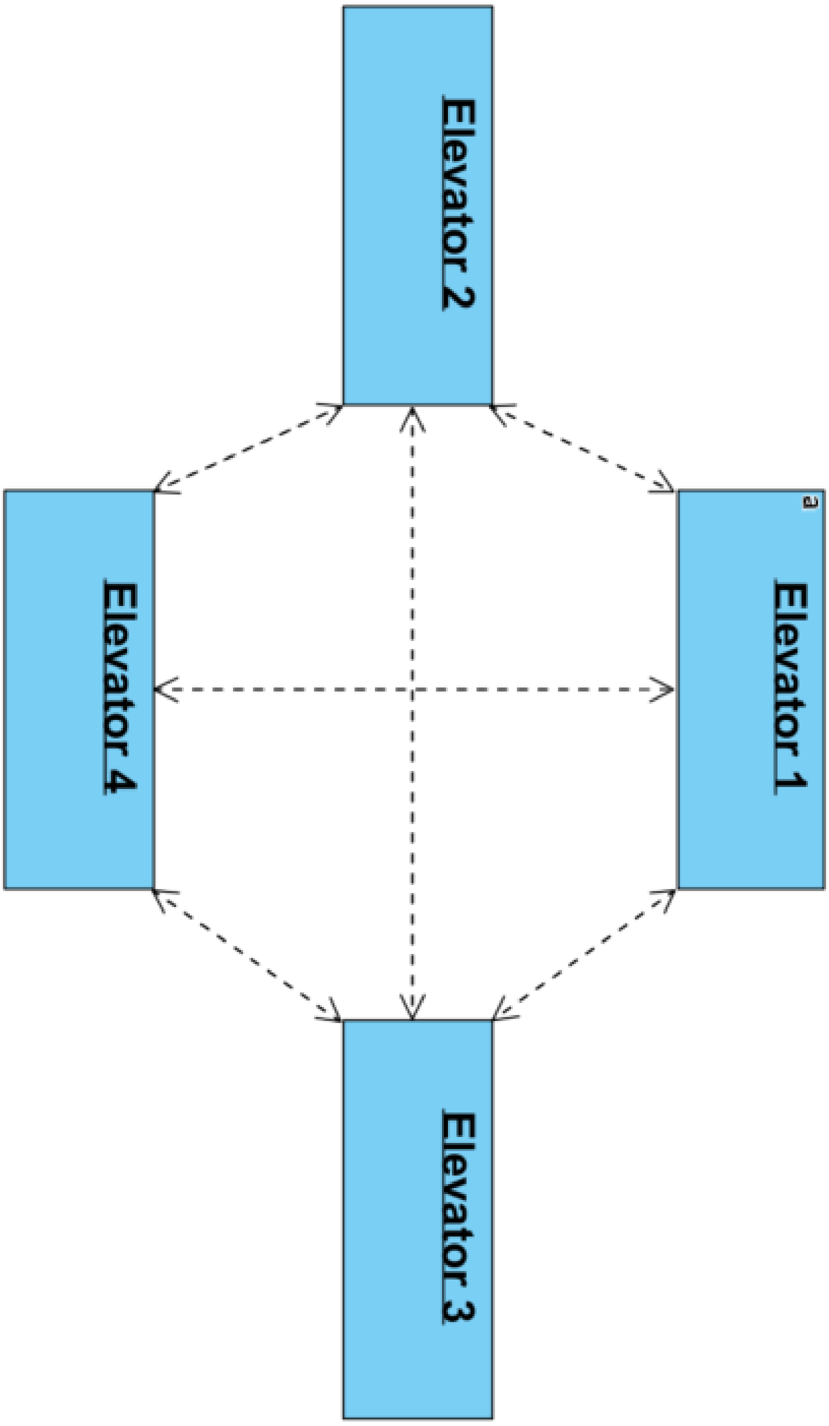
