A Glimpse of the Inner Cogs of Elevators

It’s not just a box with ropes

Part B

Dedications

To the people who chose UTM and low-key regret it but lower-key don't

Acknowledgements

Dr. Nor Shahida Hasan for your guidance

Table of contents:

Part B

The Idea

What does an elevator do? Anybody can answer that. But it gets difficult if we ask you to answer that in a way that machines can understand. If we want to accomplish that, first we must ask what machines understand. Which of these understandings are relevant to an elevator?

Machines know how to compare two numbers, and how to count. Turns out you can explain elevators with only those two functions. When you press a button on an elevator, it tells the machine to start counting up. It keeps counting and comparing the count with your signal number.

When they match, the counting stops, i.e. the elevator stops.

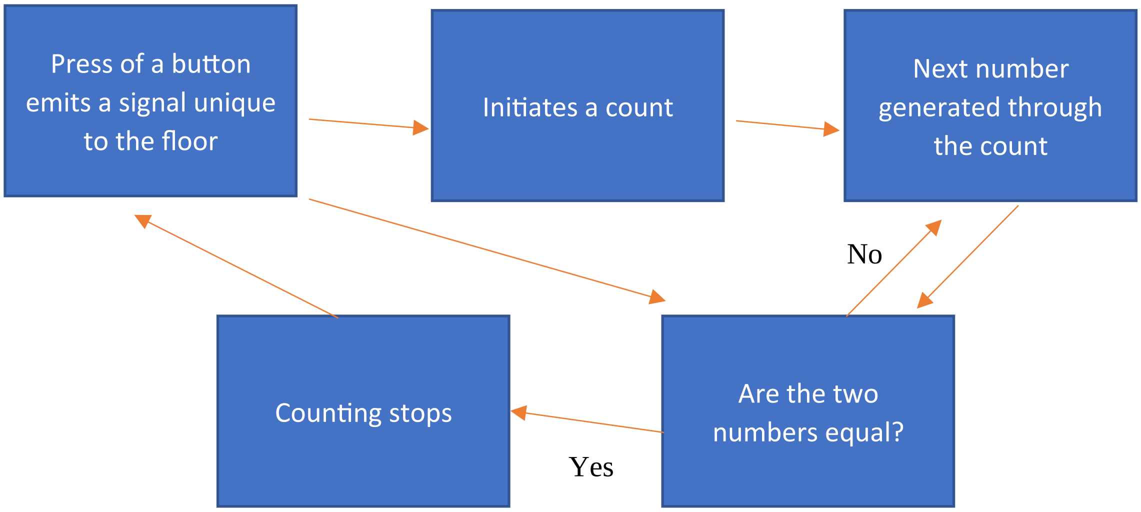

Press of a button emits a signal unique to the floor
Initiates a count
Next number generated through the count
No
Counting stops
Are the two numbers equal?
Yes

Fig: Block diagram of the elevator's working procedure

The Components

We use all the components we used for part A. Below are the new ones used in this part

Up/Down Counter: We use an up/down counter based on D-flipflops for this design. It can count up from a number and count down from any number too.

Multiplexer: Multiplexers are analogous to data filters. It helps us decide which signal through and change which gets passed from time to time.

