# SEMI E127-0308
# SPECIFICATION FOR INTEGRATED MEASUREMENT MODULE COMMUNICATIONS: CONCEPTS, BEHAVIOR, AND SERVICES (IMMC)

This standard was technically approved by the global Information & Control Committee. This edition was approved for publication by the global Audits and Reviews Subcommittee on December 20, 2007. It was available at www.semi.org in February 2008 and on CD-ROM in March 2008. Originally published July 2003; previously published November 2007.

INACTIVE

## Table of Contents

1 Purpose ....4
1.1 *Integrated Measurement* ....4
1.2 *Specification for the Integrated Measurement Module Communications (IMMC)* ....4
2 Scope ....4
2.1 *Scope of This Document* ....4
3 Limitations ....4
3.1 *Host/Integrated Equipment Interface* ....4
3.2 *Object-Based Implementation* ....4
3.3 *In-Situ Processing* ....4
4 Referenced Standards and Documents ....4
4.1 *SEMI Standards* ....4
4.2 *Other Sources* ....5
5 Terminology ....5
5.1 *Abbreviations and Acronyms* ....5
5.2 *Definitions* ....5
6 Conventions ....6
6.1 *Object Conventions* ....6
6.2 *State Model Conventions* ....7
6.3 *Service Message Representation* ....7
7 General Requirements ....8
7.1 *Protocol Requirements* ....8
7.2 *Required Standards* ....8
7.3 *Required Basic Capabilities* ....8
7.4 *State Models* ....9
7.5 *Objects* ....9
7.6 *Service Requirements* ....10
8 Overview ....10
8.1 *Background* ....10
9 Communication Ports and Clients ....10
9.1 *Ports* ....10
9.2 *Clients* ....10
9.3 *Bandwidth and Performance* ....11
10 Integrated Measurement Module Object Model ....11
10.1 *Description* ....11
11 The Integrated Measurement Module Object ....12
11.1 *Relationship of IMM Object And Physical Module* ....12
11.2 *IMM Service State Model* ....12
11.3 *Object Attributes* ....14
11.4 *System Initialization* ....16
11.5 *IMM Events and Alarms* ....17
11.6 *IMM Object Services* ....18
12 Substrate Transfer Path Object ....18
12.1 *Description* ....18



Documents Document Sharing Hub
Full document details: SEMI E127-0308

12.2 *Object State Model* ....19
12.3 *Object Attributes* ....20
12.4 *Object Services* ....21
12.5 *Creation of Substrate Objects* ....22
12.6 *Substate Verification* ....23
12.7 *Substrate Orientation* ....23
12.8 *Transfer Path Calibration* ....24
12.9 *Substrate Transfer Path Events, Errors, and Alarms* ....25
12.10 *Substrate Transfer Path Event Variables* ....25
12.11 *Substrate Transfer Path Scenarios* ....26
13 IMM Data ....32
13.1 *Overview* ....32
13.2 *Tables* ....33
14 IMM Data Table Object ....33
14.1 *Description* ....33
14.2 *IMM Data Table Object State Model* ....33
14.3 *IMM Data Table Object Attributes* ....35
14.4 *IMM Data Table Object Event Variables* ....37
14.5 *Types of IMM Data Table Objects* ....38
14.6 *IMM Data Table Object Services* ....40
15 Data Table Objects ....40
15.1 *Data Table Object Model* ....40
15.2 *Data Row Template Object* ....42
15.3 *Master Data Table Objects* ....43
16 Requirements for Objects Defined in Other Standards ....44
16.1 *Substrate Object* ....44
16.2 *Process Job Management* ....44
17 Event and Exception Management ....44
18 Message Service Definitions ....44
18.1 *List of Services* ....44
18.2 *Service Parameters* ....45
18.3 *Message Detail* ....47
19 Services Provided by Client ....50
19.1 *List of Client Services* ....50
20 IMMC Compliance ....50
20.1 *Overview of Requirements* ....50
21 Related Documents ....51
21.1 *List of Documents* ....51

## List of Tables

Table 1 Service State Model Transitions ....13
Table 2 IMM Object Attribute Definitions ....14
Table 3 Retention Rule Element Definitions ....15
Table 4 Event Variables for IMM Object ....17
Table 5 IMM Object Services ....18
Table 6 Substrate Transfer Path State Model Transitions ....20
Table 7 Substrate Transfer Path Object Attribute Definitions ....21
Table 8 List of Substrate Transfer Path Services ....22
Table 9 Event Variables for Substrate Transfer Path Object ....25
Table 10 Substrate Transfer Path Load Scenario ....26
Table 11 Substrate Transfer Path Unload Scenario ....29
Table 12 IMM Data Table Object Transitions ....34
Table 13 IMM Data Table Object Attribute Definitions ....35
Table 14 Attribute Elements for IMM Data Table Object ....36
Table 15 Event Variables for IMM Data Table Objects ....38

INACTIVE



Documents
Document Sharing Hub
Full document details: SEMI E127-0308

Table 16 IMM Raw Data Table Object Attribute Definitions ....39
Table 17 IMM Converted Data Table Object Attribute Definitions ....39
Table 18 Event Variables for Conversion Data Table Object ....39
Table 19 IMM Calibration Data Table Object Attribute Definitions ....40
Table 20 Data Table Object Attribute Definitions ....41
Table 21 Data Row Attribute Definitions ....41
Table 22 Data Row Template Attribute Definitions ....42
Table 23 Data Information Attribute Definitions ....42
Table 24 Assurance Element Definitions ....43
Table 25 Additional Substrate Attribute Definitions ....44
Table 26 List of Services ....45
Table 27 Definitions for Parameters and Parameter Elements ....45
Table 28 Definitions for RetentionRule Parameter Elements ....47
Table 29 Valid ErrorCode Enumerations ....47
Table 30 ClientConnect Service Parameters ....48
Table 31 ClientDisconnect Service Parameters ....48
Table 32 RequestRetentionConditions Service Parameters ....48
Table 33 Parameters for ChangeService ....48
Table 34 SetAutoTableSend Service Parameters ....48
Table 35 SetRetentionConditions Service Parameters ....49
Table 36 TransferPathCalibration Service Parameters ....49
Table 37 XfrCmdListRequest Service Parameters ....49
Table 38 XfrCommand Service Parameters ....50
Table 39 XfrHalt Service Parameters ....50
Table 40 XfrVerify Service Parameters ....50
Table 41 List of Client Services ....50
Table 42 IMMC Compliance ....51

INACTIVE



Documents
Document Sharing Hub
Full document details: SEMI E127-0308

## 1 Purpose

1.1 *Integrated Measurement*

1.1.1 The purpose of integrated measurement (metrology or inspection) is to facilitate intra-equipment process monitoring through rapid access to measurement data, reduce material handling between process and measurement equipment, and the opportunity to increase process monitoring with minimal or no decrease in throughput. The benefits of integrated measurement also allow Advanced Process Control systems to use the results with reduced feedback lag time.

