



UPPSALA
UNIVERSITET

PROJECT: Elevator Model using Simulink
Model-Based Design of Embedded Systems

By Asif Mohamed & Vishnu Ullas

TABLE OF CONTENT

Introduction	3
Configuration & Settings.....	3
Driver Blocks	4
Simulink Model of Elevator.....	4
Calibration & Testing	7

INTRODUCTION

The project consists of an Elevator model that runs on STM3220G-EVAL Board with one motor and an Ultrasonic sensor. The physical model is already provided to us thus we need to build a Simulink Stateflow chart to generate the code and upload it with Keil.

CONFIGURATION & SETTINGS

Firstly, we need to configure the necessary pins that we require on the STM3220G-EVAL board using STM32CubeMX. We have saved the file as *New.ioc*. The following are the general configuration of the pins.

Description	Cube Config	Pin Out
Floor button 1	GPIO_Input	PC1
Floor button 2	GPIO_Input	PH15
Floor button 3	GPIO_Input	PH13
Floor button 4	GPIO_Input	PC9
Emergency button	GPIO_Input	PA8
LED Emergency	GPIO_Output	PH14
Ultrasonic Sensor – Echo pin	TIM8_CH3	PC8
Ultrasonic Sensor – Trigger pin	TIM8_CH1	PC6
Motor Enable Channel	TIM8_CH2	PA1
Motor channel 1 input	GPIO_Output	PC2
Motor channel 2 input	GPIO_Output	PC12

Additionally, we need to manually configure the required timers.

For TIM8, we need to set the Prescaler to 96-1 to drive the timer by the 96Mhz clock in the evaluation board; The counter period needs to be 25000-1; Under PWM generation for channel 1, we need to set the Pulse to 20; We need to enable the interrupt at capture too. This setting is mostly for the Ultrasonic sensor to read time between the echoes.

For TIM2, we set the Prescaler to 96-1 and counter period to 10000-1; Under PWM gen for channel 2, we set the pulse to 20. This setting is for the Motor to send PWM pulses to drive itself.

After making all the settings, we generate the code in CubeMX.



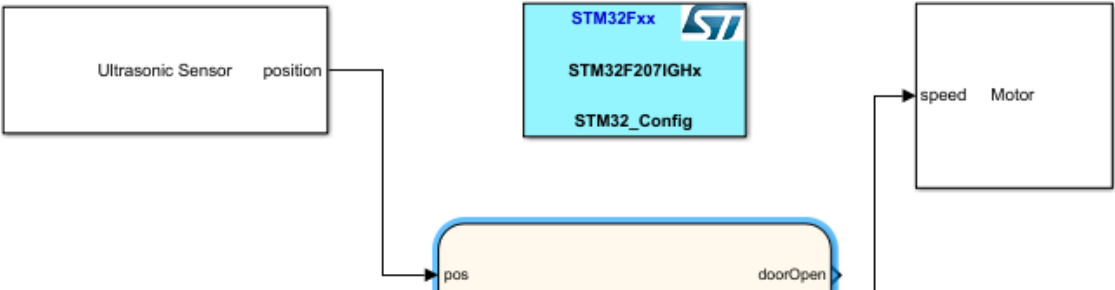
DRIVER BLOCKS

- Ultrasonic Driver

The driver reads the values of TIM8 when the echo is sent and received. The difference of this timing is used to calculate the distance. Since the sensor is fixed at the bottom, it basically gives the distance from the sensor to the elevator itself. This is inputted into our Simulink model.

