean juga akan menjalankan NodeRed secara cloud. Fungsi dari NodeRed di cloud ini yaitu akan menerima data yang dikirimkan dari Raspberry Pi dengan menggunakan MQTT in. setelah data diterima akan dapat ditampilkan di internet dengan menggunakan IP public yang diberikan oleh Digital Ocean untuk dapat melihat dashboard dari status lift tersebut.

Pada Gambar 2 merupakan letak posisi sensor limit switch yang digunakan pada penelitian ini. Sistem sensor ini akan menyalakan 2 limit switch pada saat berada di satu lantai. Jadi apabila sangkar berada di lantai 1, maka limit switch 1 dan juga limit switch 2 akan menyala. Lalu apabila sangkar berada pada lantai 2 limit switch 2 dan juga limit switch 3 akan menyala. Begitu dengan seterusnya apabila sangkar berada di lantai 6 maka limit switch 6 dan juga limit switch 7 akan menyala.

Pada penelitian ini akan membuat sistem monitoring status lift yang dimana akan menampilkan status lift, posisi lantai lift, dan juga berat sangkar. Cara kerja sistem limit switch ini akan berlaku pada lift A maupun lift B.

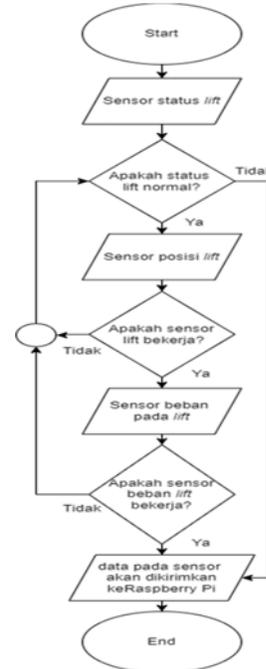


Gambar 2. Letak posisi sensor limit switch

Pada Gambar 3 merupakan program dari Machine Expert Basic yang digunakan untuk penelitian ini. Pertama program akan membaca status lift yaitu dengan mengambil data sensor kebakaran dan sensor gempa. Apabila kedua sensor tersebut tidak menyala maka sistem akan memberitahukan bahwa lift tersebut dalam keadaan normal. Lalu akan lanjut kedalam proses posisi lift. Untuk proses ini sensor limit switch akan bekerja untuk menampilkan posisi lantai dari suatu lift. Berikutnya sistem akan mendeteksi sensor berat dari sangkar lift tersebut.

Setelah 3 proses tersebut berjalan dan setiap sensor telah mengambil data, maka data tersebut akan dilanjutkan atau dikirmkan ke Raspberry Pi. Peneliti memanfaatkan Raspberry Pi untuk menjalankan NodeRed secara lokal. Pada NodeRed peneliti akan mengambil data dari PLC tersebut dengan memanfaatkan komunikasi Modbus. Komunikasi Modbus

yang digunakan dalam penelitian ini yaitu Komunikasi Modbus TCP.



Gambar 3. Diagram Alur Program Machine Expert Basic

Pada Gambar 4 merupakan training kit yang digunakan untuk menyimulasikan lift A. Untuk training kit tersebut memiliki 8 input diskrit, 2 input analog, 8 output diskrit, dan juga 1 buzzer. Peneliti menggunakan 8 input diskrit tersebut untuk menyimulasikan tombol start, tombol limit switch 1 sampai tombol limit switch 7. Berikutnya untuk output diskrit digunakan untuk menampilkan posisi lantai 1 sampai dengan lantai 6. Lalu led berikutnya digunakan untuk menyimulasikan overload. Berikutnya untuk 1 input analog digunakan untuk menyimulasikan berat sangkar. Lalu untuk buzzer pada training kit digunakan menyimulasikan pada saat berat sangkar mencapai overload.



Gambar 4. Training Kit TM221ME16T

Pada Gambar 5 merupakan training kit yang digunakan untuk menyimulasikan lift B. Pada training kit tersebut terdapat 9 input diskrit, 4 input analog, 8 output diskrit, 1 buzzer. Peneliti menggunakan training kit ini untuk menyimulasikan lift B. Untuk 9 input diskrit digunakan untuk menyimulasikan tombol

start, tombol limit switch 1 sampai limit switch 7, tombol stop. Berikutnya untuk output diskrit digunakan untuk menampilkan posisi lantai 1 hingga lantai 6, lalu untuk menyimulasikan led overload. Selanjutnya untuk 1 input analog digunakan untuk menyimulasikan berat sangkar dari lift B. Lalu untuk buzzer digunakan untuk menyimulasikan alarm apabila lift sedang overload.



