NestedMissionSequencer(nestedSequencer2)

Tight Rope v0.65

5th February 2016

1 ID Files

1.1 MissionIds

section $MissionIds$ **parents** $scj\_prelude, MissionId$

$$
\begin{array}{|l}
TopMission1ID : MissionID \\
MyMission1ID : MissionID \\
MyMission2ID : MissionID \\
MyMission3ID : MissionID \\
\hline
distinct \langle nullMissionId, TopMission1ID, MyMission1ID, \\
MyMission2ID, MyMission3ID \rangle
\end{array}
$$

1.2 SchedulablesIds

section *SchedulableIds* **parents** *scj_prelude*, *SchedulableId*

$$
\begin{array}{|l}
MySequencerID : SchedulableID \\
FirstMissionSequencerID : SchedulableID \\
SecondMissionSequencerID : SchedulableID \\
ThirdMissionSequencerID : SchedulableID \\
MyPEH1ID : SchedulableID \\
MyPEH2ID : SchedulableID \\
MyPEH3ID : SchedulableID \\
\hline
distinct \langle nullSequencerId, nullSchedulableId, MySequencerIDID, \\
FirstMissionSequencerID, SecondMissionSequencerID, \\
ThirdMissionSequencerID, MyPEH1ID, \\
MyPEH2ID, MyPEH3ID \rangle
\end{array}
$$

1.3 ThreadIds

section *ThreadIds* **parents** *scj_prelude*, *GlobalTypes*

$$
\begin{array}{l}
ThirdMissionSequencerThreadID : ThreadID \\
MyPEH2ThreadID : ThreadID \\
MyPEH1ThreadID : ThreadID \\
MyPEH3ThreadID : ThreadID \\
FirstMissionSequencerThreadID : ThreadID \\
SecondMissionSequencerThreadID : ThreadID \\
\hline
distinct\langle SafeletThreadId, nullThreadId, \\
ThirdMissionSequencerThreadID, MyPEH2ThreadID, \\
MyPEH1ThreadID, MyPEH3ThreadID, \\
FirstMissionSequencerThreadID, SecondMissionSequencerThreadID\rangle
\end{array}
$$

1.4 ObjectIds

section *ObjectIds* **parents** *scj_prelude*, *GlobalTypes*

$$
\begin{array}{|l}
MyAppObjectID : ObjectID \\
TopMission1ObjectID : ObjectID \\
FirstMissionSequencerObjectID : ObjectID \\
SecondMissionSequencerObjectID : ObjectID \\
ThirdMissionSequencerObjectID : ObjectID \\
MyMission1ObjectID : ObjectID \\
MyPEH1ObjectID : ObjectID \\
MyMission2ObjectID : ObjectID \\
MyPEH2ObjectID : ObjectID \\
MyMission3ObjectID : ObjectID \\
MyPEH3ObjectID : ObjectID \\
\hline
distinct\langle MyAppObjectID, TopMission1ObjectID, \\
FirstMissionSequencerObjectID, SecondMissionSequencerObjectID, \\
ThirdMissionSequencerObjectID, MyMission1ObjectID, \\
MyPEH1ObjectID, MyMission2ObjectID, \\
MyPEH2ObjectID, MyMission3ObjectID, \\
MyPEH3ObjectID\rangle
\end{array}
$$

2 Network

2.1 Network Channel Sets

section *NetworkChannels* **parents** *scj_prelude*, *MissionId*, *MissionIds*,
SchedulableId, *SchedulableIds*, *MissionChan*, *SchedulableChan*, *TopLevelMissionSequencerFWChan*,
FrameworkChan, *SafeletChan*

channelset *TerminateSync* ==
$\{\!|$ *schedulables_terminated*, *schedulables_stopped*, *get_activeSchedulables* $|\!\}$

channelset *ControlTierSync* ==
$\{\!|$ *start_toplevel_sequencer*, *done_toplevel_sequencer*, *done_safeletFW* $|\!\}$

channelset *TierSync* ==
$\{\!|$ *start_mission* . *TopMission*1, *done_mission* . *TopMission*1,
done_safeletFW, *done_toplevel_sequencer* $|\!\}$

channelset *MissionSync* ==
$\{\!|$ *done_safeletFW*, *done_toplevel_sequencer*, *register*,
signalTerminationCall, *signalTerminationRet*, *activate_schedulables*, *done_schedulable*,
cleanupSchedulableCall, *cleanupSchedulableRet* $|\!\}$

channelset *SchedulablesSync* ==
$\{\!|$ *activate_schedulables*, *done_safeletFW*, *done_toplevel_sequencer* $|\!\}$

channelset *ClusterSync* ==
$\{\!|$ *done_toplevel_sequencer*, *done_safeletFW* $|\!\}$

