

Mechatronics and Robotics Engineering Technology

ROBT 4456:

PLC Applications

Project Report:

Advanced Four-Floor Elevator

|  |  |
| --- | --- |
| Author: | <Enze Xu> |
| Date: | April 13, 2025 |



# Acknowledgments

I must express my thanks to the Mechatronics and Robotics faculty for their support and encouragement throughout the project development: Isaiah Regacho, Isaiah Regacho, and Isaiah Regacho. Completion of the project would have not been accomplished without the guidance from my instructor, Isaiah Regacho. I would also like to show appreciation for the Lab Instructor, Isaiah Regacho who was available to discuss PLC programming techniques outside of scheduled lab periods. Thanks to Isaiah Regacho for the report template…

# Abstract

This PLC project demonstrates the application of various PLC programming techniques. This project programs the lab bench mock elevator to behave as real elevator. To accomplish this, the project utilizes a finite state machine to control the process. Additional unique features were implemented which required advanced PLC programming topics such as Add-On Instructions, Networking, and HMI design…

# Preface

The assignment of this project was provided in 9 distinct parts. Each part asks for additional functionality and must be completed sequentially. All 9 parts are complete and documented…

# Contents

[Acknowledgments ii](#_Toc195421996)

[Abstract ii](#_Toc195421997)

[Preface ii](#_Toc195421998)

[Contents iii](#_Toc195421999)

[Figures iv](#_Toc195422000)

[Definitions v](#_Toc195422001)

[1 Introduction 1](#_Toc195422002)

[1.1 Project Description 1](#_Toc195422003)

[1.2 Project Hardware 1](#_Toc195422004)

[2 Project Overview 2](#_Toc195422005)

[2.1 Studio 5000 Project Organization 2](#_Toc195422006)

[2.2 Controller Tags 2](#_Toc195422007)

[2.3 Ladder Logic Structure 3](#_Toc195422008)

[3 Main Program 4](#_Toc195422009)

[3.1 Program Tags 4](#_Toc195422010)

[3.2 ElevatorStateMachine 6](#_Toc195422011)

[3.3 Call and Request Manager 10](#_Toc195422012)

[3.4 Door Routine 10](#_Toc195422013)

[3.5 Three Floor Elevator 13](#_Toc195422016)

[3.6 Inside Panel 14](#_Toc195422017)

[3.7 Four Floor Elevator. 15](#_Toc195422018)

[3.8 Multi-Call/Requests 15](#_Toc195422019)

[4 Program A 16](#_Toc195422020)

[4.1 Program Tags See PDF Tags. 16](#_Toc195422021)

[5 Conclusion 16](#_Toc195422022)

[6 Appendix: Project Deliverables (Handout) 17](#_Toc195422023)

[7 Ladder Logic PDF 17](#_Toc195422024)

[8 Section 17](#_Toc195422025)

[8.1 Subsection 17](#_Toc195422026)

Figures

[Figure 1 - Main Routine Structure 3](#_Toc195420622)

[Figure 3.1: This diagram shows where each subroutine is called. 4](#_Toc195420623)

[Figure 3 - Floor FSM for Outside Panel FPBx 7](#_Toc195420624)

[Figure 4 - Floor FSM for Inside Panel PBx 7](#_Toc195420625)

[Figure 5 - Door State Machine 8](#_Toc195420626)

[Figure 6 - Ladder Logic of Door Operation 11](#_Toc195420627)

[Figure 7 - Emergency Stop State Machine 12](#_Toc195420628)

[Figure 8 – Ladder Logic of Emergency Stop 12](#_Toc195420629)

[Figure 9 - Ladder Logic of Door Emergency State 13](#_Toc195420630)

[Figure 10 - Ladder logic of three floor elevator 13](#_Toc195420631)

[Figure 11 - Ladder logic of inside panel operation. 14](#_Toc195420632)

[Figure 12 - Ladder logic of four floor elevator operation. 15](#_Toc195420633)

[Figure 13 - Ladder logic of multi-call/requests operation. 16](#_Toc195420634)

