

section *sf_car_shift_logic* **parents** *stateflow_toolkit, calc_th*
s_downshifting, s_gear_state, s_fourth, s_second, s_third, s_first,
s_selection_state, s_upshifting, s_steady_state, c_shift_logic : SID

t_third_fourth, t_second_third, t_first_second, t_fourth_third, t_default_first,
t_second_first, t_third_second, t_steady_state_upshifting, t_default_steady_state,
t_upshifting_steady_state23, t_steady_state_downshifting,
t_downshifting_steady_state25, t_downshifting_steady_state24,
t_upshifting_steady_state26 : TID

C_shift_logic : State

C_shift_logic =

$$\left(\begin{array}{l} \langle \text{identifier} == c_shift_logic, \text{default} == \text{nulltransition.identifier}, \\ \text{inner} == \text{nulltransition.identifier}, \text{outer} == \text{nulltransition.identifier}, \\ \text{parent} == \text{nullstate.identifier}, \text{left} == \text{nullstate.identifier}, \\ \text{right} == \text{nullstate.identifier}, \text{substates} == \langle s_gear_state, s_selection_state \rangle, \\ \text{decomposition} == SET, \text{type} == CHART, \text{history} == \mathbf{False} \rangle \end{array} \right)$$

S_downshifting : State

S_downshifting =

$$\left(\begin{array}{l} \langle \text{identifier} == s_downshifting, \text{default} == \text{nulltransition.identifier}, \\ \text{inner} == \text{nulltransition.identifier}, \text{outer} == t_downshifting_steady_state24, \\ \text{parent} == s_selection_state, \text{left} == \text{nullstate.identifier}, \\ \text{right} == \text{nullstate.identifier}, \text{substates} == \langle \rangle, \\ \text{decomposition} == CLUSTER, \text{type} == OR, \text{history} == \mathbf{False} \rangle \end{array} \right)$$

S_gear_state : State

S_gear_state =

$$\left(\begin{array}{l} \langle \text{identifier} == s_gear_state, \text{default} == t_default_first, \\ \text{inner} == \text{nulltransition.identifier}, \text{outer} == \text{nulltransition.identifier}, \\ \text{parent} == c_shift_logic, \text{left} == \text{nullstate.identifier}, \\ \text{right} == s_selection_state, \text{substates} == \langle s_fourth, s_third, s_first, s_second \rangle, \\ \text{decomposition} == CLUSTER, \text{type} == AND, \text{history} == \mathbf{False} \rangle \end{array} \right)$$

S_fourth : State

S_fourth =

$$\left(\begin{array}{l} \langle \text{identifier} == s_fourth, \text{default} == \text{nulltransition.identifier}, \\ \text{inner} == \text{nulltransition.identifier}, \text{outer} == t_fourth_third, \\ \text{parent} == s_gear_state, \text{left} == \text{nullstate.identifier}, \\ \text{right} == \text{nullstate.identifier}, \text{substates} == \langle \rangle, \\ \text{decomposition} == CLUSTER, \text{type} == OR, \text{history} == \mathbf{False} \rangle \end{array} \right)$$

S_second : State

S_second =

$$\left(\begin{array}{l} \langle \text{identifier} == s_second, \text{default} == \text{nulltransition.identifier}, \\ \text{inner} == \text{nulltransition.identifier}, \text{outer} == t_second_third, \\ \text{parent} == s_gear_state, \text{left} == \text{nullstate.identifier}, \\ \text{right} == \text{nullstate.identifier}, \text{substates} == \langle \rangle, \\ \text{decomposition} == CLUSTER, \text{type} == OR, \text{history} == \mathbf{False} \rangle \end{array} \right)$$

S_third : State

$$S_third = \left(\begin{array}{l} \langle identifier == s_third, default == nulltransition.identifier, \\ inner == nulltransition.identifier, outer == t_third_fourth, \\ parent == s_gear_state, left == nullstate.identifier, \\ right == nullstate.identifier, substates == \langle \rangle, \\ decomposition == CLUSTER, type == OR, history == \mathbf{False} \rangle \end{array} \right)$$

S_first : State

$$S_first = \left(\begin{array}{l} \langle identifier == s_first, default == nulltransition.identifier, \\ inner == nulltransition.identifier, outer == t_first_second, \\ parent == s_gear_state, left == nullstate.identifier, \\ right == nullstate.identifier, substates == \langle \rangle, \\ decomposition == CLUSTER, type == OR, history == \mathbf{False} \rangle \end{array} \right)$$

S_selection_state : State

$$S_selection_state = \left(\begin{array}{l} \langle identifier == s_selection_state, default == t_default_steady_state, \\ inner == nulltransition.identifier, outer == nulltransition.identifier, \\ parent == c_shift_logic, left == s_gear_state, right == nullstate.identifier, \\ substates == \langle s_upshifting, s_downshifting, s_steady_state \rangle, \\ decomposition == CLUSTER, type == AND, history == \mathbf{False} \rangle \end{array} \right)$$

S_upshifting : State

$$S_upshifting = \left(\begin{array}{l} \langle identifier == s_upshifting, default == nulltransition.identifier, \\ inner == nulltransition.identifier, outer == t_upshifting_steady_state26, \\ parent == s_selection_state, left == nullstate.identifier, \\ right == nullstate.identifier, substates == \langle \rangle, \\ decomposition == CLUSTER, type == OR, history == \mathbf{False} \rangle \end{array} \right)$$

