# 1 Projekt

# 2 TrafficLight

## 2.1 Model Code

```
1   MACHINE TrafficLight
2
3  SETS colors = {red, redyellow, yellow, green}
4
5  VARIABLES tl_cars, tl_peds
6
7  INVARIANT tl_cars : colors &
8      tl_peds : {red, green} &
9      (tl_peds = red or
10     tl_cars = red)
11
12 INITIALISATION  tl_cars := red; tl_peds := red
13
14 OPERATIONS
15
16 cars_ry =
17   SELECT tl_cars = red & tl_peds = red THEN
18     tl_cars := redyellow
19   END;
20
21 cars_y =
22   SELECT tl_cars = green & tl_peds = red THEN
23     tl_cars := yellow
24   END;
25
26 cars_g =
27   SELECT tl_cars = redyellow & tl_peds = red THEN
28     tl_cars := green
29   END;
30
31 cars_r =
32   SELECT tl_cars = yellow & tl_peds = red THEN
33     tl_cars := red
34   END;
35
36 peds_r =
37   SELECT tl_peds = green & tl_cars = red THEN
38     tl_peds := red
39   END;
40
41 peds_g =
42   SELECT tl_peds = red & tl_cars = red THEN
```

```
43      tl_peds := green
44    END
45
46  END
```

Listing 1: TrafficLight MCH Code

## 2.2 Model Checking

| Modelchecking Item | Modelchecking Result |
|---|---|
| Gemischte Breiten-/Tiefensuche, Invarianten überprüfen | Modelchecking not solved |
| Gemischte Breiten-/Tiefensuche, Deadlocks finden | Modelchecking not solved |

Table 1: Modelchecking Items and Results

## 2.3 LTL Model Checking

| LTL Formular | Status |
|---|---|
| `G{tl_cars = red or tl_peds = red}` | Formular not Solved |
| `G{tl_cars = red}` | Formular not Solved |

Table 2: LTL Formulars and Results

| Pattern Name | Code | Result |
|---|---|---|

Table 3: LTL Patterns and Results

## 2.4 Symbolic Model Checking

| Symbolic Type | Configuration | Result |
|---|---|---|
| INVARIANT | | Formular not Solved |
| DEADLOCK | 1=1 | Formular not Solved |
| CHECK_WELL_DEFINEDNESS | | Formular not Solved |

| Symbolic Type | Configuration | Result |
|---|---|---|
| CHECK_STATIC_ASSERTIONS | | Formular not Solved |
| INVARIANT | cars_g | Formular not Solved |
| SYMBOLIC_MODEL_CHECK | TINDUCTION | Formular not Solved |
| FIND_REDUNDANT_INVARIANTS | | Formular not Solved |

Table 4: Symbolic Formulars and Results

### 2.5 Traces

Table 5: TrafficLight_Cars

| Position | Transition |
|---|---|
| 0 | INITIALISATION |
| 1 | cars_ry |
| 2 | cars_g |
| 3 | cars_y |
| 4 | cars_r |

Table 6: TrafficLight_Peds

| Position | Transition |
|---|---|
| 0 | INITIALISATION |
| 1 | peds_g |
| 2 | peds_r |

