

=====

An Event-B Specification of ASW_modes @ Date: 27 Oct 2008 @ Time: 05:31:40 PM

=====

MACHINE ASW_modes

REFINES Core_SW_modes

SEES ASW_modes_context

VARIABLES

TCbuffer
TMbuffer
TCpool
TMpool
TCstatus
csw_hk_on
csw_sci_on
csw_mode
mixsc_mode
mixst_mode
sixsp_mode
sixsx_mode
csw_in_sync
asw_in_sync

INVARIANTS

inv1 : *mixsc_mode* ∈ *MIXSC_MODES*
inv2 : *mixst_mode* ∈ *MIXST_MODES*
inv3 : *sixsp_mode* ∈ *SIXSP_MODES*
inv4 : *sixsx_mode* ∈ *SIXSX_MODES*
inv5 : *csw_in_sync* ∈ *BOOL*
inv6 : *asw_in_sync* ∈ *BOOL*
inv7 : *csw_in_sync* = *TRUE* ∧ *asw_in_sync* = *TRUE* ⇒ (*mixsc_mode* ↦ *csw_mode*) ∈ *ALLOWED_MIXSC_CS*
inv8 : *csw_in_sync* = *TRUE* ∧ *asw_in_sync* = *TRUE* ⇒ (*mixst_mode* ↦ *csw_mode*) ∈ *ALLOWED_MIXST_CS*
inv9 : *csw_in_sync* = *TRUE* ∧ *asw_in_sync* = *TRUE* ⇒ (*sixsp_mode* ↦ *csw_mode*) ∈ *ALLOWED_SIXSP_CS*
inv10 : *csw_in_sync* = *TRUE* ∧ *asw_in_sync* = *TRUE* ⇒ (*sixsx_mode* ↦ *csw_mode*) ∈ *ALLOWED_SIXSX_CS*

EVENTS

INITIALISATION

BEGIN

act1 : *TCbuffer* := ∅
act2 : *TMbuffer* := ∅
act3 : *TCpool* := ∅
act4 : *TMpool* := ∅
act5 : *TCstatus* := ∅
act6 : *csw_hk_on* := *FALSE*
act7 : *csw_sci_on* := *FALSE*
act8 : *csw_mode* := *csw_standby*
act9 : *mixsc_mode* :∈ *INITIAL_MIXSC_MODES*
act10 : *mixst_mode* :∈ *INITIAL_MIXST_MODES*
act11 : *sixsp_mode* :∈ *INITIAL_SIXSP_MODES*
act12 : *sixsx_mode* :∈ *INITIAL_SIXSX_MODES*

```

    act13 : csw_in_sync := TRUE
    act14 : asw_in_sync := TRUE
END

```

EVENT env_Receive_TC**REFINES** env_Receive_TC

```

ANY
  tc
  tc_pointer
WHERE
  grd1 : tc ∈ TC_SET
  grd2 : tc_pointer ∈ TC_BUFFER_ADDRESSES
  grd3 : tc_pointer ∉ dom(TCbuffer)
THEN
  act1 : TCbuffer(tc_pointer) := tc
END

```

EVENT Poll_TC**REFINES** Poll_TC

```

ANY
  tc_pointer
  tc_handler
WHERE
  grd1 : tc_pointer ∈ dom(TCbuffer)
  grd2 : tc_handler ∈ TCTM_POOL_ADDRESSES
  grd3 : tc_handler ∉ dom(TCpool)
  grd4 : tc_handler ∉ dom(TMpool)
THEN
  act1 : TCpool(tc_handler) := TCbuffer(tc_pointer)
  act3 : TCstatus(tc_handler) := Unchecked
  act2 : TCbuffer := {tc_pointer} ⧸ TCbuffer
END

```

EVENT Report_TC_check_ok**REFINES** Report_TC_check_ok

```

ANY
  tc_handler
  tm_handler
  tm
WHERE
  grd1 : tc_handler ∈ dom(TCpool)
  grd2 : tm_handler ∈ TCTM_POOL_ADDRESSES
  grd3 : tm ∈ TM_SET
  grd4 : tm_handler ∉ dom(TCpool)
  grd5 : tm_handler ∉ dom(TMpool)
  grd6 : TCstatus(tc_handler) = Unchecked
  grd7 : Valid_TC(TCpool(tc_handler)) = TRUE
  grd8 : Type_of_TM(tm) = Check_ok_TM
THEN
  act1 : TCstatus(tc_handler) := Waiting_for_Execution
  act2 : TMpool(tm_handler) := tm
END

