Stream,Function Name (Mnemonic)	*Direction*
S17,F9 Collection Event Link Request (CELR)	M,H->E,reply
Description	
Establish a Collection Event Report definition with respect to a specific Event Source.	
Structure	
L,4 1. <DATAID> 2. <EVNTSRC> 3. <CEID> 4. L,n (n is the number of Reports to be linked) 1. <RPTID$_1$> 2. <RPTID$_2$> . . n. <RPTID$_n$>	
Exception	

Stream,Function Name (Mnemonic)	*Direction*
S17,F10 Collection Event Link Acknowledge (CELA)	S,H<-E
Description	
Indicate the success or failure of a Collection Event Link Request.	
Structure	
L,3 1. <EVNTSRC> 2. <CEID> 3. <ERRCODE>	
Exception	
Item 3 should be set to zero length to indicate success.	

Stream,Function Name (Mnemonic)	*Direction*
S17,F11 Collection Event Unlink Request (CEUR)	S,H->E,reply
Description	
Request to unlink a specific Data Report from a Collection Event Report.	
Structure	
L,3 1. <EVNTSRC> 2. <CEID> 3. <RPTID>	
Exception	
Item one can be zero length, in which case the default event source is assumed.	

Stream,Function Name (Mnemonic)	*Direction*
S17,F12 Collection Event Unlink Acknowledge (CEUA)	S,H<-E
Description	
Indicates success or failure of a requested Unlink.	
Structure	
L,4 1. <EVNTSRC> 2. <CEID> 3. <RPTID> 4. <ERRCODE>	
Exception	
Item one can be zero length to indicate the default event source. Item 4 is set to zero length if the primary request was successful.	

Stream,Function Name (Mnemonic)	*Direction*
S17,F13 Trace Reset Request (TRR)	S,H->E,reply
Description	
The Host requests the equipment to clear the data and reset the specified trace reports. If n = 0, then all defined Trace Objects will be reset.	
Structure	
L,n 1. <TRID$_1$> 2. <TRID$_2$> . . n. <TRID$_n$>	
Exception	
None	

Stream,Function Name (Mnemonic)	*Direction*
S17,F14 Trace Report Reset Acknowledge (TRRA)	S,H<-E
Description	
This list in item 1 contains the identifiers of all the Trace Objects which were reset. If all Trace Objects are successfully reset, then ACKA shall be set to TRUE.	
Structure	
L,2 1. <ACKA> 2. L,m 1. L,3 1. <TRID$_1$> 2. <ERRCODE$_1$> 3. <ERRTEXT$_1$> . . m. L,3 1. <TRID$_m$> 2. <ERRCODE$_m$> 3. <ERRTEXT$_m$>	
Exception	
If ACKA is TRUE, then no errors were encountered, meaning all report requests were completed successfully and a zero-length list (m = 0) shall be sent. If some reports could not be reset, then their TRIDs shall be given in a space separated list in ERRTEXT.	

10.22 *Stream 18 Subsystem Control and Data* — Messages exchanged between component subsystems and higher level controllers. Compared to similar messages exchanged between equipment and host, subsystem messages are less complex.

<table>
<tr><td>Stream,Function Name (Mnemonic)</td><td>Direction</td></tr>
<tr><td>S18,F0 Abort Transaction (S18F0)</td><td>S,H<->E</td></tr>
<tr><td colspan="2">Description</td></tr>
<tr><td colspan="2">Used in lieu of an expected reply to abort a transaction. Function 0 is defined in every stream and has the same meaning in every stream.</td></tr>
<tr><td colspan="2">Structure</td></tr>
<tr><td colspan="2">Header only</td></tr>
<tr><td colspan="2">Exception</td></tr>
<tr><td colspan="2"></td></tr>
</table>

<table>
<tr><td>Stream,Function Name (Mnemonic)</td><td>Direction</td></tr>
<tr><td>S18,F1 Read Attribute Request (RAR)</td><td>S,H->E,reply</td></tr>
<tr><td colspan="2">Description</td></tr>
<tr><td colspan="2">This message requests the current values of specified attributes of the subsystem component indicated in TARGETID.</td></tr>
<tr><td colspan="2">Structure</td></tr>
<tr><td colspan="2">L,2
1. <TARGETID>
2. L,n n = # attribute identifiers
1. <ATTRID$_1$>
.
.
n. <ATTRID$_n$></td></tr>
<tr><td colspan="2">Exception</td></tr>
<tr><td colspan="2">If n = 0, then all attributes of the target component are requested.</td></tr>
</table>

<table>
<tr><td>Stream,Function Name (Mnemonic)</td><td>Direction</td></tr>
<tr><td>S18,F2 Read Attribute Data (RAD)</td><td>S,H<-E</td></tr>
<tr><td colspan="2">Description</td></tr>
<tr><td colspan="2">This message returns the current values of requested attributes and the current status of the requested component indicated in TARGETID. Attributes are returned in the order requested.</td></tr>
<tr><td colspan="2">Structure</td></tr>
<tr><td colspan="2">L,3
1. <TARGETID>
2. <SSACK>
3. L,n
1. <ATTRDATA$_1$>
.
.
n. <ATTRDATA$_n$></td></tr>
<tr><td colspan="2">Exception</td></tr>
<tr><td colspan="2">Both n = 0 and s = 0 if the target component is unknown.</td></tr>
</table>

Stream,Function Name (Mnemonic)	*Direction*
S18,F3 Write Attribute Request (WAR)	S,H->E,reply
Description	
This message requests the subsystem to set the value of read/write attributes of the component specified in TARGETID.	
Structure	
L,2 1. <TARGETID> 2. L,n 1. L,2 1. <ATTRID$_1$> 2. <ATTRDATA$_1$> . . n. L,2 1. <ATTRID$_n$> 2. <ATTRDATA$_n$>	
Exception	
None	

Stream,Function Name (Mnemonic)	*Direction*
S18,F4 Write Attribute Acknowledge (WAA)	S,H<-E
Description	
This message acknowledges the success of failure of the request to write attribute data to the subsystem indicated in TARGETID.	
Structure	
L, 3 1. <TARGETID> 2. <SSACK> 3. <STATUSLIST>	
Exception	
s = 0 if the target component is unknown.	

Stream,Function Name (Mnemonic)	*Direction*
S18,F5 Read Request (RR)	S,H->E,reply
Description	
The host requests the subsystem indicated in TARGETID to read information. DATASEG may be used to indicate a specific section of data to be read. DATALENGTH is used to limit the amount of data for that section.	
Structure	
L,3 1. <TARGETID> 2. <DATASEG> 3. <DATALENGTH>	
Exception	
If DATASEG and DATALENGTH are both omitted (are zero length items) then all data is requested. If DATALENGTH only is omitted, then all data within the indicated section is requested.	

Stream,Function Name (Mnemonic)	*Direction*
S18,F6 Read Data (RD)	S,H<-E

Description

This message is used to return requested information from the subsystem indicated in TARGETID or to acknowledge the results of the request.

Structure

```
L,4
  1. <TARGETID>
  2. <SSACK>
  3. <DATA>
  4. L,s
       1. <STATUS_1>
       .
       .
       s. <STATUS_s>
```

Exception

If TARGETID is unknown, then DATA is zero length.

Stream,Function Name (Mnemonic)	*Direction*
S18,F7 Write Data Request (WDR)	S,H->E,reply

Description

This message requests to write data to the subsystem component indicated in TARGETID. DATASEG may be used to indicate a specific section of data to be written or overwritten.

Structure

```
L,4
  1. <TARGETID>
  2. <DATASEG>
  3. <DATALENGTH>
  4. <DATA>
```

Exception

If DATASEG and DATALENGTH are both omitted (are zero length items) then all data is to be overwritten. If only DATALENGTH is omitted, then all data within the indicated section is to be written.

Stream,Function Name (Mnemonic)	*Direction*
S18,F8 Write Data Acknowledge (WDA)	S,H<-E

Description

This message acknowledges the success or failure of writing data to the subsystem indicated in TARGETID.

Structure

```
L,3
  1. <TARGETID>
  2. <SSACK>
  3. <STATUSLIST>
```

Exception

s = 0 if and only if TARGETID is unknown.

Stream,Function Name (Mnemonic)	*Direction*
S18,F9 Read ID Request (RIR)	S,H->E,reply
Description	
This message is used to request the subsystem indicated by TARGETID to read an identifier.	
Structure	
`<TARGETID>`	
Exception	
None	

Stream,Function Name (Mnemonic)	*Direction*
S18,F10 Read ID Data (RID)	S,H<-E
Description	
This message returns a requested material identifier MID as read by the subsystem indicated in TARGETID.	
Structure	
L,4 1. <TARGETID> 2. <SSACK> 3. <MID> 4. <STATUSLIST>	
Exception	
s = 0 if and only if TARGETID is unknown.	

Stream,Function Name (Mnemonic)	*Direction*
S18,F11 Write ID Request (WIR)	S,H->E,reply
Description	
This message is used to request the subsystem indicated by TARGETID to write an identifier.	
Structure	
L,2 1. <TARGETID> 2. <MID>	
Exception	
None	

Stream,Function Name (Mnemonic)	*Direction*
S18,F12 Write ID Acknowledge (WIA)	S,H<-E
Description	
This message acknowledges the success or failure of the subsystem specified in TARGETID in writing the ID.	
Structure	
L,3 1. <TARGETID> 2. <SSACK> 3. <STATUSLIST>	
Exception	
s = 0 if and only if TARGETID is unknown.	

Stream,Function Name (Mnemonic)	*Direction*
S18,F13 Subsystem Command Request (SCR)	S,H->E,reply
Description	
This message is used to request the subsystem indicated in TARGETID to perform a specific action.	
Structure	

L,3
  1. <TARGETID>
  2. <SSCMD>
  3. L,n
      1. <CPVAL$_1$>
      .
      .
      n. <CPVAL$_n$>	

Exception	
If n = 0, no parameters are provided.	

Stream,Function Name (Mnemonic)	*Direction*
S18,F14 Subsystem Command Acknowledge (SCA)	S,H<-E
Description	
This message reports the results from the subsystem specified in TARGETID for the requested action.	
Structure	

L,3
  1. <TARGETID>
  2. <SSACK>
  3. <STATUSLIST>	

Exception	
s = 0 if and only if TARGETID is unknown.	

Stream,Function Name (Mnemonic)	*Direction*
S18,F15 Read 2D Code Condition Request (R2DCCR)	S,H->E,reply
Description	
The host requests the subsystem indicated in TARGETID to read information on 2D Code Condition.	
Structure	
<TARGETID>	
Exception	
None	

Stream,Function Name (Mnemonic)	*Direction*
S18,F16 Read 2D Code Condition Data (R2DCCD)	S,H<-E
Description	
This message is used to return requested information on 2D Code Condition from the subsystem indicated in TARGETID or to acknowledge the results of the request.	
Structure	
L,5 1. <TARGETID> 2. <SSACK> 3. <MID> 4. <STATUSLIST> 5. <CONDITIONLIST>	
Exception	
If <SSACK> is failure, the length of the <CONDITIONLIST> must be zero.	

11 Message Documentation

11.1 *Intent* — Equipment makers using SECS-II messages must communicate the equipment-specific details of each message to the host designer in order for the host to properly adapt to the equipment. The details are communicated in a document which will follow a standard form in order to convey most clearly the information required. The following form is presented here to act as a guide for organizing the equipment-specific details.

11.2 *Standard Form SECS-II Document* — The standard form will contain three clearly labeled parts as follows.

Part I — General Information

Part II — Message Summary

Part III — Message Detail

11.2.1 Part I will contain general information on the following:

- Manufacturer and product number
- General description of equipment function
- Intended function of interface
- Software revision code
- Changes from previous versions

11.2.2 Part II will contain two lists of all messages understood and all messages sent by the equipment in terms of their stream and function codes. The first list will have pairs of columns: the first for the message received and understood and the second for the message sent in response. The second list will also have two columns: the first for the message sent and the second for the response understood. The message will be identified using the format "SxxFyy," where xx is the stream number and yy is the function number. Each transaction will be on a separate line. A "-" will indicate that one of a pair is not included. All messages not listed on the received side are implied to cause an error message to the host. All messages not listed on the sent side are assumed never to be sent from the equipment. Since some messages can be sent in two directions, the same message pair may appear in each list with the sent and received orders interchanged. A transaction listed in the standard as being allowed in two directions does not have to be implemented in both directions. This list will indicate which directions are implemented.

11.2.3 Part III will contain the details for every message listed in Part II. Messages that appear on both the sent and received sides must be detailed separately. The details shall include the following information on the data in each message:

1. For each fixed item, all the values or strings either understood or possible to send are listed along with their meaning to the equipment.
2. For each variable item, the restriction on or possible range of value or length of string.
3. Any other special interpretation of the message.

11.2.4 Each message so detailed must be clearly labeled with its stream and function code.

12 Units of Measure

12.1 *Intent* — Certain SECS-II transactions require specification of units of measure for data items passed between equipment and host. The concept of units of measure has been included as part of the SECS-II standard to enhance the ability of the host system to prompt its human operators for proper information when generating process programs, and also to facilitate automated handling of process programs by host systems and automated handling of data reported to a host by equipment.

12.2 *Units Symbols* — Under SECS-II, a units symbol is a text string of unspecified length which specifies the physical significance of a numeric value. Units of measure symbols under SECS-II may be either a SECS-II recognized unit identifier, a SECS-II unit identifier with prefix and/or suffix symbols, or an arithmetic expression of SECS-II identifiers.

12.2.1 A SECS-II units identifier is a text string which may be the full name, an abbreviation of the full name, or a special character which is unique for a specific unit of weight or measure. Identifier strings may consist of upper or lower case alphabetic characters and numerals or special characters of the ASCII character set. The first character of an identifier may not be a numeral. The case of alphabetic characters is significant (e.g., G and g, the units symbols for Gauss and gram, respectively).

12.2.2 A unit identifier may be nationally or internationally recognized, may be unique to the semiconductor industry, or, due to the special requirements of SECS-II, may be unique to this standard. Section 12.4 lists all units identifiers recognized by SECS-II. For each identifier defined in Section 12.4, six pieces of information are provided. They are:

1. Unit — Full name of the unit of measure in question.
2. Unit Identifier — SECS-II-recognized identifier for the unit.
3. Prefix Allowed — Specifies whether or not the unit identifier may be combined with a prefix symbol to generate a unit identifier which is a decimal multiple or submultiple of the base unit. Metric (or SI) units are usually capable of accepting a prefix symbol while English units may not.
4. Suffix Allowed — Specifies whether or not the unit identifier may be concatenated with a numeric suffix which provides additional information to the meaning of the associated unit symbol. The numeric suffix is composed of the ASCII digits 0 through 9 and represents a decimal value. This meaning of the numeric value is symbol-dependent and must be specified in the description section of the unit symbol's definition.
5. Equivalence — In those cases where a unit can be expressed as an arithmetic expression (of simpler units), this column will contain the expression of simpler units. For those units which are non-standard to either of the standard systems of units of measure (English or SI), this column will contain an expression which relates the non-standard unit to the equivalent unit of the standard units system. In either case, the expression provided in this column may be substituted for the corresponding SECS-II units identifier whenever required.
6. Description — Additional information as may be required to uniquely define the unit of measure in question.

12.2.3 Any SECS-II identifier which Section 12.4 indicates as being capable of taking on a prefix symbol may be appended to one of the prefix symbols shown in Table 23, forming a new unit which is a decimal multiple or submultiple of the base unit. A prefix symbol may not be used alone. It must appear concatenated to one of the identifiers in Section 12.4. Finally, only one prefix symbol may appear before any identifier. A units symbol such as "mus" (micromillisecond) is not allowed. The proper symbol is "ns" (nanosecond).

Table 23

Prefix Name	*Multiplicative Factor*	*Prefix Symbol*
exa	10^18	E
peta	10^15	P
Tera	10^12	T
giga	10^9	G
mega	10^6	M
kilo	10^3	k
hecto	10^2	h
deka (deca)	10^1	da
deci	10^-1	d
centi	10^-2	c
milli	10^-3	m
micro	10^-6	u
nano	10^-9	n
pico	10^-12	p
femto	10^-15	f
atto	10^-18	a

12.2.4 Any SECS-II identifier which Section 12.4 indicates as being capable of taking on a suffix value may have a numeric string appended to it. This decimal value allows the user to identify one of a family of symbol names with only the generic symbol name of the family being defined in Section 12.4. The meaning of a numeric suffix is dependent on the particular symbol with which it is being used and must be defined in the description section of the symbol definition.