channelset *AppSync* ==
$\bigcup\{$*SafeltAppSync*, *MissionSequencerAppSync*, *MissionAppSync*,
MTAppSync, *OSEHSync*, *APEHSync*,
$\{\!|$ *getSequencer*, *end_mission_app*, *end_managedThread_app*,
setCeilingPriority, *requestTerminationCall*, *requestTerminationRet*, *terminationPendingCall*,
terminationPendingRet, *handleAsyncEventCall*, *handleAsyncEventRet* $|\!\}\}$

channelset *ThreadSync* ==
$\{\!|$ *raise_thread_priority*, *lower_thread_priority*, *isInterruptedCall*, *isInterruptedRet*, *get_priorityLevel* $|\!\}$

channelset *LockingSync* ==
$\{\!|$ *lockAcquired*, *startSyncMeth*, *endSyncMeth*, *waitCall*, *waitRet*, *notify*, *isInterruptedCall*, *isInterruptedRet*,
interruptedCall, *interruptedRet*, *done_toplevel_sequencer*, *get_priorityLevel* $|\!\}$

channelset *Tier*0*Sync* ==
$\{\!|$ *done_toplevel_sequencer*, *done_safeletFW*,
start_mission . *MyMission*1, *done_mission* . *MyMission*1,
initializeRet . *MyMission*1, *requestTermination* . *MyMission*1 . *MySequencer* $|\!\}$

channelset *Tier*1*Sync* ==
$\{\!|$ *done_toplevel_sequencer*, *done_safeletFW*,
start_mission . *MyMission*2, *done_mission* . *MyMission*2,
initializeRet . *MyMission*2, *requestTermination* . *MyMission*2 . $|\!\}$

channelset *Tier*2*Sync* ==
$\{\!|$ *done_toplevel_sequencer*, *done_safeletFW*,
start_mission . *MyMission*3, *done_mission* . *MyMission*3,
initializeRet . *MyMission*3, *requestTermination* . *MyMission*3 . $|\!\}$

2.2 MethodCallBinder

$\mathbf{channelset}\ \mathit{MethodCallBinderSync} == \lbrace\!| \mathit{done\_toplevel\_sequencer}, |\!\rbrace$

$\mathbf{process}\ \mathit{MethodCallBinder} \mathrel{\widehat=} \mathbf{begin}$

$\mathit{BinderActions} \mathrel{\widehat=}$
$)($

$\bullet\ \mathit{BinderActions} \mathbin{\triangle} (\mathit{done\_toplevel\_sequencer} \longrightarrow \mathbf{Skip})$

$\mathbf{end}$

$\mathbf{process}\ \mathit{ApplicationB} \mathrel{\widehat=} \mathit{Application} \llbracket \mathit{MethodCallBinderSync} \rrbracket \mathit{MethodCallBinder}$

2.3 Locking

$$\textbf{process}\ \mathit{Threads} \mathrel{\widehat=} \left(\begin{array}{l} \mathit{ThreadFW}(\mathit{ThirdMissionSequencerThreadID}, 10) \\ \interleave \\ \mathit{ThreadFW}(\mathit{MyPEH2ThreadID}, 20) \\ \interleave \\ \mathit{ThreadFW}(\mathit{MyPEH1ThreadID}, 5) \\ \interleave \\ \mathit{ThreadFW}(\mathit{MyPEH3ThreadID}, 10) \\ \interleave \\ \mathit{ThreadFW}(\mathit{FirstMissionSequencerThreadID}, 5) \\ \interleave \\ \mathit{ThreadFW}(\mathit{SecondMissionSequencerThreadID}, 15) \end{array}\right)$$

$$\textbf{process}\ \mathit{Objects} \mathrel{\widehat=} \left(\begin{array}{l} \mathit{ObjectFW}(\mathit{MyAppObjectID}) \\ \interleave \\ \mathit{ObjectFW}(\mathit{TopMission1ObjectID}) \\ \interleave \\ \mathit{ObjectFW}(\mathit{FirstMissionSequencerObjectID}) \\ \interleave \\ \mathit{ObjectFW}(\mathit{SecondMissionSequencerObjectID}) \\ \interleave \\ \mathit{ObjectFW}(\mathit{ThirdMissionSequencerObjectID}) \\ \interleave \\ \mathit{ObjectFW}(\mathit{MyMission1ObjectID}) \\ \interleave \\ \mathit{ObjectFW}(\mathit{MyPEH1ObjectID}) \\ \interleave \\ \mathit{ObjectFW}(\mathit{MyMission2ObjectID}) \\ \interleave \\ \mathit{ObjectFW}(\mathit{MyPEH2ObjectID}) \\ \interleave \\ \mathit{ObjectFW}(\mathit{MyMission3ObjectID}) \\ \interleave \\ \mathit{ObjectFW}(\mathit{MyPEH3ObjectID}) \end{array}\right)$$

$$\textbf{process}\ \mathit{Locking} \mathrel{\widehat=} \mathit{Threads} \llbracket \mathit{ThreadSync} \rrbracket \mathit{Objects}$$

