nestedSequencer2

Tight Rope v0.75

24th February 2017

1 ID Files

1.1 MissionIds

section *MissionIds* **parents** *scj_prelude*, *MissionId*

$$TopMission1MID : MissionID$$
$$MyMission1MID : MissionID$$
$$MyMission2MID : MissionID$$
$$MyMission3MID : MissionID$$

$$distinct\langle nullMissionId, TopMission1MID, MyMission1MID, MyMission2MID, MyMission3MID\rangle$$

1.2 SchedulablesIds

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

$$
\begin{array}{l}
MySequencerSID : SchedulableID \\
FirstMissionSequencerSID : SchedulableID \\
SecondMissionSequencerSID : SchedulableID \\
ThirdMissionSequencerSID : SchedulableID \\
MyPEH1SID : SchedulableID \\
MyPEH2SID : SchedulableID \\
MyPEH3SID : SchedulableID \\
\hline
distinct\langle nullSequencerId, nullSchedulableId, MySequencerSID, \\
FirstMissionSequencerSID, SecondMissionSequencerSID, \\
ThirdMissionSequencerSID, MyPEH1SID, \\
MyPEH2SID, MyPEH3SID\rangle
\end{array}
$$

1.3 Non-Paradigm Objects

1.4 ThreadIds

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

$$SafeletTId : ThreadID$$
$$nullThreadId : ThreadID$$

$$distinct\langle SafeletTId, nullThreadId\rangle$$

1.5 ObjectIds

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

$$\vdash \; distinct\langle\rangle$$

2 Network

2.1 Network Channel Sets

section *NetworkChannels* **parents** *scj_prelude*, *MissionId*, *MissionIds*,
SchedulableId, *SchedulableIds*, *MissionChan*, *TopLevelMissionSequencerFWChan*,
FrameworkChan, *SafeletChan*, *AperiodicEventHandlerChan*, *ManagedThreadChan*,
OneShotEventHandlerChan, *PeriodicEventHandlerChan*, *MissionSequencerMethChan*

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 *SafeltAppSync* $\widehat{=}$
{| *getSequencerCall*, *getSequencerRet*, *initializeApplicationCall*, *initializeApplicationRet*, *end_safelet_app* |}

channelset *MissionSequencerAppSync* ==
{| *getNextMissionCall*, *getNextMissionRet*, *end_sequencer_app* |}

channelset *MissionAppSync* ==
{| *initializeCall*, *register*, *initializeRet*, *cleanupMissionCall*, *cleanupMissionRet* |}

channelset *AppSync* ==
$\bigcup\{$*SafeltAppSync*, *MissionSequencerAppSync*, *MissionAppSync*,
MTAppSync, *OSEHSync*, *APEHSync*, *PEHSync*,
{| *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 $Tier0Sync ==$
$\{\!|\ done\_toplevel\_sequencer, done\_safeletFW,$
$start\_mission\ .\ MyMission1, done\_mission\ .\ MyMission1,$
$initializeRet\ .\ MyMission1, requestTermination\ .\ MyMission1\ .\ MySequencer\ |\!\}$

channelset $Tier1Sync ==$
$\{\!|\ done\_toplevel\_sequencer, done\_safeletFW,$
$start\_mission\ .\ MyMission2, done\_mission\ .\ MyMission2,$
$initializeRet\ .\ MyMission2, requestTermination\ .\ MyMission2\ .\ |\!\}$

channelset $Tier2Sync ==$
$\{\!|\ done\_toplevel\_sequencer, done\_safeletFW,$
$start\_mission\ .\ MyMission3, done\_mission\ .\ MyMission3,$
$initializeRet\ .\ MyMission3, requestTermination\ .\ MyMission3\ .\ |\!\}$

