



SEMI E37-0413

HSMS 의 일반적 서비스(HIGH-SPEED SECS MESSAGE SERVICES (HSMS) GENERIC SERVICES)

이 표준은 기술적으로 Global Information & Control Technical Committee 에 의해 승인 되었으며, 해당 버전의 출판은 2012 년 12 월 20 일 Global Audits and Reviews Subcommittee 에 의해 승인되었다. 1995 년에 초본이 출판되었고 이전 버전은 2008 년 7 월에 출판되었다. 이번 출판본은 2013 년 4 월부터 www.semi.org 에서 이용 가능하다.

NOTICE: This translation is a REFERENCE COPY ONLY. If differences should exist between the English version and a translation in any other language, the English version is the official and authoritative version.

본 번역서는 참고용으로써, 영문원본과 한글번역본간에 차이가 발생할 경우, 영문원본을 우선으로 해야함을 알려드립니다. 본 번역서에 대해 문의사항이 있는 경우 krstandards@semi.org 로 문의 바랍니다.

NOTICE: The designation of SEMI E37 was updated during the 0413 publishing cycle to reflect the reapproval of SEMI E37.1.

SEMI E37 는 SEMI E37.1 의 재승인을 반영하기 위해 0413 Publishing Cycle 에 업데이트 되었다.

1 목적 (Purpose)

- 1.1 HSMS 는 독립적인 제조업체들이 서로에 대한 특정한 지식 없이도 연계하여 상호 운영(interoperate) 하여 생산활동을 할 수 있는 방법을 제공한다.
- 1.2 HSMS 는 고속 통신이 요구되는 곳이나 단순한 포인트 투 포인트 토폴로지(Point-to-Point Topology)가 충분치 않을 때 어플리케이션에 대한 SEMI E4 (SECS-I)의 대안이며, SEMI E4 (SECS-I)는 이들과 다른 HSMS 의 속성들이 반드시 필요치 않을 경우 어플리케이션에서 여전히 사용될 수 있다.
- 1.3 또한 HSMS 는 OSI 보다 TCP/IP 가 선호되는 경우 어플리케이션에 대한 SEMI E13 (SECS Message Services)의 대안이다.
- 1.4 사용의 세부 사항이 명확히 밝혀져 있거나 특정 어플리케이션 도메인에서 사용이 제한되는 하위 표준들이 HSMS 의 보조적인 역할을 하도록 되어 있다.

2 범위 (Scope)

- 2.1 High-Speed SECS Message Service (HSMS)는 반도체 공장 내 컴퓨터 간의 메시지 교환에 적합한 통신 인터페이스를 정의한다.

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

이 표준은 사용에 관련된 안전 사항의 명시를 의미하지 않는다. 사용자는 이 표준을 사용하기에 앞서 안전과 건강에 관련된 규제 방안을 수립하고, 규제나 기타 제한사항들에 대한 적용 방안을 규정할 책임이 있다.



3 참조 표준 및 문헌 (Referenced Standards and Documents)

3.1 SEMI 표준 (*SEMI Standards and Safety Guidelines*)

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

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

3.2 IETF 문헌 (*IETF Documents*¹)

IETF RFC 791 — Internet Protocol

IETF RFC 792 — Internet Control Message Protocol

IETF RFC 793 — Transmission Control Protocol

IETF RFC 1120 — Requirements for Internet Hosts – Communication Layers

IETF RFC 1340 — Assigned Numbers (Note: 이 RFC 는 RFC820 을 대체한다.)

3.3 POSIX 문헌 (*POSIX Document*²)

IEEE POSIX P1003.12 — Protocol Independent Interfaces (PII)

4 용어 (Terminology)

4.1 정의 (*Definitions*)

4.1.1 *application program interface (AP)* — TCP/IP 사용의 경우, TCP/IP 의 기능을 준비하거나 적용하기 위해 어플리케이션 프로그램에서 사용되는 프로그래밍 규약의 집합이다.

4.1.2 통신 실패 (*communication failure*) — SELECTED 에서 NOT CONNECTED 로의 상태 전환으로 인한 통신 연결의 실패 (§ 9 참조).

4.1.3 확정된 서비스 (*HSMS*) (*confirmed service (HSMS)*) — 개시자가 응답 개체에게 메시지를 전송함으로써 요청 받게 되는 HSMS 서비스, 이는 응답 개체가 개시자에게 응답 메시지를 통해 이 서비스의 완료를 표시할 것을 요구한다.

4.1.4 연결 (*connection*) — 두 개체 사이의 메시지 교환을 목적으로 TCP/IP LAN 상에 구축된 논리적인 연결.

4.1.5 컨트롤 메시지 (*control message*) — 두 개체 간의 HSMS 세션 관리에 사용되는 HSMS 메시지.

4.1.6 데이터 메시지 (*data message*) — HSMS 세션 내에서 특정 어플리케이션(application-specific) 데이터의 통신을 위해 사용되는 HSMS 메시지. 데이터 메시지(Data Message)는 주 메시지(Primary Message)이거나 응답 메시지(Reply Message)일 수 있다.

4.1.7 개체 (*entity*) — TCP/IP 연결의 종단점(endpoint)에 관련된 어플리케이션 프로그램.

4.1.8 헤더 (*header*) — 모든 HSMS 메시지에 선행하는 10-byte 의 데이터 요소.

¹ The IETF documents can be obtained from The Network Information Center, Network Solutions, 14700 Park Meadow Drive, Suite 200, Chantilly, VA 22021 USA

² POSIX documents can be obtained from Institute of Electrical and Electronic Engineers (IEEE), 345 East 47th Street, New York, NY 10017 USA



4.1.9 개시자 (*HSMS*) (*initiator (HSMS)*) — HSMS 서비스를 요청하는 개체. 개시자는 적절한 HSMS 메시지를 전송하여 해당 서비스를 요청한다.

4.1.10 IP 주소, 인터넷 프로토콜 주소 (*IP Address, Internet Protocol Address*) — TCP/IP 네트워크에 연결된 특정 부가 장치에 부여된 고유하게 식별하는 논리적인 주소.

4.1.11 로컬 개체 (*local entity*) — 연결의 특정 종단 점(end point)과 관련있는 로컬 개체는 해당 종단점(endpoint)과 관련된 개체이다.

4.1.12 특정 로컬 개체 (*local entity-specific*) — 프로시저, 옵션, Issue 및 구현과 관련된 일반적인 Qualifier는 이 Standard의 대상이 되지 않고, 개별 장비 업체의 재량에 맡겨 진다

4.1.13 메시지 (*message*) — 한 방향으로 된 통신의 완성된 단위. HSMS 메시지(HSMS Message)는 메시지 길이(Message Length), 메시지 헤더(Message Header) 그리고 메시지 텍스트(Message Text)로 구성된다. HSMS 메시지(HSMS Message)는 데이터 메시지(Data Message)이거나 컨트롤 메시지(Control Message)일 수 있다.

4.1.14 메시지 길이 (*message length*) — 바이트로 메시지의 길이를 나타내는 4-byte의 무 부호 정수 영역.

4.1.15 열린 트랜잭션 (*open transaction*) — 진행중인 트랜잭션(transaction).

4.1.16 포터 (*port*) — 완성된 네트워크 주소가 IP 주소(IP Address)와 TCP/IP 포트(TCP/IP Port) 번호로 지정되는 TCP/IP 연결의 종단점(endpoint).

4.1.17 포트 번호 (*port number*) — (또는 TCP 포트 번호). TCP/IP 연결의 종단점(endpoint) 역할을 하는 TCP/IP 네트워크 부가 장치 내의 포트 주소.

4.1.18 주 메시지 (*primary message*) — 홀수로된 기능(Function)을 가지는 HSMS 데이터 메시지(HSMS Data Message)이며, 데이터 트랜잭션(transaction)의 첫 번째 메시지이다.

4.1.19 설치된 포트 (*published port*) — TCP/IP 연결 요청의 수신을 위해 사용되는 특정 개체(서버)와 관련된 TCP/IP IP 주소(TCP/IP IP Address)와 포트(Port) 번호. 개체의 설치된 포트는 연결을 시작하려는 원격 개체들을 통해서만 알 수 있다.

4.1.20 수신자 (*receiver*) — 메시지를 수신하는 HSMS 개체(HSMS Entity).

4.1.21 원격 개체 (*remote entity*) — 연결의 특정 종단점(endpoint)에 관련되며 원격 개체는 연결의 반대 종단점(endpoint)과 관련된 개체이다.

4.1.22 응답(*reply*) — 짝수로된 기능(function)을 가지는 HSMS 데이터 메시지(HSMS Data Message)이며, 주 HSMS 데이터 메시지(Primary HSMS Data Message)에 대한 적절한 응답이다.

4.1.23 응답 개체(*HSMS*) (*responding entity (HSMS)*) — HSMS 서비스의 제공자. 응답 개체는 서비스를 요청하는 개시자로부터 메시지를 수신한다. 확정된 서비스의 경우, 응답 개체는 서비스를 요청하는 개시자에게 적절한 HSMS 응답 메시지를 전송하여 요청된 서비스가 완료 되었음을 알린다. 확정되지 않은 서비스의 경우, 응답 개체는 응답 메시지를 전송하지 않는다.

4.1.24 세션 (*session*) — HSMS 메시지 교환의 목적으로 두 개체 사이의 정립된 관계.



4.1.25 세션 개체 (*session entity*) — HSMS 세션에 참여하는 개체.

4.1.26 세션 ID (*session ID*) — 특정 세션 개체간의 특정 세션을 식별하는 16-bit 의 무 부호 정수.

4.1.27 스트림 (TCP/IP) (*stream (TCP/IP)*) — 다른 한 쪽으로의 전송을 위해 TCP/IP 연결의 한 종단(end)에 나타나는 바이트의 시퀀스(sequence). TCP/IP 는 나타난 스트림(stream)과 전송된 바이트의 시퀀스(sequence)가 일치함을 보장한다. HSMS 는 인접 바이트- 메시지의 블록으로 스트림(stream)을 세분화한다.

4.1.28 *T3* — HSMS 프로토콜상의 타임아웃(timeout) 응답.

4.1.29 *T5* — 개체 별 TCP/IP 연결의 중단이나 연결 설정 시도의 실패와, 동일 개체 별 새로운 TCP/IP 연결을 시작하려는 시도 사이에 최소 시간을 설정함으로써 과도한 TCP/IP 연결을 방지하는데 사용되는 HSMS 프로토콜상의 분리 타임아웃(Separation Timeout) 연결.

4.1.30 *T6* — 통신 실패가 발생한 것으로 간주되기 전에, HSMS 컨트롤 트랜잭션(transaction)을 열린(open) 상태로 유지할 수 있는 최대의 시간을 정의하는 HSMS 프로토콜상의 컨트롤 타임아웃(control Timeout) 트랜잭션(transaction)은 개시자가 필요한 요청 메시지를 전송한 시간으로부터 응답 메시지를 수신 할 때까지 열린(open) 상태로 간주된다.

4.1.31 *T7* — 통신 실패가 발생한 것으로 간주되기 전에, HSMS 연결을 위해 TCP/IP 연결 형성과 해당 연결을 이용하는 사이에 발생할 수 있는 최대 시간을 정의하는 HSMS 프로토콜의 연결 유휴 타임아웃(connection Idle Timeout).

4.1.32 *T8* —통신 실패가 발생한 것으로 간주되기 전에, 전체 HSMS 메시지 중 어떤 두 개의 연속된 바이트의 수신 사이에서 발생할 수 있는 최대 시간을 정의하는 HSMS 프로토콜의 네트워크 문자간 타임아웃(Intercharacter Timeout).

4.1.33 *TCP/IP*, 전송 컨트롤 프로토콜/인터넷 프로토콜 (*TCP/IP, Transmission Control Protocol/Internet Protocol*) — 네트워크내의 컴퓨터간에 신뢰할 수 있는 연결 형 메시지 교환을 제공하는 통신의 메소드(method).

4.1.34 *TLI*, 운송 레벨 인터페이스 (*TLI, Transport Level Interface*) — 운송 레벨(Transport Level) 프로토콜 사용에 대한 운영 시스템의 독립된 정의와 운송 프로토콜을 제공하는 TCP/IP 의 명확한 구현을 통해 규정되는 하나의 특정 API.

4.1.35 트랜잭션 (*transaction*) — 요청시 주 메시지(Primary Message)와 이와 관련된 응답 메시지(Reply message). 또한 필요하다면 요청(.req) 유형의 HSMS 컨트롤 메시지(HSMS Control Message)와 그에 대한 응답 컨트롤 메시지(Control Message) (.rsp)도 포함된다.

4.1.36 확정되지 않은 서비스 (*HSMS*) (*unconfirmed service (HSMS)*) — 개시자가 응답개체에게 메시지를 전송하여 요청되는 HSMS 서비스, 이 경우 응답 개체로부터의 완료 통보를 필요로 하지 않는다.



5 HSMS 개요 및 상태 다이어그램 (HSMS Overview and State Diagram)

5.1 High-Speed SECS Message Services (HSMS)는 TCP/IP 환경을 사용한 반도체 공장내의 컴퓨터 간의 메시지 교환에 적합한 통신 인터페이스를 정의한다. HSMS는 인접 바이트의 스트림(stream)중 신뢰할 수 있는 양 방향 동시 전송을 제공하는 TCP/IP 스트림(stream) 지원을 사용한다. 이는 SECS-I 통신이나 다른 진보된 통신 환경을 대체하여 사용될 수 있다.

