

MINIATUR 2 LIFT 5 LANTAI MENGGUNAKAN KONTROLLER 2 PLC OMRON CPM1A DENGAN *ONE TO ONE PC LINK CONNECTION* MENGGUNAKAN KABEL RS232

Sucron Handoko^{*)}, Aris Triwiyatno, and Sumardi

Jurusan Teknik Elektro, Universitas Diponegoro Semarang
Jl. Prof. Sudharto, SH, Kampus UNDIP Tembalang, Semarang 50275, Indonesia

^{*)}E-mail: sucronhandoko@gmail.com

Abstrak

Pada era sekarang, teknologi semakin berkembang, tidak terkecuali pada bidang transportasi. Lift merupakan salah satu transportasi yang memiliki arah pergerakan secara tegak. Salah satu permasalahan yang muncul pada lift adalah pembagian tugas ketika terdapat 2 lift yang bersebelahan dan saling bekerjasama dalam mengantarkan penumpang. Pada Tugas Akhir ini dilakukan perancangan miniatur 2 lift 5 lantai menggunakan kontroller 2 PLC Omron CPM1A yang dikomunikasikan dengan One to One PC Link Connection menggunakan kabel RS 232. Salah satu PLC digunakan sebagai PLC Master dan PLC yang lain sebagai PLC Slave. Masing-masing lift dikontrol oleh satu PLC sedangkan pembagian tugas diatur oleh PLC Master. Masukan pada PLC adalah sensor mekanis yang berupa limit switch dan push button sebagai sensor-sensor pada lift. Sedangkan keluaran pada PLC adalah relay dan LED yang berfungsi sebagai pengontrol aktuator dan sebagai indikator pada lift. Dari pengujian yang dilakukan diperoleh bahwa plant miniatur 2 lift 5 lantai ini dapat dikontrol menggunakan 2 PLC Omron CPM1A yang dikomunikasikan dengan one to one PC link connection menggunakan kabel RS 232. Pembagian tugas pada miniatur 2 lift 5 lantai yang bersebelahan dengan push button naik/turun di setiap lantai dapat dilakukan berdasarkan letak kedua lift, kondisi kedua lift sedang naik atau turun, serta jumlah tugas pada setiap lift.

Kata kunci: Lift, PLC Omron CPM1A, One to One PC Link Connection

Abstract

Recently, technology has been advanced, including the transportation field. Elevator is one of the vertical transportation. The synchronous of two elevator that placed side by side is one problem of elevator system so it can work together to carry the passengers. This final project purpose is to design a 2 elevators 5 floors miniature using 2 PLC Omron CPM1A controller which communicated with One to One PC Link Connection using RS 232 cable. One PLC used as a Master PLC and the other PLC as Slave PLC. Each elevator is controlled by the one PLC while the division of tasks set by the Master PLC. Input of the PLC is in the form of a mechanical sensor limit switch and push button as the sensors on the elevator. While the output of PLC is relays and LEDs that used as actuator controllers and as an indicator of the elevator. From the tests, 2 elevators 5 floors miniature plant can be controlled using 2 PLC Omron CPM1A communicated with one-to-one PC link connection using RS 232 cable. The 2 elevators 5 floors miniature plant that placed side by side with up and down push button in each floor can be synchronized by the position of 2 elevators, the condition of 2 elevators (up or down), and the task of each elevator.

Keywords: Elevator, PLC Omron CPM1A, One to One PC Link Connection

1. Pendahuluan

Pada era sekarang, teknologi semakin berkembang, tidak terkecuali pada bidang transportasi. Dalam beberapa penelitian, masalah transportasi menarik banyak perhatian dalam dunia fisika^{[6],[7]}. Lift merupakan salah satu contoh transportasi *vertical*. Salah satu permasalahan yang muncul pada lift adalah pembagian tugas ketika terdapat

2 lift yang bersebelahan dan saling bekerjasama dalam mengantarkan penumpang.

Oleh karena itu, dalam penelitian ini dirancang suatu minatur dua lift lima lantai menggunakan kontroller dua PLC (*Programmable Logic Controller*) CPM1A yang dikomunikasikan dengan *One to One PC Link Connection* menggunakan kabel RS 232.