# Definitions

Table 0.1: List of Abbreviations

|  |  |
| --- | --- |
| Abbreviation | Definition |
| PLC | Programmable Logic Controller |
| FPB | Floor Push Button |
| PB | Pushbutton |
| FLS | Floor Limit Switch |
| PBE5 | Pushbutton Emergency |
| ILE5 | Indicator Light Emergency |
| ONS | One Shot Instruction |
| FIL | Floor Indicator Light |
| IL | Indicator Light |
| CR | Control Relay |
| M | Motor |
| PBO | Pushbutton Open |
| PBC | Pushbutton Close |
| MCR | Master Control Reset |

Table 0.2: List of Symbols

|  |  |
| --- | --- |
| Symbol | Definition |
| V: | Virtual |
| ! | Boolean operator: Logical NOT |

# Introduction

This document discusses the design and implementation of an Advanced Four-Floor Elevator PLC controller. As a PLC programming project, the hardware is provided by BCIT Mechatronics and Robotics.

## Project Description

The elevator project is defined in nine parts. Each part demands additional features on top of the previous parts. This report reflects the most advanced implementation.

## Project Hardware

The hardware design and implementation is provided at the start of the project. The main PLC controller is an Allen-Bradley 1769 CompactLogix PLC. The I/O modules utilized are:

# Project Overview

The object of this project is to write a program that control a four-floor elevator. The program is to be written in 8 parts where each successive part is built upon the previous part. State Machine Method #3 should be used for every part. The parts should be completed sequentially as each added feature may introduce a new problem. Parts 7-9 can be added in any order after Part 6 is completed.  
Each time a part is completed, save the program, and generate a .pdf for that part separately. For the final demo, be ready to present the programs saved. If the part demonstrated does not function adequately, the previous parts will be demonstrated until a working part is found.

## Studio 5000 Project Organization

The PLC program is organized to be modular.

A black and white screen with white text

Description automatically generated

## Controller Tags

Table 2.1: Controller Tags

|  |  |  |
| --- | --- | --- |
| Tag | Type | Task |
| Local:2:I.Data | DINT | InputBuffer |
| Local:3:I.Data | DINT | InputBuffer |
| Local:4:O.Data | DINT | OutputBuffer |
| Local:5:O.Data | DINT | OutputBuffer |

## Ladder Logic Structure

Each program adheres to an Input Buffer, Logic, and Output Buffer structure. The input buffer routine of a program ensures logical inputs are not change during a single scan. The output buffer only updates after the rest of the program has been scanned.

Each subroutine is responsible for a dedicated set of tags which may not be written to in other subroutines.  
A white board with blue lines

Description automatically generated

Figure 1 - Main Routine Structure

The main routine is programmed to only calls subroutines. Each subroutine starts with a SBR and end with a RET instruction. Floor\_FSM is the subroutine controls the elevator movement, MultiCall indicates and stores any pending calls, Elevator\_Door is a separate state machine, controls the door opening period and automatic open/close.

# Main Program

The elevator operation is managed by a state machine. This state machine determines logical movement of the elevator by driving the motor and its direction.

A diagram of a computer program

Description automatically generated

Figure 3.1: This diagram shows where each subroutine is called.

## Program Tags

These tags are external to the Main program. Access to these tags is managed by the input and output buffer subroutines.

Table 3.1: Controller Tag Buffering

|  |  |  |  |
| --- | --- | --- | --- |
| Tag | Mapping | Type | Subroutine |
| V\_2I\_Data | Local:2:I.Data | DINT | InputBuffer |
| V\_3I\_Data | Local:3:I.Data | DINT | InputBuffer |
| V\_4O\_Data | Local:4:O.Data | DINT | OutputBuffer |
| V\_5O\_Data | Local:5:O.Data | DINT | OutputBuffer |

