

반도체및디스플레이장비 통신프로토콜 구현에 관한 연구

김두용*

*순천향대학교 정보기술공학부

초록

본 논문은 HSMS를 기반으로 하는 SECS 통신 프로토콜을 구현한다. HSMS는 TCP/IP를 이용한 이더넷 기반이기 때문에 많은 장점을 가지고 있는 윈도우 소켓을 사용한다. 윈도우 소켓은 높은 호환성을 가지며, 다양한 종류의 통신 규약을 지원한다. 윈도우소켓에서 제공하는 API 함수를 이용하여 쓰레드를 동기화 하고, 이벤트 기법을 사용하여 클라이언트 측과 서버 측의 독립적인 송수신을 가능하게 한다. 또한, 하나의 프로그램에서 서버 측과 클라이언트 측을 선택적으로 사용 가능하게 구현한다.

1. 서론

반도체및디스플레이 산업은 장치 집약적이기 때문에 제조 공정에 많은 장비들이 사용된다. 오늘날 반도체및디스플레이 산업의 발전으로 다양한 종류의 장비 수요가 증가되었으며 이로 인하여 반도체및디스플레이 장비는 점점 복잡해지고, 관리해야 할 데이터의 양은 기하급수적으로 증가하였다. 각 업체는 효율적인 장비관리의 측면에서 점차 공장자동화에 관심을 갖게 되었으며, 또한 장비 간 정보 교류 필요성을 인식하게 되었다. 통계적인 프로세스 제어 및 작업자 교육의 꾸준한 시도에도 불구하고 생산량 및 장비 활용적인 면이 일정 기준치 이상 증가하지 못하는 문제점이 대두되었다. 이에 따라 반도체및디스플레이 업계에서는 네트워크장비를 반도체및디스플레이 장비에 적용하게 되었다. 그러나 다른 통신 규약을 가진 장비들 간 통합의 어려움이 있어서 서로 다른 공급자에 의해 납품된 장비들 사이에 호환성이 없었다. 이러한 이유로 SEMI(Semiconductor Equipment and Materials International)의 장비자동화 부문(Equipment Automation Division)에서 반도체 장비와 외부 컴퓨터간의 인터페이스를 위한 데이터 통신 표준 규약의 필요성을 인식하고, SECS(SEMI Equipment Communication Standard)를 만들어 발표하였다[1]. SECS 통신 사양은 현재 대부분의 반도체 장비와 LCD 장비에 적용되는 통신 사양으로 해당 장비에 대한 표준 사양의 지원이 점차 의무화되고 있으며 SECS는 이러한 업계의 요구에 부합하는 표준으로 자리 잡게 되었다. SECS 프로토콜은 SECS-I(SEMI Equipment Communication Standard1 Message Transfer) 과 SECS-I를 기반으로 상위 계층의 통신 규약인 SECS-II(SEMI Equipment Communication Standard2 Message Content), GEM(Generic Model for Communications and Control of Manufacturing Equipment)를 개발하여 표준 규약으로 자리 잡게 되었다[2][3]. 그러나 SECS-I는 제약이 많고 속도가 느린 RS-232 기반으로 개발되었기 때문에 빠른 속도를 원하는 현대의 장비산업에서는 새로운 통신 규약이 필요로 하게 되어 HSMS(High-Speed SECS Message Services)라는 TCP/IP 이더넷에 기반을 둔 통신 규약이 개

발되었다[4]. HSMS는 SECS-I에 비해 속도가 빠르고 RS232C가 갖는 느린 속도와 케이블 길이의 제한성 등 많은 단점을 보완할 수 있으며 확장이 용이하고 길이의 제약이 없다. 본 논문은 반도체 장비 및 LCD 장비에서 적용되는 장비 간 네트워크 통신 규약인 SECS를 구현하기 위해 SECS의 하위 통신 규약인 HSMS에 기반을 두고 상위 통신 규약인 SECS-II, GEM를 구현하여 반도체및디스플레이 장비간의 통신 시스템을 구현 한다. 또한 윈도우 시스템에서 TCP/IP 소켓 프로그램을 기반으로 하나의 프로그램에서 서버 측과 클라이언트 측을 선택적으로 사용 가능하게 구현한다.

