

## Perancangan simulator lift 3 lantai menggunakan diagram keadaan

Edi Rakhman<sup>1\*</sup>, Dadan Nurdin Bagenda<sup>2</sup>, Noor Cholis Basjaruddin<sup>3</sup>

<sup>1,2,3</sup>Jurusan Teknik Elektro, Politeknik Negeri Bandung

Jln. Gegerkalong Hilir, Desa Ciwaruga, Kecamatan Parongpong, Kabupaten Bandung Barat 40559, Indonesia

<sup>1\*</sup>edr@polban.ac.id, <sup>2</sup>noorcholis@polban.ac.id, <sup>3</sup>dadannb@polban.ac.id

### ABSTRAK

Lift telah lama digunakan pada berbagai gedung bertingkat. Kelebihan lift dibandingkan dengan tangga dan eskalator adalah penggunaan ruang yang lebih efisien dan dapat digunakan oleh pengguna dengan berbagai kondisi fisik. Berbeda dengan lift, tangga dan eskalator tidak disarankan digunakan oleh pengguna kursi roda karena tidak nyaman dan berbahaya. Selain itu lift juga lebih tepat untuk mengangkut barang antar lantai dibanding tangga atau eskalator. Pengembangan lift dengan jumlah jangkauan lantai yang semakin banyak memerlukan metode dalam tahap perancangan terutama pada pengambilan keputusan ketika lift menerima permintaan dari pengguna. Pada penelitian ini, digunakan diagram keadaan dalam perancangan lift 3 lantai. Melalui diagram keadaan yang dilanjutkan dengan pembuatan *ladder diagram*, rancangan tersebut selanjutnya diimplementasikan menggunakan miniatur lift 4 lantai yang dikendalikan oleh *programmable logic controller* (PLC). Hasil pengujian rancangan pengendalian lift 3 lantai menggunakan diagram keadaan menunjukkan bahwa permintaan pengguna lift dapat dilayani dengan baik dan lift bekerja sesuai dengan rancangan.

**Kata kunci:** lift, diagram keadaan, *ladder diagram*, PLC

### ABSTRACT

*Elevators have long been used in multi-storey buildings. The advantages of elevators compared to stairs and escalators are more efficient use of space and can be used by users with various physical conditions. Unlike elevators, stairs and escalators are not recommended for use by wheelchair users because they are uncomfortable and dangerous. In addition, elevators are also more appropriate for transporting goods between floors than stairs or escalators. The development of an elevator with an increasing number of floors requires a method in the design stage, especially in making decisions when the elevator receives requests from users. In this study, status diagrams were used in designing a 3-floor elevator. Through a status diagram followed by making a ladder diagram, the design is then implemented using a 4-floor miniature elevator controlled by a programmable logic controller (PLC). The test results of the 3-floor elevator control design using the status diagram show that the requests of elevator users can be served well and the elevator works according to design.*

**Keywords:** lift, state diagram, ladder diagram, PLC

## 1. PENDAHULUAN

Hampir seluruh bangunan bertingkat lebih dari 3 lantai saat ini menggunakan lift sebagai sarana untuk berpindah antar lantai. Kelebihan lift dibanding tangga dan eskalator antara lain dibutuhkannya ruang yang relatif kecil, lebih mudah digunakan oleh pengguna dengan berbagai keadaan termasuk pengguna yang harus menggunakan kursi roda, serta singkatnya waktu yang dibutuhkan oleh pengguna untuk mencapai lantai lain. Kelebihan lain adalah dari sisi minimnya tenaga yang dibutuhkan pengguna untuk berpindah lantai dan kenyamanan yang lebih baik bagi pengguna. Selain kelebihan, lift mempunyai kekurangan antara lain jumlah pengguna pada lama waktu tertentu lebih sedikit dibanding tangga dan eskalator serta tingkat bahaya yang lebih besar dibanding tangga dan eskalator.

Penelitian dalam pengembangan lift telah banyak dilakukan, seperti pada [1] melakukan perancangan lift 3 lantai menggunakan mikrokontroler dengan dilengkapi pengendali buka-tutup pintu *box/car* lift. Kemudian, penelitian pada [2] merancang lift berbasis PLC menggunakan metode Fuzzy Logic untuk mengendalikan berat penumpang. Penelitian [3] membuat prototipe sistem aplikasi lift

barang empat lantai yang dirancang untuk mengirim dan memanggil muatan barang telah berhasil dibuat dengan menggunakan pemrograman PLC. Pada perumusan masalah dinyatakan bahwa lift barang ini dirancang/diprogram untuk dapat melakukan pengiriman barang ke lebih dari satu lantai sekaligus dalam satu perintah pengiriman [3]. Selanjutnya, penelitian [4] menitikberatkan pada bagaimana merancang dan mengimplementasi lift 3 lantai dengan PLC sebagai kontrolernya dan dimonitor via HMI. Adapula penelitian yang membuat modul praktikum trainer lift berbasis PLC pada mata pelajaran instalasi motor listrik di SMK Krian 1 Sidoarjo [5]. Penelitian [6] membuat simulator lift 4 lantai berbasis PLC yang diaplikasikan sebagai modul praktikum setingkat SMK. Keempat jurnal yang telah disebutkan ini, hanya menggunakan metode aljabar Boolean sesuai dengan urutan sekuensial kerja alat. Dalam membuat program PLC-nya tidak menggunakan metode tertentu untuk membangun *ladder diagram*.