12.2.5 Arithmetic expressions of units of measure identifiers are recognized by SECS-II as units symbols if they are formed by the following rules:

1. All units identifiers in the expression are SECS-II units identifiers defined in Section 12.4 or SECS-II prefixed units identifiers as defined above.
2. Exponentiation is denoted by a circumflex (^) between the identifier to be operated on and the exponent. Exponents may be positive or negative values. A negative value is denoted by a unary minus sign (-) between the circumflex and exponent. For positive values, the exponent will immediately follow the circumflex (A^2 or A^-2).
3. Multiplication of units identifiers is expressed by an asterisk (*) positioned between the factors to be multiplied (A*B).
4. Division of units identifiers is expressed by a slash (/) positioned between the dividend and divisor. Division may also be expressed as the product of the dividend and the divisor with a negative exponent (A/B or A*B^-1).
5. Parentheses may be used to specify the order in which the arithmetic operations will be performed.
6. Within expressions or sub-expressions where parentheses do not specify the order of operations, exponentiation will be carried out first, followed by left-to-right evaluation of all multiplication and division that is (A*B^-2*30*C^2) is equivalent to (((A/(B^2))*30)*(C^2)).

12.3 *Compliance* — For the units of measure information to have any value and to be in compliance with SECS-II, equipment and host system manufacturers must ensure that only units symbols allowed by SECS-II are used by their systems. In those instances where SECS-II does not provide a units symbol required for a particular application, the manufacturer requiring the new symbol may submit a proposal to the SEMI Communications Subcommittee requesting the enhancement. A proposal must include all the information provided by each entry of Section 12.4 as described above.

12.3.1 A proposal must undergo the full approval cycle as prescribed by SECS-II for amending a standard (acceptance by committee, balloting, etc.). As a result, the proposal should be submitted as soon as possible, so that sufficient time is available to complete the standard amendment process and to notify all interested parties of the change before the product requiring the new symbol becomes available for use in a manufacturing facility.

12.4 *SECS-II Units of Measure Identifiers* — All units of measure symbols recognized by SECS-II are defined in this section or are compound symbols based on the identifiers defined here and formed by the rules specified in Section 12.2. Portions of the information provided below have been obtained from ANSI/IEEE 260-1978, ANSI X3.5-1976, ISO 2955-1974(E), Webster's New World Collegiate Dictionary (copyright 1977), and the CRC Handbook of Chemistry and Physics (52nd edition for 1971-1972).

Table 24 SECS-II Units of Measure Identifiers

Unit	*Identifier*	*Equivalence*	*Prefix Allowed*	*Suffix Allowed*	*Description*
Non-dimensional quantities (pure numbers)	null string	None	No	No	For all quantities which have no associated unit of measure, a zero length (null) text string is the appropriate 'identifier' to use when units of measure information is required.
ampere	A	None	Yes	No	SI unit of electric current.
ampere (turn)	AT	None	Yes	No	SI unit of magnetomotive force.
angstrom	Ang	m*10^-10	Yes	No	Unit of length used when measuring wavelength of light.
atmosphere, standard	atm	101325*Pa	No	No	A unit of pressure.
atmosphere, technical	at	kgf/cm^2	No	No	A unit of pressure.
atomic mass unit (unified)	u	1.660531*10^-27*kg	No	No	One twelfth the mass of an atom of carbon 12 nuclide.
bar	bar	100*kPa	Yes	No	CGS unit of pressure.
barn	barn	10^-28*m^2	Yes	No	Unit for measuring capture cross sections of elements.
barrel (petroleum)	bbl	42*gal or 158.99*1	No	No	A unit of volume.
baud	Bd	bit/s	Yes	No	Telecommunications measure of data transfer rate equivalent to one bit of information transferred per second.
bel	B	None	Yes	No	The logarithm of the ratio of two power signals.
Becquerel	Bq	None	Yes	No	SI unit of activity of a radionuclide.
bit	bit	None	Yes	No	A unit of computer information equivalent to the choice between two alternatives (as yes or no, on or off).
boat	boat	None	No	Yes	Special SECS generic unit corresponding to a holder of wafers or packages with discrete positions. The unit capacity is specified by the symbol's suffix, if provided. Otherwise, the capacity is situation-dependent.
British thermal unit	Btu	1054.35*J	No	No	The quantity of heat required to raise the temperature of one pound of water one degree Fahrenheit at or near 39.2˚F.
byte	byte	8*bit	Yes	No	Unit of storage for computer memory.
calorie (International Table)	callIT	4.1868*J	Yes	No	Defined by the 1929 International Stream Table Conference to be 1/860 international joules or l/859.858 joules.
calorie (thermochemical)	cal	4.l840*J	Yes	No	Unit of energy defined by the NBS to be 4.184 joules. Also, called the gram calorie.
candela	cd	None	Yes	No	SI unit of luminous intensity.
candle	cd	None	Yes	No	Alternate name for candela.
carrier	carrier	None	No	Yes	Special SECS generic unit corresponding to a holder for substrates, wafers or wafer frames. The unit capacity is specified by the symbol's suffix, if provided. Otherwise, the capacity is situation-dependent.
cassette	css	None	No	Yes	Special SECS generic unit corresponding to a holder for wafers or wafer frames. The unit capacity is specified by the symbol's suffix, if provided. Otherwise, the capacity is situation-dependent.
Coulomb	C	A*s	Yes	No	SI unit of electric charge.

Unit	*Identifier*	*Equivalence*	*Prefix Allowed*	*Suffix Allowed*	*Description*
curie	Ci	3.7*l0^l0*Bq	No	No	A unit of activity of radionuclide.
cycle	c	None	Yes	No	Unit equivalent to one complete performance of a periodic process.
darcy	D	cP*(cm/s)(cm/atm) or 0.986923*um^2	No	No	Unit of permeability of a porous medium. By traditional definition, a permeability of one darcy will permit a flow of 1 cm^3/s of fluid of 1 cP viscosity through an area of 1 cm^2 under a pressure gradient of 1 atm/cm.
day (mean solar)	d	24*h	No	No	The period required for the Earth to complete one rotation about its axis.
degree (plane angle)	deg	pi/180*rad	No	No	One three hundred sixtieth part of the circumference of a circle.
degree Celsius	degC	None	No	No	Unit of temperature where 0˚C corresponds to the freezing point of water and l00˚C corresponds to the boiling point at standard atmospheric conditions.
degree Fahrenheit	degF	None	No	No	Unit of temperature where 32˚F corresponds to the freezing point of water and 212˚F corresponds to the boiling point at standard atmospheric conditions.
degree Kelvin	K	None	No	No	SI unit of temperature.
die	die	None	No	No	Special SECS generic unit corresponding to an individual integrated circuit both on a wafer and after wafer separation. Also referred to as a bar or chip.
dyne	dyn	10^-5*N	Yes	No	Unit of force in the cgs system. One dyne is the force required to provide a one grain mass with an acceleration of 1 cm/s^2.
electronvolt	eV	l.60209*10^-19*J	Yes	No	Energy acquired by a small particle carrying a unit electronic charge when it falls through a potential difference of one volt.
erg	erg	l0^-7*J	Yes	No	Unit of work or energy in the cgs system. One erg is equal to the work done or energy expended to exert a force of one dyne through a distance of 1 cm.
farad	F	A*s/V	Yes	No	SI unit of capacitance.
foot	ft	12*in	No	No	English unit of length.
footcandle	Fc	lm/ft^2	No	No	Unit of illuminance. Also called lumen per square foot.
footlambert	FL	(1/pi)*cd/ft^2	No	No	A unit of luminance. One lumen per square foot leaves a surface whose luminance is one footlambert in all directions within a hemisphere.
gal	Gal	cm/s^2	Yes	No	A unit of acceleration used especially for values of gravity.
gallon (US)	gal	23l*in^3 or 4*qt or 3.7854*1	No	No	United States version of English system unit of volume.
gallon (UK)	galUK	4.546l*l	No	No	United Kingdom version of English system unit of volume.
gauss	G	Mx/cm^2	Yes	No	Electromagnetic CGS unit of magnetic flux density.
gilbert	Gb	l0/(4*pi)*AT	Yes	No	Electromagnetic CGS unit of magnetomotive force.
grain	gr	.0022857l43*oz	No	No	English unit of weight.
gram	g	None	Yes	No	One thousandth of the SI unit of mass.

Unit	*Identifier*	*Equivalence*	*Prefix Allowed*	*Suffix Allowed*	*Description*
gram-force	gf	9.80665*N*10^-3	Yes	No	The weight of a gram mass when subjected to the mean gravitational attraction of the Earth.
gray	Gy	Unknown	Yes	No	SI unit of absorbed dose in the field of radiation dosimetry.
henry	H	V*s/A	Yes	No	SI unit of inductance.
hertz	Hz	c/s	Yes	No	SI unit of frequency.
horsepower (electric)	hp	746*W	No	No	Archaic unit of power.
hour	h	60*min	No	No	Derived unit of time.
inch	in	2.54*cm	No	No	English unit of length.
conventional inch of mercury	inHg	3386.4*Pa	No	No	Unit equivalent to the pressure required to balance a one inch high column of mercury in a manometer at 32˚F.
conventional inch of water	inH_2O	249.09*Pa	No	No	Unit equivalent to the pressure required to balance a one inch high column of water in a manometer at 4˚C.
ingot	ing	None	No	No	Special SECS generic unit corresponding to the entity of semiconductor manufacture from which wafers are made.
ion	ion	None	No	No	SECS II unique symbol equivalent to an atom that carries an electric charge as a result of losing or gaining electrons.
joule	J	N*m	Yes	No	SI unit of energy, work, and quantity of heat.
kelvin	K	None	No	No	SI unit of temperature. Also referred to as degree Kelvin.
kilopound force	klbf	l000*lbf	No	No	A multiple of the English unit of force or weight.
knot	kn	nmi/h	No	No	Unit of velocity expressed in nautical miles per hour.
lambert	L	(l/pi)*cd/cm^2	Yes	No	CGS unit of luminance. One lumen per square centimeter leaves a surface whose luminance is one lambert in all directions within a hemisphere.
leadframe	ldfr	None	No	Yes	Special SECS generic unit corresponding to a structure for leads which is removed after packaging. The structure may be fixed length or a reel. The unit capacity is specified by the symbol’s suffix, if provided. Otherwise, the capacity is situation-dependent.
liter	l	10^-3*m^3	Yes	No	A metric unit of volume.
lot	lot	None	No	No	Special SECS generic unit corresponding to a grouping of material which is undergoing the same processing operations. The amount of material represented by “1 lot” is situation-dependent.
lumen	lm	cd*sr	Yes	No	SI unit of luminous flux.
lux	lx	lm/m^2	Yes	No	SI unit of illuminance.
magazine	mgz	None	No	Yes	Special, SECS generic unit corresponding to a holder of fixed length leadframes. The unit capacity is specified by the symbols suffix, if provided. Otherwise, the capacity is situation-dependent.
maxwell	Mx	10^-8*Wb	Yes	No	Electromagnetic CGS unit of magnetic flux.
meter	m	None	Yes	No	SI unit of length.
metric ton	t	10^3*kgf	No	No	Unit of weight of force.
mho	mho	S	Yes	No	Previous name for the SI unit siemens.

Unit	*Identifier*	*Equivalence*	*Prefix Allowed*	*Suffix Allowed*	*Description*
micron	um	l0^-6*m	No	No	Alternate name for a micrometer.
conventional micron of mercury	umHg	133.32*Pa*10^-3	No	No	Unit of pressure.
mil	mil	10^-3*in	No	No	English unit of length.
mile	mi	5280*ft	No	No	English unit of length.
conventional millimeter of mercury	mmHg	133.322*Pa	No	No	Unit of pressure.
millimicron	nm	10^-9*m	No	No	Alternate name for nanometer.
minute (plane angle)	mins	deg/60	No	No	One sixtieth of a degree (plane angle).
minute(time)	min	60*s	No	No	Unit of time.
mole	mol	6.02252*l0^23	No	No	SI unit of number of entities within a substance.
month	mo	None	No	No	Unit of time.
nautical mile	nmi	1852*m	No	No	English unit of measurement.
neper	Np	0.1l51*dB	Yes	No	Unit for expressing ratios of power levels.
newton	N	kg*m/s^2	Yes	No	SI unit of force.
nit	nt	cd/m^2	Yes	No	Alternate name for the SI unit of luminance, candela per square meter.
oersted	Oe	79.577472*A/m	Yes	No	Electromagnetic CGS unit of magnetic field strength.
ohm	ohm	V/A	Yes	No	SI unit of resistance.
ounce (avoirdupois)	oz	lbf/16	No	No	English unit of weight
package	pkg	None	No	No	Special SECS generic unit corresponding to an individual entity both as a place for the die to reside and as a completed unit.
pascal	Pa	N/m^2	Yes	No	SI unit of pressure or stress.
percent	%	1/100	No	No	Ratio of parts per hundred.
phot	ph	lm/cm^2	Yes	No	CGS unit of illuminance.
pH	pH	1	No	No	Normalized measure of acidity or alkalinity.
pint (UK)	ptUK	0.56826*1	No	No	United Kingdom version of English unit of capacity.
pint (US dry)	ptUS	0.55061*1	No	No	United States version of English unit of dry capacity.
pint (US liquid)	pt	0.47318*1	No	No	United States version of English unit of liquid capacity.
plate	plt	None	No	Yes	Special SECS generic unit corresponding to a temporary fixture used to hold die during assembly operations. The unit capacity is specified by the symbol's suffix, if provided. Otherwise, the capacity is situation-dependent.
poise	P	36*N*s/m^2, or 36*kg/(m*s)	Yes	No	A CGS unit of viscosity equal to the viscosity of a fluid that would require a shearing force of one dyne to move a square centimeter area of either of two parallel layers of fluid one centimeter apart with a velocity of one centimeter per second relative to the other layer, with the space between the layers being filled with fluid.
pound	lb	0.0310810*slug	No	No	English unit of mass.
pound-force	lbf	4.4482217*N	No	No	English unit of force or weight.
poundal	pdl	0.0310810*lbf	No	No	Force required to accelerate a one pound mass at one ft/s^2.
parts per million	ppm	1/I0^6	No	No	Ratio of parts per million.

Unit	*Identifier*	*Equivalence*	*Prefix Allowed*	*Suffix Allowed*	*Description*
quart (UK)	qtUK	1.1365*1	No	No	United Kingdom version of an English unit of capacity.
quart (US dry)	qtUS	1.1012*1	No	No	United States version of an English unit of dry capacity.
quart (US liquid)	qt	0.94635*1	No	No	United States version of an English unit of liquid capacity.
rad	rd	10^-2*Gy	Yes	No	A unit of absorbed dose in the field of radiation dosimetry.
radian	rad	None	Yes	No	SI unit of plane angle.
rem	rem	10^-2*Sv	Yes	No	A unit of dose equivalent in the field of radiation dosimetry.
revolution	r	c	No	No	One complete cycle of a rotating body.
roentgen	R	Unknown	No	No	A unit of exposure in the field of radiation dosimetry.
second (plane angle)	sec	mins/60	No	No	One sixtieth of a minute of a degree.
second (time)	s	None	Yes	No	SI unit of time.
siemens	S	1/ohm	SI unit of conductance.	Yes	No
sievert	Sv	Unknown	Yes	No	SI unit of dose equivalent in the field of radiation dosimetry.
slug	slug	14.5939*kg	No	No	English unit of mass.
standard cubic centimeter per minute	sccm	cc/min	No	No	A unit of flow equivalent to one cubic centimeter of a gas at standard temperature and pressure flowing past a point in one minute.
standard liter per minute	slpm	1/min	No	No	A unit of flow equivalent to one liter of a gas at standard temperature and pressure flowing past a point in one minute.
steradian	Sr	Unknown	Yes	No	SI unit of solid angle.
stilb	sb	cd/cm^2	Yes	No	A CGS unit of luminance.
stokes	St	P*cm^3/g	Yes	No	A CGS unit of kinematic viscosity.
substrate	substrate	None	No	No	Special SECS generic unit corresponding to the entity of material being operated on, processed or fabricated.
tesla	T	N/(A*m) or Wb/m^2	Yes	No	SI unit of magnetic flux density (magnetic induction).
therm	thm	10^5*Btu	No	No	An English unit of energy.
ton (short)	ton	2000*lbf	No	No	English unit of weight.
torr	torr	mmHg	Yes	No	Pressure unit. Alternative name for millimeters of mercury.
tube	tube	None	No	Yes	Special SECS generic unit corresponding to a holder of packages arranged in a flow. The unit capacity is specified by the symbol’s suffix, if provided. Otherwise, the capacity is situation-dependent.
var	var	Unknown	Yes	No	SI unit for reactive power.
volt	V	W/A	Yes	No	SI unit of voltage.
wafer	wfr	None	No	No	Special SECS generic unit corresponding to the entity of material on which semiconductor devices are fabricated.