2. SECS 프로토콜 구성 및 분석

SECS 구성은 그림 2.1과 같다. 장비와 호스트 간에 통신을 담당하고 설정하는 하위 통신 규약인 SECS-I 및 HSMS로 구성되고, 하위 통신 규약에서 교환되는 메시지를 해석하여 의미를 규정하는 SECS-II와 SECS-II 메시지를 받을 시 장비들이 취해야 하는 절차를 규정한 GEM으로 구성되어 있다.

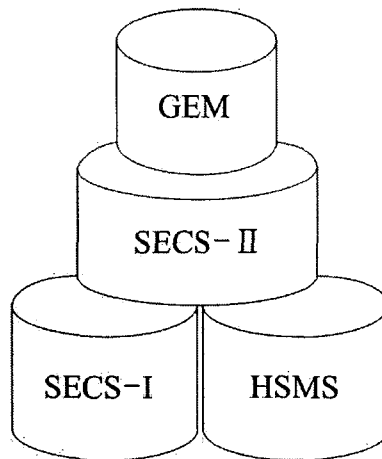


그림 2.1 SECS 프로토콜 구성도

2.1 SECS-I 메시지 구성

SECS-I는 RS-232 케이블 통신을 이용한 메시지 전송 방식으로 반도체 장비와 호스트 사이의 적절한 메시지 교환을 위해 통신 인터페이스를 정의한다. 그림 2.2는 SECS-I 메시지의 블록 구성을 나타낸다. SECS-I는 다음과 같은 특성을 갖는다[5][6].

- 비동기식 양방향 통신으로 하나의 시작 비트와 종료 비트를 갖는 10 비트 직렬 통신이다.
- 데이터를 254 바이트 크기의 블록단위로 전송한다.
- 각 블록은 10 바이트 크기의 헤더를 가지며, 헤더부분에 블록의 정보가 있다.
- 하나의 완전한 전송 단위를 메시지라 하며 1~32,767 블록으로 구성된다.
- 메시지는 요청과 응답의 한 쌍으로 구성되며 이들을 하나의 트랜잭션이라 부른다.

A	B	C	D
---	---	---	---

- A :블록의 길이 (1 바이트) = (B+C)길이
- B :헤더 (10 바이트)
- C :데이터 (0~244 바이트)
- D :책섬(2 바이트)

그림 2.2 블록 구성도

2.2 HSMS

HSMS는 SECS-I의 단점을 보완하기 위해 나온 TCP/IP기반 SECS 통신이며, 장비와 호스트 간에 통신 연결 순서를 상태 별로 정리를 하여 나타낸 것이 그림 2.3 이다.

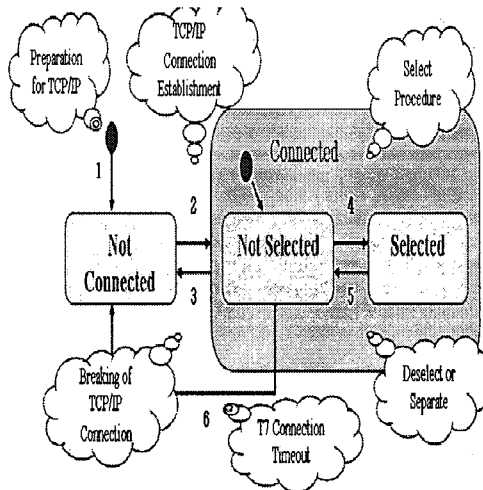


그림 2.3 HSMS 연결상태 다이어그램

HSMS 메시지는 호스트와 장비 간에 통신 설정을 완료하는 Select, 데이터를 송수신하는 Data, 양방향합의에 의한 통신 종료인 Deselect, 통신연결을 확인하는 Link Test, 일방적 통신 종료를 하는 Separate, 적절하지 않는 메시지를 받았을 때 처리하는 Reject 상태 등으로 나누어 동작한다.

2.3 SECS-II

SECS-II는 장비와 호스트 간에 메시지 전송 규약에 따라 교환되는 메시지가 해석될 수 있도록 그 구조 및 의미를 규정한다. 이 표준에서 정의된 메시지는 일반적인 반도체 제조에 필요한 대부분의 내용을 포함하고 있으며, 정의 되어 있지 않은 메시지는 장비에 따라서 필요한 메시지를 정의해서 사용할 수 있도록 허용한다.