5.2 HSMS 통신에 대한 절차는 그것이 대체하는 좀더 익숙한 SECS-I 통신과 유사하다. HSMS 통신이나 그 밖의 통신을 위해 다음의 단계들이 진행된다:

1. 두 개체 사이의 통신 연결을 설정한다. SECS-I에서, 이는 호스트와 장비를 물리적으로 연결하는 RS232 와이어(wire)를 말한다. HSMS에서의 연결은 표준 TCP/IP 연결 절차에 따라 확보된 TCP/IP 연결이다. “entity”라는 추상적인 용어는 “host”나 “equipment” 대신 사용됨에 유의한다. 이는 HSMS가 SECS-I의 대체로 사용될 때, HSMS가 좀 더 일반적인 어플리케이션을 가지기 때문이다. SECS-I 대체 어플리케이션에서 “host”는 “entity”이고 “equipment”는 “entity”이다.
2. 두 개체 간의 데이터 메시지 교환에 사용되는 어플리케이션 프로토콜 규약을 제정한다. SECS-I에 있어, 이 단계는 반도체 장비가 물리적으로 와이어(wire)의 두 종단(ends)을 연결한다는 의미를 내포한다: 프로토콜은 SECS-II이다. HSMS의 경우, 통신 연결은 HSMS 외의 다른 프로토콜을 이용한 여러 다른 TCP/IP 연결 혹은 비 TCP/IP 프로토콜을 사용한 연결과 공유될 수 있는 물리적인 연결상에 역동적으로 설정된 TCP/IP 연결이다. HSMS는 (Select Procedure라 불리는) 메시지 교환을 추가하여 HSMS Communication을 위해 독점적으로 특정 TCP/IP Connection만이 사용될 것이라는 것을 두 개체에게 확인하는데 사용한다.
3. 데이터 교환(Exchange Data). 이는 통신 연결의 일반적으로 의도된 목적이며, SECS-I과 HSMS에서 반도체 장비나 프로세스의 컨트롤(control)을 위해 SECS-II의 암호화된 메시지를 교환하기 위한 절차이다. 데이터 교환은 일반적으로 한쪽이나 양쪽 개체가 보수(maintenance)와 같은 장비의 특정 목적을 위해 off-line이 될때까지 계속된다.
4. 공식적인 통신의 종료. SECS-I에서 공식적인 통신 종료에 대한 요구사항은 없다; 오프라인된 장비는 통신을 중단한다. HSMS에서 메시지의 교환(“bilateral” Deselect 절차 또는 “unilateral” 분리 절차)은 TCP/IP 연결이 더 이상 HSMS 통신에 필요하지 않다는 것을 두 당사자에게 확인시켜주는데 사용된다.
5. 통신 연결의 중단. SECS-I에서 이는 공장 네트워크 환경의 물리적 재구성이나 보수 기간에만 발생하는 것으로, 통신 케이블을 장비나 호스트에서 물리적으로 뽑는 행위로 중단될 수 있다. HSMS에서는 TCP/IP의 동적 연결 환경을 사용하기 때문에, TCP/IP 연결은 네트워크 매개체로부터의 어떠한 물리적인 접속 중단 없이 접속 중단 절차나 해제를 통해 논리적으로 중단된다.

5.3 HSMS 내에서 지원을 받는 진단(diagnostic)의 성격을 가지는 추가적인 절차가 두 가지 있다. 그런데 이들은 일반적으로 단순한 SECS-I 연결이나 SECS-I의 직접적인 대체가 반드시 필요하지 않다. 내용은 다음과 같다:

1. *Linktest* — 해당 절차를 통해 연결 완전성(integrity)을 간단히 확인할 수 있다.



2. *Reject* — HSMS 는 SECS-II (하위 표준에 의해) 외에 프로토콜의 확장을 의도하기 때문에, (시스템 구성 오류로 인해) 상호 호환되지 않는 하위 표준을 사용하는 두 개체가 연결 될 수도 있다. 또한 초기 실행 중에 올바르게 보내지는 메시지 유형이 전송 되거나 소프트웨어의 버그로 인해 잘못된 순서로 전송을 할 수 있다. 거부 절차를 통해 이러한 상황의 발생을 표시할 수 있다

5.4 HSMS 연결 상태 다이어그램 (*HSMS Connection State Diagram*) — HSMS 상태 기기는 아래의 다이어그램으로 설명된다. 이 다이어그램에서 설명된 동작은 HSMS 의 기본 요구 사항을 정의한다: 하위 표준은 이들이나 다른 상태들을 좀 더 확장 할 수 있다.

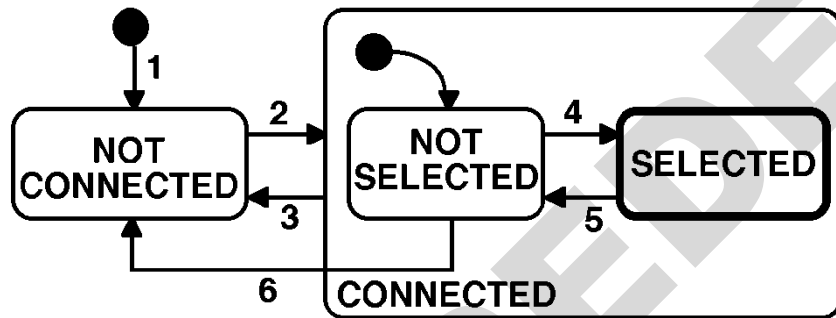


Figure 1
 HSMS 연결 상태 다이어그램
 (HSMS Connection State Diagram)

5.5 상태 설명 (*State Descriptions*)

5.5.1 *NOT CONNECTED* — 개체는 TCP/IP 연결을 시작하거나 수용할 준비가 되어 있지만, 어떤 연결도 설정하지 못했거나 이전에 설정된 모든 TCP/IP 연결이 이미 종료된 것이다.

5.5.2 *CONNECTED* — TCP/IP 연결이 설정되며, 이 상태는 NOT SELECTED 와 SELECTED, 두 가지 하위 상태를 가진다.

5.5.2.1 *NOT SELECTED* — HSMS 세션이 설정되지 않았거나, 이전에 설정된 HSMS 세션이 종료된 CONNECTED 의 하위상태.

5.5.2.2 *SELECTED* — 적어도 하나의 HSMS 세션이 설정된 CONNECTED 의 하위상태. 이는 정상적인 HSMS 의“operating” 상태이다: 이 상태에서 데이터 메시지가 교환 되기도 하며, 이는 상태 다이어그램에서 굵은 선으로 표시 되어 있다.

5.6 상태 전환 테이블 (*State Transition Table*)



Table 1 상태 전환 (State Transition)

#	Current State	Trigger	New State	Actions	Comment
1	...	TCP/IP 통신을 위한 로컬 특정 개체 준비	NOT CONNECTED	로컬 특정 개체	사용될 연결 절차에 따른 실행(action): active 혹은 passive
2	NOTCONNECTED	TCP/IP 연결이 HSMS 통신을 위해 설정됨	CONNECTED - NOT SELECTED	로컬 특정 개체	없음.
3	CONNECTED	TCP 연결의 중단	NOT CONNECTED	로컬 특정 개체	§ 6.4. 참조
4	NOT SELECTED	HSMS 선택 절차(HSMS Select Procedure)의 성공적인 완료.	SELECTED	로컬 특정 개체	HSMS 통신이 모두 설정됨: 데이터 메시지 교환이 허용됨.
5	SELECTED	HSMS 선택 해제 혹은 분리(HSMS Deselect or Separate)의 성공적인 완료	NOT SELECTED	로컬 특정 개체	이 전환은 일반적으로 HSMS 통신의 종료를 나타내며 개체는 즉각적으로 TCP/IP 연결을 종료를 수행하게됨 (상기의 Transition 3).
6	NOT SELECTED	T7 연결 타임아웃(T7 Connection Timeout)	NOT CORRECTED	로컬 특정 개체	Per § 9.2.2.

6 TCP/IP 의 사용 (Use of TCP/IP)

6.1 *TCP/IP API* — 실행에 사용되는 TCP Application Program Interface (API)의 요구되는 사양은 HSMS의 범위를 벗어난다. 주어진 HSMS의 실행은 네트워크 상의 상호운영 가능한 TCP/IP 스트림(stream) 프로토콜의 제공이 가능하다는 전제 하에 — 소켓(socket), TLI(Transport Layer Interface) 등의 — 계획된 하드웨어와 소프트웨어 플랫폼에 적절한 TCP/IP API를 사용한다.

6.1.1 부록(appendix)은 TLI(POSIX 표준 1003.12) 및 TCP/IP 통신을 위한 BSD 소켓(socket) 모델을 사용하여 표본 시나리오와 해당 표준에서 언급하고 있는 TCP/IP 절차의 예시를 포함한다.

6.2 TCP/IP 네트워크 주소화 규약 (*Network Addressing Conventions*)

6.2.1 *IP Addresses* — 주어진 Local Area Network(LAN)상에서 각각의 TCP/IP의 물리적 연결은 고유한 IP 주소(IP Address)를 가져야만 한다. IP 주소(IP Address)는 설치 시에 할당 될 수 있어야 하며, HSMS 실행 시 고정된 IP 주소(IP Address)를 선택할 수 없다. 전형적인 IP 주소(IP Address)는 192.9.200.1. 이다.

6.2.2 IP는 HSMS 프로토콜 범위 밖의 번호에 제한을 두고 있다. RFC 791의 § 2.3과 § 3의 Internet Protocol (IP) 참조.



6.2.3 TCP 포트 번호 (TCP Port Numbers) — TCP 포트 번호(TCP Port Number)는 IP 주소(IP Address)의 확장된 개념으로 간주될 수 있다.

6.2.4 HSMS 실행은 TCP/IP 실행이 사용되는 모든 범위에서 TCP 포트(TCP Port) 구성이 가능해야 한다. 전형적인 TCP 포트 번호(TCP Port Number)는 5000 이다.

6.2.5 규약(Conventions)은 HSMS 프로토콜 범위 밖의 TCP 포트 번호(TCP Port Number)를 선택하기 위해 설정된다. § 3 의 Transmission Control Protocol (TCP), RFC 793 참조.

6.3 TCP/IP 연결 설정 (Establishing a TCP/IP Connection)

6.3.1 연결 모드 (Connect Modes) — TCP/IP 연결을 설정하기 위한 절차는 RFC793 에 정의되어 있지만, RFC793 가 정의하는 모든 절차를 일반적으로 사용 가능한 API 가 지원하는 것은 아니다. 특히 RFC793 이 두 개체가 동시 접속을 시작하는 것이 허용될 때 , 이 기능은 사용가능 하도록 지원되지 않는다.

6.3.2 APIs. HSMS 는 다음 모드 중 하나로 개체를 제한한다:

- 수동 모드 (Passive Mode) — 원격 개체(Remote Entity)에 의해 시작된 연결절차를 로컬 개체가 준비하여 수용할때 사용되는 모드.
- 능동 모드 (Active Mode) — 연결절차가 로컬 개체(Local Entity)에 의해 시작되었을 때 사용되는 모드.

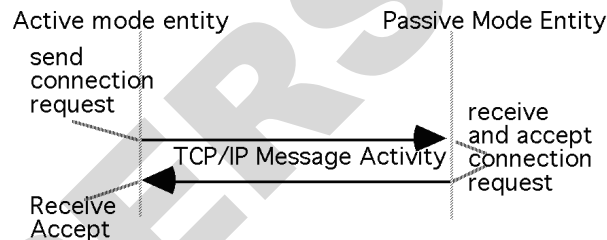


Figure 2
TCP/IP 연결 절차 개요
(TCP/IP Connection Procedure Summary)

6.3.3 부록(appendix)에서 개체가 통신 설정 단계에서 유연성을 발휘하기 위해 수동과 능동모드를 어떻게 교대로 운영되는지의 예를 보여준다.

6.3.4 수동 모드 연결 절차 (Passive Mode Connect Procedure) — 수동 로컬 개체(Passive Local Entity)가 따르는 절차가 RFC793 에 정의 되어 있으며 다음과 같이 요약된다:

1. 연결 종단점(endpoint)을 찾아서 설치된 (published)포트와 연결한다.
2. 원격 개체로부터 설치된 포트에 전송된 연결 요청을 수용한다.
3. 연결 요청을 수신하면, 이를 인식하여 수용하고 연결을 수락하였음을 나타낸다. 이 때, 연결 절차가 성공적으로 완료되며, CONNECTED 상태가 된다 (§ 5).



6.3.5 이들 절차들은 TCP/IP 중 로컬 개체 구현의 API 를 통해 실행된다. 부록(appendices)에서 TLI 와 BSD 를 이용한 상기 단계에 대한 특정 API 절차를 규정한다.

NOTE 1: 상기의 과정 중에 연결 실패가 발생할 수 있으며, 실패의 원인은 특정 로컬 개체 때문일 수도 있고, 특정 로컬 개체의 타임아웃(timeout) 후 연결 요청의 부재에 기인할 수도 있다. 취해지는 동작(예: 재 시도를 위해 1 단계로 돌아감)은 로컬 개체에 따라 달라질 수 있다

NOTE 2: 동일한 수동 모드 개체로의 다중 접속 요청과 관련된 문제들에 대해, § 9 특별 고려사항(Special Considerations) 참조.

6.3.6 능동 모드 연결 절차 (*Active Mode Connect Procedure*) — 능동 로컬 개체(Active Local Entity)가 따르는 절차는 RFC 793 에 정의되어 있으며, 이는 다음과 같이 요약된다:

1. 연결 종단 점(endpoint) 확보.
2. 수동 모드 원격 개체의 설치된 포트로의 연결 시작.
3. 원격 개체로부터 연결 요청에 대한 승인과 응답의 수신을 기다린다. 원격 개체에서의 승인의 수신은 연결 절차의 성공적인 완료를 나타내며 CONNECTED 상태로의 전환을 의미한다(§ 5).