1.2 *Specification for the Integrated Measurement Module Communications (IMMC)*

1.2.1 The purpose of the IMMC specification is to provide an object-based specification of an Integrated Measurement Module together with a standard interface between an integrated measurement module and its control and data ports where these are commonly implemented by different suppliers. The interface allows access to the properties and services of specific objects. This will facilitate the effort needed for the integration of the module into a larger system.

1.2.2 An additional purpose of this standard is to provide sufficient information through a combination of on-line services and interface documentation that an IMMC-compliant integrated measurement module may be integrated with multi-module equipment without requiring a software change in either the module or the equipment. This may require configuration changes made by the end-user through the equipment user interface where certain options are left to the module supplier.

## 2 Scope

2.1 *Scope of This Document*

2.1.1 The Integrated Measurement Module Communications specification covers concepts, behaviors, and services to be provided by a metrology or inspection module so that it may be integrated into production equipment intended for large substrates processing. However, nothing should preclude its application to smaller substrate manufacturing.

**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 *Host/Integrated Equipment Interface*

3.1.1 This specification does not apply to the communications interface between the host and integrated equipment.

3.2 *Object-Based Implementation*

3.2.1 Compliance to this standard does not require object-oriented implementation. However, it does require the appearance of implentation of those objects defined in this document.

3.3 *In-Situ Processing*

3.3.1 This specification applies only to measurement components that are contained within a module that provides its own material handling and does not measure during a process step. Interactions with in-situ measurement components are expected to differ significantly from the current specification.

## 4 Referenced Standards and Documents

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 E30.5 — Specification for Metrology Specific Equipment Model (MSEM)

SEMI E32 — Material Movement Management (MMM)

INACTIVE



Documents
Document Sharing Hub
Full document details: SEMI E127-0308

SEMI E39 — Object Services Standard: Concepts, Behavior, and Services

SEMI E40 — Standard for Processing Management

SEMI E41 — Exception Management (EM) Standard

SEMI E42 — Recipe Management Standard: Concepts, Behavior, and Message Services

SEMI E53 — Event Reporting

SEMI E58 — Automated Reliability, Availability, and Maintainability Standard (ARAMS): Concepts, Behavior, and Services

SEMI E90 — Specification for Substrate Tracking

SEMI E98 — Provisional Standard for the Object-Based Equipment Model (OBEM)

SEMI E116 —Specification for Equipment Performance Tracking

SEMI E120 —Specification for the Common Equipment Model (CEM)

SEMI E148 — Specification for Time Synchronization and Definition of the Clock Object

SEMI M20 — Practice for Establishing a Wafer Coordinate System

4.2 *Other Documents*

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

uuid: ISO/IEC 11578:1996 Information technology – Open Systems Interconnection – Remote Procedure Call (RPC); http://www.iso.ch/cate/d2229.htm

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

## 5 Terminology

NOTE 1: Terms defined in § 5 that are used in other definitions have been underlined.

5.1 *Abbreviations and Acronyms*

5.1.1 *IMM* — Integrated Measurement Module

5.1.2 *IMMC* — Specification for Integrated Measurement Module Communications: Concepts, Behavior, and Services

5.1.3 *IMMDTOSM* — IMM Data Table Object State Model

5.1.4 *SLOSM* — Substrate Location Object State Model [SEMI E90]

5.1.5 *STPO* — Substrate Transfer Path Object, an abstraction of the IMM capability to load and unload substrates into the IMM.

5.1.6 *STPOSM* — Substrate Transfer Path Object State Model

5.2 *Definitions*

5.2.1 *array* — an ordered list of numeric values. A valid data type for table row entries.

5.2.2 *conversion recipe* — a recipe or portion of a recipe that describes the steps required for taking measurement data and reporting a result. This does not need to be a physically separate recipe.

5.2.3 *error message* — a notification to the user/client that an error has occurred. It may or may not be associated with an alarm. Note: depending upon the communication protocol, error messages may or may not be treated differently from normal collection events.

5.2.4 *inspection* — an examination of an area of material to detect anomalies.

INACTIVE



Documents Document Sharing Hub

Full document details: SEMI E127-0308

5.2.5 *inspection module* — a measurement module that inspects substrates and reports information regarding anomalies. Inspection modules may determine the location of anomalies relative to a coordinate system and may also provide other types of data related to the anomaly.

5.2.6 *integrated measurement module* — a measurement module intended to be integrated into manufacturing equipment, and with the capability of receiving substrates from the equipment, measuring those substrates, and returning the substrates and the measurement results to the equipment and other concerned clients.

5.2.7 *measurement module* — an equipment module whose intended function is to measure or inspect the product and to report the results. Measurement of the product is the factory's means of gaining feedback on the manufacturing process.

5.2.8 *measurement recipe* — a recipe or portion of a recipe intended for use during a measurement, that describes among other things the locations for measurement. This does not need to be a physically separate recipe.

5.2.9 *metrology module* — a measurement module that collects and reports information on specific predetermined locations or features on a substrate with consistent data structure, or reports general information about the entire substrate.