```

EVENT Report_TC_check_fail**REFINES** Report_TC_check_fail

```

ANY
  tc_handler

```

```

    tm_handler
    tm
WHERE
    grd1 : tc_handler ∈ dom(TCpool)
    grd2 : tm_handler ∈ TCTM_POOL_ADDRESSES
    grd3 : tm ∈ TM_SET
    grd4 : tm_handler ∉ dom(TCpool)
    grd5 : tm_handler ∉ dom(TMpool)
    grd6 : TCstatus(tc_handler) = Unchecked
    grd7 : Valid_TC(TCpool(tc_handler)) = FALSE
    grd8 : Type_of_TM(tm) = Check_fail_TM
THEN
    act1 : TCpool := {tc_handler} ◁ TCpool
    act3 : TCstatus := {tc_handler} ◁ TCstatus
    act2 : TMpool(tm_handler) := tm
END

```

EVENT CSW_TC_Execution_Mode_change**REFINES** *TC_Execution_Mode_change*

```

ANY
    tc_handler
    new_mode
WHERE
    grd1 : tc_handler ∈ dom(TCpool)
    grd2 : TCstatus(tc_handler) = Waiting_for_Execution
    grd3 : Type_of_TC(TCpool(tc_handler)) ∈ MODE_CHANGING_TC
    grd4 : new_mode ∈ CSW_MODES
    grd5 : (csw_mode ↦ new_mode) ∈ CSW_TC_TRANSITIONS
    grd6 : csw_in_sync = TRUE
THEN
    act1 : TCstatus(tc_handler) := Successfully_Executed
    act3 : csw_mode := new_mode
    act2 : asw_in_sync := FALSE
END

```

EVENT CSW_Execution_Mode_sync**REFINES** *Non_TC_Execution_Mode_change*

```

ANY
    new_mode
WHERE
    grd4 : new_mode ∈ CSW_MODES
    grd5 : ((csw_mode ↦ new_mode) ∈ CSW_TC_TRANSITIONS) ∨ ((csw_mode ↦ new_mode) ∈ CSW_FDIR_TRANSITIONS) ∨ ((csw_mode ↦ new_mode) ∈ CSW_PSEUDO_TC_TRANSITIONS)
    grd1 : csw_in_sync = FALSE
    grd2 : (mixsc_mode ↦ new_mode) ∈ ALLOWED_MIXSC_CSW_MODES
    grd3 : (mixst_mode ↦ new_mode) ∈ ALLOWED_MIXST_CSW_MODES
    grd6 : (sixsp_mode ↦ new_mode) ∈ ALLOWED_SIXSP_CSW_MODES
    grd7 : (sixsx_mode ↦ new_mode) ∈ ALLOWED_SIXSX_CSW_MODES
THEN
    act3 : csw_mode := new_mode
    act1 : csw_in_sync := TRUE
END

```

EVENT FDIR_Execution_Mode_change**REFINES** *FDIR_Execution_Mode_change*

```

ANY
    new_mode

```

WHERE

$grd4 : new_mode \in CSW_MODES$
 $grd5 : ((csw_mode \mapsto new_mode) \in CSW_FDIR_TRANSITIONS) \vee ((csw_mode \mapsto new_mode) \in CSW_PSEUDO_TC_TRANSITIONS)$

