Contents

CONTEXT TrafficLightColors
EXTENDS
SETS

Light

CONSTANTS

red

green

AXIOMS

axm1: false⟨theorem⟩ $Light = \{red, green\}$

axm2: false⟨theorem⟩ $red \neq green$

END

CONTEXT PersonIO

Description of the person if physically in or out of the room

Description of the person if physically in or out of the room

EXTENDS

SETS

PersonIOState

CONSTANTS

in

out

AXIOMS

axm1: false⟨theorem⟩ $PersonIOState = \{out, in\}$

axm2: false⟨theorem⟩ $out \neq in$

END

CONTEXT SensorIO

Context describing the sensor states

Context describing the sensor states

EXTENDS

SETS

SensorState Set of sensor statesSet of sensor states

CONSTANTS

on Sensor signal onSensor signal on

off Sensor signal offSensor signal off

AXIOMS

axm1: false⟨theorem⟩ $SensorState = \{off, on\}$

axm2: false⟨theorem⟩ $off \neq on$

END

CONTEXT Room

Context describing the room states

Context describing the room states

EXTENDS

SETS

RoomState

CONSTANTS

empty

full

AXIOMS

axm1: false⟨theorem⟩ $RoomState = \{empty, full\}$

axm2: false⟨theorem⟩ $empty \neq full$

END

MACHINE m0

First implementation of the model. The entry and exit of a person is considered without any external restriction to the room capacity.

First implementation of the model. The entry and exit of a person is considered without any external restriction to the room capacity.

REFINES Room

SEES Room

VARIABLES

room

INVARIANTS

inv1: false⟨theorem⟩ $room \in RoomState$

EVENTS

Initialisation true⟨extended⟩

falsethenbegin

act1: true$room := empty$$room := empty$

end

Event Person_Go_In ⟨ordinary⟩ $\widehat{=}$

falseextendsrefines

falsewherewhen

grd1: false⟨theorem⟩ true$room = empty$$room = empty$

truethenbegin

act1: true$room := full$$room := full$

end

Event Person_Go_Out ⟨ordinary⟩ $\widehat{=}$

falseextendsrefines

falsewherewhen

grd1: false⟨theorem⟩ true$room = full$$room = full$

truethenbegin

act1: true$room := empty$$room := empty$

end

END

MACHINE m1

Second implementation of the model. The entry and exit of a person is restringed by a traffic light that denote if the room is empty or full.

Second implementation of the model. The entry and exit of a person is restringed by a traffic light that denote if the room is empty or full. m0

REFINES m0 Room,TrafficLightColors

SEES Room,TrafficLightColors

VARIABLES

room

tfl Traffic Light VariableTraffic Light Variable

INVARIANTS

inv1: false⟨theorem⟩ $tfl \in Light$

EVENTS

Initialisation true⟨extended⟩

falsethenbegin

act1: false$room := empty room := empty$

act2: true$tfl := red tfl := red$

end

Event Person_Go_In ⟨ordinary⟩ $\widehat{=}$ Person_Go_In

trueextendsrefines Person_Go_In

falsewherewhen

grd1: false⟨theorem⟩ false$room = empty room = empty$

grd2: false⟨theorem⟩ true$tfl = green tfl = green$

truethenbegin

act1: false$room := full room := full$

act2: true$tfl := red tfl := red$

end

Event Person_Go_Out ⟨ordinary⟩ $\widehat{=}$ Person_Go_Out

trueextendsrefines Person_Go_Out

falsewherewhen

grd1: false⟨theorem⟩ false$room = full room = full$

truethenbegin

act1: false$room := empty room := empty$

end

Event Traffic_Switch_Green ⟨ordinary⟩ $\widehat{=}$

This is for the signal to know the state of the environment (due to the absence of the output sensor).)

