CBC - OneRoomPerson

Rafael Fernández Ortiz

May 3, 2022

Exercise 3

In order to bring a well-defined description about how the exercise and its events are thought and modeled, we gonna introduce first of all, all variables that participate and then we will check all features or/and requirements.

Variables

Let `RoomState` := {`empty`, `full`} the set of logical state of room occupancy in `Room` context. We define `room` ∈ `RoomState` as the variable that model the logical state of room occupancy. This variable belongs to `Controller` system/context.

Let `Light` := {`red`, `green`} the set of colors for the traffic light in `TrafficLightColors` context. We define `tfl` ∈ `Light` as the variable that indicate the color of the traffic light on the entry. This variable belongs to `Communication` system/context, i.e, a piece of whole system that describe the relationship of `Environment-Controller`.

We define `wio` ∈ {0, 1} as the variable that indicate if the signal of sensor is just turned off. This variable belongs to `Communication` system/context, i.e, a piece of whole system that describe the relationship of `Environment-Controller`.

Let `SensorState` := {`off`, `on`} the set of state for the sensor in `SensorIO` context. We define `ss` ∈ `SensorState` as the variable that indicate if someone is on it (state `on`) or otherwise (state `off`). This variable belongs to `Environment`.

Let `PersonIOrState` := {`out`, `in`} the set of state for physical position of someone in `PersonIO` context. We define `p` ∈ `PersonIOrState` as the variable that indicate if someone physically in the room (state `in`) or otherwise (state `out`). This variable belongs to `Environment`.

Features / Requirements

FUN1. This system deals with the access control of a room.

Done. Trivial by refinement.

SAF2. No more than one person can be in the room.

Done. That condition is restricted by `inv_room1 : room = full ⇒ tfl = red` and the guard for the event that indicate the traffic light must be green for someone who go out from the sensor vision, can go inside the room, i.e. `Person_Go_Out_Sensor: grd1 : tfl = green`. Therefore, there is no way that someone else go inside to the room while anyone is there.

FUN3. The system must not deadlock.

I'm no sure how can I proove that.

EQP4. There is a status light outside the room with two colors: green and red.

Done. We have the variable `tfl` ∈ `Light`

FUN5. When the satus light is red, the room cannot be accessed. When te status light is green, the room can be accessed.

Done. The event that indicate whether the room can be accessed or not, is modeled by `Person_Go_Out_Sensor` event that require traffic light `tfl` have to be `green`.

EQP6. There is a presence sensor at the entrance of the room.

Done. We have the variable `ss` ∈ `SensorState`.

FUN7. The presence sensor produces an `on` signal when a person is standing on it and `off` signal otherwise.

Done. The event that model the presence sensor produces an `on` signal when a person is standing on and otherwise are `Sensor_Turn_On` event (that set the variable `ss` to `on`) and `Person_Go_Out_Sensor` event (that set the variable `ss` to `off`).

FUN8. A person inside the room can leave at any moment (using a door different from the one used to enter the room).

Done. The event that model a person inside the room can leave at any moment is `Person_Go_Out` event.

ENV9. The inside of the room cannot be seen from its outside and vice-versa.

Done. The whole system force that anyone has to obey the system laws, i.e, anyone cannot know what's happens into the room until that person is inside.

ENV10. People obey the status light.

Done. In order to someone wants to access inside the room, the traffic light must be green. This is modelled by having `tfl` = `green` as requirement for the only event that lets someone go inside.

FUN11. Anyone wishing to enter the room must step on the sensor, and wait there until the status light is green, if it is not already.

Done. This is modelled by the only two events that allow someone go inside the room which which describe that behaviour, i.e. `Sensor_Turn_On` and `Person_Go_Out_Sensor`.

ENV12. Anyone who stands on the sensor will wait there for the status light to turn green and enter the room.

Done. This is modelled by the event `Person_Go_Out_Sensor` which requires `tfl` = `green` and changes `ss` from `on` to `off`.