$grd1 : csw_in_sync = TRUE$

THEN

$act3 : csw_mode := new_mode$

$act1 : asw_in_sync := FALSE$

END**EVENT ASW_Execution_Modes_in_sync****WHEN**

$grd1 : asw_in_sync = FALSE$

$grd2 : (mixsc_mode \mapsto csw_mode) \in ALLOWED_MIXSC_CSW_MODES$

$grd3 : (mixst_mode \mapsto csw_mode) \in ALLOWED_MIXST_CSW_MODES$

$grd4 : (sixsp_mode \mapsto csw_mode) \in ALLOWED_SIXSP_CSW_MODES$

$grd5 : (sixsx_mode \mapsto csw_mode) \in ALLOWED_SIXSX_CSW_MODES$

THEN

$act3 : asw_in_sync := TRUE$

END**EVENT TC_Execution_HK_on****REFINES TC_Execution_HK_on****ANY**

$tc_handler$

WHERE

$grd1 : tc_handler \in dom(TCpool)$

$grd2 : TCstatus(tc_handler) = Waiting_for_Execution$

$grd3 : Type_of_TC(TCpool(tc_handler)) = HK_on_TC$

THEN

$act1 : TCstatus(tc_handler) := Successfully_Executed$

$act2 : csw_hk_on := TRUE$

END**EVENT TC_Execution_HK_off****REFINES TC_Execution_HK_off****ANY**

$tc_handler$

WHERE

$grd1 : tc_handler \in dom(TCpool)$

$grd2 : TCstatus(tc_handler) = Waiting_for_Execution$

$grd3 : Type_of_TC(TCpool(tc_handler)) = HK_off_TC$

THEN

$act1 : TCstatus(tc_handler) := Successfully_Executed$

$act2 : csw_hk_on := FALSE$

END**EVENT TC_Execution_SCI_on****REFINES TC_Execution_SCI_on****ANY**

$tc_handler$

WHERE

$grd1 : tc_handler \in dom(TCpool)$

$grd2 : TCstatus(tc_handler) = Waiting_for_Execution$

$grd3 : Type_of_TC(TCpool(tc_handler)) = SCI_on_TC$

THEN

$act1 : TCstatus(tc_handler) := Successfully_Executed$

act2 : csw_sci_on := TRUE
END

EVENT TC_Execution_SCI_off

REFINES TC_Execution_SCI_off

ANY

tc_handler

WHERE

grd1 : tc_handler ∈ dom(TCpool)

grd2 : TCstatus(tc_handler) = Waiting_for_Execution

grd3 : Type_of_TC(TCpool(tc_handler)) = SCI_off_TC

THEN

act1 : TCstatus(tc_handler) := Successfully_Executed

act2 : csw_sci_on := FALSE

END

EVENT Reset_Output_Buffer

REFINES Reset_Output_Buffer

ANY

tc_handler

WHERE

grd1 : tc_handler ∈ dom(TCpool)

grd2 : TCstatus(tc_handler) = Waiting_for_Execution

grd3 : Type_of_TC(TCpool(tc_handler)) = TM_Buffer_Reset_TC

THEN

act1 : TMbuffer := ∅

act2 : TCstatus(tc_handler) := Successfully_Executed

END

EVENT TC_Execution_Failure

REFINES TC_Execution_Failure

ANY

tc_handler

WHERE

grd1 : tc_handler ∈ dom(TCpool)

grd2 : TCstatus(tc_handler) = Waiting_for_Execution

THEN

act1 : TCstatus(tc_handler) := Execution_Failed

END

EVENT Report_TC_execution_success

REFINES Report_TC_execution_success

ANY

tc_handler

tm_handler

tm

WHERE

grd1 : tc_handler ∈ dom(TCpool)

grd2 : tm_handler ∈ TCTM_POOL_ADDRESSES

grd3 : tm ∈ TM_SET

grd4 : TCstatus(tc_handler) = Successfully_Executed

grd5 : tm_handler ∉ dom(TCpool)

grd6 : tm_handler ∉ dom(TMpool)

grd7 : Type_of_TM(tm) = Exec_ok_TM

THEN

act1 : TCpool := {tc_handler} ⋈ TCpool

act2 : TCstatus := {tc_handler} ⋈ TCstatus

act3 : $TMpool(tm_handler) := tm$
END

EVENT Report_TC_execution_failure

REFINES Report_TC_execution_failure

ANY

tc_handler
tm_handler
tm

WHERE

grd1 : $tc_handler \in dom(TCpool)$
grd2 : $tm_handler \in TCTM_POOL_ADDRESSES$
grd3 : $tm \in TM_SET$
grd4 : $TCstatus(tc_handler) = Execution_Failed$
grd5 : $tm_handler \notin dom(TCpool)$
grd6 : $tm_handler \notin dom(TMpool)$
grd7 : $Type_of_TM(tm) = Exec_fail_TM$