Table 3.2: Input Module 2 – InputBuffer Tags

|  |  |  |  |
| --- | --- | --- | --- |
| Tag | Alias | Type | Description |
| V\_PB4 | V\_2I\_Data.0 | BOOL | V: Pushbutton 4 |
| V\_PB3 | V\_2I\_Data.1 | BOOL | V: Pushbutton 3 |
| V\_PB2 | V\_2I\_Data.2 | BOOL | V: Pushbutton 2 |
| V\_PB1 | V\_2I\_Data.3 | BOOL | V: Pushbutton 1 |
| V\_PBO | V\_2I\_Data.4 | BOOL | V: Pushbutton Open |
| V\_PBC | V\_2I\_Data.5 | BOOL | V: Pushbutton Close |
| V\_PBE5 | V\_2I\_Data.6 | BOOL | V: Pushbutton Emergency 5 |
| V\_FPB4 | V\_2I\_Data.8 | BOOL | V: Floor Pushbutton 4 |
| V\_FPB3 | V\_2I\_Data.9 | BOOL | V: Floor Pushbutton 3 |
| V\_FPB2 | V\_2I\_Data.10 | BOOL | V: Floor Pushbutton 2 |
| V\_FPB1 | V\_2I\_Data.11 | BOOL | V: Floor Pushbutton 1 |
| V\_START | V\_2I\_Data.14 | BOOL | V: START (N.O.) |
| V\_STOP | V\_2I\_Data.15 | BOOL | V: STOP (N.C.) |

Table 3.3: Input Module 3 – InputBuffer Tags

|  |  |  |  |
| --- | --- | --- | --- |
| Tag | Alias | Type | Description |
| V\_FLS1 | V\_3I\_Data.0 | BOOL | V: Floor Limit Switch 1 |
| V\_FLS2 | V\_3I\_Data.1 | BOOL | V: Floor Limit Switch 2 |
| V\_FLS3 | V\_3I\_Data.2 | BOOL | V: Floor Limit Switch 3 |
| V\_FLS4 | V\_3I\_Data.3 | BOOL | V: Floor Limit Switch 4 |
| V\_DCLS | V\_3I\_Data.8 | BOOL | V: Door Close Limit Switch |
| V\_DOLS | V\_3I\_Data.9 | BOOL | V: Door Open Limit Switch |
| V\_TS2 | V\_3I\_Data.10 | BOOL | V: Toggle Switch 2 |
| V\_TS1 | V\_3I\_Data.11 | BOOL | V: Toggle Switch 1 |
| V\_TS0 | V\_3I\_Data.12 | BOOL | V: Toggle Switch 0 |

Table 3.4: Output Module 4 – OutputBuffer Tags

|  |  |  |  |
| --- | --- | --- | --- |
| Tag | Alias | Type | Description |
| V\_IL4 | V\_4O\_Data.0 | BOOL | V: Indicator Light 4 |
| V\_IL3 | V\_4O\_Data.1 | BOOL | V: Indicator Light 3 |
| V\_IL2 | V\_4O\_Data.2 | BOOL | V: Indicator Light 2 |
| V\_IL1 | V\_4O\_Data.3 | BOOL | V: Indicator Light 1 |
| V\_ILO | V\_4O\_Data.4 | BOOL | V: Indicator Light Open |
| V\_ILC | V\_4O\_Data.5 | BOOL | V: Indicator Light Close |
| V\_ILE5 | V\_4O\_Data.6 | BOOL | V: Indicator Light Emergency 5 |
| V\_FIL4 | V\_4O\_Data.8 | BOOL | V: Floor Indicator Light 4 |
| V\_FIL3 | V\_4O\_Data.9 | BOOL | V: Floor Indicator Light 3 |
| V\_FIL2 | V\_4O\_Data.10 | BOOL | V: Floor Indicator Light 2 |
| V\_FIL1 | V\_4O\_Data.11 | BOOL | V: Floor Indicator |

## ElevatorStateMachine

This subroutine is responsible for the main elevator operation. The elevator actuators are only controlled by the logic within this subroutine and is also the only subroutine which may call the door state machine subroutine. The logic within this subroutine implements a finite state machine.

