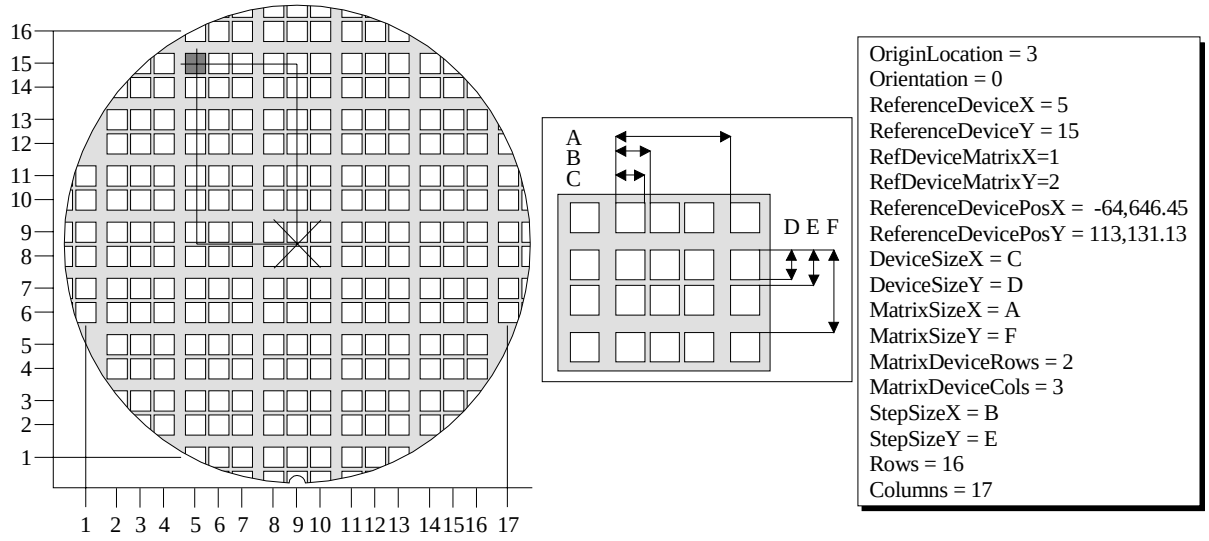


Figure 7
Example Coordinates Orientation = 0

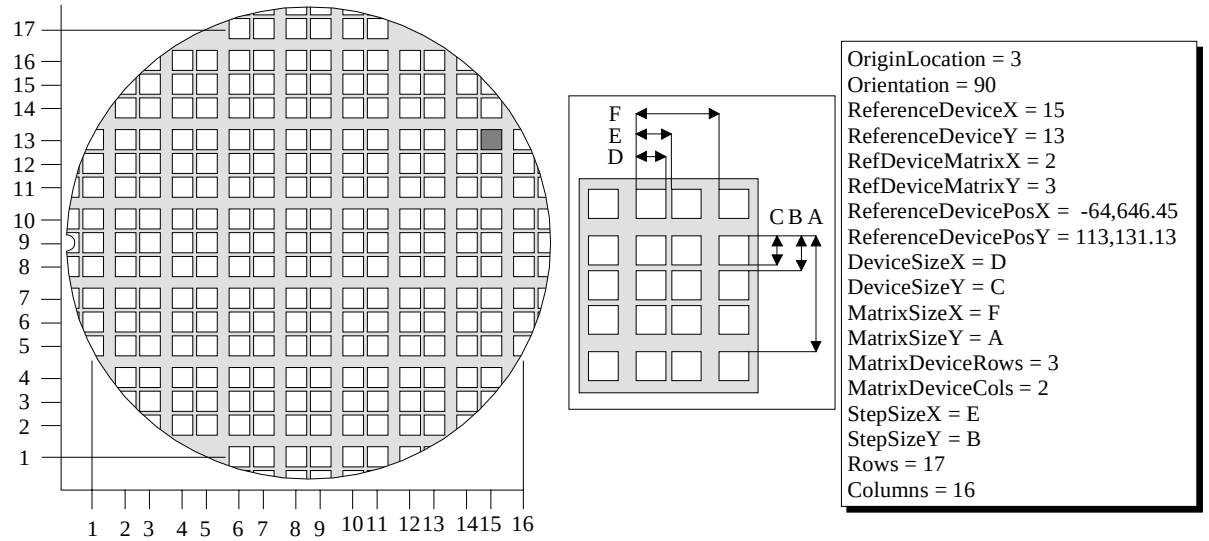


Figure 8
Example Coordinates Orientation = 90

18.9.2 *Orientation on Chuck* — The following example demonstrates the effect of probing a substrate with different settings of the OrientationOnChuck recipe attribute. The OrientationOnChuck attribute has no impact on the coordinate values communicated between the host and the equipment.

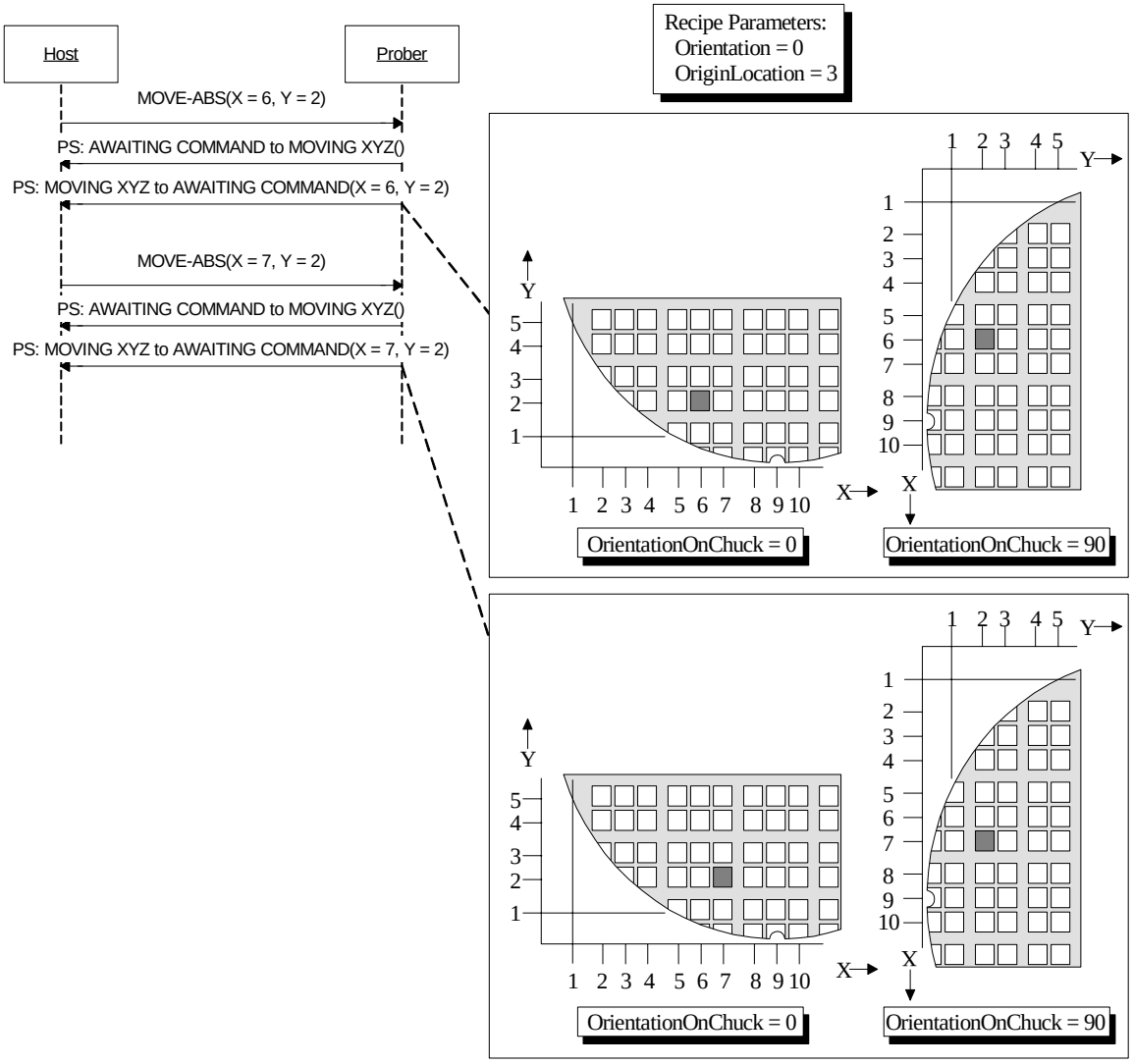


Figure 9
Example OrientationOnChuck

19 Collection Event List

19.1 *Requirements*

19.1.1 All GEM-required events are required by the PSEM300. Since Processing State Model is required by the PSEM300, all state transitions are required collection events.

19.1.2 All GEM-required events associated with the Communications, Alarm, and Spooling State Models are required. This section of the PSEM300 lists only those collection events that are not associated with a change of state or those requiring specific data variables or reports as defined in the PSEM300.

19.2 *Collection Event Table*

19.2.1 Table 3 shows events and reports required for the PSEM300 process state transition. Table 4 shows other collection events.

Table 3 Collection Events Required by Process State Transition

Transition	*Transition No.*	*Current State*	*New State*	*Typical Variable Data*
Into IDLE	1	No state	IDLE	
	6	HANDLING MATERIAL		
	18	IDLE WITH ALARMS		
Into IDLE with ALARMS	17	IDLE	IDLE with ALARMS	Alarm Id
Into SETTING UP	2	IDLE	SETTING UP	
	7	HANDLING MATERIAL		
Into AWAITING LOAD	3	SETTING UP	AWAITING LOAD	
	5	HANDLING MATERIAL		
Into HANDLING MATERIAL	4	AWAITING LOAD	HANDLING MATERIAL	
	9	AWAITING COMMAND		
Into AWAITING COMMAND	8	HANDLING MATERIAL	AWAITING COMMAND	X, Y,Z
	11	MOVING XYZ		
	13	SERVICE TASK		X,Y,Z, Task Result
Into MOVING XYZ	10	AWAITING COMMAND	MOVING XYZ	
Into SERVICE TASK	12	AWAITING COMMAND	SERVICE TASK	Task Id
	14	SERVICE TASK		
Into PROCESSING BLOCKED	15	PROCESS	PROCESS BLOCKED	
Conditional	16	PROCESSING BLOCKED	PROCESS	

Table 4 Additional Collection Events

Event Name	*Contents*	*Typical Variable Data*
CleaningMediaDepleted	The cleaning media has been depleted.	Cleaning media identifier
InkingMediaDepleted	The inking media has been depleted.	
ProbeCardRemoved	The probe card has been removed from the prober.	ProbeCardId
ProbeCardInstalled	The probe card has been installed on the prober.	ProbeCardId
ControlMapCompleted	The prober has completed indexing to each of the sites identified in the control map.	

20 Data Item Variables

20.1 The purpose of this section is to define the list of data item variables required by the PSEM300. Values of these variables will be available to the host through collection event reports and host status queries.

20.2 *Requirements*

20.2.1 All generic variable data items defined in GEM are required by all PSEM300 equipment. Any supplier-defined variables shall be documented in the same format used by this document. The following minimum information is required:

20.3 *Data Types*

20.3.1 Equipment constants remain in effect until they are overwritten either by manual entry or by a “New Equipment Constant Send” remote command. Equipment constants have various uses in PSEM300, including the following:

1) Equipment offsets that match the performance of several pieces of equipment that would otherwise perform differently due to inherent manufacturing differences. Examples are home values and motion axis scaling factors.

2) Setting the configuration of the equipment to allow for different material specifications, equipment options, material flows, frequency of automatic functions, etc.

3) Managing optional machine features. Examples are constants that tell the system whether optional substrate ID readers are present and control the configuration and function of these optional subsystems when they are present.

20.3.2 Status Variables are valid at all times. A status variable may not be changed by the host but may be changed by the equipment or operator. The value of the status variables may be queried by the host at anytime using the "Selected Equipment Status Request" remote command (See SEMI E30). Data variables are variables that are valid only upon the occurrence of specific collection events. An attempt to read a data variable at the wrong time will not generate an error, but the data reported may not have relevant meaning.

20.3.3 *Data Types and Sizes*

Table 5 Data Types and Sizes

Type	*Size*	*Definition*
String	0–32767	A string of ASCII characters from zero to "Size" characters in length.
Integer	1,2,4	An integer value that may be represented by "Size" bytes. E.g. If size=2 then the value may be –32768 to 32767.
Float	4	IEEE single-precision 32-bit floating point type [IEEE 754-1985]
Boolean	1	True or False

Table 6 Composite Data Types

Data Type	*Definition*	*Comments*
DieCoordinate	Structure composed of: XCoordinate YCoordinate	Structured data used to specify die locations on a substrate
XCoordinate	Integer	Coordinate in the X direction (column)
YCoordinate	Integer	Coordinate in the Y direction (row)

20.4 *Data Item Variable Table*

Table 7 Data Item Variable Table

Variable Name	*Data Type*	*Size*	*Description*	*Comments*
ContactCount	Integer	2	The number of times the probe card has come in contact with a substrate.	This value is reset when the host issues the SET-PROBE-CARD-ID or RESET-CONTACT-CNT remote commands. This data item must be available for the host to query at all times.
DieXCoordinate	Integer	2	The coordinate of the die in the x direction (column).	Any event associated with the movement of the chuck requires this variable.
DieYCoordinate	Integer	2	The coordinate of the die in the y direction (row).	Any event associated with the movement of the chuck requires this variable.
DieZPosition	Float	4	The position of the Chuck Z-axis in microns. A user/supplier defined unit of measure other than microns is permissible and must be documented.	Any event associated with the movement of the chuck requires this variable.
DieZCoordinate	Integer	2	The coordinate of the die in the z direction.	0:DOWN, 1:UP
InspectionResult	Boolean	1	Result of the last inspection.	True:PASS, False:FAIL

Variable Name	*Data Type*	*Size*	*Description*	*Comments*
MinXStep	Float	4	The minimum step size in microns in the X direction.	
MinYStep	Float	4	The minimum step size in microns in the Y direction.	
MinZStep	Float	4	The minimum step size in microns in the Z direction (Alternative user-defined unit of measure may be specified for the Z motion).	
ProbeCardId	String	25	The ID of the Probe Card	
TaskID	String	50	The ID of the service task being performed.	Must be available when a Service Task is being performed.

21 Recipe Management

21.1 PSEM300 does not attempt to specify how recipes are stored on the equipment or the internal structure of a recipe. Recipe parameters must be available for query by the host. The equipment must support process tuning as defined in SEMI E40.

21.2 Recipe Setup Parameters contains the minimum set of recipe attributes required by PSEM300. It is expected that additional parameters will be used to reflect the capabilities of the specific equipment configuration and technologies available for use.

21.3 *Recipes and Process Programs*

21.3.1 Recipes and Process Programs are considered to be the same in this document. The recipe of the prober consists of the main recipe with optional service recipes referenced within the main recipe.

21.4 *Recipe Requirements*

21.4.1 The PSEM300 requires that the GEM capability of recipe management be fully supported for this class of equipment. The Process Job specifies the recipe to be executed.

Table 8 Recipe Setup Parameters

Tag	*Data Type*	*Size*	*Description*	*Unit*
ProdId	String	50	Product name (i.e., Device type)	-
WaferSize	Integer	2	Wafer size	millimeters
DeviceSizeX	Float	4	Die size X	microns
DeviceSizeY	Float	4	Die size Y	microns
MatrixSizeX	Float	4	Matrix size X. For non-matrix substrates, this attribute is equal to StepSizeX.	microns
MatrixSizeY	Float	4	Matrix size Y. For non-matrix substrates, this attribute is equal to StepSizeY.	microns
MatrixDeviceRows	Integer	2	The number of device rows within an expsosure field. For non-matrix substrates, this attribute = 1. This attribute is dependant on the Orientation attribute.	-
MatrixDeviceCols	Integer	2	The number of device columns within an exposure field. For non-matrix substrates, this attribute = 1. This attribute is dependant on the Orientation attribute.	-
StepSizeX	Float	4	Step Size X. This attribute is dependant on the Orientation attribute.	microns
StepSizeY	Float	4	Step Size Y. This attribute is dependant on the Orientation attribute.	microns
ReferenceDeviceX	Integer	2	X coordinate of the reference device. This attribute is dependant on the Orientation attribute.	Column

Tag	*Data Type*	*Size*	*Description*	*Unit*
ReferenceDeviceY	Integer	2	Y coordinate of the reference device. This attribute is dependant on the Orientation attribute.	Row
RefDevicePosX	Float	4	Offset of the center of the reference device in the X-direction from the center of the substrate.	microMeters
RefDevicePosY	Float	4	Offset of the center of the reference device in the Y-direction from the center of the substrate.	microMeters
RefDeviceMatrixX	Integer	2	X coordinate of the reference device within an exposure field. For non-matrix substrates, this attribute = 1. This attribute is dependant on the Orientation attribute.	-
RefDeviceMatrixY	Integer	2	Y coordinate of the reference device within an exposure field. For non-matrix substrates, this attribute = 1. This attribute is dependant on the Orientation attribute.	-
Overdrive	Float	4	Over drive value	microns
UseControlMap	Boolean	1	Probe locations defined in control map.	-
Orientation	Integer	2	Fiducial (flat or notch) orientation. Defines the Orientation used to describe the substrate geometries. The substrate may be loaded onto the equipment chuck with an orientation established by the OrientationOnChuck attribute. OrientationOnChuck does not impact the substrate coordinate references.	Degree
OriginLocation	Integer	2	Reference Quadrant: 0 = Center, 1 = Upper Right, 2 = Upper Left 3 = Lower Left (default), 4 = Lower Right	-
OrientationOnChuck	Integer	2	Fiducial (flat or notch) orientation of the substrate loaded onto the chuck. Coordinate references are not impacted by this attribute.	-
FiducialType	Integer	2	0 = NOTCH, 1 = FLAT	-
UseHotChuck	Boolean	1	Flag set to indicate the use of temperature control.	-
HotChuckTemperature	Integer	2	Hot chuck temperature. This attribute applies when the “UseHotChuck” attribute is set to TRUE.	Celsius
HotChuckTolerance	Integer	2	Hot chuck tolerance. This attribute applies when the “UseHotChuck” attribute is set to TRUE.	Celsius
Rows	Integer	2	The number of rows of devices on the substrate. For a row to be included, a row must contain at least one complete device.	-
Columns	Integer	2	The number of columns of devices on a substrate. For a column to be included, a column must contain at least one complete.	-

22 Remote Commands

22.1 The purpose of this section is to identify remote commands, command parameters, and valid commands versus states in the processing state models.

22.2 *Requirements*

22.2.1 All the remote commands defined by PSEM300 are required.

22.2.2 The GEM remote commands START, STOP, PAUSE, ABORT, RESUME, PP-SELECT and RCP-SELECT are not used.

22.2.3 The alphanumeric strings defined by PSEM300 for remote commands and command parameters are required.

22.2.4 If additional remote commands are supported, then the “Remote Command Versus Valid States” matrix must be generated for these additional commands. Place an “X” in the table for each state in which a given command is valid.

22.3 *Remote Commands Descriptions*

22.3.1 *CLEAN-PROBES* — This command causes the probe cleaning routine to be executed. The recipe will contain sufficient settings to identify the cleaning processes. If there is more than one cleaning process defined, each

must be identified with a name. If The Process State model was in the AWAITING COMMAND state when the CLEAN-PROBES command was issued, and the cleaning task requires the substrate to be removed from the chuck, the Process State model will transition to the SERVICE TASK state the substrate will be suspended. When the cleaning task is completed, a suspended substrate must be restored to the chuck and the chuck will be returned to the pre-task coordinates prior to the transition from SERVICE TASK to AWAITING COMMAND.

