

SEMI E30-0307^E

製造装置の通信およびコントロールのための包括的モデル(GEM)

免責事項:この SEMI スタンダードは、投票により作成された英語版が正式なものであり、日本語版は日本の利用者各位の便宜のために作成したものです。万が一英語と日本語とに差異がある場合には英語版記載内容が優先されます。

本スタンダードは、global Information & Control Committee で技術的に承認されている。現版は 2006 年 11 月 21 日、global Audits and Reviews Subcommittee にて発行が承認された。2007 年 2 月に www.semi.org で、そして 2007 年 3 月に CD-ROM で入手可能となる。初版は 1992 年発行、前版は 2003 年 7 月に発行された。

^E 本文書は 2007 年 9 月に、誤字を訂正するため、編集上の修正がなされた。変更箇所は、表 11 である。

目次

1. 序文
 - 1.1 改訂の記録
 - 1.2 範囲
 - 1.3 目的
図1 GEM の範囲
 - 1.4 概要
図2 GEM の構成要素
 - 1.5 参照スタンダードおよび文書
2. 用語
3. 状態モデル
 - 3.1 状態モデル方法論
 - 3.2 通信状態モデル
図3 装置構成要素の概観の例
図4 通信状態図
表1 通信状態遷移
 - 3.3 コントロール状態モデル
図5 コントロール状態モデル
表2 コントロール状態遷移
 - 3.4 装置プロセッシング状態
図6 プロセッシング状態図
表3 プロセッシング状態遷移
4. 装置性能とシナリオ
 - 4.1 通信確立
 - 4.2 データ収集
図7 リミットの組合せ: コントロールの事例
図8 リミットの構成要素
図9 リミット状態モデル
表4 リミット状態遷移
 - 4.3 アラーム管理
図10 アラーム ALIDn についての状態図
表5 アラーム状態遷移
表6 イベントとアラームの違い
 - 4.4 リモートコントロール
 - 4.5 装置定数
 - 4.6 プロセスレシピ管理

- 4.7 材料搬送
- 4.8 装置端末サービス
- 4.9 エラーメッセージ
- 4.10 クロック
- 4.11 スプーリング
 - 図 11 スプーリング状態図
 - 表 7 スプーリング状態遷移
- 4.12 コントロール
- 5. データ項目
 - 5.1 データ項目の制約
 - 5.2 変数項目リスト
- 6. 収集イベント
 - 表 8 GEM で定義された収集イベント
- 7. SECS-II メッセージサブセット
 - ストリーム 1: 装置状態
 - ストリーム 2: 装置コントロールおよび診断
 - ストリーム 5: 例外 (アラーム) 報告
 - ストリーム 6: データ収集
 - ストリーム 7: プロセスプログラムロード
 - ストリーム 9: システムエラー
 - ストリーム 10: 端末サービス
 - ストリーム 13:
 - ストリーム 14: オブジェクトサービス
 - ストリーム 15: レシピ管理
 - ストリーム 19: レシピとパラメータの管理
- 8. GEM 準拠
 - 8.1 GEM の基本条件
 - 表 9 GEM の基本条件
 - 図 12 GEM の要求条件と性能
 - 8.2 GEM の性能
 - 表 10 GEM 性能に関連する項
 - 8.3 GEM 準拠の定義
 - 8.4 文書
 - 図 13 ホストから見た GEM
 - 表 11 GEM 準拠
 - 表 12 SML 表記法
- A. アプリケーションノート
 - A.1 工場作業要綱
 - A1.1 常時性能
 - A1.2 システムの立ち上げおよび同期
 - A1.3 生産設定
 - A1.4 処理
 - A1.5 後処理
 - A.2 装置フロントパネル
 - A2.1 ディスプレイおよび表示器
 - A2.2 スイッチ/ボタン
 - A.3 装置アラームの例
 - 表 A.3 装置の構成別アラームの例
 - A.4 トレースデータ収集例
 - A.5 ハレル表記法

- 図A5.1 ハレルステートチャート記号
- 図A5.2 OR 下位状態の例
- 図A5.3 AND 下位状態の例
- A5.1 状態の定義
- A5.2 遷移表
- 表A.5 モーター例に対する遷移
- A.6 コントロールモデルの適用例
 - 図A.6.1 単純化した“効果的”コントロールモデルの例
- A.7 リミット監視の例
 - A7.1 序文
 - A7.2 例
 - 図A7.1 バルブ監視の例
 - 図A7.2 環境監視の例
 - 図A7.3 キャリブレーションカウンタの例
- A.8 プロセスコントロールおよび装置コントロールのためのプロセスパラメータの修正
 - A8.1 序文
 - A8.2 装置定数
 - A8.3 例
 - 図A8.1 ホストレシピパラメータ修正性能がある CMP 枚葉「研磨」システム

1 序文

1.1 改訂の記録 — 本書が GEM スタンダードの最初のリリースである。

1.2 範囲 — GEM スタンダードの範囲は通信リンクを通してみた半導体製造装置の動作を定義することに限られる。ホストと装置の間で交わされるメッセージおよび関連データ項目については SEMI E5 (SECS-II) スタンダードが定義している。GEM スタンダードは、どのような状況で、どの SECS-II メッセージを使い、その結果どのような動作が起こるかということを定義する。GEM, SECS-II, および通信のためのその他の選択肢の関係を図 1 に示す。

GEM スタンダードには通信リンク内でのホストコンピュータの動作を定義しようとする考えはない。ホストコンピュータはいつでも任意の GEM メッセージシナリオを開始することができ、装置は GEM スタンダードに書かれているように応答する。ホストか装置のどちらかがある GEM メッセージシナリオを開始した場合、ホストが適切な GEM メッセージを使っていれば装置は GEM スタンダードに書かれているとおりに動作する。

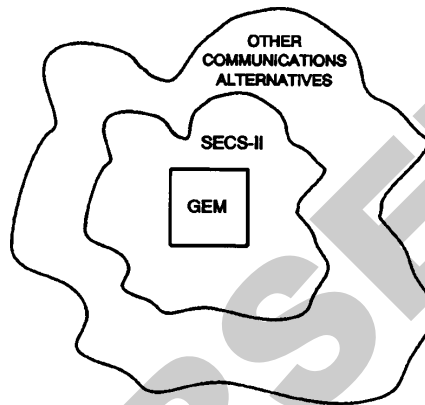


図 1
GEM の範囲

本スタンダードで記述した性能は、低レベル通信プロトコルや接続方式（例えば SECS-I, SMS, ポイントツーポイント, コネクション型, あるいはコネクションレス型）から独立するように特別にデザインされている。これらのスタンダードを GEM に適用することは要求条件ではない。

注意: 本スタンダードは、その使用に関連したすべての安全問題を取り扱うことを意図していない。このスタンダードの使用者は、その責任において、適切な安全および健康上の作業方法を確立し、また使用前に規制上の制限への適用性を判断するものである。

1.3 目的 — GEM はすべての半導体製造装置を対象に SECS-II の標準的な適用方法を定義している。GEM スタンダードは装置の動作と通信性能の一般的集合を定義しており、それによって半導体デバイス製造業者の行う製造自動化プログラムをサポートするための機能性と柔軟性を得ている。装置サプライヤは、GEM で定義した動作や性能のどれとも矛盾しない限り、GEM に含まれていない SECS-II の機能を追加することができる。追加されるものとしては、SECS-II メッセージ、収集イベント、アラーム、リモートコマンドコード、プロセッシング状態、変数データ項目（データ値、状態値もしくは装置定数）、あるいは、クラス（エッチャー、ステッパーなど）や個別装置の事例ごとに固有の機能性などがある。

GEM はデバイス製造業者と装置サプライヤの双方に経済的な利益をもたらすことを目的としている。装置サプライヤは、大部分の顧客を満足させるようなただ一つの SECS-II インタフェースを開発して市場に出せることにより利益を受け、デバイス製造業者は、あらゆる製造装置で機能が向上し SECS-II インタフェースが標準化されることで利益を受ける。標準化によって装置サプライヤとデバイス製造業者のどちらもソフトウェア開発コストを削減できる。コストを減らして機能を向上させることによって、デバイス製造業者は半導体工場の自動化をもっと速く効果的に行うことができる。GEM スタンダードの持つ柔軟性のために、デバイス製造業者は業界で共通な枠組みの中で独自の自動化を実現することが可能になる。

GEM スタンダードでは次のことを明確にしたいと考えている:

- SECS-II の通信環境のもとで半導体製造装置が公に示す動作のモデル
- 半導体製造環境の中で必要とされる情報とコントロール機能の説明
- 半導体製造装置の持つ基本的な SECS-II 通信性能の定義
- ある動作を実行する時に, SECS-II が複数の可能な方法を提供する場合, その動作を完遂するための一貫した, ただ一つの方法
- 有効な通信性能を達成するための, 標準的なメッセージの対話シーケンス

GEM スタンダードの要求条件には 2 つのタイプがある:

- GEM の基本条件
- GEM の追加性能条件

GEM の基本条件が GEM スタンダードの基盤を形成する。GEM の追加性能は, あるタイプの工場自動化のために要求される機能性, あるいは, 特定の装置タイプに適用される機能性を提供する。GEM 基本条件および GEM の追加性能条件の詳細なリストは § 8, GEM 準拠に掲載してある。図 2 は GEM スタンダードの構成要素を示している。

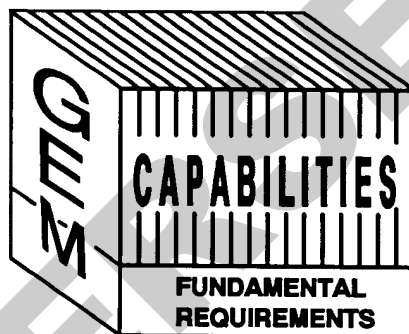


図 2
GEM の構成要素

装置サプライヤは, 特定の装置タイプについて GEM も追加性能のどれを適用するかを顧客と相談して決めなければならない。GEM スタンダードで定義された性能は, 半導体製造業者が実行する工場自動化のための要求条件を満たすように特に開発されたものなので, 特定のタイプの装置に適用される GEM 性能の大部分をほとんどのデバイス製造業者が必要とするだろうと考えられる。デバイス製造業者の中には工場自動化戦略の違いにより GEM 性能のすべては必要としないところがあるかも知れない。

1.4 概要 – 本 GEM スタンダードは次に説明するような項に分かれている。

第1 項 – 序文

この項では GEM スタンダードの改訂の記録, 範囲および目的を説明する。文書構成の概要および関連文書のリストも掲載する。

第2 項 – 定義

この項では本文書で使う用語を定義する。

第3 項－ 状態モデル

この項では状態モデルを説明するために本書全体で使用する慣用について述べる。また、すべての半導体製造装置に適用され、複数の性能に関係している基本的な状態モデルについても説明する。状態モデルはホストの観点から装置の動作を説明している。

第4 項－ 性能およびシナリオ

この項では半導体製造装置のために定義された通信性能を詳細に説明する。それぞれの性能の説明には、目的、定義、要求条件およびサポートされるシナリオが含まれている。

第5 項－ データ定義

この項では SEMI スタンダード E5 にみられるデータ項目辞書および変数項目辞書を掲載する。最初のサブセクションには使用法が限定されている(つまり許されるフォーマット) SECS-II からのデータ項目を記載する。2 番目のサブセクションにはデータ収集のためにホストが使用できる変数データ項目をリストアップし、その SECS-II の定義に制限があれば、それもすべて説明する。