THEN

act1 : $TCpool := \{tc_handler\} \triangleleft TCpool$
act2 : $TCstatus := \{tc_handler\} \triangleleft TCstatus$
act3 : $TMpool(tm_handler) := tm$

END

EVENT Pass_TM

REFINES Pass_TM

ANY

tm_handler
tm_pointer

WHERE

grd1 : $tm_handler \in dom(TMpool)$
grd2 : $tm_pointer \in TM_BUFFER_ADDRESSES$
grd3 : $tm_pointer \notin dom(TMbuffer)$

THEN

act2 : $TMbuffer(tm_pointer) := TMpool(tm_handler)$
act1 : $TMpool := \{tm_handler\} \triangleleft TMpool$

END

EVENT Drop_HK_TM

REFINES Drop_HK_TM

ANY

tm_handler

WHERE

grd1 : $tm_handler \in dom(TMpool)$
grd2 : $csw_hk_on = FALSE$
grd3 : $Type_of_TM(TMpool(tm_handler)) = HK_data_TM$

THEN

act1 : $TMpool := \{tm_handler\} \triangleleft TMpool$

END

EVENT Drop_SCI_TM

REFINES Drop_SCI_TM

ANY

tm_handler

WHERE

grd1 : $tm_handler \in dom(TMpool)$
grd2 : $csw_sci_on = FALSE$
grd3 : $Type_of_TM(TMpool(tm_handler)) = SCI_data_TM$

THEN
 $act1 : TMpool := \{tm_handler\} \triangleleft TMpool$
END

EVENT env_Deliver_TM

REFINES env_Deliver_TM

ANY
 $tm_pointer$
WHERE
 $grd1 : tm_pointer \in dom(TMbuffer)$
THEN
 $act1 : TMbuffer := \{tm_pointer\} \triangleleft TMbuffer$
END

EVENT MIXSC_TC_Execution_Mode_change

REFINES TC_ASW_Execution_Mode_change

ANY
 $tc_handler$
 new_mixsc_mode
WHERE
 $grd1 : tc_handler \in dom(TCpool)$
 $grd2 : TCstatus(tc_handler) = Waiting_for_Execution$
 $grd3 : Type_of_TC(TCpool(tc_handler)) \in MODE_CHANGING_TC$
 $grd4 : new_mixsc_mode \in MIXSC_MODES$
 $grd5 : (mixsc_mode \mapsto new_mixsc_mode) \in MIXSC_TC_TRANSITIONS$
 $grd7 : (new_mixsc_mode \mapsto csw_mode) \in ALLOWED_MIXSC_CSW_MODES$
 $grd6 : asw_in_sync = TRUE$
THEN
 $act1 : TCstatus(tc_handler) := Successfully_Executed$
 $act3 : mixsc_mode := new_mixsc_mode$
 $act2 : csw_in_sync := FALSE$
END

EVENT MIXSC_FDIR_Execution_Mode_change

ANY
 new_mixsc_mode
WHERE
 $grd4 : new_mixsc_mode \in MIXSC_MODES$
 $grd5 : (mixsc_mode \mapsto new_mixsc_mode) \in MIXSC_FDIR_TRANSITIONS$
 $grd2 : (new_mixsc_mode \mapsto csw_mode) \in ALLOWED_MIXSC_CSW_MODES$
 $grd1 : asw_in_sync = TRUE$
THEN
 $act3 : mixsc_mode := new_mixsc_mode$
 $act1 : csw_in_sync := FALSE$
END

EVENT MIXSC_Execution_Mode_sync

ANY
 new_mixsc_mode
WHERE
 $grd4 : new_mixsc_mode \in MIXSC_MODES$
 $grd5 : ((mixsc_mode \mapsto new_mixsc_mode) \in MIXSC_TC_TRANSITIONS) \vee ((mixsc_mode \mapsto new_mixsc_mode) \in MIXSC_FDIR_TRANSITIONS)$
 $grd1 : asw_in_sync = FALSE$
 $grd2 : (new_mixsc_mode \mapsto csw_mode) \in ALLOWED_MIXSC_CSW_MODES$
THEN
 $act3 : mixsc_mode := new_mixsc_mode$