S_steady_state : State

$$S_steady_state = \left(\begin{array}{l} \langle identifier == s_steady_state, default == nulltransition.identifier, \\ inner == nulltransition.identifier, outer == t_steady_state_upshifting, \\ parent == s_selection_state, left == nullstate.identifier, \\ right == nullstate.identifier, substates == \langle \rangle, \\ decomposition == CLUSTER, type == OR, history == \mathbf{False} \rangle \end{array} \right)$$

T_third_fourth : Transition

$$T_third_fourth = \left(\begin{array}{l} \langle identifier == t_third_fourth, source == snode(s_third), \\ destination == snode(s_fourth), next == t_third_second, \\ parent == s_gear_state \rangle \end{array} \right)$$

T_second_third : Transition

$$T_second_third = \left(\begin{array}{l} \langle identifier == t_second_third, source == snode(s_second), \\ destination == snode(s_third), next == t_second_first, \\ parent == s_gear_state \rangle \end{array} \right)$$

$T_first_second : Transition$

$T_first_second =$
 $\left(\begin{array}{l} \langle identifier == t_first_second, source == snode(s_first), \\ destination == snode(s_second), next == nulltransition.identifier, \\ parent == s_gear_state \rangle \end{array} \right)$

$T_fourth_third : Transition$

$T_fourth_third =$
 $\left(\begin{array}{l} \langle identifier == t_fourth_third, source == snode(s_fourth), \\ destination == snode(s_third), next == nulltransition.identifier, \\ parent == s_gear_state \rangle \end{array} \right)$

$T_default_first : Transition$

$T_default_first =$
 $\left(\begin{array}{l} \langle identifier == t_default_first, source == snode(nullstate.identifier), \\ destination == snode(s_first), next == nulltransition.identifier, \\ parent == s_gear_state \rangle \end{array} \right)$

$T_second_first : Transition$

$T_second_first =$
 $\left(\begin{array}{l} \langle identifier == t_second_first, source == snode(s_second), \\ destination == snode(s_first), next == nulltransition.identifier, \\ parent == s_gear_state \rangle \end{array} \right)$

$T_third_second : Transition$

$T_third_second =$
 $\left(\begin{array}{l} \langle identifier == t_third_second, source == snode(s_third), \\ destination == snode(s_second), next == nulltransition.identifier, \\ parent == s_gear_state \rangle \end{array} \right)$

$T_steady_state_upshifting : Transition$

$T_steady_state_upshifting =$
 $\left(\begin{array}{l} \langle identifier == t_steady_state_upshifting, source == snode(s_steady_state), \\ destination == snode(s_upshifting), next == t_steady_state_downshifting, \\ parent == s_selection_state \rangle \end{array} \right)$

$T_default_steady_state : Transition$

$T_default_steady_state =$
 $\left(\begin{array}{l} \langle identifier == t_default_steady_state, source == snode(nullstate.identifier), \\ destination == snode(s_steady_state), next == nulltransition.identifier, \\ parent == s_selection_state \rangle \end{array} \right)$

$T_upshifting_steady_state23 : Transition$

$T_upshifting_steady_state23 =$
 $\left(\begin{array}{l} \langle identifier == t_upshifting_steady_state23, source == snode(s_upshifting), \\ destination == snode(s_steady_state), next == nulltransition.identifier, \\ parent == s_selection_state \rangle \end{array} \right)$

T_steady_state_downshifting : Transition

T_steady_state_downshifting =
(
 ⟨*identifier* == *t_steady_state_downshifting*, *source* == *snode*(*s_steady_state*),
 destination == *snode*(*s_downshifting*), *next* == *nulltransition.identifier*,
 parent == *s_selection_state*⟩
)

T_downshifting_steady_state25 : Transition

T_downshifting_steady_state25 =
(
 ⟨*identifier* == *t_downshifting_steady_state25*, *source* == *snode*(*s_downshifting*),
 destination == *snode*(*s_steady_state*), *next* == *nulltransition.identifier*,
 parent == *s_selection_state*⟩
)

T_downshifting_steady_state24 : Transition

T_downshifting_steady_state24 =
(
 ⟨*identifier* == *t_downshifting_steady_state24*, *source* == *snode*(*s_downshifting*),
 destination == *snode*(*s_steady_state*), *next* == *t_downshifting_steady_state25*,
 parent == *s_selection_state*⟩
)

T_upshifting_steady_state26 : Transition

T_upshifting_steady_state26 =
(
 ⟨*identifier* == *t_upshifting_steady_state26*, *source* == *snode*(*s_upshifting*),
 destination == *snode*(*s_steady_state*), *next* == *t_upshifting_steady_state23*,
 parent == *s_selection_state*⟩
)