2.4 Program

$\mathbf{section}\ Program\ \mathbf{parents}\ scj\_prelude, MissionId, MissionIds,$
$\quad SchedulableId, SchedulableIds, MissionChan, SchedulableMethChan, MissionFW,$
$\quad SafeletFW, TopLevelMissionSequencerFW, NetworkChannels, ManagedThreadFW,$
$\quad SchedulableMissionSequencerFW, PeriodicEventHandlerFW, OneShotEventHandlerFW,$
$\quad AperiodicEventHandlerFW, ObjectFW, ThreadFW,$
$\quad MyAppApp, MySequencerApp, TopMission1App, FirstMissionSequencerApp, SecondMissionSequencerApp,$
$\quad ThirdMissionSequencerApp, MyMission1App, MyPEH1App, MyMission2App, MyPEH2App, MyMission3App, MyPE$

$$\mathbf{process}\ ControlTier \mathrel{\widehat=} \left(\begin{array}{l} SafeletFW \\ \quad [\![ControlTierSync]\!] \\ TopLevelMissionSequencerFW(MySequencer) \end{array}\right)$$

$$\mathbf{process}\ Tier0 \mathrel{\widehat=} \left(\begin{array}{l} MissionFW(TopMission1ID) \\ \quad [\![MissionSync]\!] \\ \left(\left(\begin{array}{l} SchedulableMissionSequencerFW(FirstMissionSequencerID) \\ \quad [\![SchedulablesSync]\!] \\ SchedulableMissionSequencerFW(SecondMissionSequencerID) \\ \quad [\![SchedulablesSync]\!] \\ SchedulableMissionSequencerFW(ThirdMissionSequencerID) \end{array}\right) \\ \quad [\![SchedulablesSync]\!] \end{array}\right)\right)$$

$$\mathbf{process}\ Tier1 \mathrel{\widehat=} \left(\begin{array}{l} MissionFW(MyMission1ID) \\ \quad [\![MissionSync]\!] \\ \left(PeriodicEventHandlerFW(MyPEH1ID, (NULL, time(1000, 0), NULL, nullSchedulableId))\right) \end{array}\right)$$

$$\mathbf{process}\ Tier2 \mathrel{\widehat=} \left(\begin{array}{l} MissionFW(MyMission2ID) \\ \quad [\![MissionSync]\!] \\ \left(PeriodicEventHandlerFW(MyPEH2ID, (NULL, time(1000, 0), NULL, nullSchedulableId))\right) \end{array}\right)$$

$$\mathbf{process}\ Tier3 \mathrel{\widehat=} \left(\begin{array}{l} MissionFW(MyMission3ID) \\ \quad [\![MissionSync]\!] \\ \left(PeriodicEventHandlerFW(MyPEH3ID, (NULL, time(1000, 0), NULL, nullSchedulableId))\right) \end{array}\right)$$

$$\mathbf{process}\ Framework \mathrel{\widehat=} \left(\begin{array}{l} ControlTier \\ \quad [\![TierSync]\!] \\ \left(\begin{array}{l} Tier0 \\ \quad [\![Tier0Sync]\!] \\ Tier1 \\ \quad [\![Tier1Sync]\!] \\ Tier2 \\ \quad [\![Tier2Sync]\!] \\ Tier3 \end{array}\right) \end{array}\right)$$

process *Application* $\mathrel{\widehat=}$

$$\left(\begin{array}{l} \mathit{MyAppApp} \\ ||| \\ \mathit{MySequencerApp} \\ ||| \\ \mathit{TopMission1App} \\ ||| \\ \mathit{FirstMissionSequencerApp} \\ ||| \\ \mathit{SecondMissionSequencerApp} \\ ||| \\ \mathit{ThirdMissionSequencerApp} \\ ||| \\ \mathit{MyMission1App} \\ ||| \\ \mathit{MyPEH1App}(\mathit{MyMission1ID}) \\ ||| \\ \mathit{MyMission2App} \\ ||| \\ \mathit{MyPEH2App}(\mathit{MyMission2ID}) \\ ||| \\ \mathit{MyMission3App} \\ ||| \\ \mathit{MyPEH3App}(\mathit{MyMission3ID}) \end{array}\right)$$

process *Program* $\mathrel{\widehat=} \big(\mathit{Framework} \llbracket \mathit{AppSync} \rrbracket \mathit{ApplicationB}\big) \llbracket \mathit{LockingSync} \rrbracket \mathit{Locking}$