Parameter	*Req/Ind*	*Rsp/Cnf*	*Form*	*Comment*
RemoteCommand	M	-	Text	"CLEAN-PROBES"
Cleaning Process	C	-	Text	The name of the cleaning process that is to be performed. If there is only one cleaning process defined, this parameter may be omitted.
Die List	M	-	List of type DieCoordinate, see Data Types and Sizes.	List of probe sites to be cleaned. Zero length list indicates that all sites should be cleaned.
Ack	-	M	Integer	Service Acknowledgement.

22.3.2 *INDEX* — This command is used in conjunction with Control Maps. The USE CONTROL MAP recipe attribute must be set to TRUE. The INDEX command causes the prober to move the chuck to the next unprocessed site contained in the control map.

Parameter	*Req/Ind*	*Rsp/Cnf*	*Form*	*Comment*
RemoteCommand	M	-	Text	"INDEX"
Ack	-	M	Integer	Service Acknowledgement.

22.3.3 *INK-DEVICE* — This command causes the current die location to be inked.

22.3.3.1 The Process State model transitions to the SERVICE TASK state. When the task is completed, the Process State model transitions from the SERVICE TASK to AWAITING COMMAND state.

Parameter	*Req/Ind*	*Rsp/Cnf*	*Form*	*Comment*
RemoteCommand	M	-	Text	"INK-DEVICE"
Ack	-	M	Integer	Service Acknowledgement.

22.3.4 *INK-DIE* — This command causes the die locations contained in the die-list to be inked. Optionally, the host can specify a control map containing the sites to be inked.

22.3.4.1 The Process State model transitions to the SERVICE TASK state. When the task is completed, the chuck is returned to the pre-task coordinates and the Process State model transitions from the SERVICE TASK to AWAITING COMMAND state.

Parameter	*Req/Ind*	*Rsp/Cnf*	*Form*	*Comment*	
RemoteCommand	M	-	Text	"INK-DIE"	
Die List	C	-	List of type DieCoordinate, see Data Types and Sizes.	List of die sites to be inked.	Either Die List or Control Map must be supplied.
Control Map	C	-	Text	Control Map Identifier	
Ack	-	M	Integer	Service Acknowledgement.	

22.3.5 *INSPECT-INK-MARKS* — This command causes the ink marks to be inspected. The Process model transitions to the SERVICE TASK state. When the inspection task is completed, the chuck is returned to the pre-task coordinates and the Process State model transitions from the SERVICE TASK to AWAITING COMMAND state.

Parameter	*Req/Ind*	*Rsp/Cnf*	*Form*	*Comment*
RemoteCommand	M	-	Text	"INSPECT-INK-MARKS"
Die List	C		List of type DieCoordinate, see Data Types and Sizes.	If this parameter is present, inspect the site listed. Otherwise, inspect the current site.
Ack	-	M	Integer	Service Acknowledgement.

22.3.6 *INSPECT-PTP-ALIGN* — This command causes the probe to pad alignment to be inspected. The Process State model transitions to the SERVICE TASK state. When the inspection task is completed, the chuck is returned to the pre-task coordinates and the Process State model transitions from the SERVICE TASK to AWAITING COMMAND state.

Parameter	*Req/Ind*	*Rsp/Cnf*	*Form*	*Comment*
RemoteCommand	M	-	Text	"INSPECT-PTP-ALIGN"
Ack	-	M	Integer	Service Acknowledgement.

22.3.7 *INSPECT-PROBES* — This command causes the probes to be inspected. Support for selective probe site inspection is not a fundamental requirement. When this capability exists, the supplier must document the format of the probe site list. If the Process State model is in the AWAITING COMMAND state, the Process State model transitions to the SERVICE TASK state. When the inspection task is completed, the chuck is returned to the pre-task coordinates and the Process State model transitions from the SERVICE TASK to AWAITING COMMAND state.

Parameter	*Req/Ind*	*Rsp/Cnf*	*Form*	*Comment*
RemoteCommand	M	-	Text	"INSPECT-PROBES"
Probe Site List	M	C	Supplier defined	List of probe sites to be inspected. Zero length list indicates that all probes to inspected.
Ack	-	M	Integer	Service Acknowledgement.

22.3.8 *LOAD-SUBSTRATE* — This command causes the next substrate identified in the Process Job to be loaded onto the chuck. The prober must perform all the pre-conditioning requirements for probing to begin. The transition from HANDLING MATERIAL to AWAITING COMMAND indicates that probing can commence. If the probe is unable to perform the necessary steps to prepare the substrate for probing, the prober will send an alarm and transition to the PROCESSING BLOCKED state.

Parameter	*Req/Ind*	*Rsp/Cnf*	*Form*	*Comment*
RemoteCommand	M	-	Text	"LOAD SUBSTRATE"
Ack	-	M	Integer	Service Acknowledgement.

22.3.9 *LOAD-TEST-SUBST* — This command causes a test substrate from the specified substrate location to be loaded onto the chuck.

22.3.9.1 A substrate occupying the chuck is suspended. When the host issues the UNLOAD-TEST-SUBST command, a suspended substrate is automatically resumed.

Parameter	*Req/Ind*	*Rsp/Cnf*	*Form*	*Comment*
RemoteCommand	M	-	Text	"LOAD-TEST-SUBST"
SubstLocID	M		Text	Substrate Location containing the test substrate.
Ack	-	M	Integer	Service Acknowledgement.

22.3.10 *LOCKOUT-CONTROL* — This command is used to enable and disable the equipment console during remote operations. Passing a value of Lockout=TRUE disables the equipment console and requires the user to switch the control state to ONLINE LOCAL to regain console control.

Parameter	*Req/Ind*	*Rsp/Cnf*	*Form*	*Comment*
RemoteCommand	M	-	Text	"LOCKOUT-CONTROL"
Lockout	M	C	Boolean	True = Lockout console control. User must switch to ONLINE-LOCAL mode to regain control.
Ack	-	M	Integer	Service Acknowledgement.

22.3.11 *MOVE* — This commande is used to perform a relative move operation. At least one of the optional parameters: X, Y, or Z must be included with this command. An omitted X, Y, or Z is equivalent to specifying zero.

Parameter	*Req/Ind*	*Rsp/Cnf*	*Form*	*Comment*
RemoteCommand	M	-	Text	"MOVE"
X	C	-	Integer	Relative step value. At least one parameter must be included.
Y	C	-	Integer	
Z	C	-	Integer	
Ack	-	M	Integer	Service Acknowledgement.

22.3.12 *MOVE-ABS* — This command is used to perform an absolute move. At least one of the optional parameters: X, Y, or Z must be included with this command. An omitted X, Y, or Z is equivalent to specifying the current value, resulting in no net movement in that direction.

Parameter	*Req/Ind*	*Rsp/Cnf*	*Form*	*Comment*
RemoteCommand	M	-	Text	"MOVE-ABS"
X	C	-	Integer	Absolute index. At least one parameter must be included.
Y	C	-	Integer	
Z	C	-	Integer	
Ack	-	M	Integer	Service Acknowledgement.

22.3.13 *MOVE-SUBSTEP* — This command is used to perform a sub-index step. At least one of the optional parameters: XSubstep, YSubstep, or ZSubstep must be included with this command. An omitted XSubstep, YSubstep, or ZSubstep is equivalent to specifying zero.

Parameter	*Req/Ind*	*Rsp/Cnf*	*Form*	*Comment*
RemoteCommand	M	-	Text	"MOVE-SUBSTEP"
XSubstep	C	-	Float	Relative step value. At least one parameter must be included.
YSubstep	C	-	Float	
ZSubstep	C	-	Float	
Ack	-	M	Integer	Service Acknowledgement.

22.3.14 *MOVE-TO-REF-DIE* — Move to the Reference Die coordinate as defined within the recipe.

Parameter	*Req/Ind*	*Rsp/Cnf*	*Form*	*Comment*
RemoteCommand	M	-	Text	"MOVE–TO-REF-DIE"
Ack	-	M	Integer	Service Acknowledgement.

22.3.15 *UNLOAD-TEST-SUBST* — This command unloads the test substrate from the chuck to the specified substrate location. A suspended substrate is automatically resumed.

Parameter	*Req/Ind*	*Rsp/Cnf*	*Form*	*Comment*
RemoteCommand	M	-	Text	"UNLOAD-TEST-SUBST"
SubstLocID	M		Text	Target Substrate Location.
Ack	-	M	Integer	Service Acknowledgement.

22.3.16 *RESET-CONTACT-CNT* — This command is used to reset the accumulators associated with the number of contacts made for a given probe card.

Parameter	*Req/Ind*	*Rsp/Cnf*	*Form*	*Comment*
RemoteCommand	M	-	Text	"RESET-CONTACT-CNT"
Ack	-	M	Integer	Service Acknowledgement.

22.3.17 *RESET-CONTROL MAP* — This command is used to reset the coordinate index within a current Control Map. The host can designate the coordinate index within the Control Map by providing the optional XY coordinates with this command.

Parameter	*Req/Ind*	*Rsp/Cnf*	*Form*	*Comment*
RemoteCommand	M	-	Text	"RESET-CONTROL-MAP"
X	C	-	Integer	If coordinates are designated, both X and Y must be provided. The host is responsible for providing XY coordinates that match coordinates contained in the Control Map.
Y	C	-	Integer	
Ack	-	M	Integer	Service Acknowledgement.

22.3.18 *SET-PROBE-CARD-ID* — This command is used to set the probe card ID attribute on the tool.

Parameter	*Req/Ind*	*Rsp/Cnf*	*Form*	*Comment*
RemoteCommand	M	-	Text	"SET-PROBE-CARD-ID"
ProbeCardID	M	-	Text	Prober card identifier.
Ack	-	M	Integer	Service Acknowledgement.

22.3.19 *SET-CONTROL-MAP* — This command is used to identify the control map. See Section 15 for the discussion of control maps.

Parameter	*Req/Ind*	*Rsp/Cnf*	*Form*	*Comment*
RemoteCommand	M	-	Text	"SET-CONTROL-MAP"
ControlMap	M	-	Text	Identifies the name of the control map.
X	C	-	Integer	If coordinates are designated, both X and Y must be provided. The host is responsible for providing XY coordinates that match coordinates contained in the Control Map.
Y	C	-	Integer	
Ack	-	M	Integer	Service Acknowledgement.

22.3.20 *UNLOAD-SUBSTRATE* — This command is used to unload the substrate occupying the chuck. The chuck is unoccupied upon completion of this command.

Parameter	*Req/Ind*	*Rsp/Cnf*	*Form*	*Comment*
RemoteCommand	M	-	Text	"UNLOAD-SUBSTRATE"
Ack	-	M	Integer	Service Acknowledgement.

22.3.21 *UNLOAD-ALLSUBSTRATES* — This command is used to unload substrates occupying the chuck and pipeline stages.

Parameter	*Req/Ind*	*Rsp/Cnf*	*Form*	*Comment*
RemoteCommand	M	-	Text	"UNLOAD-ALLSUBSTRATES"
SUSPEND	C	-	Boolean	If included, causes the substrate occupying the chuck to be suspended.
Ack	-	M	Integer	Service Acknowledgement.

22.3.22 *ONLINE-LOCAL* — This command is used to change the control state from On-Line Remote to On-Line Local.

Parameter	*Req/Ind*	*Rsp/Cnf*	*Form*	*Comment*
RemoteCommand	M	-	Text	"ONLINE-LOCAL"
Ack	-	M	Integer	Service Acknowledgement.

22.4 *Remote Commands and PSEM300 Process Model Mapping*

22.4.1 Each remote command either can or cannot be executed depending on a state other than the control state.

22.4.2 The following table shows remote commands and the relationship between the PSEM300 process states to be operated by these commands.

Table 9 Remote Command vs. Process State

	CLEAN-PROBES	INDEX	INK-DEVICE	INK-DIE	INSPECT-INK-MARKS	INSPECT-PROBES	INSPECT-PTP-ALIGNMENT	LOAD-TEST-SUBST	LOAD-SUBSTRATE	LOCKOUT-CONTROL	MOVE	MOVE-ABS	MOVE-SUB-STEPS	MOVE-TO-REF-DIE	ONLINE-LOCAL	RESET-CONTACT-CNT	RESET-CONTROL-MAP	SET-PROBE-CARD-ID	SET-CONTROL-MAP	UNLOAD-TEST-SUBST	UNLOAD-SUBSTRATE	UNLOAD-ALLSUBSTRATES
PROCESSING STATE																						
IDLE	O					O				O					O	O		O				
IDLE with ALARMS	O									O					O	O		O				
PROCESSING ACTIVE																						
…PROCESS																						
…SETTING UP																						
…EXECUTING																						
……AWAITING LOAD	O					O		O	O	O					O	O		O				
……HANDLING MATERIAL																						
……SUBSTRATE LOADED																						
……AWAITING COMMAND	O	O	O	O	O	O	O	O	O	O	O	O	O	O	O	O	O	O	O	O	O	O
…… MOVING XYZ																						
……SERVICE TASK																						
…PROCESSING BLOCKED																						

O: Operation is allowed.

22.5 *Restriction on the Operator by Control States*

22.5.1 For the remote commands, the operator of the prober or the host is restricted by the control states of the equipment.

Table 10 Table of Restrictions on the Operator by Control States

Remote Commands	*Operator*		*Host*	
	LOCAL	*REMOTE*	*LOCAL*	*REMOTE*
CLEAN-PROBES	O	X	X	O
INDEX	O	X	X	O
INK-DEVICE	O	X	X	O
INK-DIE	O	X	X	O
INSPECT-INK-MARKS	O	X	X	O
INSPECT-PROBES	O	X	X	O
INSPECT- PTP-ALIGN	O	X	X	O
LOAD- TEST-SUBST	O	X	X	O
LOAD-SUBSTRATE	O	X	X	O
MOVE	O	X	X	O
MOVE-ABS	O	X	X	O
MOVE-SUB-STEPS	O	X	X	O
MOVE-TO-REF-DIE	O	X	X	O
UNLOAD-TEST-SUBST	O	X	X	O
RESET-CONTACT-CNT	O	O	O	O
RESET-CONTROL-MAP	O	X	O	O
SET-CONTROL	X	X	X	O
SET-PROBE-CARD-ID	O	O	O	O
SET-CONTROL-MAP	O	X	O	O
UNLOAD-SUBSTRATE	O	X	X	O
UNLOAD-ALLSUBSTRATES	O	X	X	O
ONLINE-LOCAL	X	X	X	O

O: Operation is allowed.

X: Operation is prohibited.

23 GEM Addition Request

23.1 The purpose of this section is to specify all GEM addition requests that are required to support the equipment of this class.

23.2 *Requirements*

23.2.1 GEM addition requests required of PSEM300 are as shown below:

- Establish Communications,
- Dynamic Event Report Configuration,
- Variable Data Collection,
- Status Data Collection,
- Alarm Management,
- Remote Control,
- Equipment Constants,
- Process Program Management,
- Equipment Terminal Services,
- Clock,
- Spooling, and
- Control (host-initiated).

RELATED INFORMATION 1

NOTICE: This related information is not an official part of SEMI E130 and was derived from the work of the originating task force. This related information was approved for publication by full letter ballot procedures on September 3, 2003.

R1-1 Scenarios

R1-1.1 *Scenario I – Complete Multiple Process Job Run*

R1-1.1.1 *Job Description*

R1-1.1.1.1 A single carrier containing six wafers in two lots is processed as two separate Process Jobs. E90 state transitions are omitted for the sake of clarity. See Section R1-1.2 for a description of the substrate tracking events.

CarrierId: "CarrierA"
Process Job 1: Id: PJ01 Slots 1-3 **Substrate Ids**: SubstPJ01.01, SubstPJ01.02, SubstPJ01.03
Process Job 2: Id: PJ02 Slots 23-25 **Substrate Ids**: SubstPJ02.01, SubstPJ02.02, SubstPJ02.03

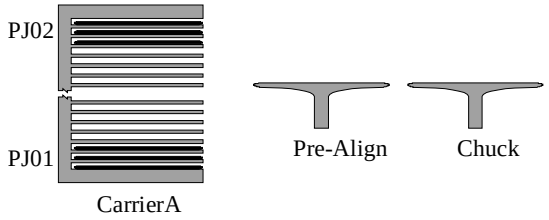


Figure R1-1
Processing Environment for Scenarios

R1-1.1.2 *Sequence Diagram*

R1-1.1.2.1 *Job Setup*

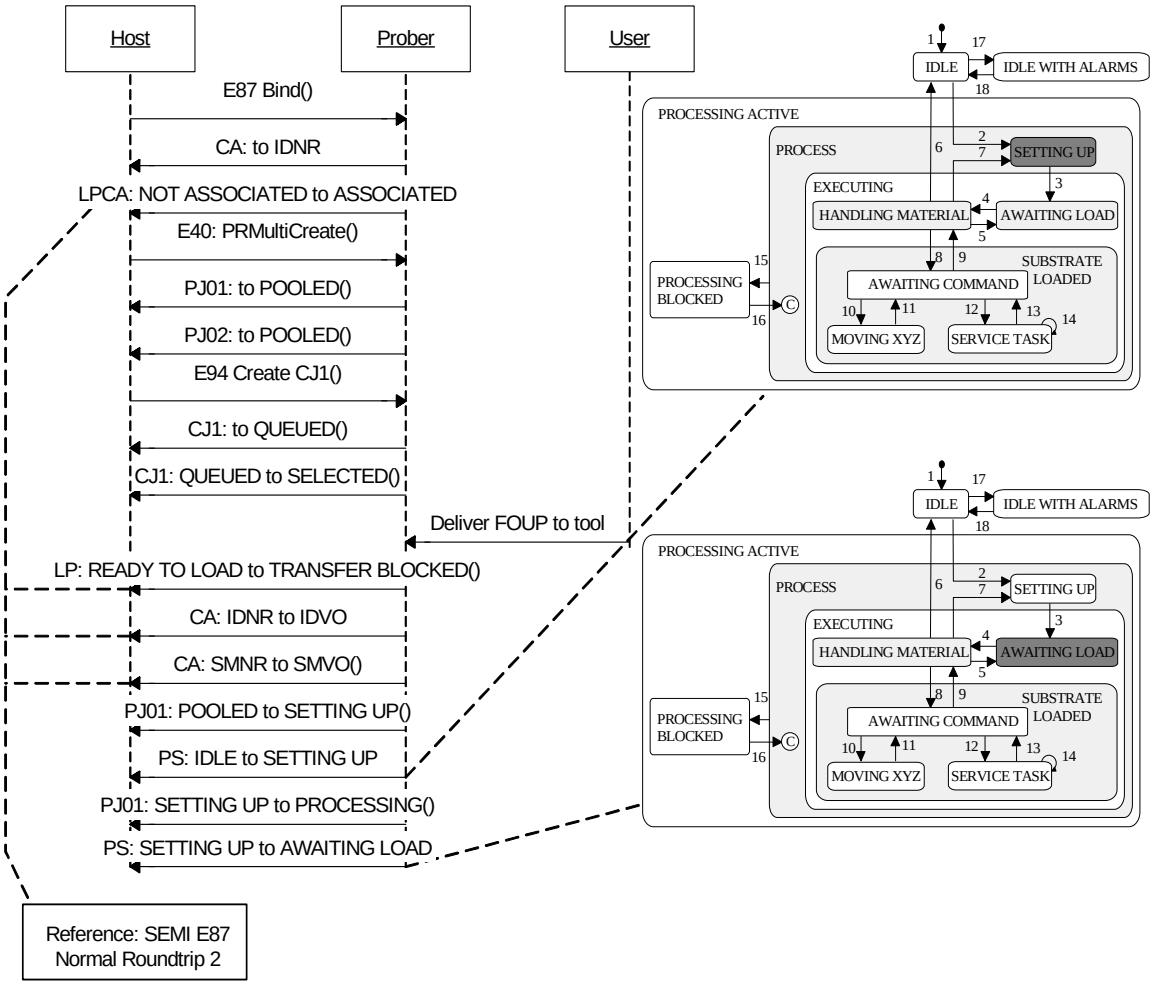


Figure R1-2
Normal Run - Job Setup Scenario

R1-1.1.2.2 *Job Start* — At this point, the Process Job is in the PROCESSING state and the prober is waiting for the host to begin processing the material identified in the PRMtlNameList.