Unit	*Identifier*	*Equivalence*	*Prefix Allowed*	*Suffix Allowed*	*Description*
waferframe	wffr	None	No	Yes	Special SECS generic unit corresponding to a temporary fixture for wafers. The unit capacity is specified by the symbol's suffix, if provided. Otherwise, the capacity is situation-dependent.
watt	W	J/s	Yes	No	SI unit of power.
watthour	Wh	3600*J	Yes	No	Unit of energy.
weber	Wb	V*s	Yes	No	SI unit of magnetic flux.
year	yr	None	No	No	Unit of time.

NOTICE: SEMI makes no warranties or representations as to the suitability of the standard set forth herein for any particular application. The determination of the suitability of the standard is solely the responsibility of the user. Users are cautioned to refer to manufacturer's instructions, product labels, product data sheets, and other relevant literature respecting any materials mentioned herein. These standards are subject to change without notice.

The user's attention is called to the possibility that compliance with this standard may require use of copyrighted material or of an invention covered by patent rights. By publication of this standard, SEMI takes no position respecting the validity of any patent rights or copyrights asserted in connection with any item mentioned in this standard. Users of this standard are expressly advised that determination of any such patent rights or copyrights, and the risk of infringement of such rights, are entirely their own responsibility.

RELATED INFORMATION 1
APPLICATION NOTES

NOTICE: The material contained in this Related Information section is not an official part of SEMI E5 and is not intended to modify or supersede the official standard. Rather, these notes describe possible methods for implementing the protocol described by the standard and are included as reference material. The standard should be referred to in all cases. SEMI makes no warranties or representations as to the suitability of the material set forth herein for any particular application. The determination of the suitability of the material is solely the responsibility of the user.

R1-1 The General Node Transaction Protocol

R1-1.1 This application note has been moved to follow SEMI E4 (SECS-I) as Application Note A7.

R1-2 Some Suggested Message Usage

R1-2.1 The number of messages implemented and the choice of messages are greatly influenced by the actual function of the equipment. To illustrate which messages might be appropriate, the following suggestions are offered for a variety of different types of equipment capabilities. It is assumed that the minimum message sets S1,F1; F2 and S9,F1; F3; F5; F7 are always implemented.

R1-2.2 For equipment which makes nondestructive in-process measurements using a fixed measurement procedure, it may be necessary only to implement S6, F9 to send the data according to a fixed format upon measurement. Optional remote control can be added with S2,F21 to start a measurement.

R1-2.3 If the equipment has a variety of measurement routines, it might be desirable to respond to S1,F5 with S1,F6, which would give the host a brief report of the test being made. The test can be thought of as a process program. Accordingly, S7,F1 and S7,F2 could be used for the host to select the program. The same messages in conjunction with S7,F3 and S7,F4 could load a new test procedure. S7,F19 could be used by the host to find out what tests were available.

R1-2.4 Some equipment which automatically processes wafers in a batch might make more extensive use of S1,F5 or S1,F3 and might include some error reporting on S5,F1. More sophisticated equipment may include some trace features with S2,F23 and S6,F1 or some control loop tuning by S2,F15.

R1-2.5 Equipment using in-line wafer movement could utilize Stream 4, S1,F9, and Stream 3 to keep track of wafers.

R1-2.6 Stream 7,F9 through F19 can be used to manage a local backup of process programs should the host fail for a short while.

R1-2.7 Microprocessor equipment can benefit from features such as provided by Stream 8 and S2,F1 through F12 which allow managing and servicing the software routines.

R1-2.8 Equipment, including a CRT, might elect to make it available to the host by including Stream 10 messages.

R1-2.9 Some equipment, such as functional testers, might have sufficient need to undertake remote file usage such as provided in Stream 13.

R1-2.10 These brief suggestions serve to illustrate that the final choice of the messages included in a given equipment depends upon its function. The messages can be viewed as interface features in the same way that other parts of the equipment are viewed as processing features or wafer handling features.

R1-3 Notes on SECS-II Data Transfers

R1-3.1 *Introduction*

R1-3.1.1 There are two primary ways to send and ask for data in SECS-II. One of these is to use the trace feature and the other is to use the event reporting method. The purpose of this note is to describe the intended operation of the messages described in the existing standard. Discussion of completeness or need for other reporting methods is left for task force and committee work.

R1-3.2 *Trace Data Collection and Reporting*

R1-3.2.1 This method of collecting data is intended for engineering and developmental use rather than routine data collection for production. The features included allow the collection of relatively large amounts of real time data over a finite amount of time. The data is generated at regular time intervals as determined by a timing generator in the equipment. The function of the host is to set up the trace and then to subsequently store the data as it is received from the equipment. It is assumed that some host resident applications will exist to analyze the data either as it is received or at some later time.

R1-3.2.2 The trace feature will only exist in equipment which implements it.

R1-3.2.3 The host sets up the trace with the S2,F23-24 transaction. At this time, the host assigns several important parameters. TRID is the trace request ID and is used later when the equipment sends back the data. Every trace data reply includes the TRID corresponding to the request that set up the trace. Several traces can theoretically be done at the same time if the equipment allows it and the TRID keeps the data for each trace distinct from other trace data. DSPER is the data sample period and is used to indicate how often the specified parameters should be sampled (that is, have their values saved). TOTSMP is the total number of samples to be made. Since TOTSMP is finite and the number of parameters is specified in this transaction, the host can reserve adequate file space for the reported data if required. The REPGSZ is the reporting group size and corresponds to the number of time samples that should be combined into one message prior to transmission. Thus, if it is desired to sample one or two parameters every second but only send those samples to the host once a minute, the reporting group size would be 60. Having the reporting group size parameter allows the host to have some control over how often it may be interrupted to handle the trace data. However, as presently defined in the standard, the trace data is reported as a single block message (S6,F1), which restricts both the number of status variables or the number of samples which can be combined into one message. The equipment may be able to accommodate this in several ways, to be described shortly.

R1-3.2.3.1 The last element in the trace initialize request is a list of status variable IDs. The trace command only allows tracing variables that have been declared and are known to the equipment as status variables. It is assumed that the equipment will report the variable values in the same order as specified in the trace request. This will allow the host to identify the values returned.

R1-3.2.4 The trace data send message, S6,F1-2, sends the trace data as a single block message to minimize the overhead in reporting data. The TRID is the first item and identifies the request that asked for the data. The next item is SMPLN, the sample number of the last sample in this message, should more than one sample be combined. The next item is STIME which is the time of the last sample in this message. These three items are followed by the list of values. If five (5) values were requested with a reporting group size of 5, then 25 values would be in this list, each group of 5 in the same order as requested and in the time order sampled. Some flexibility is allowed in how the equipment chooses to report the data to the host when the reporting group size exceeds one block of data. The equipment can send the data when it has a complete block or it can reject the request when it is set up.

R1-3.3 *Event Driven Data Reporting*

R1-3.3.1 The second major type of data reporting is initiated by some event in the equipment. Data reporting is often desired after some event such as the completion of a measurement, the completion of a lot, the completion of a wafer, the occurrence of a special event command in the recipe, or some other action which is determined by the equipment. The two aspects involved in event driven data reporting are the control of which events cause data to be sent to the host and the formatting of the data sent to the host.

R1-3.3.2 It is assumed that a set of events has been established for a particular piece of equipment and that each event can produce a report of some sort. It is further assumed that a set of equipment constants exist in the machine such that they have control over the optional reporting of the events. For example, a boolean constant may exist for each possible reporting event, and when the host sets the constant to a logical 1, the corresponding event will cause a report to be sent to the host, when the host sets the constant to logical 0, the event will not send a report. S2,F15-16 in the equipment constant send transaction can be used to control the event reporting.

R1-3.3.3 When an event causes data to be sent to the host, there are several possible conversations, depending upon the length of the data and the complexity of the formatting. S6,F3-4 is the basic data transaction, which has a very general format. The parameters provide for an overall name, DATAID, for the type of data; a collection event identification, CEID, should there be more than one event that could generate the same type of data; and a list of data sets. This structure allows reporting such data as the measurements taken on each of the wafers in a lot. The measurements on each wafer make up one data set, and the list of data sets is the whole lot. The collection event would be the lot completion, and the data ID might be film thickness measurements. Other types of organizations are possible, depending upon the type of data being sent. The same type of data might be produced by a different CEID, such as the forced termination of the lot. This collection ID would indicate that the data is incomplete for the lot. Within each data set, each data value is reported as a pair of items, one item being the name of the value and the other being the value.

R1-3.3.4 Since many simple measuring devices have only a very few types of data sets, an alternative data format is provided in S6,F9-10, which has the same form as S6,F3-4 but does not require the value name in the data set. Instead, the order of the values is fixed format for the particular DSID in that particular equipment.

R1-3.3.5 When either of the above data messages is long enough to require multiple blocks, it must be preceded by S6,F5-6 to gain permission to send a multiple block message.

R1-3.3.6 The last data control transaction is S6,F7-8, which is initiated by the host and causes a specified DATAID to be sent to the host. The implementation of this function is highly equipment-dependent. In essence, it is equivalent to the host causing an event that triggers the sending of the data. Since the equipment may be generating the data, the actual data sent depend upon the equipment implementation. The equipment can respond with a zero length item if no data can be sent.

R1-3.4 *Event Reporting*

R1-3.4.1 The third major type of data reporting is similar to that described in A3.3 above, with the following enhancements:

1. Contents of data reports are not limited to DVVALs, but may include SVs or even ECVs.
2. Contents of data reports are user programmable.

R1-3.4.2 It is assumed that the equipment vendor supplies a list of all "events" identified within a particular piece of equipment. A Collection Event Identifier (CEID) must be specified for each of these events. It is further assumed that the vendor supplies a list of all available variables within the machine. This includes Status Variables (SVs) and their identifiers (SVIDs), Equipment constants (ECVs) and their identifiers (ECIDs), and Data Values (DVVALs) and their identifiers(DVNAMEs). Each of the identifiers must be unique. The term VID (Variable Identifier) encompasses all SVIDs, ECIDs, and DVNAMEs. Likewise, the term V (Variable Data) encompasses all SVs, ECVs, and DVVALs.

R1-3.4.3 Note that a Variable (V) may be a list (format code 0). This provides for referencing a group of related data values with one identifier. Consider the following:

```
VID1 = 1 zone 1 temperature ID  Format 32
V1     zone 1 temperature value Format 52
.
.
VIDn = n zone n temperature ID  Format 32
Vn     zone n temperature value Format 52
VIDx = x all temperatures ID    Format 32
Vx  L,n                         Format 0
      1. L,2
           1. <VID1>
           2. <V1>
      .
      .
      n. L,2
           1. <VIDn>
           2. <Vn>
```

Any V in a list may also be a list (for nesting).

R1-3.4.4 In a typical initialization sequence, the host would define all the desired programmable data reports with S2,F33/S2,F34 (Define Report) transactions. Then S2,F35/S2,F36 (Link Report/Event) transactions would be used to define which reports are to be made by the equipment upon specific events (CEIDs). An individual report may be linked to more than one event. At this point the host may request reports with the S6,F15/S6,F16 (Report Request) transactions to obtain initial report data and/or to verify reports as defined and linked. Finally, the desired reports would be enabled by the host with S2,F37/S2,F38 (Enable/Disable Event Report) transactions.

R1-3.4.5 There are two methods for the equipment to send event reports to the host. S6,F13 includes the Variable Identifier (VID) with each Variable Data item (V). S6,F11 is a shorter form, without the identifiers; some users prefer this form to reduce message size.

R1-3.4.6 When any message is long enough to require multiple blocks, it must be preceded by an inquire/grant transaction.The DATAID parameter is used only to link the inquire/grant transaction with a multiblock message. This linkage is to alleviate problems in the case of interleaved messages. A unique value for DATAID must be used for each Inquire/Grant/Send/Acknowledge conversation (similar to the use of SYSBYTES in SECS-I). The DATAID parameter should not be used for any other purpose.

R1-4 Process Programs

R1-4.1 *Introduction*

R1-4.1.1 Two forms of process programs are supported by SECS-II: unformatted and formatted. The contents of an unformatted process program conform to no set standard. The format of the program is defined by the vendor of the equipment and probably bears no similarity to the format used by other vendors for their equipment. Because special programming would be required at the host to understand the equipment's unique data format, the process program is most likely generated at the machine and the host is only used as a data repository, saving the foreign data for later retransmission to the equipment. S7,F3 and S7,F6 are the SECS-II messages used to move unformatted process programs between host and equipment.

R1-4.1.2 Unformatted process programs were the original accepted means for moving processing instructions between host and equipment under SECS-II. However, the inability of a host to generate process programs for its subordinate machines was quickly recognized as a severe problem. As a result, the formatted process program and its associated transactions were added. Five transactions are provided under SECS-II for handling formatted process programs: S7,F23-24, S7,F25-26 allow movement of process programs between host and equipment; S7,F21-22 originates at a machine and provides a host with the information it needs to generate a process program for that machine; S7,F27-28 allows the equipment to tell the host whether or not the contents of the formatted process program received from the host are valid; and S7,F31-32 provides the host with the ability to ask the equipment to check the validity of a process program without actually downloading the program into the machine for production use.

R1-4.2 *Normal Sequence of Operations*

R1-4.2.1 Formatted process programs may be generated at a host or machine. The actions taken to generate one in a machine are left to the equipment manufacturer. If the process program is created at a host, a sequence of operations is assumed.

R1-4.2.1.1 Once the host's process program generator has been invoked and has been told for which machine a process program is to be created, the host editor must obtain a copy of the process capabilities data for that machine. The information may already be available on the host or it may be obtained directly from the machine. In either case, the information originates at the equipment and is obtained using S7,F21-22. (See Section R1-4.4 for additional information.)

R1-4.2.1.2 With the machine's process capabilities in its possession, the process program editor may proceed with creating the desired process program. At the conclusion of the editing session, the new machine process program will either be saved at the host or sent directly to the machine for storage and/or use.At this point, the process program is known to satisfy a number of constraints, but it is not necessarily completely acceptable to the machine due to interrelationships of the process program data which are too complex to be described in the machine process capabilities data. The host at any time may verify that a process program is truly valid by sending the process program to the machine and asking it to check the process program and tell the host whether or not the process program is, in fact, correct. If not correct, the equipment is expected to provide information on what data in the process program is unacceptable. This action is accomplished through S7,F31-32. This transaction is equivalent to S7,F23-24, with one important exception, the machine is not to do anything with the process program received under S7,F31 except acknowledge that it got the message (S7,F32) and, as soon as it is able, respond with S7,F27, which provides the host with information on the validity of the process program. In this way, a new version of a process program already held by a machine may be checked for validity without affecting the operation of the machine (i.e., a newer version of a particular process program may be checked while an older version is simultaneously being used by the equipment for material processing).

R1-4.2.1.3 At some point, a host resident process program will be required by the equipment for material processing. Transfer of a program may be accomplished in either of two ways. First, the host may initiate transfer by transmitting S7,F23. In this case, immediately upon reception of the message, the equipment is required to respond with S7,F24, which tells the host that the process program arrived and whether or not the process program is accepted for further processing by the equipment. The second means is for the equipment to initiate the transfer by asking for a process program using S7,F25. In this case, the host will send the process program to the equipment or tell the equipment it is unable to satisfy the request. S7,F23 may also be used by a piece of equipment to transfer a process program to its host for archiving. In this case, the host will respond with S7,F24 and an appropriate completion code. Likewise, a host may request a process program transfer from its machine using S7,F25. The machine will respond with S7,F26, which will contain the process program or an error indication.

R1-4.2.1.4 Following reception by the equipment of the process program, it is the machine's responsibility to check the contents of the process program for validity and respond to the host with a S7,F27 message formatted with the appropriate information about the just received process program. To complete the process program exchange transaction, the host will acknowledge the S7,F27 message with S7,F28. What is done with the process program once accepted and checked for validity is dependent on the state of the process equipment.

R1-4.3 *Equipment Process Capabilities Data*

R1-4.3.1 The underlying assumption of SECS-II formatted process programs is that processing instructions for equipment can be expressed as sequences of commands with parameters. Commands are integer codes which tell the machine what to do. The parameters of each command are numeric (integer or floating point) values, Boolean values, or text strings which specify how to carry out the particular command. This provides a very flexible structure for building process programs but does not provide the specific information (code values, types and number of parameters, legal parameter values, etc.) required by a host system to generate a process program for a particular piece of equipment. Under SECS-II, this information is provided to a host via the machine's Equipment Process Capabilities Data or PCD.