Gambar 5. Training Kit TM221CE16R

Pada Tabel 1 merupakan tabel input dan juga output pada PLC yang digunakan untuk menyimulasikan lift A dan lift B. untuk lift A yang disimulasikan dengan training kit TM221ME16T memiliki 8 input diskrit. Lalu untuk address setiap input diskrit yaitu %i0.0 digunakan untuk tombol start. Selanjutnya %i0.1 digunakan untuk menyimulasikan limit switch 1. Berikutnya address %i0.2 digunakan untuk limit switch 2. Lalu untuk address %i0.3 digunakan untuk menyimulasikan limit switch 3 dan limit switch 4 dengan address %i0.4, lalu untuk limit switch 5, limit switch 6, limit switch 7 dengan address %i0.5, %i0.6, %i0.7

Lanjut input analog dengan address %IW0.0 digunakan untuk menyimulasikan berat sangkar. Untuk output ada 8 yang digunakan untuk menyimulasikan posisi lantai 1 dengan address %Q0.0. Selanjutnya digunakan untuk lantai 2 dengan address %Q0.1. Berikutnya untuk lantai 3 dengan address %Q0.2. Lalu untuk lantai 4 oleh peneliti diberi address %Q0.3. Lanjut untuk lantai 5 dengan address %Q0.4. Selanjutnya untuk lantai 6 diberi address %Q0.5. Lalu untuk %Q0.6 digunakan untuk menyimulasikan apabila lift dalam keadaan overload maka led tersebut akan menyala. Pada address %Q0.6 diparalel juga dengan buzzer, jadi apabila lift dalam kondisi overload maka akan ada led dan buzzer yang menyala.

Untuk menyimulasikan 2 lift 1 sangkar dengan masing-masing 6 lantai peneliti kekurangan tombol fisik pada PLC. Maka dari itu peneliti memanfaatkan tombol yang terdapat pada NodeRed. Untuk input diskrit seperti tombol stop, tombol lantai 1 hingga tombol lantai 6, sensor kebakaran, dan juga sensor gempa pada penelitian ini disimulasikan dengan menggunakan tombol yang terdapat pada dashboard NodeRed di server cloud. Lalu berikutnya sisa dari output diskrit disimulasikan juga dengan memanfaatkan led yang terdapat pada dashboard NodeRed yaitu seperti limit switch 1 hingga limit switch 7, motor up dan motor down, sensor kebakaran dan sensor gempa disimulasikan dengan menggunakan led pada dashboard NodeRed.

Tabel 1. Tabel Input dan Output

Input	PLC Address		Output	Address	
	I	Memory		Q	Memory
Start	%i0.0	%M200	Lantai 1	%Q0.0	%M90
Limit Switch 1	%i0.1	%M40	Lantai 2	%Q0.1	%M91
Limit Switch 2	%i0.2	%M41	Lantai 3	%Q0.2	%M92
Limit Switch 3	%i0.3	%M42	Lantai 4	%Q0.3	%M93
Limit Switch 4	%i0.4	%M43	Lantai 5	%Q0.4	%M94
Limit Switch 5	%i0.5	%M44	Lantai 6	%Q0.5	%M95
Limit Switch 6	%i0.6	%M45	Overload	%Q0.6	%M47
Limit Switch 7	%i0.7	%M46	Buzzer	%Q0.6	%M47
Stop		%M101	Limit Switch 1		%M40
Tombol Lantai 1		%M80	Limit Switch 2		%M41
Tombol Lantai 2		%M81	Limit Switch 3		%M42
Tombol Lantai 3		%M82	Limit Switch 4		%M43
Tombol Lantai 4		%M83	Limit Switch 5		%M44
Tombol Lantai 5		%M84	Limit Switch 6		%M45
Tombol Lantai 6		%M85	Limit Switch 7		%M46
Berat Sangkar	%IW0.0	%MW100	Motor Up		%M70
Sensor Kebakaran		%M66	Motor Down		%M71
Sensor Gempa		%M67	Kebakaran		%M2
			Gempa		%M20