END

EVENT MIXSC_Store_HK_Data

REFINES Store_HK_Data

ANY

tm_handler

tm

WHERE

grd1 : *tm_handler* \in *TCTM_POOL_ADDRESSES*

grd2 : *tm* \in *TM_SET*

grd3 : *tm_handler* \notin *dom(TCpool)*

grd4 : *tm_handler* \notin *dom(TMpool)*

grd5 : *csw_hk_on* = *TRUE*

grd6 : *Type_of_TM(tm)* = *HK_data_TM*

grd7 : *mixsc_mode* \in *MIXSC_HK_MODES*

THEN

act1 : *TMpool(tm_handler)* := *tm*

END

EVENT MIXSC_Store_SCI_Data

REFINES Store_SCI_Data

ANY

tm_handler

tm

WHERE

grd1 : *tm_handler* \in *TCTM_POOL_ADDRESSES*

grd2 : *tm* \in *TM_SET*

grd3 : *tm_handler* \notin *dom(TCpool)*

grd4 : *tm_handler* \notin *dom(TMpool)*

grd5 : *csw_sci_on* = *TRUE*

grd6 : *Type_of_TM(tm)* = *SCI_data_TM*

grd7 : *mixsc_mode* \in *MIXSC_SCI_MODES*

THEN

act1 : *TMpool(tm_handler)* := *tm*

END

EVENT MIXST_TC_Execution_Mode_change

REFINES TC_ASW_Execution_Mode_change

ANY

tc_handler

new_mixst_mode

WHERE

grd1 : *tc_handler* \in *dom(TCpool)*

grd2 : *TCstatus(tc_handler)* = *Waiting_for_Execution*

grd3 : *Type_of_TC(TCpool(tc_handler))* \in *MODE_CHANGING_TC*

grd4 : *new_mixst_mode* \in *MIXST_MODES*

grd5 : (*mixst_mode* \mapsto *new_mixst_mode*) \in *MIXST_TC_TRANSITIONS*

grd7 : (*new_mixst_mode* \mapsto *csw_mode*) \in *ALLOWED_MIXST_CSW_MODES*

grd6 : *asw_in_sync* = *TRUE*

THEN

act1 : *TCstatus(tc_handler)* := *Successfully_Executed*

act3 : *mixst_mode* := *new_mixst_mode*

act2 : *csw_in_sync* := *FALSE*

END

EVENT MIXST_FDIR_Execution_Mode_change

ANY

new_mixst_mode
WHERE
 $grd4 : new_mixst_mode \in MIXST_MODES$
 $grd5 : (mixst_mode \mapsto new_mixst_mode) \in MIXST_FDIR_TRANSITIONS$
 $grd2 : (new_mixst_mode \mapsto csw_mode) \in ALLOWED_MIXST_CSW_MODES$
 $grd1 : asw_in_sync = TRUE$
THEN
 $act3 : mixst_mode := new_mixst_mode$
 $act1 : csw_in_sync := FALSE$
END

EVENT MIXST_Execution_Mode_sync

ANY
new_mixst_mode
WHERE
 $grd4 : new_mixst_mode \in MIXST_MODES$
 $grd5 : ((mixst_mode \mapsto new_mixst_mode) \in MIXST_TC_TRANSITIONS) \vee ((mixst_mode \mapsto new_mixst_mode) \in MIXST_FDIR_TRANSITIONS)$
 $grd1 : asw_in_sync = FALSE$
 $grd2 : (new_mixst_mode \mapsto csw_mode) \in ALLOWED_MIXST_CSW_MODES$
THEN
 $act3 : mixst_mode := new_mixst_mode$
END

EVENT MIXST_Store_HK_Data**REFINES** Store_HK_Data

ANY
tm_handler
tm
WHERE
 $grd1 : tm_handler \in TCTM_POOL_ADDRESSES$
 $grd2 : tm \in TM_SET$
 $grd3 : tm_handler \notin dom(TCpool)$
 $grd4 : tm_handler \notin dom(TMpool)$
 $grd5 : csw_hk_on = TRUE$
 $grd6 : Type_of_TM(tm) = HK_data_TM$
 $grd7 : mixst_mode \in MIXST_HK_MODES$
THEN
 $act1 : TMpool(tm_handler) := tm$
END