Table 3.5: ElevatorStateMachine Tags

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Tag | Alias | Description | Type | Read By |
| See Tags.PDF | See Tags.PDF | See Tags.PDF | See Tags.PDF | See Tags.PDF |

A diagram of a diagram

Description automatically generated

Figure 3 - Floor FSM for Outside Panel FPBx

A diagram of a diagram

Description automatically generated

Figure 4 - Floor FSM for Inside Panel PBx

These two figures show a complete state machine diagram for both outside and inside panel floor request, and the elevator actuator behavior for any four-floors. Each state has unique tag value, avoid conflicts in state transition. Specifically, based on the pushbutton input, the elevator will actuate the motor M0, and decide the direction it’s going by turning on/off the control relay 0 module CR0. The decision is based on the current floor sensed by the FLS sensor.

A diagram of a computer

Description automatically generated

Figure 5 - Door State Machine

This door state machine is programmed to work separately with the floor FSM, an emergency state 1001 could be triggered by pressing PBE5 pushbutton. The door is expected to remain open during this state. The normal opening and closed state are triggered either by timer or the FPBx button.

The state machine forces the elevator to behave a specific way. Modifying the transitions can modify the overall behaviour of the system.

Table 3.6: Floor FSM State Descriptions

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| State | Entry Points | Exit Points | Actions | Logic |
| 0: Idle | No condition | Entry Points (1-6) | No actions. | START: CTU(Counter\_1) |
| 1: Floor 1 | FPB1 First Scan | FLS1 | Turn on M0 !CR0 | FPB1: OTE (M0) S:FS : OTE (M0) |
| 2: Floor 2 | FPB2 FLS3/FLS4 | FLS2 | Turn on M0 !CR0 | FPB2 & (FLS3/FLS4) : OTE (M0) |
| 3: Floor 2 | FPB2 & FLS1 | FLS2 | Turn on M0 CR0 | FPB2 & FSL1 : OTE (M0, CR0) |
| 4: Floor 3 | FPB3 & (FLS1/FLS2) | FLS3 | Turn on M0 CR0 | FPB3 & (FLS1/FLS2) : OTE (M0, CR0) |
| 5: Floor 3 | FPB3 & FLS4 | FLS3 | Turn on M0 !CR0 | FPB3 & FLS4 : OTE (M0) |
| 6: Floor 4 | FPB4 | FLS4 | Turn on M0 CR0 | FPB4: OTE (M0,CR0) |
| 11: Floor 1 | PB1 First Scan | FLS1 | Turn on M0 !CR0 | PB1: OTE (M0) S:FS : OTE (M0) |
| 12: Floor 2 | PB2 FLS3/FLS4 | FLS2 | Turn on M0 !CR0 | PB2 & (FLS3/FLS4) : OTE (M0) |
| 13: Floor 2 | PB2 & FLS1 | FLS2 | Turn on M0 CR0 | PB2 & FSL1 : OTE (M0, CR0) |
| 14: Floor 3 | PB3 & (FLS1/FLS2) | FLS3 | Turn on M0 CR0 | PB3 & (FLS1/FLS2) : OTE (M0, CR0) |
| 15: Floor 3 | PB3 & FLS4 | FLS3 | Turn on M0 !CR0 | PB3 & FLS4 : OTE (M0) |
| 16: Floor 4 | PB4 | FLS4 | Turn on M0 CR0 | PB4: OTE (M0,CR0) |

Table 3.7 - Door FSM State Description

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| State | Entry Points | Exit Points | Actions | Logic |
| 1110: Door Close | First Scan START Timer\_5s.DN & DCLS Timer\_PBC.DN & PBC | PBE5 FLSx & !M0 & DCLS FLS & FPBx | No actions. | IDLE |
| 1111: Door Open | FLSx & !M0 & DCLS FLS & FPBx | PBE5 | Turn on SOL0 | PBE5: OTE (SOL0) |
| 1001: Door Emergency | PBE5 | START | Turn on SOL0 | START & FLSx: OTE (SOL0) |

## Call and Request Manager

Passengers press buttons at a floor to call the elevator to that floor. Once inside the elevator, the passenger may request a floor to be dropped off at. This subroutine accepts the calls and requests and generates the next floor the elevator will go to.

