Hardware.h -> Elevator.h

* timer.h -> fsm.h -> Orders.h
* Orders.h

Bør endres til: Hardware.h -> Elevator.h -> timer.h -> **fsm.h -> Orders.h**

**Elevator:**

**H:**

* void print\_elevator\_movement(HardwareMovement movement);
  + Printer mellom heisbevegelsene opp, ned og stopp til terminalen
* void print\_orders();
  + Printer ut en 4x3 matrise med bestillinger til terminalen
* void print\_order\_type(HardwareOrder order\_type);
  + Printer mellom bestillingstypene utside-opp, utside-ned og innside-bestilling til terminalen
* void print\_elevator\_behaviour(ElevatorBehaviour behaviour);
  + Printer ElevatorBehaviour til terminalen

**C:**

**Timer:**

**H:**

* void reset\_timer();
  + Nullstiller tidsvariabelen
* void start\_timer();
* int timed\_out();
  + Returnerer 1 hvis tiden er ute, og 0 hvis ikke

**C:**

* Bruker global klokke time(NULL) med time\_t

**Fsm:**

**H:**

* **void initialize\_elevator();**
  + Heisen gjøres klart til bruk ved å gå til nærmeste underetasje og deretter nullstille bestillingsmatrisen
* **void elevator\_arriving\_floor(int floor);**
  + Stopper heisen på etasje og åpner dør, eller hvis inititialized stopper den heisen og setter Elevator Behaviour til idle.
* **void button\_press\_event(int btn\_floor, HardwareOrder order\_type);**
  + Hvis heisen er i den aktuelle etasjen, så åpner den døren, hvis den ikke er i etasje legger den inn bestilling: Sjekker om heisen er i bevegelse, og bytter retning hvis forrige bestilling er over og på vei opp eller motsatt. Hvis heis ikke er i bevegelse, starter heisen etter den bestillingen. I tillegg printer den heisbevegelsen til terminalen og skrur på det aktuelle bestillingslyset.
* **void close\_door();**
  + Lukker døren, skrur av den aktuelle bestillingslampen og nullstiller bestillingen
* **void clear\_order\_light();**
  + Skrur av bestillingslampe spesifisert på bestillingstype og
* **int check\_door\_open();**
  + Sjekker om døren er åpen og returnerer henholdsvis 0/1 for ikke åpen /åpen
* **void stop\_button\_pressed();**
  + Hvis stopp-knappen holdes inne blir alle bestillinger og bestillingslys nullstilt og heisen står stille i nåværende posisjon frem til knappen ikke holdes inne.
* **int at\_floor();**
  + Sjekker om heisen er i en etasje og returnerer henholdsvis 0/1 for ikke i etg/ i etg
* **void clear\_all\_order\_lights();**
  + Skrur av alle bestillingslamper

**C:**

* Initialize
  + Heisen må gå til nærmeste etasje under om den ikke er i en etasje, for å kunne ta imot ordre
  + Setter ordre-matrisen lik 0
* Elevator-arriving\_floor
  + Stopper heis ved etg og åpner dør
  + Ved inititiaizen stopper den heisen og setter Elevator behaviour til idle
* Button\_press\_event
  + Hvis heis i etg -> åpne dør
  + Hvis ikke -> legg inn bestilling
    - Sjekker om heisen er i bevegelse -> bytter retning hvis forrige bestilling er over og på vei opp, og motsatt
    - Hvis heis ikke i bevegelse ->
  + Printer heisbevegelse
  + Skrur på bestillingslyset
* Close\_door
  + Skrur av bestillingslyset
  + Fjerner bestillingen
* Clear\_order\_light

**Orders:**

**H:**

* **int orders\_above(Elevator elevator);**
  + **sjekker om det er en bestilling i etasjen over**
* **int orders\_below(Elevator elevator);**
* **HardwareMovement orders\_choose\_direction(Elevator elevator);**
* **int should\_elevator\_stop(Elevator elevator);**
* **Elevator clear\_elevator\_order(Elevator elevator);**
* **Elevator clear\_all\_orders(Elevator elevator);**

**C:**