5.2.10 *object-based* — a programming language, or database, is called object-based if it supports the concept of data abstraction, but partly or entirely lacks more advanced concepts such as class, inheritance, polymorphism, and so on. [Oestereich, Bernd, "Developing Software with UML," Addison-Wesley (1999)]

5.2.11 *substrate context information* — information concerning the substrate that may be useful to for analysis, such as process flow step, substrate orientation, the identifier of the process equipment/chamber most likely to have affected results, the recipe run on that equipment/chamber, etc.

5.2.12 *substrate orientation* — the angle of rotation from normal. For wafers, this is the angle of rotation from the primary fiducial.

## 6 Conventions

6.1 *Object Conventions*

6.1.1 This document conforms to the conventions for objects established by SEMI E39, including object diagrams, object terminology, and requirements for standardized objects. However, the notation used for object diagrams is the Universal Modeling Language (UML) notation (see ¶ 4.2 for additional detail).

6.1.2 *Formal Name of an Object* — The text capitalizes formal object name references, similar to the way capitalization is normally used when discussing entities. When describing something in the general (like cities) lower case is used, but when a specific entity is of interest (New York City), then first letters are capitalized. Where words are concatenated, they retain their capitalization to enhance readability.

6.1.3 *Object Attributes* — Attribute tables define those public attributes that can be read and set through the basic OSS services GetAttr and SetAttr. Simple attributes have a single data type as their value. Complex or compound attributes are made up of an ordered set of other elements, either a list or a structure. See SEMI Compilation of Terms for the definition of "form" for both the list of valid data type formats and for the different types of formats themselves.

6.1.4 *Attribute Definitions* — Attributes are formally defined in an attribute definition table with the following form:

| *Attribute Name* | *Definition* | *Access* | *Reqd* | *Form* |
|---|---|---|---|---|
| The formal name of the attribute. | Defines the requirements of the attribute. | RO or RW | Reqd or Optional | Data type: see SEMI E39, ¶ 4.5. Complex attributes must declare the order of their components. |

6.1.5 *Complex Attribute Data* — The individual data items of Complex attributes are defined in a separate Attribute Definition Table as individual attributes. However, these data items are not Attributes and the SEMI E39 GetAttr and SetAttr services are not valid for them.

INACTIVE



Documents
Document Sharing Hub
Full document details: SEMI E127-0308

6.1.6 *Lists of Data* — Lists of data shall be identified as 0+ when a null list is permitted. Otherwise, they shall be identified as 1+.

6.2 *State Model Conventions*

6.2.1 This document uses the Harel state chart convention for describing dynamic operation of defined objects. The outline of this convention is described in an attachment of SEMI E30. The official definition of this convention is described in "State charts: A Visual Formalism for Complex Systems" written by D. Harel in Science of Computer Programming 8, 1987.

6.2.2 The Harel convention has no concepts of "creation" and "extinction" for expressing the state model of a temporary entity. The "job" described in this document is such an entity, and a copy of the same state model is used for an independent job newly created. In this document, a circle with a black circle inside is used for expressing extinction of an entity. A filled black circle denotes the entry to the state model (the entity creation). This is the convention used in the OMT notation as well.

6.2.3 Transition tables are provided in conjunction with the state diagrams to explicitly describe the nature of each state transition as shown in the state model diagram. A transition table contains columns for Transition number, Previous State, Trigger, New State, Actions, and Comments. The "trigger" (column 3) for the transition occurs while in the "previous" state. The "actions" (column 5) includes a combination of:

1. Actions taken upon exit of the previous state.
2. Actions taken upon entry of the new state.
3. Actions taken which are most closely associated with the transition.

No differentiation is made between these cases.

| *#* | *Previous State* | *Trigger* | *New State* | *Actions* | *Comments* |
|---|---|---|---|---|---|
| | | | | | |

6.2.4 State models are referred to in the Event Variables tables as an acronym consisting of the first letters of the state model name, ending in "SM" for State Model. Specific transitions with the state model are represented with the state model acronym plus "-Tn", where n represents the number of the transition. State model transitions without numbers are not expected to be reported.

6.3 *Service Message Representation*

6.3.1 Services are functions or methods that may be provided by either the equipment or the host. A service message may be either a request message, which always requires a response, or a notification message, which does not require a response.

6.3.2 *Service Definition Table* — A service definition table defines the specific set of messages for a given service resource, as shown in the following table:

| *Message Service Name* | *Type* | *Description* |
|---|---|---|
| | | |

6.3.2.1 Type can be either "N" = Notification or "R" = Request & Response.

6.3.2.2 Notification type messages are initiated by the service provider (e.g., the equipment) and the provider does not expect to get a response from the service user. Request messages are initiated by a service user (e.g., the host). Request messages ask for data or an activity from the provider. Request messages expect a specific response message (no presumption on the message content).

6.3.3 *Service Parameter Dictionary* — A service parameter dictionary table defines the description, format and its possible value for parameters used by services, as shown in the table below. A row is provided in the table for each parameter of a service. See SEMI Compilation of Terms for the definition of "form" for both the list of valid data type formats and for the different types of these formats.

INACTIVE



Documents Document Sharing Hub
Full document details: SEMI E127-0308

| *Parameter Name* | *Description* | *Format: Possible Value* |
|---|---|---|
| | | |

6.3.4 *Service Message Definition* — A service message definition table defines the parameters used in a service, as shown in the following table:

| *Parameter* | *Req/Ind* | *Rsp/Cnf* | *Comment* |
|---|---|---|---|
| | | | |

6.3.4.1 The columns labeled REQ/IND and RSP/CNF link the parameters to the direction of the message. The message sent by the initiator is called the "Request". The receiver terms this message the "Indication" or the request. The receiver may then send a "Response" which the original sender terms the "Confirmation".

6.3.4.2 The following codes appear in the REQ/IND and RSP/CNF columns and are used in the definition of the parameters (e.g., how each parameter is used in each direction):

| | |
|---|---|
| M | Mandatory Parameter — Must be given a valid value. |
| C | Conditional Parameter — May be defined in some circumstances and undefined in others. Whether a value is given may be completely optional or may depend on the value of the other parameter. |
| U | User-Defined Parameter. |
| - | The parameter is not used. |
| = | (For response only) Indicates that the value of this parameter in the response must match that in the primary (if defined). |

## 7 General Requirements

7.1 *Protocol Requirements*

7.1.1 The communication protocol used must support use of all data types, including lists, structures, and arrays.

7.2 *Required Standards*

7.2.1 This section discusses specific SEMI standards that have been assumed for the proper operation of an integrated measurement module (IMM) as specified in this document.