3 Safelet

section $\mathit{MyAppApp}$ **parents** $\mathit{scj\_prelude}, \mathit{SchedulableId}, \mathit{SchedulableIds}, \mathit{SafeletChan}$

process $\mathit{MyAppApp} \mathrel{\widehat=}$ **begin**

$$\mathit{InitializeApplication} \mathrel{\widehat=} \left(\begin{array}{l} \mathit{initializeApplicationCall} \longrightarrow \\ \mathit{initializeApplicationRet} \longrightarrow \\ \mathbf{Skip} \end{array}\right)$$

$$\mathit{GetSequencer} \mathrel{\widehat=} \left(\begin{array}{l} \mathit{getSequencerCall} \longrightarrow \\ \mathit{getSequencerRet}\,!\,\mathit{MySequencerID} \longrightarrow \\ \mathbf{Skip} \end{array}\right)$$

$$\mathit{immortalMemorySizeMeth} \mathrel{\widehat=} \mathbf{var}\ \mathit{ret} : \mathbb{Z} \bullet \left(\begin{array}{l} \mathit{immortalMemorySizeCall}\,.\,\mathit{MyApp} \longrightarrow \\ \left(\mathit{ret} := 10000\right); \\ \mathit{immortalMemorySizeRet}\,.\,\mathit{MyApp}\,!\,\mathit{ret} \longrightarrow \\ \mathbf{Skip} \end{array}\right)$$

$$\mathit{Methods} \mathrel{\widehat=} \left(\begin{array}{l} \mathit{GetSequencer} \\ \Box \\ \mathit{InitializeApplication} \\ \Box \\ \mathit{immortalMemorySizeMeth} \end{array}\right) ;\ \mathit{Methods}$$

$\bullet\ (\mathit{Methods}) \mathbin{\triangle} (\mathit{end\_safelet\_app} \longrightarrow \mathbf{Skip})$

end

4 Top Level Mission Sequencer

section *MySequencerApp* **parents** *TopLevelMissionSequencerChan*, *MissionId*, *MissionIds*, *SchedulableId*, *MySequencerClass*

process *MySequencerApp* $\widehat{=}$ **begin**

State

$this : \mathbf{ref}\ MySequencerClass$

state *State*

Init

$State\,'$

$this' = \mathbf{new}\ MySequencerClass()$

$$GetNextMission \mathrel{\widehat{=}} \mathbf{var}\ ret : MissionID \bullet \left(\begin{array}{l} getNextMissionCall\,.\,MySequencer \longrightarrow \\ ret := this\,.\,getNextMission(); \\ getNextMissionRet\,.\,MySequencer\,!\,ret \longrightarrow \\ \mathbf{Skip} \end{array}\right)$$

$$Methods \mathrel{\widehat{=}} \big(GetNextMission\big)\ ;\ Methods$$

$$\bullet\ (Init\ ;\ Methods) \bigtriangleup (end\_sequencer\_app\,.\,MySequencer \longrightarrow \mathbf{Skip})$$

end

class *MySequencerClass* $\widehat{=}$ **begin**

state *State*

$myMission : TopMission1$
$done : \mathbb{B}$

state *State*

initial *Init*

$State'$

$myMission' = TopMission1$
$done' = false$

protected *getNextMission* $\widehat{=}$ **var** $ret : MissionID$ $\bullet$

$$\left(\begin{array}{l}\mathbf{if}\ (done = \mathbf{True} = \mathbf{False}) \longrightarrow \\ \qquad \left(\begin{array}{l} this\,.\,done := true; \\ ret := myMission \end{array}\right) \\ [\!]\ (done = \mathbf{True} = \mathbf{False}) \longrightarrow \\ \qquad \left(ret := nullMissionId\right) \\ \mathbf{fi} \end{array}\right)$$

$\bullet$ **Skip**

end

5 Missions

5.1 TopMission1

section $\mathit{TopMission1App}$ **parents** $\mathit{scj\_prelude}, \mathit{MissionId}, \mathit{MissionIds},$
$\mathit{SchedulableId}, \mathit{SchedulableIds}, \mathit{MissionChan}, \mathit{SchedulableMethChan}$
$, \mathit{TopMission1MethChan}$

process $\mathit{TopMission1App} \mathrel{\widehat=}$ **begin**

$$\begin{array}{|l}\hline \textit{State} \\ \mathit{this} : \mathbf{ref}\ \mathit{TopMission1Class} \\ \hline \end{array}$$

state State

$$\begin{array}{|l}\hline \textit{Init} \\ \mathit{State}' \\ \hline \mathit{this}' = \mathbf{new}\ \mathit{TopMission1Class}() \\ \hline \end{array}$$

$$\mathit{InitializePhase} \mathrel{\widehat=}$$
$$\left(\begin{array}{l}\mathit{initializeCall} \,.\, \mathit{TopMission1} \longrightarrow \\ \mathit{register}\,!\,\mathit{FirstMissionSequencer}\,!\,\mathit{TopMission1} \longrightarrow \\ \mathit{register}\,!\,\mathit{SecondMissionSequencer}\,!\,\mathit{TopMission1} \longrightarrow \\ \mathit{register}\,!\,\mathit{ThirdMissionSequencer}\,!\,\mathit{TopMission1} \longrightarrow \\ \mathit{initializeRet} \,.\, \mathit{TopMission1} \longrightarrow \\ \mathbf{Skip}\end{array}\right)$$