2.2 Locking

section *NetworkLocking* **parents** *scj_prelude*, *GlobalTypes*, *FrameworkChan*, *MissionId*, *MissionIds*, *ThreadIds*, *NetworkChannels*, *ObjectFW*, *ThreadFW*, *Priority*

process *Threads* $\widehat{=}$
$(\mathbf{Skip})$

process *Objects* $\widehat{=}$
$(\mathbf{Skip})$

process *Locking* $\widehat{=}$ *Threads* $[\![$ *ThreadSync* $]\!]$ *Objects*

2.3 Program

section *Program* **parents** *scj_prelude*, *MissionId*, *MissionIds*,
SchedulableId, *SchedulableIds*, *MissionChan*, *SchedulableMethChan*, *MissionFW*,
SafeletFW, *TopLevelMissionSequencerFW*, *NetworkChannels*, *ManagedThreadFW*,
SchedulableMissionSequencerFW, *PeriodicEventHandlerFW*, *OneShotEventHandlerFW*,
AperiodicEventHandlerFW, *ObjectFW*, *ThreadFW*,
MyAppApp, *MySequencerApp*, *TopMission*1*App*, *FirstMissionSequencerApp*, *SecondMissionSequencerApp*,
ThirdMissionSequencerApp, *MyMission*1*App*, *MyPEH*1*App*, *MyMission*2*App*, *MyPEH*2*App*, *MyMission*3*App*, *MyPE*

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

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

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

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

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

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

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

process $Program \mathrel{\widehat{=}} \big(Framework \;[\![\; AppSync \;]\!]\; Application\big) \;[\![\; LockingSync \;]\!]\; Locking$

3 Safelet

section *MyAppApp* **parents** *scj_prelude*, *SchedulableId*, *SchedulableIds*, *SafeletChan*, *MethodCallBindingChannels*

process $MyAppApp \mathrel{\widehat=} \textbf{begin}$

$$InitializeApplication \mathrel{\widehat=} \begin{pmatrix} initializeApplicationCall \longrightarrow \\ initializeApplicationRet \longrightarrow \\ \textbf{Skip} \end{pmatrix}$$

$$GetSequencer \mathrel{\widehat=} \begin{pmatrix} getSequencerCall \longrightarrow \\ getSequencerRet\,!\,MySequencerSID \longrightarrow \\ \textbf{Skip} \end{pmatrix}$$

$$Methods \mathrel{\widehat=} \begin{pmatrix} GetSequencer \\ \Box \\ InitializeApplication \end{pmatrix} ;\ Methods$$

$\bullet\ (Methods) \triangle (end\_safelet\_app \longrightarrow \textbf{Skip})$

end

4 Top Level Mission Sequencer

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

process $MySequencerApp \mathrel{\widehat=} \textbf{begin}$

State

$this : \textbf{ref}\ MySequencerClass$

state *State*

Init

$State'$

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

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

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

$\bullet\ (Init\ ;\ Methods) \triangle (end\_sequencer\_app\ .\ MySequencerSID \longrightarrow \textbf{Skip})$

end

section *MySequencerClass* **parents** *scj_prelude*, *SchedulableId*, *SchedulableIds*, *SafeletChan*
, *MethodCallBindingChannels*, *MissionId*, *MissionIds*

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

state *State*

$$done : \mathbb{B}$$

state *State*

initial *Init*

$$State'$$

$$done' = false$$

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

$$\left(\begin{array}{l} \textbf{if } (done = \textbf{False}) \longrightarrow \\ \quad \left(\begin{array}{l} done := \textbf{True}; \\ ret := TopMission1MID \end{array}\right) \\ [] \neg (done = \textbf{False}) \longrightarrow \\ \quad \left(ret := nullMissionId\right) \\ \textbf{fi} \end{array}\right)$$

• **Skip**

end

5 Missions

5.1 TopMission1

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

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

State

$this : \mathbf{ref}\ TopMission1Class$

state $State$

Init

$State'$

$this' = \mathbf{new}\ TopMission1Class()$

$$InitializePhase \mathrel{\widehat=} \left(\begin{array}{l} initializeCall\ .\ TopMission1MID \longrightarrow \\ register\ !\ FirstMissionSequencerSID\ !\ TopMission1MID \longrightarrow \\ register\ !\ SecondMissionSequencerSID\ !\ TopMission1MID \longrightarrow \\ register\ !\ ThirdMissionSequencerSID\ !\ TopMission1MID \longrightarrow \\ initializeRet\ .\ TopMission1MID \longrightarrow \\ \mathbf{Skip} \end{array}\right)$$

$$CleanupPhase \mathrel{\widehat=} \left(\begin{array}{l} \mathbf{var}\ \mathbb{B} : ret \bullet \\ cleanupMissionCall\ .\ TopMission1MID \longrightarrow \\ cleanupMissionRet\ .\ TopMission1MID\ !\ \mathbf{True} \longrightarrow \\ \mathbf{Skip} \end{array}\right)$$