7.2.2 *SEMI E40* — Fundamental compliance to SEMI E40 for Processing Management is required. A process job queue is not required unless the IMM is capable of holding multiple substrates.

7.2.3 *SEMI E90* — Fundamental compliance to SEMI E90 for Substrate Objects and Substrate Location Objects is required.

7.2.4 *SEMI E116* — Fundamental compliance to SEMI E116 (Equipment Performance Tracking) is required. The IMM is to be considered an EPT Module. Guidance for implementing EPT for the IMM is provided in Related Information 2.

7.3 *Required Basic Capabilities*

7.3.1 The requirements in this section are concerned with basic capabilities that may be satisfied in different ways. Because of this flexibility, the method supported must be clearly documented by the IMM supplier. The IMM Control Client should allow on-line configuration by a user to achieve seamless "plug and play" integration.

NOTE 2: New standards are evolving and may become the required method for satisfying these basic capabilities in the future.

7.3.1.1 These capabilities are specified between the IMM and its control client. If, as expected for integrated measurement modules, the client is located within or near the process tool, it is up to the client to have the ability to communicate the appropriate information (alarm reporting, recipe upload/download) to the host.

INACTIVE



Documents
Document Sharing Hub
Full document details: SEMI E127-0308

7.3.2 *Clock Management* — The ability to provide precise time/date information for reporting purposes is required. In addition, standard methods for clock synchronization such as SEMI E148 shall be used and complied to when enabled, otherwise current mechanism as defined in SEMI E30 shall be used.

7.3.3 *Event Management* — The ability for the IMM to report events and associated data of interest to its client is required. Data provided with these events shall be configurable and shall be valid at the time of the event. Each event report shall contain the date and time that the event occurred (and not the time that the report was sent). This requirement may be satisfied through compliance to a standard event mechanism, such as the event reporting capability defined in SEMI E30 or SEMI E53.

7.3.4 *Alarm Management* — The ability to report any change in its alarm conditions is required. An alarm condition is either set or cleared. A set alarm represents danger to the factory, the module, the equipment, the substrates, or the process. This capability may be satisfied through compliance to a standard alarm mechanism such as is defined in SEMI E30 or SEMI E41.

7.3.5 *Exception Management* — An exception is an error. Exceptions may or may not be alarms, either because they do not represent danger or because they represent occurrences that do not have a set or clear state. Exceptions shall be reported and time-stamped at the time the exception occurs. This capability may be satisfied through compliance a standard for exception management such to SEMI E41 or through reporting exceptions as events.

7.3.6 *Recipe Management* — Compliance to this standard requires the support of recipe upload and download and directory functions as specified in one of the SEMI standards defining these methods, such as SEMI E30 for process programs or SEMI E42 recipes.

NOTE 3: New standards for recipe management may be used for these purposes in the future, as they become available.

7.4 *State Models*

7.4.1 *State Model Transitions* — All numbered transitions in all state models defined in this standard represent uniquely identifiable and reportable events unless otherwise noted.

7.4.2 State models in this specification represent behavior as it appears to the client and are used to provide events to the communications partner. Actual state models implemented may differ. For example, a certain state model transition in this document may combine multiple transitions as implemented. It is critical that event representation of state model transitions as specified in the state models defined in this document be followed for reporting.

7.5 *Objects*

7.5.1 *SEMI E39* — Fundamental compliance to SEMI E39 is required for all objects defined in, or required by, this standard. In addition, the Integrated Measurement Module is considered an owner of all objects that it uses, manages, or that are components. If the IMM supports all object types specified in this standard, then it is not necessary to support the GetType service in SEMI E39. If all object types implemented have only those attributes specified by name in this standard, then the service GetAttrName is not required. Filtering for GetAttr and SetAttr is not required. Otherwise, to provide for a self-configuring interface, owner objects shall conform to the following additional services defined in SEMI E39:

- GetType
- GetAttrName
- GetServiceNames
- GetServiceParameters

7.5.2 All objects defined in this standard shall satisfy the fundamental requirements of SEMI E39. It shall be possible for the client to inquire about the attributes of one or more objects through the service GetAttr. It shall be possible for the client to change the value of attributes designated as read/write (RW) using the service SetAttr.

7.5.3 For purposes of communicating using SEMI E39, all objects other than the IMM object itself shall be owned directly by the IMM. The IMM itself is considered to be owned by its Client. For purposes of normal operations during manufacturing, the factory host considers the Control Client as the IMM owner.

INACTIVE



Documents Document Sharing Hub
Full document details: SEMI E127-0308

7.6 *Service Requirements*

7.6.1 This document does not redefine services defined in other SEMI standards. However, for emphasis, those that are beyond the requirements of fundamental compliance to other standards are included in the lists of services as applicable.

## 8 Overview

8.1 *Background*

8.1.1 Both modern manufacturing processes and modern manufacturing equipment are increasingly complex. A single installation of equipment may have hundreds of sensors and actuators. In order to manage this complexity, better methods of referencing the internal components of equipment are needed. Use of the object paradigm provides a means for the equipment to describe its internal composition to the factory in a natural way.

8.1.2 Definition of standardized objects allows the factory to be specific about its requirements and its need for information.

8.1.3 The measurement module is not concerned about the role of its client. Depending on the architectural layout, it might be communicating with the controller of the integrated equipment, the controller of the equipment's front-end carrier handling module or the factory host.

8.1.4 The IMM is one module of many that receive instructions about a specific substrate. The substrate may be given to the IMM at any point in a sequence of individual process steps. While it typically may be the first or the last in the sequence, or both, it may also appear at any point in the middle.