第6 項－ 収集イベント

この項では必要な収集イベントのリストとそれに関連したデータを記述する。

第7 項－ SECS メッセージサブセット

この項では GEM スタンダードで定義されているすべての性能を実現するのに必要な SECS-II メッセージのリストを掲載する。

第8 項－ GEM 準拠

この項では GEM の基本条件および GEM の追加性能を述べており、スタンダードのこれ以外の項で詳細な要求条件が載っている部分についての参照リストを提供した。この項では、装置サプライヤやデバイス製造業者がこのスタンダードへの準拠を説明するときに使用できる標準的な用語や文書化の方法も定義する。

A 項－ アプリケーションノート

これらの項では追加の説明や例を記述する。

A1 項－ 工場での作業要綱

この項では、必要とされる SECS 性能が工場での典型的な作業シーケンスの中でどのように使われるか、その概要を説明する。この項は動作が実行される典型的なシーケンスに基づいて構成される。

A2 項－ 装置フロントパネル

この項では、必要とするフロントパネルボタンや指示装置やスイッチを本書で定義したように設置する際のガイダンスを記述する。フロントパネルに対する要求事項の要約も掲載する。

A3 項－ 装置アラームの例

この項ではさまざまな装置の構成に関連したアラームの例を載せる。

A4 項－ トレースデータ収集の例

この項ではホストが開始するトレースの例、および装置が送信する定期的なトレースデータメッセージの例を載せる。

A5 項－ ハレル表記法

この項では、状態モデルを説明するために本書で使ったデイビッド・ハレルの“ステートチャート”表記法について説明する。

A6 項－ コントロール状態アプリケーションの例

この項では装置コントロールモデルとホストの相互作用の一例を示す。

A7 項 – リミット監視の例

この項ではリミット監視の例を載せる。

1.5 参照スタンダードおよび文書

1.5.1 SEMI スタンダード – 次の SEMI スタンダード文書が GEM スタンダードに関連している。GEM に述べられているこれらスタンダードの特定の部分が GEM スタンダードの条項を構成している。

SEMI E4 – SEMI Equipment Communications Standard 1 – Message Transfer (SECS-I{ XE "SECS-I" })

SEMI E5 – SEMI Equipment Communications Standard 2 – Message Content (SECS-II{ XE "SECS-II" })

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

SEMI E37 – High-Speed SECS Message Services (HSMS) Generic Services

SEMI E139 – Specification for Recipe and Parameter Management (RaP)

1.5.2 その他の文書

Harel, D., "Statecharts: A Visual Formalism for Complex Systems," Science of Computer Programming 8 (1987) 231-274¹.

注意: 別途指示のない場合、引用されたすべての文書は最新の出版バージョンである。

2 用語

2.1 アラーム (alarm) – アラームとは、人間、装置あるいは処理している材料に危険を及ぼすかもしれないような、装置における異常状態に関連している。そのような異常状態は物理的な安全限界に基づいて装置製造業者が定義する。アラームの存在によって潜在的に影響を受けるような装置の動作は禁止される。

2.1.1 プロセスの許容値に関連してコントロールリミットを超過してもアラームを生成することはないし、処理の開始や完了といった装置の通常のイベントもアラームを生成しないことに注意する。

2.2 性能 (capabilities) – 性能とは半導体製造装置によって実行されるオペレーションである。オペレーションは SECS-II メッセージシーケンス (またはシナリオ) を使って通信インタフェースを介して始められる。性能の一例としてはアラームのセットおよび解除がある。

2.3 収集イベント (collection event) – 収集イベントとは、ホストにとって重要であると考えられる、装置におけるイベント (あるいは関連したイベントの集まり) である。

2.4 通信の喪失 (communication failure) – 通信の喪失は、確立された通信リンクが失われたときに起こるといわれている。そのような喪失は、プロトコル仕様である。通信喪失のプロトコル仕様の定義として、該当するプロトコル規格を参照のこと。(例 SEMI E4 または SEMI E37)

2.5 通信の誤り (communication fault) – 通信の誤りは、装置が予期していたメッセージを受け取れなかったとき、ならびにトランザクションタイムアウト、会話タイムアウトが発生したときに起こる。

2.6 コントロール (control) – コントロールするとは命令指示するという行為である。

2.7 装置モデル (equipment model) – 装置モデルとは、自動化された製造環境をサポートするために製造装置が実行しなければならない性能、シナリオ、および SECS-II メッセージに基づいた定義である (包括的装置モデルも参照のこと)。

2.8 イベント (event) – イベントとは装置にとって重要な、検出可能な出来事である。

2.9 GEM 準拠 (GEM compliance) – “GEM 準拠” という言葉は個々の GEM 性能に関して定義されており、ある性能が GEM スタンダードに忠実に従っていることを示す。GEM 準拠については § 8 でより詳しく述べる。

¹ Elsevier Science, P.O. Box 945, New York, NY 10159-0945; Website: <http://www.elsevier.nl>

2.10 包括的装置モデル (Generic Equipment Model) — 包括的装置モデルはあらゆるタイプの装置について述べる際の参照モデルとして使う。それはほとんどの装置に適用できる機能を持っているが、特定の装置が必要とする特有な条件については述べていない。

2.11 ホスト (host) — SEMI E4 および E5 スタンダードはホストを“装置と通信する知的なシステム”と定義している。

2.12 メッセージ誤り (message fault) — メッセージ誤りは、欠陥があるために処理することができないメッセージを装置が受信したときに起こる。

2.13 作業要綱 (operational script) — 作業要綱とは実際の工場での典型的な作業手順になるようにいくつかのシナリオを並べたものである。手順の例としては、システムの初期化、装置のセットアップ、およびプロセッシングがある。

2.14 オペレータ (operator) — 装置の持つ機能 (例えばプロセッシング) を実行するために装置を操作する人。オペレータは通常、装置の操作盤を介して装置を動かす。

2.15 プロセスユニット (process unit) — プロセスユニットとは、単独の動作開始コマンド、プロセスプログラムなどを介して単体として処理されるのが普通であるような材料のことを言う。一般的なプロセスユニットにはウェーハ、カセット、マガジン、それにポートがある。

2.16 プロセッシングサイクル (processing cycle) — プロセッシングサイクルとは、典型的なプロセスユニットに含まれる材料の全部が処理されるシーケンスの事である。動作や時間を測るものとして使われることがよくある。

2.17 シナリオ (scenario) — シナリオとは、ある性能を実行するための順番になるように並べられた SECS-II メッセージの集まりである。わかり易くするためにその他の情報もシナリオに含まれることがある。

2.18 SECS-I — SEMI 半導体製造装置通信スタンダード 1 (SEMI E4)。このスタンダードは、EIA RS232-C を基にした電氣的な信号レベルを持つメッセージ転送プロトコルのための方法を規定している。

2.19 SECS-II — SEMI 半導体製造装置通信スタンダード 2 (SEMI E5)。このスタンダードはメッセージのグループ、および半導体製造装置のコントロールに関連したメッセージについて個々の構文と意味を規定している。

2.20 SMS — SECS メッセージサービス。SECS-I の代わりになるもので、ネットワークで SECS-II フォーマットのメッセージを送信する際に使う。

2.21 状態モデル (state model) — 状態モデルとは状態と状態遷移を集めたもので、状態と状態遷移が結びついてシステムの動作を説明する。このモデルには、状態、状態内で起こる可能性のある動作／応答動作、状態間の遷移を引き起こすイベント、および状態間を遷移するプロセスなどを説明する諸条件の定義を含んでいる。

2.22 システムデフォルト (system default) — 装置の動作を表すモデルの状態のことを指し、システム初期化の直後に有効となる。また指定された装置変数がシステム初期化の直後にとることになっている値のことも指す。

2.23 システム初期化 (system initialization) — パワーアップ、システムの開始、および、システムのリセットの際に装置が実行するプロセス。このプロセスは、装置が適切にかつ動作モデルに従って動作するような状態に装置を持って行くためのものである。

2.24 ユーザ (user) — 工場を代表し、工場の作業モデルを実行する人(々)。ユーザは、装置が工場の作業に適合するためのセットアップや設定変更を行う責任があると見なされる。

3 状態モデル

以後の項では半導体製造装置のための状態モデルを記述する。状態モデルはホストの観点からみた装置の動作を簡潔で分かりやすい形式で説明している。状態モデルは異なった装置に関して、ある部分 (例えば、通信) では全く同じであり、別の部分 (例えば、プロセッシング) では違って来る。別々にモデル化して、それから組み合わせることができるような対等な部品に装置を分けるのがよいと考えられる。装置の構成要素の概要の例を図 3 に示す。

装置製造業者は、状態モデルの方法を使って自分達の作った装置の動作を文書化しなければならない。状態モデルは § 3.1, § A5, および関連文書に記載されている。状態モデルの記述は下記の 3 つの要素を含んでいなければならない:

- 状態図システムあるいはシステムの構成要素がとり得る状態、およびある状態から別の状態への遷移のすべてを示したもの。状態および遷移の一つ一つには名前がなければならない。ハレル表記法 (§ A5 参照) の使用を推奨する。
- 遷移表すべての遷移、開始状態および最終状態、どのような刺激で遷移が引き起こされるか、遷移の結果としてどのような動作がとられるかをリストアップしたもの。
- 各状態の定義その状態が活動状態 (ACTIVE) になった時のシステムの動作を指定したもの。

上記の要素の例をアプリケーションノート A5 に記載する。

EQUIPMENT	
COMMUNICATIONS	PROCESSING
CONTROL	ALARMS
SPOOLING	LIMITS MONITORIN
PROCESS RECIPES	MATERIAL MOVEMEN
TRACE REPORTS	TERMINAL SERVICES

図 3
装置構成要素の概観の例

状態モデルを使用する利点には以下のようなものがある:

1. 装置状態モデルは仕様作成ツールとして有用である。
2. ホストシステムは状態モデルに基づいて装置の動作を予測できる。
3. エンドユーザと装置プログラマーは装置の動作について共通の表現方法を持つことができ、そこを仕事の開始点にできる。
4. あらゆる装置の状態に適合する“正しい”オペレーションを定義できる。
5. 外部イベントの発生通知を内部の状態遷移に関連づけることができる。
6. 外部コマンドを状態遷移に関連づけることができる。
7. 装置コントロールのさまざまな様相を説明した状態モデルの構成要素を互いに関連づけることができる(例: 材料移送状態モデルを持つプロセッシング状態モデル; 内部装置安全システムを持つプロセッシング状態モデル)。

3.1 状態モデル方法論 — 本書で説明するさまざまな性能から期待される機能性を文書化するために、デビッド・ハレルの開発した“ステートチャート”を採用した。ハレルの著作は § 1.5 の参考文献欄に掲載されており、それはこの表記法を十分に理解するための“必読書”であると考えられる。本項および後の項に使われている慣用は性能の動的機能性を次の 3 項目で説明するためにある:

定義された状態あるいは下位状態の文項説明,

状態間の可能な遷移を説明した表、
状態と遷移の関係を示すための図で、ハレルが定義した記号を使ったもの。

これら 3 つの項目の組合せによってシステムや構成要素のための状態モデルを定義する。ハレル表記法の要約および、この方法を使って動作を定義するのに使う文項、表、図などの詳細な説明はアプリケーションノート A5 に掲載する。

状態モデルの基本単位は状態である。状態とはいくつかの条件の静的な集合である。条件が成立すると、それが現行の状態になる。これら条件の中にはセンサの読み取り値、スイッチの位置、時刻などが含まれるかも知れない。刺激(状態遷移を引き起こす)への応答動作の説明も状態定義の一部となる(例えば、メッセージ S_x, F_y を受信したら、応答メッセージ $S_x, F_y + 1$ を発生する)。刺激には非常に多くのものが考えられるが、半導体装置については SECS メッセージ受信、タイムアウト、装置ターミナルからのオペレータ入力、センサからの読み取り値の変化などが含まれるだろう。