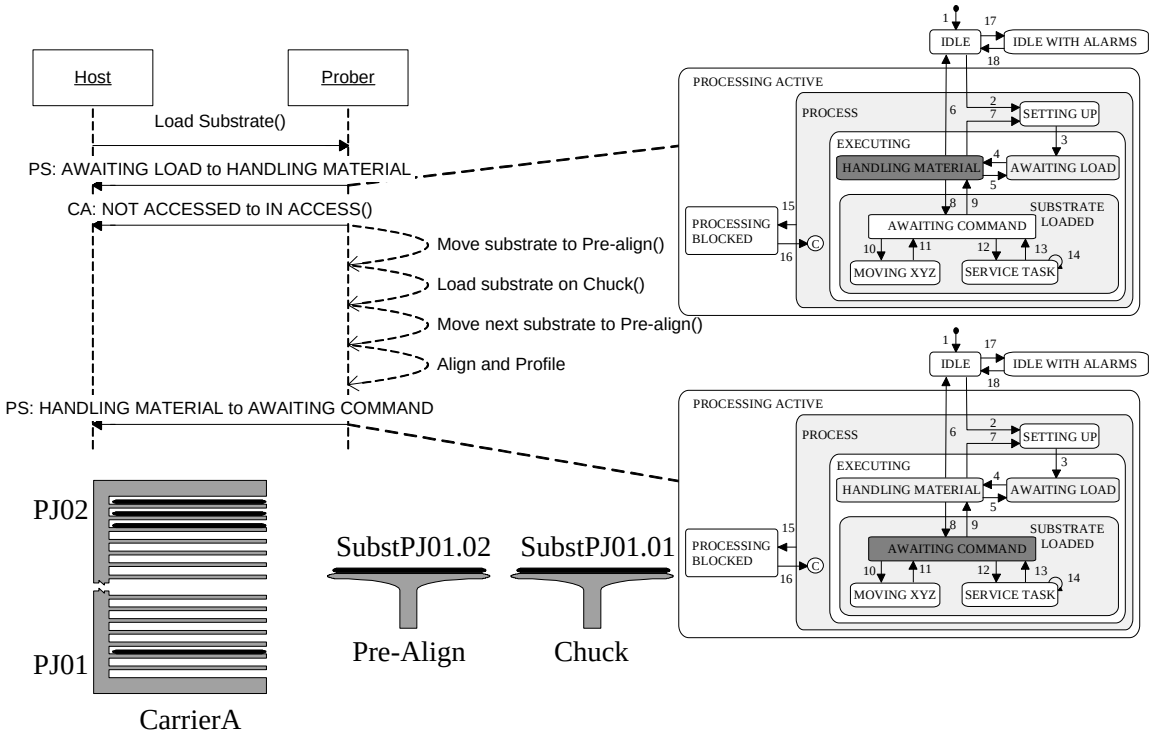


Figure R1-3
Normal Run - Job Start Scenario

R1-1.1.2.3 *Process Material*

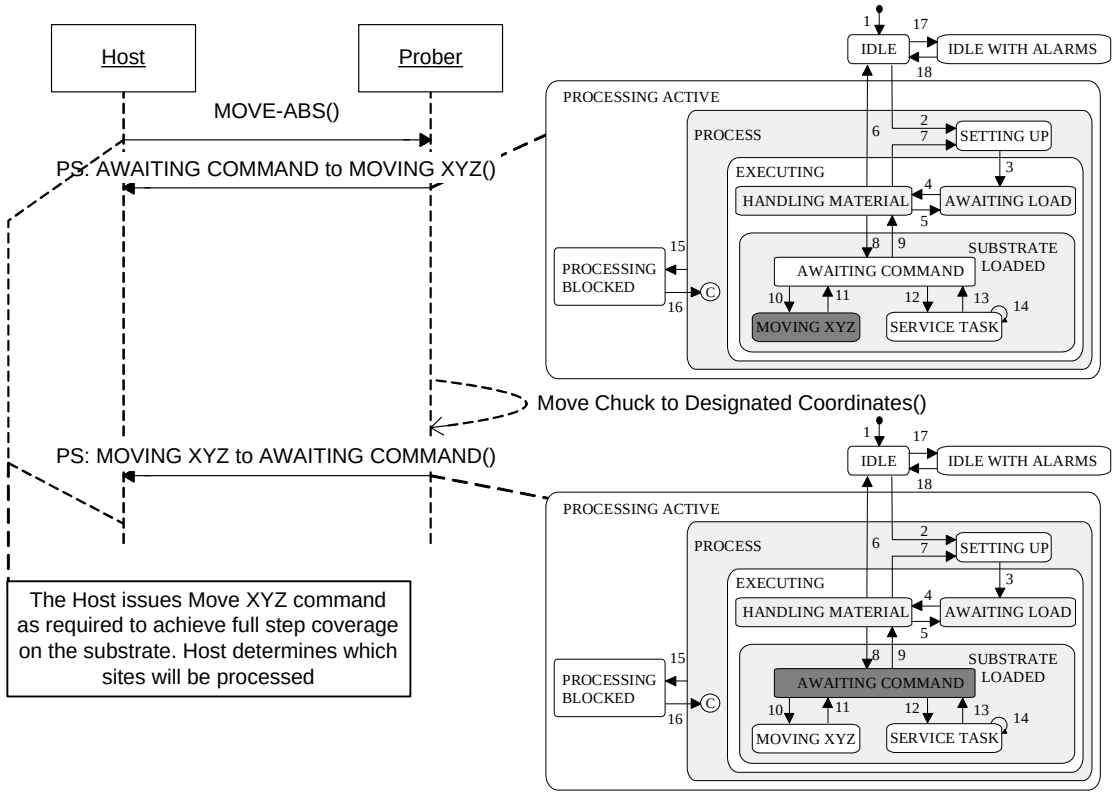


Figure R1-4
Normal Run - Material Processing Scenario

R1-1.1.2.4 *Load 2nd Substrate*

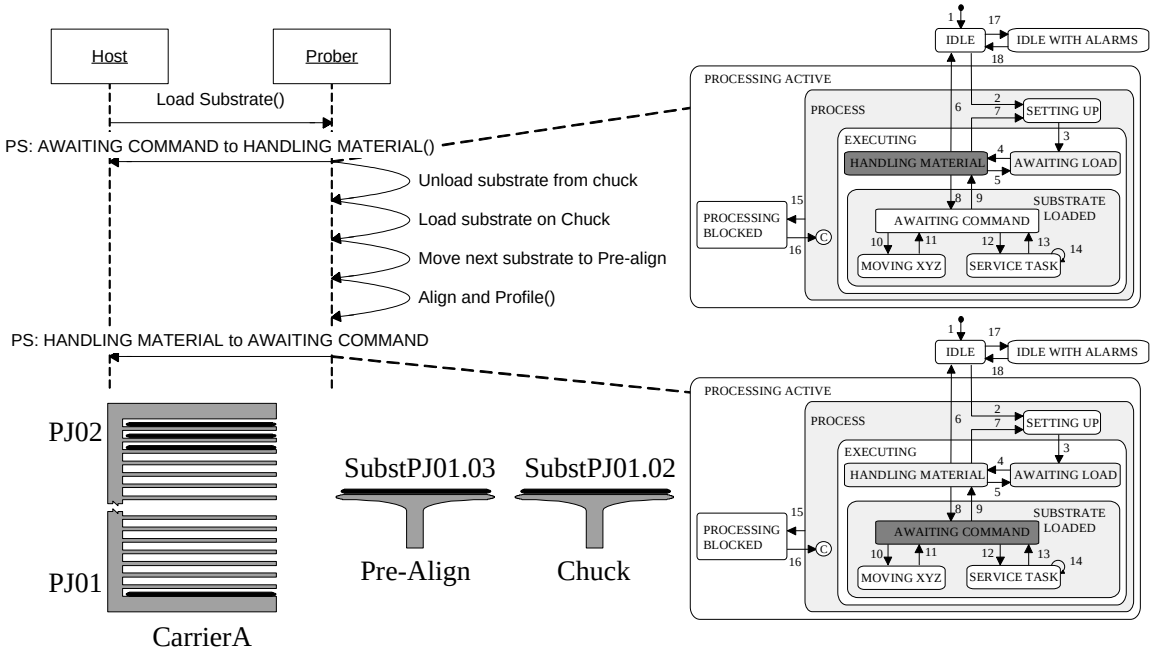


Figure R1-5
Normal Run - Load 2nd Substrate Scenario

Process Substrate SubstPJ01.02 (See Section R1-1.1.2.3)

R1-1.1.2.5 *Load 3rd Substrate – PJ02 Dispatched*

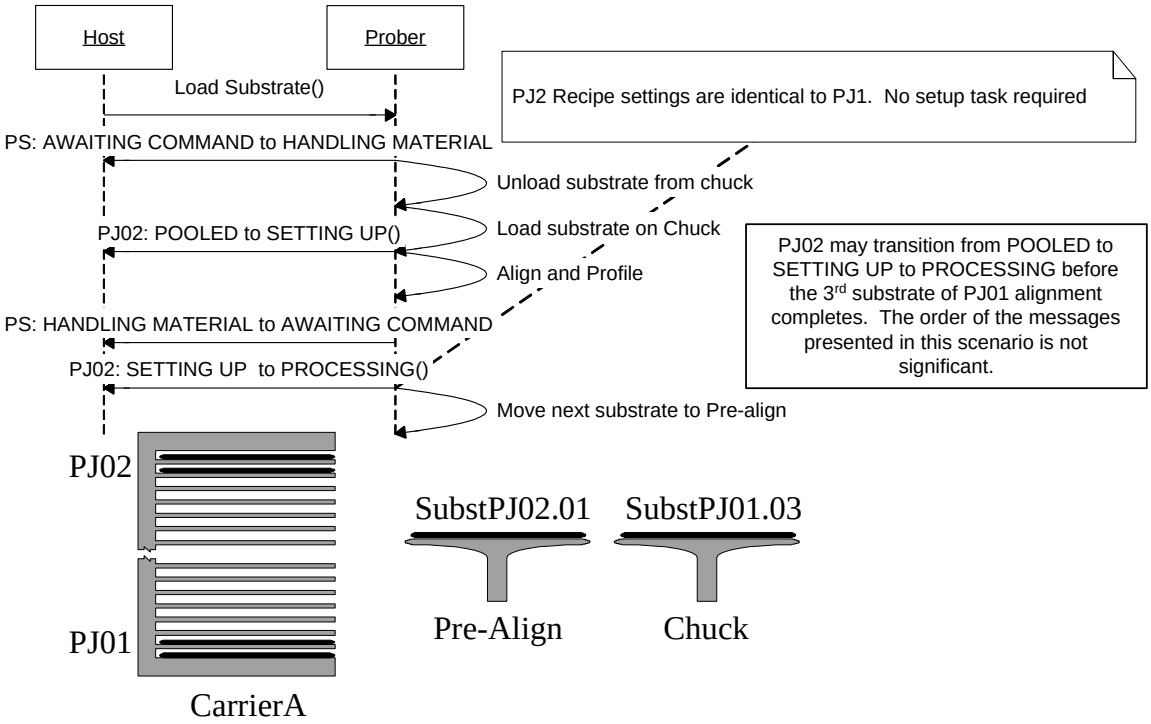


Figure R1-6
Normal Run - Load 3rd Substrate Scenario

Process Substrate SubstPJ01.03 (See Section R1-1.1.2.3)

R1-1.1.2.6 *Load 1st Substrate of PJ02. PJ01 Terminates*

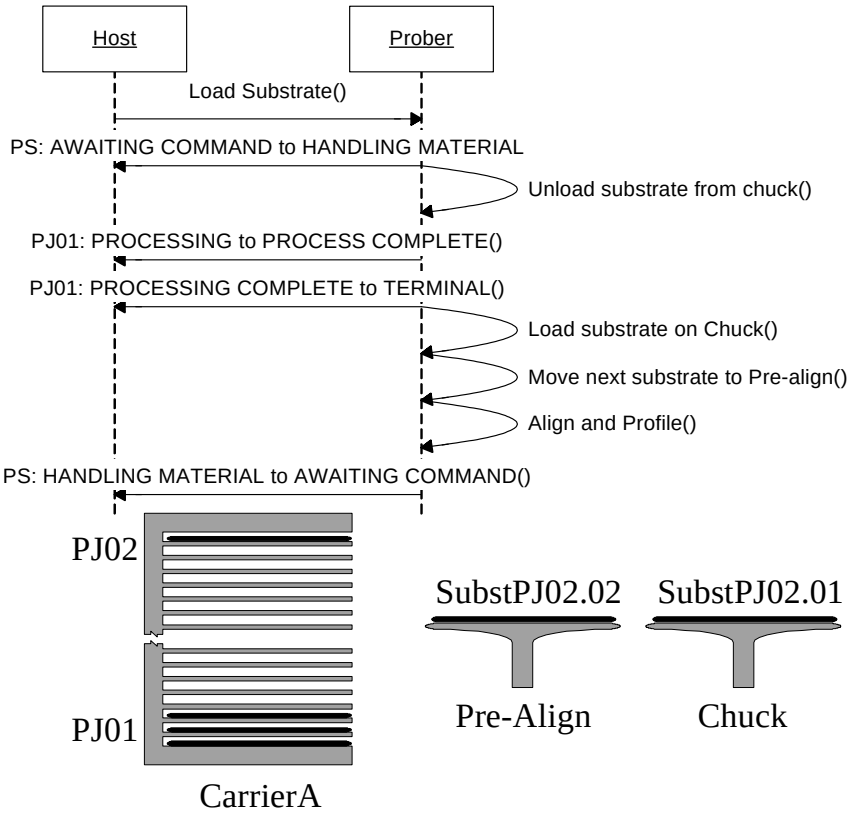


Figure R1-7
Normal Run - Load 1st Substrate of PJ02 Scenario

Process Substrate SubstPJ01.01 (See Section R1-1.1.2.3)

R1-1.1.2.7 *Load PJ02 Substrate 2*

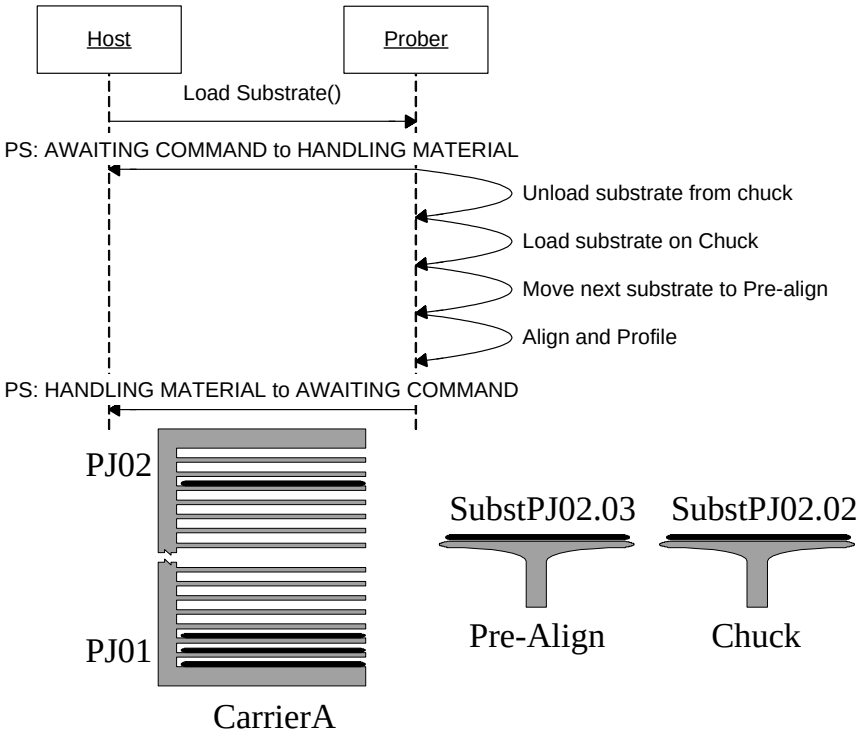


Figure R1-8
Normal Run - Load 2nd Substrate of PJ02 Scenario

Process Substrate SubstPJ02.02 (See Section R1-1.1.2.3)

R1-1.1.2.8 *Load PJ02 substrate 3*

Message sequence identical to Section R1-1.1.2.7

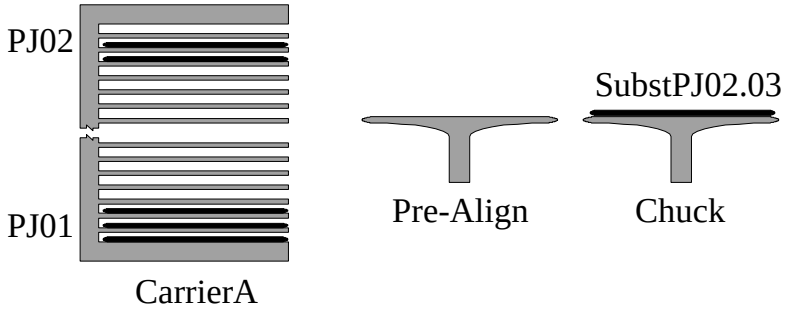


Figure R1-9
Normal Run - Load 3rd Substrate of PJ02 Scenario

Process Substrate SubstPJ02.03 (See Section R1-1.1.2.3)

R1-1.1.2.9 *Unload PJ02 substrate 3*

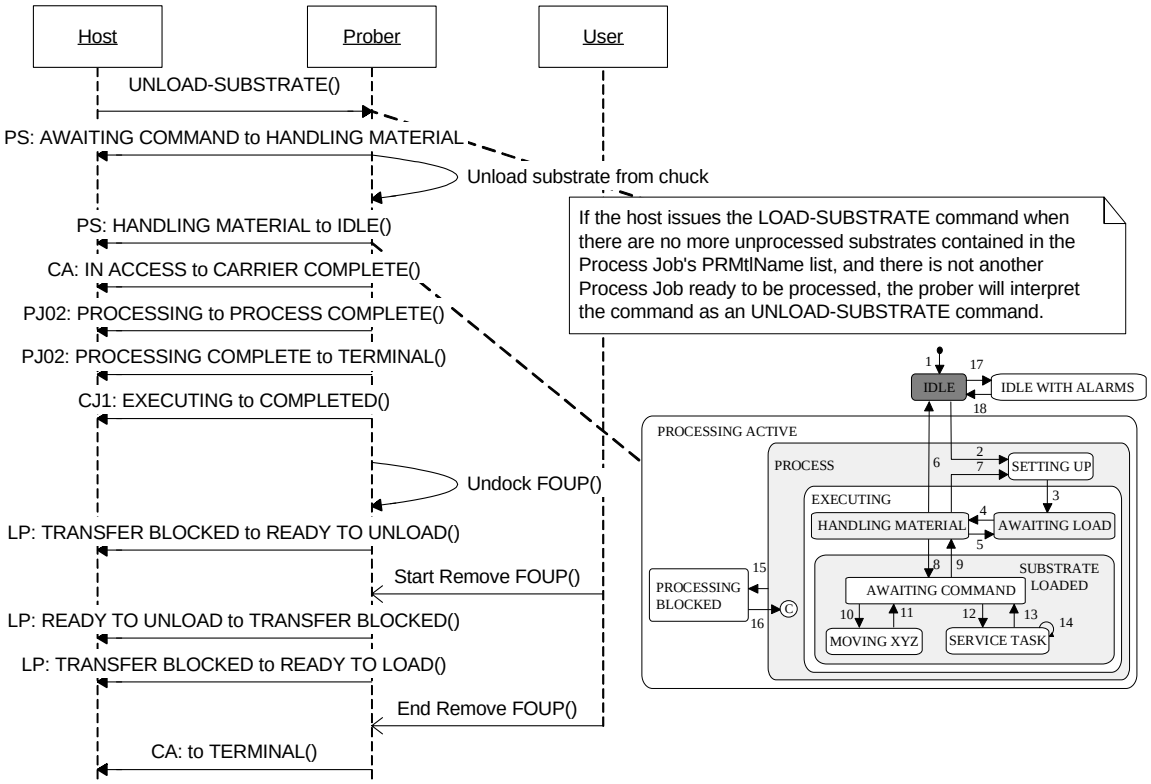


Figure R1-10
Normal Run – Unload 3rd Substrate of PJ02 Scenario

R1-1.2 *Scenario II – Substrate Tracking Events* — This scenario demonstrates the E90 Substrate tracking events corresponding to the job defined in Scenario I. E40 and E94 messages are omitted for clarity.