8.1.5 The IMM is given a substrate, a recipe, and a process job. It may or may not have an alignment station, and it may or may not be able to contain more than one substrate at a time. For example, it might be able to align one substrate while it is executing a process job on a previous substrate. Also, the IMM may or may not have an integrated data analysis subsystem. Modules with data analysis capability may require additional processing time to complete the analysis. Finally, the substrate is unloaded and the results of the measurement are sent on the data port.

## 9 Communication Ports and Clients

9.1 *Ports*

9.1.1 The IMM shall provide two or more communication ports, one of which to be used for control as well as information and is called the primary port. Other ports are called secondary ports and are mainly used for sending and receiving information to applications that may be either on- or off-tool, such as remote process control and/or fault detection, archiving substrate data tables, uploading and downloading recipes, etc.

9.1.2 Control in this case signifies the request for services that can change the hardware state of the integrated metrology module, including substrate exchange, process jobs, and maintenance-related activities. In general, the controller of the process tool, or one of its major components, will require the use of the primary port for control, although there may be special circumstances where this is not true.

9.1.3 The IMM shall support one or more additional communication ports to be used mainly for sending and/or receiving information, such as event reports, alarm-related messages, recipes, and data tables. It is recommended that at least one of the additional ports be physically separate to allow a connection to a different network.

9.1.4 The requirement to support a second or multiple ports is a requirement on the IMM side of the communication only. The decision to utilize these ports, and the manner in which they are utilized, is left to the application of the integrated metrology tool and the factory.

9.2 *Clients*

9.2.1 There are two types of clients, the Control Client and the Data Client.

9.2.2 All clients are able to configure the following independently of other clients:

- event reports,
- alarm reports, and
- table retention conditions (¶ 11.3.4).

INACTIVE



Documents
Document Sharing Hub
Full document details: SEMI E127-0308

9.2.2.1 In addition, each client may request these settings at any time and receive only their own settings in response.

9.2.3 The first message that the IMM receives from a potential client must be the ClientConnect Service request. This message is used to establish a client as either the Control Client or a Data Client. All other service requests from an un-established client shall be rejected. At any point in time, there shall be no more than one Control Client.

9.2.4 *The Control Client*

9.2.4.1 The Control Client is a client who intends to request services that may change the physical state of the hardware of the IMM. Such services include those used for substrate exchange, process jobs, and any maintenance activities. In general, this will be the controller of the integrated metrology tool or one of its major components. It is up to the implementer whether clients connecting through a port other than the primary port can establish themselves as the Control Client.

9.2.4.2 The Control Client shall be able to use all IMM capabilities specified in this standard. Only the Control Client shall be permitted to use the IMM Process Job services, or to issue service requests that change the state of the hardware.

9.2.4.3 Once a Control Client is established through use of the ClientConnect service, it remains established as the Control Client until it disconnects by sending the ClientDisconnect Service request.

9.2.4.3.1 During the time that a Control Client is established, any requests to be a Control Client shall be rejected, and any requests for services that can change the state of the hardware from other clients shall be rejected.

9.2.4.3.2 Once the Control Client has disconnected, then another client may request to establish itself as the Control Client.

9.2.5 *Data Clients*

9.2.5.1 The Data Client is a client who intends to request services for information only and not for any services that may change the physical state of the hardware of the IMM. Each Data Client may request information at any time.

9.2.5.2 The use of the Data Client is optional for the end-user.

9.3 *Bandwidth and Performance*

9.3.1 The IMM is responsible for managing its own bandwidth and performance. This standard allows for delays in message transmission in a multi-client environment in order to protect the integrity of the IMM measurement and data conversion abilities.

## 10 Integrated Measurement Module Object Model

10.1 *Description*

10.1.1 The Integrated Measurement Module (IMM) conforms to the architectural specification of the EquipmentModule object as defined in SEMI E120 (CEM).

10.1.2 Figure 1 shows an object model for the Integrated Measurement Module and its significant components.

INACTIVE



Documents Document Sharing Hub
Full document details: SEMI E127-0308

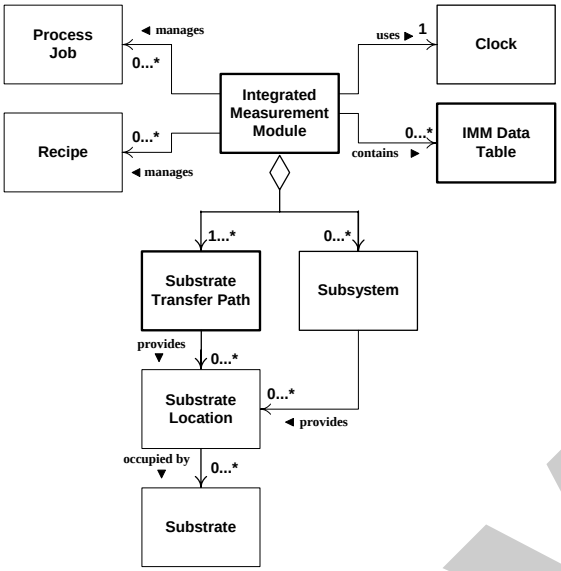


**Figure 1**
**IMM Object Model**

10.1.3 When SEMI E148 is used, the IMM has exactly one internal TS-Clock object. It must have one or more measurement subsystems capable of holding one or more substrates in a substrate location. It may have additional components, such as OCR subsystems. Pattern recognition and analysis subsystem typically do not hold any material. The Clock shall be provided as a SEMI E148 and a SEMI E39 compliant TS-Clock object. Otherwise, current means to set the clock using methods as defined in SEMI E30 shall be used. Subsystems shown in Figure 1 are not required to be visible as objects externally.

## 11 The Integrated Measurement Module Object

11.1 *Relationship of IMM Object and Physical Module*

11.1.1 The IMM object represents the IMM module and all of its internal components to its client. It is responsible for ensuring proper behavior in all cases. In this document, the IMM object is considered synonymous with the IMM module. The IMM provides all services to the client other than those specifically directed to the STPO, including those specified by other SEMI standards.

11.2 *IMM Service State Model*

11.2.1 The IMM is either in a state where it can be used for normal measuring purposes or it is not. This is reflected in the Service State Model as shown in Figure 2.