R1-4.3.2 A PCD provides three levels of information global data pertaining to the entire process program; definition of each possible command understood by the machine; and definition of each command parameter. Global process program definition data consist of MDLN, SOFTREV, CMDMAX, BYTMAX, and the list of command descriptors.

R1-4.3.3 MDLN and SOFTREV provide the same data to the host as the equipment's response to the S1,F1 host interrogative, "Are you there?" They are included in the PCD to provide a means of distinguishing between PCDs for different machines and revisions of PCDs for the same piece of equipment. Also, when a process program is generated, the MDLN/SOFTREV values of the PCD are provided in the process program to allow the machine an unambiguous method of determining if the process program was generated from a PCD it understands.

R1-4.3.4 BYTMAX and CMDMAX are two integer values which allow the equipment to limit the size of the process program which will be generated. BYTMAX specifies the maximum number of bytes a process program may occupy. CMDMAX specifies the maximum number of commands which may appear in the process program. Either value may be zero, which indicates that no maximum limit is being imposed by the equipment.

R1-4.3.5 The PCD command list identifies (in no particular order) each of the unique operations its associated machine is capable of performing. These operations may correspond to processing operations of the equipment (bake, spin), initialization of equipment components (set beamline controls), definition of data values referenced by later commands (define bond coordinates or inspection points), or even "pseudo-operations," which allow conditional execution of the process program (go to X; if temperature out of range, then go to y; repeat ramping until speed 200; etc.).

R1-4.3.6 Each command in the PCDlist has a number of data values associated with it which provide the host with the command's personality. These are CCODE, CNAME, RQCMD, BLKDEF, BCDS, IBCDS, NBCDS, ACDS, IACDS, NACDS, and the commands parameter list.

R1-4.3.7 CCODE defines the unique numeric code which the equipment recognizes as representing the command being defined. CNAME is a text string which hopefully describes the function of the command. The string must be unique for each command since humans generating process programs at the host will use them, and the host process program generator will translate the CNAME to the corresponding CCODE.

R1-4.3.8 RQCMD. This Boolean value allows equipment to specify whether or not a command must appear at least once within their process program. If true, the command must be used. If RQCMD is set false, the command may or may not be used in the process program at the discretion of the person creating the process program.

R1-4.3.9 In addition to the information the PCD provides on allowed data content within a process program, it also can provide information to the host on possible interdependencies between the commands. Specifically, through the PCD the host can know such things as: command code A must appear before command code B; command code A must come after command code D; command code A must immediately precede command code X; command code A must not come before command code E; command code A must not come after command code F; and/or command code A must immediately come after command code T. Each of the PCD entries, BCDS (before codes), ACDS (after codes), IBCDS (immediately before codes), IACDS (immediately after codes), NBCDS (not before codes), and NACDS (not after codes), is a SECS item which may contain one or more command codes. Each particular item defines the relation to be satisfied. The elements of the item identify the command codes which are to satisfy the relation with the command being defined. A zero length item indicates that no restrictions apply for that type of checking. For example, if the values of the various fields take on the values shown in Figure R1-1, the host process program editor will assure that the TEST command (code 10) will occur before commands with codes 5, 6, and 8; that it will come after commands with codes 100 and 2; that TEST will not appear after the command with code 20; and that each occurrence of command code 3 will have a TEST command immediately before it, subject to the block checking limitations described elsewhere.

CNAME = TEST

CCODE = 10

BCDS = 5,6,8

IBCDS = 3

NBCDS = none

ACDS = 100.2

IACDS = none

NACDS = 20

Figure R1-1

R1-4.3.10 Associated with before/after checking is the concept of a block which allows setting of limits on before/after checking. A block consists of a start block command, a block terminator command and possibly body commands, commands which are included between the start and terminator commands. There are no specific command codes for start, or terminator commands in SECS-II formatted process programs. Instead, being a start block, terminator block, or body command is merely an attribute of each command defined in the PCD. The field BLKDEF defines this attribute for each command. A positive one indicates the command starts a new block. Zero indicates that the command is a body command and neither starts nor terminates a block. A value of negative one indicates that the command is a terminator command.

R1-4.3.11 Before/after checking for a particular command is performed only with other commands within the same block. To be within the same block, a command must have the same nesting level as the command of interest or the command must be a contained block.

R1-4.3.11.1 The example data in Figure R1-2 shows six grouping of commands for before/after checking: (A,B',N), (B,C,D',G',M), (D,E,F), (G,H',L), (H,I',K), (I,J). A letter followed by an apostrophe (') indicates a block which has been collapsed to a command and has the before/after attributes of its start block command. Note that body and terminator commands occur in only one grouping, while block start commands occur in two. Also, note that the outermost block is assumed to begin with the first command of the process program and to end with the last command.

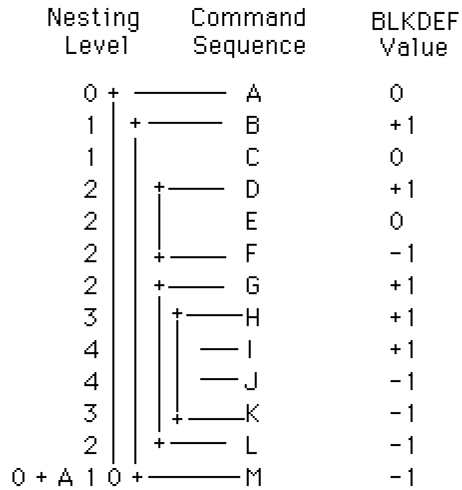

Figure R1-2

R1-4.3.12 Each command's parameter list defines the parameters required by the equipment to carry out each particular command. The order in which each parameter descriptor appears in the PCD parameter list also defines the order in which parameters will appear in a process program command parameter list. Each parameter is one of three possible types: numeric, text, or Boolean.

R1-4.3.13 Regardless of parameter type, the first four elements of any parameter descriptor list are the same. The first field, PNAME, specifies the text string which names the parameter. This data will be displayed by a host when prompting a human for the parameter data. The second field, RQPAR, specifies if the value must be specified at the time the process program is generated (true) or if specifying the data is optional (false). The third field, PDFLT, identifies the type of data to be accepted for this parameter as well as providing default values to include in the process program if the RQPAR is false and no data is input for the parameter when the process program is generated. PDFLT will have zero length if no default value is provided.

R1-4.3.14 The final field, PMAX, specifies the maximum length of the parameter data placed in a process program. For numeric and Boolean data, it specifies the maximum number of data entries in the SECS-II item. For a string parameter, it specifies the maximum number of characters acceptable to the machine. In either case, negative values are invalid and a value of zero indicates there is no length restriction.

R1-4.3.15 For numeric and Boolean parameters which are multi-valued items, usage of PDFLT becomes a bit more complex. In these cases, PDFLT may also be a vector of values. When default values are to be included in a process program, the entry of the default vector in the same ordinal position as the parameter entry requiring the default is used. If the parameter is allowed to have N entries but only M defaults are provided, the last N-M parameter entries will have no defaults.

R1-4.3.16 If the numeric or Boolean vector parameter is required to be entered, PDFLT will contain no default data values, but dummy values must be provided so that the length of the item specifies the minimum number of entries the equipment expects to receive for the parameter. If this minimum number of entries exceeds the maximum number of entries allowed for the parameter (PMAX), then only PMAX entries will be provided in the process program.

R1-4.3.17 Numeric parameters may be any of the SECS recognized floating point or integer data types. PDFLT identifies the particular type. ULIM and LLIM will be of the same data type and specify the range of legal values for the parameter (LLIM ≤ x ≤ ULIM). UNITS is a character string formed according to E5, Section 12, which specifies the expected units of measure of the numeric value. RESC specifies whether the resolution of the data item to be entered is to be in terms of a fundamental increment or significant digits. In the case of the former, RESV will be of the same type as the expected parameter and will specify the base increment. In the case of the latter, RESV will be an integer and will specify the number of significant digits to accept for the parameter.

R1-4.3.18 In addition to the standard fields described above, string parameter descriptors have one unique field. This field provides a set of template strings. A text parameter will be assumed valid by a host process program generator if it matches one of the template strings. A match occurs if the input string is at least as long as the corresponding template and each position of the template and data strings match. A null string specified as a template will result in a match with any data string. A null data string will match only a null template string. A null template list indicates all strings are acceptable to the equipment.

R1-4.4 *Equipment Capabilities Descriptor Availability*

R1-4.4.1 Ideally, each piece of equipment should be able to respond to a host PCD request at any time. However, inasmuch as an equipment's PCD may be rather large and the equipment may have limited storage capacity, constant availability may be impossible. In these cases, some compromise will have to be made such as making it available only at machine initialization or when idle. In extremely severe cases, an equipment manufacturer may have to provide the PCD data with the rest of the machine documentation, requiring his customer to manually enter the data into his host system.

R1-4.4.2 In light of this difficulty, host systems should maintain copies of PCD's for each piece of equipment under its control and not expect to be able to obtain the PCD from the machine whenever it is required. Doing so, in fact, will permit more flexible process program development in the host, allowing creation of process programs even when equipment is not online and encourage the use of formatted process programs by equipment manufacturers.

R1-5 Suggested Baseline SECS Equipment Implementation

R1-5.1 *Purpose and Scope*

R1-5.1.1 This document provides a recommendation prepared by the Rigid Disk Subcommittee for generating a baseline implementation of the SECS (SEMI Equipment Communication Standard) standards on production process and test equipment. This document is not a tutorial to aid in understanding SECS but rather serves as an introduction to the requirements of SECS, and a brief guide to the selection of SECS messages for equipment. Actual system requirements of many implementations are beyond the scope of this document. The full standards, SEMI Equipment Communications Standard I (SECS-I), SEMI E4 and SEMI Equipment Communications Standard II (SECS-II), SEMI E5 should be consulted by all users.

R1-5.2 *Introduction*

R1-5.2.1 The SECS standards are an existing and developed set of communication standards currently used by the semiconductor and other industries to support automated production. The standards provide a means for communicating information and control between production equipment and a "host" computer. This transfer of information and control can be used to provide production tracking and location of WIP (Work-In-Process), scheduling of WIP and control of material transfer at the equipment. The process measurements and records can be used to provide process engineers with a database for statistical process control. SECS messages are appropriate to a wide variety of applications, including measurement, processing, and material transport equipment.

R1-5.3 *SECS-I Standard*

R1-5.3.1 SECS-I defines the lower protocol layers of a point-to-point interface between equipment and a host computer system. The standard requires a simple, well understood physical interface, RS-232. The SECS-I protocol allows the equipment control over the protocol: the equipment is the master of the link and can initiate the transfer of a message to the host. Likewise, the equipment can regulate the receipt of a message by its response to a handshake to receive the message. Thus, the interface takes place at the convenience of the equipment, and equipment with very limited computer resources can still support the standard.

R1-5.4 *SECS-II Standard*

R1-5.4.1 SECS-II defines the higher layer in the protocol, including message content, structure, and data types and their formats. SECS-II defines messages in sets with related functions called Streams. The actual content of the messages are specific to an application, but it is possible for a properly designed host software system to unpack a message and present the data in a meaningful way with no prior definition as to the content. Equipment need only implement those messages appropriate to meet its system requirements; thus, very simple equipment will require implementation of few messages.

R1-5.4.2 Application Note A2 contains some suggestions for message utilization, including some information on minimum message sets. It is the intent of this report to expand in a somewhat different direction, to identify those messages which typically constitute a sufficient set given a selected equipment function.

R1-5.4.3 To select a message set, the requirements for the equipment must be identified. This requirement, in turn, determines a message set. Identified below are a number of types of equipment requirements and a baseline set of messages supporting those requirements. The message sets identified are baseline recommendations. Actual equipment implementations may need more messages than those specified here to satisfy all system requirements. The published standards should be consulted for all applications. Issues such as handling of multi-block messages, optional replies, and others are described in detail in the SEMI specifications and are not a topic of this baseline recommendation.

R1-5.4.4 The implementation for a specific equipment type begins by specifying which tasks that equipment is required to perform from the following list, and then studying the expanded descriptions for those tasks chosen in the SECS-II implementation section.

Typical Tasks for Measurement and Process:

1. Measurement or Action Reports
2. Equipment Alarm Reports
3. Remote Request for Equipment Condition or State
4. Operator Interface to the Host
5. Remote Access to Process Programs
6. Remote Commands

Typical Tasks for Material Control and Transport:

1. Material Status Information
2. Material Transport Control

Additional Tasks for Special Situations:

1. File Transfer

R1-5.5 *Baseline SECS Implementation Recommendations: SECS-I*

R1-5.5.1 The baseline SECS-I requirement for equipment is to fully implement the SECS-I protocol as defined in SEMI E4. This requirement applies to all SECS compatible equipment independent of the equipment's function. The flow chart (Figure 2 of SEMI E4) illustrates the block transfer protocol of SECS-I. The body of the SECS-I standard describes protocol timeouts and other requirements that are essential components of the specification.

R1-5.6 *Baseline SECS Implementation Recommendations: SECS-II*

R1-5.6.1 The SECS-II implementation begins with a choice or selection of messages for the equipment. In order for equipment to meet the baseline requirements for a viable SECS-II interface, the equipment must be capable of generating a certain set of messages and recognizing another set. The messages required for either set will depend on tasks of equipment and on system requirements for production and process control by the host. Equipment must accept S1,F1 and send S1,F2. Note that implementation in the reverse direction is optional. In order to ensure a viable data communication link, certain messages are required:

Messages for All Equipment — Required by SECS-II

a. Messages Generated by the Equipment

S1,F2:On Line Data

S9,F1: Unrecognized Device ID

S9,F3: Unrecognized Stream Type

S9,F5: Unrecognized Function Type

S9,F7: Illegal Data

b. Messages Recognized by the Equipment

S1,F1: Are You There Request

R1-5.6.2 In addition to those messages which are required, the following are strongly recommended. These messages are used for diagnostic purposes:

Messages for All Equipment — Strongly Recommended

A. Messages Generated by the Equipment

1. S2,F25: String Diagnostic Request

B. Messages Recognized by the Equipment

1. S2,F26: String Diagnostic Data

R1-5.6.3 Note that messages are identified by a "stream" number and a "function" number. All primary messages have an odd-numbered function, and the corresponding reply is the next consecutive even-numbered function. Thus, many messages are paired. For example, the Stream 1, Function 1 (S1,F1) message generated by either host of equipment has the reply of Stream 1, Function 2 (S1,F2).

R1-5.6.4 *Reports of Measurements or Process Actions* — The simplest task for measurement and process equipment is to report their measurements or actions to the host. This report could also include equipment setup parameters and sample identification. The ability to accurately transfer this data from equipment to a factory host is of prime importance in automating production. Depending on the equipment requirements, the function can be far greater.

R1-5.6.4.1 *Measurements or Action Reports* — Given an equipment requirement to relay data to a host computer, the equipment must handle one or both of the following messages as defined by SEMI E5.

S6,F3: Discrete Variable Send

S6,F9: Formatted Variable Send

R1-5.6.4.2 *Equipment Alarm Reports* — These messages are unsolicited by the host, they transmit information warning of conditions threatening personal safety, equipment safety, or out of limit equipment parameters which may cause harm to the product or indicate equipment malfunction:

S5,F1: Alarm Report Send

R1-5.6.4.3 *Remote Request for Equipment Condition or State* — This is the method where the host requests data from the equipment.

Messages Transmitted to the Equipment:

S6,F7: Data Transfer Request

Messages Transmitted by the Equipment in Reply:

S6,F8: Data Transfer Data

R1-5.6.4.4 *Operator Interface to the Host* — Many equipment systems will have an interactive operator's console; some will have computer terminals used for this purpose. SECS-II has a message type to transfer text from host to the equipment console. There are also messages through which equipment operators may transmit text from their console, directly to the host. This text may include desired information which is not accessible by the equipment computer directly.

Message Transmitted by the Equipment:

S10,F1: Terminal Request

Message Recognized by the Equipment:

S10,F3: Terminal Display, Single

R1-5.6.4.5 *Remote Access to Process Programs* — SECS-II defines a means for storing or retrieving equipment process programs. This function allows upload and download of such programs through the SECS interface. By this means, programs for equipment may be archived in the host computer system. Either the host or the equipment may request the transfer of a Process Program from the other. To do this:

The Requestor transmits:

S7,F5: Process Program Request

The Sender of the Process Program replies:

S7,F6: Process Program Data

In addition, either host or equipment may initiate sending the Process Program. In this case:

The sender transmits:

S7,F1: Process Program Load Inquire,

Receiver replies with:

S7,F2: Process Program Load Grant,

Sender transmits:

S7,F3: Process Program Send,

The receiver answers:

S7,F4: Process Program Acknowledge.