6.3.7 이들 절차들은 TCP/IP 중 로컬 개체 구현의 API 를 통해서 실행된다. 부록(appendix)에서 TLI 와 BSD 를 이용하여 상기 단계에 대한 특정 API 절차를 규정한다.

NOTE 3: 상기 단계 중에 실패가 발생할 수 있으며, 실패에 대한 원인은 특정 로컬 개체 문제일 수도 있고 특정 로컬 개체의 타임아웃(timeout) 후 승인 메시지의 부재일 수도 있다. 취해지는 동작(예: 재 시도를 위해 1 단계로 돌아감)은 로컬 개체에 따라 달라질 수 있지만, 만일 로컬 개체가 연결을 재 시도한다면, 이는 T5 연결 분리 타임아웃(timeout)을 따른다(“Special Considerations” 참조).

6.4 TCP/IP 연결 종료 (*Terminating a TCP/IP Connection*) — 연결 종료는 연결(Connection) 설정과 논리적으로 정반대이다. 로컬 개체(Local Entity's)의 측면에서 TCP/IP 연결은 언제든지 중단될 수 있지만, HSMS 는 연결이 CONNECTED 상태의 하위상태인 NOT SELECTED 일 때만 연결을 종료할 수 있다.

6.5 연결 종료에 대한 절차는 RFC793 에 정의되어 있으며, 어떤쪽이든 한 개체가 연결을 종료할 수 있다. NOT CONNECTED 상태에 진입하면, 이는 HSMS 통신의 종료를 나타낸다. 부록(appendix)에서 TLI 와 BSD API 를 사용하여 해제와 연결종단에 대한 절차를 설명한다.

7 HSMS 메시지 교환 절차 (HSMS Message Exchange Procedures)

7.1 HSMS 는 이전 섹션에서 소개된 절차에 따라 설정된 TCP/IP 연결을 통해 개체들간의 모든 메시지 교환에 대한 절차를 정의한다. 개요 글에서 설명된 것처럼, 연결이 설정되면 두 개체는 선택(Select) 절차에 따라 HSMS 통신을 설정한다. 데이터 메시지는 언제든지 어떤 방향으로든 양방향으로 교환될 수 있고, 개체들이 HSMS 통신으로 종료하고자 한다면, HSMS 통신 종료를 위해 선택 해제(Deselect)와 분리(Separate) 절차를 사용한다.

7.2 HSMS 메시지의 전송 및 수신 (*Sending and Receiving HSMS Messages*) — 모든 HSMS 절차는 HSMS 메시지 교환과 관련이 있다. 이들 메시지는 표준의 우선 순위에 따라 이전에 설정된 TCP/IP 연결을 사용하여



TCP/IP 스트림(stream)대로 전송되고 수신된다. 특히 “Urgent” 데이터의 사용을 HSMS 는 지원하지 않는다 (전송 및 수신 절차에 관련된 상세정보는 RFC 793 참조).

7.2.1 부록(appendix)에서는 TLI 와 BSD 소켓(socket) APIs 를 이용한 HSMS 메시지 전송 및 수신 예 제공한다.

7.3 선택 절차 (*Select Procedure*) — 선택(Select) 절차는 컨트롤(control) 트랜잭션(transaction)에서 Select.req 와 Select.rsp 메시지들을 이용한 TCP/IP 연결상에 HSMS 통신을 설정하는데 사용된다.

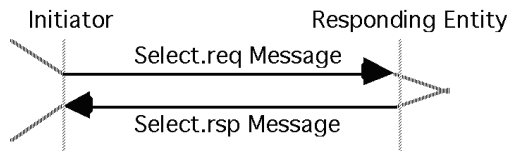


Figure 3
선택 컨트롤 트랜잭션
(Select Control Transaction)

7.4 HSMS 가 CONNECTED 상태에서 언제든지 선택(Select)을 허용하더라도, 하위 표준은 NOT SELECTED 하위 상태에서 연결이 유지되도록 추가적으로 요구할 수도 있다 (“Special Considerations”참조).

7.4.1 개시자 절차 (*Initiator Procedure*) — 개시자로 이어지는 절차는 다음과 같다.

1. 선택(select) 절차의 개시자가 응답 개체에 Select.req 메시지를 전송한다.
2. 개시자가 0 인 선택 상태(Select Status)의 Select.req 를 수신하면, HSMS 선택(HSMS Select) 절차는 성공적으로 완료되며, SELECTED 상태로 전환된다. (§ 5 참조).
3. 개시자가 0 이 아닌 선택 상태(Select Status)의 Select.req 를 수신한다면, 선택(Select)이 성공적으로 완료되지 못한다 (상태 전환 없음).
4. Select.req 를 수신하기 전에 개시자에서 T6 타임아웃(timeout)이 만료되면, 이는 통신 실패로 간주된다 (“Special Considerations” 참조).

7.4.2 응답 개체 절차 (*Responding Entity Procedure*) — 응답 개체로 이어지는 절차는 다음과 같다.

1. 응답 개체가 Select.req 를 수신한다.
2. 응답 개체가 선택(select)을 수락할 수 있다면, 응답 개체는 0 인 선택 상태(Select Status) 의 Select.req 를 전송한다. 응답 개체에 대한 HSMS 선택 절차(HSMS Select Procedure)가 성공적으로 완료되고, SELECTED 상태로 전환된다 (§ 5 참조).
3. 응답 개체가 선택을 허용할 수 없다면, 응답 개체는 0 이 아닌 선택 상태(Select Status)의 Select.req 를 전송한다. 응답 개체에 대한 HSMS 선택 절차(HSMS Select Procedure)가 성공적으로 완료되지 않는다 (상태 전환 없음).



7.4.3 동시 선택 절차 (*Simultaneous Select Procedures*) — 하위 표준이 선택(Select)의 사용을 제한하지 않는다면 양쪽 개체는 동일한 Session ID의 선택 절차(Select Procedures)를 동시에 시작하는 것이 가능하다. 그런 경우, 각각의 개체는 Select.rsp로 응답하여 다른 개체의 선택 요청을 수용한다.

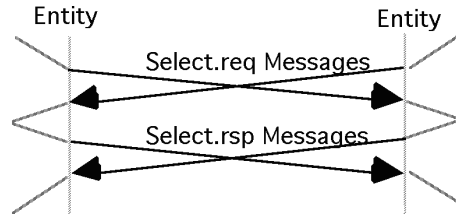


Figure 4
동시 선택 트랜잭션
(Simultaneous Select Transactions)

7.5 데이터 절차 (*Data Procedure*) — SELECTED 상태로 연결되어 있다면 HSMS 데이터 메시지는 어떤 개체에 의해서도 시작될 수 있으며, 연결이 SELECTED 상태가 아니라면 데이터 메시지 수신은 거부(reject) 절차로 이어진다 (§ 7.7 참조).

7.6 데이터 메시지는 “Primary” 혹은 “Reply” 데이터 메시지등의 데이터 트랜잭션(transaction) 중 일부로 추가 정의될 수 있다. 데이터 트랜잭션(transaction) 중, 트랜잭션(transaction)의 개시자는 주(primary) 메시지를 응답 개체에게 전송한다. 주(Primary) 메시지가 응답을 기대하고 있음을 나타내면, 응답 개체가 주(Primary) 메시지에 대한 응답으로 응답(Reply) 메시지를 전송한다.

7.6.1 데이터 트랜잭션(Data Transaction)은 다음의 유형을 지원한다:

1. 응답을 기대하는 주 메시지(Primary Message)와 관련된 응답 메시지(Reply Message)
2. 응답을 기대하지 않는 주 메시지(Primary Message)

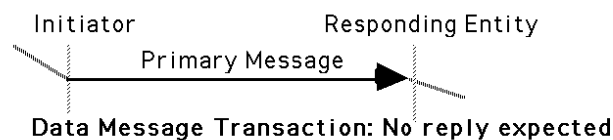
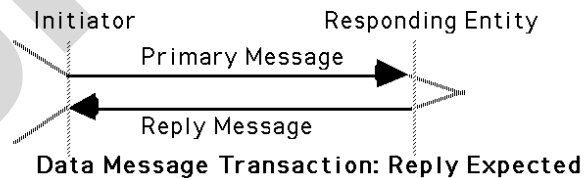


Figure 5
데이터 메시지 트랜잭션
(Data Message Transaction)



7.6.2 이들 트랜잭션(transaction)에 대한 특정한 절차는 어플리케이션 레이어(layer)에 의해 결정되며, 다른 표준(Standards)이 적용된다 (예, SECS-II 의 부호화된 메시지를 이용한 GEM 장비에 대한 SEMI E5 와 SEMI E30).

7.6.3 적용 가능한 상위 레이어(layer)의 표준은 메시지 유형으로 식별되며, 그 유형은 § 8 에서 정의된 특정 형식중 하나로 결정된다. HSMS 메시지에 대한 정상적인 유형은 SECS-II 텍스트이다. T3 응답 타임아웃(T3 Reply Timeout)에 관해서는 “Special Considerations”을 참조할 것.

7.7 선택해제 절차 (*Deselect Procedure*) — 선택해제(Deselect) 절차는 TCP/IP 연결을 중단하기에 앞서 개체에 대해 HSMS 통신의 원만한 종료를 위해 사용된다. HSMS 는 SELECTED 상태의 연결을 필요로 하며, 그 절차는 다음과 같다.

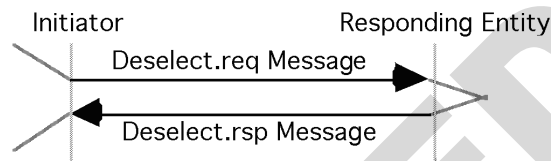


Figure 6
선택해제 컨트롤 트랜잭션
(Deselect Control Transaction)

7.7.1 개시자 절차 (*Initiator Procedure*)

1. 선택 해제(Deselect) 절차의 개시자가 응답 개체에 Deselect.req 메시지를 전송한다.
2. 개시자가 0 인 선택해제 상태(Deselect Status)의 Deselect.rsp 를 수신하면, 선택해제(Deselect) 절차는 성공적으로 종료되며, NOT SELECTED 상태로 전환된다. (§ 5 참조).
3. 개시자가 0 이 아닌 선택해제 상태(Deselect Status)의 Deselect.rep 를 수신하면, 선택해제(Deselect) 절차는 성공적으로 종료되지 못하고, 어떠한 상태 변화도 발생하지 않는다.
4. 개시자가 Deselect.rsp 를 수신하기 전에 T6 타임아웃(timeout)이 만료되면, 이는 통신 실패로 간주된다 (“Special Considerations” 참조).

7.7.2 응답 개체 절차 (*Responding Entity Procedure*)

1. 응답 개체가 Deselect.req 메시지를 수신한다.
2. 응답 개체가 SELECTED 상태이고, 선택해제(Deselect)를 허용한다면, 0 응답 코드의 Deselect.rsp 를 이용하여 응답한다. 응답 개체의 선택해제(Deselect) 절차가 성공적으로 완료되고, NOT SELECTED 상태로 전환된다 (§ 5 참조).
3. 응답 개체가 선택해제(Deselect)를 허용하지 않는다면, 이는 SELECTED 상태가 아니거나 로컬 조건이 Deselect 를 허용하지 못하기 때문에 0 이 아닌 응답코드의 Deselect.rsp 를 이용하여 응답한다. 응답 개체의 선택해제(Deselect) 절차가 성공적으로 종료되지 못하고, 어떠한 상태 변화도 발생하지 않는다.



7.7.3 동시 선택해제 절차 (*Simultaneous Deselect Procedures*) — 하위 표준이 선택해제(Deselect)의 사용을 제한하지 않는다면, 동일한 Session ID's 로 양쪽 개체에 대한 선택해제 절차(Deselect Procedures)의 동시 시작이 가능하다. 이 경우, 각각의 개체는 deselect.rsp 의 응답을 통해 다른 개체의 선택해제(Deselect) 요청을 승인한다.

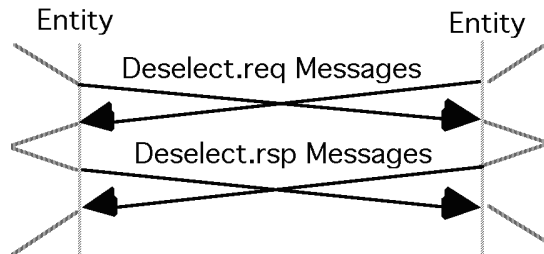


Figure 7
 동시 선택해제 트랜잭션
 (Simultaneous Deselect Transactions)

7.8 Linktest 절차 (*Linktest Procedure*) — Linktest 는 HSMS 통신과 TCP/IP 의 운용상의 완전성(integrity)을 결정하는데 사용되며, CONNECTED 상태에서 언제든지 유효하다.

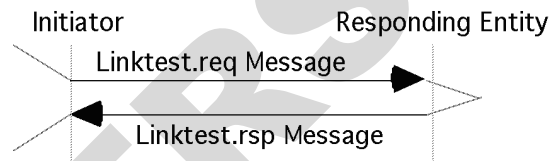


Figure 8
 Linktest 컨트롤 트랜잭션
 (Linktest Control Transaction)

7.8.1 개시자 절차 (*Initiator Procedure*)

1. Linktest 절차의 개시자가 응답 개체에 Linktest.req 메시지를 전송한다.
2. T6 타임아웃(timeout)내에 개시자가 Linktest.req 를 수신한 경우, Linktest 는 성공적으로 완료된다.
3. 개시자가 Linktest.rsp 를 수신하기 전에 T6 타임아웃(timeout)이 만료 된다면, 이는 통신 실패로 간주된다 (“Special Considerations” 참조).