Pada penelitian ini dirancang lift 4 lantai untuk kegiatan praktikum mekatronika dengan menggunakan *project based learning* [7] sebagai metode pembelajarannya. Pada proses perancangannya dititikberatkan pada bagaimana merancang *ladder diagram* PLC (program PLC). Model lift 4 lantai itu dijadikan acuan untuk membuat program dasar PLC CP1E 30 I/O yang jika diproyeksikan ke lantai lebih banyak akan mudah dikembangkan. Tujuan penelitian ini adalah merancang lift 4 lantai dengan menggunakan *ladder diagram* PLC berdasarkan metode diagram keadaan (*state diagram*) untuk menghasilkan program yang lebih sistematis.

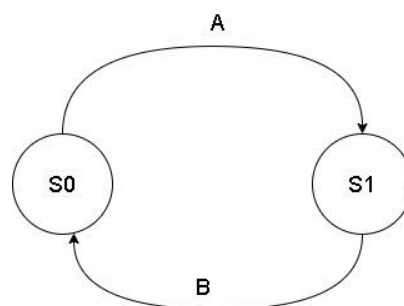
## 2. METODE PENELITIAN

Pada bagian ini akan dijelaskan teori diagram keadaan, serta rancangan diagram keadaan dan rancangan *ladder diagram* untuk lift 3 lantai. Melalui perancangan diagram keadaan dapat dibuat *ladder diagram* yang selanjutnya diimplementasikan menggunakan miniatur lift 4 lantai yang dikendalikan oleh PLC.

### 2.1 Diagram keadaan

*State diagram* atau selanjutnya disebut diagram keadaan adalah sebuah metode yang merepresentasi visual dari perilaku sistem yang terdiri dari serangkaian status atau keadaan yang mungkin dialami oleh sistem tersebut. Diagram keadaan digunakan dalam rekayasa perangkat lunak dan teknik sistem lainnya untuk menggambarkan bagaimana sistem bereaksi terhadap masukan dan bagaimana dapat bertransisi dari satu keadaan ke keadaan lainnya.

Hubungan yang erat antara diagram keadaan dengan *ladder diagram* bahwa *ladder diagram* dapat dibentuk dari persamaan aljabar Boolean (yang dibentuk dari gerbang NOT, OR, dan AND, serta turunannya) [8]. Sementara, diagram keadaan dapat dianalisis dan menghasilkan persamaan aljabar Boolean. Gambar 1 dapat menjadi contoh ilustrasi yang menghubungkan kedua entitas tersebut.



Gambar 1. Hubungan dua status

Dari Gambar 1, persamaan Aljabar Boolean diperoleh untuk pengguna S0 seperti pada persamaan (1)

$$S0 = (S0 + B). \bar{A} \quad (1)$$

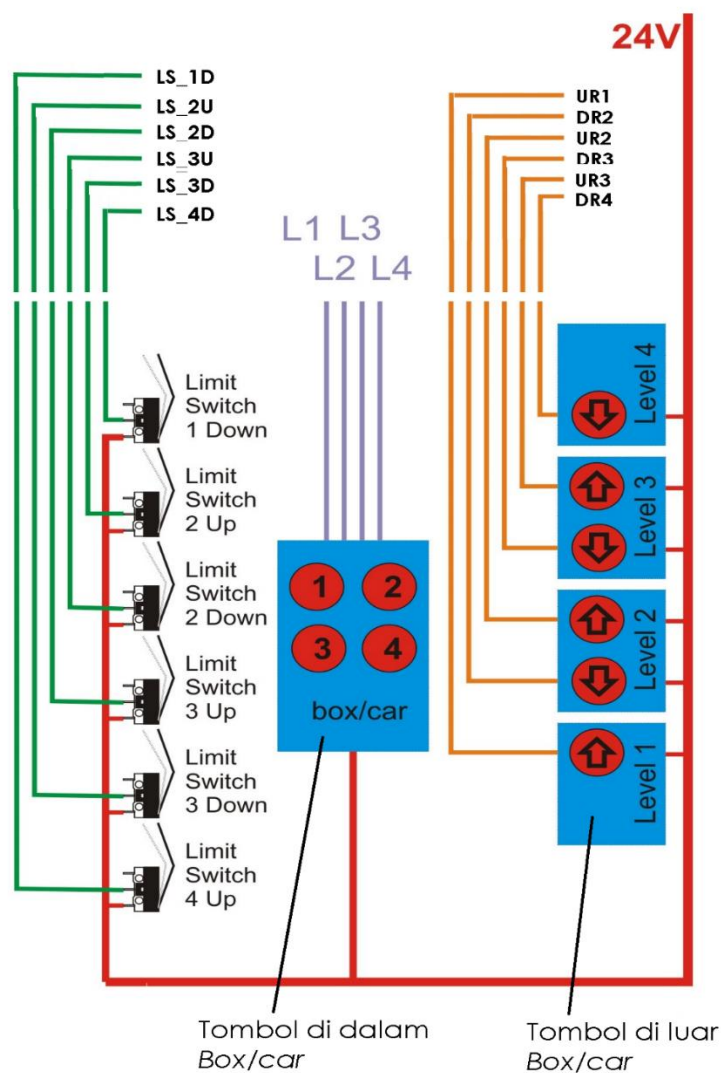
dan untuk *state* S1 seperti pada persamaan (2)

