

SEMI E41-95

EXCEPTION MANAGEMENT (EM) STANDARD

This Standard was technically approved by the Information & Control Global Technical Committee. Available at www.semi.org in 1995.

NOTICE: This Standard or Safety Guideline has an Inactive Status because the conditions to maintain Current Status have not been met. Inactive Standards or Safety Guidelines are available from SEMI and continue to be valid for use.

1 Purpose

1.1 Interactive exception handling enhances the error recovery ability while maintaining automated control in the factory. This Standard addresses the communications needs within the semiconductor manufacturing environment with respect to equipment exception handling.

1.2 This Standard specifies capabilities to be provided by the exception agent for effective reporting and interaction with respect to abnormal situations in the equipment. It describes the *concept* of exception management, the *behavior* of the equipment in relation to interactive exception handling, and the *messaging services* which are needed to provide the functionality.

1.3 The communications services defined here will enable standards-based interoperability of independent systems. They shall allow application software to be developed which can assume the existence of these services and allow software products to be developed which offer them.

1.4 Implementation of automated exception management will help reduce error recovery time and avoid changing from automatic to manual equipment control in many situations. The adoption of the standards described will greatly reduce the effort required to integrate compliant equipment components. Compliance requires a specific set of standard services.

2 Scope

2.1 The current scope of this Standard is interactive exception handling within a cluster tool.

2.2 While the functionality provided may be applied to other multi-resource equipment, it may not provide the flexibility required for automated management and command by the factory of all types of equipment. It is anticipated that this standard will be extended to accommodate management of exceptions in other types of multi-resource equipment and by the factory of all types of equipment.

2.3 This Standard supports exception condition reporting, including alarms, by an exception agent to a decision authority. The exception agent also has the ability to enable and disable reporting on each exception condition.

2.4 Interactive exception handling is supported through selection by the decision authority of recovery actions in certain situations. The recovery actions are performed by the exception agent with the goal of resolving the abnormal situation and allowing normal equipment operation to continue.

2.5 This Standard presents a solution from the concepts and behavior down to the messaging services. It does not define the messaging protocol.

2.6 A messaging service includes the identification that a message shall be exchanged and definition of the data which is contained in that message. It does not include information on the structure of the message, how the data is represented within the message, or how the message is exchanged. This additional information is contained within the *message protocol*.

2.7 The defined services may be applied to multiple protocols. Information on the mapping of exception management services to special protocols (e.g., SECS-II) are added as adjunct standards.

2.8 The services assume a communications environment in which a reliable connection has been established between the user of the services and the provider of the services. Establishing, maintaining, releasing a connection, and handling communication failures is beyond the scope of this Standard.

NOTICE: SEMI Standards and Safety Guidelines do not purport to address all safety issues associated with their use. It is the responsibility of the users of the Documents to establish appropriate safety and health practices, and determine the applicability of regulatory or other limitations prior to use.

3 Referenced Standards and Documents

3.1 SEMI Standards and Safety Guidelines

3.2 The following SEMI Standard is related to the Exception Management Standard:

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

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

4 Definitions

4.1 The following definitions are arranged in alphabetical order. Some are defined using terms defined elsewhere within this section. No references beyond this section should be necessary for a basic understanding of these terms.

4.1.1 *agent* — an intelligent system within a factory that provides one or more service resources and uses the services of other agents. A generalization of host, equipment, cell, cluster, cluster module, station controller, work station. Agents are associated with a physical system or a collection of physical systems, including computer platforms.

4.1.2 *alarm* — an alarm is related to any abnormal situation of the equipment that may endanger people, equipment, or material being processed.

4.1.3 *clearing* — exception agent to decision authority reporting that an abnormal situation related to an exception condition is no longer apparent or relevant.

4.1.4 *decision authority* — an entity requiring to be notified of significant exception condition changes and which decides how to proceed to resolve abnormal situations related to recoverable error conditions. The decision authority may be represented by a supervisory controller interacting with an operator who may ultimately choose the recovery action.

4.1.5 *error condition* — an exception condition which is not an alarm and which may support recovery actions requested by a decision authority.