Pada Tabel 2 merupakan jumlah penggunaan Modbus read pada NodeRed lokal yang dijalankan di Raspberry Pi. Untuk setiap lift menggunakan 20 Modbus read. Pertama digunakan untuk membaca nilai dari PLC untuk tombol on dan off. Berikutnya digunakan untuk mengambil data dari lantai 1 sampai lantai 6 agar dapat menampilkan posisi dari sebuah lift tersebut berada pada lantai berapa. Lalu berikutnya digunakan untuk mengambil data motor dari lift tersebut apakah sedang naik atau turun.

Selanjutnya digunakan untuk memantau dari limit switch setiap lift dari limit switch 1 sampai limit switch 7. Lalu juga digunakan untuk mengambil nilai dari sensor berat sangkar yang akan ditampilkan pada dashboard NodeRed menggunakan gauge. Terakhir digunakan untuk mengambil nilai dari pintu sangkar apakah pintu tersebut sedang tertutup atau terbuka.

Pada Tabel 3 merupakan jumlah dari penggunaan Modbus write pada NodeRed lokal yang dijalankan pada Raspberry Pi. ada 10 Modbus write yang digunakan untuk masing-masing lift. Pertama digunakan untuk tombol on dan off yang akan mengirim data ke PLC. Berikutnya digunakan untuk tombol lantai lift mulai dari tombol lantai 1 sampai dengan tombol lantai 6. Tombol tersebut digunakan untuk memerintahkan lift berjalan ke lantai yang ditekan dari layar dashboard monitoring. Berikutnya digunakan untuk menyimulasikan sensor gempa dan juga sensor kebakaran.

Pada Tabel 4 merupakan jumlah penggunaan MQTT in secara lokal dan juga MQTT out yang berada pada server cloud. Jumlah penggunaan ada 10 MQTT in pada NodeRed lokal. Pertama digunakan untuk mendapatkan nilai dari on dan off dengan topic onA dan offA untuk lift A. Selanjutnya digunakan untuk lantai 1 sampai lantai 6 dengan diberi topic lantai1A, lantai2A, lantai3A, lantai4A, lantai5A, lantai6A. Lalu untuk sensor kebakaran dan juga sensor gempa peneliti memberikan

topic kebakaranA dan juga gempaA. Selanjutnya penggunaan MQTT in di lokal untuk lift B sama jumlahnya yaitu 10. Lalu untuk topic yang digunakan untuk lift B peneliti juga memberikan topic yang sama dengan lift A. Perbedaannya hanyalah diberikan huruf B pada akhir topic untuk membedakan dari lift A.

Tabel 2. Jumlah Modbus Read

Modbus Read						
No	Lift A			Lift B		
	Input dan Output	PLC Address		Input dan Output	PLC Address	
		Memory	I/O		Memory	I/O
1	On	%M60	%I0.0	On	%M60	%I0.0
2	Off	%M40		Off	%M101	%I0.8
3	Lantai 1	%M90		Lantai 1	%M90	
4	Lantai 2	%M91		Lantai 2	%M91	
5	Lantai 3	%M92		Lantai 3	%M92	
6	Lantai 4	%M93		Lantai 4	%M93	
7	Lantai 5	%M94		Lantai 5	%M94	
8	Lantai 6	%M95		Lantai 6	%M95	
9	Motor Up	%M70		Motor Up	%M70	
10	Motor Down	%M71		Motor Down	%M71	
11	Limit Switch 1	%M40	%I0.1	Limit Switch 1	%M40	%I0.1
12	Limit Switch 2	%M41	%I0.2	Limit Switch 2	%M41	%I0.2
13	Limit Switch 3	%M42	%I0.3	Limit Switch 3	%M42	%I0.3
14	Limit Switch 4	%M43	%I0.4	Limit Switch 4	%M43	%I0.4
15	Limit Switch 5	%M44	%I0.5	Limit Switch 5	%M44	%I0.5
16	Limit Switch 6	%M45	%I0.6	Limit Switch 6	%M45	%I0.6
17	Limit Switch 7	%M46	%I0.7	Limit Switch 7	%M46	%I0.7
18	Berat	%M100	%IW0.0	Berat	%M100	%IW0.0
19	Overload	%M47	%Q0.7	Overload	%M47	%Q0.7
20	Pintu	%M300		Pintu	%M300	