$$\mathit{CleanupPhase} \mathrel{\widehat=}$$
$$\left(\begin{array}{l}\mathit{cleanupMissionCall} \,.\, \mathit{TopMission1} \longrightarrow \\ \mathit{cleanupMissionRet} \,.\, \mathit{TopMission1}\,!\,\mathbf{True} \longrightarrow \\ \mathbf{Skip}\end{array}\right)$$

$$\mathit{Methods} \mathrel{\widehat=} \left(\begin{array}{l}\mathit{InitializePhase} \\ \Box \\ \mathit{CleanupPhase}\end{array}\right) ;\ \mathit{Methods}$$

$$\bullet\ (\mathit{Init}\ ;\ \mathit{Methods}) \mathbin{\triangle} (\mathit{end\_mission\_app} \,.\, \mathit{TopMission1} \longrightarrow \mathbf{Skip})$$

end

5.2 Schedulables of TopMission1

section *FirstMissionSequencerApp* **parents** *TopLevelMissionSequencerChan*, *MissionId*, *MissionIds*, *SchedulableId*, *FirstMissionSequencerClass*

process *FirstMissionSequencerApp* $\widehat{=}$ **begin**

$$
\begin{array}{l}
GetNextMission \mathrel{\widehat{=}} \mathbf{var}\ ret : MissionID \bullet \\
\left(\begin{array}{l}
getNextMissionCall . FirstMissionSequencer \longrightarrow \\
ret := this . getNextMission(); \\
getNextMissionRet . FirstMissionSequencer ! ret \longrightarrow \\
\mathbf{Skip}
\end{array}\right)
\end{array}
$$

$$
\begin{array}{l}
Methods \mathrel{\widehat{=}} \\
\left(GetNextMission \right) ;\ Methods
\end{array}
$$

$$\bullet\ (Methods) \triangle (end\_sequencer\_app . FirstMissionSequencer \longrightarrow \mathbf{Skip})$$

end

class *FirstMissionSequencerClass* $\widehat{=}$ **begin**

state *State*

$$
\begin{array}{l}
\textbf{ref}\ \mathit{myMissionClass} : \mathit{MissionClass} \\
\mathit{done} : \mathbb{B}
\end{array}
$$

state *State*

initial *Init*

$$
\begin{array}{l}
\mathit{State}\,' \\
\hline
\textbf{ref}\ \mathit{myMissionClass}' = \textbf{new}\ \mathit{MissionClass}() \\
\mathit{done}' = \mathit{false}
\end{array}
$$

$$
\textbf{protected}\ \mathit{getNextMission}\ \widehat{=}\ \textbf{var}\ \mathit{ret} : \mathit{MissionID} \bullet
\left(\begin{array}{l}
\textbf{if}\ (\mathit{done} = \textbf{True} = \textbf{False}) \longrightarrow \\
\qquad \left(\begin{array}{l} \mathit{this} \,.\, \mathit{done} := \mathit{true}; \\ \mathit{ret} := \mathit{myMission} \end{array}\right) \\
[\!]\ (\mathit{done} = \textbf{True} = \textbf{False}) \longrightarrow \\
\qquad \left(\mathit{ret} := \mathit{nullMissionId}\right) \\
\textbf{fi}
\end{array}\right)
$$

$\bullet$ **Skip**

end

section *SecondMissionSequencerApp* **parents** *TopLevelMissionSequencerChan*, *MissionId*, *MissionIds*, *SchedulableId*, *SecondMissionSequencerClass*

process *SecondMissionSequencerApp* $\widehat{=}$ **begin**

$$
\begin{array}{l}
GetNextMission \mathrel{\widehat{=}} \mathbf{var}\ ret : MissionID \bullet \\
\left(\begin{array}{l}
getNextMissionCall\,.\,SecondMissionSequencer \longrightarrow \\
ret := this\,.\,getNextMission(); \\
getNextMissionRet\,.\,SecondMissionSequencer\,!\,ret \longrightarrow \\
\mathbf{Skip}
\end{array}\right)
\end{array}
$$

$$
\begin{array}{l}
Methods \mathrel{\widehat{=}} \\
\big(GetNextMission\big)\ ;\ \ Methods
\end{array}
$$

$$\bullet\ (Methods) \triangle (end\_sequencer\_app\,.\,SecondMissionSequencer \longrightarrow \mathbf{Skip})$$

end

class *SecondMissionSequencerClass* $\widehat{=}$ **begin**

$$\begin{array}{|l}
\multicolumn{1}{l}{\textbf{state}\ \mathit{State}} \\ \hline
\textbf{ref}\ \mathit{myMissionClass} : \mathit{MissionClass} \\
\mathit{done} : \mathbb{B} \\ \hline
\end{array}$$

