mission1 Report

4th October 2016

1 ID Files

1.1 MissionIds

$$\textbf{section}\ MissionIds\ \textbf{parents}\ scj\_prelude, MissionId$$

$$\begin{array}{|l}
MyMissionMID : MissionID \\
\hline
distinct \langle nullMissionId, MyMissionMID \rangle
\end{array}$$

1.2 SchedulablesIds

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

$$
\begin{array}{|l}
\mathit{mainSequencerSID} : \mathit{SchedulableID} \\
\mathit{APEHSID} : \mathit{SchedulableID} \\
\mathit{PEHSID} : \mathit{SchedulableID} \\
\hline
\mathit{distinct}\langle \mathit{nullSequencerId}, \mathit{nullSchedulableId}, \mathit{mainSequencerSID}, \\
\mathit{APEHSID}, \mathit{PEHSID}\rangle
\end{array}
$$

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* . *MyMission*, *done_mission* . *MyMission*,
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* ⦄

2.2 Locking

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

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

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

process *Locking* $\widehat{=}$ *Threads* $\llbracket$ *ThreadSync* $\rrbracket$ *Objects*

2.3 Program

$$\begin{array}{l}
\mathbf{section}\ Program\ \mathbf{parents}\ scj\_prelude, MissionId, MissionIds, \\
\quad SchedulableId, SchedulableIds, MissionChan, SchedulableMethChan, MissionFW, \\
\quad SafeletFW, TopLevelMissionSequencerFW, NetworkChannels, ManagedThreadFW, \\
\quad SchedulableMissionSequencerFW, PeriodicEventHandlerFW, OneShotEventHandlerFW, \\
\quad AperiodicEventHandlerFW, ObjectFW, ThreadFW, \\
\quad MyAppApp, mainSequencerApp, MyMissionApp, APEHApp, PEHApp
\end{array}$$

$$\begin{array}{l}
\mathbf{process}\ ControlTier \mathrel{\widehat=} \\
\left(\begin{array}{l}
SafeletFW \\
\quad \llbracket ControlTierSync \rrbracket \\
TopLevelMissionSequencerFW(mainSequencer)
\end{array}\right)
\end{array}$$

$$\begin{array}{l}
\mathbf{process}\ Tier0 \mathrel{\widehat=} \\
\left(\begin{array}{l}
MissionFW(MyMissionID) \\
\quad \llbracket MissionSync \rrbracket \\
\left(\begin{array}{l}
OneShotEventHandlerFW(APEHID, (time(5,0), null)) \\
\quad \llbracket SchedulablesSync \rrbracket \\
AperiodicEventHandlerFW(PEHID, (time(60,0), time(5,0), NULL, nullSchedulableId))
\end{array}\right)
\end{array}\right)
\end{array}$$

$$\begin{array}{l}
\mathbf{process}\ Framework \mathrel{\widehat=} \\
\left(\begin{array}{l}
ControlTier \\
\quad \llbracket TierSync \rrbracket \\
(Tier0)
\end{array}\right)
\end{array}$$

$$\begin{array}{l}
\mathbf{process}\ Application \mathrel{\widehat=} \\
\left(\begin{array}{l}
MyAppApp \\
\interleave \\
mainSequencerApp \\
\interleave \\
MyMissionApp \\
\interleave \\
APEHApp(MyMissionID) \\
\interleave \\
PEHApp(apehID)
\end{array}\right)
\end{array}$$

$$\mathbf{process}\ Program \mathrel{\widehat=} \left(Framework \llbracket AppSync \rrbracket Application\right) \llbracket LockingSync \rrbracket Locking$$

3 Safelet

$\mathbf{section}\ MyAppApp\ \mathbf{parents}\ scj\_prelude, SchedulableId, SchedulableIds, SafeletChan, MethodCallBindingChannels$

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

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

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

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

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

$\mathbf{end}$

4 Top Level Mission Sequencer

$\mathbf{section}\ mainSequencerApp\ \mathbf{parents}\ TopLevelMissionSequencerChan,$
$\quad MissionId, MissionIds, SchedulableId, SchedulableIds, mainSequencerClass, MethodCallBindingChannels$