R1-1.2.1 *Job Setup*

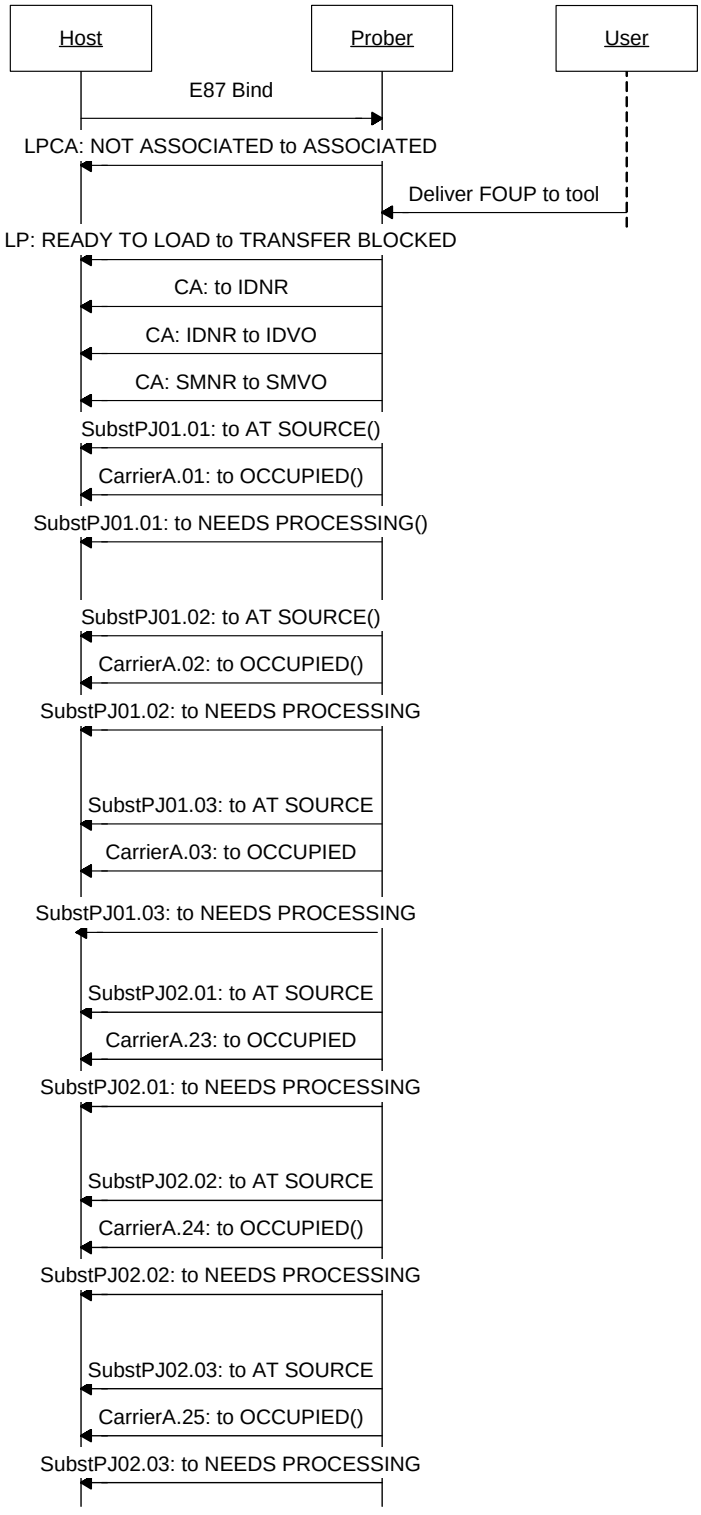


Figure R1-11
Substrate Tracking – Job Setup Scenario

R1-1.2.2 *Job Start*

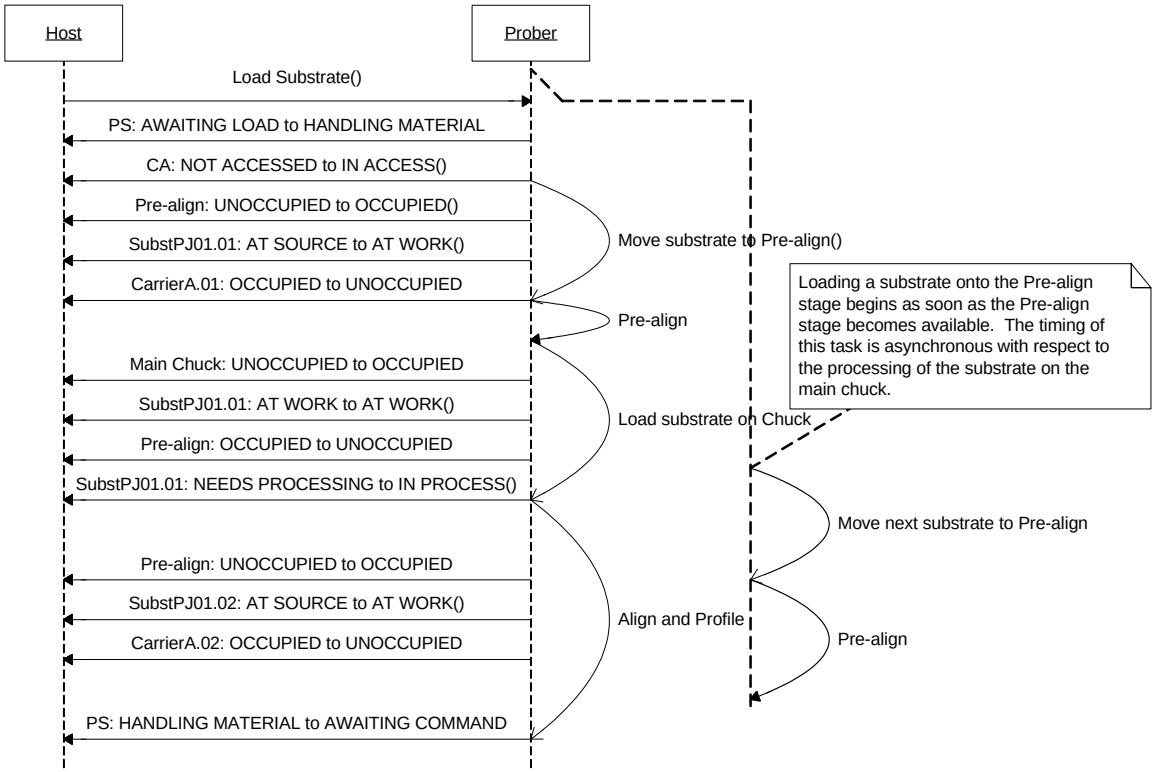


Figure R1-12
Substrate Tracking – Job Start Scenario

R1-1.2.3 *Process Material* — No Substrate Tracking events occur as the substrate on the main chuck is processed.

R1-1.2.4 *Load 2nd Substrate*

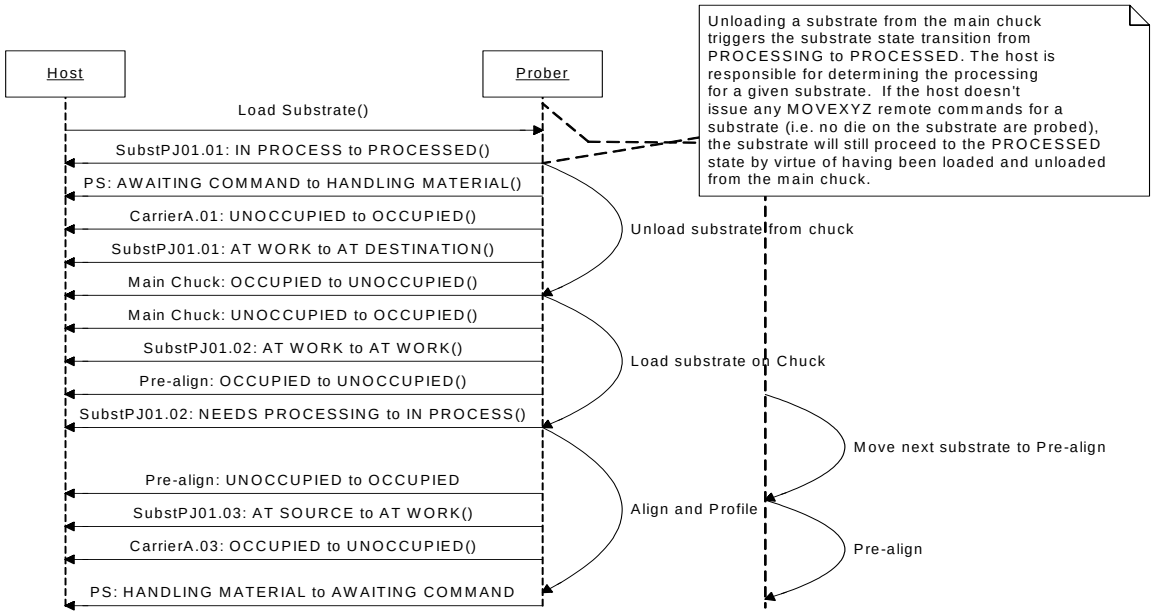


Figure R1-13
Substrate Tracking – Load 2nd Substrate Scenario

R1-1.2.5 *Process Remaining Material* — The Substrate Tracking events are identical for the remaining substrates in the two process jobs.

R1-1.3 *Scenario III – PJ1 is Stopped on 2nd Substrate* — This scenario involves the same job description as Scenario I. After the 2nd substrate from PJ1 is loaded on the chuck, the host issues a PJStop service message. PJ1 is unloaded and PJ2 continues processing. The purpose of this scenario is to demonstrate the affect of an abnormal job event as it relates to the Process State model.

R1-1.3.1 *Preconditions* — This section provides a summary of the sequence steps from scenario I leading up to the PJStop.

R1-1.3.1.1 Job setup. See Section R1-1.1.2.1

R1-1.3.1.2 Job Start. See Section R1-1.1.2.2.

R1-1.3.1.3 Process Material. See Section R1-1.1.2.3

R1-1.3.1.4 Load 2nd substrate. See Section R1-1.1.2.4.

R1-1.3.1.5 Process Material. See Section R1-1.1.2.3

R1-1.3.1.6 *Physical View*

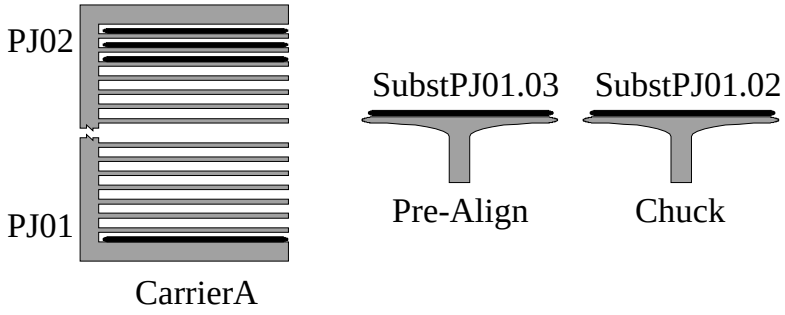


Figure R1-14
Physical View after 2nd Substrate Loaded

R1-1.3.1.7 *State Models*

Table R1-1 Object States after 2nd Substrate Loaded

Object Reference	*Description*	*States Occupied*
PS	Process State model	AWAITING COMMAND
CJ1	Control Job 1	EXECUTING
PJ01	Process Job 1	PROCESSING
PJ02	Process Job 2	PROCESSING
SubstPJ01.01	PJ1 Substrate 1	AT DESTINATION; PROCESSED
SubstPJ01.02	PJ1 Substrate 2	AT WORK; IN PROCESSED
SubstPJ01.03	PJ1 Substrate 3	AT WORK; NEEDS PROCESSING
SubstPJ02.01	PJ2 Substrate 1	AT SOURCE; NEEDS PROCESSING
SubstPJ02.02	PJ2 Substrate 2	AT SOURCE; NEEDS PROCESSING
SubstPJ02.03	PJ2 Substrate 3	AT SOURCE; NEEDS PROCESSING

R1-1.3.1.8 *Stop PJ1*

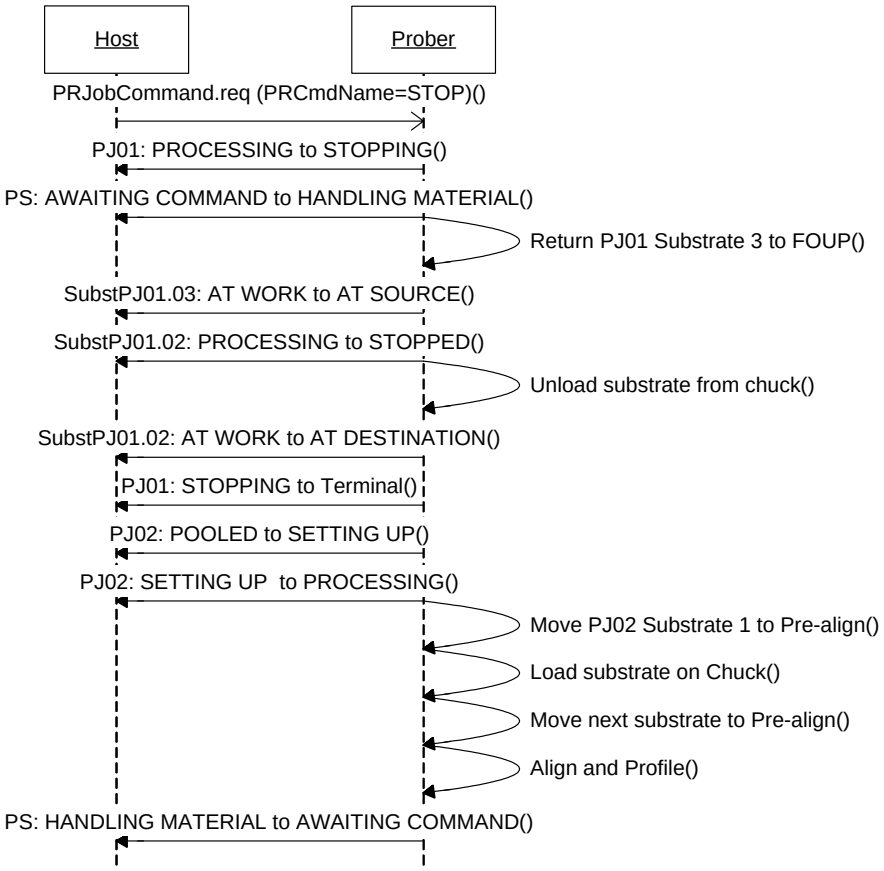


Figure R1-15
Stop PJ01 Scenario

R1-1.3.1.9 *Post Conditions*

R1-1.3.1.9.1 *State Models*

Table R1-2 Object States After PJ01 Stopped

Object Reference	*Description*	*States Occupied*
PS	Process State model	AWAITING COMMAND
CJ1	Control Job 1	EXECUTING
PJ01	Process Job 1	Terminal (non existent)
PJ02	Process Job 2	PROCESSING
SubstPJ01.01	PJ1 Substrate 1	AT DESTINATION; PROCESSED
SubstPJ01.02	PJ1 Substrate 2	AT DESTINATION; STOPPED
SubstPJ01.03	PJ1 Substrate 3	AT SOURCE; NEEDS PROCESSING
SubstPJ02.01	PJ2 Substrate 1	AT WORK; IN PROCESS
SubstPJ02.02	PJ2 Substrate 2	AT WORK; NEEDS PROCESSING
SubstPJ02.03	PJ2 Substrate 3	AT SOURCE; NEEDS PROCESSING

R1-1.3.1.10 *Post Processing* — The following sequence would be followed to complete the job:

R1-1.3.1.10.1 Load PJ02 Substrate 2. See Section R1-1.1.2.7

R1-1.3.1.10.2 Process Material. See Section R1-1.1.2.3

R1-1.3.1.10.3 Load PJ02 Substrate 3. See Section R1-1.1.2.8

R1-1.3.1.10.4 Process Material. See Section R1-1.1.2.3

R1-1.3.1.10.5 Unload PJ02 Substrate 3. See Section R1-1.1.2.9

R1-1.4 *Scenario IV – Alarm Condition Causes a Process State Model Transition to the PROCESSING BLOCKED State* — The purpose of this scenario is to demonstrate the behavior of prober when an alarm condition occurs that causes the processing resource to become unavailable. The Process State model transitions to the PROCESSING BLOCKED state. The condition does not cause a corresponding transition to the PAUSING state within an executing Process Job or a transition to the PAUSED state within the active Control Job.