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

$$\bullet\ (Init ;\ Methods) \triangle (end\_mission\_app\ .\ TopMission1MID \longrightarrow \mathbf{Skip})$$

end

5.2 Schedulables of TopMission1

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

process $FirstMissionSequencerApp \mathrel{\widehat=} \textbf{begin}$

State

$this : \textbf{ref}\ FirstMissionSequencerClass$

state *State*

Init

$State'$

$this' = \textbf{new}\ FirstMissionSequencerClass()$

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

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

$\bullet\ (Init\ ;\ Methods) \triangle (end\_sequencer\_app\,.\,FirstMissionSequencerSID \longrightarrow \textbf{Skip})$

end

section *FirstMissionSequencerClass* **parents** *scj_prelude*, *SchedulableId*, *SchedulableIds*, *SafeletChan*, *MethodCallBindingChannels*, *MissionId*, *MissionIds*

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

state *State*

$$done : \mathbb{B}$$

state *State*

initial *Init*

$$State'$$

$$done' = false$$

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

$$\left(\begin{array}{l} \textbf{if } (done = \textbf{False}) \longrightarrow \\ \quad \left(\begin{array}{l} done := \textbf{True}; \\ ret := MyMission1MID \end{array}\right) \\ [] \neg (done = \textbf{False}) \longrightarrow \\ \quad \left(ret := nullMissionId\right) \\ \textbf{fi} \end{array}\right)$$

$\bullet$ **Skip**

end

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

process $SecondMissionSequencerApp \mathrel{\widehat=} \textbf{begin}$

State

$this : \textbf{ref}\ SecondMissionSequencerClass$

state *State*

Init

$State'$

$this' = \textbf{new}\ SecondMissionSequencerClass()$

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

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

$\bullet\ (Init\ ;\ Methods) \triangle (end\_sequencer\_app\,.\,SecondMissionSequencerSID \longrightarrow \textbf{Skip})$

end

section *SecondMissionSequencerClass* **parents** *scj_prelude*, *SchedulableId*, *SchedulableIds*, *SafeletChan*, *MethodCallBindingChannels*, *MissionId*, *MissionIds*

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

state *State*

$$done : \mathbb{B}$$

state *State*

initial *Init*

$$State'$$

$$done' = false$$

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

$$\left(\begin{array}{l} \textbf{if } (done = \textbf{False}) \longrightarrow \\ \quad \left(\begin{array}{l} done := \textbf{True}; \\ ret := MyMission2MID \end{array}\right) \\ [] \neg (done = \textbf{False}) \longrightarrow \\ \quad \left(ret := nullMissionId\right) \\ \textbf{fi} \end{array}\right)$$

• **Skip**

end

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

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

State

$this : \mathbf{ref}\ ThirdMissionSequencerClass$

state *State*

Init

$State'$

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

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

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

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

end

section *ThirdMissionSequencerClass* **parents** *scj_prelude*, *SchedulableId*, *SchedulableIds*, *SafeletChan*, *MethodCallBindingChannels*, *MissionId*, *MissionIds*

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

state *State*

$$done : \mathbb{B}$$

state *State*

initial *Init*

$$State'$$

$$done' = false$$

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

$$\left(\begin{array}{l} \textbf{if } (done = \textbf{False}) \longrightarrow \\ \quad \left(\begin{array}{l} done := \textbf{True}; \\ ret := MyMission3MID \end{array}\right) \\ [] \neg (done = \textbf{False}) \longrightarrow \\ \quad \left(ret := nullMissionId\right) \\ \textbf{fi} \end{array}\right)$$

