

1 Network

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

```
channelset TerminateSync ==  
    { schedulables_terminated, schedulables_stopped, get_activeSchedulables }
```

```
channelset SafeletTierSync ==  
    { start_toplevel_sequencer, done_toplevel_sequencer, done_safeletFW }
```

```
channelset TierSync ==  
    { start_mission.MainMission, done_mission.MainMission,  
      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 Tier0Sync ==  
    TierCommonSync  
    ∪  
    { start_mission.MainMission, done_mission.MainMission,  
      initializeRet.MainMission, requestTermination.MainMission.MainMissionSequencer }
```

section *Program* **parents** *scj_prelude, MissionId, MissionIds,*
SchedulableId, SchedulableIds, MissionChan, SchedulableMethChan, MissionFW,
SafeletFW, TopLevelMissionSequencerFW, NetworkChannels, ManagedThreadFW,
SchedulableMissionSequencerFW, PeriodicEventHandlerFW, OneShotEventHandlerFW,
AperiodicEventHandlerFW, TestSafeletApp, MainMissionSequencerApp,
MainMissionApp, NestedMissionSequencerAppNestedMissionApp, NestedOneShotEventHandlerApp

process *ControlTier* $\hat{=}$

$$\left(\begin{array}{l} \textit{SafeletFW} \\ \llbracket \textit{TierSync} \rrbracket \\ \textit{TopLevelMissionSequencerFW}(\textit{MainMissionSequencer}) \end{array} \right)$$

process *Tier0* $\hat{=}$

$$\left(\begin{array}{l} \textit{MissionFW}(\textit{MainMission}) \\ \llbracket \textit{MissionSync} \rrbracket \\ (\textit{SchedulableMissionSequencerFW}(\textit{NestedMissionSequencer})) \end{array} \right)$$

process *Tier1* $\hat{=}$

$$\left(\begin{array}{l} \textit{MissionFW}(\textit{NestedMission}) \\ \llbracket \textit{MissionSync} \rrbracket \\ (\textit{OneShotEventHandlerFW}(\textit{NestedOneShotEventHandler})) \end{array} \right)$$

process *Framework* $\hat{=}$

$$\left(\begin{array}{l} \textit{ControlTier} \\ \llbracket \textit{TierSync} \rrbracket \\ \left(\begin{array}{l} \textit{Tier0} \\ \llbracket \textit{Tier0Sync} \rrbracket \end{array} \right) \\ \textit{Tier1} \end{array} \right)$$

process *Application* $\hat{=}$

$$\left(\begin{array}{l} \textit{TestSafeletApp} \\ ||| \\ \textit{MainMissionSequencerApp} \\ ||| \\ \textit{MainMissionApp} \\ ||| \\ \textit{NestedMissionSequencerApp} \\ ||| \\ \textit{NestedMissionApp} \\ ||| \\ \textit{NestedOneShotEventHandlerApp} \end{array} \right)$$

process *Program* $\hat{=}$ *Framework* $\llbracket \textit{AppSync} \rrbracket$ *Application*

2 ID Files

2.1 MissionIds

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

MainMission : MissionID
NestedMission : MissionID

distinct(*nullMissionId, schedulable, schedulable*)

2.2 SchedulablesIds

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

NestedMissionSequencer : SchedulableID
NestedOneShotEventHandler : SchedulableID

distinct(*nullSequencerId, nullSchedulableId, schedulable, schedulable*)

3 Safelet

section *TestSafeletApp* **parents** *scj_prelude, SchedulableId, SchedulableIds, SafeletChan*

process *TestSafeletApp* $\hat{=}$ **begin**

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

GetSequencer $\hat{=}$
 $\left(\begin{array}{l} \textit{getSequencerCall} \longrightarrow \\ \textit{getSequencerRet} ! \textit{MainMissionSequencer} \longrightarrow \\ \mathbf{Skip} \end{array} \right)$

Methods $\hat{=}$
 $\left(\begin{array}{l} \textit{GetSequencer} \\ \square \\ \textit{InitializeApplication} \end{array} \right); \textit{Methods}$

• $(\textit{Methods}) \triangle (\textit{end_safelet_app} \longrightarrow \mathbf{Skip})$

end

4 Top Level Mission Sequencer

section *MainMissionSequencerApp* **parents** *TopLevelMissionSequencerChan*,
MissionId, *MissionIds*, *SchedulableId*

process *MainMissionSequencerApp* $\hat{=}$ **begin**

GetNextMission $\hat{=}$
 $\left(\begin{array}{l} \textit{getNextMissionCall} . \textit{MainMissionSequencer} \longrightarrow \\ \textit{getNextMissionRet} . \textit{MainMissionSequencer} ! \textit{MainMission} \longrightarrow \\ \mathbf{Skip} \end{array} \right)$