SECS-II 데이터는 아이템과 리스트로 구성된다. 헤더에서 정의된 스트림(stream)번호 및 함수

(function)번호에 따라서 데이터의 아이템과 리스트 구성방법, 형식이 정의된다. 사용자는 사전에 스트림번호 및 함수번호에 따른 리스트와 아이템으로 구성된 사전을 이용하여 데이터 값을 입출력 한다.

2.4 GEM

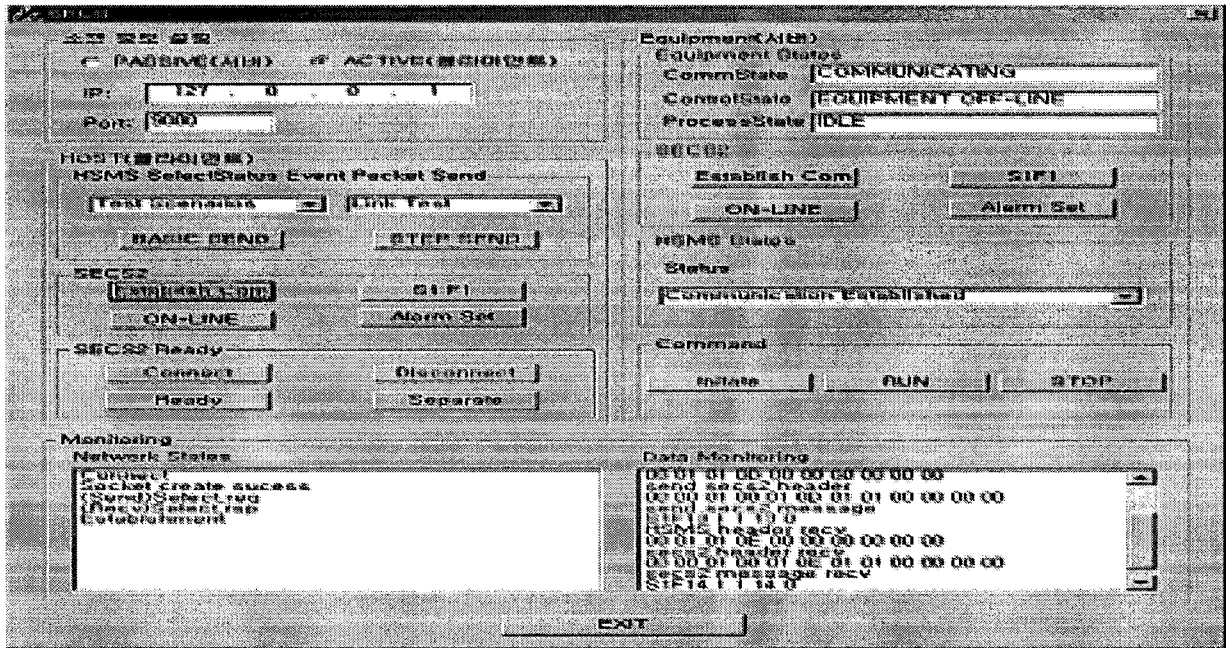
GEM은 SECS-II 메시지를 받을 때 마다 장비들이 취해야 하는 행동 양식과 절차이다. GEM은 메시지에 대해 상응하는 동작을 구현하는데 사전에 정의된 일련의 시나리오에 따라 프로그램이 동작하도록 만든 것이다. 필수 기본 사양은 반드시 제공되어야 하고 선택 사양은 필요한 기능을 선택적으로 만들어 첨가한다. 다음은 필수 기본 사양들의 내용이다.