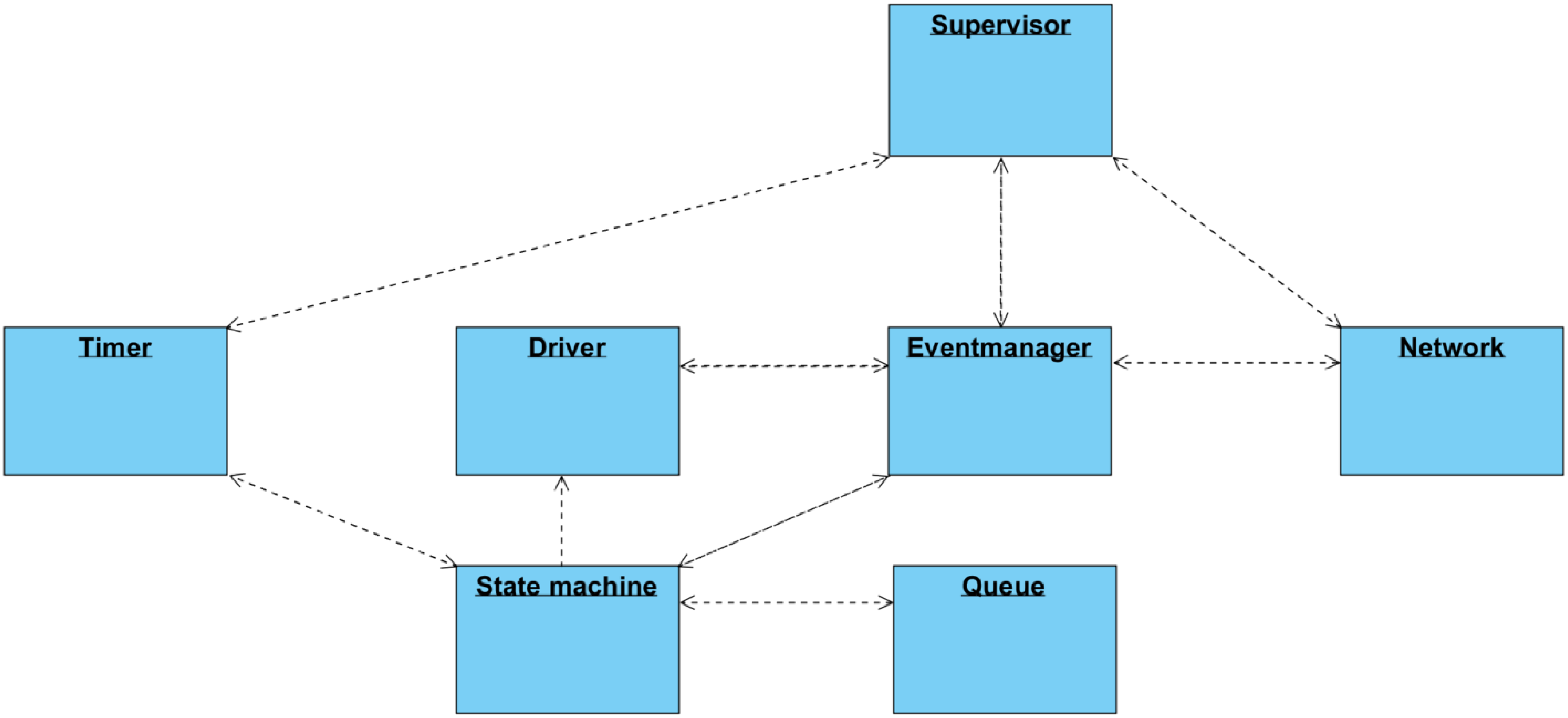
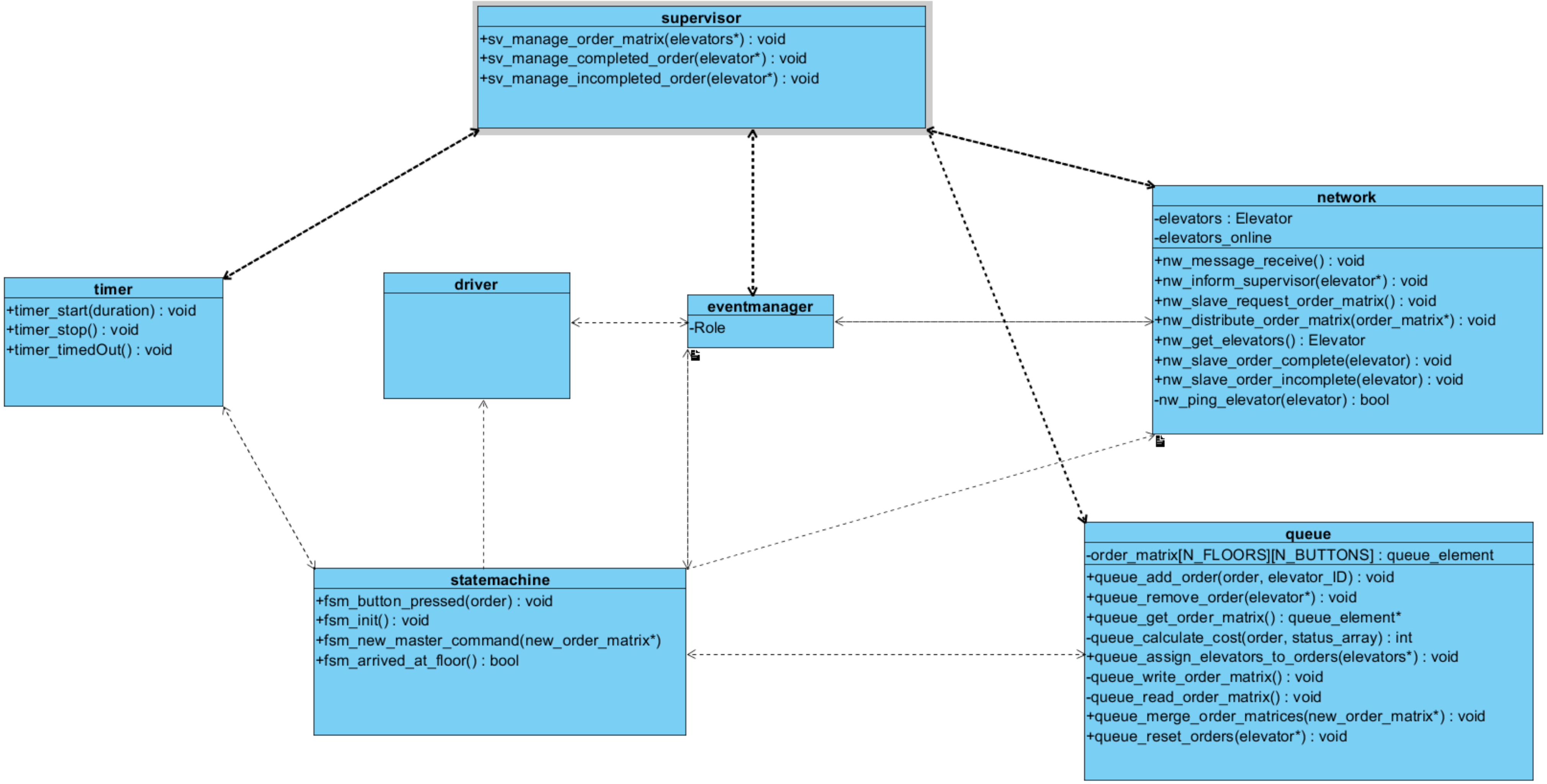