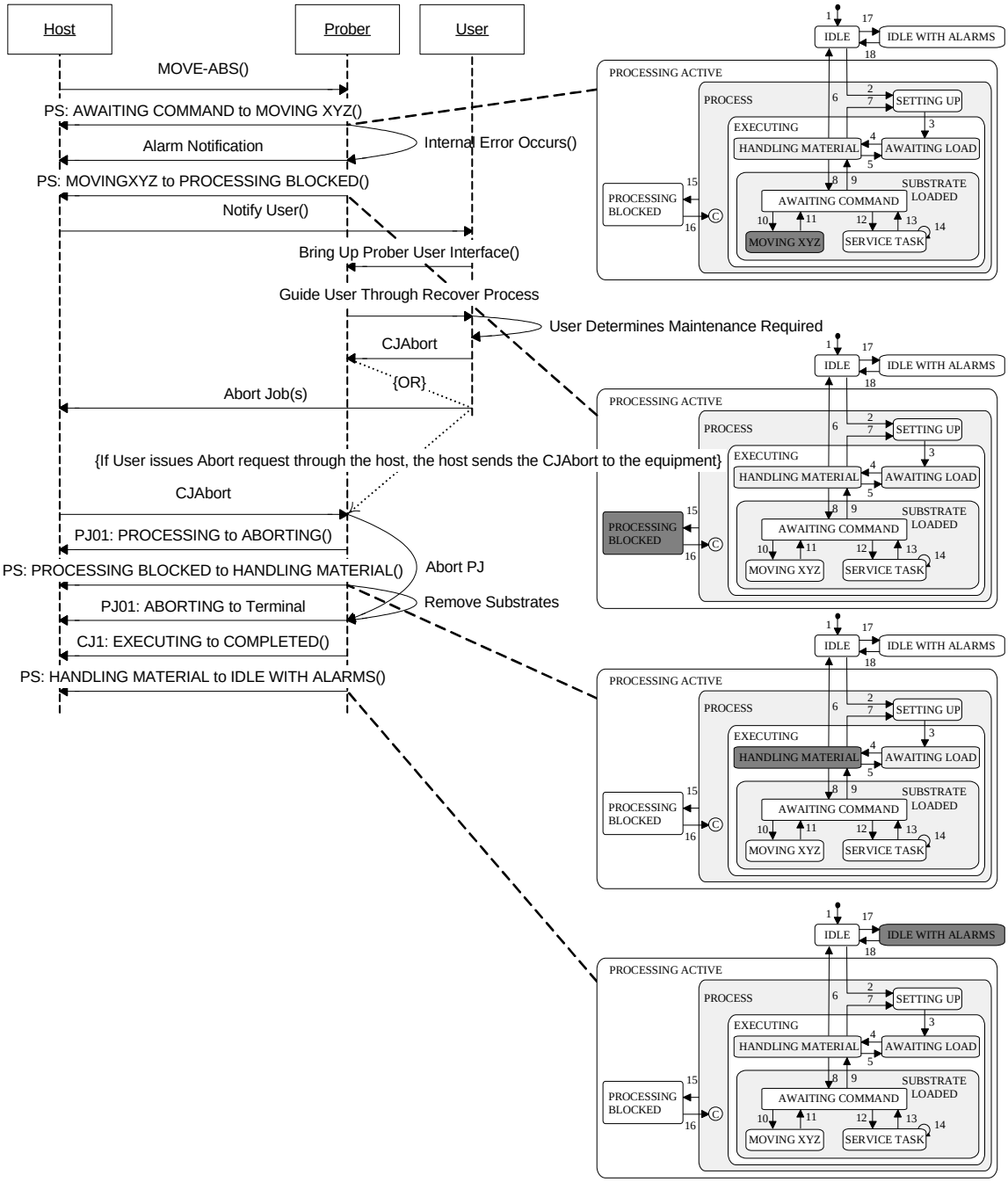
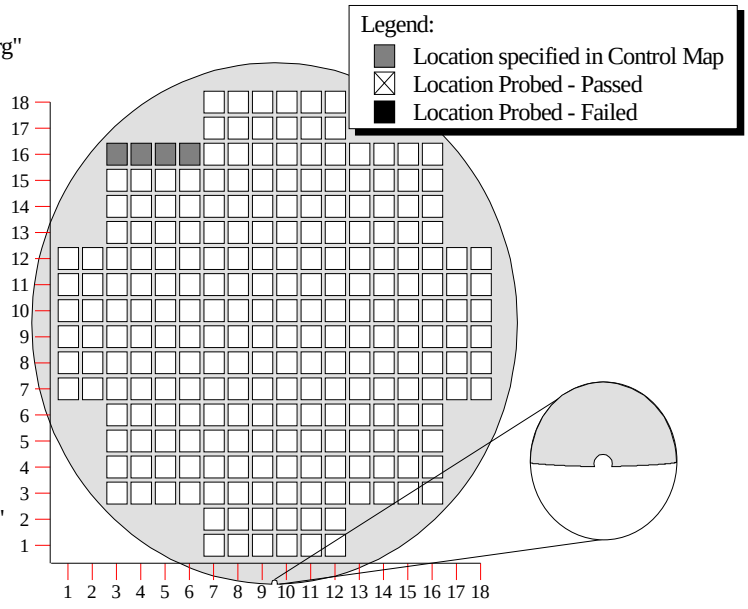


Figure R1-16
Prober Alarm Scenario

R1-1.5 *Scenario V – Processing Material using Control Maps* — Control maps can be used for identifying the sites to be probed per substrate. The specification for the download of maps is outside the scope of this specification. The format of the maps is user defined.

R1-1.5.1 *Example Settings* — For the sake of demonstration, the control map pertaining to the substrate in this example is communicated using the SEMI G85 map format. The recipe parameter “USE CONTROL MAP” is set to true.

```
<?xml version="1.0"?>
<Map xmlns:semi="http://www.semi.org"
  WaferId="39211USEC4"
  FormatRevision="G85-1101">
  <Device
      ProductId="Example300"
      ProductCode="xx"
      LotId="1234567890"
      Orientation="0"
      WaferSize="300"
      DeviceSizeX="15968.77"
      DeviceSizeY="15968.77"
      StepSizeX="15968.77"
      StepSizeY="15968.77"
      Rows="18"
      Columns="18"
      SupplierName="xx"
      OriginLocation="3"
      CreateDate="2002100815124800"
      Status="70"
      BinType="Decimal"
      NullBin="2">
     <Bin BinCode="1" BinCount="4"   BinQuality="Good" BinDescription="Test Site"/>
     <Bin BinCode="0" BinCount="240" BinQuality="Good" BinDescription="Non Test Site"/>
     <Bin BinCode="2" BinCount="80"  BinQuality="Fail" BinDescription="Non Testable Site"/>
     <Data MapName="PSEM300 Example" Version="1">
      <Row><![CDATA[2 2 2 2 2 2 0 0 0 0 0 0 2 2 2 2 2 2]]></Row>
      <Row><![CDATA[2 2 2 2 2 2 0 0 0 0 0 0 2 2 2 2 2 2]]></Row>
      <Row><![CDATA[2 2 0 0 0 0 0 0 0 0 0 0 0 0 2 2]]></Row>
      <Row><![CDATA[2 2 0 0 0 0 0 0 0 0 0 0 0 0 2 2]]></Row>
      <Row><![CDATA[2 2 0 0 0 0 0 0 0 0 0 0 0 0 2 2]]></Row>
      <Row><![CDATA[2 2 0 0 0 0 0 0 0 0 0 0 0 0 2 2]]></Row>
      <Row><![CDATA[0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0]]></Row>
      <Row><![CDATA[0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0]]></Row>
      <Row><![CDATA[0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0]]></Row>
      <Row><![CDATA[0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0]]></Row>
      <Row><![CDATA[0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0]]></Row>
      <Row><![CDATA[0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0]]></Row>
      <Row><![CDATA[2 2 0 0 0 0 0 0 0 0 0 0 0 0 2 2]]></Row>
      <Row><![CDATA[2 2 0 0 0 0 0 0 0 0 0 0 0 0 2 2]]></Row>
      <Row><![CDATA[2 2 0 0 0 0 0 0 0 0 0 0 0 0 2 2]]></Row>
      <Row><![CDATA[2 2 1 1 1 1 0 0 0 0 0 0 0 0 2 2]]></Row>
      <Row><![CDATA[2 2 2 2 2 2 0 0 0 0 0 0 2 2 2 2 2 2]]></Row>
      <Row><![CDATA[2 2 2 2 2 2 0 0 0 0 0 0 2 2 2 2 2 2]]></Row>
     </Data>
  </Device>
</Map>
```



Site Selection List		
Die Number	Row	Col
1	16	3
2	16	4
3	16	5
4	16	6

Figure R1-17
Control Map Usage Example Settings

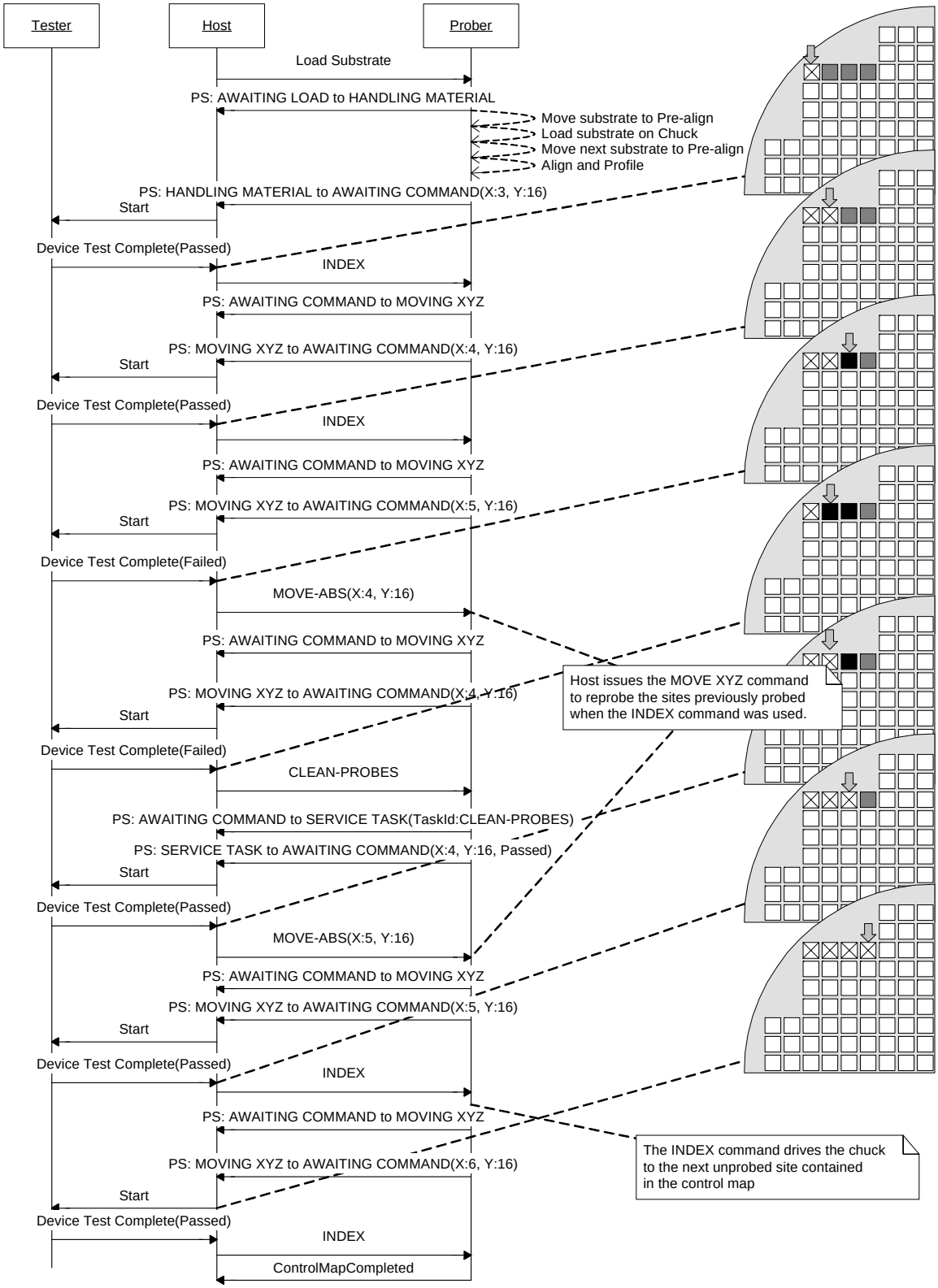


Figure R1-18
Control Map Usage Scenario

R1-1.6 *Scenario VI – Recoverable Alarm* — An error occurs when the prober is responding to a MOVE XYZ command (the chuck could not complete the motion). The host is able to restart processing after the chuck is initialized.

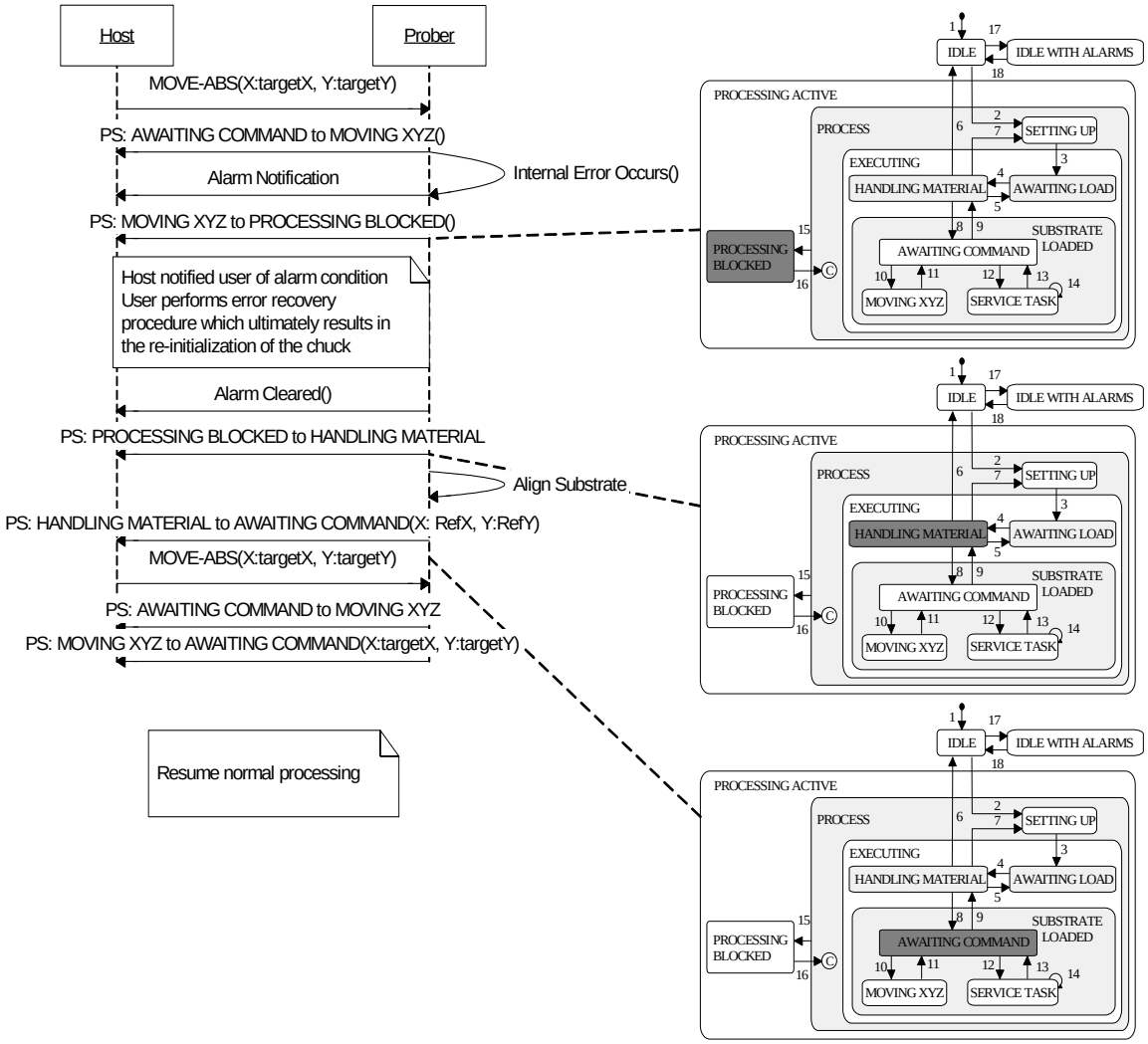


Figure R1-19
Recoverable Alarm Scenario

R1-1.7 *Scenario VII – Inspect and Clean Probes* — The host issues an INSPECT-PROBES command during substrate processing. The probe inspection fails so the host issues the CLEAN-PROBES command. The prober is configured to automatically inspect the probes after the probe cleaning operation. The host resumes processing after the successful clean/inspection:

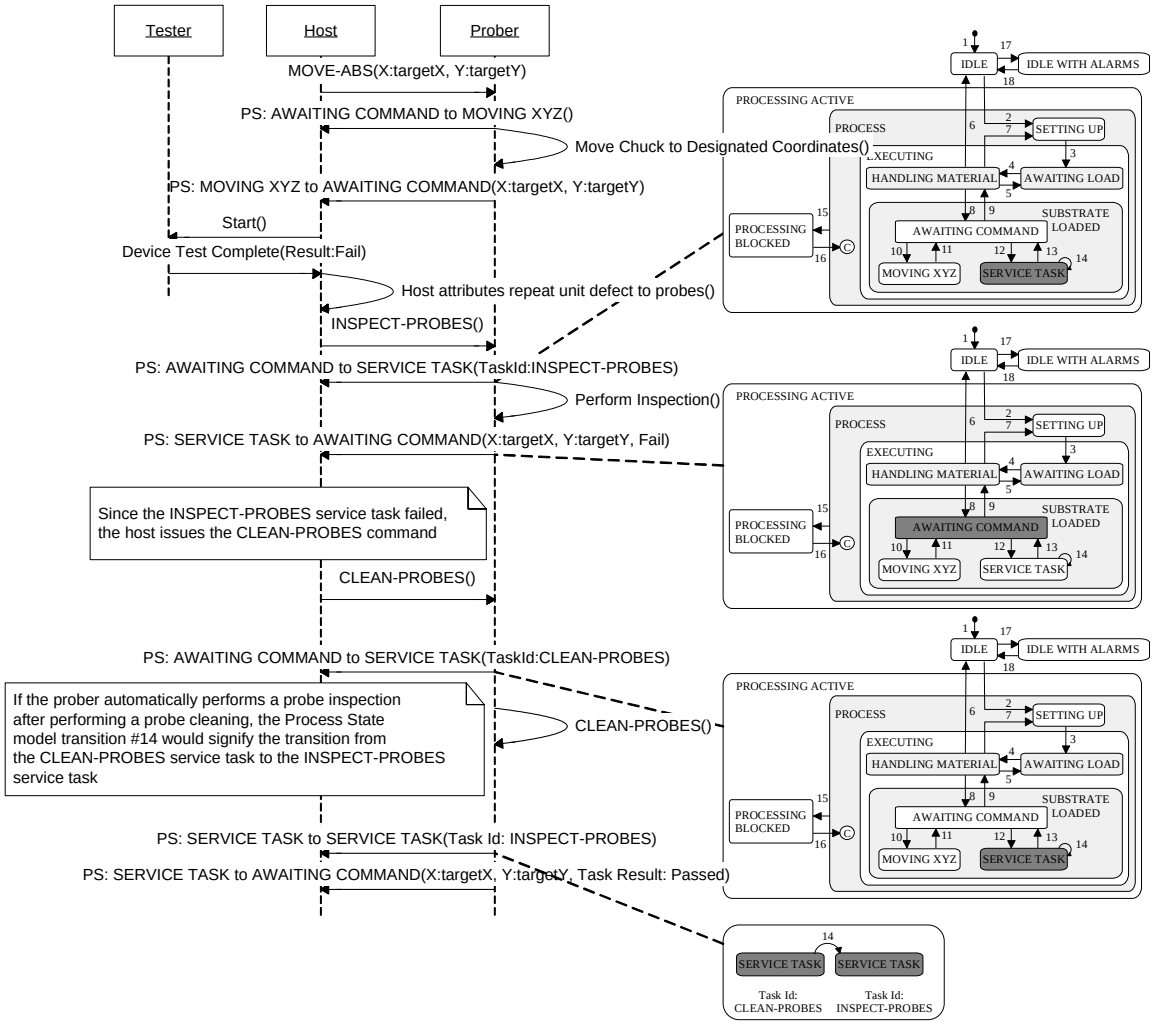


Figure R1-20
Probe Inspection and Cleaning Scenario

R1-1.7.2 *Cleaning Probes – Detail* — This scenario assumes the recipe is setup for cleaning probes using a cleaning substrate. The cleaning substrate is stored in an internal storage site. The substrates currently loaded on the chuck and pre-align stage(s) are returned to the source carrier so that the cleaning substrate can be loaded on the chuck. When the cleaning substrate is removed from the chuck, the substrate(s) that occupied the chuck/pre-align stage(s) are restored.

