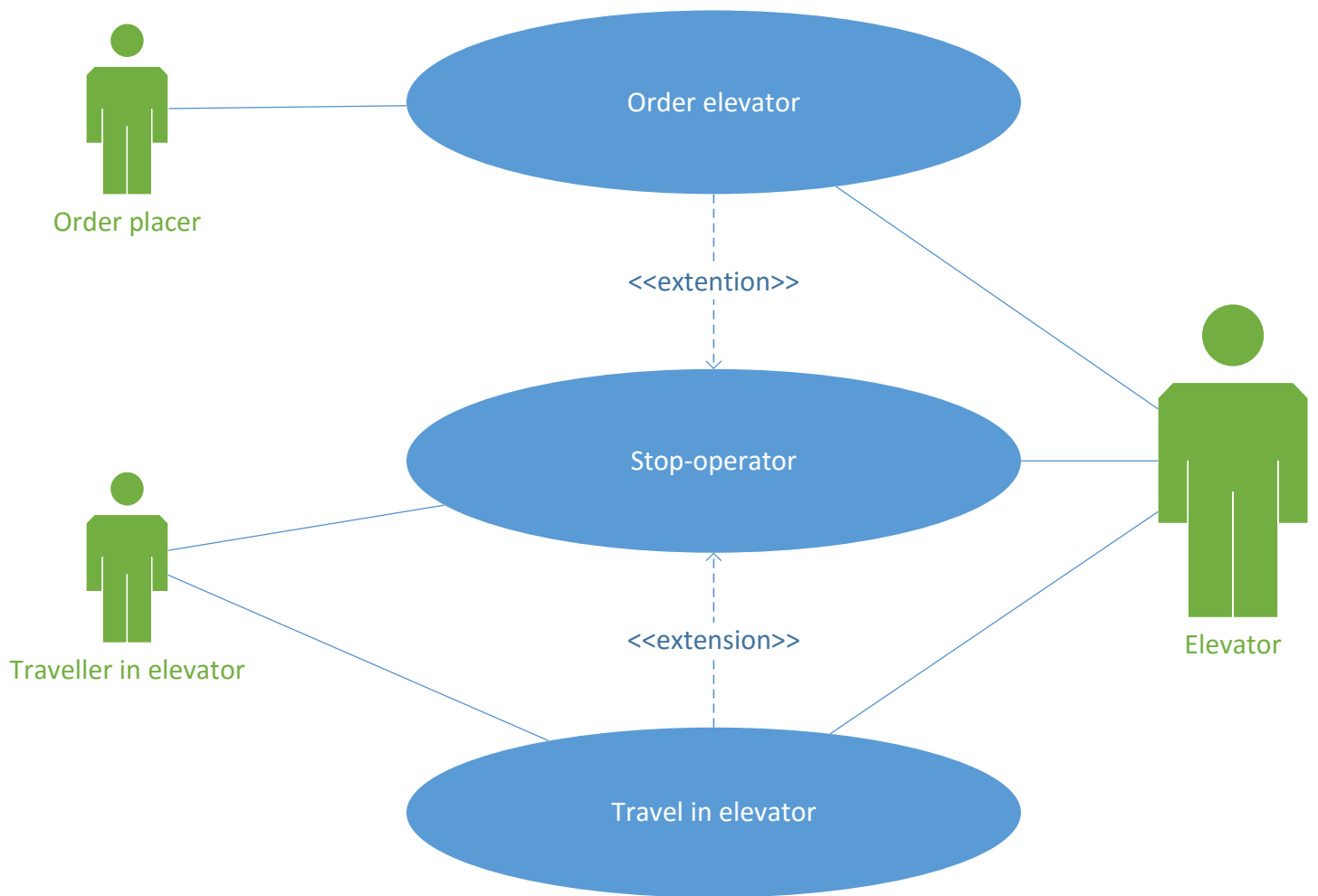


Use-case Diagram:



Use-case descriptions

UCD 1 – Call elevator

Precondition:

- Elevator in initialized state
- Stop-button not held in

Trigger: Call-button in given floor is pressed

Success-scenario:

1. Call-button lights up, and the order is registered.
2. Call-button light stays lit and order is awaiting in queue.
3. The elevator arrives at the floor it's ordered from.
4. The elevator stops, if the floor is next in queue.
5. Door opens, open-door indicator lights.
6. Call-button light goes out.
7. Order is removed from queue.
8. After 3 seconds of open door, the door closes, open-door light goes out.
9. Elevator stays still with door closed in an initialized state.