R1-5.6.4.6 *Remote Commands* — The host can initiate a command to the equipment in a manner similar to an operator pressing a button.

The host transmits:

S2,F21: Remote Command Send,

The equipment replies:

S2,F22: Remote Command Acknowledge.

R1-5.6.4.6.1 Execution of the remote command may cause the equipment to transmit a message to the host at a time greater than the reply time required for the S2,F22 message. If this type of reply is desired, the

S6,F3: Discrete Variable Data Send may be used to transfer data to the host.

R1-5.6.5 *Typical Tasks for Material Control and Transport*

R1-5.6.5.1 *Material Status Information* — The host may query the equipment for material-in-process information. The information is transmitted only as a answer to a host request.

The equipment recognizes:

S3,F1: Material Status Request.

The equipment transmits:

S3,F2: Material Status Data.

R1-5.6.5.2 *Material Transport Control* — The SECS-II protocol includes the means to affect automated transfer of material from one SECS-compatible device to another. Baseline compatibility requires the equipment to perform a simple material transfer process, an actual implementation may require means for graceful error recovery as well. This recommendation does not include messages to handle error conditions.

Receiving Material:

Equipment Recognizes S4,F1:

Ready to Send Material

Equipment Transmits:

S4,F3: Send Material,

S4,F5: Handshake

Sending Material:

Equipment Transmits:

S4,F1: Ready to Send Material

Equipment Recognizes:

S4,F3: Send Material,

S4,F5: Handshake Complete

Equipment Recognizes:

S4,F2: RTS Acknowledge

R1-5.7 *Conclusion* — The baseline requirements for equipment using the SECS standards includes all of SECS-I and a limited selection of messages from SECS-II. The choice of SECS-II messages, and data contained therein, is dictated by the equipment and system requirements. The benefits in using the standards are many, including support for growth in equipment function, and standardization needed for effective automation. The results include automated process monitoring and all of the associated benefits.

NOTICE: SEMI makes no warranties or representations as to the suitability of the standard set forth herein for any particular application. The determination of the suitability of the standard is solely the responsibility of the user. Users are cautioned to refer to manufacturer's instructions, product labels, product data sheets, and other relevant literature respecting any materials mentioned herein. These standards are subject to change without notice.

The user's attention is called to the possibility that compliance with this standard may require use of copyrighted material or of an invention covered by patent rights. By publication of this standard, SEMI takes no position respecting the validity of any patent rights or copyrights asserted in connection with any item mentioned in this standard. Users of this standard are expressly advised that determination of any such patent rights or copyrights, and the risk of infringement of such rights, are entirely their own responsibility.

Copyright by SEMI® (Semiconductor Equipment and Materials International), 3081 Zanker Road, San Jose, CA 95134. Reproduction of the contents in whole or in part is forbidden without express written consent of SEMI.

SEMI E30-1103
GENERIC MODEL FOR COMMUNICATIONS AND CONTROL OF MANUFACTURING EQUIPMENT (GEM)

This standard was technically approved by the Global Information & Control Committee and is the direct responsibility of the Japanese Information & Control Committee. Current edition approved by the Japanese Regional Standards Committee on August 8, 2003. Initially available at www.semi.org October 2003; to be published November 2003. Originally published in 1992; previously published July 2003.

CONTENTS

SEMI E30-1103
GENERIC MODEL FOR COMMUNICATIONS AND CONTROL OF MANUFACTURING EQUIPMENT (GEM)

This standard was technically approved by the Global Information & Control Committee and is the direct responsibility of the Japanese Information & Control Committee. Current edition approved by the Japanese Regional Standards Committee on August 8, 2003. Initially available at www.semi.org October 2003; to be published November 2003. Originally published in 1992; previously published July 2003.

1 Introduction

1.1 *Revision History* — This is the first release of the GEM standard.

1.2 *Scope* — The scope of the GEM standard is limited to defining the behavior of semiconductor equipment as viewed through a communications link. The SEMI E5 (SECS-II) standard provides the definition of messages and related data items exchanged between host and equipment. The GEM standard defines which SECS-II messages should be used, in what situations, and what the resulting activity should be. Figure 1.1 illustrates the relationship of GEM, SECS-II and other communications alternatives.

The GEM standard does NOT attempt to define the behavior of the host computer in the communications link. The host computer may initiate any GEM message scenario at any time and the equipment shall respond as described in the GEM standard. When a GEM message scenario is initiated by either the host or equipment, the equipment shall behave in the manner described in the GEM standard when the host uses the appropriate GEM messages.

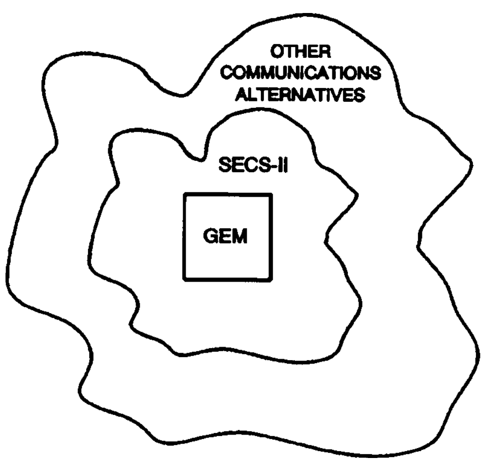

Figure 1.1
GEM Scope

The capabilities described in this standard are specifically designed to be independent of lower-level communications protocols and connection schemes (e.g., SECS-I, SMS, point-to-point, connection-oriented or connectionless). Use of those types of standards is not required or precluded by this standard.

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.

1.3 *Intent* — GEM defines a standard implementation of SECS-II for all semiconductor manufacturing equipment. The GEM standard defines a common set of equipment behavior and communications capabilities that provide the functionality and flexibility to support the manufacturing automation programs of semiconductor device manufacturers. Equipment suppliers may provide additional SECS-II functionality not included in GEM as long as the additional functionality does not conflict with any of the behavior or capabilities defined in GEM. Such additions may include SECS-II messages, collection events, alarms, remote command codes, processing states, variable data items (data values, status values or equipment constants), or other functionality that is unique to a class (etchers, steppers, etc.) or specific instance of equipment.

GEM is intended to produce economic benefits for both device manufacturers and equipment suppliers. Equipment suppliers benefit from the ability to develop and market a single SECS-II interface that satisfies most customers. Device manufacturers benefit from the increased functionality and standardization of the SECS-II interface across all manufacturing equipment. This standardization reduces the cost of software development for both equipment suppliers and device manufacturers. By reducing costs and increasing functionality, device manufacturers can automate semiconductor factories more quickly and effectively. The flexibility provided by the GEM standard also enables device manufacturers to implement unique automation solutions within a common industry framework.

 SEMI E30-1103 © SEMI 1992, 2003

The GEM standard is intended to specify the following:

— A model of the behavior to be exhibited by semiconductor manufacturing equipment in a SECS-II communication environment,

— A description of information and control functions needed in a semiconductor manufacturing environment,

— A definition of the basic SECS-II communications capabilities of semiconductor manufacturing equipment,

— A single consistent means of accomplishing an action when SECS-II provides multiple possible methods, and

— Standard message dialogues necessary to achieve useful communications capabilities.

The GEM standard contains two types of requirements:

— fundamental GEM requirements and

— requirements of additional GEM capabilities.

The fundamental GEM requirements form the foundation of the GEM standard. The additional GEM capabilities provide functionality required for some types of factory automation or functionality applicable to specific types of equipment. A detailed list of the fundamental GEM requirements and additional GEM capabilities can be found in Chapter 8, GEM Compliance. Figure 1.2 illustrates the components of the GEM standard.

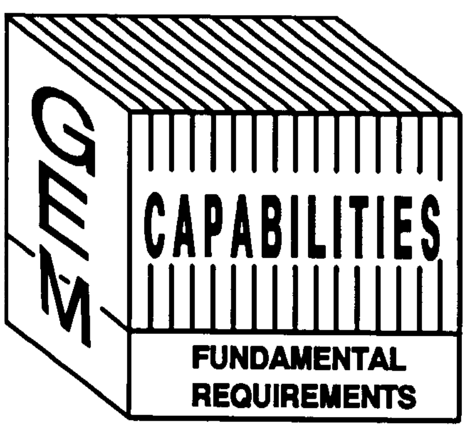

Figure 1.2
GEM Components

Equipment suppliers should work with their customers to determine which additional GEM capabilities should be implemented for a specific type of equipment. Because the capabilities defined in the GEM standard were specifically developed to meet the factory automation requirements of semiconductor manufacturers, it is anticipated that most device manufacturers will require most of the GEM capabilities that apply to a particular type of equipment. Some device manufacturers may not require all the GEM capabilities due to differences in their factory automation strategies.

1.4 *Overview* — The GEM standard is divided into sections as described below.

Section 1 — Introduction

This section provides the revision history, scope and intent of the GEM standard. It also provides an overview of the structure of the document and a list of related documents.

Section 2 — Definitions

This section provides definitions of terms used throughout the document.

Section 3 — State Models

This section describes the conventions used throughout this document to depict state models. It also describes the basic state models that apply to all semiconductor manufacturing equipment and that pertain to more than a single capability. State models describe the behavior of the equipment from a host perspective.

Section 4 — Capabilities and Scenarios

This section provides a detailed description of the communications capabilities defined for semiconductor manufacturing equipment. The description of each capability includes the purpose, definitions, requirements, and scenarios that shall be supported.

Section 5 — Data Definitions

This section provides a reference to the Data Item Dictionary and Variable Item Dictionary found in SEMI Standard E5. The first subsection shows those data items from SECS-II which have been restricted in their use (i.e., allowed formats). The second subsection lists variable data items that are available to the host for data collection and shows any restrictions on their SECS-II definitions.

Section 6 — Collection Events

This section provides a list of required collection events and their associated data.

Section 7 — SECS Message Subset

This section provides a composite list of the SECS-II messages required to implement all capabilities defined in the GEM standard.

Section 8 — GEM Compliance

This section describes the fundamental GEM requirements and additional GEM capabilities and provides references to other sections of the standard where detailed requirements are located. This section also defines standard terminology and documentation that can be used by equipment suppliers and device manufacturers to describe compliance with this standard.

Section A — Application Notes

These sections provide additional explanatory information and examples.

Section A.1 — Factory Operational Script

This section provides an overview of how the required SECS capabilities may be used in the context of a typical factory operation sequence. This section is organized according to the sequence in which actions are typically performed.

Section A.2 — Equipment Front Panel

This section provides guidance in implementing the required front panel buttons, indicators, and switches as defined in this document. A summary of the front panel requirements is provided.

Section A.3 — Examples of Equipment Alarms

This section provides examples of alarms related to various equipment configurations.

Section A.4 — Trace Data Collection Example

This section provides an example of trace initialization by the host and the periodic trace data messages that might be sent by the equipment.

Section A.5 — Harel Notation

This section explains David Harel's "Statechart" notation that is used throughout this document to depict state models.

Section A.6 — Example Control Model Application

This section provides one example of a host's interaction with an equipment's control model.

Section A.7 — Examples of Limits Monitoring

This section contains four limits monitoring examples to help clarify the use of limits and to illustrate typical applications.

1.5 *Applicable Documents*

1.5.1 *SEMI Standards* — The following SEMI standards are related to the GEM standard. The specific portions of these standards referenced by GEM constitute provisions of the GEM standard.

SEMI E4 — SEMI Equipment Communications Standard 1 — Message Transfer (SECS-I)

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

SEMI E13 — Standard for SEMI Equipment Communication Standard Message Service (SMS)

SEMI E23 — Specification for Cassette Transfer Parallel I/O Interface

1.5.2 *Other References*

Harel, D., "Statecharts: A Visual Formalism for Complex Systems," Science of Computer Programming 8 (1987) 231-274[1].

NOTICE: As listed or revised, all documents cited shall be the latest publications of adopted standards.

2 Definitions

2.1 *alarm* — An alarm is related to any abnormal situation on the equipment that may endanger people, equipment, or material being processed. Such abnormal situations are defined by the equipment manufacturer based on physical safety limitations. Equipment activities potentially impacted by the presence of an alarm shall be inhibited.

2.1.1 Note that exceeding control limits associated with process tolerance does not constitute an alarm nor do normal equipment events such as the start or completion of processing.

2.2 *capabilities* — Capabilities are operations performed by semiconductor manufacturing equipment. These operations are initiated through the communications interface using sequences of SECS-II messages (or scenarios). An example of a capability is the setting and clearing of alarms.

2.3 *collection event* — A collection event is an event (or grouping of related events) on the equipment that is considered to be significant to the host.

2.4 *communication failure* — A communication failure is said to occur when an established communications link is broken. Such failures are protocol specific. Refer to the appropriate protocol standard (e.g., SEMI E4 or

1 Elsevier Science, P.O. Box 945, New York, NY 10159-0945, http://www.elvesier.nl/homepage/browse.htt

SEMI E37) for a protocol-specific definition of communication failure.

2.5 *communication fault* — A communication fault occurs when the equipment does not receive an expected message, or when either a transaction timer or a conversation timer expires.

2.6 *control* — To control is to exercise directing influence.

2.7 *equipment model* — An equipment model is a definition based on capabilities, scenarios, and SECS-II messages that manufacturing equipment should perform to support an automated manufacturing environment. (See also Generic Equipment Model.)

2.8 *event* — An event is a detectable occurrence significant to the equipment.

2.9 *GEM compliance* — The term "GEM Compliance" is defined with respect to individual GEM capabilities to indicate adherence to the GEM standard for a specific capability. Section 8 includes more detail on GEM Compliance.

2.10 *Generic Equipment Model* — The Generic Equipment Model is used as a reference model for any type of equipment. It contains functionality that can apply to most equipment, but does not address unique requirements of specific equipment.

2.11 *host* — The SEMI E4 and E5 standards define Host as "the intelligent system that communicates with the equipment."

2.12 *message fault* — A message fault occurs when the equipment receives a message that it cannot process because of a defect in the message.

2.13 *operational script* — An operational script is a collection of scenarios arranged in a sequence typical of actual factory operations. Example sequences are system initialization powerup, machine setup, and processing.

2.14 *operator* — A human who operates the equipment to perform its intended function (e.g., processing). The operator typically interacts with the equipment via the equipment supplied operator console.

2.15 *process unit* — A process unit refers to the material that is typically processed as a unit via single run command, process program, etc. Common process units are wafers, cassettes, magazines, and boats.

2.16 *processing cycle* — A processing cycle is a sequence wherein all of the material contained in a typical process unit is processed. This is often used as a measure of action or time.

2.17 *scenario* — A scenario is a group of SECS-II messages arranged in a sequence to perform a capability. Other information may also be included in a scenario for clarity.

2.18 *SECS-I* — SEMI Equipment Communications Standard 1 (SEMI E4). This standard specifies a method for a message transfer protocol with electrical signal levels based upon EIA RS232-C.

2.19 *SECS-II* — SEMI Equipment Communications Standard 2 (SEMI E5). This standard specifies a group of messages and the respective syntax and semantics for those messages relating to semiconductor manufacturing equipment control.

2.20 *SMS* — SECS Message Service. An alternative to SECS-I to be used when sending SECS-II formatted messages over a network.

2.21 *state model* — A State Model is a collection of states and state transitions that combine to describe the behavior of a system. This model includes definition of the conditions that delineate a state, the actions/reactions possible within a state, the events that trigger transitions to other states, and the process of transitioning between states.

2.22 *system default* — Refers to state(s) in the equipment behavioral model that are expected to be active at the end of system initialization. It also refers to the value(s) that specified equipment variables are expected to contain at the end of system initialization.

2.23 *system initialization* — The process that an equipment performs at power-up, system activation, and/or system reset. This process is expected to prepare the equipment to operate properly and according to the equipment behavioral models.

2.24 *user* — A human or humans who represent the factory and enforce the factory operation model. A user is considered to be responsible for many setup and configuration activities that cause the equipment to best conform to factory operations practices.

3 State Models

The following sections contain state models for semiconductor manufacturing equipment. These state models describe the behavior of the equipment from a host perspective in a compact and easy to understand format. State models for different equipment will be identical in some areas (e.g., communications), but may vary in other areas (e.g., processing). It is desirable to divide the equipment into parallel components that can be modeled separately and then combined. An example of a component overview of an equipment is provided as Figure 3.0.