$$S1 = (S1 + A). \bar{B} \tag{2}$$

Pengguna S0 tersebut ekuivalen dengan S0 yang di-OR-kan dengan variabel B dan hasilnya di-AND-kan dengan NOT A. Begitu juga untuk S1 bahwa S1 ekuivalen dengan S0 yang di-OR-kan dengan A dan hasilnya di-AND-kan dengan NOT B. S0 dan S1 menggambarkan variabel keadaan. Variabel A dan B menggambarkan transisi. Baik S0, S1, A, dan B adalah variabel bilangan biner sesuai dengan ranah aljabar Boolean.

### 2.2 Rancangan Diagram keadaan Lift 3 Lantai

Program yang dirancang menggunakan diagram keadaan dimaksudkan adalah program dasar kendali lift 3 dari 4 lantai yang disediakan. Gambar 2 adalah skema tombol dan *limit switch* lift 4 lantai yang menjadi acuan untuk merancang diagram keadaan. Ada 2 kasus yang menjadi dasar program tersebut. Pertama untuk pengguna yang berada di setiap lantai sebelum masuk *box/car* lift yang diilustrasikan oleh Gambar 3 dan posisi pengguna di dalam *box/car* yang diilustrasikan oleh Gambar 4.

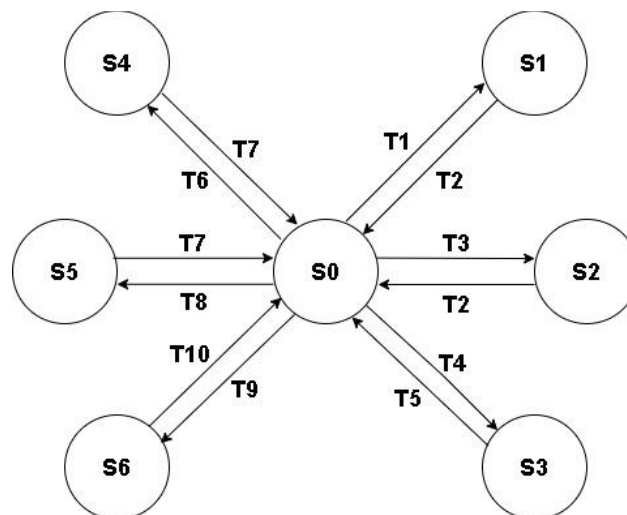


Gambar 2. Skema tombol dan *limit switch* lift 4 lantai

Gambar adalah diagram keadaan yang menggambarkan cara kerja lift 3 lantai untuk pengguna berada di setiap lantai dan masih di luar *box/car*, yang diambil dari skema lift 4 lantai (Gambar 2). Pengguna S0 adalah keadaan awal (inisial) untuk semua Keluaran dalam keadaan mati (OFF) dan

menjadi sentral diagram keadaannya. Hal ini karena sifat lift yang selalu akan berhenti di posisi setiap lantai. Itu artinya motor UP dan DOWN yang menggerak *box/car* akan berhenti manakala terkena setiap *limit switch* yang diatur, baik saat *box/car* naik atau turun.

Dengan mengambil asumsi lift bekerja hanya 3 lantai terdapat tombol-tombol di setiap pintu masuk lift di setiap lantai, yakni tombol (dengan variabel) UR\_1 di lantai 1, DR\_2 dan UR\_2 di lantai 2, serta DR\_3 di lantai 3. Sementara DR\_4 dan UR\_3 tidak digunakan karena hanya 3 lantai yang difungsikan. Maksud permintaan (*request*) tombol lantai adalah kondisi pengguna menekan tombol untuk setiap lantai. Sebagai contoh, jika pengguna menekan tombol UR\_1 di lantai 1 dan sementara lift berada pada posisi lantai 3 (*limit switch* LS\_3D berlogika 1), maka motor DOWN akan aktif yang membuat *box/car* akan bergerak turun dan berhenti di lantai 1, dengan mekanisme dihentikan oleh *limit switch* LS\_1D. Terlihat di Gambar 3. Dari pengguna S0, T1 berlogika 1 karena logika tombol UR\_1 (tombol UP di lantai 1) serta *limit switch* LS\_3D berlogika 1 bersamaan. Hal yang sama terjadi jika lift berada di lantai 2. Logika T2 ekuivalen dengan UR\_1 dan *limit switch* LS\_2D berlogika 1 (bersamaan). Untuk kasus ini, mekanisme ini tidak berlaku (dan tidak digambarkan di diagram keadaan), seandainya *box/car* berada di lantai 1. Jadi, pengguna S1 s.d. S3 menggambarkan *pengguna* berada di lantai 1 atau 2 yang hendak ke lantai yang lebih tinggi. Demikian juga, jika pengguna berada di lantai 3 atau 2 yang hendak turun, maka mekanismenya digambarkan pada pengguna S4 s.d. S6. Perlu diketahui keadaan S0 adalah kondisi keluaran yang tidak aktif semua pada saat *box/car* sudah mencapai lantai tertentu.