本書ならびにこの項で説明している状態モデルをもっと明確に理解するためには、状態とイベントを区別し、それらの関係をはっきりさせることが役に立つ。イベントは静的であるというよりむしろ動的(ダイナミック)である。イベントは条件の変化を表す。もっと明確に言うそうといった変化を検出したことを表す。イベントにはセンサ読み取り値のリミットオーバー、スイッチの位置変化、タイムリミットオーバーなどが含まれるだろう。

新しく活動状態になった状態への変化(状態遷移)は常に条件の変更、つまりイベントの発生によって生じなければならない。付け加えると、状態遷移それ自体がイベントということになる。事実、装置に発生するイベントは数多くあるため、それが検出可能かどうか、関心があるかどうか、ということに基づいてイベントを分類することが重要である。本書ではイベントという言葉は狭義に定義しており、装置にとって重要な、検出可能な出来事であるとしている。

イベントの更に狭義の定義は“収集イベント”という言葉で代表される。これはホストにとって重要であると考えられる、装置におけるイベント(もしくは関連したイベントの集まり)である。(有効である場合に)ホストに報告されるのはこういったイベントである。この定義によると、ある装置についての収集イベントのリストは通常、全イベントのサブセットに過ぎないということになる。本書の状態モデルのレベルは、ホストが関心を持つ程度の詳しさに限定されている。したがって、このスタンダードで定義したすべての状態遷移は特記されていない限り収集イベントに対応することになる。

3.2 通信状態モデル — 通信状態モデルはホストとの通信リンクの有無について装置の動作を定義している。

§ 4.1 は本項を拡張して通信確立性能について定義する。どちらの視点も装置とホストの物理的な接続関係というよりは論理的な接続関係を視点に置いている。

3.2.1 用語説明 — 通信の喪失(communication failure)、接続トランザクションの喪失(connection transaction failure)、通信リンク(communication link)という言葉は本文書内で使うために定義されたものであり、ここ以外の場所で使われている同じ言葉あるいは類似した言葉と混同してはならない。

- 通信の喪失のプロトコル独自の定義については SEMI E4 (SECS-I)あるいは SEMI E37 (HSMS)を参照すること。
- 接続トランザクションの喪失は通信を確立しようとした時に発生し、次の事柄によって引き起こされる:
 - 通信の喪失
 - 応答タイムアウトリミット内に $S1, F14$ 応答を受信できなかったとき、あるいは
 - $S1, F14$ を受信したがフォーマットが適切でなかった、あるいは $COMMACK^2$ が 0 にセットされていなかった。
- 応答タイムアウト間隔は返信を必要とする一次メッセージ全部が正しく伝達されたあとと始まる。(応答タイムアウトのプロトコル独自の定義については SEMI E4 (SECS-I)あるいは SEMI E37 (HSMS)を参照すること。)

2 通信確立承認コード。§ 4.1 で定義されている。このデータ項目の詳細な定義については SEMI E5 スタンダードを参照。

- 通信リンクは“許諾”という返信をもって S1,F13/F14 トランザクションが最初に完了したあとに確立される。このリンクの確立は物理的と言うより論理的なものである。
- 実装方法によっては、出力メッセージを送信する前に一時的に蓄えておくメカニズムを持つこともある。このように蓄えたメッセージを表すのにキュー（待ち行列）という名詞を使う。メッセージをキューの中に置くときにキューに入れるといい、蓄えから取り除くときにデキューするという。
- 送信という言葉にはメッセージを“送信するためにキューに入れる”ことやメッセージを“送信しようとするプロセスを始める”ことが含まれる。送信という言葉にはメッセージの送信が正常に終了したという意味は含まれていない。
- ホスト初期化によって、あるいは通信の喪失をホストが独自に検出して、装置との通信を確立しようとすることがある。このように、ホストは S1,F13/F14 トランザクションをいつでも開始することができる。

3.2.2 通信遅延タイマー — 通信遅延タイマー (CommDelay timer): これは内部タイマーで、S1,F13 を送信する間隔を測るのに使用する。

この間隔は Establish Communications Timeout の値と等しい。この通信遅延タイマーはホストからは直接は見えない。

通信確立タイムアウト (Establish Communications Timeout): これはユーザが設定できる装置定数で、S1,F13 を送信する間隔を秒単位で決める。この値を使って通信遅延タイマーの初期設定を行う。

通信遅延タイマーの初期化は時間測定を開始するために行われる。通信遅延タイマーは遅延タイマータイムアウト待ち (WAIT DELAY) 状態に入ったときにのみ初期化される。

通信遅延タイマーが時間切れになるのは、タイマーが“タイムアウト”になり送信間隔の残り時間がゼロになったときである。遅延タイマータイムアウト待ち (WAIT DELAY) 状態の時にタイマーが時間切れになった場合、それが引金となって S1,F13 を新たに送信しようとする試みが行われ、さらに通信確立要求確認待ち (WAIT CRA³) 状態へ遷移するトリガーとなる。

3.2.3 規定

- S1,F13 を送信しようとする試みは通信確立要求確認待ち (WAIT CRA) 状態に遷移する際にのみ行われる。この時、通信遅延タイマーは“時間切れ”にセットしておかなければならない。
- 通信遅延タイマーは遅延タイマータイムアウト待ち (WAIT DELAY) 状態に遷移したときにのみ初期化される。S1,F13 を送信しようとする試みが次に行われるのは、通信確立要求確認待ち (WAIT CRA) 状態に遷移したときのみである。

3.2.4 通信状態 — SECS 通信には大きく分けて 2 つの状態がある: 通信無効 (DISABLED) と通信有効 (ENABLED) である。システムがデフォルトでとる状態はユーザが (例えば、ジャンパーセッティングや不揮発性の記憶変数を介して) 装置で設定変更することができなければならない。

一旦システムが初期化されたら、オペレータは装置ターミナル機能やモーメンタリスイッチを使っていつでも通信状態を選択し直すことができる。システムのデフォルトと矛盾するかも知れないのでツープозиションタイプのスイッチを使ってはならない。

通信有効状態には 2 つの下位状態がある。以下に説明する、通信中断 (NOT COMMUNICATING) と通信実行 (COMMUNICATING) である。装置は、通信中断と通信実行の下位状態を含めた現在の通信状態を常に表示してオペレータに知らせなければならない。

接続トランザクション喪失の時には、ユーザが設定できる装置定数 Establish Communications Timeout により、通信中断 (NOT COMMUNICATING) 下位状態での、S1,F13 (通信確立要求) の送信時間間隔を設定する。

3 CRA とは Establish Communication Request Acknowledge (通信確立要求確認) (S1,F14) という言葉を覚えておくために定義した略語である。

図 4 に通信状態モデルの上位と下位の状態間の関係を示す。状態遷移を引き起こすイベントと発生する動作を表 1 に示した。通信状態モデルを次のように定義する。

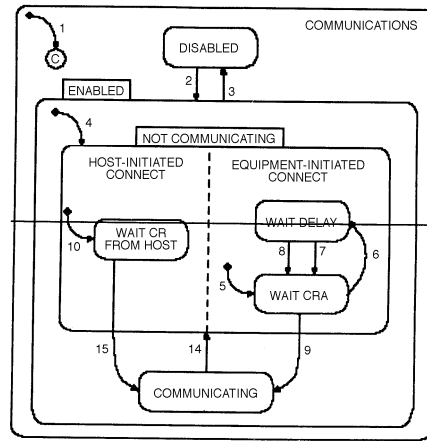


図 4
通信状態図

通信無効(DISABLED)

この状態にあるときにはホストコンピュータとの SECS-II 通信は存在しない。オペレータが通信有効から通信無効にスイッチを切り換えたなら、すべての SECS-II 通信は即座に停止される。送信のためにキューに入っていたメッセージはすべて破棄され、実行中のトランザクションや会話に対しては、それ以降の動作をすべて終了する⁴。その時に転送中であったメッセージの取り扱いは下位レベルのメッセージトランスファープロトコルの問題であり、このスタンダードでは取り扱わない。

通信無効(DISABLED)状態はシステムデフォルトの一つである。

通信有効(ENABLED)

通信有効(ENABLED)には 2 つの下位状態がある。通信実行(COMMUNICATING)と通信中断(NOT COMMUNICATING)である。システム初期化の間、あるいはオペレータの選択により、通信有効になったときには、通信が正式に確立されるまで、必ず通信中断という下位状態にある。SECS-II に基づいた通信をサポートすることができるように、(SECS-I のような)下位レベルのプロトコルは正常に機能することを前提にしている。

通信有効(ENABLED)状態はシステムデフォルトの一つである。

通信有効(ENABLED)／通信中断(NOTCOMMUNICATING)

この下位状態にある間は、S1,F13, S1,F14 と S9,Fx 以外のメッセージは送信されない。装置は S1,F13 か S1,F14(通信確立確認)以外のメッセージをホストから受信してもすべて破棄する。装置はまたホストコンピュータとの通信確立が成功するまで定期的に S1,F13 を送信して通信を確立しようとする。しかし、どのような時でも装置が開始するただ一つの S1,F13 トランザクションのみがオープンとなる。

通信中断状態には 2 つの AND 下位状態がある。ホスト開始接続(HOST-INITIATED CONNECT)と装置開始接続(EQUIPMENT-INITIATED CONNECT)である。装置が通信中断状態にあれば、このどちらも機能する。これら 2 つの下位状態は装置およびホストが同じ時間帯に通信確立を実行するときの装置の動作を明らかにしている⁵。

4 SECS-II トランザクションおよび会話プロトコルの定義については SEMI E5, § 5 を参照のこと。

5 ハレル表記によれば、どんな AND 下位状態でもそこから出ることは、その親の状態から出ることを意味する。したがって、その親の状態のすべての下位状態からも同じく出ることを意味する。

通信中断 (NOT COMMUNICATING) / 装置開始接続 (EQUIPMENT INITIATED-CONNECT)

この状態には通信確立要求確認待ち (WAIT CRA) と遅延タイムアウト待ち (WAIT DELAY) の 2 つの下位状態がある。通信中断状態になると、ただちに装置開始接続状態となり通信確立要求確認待ちへの状態遷移が起こるとともに、通信遅延タイマーが“時間切れ”にセットされ、すぐに S1,F13 を送信しようとする。

通信中断 (NOT COMMUNICATING) / 装置開始接続 (EQUIPMENT-INITIATED CONNECT) / 通信確立要求確認待ち (WAIT CRA)

通信確立要求が送信された。装置はホストがその要求を承認するのを待つ。

通信中断 (NOT COMMUNICATING) / 装置開始接続 (EQUIPMENT-INITIATED CONNECT) / 遅延タイムアウト待ち (WAIT DELAY)

接続トランザクションが失敗した。通信遅延タイマーを初期化する。装置はタイマーの時間切れを待つ。

通信中断 (NOT COMMUNICATING) / ホスト開始接続 (HOST INITIATEDCONNECT)

この状態は、通信中断状態においてホストが開始した S1,F13 に対応する装置の動作を説明する。

通信中断 (NOT COMMUNICATING) / ホスト開始接続 (HOST INITIATEDCONNECT) / ホストからの通信確立要求待ち (WAIT CR FROM HOST)

装置はホストからの S1,F13 を待つ。S1,F13 を受信すると、装置は COMMACK=0 の S1,F14 を送信する。

有効 (ENABLED) / 通信実行 (COMMUNICATING)

通信が確立された。装置は S1,F13 も含めて、あらゆるメッセージをホストから受信することができる。装置が通信実行状態にあるときは、ホストコンピュータとの間の SECS 通信は維持されなければならない。通信が無効にされるか通信の喪失が発生するまでこの状態は持続する。通信実行下位状態の間に装置がホストから S1,F13 を受信したときには、COMMACK を 0 にセットした S1,F14 で応答しなければならない。以前に送信した S1,F13 に対して S1,F14 を受信しても、装置は何もする必要はない。