e_UP, *e_DOWN*, *ENULL* : EVENT

channel *o_gear* : ℝ; *i_speed* : ℝ; *i_throttle* : ℝ

process *P_shift_logic* ≐ begin

StateflowChart

identifier = *c_shift_logic*
states = {(*c_shift_logic*, *C_shift_logic*), (*s_downshifting*, *S_downshifting*),
 (*s_gear_state*, *S_gear_state*), (*s_fourth*, *S_fourth*), (*s_second*, *S_second*),
 (*s_third*, *S_third*), (*s_first*, *S_first*), (*s_selection_state*, *S_selection_state*),
 (*s_upshifting*, *S_upshifting*), (*s_steady_state*, *S_steady_state*)}
transitions = {(*t_third_fourth*, *T_third_fourth*), (*t_second_third*, *T_second_third*),
 (*t_first_second*, *T_first_second*), (*t_fourth_third*, *T_fourth_third*),
 (*t_default_first*, *T_default_first*), (*t_second_first*, *T_second_first*),
 (*t_third_second*, *T_third_second*),
 (*t_steady_state_upshifting*, *T_steady_state_upshifting*),
 (*t_default_steady_state*, *T_default_steady_state*),
 (*t_upshifting_steady_state23*, *T_upshifting_steady_state23*),
 (*t_steady_state_downshifting*, *T_steady_state_downshifting*),
 (*t_downshifting_steady_state25*, *T_downshifting_steady_state25*),
 (*t_downshifting_steady_state24*, *T_downshifting_steady_state24*),
 (*t_upshifting_steady_state26*, *T_upshifting_steady_state26*)}
junctions = {}

$SimulationInstance$ $v_gear, v_up_th, v_speed, v_down_th, v_throttle : \mathbb{R}$

$InitSimulationInstance$ $SimulationInstance'$ $v_gear' = 0$ $v_up_th' = 0$ $v_speed' = 0$ $v_down_th' = 0$ $v_throttle' = 0$
--

$SimulationData$ $state_status : SID \rightarrow \mathbb{B}$ $state_history : SID \rightarrow SID$ $dom\ state_status = dom\ states$ $dom\ state_history = \{j : ran\ junctions \mid j.history = \mathbf{True} \bullet j.parent\}$ $\forall s : ran\ states \mid s.decomposition = CLUSTER \bullet$ $\#\{ss : ran\ s.substates \mid state_status(ss) = \mathbf{True}\} \leq 1$

$InitSimulationData$ $SimulationData'$ $state_status' = \{n : dom\ states \bullet n \mapsto \mathbf{False}\}$ $state_history' = \{n : dom\ state_history' \bullet n \mapsto nullstate.identifier\}$

$ActivateNoHistory$ $\Delta SimulationData$ $x? : SID$ $x? \in dom\ state_status$ $(parent\ (states\ x?)).history = \mathbf{False}$ $state_history' = state_history$ $state_status' = state_status \oplus \{x? \mapsto \mathbf{True}\}$
--

$ActivateWithHistory$ $\Delta SimulationData$ $x? : SID$ $x? \in dom\ state_status$ $(parent\ (states\ x?)).history = \mathbf{True}$ $state_history' = state_history \oplus \{(states\ x?).parent \mapsto x?\}$ $state_status' = state_status \oplus \{x? \mapsto \mathbf{True}\}$

$Activate == (ActivateWithHistory \vee ActivateNoHistory) \wedge \exists SimulationInstance$

$Deactivate$ $\Delta SimulationData$ $\exists SimulationInstance$ $x? : SID$
$x? \in \text{dom } state_status$ $state_history' = state_history$ $state_status' = state_status \oplus \{x? \mapsto \mathbf{False}\}$

$InitState == (InitSimulationInstance) \wedge (InitSimulationData)$

$\mathbf{state } shift_logic_state == (SimulationInstance) \wedge (SimulationData)$

$entryaction_downshifting \hat{=} (executeentryaction.(s_downshifting) \longrightarrow \mathbf{Skip})$
 $entryaction_gear_state \hat{=} (executeentryaction.(s_gear_state) \longrightarrow \mathbf{Skip})$
 $entryaction_fourth \hat{=} (executeentryaction.(s_fourth) \longrightarrow (v_gear := 4 ; \mathbf{Skip}))$
 $entryaction_second \hat{=} (executeentryaction.(s_second) \longrightarrow (v_gear := 2 ; \mathbf{Skip}))$
 $entryaction_third \hat{=} (executeentryaction.(s_third) \longrightarrow (v_gear := 3 ; \mathbf{Skip}))$
 $entryaction_first \hat{=} (executeentryaction.(s_first) \longrightarrow (v_gear := 1 ; \mathbf{Skip}))$
 $entryaction_selection_state \hat{=} (executeentryaction.(s_selection_state) \longrightarrow \mathbf{Skip})$
 $entryaction_upshifting \hat{=} (executeentryaction.(s_upshifting) \longrightarrow \mathbf{Skip})$
 $entryaction_steady_state \hat{=} (executeentryaction.(s_steady_state) \longrightarrow \mathbf{Skip})$
 $entryactions \hat{=} \left(\begin{array}{l} entryaction_downshifting \square entryaction_gear_state \square \\ entryaction_fourth \square entryaction_second \square entryaction_third \square \\ entryaction_first \square entryaction_selection_state \square \\ entryaction_upshifting \square entryaction_steady_state \end{array} \right)$