## 3 PitmanController_TIME_MC_v4

### 3.1 Model Code

```
1  MACHINE PitmanController_TIME_MC_v4
2
3 INCLUDES BlinkLamps_v3, Sensors, GenericTimersMC
4 /*
5  The BlinkLamps machine takes care of flashing the lights and
       the
6  remaining blinks (for tip blinking).
```

```
 7  The main machine only has to worry about setting
        active_blinkers and
 8  for setting the blinkers to continuous or tip-blinking
 9  v5 uses v3 BlinkLamps
10 */
11
12
13 CONSTANTS
14   pitman_direction /* "Convert Pitman position into blink
        direction" */
15 PROPERTIES
16   pitman_direction = {Neutral |-> neutral_blink, Downward5 |->
        left_blink, Downward7 |-> left_blink,
17                       Upward5 |-> right_blink, Upward7 |->
                             right_blink}
18
19 INVARIANT
20  /* "SAF-H1" */ (hazardWarningSwitchOn = switch_on =>
        active_blinkers=BLINK_DIRECTION)
21  /* "ELS-8: As long as the hazard warning light switch is
        pressed (active), all
22           direction indicators flash synchronously. " */ &
23
24   /* "SAF-H2" */ (hazardWarningSwitchOn = switch_off &
        remaining_blinks = -1
25    => active_blinkers = {pitman_direction(pitmanArmUpDown)} ) &
26
27   /* "SAF-H3" */ (pitmanArmUpDown:PITMAN_DIRECTION_BLINKING &
        engineOn=TRUE
28    => {pitman_direction(pitmanArmUpDown)} <: active_blinkers) &
29
30     /* "SAF-H4" */ (engineOn=FALSE & hazardWarningSwitchOn =
          switch_off => active_blinkers={})
31   &
32 // new invariants required for Rodin Proof:
33
34  /* "SAF-H0" */ (hazardWarningSwitchOn = switch_on =>
        remaining_blinks = -1) &
35
36  /* "SAF-H3b"*/ (pitmanArmUpDown ∈ PITMAN_DIRECTION_BLINKING &
        engineOn=TRUE
37         ⇒ remaining_blinks = -1)
38
39 ASSERTIONS
40 /* "thm1" */ pitman_direction : PITMAN_POSITION --> DIRECTIONS
41
42 INITIALISATION
43   AbsoluteSetDeadline(blink_deadline,500)
44 OPERATIONS
```

```
45  ENV_Pitman_Tip_blinking_start(newPos) =
46  SELECT newPos : PITMAN_TIP_BLINKING &
47     newPos /= pitmanArmUpDown THEN
48     // ELS-2, ELS-5
49     SET_Pitman_Tip_blinking_short(newPos);
50     IF hazardWarningSwitchOn = switch_off
51        /* "ELS-13: If the warning light is activated,
52                 any tip-blinking will be ignored ..." */
53        & engineOn = TRUE
54       THEN
55          SET_BlinkersOn(pitman_direction(newPos),3)
56      END;
57     AddDeadline(tip_deadline,500)
58  END;
59
60
61  RTIME_Tip_blinking_Timeout(delta) =
62   SELECT
63     /* "grdTip" */ delta ∈ 0..500
64   THEN
65     IF pitmanArmUpDown : PITMAN_TIP_BLINKING & remaining_blinks
         > 1 &
66        active_blinkers = {pitman_direction(pitmanArmUpDown)}
            THEN
67        // after 0.5 seconds a Tip blinking is cancelled and
            replaced by a continuous blinking
68        // ELS-4: If the driver holds the pitman arm for more
            than 0.5 seconds in position "tip-blinking left",
69        // flashing cycles are initiated for all direction
70        // indicators on the left (see Req. ELS-1) until the
            pitman arm leaves the position "tip-blinking left".
71       SET_RemainingBlinks(-1)
72     END;
73     IncreaseTimeUntilDeadline(tip_deadline,delta)
74  END;
75
76  RTIME_BlinkerOn(delta) =
77   SELECT