2 Metode

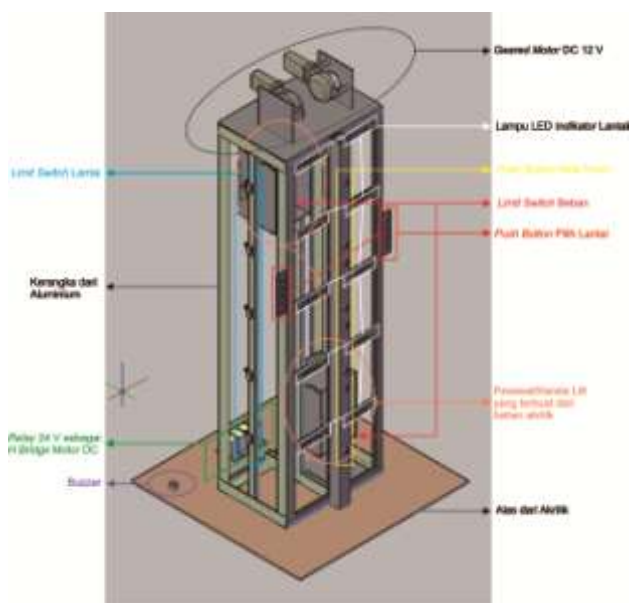
2.1 Metode Pengoperasian Lift

➤ Metode *Duplex-Collective*

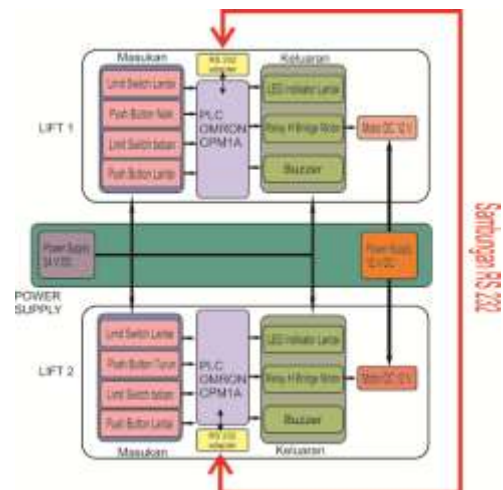
Prinsip bekerja dari *duplex-collective* sebenarnya sama dengan prinsip kerja *selective-collective* yang menggunakan 2 buah pesawat lift yang di posisikan berdampingan. Hal ini dilakukan untuk mengatasi panggilan lain jika pada salah satu pesawat lift sedang bekerja atau sedang melakukan rute yang berbeda dari panggilan, maka pesawat lift yang lain akan mengambil inisiatif untuk menjawab panggilan tersebut, sementara pesawat lift yang satu tetap bekerja.

2.2 Perancangan Perangkat Keras (*Hardware*)

Gambar 1 dan gambar 2 menjelaskan perancangan perangkat keras dan diagram blok perancangan perangkat keras miniatur lift.



Gambar 1 Perancangan perangkat keras.



Gambar 2 Diagram blok perancangan perangkat keras miniatur lift.

Berikut ini penjelasan mengenai perangkat keras yang digunakan pada tugas akhir ini :

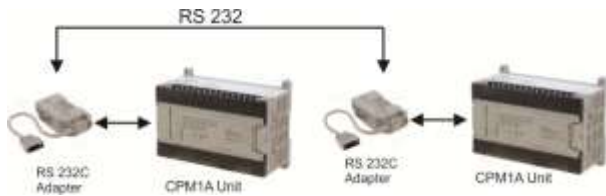
1. *Power supply* merupakan sumber energi yang digunakan dalam tugas akhir ini. Pada tugas akhir ini digunakan dua *power supply* yaitu 24 V DC untuk modul masukan serta keluaran PLC dan 12 V DC untuk motor DC.
2. *Limit switch* lantai di pasang pada setiap lantai dan digunakan untuk sensor kehadiran lift pada setiap lantai.
3. *Push button* naik/turun digunakan untuk memberi masukan ketika masih berada di luar lift. *Push button* merah untuk naik dan *push button* hijau untuk turun.
4. Untuk simulasi ketika penumpang sudah berada didalam lift, *push button* lantai digunakan untuk memilih di lantai mana penumpang tersebut ingin berhenti. Dalam miniatur ini *push button* lantai diletakkan di sebelah kanan dan kiri lift.
5. PLC Omron CPM1A digunakan sebagai pengendali (*controller*) dari pergerakan lift. PLC yang digunakan ada 2 buah, salah satu sebagai *master* dan yang lain sebagai *slave*.
6. *RS 232 adapter* digunakan untuk komunikasi PLC *master* dan PLC *slave* dengan menggunakan sambungan RS 232.
7. Untuk simulasi ketika beban didalam lift melebihi kapasitas maka di gunakan *limit switch* beban. *Limit switch* ini di pasang di bagian alas pesawat/kereta lift.
8. Motor DC 12 V yang dilengkapi dengan *gearbox* digunakan untuk menggerakkan lift naik dan turun.
9. Lampu LED digunakan sebagai indikator di lantai mana lift tersebut berada serta untuk memberi keterangan lift dalam kondisi naik atau turun. LED biru sebagai indikator posisi lift, LED merah sebagai tanda lift sedang turun, dan LED hijau menandakan lift sedang naik.
10. Relay DC 24 V digunakan sebagai rangkaian *driver motor H bridge*.

11. *Buzzer* digunakan sebagai alarm ketika beban dalam lift melebihi kapasitas.