Tabel 3. Jumlah Modbus Write

Modbus Write					
No	Lift A			Lift B	
	Input dan Output	PLC Address		Input dan Output	PLC Address
		Memory			Memory
1	On	%M200		On	%M200
2	Off	%M101		Off	%M101
3	Tombol Lantai 1	%M80		Tombol Lantai 1	%M80
4	Tombol Lantai 2	%M81		Tombol Lantai 2	%M81
5	Tombol Lantai 3	%M82		Tombol Lantai 3	%M82
6	Tombol Lantai 4	%M83		Tombol Lantai 4	%M83
7	Tombol Lantai 5	%M84		Tombol Lantai 5	%M84
8	Tombol Lantai 6	%M85		Tombol Lantai 6	%M85
9	Sensor Kebakaran	%M86		Sensor Kebakaran	%M86
10	Sensor Gempa	%M87		Sensor Gempa	%M87

Tabel 4. Jumlah MQTT In (Lokal) dan MQTT Out (Server Cloud)

MQTT In (Lokal) dan MQTT Out (Server Cloud)				
No	Lift A		Lift B	
	Input dan Output	Topic	Input dan Output	Topic
1	On	onA	On	onB
2	Off	offA	Off	offB
3	Lantai 1	lantai1A	Lantai 1	lantai1B
4	Lantai 2	lantai2A	Lantai 2	lantai2B
5	Lantai 3	lantai3A	Lantai 3	lantai3B
6	Lantai 4	lantai4A	Lantai 4	lantai4B
7	Lantai 5	lantai5A	Lantai 5	lantai5B
8	Lantai 6	lantai6A	Lantai 6	lantai6B
9	Sensor Kebakaran	kebakaranA	Sensor Kebakaran	kebakaranB
10	Sensor Gempa	gempaA	Sensor Gempa	gempaB

Tabel 5. Jumlah MQTT Out (Lokal) dan MQTT In (Server Cloud)

MQTT Out (Local)				
No	Lift A		Lift B	
	Input dan Output	Topic	Input dan Output	Topic
1	Power On	poweronA	Power On	poweronB
2	Power Off	poweroffA	Power Off	poweroffB
3	Posisi lantai	lantaiA	Posisi lantai	lantaiB
4	Limit Switch 1	ls1A	Limit Switch 1	ls1B
5	Limit Switch 2	ls2A	Limit Switch 2	ls2B
6	Limit Switch 3	ls3A	Limit Switch 3	ls3B
7	Limit Switch 4	ls4A	Limit Switch 4	ls4B
8	Limit Switch 5	ls5A	Limit Switch 5	ls5B
9	Limit Switch 6	ls6A	Limit Switch 6	ls6B
10	Limit Switch 7	ls7A	Limit Switch 7	ls7B
11	Berat	beratA	Berat	beratB
12	Full	fullA	Full	fullB
13	Motor Up	upA	Motor Up	upB
14	Motor Down	downA	Motor Down	downB
15	Pintu	pintuA	Pintu	pintuB

Pada Tabel 5 ditampilkan jumlah penggunaan MQTT Out (Lokal) dan MQTT In (Server Cloud). Total penggunaan ada 15 MQTT in pada server cloud. Pertama digunakan untuk mengirimkan pada PLC yaitu tombol on dan off yang terdapat pada dashboard dengan topic poweronA dan juga poweroffA. Lalu selanjutnya digunakan untuk membaca posisi lantai dengan topic lantaiA. Berikutnya digunakan untuk mengirimkan nilai dari limit switch 1 sampai dengan limit switch 7 dengan topic ls1A, ls2A, ls3A, ls4A, ls5A, ls6A, ls7A. Selanjutnya digunakan untuk mengirimkan nilai dari sensor berat sangkar dengan topic beratA. Lalu digunakan juga untuk motor up dan juga motor down dengan topic upA dan downA.