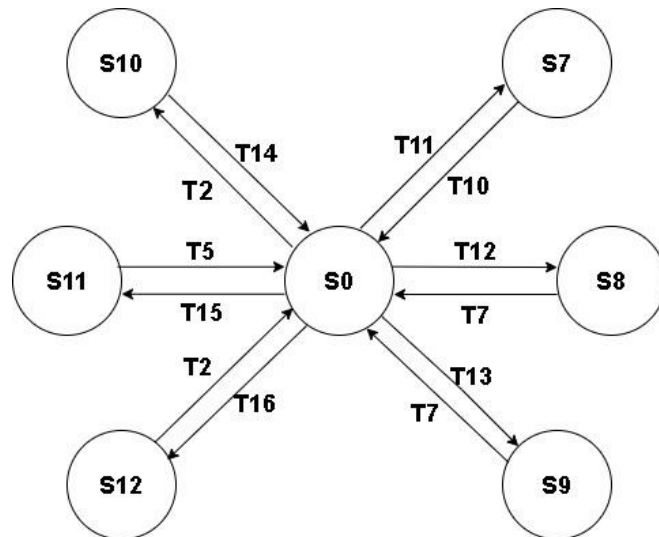
Gambar 3. Diagram keadaan lift 3 lantai untuk permintaan (*request*) di luar *box/car*

Kasus berikutnya adalah menjelaskan mekanisme permintaan (*request*) saat pengguna berada di dalam *box/car* untuk memilih lantai yang dikehendaki. Jika pengguna menekan tombol yang sesuai lantainya, maka lift tidak akan bereaksi. Misalkan, pengguna sedang berada di lantai 1 dan sudah masuk ke *box/car*. Hanya ada 2 kemungkinan memilih tombol lantai, yakni nomor 2 atau 3. Begitu juga di lantai 2, permintaannya untuk lantai 1 dan 3 dan di lantai 3 untuk lantai 1 dan 2. Dengan ada 6 kemungkinan permintaan. Dari kasus ini jumlah *pengguna* yang dihasilkan adalah 6 buah. Diagram keadaan yang dihasilkan tersaji pada Gambar 4.

Untuk merancang *ladder diagram* dari sebuah diagram keadaan, diperoleh tiga kelompok persamaan aljabar Boolean [9], yakni:

- Kelompok persamaan transisi
- Kelompok persamaan pengguna
- Kelompok persamaan keluaran

Untuk kelompok persamaan transisi dan persamaan pengguna (keadaan) disajikan ke dalam Tabel 1 dan Tabel 2 karena dilengkapi dengan keterangan transisi dan keadaan tertentu. Penempatan kelompok *ladder diagram* haruslah dimulai dari program transisi, lalu *state*, dan diakhir dengan kelompok *ladder diagram* keluaran.



Gambar 4. Diagram keadaan lift 3 lantai untuk permintaan tombol di dalam *box/car*

Kelompok persamaan transisi:

Tabel 1. Persamaan transisi

Kode dan Persamaan Transisi	Keterangan
$T1 = (UR\_1).(LS\_3U)$	UR lantai 1, posisi <i>box/car</i> di lt.3
$T2 = (LS\_1D)$	sensor posisi lt.1 akibat UR_1
$T3 = (UR\_1).(LS\_2U)$	UR lantai 1, posisi <i>box/car</i> di lt.2
$T4 = (UR\_2).(LS\_3U)$	UR lantai 2, posisi <i>box/car</i> di lt.3
$T5 = LS\_2D$	sensor posisi lt.2 akibat UR_2
$T6 = (UR\_3).(LS\_1D)$	DR lantai 3, posisi <i>box/car</i> di lt.1
$T7 = (LS\_3U)$	sensor posisi lt.3 akibat DR_3
$T8 = (DR\_3).(LS\_2D)$	DR lantai 3, posisi <i>box/car</i> di lt.2
$T9 = (DR\_2).(LS\_1D)$	DR lantai 2, posisi <i>box/car</i> di lt.1
$T10 = LS\_2U$	sensor posisi lt.2 akibat DR_2
$T11 = (LS\_1D).(L\_2)$	di dalam <i>box/car</i> di lt.1 Up ke lt.2
$T12 = (LS\_1D).(L\_3)$	di dalam <i>box/car</i> di lt.1 Up ke lt.3
$T13 = (LS\_2D).(LS\_2U).(L\_3)$	di dalam <i>box/car</i> di lt.2 Up ke lt.3
$T14 = (LS\_2D).(LS\_2U).(L\_1)$	di dalam <i>box/car</i> di lt.2 Down ke lt.1
$T15 = (LS\_3U).(L\_2)$	di dalam <i>box/car</i> di lt.3 Down ke lt.2
$T16 = (LS\_3U).(L\_1)$	di dalam <i>box/car</i> di lt.3 Down ke lt.1