7.8.2 응답 개체 절차 (*Responding Entity Procedure*)

1. 응답 개체가 개시자로부터 Linktest.req 를 수신한다.
2. 응답 개체가 Linktest.rsp 를 전송한다.

7.9 분리 절차 (*Separate Procedure*) — 분리(Separate) 절차는 하나의 개체가 TCP/IP 연결(TCP/IP Connection)을 중단하기에 앞서 갑자기 HSMS 통신을 종료할 때 사용된다. HSMS 가 분리(Separate)를 사용할



때 연결은 SELECTED 상태여야 한다. 응답 개체는 응답을 전송하지 않고 로컬 상태와 관계 없이 반드시 통신이 종료되어야 하며, 그 절차는 다음과 같다.

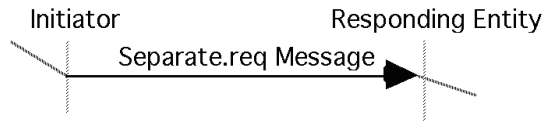


Figure 9
분리 컨트롤 트랜잭션
(Separate Control Transaction)

7.9.1 개시자 절차(Initiator Procedure)

1. 선택 절차의 개시자는 Separate.req 메시지를 응답 개체에게 전송한다. 개시자의 분리(Separate) 절차가 성공적으로 완료되고, NOT SELECTED 상태로 전환된다. (§ 5 참조)

7.9.2 응답 개체 절차 (Responding Entity Procedure)

1. 응답 개체가 개시자로부터 Separate.req 를 수신한다.
2. 응답 개체가 SELECTED 상태인 경우, 분리(Separate) 절차가 성공적으로 완료된다.
3. 응답 개체가 SELECTED 상태가 아닌 경우, Separate.req 는 무시된다.

7.10 거부 절차 (Reject Procedure) — 거부(Reject) 절차는 유효하지 않은 HSMS 메시지가 부적절한 상황에서 수신된 것에 대한 응답으로 사용된다. 거부(Reject) 절차는 HSMS 를 이용하여 배포된 어플리케이션의 개발 중 유용한 진단 정보를 제공 하는데 지원되며, 그 절차는 다음과 같다:

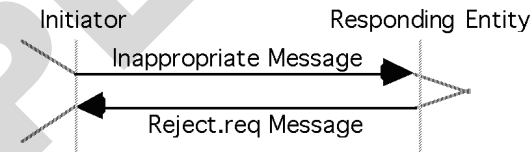


Figure 10
거부 트랜잭션
(Reject Transaction)

7.10.1 개시자(부적합한 메시지의 송신자) 절차 (Initiator (Sender of Inappropriate Message) Procedure)

1. 부적합한 메시지의 개시자는 Reject.req 를 수신 시 적절한 조치를 취한다 (특정 로컬개체).

7.10.2 응답 개체 절차 (Responding Entity Procedure)

1. 부적합한 메시지를 수신한 개체는 Reject.req 메시지로 응답한다.

7.10.3 HSMS 는 NOT SELECTED 상태에서 데이터 메시지 수신에 대한 거부(Reject) 절차를 따르거나, 메시지를 수신한 개체에 대해 정의되지 않은 S Type 이나 P Type (다음 섹션 참조: Message Format) 메시지의



수신이 요구된다. 하위 표준에서 거부 절차(Reject Procedure)가 요구되는 다른 환경들이 정의 되기도 한다. 일반적으로 거부(Reject) 메시지의 수신은 부적절하게 구성된 시스템이나 소프트웨어 프로그래밍 오류를 나타낸다.

8 HSMS 메시지 형식 (HSMS Message Format)

8.1 이 섹션에서는 이전 섹션에서 소개된 절차에서 사용되는 메시지의 세부적인 형식을 정의한다.

8.2 일반적인 메시지 형식 (General Message Format)

8.2.1 바이트 구조(Byte Structure) — HSMS 내에서, 1 바이트는 8 비트를 의미한다. 1 바이트 내의 비트는 Bit 7(최상위)부터 Bit 0(최하위)로 순서로 숫자가 매겨진다.

8.2.2 메시지 형식(Message Format) — HSMS 메시지(HSMS Message)는 다음의 순서로 바이트의 단일 인접 스트림(stream)으로 전송된다:

Table 2 HSMS 메시지 형식 (HSMS Message Format)

<i>Number of Bytes</i>	<i>Description</i>
4 Bytes	메시지 길이(Message Length). MSB 우선(MSB First). 메시지 텍스트(Message Text)와 더불어 메시지 헤더(Message Header)에서 바이트 수를 지정한다.
10 Bytes	메시지 헤더(Message Header).
0–n Bytes	메시지 텍스트(Message Text). 메시지 헤더의 P Type 영역에 의해 보다 구체화된 형식

8.2.3 메시지 길이 (Message Length) — 메시지 길이(Message Length)는 메시지 내용(Message Text)과 더불어 메시지 헤더(Message Header)의 길이를 바이트로 보여주는 4 바이트의 무 부호 정수 값이다. 메시지 길이(Message Length)는 맨 처음최상위 바이트(MSB)에서 마지막 최하위 바이트(LSB) 순으로 전송된다.

8.2.4 가능한 최소 메시지 길이(Message Length)는 10 이며(Header 단독), 가능한 최대 메시지 길이(Message Length)는 구현에 따라 달라진다.

8.2.5 메시지 헤더 (Message Header) — 메시지 헤더(Message Header)는 10-byte 필드이다. 헤더의 바이트들은 byte 0 (전송된 첫 번째 바이트)에서 byte 9 (전송된 최종 바이트) 순으로 숫자가 매겨지며, 메시지 헤더의 형식(Message Header)은 다음과 같다:

Table 3 HSMS 메시지 헤더 (HSMS Message Header)

<i>Bytes</i>	<i>Description</i>
0–1	Session ID (Device ID)
2	Header Byte 2
3	Header Byte 3



<i>Bytes</i>	<i>Description</i>
4	PType
5	SType
6–9	System Bytes

8.2.6 물리적인 바이트 순서는 SECS-I 헤더에 최대한 근접하게 대응되도록 설계되어 있다.

8.2.6.1 세션 ID (*Session ID*) — 세션 ID(Session ID)는 16-bit의 무 부호 정수 값이며, 이는 헤더의 바이트 0 과 1 (byte 0 은 MSB, 1 은 LSB)이 값을 채우고 있고, 컨트롤 메시지 (특히 Select 와 Deselect)와 뒤따르는 데이터 메시지 간에 서로 참조하여 결합하도록 하는데 그 목적이 있다. HSMS 하위 표준의 역할은 이들 연결을 구체화 하는데 있다.

8.2.6.2 헤더 바이트 2 (*Header Byte 2*) — 이 헤더 바이트는 다른 HSMS 메시지에 대해 다른 방법으로 사용된다. 컨트롤 메시지(Control Message, 아래의 S Type 참조)는 0 또는 상태 코드를 포함한다. P type(아래 참조) = 0 인 데이터 메시지(Data Message)는 W-Bit 와 SECS 스트림(SECS Stream)을 포함한다. P type 이 0 이 아닌 데이터 메시지(Data Message)는 “Special Considerations” 참조.

8.2.6.3 헤더 바이트 3 (*Header Byte 3*) — 이 헤더 바이트는 다른 HSMS 메시지들에 대해 다른 방법으로 사용된다. 컨트롤 메시지(Control Message)는 0 또는 상태코드를 포함한다. P Type(아래 참조)= 0 인 데이터 메시지(Data Message)는 SECS 기능(SECS Function)을 포함한다. P Type 이 0 이 아닌 데이터 메시지(Data Message)는 “Special Considerations”를 참조.

8.2.6.4 PType — P Type (Presentation Type)은 헤더의 byte 4 를 점유하는 8-bit의 무 부호 정수 값을 가지며, 프레젠테이션 레이어(presentation layer) 메시지 유형을 정의하는 열거된 유형으로 사용된다: 메시지 헤더(Message Header)와 메시지 텍스트(Message Text)는 어떻게 부호화 될까. P Type = 0 인 경우에만 HSMS 에 의해 SECS-II 메시지가 부호화 된다. 0 이 아닌 P Type 값은 “Special Considerations”참조.

Table 4 PType

<i>Value</i>	<i>Description</i>
0	SECS-II Encoding (SECS-II 부호화)
1–127	하위 표준에 대한 예약
128–255	예약, 사용하지 않음

8.2.6.5 SType — S Type (Session Type)은 헤더 바이트 5 를 차지하고 있는 1 -byte 의 무부호 정수 값이다.

8.2.6.6 SType 은 해당 메시지가 HSMS 데이터 메시지 (HSMS Data Message, value=0)인지 또는 HSMS 컨트롤 메시지(HSMS Control Message)중 하나인지를 식별하는 열거된 형태이다. 테이블 내에서 명시적으로 정의되지 않은 값들은 “Special Considerations”에 명시되어 있다.



Table 5 SType

<i>Value</i>	<i>Description</i>
0	Data Message
1	Select.req
2	Select.rsp
3	Deselect.req
4	Deselect.rsp
5	Linktest.req
6	Linktest.rsp
7	Reject.req
8	(사용하지 않음)
9	Separate.req
10	(사용하지 않음)
11-127	하위 표준에 대한 예약
128-255	예약, 사용하지 않음

8.2.6.7 시스템 바이트 (*System Bytes*) — 시스템 바이트(System Byte)는 헤더 바이트 6-9 를 차지하는 4-바이트 필드이다. 시스템 바이트(System Byte)는 열린(open) 트랜잭션(transaction)의 집합에서만 트랜잭션(transaction)을 식별 하는데 사용된다.

8.2.6.8 유일성 (*Uniqueness*) — 주 데이터 메시지(Primary Data Message)의 시스템 바이트(System Bytes), Select.req, Deselect.req 혹은 Linktest.req 메시지는 연결의 동일한 종단 점(end)에서 시작된 현재 열린(open) 트랜잭션(transaction)과 다른 고유한 것이어야 하며, 또한 가장 최근에 완료된 트랜잭션(transaction)과도 고유해야 한다.

8.2.6.9 응답 메시지 (*Reply Message*) — 응답 데이터 메시지(Reply Data Message)의 시스템 바이트(System Bytes)는 대응되는 주 메시지(Primary Message)의 시스템 바이트와 동일해야 한다. Select.rsp, Deselect.rsp 혹은 Linktest.rsp 의 시스템 바이트(System Bytes)는 각각의 “req” 메시지의 시스템 바이트와 동일해야 한다.

8.3 유형별 HSMS 메시지 형식 (*HSMS Message Formats by Type*) — HSMS 메시지에서 헤더 바이트의 구체적인 해석은 S Type 필드의 값에 의해 정의된 특정한 HSMS 메시지에 따라 다르다. 정의된 메시지의 전체 집합은 P Type = 0 (SECS-II 메시지 형식)의 결과로 보여주는 아래 표에 정리된 바와 같다.

Table 6 HSMS 메시지 형식 요약 (HSMS Message Format Summary)

	<i>Message Header</i>						
<i>Message Type</i>	<i>Bytes 0-1 SessionID</i>	<i>Byte 2</i>	<i>Byte 3</i>	<i>Byte 4 PType</i>	<i>Byte 5 SType</i>	<i>Bytes 6-9 System Bytes</i>	<i>Message Text</i>
Data Message	*	W-Bit and SECS Stream	SECS Function	0	0	Primary: UniqueReply: primary 와 동일	Text
Select.req	*	0	0	0	1	고유함	none



	Message Header						
Message Type	Bytes 0-1 SessionID	Byte 2	Byte 3	Byte 4 PType	Byte 5 SType	Bytes 6-9 System Bytes	Message Text
Select.rsp	Same as .req	0	Select Status	0	2	.req 와 동일	none
Deselect.req	*	0	0	0	3	고유함	none
Deselect.rsp	Same as .req	0	Deselect Status	0	4	.req 와 동일	none
Linktest.req	0xFFFF	0	0	0	5	고유함	none
Linktest.rsp	0xFFFF	0	0	0	6	.req 와 동일	none
Reject.req	거부된 메시지와 동일	거부된 메시지의 PType 또는 SType	Reason Code	0	7	거부된 메시지와 동일	none
Separate.req	*	0	0	0	9	고유함	none

#1 * 하위 표준에 의해 정의된 추가 사양을 나타냄.

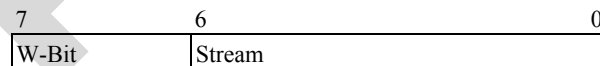
8.3.1 *SType=0: Data Message* — SType = 0 인 HSMS 메시지는 Primary 나 Reply 중 데이터(Data) 메시지를 전송하는 HSMS 데이터 절차에 의해 사용된다. 메시지 형식은 다음과 같다:

8.3.2 HSMS 메시지 길이(HSMS Message Length)는 항상 10 (헤더 단독 길이) 이거나 그 이상이다.

8.3.3 HSMS 메시지 헤더(HSMS Message Header)는 다음과 같다:

8.3.3.1 *Session ID* — 위에 설명된 것과 같이 구체적인 값은 하위 표준을 따른다.

8.3.3.2 *Header Byte 2* — P Type 의 값 = 0 (SECS-II)인 메시지에 대해, 헤더 바이트 2 는 아래와 같은 형식을 가진다.