2.2.1 Komunikasi One to One PC Link Connection antara 2 PLC Omron CPM1A

Komunikasi *One to One PC Link Connection* digunakan untuk mengkomunikasikan 2 PLC Omron CPM1A yang digunakan pada tugas akhir ini. Komunikasi ini diperlukan agar kedua PLC dapat saling bertukar data sehingga kedua lift dapat disinkronkan kondisinya. Komunikasi ini juga mempermudah dalam pembagian tugas dari kedua lift yang dilakukan oleh PLC *master*. Hal ini dikarenakan dengan komunikasi ini maka PLC *master* akan menerima data tentang kondisi lift 2 dari PLC *slave* sehingga PLC *master* dapat menentukan pembagian tugas berdasarkan kondisi kedua lift.

Untuk komunikasi ini kedua PLC membutuhkan RS 232 *adapter* yang disambungkan seperti gambar 3.



Gambar 3 One to one PLC Link Connections PLC Omron CPM1A.

2.3 Perancangan Perangkat Lunak (Software)

Pada perancangan perangkat lunak ini digunakan *software CX-Programmer Ver 9* untuk membuat program dan menanamkan program ini pada PLC. Program yang dibuat ada dua, yaitu program pada PLC *master* dan program pada PLC *slave*. Pada dasarnya, kedua program ini hampir sama, tetapi karena pembagian tugas ada pada PLC *master* maka PLC *master* memiliki program yang relatif lebih banyak daripada *slave*.

2.3.1 Program PLC Master

Diagram alir (*flowchart*) pada program yang ditanamkan ke PLC *master* seperti pada gambar 4.

➤ **First Scanning / First Cycle**

Ketika PLC terhubung dengan *power supply*, maka PLC akan menjalankan program awal yaitu *first scanning* atau *first cycle*. *First cycle* ini hanya akan dijalankan sekali pada saat awal pertama kali PLC. Pada *first cycle* ini dideteksi kondisi limit switch lantai, sehingga pada awal dihidupkan LED indikator lantai akan langsung menyala menunjukkan keberadaan lift. Akan tetapi jika lift tidak berada tepat pada lantai tertentu maka *first cycle* akan mengaktifkan *relay* motor turun sehingga lift akan turun

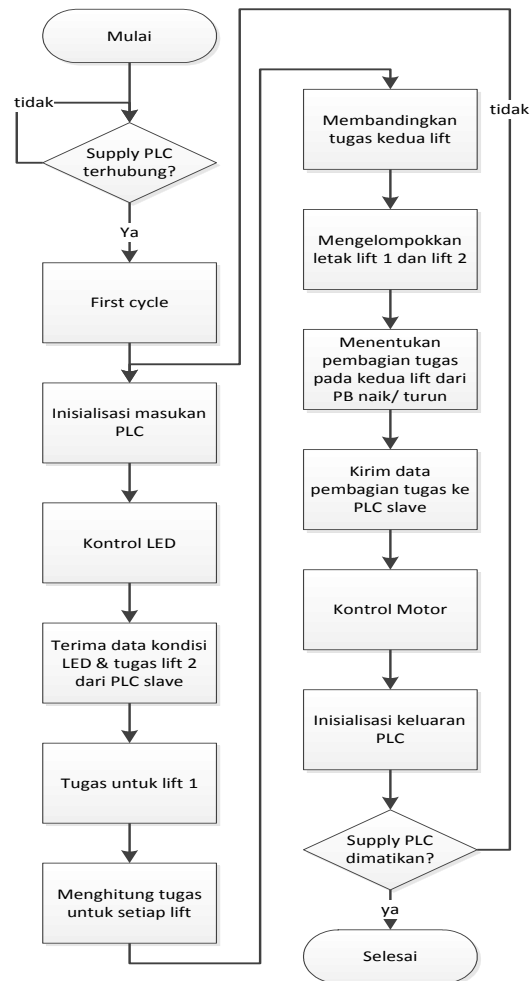
menuju lantai di bawahnya.

➤ **Inisialisasi Masukan dan Keluaran PLC**

Inisialisasi masukan PLC dilakukan dengan mengubah masukan sebenarnya diubah ke alamat bayangan pada PLC. Alamat *channel input* yang sesungguhnya diubah ke alamat imajiner atau alamat bayangan dengan memanfaatkan memori *internal* PLC. Tujuan dari pembuatan alamat imajiner ini adalah untuk kemudahan *maintenance* dan perubahan masukan PLC yang dapat terjadi sewaktu-waktu.

➤ **Kontrol LED Indikator**

Untuk kontrol indikator LED diperoleh dari kondisi *limit switch* lantai. Ketika kontaktor *limit switch* lantai 1 memberikan masukan *high* maka ini akan menghidupkan kontaktor LED indikator lantai 1. Begitu juga untuk *limit switch* lantai 2 yang akan menghidupkan LED indikator lantai 2 dan seterusnya.