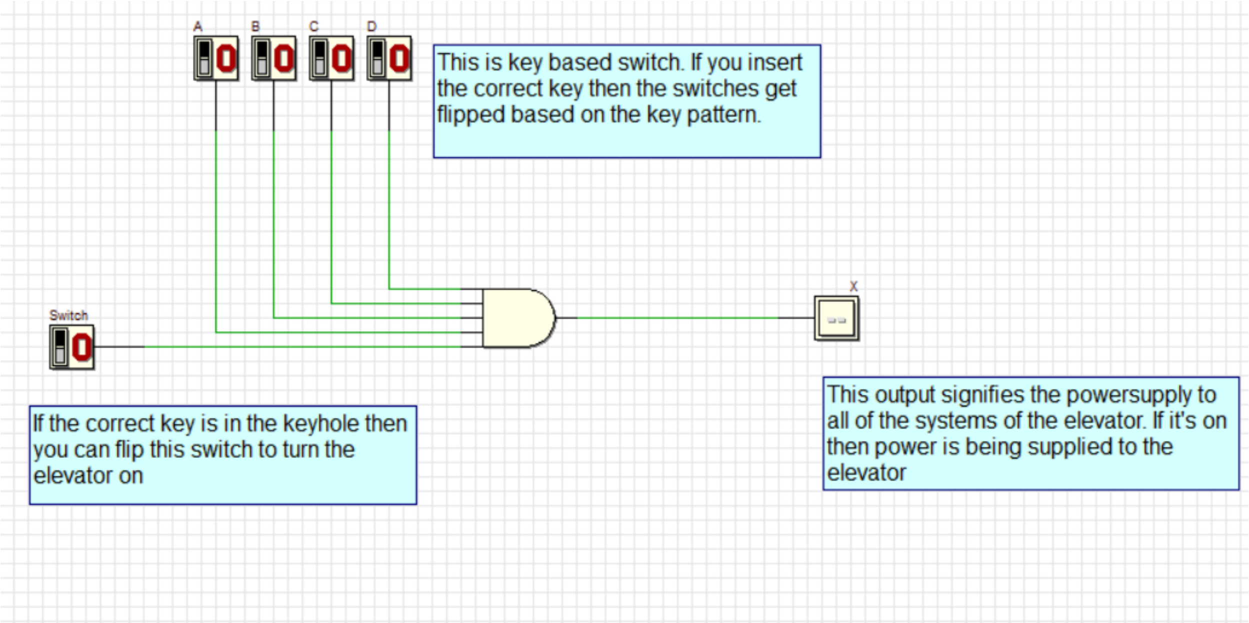
S-R Latch: S-R latch is component used to save one bit of information. The saved bit does not change if the inputs are not given in specific combinations.

The Design

The design section is split into the following sections according to features-

- Authentication
- Call receiving
- Call following and ignoring
- Emergency
- Resulting simulation

Authentication



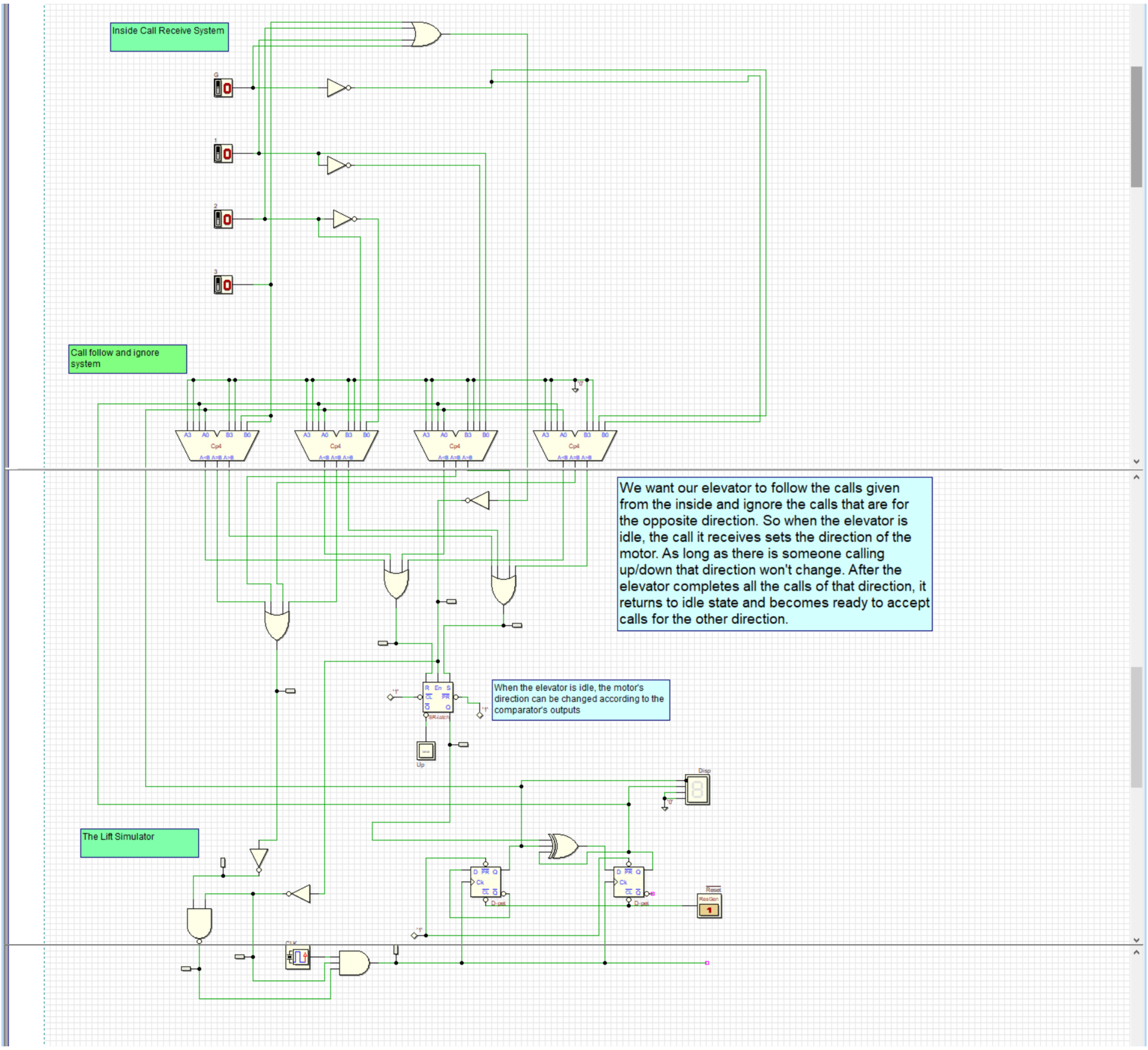
This is key based switch. If you insert the correct key then the switches get flipped based on the key pattern.