$\bullet$ **Skip**

end

5.3 MyMission1

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

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

State

$this : \mathbf{ref}\ MyMission1Class$

state $State$

Init

$State'$

$this' = \mathbf{new}\ MyMission1Class()$

$$InitializePhase \mathrel{\widehat=} \begin{pmatrix} initializeCall\ .\ MyMission1MID \longrightarrow \\ register\ !\ MyPEH1SID\ !\ MyMission1MID \longrightarrow \\ initializeRet\ .\ MyMission1MID \longrightarrow \\ \mathbf{Skip} \end{pmatrix}$$

$$CleanupPhase \mathrel{\widehat=} \begin{pmatrix} \mathbf{var}\ \mathbb{B} : ret \bullet \\ cleanupMissionCall\ .\ MyMission1MID \longrightarrow \\ cleanupMissionRet\ .\ MyMission1MID\ !\ \mathbf{True} \longrightarrow \\ \mathbf{Skip} \end{pmatrix}$$

$$Methods \mathrel{\widehat=} \begin{pmatrix} InitializePhase \\ \Box \\ CleanupPhase \end{pmatrix} ;\ Methods$$

$$\bullet\ (Init\ ;\ Methods) \triangle (end\_mission\_app\ .\ MyMission1MID \longrightarrow \mathbf{Skip})$$

end

5.4 Schedulables of MyMission1

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

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

State

$this : \mathbf{ref}\ MyPEH1Class$

state $State$

Init

$State'$

$this' = \mathbf{new}\ MyPEH1Class()$

$$handleAsyncEvent \mathrel{\widehat=}$$
$$\left(\begin{array}{l} handleAsyncEventCall\,.\,MyPEH1SID \longrightarrow \\ \left(\begin{array}{l} count := count + 1; \\ \mathbf{if}\ (count = 10) \longrightarrow \\ \quad \left(\begin{array}{l} requestTerminationCall\,.\,m\,.\,MyPEH1SID \longrightarrow \\ requestTerminationRet\,.\,m\,.\,MyPEH1SID\,?\,requestTermination \longrightarrow \end{array}\right) \\ \quad \mathbf{Skip} \\ [\!]\ \neg\,(count = 10) \longrightarrow \mathbf{Skip} \\ \mathbf{fi} \end{array}\right) \\ handleAsyncEventRet\,.\,MyPEH1SID \longrightarrow \\ \mathbf{Skip} \end{array}\right);$$

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

$$\bullet\ (Init\ ;\ Methods) \triangle (end\_periodic\_app\,.\,MyPEH1SID \longrightarrow \mathbf{Skip})$$

end

section $MyPEH1Class$ **parents** $scj\_prelude, SchedulableId, SchedulableIds, SafeletChan$, $MethodCallBindingChannels$

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

state $State$

$count : \mathbb{Z}$

state $State$

initial $Init$

$State'$

$count' = 0$

- **Skip**

end

5.5 MyMission2

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

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

State

$this : \mathbf{ref}\ MyMission2Class$

state *State*

Init

$State'$

$this' = \mathbf{new}\ MyMission2Class()$

$$InitializePhase \mathrel{\widehat=} \begin{pmatrix} initializeCall\ .\ MyMission2MID \longrightarrow \\ register\ !\ MyPEH2SID\ !\ MyMission2MID \longrightarrow \\ initializeRet\ .\ MyMission2MID \longrightarrow \\ \mathbf{Skip} \end{pmatrix}$$

$$CleanupPhase \mathrel{\widehat=} \begin{pmatrix} \mathbf{var}\ \mathbb{B} : ret \bullet \\ cleanupMissionCall\ .\ MyMission2MID \longrightarrow \\ cleanupMissionRet\ .\ MyMission2MID\ !\ \mathbf{True} \longrightarrow \\ \mathbf{Skip} \end{pmatrix}$$

$$Methods \mathrel{\widehat=} \begin{pmatrix} InitializePhase \\ \Box \\ CleanupPhase \end{pmatrix} ;\ Methods$$

$$\bullet\ (Init\ ;\ Methods) \triangle (end\_mission\_app\ .\ MyMission2MID \longrightarrow \mathbf{Skip})$$