EVENT MIXST_Store_SCI_Data**REFINES** Store_SCI_Data

ANY
tm_handler
tm
WHERE
 $grd1 : tm_handler \in TCTM_POOL_ADDRESSES$
 $grd2 : tm \in TM_SET$
 $grd3 : tm_handler \notin dom(TCpool)$
 $grd4 : tm_handler \notin dom(TMpool)$
 $grd5 : csw_sci_on = TRUE$
 $grd6 : Type_of_TM(tm) = SCI_data_TM$
 $grd7 : mixst_mode \in MIXST_SCI_MODES$
THEN
 $act1 : TMpool(tm_handler) := tm$
END

EVENT SIXSP_TC_Execution_Mode_change**REFINES** TC_ASW_Execution_Mode_change**ANY***tc_handler**new_sixsp_mode***WHERE***grd1* : *tc_handler* $\in \text{dom}(\text{TCpool})$ *grd2* : *TCstatus(tc_handler)* = *Waiting_for_Execution**grd3* : *Type_of_TC*(*TCpool(tc_handler)*) $\in \text{MODE_CHANGING_TC}$ *grd4* : *new_sixsp_mode* $\in \text{SIXSP_MODES}$ *grd5* : (*sixsp_mode* \mapsto *new_sixsp_mode*) $\in \text{SIXSP_TC_TRANSITIONS}$ *grd7* : (*new_sixsp_mode* \mapsto *csw_mode*) $\in \text{ALLOWED_SIXSP_CSW_MODES}$ *grd6* : *asw_in_sync* = *TRUE***THEN***act1* : *TCstatus(tc_handler)* := *Successfully_Executed**act3* : *sixsp_mode* := *new_sixsp_mode**act2* : *csw_in_sync* := *FALSE***END****EVENT SIXSP_FDIR_Execution_Mode_change****ANY***new_sixsp_mode***WHERE***grd4* : *new_sixsp_mode* $\in \text{SIXSP_MODES}$ *grd5* : (*sixsp_mode* \mapsto *new_sixsp_mode*) $\in \text{SIXSP_FDIR_TRANSITIONS}$ *grd2* : (*new_sixsp_mode* \mapsto *csw_mode*) $\in \text{ALLOWED_SIXSP_CSW_MODES}$ *grd1* : *asw_in_sync* = *TRUE***THEN***act3* : *sixsp_mode* := *new_sixsp_mode**act1* : *csw_in_sync* := *FALSE***END****EVENT SIXSP_Execution_Mode_sync****ANY***new_sixsp_mode***WHERE***grd4* : *new_sixsp_mode* $\in \text{SIXSP_MODES}$ *grd5* : ((*sixsp_mode* \mapsto *new_sixsp_mode*) $\in \text{SIXSP_TC_TRANSITIONS}$) \vee ((*sixsp_mode* \mapsto *new_sixsp_mode*) $\in \text{SIXSP_FDIR_TRANSITIONS}$)*grd1* : *asw_in_sync* = *FALSE**grd2* : (*new_sixsp_mode* \mapsto *csw_mode*) $\in \text{ALLOWED_SIXSP_CSW_MODES}$ **THEN***act3* : *sixsp_mode* := *new_sixsp_mode***END****EVENT SIXSP_Store_HK_Data****REFINES** Store_HK_Data**ANY***tm_handler**tm***WHERE***grd1* : *tm_handler* $\in \text{TCTM_POOL_ADDRESSES}$ *grd2* : *tm* $\in \text{TM_SET}$ *grd3* : *tm_handler* $\notin \text{dom}(\text{TCpool})$ *grd4* : *tm_handler* $\notin \text{dom}(\text{TMpool})$ *grd5* : *csw_hk_on* = *TRUE*

```

    grd6 : Type_of_TM(tm) = HK_data_TM
    grd7 : sixsp_mode ∈ SIXSP_HK_MODES
  THEN
    act1 : TMpool(tm_handler) := tm
  END