Extensions:

4a: The elevator doesn't stop (another order precedes it)

4a.1: Jump to step 2.

General exception: Stop button is pressed: Go to stop operator Use-case

Requirements for success:

Same as preconditions

Minimal guarantee:

Elevator stops

UCD 2 – Passenger travelling in elevator

Precondition:

- Stop button not pressed
- Elevator in initialized state

Trigger: Order-button inside elevator pressed.

Success-scenario:

1. Order-button inside elevator lights up.
2. Order is placed in queue.
3. Order-button inside elevator stays lit.
4. Elevator starts moving.
5. Elevator arrives at designated floor and stops.
6. Order-button light inside elevator goes out.
7. Door opens, open-door indicator lights up.
8. Order is removed from queue.
9. After 3 seconds of open door, the door closes, open-door light goes out.

Exceptions:

Trigger exception: Elevator's position and requested floor is the same: Jump to step 8.

General exception: Stop button is pressed: Go to stop operator Use-case

Requirement for success:

Same as preconditions

Minimal Guarantee:

The elevator stops.

UCD 3 – Stop-button operator

Preconditions:

None

Trigger: Stop button is pressed

Success-scenario:

1. The elevator stops
2. All orders are deleted, and further orders are ignored.
3. Stop button is released.
4. The elevator stays in place.
5. Orders can be registered again

Exceptions:

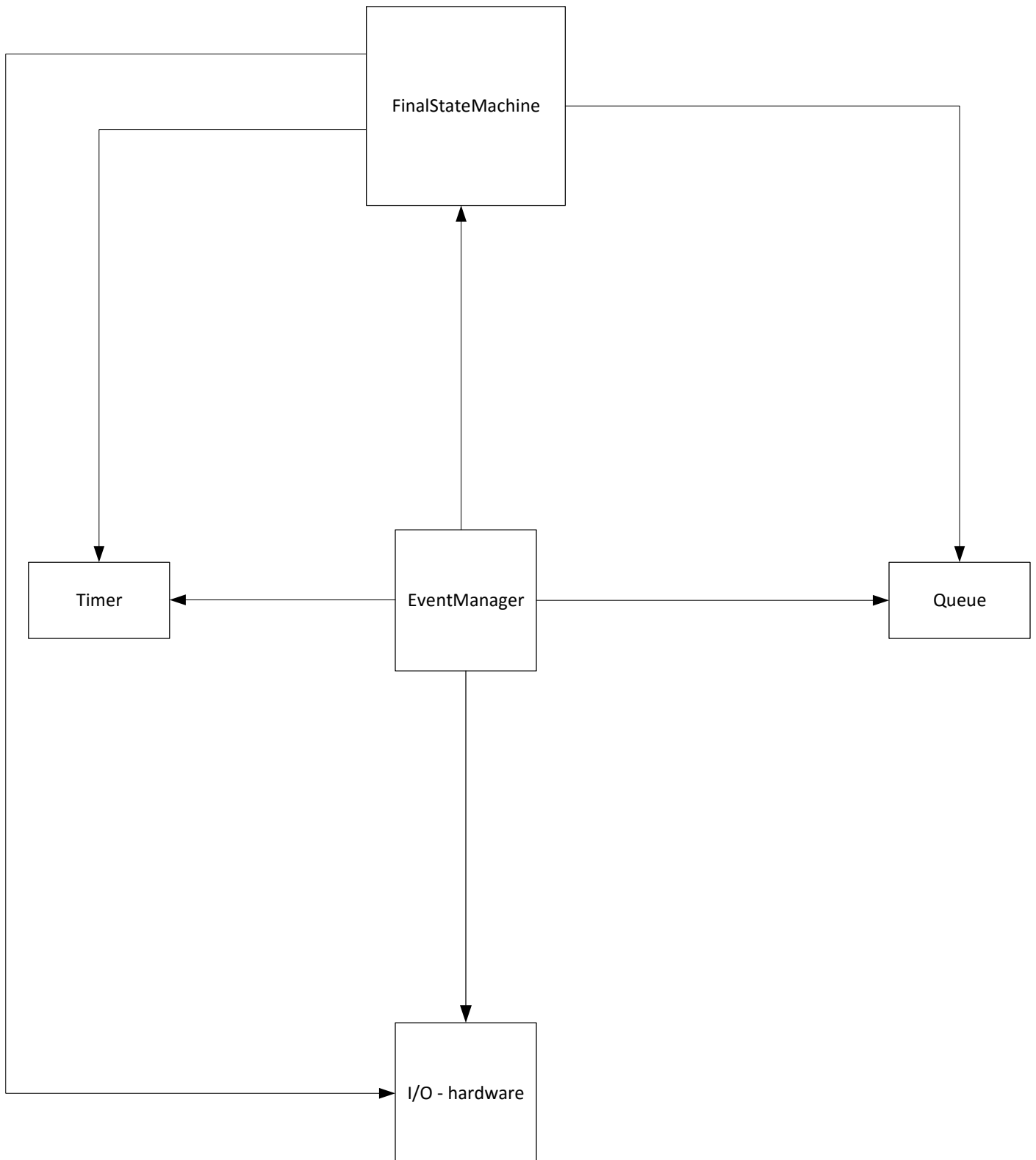
1a: Elevator is in an uninitialized state.