通信喪失が発生した場合には、装置は通信中断下位状態へ戻り、ホストとの通信を再確立しようと試みる。

装置は、ホスト開始接続/ホストからの通信確立要求待ち状態で、ホストからの S1,F13 を受信したときに、同時に装置開始接続/通信確立完了待ち状態で、ホストからの S1,F14 を待っていることも有り得る。この状態が発生したときには、装置およびホストの各々がオープンな S1,F13/S1,F14 トランザクションを持つことになる。どちらでも S1,F13/S1,F14 トランザクションが正常に終了し、通信が確立されると、これら 2 つのトランザクションのうちはじめに完了したほうが通信中断から通信実行への遷移を引き起こす。この場合、もう一つのトランザクションは通信実行への遷移にかかわらず正常にトランザクションが完了するまではオープンのままとなる。

通信実行状態になったときに、装置が以前に受信した S1,F13 に対する S1,F14 をまだ送信していなかったとしても⁶、S1,F14 の応答は通常どおり返信される。S1,F14 の送信失敗は他の通信喪失と同様に取り扱いわれる。

装置が開始した S1,F13/S1,F14 が通信が実行状態になっても依然としてオープンである場合には、その後発生したホストからの返信メッセージの受信異常は、通信の誤り (COMMUNICATION FAULT) とみなされる。トランザクションタイマータイムアウトが発生したときには S9,F9 が送信されなければならない⁷ (通信誤り (COMMUNICATION FAULTs) およびメッセージの欠陥 (message faults) の定義および、ストリーム 9 エラーメッセージの詳細については § 4.9 を参照すること)。

⁶ これは通信実行状態への遷移が起きたときに、開始されたがまだ完了していないトランザクションがあればそれも含む。

⁷ トランザクションタイマーは SMS (SEMI E13) のようなプロトコルに関しては必要条件ではない。

3.2.5 状態遷移 – 表 1 には、図 4 に掲載した状態遷移の完全な説明を載せた。

オペレータが通信無効 (DISABLED) 状態から通信有効 (ENABLED) 状態にスイッチを切り換えても何の収集イベントも起こらない。と言うのは、通信が確立されるまでどんなメッセージも送信できないからである。通信確立というプロセスは通信が有効 (ENABLED) になっているということをホストに知らせる役目を果たしている。通信状態モデルに関してはこれ以外の収集イベントは定義されていない。

表 1 通信状態遷移

#	現在の状態	トリガー	新しい状態	動作	コメント
1	(通信状態に入る)	システムの初期化	システムデフォルト	なし	システムのデフォルトは通信無効または有効に設定される。
2	DISABLED (通信無効)	オペレータが通信無効から通信有効に切り換える。	ENABLED (通信有効)	なし	SECS-II 通信が有効になる。
3	ENABLED (通信有効)	オペレータが通信有効から通信無効に切り換える。	DISABLED (通信無効)	なし	SECS-II 通信が禁止される。
4	(通信有効に入る)	通信有効状態に入る。	NOT COMMUNICATING (通信中断)	なし	システムの初期化から通信有効に入ってもよいし、オペレータが通信有効に切り換えてもよい。
5	(装置開始接続に入る)	(通信中断状態に入る。)	WAIT CRA (通信確立要求確認待ち)	通信初期化。 CommDelay タイマーを時間切れにセット。 S1,F13を送信。	通信確立開始。
6	WAIT CRA (通信確立要求確認待ち)	通信トランザクションの失敗	WAIT DELAY (遅延タイマータイムアウト待ち)	CommDelay タイマーを初期化する。送信するためにキューに入れておいたメッセージをすべてキューから出す。	適切な場合には、キューから出されたメッセージは生成順にスプールバッファに入れる。タイマーが時間切れになるのを待つ。
7	WAIT DELAY (遅延タイマータイムアウト待ち)	通信遅延タイマーが時間切れになる。	WAIT CRA (通信確立要求確認待ち)	S1,F13 送信。	S1,F14 を待つ。ホストからの S1,F13 を受信することもある。
8	WAIT DELAY (遅延タイマータイムアウト待ち)	S1,F13 以外のメッセージを受信する。	WAIT CRA (通信確立要求確認待ち)	メッセージを捨てる。応答なし。通信遅延タイマーを“時間切れ”にセットする。S1,F13 送信。	通信を確立するチャンスがあることを意味する。
9	WAIT CRA (通信確立要求確認待ち)	待っていた、COMMACK=0 の S1,F14 を受信する。	COMMUNICATING (通信実行)	なし	通信が確立される。
10	(ホスト開始接続に入る)	(通信中断状態に入る。)	WAIT CR FROM HOST (ホストからの通信確立待ち)	なし	ホストからの S1,F13 を待つ。
14	COMMUNICATING (通信実行)	通信の喪失。(通信の喪失のプロトコル独自の定義については SEMI E4(SECS-I)または SEMI E37(HSMS)を参照すること。)	NOT COMMUNICATING (通信中断)	送信のためにキューに入っていたメッセージをすべてキューから取り除く。	キューから取り除かれたメッセージは必要に応じてスプールバッファに入れる。

#	現在の状態	トリガー	新しい状態	動作	コメント
15	WAIT CR FROM HOST (ホストからの通信確立要求待ち)	S1,F13を受信する。	COMMUNICATING (通信実行)	COMMACK=0 で S1,F14を送信する	通信が確立されている。

3.3 コントロール状態モデル コントロール状態モデルは、ホストと装置が協調的に動作するときの協調の程度(レベル)を定義したものである。ここではホストが装置をコントロールする場合のさまざまなレベルにおいて、オペレータがどのように介入するかということも説明する。通信状態モデルが、ホストと装置間のメッセージ交換能力について述べているのに対し、コントロールモデルは、装置がメッセージを受信したときにどのように動作しなければならないかについて述べている。

コントロールモデルはホストに対し次の 3 つの基本的なコントロールのレベルを提供する。もっとも高度なレベル(リモート)においては、ホストは実現可能な範囲での完全な装置コントロールを行う。中間のレベル(ローカル)では、ホストは装置内のすべての情報にアクセスすることを許されるが、装置の操作に関してはある程度の制限が設けられる。もっとも低いレベル(オフライン)では、装置はホストからのコントロールを一切受け付けず⁸、非常に限られた範囲の情報しか提供しない⁹。

コントロールモデルと通信モデル(実装されている場合)は互いに直接的に影響し合うことはない。つまり、一方のモデルの動作または状態が、他方のモデルの動作の変化を引き起こす直接の原因とはならない。しかし、通信状態が通信中断(NOT COMMUNICATING)のときは、ほとんどのメッセージが機能しないことも確かである。メッセージが伝達されなければコントロール性能や他のすべての GEM の性能も影響を受ける。

図 5 を参照のこと。コントロール下位状態および状態遷移を定義した。

オフライン(OFF-LINE)

オフライン状態では、オペレータが装置の操作盤から操作を行う。装置がオフライン状態であってもメッセージの伝達は可能である。しかし、自動化を目的とするメッセージの使用は厳しく制限される。オフライン状態の時には、装置は通信確立のためのメッセージ、もしくはホストからのオンライン移行要求に対してしか応答しない。

オフライン状態の間は、S1,F13 もしくは S1,F17 を除いてはホストからの一次メッセージに対し、装置は常に Sx,F0 を返信する。装置は S1,F13 および S1,F17 については処理を行い応答を返す。S1,F17 は、ホストが装置にオンライン状態への移行を要求するときに使われる。装置はホストオフライン状態にあるときにのみこの要求を受け付ける。(後述の遷移 11 に関する定義を参照のこと)。

オフライン状態の間は、装置は S1,F13¹⁰、S9,Fx¹¹と S1,F1 を除いては、一次メッセージを送信をしようとするのではない(オンライン試行下位状態参照)。装置は S1,F14 もしくは S1,F2 以外の応答メッセージをホストから受信すると、そのメッセージを破棄する。

システムがオフライン状態にあるときはメッセージのスプールは発生しない。通信中断状態となる時に、スプール活動状態となる可能性がある。この状態はオフラインの時に発生するかもしれないが、装置は上記12以外のメッセージを送信することはないのだから、結果的にスプールされるメッセージは発生しない。

オフラインには 3 つの下位状態がある: 装置オフライン(EQUIPMENT OFF-LINE)、オンライン試行(ATTEMPT ON-LINE)、およびホストオフライン(HOST OFF-LINE)である。

オフライン/装置オフライン(OFF-LINE/EQUIPMENT OFF-LINE)

この状態の時には、システムはオフライン状態を保ち、オペレータからのオンライン移行指示を待つ。

8 ホストは通信確立動作を行うことはできる。これは装置の操作に影響を与えないためコントロール操作の範疇には入らない。

9 ホストは S1,F13/F14 トランザクションを利用すれば装置の確認はできる。

10 S1,F13 の送信は通信状態モデルに基づいて行われる。

11 S9,Fx メッセージはオフライン時に正規に応答するメッセージの返信としてのみ発行される(つまり、S1,F13 および S1,F17)。

12 装置は S1,F1 もしくは S1,F13 を送信するかもしれないが、ストリーム 1 のメッセージはスプーリングに適していないので、やはりスプールには入らない。

オフライン／オンライン確立試行(OFF-LINE/ATTEMPT ON-LINE)

オンライン確立試行状態の時には、装置はオペレータからのオンライン状態への移行指示に応答している。この状態に入るとすぐに装置は S1,F1 をホストに送信する。

この状態のときは、オペレータによるオンラインまたはオフラインへのスイッチ切り替え操作には反応しないことに注意する。

オフライン／ホストオフライン(OFF-LINE/HOST OFF-LINE)

ホストオフラインとは、オペレータが装置をオンラインにしようとしたが、ホストの許可が得られない状態である。

この状態に入るには、オンラインに移行する試みが失敗したか、ホストが装置にオンラインからオフラインへと移行するよう要求したかのどちらかである(詳細については遷移表 5 を参照)。この状態の時には、装置はホストからのオンライン移行要求(S1,F17)を受け入れる。ホストオフライン状態以外の時にはそのような要求は拒絶する。

オンライン(ON-LINE)

オンライン状態の時には、SECS-II メッセージが交換され、それに基づいて動作が実行される。装置の操作盤から実際に実行できる性能はホストからも同様に実行できなければならない。

オンライン状態のときは、Sx,F0 メッセージを使う必要性はない。また使わないよう推奨する。もし使用するとしたら、メッセージの異常に関連して、トランザクションを終了するときだけである。

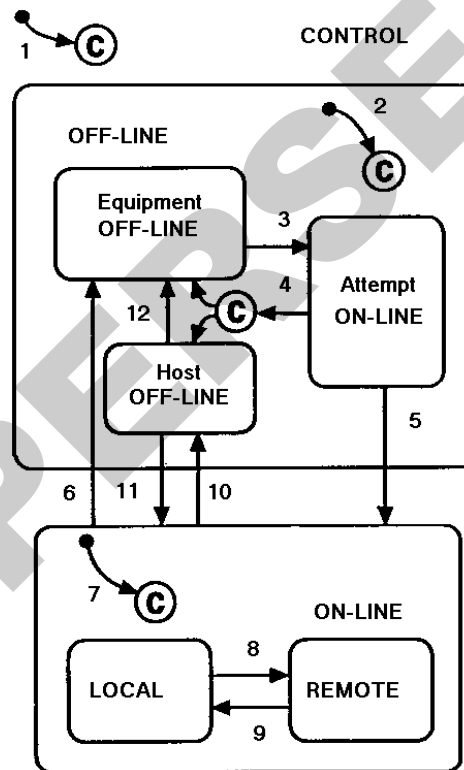


図 5
コントロール状態モデル

オンライン／ローカル(ON-LINE/LOCAL)