R1-1.7.2.1 *Points of Interest*

R1-1.7.2.1.1 The service task involves moving substrates between substrate locations. This is not reflected in the Process State model. The host is not involved in the service task. Therefore, there is no need for the Process State model to transition to the HANDLING MATERIAL state.

R1-1.7.2.2 *Pre-Conditions*

R1-1.7.2.2.1 *Job Setup* — See Section R1-1.1.2.1

R1-1.7.2.2.2 *Job Start* — See Section R1-1.1.2.2.

R1-1.7.2.2.3 *State Models*

Table R1-3 Object States for Probe Cleaning Scenario

Object Reference	*Description*	*States Occupied*
PS	Process State model	AWAITING COMMAND
CJ1	Control Job 1	EXECUTING
PJ01	Process Job 1	PROCESSING
PJ02	Process Job 2	POOLED
SubstPJ01.01	PJ1 Substrate 1	AT WORK; IN PROCESS
SubstPJ01.02	PJ1 Substrate 2	AT WORK; NEEDS PROCESSING
SubstPJ01.03	PJ1 Substrate 3	AT SOURCE; NEEDS PROCESSING
SubstPJ02.01	PJ2 Substrate 1	AT SOURCE; NEEDS PROCESSING
SubstPJ02.02	PJ2 Substrate 2	AT SOURCE; NEEDS PROCESSING
SubstPJ02.03	PJ2 Substrate 3	AT SOURCE; NEEDS PROCESSING

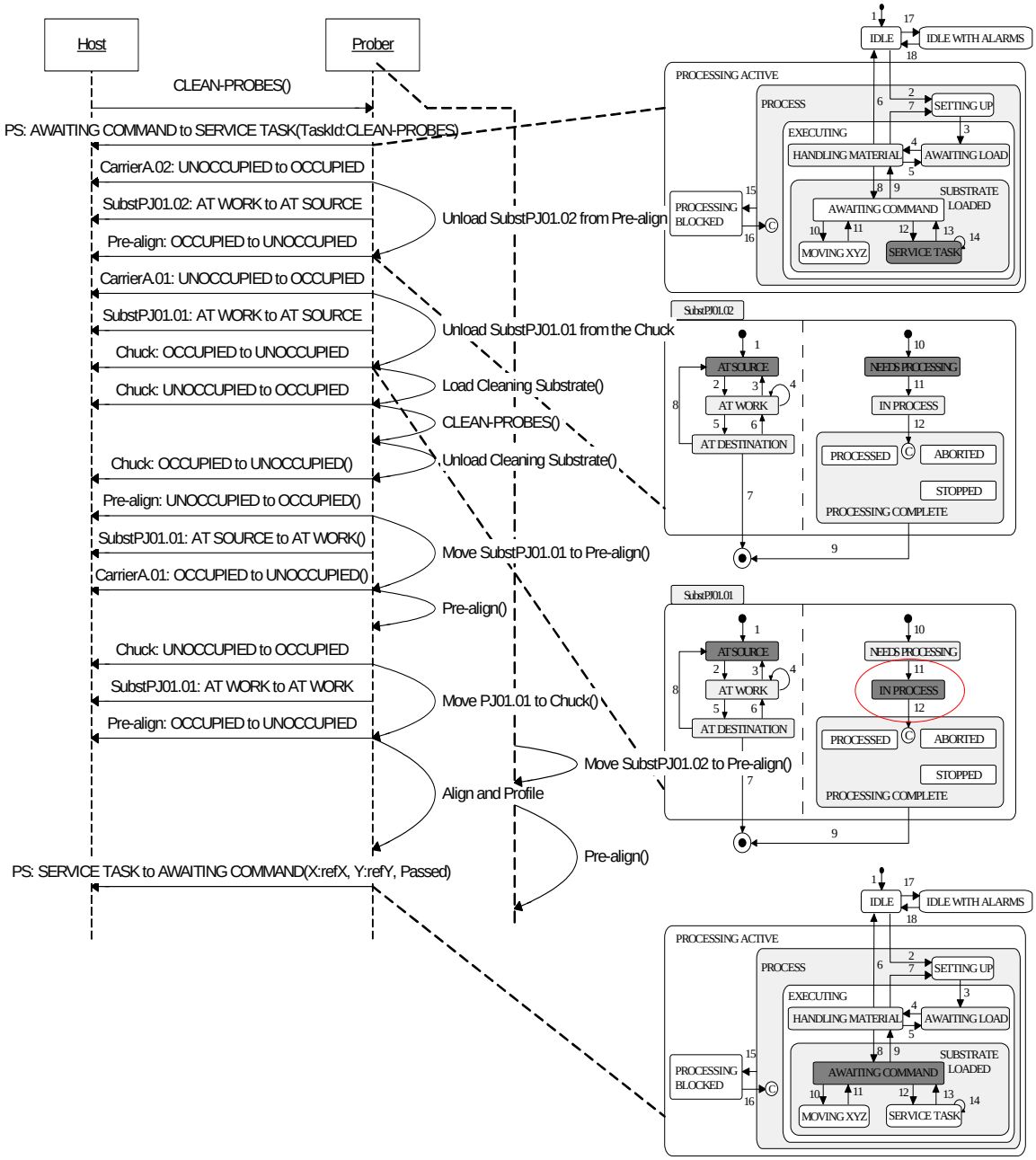


Figure R1-21
Clean Probes Detailed Scenario

R1-1.7.2.3 *Post Conditions* — The post conditions match the pre-conditions.

R1-1.8 *Scenario VIII – Pre-align Failure on 1st Substrate* — The prober fails to setup the 1st substrate of a job (the chuck is unoccupied). The user manually aligns substrate and presses Start on the prober console to resume processing.

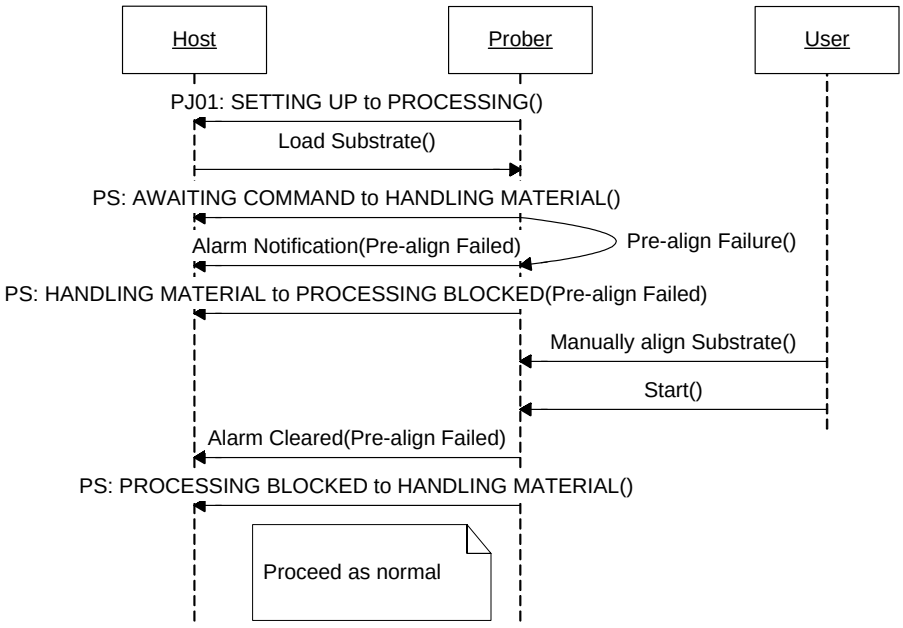


Figure R1-22
Pre-Align Failure on 1st Substrate Scenario

R1-1.9 *Scenario IX – Pre-align Fails on Subsequent Substrate. User Clears Condition Prior to the Host Issuing the Next LOAD-SUBSTRATE Command* — The prober fails to pre-align a substrate while the host is processing a substrate on the main chuck. The user manually aligns the substrate and presses Start on the prober console to clear the alarm. The next LOAD-SUBSTRATE command from the host proceeds as normal.

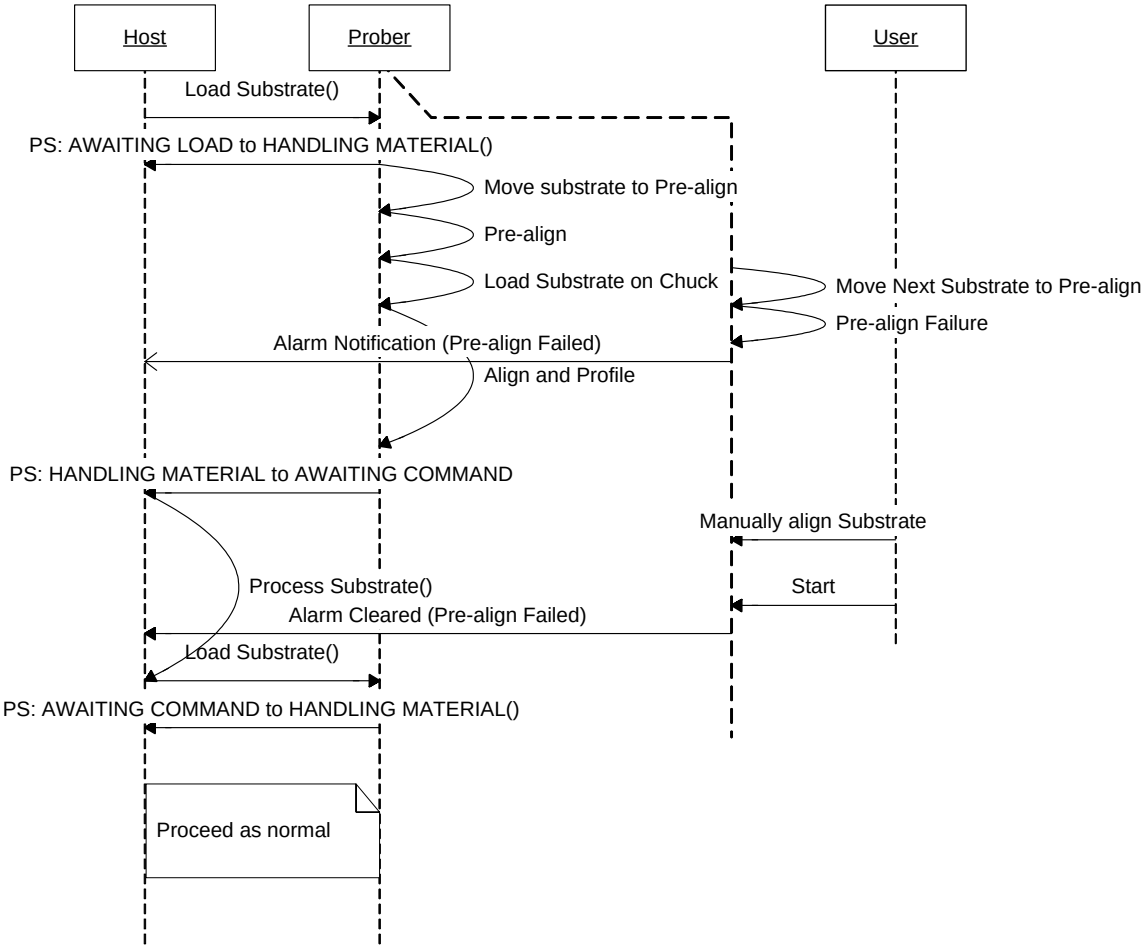


Figure R1-23
Pre-Align Failure 2 – Alarm Cleared by User

R1-1.10 *Scenario X – Pre-align Fails on Subsequent Substrate. User Fails to Clear the Condition Prior to the Host Issuing the Next LOAD-SUBSTRATE Command* — The prober fails to pre-align a substrate while the host is processing a substrate on the main chuck. The user manually aligns the substrate but fails to complete the manual alignment prior to the host issuing the LOAD-SUBSTRATE command. The Process State model transitions to the PROCESSING BLOCKED state until the user clears the alarm. When the alarm is cleared, the Process State model transitions to the HANDLING MATERIAL state and processing resumes as normal.

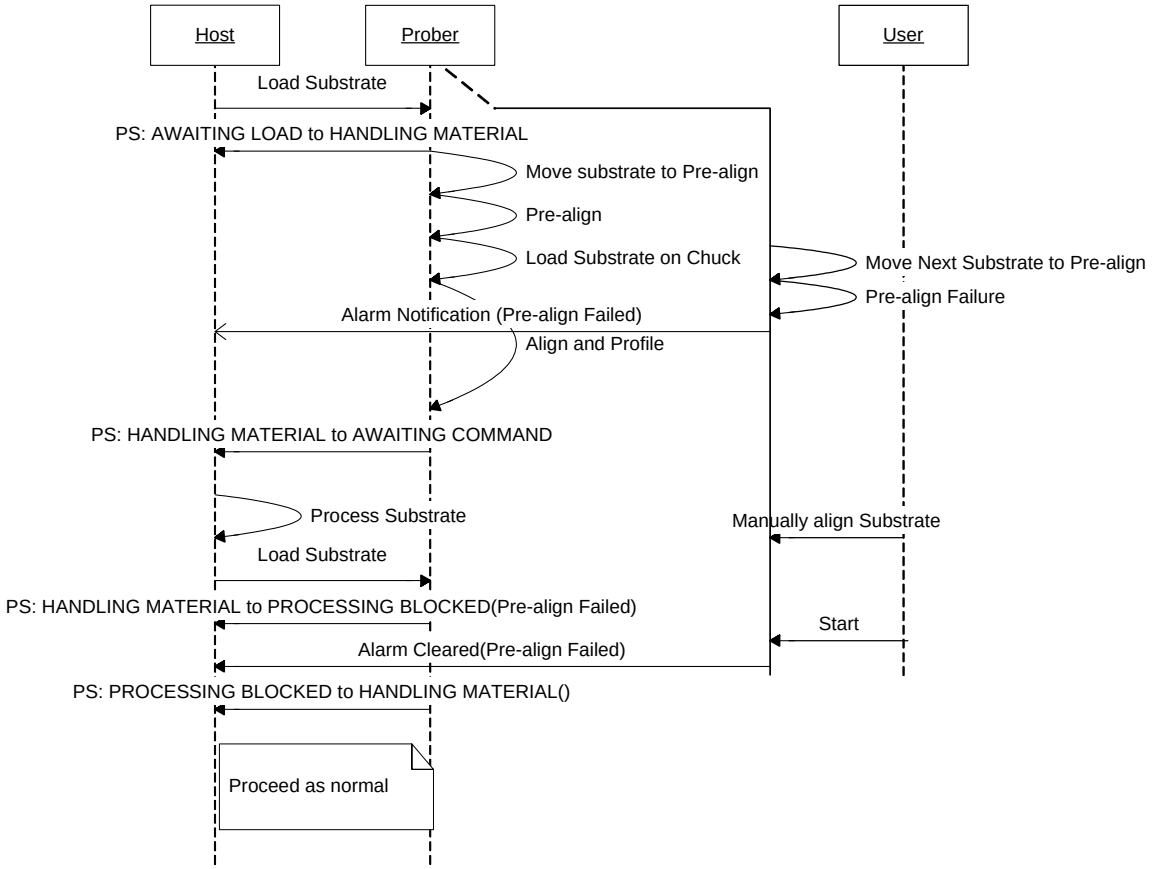


Figure R1-24
Pre-Align Failure 3 – User Failed to clear Alarm

R1-1.11 *Scenario XI – Correlation Substrate Testing*

R1-1.11.1 *Overview* — This scenario describes the use of a correlation substrate. The correlation substrate is loaded on the chuck to verify the test results for the current setup. From the Process/Control Job perspective, correlation substrate testing is a normal task available to the host. Therefore, there are no job related collection events associated with correlation testing.

R1-1.11.1.1 The correlation substrate is not part of a Process Job. The correlation substrate must be run using the same recipe as the currently loaded process job.

R1-1.11.2 *Pre-Conditions*

R1-1.11.2.1 *Job Setup* — See Section R1-1.1.2.1

R1-1.11.2.2 *Job Start* — See Section R1-1.1.2.2.

R1-1.11.2.3 *Physical View*

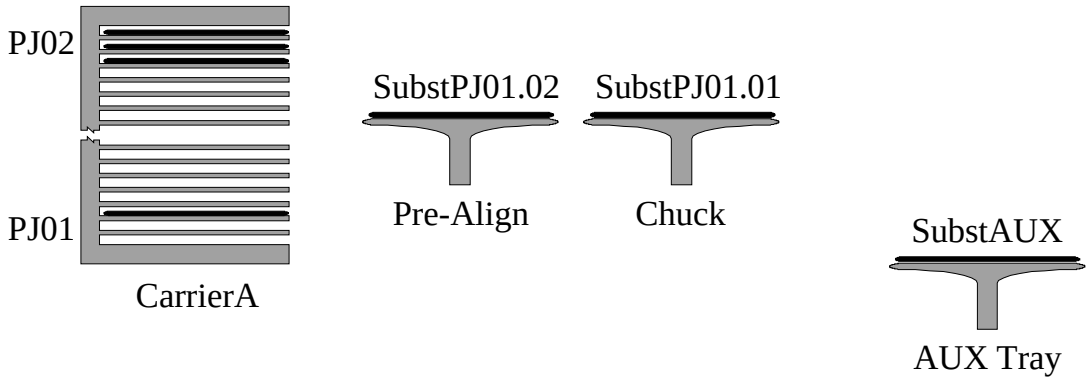


Figure R1-25
Correlation Scenario - Physical View

R1-1.11.2.4 *State Models*

Table R1-4 Correlation Scenario Object States

Object Reference	*Description*	*States Occupied*
PS	Process State model	AWAITING COMMAND
CJ1	Control Job 1	EXECUTING
PJ01	Process Job 1	PROCESSING
PJ02	Process Job 2	POOLED
SubstPJ01.01	PJ1 Substrate 1	AT WORK; IN PROCESS
SubstPJ01.02	PJ1 Substrate 2	AT WORK; NEEDS PROCESSING
SubstPJ01.03	PJ1 Substrate 3	AT SOURCE; NEEDS PROCESSING
SubstPJ02.01	PJ2 Substrate 1	AT SOURCE; NEEDS PROCESSING
SubstPJ02.02	PJ2 Substrate 2	AT SOURCE; NEEDS PROCESSING
SubstPJ02.03	PJ2 Substrate 3	AT SOURCE; NEEDS PROCESSING
SubstAUX	Correlation Substrate	AT SOURCE; NEEDS PROCESSING

R1-1.11.2.5 *Process Correlation Substrate* — The substrates in the pipeline are returned to the source FOUP. Substrate SubstPJ01.01 is suspended and remains in the IN PROCESS state.