1a.1: Elevator enters an initialized state and stops. Proceed with next step.

1b: Elevator is not in-between floors.

1b.1: The door opens for as long as the button is pressed. Proceed with next step.

Communication Diagram: *General*



Communication Diagram:

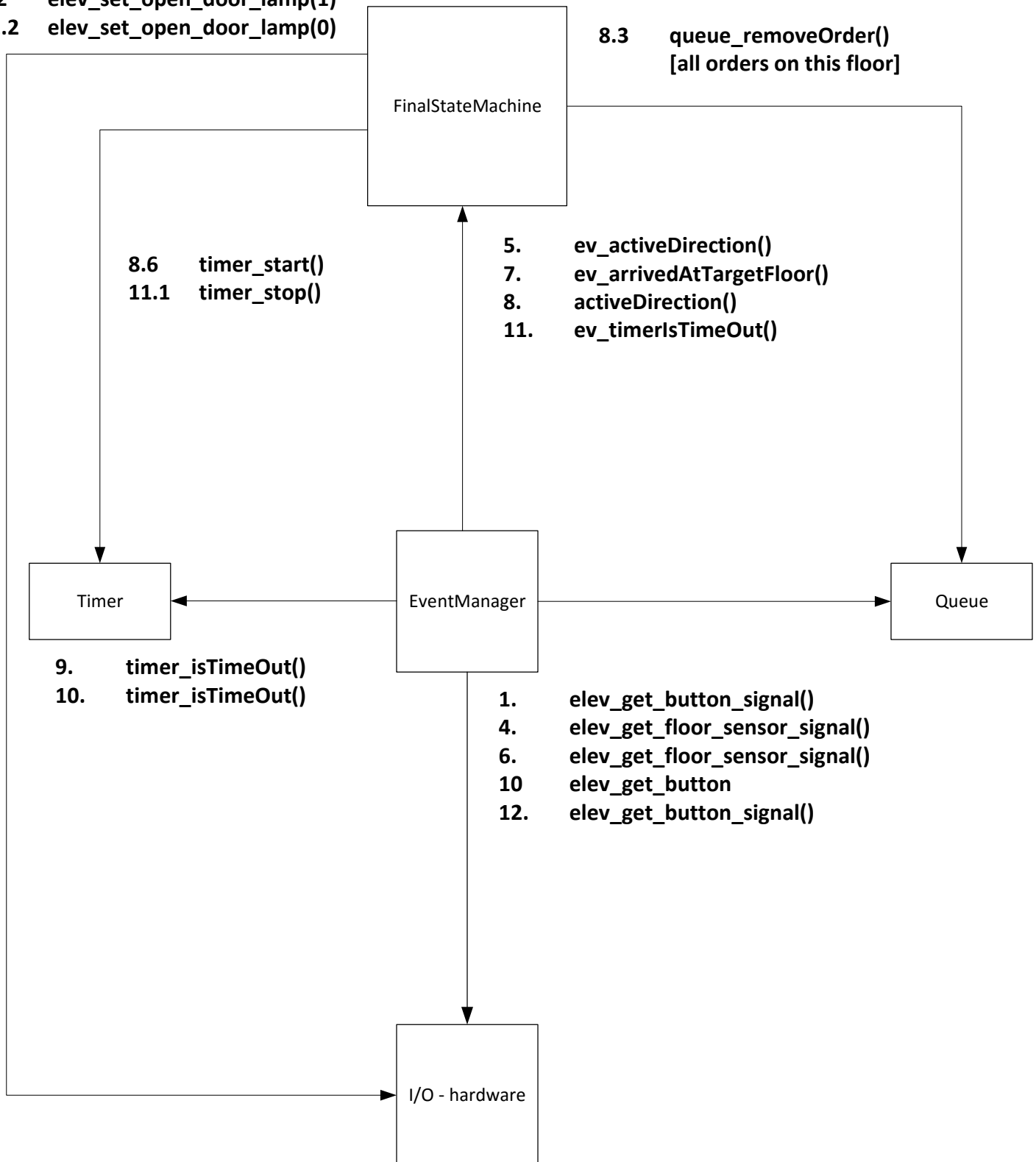
Use-case:

Call elevator,

Passenger travelling in elevator

- 5.1 elev_set_motor_direction()
- 7.1 elev_set_motor_direction(0)
- 8.1 elev_set_button_lamp(..,0)
[all buttons for this floor]
- 8.2 elev_set_open_door_lamp(1)
- 11.2 elev_set_open_door_lamp(0)

- 8.3 queue_removeOrder()
[all orders on this floor]



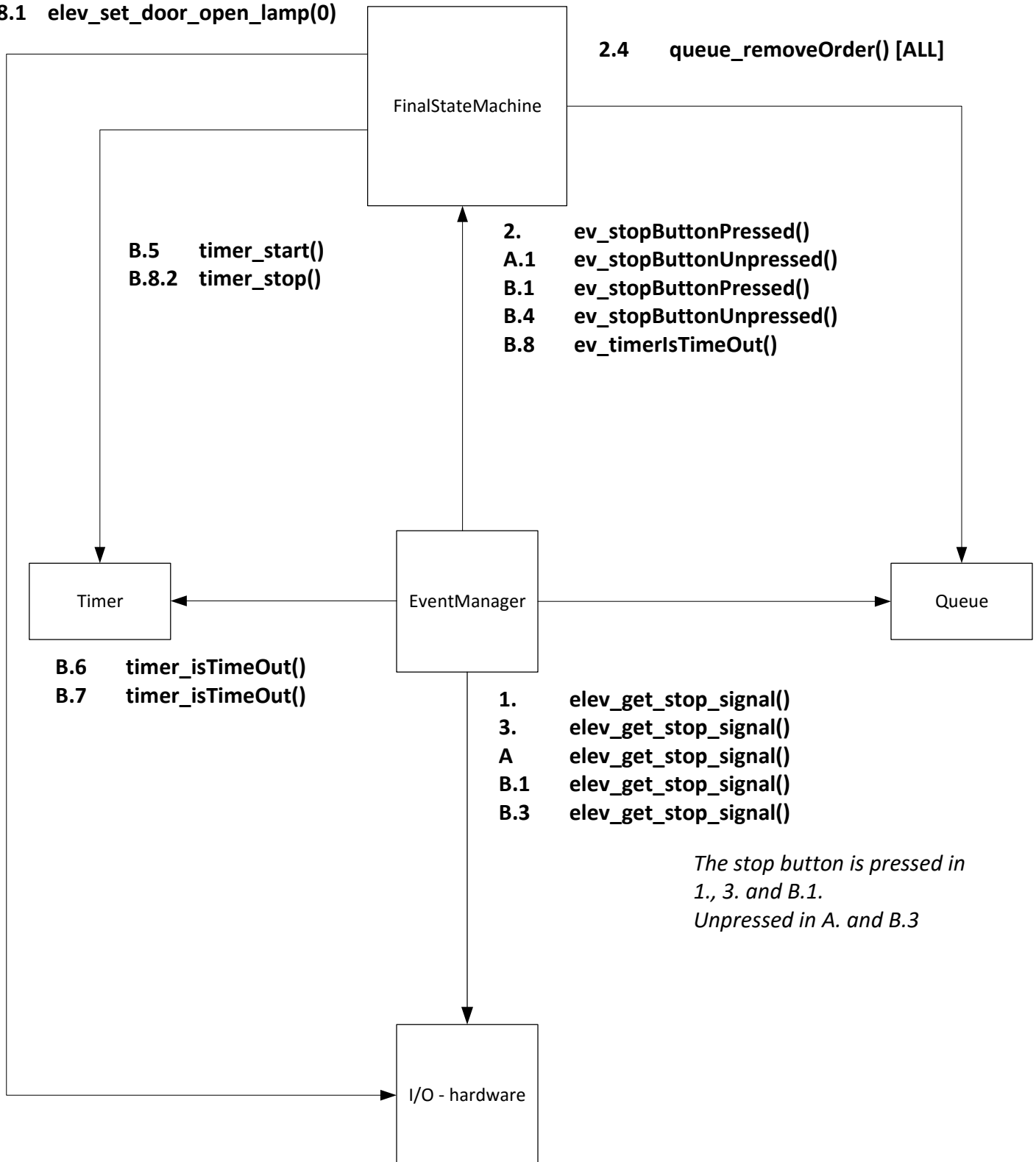
*If elev_get_floor_sensor_signal() returns -1
(is between floors) run all «A» scenarios ,
else run all «B» scenarios.*