Kelompok persamaan keadaan:

Tabel 2. Persamaan keadaan

Kode dan persamaan pengguna	Status keluaran
$S1 = (S1 + T1). \overline{T2}$	MOTOR_DN=1 , MOTOT_UP=0
$S2 = (S2 + T3). \overline{T2}$	MOTOR_DN=1 , MOTOT_UP=0
$S3 = (S3 + T4). \overline{T5}$	MOTOR_DN=1 , MOTOT_UP=0
$S4 = (S4 + T6). \overline{T7}$	MOTOR_DN=0 , MOTOT_UP=1
$S5 = (S5 + T8). \overline{T7}$	MOTOR_DN=0 , MOTOT_UP=1
$S6 = (S6 + T9). \overline{T10}$	MOTOR_DN=0 , MOTOT_UP=1
$S7 = (S7 + T11). \overline{T2}$	MOTOR_DN=0 , MOTOT_UP=1
$S8 = (S8 + T12). \overline{T2}$	MOTOR_DN=0 , MOTOT_UP=1
$S9 = (S9 + T13). \overline{T5}$	MOTOR_DN=0 , MOTOT_UP=1
$S10 = (S10 + T14). \overline{T7}$	MOTOR_DN=1 , MOTOT_UP=0
$S11 = (S11 + T15). \overline{T7}$	MOTOR_DN=1 , MOTOT_UP=0
$S12 = (S12 + T16). \overline{T10}$	MOTOR_DN=1 , MOTOT_UP=0

Kelompok persamaan keluaran:

$$\text{MOTOR\_UP} = S4+S5+S6+S7+S8+S9$$

$$\text{MOTOR\_DN} = S1+S2+S3+S10+S11+S12$$

$$\text{LANTAI\_1} = S1+S2+S10+S12$$

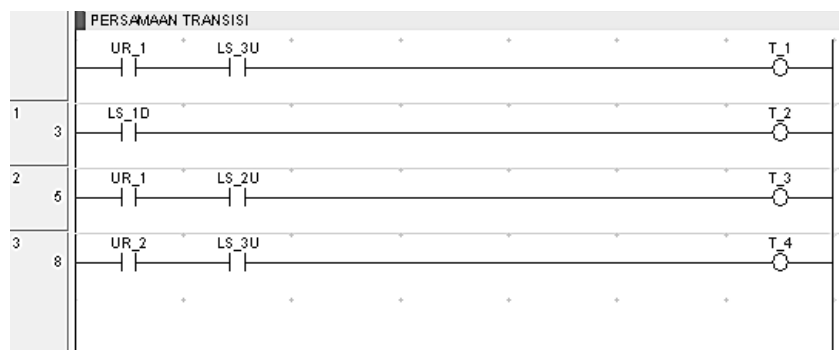
$$\text{LANTAI\_2} = S3+S6+S7+S11$$

$$\text{LANTAI\_3} = S4+S5+S8+S9$$

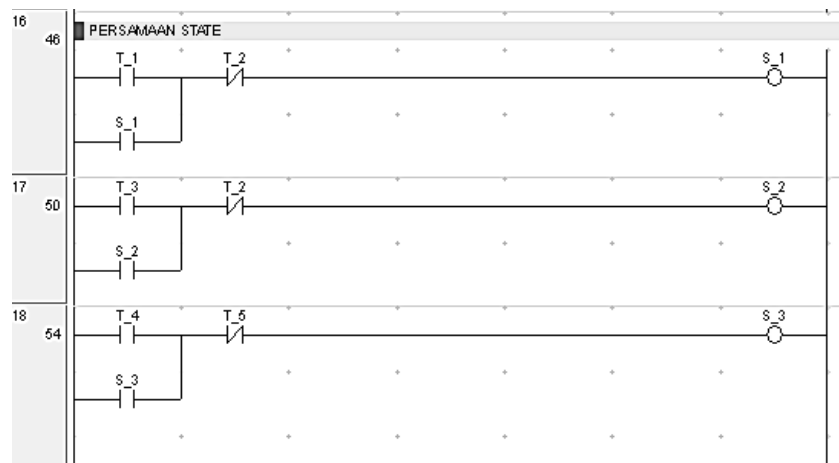
Kelompok persamaan keluaran ini diambil berdasarkan **Error! Reference source not found.** dan Tabel 2, yakni hubungan antara pengguna dan keluaran. Terdapat 5 keluaran PLC, yaitu keluaran MOTOR\_UP dan MOTOR\_DN digunakan untuk menggerakkan *box/car* ke atas maupun ke bawah. Sedangkan LANTAI\_1, LANTAI\_2, serta LANTAI\_3 digunakan untuk status *box/car* sedang proses bergerak ke lantai mana. Untuk menginterpretasi Tabel 1 dan Tabel 2, digunakan sifat tabel karakteristik gerbang logika digital [10], [13] agar menghasilkan kode program *ladder diagram*, seperti halnya dijelaskan pada bagian berikutnya.