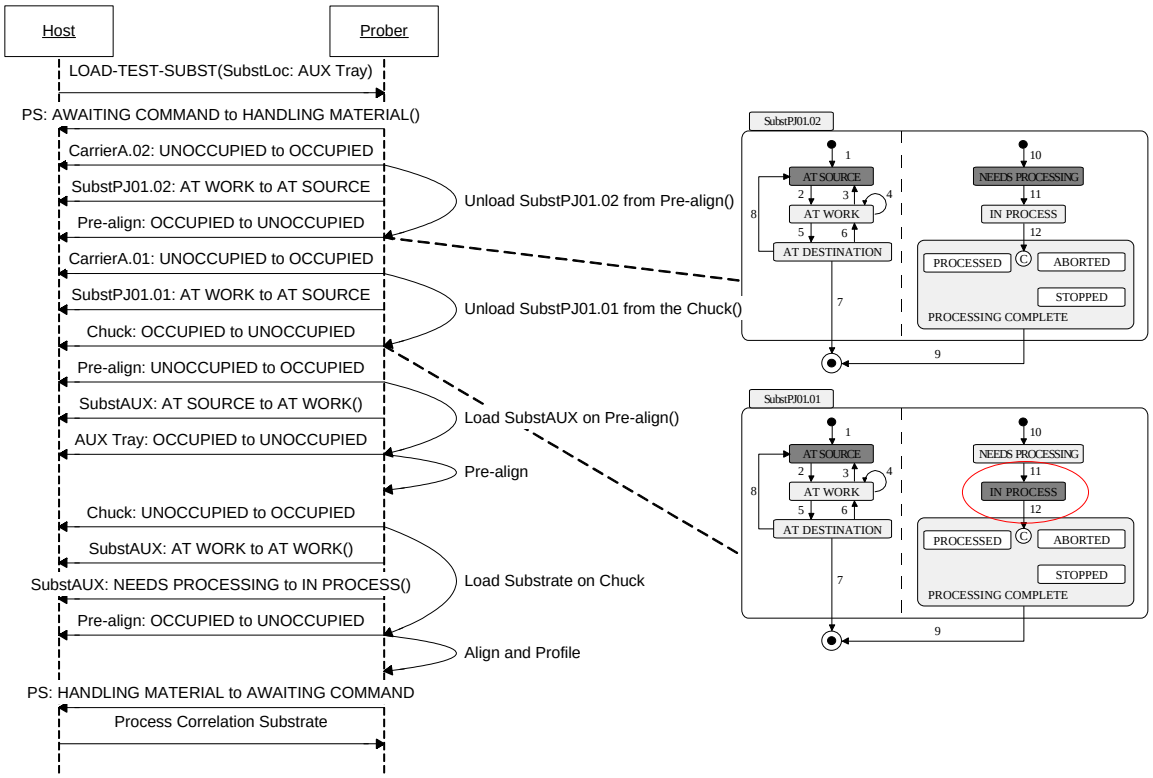


Figure R1-26
Correlation Substrate Processing Scenario

R1-1.11.2.6 *Unload Correlation Substrate* — Unloading the correlation substrate triggers the prober to restore the substrates to the state that existed prior to the host issuing the LOAD-TEST-SUBST command.

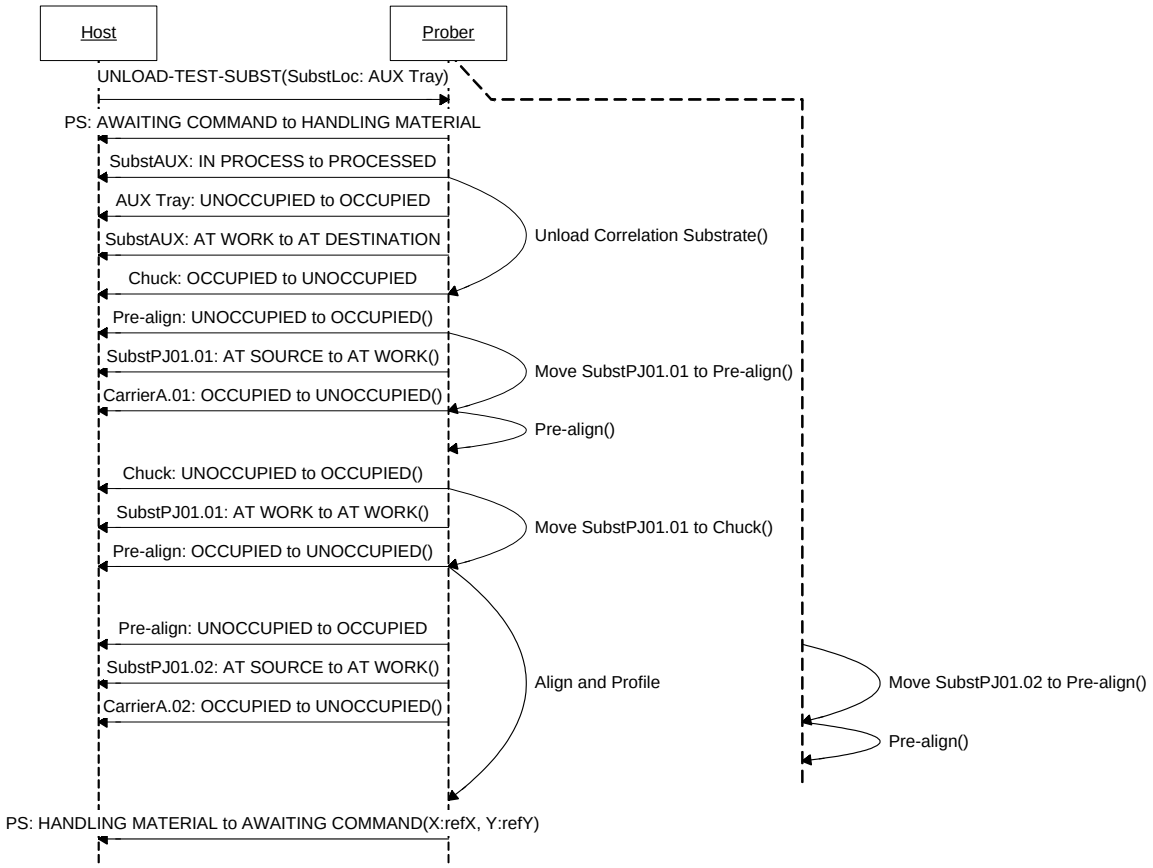


Figure R1-27
Unload Correlation Substrate Scenario

R1-1.11.2.7 *Post-Conditions* — The prober has restored the substrates to their original condition that preceded the host issuing the LOAD-TEST-SUBST command. SubstPJ01.01 is on the chuck. The chuck is positioned at the coordinates: refX,refY.

R1-1.11.2.7.1 If a control map is being used, the control map is reset and the chuck is positioned on the first site contained in the map.

R1-2 Additional Non-Standardized Behaviors

This section contains a list of behaviors that are desirable but outside the scope of the PSEM300 standard.

R1-2.1 *Off-Board Inspection*

R1-2.1.1 When the need exists to remove a substrate for external inspection, the prober should remember that the substrate has been removed from the equipment. The Process Job pertaining to the given substrate should not terminate until the substrate has been returned to the correct carrier substrate location as designated in the Control Job's MtrlOutSpec (or to the source location if the MtrlOutSpec was not provided).

R1-2.1.2 Support for off-board inspection is not part of this specification due to the challenge of tracking a substrate that has been removed from the tool. However, this feature can be requested as a user defined feature. No guidance is available within this specification regarding the tracking of the substrate or the related impact on the PJ/CJ and Carrier states.

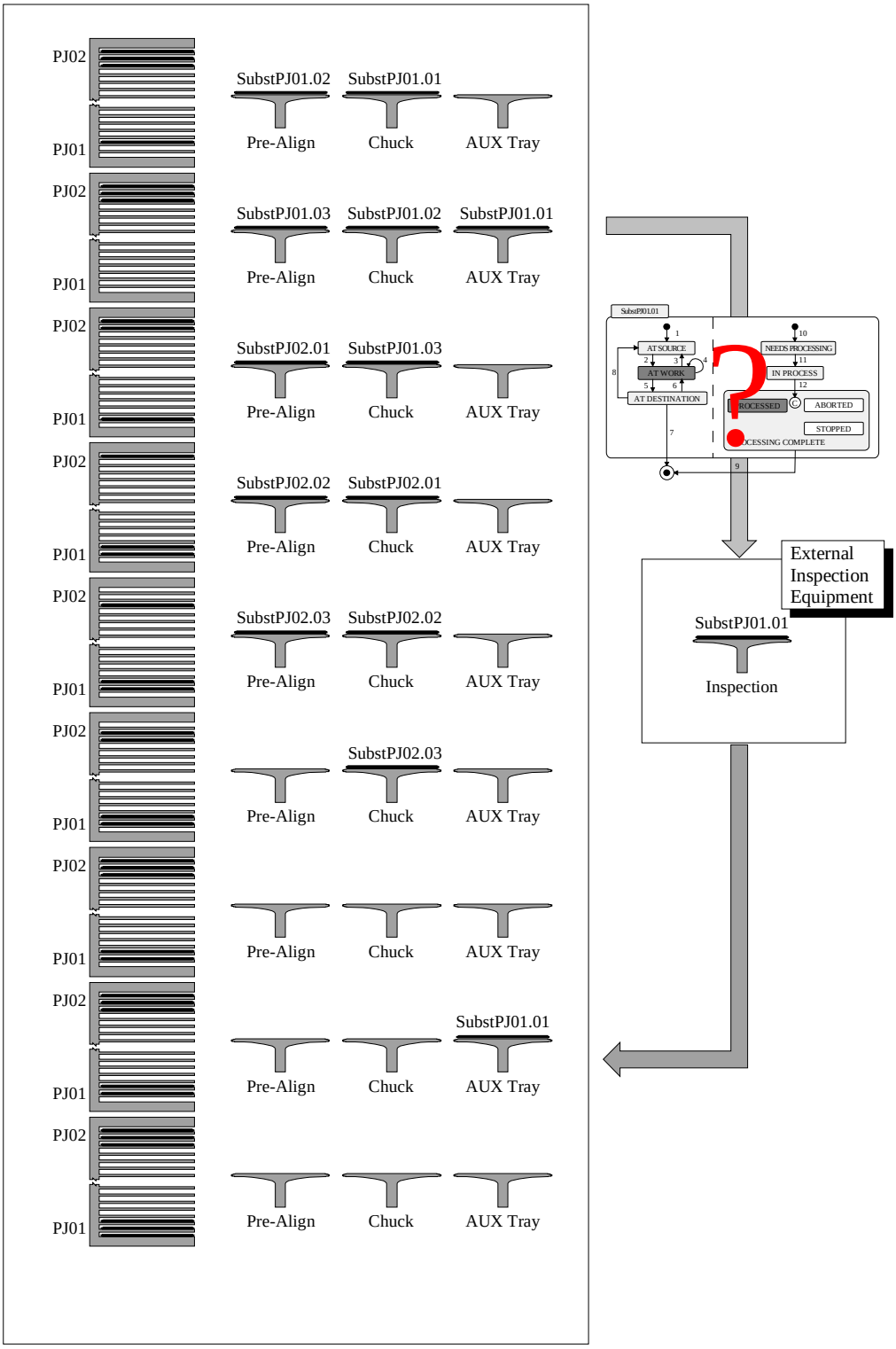


Figure R1-28
Off-Board Inspection Scenario

RELATED INFORMATION 2

NOTICE: This related information is not an official part of SEMI E130 and was derived from the work of the originating task force. This related information was approved for publication by full letter ballot procedures on September 3, 2003.

R2-1 Application for Self-Indexing Probing

R2-1.1 Following ideas for implementation of specific integrations are not a part of this specification but applications for some regency solution to maximize compliance to this specification.

R2-1.2 Some wafer prober with combined specific IC testers may index dice automatically by itself synchronized with such communication as GPIB or RS-232C between tester and prober. According state model in this specification, a series of this automated indexing and testing could be execution of a service task (hatched area in following diagram).

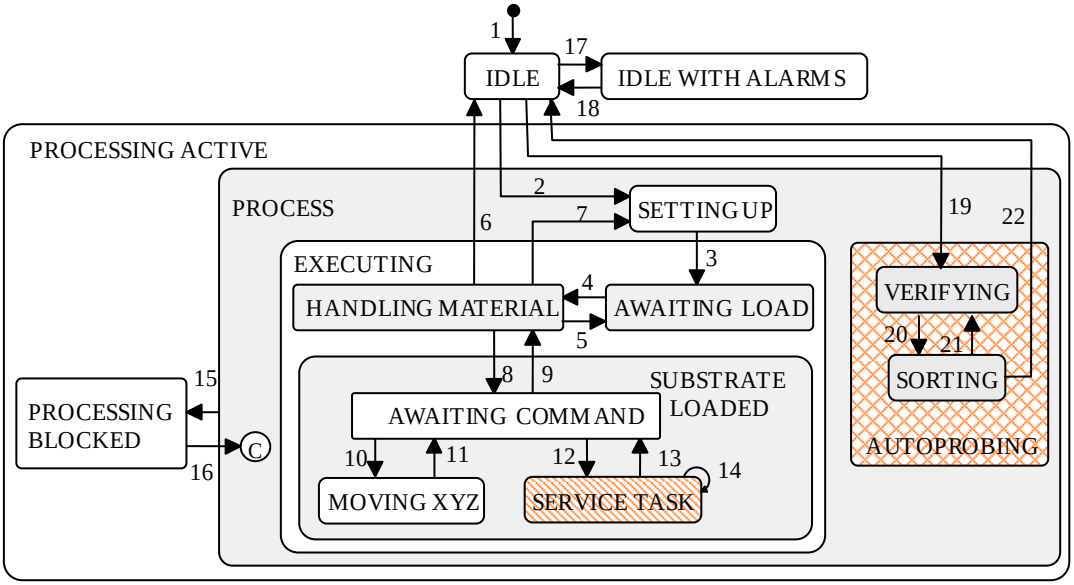


Figure R2-1
Self-Indexing State Model

R2-1.2.1 For the other application to distinguish this normal probing task from such generic service tasks as Cleaning Probes, adding substate “AUTOPROBING” in the state of “PROCESS” is one of ideas as an extension of this specification. The diagram above illustrates the extension (cross-hatched substate) of a part of the state model in this specification. This extension also eliminates loading/unloading remote commands and related states, and is more close to processing state in SEMI E91 (PSEM).

R2-1.2.2 Above diagram illustrates the extension (cross-hatched substate) of a part of the state model in this specification. Clarification of extended sub-states and transitions are given below.

R2-1.3 *AUTOPROBING* — Automatic processing state without instruction of host. Programmed such service operations as inking cleaning are included in this states as well as probing sequence with automated self indexing dice. Usually prober communicates with tester through serial or parallel cable in this state.

R2-1.4 *VERIFYING* — One of substates of AUTOPROBING. This states starts before loading the first wafer of PJ and concludes when all set-up conditions and all alignments for the first wafer of PJ including needle-pad alignment for the first dice on the first wafer is completed.

R2-1.5 *SORTING* — One of substates of AUTOPROBING. Prober executes processing automatically without specific host instruction. The prober needs thecapability to index to the right dice by itself, communicate with the tester, and execute programmed sequences including such service operation as cleaning and inking.

Table R2-1 Self-Indexing State Transition Table

#.	*Current State*	*Trigger*	*New State*	*Action*	*Comment*
19	IDLE	A process job has been dispatched.	VERIFYING	Actions taken correspond to preconditioning activities within the E40 Process Job including loading the first wafer. Check all alignment and set-up conditions.	A part of verification may be done by human operator. Some prompt, usually sound and/or light, to instruct human verification may be expected when such operation is required.
20	VERIFYING	All required alignment for the first wafer of the PJ is completed including verification of needle-pad alignment for the first dice.	SORTING	Start automated process.	
21	SORTING	Such processing condition as recipe parameter is updated.	VERIFYING	Check all alignment and set-up conditions.	An optional transition. Same as transition 19.
22	SORTING	The PJ is completed and all materials of the PJ is restored in carriers.	IDLE		There are no active process jobs at this time.

SEMI E130.1-1104
SPECIFICATION FOR SECS-II PROTOCOL FOR PROBER SPECIFIC EQUIPMENT MODEL FOR 300 mm ENVIRONMENT (PSEM300)

This specification was technically approved by the Global Information & Control Committee and is the direct responsibility of the North American Information & Control Committee. Current edition approved by the North American Regional Standards Committee on July 11, 2004. Initially available at www.semi.org September 2004; to be published November 2004.

1 Purpose

1.1 This document maps the services and data of SEMI E130 to SECS-II streams and functions, and data definitions.

2 Scope

2.1 This is a specification covering equipment supporting automated communication of the prober equipment.

2.2 This document applies to all implementations of SEMI E130 that use the SECS-II message protocol (SEMI E5). Compliance to this standard requires compliance to both SEMI E130 and SEMI E5.

NOTICE: This standard does not purport to address safety issues, if any, associated with its use. It is the responsibility of the users of this standard to establish appropriate safety and health practices and determine the applicability of regulatory or other limitations prior to use.

3 Limitations

3.1 This specification applies to semiconductor equipment that also use SEMI E30 GEM standard.

3.2 PSEM300-compliant equipment must support High Speed Messaging Service (SEMI E37) communication standard for sending SEMI E5 messages over TCP/IP.

4 Referenced Standards

4.1 *SEMI Standards*

SEMI E5 — SEMI Equipment Communications Standard 2 Message Content (SECS-II)

SEMI E30 — Generic Model for Communications and Control of Manufacturing Equipment (GEM)

SEMI E37 — High Speed SECS Message Services (HSMS) Generic Services

SEMI E123 — Handler Specific Equipment Model (HSEM)

NOTICE: Unless otherwise indicated, all documents cited shall be the latest published versions.

5 Service Message Mapping

5.1 This section shows the specific SECS-II streams and functions that shall be used for SECS-II implementation of the services or remote commands defined in SEMI E130, as well as the parameter mapping for data attached to services.

5.2 *Services Message Mapping*

5.2.1 Table 1 defines the relationships between SEMI E130 services and SECS-II messages. Mapping of service parameters to the SECS-II data items is provided in a separate table. Conventions and definitions of table fields are as described below.

5.2.2 *Service Name* — Name of the service or remote command defined in SEMI E130.

5.2.3 *Stream, Function* — Specifies the SECS-II stream and function (SxFx) mapped to the service messages. Following convention of SECS-II, request and notification messages are mapped to the odd-numbered functions and response or acknowledgement messages are mapped to the corresponding even-numbered functions.

5.2.4 *SECS-II Message Name* — Name of the SECS-II message.