This is for the signal to know the state of the environment (due to the absence of the output sensor).)

falseextendsrefines

falsewherewhen

grd1: false⟨theorem⟩ true$room = empty room = empty$

grd2: false⟨theorem⟩ true$tfl = red tfl = red$

truethenbegin

act1: true$tfl := green tfl := green$

end

END

MACHINE m2 m1
REFINES m1 Room,TrafficLightColors,SensorIO,PersonIO
SEES Room,TrafficLightColors,SensorIO,PersonIO
VARIABLES

room
tfl Traffic Light variableTraffic Light variable
ss Sensor variableSensor variable
wio Wire from sensor to controllerWire from sensor to controller
p Person In/Out variablePerson In/Out variable

INVARIANTS

inv_room1: false⟨theorem⟩ $room = full \Rightarrow tfl = red$
inv_room2: false⟨theorem⟩ $room = full \Rightarrow p = in$
inv_ss1: false⟨theorem⟩ $ss \in SensorState$
inv_ss2: false⟨theorem⟩ $ss = on \Rightarrow wio = 0$
inv_ss3: false⟨theorem⟩ $ss = off \land wio = 0 \Rightarrow tfl = red$
inv_wio1: false⟨theorem⟩ $wio \in \{0, 1\}$
inv_wio2: false⟨theorem⟩ $wio = 1 \Rightarrow tfl = green$
inv_wio3: false⟨theorem⟩ $wio = 0 \Leftrightarrow (p = in \land room = full) \lor (p = out \land room = empty)$
inv_wio4: false⟨theorem⟩ $wio = 1 \Rightarrow p = in \land room = empty$
inv_p1: false⟨theorem⟩ $p \in PersonIOState$
inv_p2: false⟨theorem⟩ $p = out \Rightarrow room = empty$

EVENTS

Initialisation true⟨extended⟩

falsethenbegin

act1: false$room := empty$$room := empty$
act2: false$tfl := red$$tfl := red$
act4: true$wio := 0$$wio := 0$
act3: true$ss := off$$ss := off$
act5: true$p := out$$p := out$

end

Event Person_Go_In ⟨ordinary⟩ $\widehat{=}$

Event where the controller knows the signal from the cable (which assumes a person has entered) and then changes the state.
Event where the controller knows the signal from the cable (which assumes a person has entered) and then changes the state. Person_Go_In

falseextendsrefines Person_Go_In

falsewherewhen

grd1: false⟨theorem⟩ true$room = empty$$room = empty$
grd2: false⟨theorem⟩ true$wio = 1$$wio = 1$

truethenbegin

act1: true$room := full$$room := full$
act2: true$wio := 0$$wio := 0$
act3: true$tfl := red$$tfl := red$

end

Event Person_Go_Out ⟨ordinary⟩ $\widehat{=}$

Exit event. As there is no exit sensor, we can assume that both states (To be inside and room is full) change simultaneously.
Exit event. As there is no exit sensor, we can assume that both states (To be inside and room is full) change simultaneously. Person_Go_Out

trueextendsrefines Person_Go_Out

falsewherewhen

grd1: false⟨theorem⟩ false$room = full$$room = full$
grd2: false⟨theorem⟩ true$p = in$$p = in$

truethenbegin

act1: false$room := empty$$room := empty$

act2: true$p := out$$p := out$

end

Event Traffic_Switch_Green ⟨ordinary⟩ $\widehat{=}$

Event where sensor is on and it is satisfied that entry can be allowed.

Event where sensor is on and it is satisfied that entry can be allowed. Traffic_Switch_Green

trueextendsrefines Traffic_Switch_Green

falsewherewhen

grd1: false⟨theorem⟩ false$room = empty$$room = empty$

grd2: false⟨theorem⟩ false$tfl = red$$tfl = red$

grd3: false⟨theorem⟩ true$ss = on$$ss = on$

truethenbegin

act1: false$tfl := green$$tfl := green$

end

Event Sensor_Turn_On ⟨ordinary⟩ $\widehat{=}$

Event where sensor is on with no one is staging

Event where sensor is on with no one is staging

falseextendsrefines

falsewherewhen

grd1: false⟨theorem⟩ true$ss = off$$ss = off$

grd2: false⟨theorem⟩ true$wio = 0$$wio = 0$

truethenbegin

act1: true$ss := on$$ss := on$

end

Event Person_Go_Out_Sensor ⟨ordinary⟩ $\widehat{=}$

Event where situation satisfied that person who is waiting, can go inside.

Event where situation satisfied that person who is waiting, can go inside.

falseextendsrefines

falsewherewhen

grd1: false⟨theorem⟩ true$tfl = green$$tfl = green$

grd2: false⟨theorem⟩ true$ss = on$$ss = on$

truethenbegin

act1: true$p := in$$p := in$

Person physically go inside

Person physically go inside

act2: true$ss := off$$ss := off$

sensor is turned off

sensor is turned off

act3: true$wio := 1$$wio := 1$

end

END