8.3.3.3 헤더 바이트 2 (Header Byte 2)의 최 상위 비트(bit 7)는 W-Bit 이며, 주 메시지(Primary Message)에서의 W-Bit 는 주 메시지(Primary Message)가 응답(Reply) 메시지를 예상하는지의 여부를 나타낸다. 응답(Reply)이 예상되는 주 메시지(Primary Message)는 W-Bit 를 1 로 설정 해야 하며, 응답(Reply)을 예상하지 않는 주 메시지(Primary Message)는 W-Bit 를 0 으로 설정해야 한다. 응답 메시지(Reply Message)는 언제나 W-Bit 를 0 으로 설정해야 한다. 헤더 바이트 2(Header Byte 2)의 낮은 순위 7 비트(bits 6-0)는 메시지에 대해 SECS 스트림(SECS Stream)을 포함한다. 스트림(Stream)은 7-bit 의 무 부호 정수 값이며 이는 메시지의 주된 주제(major topic)를 식별하고, 그 사용은 SEMI E5(SECS-II)에 정의되어 있다.

8.3.3.4 헤더 바이트 3 (*Header Byte 3*) — PType 값 = 0 을 가진 메시지에 대해, 헤더 바이트(Byte) 3 은 메시지에 대한 SECS 기능(SECS Function)을 포함하고, 그 기능(Function)은 메시지의(Stream 내에서) 부차적인 주제를 식별하는 8-비트의 무 부호 정수 값이며, 그 사용은 SEMI E5(SECS-II)에 정의되어 있다.



기능(Function) 중 최하위 비트 (bit 0)는 데이터 메시지(Data Message)가 Primary 인지 Reply 인지의 여부를 정의한다. 값 1 은 Primary 를 나타내며, 값 0 은 Reply 를 나타낸다.

8.3.3.5 *PType* — SECS-II 메시지에 대해 P Type = 0 으로 설정.

- SType = 0

8.3.3.6 시스템 바이트 (*System Bytes*) — PType = 0 인 경우(SECS-II)는 다음의 정의가 적용된다. 주 메시지(Primary Message)에 대해, 시스템 바이트(System Byte)는 연결(Connection)의 동일한 종단 점(end)에서 시작된 모든 다른 열린 트랜잭션(transaction)로 부터 해당 트랜잭션(transaction)을 식별하는 고유한 값을 포함한다. 응답 메시지(Reply Message)의 경우, 시스템 바이트(System Bytes)는 해당 주 메시지(Primary Message)와 같은 동일한 값을 포함한다.

8.3.3.7 HSMS 메시지의 텍스트 (HSMS Message Text)는 PType 필드에서 구체화된 대로 포맷된 데이터 메시지(Data Message)의 텍스트를 포함한다. PType = 0 인 경우, 텍스트는 SECS-II 메시지로 포맷된다.

NOTE 4: 일부 데이터 메시지(Data Message)는 텍스트 없이 헤더만으로 구성된다.

8.3.4 *SType=1: Select.req* — SType 1 인 HSMS 메시지는 “Select Request” 컨트롤 메시지(Control Message)이며, HSMS 통신 설정을 위해 절차의 개시자에 의해 사용된다. 형식은 다음과 같다:

8.3.4.1 메시지 길이(Message Length)는 항상 10 이다 (헤더만).

8.3.4.2 HSMS 메시지 헤더(HSMS Message Header)는 다음과 같다:

8.3.4.3 *SessionID* — 위에 설명된 것과 같이 구체적인 값은 하위 표준을 따른다.

- Header Byte 2 = 0
- Header Byte 3 = 0
- PType = 0
- SType = 1

8.3.4.4 시스템 바이트 (*System Bytes*) — 열린(open) 트랜잭션(transaction)중 고유 값.

8.3.5 *SType=2: Select.rsp* — SType 2 를 가지는 HSMS 메시지는 “Select Response” 컨트롤 메시지(Control Message)이며, HSMS 통신의 설정을 위한 절차상 Select.req 컨트롤 메시지(Control Message)의 응답으로 사용된다. 메시지의 형식은 다음과 같다:

8.3.6 메시지 길이(Message Length)는 항상 10 이다 (헤더만).

8.3.7 HSMS 메시지 헤더(HSMS Message Header)는 다음과 같다:

8.3.7.1 *SessionID* — 대응하는 Select.req 상의 Session ID 값은 반드시 동일해야 한다.

- Header Byte 2 = 0

8.3.7.2 헤더 바이트 3 (*Header Byte 3*) — SelectStatus. 0 코드는 선택 작업 (Select operation)의 성공을 나타내며, 0 이 아닌 코드는 실패를 나타낸다.



Table 7 SelectStatus

<i>Value</i>	<i>Description</i>
0	통신의 설정(Communication Established). 선택(Select)이 성공적으로 완료됨.
1	이미 통신이 활성화 되어 있음(Communication Already Active). 이전의 선택에서 이번 선택에서 선택된 개체로 이미 통신을 설정함.
2	연결이 준비 되지 않음(Connection Not Ready). 연결(Connection)이 선택 요청을 승인할 준비가 되지 않음.
3	연결 소진(Connect Exhaust). 연결이 승인되었지만, 개체는 이미 개별 TCP/IP 연결을 서비스 하고 있고 주어진 시간에 하나 이상의 연결 제공이불가능함.
4-127	선택 실패에 대한 하위 표준에서 구체화되는 원인을 위해 유보
128-255	선택 실패에 대한 로컬 개체에서 구체화되는 원인을 위해 유보.

- PType = 0
- SType = 2

8.3.7.3 시스템 바이트 (*System Bytes*) — 해당 Select.req 에 대응하는 시스템 바이트(System Bytes)의 값과 동일하다.

8.3.8 *SType=3: Deselect.req* — SType 3 인 HSMS 메시지는 “Deselect Request” 컨트롤 메시지(Control Message)로서 HSMS 통신을 종료하기 위해 선택(Select) 절차의 개시자에 의해 사용된다. 메시지 형식은 다음과 같다:

8.3.9 메시지 길이(Message Length)는 항상 10 이다 (헤더만).

8.3.10 HSMS 메시지 헤더(HSMS Message Header)는 다음과 같다:

8.3.10.1 *SessionID* — SessionID 는 종료된 특정 HSMS 세션을 나타내기 위해 이전에 전송된 Select.req 의 Session ID 값과 일치해야 하며, 하위 표준에서 구체화된 내용에 따른다

- Header Byte 2 = 0
- Header Byte 3 = 0
- PType = 0
- SType = 3

8.3.10.2 시스템 바이트 (*System Bytes*) — 열린(open) 트랜잭션(transaction)중 고유 값.

8.3.11 *SType=4: Deselect.rsp* — SType 4 인 HSMS 메시지는 “Deselect Response” 컨트롤 메시지(Control Message)로서 HSMS 통신 종료를 위한 선택 해제(Deselect) 절차에서 Deselect.req 컨트롤 메시지(Control message)에 대한 응답으로 사용된다. 메시지 형식은 다음과 같다:



8.3.12 메시지 길이(Message Length)는 항상 10 이다 (헤더만).

8.3.13 HSMS 메시지 헤더(HSMS Message Header)는 다음과 같다:

8.3.13.1 *SessionID* — Deselect.req 에 대응하는 Session ID 는 반드시 동일 해야 한다

- Header Byte 2 = 0

8.3.13.2 헤더 바이트 3 (*Header Byte 3*) — DeselectStatus. 0 코드는 선택 해제 작업 (Deselect operation)의 성공을 나타낸다. 0 이 아닌 코드는 실패를 나타낸다.

Table 8 선택 해제 상태 (Deselect Status)

Value	Description
0	종료된 통신(Communication Ended). 선택 해제 (Deselect)가 성공적으로 완료됨.
1	통신이 설정되지 않음(Communication Not Established). HSMS 통신이 선택으로 설정되지 않았거나, 이전의 선택 해제로 (Deselect)로 이미 종료됨.
2	통신이 진행 중(Communication Busy). 세션이 응답 개체에 의해 여전히 사용 중으로 통신이 매끄럽게 종료될 수 없음. 이 경우 최초 요청자가 반드시 통신을 종료해야 한다면, 분리 절차는 반드시 최후의 수단으로 사용 되어야 함.
3-127	선택 해제(Deselect) 실패에 대한 하위 표준에서 구체화되는 원인을 위해 유보
128-255	선택 해제(Deselect)실패에 대한 로컬 개체 에서 구체화되는 원인을 위해 유보

- PType = 0
- SType = 4

8.3.13.3 시스템 바이트 (*System Bytes*) —Deselect.req 에 대응하는 시스템 바이트는 동일하다.

8.3.14 *SType=5: Linktest.req* — SType 5 인 HSMS 메시지는 “Linktest Request” 컨트롤 메시지(Control Message)이며, HSMS 연결(HSMS Connection)의 완전성(integrity)을 검증하는데 사용되거나, 주기적인 하트비트(heartbeat)로 사용된다. 메시지 형식은 다음과 같다:

8.3.15 메시지 길이(Message Length)는 항상 10 이다 (헤더만).

8.3.16 HSMS 메시지 헤더(HSMS Message Header)는 다음과 같다:

- SessionID = 0xFFFF (in binary, all ones)
- Header Byte 2 = 0
- Header Byte 3 = 0
- PType = 0
- SType = 5



8.3.16.1 시스템 바이트 (*System Bytes*) — 열린(open) 트랜잭션(transaction)내에서 고유 값.

8.3.17 *SType=6: Linktest.rsp* — *SType* 6 인 HSMS 메시지는 “Linktest Response” 컨트롤 메시지(Control Message)이며, 이는 Linktest 절차 중 Linktest.req 컨트롤 메시지(Control message)에 응답으로 사용된다. 메시지 형식은 다음과 같다:

8.3.18 메시지 길이(Message Length)는 항상 10 이다 (헤더만).

8.3.19 HSMS 메시지 헤더(HSMS Message Header)는 다음과 같다:

- SessionID = 0xFFFF (binary, all ones)
- Header Byte 2 = 0
- Header Byte 3 = 0
- PType = 0
- SType = 6

8.3.19.1 시스템 바이트 (*System Bytes*) — 해당 Linktest.req 에 대응하는 시스템 바이트는 동일하다.

8.3.20 *SType=7: Reject.req* — *SType* 7 인 HSMS 메시지는 해당 시점에서 유효하지 않거나 메시지의 수신자가 지원하지 않는 수신된 유효한 HSMS 메시지의 응답으로 사용된다. 이는 하위 표준 사용 시도나 메시지수신자가 지원하지 않는 사용자 정의의 확장을 처리하기 위한 것이다 (예를 들어, 어떤 값과 동일한 *SType* 은 해당 표준에서 정의되지 않는다). 이는 개체가 대응 하는 열린(open) 트랜잭션(transaction)이 없는 경우에 응답으로써 (짝수의 *S Type*) 컨트롤 메시지(control message)를 수신했을 경우에 사용되어야 한다.

8.3.21 HSMS 메시지 헤더(HSMS Message Header)는 다음과 같다:

8.3.21.1 *SessionID* — 거부된 메시지의 Session ID 의 값과 동일하다.

8.3.21.2 헤더 바이트 2 (*Header Byte 2*) — ReasonCode = P Type Not Supported 의 경우, 거부된 메시지의 PType 과 동일하며, 그렇지 않다면 거부당한 메시지의 S Type 값과 동일하다.

8.3.21.3 헤더 바이트 3 (*Header Byte 3*) — 원인(Reason) 코드 (항상 0 이 아님)

Table 9 ReasonCode

<i>Value</i>	<i>Description</i>
1	지원되지 않는 S Type (SType Not Supported). HSMS 표준이나 개체가 지원하는 특정 하위 표준에서 정의 되지 않는 S Type 값을 가진 메시지가 수신됨.
2	지원되지 않는 PType(PType Not Supported). 위와 같으나 P Type 에 적용.
3	열려있지 않은 트랜잭션 (Transaction Not Open). 이에 해당하는 두드러진 요청 메시지가 없을 때 응답 컨트롤 메시지(Response control message)를 수신한다.
4	선택되지 않은 개체 (Entity Not Selected). SELECTED 상태가 아닐 때 데이터 메시지를 수신한다.
4-127	거부(reject)에 대한 하위 표준에서 구체화하는 원인을 위한 유보.



Value	Description
128-255	거부(reject)에 대한 로컬 개체에서 구체화하는 원인을 위한 유보.

- PType = 0
- SType = 7

8.3.21.4 시스템 바이트 (*System Bytes*) — 거부된 메시지에 대응하는 시스템 바이트(System Byte)와 동일하다.

8.3.22 *SType=9: Separate.req* — SType = 9 인 HSMS 메시지는 즉각적으로 HSMS 통신을 종료하는데 사용된다. SType 값의 예외와 함께, 이는 Deselect.req 메시지와 동일하다. 그 목적은 즉각적이고 예외 없는 HSMS 통신을 종료하는 것이며, 어떠한 응답도 정의되지 않는다.

9 특별 고려 사항 (Special Considerations)

9.1 일반적인 고려사항 (*General Considerations*)

9.1.1 통신 실패 (*Communications Failures*) — 통신의 실패가 감지되었다면, 개체는 TCP/IP 연결을 종료해야 한다. 연결의 종료 시, 해당 시점에서 개체는 통신을 재설정하는 시도를 할 수 있다.