$duringaction_downshifting \hat{=} (executeduringaction.(s_downshifting)?ce \longrightarrow \mathbf{Skip})$
 $duringaction_gear_state \hat{=} (executeduringaction.(s_gear_state)?ce \longrightarrow \mathbf{Skip})$
 $duringaction_fourth \hat{=} (executeduringaction.(s_fourth)?ce \longrightarrow \mathbf{Skip})$
 $duringaction_second \hat{=} (executeduringaction.(s_second)?ce \longrightarrow \mathbf{Skip})$
 $duringaction_third \hat{=} (executeduringaction.(s_third)?ce \longrightarrow \mathbf{Skip})$
 $duringaction_first \hat{=} (executeduringaction.(s_first)?ce \longrightarrow \mathbf{Skip})$

$duringaction_selection_state \hat{=} executeduringaction.(s_selection_state)?ce \longrightarrow$
 $\left(\begin{array}{l} \mathbf{var } _aux : \mathbb{R} \times \mathbb{R} \bullet _aux := \text{calc_th}(v_gear, v_throttle); \\ v_down_th := _aux.1 ; v_up_th := _aux.2 \end{array} \right)$
 $duringaction_upshifting \hat{=} (executeduringaction.(s_upshifting)?ce \longrightarrow \mathbf{Skip})$
 $duringaction_steady_state \hat{=} (executeduringaction.(s_steady_state)?ce \longrightarrow \mathbf{Skip})$
 $duringactions \hat{=} \left(\begin{array}{l} duringaction_downshifting \square duringaction_gear_state \square \\ duringaction_fourth \square duringaction_second \square \\ duringaction_third \square duringaction_first \square \\ duringaction_selection_state \square duringaction_upshifting \square \\ duringaction_steady_state \end{array} \right)$

$$\begin{aligned}
& \textit{exitaction_downshifting} \hat{=} (\textit{executeexitaction}.(s_downshifting) \longrightarrow \mathbf{Skip}) \\
& \textit{exitaction_gear_state} \hat{=} (\textit{executeexitaction}.(s_gear_state) \longrightarrow \mathbf{Skip}) \\
& \textit{exitaction_fourth} \hat{=} (\textit{executeexitaction}.(s_fourth) \longrightarrow \mathbf{Skip}) \\
& \textit{exitaction_second} \hat{=} (\textit{executeexitaction}.(s_second) \longrightarrow \mathbf{Skip}) \\
& \textit{exitaction_third} \hat{=} (\textit{executeexitaction}.(s_third) \longrightarrow \mathbf{Skip}) \\
& \textit{exitaction_first} \hat{=} (\textit{executeexitaction}.(s_first) \longrightarrow \mathbf{Skip}) \\
& \textit{exitaction_selection_state} \hat{=} (\textit{executeexitaction}.(s_selection_state) \longrightarrow \mathbf{Skip}) \\
& \textit{exitaction_upshifting} \hat{=} (\textit{executeexitaction}.(s_upshifting) \longrightarrow \mathbf{Skip}) \\
& \textit{exitaction_steady_state} \hat{=} (\textit{executeexitaction}.(s_steady_state) \longrightarrow \mathbf{Skip}) \\
& \textit{exitactions} \hat{=} \left(\begin{array}{l} \textit{exitaction_downshifting} \sqcap \textit{exitaction_gear_state} \sqcap \\ \textit{exitaction_fourth} \sqcap \textit{exitaction_second} \sqcap \\ \textit{exitaction_third} \sqcap \textit{exitaction_first} \sqcap \\ \textit{exitaction_selection_state} \sqcap \textit{exitaction_upshifting} \sqcap \\ \textit{exitaction_steady_state} \end{array} \right)
\end{aligned}$$

$$\begin{aligned}
& \textit{conditionaction_third_fourth} \hat{=} (\textit{executeconditionaction}.(t_third_fourth) \longrightarrow \mathbf{Skip}) \\
& \textit{conditionaction_second_third} \hat{=} (\textit{executeconditionaction}.(t_second_third) \longrightarrow \mathbf{Skip}) \\
& \textit{conditionaction_first_second} \hat{=} (\textit{executeconditionaction}.(t_first_second) \longrightarrow \mathbf{Skip}) \\
& \textit{conditionaction_fourth_third} \hat{=} (\textit{executeconditionaction}.(t_fourth_third) \longrightarrow \mathbf{Skip}) \\
& \textit{conditionaction_default_first} \hat{=} (\textit{executeconditionaction}.(t_default_first) \longrightarrow \mathbf{Skip}) \\
& \textit{conditionaction_second_first} \hat{=} (\textit{executeconditionaction}.(t_second_first) \longrightarrow \mathbf{Skip}) \\
& \textit{conditionaction_third_second} \hat{=} (\textit{executeconditionaction}.(t_third_second) \longrightarrow \mathbf{Skip})
\end{aligned}$$