オペレータが直接装置の操作を行う。装置の操作盤からあらゆる操作ができる。

ホストはローカル状態において次のような性能ならびに制限を持つことになる：

- ホストは、物理的な動作を起こさせたり処理を開始させるリモートコマンドの使用を禁止される。処理中は、ホストはその処理に影響を与えるリモートコマンドを使うことはできない。
- 処理中は、ホストはその処理に影響を与える装置定数の修正はできない。その他の装置定数については処理中でも変更できる。処理中でなければ、ホストはすべての利用可能な装置定数を修正できる。
- ホストは、装置のレシピ格納領域に対しレシピのアップロードおよびダウンロードを開始できる。ホストは、現在実行中のレシピに影響を与えない限り、次に実行するレシピを選択できる。
- ホストは、アラーム、イベントレポート、トレースデータレポートを含むデータの自動報告機能を装置に対し設定できる。ホストはそういった報告を適時受信する。
- ホストは、ステータスデータ、装置定数、イベントレポート、プロセスレシピディレクトリ、およびアラームを含むデータを装置に要求できる。
- 装置は、GEM で定義されている端末サービスを実行できる。

ホストは、以上述べた内容により特に制限されない限り、他のどんな機能でもローカル状態で使うことができる。

注 1: 上に述べた機能のうち、対象とする装置で実装されないものについては考慮しなくてもよい。

オンライン/リモート(ON-LINE/REMOTE)

リモートコントロールの GEM 機能 (§ 4.4 参照) に対応する装置であれば、リモート状態の時には、ホストは通信インタフェースを介して可能な範囲で、装置を操作する。リモート状態のとき、装置はホストの性能に何等の制限を加えない。ホストによってコントロールされる程度は工場によって色々と異なってくる。あるときにはリモートコントロールによる処理が行われている間でも、オペレータによる装置の操作が必要になるケースもあるかも知れない。それらの操作にはセットアップやオペレータによる補助などが含まれる。このようなさまざまな状況に対応できる柔軟性を持つよう、この状態は考えられている。

オペレータは特定の性能について制限を受けることがある。この制限は設定変更が可能であり、ホストと競合することなくオペレータが必要な性能を実施できるよう装置を設定できる。設定の種類には次のようなものが含まれる(ただし、ここに挙げたものにだけに限定はされない):

- 装置定数の変更(処理関連)
- 装置定数の変更(処理関連以外)
- プロセスレシピダウンロード開始
- プロセスレシピ選択
- プロセスレシピ開始
- プロセスレシピ一時停止/再開
- オペレータによる補助操作
- 装置に対する材料搬送
- 装置指定のコマンド(必要ならコマンド毎)

注 2: 上記の性能のうち、装置で実現されていないものについては無視してよい。

オペレータがローカル状態で使用できる性能をリモート状態になったからといって無条件に制限してはならない。

サプライヤは、必要があれば、上記のリストに含まれていない性能についても設定変更可能な制限を加えてもよい。ホストが利用できないようなオペレータ機能については、ここで述べたような設定変更機能は必要とされない。

コントロール機能はある範囲内でホストと現場のオペレータの間で分担される必要がある。少なくとも、オペレータはコントロール状態の変更、緊急停止、処理の中断(例えば、ストップ、アバート、ポーズなど)の能力を持たなければならない。上記のうち緊急停止以外の機能は使用が制限されることもある¹³。

ホスト側のソフトウェアはオペレータに割り当てられている性能と同等(コンパチブル)の性能を持つようデザインされなければならない。

表 2 コントロール状態遷移

#	現在の状態	トリガー	新しい状態	動作	コメント
1	(未定義)	コントロール状態に入る。 (システム立ち上げ)	CONTROL コントロール状態(下位状態は設定により異なる)	なし	装置はデフォルト設定でオンラインもしくはオフラインになる(注 1 参照)。
2	(未定義)	オフライン状態に入る。	OFF-LINE オフライン状態(下位状態は設定により異なる)	なし	装置はデフォルト設定でオフラインのどんな下位状態にもなる。
3	EQUIPMENT OFF-LINE (装置オフライン)	オペレータがスイッチをオンラインに切り替える。	ATTEMPT ON-LINE オンライン試行	なし	オンライン確立状態にあるときはいつでも S1,F1 が送信されることに注意。
4	ATTEMPT ON-LINE (オンライン確立試行)	S1,F0	設定条件により異なる新しい状態	なし	通信の喪失、返信タイムアウト、もしくは S1,F0 の受信による(注 2 参照)。設定条件により装置オフライン、もしくはホストオフラインに移行する。
5	ATTEMPT ON-LINE (オンライン確立試行)	装置はホストから期待した S1,F2 を受信する。	ON-LINE オンライン	なし	ホストは遷移 7 でオンラインに移行することを通知される。
6	ON-LINE (オンライン)	オペレータがスイッチをオフラインに切り替える。	EQUIPMENT OFF-LINE 装置オフライン	なし	“装置オフライン”イベント発生(注 3 参照)。オフラインのとき、イベント返信メッセージは捨てられる。
7	(未定義)	オンライン状態に入る	ON-LINE オンライン状態(下位状態はリモート/ローカルのスイッチの設定によって決まる)。	なし	“コントロール状態ローカル”または“コントロール状態リモート”イベント発生。イベントレポートは実際に移行したオンラインの下位状態を示す。
8	LOCAL (ローカル)	オペレータがフロントパネルのスイッチをリモートにセットする。	REMOTE リモート	なし	“コントロール状態リモート”イベント発生。
9	REMOTE (リモート)	オペレータがフロントパネルのスイッチをローカルにセットする。	LOCAL ローカル	なし	“コントロール状態ローカル”イベント発生。
10	ON-LINE (オンライン)	装置は“オフライン切り替え”メッセージ(S1,F15)をホストから受け取る。	HOST OFF-LINE ホストオフライン	なし	“装置オフライン”イベント発生。
11	HOST OFF-LINE (ホストオフライン)	装置はオンライン移行要求(S1,F17)を了解する。	ONLINE オンライン	なし	ホストは遷移 7 でオンラインに移行することを通知される。
12	HOST OFF-LINE (ホストオフライン)	オペレータがスイッチをオフラインに切り替える。	EQUIPMENT OFF-LINE 装置オフライン	なし	“装置オフライン”イベント発生

注 1 遷移 1 および 2 で述べている条件設定は単一設定でなければならない。これによってユーザは、装置オフライン、オンライン確立試行、ホストオフライン、オンラインのどれに入るか選択することができる。

注 2 通信の喪失は、プロトコル仕様である。通信喪失のプロトコル仕様の定義として、該当するプロトコル規格を参照のこと。(例 SEMI E4 または SEMI E37)

13 オペレータの操作を制限する方法(パスワード、キー等)に関する規定はこの文書の範囲外である。

注 3 装置に対してホストが開始したトランザクションはすべて完了しなければならない。そうするには、イベントメッセージの送信に先立ってホストに適切な返信メッセージを送信するか、イベントメッセージの後で(つまりトランザクション終了後) Sx,F0 を送信する。

3.4 装置プロセッシング状態 — 装置が意図した機能を実行するためにとる動作は文書として示されねばならない。このプロセッシング状態モデルは個々の装置のプロセス、技術、スタイルに強く依存している。しかし、それらモデルの中にも共通の側面があると考えられる。

図 6 のプロセッシング状態図は実施するモデルの一例として掲載した。このモデルは、プロセッシング状態モデルを文書によって表現するときのあるべき姿を表しているだけであり、これらの状態をそのまま実施しなければならないということではない。

装置はそれぞれのプロセッシング状態遷移において収集イベントを生成しなければならない。また現在のプロセッシング状態と以前のプロセッシング状態を表す値である状態変数(現プロセス状態(Process State), 前プロセス状態(Previous Process State))も提供しなければならない。

例に示したプロセッシング状態図について言えば、初期状態 INIT は実際のプロセッシング状態の一つではないことに注意する。これは、装置のシステム初期化が終了するとすぐに IDLE プロセッシング状態に入ることを示すためだけに書かれている。図 3.4 で記述した装置プロセッシング状態と状態遷移(番号が打ってある)についての詳細な説明をこのあとに載せた。

3.4.1 装置プロセッシング状態の説明

IDLE

装置は指示待ち状態にある。

PROCESSING ACTIVE

この状態はすべての下位状態の親となるもので、プロセスレシピ実行の一連の動作記述はその下位状態にある。

PROCESS

この状態は、プロセスプログラムの準備と実行に関連する下位状態の親となる。

SETUP

この状態にあるときは、プロセス実行に必要な外的条件を満たすための準備をしている。例えば、材料は装置にあり、入出力ポートは正しい状態にあり、温度や圧力といったパラメータは制限値内にある、などの準備を整える。もしすべてのセットアップ操作が完了すれば、ここは通過可能な状態となり、次の状態への遷移が起こる。

READY

この状態にあるとき、装置はプロセスを実行する準備が整っており、オペレータもしくはホストからの開始コマンドを待っている。

EXECUTING

装置がプロセスレシピを自動的に実行しており、外部からの干渉なしに動作し続けることができる状態である。

PAUSE

この状態の時、プロセスは一時停止され装置は指示を待っている。

それぞれの状態遷移は下記の表に定義した。この表に載っている遷移はすべて収集イベントと見なされなければならないことに注意する。

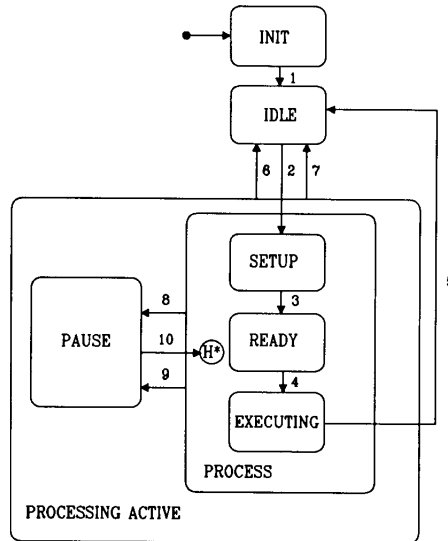


図 6
プロセッシング状態図

表 3 プロセッシング状態遷移

#	現在の状態	トリガー	新しい状態	動作	コメント
1	INIT	装置の初期化完了。	IDLE	なし	なし
2	IDLE	セットアップ命令が出された。	SETUP	なし	なし
3	SETUP	セットアップ活動はすべて完了し、装置は開始指示を受信する用意が整っている。	READY	この活動は装置によって異なる。	なし
4	READY	装置はホストもしくはオペレータコンソールから開始指示 (START) を受信した。	EXECUTING	この活動は装置によって異なる。	なし
5	EXECUTING	プロセッシング作業が完了した。	IDLE	なし	なし
6	PROCESSING ACTIVE	装置はホストもしくはオペレータコンソールから停止指示 (STOP) を受信した。	IDLE	なし	なし
7	PROCESSING ACTIVE	装置はホストもしくはオペレータコンソールから中断指示 (ABORT) を受信した。	IDLE	この活動は装置によって異なる。	なし
8	PROCESS	装置はたとえばアラームのような条件により、一時停止 (PAUSE) することに決めた。	PAUSE	この活動は装置によって異なる。	このタイプの異常については普通オペレータの補助が必要である。
9	PROCESS	装置はホストもしくはオペレータコンソールから一時停止指示 (PAUSE) を受信した。	PAUSE	この活動は装置によって異なる。	なし
10	PAUSE	装置はホストもしくはオペレータコンソールから一時停止解除指示 (RESUME) を受信した。	前 PROCESS サブステート	この活動は装置によって異なる。	なし