state *State*

$$\begin{array}{|l}
\multicolumn{1}{l}{\textbf{initial}\ \mathit{Init}} \\ \hline
\mathit{State}\,' \\ \hline
\textbf{ref}\ \mathit{myMissionClass}' = \textbf{new}\ \mathit{MissionClass}() \\
\mathit{done}' = \mathit{false} \\ \hline
\end{array}$$

$$\textbf{protected}\ \mathit{getNextMission}\ \widehat{=}\ \textbf{var}\ \mathit{ret} : \mathit{MissionID} \bullet \\
\left(\begin{array}{l}
\textbf{if}\ (\mathit{done} = \textbf{True} = \textbf{False}) \longrightarrow \\
\qquad \left(\begin{array}{l} \mathit{this}\,.\,\mathit{done} := \mathit{true}; \\ \mathit{ret} := \mathit{myMission} \end{array}\right) \\
[\!]\ (\mathit{done} = \textbf{True} = \textbf{False}) \longrightarrow \\
\qquad \left(\mathit{ret} := \mathit{nullMissionId}\right) \\
\textbf{fi}
\end{array}\right)$$

$\bullet$ **Skip**

end

section *ThirdMissionSequencerApp* **parents** *TopLevelMissionSequencerChan*, *MissionId*, *MissionIds*, *SchedulableId*, *ThirdMissionSequencerClass*

process *ThirdMissionSequencerApp* $\widehat{=}$ **begin**

$$
\begin{array}{l}
\mathit{GetNextMission} \mathrel{\widehat=} \mathbf{var}\ \mathit{ret} : \mathit{MissionID} \bullet \\
\left(\begin{array}{l}
\mathit{getNextMissionCall} \,.\, \mathit{ThirdMissionSequencer} \longrightarrow \\
\mathit{ret} := \mathit{this} \,.\, \mathit{getNextMission}(); \\
\mathit{getNextMissionRet} \,.\, \mathit{ThirdMissionSequencer}\ !\ \mathit{ret} \longrightarrow \\
\mathbf{Skip}
\end{array}\right)
\end{array}
$$

$$
\begin{array}{l}
\mathit{Methods} \mathrel{\widehat=} \\
\big(\mathit{GetNextMission}\big)\ ;\ \ \mathit{Methods}
\end{array}
$$

$$\bullet\ (\mathit{Methods}) \mathbin{\triangle} (\mathit{end\_sequencer\_app} \,.\, \mathit{ThirdMissionSequencer} \longrightarrow \mathbf{Skip})$$

end

class *ThirdMissionSequencerClass* $\widehat{=}$ **begin**

state *State*

$myMission : MyMission3$
$done : \mathbb{B}$

state *State*

initial *Init*

$State\,'$

$myMission' = MyMission3$
$done' = false$

protected *getNextMission* $\widehat{=}$ **var** $ret : MissionID \bullet$

$$\left(\begin{array}{l}
\textbf{if}\ (done = \textbf{True} = \textbf{False}) \longrightarrow \\
\qquad \left(\begin{array}{l} this\,.\,done := true; \\ ret := myMission \end{array}\right) \\
[\!]\ (done = \textbf{True} = \textbf{False}) \longrightarrow \\
\qquad \left(ret := nullMissionId\right) \\
\textbf{fi}
\end{array}\right)$$

$\bullet$ **Skip**

end

5.3 MyMission1

section $\mathit{MyMission1App}$ **parents** $\mathit{scj\_prelude}, \mathit{MissionId}, \mathit{MissionIds},$
$\mathit{SchedulableId}, \mathit{SchedulableIds}, \mathit{MissionChan}, \mathit{SchedulableMethChan}$
$, \mathit{MyMission1MethChan}$

process $\mathit{MyMission1App} \widehat{=}$ **begin**

$$\begin{array}{|l}\hline \textit{State} \\ \mathit{this} : \mathbf{ref}\ \mathit{MyMission1Class} \\ \hline\end{array}$$

state State

$$\begin{array}{|l}\hline \textit{Init} \\ \mathit{State}' \\ \hline \mathit{this}' = \mathbf{new}\ \mathit{MyMission1Class}() \\ \hline\end{array}$$

$$\mathit{InitializePhase} \widehat{=}$$
$$\left(\begin{array}{l}\mathit{initializeCall}\,.\,\mathit{MyMission1} \longrightarrow \\ \mathit{register}\,!\,\mathit{MyPEH1}\,!\,\mathit{MyMission1} \longrightarrow \\ \mathit{initializeRet}\,.\,\mathit{MyMission1} \longrightarrow \\ \mathbf{Skip}\end{array}\right)$$