$$\begin{aligned}
& \text{conditionaction_steady_state_upshifting} \hat{=} \\
& \quad (\text{executeconditionaction}.(t_steady_state_upshifting) \longrightarrow \mathbf{Skip}) \\
& \text{conditionaction_default_steady_state} \hat{=} \\
& \quad (\text{executeconditionaction}.(t_default_steady_state) \longrightarrow \mathbf{Skip}) \\
& \text{conditionaction_upshifting_steady_state23} \hat{=} \\
& \quad (\text{executeconditionaction}.(t_upshifting_steady_state23) \longrightarrow \mathbf{Skip}) \\
& \text{conditionaction_steady_state_downshifting} \hat{=} \\
& \quad (\text{executeconditionaction}.(t_steady_state_downshifting) \longrightarrow \mathbf{Skip}) \\
& \text{conditionaction_downshifting_steady_state25} \hat{=} \\
& \quad \text{executeconditionaction}.(t_downshifting_steady_state25) \longrightarrow \\
& \quad \left(\begin{array}{l} \mathbf{var_b} : \mathbb{B} \bullet \text{broadcast}(e_DOWN, s_gear_state); \text{check}(_b); \\ \left(\begin{array}{l} \mathbf{if_b} = \mathbf{True} \longrightarrow \mathbf{Skip} \\ \square \neg (_b = \mathbf{True}) \longrightarrow \mathbf{Skip} \\ \mathbf{fi} \end{array} \right) \end{array} \right) \\
& \text{conditionaction_downshifting_steady_state24} \hat{=} \\
& \quad (\text{executeconditionaction}.(t_downshifting_steady_state24) \longrightarrow \mathbf{Skip}) \\
& \text{conditionaction_upshifting_steady_state26} \hat{=} \\
& \quad \text{executeconditionaction}.(t_upshifting_steady_state26) \longrightarrow \\
& \quad \left(\begin{array}{l} \mathbf{var_b} : \mathbb{B} \bullet \text{broadcast}(e_UP, s_gear_state); \text{check}(_b); \\ \left(\begin{array}{l} \mathbf{if_b} = \mathbf{True} \longrightarrow \mathbf{Skip} \\ \square \neg (_b = \mathbf{True}) \longrightarrow (\mathbf{Skip}) \\ \mathbf{fi} \end{array} \right) \end{array} \right) \\
& \text{conditionactions} \hat{=} \left(\begin{array}{l} \text{conditionaction_third_fourth} \square \text{conditionaction_second_third} \square \\ \text{conditionaction_first_second} \square \text{conditionaction_fourth_third} \square \\ \text{conditionaction_default_first} \square \text{conditionaction_second_first} \square \\ \text{conditionaction_third_second} \square \\ \text{conditionaction_steady_state_upshifting} \square \\ \text{conditionaction_default_steady_state} \square \\ \text{conditionaction_upshifting_steady_state23} \square \\ \text{conditionaction_steady_state_downshifting} \square \\ \text{conditionaction_downshifting_steady_state25} \square \\ \text{conditionaction_downshifting_steady_state24} \square \\ \text{conditionaction_upshifting_steady_state26} \end{array} \right)
\end{aligned}$$

$$\begin{aligned}
& \text{transitionaction_third_fourth} \hat{=} (\text{executetransitionaction}.(t_third_fourth) \longrightarrow \mathbf{Skip}) \\
& \text{transitionaction_second_third} \hat{=} (\text{executetransitionaction}.(t_second_third) \longrightarrow \mathbf{Skip}) \\
& \text{transitionaction_first_second} \hat{=} (\text{executetransitionaction}.(t_first_second) \longrightarrow \mathbf{Skip}) \\
& \text{transitionaction_fourth_third} \hat{=} (\text{executetransitionaction}.(t_fourth_third) \longrightarrow \mathbf{Skip}) \\
& \text{transitionaction_default_first} \hat{=} (\text{executetransitionaction}.(t_default_first) \longrightarrow \mathbf{Skip}) \\
& \text{transitionaction_second_first} \hat{=} (\text{executetransitionaction}.(t_second_first) \longrightarrow \mathbf{Skip}) \\
& \text{transitionaction_third_second} \hat{=} (\text{executetransitionaction}.(t_third_second) \longrightarrow \mathbf{Skip}) \\
& \text{transitionaction_steady_state_upshifting} \hat{=} \\
& \quad (\text{executetransitionaction}.(t_steady_state_upshifting) \longrightarrow \mathbf{Skip}) \\
& \text{transitionaction_default_steady_state} \hat{=} \\
& \quad (\text{executetransitionaction}.(t_default_steady_state) \longrightarrow \mathbf{Skip}) \\
& \text{transitionaction_upshifting_steady_state23} \hat{=} \\
& \quad (\text{executetransitionaction}.(t_upshifting_steady_state23) \longrightarrow \mathbf{Skip}) \\
& \text{transitionaction_steady_state_downshifting} \hat{=} \\
& \quad (\text{executetransitionaction}.(t_steady_state_downshifting) \longrightarrow \mathbf{Skip}) \\
& \text{transitionaction_downshifting_steady_state25} \hat{=} \\
& \quad (\text{executetransitionaction}.(t_downshifting_steady_state25) \longrightarrow \mathbf{Skip}) \\
& \text{transitionaction_downshifting_steady_state24} \hat{=} \\
& \quad (\text{executetransitionaction}.(t_downshifting_steady_state24) \longrightarrow \mathbf{Skip}) \\
& \text{transitionaction_upshifting_steady_state26} \hat{=} \\
& \quad (\text{executetransitionaction}.(t_upshifting_steady_state26) \longrightarrow \mathbf{Skip}) \\
& \text{transitionactions} \hat{=} \left(\begin{array}{l}
\text{transitionaction_third_fourth} \square \text{transitionaction_second_third} \square \\
\text{transitionaction_first_second} \square \text{transitionaction_fourth_third} \square \\
\text{transitionaction_default_first} \square \text{transitionaction_second_first} \square \\
\text{transitionaction_third_second} \square \\
\text{transitionaction_steady_state_upshifting} \square \\
\text{transitionaction_default_steady_state} \square \\
\text{transitionaction_upshifting_steady_state23} \square \\
\text{transitionaction_steady_state_downshifting} \square \\
\text{transitionaction_downshifting_steady_state25} \square \\
\text{transitionaction_downshifting_steady_state24} \square \\
\text{transitionaction_upshifting_steady_state26}
\end{array} \right)
\end{aligned}$$