4 装置性能とシナリオ

この項では GEM で要求される性能の詳細と、その性能を適用するためのシナリオを記述した。性能とは半導体製造装置によって遂行されるオペレーションである。このオペレーションは SECS-II メッセージを使った通信インタフェースを介して開始される。シナリオとは性能を遂行するための順序に並べられた SECS-II メッセージの集まりである。理解を助けるためにその他の情報もシナリオに含まれることがある。読者のために、性能一つ一つについて目的、関連する定義、詳細な説明、要求条件、シナリオを記述した。

本項では下記の性能について述べる:

- 通信確立
- イベント通知
- 動的なイベント報告の設定
- 変数データ収集
- トレースデータ収集
- リミット監視
- 状態データ収集
- オンラインの確認
- アラーム管理
- リモートコントロール
- 装置定数
- プロセスレシビ管理
- 材料移送
- 装置ターミナルサービス
- エラーメッセージ
- クロック
- スプーリング
- コントロール

4.1 通信確立 – 通信確立性能は、システムの立ち上げのあと、あるいは、通信相手との通信が途絶えたあとに通信を正式に確立する手段を提供し、それによって、通信相手に通信が不能な期間があったことを知らせることができる。

4.1.1 目的 – ホストと装置との間の通信は、通信確立要求/通信確立確認のトランザクションを使うことによって正式に確立される。

S1,F1/F2 のトランザクションは通信確立以外の目的にも使われることがあり、いつ起こるかも知れないので、通信確立の目的にこのトランザクションを使うことは曖昧である。(訳者注): S1,F1/F2 の使用方法については別途“コントロール状態モデル”で正確に定義されている。

通信状態モデルと関連して S1,F13/F14 を使うことにより、装置とホストの間で相互に通信が不可能な期間があったことを互いに通知できる。このトランザクションの正常終了は、ホストと装置の間でなんらかの同期が必要かもしれないということを示している。

4.1.2 定義

通信了解コード(COMMACK) – 通信確立確認メッセージで返される了解コード。このデータ項目の完全な定義については SEMI E5 スタンドダードを参照のこと。

通信確立タイムアウト(Establish Communications Timeout) – 通信確立要求を再び送信する間隔を表す装置定数。この値が何秒間隔かを定める。この変数データ項目の完全な定義については SEMI E5 スタンドダードを参照のこと。

4.1.3 説明 – 通信リンクの一方に異常が発生し、他方がそれに気づかないというような場合には幾つかの潜在的な問題がある。ホストから見れば、通信の喪失は多くの考えうる動作の原因をつくり出す。ホストによって制御さ

れている装置の設定値をリセットする必要がある場合もあるだろうし、通信が喪失している間にも装置が自動的に処理を続行したために装置の状態が変わっているかも知れない。通信確立のための正規のプロトコル定義は、装置の現在の状態とホストの常態の同期化が必要なことをホストに知らせる手段を提供する。

装置は下記の条件のどちらかが満たされたら、いつでも正式に通信が確立されたものと見なす¹⁴:

- 装置は通信確立要求をホストに送信し、トランザクションタイムアウトの時間内に確認の了解コードを持った通信確立確認をホストから受信した、もしくは
- 装置は通信確立要求をホストから受信し、確認の了解付きの通信確立要求確認の応答を正常に送信できた。

装置がホストに通信確立要求を送信することで、装置との同期が必要なことをホストに知らせることになる。

装置が通信を確立しようとしているときには、上記のように通信が正式に確立するまで通信確立要求を定期的を送信する。送信する間隔はユーザが設定できなければならず、接続トランザクションの失敗が検出されたときからこの間隔で送信が始まる (§ 3.2 参照)。

通信を確立しようとする試みは下位レベルの接続機能の問題ではなく、ホストが装置との同期化を行う必要があるかもしれない、ということをお互いに相手に知らせるために使われる論理的なアプリケーションの問題である。

4.1.4 要求条件

- 装置は通信状態モデルをサポートする (§ 3.2 参照)。
- 装置は上記の通信確立タイムアウト (Establish Communications Timeout) 装置定数を提供する。

4.1.5 シナリオ

4.1.5.1 ホストからの通信確立

コメント	ホスト	装置	コメント
通信確立要求	S1,F13-->	<--S1,F14 COMMACK = 受諾で応答 通信状態 = 通信実行 (COMMUNICATING)	通信状態は通信有効 (ENABLED) である (下位状態は何でもよい)

4.1.5.2 装置からの通信確立, ホストの確認応答

コメント	ホスト	装置	コメント
通信確立要求確認	S1,F14-->	<--S1,F13 通信確立要求	通信状態 = 通信中断 (NOT COMMUNICATING) [LOOP] [LOOP] ----SEND
		[IF] S1,F14 がタイムアウトなしに受信されたら [THEN] ループを出る----SEND [ELSE] EstablishCommunications-Timeout の間隔分遅延する [ENDIF] [END_LOOP] ----SEND [IF] COMMACK = 受諾 [THEN] 通信状態 = 通信実行 (COMMUNICATING) ループを出る---- [ELSE] 遅延するためにタイマーをリセットし, Establish-CommunicationsTimeout で指定された間隔分遅延する [ENDIF] [END_LOOP]	

¹⁴ この 2 つの条件のどちらかが満たされると、その結果、通信実行 (COMMUNICATING) 下位状態への遷移がおこる。詳細については § 3.2 を参照のこと。

4.1.5.3 通信確立動作の同時発生 — インターリーブをサポートする装置に関しては、ホストまたは装置のどちらも通信相手からの要求を受信する前に通信確立要求を送信することがある。どちらかの通信確立要求が正常に受け入れられると通信が確立するので、どちらが先に要求を出したかは問題ではない。ホストと装置の役割が入れ代わることもありうる。

装置は S1,F14 を送信する前にホストからの S1,F14 を受信

コメント	ホスト	装置	コメント
		通信状態=通信中断	
		<--S1,F13	通信確立要求
通信確立要求	S1,F13-->		
COMMACK=受諾で応答	S1,F14-->		ホストから S1,F14 を受信、通信が確立され ¹⁵ 、通信状態=通信実行 (COMMUNICATING)
		<--S1,F14	COMMACK=受諾 ¹⁶ で応答

装置は S1,F14 を受信する前にホストへ S1,F14 を送信

コメント	ホスト	装置	コメント
		通信状態=通信中断 (NOT COMMUNICATING)	
		<--S1,F13	通信確立要求
通信確立要求	S1,F13-->		
		<--S1,F14	COMMACK=受諾 ¹⁵ で返信。 通信が確立された ¹⁵ そして 通信状態=通信実行 (COMMUNICATING)
COMMACK=受諾で応答	S1,F14-->		ホストから S1,F14 を受信した

4.2 データ収集 — データ収集により、ホストは装置の状態をモニタすることが出来る。データ収集は、イベント報告、トレースデータ報告、リミットモニタリング、指定状態や変数データの問い合わせにより行われる。

4.2.1 イベントデータ収集 — イベントデータ収集は、動的で柔軟性に富んだ手段をユーザに提供する。データのホストに対する表現や報告を、それぞれの必要性に応じて装置上で調整出来る。イベントをベースとしたデータ収集の方式は、装置の動作を自動的にホストに知らせることになるため、ホストが装置をモニタしたり、装置との同期を維持したりするのに役に立つ。

イベントデータ収集は 2 つの論理的な部分に分けられる: イベントが発生したときにホストへ送られる通知と、イベントの通知に付随したデータの動的な設定変更である。

4.2.1.1 イベントの通知 — この項は装置の収集イベントが発生したときにホストへ通知する手段について述べる。

4.2.1.1.1 目的 — この性能によってホストは、装置のオペレーションの特定の時点でデータを受け取ることが出来る。この非同期の報告によってホストは情報を求めて装置をポーリングする必要がなくなる。装置で発生するイベントはホストの側の動作実行の引金となるかもしれない。また、装置の状態モデルに応じてイベントが発生することを知っていると、ホストは装置の状態を追跡することができる。装置の動作は装置の現在の状態に関連している。このようにして、この性能は、装置がどのようにふるまい、ホストの動作に対してどのように応答するのかをホストが理解するのを助ける。

4.2.1.1.2 定義

4.2.1.1.2.1 収集イベント (Collection Event) — ホストにとって重要であると思われる、装置での事象 (あるいは関連した事象の集まり)

4.2.1.1.2.2 収集イベント ID (CEID-Collection Event ID) — 収集イベントの固有の識別子。このデータ項目の完全な定義については SEMI E5 スタンダードを参照のこと。

15 S1,F13/F14 トランザクションが COMMACK=0 で正常に終了したとき通信は確立される。

16 オープンな S1,F13 があっても、S1,F14 が正常に伝達されれば通信は確立される。

4.2.1.1.2.3 イベント(Event) – 装置にとって重要な、検出可能な出来事。

4.2.1.1.2.4 レポート(Report) – 装置で前もって定義しておいた変数の集まり、あるいはホストが S2,F33/F34 を使って定義した変数の集まり。

4.2.1.1.3 詳細な説明 – 装置サプライヤは定義済み収集イベントのリストを提供しなければならない。個々の性能や状態モデルによっては特定の収集イベントが必要となる。収集イベントの例を下記に示した：

- ホストの指示要求によって始められたそれぞれの動作の完了
- 選択されたプロセス動作や材料移載動作
- 装置が検出したオペレータの動作
- 状態遷移
- 装置でのアラーム状態の設定もしくは解除
- アラームとは見なされない例外的な状態

必要な収集イベントのリストは § 6 に載せた。

不必要なメッセージを除外するためにユーザはイベント毎に収集イベントレポートを無効にすることができる。イベントレポートメッセージは、収集イベント ID (CEID) が有効になっていれば、特定の収集イベントが発生するとすぐに送信される。各々のイベントメッセージには、ひとつのレポート、または複数のレポートが付随しており、それぞれのレポートは変数データを含んでいる。§ 4.2.1.2 ではイベントレポートを動的に作り上げるための性能を説明している。イベントレポートメッセージに含まれているデータの値はそのイベントが発生したときのものでなければならない。つまり、イベントレポートはそのイベントが発生したその時のデータにより作られる。

装置は、ホストが特定のイベントレポートからのデータを要求できるように S6,F15/F16 トランザクションを提供する。

4.2.1.1.4 要求条件

1. 装置サプライヤは、装置について定義された収集イベント全部とそれぞれのイベントが発生する条件を記した書類を提供する。
2. 装置サプライヤは、報告の際に使用できるさまざまな収集イベントの一つ一つについて固有の CEID を提供する。
3. 装置サプライヤは、各イベントのレポートを有効にしたり無効にしたりするための手段を提供する。この手段はホストインタフェース (§ 4.2.1.2 参照)、もしくは装置のオペレータコンソールのどちらかを通じて行うことができる。
4. 各イベントについて装置サプライヤは下記のうちのどちらかを提供する：
 - イベントにリンクしていて、そのイベントに関連したデータを持つレポートの初期設定、もしくはユーザが装置のオペレータコンソールまたはホストインタフェース (§ 4.2.1.2 参照) を通じてそのイベントにリンクしたデータを設定できる能力。

4.2.1.1.5 シナリオ

装置での収集イベント発生

コメント	ホスト	装置	コメント
マルチブロック許可	S6,F6-->		[IF] イベントレポートがマルチブロックなら
		<--S6,F5	[THEN] マルチブロックの問い合わせを送信する
			[ENDIF]
イベントレポート確認	S6,F12-->	<--S6,F11	イベントレポート送信