Gambar 4 Diagram alir (*flowchart*) pada program PLC *master*.

➤ Data dari PLC Slave

Untuk menentukan pembagian tugas pada kedua lift maka PLC *master* memerlukan data tentang kondisi LED indikator lift 2. Hal ini diperlukan karena dari kondisi LED ini diketahui letak lift 2 sedang berada pada lantai berapa serta sedang dalam kondisi naik atau turun. Dengan mengetahui kondisi LED 2 dari indikator LED, kemudian dibandingkan dengan letak dan kondisi lift 1 dan ditentukan pembagian tugas pada kedua lift. Selain dari letak dan kondisi kedua lift, juga diperlukan tentang jumlah tugas yang diterima oleh kedua lift. Oleh karena itu diperlukan data tentang tugas pada lift 2 yang dikirimkan ke PLC *master*. Selain kondisi LED dan tugas pada lift 2, data *push button* turun juga dikirimkan ke PLC *master*. Hal ini diperlukan untuk proses pembagian tugas dari kedua lift dimana ketika *push button* turun ini ditekan maka PLC *master* yang akan menentukan ke lift mana tugas ini akan diberikan.

➤ Tugas Lift 1

Setelah menerima data dari PLC *slave* tentang tugas pada lift 2, maka PLC *master* juga perlu menentukan tugas untuk lift 1. Tugas ini di beri nama kontaktor *latch*, di mana nantinya kontaktor ini yang akan membuat *relay* motor naik atau *relay* motor turun menyala. *Latch* ini akan aktif dengan pemicu dari *push button* pilih lantai (PB pilih lantai) yang mensimulasikan ketika penumpang sudah berada dalam lift. Selain itu, *latch* ini juga dapat aktif dari hasil pembagian tugas yang dilakukan PLC *master* nantinya. Hasil pembagian tugas ini menggunakan kontaktor PB *sinkron* lantai. Agar kontaktor *latch* ini tetap aktif meskipun pemicunya sudah tidak lagi aktif maka di gunakan fungsi *keep*. Sedangkan untuk mematikan kembali kontaktor *latch* ini menggunakan *limit switch* lantai, karena hal ini menandakan bahwa lift sudah sampai pada lantai yang dituju dan tugas untuk lantai sudah dilaksanakan.

➤ Penghitungan Tugas dari Masing – Masing Lift

Ketika PLC *master* sudah mempunyai data tentang tugas dari masing-masing lift maka dari data tersebut perlu untuk dihitung agar nantinya dapat dibandingkan. Penghitungan tugas ini menggunakan fungsi *counter* pada PLC. *Counter* yang digunakan adalah *reversible counter* dimana *counter* ini dapat digunakan untuk menambah dan mengurangi jumlah hitungannya. Pada *counter* ini juga terdapat fungsi *reset* untuk mengembalikan *counter* ke kondisi awal yaitu pada perhitungan nol. Fungsi *reset* ini dihubungkan ke *first scanning* sehingga ketika awal PLC dihidupkan semua *counter* pada posisi nol. Untuk menambah penghitungan digunakan perubahan kondisi tugas (*latch*) dari *low* ke *high* pada tiap lantainya. Perubahan kondisi dari *low* ke *high* ini dideteksi dengan menggunakan fungsi *Differential Up* (DIFU) pada PLC. Sedangkan untuk mengurangi penghitungan digunakan

perubahan kondisi tugas (*latch*) dari *high* ke *low*. Fungsi yang digunakan untuk mendeteksi perubahan kondisi dari *high* ke *low* pada PLC adalah *Differential Down* (DIFD).

➤ Perbandingan Tugas Kedua Lift (*Compare*)

Ketika tugas kedua lift sudah dihitung maka perlu dilakukan perbandingan untuk pertimbangan ketika proses pembagian tugas. Perbandingan tugas ini dilakukan dengan fungsi *Compare* (CMPR) pada PLC. Fungsi CMPR ini dikombinasikan dengan *special bit* yang ada pada PLC. *Special bit* yang dimaksudkan adalah *Equal* (EQ) atau sama dengan, *Grather Than* (GT) atau lebih dari, dan *Less Than* (LT) atau kurang dari. Cara kerja dari ketiga bit ini adalah sebagai berikut :

- Jika tugas pada lift 1 sama jumlahnya dengan tugas pada lift 2 maka kontaktor EQ akan aktif.
- Jika tugas pada lift 1 lebih dari tugas pada lift 2 maka kontaktor GT akan aktif.
- Jika tugas pada lift 1 kurang dari tugas pada lift 2 maka kontaktor LT akan aktif.

Selain untuk membandingkan dengan tugas dari lift yang lain, tugas pada setiap lift dibandingkan dengan nol untuk mendeteksi jika lift tersebut sedang tidak memiliki tugas. Hal ini dilakukan dengan *special bit* EQ, dimana jika tugas lift sama dengan nol maka kontaktor ini akan aktif.