9.2 TCP/IP 고려 사항 (*TCP/IP Considerations*)

9.2.1 연결 분리 시간 (T5) (*Connect Separation Time (T5)*) — 연결 절차가 일부 네트워크 활동을 시작하며, 연결을 승인할 준비가 되지 않은 개체의 IP 주소(IP Address)나 포트 번호(Port Number)로 능동 모드 연결 절차를 빈번하게 사용하는 것은 TCP/IP 운영에 좋지 않은 영향을 줄 수도 있다. 수동 모드는 네트워크 활동을 발생시키지 않으며, 로컬 어플리케이션의 성능에 영향을 줄 수는 있지만 네트워크에 나쁜 영향을 주지는 않는다. 능동 모드에서 연결을 시작하는 개체는 여기에 설명된 절차에 상응하는 방법으로 연결 절차를 수행하는 것을 제한해야 한다.

9.2.1.1 (성공적이던 성공적이지 못하던 간에) 여하한 방법으로 능동 연결 절차의 종료 후, 개체(Entity)는 T5 연결 분리 시간(T5 Connect Separation Time)이 만료 되기(lapsed)전까지는 다른 능동 연결 절차(동일 원격 개체(Remote Entity)에 대해)를 시작해서는 안 된다. 연결 작업의 분리는 T5 연결 분리 시간(T5 Connect Separation Time) 간격의 합에, 연결 작업 자체에 소요된 시간이다 더해진다.

9.2.2 *NOT SELECTED Timeout (T7)* — NOT SELECTED 상태의 진입은 상태 전환 #2 (TCP/IP 연결의 설정)로 달성된다. 개체가 SELECTED 상태에 진입하기 전에 얼마나 오랫동안 NOT SELECTED 상태로 유지될 것 인지, NOT CONNECTED 상태로 돌아갈 것인지에 대한 시간 제한을 가진다.

9.2.2.1 특히 하나 이상의 TCP/IP 연결을 승인할 수 없는 일부 개체들은, 다른 개체와의 통신이 불가능하게 될 것 처럼 NOT SELECTED 상태에 머무름으로써 이들 운영에 손상을 입힐수 있다. 그러한 개체들은 통신이 NOT SELECTED 상태에서 T7 타임아웃(timeout) 기간 보다 길게 유지될 경우 TCP/IP 연결을 끊어야 한다 (상태 전환 #3).

9.2.3 네트워크 문자간 타임아웃 (T8) (*Network Intercharacter Timeout (T8)*) — TC/IP 가 메시지 프로토콜이라기 보다는 스트림(stream)인 까닭에 단일 HSMS 메시지의 모든 부분을 이루는 바이트가



TCP/IP 프로토콜의 규정 위반 없이 개별 TCP/IP 메시지를 통해 전송될 수 있다. 이들 개별 메시지들은 상당한 기간이 걸려 분리될 수도 있기 때문에, 네트워크 문자간 타임아웃 (T8) (Network Intercharacter Timeout (T8))이 정의된다.

9.2.3.1 T8 을 필요로 하는 통신 이슈들이 전적으로 메시지의 송신자의 통제하에 있지않다는 점을 제외하고 T8 은 SECS-I T1 타이머(timer)의 목적과 유사하므로, 메시지 수신인에 대해서만 정의한다. 특히, 부분 메시지가 수신되고, T8 타임아웃(timeout) 기간이 전체 메시지의 수신 전에 만료 된다면, 수신 개체는 위에 설명된 경우를 통신 실패로 간주한다.

9.2.4 단일 설치 포트로의 다중 연결 요청 (*Multiple Connection Requests Directed to a Single Published Port*) — 설치된 포트에서 수동 개체가 연결을 수락하면, TCP/IP 는 (필수는 아니지만) 개체가 동일한 설치 포트방향하는 다른 추가 연결을 받아들여 승인하는 것을 허용한다.

9.2.4.1 HSMS 는 (필수는 아니지만)개체들이 위와 같은 방식으로 운영되하는것을 허용하지만, HSMS 준수 목적으로 형성된 각각의 연결은 동일한 설치 포트의 다른 연결들이 완전히 독립적으로 이루어진것처럼 HSMS 상태 다이어그램 내에서 정의된 동작을 나타내야 한다.

9.2.4.1.1 수동모드의 개체로 추가 연결 요청의 거부 (*Rejection of Additional Connection Requests by a Passive Mode Entity*) — HSMS 통신을 위해 하나 이상의 TCP/IP 연결 서비스가 불가한 수동모드 개체는 추가 연결 요청에 대해 다음의 3 가지 절차 중 하나를 따른다.

- 연결을 승인, 그러나 Communication Already Active 응답 코드를 가진 후속 HSMS 선택 절차를 따름. HSMS 상태 다이어그램(HSMS State Diagram)의 목적으로, 연결 절차는 성공적으로 CONNECTED 상태로 전환되어 종료되지만, HSMS 통신은 설정되지 않고 NOT SELECTED 하위 상태에 머문다. 연결이 거부당한(HSMS Select Procedure 참조) 원인에 대해 원격 개체에 가장 많은 정보를 제공하는 것을 선택하는것이 선호되지만, 로컬 개체에 추가적인 실행 필수요건이 부과된다.
- 연결 요청에 대한 능동적 거부. 이는 t_snddis 절차를 이용하여 TLI 실행으로 수행될 수 있으며, 원격 개체의 성공적이지 못한 연결 절차의 종료를 야기할 수 있다. API's 특히 BSD Sockets 의 실행들은 능동 거부(active reject)를 시작하는 것을 허용하지 때문에 이 옵션은 모든 실행에서 사용될 수는 없다. 하지만, BSD Sockets 을 포함하는 모든 TCP/IP 실행이 원격 개체로부터 능동 거부(active reject)에 대해 적절하게 응답한다.
- 연결 요청의 수락 거절. 로컬 개체에서 어떠한 조치도 취해지지 않으며: 원격 개체의 연결 절차가 결국에는 시간의 만료되어 종료된다. 원격 개체에 상당한 지연이 발생할 수 있으므로, 권장 되지는 않지만 이 옵션이 허용된다. 하지만 이는 네트워크 자원(resource) 제한이 있을 경우에만 실행할 수 있는 대안이다.

9.2.4.1.2 수동 로컬 개체의 문서화는 연결을 거절하기 위해 로컬 개체가 사용하는 방법을 나타내야만한다.

9.3 특정 HSMS 고려사항 (HSMS-Specific Considerations)

9.3.1 컨트롤 트랜잭션 T6 컨트롤 타임아웃(*Control Transactions T6 Control Timeout*) — 컨트롤 메시지의 번호는 메시지 교환이나 트랜잭션(transaction)에 필요한 절차의 일부이다: 컨트롤(control) 서비스 개시자의 <xx>.req 는, <xx>.req 의 수신자로부터 이에 대한 응답으로 <xx>.rsp 가 뒤에 따른다. 컨트롤(control) 트랜잭션(transaction)은 <xx>.req 요청이 전송된 시점으로부터 <xx>.rsp 를 수신한 시점까지 열린(open) 상태로 간주된다.



9.3.1.1 컨트롤(control) 트랜잭션(transaction)이 열린 상태로 유지된 시간은 T6 컨트롤 트랜잭션(transaction) 타임아웃(timeout)을 따른다. 컨트롤(control) 트랜잭션(transaction)의 시작 시, 로컬 개체는 T6 타임아웃(timeout) 값과 같도록 타이머를 설정해야 한다. 타이머의 만료 전에 트랜잭션(transaction)이 적절하게 완료된 경우, 타이머가 취소된다. 트랜잭션(transaction)의 적절한 종료 전에 타이머가 만료 되었다면, 트랜잭션(transaction)이 개시자에 의해 종료된 것으로 간주하고 HSMS 통신의 실패로 간주된다.

9.3.2 절차 및 “stateless” 트랜잭션(*Procedures and “Stateless” Transactions*) — 대부분의 HSMS 컨트롤(control) 절차는 트랜잭션(transaction)과 관련이 있다: 개시자는 응답 개체에게 요청 메시지를 전송하고 응답 메시지를 기다린다. 응답 개체는 개시자의 요청 메시지를 수신하고 응답을 전송한다.

9.3.2.1 이러한 트랜잭션(transaction)은 다음과 같은 경우 “stateless”임을 유의: 트랜잭션(transaction)의 개시자가 응답을 기다리는 동안, 응답 외에 다른 메시지를 수신할 수 있고, 이 메시지가 기존의 트랜잭션(transaction)이 시작되었던 당시의 개시자의 상태에 유효한 메시지일 수 있다. 예를 들어, 두 개체가 동시에 트랜잭션(transaction)을 시작할 수 있고, 그 결과 “TRANSACTION OPEN”이나 “TRANSACTION NOT OPEN”의 어떤 상태도 HSMS 상태 기기에 반영되지 않는다. 실행시에는 이러한 상태 정보의 사용은 로컬 개체에 특성화 된 이슈로 엄격하게 한정한다.

9.3.3 대체 메시지 유형과 헤더 바이트 값 (Alternative Message Types and Header Byte Values) — HSMS 표준은 PType 이나 SType 필드의 모든 가능한 열거된 값을 정의하지 않는다. 또한 헤더(Header) 바이트 2 와 3 은 SType 이 0 인 메시지의 PType 에 의해 결정된 형식을 가지지만, 그 외의 모든 SType 값에 따라 달라지게 된다. 메시지 텍스트 형식화(formatting) PType 에 의해 정의되지만, 오직 데이터 메시지에 한정한다.

9.3.3.1 하위 표준은 이러한 규약에 부합해야만 하며, 특히, SType = 0 이고, PType 값이 0 이 아니라고 규정하는 하위 표준은 메시지 텍스트 부호화와 헤더 바이트 2 와 의 해석 모두를 구체화한다. STypes 이 0 은 아니지만, 이 표준에 구체화되었다면, PType 은 반드시 0 이어야 하고 어떠한 메시지 텍스트도 전송되지 않는다. 하위 표준에서 정의된 S Type 에 대해, 헤더 바이트 2 와 3 의 의미는 SType 값마다 구체화되고, 앞서 나온 설명된 것들과 일관된 방법으로 PType 필드가 사용되는 한 이들 SType 은 선택적으로 메시지 텍스트를 정의한다.

9.4 SECS-II 고려사항 (*SECS-II Considerations*) — SECS-II 표준 (SEMI E5)은 SECS-I (SEMI E4)를 언급한다. 이 섹션은 HSMS 가 SECS-II 메시지 전송에 사용될 때 SECS-II 에 한정적인 이슈들을 언급한다.

9.4.1 응답 매칭 (*Reply Matching*) — 송신자가 W-Bit 1 로 주 메시지(Primary Message)를 전송하면 (Reply Expected), 송신자는 다음 요구사항이 만족하는 헤더를 가진 응답 메시지(Reply message)를 예상한다.

- 응답(Reply)의 SessionID 는 주 메시지(Primary Message)의 Session ID 와 반드시 일치해야 한다.
- 응답(Reply)의 스트림(Stream)은 반드시 주 메시지(Primary Message)의 스트림(Stream)과 일치해야 한다.
- 응답(Reply)의 기능(Function)은 주 메시지(Primary Message)의 기능(Function)보다 커야 하거나, 응답(Reply)의 기능(Function)이 0 이어야 한다 (Function Zero Reply).
- 응답(Reply)의 시스템 바이트(System Bytes)는 반드시 주 메시지(Primary Message)의 시스템 바이트(System Byte)와 일치해야 한다.



9.4.1.1 T3 응답 타임아웃 (*T3 Reply Timeout*) — T3 응답 타임아웃(timeout)은 HSMS 메시지 프로토콜이 응답(Reply) 메시지를 기다릴 수 있는 시간을 제한한다.

9.4.1.2 W-Bit1 (Reply Expected)인 주 메시지(Primary Message)를 전송한 후, 송신자는 반드시 T3 값으로 초기화된 응답 타이머를 시작해야 한다. 응답 타이머가 만료되기 전에 송신자가 응답 메시지(Reply Message)를 수신하지 못했다면, T3 타임아웃 오류(T3 Timeout Error)가 발생한다. 송신자는 트랜잭션(transaction)을 종료하고 더 이상 응답 메시지(Reply Message)를 기대하지 않는다.

9.4.1.3 응답(Reply)이 예상되는 경우 각각의 열린(open) 트랜잭션(transaction)은 개별 응답 타이머가 필요하다.

9.4.2 스트림 9 메시지 (*Stream 9 Messages*) — SECS-II 표준은 SECS-II 데이터 아이템(Data Item) MHEAD 나 SHEAD 가 포함된 메시지 텍스트와, 오류 메시지 S9F1, S9F3, S9F5, S9F7, S9F9, S9F11 를 정의하고, 10-byte 의 SECS-I 블록 헤더를 포함할 것을 정의한다.

9.4.2.1 HSMS 를 가진 SECS-II 사용시, MHEAD 및 SHEAD 는 10 바이트의 HSMS 메시지 헤더(HSMS Message Header)를 포함해야 한다.

10 HSMS 문서화 (HSMS Documentation)

10.1 HSMS 실행은 다음의 정보를 문서화 하는 것이 요구된다:

1. 프로토콜 매개변수(parameter) 설정에 대한 메소드(method) (§ 10.1 참조).
2. 각 매개변수(parameter)에 대한 허용 범위와 resolution.
3. TCP/IP 연결 설정을 위해 실행에 수동 모드가 사용되었을 경우, 연결 요청을 거부할 때 옵션이 사용된다.
4. 수신할 수 있는 최대 메시지 사이즈.
5. 전송된 메시지의 예상되는 최대 사이즈.
6. 지원되는 동시에 열린(open) 트랜잭션(transaction)의 최대 수.

10.2 매개변수 설정 (*Parameter Setting*) — HSMS 의 실행시 아래의 매개변수(parameter) 설치(installation) 시간 설정을 대비 해야 한다. 모든 매개변수(parameter)의 범위와 resolution 이 적어도 테이블 내에 표시 되어야 하며, 모든 매개변수(parameter)는 전원이 나가거나 시스템 소프트웨어가 재 시작될 경우, 설정이 유지되는 방식으로 반드시 저장되어야 한다.