ホストのイベントレポート要求

コメント	ホスト	装置	コメント
イベントレポート要求	S6,F15-->	<--S6,F16	イベントレポート送信

4.2.1.2 動的なイベントの設定変更 — この項ではホストが装置のイベントレポートの設定を動的に修正変更できる性能について説明する。

4.2.1.2.1 目的 — この性能は製造環境に応じて必要となる柔軟なデータ報告機能を持たせるために定義された。この性能によってホストは必要に応じてデータの流れを増減することができる。たとえば、装置の性能が低下した場合には、その問題の診断を助けるためにその装置からのデータの流れを増やすこともできる。

4.2.1.2.2 定義

4.2.1.2.2.1 Events Enabled — 現在有効になっている収集イベント ID (CEID) のリストで構成される変数データ項目。この変数データ項目の完全な定義については SEMI E5 スタンダードを参照のこと。

4.2.1.2.2.2 レポート ID (RPTID) — レポートに固有な識別子。このデータ項目の完全な定義については SEMI E5 スタンダードを参照のこと。

4.2.1.2.2.3 変数データ (V) — 状態変数 (SV), データ値 (DVVAL), あるいは装置定数の値 (ECV) を含んでいるデータ項目。このデータ項目の完全な定義については SEMI E5 スタンダードを参照のこと。

4.2.1.2.2.4 変数 ID (VID) — 変数データ項目に固有な識別子。変数 ID (VID) のセットは状態変数 ID (SVID), 装置定数 ID (ECID), データ値 (DVVAL) のための ID (DVNAME) 全部が結合したものである。このデータ項目の完全な定義については SEMI E5 スタンダードを参照のこと。

4.2.1.2.3 詳細な説明 — 装置は SECS-II インタフェースを通じて下記のイベントレポートの設定変更機能をサポートする:

- ホストによるレポートの定義／削除
- ホストによる定義済みレポートの指定収集イベントへのリンク／リンク解除
- ホストによる指定収集イベントレポートの有効／無効設定

注 3: 装置が (オペレータコンソールから) レポートを定義したり、レポートをイベントにリンクしたりする別の手段を与えることもある。ただし、これは要求条件ではない。

装置は収集イベントについて、個別にあるいは一括して、その収集イベントに対するレポートの有効、無効を指示するホストからの要求を受け付けることができる。このために有効な収集イベントのリストで構成される状態変数 (SV) が必要となる (§ 5.2, 変数項目リスト, EventsEnabled 変数参照)。

イベントレポートメッセージにレポートを付けてもよい (S6,F11)。そのレポートは希望するイベントに特にリンクされていて、通常そのイベントに関連した変数データを含んでいる。レポートは装置サブライヤが提供してもよい。ユーザが作ってもよい。ユーザはレポートを作りそれを SECS-II インタフェースを通じてイベントにリンクすることができなければならない。

イベントレポートメッセージで報告されるデータを構成するのは、状態変数 (SV), 装置定数値 (ECV), もしくはデータ値 (DVVAL) である。データの値が有効なのはあるイベントに関してのみであり、その値はそのイベントにリンクされたレポートの中だけ含まれることに注意する。

4.2.1.2.4 要求条件

- 装置ベンダーは、装置から得られる変数データすべてを文書にしたものを提供する。これは、変数名、変数タイプあるいはクラス (SV, ECV, DVVAL), 単位、フォーマットコード、値のとり得る範囲、その変数の意味と使用法の説明を含む。

- 装置ベンダーは、装置でデータ収集のために使うさまざまな変数データ(V)について固有の変数 ID(VID)を提供する。たとえば、装置定数値(ECV)やデータ値(DVVAL)の変数 ID(VID)と同じ変数 ID(VID)を持つ状態変数(SV)はない、ということである。
- レポート定義とイベントデータ収集にはすべての変数データを使うことができる。必要な変数データのリストについては § 5.2, 変数項目リストを参照。
- レポート定義, レポートからイベントへのリンク, イベントレポートの有効/無効状態はすべて不揮発性記憶装置に保持する。

4.2.1.2.5 シナリオ

収集イベントレポートの設定

コメント	ホスト	装置	コメント
[IF] レポート定義がマルチブロックである			
[THEN] マルチブロック問い合わせ送信	S2,F39-->	<--S2,F40 マルチブロック許可	
[ENDIF]			
規定レポートの送信	S2,F33-->	DATAIDs, RPTIDs, VIDsを受信	
		<--S2,F34 DRACK ¹⁷ =0 レポートは OK	
[IF] リンクイベント/レポートがマルチブロックである			
[THEN] マルチブロック問い合わせ送信	S2,F39-->	<--S2,F40 マルチブロック許可	
[ENDIF]			
レポートをイベントにリンク	S2,F35-->	CEIDs およびそれに対応する RPTIDsを受信	
		<--S2,F36 LRACK=0 イベントのリンクを確認	
収集イベントを有効にする	S2,F37-->	有効/無効コード(CEEDs)とそれぞれのイベントレポートの CEIDsを受信	
		<--S2,F38 ERACK=0 OK, 該当する収集イベントが発生したら指定されたレポートを作る	

4.2.2 変数データ収集

4.2.2.1 目的 — ホストはこの性能を使って装置のデータ変数について問い合わせることができるので、システムの立ち上げおよび同期化の際に有用である。

4.2.2.2 定義

4.2.2.2.1 レポート ID (RPTID) — レポートにつける固有の識別子。このデータ項目の完全な定義については SEMI E5 スタンダードを参照のこと。

4.2.2.2.2 変数データ(V) — 変数データ項目には状態、離散型データ、もしくは定数データが含まれている。このデータ項目の完全な定義については SEMI E5 スタンダードを参照のこと。

4.2.2.3 詳細な説明 — ホストはレポート ID(RPTID)を指定することによってデータ変数を含むレポートを装置に要求することができる。レポートはそれより前に定義済みであることを前提とする(例えば、レポート定義 S2,F33 トランザクションを使う; § 4.2.1 参照)。レポートに含まれる状態変数データ(SV)や装置定数(ECV)の値は現在のものでなければならない。離散型のデータ値(DVVAL)は特定の収集イベントが発生したその時だけ有効である。ハードウェアおよび/あるいはソフトウェアの状況による制約のため、装置に DVVAL を規定できない場合はゼロレンジスアイテムが報告される。

17 レポート定義承認コード, このデータ項目の完全な定義については SEMI E5 スタンダードを参照のこと。

4.2.2.4 要求条件

- 変数データ項目 (V) とそれに関連した計測の単位は装置ベンダーが提供しなければならない。

4.2.2.5 シナリオ

ホストのレポート要求

コメント	ホスト	装置	コメント
レポート RPTID に含まれるデータ変数を要求	S6,F19-->	<--S6,F20	与えられた RPTID についての変数データのリストを返信

4.2.3 トレースデータ収集

4.2.3.1 目的 — トレースデータ収集は定期的にデータをサンプリングする手段を提供する。時間ベースのデータ収集が役に立つのは、ある時間枠の中での傾向や繰り返し処理を追跡したり、連続データをモニタしたりするときである。

4.2.3.2 定義

4.2.3.2.1 データ収集時間 (DSPER) — サンプリング間隔。このデータ項目の完全な定義については SEMI E5 スタンダードを参照のこと。

4.2.3.2.2 応答グループ数 (REPGSZ) — ホストに伝達される一つのトレース報告に含まれるサンプル数。このデータ項目の完全な定義については SEMI E5 スタンダードを参照のこと。

4.2.3.2.3 状態変数データ (SV) — (トレース報告に含まれる) 状態変数データ項目。このデータ項目の完全な定義については SEMI E5 スタンダードを参照のこと。

4.2.3.2.4 状態変数 ID (SVID) — 状態変数に固有の識別子。このデータ項目の完全な定義については SEMI E5 スタンダードを参照のこと。

4.2.3.2.5 サンプル数 (TOTSMP) — トレース期間全体で収集されるサンプルの数。このデータ項目の完全な定義については SEMI E5 スタンダードを参照のこと。

4.2.3.2.6 トレース要求 ID (TRID) — トレース要求定義に関した識別子。このデータ項目の完全な定義については SEMI E5 スタンダードを参照のこと。

4.2.3.3 詳細な説明 — 装置はホストの指示に従い (S2,F23) トレース報告を作成する。トレース報告 (S6,F1) について、ホストはそのトレース報告の名称 (TRID)、データサンプリング間隔 (DSPER)、サンプル数 (TOTSMP)、トレース報告ごとのサンプル数: 応答グループ数 (REPGSZ)、どのデータをその報告につけて送信するかという状態変数 ID (SVID) のリストを明確に指示する。ホストに送られるトレース報告の数は、サンプルの総数を応答グループ数で割ったもので決まる (TOTSMP/REPGSZ)。

装置はホストが決めた間隔で指定されたデータ (SV) のサンプリングを行い、あらかじめ定義されたトレース報告を指定された応答グループ数 (REPGSZ) でホストに送信する。トレース報告の定義は、最終トレース報告が送信された後、装置側で自動的に削除する。

ホストはトレース要求定義上の TRID と同じ ID を指定することによって現在実行中のトレース機能を修正したり、再度、実行しなおしたりすることができる。ホストが TRID を指定した時点で古いトレースは終了し、新しいトレースが始められる。あるいは、特定の TRID について TOTSMP=0 を指定することによって、トレースが完了する前にそのトレースを終了させるよう装置に指示することができる。ホストがそのように指示した時点でトレース報告の設定は削除される。

詳細例をアプリケーションノート A4 に掲載した。

4.2.3.4 要求条件

- 装置は定期的なサンプリングを実行し、トレース報告をホストへ伝達するための機能 (例えば内部クロック) を持つ。
- 装置は最低 4 つのトレースを同時にサポートする。同じ状態変数 ID (SVID) を同時に複数のトレースで収集することができる。
- トレースデータ収集のために使うことが出来る。ただしシングルブロックに納まらない SV は例外とする。

注 4: SEMI E5 ではリストフォーマットの SV を提供している。これは実際には可変リストであるので、そのような SV をトレースデータ収集で使うことには潜在的な問題がある。これは SEMI E5 スタンドアードの問題である。リストフォーマットを使って SV を使う際には注意が必要である。

4.2.3.5 シナリオ

ホストのトレース報告開始

コメント	ホスト	装置	コメント
トレースデータ初期化要求	S2,F23-->	<--S2,F24	確認, トレース開始 [DO] TOTSMP/REPGSZ 回数繰り返す。 [DO] DSPER の間隔で SVID ₁ ,...,SVID _n のデータを REPGSZ 回収集する。 [END_DO]
受信確認	S6,F2-->	<--S6,F1	SV ₁ ,...,SV _n を送信 [END_DO]
オプション: 完了前にトレース終了を要求 (TOTSMP=0)	S2,F23-->	<--S2,F24	トレース中断を確認

4.2.4 リミット監視 — この性能は指定された装置変数の監視に関係しており, 3 つの基本的な側面を持つ。

- 監視を行う領域とリミットの標準的な集合を定義する。
- 指定された装置変数が監視領域間を遷移したときにホストに報告する。
- 指定された装置変数について, その変数リミットの属性値を修正する能力をホストに与える。

4.2.4.1 目的 — このリミット監視性能は, 装置間で一貫性があり, 柔軟で効果的, 非同期的な方法によって装置の状態を監視する手段をホストに提供する。これによってホストは現在の状態値をたえず装置に問い合わせる必要がなくなる。さらに, この性能によってホストは必要に応じて監視の範囲を変更することができるようになる。この性能は製造工程と診断/テスト工程用シナリオの両方に適用でき, 統計的なプロセスコントロールにも応用できる。