end

5.6 Schedulables of MyMission2

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

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

State

$this : \mathbf{ref}\ MyPEH2Class$

state *State*

Init

$State'$

$this' = \mathbf{new}\ MyPEH2Class()$

$$handleAsyncEvent \mathrel{\widehat=}$$
$$\left(\begin{array}{l} handleAsyncEventCall\,.\,MyPEH2SID \longrightarrow \\ \left(\begin{array}{l} count := count + 1; \\ \mathbf{if}\ (count = 10) \longrightarrow \\ \quad \left(\begin{array}{l} requestTerminationCall\,.\,m\,.\,MyPEH2SID \longrightarrow \\ requestTerminationRet\,.\,m\,.\,MyPEH2SID\,?\,requestTermination \longrightarrow \end{array}\right) \\ \quad \mathbf{Skip} \\ [\!]\ \neg\,(count = 10) \longrightarrow \mathbf{Skip} \\ \mathbf{fi} \end{array}\right) \\ handleAsyncEventRet\,.\,MyPEH2SID \longrightarrow \\ \mathbf{Skip} \end{array}\right);$$

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

$$\bullet\ (Init\ ;\ Methods) \triangle (end\_periodic\_app\,.\,MyPEH2SID \longrightarrow \mathbf{Skip})$$

end

section *MyPEH2Class* **parents** *scj_prelude*, *SchedulableId*, *SchedulableIds*, *SafeletChan*
, *MethodCallBindingChannels*

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

state *State*

$$count : \mathbb{Z}$$

state *State*

initial *Init*

$$State'$$

$$count' = 0$$

- **Skip**

end

5.7 MyMission3

section $MyMission3App$ **parents** $scj\_prelude$, $MissionId$, $MissionIds$, $SchedulableId$, $SchedulableIds$, $MissionChan$, $SchedulableMethChan$, $MyMission3MethChan$, $MethodCallBindingChannels$

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

State

$this : \mathbf{ref}\ MyMission3Class$

state $State$

Init

$State'$

$this' = \mathbf{new}\ MyMission3Class()$

$$InitializePhase \mathrel{\widehat=} \begin{pmatrix} initializeCall\ .\ MyMission3MID \longrightarrow \\ register\ !\ MyPEH3SID\ !\ MyMission3MID \longrightarrow \\ initializeRet\ .\ MyMission3MID \longrightarrow \\ \mathbf{Skip} \end{pmatrix}$$

$$CleanupPhase \mathrel{\widehat=} \begin{pmatrix} \mathbf{var}\ \mathbb{B} : ret \bullet \\ cleanupMissionCall\ .\ MyMission3MID \longrightarrow \\ cleanupMissionRet\ .\ MyMission3MID\ !\ \mathbf{True} \longrightarrow \\ \mathbf{Skip} \end{pmatrix}$$

$$Methods \mathrel{\widehat=} \begin{pmatrix} InitializePhase \\ \Box \\ CleanupPhase \end{pmatrix} ;\ Methods$$

$$\bullet\ (Init\ ;\ Methods) \triangle (end\_mission\_app\ .\ MyMission3MID \longrightarrow \mathbf{Skip})$$

end

5.8 Schedulables of MyMission3

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

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

State

$this : \mathbf{ref}\ MyPEH3Class$

state *State*

Init

$State'$

$this' = \mathbf{new}\ MyPEH3Class()$

$handleAsyncEvent \mathrel{\widehat=}$
$\Big(handleAsyncEventCall\,.\,MyPEH3SID \longrightarrow$
$\Big(\Big(count := count + 1;$
$\mathbf{if}\ (count = 10) \longrightarrow$
$\Big(requestTerminationCall\,.\,m\,.\,MyPEH3SID \longrightarrow$
$requestTerminationRet\,.\,m\,.\,MyPEH3SID\,?\,requestTermination \longrightarrow\Big)$
$\mathbf{Skip}$
$[]\ \neg\ (count = 10) \longrightarrow \mathbf{Skip}$
$\mathbf{fi}\Big)\Big)\ ;$
$handleAsyncEventRet\,.\,MyPEH3SID \longrightarrow$
$\mathbf{Skip}\Big)$

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

$\bullet\ (Init\ ;\ Methods) \triangle (end\_periodic\_app\,.\,MyPEH3SID \longrightarrow \mathbf{Skip})$

end

section *MyPEH3Class* **parents** *scj_prelude*, *SchedulableId*, *SchedulableIds*, *SafeletChan*
, *MethodCallBindingChannels*

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

state *State*

$$count : \mathbb{Z}$$

state *State*

initial *Init*

$$State'$$

$$count' = 0$$

- **Skip**

end