### 2.3 Rancangan Ladder Diagram

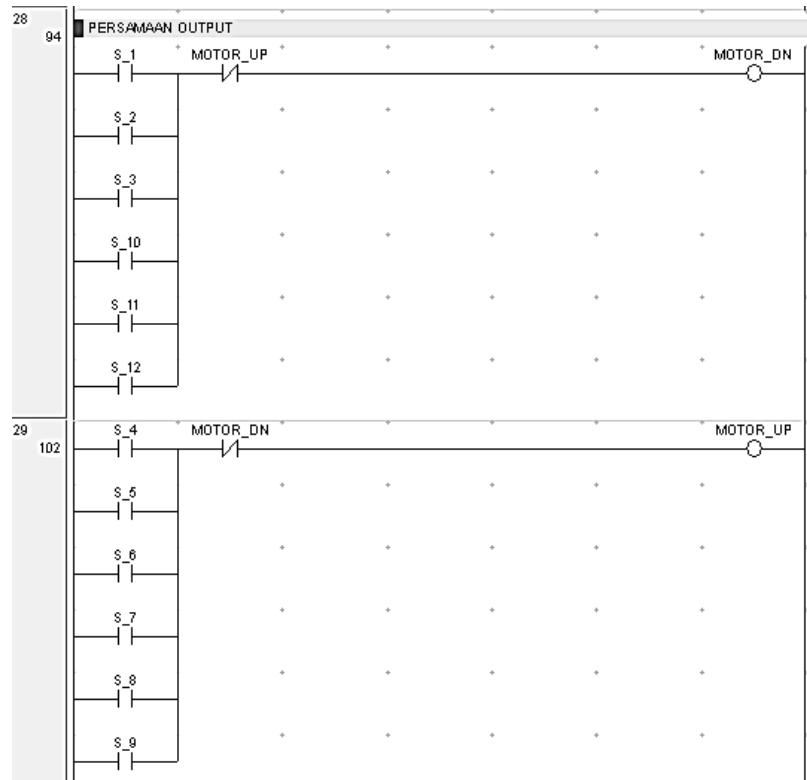
Ketiga kelompok persamaan yang telah diuraikan di atas selanjutnya diterapkan pada *ladder diagram* PLC. Keluaran persamaan aljabar Boolean menjadi keluaran pada *ladder diagram* [12], [14], [15]. Pengujian dimulai dilakukan dengan menggunakan simulator yang dimiliki oleh perangkat lunak PLC yang digunakan. Parameter yang diamati adalah interaksi hubungan antar masukan PLC dengan keluaran. Gambar 5 dan Gambar 6 menggambarkan *ladder diagram* untuk persamaan transisi dan persamaan keadaan yang tidak seluruhnya ditampilkan. Dengan membaca Tabel 1 dan Tabel 2 saja sudah tergambar hubungan antara bagian keluaran *ladder diagram* Ti dan Si (i adalah indeks variabel) dengan masukannya sudah mewakili gambaran implementasi *ladder diagram*nya. Sedangkan untuk keluaran di sini hanya menyajikan keluaran MOTOR\_UP dan MOTOR\_DOWN seperti yang tersaji pada Gambar 7.