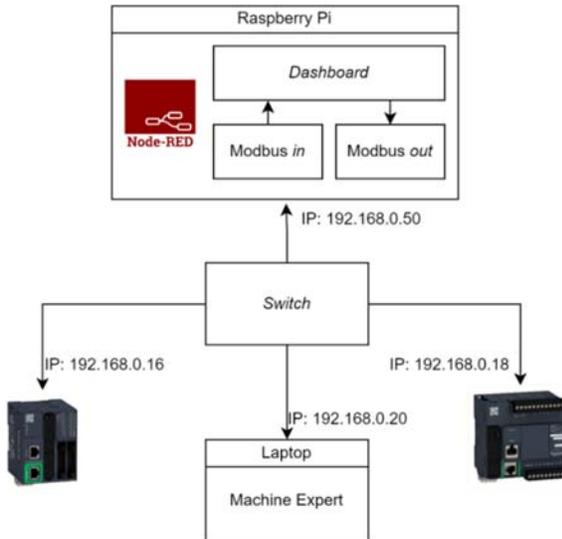
Terakhir digunakan untuk full dan juga pintu dengan topic fullA dan pintuA. Lalu untuk lift B total penggunaan MQTT out di lokal juga berjumlah 15 lalu untuk topic setiap akhiran diberikan huruf B untuk membedakan bahwa topic tersebut untuk lift B.

III. PENGUJIAN SISTEM DAN ANALISA

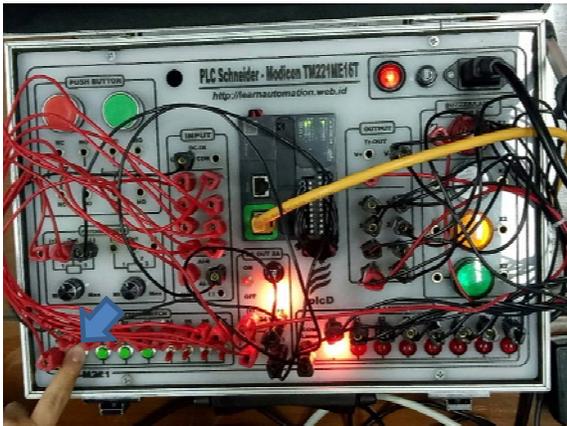
Untuk pengujian sistem dan analisa akan dilakukan pengujian komunikasi antara PLC dan juga Raspberry Pi dengan memanfaatkan Modbus TCP. Lalu berikutnya akan menguji sistem Modbus/MQTT dengan memanfaatkan MQTT broker pada server cloud (Digital Ocean).

A. Pengujian Komunikasi Modbus

Pada Gambar 1 Gambar 6 merupakan blok diagram sistem untuk pengujian komunikasi Modbus. Pada pengujian ini hanya menggunakan 2 buah PLC dan juga 1 buah Raspberry Pi. Tujuan dari pengujian ini supaya dapat mengetahui bahwa sebuah PLC yang digunakan pada penelitian ini sudah dapat mengirimkan data ke NodeRed lokal di Raspberry Pi.



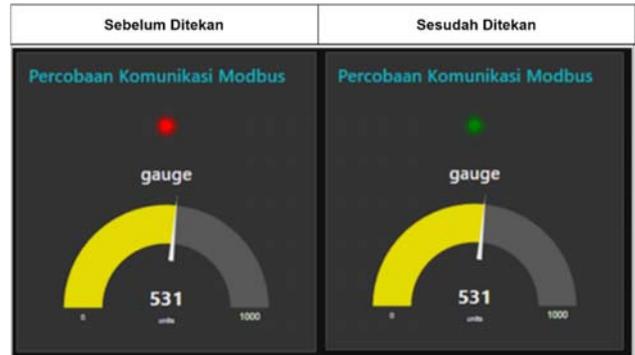
Gambar 6. Blok Diagram Sistem Pengujian Modbus



Gambar 7. Pengujian Tombol Pada PLC

Pada Gambar 7 peneliti mencoba dengan menekan tombol fisik pada PLC. Lalu akan dapat diterima pada dashboard NodeRed. Pada Gambar 8 merupakan hasil dari dashboard NodeRed yang dijalankan pada Raspberry Pi. Pada bagian sebelah kiri merupakan hasil dashboard NodeRed pada saat peneliti

sebelum menekan tombol pada PLC led akan berwarna merah apabila sebelum ditekan. Lalu pada bagian sebelah kanan merupakan hasil dashboard pada saat peneliti menekan tombol fisik pada PLC. Tampak pada dashboard NodeRed berubah menjadi warna hijau.