4.1.6 *exception agent* — the entity which manages access to and reporting of information on abnormal situations in equipment. It achieves this by defining exception conditions, each related to a significant abnormal situation. It may provide services for a decision authority to direct the recovery from certain situations.

4.1.7 *exception condition* — a condition managed by an exception agent for reporting on and recovery from an abnormal situation in the equipment.

4.1.8 *form* — type of data representing information contained in an object attribute or service message parameter. The data types are detailed in § 4.2.

4.1.9 *fundamental requirements* — the requirements for information and behavior that must be satisfied for compliance with a standard. Fundamental requirements apply to specific areas of application, objects, or services.

4.1.10 *posting* — all exception agent to decision authority reporting associated with an exception condition while the related abnormal situation is apparent and relevant.

4.1.11 *recovery action* — an operation associated with an error condition with the aim of resolving the abnormal situation detected. It may supply information to the exception agent or request the exception agent to perform some activity.

4.1.12 *service* — the set of messages and definition of the behavior of a service provider that enables remote access to a particular functionality.

4.1.13 *service-provider* — the software control entity that is the provider of any of the related services.

4.1.14 *service-user* — the software control entity that is the user of any of the related services.

4.2 Data Type

4.2.1 *form* — type of data: positive integer, unsigned integer, integer, enumerated, Boolean, text, formatted text, structure, list, ordered list.

4.2.2 *positive integer* — may take the value of any positive whole number. Messaging protocol may impose a limit on the range of possible values.

4.2.3 *unsigned integer* — may take the value of any positive integer or zero. Messaging protocol may impose a limit on the range of possible values.

4.2.4 *integer* — may take on the value of any negative or unsigned integer. Messaging protocol may impose a limit on the range of possible values.

4.2.5 *enumerated* — may take on one of a limited set of possible values. These values may be given logical names, but they may be represented by any single-item data type.

4.2.6 *boolean* — may take on one of two possible values, equating to TRUE or FALSE.

4.2.7 *text* — a text string. Messaging protocol may impose restrictions, such as length or ASCII representation.

4.2.8 *formatted text* — a text string with an imposed format. This could be by position, by use of special characters, or both.

4.2.9 *structure* — a complex structure consisting of a specific set of items, of possibly mixed data types, in a specified arrangement.

4.2.10 *list* — a set of one or more items that are all of the same form (one of the above forms).

4.2.11 *ordered list* — a list for which the order in which items appear is significant.

5 Conventions

5.1 *Harel State Model* — This Document uses the Harel State Chart notation to describe the dynamic behavior of the objects defined. An overview of this notation is presented in Related Information 1 of SEMI E30. The formal definition of this notation is presented in Science of Computer Programming 8, “Statecharts: A Visual Formalism for Complex Systems,” by D. Harel, 1987.

Transition tables are provided in conjunction with the state diagrams to describe explicitly the nature of each state transition. A transition contains columns for Transition #, Current State, Trigger, New State, Action(s). The ‘trigger’ (column 3) for the transition occurs while in the ‘current’ state. The ‘actions’ (column 5) include a combination of (1) actions taken upon exit of the current state, (2) actions taken upon entry of the new state, and (3) actions taken which are most closely associated with the transition. No differentiation is made.

5.2 *Object Attribute Representation* — The object information models for standardized objects will be supported by an attribute definition table with the following column headings:

<i>Attribute Name</i>	<i>Definition</i>	<i>Access</i>	<i>Rqmt</i>	<i>Form</i>
The formal text name of the attribute	Description of the information contained	RO or RW	Y or N	(see below)

The Access column uses RO (Read Only) or RW (Read and Write) to indicate the access that service-users have to the attribute.

A ‘Y’ or ‘N’ in the requirement (Rqmt) column indicates whether or not this attribute must be supported in order to meet fundamental compliance for the service.

The Form column is used to indicate the format of the attribute. (See § 4.1 for definitions.)

5.3 Service Message Representation

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

<i>Message Service Name</i>	<i>Type</i>	<i>Description</i>
Message Name	N or R	The intent of the service.

Type can be either N = Notification or R = Request.

Notification type messages are initiated by the service provider, and the provider does not expect to get a response from the consumer/subscriber.

Request messages are initiated by a service consumer or subscriber. Request messages ask for data or an activity from the provider. Request messages expect a specific response message (no presumption on the message content).