INACTIVE



Documents
Document Sharing Hub
Full document details: SEMI E127-0308

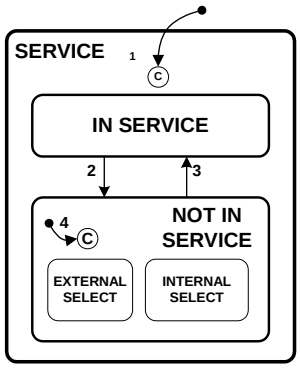


**Figure 2**
**IMM Service State Model**

11.2.2 The IMM Service state model shows its availability for process jobs. It has two states, IN SERVICE and NOT IN SERVICE. The IMM shall only accept or initiate process jobs while in the IN SERVICE state.

11.2.3 *State Definitions*

11.2.3.1 *IN SERVICE* — In this state, the IMM is capable of performing all its normal activities for measurement. It is able to accept and execute a process job or may be currently executing a process job. No faults or exceptions exist that would interfere with its capability of loading, measuring, and unloading a substrate.

11.2.3.2 *NOT IN SERVICE* — In this state, the IMM shall not accept new process jobs or initiate execution of existing process jobs. It may be able to perform special services required for maintenance and diagnostics, including calibration and associated loading/unloading activities. Either the external Control Client or the IMM may put it in an NOT IN SERVICE state. Once put into NOT IN SERVICE, only the one who put it NOT IN SERVICE may change the state back to IN SERVICE. This is reflected in the two substates, EXTERNAL SELECT and INTERNAL SELECT.

11.2.3.3 *EXTERNAL SELECT* — The Control Client has placed the IMM out of service.

11.2.3.4 *INTERNAL SELECT* — The IMM has placed itself out of service due to a fault condition.

11.2.4 *Service State Model Transitions* — Transitions are defined for the Service State Model in Table 1.

**Table 1 Service State Model Transitions**

| # | *Previous State* | *Trigger* | *New State* | *Comment* |
|---|---|---|---|---|
| 1 | (no state) | IMM initialization. | IN SERVICE or NOT IN SERVICE | Initial state is determined first by existence of exceptions or alarm conditions. If none, then initial state shall be the last known state. |
| 2 | IN SERVICE | The IMM detects one of the conditions for NOT IN SERVICE<br>OR<br>The IMM client has issued a ChangeService request. | NOT IN SERVICE | Transitions 2 and 4 are combined for reporting purposes. |
| 3 | NOT IN SERVICE | The IMM or communications partner has returned the IMM to IN SERVICE. | IN SERVICE | |
| 4 | System initialization<br>or<br>IN SERVICE | Entry into NOT IN SERVICE:<br>4-1 Following system initialization,<br>4-2 IMM initiated state change, and<br>4-3 Client requested state change. | EXTERNAL SELECT (trigger 4-3)<br>or INTERNAL SELECT (triggers 4-1 or 4-2) | Transitions 2 and 4 are combined for reporting purposes. |

INACTIVE



Documents Document Sharing Hub
Full document details: SEMI E127-0308

11.2.5 The Control Client may request a state change through a Service request at any time.

11.2.6 A transition to OUT OF SERVICE shall not interrupt any process job currently executing. It is recommended that all transitions to OUT OF SERVICE be delayed until such process jobs have completed, either normally or abnormally. This avoids confusion or an unintended internally-generated abort of process jobs. Process jobs must be aborted separately and deliberately when necessary.

11.3 *Object Attributes*

11.3.1 Table 2 defines the attributes of the IMM Object as shown in Figure 3.

**Table 2 IMM Object Attribute Definitions**

| *Attribute Name* | *Definition* | *Access* | *Reqd* | *Form* |
|---|---|---|---|---|
| ObjType | Defines object type. | RO | Y | Text = "EquipmentModule" |
| ObjID | Identifier of specific module. | RO | Y | Conforms to requirement of ObjID [SEMI E39]. |
| EventList | List of event name/event number pairs. | RO | Y | List of text/integer pairs. |
| RetentionRules | Retention Rules supported by the IMM. | RO | Y | List of 0+ conditions. |
| Service | Current status of the Service State Model. | RO | Y | Enumerated:<br>InService<br>NotInService |
| SubstrateList | A list of zero or more ObjID for all substrates currently within the IMM, regardless of location. | RO | Y | List of 0+ text. |
| TableCapacity | Indicates maximum number of tables that may be stored. | RO | Y | Integer. |
| TableCount | Indicates number of tables currently stored. | RO | Y | Integer. |
| TableStorageAlert | Remaining table storage capacity. Either the number of additional tables that can be stored (integer) or the remaining percentage of total capacity (floating point) available. When remaining capacity reaches this value, the Table Storage Capacity Warning event message is sent. | RO | Y | Integer or floating point. |

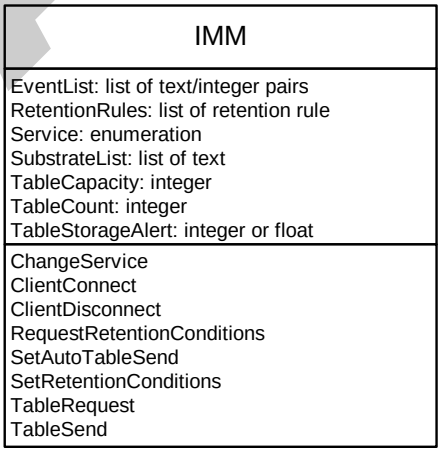


**Figure 3**
**IMM Object**

INACTIVE



Documents
Document Sharing Hub
Full document details: SEMI E127-0308

11.3.2 The attribute EventList identifies all of the errors that may be generated by the IMM object. Each such event shall have both a descriptive name and a numeric identifier, each of which shall be unique within the EventList list.

11.3.3 The attribute TableCapacity identifies the maximum number of tables that the IMM is capable of storing. The client may use this information for determining its own retention rules.

11.3.4 *Retention Conditions* — A question of concern to the IMM is the method of determining when a data table is no longer required. Each client is able to select individual conditions for retaining raw data and conversion tables. These conditions are the client-selected RetentionRules.