Equipment manufacturers must document the operation-al behavior of their equipment using state model meth-odology. State models are discussed in Sections 3.1 and A.5 and in a referenced article. Documentation of a state model shall include the following three elements:

— A *state diagram* showing the possible states of the system or components of a system and all of the possible transitions from one state to another. The states and transitions must each be labeled. Use of the Harel notation (see A.5) is recommended.

— A *transition table* listing each transition, the beginning and end states, what stimulus triggers the transition, and any actions taken as a result of the transition.

— A *definition of each state* specifying system behavior when that state is active.

Examples of the above elements are provided in Section A.5.

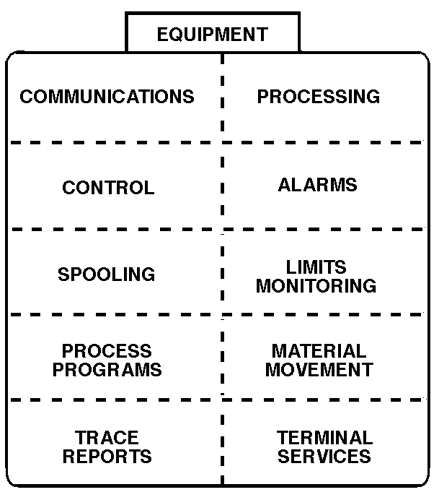

Figure 3.0
Example Equipment Component Overview

The benefits of providing state models are:

1. State machine models are a useful specification tool,

2. A host system can anticipate machine behavior based upon the state model,

3. End-users and equipment programmers have a common description of machine behavior from which to work,

4. "Legal" operations can be defined pertaining to any machine state,

5. External event notifications can be related to internal state transitions,

6. External commands can be related to state transitions,

7. State model components describing different aspects of machine control can be related to one another (example: processing state model with material transport state model; processing state model with internal machine safety systems).

3.1 *State Model Methodology* — To document the expected functionality of the various capabilities described in this document, the "Statechart" notation developed by David Harel has been adopted. An article by Harel is listed in Section 1.5 and should be considered "must" reading for a full understanding of the notation. The convention used in this and following sections is to describe the dynamic functionality of a capability with three items: a textual description of each state or substate defined, a table that describes the possible transitions from one state to another, and a graphical figure that uses the symbols defined by Harel to illustrate the relationships of the states and transitions. The combination of these items define the state model for a system or component. A summary of the Harel notation and a more detailed description of the text, table, and figure used to define behavior with this methodology is contained in the Application Note A.5.

The basic unit of a state model is the state. A state is a static set of conditions. If the conditions are met, the state is current. These conditions might involve sensor readings, switch positions, time of day, etc. Also part of a state definition is a description of reactions to specific stimuli (e.g., if message Sx,Fy is received, generate reply message Sx,Fy + 1). Stimuli may be quite varied but for semiconductor equipment would include received SECS messages, expired timers, operator input at an equipment terminal, and changes in sensor readings.

To help clarify the interpretation of this document and the state models described herein, it is useful to distin-guish between a state and an event and the relationship of one to the other. An event is dynamic rather than static. It represents a change in conditions, or more specifically, the awareness of such a change. An event might involve a sensor reading exceeding a limit, a switch changing position, or a time limit exceeded.

A change to a new active state (state transition) must always be prompted by a change in conditions, and thus an event. In addition, a state transition may itself be termed an event. In fact, there are many events that may occur on an equipment, so it is important to classify events based on whether they can be detected and whether they are of interest. In this document, the term event has been more narrowly defined as a detectable occurrence that is significant to the equipment.

A further narrowing of the definition of event is represented by the term "collection event," which is an event (or group of related events) on the equipment that is considered significant to the host. It is these events that (if enabled) are reported to the host. By this definition, the list of collection events for an equipment would typically be only a subset of total events. The state models in this document are intended to be limited to the level of detail in which the host is interested. Thus, all state transitions defined in this standard, unless otherwise specified, shall correspond to collection events.

3.2 *Communications State Model* — The Communications State Model defines the behavior of the equipment in relation to the existence or absence of a communications link with the host. Section 4.1 expands on this section by defining the Establish Communications capability. This model pertains to a logical connection between equipment and host rather than a physical connection.

3.2.1 *Terminology* — The terms communication failure, connection transaction failure, and communication link are defined for use within this document only and should not be confused with the same or similar terms used elsewhere.

- See SEMI E4 (SECS-I) or SEMI E37 (HSMS) for a protocol specific definitions of communications failure.
- A connection transaction failure occurs when attempting to establish communications and is caused by
  - — a communication failure,
  - — the failure to receive an S1,F14 reply within a reply timeout limit, or
  - — receipt of S1,F14 that has been improperly formatted or with COMMACK[2] not set to 0.
- A reply timeout period begins after the successful transmission of a complete primary message for which a reply is expected. (See SEMI E4 (SECS-I) or SEMI E37 (HSMS) for a protocol-specific definition of reply timeout.)
- A communication link is established following the first successful completion of any one S1,F13/F14 transaction with an acknowledgement of "accept". The establishment of this link is logical rather than physical.
- Implementations may have mechanisms which allow outgoing messages to be stored temporarily prior to being sent. The noun queue is used to cover such stored messages. They are queued when placed within the queue and are dequeued by removing them from this storage.
- Send includes "queue to send" or "begin the process of attempting to send" a message. It does not imply the successful completion of sending a message.
- The host may attempt to establish communications with equipment at any time due to the initialization of the host or by independent detection of a communications failure by the host. Thus, the host may initiate an S1,F13/F14 transaction at any time.

3.2.2 *CommDelay Timer* — The CommDelay timer represents an internal timer used to measure the interval between attempts to send S1,F13. The length of this interval is equal to the value in the EstablishCommunicationsTimeout. The CommDelay timer is not directly visible to the host.

EstablishCommunicationsTimeout is the user-configurable equipment constant that defines the delay, in seconds, between attempts to send S1,F13. This value is used to initialize the CommDelay timer.

The CommDelay timer is initialized to begin timing. The CommDelay timer is initialized only when the state WAIT DELAY is entered.

The CommDelay timer is expired when it "times out," and the time remaining in the interval between attempts to send is zero. When the timer expires during the state WAIT DELAY, it triggers a new attempt to send S1,F13 and the transition to the state WAIT CRA[3] .

3.2.3 *Conventions*

- The attempt to send S1,F13 is made only upon transit into the state WAIT CRA. The CommDelay Timer should be set to "expired" at this time.
- The CommDelay timer is initialized only upon transit into the state WAIT DELAY. A next

2 Establish Communications Acknowledge Code, defined in Section 4.1. See the SEMI E5 Standard for further definition of this Data Item.

3 CRA is the mnemonic defined for Establish Communications Request Acknowledge (S1,F14).

attempt to send S1,F13 shall occur only upon a transit to the state WAIT CRA.

3.2.4 *Communication States* — There are two major states of SECS communication, DISABLED and ENABLED. The system default state must be user-configurable at the equipment (e.g., via a jumper setting or non-volatile memory variable).

Once system initialization has been achieved, the opera-tor shall be able to change the communication state selection at any time via equipment terminal functions or momentary switch. A two-position type switch must not be used due to possible conflict with the system default.

The ENABLED state has two substates, NOT COMMUNICATING and COMMUNICATING, described below. The equipment must inform the operator of the current communication state via continuous display at the equipment, including the NOT COMMUNICATING and COMMUNICATING sub-states.

In the event of a connection transaction failure, a user-configurable equipment constant EstablishCommunicationsTimeout is used to establish the interval between attempts to send an S1,F13 (Establish Communications Request) while in the NOT COMMUNICATING substate.

Figure 3.2.1 shows the relationship between the superstates and substates of the Communications State Model. A description of the events triggering state transitions and the actions taken is given in Table 3.2.

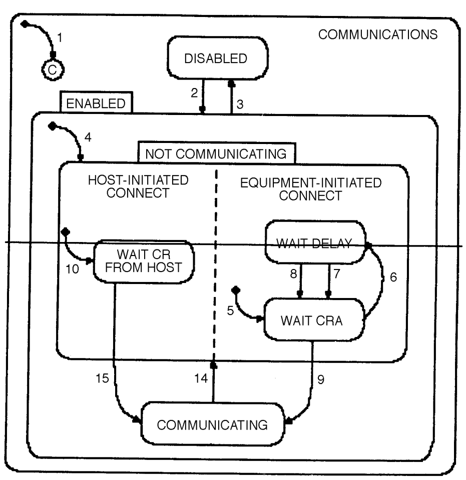

Figure 3.2.1
Communications State Diagram

The states of the Communications State Model are defined as follows:

DISABLED

In this state SECS-II communication with a host computer is non-existent. If the operator switches from ENABLED to DISABLED, all SECS-II communications must cease immediately. Any messages queued to send shall be discarded, and all further action on any open transactions and conversations shall be terminated.[4] Handling of messages currently being transmitted is an issue for lower level message transfer protocols and is not addressed in this standard.

The DISABLED state is a possible system default.

ENABLED

ENABLED has two substates, COMMUNICATING and NOT COMMUNICATING. Whenever communications are enabled, either during system initialization or through operator selection, the substate of NOT COMMUNICATING is active until communications are formally established. Lower-level protocols (such as SECS-I) are assumed to be functioning normally in that they are capable of supporting the communication of SECS-II syntax.

The ENABLED state is a possible system default.

ENABLED/NOT COMMUNICATING

No messages other than S1,F13, S1,F14, and S9,Fx shall be sent while this substate is active. The equipment shall discard any messages received from the host other than S1,F13 or S1,F14 (Establish Communications Acknowledge). It shall also periodically attempt to establish communication with a host computer by issuing an S1,F13 until communications are successfully established. However, only one equipment-initiated S1,F13 transaction may be open at any time.

The NOT COMMUNICATING state has two AND substates, HOST-INITIATED CONNECT and EQUIPMENT-INITIATED CONNECT, both of which are active whenever the equipment is NOT COMMUNICATING. These two substates clarify the behavior of the equipment in the event that both the equipment and the host attempt to establish communications during the same period of time[5] .

4 Refer to SEMI E5, Section 5, for definitions of SECS-II transaction and conversation protocols.

5 Note that in the Harel notation, an exit from any AND substate is an exit from the parent state and thus from all other AND substates of that parent substate.

NOT COMMUNICATING/EQUIPMENT-INITIATED CONNECT

This state has two substates, WAIT CRA and WAIT DELAY. Upon any entry to the NOT COMMUNICATING state, whenever EQUIPMENT-INITIATED CONNECT first becomes active, a transition to WAIT CRA occurs, the CommDelay timer is set to “expired,” and an immediate attempt to send S1,F13 is made.

NOT COMMUNICATING/EQUIPMENT-INITIATED CONNECT/WAIT CRA

An Establish Communications Request has been sent. The equipment waits for the host to acknowledge the request.

NOT COMMUNICATING/EQUIPMENT-INITIATED CONNECT/WAIT DELAY

A connection transaction failure has occurred. The CommDelay timer has been initialized. The equipment waits for the timer to expire.

NOT COMMUNICATING/HOST-INITIATED CONNECT

This state describes the behavior of the equipment in response to a host-initiated S1,F13 while NOT COMMUNICATING is active.

NOT COMMUNICATING/HOST-INITIATED CONNECT/WAIT CR FROM HOST

The equipment waits for an S1,F13 from the host. If an S1,F13 is received, the equipment attempts to send an S1,F14 with COMMACK = 0.

ENABLED/COMMUNICATING

Communications have been established. The equipment may receive any message from the host, including S1,F13. When the equipment is COMMUNICATING, SECS communications with a host computer must be maintained. This state remains active until communications are disabled or a communication failure occurs. If the equipment receives S1,F13 from the host while in the COMMUNICATING substate, it should respond with S1,F14 with COMMACK set to zero. If the equipment receives S1,F14 from a previously sent S1,F13, no action is required.

In the event of communication failure, the equipment shall return to the NOT COMMUNICATING substate and attempt to re-establish communications with the host.

It is possible that the equipment may be waiting for an S1,F14 from the host in EQUIPMENT-INITIATED CONNECT/WAIT CRA at the time an S1,F13 is received from the host in HOST-INITIATED CONNECT/WAIT CR FROM HOST. When this situation occurs, both equipment and host have an open S1,F13/S1,F14 transaction. Since communications are successfully established on the successful completion of any S1,F13/S1,F14 transaction, either of these two transactions may be the first to complete successfully and to cause the transition from NOT COMMUNICATING state to COMMUNICATING. In this event, the other transaction shall remain open regardless of the transition to COMMUNICATING until it is closed in a normal manner.

If the equipment has not yet sent[6] an S1,F14 to a previously received S1,F13 at the time when COMMUNICATING becomes active, the S1,F14 response shall be sent in a normal manner. A failure to send the S1,F14 is then treated as any other communication failure.

If the equipment-initiated S1,F13/S1,F14 is still open when the transition to COMMUNICATING occurs, subsequent failure to receive a reply from the host is considered a communication fault by equipment. An S9,F9 should be sent when a transaction timer timeout occurs[7]. (See Section 4.9 for definitions of communication faults and message faults, as well as detail on Stream 9 Error Messages.)

3.2.5 *State Transitions* — Table 3.2 contains a full description of the state transitions depicted in Figure 3.2.1.

When the operator switches from the DISABLED state to the ENABLED state, no collection event shall occur, since no messages can be sent until communications have been established. The process of establishing communications serves to notify the host that communications are ENABLED. No other collection events are defined for the Communications State Model.

6 This includes transmissions that may have started but not yet successfully completed at the time that the transition to COMMUNICATING occurs.

7 The existence of a transaction timer is not a requirement in some protocols, such as SMS (SEMI E13).

Table 3.2 Communications State Transition Table

#	*Current State*	*Trigger*	*New State*	*Action*	*Comment*
1	(Entry to COMMUNICA-TIONS)	System initialization.	System Default	None.	The system default may be set to DISABLED or ENABLED.
2	DISABLED	Operator switches from DISABLED to ENABLED.	ENABLED	None.	SECS-II communications are enabled.
3	ENABLED	Operator switches from ENABLED to DISABLED.	DISABLED	None.	SECS-II communications are prohibited.
4	(Entry to ENABLED)	Any entry to ENABLED state.	NOT COMMUNICATING	None.	May enter from system initialization to ENABLED or through operator switch to ENABLED.
5	(Entry to EQUIPMENT-INITIATED CONNECT)	(Any entry to NOT COMMUNICA-TING)	WAIT CRA	Initialize communications. Set CommDelay timer "expired." Send S1,F13.	Begin the attempt to establish communications.
6	WAIT CRA	Connection transaction failure.	WAIT DELAY	Initialize CommDelay timer. Dequeue all messages queued to send.	If appropriate, dequeued messages shall be placed in spool buffer in the order generated. Wait for timer to expire.
7	WAIT DELAY	CommDelay timer expired.	WAIT CRA	Send S1,F13	Wait for S1,F14. May receive S1,F13 from Host.
8	WAIT DELAY	Received a message other than S1,F13.	WAIT CRA	Discard message. No reply. Set CommDelay timer "expired". Send S1,F13.	Indicates opportunity to establish communications.
9	WAIT CRA	Received expected S1,F14 with COMMACK = 0.	COMMUNICATING	None.	Communications are established.
10	(Entry to HOST-INITIATED CONNECT)	(Any entry to NOT COMMUNICA-TING)	WAIT CR FROM HOST	None.	Wait for S1,F13 from Host.
14	COMMUNICATING	Communication failure. (See SEMI E4 or SEMI E37 for a protocol-specific definition of communication failure.)	NOT COMMUNICATING	Dequeue all messages queued to send.	Dequeued messages may be placed in spool buffer as appropriate.
15	WAIT CR FROM HOST	Received S1,F13.	COMMUNICATING	Send S1,F14 with COMMACK = 0.	Communications are established.

3.3 *Control State Model* — The CONTROL state model defines the level of cooperation between the host and equipment. It also specifies how the operator may interact at the different levels of host control. While the COMMUNICATIONS state model addresses the ability for the host and equipment to exchange messages, the CONTROL model addresses the equipment's responsibility to act upon messages that it receives.

The CONTROL model provides the host with three basic levels of control. In the highest level (REMOTE), the host may control the equipment to the full extent possible. The middle level (LOCAL) allows the host full access to information, but places some limits on how the host can affect equipment operation. In the lowest level (OFF-LINE), the equipment allows no host control[8] and only very limited information.[9]

The control model and communications model (when implemented) do not interact directly. That is, no action or state of one model directly causes a change in behavior of the other. It is true, however, that when the communication state is NOT COMMUNICATING then most message transaction are not functional. When messages cannot be transmitted, the control capabilities and all other GEM capabilities are affected.