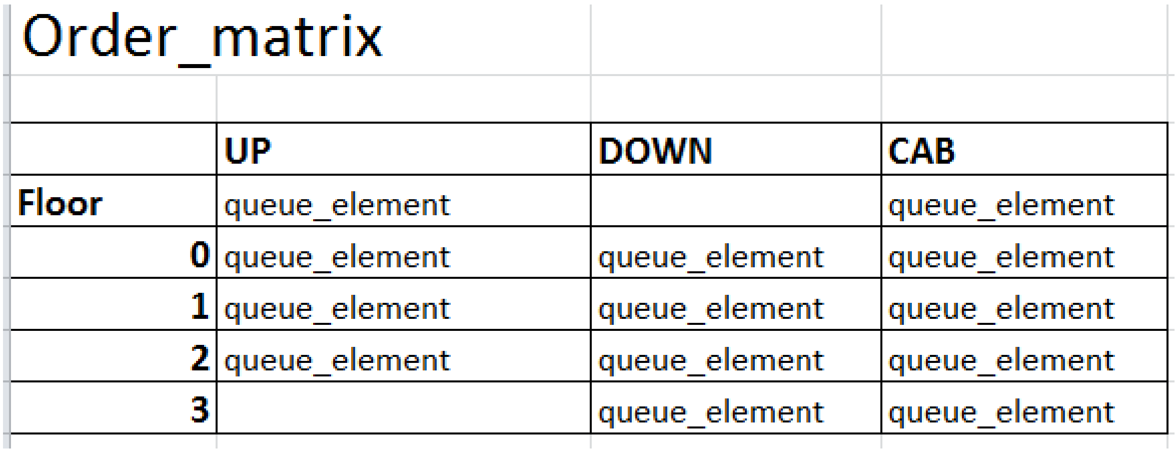