5.3.2 Service Parameter Dictionary — A service parameter dictionary table defines the parameters used in a service, as shown in the following table:

<i>Parameter</i>	<i>Form</i>	<i>Description</i>
Parameter X	Data type	A parameter called X is B in A.

A row is provided in the table for each parameter of the service. The first column contains the name of the parameter. This is followed by columns describing the form and contents of the corresponding primitive.

The Form column is used to indicate the type of data contained in a parameter. (See § 4.1 for definitions.)

The Description column in the Service Parameter Dictionary table describes the meaning of the parameter, the values it can assume, and any interrelationships with other parameters.

To prevent the definition of numerous parameters named ‘XxxList,’ this Document adopts the convention of referring to the list as ‘(List of) Xxx.’ In this case, the definition of the variable Xxx will be given, not of the list. The term ‘list’ indicates a collection (or set) of zero or more items of the same data type. Where a list is used in both the request and the response, the list order in the request is retained in the response. A list must contain at least one element unless zero elements are specifically allowed.

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

<i>Parameter</i>	<i>Req/Ind</i>	<i>Rsp/Cnf</i>	<i>Description</i>
Parameter X	(see below)	(see below)	A description of the service.

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.’

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 another 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).

6 Overview

Exception management is concerned with the interactive handling of equipment exception conditions. This Standard defines the services provided by which abnormal situations are reported by an exception agent (service-provider) to a decision authority (service-user). In the case of recoverable situations, services are provided for a decision authority to choose how to proceed to resolve the abnormal situation.

The exception agent is the equipment entity which manages access to and reporting of information on abnormal situations. It achieves this by defining exception conditions, each related to a significant abnormal situation. All services are defined in terms of these exception conditions.

Exception management allows for the decision authority to direct the resolution of an abnormal situation. The decision authority selects a recovery action to be performed to resolve the situation from among the options supplied with the exception condition. The exception agent performs the requested recovery action, which may or may not resolve the situation.

The services are fully defined in terms of the functionality provided by the exception agent (service-provider) and as such do not dictate the architecture of the decision authority (service-user).

This Standard describes the concepts and exception condition model on which the communications are based, followed by the detailed behavioral model used. It then describes the standard object attributes and message services in detail.

6.1 Compliance — Compliance with this Standard includes adherence to all stated requirements in this Document where implemented. This includes defined message services and state models.

There are two levels of compliance defined. The first is alarm reporting support. The second extends support to include interactive handling of recoverable exception conditions. Required capabilities are listed in Fundamental Requirements, § 10.4.

7 Concepts

7.1 Exception Management Model — The exception management model describes the mechanism for interactive handling of equipment exception conditions.

An exception condition is a condition monitored in equipment by an exception agent (service-provider) for detecting an abnormal situation. Exception conditions are limited in this Standard to those accessible by a remote decision authority (service-user).

An exception condition is persistent, existing whether or not the abnormal situation currently exists. The abnormal situation is indicated by the exception condition state becoming SET. An exception agent is an entity which manages remote decision authority access to any number of exception conditions. Each exception condition is identified by a name which is unique for the exception agent.

A decision authority is a remote entity requiring to be notified of exception condition information and which decides appropriate actions to take to resolve abnormal situations. The decision authority may be represented by a supervisory controller interacting with an operator who may ultimately choose the recovery action.

The exception agent notifies the decision authority on detection of an abnormal situation related to an exception condition (i.e., state is SET) and again when it is no longer apparent. It also provides access to exception condition attribute data and for the execution of recovery actions requested by the decision authority.

Exception management defines two types of exception conditions: alarms and error conditions.

An alarm is related to any abnormal situation on the equipment that may endanger people, equipment, or material being processed. Alarms do not provide for decision authority involvement in the resolution of the abnormal situation.

An error condition is related to any abnormal situation detected which is made accessible to a decision authority. An error condition may supply a list of possible recovery actions from which the decision authority can select to attempt to resolve the abnormal situation, thereby resulting in the error condition state becoming CLEARED.

Being persistent, exception condition attributes may be queried at any time, and reporting of changes, such as state SET/CLEARED, can be disabled by the decision authority.