➤ Pengelompokan Letak Lift

Setelah dibandingkan jumlah tugas kedua lift, maka hal yang perlu dipertimbangkan untuk pembagian tugas adalah letak kedua lift. Untuk lebih menyederhanakan kondisi letak kedua lift maka diperlukan kontaktor-kontaktor penyederhana letak lift. Hal ini dilakukan dengan mengelompokkan letak lift itu ketika *push button* naik atau turun pada lantai tertentu, misal pada lantai dua maka letak lift dikelompokkan menjadi 3 yaitu di bawah lantai dua, pada lantai dua, dan di atas lantai dua. Berikut ini penjelasan pengelompokan tersebut :

- Pengelompokkan letak lift di bawah lantai dua ketika *limit switch* lantai dua tidak aktif dan LED indikator lantai satu aktif atau LED indikator lantai dua aktif tetapi disertai dengan aktifnya LED indikator turun.
- Letak lift dikelompokkan pada lantai dua jika *limit switch* lantai dua aktif.
- Letak lift dikelompokkan di atas lantai dua ketika *limit switch* lantai dua tidak aktif dan LED indikator lantai 3 atau 4 atau 5 aktif, ataupun LED indikator lantai dua aktif tetapi disertai dengan aktifnya LED indikator naik.

Untuk letak lantai yang lain juga mengikuti aturan seperti pada lantai dua tersebut.

➤ Pembagian Tugas pada Kedua Lift

Pembagian tugas pada kedua lift dilakukan berdasarkan pengelompokkan letak kedua lift, kondisi lift sedang naik ataukah turun, serta berdasarkan jumlah tugas kedua lift. Hal ini dilakukan ketika satah satu *push button* naik atau turun ditekan sehingga harus ada salah satu lift yang menuju ke lantai tersebut untuk selanjutnya mengantarkan penumpang ke lantai tujuannya. Pada dasarnya logika pembagian tugas ketika *push button* (PB) lantai satu ditekan sama dengan ketika PB lantai 5 ditekan. Perbedaannya adalah jika pada lantai 1 letak lift dikelompokkan menjadi dua, yaitu pada lantai 1 dan di atas lantai 1 sedangkan pada lantai 5 di kelompokkan menjadi pada lantai 5 dan dibawah lantai 5. Jadi, logika yang digunakan sama untuk dua PB, hanya saja kondisinya berkebalikan, jika pada lantai 1 kondisinya naik, maka pada lantai 5 kondisinya turun. Hal ini juga berlaku untuk PB pada lantai yang lain yaitu :

- PB naik lantai 1 logikanya sama dengan PB turun lantai 5.
- PB naik lantai 2 logikanya sama dengan PB turun lantai 4.
- PB turun lantai 2 logikanya sama dengan PB naik lantai 4.
- PB naik lantai 3 logikanya sama dengan PB turun lantai 3.

➤ Pengiriman Data ke Lift 2 (PLC Slave)

Ketika telah ditentukan pembagian tugas kedua lift, maka jika lift 2 yang mendapatkan tugas harus dikirim kembali ke lift 2 menggunakan *link relay*.

➤ Kontrol Motor

Dari semua proses sebelumnya maka dilakukan pengontrolan motor. Untuk kontrol motor ini dipengaruhi oleh kontaktor tugas (*latch*), *limit switch* lantai 1 dan lantai 5, kontaktor untuk prioritas naik atau turun, kontaktor dari *first cycle*, *limit switch* beban, LED indikator lantai, serta *limit switch* lantai. Untuk mngaktifkan kotaktor motor naik diperoleh dari kombinasi LED indikator dan kontaktor tugas (*latch*). Sedangkan untuk mematikan kontaktor motor naik digunakan kontaktor NC dari prioritas turun, *limit switch* lantai 5, kontaktor motor naik mati *timer*, serta kontaktor motor turun. Prioritas turun digunakan untuk memprioritaskan kondisi turun lift, sedangkan *limit switch* lantai 5 menandakan jika lift sudah berada pada lantai paling atas sehingga kontaktor motor naik harus mati. Kontaktor motor naik mati *timer* merupakan kontaktor yang aktif jika lift harus berhenti di beberapa lantai sehingga pada setiap lantai harus berhenti menggunakan *timer* untuk menggantikan fungsi membuka dan menutupnya pintu lift. Agar motor naik tidak aktif secara bersamaan dengan motor turun, maka pada kontaktor

motor naik diserikan dengan kontaktor motor turun, begitu juga sebaliknya.