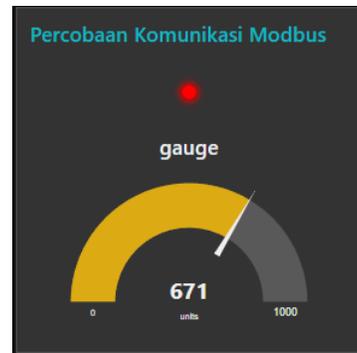


Gambar 8. Tampilan Dashboard Sebelum dan Sesudah Ditekan

Pada Gambar 9 peneliti mencoba komunikasi Modbus antara PLC dengan Raspberry Pi dengan data analog. Peneliti mencoba dengan memutar potensiometer fisik pada PLC untuk mengubah nilai dari data analog tersebut. Pada Gambar 10 terlihat tampilan *dashboard gauge* berubah dengan berjalan naik.



Gambar 9. Pengujian Potensiometer Pada PLC



Gambar 10. Tampilan Dashboard Untuk Data Analog

Pada Tabel 6 merupakan hasil dari pengujian sistem komunikasi Modbus. Pada pengujian ini peneliti menguji sebanyak 10 kali. Percobaan tersebut didapati nilai diatas 1 detik sebanyak 2 kali percobaan. Sebanyak 8 kali percobaan didapati waktu dibawah 1 detik. Dalam pengujian tersebut

didapatkan data dengan rata-rata 0,74 detik. Jadi untuk komunikasi Modbus tersebut dapat dikatakan sangat cepat.

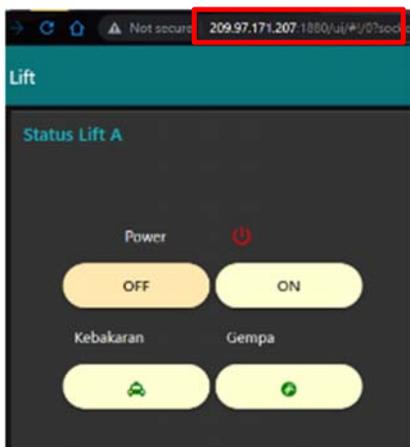
Tabel 6. Pengujian Sistem Modbus Read dan Write

Tombol	Percobaan	waktu
	1	00.00.62
	2	00.00.64
	3	00.00.45
	4	00.00.98
	5	00.01.06
	6	00.01.06
	7	00.00.78
	8	00.00.66
	9	00.00.72
	10	00.00.46

B. Pengujian Komunikasi Modbus/MQTT Pada Sistem Lift

Pada pengujian kali ini peneliti bertujuan untuk menguji komunikasi antara Modbus/MQTT. Untuk pengujian komunikasi Modbus/MQTT menggunakan sistem seperti pada Gambar 1. Dengan menggunakan 2 buah PLC untuk menyimulasikan 2 buah lift 1 sangkar yang masing-masing 6 lantai. Lalu menggunakan Raspberry Pi untuk menerima data dari PLC secara lokal dengan memanfaatkan komunikasi Modbus pada NodeRed. Setelah itu data tersebut akan dikirimkan ke server cloud dengan menjalankan NodeRed pada Digital Ocean dengan memanfaatkan MQTT pada NodeRed di lokal maupun di server cloud. Setelah itu akan memanfaatkan dashboard pada server cloud untuk dapat melakukan monitoring status lift.

Selain itu juga peneliti ingin mencoba untuk mengakses dashboard dengan menggunakan internet yang berbeda. Pada Gambar 11 merupakan hasil dashboard pada saat mengakses dengan menggunakan IP public yang diberikan oleh server cloud (Digital Ocean). Dengan cara mengakses 209.97.171.207:1880/ui.



Gambar 11. Tampilan Dashboard Monitoring Lift

Pada Gambar 12 merupakan hasil dashboard monitoring pada saat peneliti menekan tombol on. Pada saat tombol on ditekan maka power akan berubah warna menjadi hijau yang menandakan bahwa power lift A tersebut sedang menyala. Lalu apabila power lift A tersebut tidak menyala maka warna akan

berubah menjadi merah yang menandakan bahwa lift A tersebut sedang mati seperti pada Gambar 11.