Elevator 2
Elevator 4
Elevator 1
a
Elevator 3

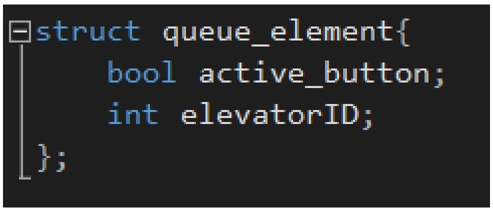

Supervisor
Timer
Driver
Eventmanager
Network
State machine
Queue


supervisor
+sv_manage_order_matrix(elevators*) : void
+sv_manage_completed_order(elevator*) : void
+sv_manage_incompleted_order(elevator*) : void
timer
+timer_start(duration) : void
+timer_stop() : void
+timer_timedOut() : void
driver
eventmanager
-Role
network
-elevators : Elevator
-elevators_online
+nw_message_receive() : void
+nw_inform_supervisor(elevator*) : void
+nw_slave_request_order_matrix() : void
+nw_distribute_order_matrix(order_matrix*) : void
+nw_get_elevators() : Elevator
+nw_slave_order_complete(elevator) : void
+nw_slave_order_incomplete(elevator) : void
-nw_ping_elevator(elevator) : bool
statemachine
+fsm_button_pressed(order) : void
+fsm_init() : void
+fsm_new_master_command(new_order_matrix*)
+fsm_arrived_at_floor() : bool
queue
-order_matrix[N_FLOORS][N_BUTTONS] : queue_element
+queue_add_order(order, elevator_ID) : void
+queue_remove_order(elevator*) : void
+queue_get_order_matrix() : queue_element*
-queue_calculate_cost(order, status_array) : int
+queue_assign_elevators_to_orders(elevators*) : void
-queue_write_order_matrix() : void
-queue_read_order_matrix() : void
+queue_merge_order_matrices(new_order_matrix*) : void
+queue_reset_orders(elevator*) : void

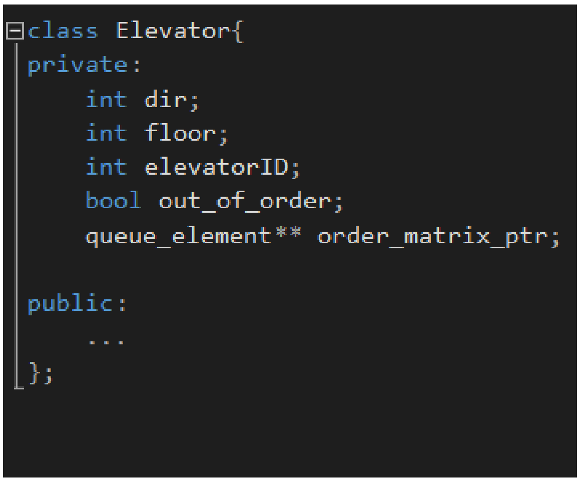


Order_matrix

	UP	DOWN	CAB
Floor	queue_element		queue_element
0	queue_element	queue_element	queue_element
1	queue_element	queue_element	queue_element
2	queue_element	queue_element	queue_element
3		queue_element	queue_element



```
struct queue_element{
    bool active_button;
    int elevatorID;
};
```



```
class Elevator{
private:
    int dir;
    int floor;
    int elevatorID;
    bool out_of_order;
    queue_element** order_matrix_ptr;

public:
    ...
};
```

Case: Button pressed at slave

```
//slave

fsm_button_pressed()

queue_add_order(order, elevatorID)

nw_inform_supervisor(elevator)


//supervisor

nw_message_receive() --> case: supervisor_informed

sv_manage_order_matrix(elevators*)

queue_merge_order_matrices(new_order_matrix*)

queue_assign_elevators(elevators*)

nw_distrubute_order_matrix(order_matrix*)


//slaves

nw_message_receive() --> case: new_order_matrix

fsm_new_master_command(new_order_matrix*)

queue_merge_order_matrices(new_order_matrix*)
```

Case: Slave order incomplete

```
//slave
Eventmanager:
timer_start(30)
if timed_out()
    nw_slave_order_incomplete(elevator*)
    elevator.out_of_order = 1
    fsm_init()


//Supervisor
nw_message_received()--> case: order_incomplete(elevator*)
sv_manage_incomplete_order(elevator*)
queue_reset_orders(elevator*)
queue_assign_elevators_to_orders(elevator*)
nw_distribute_order_matrix(order_matrix*)
```