Gambar 5. Cuplikan *ladder diagram* untuk persamaan transisi



Gambar 6. Cuplikan *ladder diagram* untuk persamaan keadaan



Gambar 7. Ladder diagram untuk persamaan keluaran

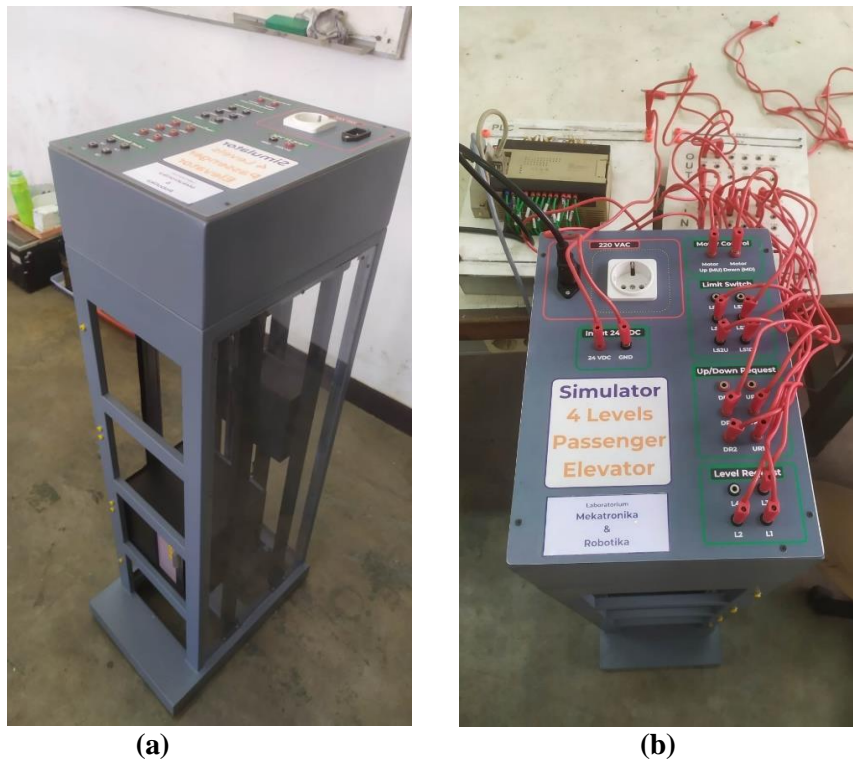
### 3. HASIL DAN PEMBAHASAN

Perancangan menggunakan diagram keadaan dan *ladder diagram* selanjutnya diimplementasikan pada miniatur lift berbasis PLC yang telah dikembangkan. Ukuran miniatur lift adalah 425 mm x 280 mm x 1015 mm dibuat dengan material utama besi dan akrilik. Kecepatan linier lift adalah 1 cm/detik.

Hasil pengujian berupa relasi antara keluaran dengan *state* tersaji pada Tabel 3. Dari tabel tersebut, motor menggerakkan *box/car* turun disebabkan oleh Keadaan S1, S2, S3, S10, S11, atau S12 berlogika 1. Sedangkan *box/car* akan naik oleh keadaan S4, S5, S6, S7, S8, atau S9 berlogika 1. Mekanisme yang mendasar agar saat keadaan *box/car* sedang turun oleh variabel Keluaran MOTOR\_DN berlogika 1, maka variabel Keluaran MOTOR\_DN pada rung nomor baris 28 (Gambar 7) dimatikan oleh MOTOR\_DN pada rung baris nomor 29. Istilah ini dikenal dengan *interlock*. Hal ini berlaku sebaliknya saat MOTOR\_UP berlogika 1. Karena variabel Keluaran LANTAI\_1, LANTAI\_2, dan LANTAI\_3 tidak dihubungkan ke modul keluaran seperti lampu indikator atau pun penyaji digital (*seven segment*), maka pengamatannya hanya sebatas pada komputer saja melalui perangkat lunak PLC yang digunakan.

Karena pada penelitian ini hanya merancang sistem yang bersifat mendasar, maka ada beberapa hal yang tidak dapat dilakukan atau dilaksanakan oleh program PLC, diantaranya:

- Belum memberi jalan/peluang bagi lebih dari 1 pengguna bersamaan dalam memasukan input saat pengguna berada di lantai yang berbeda. Data yang akan terbaca oleh PLC adalah pengguna yang paling pertama yang menekan tombol.
- Belum memberi informasi dalam bentuk *display* yang menunjukkan posisi lift yang menunjukkan posisi saat ini dan akan bergerak menuju lantai mana.
- Tidak melibatkan kendali buka-tutup pintu *box/car*.
- Orientasi program yang diajukan hanya untuk 3 lantai, meskipun prototipe lift memiliki 4 lantai



Gambar 2. Realisasi perangkat keras lift: (a) lift 4 lantai dilihat dari samping depan, (2) panel lift

Adapun pertimbangan menggunakan rancangan untuk 3 lift didasarkan pemikiran bahwa 3 lantai adalah dasar untuk pijakan awal merancang *ladder diagram* berdasarkan metode diagram keadaan. Tombol untuk setiap lantai diletakkan di sebelah kiri pintu. Untuk lantai paling bawah dan paling atas hanya disediakan 1 tombol. Lift dirancang hanya terdiri dari *box/car* yang digerakkan oleh motor AC 1 fasa. Sementara, kontrolernya, berupa modul terpisah yang dihubungkan melalui sejumlah kabel jumper seperti pada Gambar 5(b). Sementara, di lantai di antaranya, dipasang 2 tombol. Dari rancangan ini akan dengan mudah dikembangkan ke lebih dari 3 lantai. Perancangan fisik lift 4 lantai dimaksudkan di sini adalah untuk memberi peluang pembelajaran bagi mahasiswa mengembangkan sistem kendali sampai dengan 4 lantai. Lift 4 lantai sebagai modul praktikum adalah hasil pertimbangan dari segi biaya pembuatan dan keterbatasan jumlah kanal modul masukan-keluaran PLC yang ada. Penambahan lantai untuk dikendalikan, akan menambah jumlah pengguna dan juga transisinya.

Tabel 3. Relasi status dengan luaran

State	Lantai_1	Lantai_2	Lantai_3	MOTOT_UP	MOTOR_DN
S1	1				1
S2	1				1
S3		1			1
S4			1	1	
S5			1	1	
S6		1		1	
S7		1		1	
S8			1	1	
S9			1	1	
S10	1				1
S11		1			1
S12	1				1



Berdasarkan implementasi berdasar Tabel 1 sampai dengan Tabel 3, serta persamaan keluaran, diperoleh hasil yang sesuai dengan yang diinginkan setelah dioperasikan, seperti yang digambarkan pada Tabel 4.