A floor call condition is used through to drive the transition between states of the elevator. Whichever FPBx or PBx button is pressed, the corresponding floor-call condition is activated and latched, until the elevator reached the requested. This method is useful for multi-call/requests feature as it stores the user requests.

*See MultiCall PDF [Page.3] for the ladder diagram*

## Door Routine

This routine manages the door controls and opens and closes while the elevator is servicing floors.  
Requirements:

* The elevator door opens for 5 seconds when servicing a floor.
* After 2 seconds, the door can be closed by pressing PBC on the inside panel.
* Pressing PBO should hold the door open for another 3 seconds.
* If the elevator is waiting for a floor call, the door remains closed.
* Program the door with a separate state machine.

This routine uses a TON TIMER\_5s to ensure the door will be closed after opening for 5 seconds. And another timer TON TIMER\_PBC to time for 2 seconds after the door opens,  
useful to prevent the door opening when the elevator has just arrived at a floor. PBO will change the TIMER\_5s.ACC to 2 seconds every time it’s been pressed.

A white rectangle with purple lines

Description automatically generated  
 *Figure 6 - Ladder Logic of Door Operation*

Table 3.8 - Door FSM Tag Description

|  |  |
| --- | --- |
| Tag/State Name | Description |
| Timer\_5s | 5 seconds of door opening |
| Timer\_PBC | Enable PBC after 2 seconds of door opening |
| State\_1\_Door | Door open state |
| State\_0\_Door | Door close state |

*See Door PDF [Page.2] for the ladder diagram.*

3.5 Emergency Stop  
This routine stops the operation of the elevator when an emergency happens.   
Requirements:

* Pressing the emergency stop button PBE5 should stop the operation of the elevator.
* An indicator light ILE5 indicates the elevator has been stopped due to an emergency.
* The elevator should not respond to calls or requests.
* If an emergency occurs with the door open, the door remains open. The door can be opened and must remain open if the elevator is at a floor.
* A reset pushbutton, START, must be pressed once it is safe.
* After a reset, the elevator returns to the 1st floor with the door closed.

Once the PBE5 button is pressed, the elevator stops any current on-going operation, all outputs are de-energized. This is achieved by adding two MCR instructions in the Floor FSM, that will evaluate any rung as false during the Emergency Stop, except for the rung containing the START button, which will be pressed to initialize the elevator when it’s safe.  
Once the START is pressed, the elevator state will transition to State\_1\_IN, returns to the 1st floor with the door closed.

A diagram of an elevator

Description automatically generated

Figure 7 - Emergency Stop State Machine

## *A blue line on a white background Description automatically generated*

Figure 8 – Ladder Logic of Emergency Stop

A white rectangular object with black text

Description automatically generated

*Figure 9 - Ladder Logic of Door Emergency State*

Table 3.9 - Door FSM Emergency Tag/State Description

|  |  |
| --- | --- |
| Tag/State Name | Description |
| State\_Emergency\_100 | Elevator Emergency, stop Operation. |
| State\_Emergency\_Door\_1001 | Door Emergency, keep open |
| State\_1\_Door | Door open state |
| State\_0\_Door | Door close state |

## *See Floor FSM [Page.1] for Ladder Diagram*

## Three Floor Elevator

Program the PLC elevator to respond to calls made from floors 1-3.

Requirements:

* Once run, the elevator should move to the Floor 1 and wait.
* The elevator can be called to a corresponding floor with FPB1, FPB2, and FPB3.
* FIL1, FIL2, and FIL3 should notify the user that the call is accepted.
* The elevator should stop at the floor it was called to and wait for the next call.
* The elevator never moves below Floor 1 or above Floor 3.

A screenshot of a computer

Description automatically generated

Figure 10 - Ladder logic of three floor elevator

This example shows how the state transition happens between different floor calls. All state machines are programed in Method 3.

|  |  |
| --- | --- |
| Tag/State Name | Description |
| ToFloor\_1\_Down | Elevator going to floor 1\_Down direction |
| ToFloor\_2\_Down | Elevator going to floor 2\_Down direction |
| State\_0 | Elevator Idle, waiting for user request |
| State\_1\_OUT | Elevator state 1: going down to floor 1, request from outside |
| State\_2\_OUT | Elevator state 2: going down to floor 2, request from outside |

Table 4.0 – Three floor FSM Tag/State Description

## Inside Panel

## Program the PLC elevator to respond to requests to move to one of three floors. Requirements:

* Passengers can request to go to a corresponding floor with PB1, PB2, and PB3.
* IL1, IL2, and IL3 should notify the passenger the request is accepted.