$condition_third_fourth \hat{=} (evaluatecondition.(t_third_fourth)!(\mathbf{True}) \longrightarrow \mathbf{Skip})$
 $condition_second_third \hat{=} (evaluatecondition.(t_second_third)!(\mathbf{True}) \longrightarrow \mathbf{Skip})$
 $condition_first_second \hat{=} (evaluatecondition.(t_first_second)!(\mathbf{True}) \longrightarrow \mathbf{Skip})$
 $condition_fourth_third \hat{=} (evaluatecondition.(t_fourth_third)!(\mathbf{True}) \longrightarrow \mathbf{Skip})$
 $condition_default_first \hat{=} (evaluatecondition.(t_default_first)!(\mathbf{True}) \longrightarrow \mathbf{Skip})$
 $condition_second_first \hat{=} (evaluatecondition.(t_second_first)!(\mathbf{True}) \longrightarrow \mathbf{Skip})$
 $condition_third_second \hat{=} (evaluatecondition.(t_third_second)!(\mathbf{True}) \longrightarrow \mathbf{Skip})$
 $condition_steady_state_upshifting \hat{=}$

$$\left(\begin{array}{l} \mathbf{if}((v_speed >_{\mathcal{A}} v_up_th) \neq 0) \longrightarrow \\ \quad evaluatecondition.(t_steady_state_upshifting)!(\mathbf{True}) \longrightarrow \mathbf{Skip} \\ \square \neg (((v_speed >_{\mathcal{A}} v_up_th) \neq 0)) \longrightarrow \\ \quad evaluatecondition.(t_steady_state_upshifting)!(\mathbf{False}) \longrightarrow \mathbf{Skip} \\ \mathbf{fi} \end{array} \right)$$
 $condition_default_steady_state \hat{=}$
 $(evaluatecondition.(t_default_steady_state)!(\mathbf{True}) \longrightarrow \mathbf{Skip})$
 $condition_upshifting_steady_state23 \hat{=}$

$$\left(\begin{array}{l} \mathbf{if}((v_speed <_{\mathcal{A}} v_up_th) \neq 0) \longrightarrow \\ \quad evaluatecondition.(t_upshifting_steady_state23)!(\mathbf{True}) \longrightarrow \mathbf{Skip} \\ \square \neg (((v_speed <_{\mathcal{A}} v_up_th) \neq 0)) \longrightarrow \\ \quad evaluatecondition.(t_upshifting_steady_state23)!(\mathbf{False}) \longrightarrow \mathbf{Skip} \\ \mathbf{fi} \end{array} \right)$$
 $condition_steady_state_downshifting \hat{=}$

$$\left(\begin{array}{l} \mathbf{if}((v_speed <_{\mathcal{A}} v_down_th) \neq 0) \longrightarrow \\ \quad evaluatecondition.(t_steady_state_downshifting)!(\mathbf{True}) \longrightarrow \mathbf{Skip} \\ \square \neg (((v_speed <_{\mathcal{A}} v_down_th) \neq 0)) \longrightarrow \\ \quad evaluatecondition.(t_steady_state_downshifting)!(\mathbf{False}) \longrightarrow \mathbf{Skip} \\ \mathbf{fi} \end{array} \right)$$
 $condition_downshifting_steady_state25 \hat{=}$

$$\left(\begin{array}{l} \mathbf{if}((v_speed \leq_{\mathcal{A}} v_down_th) \neq 0) \longrightarrow \\ \quad evaluatecondition.(t_downshifting_steady_state25)!(\mathbf{True}) \longrightarrow \mathbf{Skip} \\ \square \neg (((v_speed \leq_{\mathcal{A}} v_down_th) \neq 0)) \longrightarrow \\ \quad evaluatecondition.(t_downshifting_steady_state25)!(\mathbf{False}) \longrightarrow \mathbf{Skip} \\ \mathbf{fi} \end{array} \right)$$
 $condition_downshifting_steady_state24 \hat{=}$

$$\left(\begin{array}{l} \mathbf{if}((v_speed >_{\mathcal{A}} v_down_th) \neq 0) \longrightarrow \\ \quad evaluatecondition.(t_downshifting_steady_state24)!(\mathbf{True}) \longrightarrow \mathbf{Skip} \\ \square \neg (((v_speed >_{\mathcal{A}} v_down_th) \neq 0)) \longrightarrow \\ \quad evaluatecondition.(t_downshifting_steady_state24)!(\mathbf{False}) \longrightarrow \mathbf{Skip} \\ \mathbf{fi} \end{array} \right)$$
 $condition_upshifting_steady_state26 \hat{=}$