Table 10 매개변수 설정 (Parameter Setting)

<i>Parameter Name</i>	<i>Value Range</i>	<i>Resolution</i>	<i>Typical Value</i>	<i>Description</i>
T3 Reply Timeout	1–120 seconds	1 second	45 seconds	응답 타임아웃(Reply timeout). 응답 메시지를 예상하는 개체가 응답을 기다리는 최대 시간을 지정한다.



<i>Parameter Name</i>	<i>Value Range</i>	<i>Resolution</i>	<i>Typical Value</i>	<i>Description</i>
T5 Connect Separation Timeout	1–240 seconds	1 second	10 seconds	연결 분리 타임아웃(Connection Separation Timeout). 주어진 원격 개체에 연속적인 연결을 시도들 사이에 소요되는 시간을 지정한다.
T6 Control Transaction Timeout	1–240 seconds	1 second	5 seconds	컨트롤 트랜잭션 타임아웃(Control Transaction Timeout). 통신 실패로 간주되기 전, 컨트롤(control) 트랜잭션(transaction)이 열린 상태로 유지되는 시간을 지정한다.
T7 NOT SELECTED Timeout	1–240 seconds	1 second	10 seconds	통신 실패로 간주되기 전, TCP/IP 연결이 NOT SELECTED 상태(즉, HSMS 활동이 없는)로 유지되는 시간.
T8 Network Intercharacter Timeout	1–120 seconds	1 second	5 seconds	통신 실패로 간주되기 전, 만료된 단일 HSMS 메시지의 연속적인 바이트 사이의 최대 시간.
Connect Mode	PASSIVE, ACTIVE	-	-	연결 모드(Connect Mode). HSMS 연결이 이루어지는 동안 해당 로컬 개체가 사용할 로직을 지정한다.
Local Entity IP Address and Port number	TCP/IP 규약에 따라 결정된다	-	-	PASSIVE 모드로 운영되는 개체에 대해 요구된다. 입력된 연결 요청을 수신하는 로컬 개체의 주소를 결정한다.
Remote Entity IP Address and Port Number	TCP/IP 규약에 따라 결정된다	-	-	ACTIVE 모드에서 운영되는 개체에 대해 요구된다. 로컬 개체가 연결을 시도하는 원격개체의 주소를 결정한다.

#1 위에 보여지는 매개변수(Parameter) 기본값은 10 node 나 그 이하의 작은 네트워크를 위한 것이며, 보다 큰 네트워크 구성을 위한 설정은 별도의 조정이 필요하다.



APPENDIX 1

NOTICE: The material in this Appendix is an official part of SEMI E37 and was approved by full letter ballot procedures.

이 부록의 내용은 SEMI E39 의 공식적인 부분이며 비 무기명 투표 절차로 승인되었다.

A1-1 TLI 를 사용하는 TCP/IP 절차 및 BSD 소켓 인터페이스 (TCP/IP Procedures Using TLI and BSD Socket Interfaces)

A1-1.1 수동 모드 연결 절차 (Passive Mode Connect Procedure)

Table A1-1 능동 모드 연결 절차 (Passive Mode Connect Procedure)

<i>Intended Action</i>	<i>TLI Construct</i>	<i>BSD Construct</i>	<i>Comment</i>
연결 종단 점(endpoint)을 획득하고 설치된 포트에 이를 연결한다.	tep = t_open(...) t_bind(tep,...)	skt = socket(...) bind(skt,...)	BSD 는 연결 종단 점(endpoint)의 “socket”을 의미한다. TLI 는 TEP (transport end point)를 의미한다.
연결을 시도하는 Socket 을 허용한다.	...	listen(skt,...)	TLI 에서, BSD listen 의 등가는 필요하지 않다.
연결 절차: 연결 요청을 수신하고 이를 승인한다.	t_listen(tep,...) t_accept(tep,...)	accept(skt,...)	
연결 절차: 연결 요청을 수신하지만, 이를 거부(reject)한다.	t_listen(tep,...) t_snddis(tep,...)	...	요청을 수신하여 승인하는 것은 단일 운영(single operation)이므로, BSD API 는 연결 요청의 거부(reject)의 발신을 지원하지 않는다.

A1-1.2 능동 모드 연결 절차 (Active Mode Connect Procedure)

Table A1-2 능동 모드 연결 절차 (Active Mode Connect Procedure)

<i>Intended Action</i>	<i>TLI Construct</i>	<i>BSD Construct</i>	<i>Comment</i>
연결 종단점(endpoint)을 획득한다.	tep = t_open(...) t_bind(tep,...)	skt = socket(...)	TLI 는 능동 개체에 대해 null 주소에 결부시키는 것이 요구된다.
연결 절차: 연결 요청을 전송하고 수동 개체로부터 승인 받거나 거부(reject)를 수신한다.	t_connect (tep,...) t_rcvconnect(tep,...)	connect(skt,...)	BSD 연결은 TLI 기반의 원격 개체로부터 능동 거부(reject)를 적절하게 처리한다.



A1-1.3 연결 종료 (Terminating the Connection)

Table A1-3 연결 종료 (Terminating the Connection)

<i>Intended Action</i>	<i>TLI Construct</i>	<i>BSD Construct</i>	<i>Comment</i>
연결을 해제하고 연결 종단점(endpoint)을 해제한다.	t_sndrel(tep,...) t_close(tep)	close(skt)	BSD 종료의 “좋은 점(gracefulness)”은 로컬 실행 기능(function)에 있다.
접속 중단과 연결 종단점(endpoint)을 해제한다.	t_snddis(tep,...) t_close(tep)	shutdown(skt,2) close(skt)	두 번째 arg = 2 인 경우, 섯다운(shutdown)으로 즉시 추가적인 전송과 수신을 중지한다.

A1-1.4 HSMS 메시지 전송 및 수신 (Sending and Receiving HSMS Messages)

Table A1-4 HSMS 메시지 전송 및 수신 (Sending and Receiving HSMS Messages)

<i>Intended Action</i>	<i>TLI Construct</i>	<i>BSD Construct</i>	<i>Comment</i>
HSMS 메시지전송.	hdr->Len = length; t_snd(tep,hdr,14,0); t_snd(tep,Text,hdr->Len,0);	hdr->Len = length; write(skt,hdr,14); write(skt,Text,hdr->Len);	이 절차는 길이 바이트와 헤더가 결합하여 첫 번째로 단일 14-byte 아이템을 전송하고 그 이후에 텍스트가 뒤따르는 “typical” 실행 스타일을 보여준다. 이는 모든 것을 결합하는 것이 허용되지 않는다는 것을 의미 하지는 않는다.
HSMS 메시지와 수신.	t_rcv(tep,hdr,14,...); t_rcv(tep,Text,hdr->Len,...);	read(skt,hdr,14); read(skt,Text,hdr->Len);	수신에 대해서 위와 동일.

A1-2 HSMS 시나리오 (HSMS Scenarios)

A1-2.1 다음의 시나리오는 전형적인 완성 세션에 사용되는 HSMS 절차를 설명하기 위해 제공된다. 용어, 절차 명, 메시지 명은 이 문서의 나머지 부분에 추가적으로 설명 된다. 또한 각 개체가 HSMS 선택(Select) 절차, 선택해제(Deselect) 혹은 분리(Separate) 절차, HSMS 데이터 메시지(HSMS Data Messages)와 트랜잭션(transaction)을 시작할 수 있음을 유의해야 한다. 편의를 위해, 시나리오는 모든 트랜잭션(transaction)의 개시자를 왼쪽에 있는 개체로 보여준다.

A1-2.2 HSMS 통신의 시작 (*Begin HSMS Communication*) — 이 시나리오는 TCP/IP 연결 절차, HSMS 선택 절차, 데이터 메시지의 교환을 설명한다. TCP 에 대한 데이터 메시지 활동(activity)은 설명의 목적으로만 쓰이는 점에 유의해야 한다. 실질적으로 네트워크 활동은 다양할 수 있다. 예를 들어, 데이터 메시지가 보여지는 것과 같이 t-snd (또는 쓰기)에 별도의 콜(call)으로 전송될 때도 TCP/IP 실행은 헤더의 완충 역할을 하고, 텍스트를단일 패킷(packet)으로 전송하거나, 텍스트를 다중 패킷(packet)으로 나눌 수도 있다.



Table A1-5 HSMS 통신 시작 (Begin HSMS Communication)

<i>BSD API Calls</i>	<i>TLI API Calls</i>	<i>Network Activity</i>	<i>TLI API Calls</i>	<i>BSD API Calls</i>
연결 요청을 시작하기 위한 준비.			연결 요청을 수신하기 위한 준비.	
skt = socket(...);	tep = t_open(...); t_bind(tep, ...);		tep = t_open(...); t_bind(tep,...);	skt = socket(...);bind(skt,...); listen(skt,...);
연결 요청을 시작하고 응답을 기다림.			연결 요청을 수신, 이를 승인하여 응답을 전송.	
connect(skt,...);	t_connect(tep,...); t_rcvconnect(tep,...);	TCP/IP Connect Req Msg(s) TCP/IP Accept Msg(s)	t_listen(tep,...);t_acce pt(tep,...);	accept(skt,...);
HSMS 선택(Select) 절차의 시작: 요청을 전송하고 응답을 수신한다.		Select.req Message Select.rsp Message	HSMS 선택 절차에 응답한다. 요청을 수신하고 응답을 전송한다.	
write(skt,hdr,14); read(skt,hdr,14);	t_snd(tep,hdr,14,0); t_rcv(tep,hdr,14,...);		t_rcv(tep,hdr,14,...); t_snd(tep,hdr,14,0);	read(skt,hdr,14); write(skt,hdr,14);
길이 바이트와 헤더로된 HSMS 데이터 메시지를 전송한다. 텍스트(Text)가 뒤따른다.		HSMS Data Message (hdr) HSMS Data Message (text)	길이 바이트와 헤더로된 HSMS 데이터 메시지를 수신한다. 텍스트(Text)가 뒤따른다	
hdr->Len = Length; write(skt,Text,...);	hdr->Len = Length; t_snd(tep,hdr,14,0); t_snd(tep,Text,...);		t_rcv(tep,hdr,14,...); t_rcv(tep,Text,...);	read(skt,hdr,14);read(s kt,Text,...);

A1-2.3 선택 해제를 사용하는 종료된 통신 (*Ending Communication Using Deselect*) — 이 시나리오는 HSMS 세션(HSMS Session)을 종료하는 선택해제(Deselect) 절차를 사용하여 HSMS 세션(HSMS Session)의 종료를 설명한다.

Table A1-6 선택 해제를 사용한 통신 종료 (Ending Communication Using Deselect)

<i>BSD API Calls</i>	<i>TLI API Calls</i>	<i>Network Activity</i>	<i>TLI API Calls</i>	<i>BSD API Calls</i>
Deselect.req 를 전송하고 Deselect.rsp 를 수신한다.			Deselect.req 를 수신하고 Deselect.rsp 를 전송한다.	
write(skt,hdr,14); read(skt,hdr,14);	t_snd(tep,hdr,14,0); t_rcv(tep,hdr,14,...);	Deselect.req Message Deselect.rsp Message	t_rcv(tep,hdr,14,...); t_snd(tep,hdr,14,0);	read(skt,hdr,14); write(skt,hdr,14);
TCP/IP 연결 종료.		TCP/IP Disconnect Msg(s)	연결 종료에 응답한다.	
shutdown(skt,2); close(skt);	t_snddis(tep); t_close(tep);		t_rcvdis(tep,...);	close(skt);

A1-3 HSMS 대체 모드 연결 절차 (HSMS Alternating Mode Connect Procedure)

A1-3.1 일부 사용자는 특정 시점에서 주어진 개체가 사용하게 될 연결모드(능동 혹은 수동)가 어떤 것 일지 결정하지 않게 하는 특별한 요구사항을 가지기도 한다. 이와 같은 경우, 로컬 개체(Local Entity)는 연결이 성공적으로 설정될 때까지 능동 모드와 수동 모드의 연결 절차를 번갈아서 시도 한다. 로컬 개체가 수동



상태일 때 설치된 포트를 제공해야 함을 유의해야 한다. 대체 로컬 개체(Alternating Local Entity)에서 일반적인 논리 시퀀스(sequence)는 다음과 같다:

1. T5 연결 분리 타임아웃(timeout)과 같거나 긴 `t_rcvconnect`에 대해 타임아웃(timeout) 값을 사용하여 § 6.3.3에 설명된 것과 같이 능동 연결 절차를 시도한다.
2. 능동 연결 절차가 성공적으로 완료된다면, 대체 모드 연결 절차도 성공적으로 완료된다.
3. 능동 연결 절차가 성공적으로 종료되지 못한다면, § 6.3.2에 설명된 것과 같이 T5의 연결 분리 타임아웃(timeout)과 같거나 긴 `t_listen`에 대한 타임아웃(timeout)으로 수동 연결 절차를 시도한다.
4. 수동 연결 절차가 성공적으로 완료된다면, § 6.3.2에 설명된 것과 같이 대체 모드 연결 절차가 성공적으로 완료된다.
5. 수동 연결 절차가 성공적으로 종료되지 못한다면, 로컬 개체는 대체 모드 절차를 지속하기 위해 1 단계로 되돌아가거나 성공적으로 종료되지 못한다. 연결을 구성하기 위해 반복적으로 시도되는 위 단계의 시퀀스(sequence)의 횟수는 로컬 개체 특성화 된 이슈이다.