Refer to Figure 3.3 as the CONTROL substates and state transitions are defined.

OFF-LINE

When the OFF-LINE state is active, operation of the equipment is performed by the operator at the operator console. While the equipment is OFF-LINE, message transfer is possible. However the use of messaging for any automation purpose is severely restricted. While the OFF-LINE state is active, the equipment will only respond to those messages used for the establishment of communications or a host request to activate the ON-LINE state.

While OFF-LINE, the equipment will respond with an Sx,F0 to any primary message from the host other than S1,F13 or S1,F17. It will process and respond to S1,F13 and S1,F17. S1,F17 is used by the host to request the equipment to transition to the ON-LINE state. The equipment will accept this request and send a positive response only when the HOST OFF-LINE state is active (see transition 11 definition below).

8 The host may establish communications. This does not affect equipment operation and for that reason is not termed a control operation.

9 The host may determine the equipment identification via the S1F13/F14 transaction.

While the OFF-LINE state is active, the equipment shall attempt to send no primary message other than S1,F13,[10] S9,Fx,[11] and S1,F1 (see ATTEMPT ON-LINE substate). If the equipment receives a reply message from the host other than S1,F14 or S1,F2, this message is discarded.

No messages enter the spool when the system is OFF-LINE. Spooling may be active when the Communications State of NOT COMMUNICATING is active. This might occur during OFF-LINE, but since the equipment will not attempt to send messages except as mentioned in the previous paragraph[12], no messages will enter the spool.

OFF-LINE has three substates: EQUIPMENT OFF-LINE, ATTEMPT ON-LINE, and HOST OFF-LINE.

OFF-LINE/EQUIPMENT OFF-LINE

While this state is active, the system maintains the OFF-LINE state. It awaits operator instructions to attempt to go ON-LINE.

OFF-LINE/ATTEMPT ON-LINE

While the ATTEMPT ON-LINE state is active, the equipment has responded to an operator instruction to attempt to go to the ON-LINE state. Upon activating this state, the equipment attempts to send an S1,F1 to the host.

Note that when this state is active, the system does not respond to operator actuation of either the ON-LINE or the OFF-LINE switch.

10 Sending of S1,F13 is based upon the COMMUNICATIONS state model.

11 S9,Fx messages may be issued only in response to the messages to which the equipment will normally respond while OFF-LINE (i.e. S1,F13 and S1,F17).

12 The equipment may send S1,F1 or S1,F13, but since Stream 1 messages are not eligible for spooling, they will not enter the spool either.

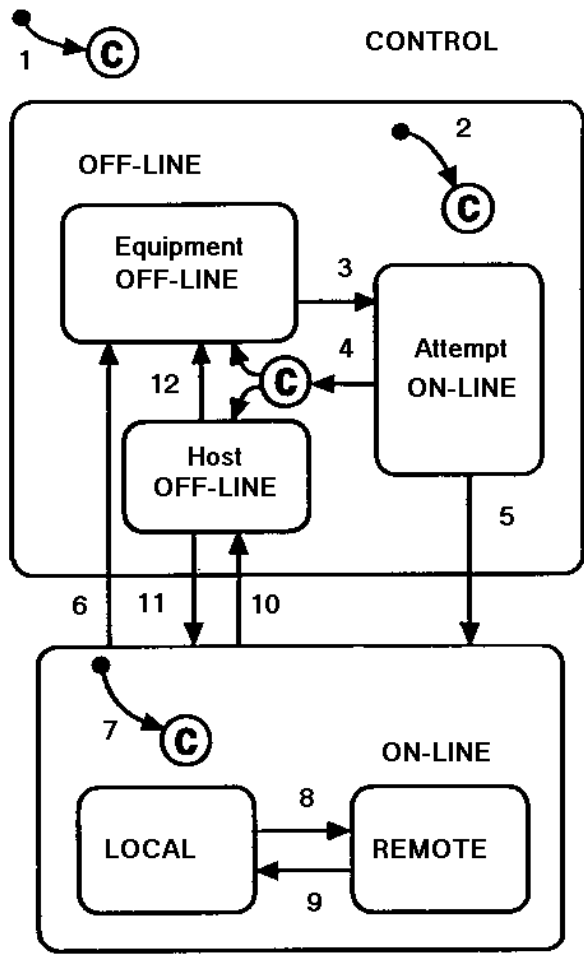

Figure 3.3
CONTROL State Model

OFF-LINE/HOST OFF-LINE

While the HOST OFF-LINE state is active, the operator's intent is that the equipment be ON-LINE. However, the host has not agreed. Entry to this state may be due to a failed attempt to go ON-LINE or to the host's request that the equipment go OFF-LINE from ON-LINE (see the transition table for more detail). While this state is active, the equipment shall positively respond to any host's request to go ON-LINE (S1,F17). Such a request shall be denied when the HOST OFF-LINE state is not active.

ON-LINE

While the ON-LINE state is active, SECS-II messages may be exchanged and acted upon. Capabilities that may be available to the host should be similar to those available from the operator console wherever practical.

The use of Sx,F0 messages is not required while the ON-LINE state is active. Their use is discouraged in this case. The only allowed use is to close open transactions in conjunction with message faults.

ON-LINE/LOCAL

Operation of the equipment is implemented by direct action of an operator. All operation commands shall be available for input at the local operator console of the equipment.

The host shall have the following capabilities and restrictions when the LOCAL state is active:

— The host shall be prohibited from the use of remote commands that cause physical movement or which initiate processing. During processing, the host shall be prohibited from the use of any remote command that affects that process.

— During processing, the host shall be prohibited from modifying any equipment constants that affect that process. Other equipment constants shall be changeable during processing. The host shall be able to modify all available equipment constants when no processing is in progress.

— The host shall be capable of initiating the upload and download of recipes to/from the recipe storage area on the equipment. The host shall be capable of selecting recipes for execution so long as this action does not affect any currently executing recipe.

— The host shall be able to configure automatic data reporting capabilities including alarms, event reporting, and trace data reporting. The host shall receive all such reports at the appropriate times.

— The host shall be able to inquire for data from the equipment, including status data, equipment constants, event reports, process program directories, and alarms.

— The equipment shall be able to perform Terminal Services as defined in GEM.

The host shall be allowed any other capabilities that were not specifically restricted in the above items as long as the LOCAL state is active.

NOTE 2: Capabilities mentioned above which are not implemented on a specific equipment may be ignored in this context.

ON-LINE/REMOTE

For equipment which supports the GEM capability of remote control (see Section 4.4), while the REMOTE state is active, the host shall have access, through the communications interface, to the necessary commands to operate the equipment through the full process cycle in an automated manner. The equipment does not restrict any host capabilities when REMOTE is active. The degree of control executed by the host may vary from factory to factory. In some cases, the operator maybe required to interact during remotely controlled processes. This interaction may involve set-up operations, operator assist situations, and others. This state is intended to be flexible enough to accommodate these different situations.

To support the different factory automation policies and procedures, it shall be possible to configure the equipment to restrict the operator in specific non-emergency procedures. These restrictions shall be configurable so that the equipment may be set up to allow the operator to perform necessary functions without contention with the host. The categories for configuration shall include (but are not limited to):

— change equipment constants (process-related),

— change equipment constants (non-process-related),

— initiate process program download,

— select process program,

— start process program,

— pause/resume process program,

— operator assist,

— material movement to/from equipment,

— equipment-specific commands (on a command-by-command basis if needed).

NOTE 3: Capabilities mentioned above which are not implemented on a specific equipment may be ignored in this context.

No capabilities that are available to the operator when the LOCAL state is active should be unconditionally restricted when the REMOTE state is active. The supplier may provide for configurable restriction of operator capabilities not included in the list above if desired. No configurability is necessary for any operator functions not available to the host.

The control functions must be shared to some degree between the host and the local operator. At the very least, the operator must have the capability to change the CONTROL state, actuate an Emergency Stop, and interrupt processing (e.g., STOP, ABORT, or PAUSE). All of these capabilities except Emergency Stop may be access-limited.[13]

The host software should be designed to be compatible with the capabilities allotted to the operator.

13 Definition of the method of limiting operator access (password, key, etc.) to a capability is not within the scope of this document.

Table 3.3 CONTROL State Transition Table

#	*Current State*	*Trigger*	*New State*	*Action*	*Comments*
1	(Undefined)	Entry into CONTROL state (system initialization).	CONTROL (Substate conditional on configuration).	None	Equipment may be configured to default to ON-LINE or OFF-LINE . (See NOTE 1.)
2	(Undefined)	Entry into OFF-LINE state.	OFF-LINE (Substate conditional on configuration.)	None	Equipment may be configured to default to any substate of OFF-LINE.
3	EQUIPMENT OFF-LINE	Operator actuates ON-LINE switch.	ATTEMPT ON-LINE	None	Note that an S1,F1 is sent whenever ATTEMPT ON-LINE is activated.
4	ATTEMPT ON-LINE	S1,F0.	New state conditional on configuration.	None	This may be due to a communication failure (See NOTE 2), reply timeout, or receipt of S1,F0. Configuration may be set to EQUIPMENT OFF-LINE or HOST OFF-LINE.
5	ATTEMPT ON-LINE	Equipment receives expected S1,F2 message from the host.	ON-LINE	None	Host is notified of transition to ON-LINE at transition 7.
6	ON-LINE	Operator actuates OFF-LINE switch.	EQUIPMENT OFF-LINE	None	"Equipment OFF-LINE" event occurs. (See NOTE 3.) Event reply will be discarded while OFF-LINE is active.
7	(Undefined)	Entry to ON-LINE state.	ON-LINE (Substate conditional on REMOTE/LOCAL switch setting.)	None	"Control State LOCAL" or "Control State REMOTE" event occurs. Event reported based on actual ON-LINE substate activated.
8	LOCAL	Operator sets front panel switch to REMOTE.	REMOTE	None	"Control State REMOTE" event occurs.
9	REMOTE	Operator sets front panel switch to LOCAL.	LOCAL	None	"Control State LOCAL" event occurs.
10	ON-LINE	Equipment accepts "Set OFF-LINE" message from host (S1,F15).	HOST OFF-LINE	None	"Equipment OFF-LINE" event occurs.
11	HOST OFF-LINE	Equipment accepts host request to go ON-LINE (S1,F17).	ON-LINE	None	Host is notified to transition to ON-LINE at transition 7.
12	HOST OFF-LINE	Operator actuates OFF-LINE switch.	EQUIPMENT OFF-LINE	None	"Equipment OFF-LINE" event occurs.

NOTE 1: The configuration mentioned for transitions 1 and 2 should be a single setting. This would provide the user with a choice of entering the EQUIPMENT OFF-LINE, ATTEMPT ON-LINE, HOST OFF-LINE, or ON-LINE states.

NOTE 2: Communication failures are protocol specific. Refer to the appropriate protocol standard (e.g., SEMI E4 or SEMI E37) for a protocol-specific definition of communication failure.

NOTE 3: Any host initiated transaction open at the equipment must be completed. This may happen either by sending the appropriate reply to the host prior to sending the event message or by sending an Sx,F0 message following the event message (i.e., after the transaction).

3.4 *Equipment Processing States* — The behavior of the equipment in the performance of its intended function must be documented. This processing state model is highly dependent on the equipment process, technology, and style. However, there are expected to be common aspects to these models.

The Processing State Diagram, Figure 3.4, is provided as an example of an implementation model. This model demonstrates the expected nature of the processing state model documentation. There is no requirement that these specific states be implemented.

The equipment must generate collection events for each processing state transition, as well as provide status variables (ProcessState, PreviousProcessState) whose values are the current processing state and the previous processing state.

In referring to the Processing State Diagram, note that the initialization state INIT is not an actual processing state. It is shown here simply to indicate that the IDLE processing state is entered upon completion of equipment system initialization. On the following pages detailed descriptions are provided for the equipment processing states and state transitions (numbered) as shown in the diagram.

3.4.1 *Description of Equipment Processing States*

IDLE

In this state the equipment is awaiting instructions.

PROCESSING ACTIVE

This state is the parent of all substates where the context of process program execution exists.

PROCESS

This state is the parent of those substates that refer to the active preparation and execution of a process program.

SETUP

In this state all external conditions necessary for process execution are satisfied, such as ensuring material is present at the equipment, input/output ports are in the proper state, parameters such as temperature and pressure values are within limits, etc. If all setup operations are already complete, then this becomes a fall through state and a transition to the next state takes place.

READY

In this state the equipment is ready for process execution and is awaiting a START command from the operator or the host.

EXECUTING

Executing is the state in which the equipment is executing a process program automatically and can continue to do so without external intervention.

PAUSE

In this state processing is suspended and the equipment is awaiting a command.

Each state transition is defined in the following table. Note that all transitions in this table should be considered collection events.

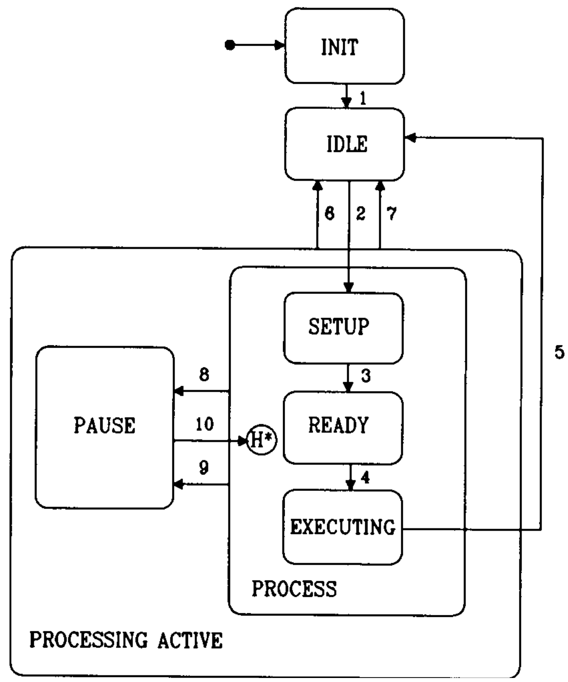

Figure 3.4
Processing State Diagram

Table 3.4 Processing State Transition Table

#	*Current State*	*Trigger*	*New State*	*Action*	*Comments*
1	INIT	Equipment initialization complete.	IDLE	None	None
2	IDLE	Commit has been made to set up.	SETUP	None	None
3	SETUP	All setup activity has completed and the equipment is ready to receive a START command.	READY	This activity is equipment-specific.	None
4	READY	Equipment has received a START command from the host or operator console.	EXECUTING	This activity is equipment-specific.	None
5	EXECUTING	The processing task has been completed.	IDLE	None	None
6	PROCESSING ACTIVE	Equipment has received a STOP command from host or operator console.	IDLE	None	None
7	PROCESSING ACTIVE	Equipment has received an ABORT command from host or operator console.	IDLE	This activity is equipment-specific.	None
8	PROCESS	The equipment decides to PAUSE due to a condition such as alarm.	PAUSE	This activity is equipment-specific.	For this type of problem, an operator assist is usually required.
9	PROCESS	Equipment has received a PAUSE command from host or operator console.	PAUSE	This activity is equipment-specific.	None
10	PAUSE	Equipment has received a RESUME command from host or operator console.	Previous PROCESS substate	This activity is equipment-specific.	None

4 Equipment Capabilities and Scenarios

This section describes the details of the capabilities required by GEM and provides scenarios for their use. Capabilities are operations performed by semiconductor manufacturing equipment. These operations are initiated through the communications interface using SECS-II messages. A scenario is a group of SECS-II messages arranged in a sequence to perform a capability. Other information may be included with the scenario for clarity. For each capability, the reader is provided with a statement of purpose, pertinent definitions, a detailed description, requirements, and scenarios.

The following capabilities are discussed:

- Establish Communications
- Event Notification
- Dynamic Event Report Configuration
- Variable Data Collection
- Trace Data Collection
- Limits Monitoring
- Status Data Collection
- On-line Identification
- Alarm Management
- Remote Control
- Equipment Constants
- Process Program Management
- Material Movement
- Equipment Terminal Services
- Error Messages

Clock

Spooling

Control

4.1 *Establish Communications* — The Establish Communications capability provides a means of formally establishing communications following system initialization or any loss of communications between communicating partners, and thus of notifying the communication partner that a period of non-communication has occurred.

4.1.1 *Purpose* — Communications between host and equipment are formally established through use of the Establish Communications Request/Establish Communications Acknowledge transaction.