If the correct key is in the keyhole then you can flip this switch to turn the elevator on

This output signifies the powersupply to all of the systems of the elevator. If it's on then power is being supplied to the elevator

Fig: Authentication circuit

The given diagram is a simple circuit to authenticate launch. Four outputs of 5 input AND gate come from switches inside a keyhole. So, if the correct key is entered, it presses all the keys. After that you can turn the elevator on by pressing its switch.



We want our elevator to follow the calls given from the inside and ignore the calls that are for the opposite direction. So when the elevator is idle, the call it receives sets the direction of the motor. As long as there is someone calling up/down that direction won't change. After the elevator completes all the calls of that direction, it returns to idle state and becomes ready to accept calls for the other direction.

When the elevator is idle, the motor's direction can be changed according to the comparator's outputs

Fig: Elevator Control System

Call Receiving

This part is simple as well. As we are simulating the inside system of the elevator, we only need buttons for the floors. So, we have four buttons- G, 1, 2, 3- and combination of NOT gates to produce four unique two-line combinations for each of the buttons. The press of a button inputs the call into the next part.

Call Following and Ignoring

Now this is where things get interesting. We want the elevator to follow the calls from the inside, but we want it to ignore the calls that or for the other side, i.e., if the lift is going down, we only want it to accept calls below. So, we have an S-R latch to store the state of the lift (up or down). This state can only be rewritten when the elevator is idle after completing all the calls of one direction.

That summarizes the ignoring system. We also want our elevator to follow calls and stop at the right floors. For that purpose, we have four comparators. If the comparators produce and equality signal, a stop signal shuts down the clock of the elevator stopping it.

These comparators also contribute to the ignoring system. They detect if someone is calling up or down. The state storing S-R latch's output is dependent on the comparators.