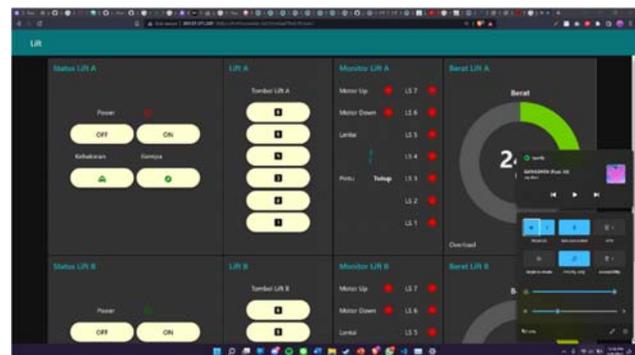
Gambar 12. Hasil Dashboard Pada Saat Tombol On Ditekan

Pada Tabel 7 merupakan hasil pengujian sistem komunikasi Modbus/MQTT. Pada percobaan sebanyak 10 kali peneliti mendapatkan data bahwa sebanyak 3 kali percobaan diatas 1 detik. Lalu untuk percobaan sebanyak 7 kali didapatkan data dibawah 1 detik. Dengan rata-rata waktu 0,84 detik.

Tabel 7. Pengujian Sistem Modbus/MQTT

Tombol On/Off	Percobaan	Waktu
	1	00.00.63
	2	00.00.95
	3	00.00.86
	4	00.01.09
	5	00.00.57
	6	00.00.83
	7	00.00.76
	8	00.01.00
	9	00.00.72
	10	00.01.00

C. Penyajian Hasil Dashboard Monitoring Lift



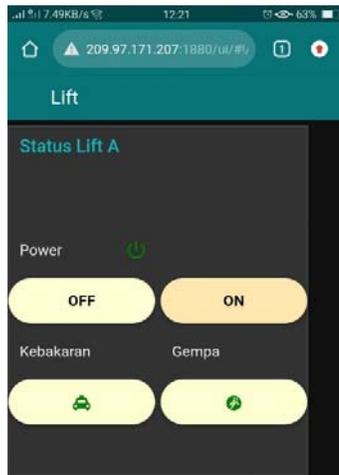
Gambar 13. Hasil Dashboard pada dekstop 1

Pada subbab ini peneliti akan menyajikan hasil dashboard monitoring lift pada saat diakses menggunakan beberapa

dekstop dan mobile device dengan menggunakan jaringan yang berbeda. [2]



Gambar 14. Hasil Dashboard pada dekstop 2



Gambar 15. Hasil Dashboard pada Mobile Device

IV. KESIMPULAN

Dari penelitian yang sudah dilakukan peneliti mendapatkan kesimpulan yang didapat sebagai berikut:

1. Sistem dapat berkomunikasi antara PLC dan NodeRed di Raspberry Pi dengan memanfaatkan Modbus TCP dengan rata-rata waktu yang cukup cepat yaitu 0,74 detik dalam 10 kali percobaan.
2. Sistem dapat berkomunikasi antara Raspberry Pi dan cloud server (Digital Ocean) dengan memanfaatkan MQTT di NodeRed dengan rata-rata waktu yang relatif cepat yaitu 0,84 detik dalam 10 kali percobaan.
3. Monitoring berhasil dilakukan melalui dashboard NodeRed dan digital output pada PLC dapat menyala. Selain itu juga dashboard pada tugas akhir ini dapat diakses melalui web browser dengan percobaan melalui 2 dekstop dengan jaringan yang berbeda-beda dan juga 1 mobile device dengan jaringan yang berbeda juga.

DAFTAR PUSTAKA

- [1] Ali, Mohammed Hasan, "Design and implementation of an electrical lift controlled using PLC", *International Journal of Electrical and Computer Engineering*, vol. 8, no. 4, 2018.

B. Yuwono, S. P. Nugroho, and H. Heriyanto, "Pengembangan Model Public Monitoring System Menggunakan Raspberry Pi," *Telematika*, vol. 12, no. 2, Jul. 2015.

- [3] Silva, Claudio Rodrigues M., and Felipe Augusto Cortez M. Silva, "An IoT gateway for Modbus and MQTT integration", *2019 SBMO/IEEE MTT-S International Microwave and Optoelectronics Conference (IMOC)*, IEEE, 2019.