
Pseudo-code for Loading SM (Lowest Level)

States: ASKING_FOR_COW, WAITING_FOR_COW

Events Posted: ES_ASKED, ES_RESPONSE_READY

RunLoadingSM

Takes ES_Event CurrentEvent, returns ES_NO_EVENT

Set MakeTransition variable to false, because we are not making a transition currently

Set state type variable NextState to CurrentState

Set event type EntryEventKind to ES_ENTRY (default to normal entry to new state)

Set event type ReturnEvent to ES_NO_EVENT, assuming no error

Switch (CurrentState)

Case ASKING_FOR_COW

Execute During function for ASKING_FOR_COW

If the event is active (not ES_NO_EVENT)

Switch (Event)

Case: ES_ASKED

Set NextState to WAITING_FOR_COW

Set MakeTransition to true

Set ReturnEvent to ES_NO_EVENT

End Case

End Switch

End if

End Case

Case WAITING_FOR_COW

Call the WAITING_FOR_COW during function

Set CurrentEvent to returned event from during function

If the CurrentEvent is active (not ES_NO_EVENT)

Switch (CurrentEvent)

Case ES_RESPONSE_READY

Set next state to ASKING_FOR_COW

Set MakeTransition to false

Set ReturnEvent to ES_NO_EVENT (consumed)

End Case

End Switch

End if

End Case

```
If MakeTransition is true (we are transitioning to a different
state)
    Set the CurrentEvent to ES_EXIT
    Call RunLoadingSM with CurrentEvent
    Set CurrentState to NextState
    Call RunLoadingSM with ES_ENTRY event (start the entry
function for the new state)
Endif
Return ReturnEvent
End RunLoadingSM
```

StartLoadingSM

Takes ES_EVENT Current Event, returns nothing

```
Initialize CurrentState to ASKING_FOR_COW
Call RunLoadingSM with Current Event (ES_ENTRY event)
```

DuringAskingForCOW

Takes event, returns event

```
If event is ES_ENTRY or ES_ENTRY_HISTORY
    Turn on the visible LED to signal asking for COW
    Turn on the first pulse of the IR LED to ask for COW
    Increment the number of pulses
    Start the timer for the length of the on time (one shot)
Else if event is ES_EXIT
    Reset the counter for number of pulses
    Turn the visible LED off to signal end of asking
Else
    Nothing
Endif
Return Event (this event is either an event that CheckinSM needs to
handle, or ES_NO_EVENT if a lower level SM handled it)
```

DuringWaitingForCOW

Takes event, returns event

```
If event is ES_ENTRY or ES_ENTRY_HISTORY
    Start a framework timer for the length of waiting for a cow to
load
Else if event is ES_EXIT
Else
    If the COW loading timer has expired
```

```

    Add a COW to the COW variable in MasterSM
    Get the current number of COWs
    If the number of COWs is less than the max number of COWs,
        Then reload by posting an ES_LOAD event to MasterSM
    Else if not
        Post an ES_QUERY event to Comm
        Make parameter GameStatusCMD
    Endif
Else if the event is ES_RESPONSE_READY
    Get the team value
    If team is RED
        Get the next red location from Comm
    Else if team is GREEN
        Get the next green location from Comm
    Endif
    If the next location is shooting
        Post an ES_DRIVE_SHOOT event to MasterSM
    Else if it is checkin
        Post an ES_DRIVE_CHECKIN event to MasterSM
    endif
Endif
Return Event (this event is either an event that LoadingSM needs to
handle, or ES_NO_EVENT if a lower level SM handled it)

```

StartAskTimer

```

Takes an amount of time, returns nothing
    Enable the clock for the one shot interrupt for the amount of
    time passed in
End StartInputCapture

```

AskTimerISR

```

    Clear the source of the interrupt
    If the number of pulses is event
        Set the IR LED high
        Set the amount of time for timer as on-time time
    Else
        Set the IR LED low
        Set the amount of time for timer as lo-time time
    Endif
    If the number of pulses is equal to required number of pulses
        Post an ES_ASKED event to MasterSM to move to
        WAITING_FOR_COW state
    Else

```

Increment number of pulses

Start the AskTimer for amount of time above