

Contents

CONTEXT cruise_control	2
MACHINE M0	3

CONTEXT cruise_control

SETS

BEFORECC

CONSTANTS

pa

po

undefined

AXIOMS

axm1: $partition(BEFORECC, \{pa\}, \{po\}, \{undefined\})$

END

MACHINE M0**SEES** cruise_control**VARIABLES**

PO
 PA
 BRAKE
 CC
 PC
 engrun
 beforecc

INVARIANTS

typeof_PO: $PO \in \text{BOOL}$
typeof_PA: $PA \in \text{BOOL}$
typeof_BRAKE: $BRAKE \in \text{BOOL}$
typeof_CC: $CC \in \text{BOOL}$
typeof_PC: $PC \in \text{BOOL}$
distinct_states_in_iUML: $\text{TRUE} \in \{PO, PA, BRAKE, CC, PC\} \Rightarrow \text{partition}(\{\text{TRUE}\}, \{PO\} \cap \{\text{TRUE}\}, \{PA\} \cap \{\text{TRUE}\}, \{BRAKE\} \cap \{\text{TRUE}\}, \{CC\} \cap \{\text{TRUE}\}, \{PC\} \cap \{\text{TRUE}\})$
inv1: $\text{engrun} \in \text{BOOL}$
inv2: $\text{beforecc} \subseteq \text{BEFORECC}$

EVENTS**Initialisation****begin**

init_PO: $PO := \text{TRUE}$
init_PA: $PA := \text{FALSE}$
init_BRAKE: $BRAKE := \text{FALSE}$
init_CC: $CC := \text{FALSE}$
init_PC: $PC := \text{FALSE}$
act1: $\text{engrun} := \text{FALSE}$
act2: $\text{beforecc} := \{\text{undefined}\}$

end**Event** PedalOnly $\langle \text{ordinary} \rangle \hat{=}$ **when****isin_PA_or_isin_CC:** $(PA = \text{TRUE} \vee CC = \text{TRUE})$ **then**

leave_PA: $PA := \text{FALSE}$
leave_CC: $CC := \text{FALSE}$
enter_PO: $PO := \text{TRUE}$
act1: $\text{engrun} := \text{FALSE}$

end**Event** PedalAssist $\langle \text{ordinary} \rangle \hat{=}$ **when****isin_PO_or_isin_CC:** $(PO = \text{TRUE} \vee CC = \text{TRUE})$ **then**

leave_PO: $PO := \text{FALSE}$
leave_CC: $CC := \text{FALSE}$
enter_PA: $PA := \text{TRUE}$
act1: $\text{engrun} := \text{TRUE}$

end**Event** Brake $\langle \text{ordinary} \rangle \hat{=}$ **when****isin_PA_or_isin_PC_or_isin_PO:** $(PA = \text{TRUE} \vee PC = \text{TRUE} \vee PO = \text{TRUE})$ **then**

leave_PA: $PA := \text{FALSE}$
leave_PC: $PC := \text{FALSE}$

```

    leave_P0: PO := FALSE
    enter_BRAKE: BRAKE := TRUE
    act1: engrun := FALSE
end
Event Stop ⟨ordinary⟩ ≐
  when
    isin_BRAKE: BRAKE = TRUE
  then
    leave_BRAKE: BRAKE := FALSE
    act1: engrun := FALSE
  end
Event PedalCharge ⟨ordinary⟩ ≐
  when
    isin_P0: PO = TRUE
  then
    leave_P0: PO := FALSE
    enter_PC: PC := TRUE
    act1: engrun := TRUE
  end
Event PedalOnly2CruiseControl ⟨ordinary⟩ ≐
  when
    isin_P0: PO = TRUE
  then
    leave_P0: PO := FALSE
    enter_CC: CC := TRUE
    act1: engrun := TRUE
    act2: beforecc := {po}
  end
Event PedalAssist2CruiseControl ⟨ordinary⟩ ≐
  when
    isin_PA: PA = TRUE
  then
    leave_PA: PA := FALSE
    enter_CC: CC := TRUE
    act1: engrun := TRUE
    act2: beforecc := {pa}
  end
Event BrakeCruiseControl2PedalAssist ⟨ordinary⟩ ≐
  when
    isin_CC: CC = TRUE
    grd1: beforecc = {pa}
  then
    leave_CC: CC := FALSE
    enter_PA: PA := TRUE
    act1: engrun := TRUE
  end
Event BrakeCruiseControl2PedalOnly ⟨ordinary⟩ ≐
  when
    isin_CC: CC = TRUE
    grd1: beforecc = {po}
  then
    leave_CC: CC := FALSE
    enter_P0: PO := TRUE
    act1: engrun := FALSE
  end
END

```