A1-3.2 대체 모드 사이클 타임 (*Alternating Mode Cycle Time*) — 대체 모드 사이클 타임(Alternating Mode Cycle Time)은 대체 모드(Alternating Mode) 개체의 연결 절차(Connect Procedure)가 반복되는 절차 사이의 시간을 의미하며, 상기의 절차에서, 이 시간은 1 단계 재 시작에 앞서 5 단계의 완료와 1 단계의 시작 사이의 간격에 해당한다. 이 시간은 실행시에 다르게 설정할 수 있다.

A1-3.2.1 두 개체가 모두 대체 모드 연결 절차(Alternating Mode Connect Procedure)를 사용하는 경우, 잠금(lock) 단계(모두 능동 모드이거나 모두 수동 모드)에서 두 개체가 연결을 시도하지 않도록, 이들 모두가 서로 다른 대체 모드 사이클 타임을 이용하는 것이 바람직하다. 대체 모드 사이클 타임(Alternating Mode Cycle Time)을 조정하는 것은 T5 조정으로 가능하며, 두 개체의 사이클 타임을 다르게 한다:

A1-3.3 HSMS 연결 조합 (*HSMS Connect Combinations*) — 능동이나 수동 모드간에 변환이 가능하도록 설정된 개체는 수동이나 능동 모드의 원격 개체와 연결이 가능하다. 아래의 리스트는 해당 특정 연결 전략을 가진 표준을 이용한 가능한 조합을 요약하고 있다.

1. ACTIVE로 설정된 Entity “A”는 ALTERNATING이나 PASSIVE로 설정된 Entity “B”와 연결될 수 있고, Entity “A”는 항상 연결(Connection)을 설정한다.
2. ALTERNATING으로 설정된 Entity “A”는 PASSIVE로 설정된 Entity “B”와 연결될 수 있고, Entity “A”는 항상 연결(Connection)을 설정한다.
3. ALTERNATING으로 설정된 Entity “A”는 ALTERNATING으로 설정된 Entity “B”와 연결될 수 있고, 한쪽 종단 점(end)이 연결(Connection)을 설정할 수 있다. 해당 문서에서 설명된 순차 로직이 아닌 다중 회선 연결 로직 (multi-threaded connect logic)을 이용한 실행시, HSMS 연결의 양쪽 종단 점(end)이 동시에 연결을 시도할 수도 있다. 이 경우, 두 가지 개별 TCP/IP 연결이 설정될 수 있으며, 한쪽 연결이 유지되고 다른 한쪽이 종료되도록 규약을 제정하는 것이 필요하다.
4. 양쪽 개체 모두가 ACTIVE로 설정되거나 모두 PASSIVE로 설정되는 것은 허용되지 않는다.



A1-4 HSMS 가 아닌 TCP/IP 프로토콜 (Non-HSMS TCP/IP Protocols)

A1-4.1 전형적인 TCP/IP 실행에서, HSMS 는 동일한 IP 주소(IP Address)에서 TCP/IP 기반의 프로토콜과 공존할 수 있다. 예를 들어, SECS-II 메시지 트랜잭션(transaction)은 어플리케이션이 대용량 데이터 파일을 전송하는 TCP/IP FTP (File Transfer Protocol) 시퀀스(sequence)를 시작하도록 트리거(trigger) 할 수 있다.

A1-5 TCP/IP 가 아닌 프로토콜 (Non-TCP/IP Protocols)

A1-5.1 HSMS 개체로서 동일한 네트워크상의 TCP/IP 외에 다른 프로토콜의 이용은 가능하지만 해당 표준의 범위를 벗어난다. 일반적으로 다른 프로토콜이 네트워크상의 TCP/IP 나 HSMS 개체에 영향을 주지 않는다면, 이들 프로토콜은 사용이 가능하다.

A1-6 다중 LANs (Multiple LANs)

A1-6.1 HSMS 사양은 오직 단일 TCP/IP LAN 만이 고려대상이며, 상호 연결이 가능한 다중 LANs 은 HSMS 의 범위 밖이다. 그러나, TCP/IP 실행은 일반적으로 게이트웨이(gateways), 라우터(router)와 유사한 개체를 통해 매끄럽게 구성되도록 지원하기 때문에, 상호 연결된 LAN's 을 통해 HSMS 연결(HSMS Connection)을 설정하는 것이 가능하기도 하다.

A1-7 TCP/IP 의 물리적 레이어 (TCP/IP Physical Layer)

A1-7.1 HSMS 는 IP 의 물리적인 레이어(layer)를 지정하지 않으며, TCP/IP 가 지원하는 물리적 레이어(layer)의 사용이 가능하다. 대부분, TCP/IP 실행은 물리적 레이어(layer)로써 Ethernet (IEEE 802.3)을 사용하지만, 일부 TCP/IP 실행은 다른 프로토콜 (예를 들어 Token Bus, IEEE 802.4, 5)을 사용한다. 주어진 설치(installation) 상 상호 운영 능력을 보장하기 위해서는 물리적 레이어(layer)에 대한 추가적인 로컬 표준을 설정하는 것이 권장된다.

A1-8 기정 TCP/IP 번호 (Well-Known TCP/IP Port Numbers)

A1-8.1 일부 TCP/IP 기반의 프로토콜은 다른 프로토콜에는 사용할 수 없는 특정 “Well Known” TCP 포트 번호(TCP Port Number)를 지정하여 설치한다. HSMS 는 특정 “Well Known” TCP 포트 번호(TCP Port Number)를 지정하지는 않지만, 대신 이것을 설정 가능하게 할것을 요구한다. IETF 는 RFC1340 에서 “Assigned Well Known Port Number”를 정의한다.

A1-9 연결 해제와 재 연결 사이의 지연 (Delay Between Disconnect and Re-Connect)

A1-9.1 일부 TCP/IP 시스템은 연결의 양쪽 종단 점(end)에서 동일한 TCP 포트를 사용했을 때, 연결이 종료되었다 즉시 재 연결되는 어플리케이션의 문제가 나타난다. 이러한 시스템을 이용할 때에는 재 연결 전에 연결 중단 후 잠시 지연하는 것을 권장한다. TCP/IP 실행에 따라 지연 시간(delay time)은 다양하지만, 일반적으로 “Maximum Segment Lifetime” 즉 MSL 보다 2 배 정도로 계산된다. 예를 들어, BSD 를 기반으로 한 대부분의 TCP/IP 시스템은 (예를 들어 Sun, AIX) 30 초의 MSL 을 사용하는데, 이때 60 초의 지연 시간(delay)이 적절하다. 긴급한 연결이 필요할 때는, 사용중인 Connect Mode 가 허용하는 한도 내에서 어플리케이션은 다른 포트를 이용해야 한다. 일부 TCP/IP 시스템에서는 이러한 문제가 발생하지 않는다.



A1-10 사용자 정의의 메시지 유형 (User-Defined Message Types)

A1-10.1 장비 공급 업체는 기본 수준의 HSMS 나 다른 하위 표준에서 찾을 수 없는 추가 기능의 개발이 적절하다는 것을 인식하고 있다. 이는 제안된 새로운 하위 표준의 테스트와 개발 중에 가능하다. 새로운 메시지 유형을 통한 사용자 정의의 확장은 인트라 벤더(intra-vendor) 통신 인터페이스로 국한하여 허용된다: 이러한 확장을 요구하는 인터 벤더(inter-vendor) 통신 인터페이스는 HSMS 표준을 따르지 않는 것으로 간주된다.

A1-10.2 장비 업체가 확장이나 표준을 벗어나는 것이 필요하다고 판단한다면, SType 이나 PType 의 “Reserved, not used,” 값을 완전히 분리된 비슷한 수준의 표준을 사용하기 보다는 HSMS 실행의 재사용 승인을 통한 실행으로 간소화 할 수 있다. PType 이나 SType 에 대한 “Reserved, not used” 범위 내에 유지됨으로써, 실행자는 SType 이나 PType 의 새로운 값을 정의하는 미래의 하위 표준이 사용자 정의의 확장과 충돌하지 않을 것이라는 것을 확신할 수 있다.

A1-11 SECS-1 과 HSMS 의 비교 (Comparison of SECS-I and HSMS)

A1-11.1 아래의 테이블은 SECS-1 과 HSMS 의 주요 기능을 비교한다.

Table A1-7 SECS-1 과 HSMS 의 비교 (Comparison of SECS-I and HSMS)

<i>Feature</i>	<i>SECS-I</i>	<i>HSMS</i>
통신 프로토콜 기반 (Communications Protocol Base)	RS-232	TCP/IP
물리적 레이어 (Physical Layer)	25 핀 커넥터와 4-wire 시리얼(serial) 케이블	물리적 레이어(layer)가 정의되지 않는다. HSMS 는 물리적 개체가 지원되는 TCP/IP 를 허용한다. 일반적인 예는 Ethernet (IEEE 802.3)과 thin coax (10-BASE-2)이다.
통신 속도 (Communications Speed)	일반적으로 대략 1000 bytes/second(대략 9600 baud).	일반적으로 10 MBits/second (대략 일반적인 Ethernet).
연결 (Connections)	SECS-I 연결 당 하나의 물리적 RS-232 케이블	하나의 물리적인 네트워크 케이블이 대다수의 HSMS 연결(HSMS Connection)을 지원 할 수 있다.



<i>Feature</i>	<i>SECS-I</i>	<i>HSMS</i>
메시지 형식 (Message Format)	<p>메시지 텍스트는 SECS_II 데이터 항목(SECS_II Data Item)이다.</p> <p>대략 256 바이트 크기의 전송 블록의 시리즈로 SECS-II 메시지를 전송한다. 각각의 블록은 1-byte 의 블록 길이와, 10-byte 의 블록 헤더(Block Header), 텍스트 및, 2-byte 의 Checksum 을 가진다.</p>	<p>메시지 텍스트(Message Text)는 SECS-II 데이터 아이템(SECS_II Data Item)이다.</p> <p>TCP/IP 바이트 스트림(stream)으로 SECS-II 메시지를 전송한다. 메시지는 4 바이트의 메시지 길이(Message Length)와 10 바이트의 메시지 헤더(Message Header), 텍스트를 가진다. TCP/IP 레이어(layer)는 사용되는 물리적 레이어(layer)에 따른 블록화 제한을 도입하지만, 이 블록화는 TCP/IP API 에 의거하고 HSMS 범위의 밖이다.</p>
헤더 (Header)	<p>메시지의 각 블록에 10 바이트의 헤더. 헤더(Header) 바이트 4-5 는 E-bit 와 블록 번호(Block Number)를 포함한다.</p>	<p>전체 메시지에 대한 하나의 10 바이트의 헤더(Header). 헤더(Header) 바이트 4-5 는 PType 과 SType 을 포함한다. SType= 0 일 경우(Data Message) 헤더 바이트 2-3 은 W-Bit, 스트림(Stream) 및 기능(Function)이다. SType 이 0 이 아닐 경우(Control Message), 바이트 2-3 은 다른 용도를 가지며, R-Bit 는 아니다.</p>
최대 메시지 사이즈	<p>대략 7.9 백만 바이트로 제한 (블록당 244 텍스트 바이트를 32767 블록에 곱한).</p>	<p>메시지 사이즈는 4-바이트 메시지 길이로 제한된다 (대략 4 GBytes). TCP/IP 와 HSMS 의 로컬 실행은 실제로 이것을 제한한다.</p>
프로토콜 매개변수 (공통) Protocol Parameters (Common)	T3 Reply Timeout Device ID.	T3 Reply Timeout SessionID (Device ID 와 유사).
프로토콜 매개변수 (SECS-I 단독) (Protocol Parameters (SECS-I only))	Baud Rate T1 Inter-Character Timeout T2 Block Protocol Timeout T4 Inter-Block Timeout RTY Retry Count Host/Equipment	HSMS 에서 사용되지 않음. TCP/IP 레이어(layer)로 해당 이슈들을 해결한다.
프로토콜 매개변수 (HSMS 단독) (Protocol Parameters (HSMS Only))	SECS-I 에서 요구되지 않는다.	IP Address and Port of Passive Entity. T5 Connect Separation Timeout. T6 Control Transaction Timeout. T7 NOT SELECTED Timeout. T8 Network Intercharacter Timeout.



NOTICE: Semiconductor Equipment and Materials International (SEMI) makes no warranties or representations as to the suitability of the Standards and Safety Guidelines set forth herein for any particular application. The determination of the suitability of the Standard or Safety Guideline 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 or equipment mentioned herein. Standards and Safety Guidelines are subject to change without notice.

By publication of this Standard or Safety Guideline, SEMI takes no position respecting the validity of any patent rights or copyrights asserted in connection with any items mentioned in this Standard or Safety Guideline. Users of this Standard or Safety Guideline 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.

SEMI 는 특정 어플리케이션에 대해 계약에 명시된 표준(들)의 적합성에 대해 아무런 보증이나 진술도 하지 않는다. 표준의 적합성의 결정은 전적으로 사용자의 책임이며, 사용자는 여기에 언급된 제조업체의 지시사항, 제품 라벨, 제품 데이터 시트 및 재료나 장비가 기타 관련 문헌을 참조할 때 주의 해야 한다. 이러한 기준은 예고 없이 변경될 수 있다.

이 표준의 출판으로, 반도체 장비 및 재료 국제 (SEMI)는 모든 특허 권리나 저작권의 유효성을 존중하여 언급된 모든 항목과 관련하여 제기하지 않는다. 이 표준의 사용자가 명시적으로 모든 채널 특허권이나 저작권 및 권리의 침해의 위험한 결정은 전적으로 자신의 책임임을 주지해야 한다.