11.3.4.1 The IMM has a RetentionRules attribute that lists the specific retention rule elements supported by the IMM (see Table 3 for definitions of retention rule elements). An empty (null) list signifies that there are no retention rules that can be met to cause the table to be either deleted or retained.

11.3.4.2 The IMM maintains a list of rules for each client. This is a single level or two level list of rules from the RetentionRules list. This list is available on the RequestRetentionRules request by the client and is not publicly available as an attribute.

11.3.4.2.1 Once all retention rules have been satisfied for a given table for all clients, the IMM may delete the table at its own discretion based on internal requirements that are outside the scope of this standard. The IMM is responsible for managing both its storage space and those tables that do not need to be retained for any client.

11.3.4.2.2 As a single level list, when any item list is satisfied, the table may be deleted (a logical OR condition).

11.3.4.2.3 If one or more conditions need to be satisfied before a table may be deleted (a logical AND condition), the items should appear as a list of RetentionRules that occupy one element of the top level list. For instance, if the retention rule is that conditions A and B both need to be met, or condition C must be met, then the DefaultDataRetention rule is a two element list with one element being a 2 item list of A and B and the other is just condition C:

(A,B), C

11.3.4.2.4 An empty (null) list is equivalent to a “No Retention Rule” condition.

11.3.4.2.5 Actual representation of this list is dependent upon the communication protocol used.

11.3.4.3 Until or unless set by the client, the client’s retention rule is a zero element list, signifying that no retention rule has been set.

11.3.4.4 Except when required by automatic rollover (¶ 11.3.5.2), no IMM Data Table Object shall be deleted until all client retention rules have been satisfied.

11.3.4.5 The definitions of individual standard retention rule elements are provided in Table 3. Additional rules may be defined in the future. All of the individual rules shall be integers, where a non-value indicates that this rule is enabled. Otherwise it is disabled. The default value for each element is zero.

**Table 3 Retention Rule Element Definitions**

| *Retention Rule* | *Definition* |
|---|---|
| After Transfer | The table may be deleted after it has been sent to the client. Any non-zero value indicates that this rule is enabled. A value of zero disables it. |
| Client Delete | If ClientDelete is non-zero, tables are retained until deleted by the client. However, each time the table capacity drops below the value in ClientDelete, the TableStorageCapacityWarning event is generated and sent to the client, so the client can then delete tables no longer needed. If the client fails to delete sufficient tables, then the Automatic Rollover deletes the oldest tables as described below. |
| Maximum Tables Stored | This rule sets a value for the maximum number of tables that should accumulated before Automatic Rollover takes effect and the oldest table(s) are deleted when there is insufficient room to store a new table. Maximum value for this rule is Table Capacity – 1. |
| RetentionTime | The value in RetentionTime indicates the maximum length of time, in hours, that a table should be retained in storage. The table may be deleted after this period of time has expired. Any internal timers used are not visible to the client. |

INACTIVE



Documents Document Sharing Hub
Full document details: SEMI E127-0308

11.3.5 *Table Storage Management* — All tables shall be stored in non-volatile storage once they have been completed.

11.3.5.1 *Calibration Table* — The latest instance of Calibration Table shall always be retained in storage.

11.3.5.2 *Automatic Rollover* — In the event that there is not enough storage capacity remaining to store a new table, then the oldest tables of the same type as the new table (for either raw data or converted data) shall be deleted, one at a time, until there is room to store the new table, regardless of the presence or absence of retention rules.

NOTE 4: The size of Raw Data Table Objects and Converted Data Table Objects may be significantly different.

11.3.5.3 TableStorageAlert is a constant determined by the supplier that represents either the approximate number of tables (total) that may still be stored, or a percentage of total remaining storage capacity (TableCapacity). When the remaining capacity drops to the level indicated by TableStorageAlert, a Table Storage Capacity Warning event is sent (¶ 11.4.5).

11.4 *System Initialization* — Whenever the Integrated Measurement Module is powered on or reset, it goes through a system initialization before it is acknowledges communications with, or establishes, any potential client. Initialization activities include both software initialization and hardware initialization. On the completion of initialization, the IMM is in a known condition, and the following requirements have been satisfied:

- All state models have been set to their entry state,
- Communications have been initialized,
- The physical status of the IMM is known,
- The absence or presence of substrates is known,
- Exceptions, including all alarm conditions, have been determined,
- No process jobs are defined,
- No clients have been established, and
- All internal equipment substrate locations have been checked for the presence of substrates, and the total count of substrates detected has been compared with the number expected.

11.4.1 *Substrate Count Discrepancy*

11.4.1.1 The determination of expected substrate count requires that sufficient information about actual count must be maintained in non-volatile memory during run-time so that is available to accurately determine expected count during system initialization. For example:

- The content of the IMM attribute SubstrateList is saved in non-volatile memory whenever it is updated, and expected count is determined by the number of valid entries in the list, or
- A list of occupied internal substrate locations is retained in non-volatile memory each time any one is updated, and the expected count is determined by the number of valid entries in the list, or
- A simple count of the number of current substrates is retained in non-volatile memory.

11.4.1.2 If both the expected count and the actual count are zero, and there are no other errors or alarms, then the IMM Service state is IN SERVICE, and the IMM EPT state is IDLE.

11.4.1.3 If both the expected count and the actual count are equal and non-zero, or the actual count exceeds the expected count, then the IMM Service state is OUT OF SERVICE, and the IMM EPT state is BLOCKED. In this case, the Unexpected Substrate Alarm is set. The substrates shall be unloaded in an orderly manner. When all substrates have been unloaded, and all alarms are cleared, the IMM Service state changes to IN SERVICE, and the IMM EPT state changes to IDLE. The IMM may provide a method for a maintenance technician to determine if the "unexpected substrate" is actually present (e.g., the physical module may provide a view plate) and to indicate whether or not it is safe to unload. Such methods are beyond the scope of this standard.