4.2.4.2 定義

4.2.4.2.1 リミット変数 (Limit Variable) — 領域間遷移収集イベントを生成するような装置変数の変数 ID (VID) を含むデータ値 (DVVAL)。

4.2.4.2.2 イベントリミット (Event Limit) — リミット変数 (Limit Variable) により超過されたリミットのリミット ID (LIMITID) を含むデータ値 (DVVAL)。

4.2.4.2.3 遷移タイプ (Transition Type) — 発生した領域間遷移の方向を定義するデータ値 (DVVAL): 0 = 下位領域から上位領域への遷移, 1 = 上位領域から下位領域への遷移。

4.2.4.2.4 リミット (Limit) — この項では変更可能な監視用“障壁”を完全に表す変数リミット属性の集合を代表するのに使う。その属性には, 変数 ID (VID), 単位, デッドバンド上限値 (UPPERDB), デッドバンド下限値 (

LOWERDB), リミット最大値 (LIMITMAX), リミット最小値 (LIMITMIN) を含む。文脈によってはもっと狭義に、デッドバンド上限値 (UPPERDB) とデッドバンド下限値 (LOWERDB) の組合せと解釈することもある。

4.2.4.2.5 リミット ID (LIMITID_n) — 監視する装置変数のリミットの集合のうち、(UPPERDB と LOWERDB によって定義した) 特定のリミットの識別子のことを指す。リミット ID は 1 から始まって (7 以下) のとり得るリミットの数まで、連続して番号が打たれる。

4.2.4.2.6 監視領域 (Monitoring Zone) — ホストにとって関心のある変数がとり得る値の範囲のサブセット。一つのリミットは範囲を 2 つの領域に分ける。複数のリミットは組み合わせられて範囲を更に細かく分ける。

4.2.4.2.7 領域間遷移 (Zone Transition) — 変数値がある監視領域から別の監視領域へ動くこと。この遷移は収集イベントであり対応する収集イベント ID (CEID) を持っている。

4.2.4.2.8 デッドバンド (Dead Band) — 2 つの領域が重なっている部分で、リミット上あるいはその近くにいる変数が始終、領域間遷移をする (つまり、チャタリングする) のを防ぐために設けられている。

4.2.4.2.9 デッドバンド上限値 (UPPERDB) — リミットのデッドバンドの上限を決める変数リミット属性¹⁸。この値はある変数 ID (VID) にひとつのリミット ID (LIMITID) を割り当てる。デッドバンド上限値 (UPPERDB) とデッドバンド下限値 (LOWERDB) は組になって一つのリミットを定義する。

4.2.4.2.10 デッドバンド下限値 (LOWERDB) — リミットのデッドバンドの下限を決める変数リミット属性¹⁸。この値はある変数 ID (VID) にひとつのリミット ID (LIMITID) を割り当てる。デッドバンド上限値 (UPPERDB) とデッドバンド下限値 (LOWERDB) は組になって一つのリミットを定義する。

4.2.4.2.11 上位領域 (UPPERZONE) — リミットより上にある値の範囲。

4.2.4.2.12 下位領域 (LOWERZONE) — リミットより下にある値の範囲。

4.2.4.2.13 リミット最大値 (LIMITMAX) — ある装置変数のリミットの最高値。この値は装置ベンダーがセットし、普通、監視される変数がとることのできる最高値と一致する。

4.2.4.2.14 リミット最小値 (LIMITMIN) — ある装置変数のリミットの最小値¹⁹。この値は装置ベンダーがセットし、普通、監視される変数がとることのできる最小値と一致する。

4.2.4.2.15 未定義 — 変数リミットに関して使われるときには、特定のリミットに関する領域間遷移の監視/報告が無効になっていることを示す。

4.2.4.3 説明 — リミット監視性能によってホストは、浮動小数点、整数、ブール数といったタイプの指定装置状態変数 (SV) に適用できる、最低 7 つの設定変更可能なリミットや障壁を持つことができる。こういった障壁のうちの一つの境界を越えると、収集イベントが発生して、監視領域や監視される変数の状態に起こる変化に注意するようホストに警告する。この 7 つのリミットはホストシステムの要求に適合できるようさまざまな組み合わせられる¹⁹。変数監視を実行するためにリミットのうちの 5 つを組み合わせさせた例を図 7 に掲載した²⁰。この項ではリミット監視のキーとなる点を説明している。リミット監視を実現した詳細な例をアプリケーションノート A7 に載せた。

注 5: SEMI E5 スタンダードではリスト構造の状態変数 (SV) も許されているが、この性能のもとではそういった変数リストは許されていない。

4.2.4.3.1 監視リミットの特性 — リミットは、そのリミットが対応している変数 ID (VID), その変数の単位、そのリミットがとり得る最大値および最小値 (LIMITMAX と LIMITMIN), リミットに定められた境界 (UPPERDB と LOWERDB) を含む属性の集まりによって定義されている。図 8 参照。デッドバンド上限値 (UPPERDB) とデッドバンド下限値 (LOWERDB) の値には制限があって、それは下記のような式に表される:

リミット最大値 (LIMITMAX) ≥ デッドバンド上限値 (UPPERDB) ≥ デッドバンド下限値 (LOWERDB) ≥ リミット最小値 (LIMITMIN)

18 フォーマットと単位は監視する変数のフォーマットと同じでなければならない。

19 装置は変数一つについて最低 7 つのリミットを提供するが、ホストは 7 つ全部を使う必要はないことに注意する。

20 この図は、ホストに報告されるリミット超過の動きを示したのではなく、装置が知ることができると考えられる読み取り値を示したものである。報告についてはこの項の後の方で説明する。

リミットは変数がとり得る値の範囲を上位領域と下位領域の 2 つの部分に分ける。いかなる時でも監視される変数はこの 2 つの部分のどちらかに存在する。しかし、図 8 で示したように、この 2 つの領域には重なっている部分があり、これをデッドバンドと呼ぶ。

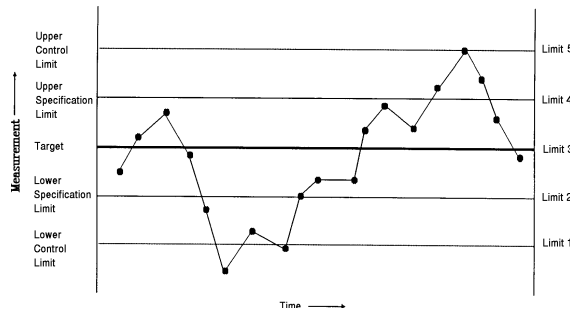


図 7

リミットの組合せ：コントロールの事例

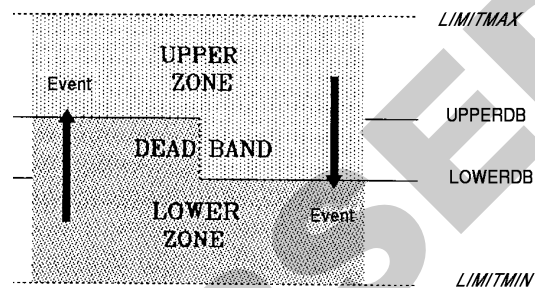


図 8

リミットの構成要素

デッドバンドはリミット監視、特に浮動小数点の変数に関する重要な概念である。その目的はチャタリングと呼ばれる現象 — 変数値が領域の境界近くにあるとき、値が小刻みに変動し領域間を行き来すること — を防ぐことにある。実際には、変数の値は領域遷移が起こる前にデッドバンドの反対側の境界に至る必要がある。このように、変数値がデッドバンドの上限 (UPPERDB) に至り、上位領域に遷移したら、その変数値はデッドバンドの下限 (LOWERDB) に落ちるまで下位領域には戻らない。デッドバンド上限値 (UPPERDB) とデッドバンド下限値 (LOWERDB) の差は、検出の必要が無いと考えられる変動の通常の振幅より常に大きくなければならない。デッドバンドの幅がゼロにセットされる場合もある (つまり、UPPERDB=LOWERDB)。整数値がリミット上にあるときには一見、現在の領域が曖昧になるように見える。しかし、その値の動きを考えるとそのようなことはない。デッドバンド上限値 (UPPERDB) = デッドバンド下限値 (LOWERDB) = 100 として考えてみよう。下記のリストには変数を連続して読み取った値と、それにより決定される領域を示した:

- 99 下位領域 (最初の読み取り値)
- 101 上位領域 (領域間遷移)
- 100 下位領域 (領域間遷移)
- 100 下位領域
- 99 下位領域
- 100 上位領域 (領域間遷移)

ある領域から別の領域へ遷移が起こると収集イベントが発生し、S6,F11 で報告される。ホストはそのイベントについてのイベントレポートを有効としたときに通知を受られるオプションを持っている。監視性能を持つ変数各々について

て、その変数に関する領域間遷移を示すための収集イベント ID (CEID) をひとつ決める。遷移イベントの性質を決定づけるためにデータ値 (DVVAL) を 3 つ、定義する。

- **リミット変数 (Limit Variable)** – 監視される変数の変数 ID (VID) で、収集イベントが参照する。
- **イベントリミット (Event Limit)** – リミット変数 (Limit Variable) が到達したり越えたりするリミットのリミット ID (LIMITID) を含んでいる。
- **遷移タイプ (Transition Type)** – 発生した領域間遷移の方向を決める: 0 = 下位領域から上位領域への遷移, 1 = 上位領域から下位領域への遷移。

サンプリング周期はリミット監視の重要な要素の一つで、装置仕様を決めるときに考慮されなければならない。変数値の変化がサンプリング周期に比べて相対的に速い場合には、いくつかの領域遷移が見落とされたり、値を読み取る合間にいくつもの領域間遷移が起こったりする可能性がある。領域間遷移が“同時に”起こったり、リミットに全く同じ値を定義したりする可能性があるため、イベントリミット (Event Limit) のデータ値 (DVVAL) は、変数についての複数の領域間遷移のリストを一つの収集イベントで報告することができるように定義している。

リミットを越える両方向への遷移を報告するのに、ただ一つの収集イベント ID (CEID) を使用するというのも強調しておかなければならない。つまり、一方向への遷移報告のみ行い、逆方向への遷移報告はしないという設定はできない。

個々の変数についての個々のリミットの機能性は図 9 に示した状態モデルで説明することができる。以下に、3 つの状態をさらに詳しく説明し、その後に遷移を定義した表を載せた。

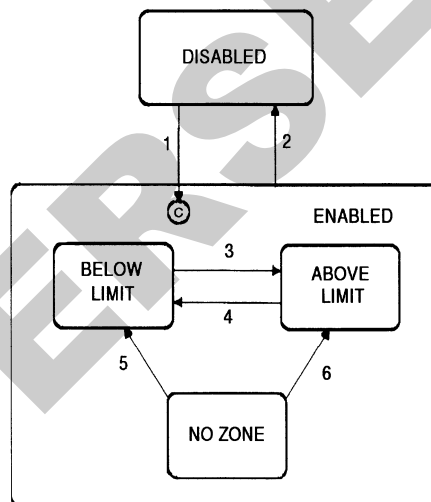


図 9
リミット状態モデル

リミット以上 (ABOVE LIMIT)

図 4.2.3 変数の値が増加して、デッドバンドの上側の境界であるデッドバンド上限値 (UPPERDB) と等しいか、それを越えたとき、その変数はリミット以上であると見なされる。ホストがどのように利用するかによってこの状態の重要性が決まる。

リミット以下 (BELOW LIMIT)

変数の値が減少して、デッドバンドの下側の境界であるデッドバンド下限値 (LOWERDB) と等しいか、それより少なくなったとき、その変数はリミット以下であると見なされる。ホストがどのように利用するかによってこの状態の重要性が決まる。

領域不定 (NO ZONE)