78     /* "grdTip" */ delta ∈ 0..500
79   THEN
80     TIME_BlinkerOn;
81     IncreaseTimeUntilCyclicDeadline(blink_deadline,delta,500)
82  END;
83
84  RTIME_BlinkerOff(delta) =
85   SELECT
86     /* "grdTip" */ delta ∈ 0..500
87   THEN
88     TIME_BlinkerOff;
```

```
89      IncreaseTimeUntilCyclicDeadline(blink_deadline,delta,500)
90    END;
91
92    RTIME_Nothing(delta,newOnCycle) =
93     SELECT
94      /* "grdDelta" */ delta ∈ 0..500 &
95      newOnCycle : BOOL
96     THEN
97      TIME_Nothing(newOnCycle);
98      IncreaseTimeUntilCyclicDeadline(blink_deadline,delta,100)
99    END;
100
101   RTIME_Passes(delta) = SELECT delta : {100}
102    THEN
103     IncreaseTime(delta)
104   END;
105
106   ENV_Turn_EngineOn =
107   BEGIN
108     SET_EngineOn;
109     IF pitmanArmUpDown :PITMAN_DIRECTION_BLINKING &
110        hazardWarningSwitchOn = switch_off THEN
111        SET_BlinkersOn(pitman_direction(pitmanArmUpDown), -1)
112     END
113   END;
114
115   ENV_Turn_EngineOff =
116   BEGIN
117     SET_EngineOff;
118     IF hazardWarningSwitchOn = switch_off
119      /* "ELS-8 As long as the hazard warning light switch is
            pressed (active),
120               all direction indicators flash synchronously." */
121               // TO DO: pluse ratio 1:2 if ignition key is in
                     lock
122     THEN
123          SET_AllBlinkersOff
124     END
125   END;
126
127   ENV_Pitman_DirectionBlinking (newPos) =
128   // corresponds to pitmanArmUpDown = 2 or 4 (Upward/Downward7)
129   // ELS-1, ELS-5
130   PRE newPos : PITMAN_POSITION & newPos /= pitmanArmUpDown THEN
131      IF hazardWarningSwitchOn = switch_off & engineOn = TRUE
           THEN
132         SET_BlinkersOn(pitman_direction(newPos), -1)
133      END;
134      SET_Pitman_DirectionBlinking(newPos)
```

```
135   END;
136
137   ENV_Pitman_Reset_to_Neutral =
138   // ELS-1, ELS-5
139   BEGIN
140      SET_Pitman_Reset_to_Neutral;
141      IF hazardWarningSwitchOn = switch_off & remaining_blinks =
             -1 THEN
142         SET_AllBlinkersOff
143      END
144   END;
145
146
147
148   ENV_Hazard_blinking(newSwitchPos) = SELECT newSwitchPos :
         SWITCH_STATUS & newSwitchPos /= hazardWarningSwitchOn THEN
149   // ELS-1, ELS-5
150      IF newSwitchPos = switch_on // hazardWarningSwitchOn =
           switch_off
151      THEN
152         SET_AllBlinkersOn
153       ELSIF newSwitchPos = switch_off // hazardWarningSwitchOn
            = switch_on
154       THEN
155         IF pitmanArmUpDown = Neutral or engineOn = FALSE THEN
156            SET_AllBlinkersOff
157         ELSIF  pitmanArmUpDown /: PITMAN_DIRECTION_BLINKING
              THEN
158            // ELS-12 : When hazard warning is deactivated again
                 , the pitman arm is in
159            //  position \direction blinking left" or \direction
                  blinking right" ignition is On,
160            //  the direction blinking cycle should be started (
                 see Req. ELS-1).
161            SET_AllBlinkersOff
162         ELSE
163            SET_BlinkersOn(pitman_direction(pitmanArmUpDown),
                 remaining_blinks) // remaining_blinks must be ≠ 0
164         END
165      END;
166      SET_Hazard_blinking(newSwitchPos)
167   END
168
169 END
```