$$\mathit{CleanupPhase} \widehat{=}$$
$$\left(\begin{array}{l}\mathit{cleanupMissionCall}\,.\,\mathit{MyMission1} \longrightarrow \\ \mathit{cleanupMissionRet}\,.\,\mathit{MyMission1}\,!\,\mathbf{True} \longrightarrow \\ \mathbf{Skip}\end{array}\right)$$

$$\mathit{Methods} \widehat{=} \left(\begin{array}{l}\mathit{InitializePhase} \\ \Box \\ \mathit{CleanupPhase}\end{array}\right) ;\ \mathit{Methods}$$

$$\bullet\ (\mathit{Init}\ ;\ \mathit{Methods}) \bigtriangleup (\mathit{end\_mission\_app}\,.\,\mathit{MyMission1} \longrightarrow \mathbf{Skip})$$

end

5.4 Schedulables of MyMission1

section $MyPEH1App$ **parents** $PeriodicEventHandlerChan, SchedulableId, SchedulableIds$

process $MyPEH1App \mathrel{\widehat=}$
$\quad m : MissionID \bullet$ **begin**

$$
\begin{array}{l}
handleAsyncEvent \mathrel{\widehat=} \\
\left(\begin{array}{l}
handleAsyncEventCall \,.\, MyPEH1 \longrightarrow \\
\left(\begin{array}{l}
count := count + 1; \\
\mathbf{if}\ (count = 10) \longrightarrow \\
\qquad \big(requestTerminationCall \,.\, m \,.\, MyPEH1 \longrightarrow \quad requestTerminationRet \,.\, m \,.\, MyPEH1 \,?\, requestTermination \longrightarrow \quad \mathbf{Sk} \\
\ [\!]\ (count = 10) \longrightarrow \mathbf{Skip} \\
\mathbf{fi}\,; \\
\mathbf{Skip}
\end{array}\right) \\
handleAsyncEventRet \,.\, MyPEH1 \longrightarrow \\
\mathbf{Skip}
\end{array}\right)
\end{array}
$$

$$
\begin{array}{l}
Methods \mathrel{\widehat=} \\
\big(handleAsyncEvent\big)\ ;\ \ Methods
\end{array}
$$

$\bullet\ (Methods) \mathbin{\triangle} (end\_periodic\_app \,.\, MyPEH1 \longrightarrow \mathbf{Skip})$

end

class $\mathit{MyPEH1Class} \mathrel{\widehat=}$ **begin**

```
── state State ──────────────
│ count : ℤ
│ m : Mission
└─────────────────────────────
```

state *State*

```
── initial Init ─────────────
│ State′
├───────────
│ count′ = 0
└─────────────────────────────
```

• **Skip**

end

5.5 MyMission2

section *MyMission2App* **parents** *scj_prelude*, *MissionId*, *MissionIds*, *SchedulableId*, *SchedulableIds*, *MissionChan*, *SchedulableMethChan* , *MyMission2MethChan*

process $MyMission2App \mathrel{\widehat=}$ **begin**

$$\begin{array}{|l}\hline \textit{State} \\ \quad this : \mathbf{ref}\ MyMission2Class \\ \hline\end{array}$$

state *State*

$$\begin{array}{|l}\hline \textit{Init} \\ \quad State' \\ \hline \quad this' = \mathbf{new}\ MyMission2Class() \\ \hline\end{array}$$

$$InitializePhase \mathrel{\widehat=} \left(\begin{array}{l} initializeCall\,.\,MyMission2 \longrightarrow \\ register\,!\,MyPEH2\,!\,MyMission2 \longrightarrow \\ initializeRet\,.\,MyMission2 \longrightarrow \\ \mathbf{Skip} \end{array}\right)$$

$$CleanupPhase \mathrel{\widehat=} \left(\begin{array}{l} cleanupMissionCall\,.\,MyMission2 \longrightarrow \\ cleanupMissionRet\,.\,MyMission2\,!\,\mathbf{True} \longrightarrow \\ \mathbf{Skip} \end{array}\right)$$

$$Methods \mathrel{\widehat=} \left(\begin{array}{l} InitializePhase \\ \Box \\ CleanupPhase \end{array}\right) \mathbin{;}\ Methods$$

$$\bullet\ (Init \mathbin{;}\ Methods) \mathbin{\triangle} (end\_mission\_app\,.\,MyMission2 \longrightarrow \mathbf{Skip})$$

end

5.6 Schedulables of MyMission2

section $MyPEH2App$ **parents** $PeriodicEventHandlerChan, SchedulableId, SchedulableIds$

process $MyPEH2App \mathrel{\widehat=}$
    $m : MissionID \bullet$ **begin**

```
$handleAsyncEvent \mathrel{\widehat=}$
(  $handleAsyncEventCall\,.\,MyPEH2 \longrightarrow$
   (  $count := count + 1;$
      **if** $(count = 10) \longrightarrow$
            $\big(requestTerminationCall\,.\,m\,.\,MyPEH2 \longrightarrow \quad requestTerminationRet\,.\,m\,.\,MyPEH2\,?\,requestTermination \longrightarrow$ **Sk**
      $[\!]\ (count = 10) \longrightarrow$ **Skip**
      **fi** ;
      **Skip**
   )
   $handleAsyncEventRet\,.\,MyPEH2 \longrightarrow$
   **Skip**
)
```