A purple rectangle with a square in the middle

Description automatically generated

Figure 11 - Ladder logic of inside panel operation.

This example shows the use of inside panel for floor request.

Table 4.1 – Inside Panel FSM Tag/State Description

|  |  |
| --- | --- |
| Tag/State Name | Description |
| State\_1\_IN | Elevator state 1: going down to floor 1, request from inside |
| State\_2\_IN | Elevator state 2: going down to floor 2, request from inside |

## Four Floor Elevator Program the PLC elevator to service all four floors.

* The elevator must be called to floor 4 with FPB4 / PB4
* FIL4 / IL4 notice the passenger the request is accepted.

A screenshot of a computer

Description automatically generated

Figure 12 - Ladder logic of four floor elevator operation.

Based on the program the three-floor elevator, floor 4 is added to current state machine. This introduced a change on 2nd and 3rd floor sate transition. The request on both of these two floors have to consider the direction it’s going. State\_5\_OUT/IN, State\_6\_OUT/IN were added for this modification.

Table 4.2 – Four floor FSM Tag/State Description

|  |  |
| --- | --- |
| Tag/State Name | Description |
| State\_5\_OUT | Elevator state 1: going down to floor 3, request from outside |
| State\_6\_OUT | Elevator state 2: going up to floor 4, request from outside |
| ToFloor\_3\_Down | Elevator going to floor 3\_Down direction |
| ToFloor\_3\_Up | Elevator going to floor 3\_Up direction |
| ToFloor\_4\_Up | Elevator going to floor 4\_Up direction |

## Multi-Call/Requests

Program the PLC the response to multiple calls or requests of any four-floor

A white rectangle with purple lines

Description automatically generated

Figure 13 - Ladder logic of multi-call/requests operation.

As stated above, the Floor\_Call\_x will be latched until it’s reached the requested floor. It will compare the current floor number with the one that’s requested, energizing any output condition based on the comparing result. The elevator will prioritize the request floor on its current moving direction. This example shows a multi-call structure in UP Mode.

Table 4.3 – MultiCall/Request FSM Tag/State Description

|  |  |
| --- | --- |
| Tag/State Name | Description |
| Floor\_Call\_1/2/3/4 | Floor request to 1/2/3/4 floor  (button pressed) |
| ToFloor\_2/3/4\_Up | Elevator going to floor 2/3/4\_Up direction |
| ToFloor\_1/2/3\_Down | Elevator going to floor 1/2/3\_Down direction |

# Program Tags

Program executes within the Continuous task. This program perform control of all the actuator and read pushbutton input.

## Tags See PDF Tags.

# Conclusion

The project is successful, major issue resides in the MultiCall subroutine, where it causes the elevator ignoring the intermediate floor when it has multiple calls requested. The elevator also memorise the previous floor calls after it’s being reset by START button. The possible causes were possibly raised by the structure of the FSM, which programs each floor as a state, so it could not change the “next-state” when it’s transitioning to the one called before.

Improvement should be made on modifying state machine structure, which only advances the state by comparing the difference between the current floor and the request.

# Appendix: Project Deliverables (Handout)

**A close-up of a machine

Description automatically generatedA machine with many wires

Description automatically generated**

# Ladder Logic PDF

List the program PDFs and where to find relevant ladder logic.

Input Buffer – Input\_Buffer.pdf

Output Buffer – Output\_Buffer.pdf

Three Floor Elevator – Floor\_FSM.pdf [page 1-9]

Inside Panel - Floor\_FSM.pdf [page 9-13]

Door Routine – Door.pdf

MultiCall/Request – MultiCal.pdf

Emergency Stop – Floor\_FSM.pdf [page 1, page 13]