$$\left(\begin{array}{l} \mathbf{if}((v_speed \geq_{\mathcal{A}} v_up_th) \neq 0) \longrightarrow \\ \quad evaluatecondition.(t_upshifting_steady_state26)!(\mathbf{True}) \longrightarrow \mathbf{Skip} \\ \square \neg (((v_speed \geq_{\mathcal{A}} v_up_th) \neq 0)) \longrightarrow \\ \quad evaluatecondition.(t_upshifting_steady_state26)!(\mathbf{False}) \longrightarrow \mathbf{Skip} \\ \mathbf{fi} \end{array} \right)$$

$$conditions \hat{=} \left(\begin{array}{l} condition_third_fourth \square condition_second_third \square \\ condition_first_second \square condition_fourth_third \square \\ condition_default_first \square condition_second_first \square \\ condition_third_second \square \\ condition_steady_state_upshifting \square \\ condition_default_steady_state \square \\ condition_upshifting_steady_state23 \square \\ condition_steady_state_downshifting \square \\ condition_downshifting_steady_state25 \square \\ condition_downshifting_steady_state24 \square \\ condition_upshifting_steady_state26 \end{array} \right)$$

$$\begin{aligned} trigger_third_fourth &\hat{=} checktrigger.(t_third_fourth)?e \longrightarrow \\ &\left(\begin{array}{l} \mathbf{if} \ e = e_UP \longrightarrow result.(t_third_fourth).(e)!(\mathbf{True}) \longrightarrow \mathbf{Skip} \\ \square \neg (e = e_UP) \longrightarrow result.(t_third_fourth).(e)!(\mathbf{False}) \longrightarrow \mathbf{Skip} \\ \mathbf{fi} \end{array} \right) \\ trigger_second_third &\hat{=} checktrigger.(t_second_third)?e \longrightarrow \\ &\left(\begin{array}{l} \mathbf{if} \ e = e_UP \longrightarrow result.(t_second_third).(e)!(\mathbf{True}) \longrightarrow \mathbf{Skip} \\ \square \neg (e = e_UP) \longrightarrow result.(t_second_third).(e)!(\mathbf{False}) \longrightarrow \mathbf{Skip} \\ \mathbf{fi} \end{array} \right) \\ trigger_first_second &\hat{=} checktrigger.(t_first_second)?e \longrightarrow \\ &\left(\begin{array}{l} \mathbf{if} \ e = e_UP \longrightarrow result.(t_first_second).(e)!(\mathbf{True}) \longrightarrow \mathbf{Skip} \\ \square \neg (e = e_UP) \longrightarrow result.(t_first_second).(e)!(\mathbf{False}) \longrightarrow \mathbf{Skip} \\ \mathbf{fi} \end{array} \right) \\ trigger_fourth_third &\hat{=} checktrigger.(t_fourth_third)?e \longrightarrow \\ &\left(\begin{array}{l} \mathbf{if} \ e = e_DOWN \longrightarrow result.(t_fourth_third).(e)!(\mathbf{True}) \longrightarrow \mathbf{Skip} \\ \square \neg (e = e_DOWN) \longrightarrow result.(t_fourth_third).(e)!(\mathbf{False}) \longrightarrow \mathbf{Skip} \\ \mathbf{fi} \end{array} \right) \end{aligned}$$