Communication Diagram:

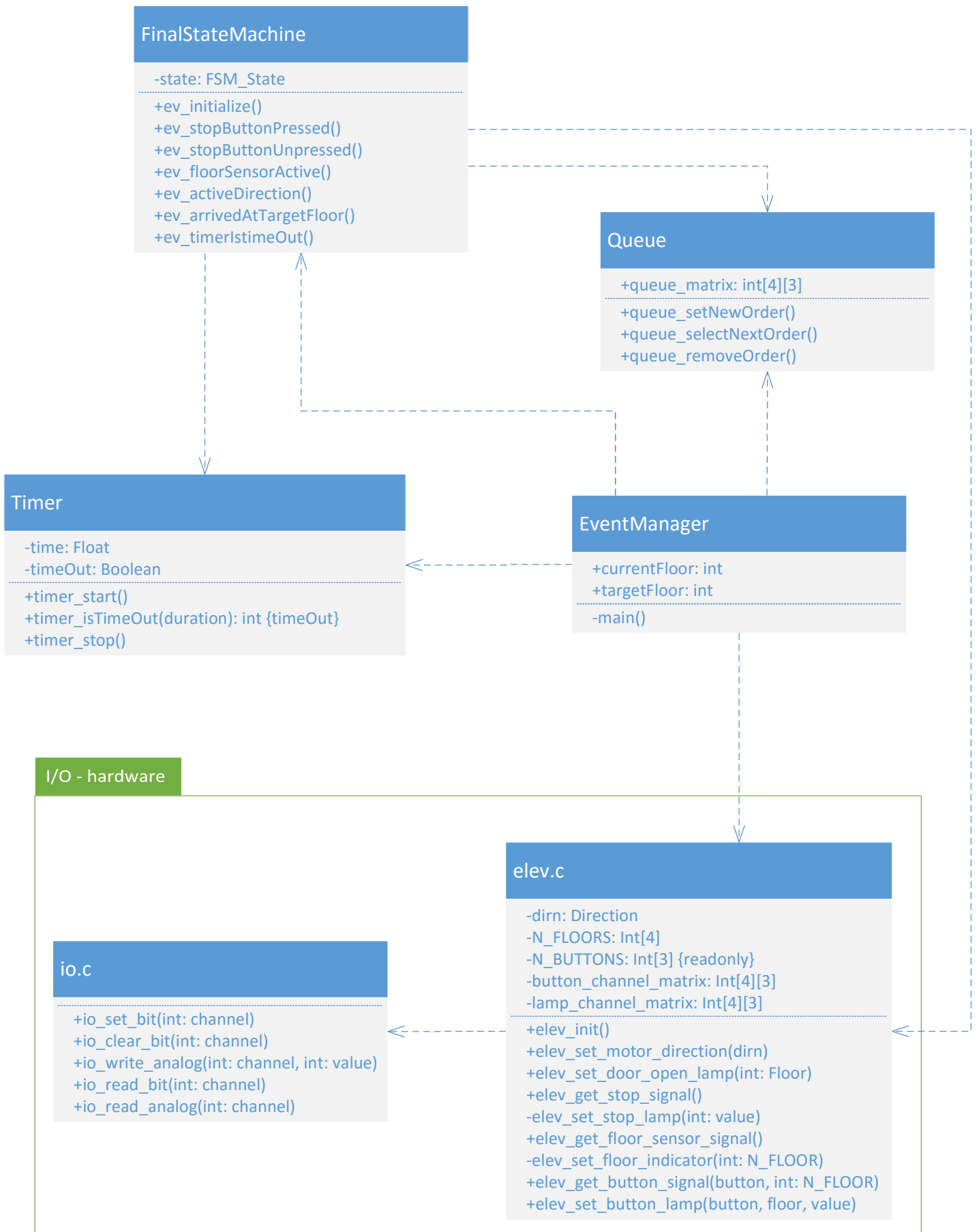
Use-case:

Stop-button operator

- 2.1 elev_set_motor_direction(0)
- 2.2 elev_get_floor_sensor_signal()
- 2.3 elev_set_button_lamp(0) [ALL]
- B.2 elev_set_door_open_lamp(1)
- B.8.1 elev_set_door_open_lamp(0)

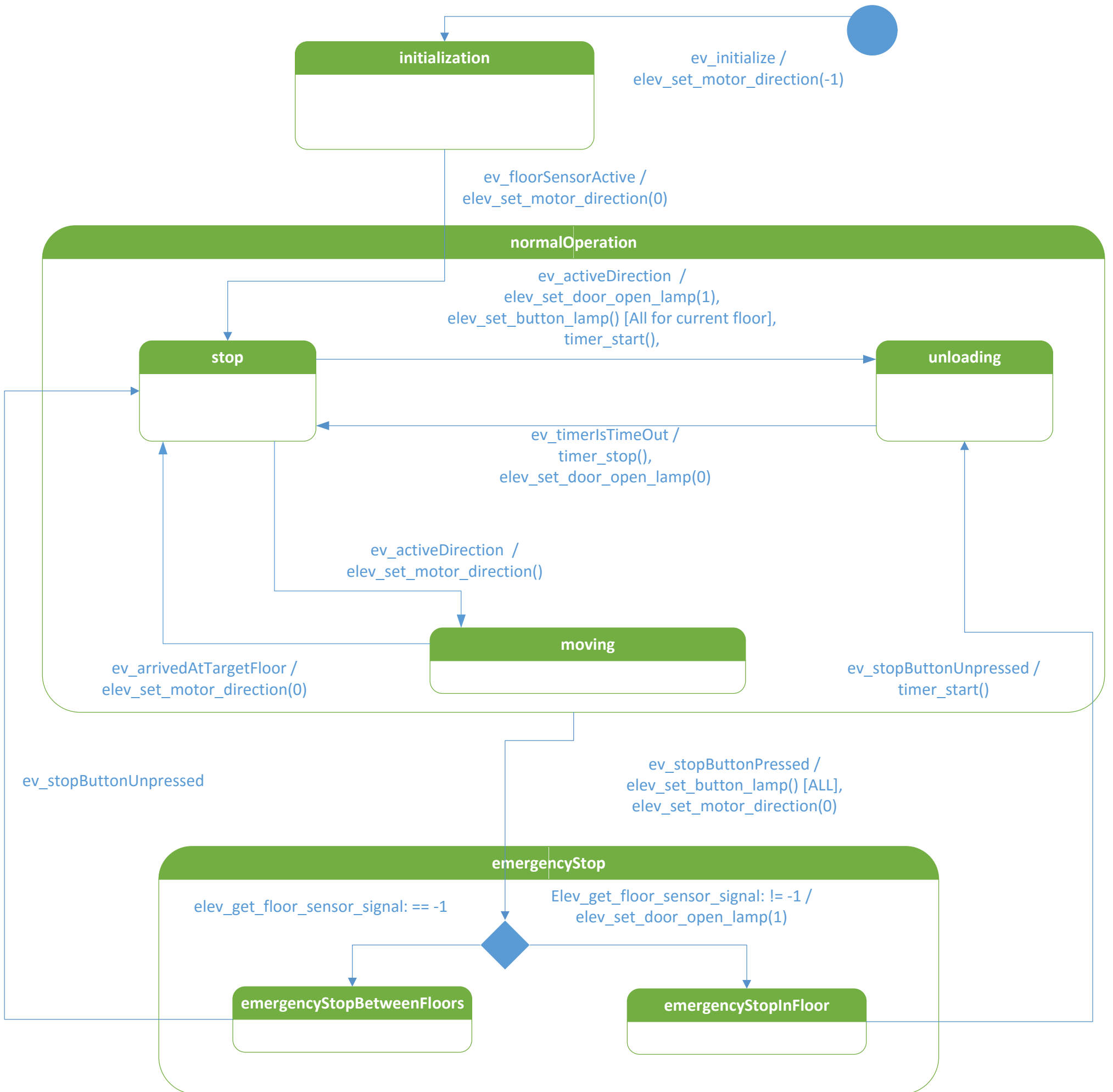


Class Diagram



State Machine Diagram

FinalStateMachine



Sequence Diagram – minimum requirement