Listing 4: PitmanController_TIME_MC_v4 MCH Code

## 3.2 Model Checking

| Modelchecking Item | Modelchecking Result |
|---|---|
| Gemischte Breiten-/Tiefensuche, Invarianten überprüfen | Modelchecking not solved |
| Gemischte Breiten-/Tiefensuche, Deadlocks finden | Modelchecking not solved |

Table 7: Modelchecking Items and Results

## 3.3 LTL Model Checking

| LTL Formular | Status |
|---|---|
| 1 G{engineOn = TRUE} | Formular not Solved |

Table 8: LTL Formulars and Results

| Pattern Name | Code | Result |
|---|---|---|

Table 9: LTL Patterns and Results

## 3.4 Traces

Table 10: PitmanController_TIME_v3_Trace7

| | Position | Transition |
|---|---|---|
| 0 | | SETUP_CONSTANTS |
| 1 | | INITIALISATION |
| 2 | | RTIME_Nothing |
| 3 | | RTIME_Nothing |
| 4 | | RTIME_Nothing |
| 5 | | RTIME_Nothing |
| 6 | | RTIME_Nothing |
| 7 | | RTIME_Nothing |
| 8 | | RTIME_Nothing |

| Position | Transition |
|---|---|
| 9 | RTIME_Nothing |
| 10 | RTIME_Nothing |
| 11 | RTIME_Nothing |
| 12 | RTIME_Nothing |
| 13 | RTIME_Nothing |
| 14 | RTIME_Nothing |
| 15 | RTIME_Nothing |
| 16 | RTIME_Nothing |
| 17 | RTIME_Nothing |
| 18 | ENV_Turn_EngineOn |
| 19 | RTIME_Nothing |
| 20 | RTIME_Nothing |
| 21 | RTIME_Nothing |
| 22 | RTIME_Nothing |
| 23 | RTIME_Nothing |
| 24 | RTIME_Nothing |
| 25 | RTIME_Nothing |
| 26 | RTIME_Nothing |
| 27 | RTIME_Nothing |
| 28 | RTIME_Nothing |
| 29 | RTIME_Nothing |
| 30 | RTIME_Nothing |
| 31 | RTIME_Nothing |
| 32 | RTIME_Nothing |
| 33 | RTIME_Nothing |
| 34 | RTIME_Nothing |
| 35 | RTIME_Nothing |
| 36 | RTIME_Nothing |
| 37 | RTIME_Nothing |
| 38 | RTIME_Nothing |
| 39 | ENV_Pitman_DirectionBlinking |

| Position | Transition |
|---|---|
| 40 | RTIME_BlinkerOn |
| 41 | RTIME_BlinkerOff |
| 42 | RTIME_BlinkerOn |
| 43 | RTIME_Passes |
| 44 | RTIME_Passes |
| 45 | ENV_Pitman_Reset_to_Neutral |
| 46 | RTIME_Nothing |
| 47 | RTIME_Nothing |
| 48 | RTIME_Nothing |
| 49 | RTIME_Nothing |
| 50 | RTIME_Nothing |
| 51 | RTIME_Nothing |
| 52 | RTIME_Nothing |
| 53 | RTIME_Nothing |
| 54 | RTIME_Nothing |
| 55 | RTIME_Nothing |
| 56 | RTIME_Nothing |
| 57 | RTIME_Nothing |
| 58 | RTIME_Nothing |
| 59 | RTIME_Nothing |
| 60 | RTIME_Nothing |
| 61 | RTIME_Nothing |
| 62 | RTIME_Nothing |
| 63 | RTIME_Nothing |
| 64 | RTIME_Nothing |
| 65 | RTIME_Nothing |
| 66 | RTIME_Nothing |
| 67 | RTIME_Nothing |
| 68 | RTIME_Nothing |
| 69 | RTIME_Nothing |
| 70 | RTIME_Nothing |

| Position | Transition |
|---|---|
| 71 | RTIME_Nothing |
| 72 | RTIME_Nothing |
| 73 | RTIME_Nothing |
| 74 | RTIME_Nothing |
| 75 | RTIME_Nothing |
| 76 | RTIME_Nothing |
| 77 | RTIME_Nothing |
| 78 | RTIME_Nothing |
| 79 | RTIME_Nothing |
| 80 | RTIME_Nothing |
| 81 | RTIME_Nothing |
| 82 | RTIME_Nothing |
| 83 | RTIME_Nothing |
| 84 | RTIME_Nothing |
| 85 | RTIME_Nothing |
| 86 | RTIME_Nothing |
| 87 | RTIME_Nothing |
| 88 | RTIME_Nothing |
| 89 | RTIME_Nothing |
| 90 | RTIME_Nothing |
| 91 | ENV_Pitman_Tip_blinking_start |
| 92 | RTIME_BlinkerOn |
| 93 | RTIME_Passes |
| 94 | ENV_Pitman_Reset_to_Neutral |
| 95 | RTIME_Tip_blinking_Timeout |
| 96 | RTIME_Passes |
| 97 | RTIME_BlinkerOff |
| 98 | RTIME_Passes |
| 99 | RTIME_Passes |
| 100 | ENV_Hazard_blinking |
| 101 | RTIME_BlinkerOn |

| Position | Transition |
|---|---|
| 102 | RTIME_BlinkerOff |
| 103 | RTIME_BlinkerOn |
| 104 | RTIME_BlinkerOff |
| 105 | RTIME_BlinkerOn |
| 106 | RTIME_Passes |
| 107 | RTIME_Passes |
| 108 | ENV_Pitman_Tip_blinking_start |
| 109 | RTIME_Passes |
| 110 | ENV_Pitman_Reset_to_Neutral |
| 111 | RTIME_BlinkerOff |
| 112 | RTIME_Tip_blinking_Timeout |
| 113 | RTIME_BlinkerOn |
| 114 | RTIME_Passes |
| 115 | RTIME_Passes |
| 116 | ENV_Hazard_blinking |
| 117 | RTIME_Nothing |
| 118 | RTIME_Nothing |
| 119 | RTIME_Nothing |
| 120 | RTIME_Nothing |
| 121 | RTIME_Nothing |