$\mathbf{process}\ mainSequencerApp \mathrel{\widehat=}$
$\quad name : String \bullet \mathbf{begin}$

$State$

$$this : \mathbf{ref}\ mainSequencerClass$$

$\mathbf{state}\ State$

$Init$

$$State\,'$$

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

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

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

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

end

$\mathbf{section}\ mainSequencerClass\ \mathbf{parents}\ scj\_prelude, SchedulableId, SchedulableIds, SafeletChan$
$, MethodCallBindingChannels, MissionId, MissionIds$

$\mathbf{class}\ mainSequencerClass \mathrel{\widehat=} \mathbf{begin}$

$\mathbf{state}\ State$

$notReleased : \mathbb{B}$

$\mathbf{state}\ State$

$\mathbf{initial}\ Init$

$State~'$

$notReleased = \mathbf{True}$

$$
\mathbf{protected}\ getNextMission \mathrel{\widehat=} \mathbf{var}\ ret : MissionID \bullet
\left(\begin{array}{l}
\mathbf{if}\ notReleased = \mathbf{True} \longrightarrow \\
\qquad \left(\begin{array}{l}
\mathbf{var}\ mission : MissionID \bullet mission := MyMissionMID; \\
this \,.\, notReleased := \mathbf{False}; \\
ret := mission
\end{array}\right) \\
[\!]\ \neg\ notReleased = \mathbf{True} \longrightarrow \\
\qquad \left(ret := nullMissionId\right) \\
\mathbf{fi}
\end{array}\right)
$$

$\bullet$ **Skip**

end

5 Missions

5.1 MyMission

$\mathbf{section}\ MyMissionApp\ \mathbf{parents}\ scj\_prelude, MissionId, MissionIds,$
$\quad SchedulableId, SchedulableIds, MissionChan, SchedulableMethChan, MyMissionMethChan$
$, MethodCallBindingChannels$

$\mathbf{process}\ MyMissionApp \mathrel{\widehat=} \mathbf{begin}$

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

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

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

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

$\mathbf{end}$

5.2 Schedulables of MyMission

section $APEHApp$ **parents** $AperiodicEventHandlerChan, SchedulableId, SchedulableIds, MethodCallBindingChannels$

process $APEHApp \mathrel{\widehat{=}}$
$controllingMission : MissionID \bullet$ **begin**

$$
\begin{array}{l}
handleAsyncEvent \mathrel{\widehat{=}} \\
\left(\begin{array}{l}
handleAsyncEventCall \,.\, APEHSID \longrightarrow \\
\left(\begin{array}{l}
requestTerminationCall \,.\, controllingMission \,.\, APEHSID \longrightarrow \\
requestTerminationRet \,.\, controllingMission \,.\, APEHSID \,?\, requestTermination \longrightarrow \\
\mathbf{Skip}
\end{array}\right) ; \\
handleAsyncEventRet \,.\, APEHSID \longrightarrow \\
\mathbf{Skip}
\end{array}\right)
\end{array}
$$

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

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

end

$\mathbf{section}\ \mathit{PEHApp}\ \mathbf{parents}\ \mathit{PeriodicEventHandlerChan}, \mathit{SchedulableId}, \mathit{SchedulableIds}, \mathit{MethodCallBindingChannels}$

$$\begin{array}{l}
\mathbf{process}\, \mathit{PEHApp} \mathrel{\widehat=} \\
\quad \mathit{apeh} : \mathit{SchedulableID} \bullet \mathbf{begin}
\end{array}$$

$$\begin{array}{l}
\mathit{handleAsyncEvent} \mathrel{\widehat=} \\
\left(\begin{array}{l}
\mathit{handleAsyncEventCall} \,.\, \mathit{PEHSID} \longrightarrow \\
\left(\begin{array}{l}
\mathit{releaseCall} \,.\, \mathit{apeh} \,.\, \mathit{PEHSID} \longrightarrow \\
\mathbf{Skip}
\end{array}\right) ; \\
\mathit{handleAsyncEventRet} \,.\, \mathit{PEHSID} \longrightarrow \\
\mathbf{Skip}
\end{array}\right)
\end{array}$$

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

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

$\mathbf{end}$