11.4.1.4 If the expected count is greater than the actual count, then a Missing Substrate Alarm is set. The IMM remains OUT OF SERVICE and BLOCKED until the alarm condition is cleared. The IMM may provide a method for a maintenance technician to determine if the "lost substrate" is actually present (e.g., the physical module may provide a view plate) and to indicate whether or not it is safe to unload. Such methods are beyond the scope of this

INACTIVE



Documents
Document Sharing Hub
Full document details: SEMI E127-0308

standard. Note that sensor errors can result in an apparent loss or gain that is not real. Note also that it may be possible for a maintenance technician to interact with the IMM as a “control client”.

11.5 *IMM Events and Alarms*

11.5.1 In addition to the special events and alarms described below, all state transitions shall be capable of triggering an event, including all state models required by other standards required by IMMC.

11.5.2 *Alarms* — Various alarms caused by hardware-specific malfunction may occur at any time, including those detected by the Substrate Transfer Path Object (STPO). In addition, alarms may be caused by a discrepancy between actual and expected substrate counts.

11.5.2.1 *Missing Substrate Alarm* — This alarm is set whenever a substrate is no longer detected. The substrate is either broken (and must be manually removed) or a sensor has failed so that it can no longer detect a substrate. This condition shall cause the IMM Service state to set to OUT OF SERVICE and the IMM EPT state to be set to BLOCKED so long as this condition exists. This condition may occur at any time, either during system initialization or runtime. In either case, the alarm condition shall be cleared before processing continues. See ¶ 11.4.1.4 for further detail.

11.5.2.2 *Unexpected Substrate Alarm* — This alarm is set during system initialization when substrates have been detected, and the actual substrate count equals or exceeds the number expected. See ¶ 11.4.1.3 for further detail.

11.5.3 *SubstrateListUpdate* — The SubstrateList attribute is a list of the ObjIDs for all substrates currently present within the IMM module. Whenever a Substrate Object is created or deleted, this attribute is updated and an event message is sent with the ObjID of the Substrate Object associated with the change and whether it was added or removed.

11.5.4 *Table Storage Capacity Warning* — A notification sent to the client when table capacity decreases below TableStorageAlert and each time the storage further decreases.

11.5.5 *Table Storage Capacity Overflow* — Tables have been removed due to capacity overflow. Note: This is a trigger for Transition 6 in the IMM Data Table state model, ¶ 14.2.

11.5.6 *Event Variables* — Certain variables shall be provided that are associated with certain events and contain the values that were valid at the time the event occurred. For the IMM object, these variables are specified in Table 4. The names of variables as specified shall be implemented exactly as text strings.

**Table 4 Event Variables for IMM Object**

| *Variable Name* | *Description* | *Type* | *Event* | *Comment* |
|---|---|---|---|---|
| DateTime | The date and time that the event occurred. | Timestamp conforms to the format as specified by the TimeFormat equipment constant value setting (see SEMI E5). | All | |
| Service | Current state of Service State Model. | Enumerated IN SERVICE OUT OF SERVICE | All | |
| SubstrateListChangeType | Indicates the type of change: addition or deletion. | Enumerated Addition Deletion | SubstrateListUpdate | |

INACTIVE



Documents Document Sharing Hub
Full document details: SEMI E127-0308

| *Variable Name* | *Description* | *Type* | *Event* | *Comment* |
|---|---|---|---|---|
| SubstrateID | The last SubstrateID added to or removed from the SubstrateList attribute. | Text. | SubstrateListUpdate | |
| TableCount | The number of tables currently stored. | Integer. | TableStorageCapacityWarning, TableStorageCapacityOverflow | |

11.6 *IMM Object Services*

11.6.1 Table 5 defines the services specific to the Integrated Measurement Module. Message details are defined in § 19.

**Table 5 IMM Object Services**

| *Message Service Name* | *Description* | *Type* | *Reqd* |
|---|---|---|---|
| ChangeService | Set the IMM in service or out of service. | R | Y |
| ClientConnect | A request to become a Control Client or Data Client. | R | Y |
| ClientDisconnect | A request to terminate the Client role. | R | Y |
| RequestRetentionConditions | A request to send the rules and retention time for this client. | R | Y |
| SetAutoTableSend | A request to set AutoTableSend on or off for this client. | R | Y |
| SetRetentionConditions | The client sends a one or two level list of 0+ Retention Rules and an optional minimum retention time. | R | Y |
| TableRequest | The client requests a specific table, or table subset. | R | Y |
| TableSend | Used by the client to send a table or delete an existing table. Used by the IMM to request to send a table to the client by the AutoTableSend capability. | R | Y |

## 12 Substrate Transfer Path Object

12.1 *Description*

12.1.1 The Substrate Transfer Path object (STPO) manages the spatial envelope used for loading (receiving) and unloading (sending) a substrate, along with all mechanisms used in effecting a physical transfer. It has one or more Equipment Substrate Location objects for holding substrates after they are received and before they are sent. Each Substrate Location can hold exactly one substrate when it is occupied. Substrate and Substrate Location objects are defined in SEMI E90.

12.1.2 If substrates can be loaded into, and/or unloaded from, more than one substrate location, then each of these locations shall be represented by a separate STPO object. This allows the Control Client to specify where it intends to load or unload.

12.1.3 The STPO object allows the IMM to interact with an external substrate handler to transfer substrates to and from a Substrate Port (a substrate location together with any necessary mechanisms), and also to exchange properties of the Substrate Object being transferred.

12.1.4 A Substrate Transfer has two required parts, which may occur in either order: the substrate transfer and the substrate information transfer. The transfer procedure consists of the messages and physical actions necessary to effect the loading or unloading of a substrate to or from the Substrate Transfer Location. The exchange is not complete until both parts have been successfully completed.

12.1.5 The minimum information required for processing is SubstrateID. Additional information is used to store as context data in the IMM Data Table Objects discussed in § 13. The method provided for obtaining the additional information is the same as the method for providing SubstrateID.

12.1.6 The STPO shall provide two ordered lists of the sequence of text micro commands that are needed for loading and for unloading a substrate to and from the Substrate Transfer Location. The Control Client may obtain

INACTIVE



Documents
Document Sharing Hub
Full document details: SEMI E127-0308