Emergency

This is another simple part. If the emergency button is pressed inside the elevator, it ignores all calls from the outside system

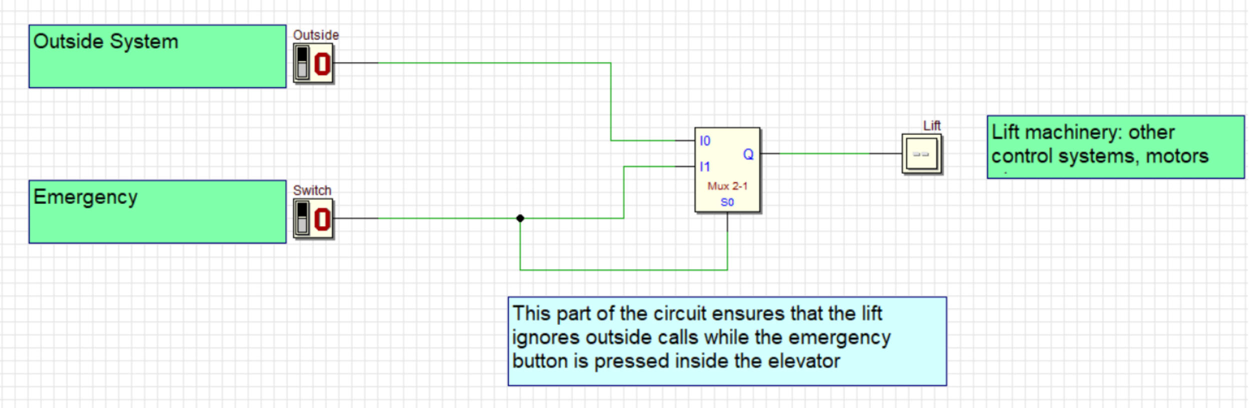

Outside System
Outside
0
I0
I1
Q
Mux 2-1
S0
Lift
Lift machinery: other control systems, motors
Emergency
Switch
0
This part of the circuit ensures that the lift ignores outside calls while the emergency button is pressed inside the elevator

Fig: Emergency Response System

Resulting Simulation

Finally, we have the most beautiful part of the elevator: an up/down counter. This synchronous up/down counter was designed using D flip-flops. The clock stops, counts up or down based on the inputs from the call follow system. The position number produced by the counter is sent as input to the call follow system so that comparators can compare that value with the desired floors value.

Limitations

There are two key limitations in the design. First one is the absence of a Comparator EN. When no button is pressed, a comparator still takes in the output produced by the 0s of the button. We want our comparators to stop making decisions when buttons are not pressed but we are unable to do that because we do not have Comparator EN in the production software.

Secondly, we want a call button to be reset when the elevator has stopped at the floor. We have not been able to implement that owing to lack of components.

These limitations have resulted in a lack of function in the state storing S-R latch. Thus, it is impossible to control the elevator after full integration. For demo purposes we can implement it using input switches where required.

Attempts have been made to bypass the Comparator EN but we have not accomplished to solve the latch problem.

Expanding to the Outside

Our simulation only shows the workings of the inside system. If we wish to expand it to accommodate outside call, we will not have to make any major changes. We will have to get 6 switches (two buttons- up, down- for floor 1 and 2). And we have to add all up switches with the up-state output and down switches with down state. Then get enough comparators to make decisions. So, in short, we just have to do everything we did here, once more.

Conclusion

This project has pushed us to think outside the box. We started from a counter that counted up and stopped on command and went on to a total elevator. For us, this project would go on to be a highlight of the year. Also, this project would be more than a memory as it has familiarized us with professional communication, teamwork, and of course topics of electrical engineering. We do regret that we were not able to get our elevator to the fullest of its glory, but we did the best within our capabilities, and it was a great experience to be able to do so on a creative project like this.

References and Further Readings

- Digital Fundamentals – Eighth Edition – Thomas L. Floyd
- Digital Logic Module – Fifth Edition – School of Computing, Faculty of Engineering, Universiti Teknologi Malaysia
- New Designs Going Up-Working Knowledge on Elevators by Mark Fischetti https://www.scientificamerican.com/article/how-do-elevators-work/

Appendices

i) List of members and responsibilities

Adrian Norman Bin Khusairi
A21MJ5028

Report writing, Brainstorming, and Reviews

Siam Rahman
A21MJ0143

Designing

Saiful Islam
A21MJ0142

Designing

Ahmad Hafiz Ali Bin Hassan Ali
A21MJ5012

Report Writing, Brainstorming, and Reviews

Farzad Khan Chowdhury
A21MJ0138

Report writing, Brainstorming and Reviews