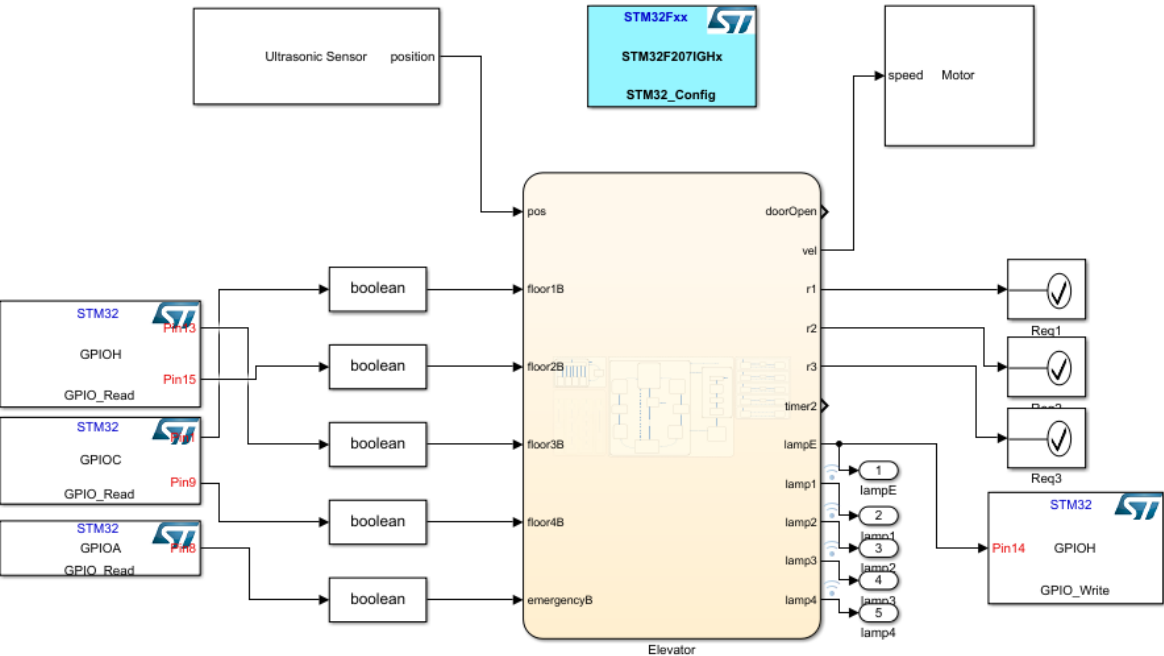
- Motor Driver

This driver takes input from the model and that value is used to calculate the duty cycle of the PWM that enables the motor to rotate clockwise and anti-clockwise.



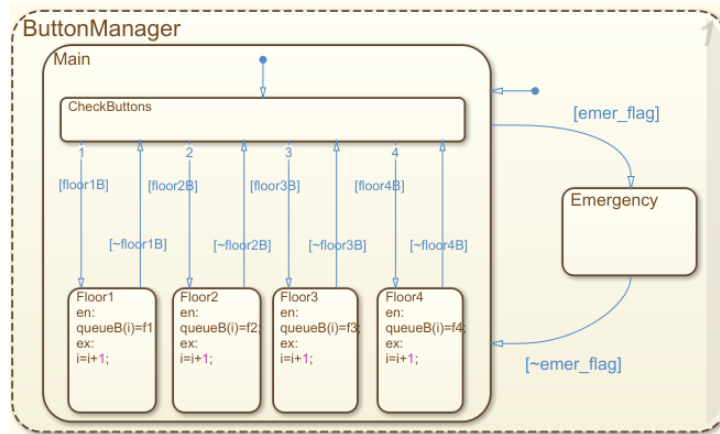
SIMULINK MODEL OF ELEVATOR

The model consists of 4 main states in the Stateflow chart that runs parallelly.



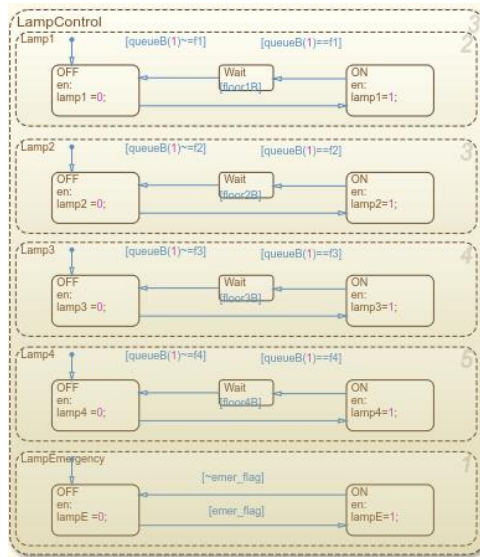
- Button Manager

Using flags, we check for button presses and add the specific floor number into the button queue that we created.