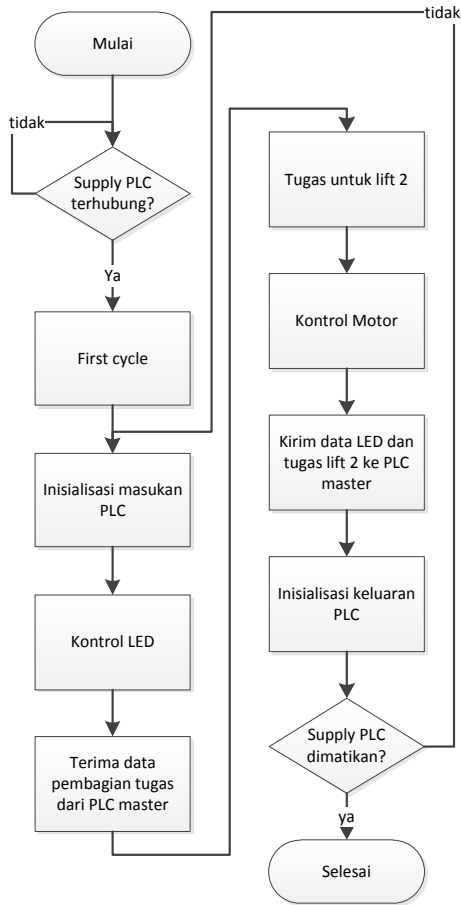
Pada prinsipnya yang memicu aktifnya motor turun sama dengan yang memicu aktifnya motor naik. Perbedaannya hanya terdapat pada *first cycle* yang juga mempengaruhi motor turun.

2.3.2 Program PLC Slave

Diagram alir (*flowchart*) pada program yang ditanamkan ke PLC *slave* seperti pada gambar 5.

Seperti halnya pada PLC master, PLC slave akan menjalankan program ketika ada *power supply* yang terhubung ke PLC. Kemudian akan dilakukan *first cycle* dan inialisasi masukan PLC. Kontrol LED dilakukan berdasarkan kondisi *limit switch* lantai. Setelah itu melalui komunikasi *one to one PC link*, PLC *slave* menerima data pembagian tugas dari lift 1. Dari pembagian tugas tersebut serta dari *push button* pilih lantai, ditentukan tugas untuk lift 2. Selanjutnya dari tugas tersebut dilakukan kontrol motor. Setelah itu dilakukan pengiriman data tentang kondisi LED dan tugas pada lift 2 ke PLC *master*. Selain pada masukan yang diinisialisasikan, pada keluaran juga diinisialisasikan. Program ini akan terus dijalankan dari inialisasi masukan sampai inialisasi keluaran dan akan berhenti ketika *supply* ke PLC dimatikan.

Penjelasan lebih detail tentang semua bagian tersebut sebagian besar sama seperti pada PLC *master*. Perbedaannya hanya pada pengiriman data dari dan ke PLC *master*. Selain itu juga pada PLC *slave* ini tidak terdapat pembagian tugas, jadi program pada PLC *slave* ini lebih sederhana



Gambar 5 Diagram alir (flowchart) pada program PLC slave.

➤ **Penerimaan Data dari PLC Master**

Dari pembagian tugas yang dilakukan PLC master, maka perlu dikirimkan ke PLC slave jika tugas tersebut diberikan ke lift 2.

➤ **Pengiriman Data ke PLC Master**

Dikarenakan PLC master butuh data kondisi LED dan tugas pada lift 2, maka perlu dilakukan pengiriman data tersebut.

3. Hasil dan Analisa

3.1 Pengujian Sistem Secara Keseluruhan

Pada pengujian ini di lakukan beberapa pengujian yang merupakan penggabungan antara push button naik/turun, push button pilih lantai, dan limit switch beban. Pada pengujian ini di lakukan simulasi seperti layaknya lift sebenarnya yaitu ketika ada penumpang di lantai tertentu menekan push button naik/turun, maka salah satu lift akan menjemput penumpang tersebut. Kemudian ketika penumpang sudah berada dalam lift maka push button pilih lantai akan di tekan untuk menuju lantai yang

diinginkan. Selain itu dilakukan juga pengujian jika lebih dari satu push button naik/turun yang ditekan serta pengujian jika power supply dimatikan secara mendadak. Tabel 1 menggambarkan hasil pengujian lift ini.

Tabel 1 Hasil pengujian sistem secara keseluruhan.

