# Stimulus Assembly Design

There are four assemblies associated with motion control in the stimulus prototype design.

# Assembly State Machines

The assemblies are very simple state machines to the extent that each command works in isolation: commands are only accepted in the ready state. Note: we need to add states for when commands fail or when the health of the HCD fails.

## Stimulus Optic Assembly



|  |  |  |  |
| --- | --- | --- | --- |
| Command | State | Allowed Commands | Completion State |
| N/A | ready | All | N/A |
| homeMechanism | homing | None | ready |
| selectOptic | selecting | None | ready |
| moveStageToPosition | moveToPos | None | ready |
| setSelectionPoints | setting | None | ready |

This state machine is the same for all motion control assemblies.

All ‘events’ from assemblies are in fact telemetries in the stimulus prototype.

## Assembly Object/Actor Design

Package: org.tmt.aps.ics.stimulusOptic

StimulusOpticAssembly – creates CommandHandler, DiagPublisher, EventPublisher.

AssemblyContext – data class used to define component and the set of submit commands that can be used to command it.

ConfigValidation – validation methods for each of the configuration commands.

CommandHandler – uses event service and reference to HCD actor. Handles all commands from the CCS to the assembly – how this is proxied from the assembly to this actor in the vslice has not been discovered.

For each command received, an actor of the specified type is created and the command and HCD references are passed to it, and the message CommandStart is sent to them. In the vertical slice, these actors are:

DatamCommand

MoveCommand

PositionCommand

SetElevationCommand

FollowCommand

For each of the actors the plan is essentially the same: once the commandStart message is received, the setupConfig is submitted to the HCD using actor message passing.

DiagPublisher – looks like the Telemetry publisher. Publishes telemetry in operations and diagnostic modes.

EventPublisher

Algorithms – static object with various conversions and verification computations.

AlarmMonitor – not sure when this is created, need to look it up. Need to think of what alarms will be required for the Stimulus.

EventSubscriber – there should be an actor whose sole job is to listen to telemetry from the HCD, perform appropriate conversions and pass these upline. Is this a pattern that other components would use?