```

EVENT SIXSP_Store_SCI_Data**REFINES** Store_SCI_Data

```

  ANY
    tm_handler
    tm
  WHERE
    grd1 : tm_handler ∈ TCTM_POOL_ADDRESSES
    grd2 : tm ∈ TM_SET
    grd3 : tm_handler ∉ dom(TCpool)
    grd4 : tm_handler ∉ dom(TMpool)
    grd5 : csw_sci_on = TRUE
    grd6 : Type_of_TM(tm) = SCI_data_TM
    grd7 : sixsp_mode ∈ SIXSP_SCI_MODES
  THEN
    act1 : TMpool(tm_handler) := tm
  END

```

EVENT SIXSX_TC_Execution_Mode_change**REFINES** TC_ASW_Execution_Mode_change

```

  ANY
    tc_handler
    new_sixsx_mode
  WHERE
    grd1 : tc_handler ∈ dom(TCpool)
    grd2 : TCstatus(tc_handler) = Waiting_for_Execution
    grd3 : Type_of_TC(TCpool(tc_handler)) ∈ MODE_CHANGING_TC
    grd4 : new_sixsx_mode ∈ SIXSX_MODES
    grd5 : (sixsx_mode ↦ new_sixsx_mode) ∈ SIXSX_TC_TRANSITIONS
    grd7 : (new_sixsx_mode ↦ csw_mode) ∈ ALLOWED_SIXSX_CSW_MODES
    grd6 : asw_in_sync = TRUE
  THEN
    act1 : TCstatus(tc_handler) := Successfully_Executed
    act3 : sixsx_mode := new_sixsx_mode
    act2 : csw_in_sync := FALSE
  END

```

EVENT SIXSX_FDIR_Execution_Mode_change

```

  ANY
    new_sixsx_mode
  WHERE
    grd4 : new_sixsx_mode ∈ SIXSX_MODES
    grd5 : (sixsx_mode ↦ new_sixsx_mode) ∈ SIXSX_FDIR_TRANSITIONS
    grd2 : (new_sixsx_mode ↦ csw_mode) ∈ ALLOWED_SIXSX_CSW_MODES
    grd1 : asw_in_sync = TRUE
  THEN
    act3 : sixsx_mode := new_sixsx_mode
    act1 : csw_in_sync := FALSE
  END

```

EVENT SIXSX_Execution_Mode_sync

ANY

new_sixsx_mode
WHERE
grd4 : *new_sixsx_mode* ∈ *SIXSX_MODES*
grd5 : ((*sixsx_mode* ↦ *new_sixsx_mode*) ∈ *SIXSX_TC_TRANSITIONS*) ∨ ((*sixsx_mode* ↦ *new_sixsx_mode*) ∈ *SIXSX_FDIR_TRANSITIONS*)
grd1 : *asw_in_sync* = *FALSE*
grd2 : (*new_sixsx_mode* ↦ *csw_mode*) ∈ *ALLOWED_SIXSX_CSW_MODES*
THEN
act3 : *sixsx_mode* := *new_sixsx_mode*
END

EVENT SIXSX_Store_HK_Data

REFINES Store_HK_Data

ANY
tm_handler
tm
WHERE
grd1 : *tm_handler* ∈ *TCTM_POOL_ADDRESSES*
grd2 : *tm* ∈ *TM_SET*
grd3 : *tm_handler* ∉ *dom(TCpool)*
grd4 : *tm_handler* ∉ *dom(TMpool)*
grd5 : *csw_hk_on* = *TRUE*
grd6 : *Type_of_TM(tm)* = *HK_data_TM*
grd7 : *sixsx_mode* ∈ *SIXSX_HK_MODES*
THEN
act1 : *TMpool(tm_handler)* := *tm*
END

EVENT SIXSX_Store_SCI_Data

REFINES Store_SCI_Data

ANY
tm_handler
tm
WHERE
grd1 : *tm_handler* ∈ *TCTM_POOL_ADDRESSES*
grd2 : *tm* ∈ *TM_SET*
grd3 : *tm_handler* ∉ *dom(TCpool)*
grd4 : *tm_handler* ∉ *dom(TMpool)*
grd5 : *csw_sci_on* = *TRUE*
grd6 : *Type_of_TM(tm)* = *SCI_data_TM*
grd7 : *sixsx_mode* ∈ *SIXSX_SCI_MODES*
THEN
act1 : *TMpool(tm_handler)* := *tm*
END

END