No	Posisi Lift		Kondisi Lift		Push button / Limit Switch/ Gangguan	Aksi yang dilakukan	
	Lift 1	Lift 2	Lift 1	Lift 2		Lift 1	Lift 2
1	1	1	↑	↑	PB naik Lt. 2	naik ke Lt. 2	tetap di Lt. 1
2	2	1	↑	↑	PB pilih Lt. 4 lift 1	naik menuju Lt. 4	tetap di Lt. 1
3	3	1	↑	↑	PB turun Lt. 3	sedang naik menuju Lt. 4	naik menuju Lt. 3
4	4	3	↑	↓	LS beban lift 1	buzzer menyala	kondisi lift berubah ↓ buzzer menyala
5	4	3	↑	↓	LS beban lift 2	tetap di Lt. 4	di Lt. 3
6	4	3	↑	↓	PB pilih Lt. 5 & Lt. 2 lift 1	Naik ke Lt. 5	tetap di Lt. 3
7	5	3	↓	↓	PB pilih Lt. 2 & Lt.5 lift 2	Turun ke Lt. 2 (lanjutan tugas no 5)	turun ke Lt. 2
8	2	2	↓	↓	PB naik Lt.3 & PB turun Lt. 3	Naik ke Lt. 3	Naik ke Lt. 5 (lanjutan tugas no 7)
9	3	5	↑	↓	PB naik Lt. 4	Naik ke Lt. 4	Turun ke Lt. 3 (lanjutan tugas no 8)
10	4	3	↑	↓	PB naik Lt. 1 & PB pilih Lt. 5 lift 1	naik ke Lt. 5	Turun ke Lt.1
11	4-5	3-2	↑	↓	Supply di matikan	Lift berhenti antara lantai 4 & 5	Lift berhenti antara lantai 3 & 2
12	4-5	3-2	↑	↓	Supply di nyalakan kembali	Turun ke Lt.4	Turun ke Lt. 2
13	4	2	↓	↓	PB pilih Lt. 5 lift 1 & PB naik Lt. 1	Naik ke Lt. 5	Turun ke Lt. 1
14	5	1	↓	↑	PB turun Lt. 5	Tetap di Lt. 5	Tetap di Lt.1
15	5	1	↓	↑	PB pilih Lt. 3 Lift 1	Turun ke Lt. 3	Tetap di Lt. 1

Tabel 1 diperoleh dengan simulasi yang mirip pada pengoperasian lift sebenarnya. Penjelasan simulasi yang dilakukan pada tabel 1 adalah sebagai berikut :

1. Ketika posisi awal lift berada pada lantai 1 semua dan ada seorang penumpang di lantai 2 ingin naik ke lantai 4, maka ia menekan *push button* naik pada lantai 2 dan lift 1 naik menuju lantai 2.
2. Setelah orang tersebut masuk ke dalam lift ia menekan *push button* pilih lantai 4, karena tujuan orang tersebut adalah lantai 4. Oleh karena itu lift 1 tersebut mengantarkan ke lantai 4.
3. Ketika lift 1 dalam perjalanan menuju lantai 4, ada penumpang lain yang berada pada lantai 3 ingin turun dengan menekan *push button* turun lantai 3. Hal ini membuat lift 2 yang dari awal belum memperoleh tugas naik menuju Lt. 3 untuk mengantarkan penumpang tersebut.
4. Ketika lift 1 sampai ke lantai 4 ada banyak penumpang yang naik sehingga kelebihan beban dan menyebabkan *limit switch* beban on. Hal ini menyebabkan *buzzer* menyala sebagai tanda alarm tanda kelebihan beban.
5. Begitu juga ketika lift 2 berada di lantai 3, kemudian ada penumpang yang melebihi kapasitas beban sehingga *buzzer* lift 2 pun menyala.
6. Pada pengujian ke 6 dilakukan penekanan 2 *push button* pilih lantai sekaligus pada lift 1, yaitu pada lantai 5 dan lantai 2. Hal ini merupakan simulasi ketika ada banyak penumpang yang naik ke lift 1 dan ada yang ingin naik, ada pula yang ingin turun. Lift 1 mengerjakan tugas sesuai prioritasnya terlebih dahulu yaitu naik dulu ke lantai 5.
7. Pada lift 2 juga dilakukan hal yang sama seperti pengujian no 6. *Push button* pilih lantai yang ditekan adalah lantai 2 dan lantai 5. Lift 2 juga menjalankan tugas yang sesuai prioritasnya terlebih dahulu yaitu turun ke lantai 2. Sementara itu lift 1 melanjutkan tugasnya untuk turun ke lantai 2.
8. Kemudian dilakukan pengujian ketika ada beberapa penumpang pada lantai 3 dan mereka ada yang ingin naik dan ada yang ingin turun. Kedua *push button* naik dan turun pada lantai 3 ditekan. Hal ini membuat lift 1 yang sudah tidak memiliki tugas naik menuju lantai 3 sedangkan lift 2 melanjutkan tugas sebelumnya untuk naik ke lantai 5.
9. Disaat lift 1 sedang kondisi naik dari lantai 3, ada penumpang juga pada lantai 4 yang ingin naik, sehingga lift 1 naik ke lantai 4. Lift 2 turun ke lantai 3 untuk menjemput penumpang yang ingin turun pada pengujian nomor 8.
10. Setelah lift 1 mengambil penumpang pada lantai 3 dan 4 untuk naik, penumpang menekan *push button* pilih lantai 5. Disaat yang bersamaan ada penumpang yang ingin naik pada lantai 1 dan menekan *push button* naik lantai 1. Lift 1 naik ke lantai 5 sedangkan lift 2 turun ke lantai 1.
11. Pada pengujian ini dilakukan pengujian jika power *supply* mati mendadak semua lift akan berhenti diposisi akhirnya. Lift 1 berhenti diantara lantai 4 dan 5, sedangkan lift 2 berhenti diantara lantai 3 dan 2.
12. Ketika power *supply* kembali dinyalakan maka lift turun menuju lantai terdekat yang berada dibawahnya. Lift 1 turun ke lantai 4 dan lift 2 turun ke lantai 3. Penumpang harus menekan kembali *push button* yang diinginkan untuk naik atau turun ataupun *push button* pilih lantai. Hal ini dikarenakan data pada controller di reset.
13. Penumpang menekan kembali *push button* naik lantai 1 dan *push button* pilih lantai 5 lift 1. Hal ini menyebabkan lift 1 naik ke lantai 5 dan lift 2 turun ke lantai 1.
14. Disaat yang sama ada penumpang yang menekan *push button* turun lantai 5 dan ingin turun ke lantai 3. Dikarenakan lift 1 sudah berada pada lantai 5 maka lift tidak mengalami perubahan posisi.
15. Setelah penumpang tersebut masuk ke lift 1, dia menekan *push button* pilih lantai 3 pada lift 1 untuk turun ke lantai 3. Hal ini menyebabkan lift 1 turun ke lantai 3 untuk mengantarkan penumpang tersebut.