- Lamp Manager

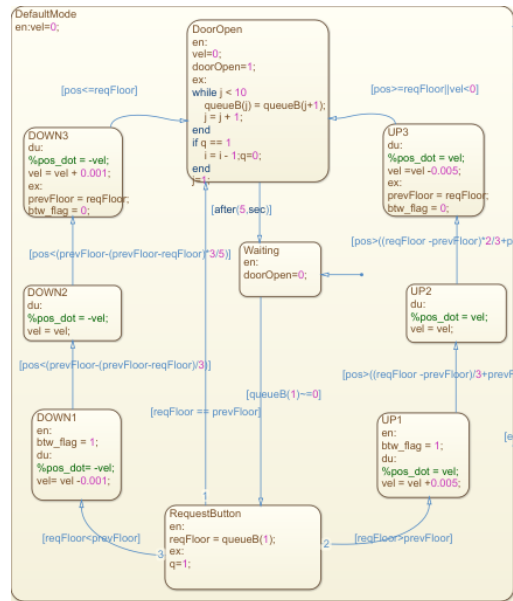
Based on the Button presses, the respective lamps are turned on. They only turn off when their respective floors have been completed.



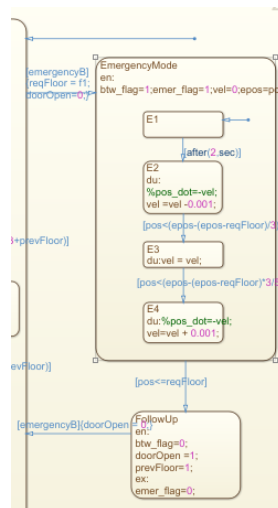
- ElevatorController

There are two modes in this model: DefaultMode and EmergencyMode.

The chart begins with the DefaultMode, it begins with waiting state and only transitions when the queue is not empty. When there is a request, the model decides whether to go UP or DOWN. In both the states, we need to change the speed of the motor gradually such that the elevator doesn't make sudden movements. So, every 1/3 of the distance the elevator needs to move, the speed slowly increments and becomes steady at after 1/3rd. Similarly, before the 2/3rd distance is covered, the speed starts decrementing gradually.



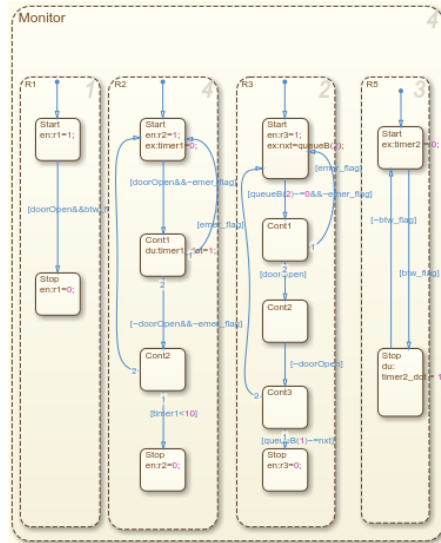
The emergencyMode state is enabled only if the emergency button is detected. It waits for 2 secs and then moves to Floor 1 automatically. No button presses will be registered at this point until the emergency button is pressed again to reset the elevator back. But the queue will remain as it is and continue from before.



The speed varies from UP and DOWN state. This is due to gravity playing a role when the elevator moves downwards. There is less stress on the motor when going down.

- Monitor

This state is used mainly to check for the given requirements that the elevator needs to follow. If any requirement is violated, the system stops itself.



This model is build into the Code that we generated earlier in STM32CubeMX. Using Keil, the code is fed into the STM3220G-EVAL board.

CALIBRATION & TESTING

There were a couple of changes we needed to calibrate with the physical model.

- The distances from each floor to the sensor had to be calibrated accordingly. The first floor was approximately at 10cm. The second floor was approximately at 30cm. The third floor was approx. at 48-50cm. The fourth floor was approx. at 66-70cm. But the readings were not accurate at certain situations.
- The motor also needed some calibrations when we drove it. As explained before, the Up and Down movement were different even thou they had same values. The speed is decreased a little when moving down to not crash the elevator at the bottom floor.
- We gave a delay of 5 secs to the door function so that the elevator remains on every floor at least for 5 secs.
- We made the elevator takes the first value given by the ultrasonic only for the 1st floor and compares itself to the logic within and then the resulting output, velocity, is fed into the motor block as speed.