$Methods \mathrel{\widehat=}$
$\big(handleAsyncEvent\big)\ ;\ \ Methods$

$\bullet\ (Methods) \mathbin{\triangle} (end\_periodic\_app\,.\,MyPEH2 \longrightarrow \mathbf{Skip})$

end

class $\mathit{MyPEH2Class} \mathrel{\widehat=}$ **begin**

state *State*

$$\begin{array}{|l}
\mathit{count} : \mathbb{Z} \\
m : \mathit{Mission} \\
\hline
\end{array}$$

state *State*

initial *Init*

$$\begin{array}{|l}
\mathit{State}' \\
\hline
\mathit{count}' = 0 \\
\hline
\end{array}$$

$\bullet$ **Skip**

end

5.7 MyMission3

section *MyMission3App* **parents** *scj_prelude*, *MissionId*, *MissionIds*,
SchedulableId, *SchedulableIds*, *MissionChan*, *SchedulableMethChan*
, *MyMission3MethChan*

process *MyMission3App* $\widehat{=}$ **begin**

$$\begin{array}{|l}
\hline \textit{State} \\
\textit{this} : \mathbf{ref}\ \textit{MyMission3Class} \\
\hline
\end{array}$$

state *State*

$$\begin{array}{|l}
\hline \textit{Init} \\
\textit{State}\,' \\
\hline
\textit{this}' = \mathbf{new}\ \textit{MyMission3Class}() \\
\hline
\end{array}$$

$$\textit{InitializePhase} \mathrel{\widehat{=}} \\ \left(\begin{array}{l} \textit{initializeCall} \,.\, \textit{MyMission3} \longrightarrow \\ \textit{register}\,!\,\textit{MyPEH3}\,!\,\textit{MyMission3} \longrightarrow \\ \textit{initializeRet} \,.\, \textit{MyMission3} \longrightarrow \\ \mathbf{Skip} \end{array}\right)$$

$$\textit{CleanupPhase} \mathrel{\widehat{=}} \\ \left(\begin{array}{l} \textit{cleanupMissionCall} \,.\, \textit{MyMission3} \longrightarrow \\ \textit{cleanupMissionRet} \,.\, \textit{MyMission3}\,!\,\mathbf{True} \longrightarrow \\ \mathbf{Skip} \end{array}\right)$$

$$\textit{Methods} \mathrel{\widehat{=}} \left(\begin{array}{l} \textit{InitializePhase} \\ \Box \\ \textit{CleanupPhase} \end{array}\right) \ ; \ \textit{Methods}$$

$$\bullet\ (\textit{Init}\ ;\ \ \textit{Methods}) \bigtriangleup (\textit{end\_mission\_app} \,.\, \textit{MyMission3} \longrightarrow \mathbf{Skip})$$

end

5.8 Schedulables of MyMission3

section *MyPEH3App* **parents** *PeriodicEventHandlerChan*, *SchedulableId*, *SchedulableIds*

$$\mathbf{process}\ MyPEH3App \mathrel{\widehat=}$$
$$\quad m : MissionID \bullet \mathbf{begin}$$

$$handleAsyncEvent \mathrel{\widehat=}$$
$$\left(\begin{array}{l} handleAsyncEventCall\,.\,MyPEH3 \longrightarrow \\ \left(\begin{array}{l} count := count + 1; \\ \mathbf{if}\ (count = 10) \longrightarrow \\ \qquad \big(requestTerminationCall\,.\,m\,.\,MyPEH3 \longrightarrow \quad requestTerminationRet\,.\,m\,.\,MyPEH3\,?\,requestTermination \longrightarrow \quad \mathbf{S} \\ []\ (count = 10) \longrightarrow \mathbf{Skip} \\ \mathbf{fi}\,; \\ \mathbf{Skip} \end{array}\right. \\ handleAsyncEventRet\,.\,MyPEH3 \longrightarrow \\ \mathbf{Skip} \end{array}\right.$$

$$Methods \mathrel{\widehat=}$$
$$\big(handleAsyncEvent\big)\ ;\ \ Methods$$

$$\bullet\ (Methods) \mathbin{\triangle} (end\_periodic\_app\,.\,MyPEH3 \longrightarrow \mathbf{Skip})$$

end

class $\mathit{MyPEH3Class} \mathrel{\widehat=}$ **begin**

state *State*

$\mathit{count} : \mathbb{Z}$
$m : \mathit{Mission}$

state *State*

initial *Init*

State'

$\mathit{count}' = 0$

• **Skip**

end