4. Kesimpulan

Berdasarkan pengujian dan analisis yang dilakukan pada *plant* miniatur 2 lift ini, didapatkan kesimpulan bahwa *plant* miniatur 2 lift ini dapat dikontrol menggunakan 2 PLC Omron CPM1A yang dikomunikasikan dengan *one to one PC link connection* menggunakan kabel RS 232. Pembagian tugas dari *push button* naik/turun setiap lantai pada miniatur 2 lift ini berdasarkan letak lift, kondisi lift sedang naik atau turun, serta jumlah tugas pada setiap lift. Simulasi ketika lift mengalami kelebihan beban menyebabkan alarm *buzzer* menyala dan akan mati kembali ketika beban di dalam lift sudah tidak melebihi kapasitas. Jika *power supply* mati secara mendadak maka seluruh sistem lift akan berhenti dan ketika *power supply* dinyalakan kembali maka program lift akan masuk ke *first cycle* untuk menyalakan indikator letak lift atau untuk menurunkan lift menuju lantai terdekat yang ada dibawahnya. Setelah *power supply* dinyalakan kembali, sistem memerlukan penekanan kembali pada masukan PLC untuk menyelesaikan tugas yang terakhir.

Saran yang dapat dilakukan untuk pengembangan sistem lebih lanjut antara lain dapat ditambahkan sistem monitoring posisi lift pada software komputer. Selain itu dapat pula ditambahkan pintu lift dan sistem keamanan yang lebih detail agar lebih menyerupai lift yang sebenarnya. Adanya kontrol kecepatan motor penggerak sehingga pergerakan lift saat *stop to maks* dan *maks to stop* lebih halus dengan *softstart* dan *softstop* akan menambah kemiripan sistem dengan lift aslinya.

Daftar Pustaka

- [1]. Faishal, Ahmad, "LIFT CONTROL BY USING PROGRAMMABLE LOGIC CONTROLLER (PLC) FOR STUDENTS KITS", Skripsi-S1, Teknik Elektro Universiti Teknologi Malaysia, 2009.
- [2]. Hasan, Ali, "Hydraulic Elevators Basic Components", <http://alihassanelashmawy.blogspot.com/2012/04/hydraulic-elevators-basic-components.html>, 2012.
- [3]. Kim, Hun Mo, "A Design for Group Controller of Building Using Adaptive Dual Fuzzy Algorithm", Sungkyunkwan University, Mechanical Engineering, 2005.
- [4]. Kurniawan, M. Supono, *Perancangan Simulasi Supervisory Control And Data Acquisition pada Prototipe Sistem Listrik Redundant*, Skripsi-S1, Teknik Elektro UNDIP, 2012.
- [5]. Kusuma, Yuriadi, "Elevator", Universitas Mercubuana, 2010.
- [6]. Nagatani, T., "Complex behavior of elevators in peak traffic", Shizuoka University, Mechanical Engineering, 2003.
- [7]. Nagatani, T., "Dynamical transitions in peak elevator traffic", Shizuoka University, Mechanical Engineering, 2003.
- [8]. Steven, *Perancangan Simulator Lift Gedung 6 Lantai Menggunakan Mikrokontroler Atmega8535*, Skripsi-S1, Teknik Elektro UNDIP, 2011.