Methods $\hat{=}$
 $(\textit{GetNextMission}) ; \textit{Methods}$

$\bullet (\textit{Methods}) \triangle (\textit{end_sequencer_app} . \textit{MainMissionSequencer} \longrightarrow \mathbf{Skip})$

end

5 Missions

5.1 MainMission

section *MainMissionApp* **parents** *scj_prelude*, *MissionId*, *MissionIds*,
SchedulableId, *SchedulableIds*, *MissionChan*, *SchedulableMethChan*

process *MainMissionApp* $\hat{=}$ **begin**

InitializePhase $\hat{=}$
$$\left(\begin{array}{l} \textit{initializeCall} . \textit{MainMission} \longrightarrow \\ \textit{register} ! \textit{NestedMissionSequencer} ! \textit{MainMission} \longrightarrow \\ \textit{initializeRet} . \textit{MainMission} \longrightarrow \\ \mathbf{Skip} \end{array} \right)$$

CleanupPhase $\hat{=}$
$$\left(\begin{array}{l} \textit{cleanupMissionCall} . \textit{MainMission} \longrightarrow \\ \textit{cleanupMissionRet} . \textit{MainMission} ? \mathbf{False} \longrightarrow \\ \mathbf{Skip} \end{array} \right)$$

Methods $\hat{=}$
$$\left(\begin{array}{l} \textit{InitializePhase} \\ \square \\ \textit{CleanupPhase} \end{array} \right) ; \textit{Methods}$$

• $(\textit{Methods}) \triangle (\textit{end_mission_app} . \textit{MainMission} \longrightarrow \mathbf{Skip})$

end

5.2 Schedulables of MainMission

section *NestedMissionSequencerApp* **parents** *TopLevelMissionSequencerChan*,
MissionId, *MissionIds*, *SchedulableId*

process *NestedMissionSequencerApp* $\hat{=}$ **begin**

GetNextMission $\hat{=}$
 $\left(\begin{array}{l} \text{getNextMissionCall} . \text{NestedMissionSequencer} \longrightarrow \\ \text{getNextMissionRet} . \text{NestedMissionSequencer} ! \text{NestedMission} \longrightarrow \\ \mathbf{Skip} \end{array} \right)$

Methods $\hat{=}$
 $(\text{GetNextMission}) ; \text{Methods}$

$\bullet (\text{Methods}) \triangle (\text{end_sequencer_app} . \text{NestedMissionSequencer} \longrightarrow \mathbf{Skip})$

end

5.3 NestedMission

section *NestedMissionApp* **parents** *scj_prelude*, *MissionId*, *MissionIds*,
SchedulableId, *SchedulableIds*, *MissionChan*, *SchedulableMethChan*

process *NestedMissionApp* $\hat{=}$ **begin**

InitializePhase $\hat{=}$
 $\left(\begin{array}{l} \text{initializeCall} . \text{NestedMission} \longrightarrow \\ \text{register} ! \text{NestedOneShotEventHandler} ! \text{NestedMission} \longrightarrow \\ \text{initializeRet} . \text{NestedMission} \longrightarrow \\ \mathbf{Skip} \end{array} \right)$

CleanupPhase $\hat{=}$
 $\left(\begin{array}{l} \text{cleanupMissionCall} . \text{NestedMission} \longrightarrow \\ \text{cleanupMissionRet} . \text{NestedMission} ? \mathbf{False} \longrightarrow \\ \mathbf{Skip} \end{array} \right)$

Methods $\hat{=}$
 $\left(\begin{array}{l} \text{InitializePhase} \\ \square \\ \text{CleanupPhase} \end{array} \right) ; \text{Methods}$

$\bullet (\text{Methods}) \triangle (\text{end_mission_app} . \text{NestedMission} \longrightarrow \mathbf{Skip})$

end

5.4 Schedulables of NestedMission

section *NestedOneShotEventHandlerApp* **parents** *OneShotEventHandlerChan*, *SchedulableId*, *SchedulableIds*

process *NestedOneShotEventHandlerApp* $\hat{=}$ **begin**

Methods $\hat{=}$
handlerAsyncEvent ; *Methods*

handlerAsyncEvent $\hat{=}$
$$\left(\begin{array}{l} \text{handleAsyncEventCall} . \text{NestedOneShotEventHandler} \longrightarrow \\ \text{handleAsyncEventRet} . \text{NestedOneShotEventHandler} \longrightarrow \\ \mathbf{Skip} \end{array} \right)$$

• (*Methods*) \triangle (*end_oneShot_app* . *NestedOneShotEventHandler* \longrightarrow **Skip**)

end