$$\begin{aligned}
& \text{trigger_default_first} \hat{=} \text{checktrigger}.(t_default_first)?e \longrightarrow \\
& \quad \text{result}.(t_default_first).(e)!(\mathbf{True}) \longrightarrow \mathbf{Skip} \\
& \text{trigger_second_first} \hat{=} \text{checktrigger}.(t_second_first)?e \longrightarrow \\
& \quad \left(\begin{array}{l} \mathbf{if} \ e = e_DOWN \longrightarrow \text{result}.(t_second_first).(e)!(\mathbf{True}) \longrightarrow \mathbf{Skip} \\ \square \neg (e = e_DOWN) \longrightarrow \text{result}.(t_second_first).(e)!(\mathbf{False}) \longrightarrow \mathbf{Skip} \\ \mathbf{fi} \end{array} \right) \\
& \text{trigger_third_second} \hat{=} \text{checktrigger}.(t_third_second)?e \longrightarrow \\
& \quad \left(\begin{array}{l} \mathbf{if} \ e = e_DOWN \longrightarrow \text{result}.(t_third_second).(e)!(\mathbf{True}) \longrightarrow \mathbf{Skip} \\ \square \neg (e = e_DOWN) \longrightarrow \text{result}.(t_third_second).(e)!(\mathbf{False}) \longrightarrow \mathbf{Skip} \\ \mathbf{fi} \end{array} \right) \\
& \text{trigger_steady_state_upshifting} \hat{=} \text{checktrigger}.(t_steady_state_upshifting)?e \longrightarrow \\
& \quad \text{result}.(t_steady_state_upshifting).(e)!(\mathbf{True}) \longrightarrow \mathbf{Skip} \\
& \text{trigger_default_steady_state} \hat{=} \text{checktrigger}.(t_default_steady_state)?e \longrightarrow \\
& \quad \text{result}.(t_default_steady_state).(e)!(\mathbf{True}) \longrightarrow \mathbf{Skip} \\
& \text{trigger_upshifting_steady_state23} \hat{=} \text{checktrigger}.(t_upshifting_steady_state23)?e \longrightarrow \\
& \quad \text{result}.(t_upshifting_steady_state23).(e)!(\mathbf{True}) \longrightarrow \mathbf{Skip} \\
& \text{trigger_steady_state_downshifting} \hat{=} \text{checktrigger}.(t_steady_state_downshifting)?e \longrightarrow \\
& \quad \text{result}.(t_steady_state_downshifting).(e)!(\mathbf{True}) \longrightarrow \mathbf{Skip} \\
& \text{trigger_downshifting_steady_state25} \hat{=} \text{checktrigger}.(t_downshifting_steady_state25)?e \longrightarrow \\
& \quad \text{result}.(t_downshifting_steady_state25).(e)!(\mathbf{True}) \longrightarrow \mathbf{Skip} \\
& \text{trigger_downshifting_steady_state24} \hat{=} \text{checktrigger}.(t_downshifting_steady_state24)?e \longrightarrow \\
& \quad \text{result}.(t_downshifting_steady_state24).(e)!(\mathbf{True}) \longrightarrow \mathbf{Skip} \\
& \text{trigger_upshifting_steady_state26} \hat{=} \text{checktrigger}.(t_upshifting_steady_state26)?e \longrightarrow \\
& \quad \text{result}.(t_upshifting_steady_state26).(e)!(\mathbf{True}) \longrightarrow \mathbf{Skip} \\
& \text{triggers} \hat{=} \left(\begin{array}{l} \text{trigger_third_fourth} \square \text{trigger_second_third} \square \\ \text{trigger_first_second} \square \text{trigger_fourth_third} \square \\ \text{trigger_default_first} \square \text{trigger_second_first} \square \\ \text{trigger_third_second} \square \text{trigger_steady_state_upshifting} \square \\ \text{trigger_default_steady_state} \square \text{trigger_upshifting_steady_state23} \square \\ \text{trigger_steady_state_downshifting} \square \text{trigger_downshifting_steady_state25} \square \\ \text{trigger_downshifting_steady_state24} \square \text{trigger_upshifting_steady_state26} \square \end{array} \right) \\
& \text{getevents} \hat{=} (\text{events}!(\langle ENULL \rangle) \longrightarrow \mathbf{Skip}) \\
& \text{getstate} \hat{=} (\text{state}?x : (x \in \text{dom}(\text{states}))!(\text{states}(x)) \longrightarrow \mathbf{Skip}) \\
& \text{getjunction} \hat{=} (\text{junction}?x : (x \in \text{dom}(\text{junctions}))!(\text{junctions}(x)) \longrightarrow \mathbf{Skip}) \\
& \text{gettransition} \hat{=} (\text{transition}?x : (x \in \text{dom}(\text{transitions}))!(\text{transitions}(x)) \longrightarrow \mathbf{Skip}) \\
& \text{getchart} \hat{=} (\text{chart}!(\text{states}(\text{identifier})) \longrightarrow \mathbf{Skip})
\end{aligned}$$

$broadcast \hat{=} e : EVENT; dest : SID \bullet local_event!(e, states(dest)) \longrightarrow$
 $\mu X \bullet (AllActions ; X \sqcap end_local_execution \longrightarrow \mathbf{Skip})$
 $check \hat{=} \mathbf{res} \text{ erl} : \mathbb{B} \bullet \mu X \bullet (InterfaceActions ; X \sqcap interrupt?x \longrightarrow \text{erl} := x)$
 $status \hat{=} (status?x : (x \in \text{dom}(state_status))!(state_status(x)) \longrightarrow \mathbf{Skip})$
 $history \hat{=} (history?x : (x \in \text{dom}(state_history))!(state_history(x)) \longrightarrow \mathbf{Skip})$
 $activation \hat{=} (activate?x \longrightarrow (Activate))$
 $deactivation \hat{=} (deactivate?x \longrightarrow (Deactivate))$
 $ChartActions \hat{=} \left(\begin{array}{l} entryactions \sqcap duringactions \sqcap \\ exitactions \sqcap conditionactions \sqcap \\ transitionactions \end{array} \right); end_action \longrightarrow \mathbf{Skip}$
 $InterfaceActions \hat{=} \left(\begin{array}{l} getevents \sqcap getchart \sqcap getstate \sqcap getjunction \sqcap gettransition \sqcap \\ status \sqcap history \sqcap activation \sqcap deactivation \end{array} \right)$
 $Inputs \hat{=} read_inputs \longrightarrow \left(\begin{array}{l} i_speed?x \longrightarrow v_speed := x \\ \llbracket \{v_speed\} \mid \{v_throttle\} \rrbracket \\ i_throttle?x \longrightarrow v_throttle := x \end{array} \right)$
 $Outputs \hat{=} write_outputs \longrightarrow o_gear!(v_gear) \longrightarrow \mathbf{Skip}$
 $AllActions \hat{=} \left(\begin{array}{l} conditionactions \sqcap triggers \sqcap Inputs \sqcap Outputs \sqcap \\ ChartActions \sqcap InterfaceActions \end{array} \right)$
 $\bullet (InitState); \left(\mu X \bullet \left(\mu Y \bullet \left(\begin{array}{l} AllActions ; Y \\ \square \\ end_cycle \longrightarrow \mathbf{Skip} \end{array} \right) \right); X \right)$
end

process $shift_logic \hat{=} (P_shift_logic \llbracket interface \cup \{end_cycle\} \rrbracket Simulator) \setminus interface$