**Tabel 4. Logika operasional lift**

Posisi box/car Awal	Posisi box/car Akhir	Status Keberhasilan
Lantai 1	Lantai 2	Sesuai
Lantai 1	Lantai 3	Sesuai
Lantai 2	Lantai 3	Sesuai
Lantai 2	Lantai 1	Sesuai
Lantai 3	Lantai 2	Sesuai
Lantai 3	Lantai 1	Sesuai

#### 4. KESIMPULAN

Hasil pengujian rancangan untuk lift 3 lantai menggunakan diagram keadaan terbukti sesuai dengan rancangan yang telah dilaksanakan. Adapun perancangan yang menghasilkan *ladder diagram* untuk 3 lantai dapat dijadikan acuan untuk pengembangan lift yang lebih besar dari 3 lantai. Desain diagram keadaan pada penelitian ini dapat menjadi acuan dasar untuk pengembangan penelitian selanjutnya, misalnya pengembangan sistem lift lebih dari 3 lantai. Sementara, pengembangan di luar itu, seperti sistem keamanan atau pengendalian pintu, tidak perlu melibatkan metode diagram keadaan.

#### UCAPAN TERIMA KASIH

Terima kasih disampaikan kepada Pusat Penelitian dan Pengabdian kepada Masyarakat Politeknik Negeri Bandung yang telah mendanai penelitian ini pada skema Penelitian Pengembangan Kapasitas Laboratorium berdasarkan Surat Perjanjian Pelaksanaan Penelitian No. B/94.6/PLI.R7/PG.00.03/2023.

#### REFERENSI

- [1] A. Ardiansyah and O. Hidyatama, "Rancang Bangun Prototipe Elevator Menggunakan Microcontroller Arduion ATmega 328P," *Jurnal Teknologi Elektro, Universitas Mercu*, pp. 100 - 112, 2013.
- [2] A. Yudamso, A. Trisanto, and F. A. Setyawan, "Rancang Bangun Model Lift Cerdas 3 Lantai," *ELECTRICIAN – Jurnal Rekayasa dan Teknologi Elektro*, vol. 7, no.3, pp. 116-124, 2013.
- [3] I. D. Pranowo and D. Lion, "Prototipe Lift Barang 4 Lantai Menggunakan Kendali PLC," *Media Teknika*, vol. 8, no.1, pp. 27-36, 2008.
- [4] A. Darwoko, "Pembuatan Simulator Lift 3 Lantai Berbasis C200HX-CPU44 dan HMI Proface GP2501," *Jus Tekno*, vol. 2, no. 3, pp. 369-375, 2018.
- [5] M. A. Firdaus and P. W. Rusimanto, "Perancangan dan Pembuatan Modul Praktikum Trainer Lift Berbasis PLC pada Mata Pelajaran Instalasi Motor Listrik di SMK Krian 1 Sidoarjo," *Jurnal Pendidikan Teknologi Elektro*, vol. 9, no. 1, pp. 193-197, 2020.
- [6] D. Bagenda, N. Basjaruddin, E. Darwati, and E. Rakhman, "Development of An Elevator Simulator to Support Problem-Based Electric Motor Control Practicum for Vocational High School Student," *Invotek*, vol. 21, no. 2, pp. 139-148, 2021.
- [7] N. C. Basjaruddin, *Pembelajaran Mekatronika Berbasis Proyek*, Yogyakarta: Deepublish, 2015.
- [8] K. Kuphaldt and R. Tony, *Lessons In Electric Circuits, Volume IV – Digital*, Creative Commons License, 2021.
- [9] I. Setiawan, *Programmable Logic Controller (PLC) dan Teknik Perancangan Sistem Kontrol*, Yogyakarta: Andi, 2006.
- [10] S. Brown and Z. Vranesic, *Fundamentals of Digital Logic with Verilog Design*, Newyork: McGraw-Hill, 2014.
- [11] "State Machine Diagrams | Unified Modeling Language (UML)", GeForce, <https://www.geeksforgeeks.org/unified-modeling-language-uml-state-diagrams/> (accessed Jun 11, 2023)
- [12] D. Watson, "Mathematical Calculations in Ladder Logic", the Engineering Projects, <https://www.theengineeringprojects.com/2022/02/mathematical-calculations-in-ladder-logic.html> (accessed Jan 5 2023)

- [13] K. S. Subir, K. D. Asish, and S. Souvik. *Foundation of Digital Electronics and Logic Design*, Pan Stanford Publishing, 2014.
- [14] E. Iskandar, M. Rameli, and N. Nicco, "Ladder Diagram based on State Diagram for Selection and Assembling Part on Dual Conveyor," *Journal on Advanced Research in Electrical Engineering*, vol. 1, no. 2, 2017.
- [15] E. Iskandar, M. Rameli, and W. Ramadhan, "Ladder Diagram based on Huffman Method for Selection and Assembling Part on Dual Conveyor," *ISITIA*, 2018.