**TemplateHSM State Pseudo Code**

**Key:**

Capital Case used for variables and function names (Ed’s Convention)

/\* comments \*/

// place holder for code

Only code needed for lowest level state (During func)

Write Durings for everything, but comment out the function call if not needed

**Start ThisHSMS (CurrentEvent)**

 // Handle history or irregular entry here

If not history entry

 CurrentState = initial state

 End

 /\* Call ENTRY function for ThisHSMS \*/

 Run ThisHSMS(CurrentEvent)

End Start

**ThisHSMS\_t Run ThisHSMS (CurrentEvent)**

 Init MakeTransition, NextState, EntryEventKind, ReturnEvent

 /\* React differently based on the state you’re in \*/

Switch (CurrentState)

 Case SubState A:

 /\* Execute During to respond to event \*/

 CurrentEvent = DuringStateA(CurrentEvent)

 /\* Handle all events relevant to SubState A \*/

 If ES\_NO\_EVENT does not == CurrentEvent

 Switch (CurrentEvent)

 Case EventOfInterest

 // Any transition action here

 NextState = // State to change to

 MakeTransition = True // Internal transition -> False

 EntryEventKind = // History or Normal

 ReturnEvent = // Consume event -> ES\_NO\_EVENT

 Break

 End switch

 End if

 Break

 Case SubState B:

 /\* Execute During to respond to event \*/

 CurrentEvent = DuringSubStateB (CurrentEvent);

 // Final state – no transitions

 Break;

 End switch

 /\* Handle transitions between states \*/

 If (MakeTransition)

 /\* Exit this state \*/

 CurrentEvent.EventType = EV\_EXIT;

 Run ThisHSMS(CurrentEvent)

/\* Enter next state \*/

CurrentState = NextState; // Change to next state

CurrentEvent.EventType = EventEntryKind;

Run ThisHSMS(CurrentEvent)

 End if

Return ReturnEvent

End Run

**ThisHSMS\_t During ThisHSMS (CurrentEvent)**

 Init ReturnEvent

 If (EV\_ENTRY == CurrentEvent)

 // Execute Entry actions here

 /\* Start lower level SM(s) \*/

 // StartLowerLevelSM(CurrentEvent)

 Else if (EV\_ExIt == CurrentEvent)

 /\* Exit lower level SM(s) \*/

 // RunLowerLevelSM(CurrentEvent)

 // Execute Exit actions here

 Else

 /\* Pass event to lower level SM(s) \*/

 // ReturnEvent = RunLowerLevelSM(CurrentEvent)

 // Execute any internal event response here

 End if

 return ReturnEvent // or CurrentEvent to avoid event consumption

End During

**ThisHSMS\_t Query ThisHSMS (CurrentEvent)**

 Return CurrentState;

End Query

**During LowestLevelState (CurrentEvent)**

 Init ReturnEvent

 If (EV\_ENTRY == CurrentEvent)

 // Execute Entry actions here

 /\* No lower level SM(s) \*/

 Else if (EV\_ExIt == CurrentEvent)

 /\* No lower level SM(s) \*/

 // Execute Exit actions here

 Else

 /\* No lower level SM(s) \*/

 // Execute any internal event response here

 End if

 return ReturnEvent

End During