The use of S1,F1/F2 for this purpose is ambiguous since the transaction can be used for other purposes and may occur at any time.

The S1,F13/F14 transaction, used in conjunction with the Communications State Model, provides a means for equipment to notify the host, or the host to notify the equipment, that there has been a period of inability to communicate. The successful completion of this transaction also signals a possible need for synchronization activities between host and equipment.

4.1.2 *Definitions*

COMMACK — Acknowledge code returned in the Establish Communications Acknowledge message. See the SEMI E5 Standard for a full definition of this data item.

EstablishCommunicationsTimeout — An equipment constant used to initialize the interval between attempts to re-send an Establish Communications Request. This value specifies the number of seconds for the interval. See the SEMI E5 Standard for a full definition of this variable data item.

4.1.3 *Description* — There are potential problems when one side of the communications link fails and the other side does not detect it. From the point of view of the host, a loss of communications has many possible causes. In some cases, host-controlled settings on the equipment may need to be reset. In other cases, the equipment may have continued an automatic processing sequence during the period of no communication and may have changed states. The definition of a formal protocol for establishing communications alerts the host to the need to synchronize itself with the equipment's current status.

Equipment shall consider communications as formally established whenever either of the following conditions have been satisfied:[14]

— Communications Request has been sent to the host and an Establish Communications Acknowledge has been received within the transaction timeout period and with an acknowledge code of Accept, or

— Communications Request has been received from the host, and an Establish Communications Acknowledge response has been successfully sent with an acknowledge code of Accept.

When the equipment sends an Establish Communications Request to the host, this notifies the host of the possible need to synchronize itself with the equipment.

When the equipment is attempting to establish communications, an Establish Communications Request shall be sent periodically until communications have been formally established as described above. The interval between attempts must be user-configurable and begins as soon as a connection transaction failure is detected (see Section 3.2).

Attempting to establish communications is not a low-level connectivity issue, but rather a logical application issue used by either party to notify its partner that the host may need to perform synchronization activities with the equipment.

4.1.4 *Requirements*

— Equipment must support the Communication State Model (see Section 3.2).

— Equipment must provide the EstablishCommunicationsTimeout equipment constant described above.

14 Satisfaction of either of these conditions will result in a transition to the COMMUNICATING substrate. See Section 3.2 for further detail.

4.1.5 *Scenarios*

4.1.5.1 *Host Attempts to Establish Communications*

```
COMMENT                        HOST    EQUIPMENT                          COMMENT

                                                 Communications state is Enabled
                                                 (any substate)
Establish Communications  S1,F13-->
Request
                                 <--S1,F14       Reply COMMACK = Accept and
                                                 Communications state =
                                                 COMMUNICATING
```

4.1.5.2 *Equipment Attempts to Establish Communications and Host Acknowledges*

```
COMMENT                        HOST    EQUIPMENT                          COMMENT

                    Communications State = NOT COMMUNICATING
                                                 [LOOP]
                                                 [LOOP]--SEND
                                 <--S1,F13       Establish Communications Request
Establish Communications  S1,F14-->
Acknowledge                                      [IF] S1,F14 rcvd w/o timeouts
                                                 [THEN] exit_loop--SEND
                                                 [ELSE] Delay for interval in
                                                        Establish Commuications-
                                                        Timeout
                                                 [ENDIF]
                                                 [END_LOOP]--SEND
                                                 [IF] COMMACK = Accept
                                                 [THEN] Communications state=
                                                        Communicating
                                                        exit_loop--
                                                 [ELSE] Reset timer for delay,
                                                        and delay for interval
                                                        specified in
                                                        EstablishCommunications—
                                                        Timeout
                                                 [ENDIF]
                                                 [END_LOOP]
```

4.1.5.3 *Simultaneous Attempts to Establish Communications* — For equipment that supports interleaving, it is possible that either the host or equipment could send an Establish Communications Request before receiving the request from its partner. As communications are established by the successful acceptance of any one Establish Communications Request, it is immaterial who sends the request first. The roles of host and equipment may be reversed.

Equipment Receives S1,F14 From Host Before Sending S1,F14:

COMMENT	**HOST**	**EQUIPMENT**	**COMMENT**
	Communications State = NOT COMMUNICATING		
		<--S1,F13	Establish Communications Request
Establish Communications Request	S1,F13-->		
Reply COMMACK = Accept	S1,F14-->		S1,F14 received from Host and Communications established[15] and Communications state = COMMUNICATING
		<--S1,F14	Reply COMMACK = Accept[16]

Equipment Sends S1,F14 To Host Before Receiving S1,F14:

COMMENT	**HOST**	**EQUIPMENT**	**COMMENT**
	Communications State = NOT COMMUNICATING		
		<--S1,F13	Establish Communications Request
Establish Communications Request	S1,F13-->		
		<--S1,F14	Reply COMMACK = Accept[15] Communications established[15] and Communications state = COMMUNICATING
Reply COMMACK = Accept	S1,F14-->		S1,F14 received from Host

4.2 *Data Collection* — Data collection allows the host to monitor equipment activity via event reporting, trace data reporting, limits monitoring, and query of selected status or other variable data.

4.2.1 *Event Data Collection* — Event data collection provides a dynamic and flexible method for the user to tailor the equipment to meet individual needs with respect to data representation and presentation to the host. The event-based approach to data collection provides automatic notification to the host of equipment activities and is useful in monitoring the equipment and in maintaining synchronization with the equipment.

Event data collection may be broken into two logical parts: host notification when an event occurs and dynamic configuration of the data attached to the event notification.

4.2.1.1 *Event Notification* — This section describes the method of notifying the host when equipment collection events occur.

4.2.1.1.1 *Purpose* — This capability provides data to the host at specified points in equipment operation. This asynchronous reporting eliminates the need for the host to poll the equipment for this information. Events on the equipment may trigger activity on the part of the host. Also, knowledge of the occurrence of events

15 Communications are establisted at the successful completion of the S1,F13/14 transaction where COMMACK is set to zero.

16 Communications are established on the successful transmission of S1,F14, even if there is an open S1,F13.

related to the equipment state models allows the host to track the equipment state. An equipment's behavior is related to its current state. Thus, this capability helps the host understand how an equipment will behave and how it will react to host behavior.

4.2.1.1.2 *Definitions*

Collection Event — An event (or grouping of related events) on the equipment that is considered significant to the host.

Collection Event ID (CEID) — A unique identifier of a collection event. See the SEMI E5 Standard for a full definition of this data item.

Event — A detectable occurrence significant to the equipment.

Report — A set of variables predefined by the equipment or defined by the host via S2,F33/F34.

4.2.1.1.3 *Detailed Description* — The equipment supplier must provide a set of predefined collection events. Specific collection events are required by individual capabilities and state models. Examples of collection events include:

— The completion of each action initiated by a host requested command,

— Selected processing and material handling activities,

— Operator action detected by the equipment,

— A state transition,

— The setting or clearing of an alarm condition on the equipment, and

— Exception conditions not considered alarms.

See Section 6 for a list of required collection events.

The reporting of a collection event may be disabled per event by the user to eliminate unwanted messages. An event report message shall be sent to the host upon the occurrence of a particular collection event if the collection event (CEID) has been enabled. Attached to each event message is one or more event reports which contain variable data. Section 4.2.1.2 describes the capability which allows for the dynamic customization of event reports. The values of any data contained in an event report message must be current upon the occurrence of the event. This implies that event reports be built at the time of the event occurrence.

The equipment shall also provide the S6,F15/F16 transaction to allow the host to request the data from a specific event report.

4.2.1.1.4 *Requirements*

— The equipment supplier shall provide documentation of all collection events defined on the equipment and the conditions for each event to occur.

— The equipment supplier shall provide unique CEIDs for each of the various collection events that are available for reporting.

— The equipment supplier shall provide a method for enabling and disabling the reporting of each event. This method shall either be available via the host interface (see Section 4.2.1.2) or the equipment's operator console.

— For each event, the equipment supplier shall provide either

- a default set of report(s) linked to the event which contain data pertinent to that event, or
- the ability for the user to configure the data linked to that event via the equipment's operator console or host interface (see Section 4.2.1.2).

4.2.1.1.5 *Scenarios*

Collection Event Occurs on the Equipment:

```
COMMENT                          HOST   EQUIPMENT                          COMMENT

                                                  [IF] Event Report is Multi-block
                                         <--S6,F5    [THEN] send Multi-block inquire
Multi-block grant               S6,F6-->
                                                  [ENDIF]
                                         <--S6,F11   Equipment sends Event Report
Host acknowledges Event Report S6,F12-->
```

Host Requests Event Report:

```
COMMENT                           HOST   EQUIPMENT                        COMMENTS

Host requests an event report S6,F15-->
                                      <--S6,F16    Equipment sends event report.
```

4.2.1.2 *Dynamic Event Report Configuration* — This section describes a capability which allows the host to dynamically modify the equipment event reporting setup.

4.2.1.2.1 *Purpose* — This capability is defined to provide the data reporting flexibility required in some manufacturing environments. It allows the host to increase or decrease the data flow according to need. For example, if the performance of an equipment degrades, the data flow from that equipment may be increased to help diagnose the problem.

4.2.1.2.2 *Definitions*

EventsEnabled — A variable data item that consists of a list of currently enabled collection events (CEIDs). See SEMI E5 for a full definition of this variable data item.

Report ID (RPTID) — A unique identifier of a specific report. See SEMI E5 for a full definition of this data item.

Variable Data (V) — A data item containing status (SV), data (DVVAL), or constant (ECV) values. See SEMI E5 for a full definition of this data item.

Variable Data ID (VID) — A unique identifier of a variable data item. The set of VID's is the union of all SVID's, ECID's, and ID's for DVVAL's (DVNAME's). See SEMI E5 for a full definition of this data item.

4.2.1.2.3 *Detailed Description* — The equipment shall support the following event report configuration functionality through the SECS-II interface:

— Host definition/deletion of custom reports,

— Host linking/unlinking of defined reports to specified collection events, and

— Host enabling/disabling the reporting of specified collection events.

NOTE 4: The equipment may also supply alternative means for defining reports and linking reports to events (e.g., via the operator console). Implementation of alternate means is not required.

The equipment can be instructed by the host to enable or disable reporting of collection events on an individual or collective basis. A status value (SV) shall be available that consists of a list of enabled collection events. (See Section 5.2, Variable Item List, EventsEnabled variable.)

Reports may be attached to an event report message (S6,F11). These reports are specifically linked to the desired event and typically contain variable data relating to that event. The reports may be provided by the equipment supplier or created by the user. The user must be able to create reports and link them to events via the SECS-II interface.

The data reported in the event report messages may consist of Status Values (SV's), Equipment Constant Values (ECV's), or Data Values (DVVAL's). Note that data values shall be valid and current on certain events and certain states and might not be current at other times. The implementor shall document when a data value will be current and available for reporting.

4.2.1.2.4 *Requirements*

— The equipment manufacturer must provide documentation of all variable data available from the equipment. This is to include variable name, variable type or class (SV, ECV, DVVAL), units, format codes, possible range of values, and a description of the meaning and use of this variable.

— The equipment manufacturer must provide unique VIDs for the various variable data (V) available for data collection in the equipment. For example, this means that no SV shall have a VID which is the same as the VID of any ECV or DVVAL.

— All variable data must be available for report definition and event data collection. See Section 5.2, Variable Item List, for a list of required variable data.

— All report definitions, report-to-event links, and enable/disable status of event reports must be retained in non-volatile storage.

4.2.1.2.5 *Scenario*

Collection Event Reporting Set-up:

```
COMMENT                                 HOST   EQUIPMENT                         COMMENT

[IF] Define Report is Multi-block
[THEN] send Multi-block inquire S2,F39-->
                                               <-- S2,F40   Multi-block grant.
[ENDIF]
Send report definitions S2,F33-->                           DATAIDs, RPTIDs, and VIDs
                                                            received.
                                               <-- S2,F34   DRACK[17]  = 0 the reports are OK
[IF] Link Event/Report is Multi-block
[THEN] send Multi-block inquire S2,F39-->
                                               <-- S2,F40   Multi-block grant.
[ENDIF]
Link reports to events S2,F35-->                            CEIDs and the corresponding
                                                            RPTIDs are received.
                                               <-- S2,F36   LRACK = 0 the event linkages
                                                            are acceptable.
Enable specific collection S2,F37-->                        Enable/ disable codes (CEEDs)
events                                                      and the respective event
                                                            reporting CEIDs received.
                                               <-- S2,F38   ERACK = 0 OK, will generate the
                                                            specified reports when the
                                                            appropriate collection events
                                                            happen
```

4.2.2 *Variable Data Collection*

4.2.2.1 *Purpose* — This capability allows the host to query for the equipment data variables and is useful during initialization and synchronization.

4.2.2.2 *Definitions*

Report ID (RPTID) — A unique identifier of a specific report. See SEMI E5 for a full definition of this data item.

Variable Data (V) — A variable data item containing status, discrete, or constant data. See SEMI E5 for a full definition of this data item.

4.2.2.3 *Detailed Description* — The host may request a report containing data variables from the equipment by specifying the RPTID. It is assumed that the report has been previously defined (e.g., using the Define Report S2,F33 transaction (see Section 4.2.1)). The values of any status variables (SV's) and equipment constants (ECV's) contained within the report must be current. Discrete data values (DVVAL's) are only guaranteed to be valid upon the occurrence of a specific collection event. If DVVAL cannot be specified in equipment due to some restrictions depend on hardware and/or software conditions, the zero length item is reported.

4.2.2.4 *Requirements*

— Variable data items (V's) and associated units of measure must be provided by the equipment manufacturer.

17 Define Report Acknowledge Code, see SEMI E5 for full definition of this Data Item.

4.2.2.5 *Scenario*

Host Requests Report:

```
COMMENT                           HOST   EQUIPMENT                            COMMENT

Host requests data variables
contained in report RPTID    S6,F19-->
                                        <--S6,F20     Equipment responds with a list of
                                                      variable data for the given
                                                      RPTID.
```

4.2.3 *Trace Data Collection*

4.2.3.1 *Purpose* — Trace data collection provides a method of sampling data on a periodic basis. The time-based approach to data collection is useful in tracking trends or repeated applications within a time window, or monitoring of continuous data.

4.2.3.2 *Definitions*

Data Sample Period (DSPER) — The time delay between samples. See SEMI E5 for a full definition of this data item.

Reporting Group Size (REPGSZ) — The number of samples included per trace report transmitted to the host. See SEMI E5 for a full definition of this data item.

Status Variable (SV) — Status data item (included in trace report). See SEMI E5 for a full definition of this data item.

Status Variable ID (SVID) — A unique identifier of a status variable. See SEMI E5 for a full definition of this data item.

Total Samples (TOTSMP) — Number of samples to be taken during a complete trace period. See SEMI E5 for a full definition of this data item.

Trace Request ID (TRID) — An identifier associated with a trace request definition. See SEMI E5 for a full definition of this data item.

4.2.3.3 *Detailed Description* — The equipment shall establish a trace report as instructed by the host (S2,F23). For a trace report (S6,F1), the host shall designate a name for the trace report (TRID), a time interval for data sampling (DSPER), the total number of samples to be taken (TOTSMP), the number of samples per trace report (REPGSZ), and a listing of which data will be sent with the report (SVID's). The number of trace reports sent to the host is determined by total samples divided by reporting group size (TOTSMP/REPGSZ).

The equipment shall sample the specified data (SV's) at the interval designated by the host (DSPER) and shall send a predefined trace report to the host for the specified reporting group size (REPGSZ). The trace report definition shall be automatically deleted from the equipment after the last trace report has been sent.

The host may modify or re-initiate a trace function currently in progress by specifying the same TRID in a trace request definition, at which point the old trace shall be terminated and the new trace shall be initiated, or the host can instruct the equipment to terminate a trace report prior to its completion by specifying TOTSMP = 0 for that TRID, at which point the trace report definition shall be deleted.

A detailed example is included as Application Note A.4.

4.2.3.4 *Requirements*

— The equipment must have a local mechanism (e.g., internal clock) for triggering the periodic sampling and transmission of trace reports to the host.

— A minimum of four (4) concurrent traces shall be supported by the equipment. The same SVID may be collected in multiple traces simultaneously.

— All SVID's available at the equipment shall be supported for trace data collection. The exception to this is any SV that will not fit into a single block.

NOTE 5: SEMI E5 provides for SV's to be of a list format. Since this may in practice be a variable list, there is a potential problem with such an SV supported by the Trace Data Collection capability. This is a problem with the SEMI E5 standard. Care should be exercised in the use of SV's using the list format.