Table 1 Services Message Mapping Table

Service Name	*Stream, Function*	*SECS-II Message Name*
CLEAN-PROBES	S2,F49/F50	Enhanced Remote Command Send/Acknowledge
INDEX	S2,F49/F50	Enhanced Remote Command Send/Acknowledge
INK-DEVICE	S2,F49/F50	Enhanced Remote Command Send/Acknowledge
INK-DIE	S2,F49/F50	Enhanced Remote Command Send/Acknowledge
INSPECT-INK-MARKS	S2,F49/F50	Enhanced Remote Command Send/Acknowledge

Service Name	*Stream, Function*	*SECS-II Message Name*
INSPECT-PROBES	S2,F49/F50	Enhanced Remote Command Send/Acknowledge
INSPECT- PTP-ALIGN	S2,F49/F50	Enhanced Remote Command Send/Acknowledge
LOCKOUT-CONTROL	S2,F49/F50	Enhanced Remote Command Send/Acknowledge
LOAD- TEST-SUBST	S2,F49/F50	Enhanced Remote Command Send/Acknowledge
LOAD-SUBSTRATE	S2,F49/F50	Enhanced Remote Command Send/Acknowledge
MOVE	S2,F49/F50	Enhanced Remote Command Send/Acknowledge
MOVE-ABS	S2,F49/F50	Enhanced Remote Command Send/Acknowledge
MOVE-SUB-STEPS	S2,F49/F50	Enhanced Remote Command Send/Acknowledge
MOVE-TO-REF-DIE	S2,F49/F50	Enhanced Remote Command Send/Acknowledge
UNLOAD-TEST-SUBST	S2,F49/F50	Enhanced Remote Command Send/Acknowledge
RESET-CONTACT-CNT	S2,F49/F50	Enhanced Remote Command Send/Acknowledge
RESET-CONTROL-MAP	S2,F49/F50	Enhanced Remote Command Send/Acknowledge
SET-CONTROL	S2,F49/F50	Enhanced Remote Command Send/Acknowledge
SET-PROBE-CARD-ID	S2,F49/F50	Enhanced Remote Command Send/Acknowledge
SET-CONTROL-MAP	S2,F49/F50	Enhanced Remote Command Send/Acknowledge
UNLOAD-SUBSTRATE	S2,F49/F50	Enhanced Remote Command Send/Acknowledge
UNLOAD-ALLSUBSTRATES	S2,F49/F50	Enhanced Remote Command Send/Acknowledge
ONLINE-LOCAL	S2,F49/F50	Enhanced Remote Command Send/Acknowledge

5.3 *Event Message Mapping*

5.3.1 Table 2 defines the relationships between SEMI E130 collection events and SECS-II messages. Conventions and table field definitions similar to mapping table for the services, here Event Name specifies the event defined in SEMI E130. This specification follows SEMI E30 Dynamic Report Configuration to define and attach reports to collection events.

Table 2 Event Message Mapping Table

Event Name	*Stream, Function*	*SECS-II Message Name*
All events defined in SEMI E130.	S6F11/12	Event Report Send/Acknowledge

5.4 *Service Parameter Mapping*

5.4.1 Table 3 defines the relationships between SEMI E130 service parameters and SECS-II data definitions or parameter fields in the mapped streams and functions. Parameters for acknowledgements or responses will follow SECS-II specification for that stream and function, no service specific error codes are defined in this specification. Descriptions of each table column are described below.

5.4.2 *Service* — Specifies the service whose parameters are described.

5.4.3 *SECS-II Field* — Specifies the SECS-II data item or message parameter used by the service. In this specification, the DATAID and OBJSPEC fields are unspecified for services using the S2F49 SECS-II message. Their values are to be ignored or used for implementation specific purposes.

5.4.4 *Values* — Value or SECS II format for the specified field. Formats are specified using the SML notation as defined in appendices of SEMI E30 (GEM).

5.4.5 *Req.* — Indicates whether the specified field or parameter is required or not. If an optional CPNAME parameter is used, the corresponding CPVAL must also be given.

5.4.6 *Description* — Provides a brief description of the SECS-II field in relation to the service parameters defined in SEMI E130.

Table 3 Service Parameter to SECS-II Data Items Mapping

Service	*SECS-II Field*	*Values*	*Req.*	*Description*
CLEAN-PROBES	RCMD	"CLEAN-PROBES"	Y	
	CPNAME	"CLEANPROCESS"	N	Clean Process parameter.
	CPVAL	A [n]		Specifies name of the cleaning process to be performed.
	CPNAME	"DIELIST"	Y	Die List parameter.
	CPVAL	L, n 1. L, 2 1. U4 2. U4 : n. L, 2 1. U4 2. U4		List of probe sites to be cleaned. Zero-length list indicates that all sites should be cleaned. Each site is represented by a die coordinate structure specifying the X and Y coordinate: L, n (n = number of sites selected) 1. L, 2 1. X Coordinate 2. Y Coordinate … n. L, 2 1. X Coordinate 2. Y Coordinate
INDEX	RCMD	"INDEX"	Y	
INK-DEVICE	RCMD	"INK-DEVICE"	Y	
INK-DIE	RCMD	"INK-DIE"	Y	
	CPNAME	"DIELIST"	N	Die List parameter.
	CPVAL	L, n 1. L, 2 1. U4 2. U4 : n. L, 2 1. U4 2. U4		List of die sites to be inked. "DIELIST" structure as defined above. Either Die List or Control Map must be supplied.
	CPNAME	"CONTROLMAP"	N	Control Map parameter.
	CPVAL	A [n]		Control Map identifier.
INSPECT-INK-MARKS	RMCD	"INSPECT-INK-MARKS"	Y	
	CPNAME	"DIELIST"	N	Die List parameter.
	CPVAL	L, n 1. L, 2 1. U4 2. U4 : n. L, 2 1. U4 2. U4		List of sites to inspect. Current site is assumed if not provided.
INSPECT-PROBES	RCMD	"INSPECT-PROBES"	Y	
	CPNAME	"PROBESITELIST"	N	Probe Site List parameter.
	CPVAL	Supplier Defined Format		Supplier defined list of probe sites to inspect. Zero-length list indicates all sites.
INSPECT- PTP-ALIGN	RCMD	"INSPECT- PTP-ALIGN"	Y	

Service	*SECS-II Field*	*Values*	*Req.*	*Description*
LOAD- TEST-SUBST	RCMD	“LOAD- TEST-SUBST”	Y	
	CPNAME	“SUBSTLOCID”	Y	SubstLocID parameter.
	CPVAL	A [n]		Substrate Location containing the test substrate.
LOAD-SUBSTRATE	RCMD	“LOAD-SUBSTRATE”	Y	
LOCKOUT-CONTROL	RCMD	“LOCKOUT-CONTROL”	Y	
	CPNAME	“LOCKOUT”	N	Lockout parameter.
	CPVAL	BOOLEAN		True = Lockout console control.
MOVE	RCMD	“MOVE”	Y	At least one of X, Y, or Z parameter must be supplied. An omitted X, Y, or Z is equivalent to specifying zero.
	CPNAME	“X”	N	X parameter.
	CPVAL	U4		Relative step value in X direction.
	CPNAME	“Y”	N	Y parameter.
	CPVAL	U4		Relative step value in Y direction.
	CPNAME	“Z”	N	Z parameter.
	CPVAL	U4		Relative step value in Z direction.
MOVE-ABS	RCMD	“MOVE-ABS”	Y	At least one of X, Y, or Z parameter must be supplied. An omitted X, Y, or Z is equivalent to specifying current value.
	CPNAME	“X”	N	X parameter.
	CPVAL	U4		Absolute index in X direction.
	CPNAME	“Y”	N	Y parameter.
	CPVAL	U4		Absolute index in Y direction.
	CPNAME	“Z”	N	Z parameter.
	CPVAL	U4		Absolute index in Z direction.
MOVE-SUB-STEPS	RCMD	“MOVE-SUB-STEPS”	Y	At least one of XSubstep, YSubstep or Zsubstep must be supplied. An omitted XSubstep, YSubstep, or ZSubstep is equivalent to specifying zero.
	CPNAME	“XSUBSTEP”	N	XSubstep parameter.
	CPVAL	F4		Relative substep value in X direction.
	CPNAME	“YSUBSTEP”	N	YSubstep parameter.
	CPVAL	F4		Relative substep value in Y direction.
	CPNAME	“ZSUBSTEP”	N	ZSubstep parameter.
	CPVAL	F4		Relative substep value in Z direction.
MOVE-TO-REF-DIE	RCMD	“MOVE-TO-REF-DIE”	Y	
RESET-CONTACT-CNT	RCMD	“RESET-CONTACT-CNT”	Y	
RESET-CONTROL-MAP	RCMD	“RESET-CONTROL-MAP”	Y	If coordinates are designated, both X and Y coordinate must be provided.
	CPNAME	“X”	N	X parameter.
	CPVAL	U4		X coordinate value.
	CPNAME	“Y”	N	Y parameter.
	CPVAL	U4		Y coordinate value.
SET-PROBE-CARD-ID	RCMD	“SET-PROBE-CARD-ID”	Y	
	CPNAME	“PROBECARDID”	Y	Probe Card ID parameter.
	CPVAL	A [n]		Probe card identifier.

Service	*SECS-II Field*	*Values*	*Req.*	*Description*
SET-CONTROL-MAP	RCMD	"SET-CONTROL-MAP"	Y	
	CPNAME	"CONTROLMAP"	Y	ControlMap parameter.
	CPVAL	A [n]		Control Map identifier.
	CPNAME	"X"	N	X parameter.
	CPVAL	U4		X coordinate value.
	CPNAME	"Y"	N	Y parameter.
	CPVAL	U4		Y coordinate value.
UNLOAD-SUBSTRATE	RCMD	"UNLOAD-SUBSTRATE"	Y	
UNLOAD-TEST-SUBST	RCMD	"UNLOAD-TEST-SUBST"	Y	
	CPNAME	"SUBSTLOCID"	Y	SubstLocID parameter.
	CPVAL	A [n]		Target substrate location.
UNLOAD-ALLSUBSTRATES	RCMD	"UNLOAD-ALLSUBSTRATES"	Y	
	CPNAME	"SUSPEND"	N	SUSPEND parameter.
	CPVAL	BOOLEAN		If TRUE, causes substrate occupying chuck to be suspended.
ONLINE-LOCAL	RCMD	"ONLINE-LOCAL"	Y	

6 Variable Data Item Mapping

6.1 The purpose of this section is to define the list of variable items required by SEMI E130. Values of these variables will be available to the host through collection event reports and host status queries.

6.2 *Requirements*

6.2.1 Any supplier-defined variables shall be documented in the same format used by this document. The following minimum information is required:

<variable name> **Class:** <ECV, SV, or DVVAL> **Format:** <SML>

Description: <If class = DVVAL, description must contain statement of when data is valid.>

<If format = ASCII, then a length is required. It is assumed to be left-justified unless otherwise noted.>

6.3 *Data Types*

6.3.1 Equipment Constants (ECVs) can be changed by the host using S2,F15 (New Equipment Constant Send). The operator may be able to change some values, but the equipment does not change the values on its own. The value of an equipment constant may be queried by the host at any time, using the S2,F13/14 transaction. They reside in non-volatile memory of the equipment. Equipment constants remain in effect until they are overwritten either by manual entry or by a S2,F15.

6.3.2 Status Variables (SVs) are valid at all times. An SV may not be changed by the host but may be changed by the equipment or operator. The value of status variables may be queried by the host at anytime using the S1,F3/4 or S6,F19/20 transactions.

6.3.3 DVVALs are variables that are valid only upon the occurrence of specific collection events. An attempt to read a variable item at the wrong time does not generate an error, but the data reported may not have relevant meaning.

6.4 For multiple sites scenarios, if a variable is described as List by number of sites, then a list is expected, one value for each site.

Table 4 Variable Item Mapping Table

Variable Name	*Description*	*Class*	*SECS-II Type*
ContactCount	The number of times the probe card has come in contact with a substrate.	SV	U2
DieXCoordinate	The coordinate of the die in the x direction (column).	SV	I2
DieYCoordinate	The coordinate of the die in the y direction (row).	SV	I2
DieZPosition	The position of the Chuck Z-axis in microns. A user/supplier defined unit of measure other than microns is permissible and must be documented.	SV	F4
DieZCoordinate	The coordinate of the die in the z direction.	SV	I2
InspectionResult	Result of the last inspection.	SV	BOOLEAN
MinXStep	The minimum step size in microns in the X direction.	SV	F4
MinYStep	The minimum step size in microns in the Y direction.	SV	F4
MinZStep	The minimum step size in microns in the Z direction (Alternative user-defined unit of measure may be specified for the Z motion).	SV	F4
ProbeCardId	The ID of the Probe Card.	SV	A [25]
TaskID	The ID of the service task being performed.	SV	A [50]

SEMI E132-0305
SPECIFICATION FOR EQUIPMENT CLIENT AUTHENTICATION AND AUTHORIZATION

This specification was technically approved by the Global Information & Control Committee and is the direct responsibility of the North American Information & Control Committee. Current edition approved by the North American Regional Standards Committee on July 11, 2004 and August 16, 2004. Initially available at www.semi.org September 2004; to be published November 2004. Originally published March 2004; previously published July 2004.

NOTICE: The designation of SEMI E132 was updated during the 0305 publishing cycle to reflect the creation of SEMI E132.1.

1 Purpose

1.1 This specification describes a method for restricting access to communication with equipment by requiring clients to authenticate to the equipment before proceeding with subsequent communication, and provides a flexible authorization scheme to control client application access. Authorization allows parties who want to exchange information with or control the equipment to do so on a need-to-know and/or need-to-use basis.

1.2 The authorization scheme specified in this standard allows equipment vendor the flexibility to provide access control at any level of granularity, ranging from no access control restrictions, predefined role-based access control, to very fine-grained control.

2 Scope

2.1 This is a standard that applies to all semiconductor manufacturing equipment that requires authentication and authorization for its services. It does not apply to communication that is governed by the SEMI E30 communication and control state models.

2.2 This standard does not require data transmitted over an established session to be encrypted, encryption is only required as specified by the authentication protocol. It is assumed that the interface specified by this standard will be operating in an environment where there are no malicious attacks such as inside a closed factory network.

NOTICE: This standard does not purport to address safety issues, if any, associated with its use. It is the responsibility of the users of this standard to establish appropriate safety and health practices and determine the applicability of regulatory or other limitations prior to use.

3 Limitations

3.1 An adjunct standard will need to be defined that maps specific technologies and protocols to the defined concepts in this document.

4 Referenced Standards

4.1 *SEMI Standards*

SEMI E30 — Generic Model for Communications and Control of SEMI Equipment (GEM)

4.2 *Other Standards*

Uniform Resource Identifiers (URI): Generic Syntax, IETF RFC 2396, August 1998 (**http://www.ietf.org/rfc/rfc2396.txt**)

NOTICE: Unless otherwise indicated, all documents cited shall be the latest published versions.

5 Referenced Documents

5.1 Unified Modeling Language (UML) Specification, Version 1.4, OMG Specification 01-09-67, **http://www.omg.org/technology/documents/modeling_spec_catalog.htm**

6 Terminology

6.1 *Definitions*

6.1.1 *administration* — the process of configuring equipment client identities, associated credentials, and setting authorizations.

6.1.2 *Asymmetric Key Cryptography* — see *Public Key Cryptography.*

6.1.3 *authenticated communications* — communications with equipment compliant to this specification to establish an authenticated session, to grant authorizations and for administration. This includes any communications over the authenticated session. Note that authenticated communications are not necessarily encrypted.

6.1.4 *authenticated session* — also referred to as *session* in this document, this is an application-level concept defined by this specification and is independent of any physical or logical connection or communication protocol "session" layer used in the actual exchange of messages. An *authenticated session* or *session* follows the session establishment procedure, state models, and the defined services as described in this specification.

6.1.5 *authentication* — the process of determining whether a user or process is, in fact, who or what it is declared to be. In private and public computer networks, authentication is often achieved through the use of passwords. Knowledge of the password is assumed to guarantee that the user is authentic. This standard refers only to the authentication of application processes, and does not support or address authentication of human users.

6.1.6 *authorization* — the process of granting the privilege to perform a specific action to a user or process. Note that the user or process must be authenticated before authorization may take place.

6.1.7 *client* — an application process that communicates with the equipment to access equipment resources. This specification uses the terms "client" and "client application" interchangeably.

6.1.8 *communication session* — a series of two-way message exchanges between a client and the equipment. A session typically begins and ends at the request of the client.

6.1.9 *cryptographic algorithm* — a transformation of meaningful data into some meaningless data, a process called encryption. If the transformation is reversible, then the reverse process is called decryption. Typically it is computationally difficult to correctly decrypt the data without knowing both the algorithm used and some additional information, referred to as keys.

6.1.10 *encryption* — see *cryptographic algorithm.*

6.1.11 *message digest* — see *one-way hash.*

6.1.12 *one-way hash* — a one-way transformation of an arbitrary length of data into a fixed-length code. The transformation is computationally difficult to reverse, and unlikely to have collisions. The same message will always result in the same hash, and any slight modification to the original message will result in a different hash, thus providing message integrity. The terms One-way Hash, Hash and Digest are used interchangeably in this document.

6.1.13 *Persistent Session* — an *Authenticated Session* established that survives equipment shutdowns or power offs, it is set to persistent by a client through an equipment service defined in this specification. Behavior of a persistent session during equipment shutdown and startup is as defined by the session state model described in this document.

6.1.14 *Public Key Cryptography (PKC)* — also known as Asymmetric Key Cryptography, a cryptographic algorithm that employs two keys for encryption and decryption of data. One key is designated as the public key and can be freely advertised, while the other key, designated as the private key, is never revealed. Data encrypted with one key can only be decrypted with the other key. PKC is widely used for key exchange and message signing.

6.1.15 *Secret Key Cryptography (SKC)* — also known as Symmetric Key Cryptography, a cryptographic algorithm that employs one single key for both encryption and decryption. SKC is widely used for bulk encryption of data due to its speed over PKC.

6.1.16 *Session* — in this document, the term "session" refers to an *Authenticated Session* unless specified otherwise. It has no relation to the sessions that may be defined elsewhere.

6.1.17 *Symmetric Key Cryptography* — see *Secret Key Cryptography.*

7 Conventions

7.1 *Unified Modeling Language (UML) Notation* — All class and sequence diagrams in this standard make use of the Unified Modeling Language notation.

7.2 *Attribute Tables* — The table below provides an example of the tables used to list and describe attributes of classes defined in this specification.

Table 1 Attribute Table Format

Attribute Name	*Definition*	*Form*
		See list below.

7.2.1 *Form* — Defines the data type of the attribute. The terms used to describe data types used in this column are defined in the SEMI Compilation of Terms, or are included as part of the specification. Refer to the compilation of terms for the definition of SEMI type name meanings.

7.2.2 *Association Tables* — The table below provides an example of the tables used to list and describe associations between classes defined in this specification.

Table 2 Association Table Format

Association Role Name	*Definition*	*Comments*

7.2.2.1 *Association Role Name* — The name of the association role being specified.

7.2.2.2 *Definition* — Describes the function or purpose of the association.

7.2.2.3 *Comments* — Any additional comments or notes regarding the association.

7.3 *Operation Definition Tables* — The table below provides an example of the tables used to list and describe the interface operations defined in this specification.

Table 3 Operation Definition

Operation	*Description*	*Type*	*Requestor/Sender*	*Responder/Receiver*
		See list below.		

7.3.1 *Operation* — Specifies the name of the operation.

7.3.2 *Type* — Specifies the messaging semantics of the operation. Only Request-Response (RR) and Fire-and-Forget (FF) semantics are used in this specification.

7.3.3 *Requestor/Sender* — For RR semantics, identifies the entity that makes the request, for FF semantics, identifies the entity that sends the message. Can be ‘Client’, ‘Admin’, or ‘Equipment’. Client can be any of the applications communicating to equipment compliant to this specification.

7.3.4 *Responder/Receiver* — For RR semantics, identifies the entity that responds to the request, for FF semantics, identifies the entity that receives the message. Can be ‘Client’, ‘Admin’, or ‘Equipment’.

7.4 *Operation Argument Definition Table* — The table below provides an example of the tables used to list and describe arguments for interface operations defined in this specification.

Table 4 Operation Argument Definitions

Argument	*Description*	*Kind*	*Form*
		See list below.	**See list below.**

7.4.1 *Argument* — Specifies the name of the argument.