- 장비의 실행 상태 모델
- 호스트에서 초기화 설정
- 이벤트 통지
- 온라인상태 확인
- 에러 메시지 처리
- 작업자에 의한 제어

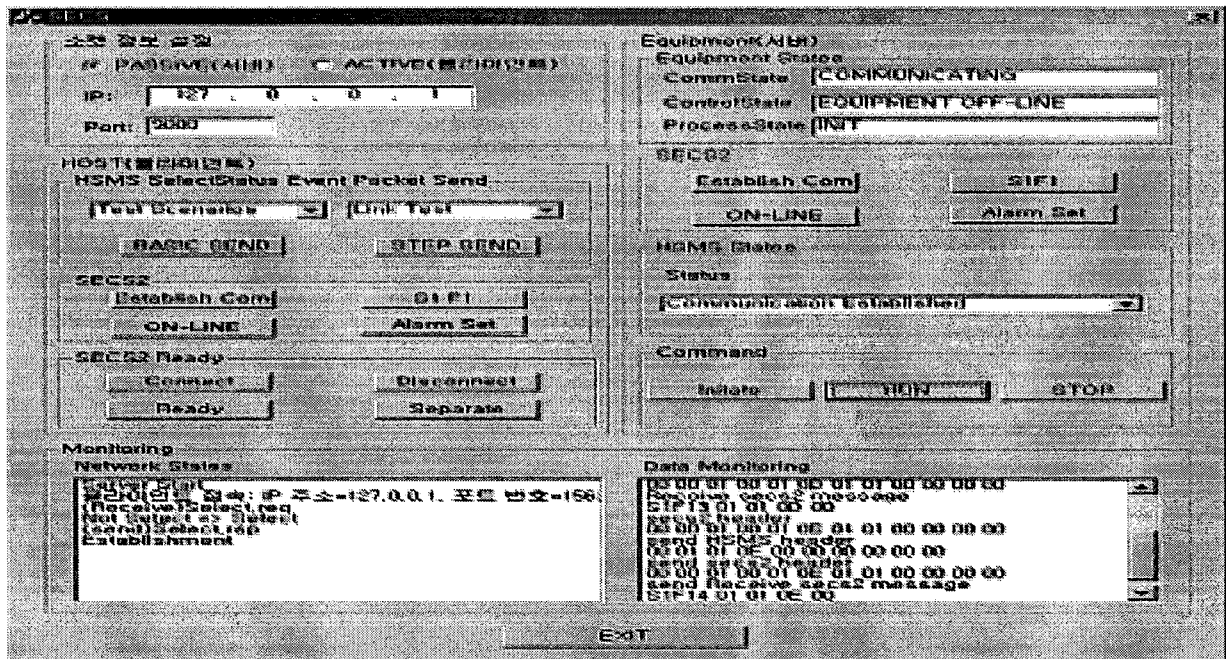
3. SECS 프로토콜 구현

본 논문은 SECS 통신 규약을 능동 모드와 수동 모드로 나누어 구현 하였다. 능동 모드는 네트워크 통신상에서 클라이언트의 기능을 하고 수동 모드는 서버 기능을 한다. 능동 모드는 주로 호스트에서 수동 모드는 장비에서 사용한다. 수동 모드는 서버 소켓으로 구현하였다. 소켓을 생성한 후 클라이언트가 접속하기를 대기하며, 클라이언트가 접속하면 새로운 소켓을 생성한다. 능동 모드는 클라이언트 기능을 가지고 있기 때문에 클라이언트 소켓으로 구현하였다. 그러나 일반적인 클라이언트 소켓 기능으로 장비 간의 정상적인 통신을 할 수 없다. 장비 간의 통신에서는 각 장비와 호스트 마다 독립적으로 데이터를 수신하거나 송신 할 수 있어야 하지만 일반적인 클라이언트 경우에는 독립적으로 메시지 수신 대기를 하지 못하기 때문이다. 이 문제를 해결하기 위해서 본 논문에서는 윈도우에서 지원하는 WaitForMultipleObject 함수와 쓰레드기법 및 이벤트 기법을 사용하였다. 쓰레드 기법은 서버 쓰레드, 클라이언트 쓰레드를 사용하였고 이벤트 기법에는 읽기 이벤트, 쓰기 이벤트를 사용하였다. WaitForMultipleObject 함수는 두 개 쓰레드가 동시에 진행되어서는 안되는 상황이 있을 때 매개체 역할을 한다. 이것을 동기화 객체라고 하고 동기화 객체를 사용하여 쓰레드 진행 가능을 판단하고 이에 근거하여 쓰레드가 계속 진행되거나 대기한다. 서버를 실행시키면 이벤트를 생성하고 각 쓰레드를 생성시킨다. 이후 서버 쓰레드에서 소켓이 생성되고 읽기 쓰레드와 쓰기 쓰레드를 생성한다. 클라이언트가 접속을 하면 클라이언트 쓰레드 및 읽기 쓰레드, 쓰기 쓰레드를 생성한다[7][8][9]. 본 논문에서는 GEM의 필수 기본 사양들과 선택 사양 중에 장비 측 초기화 설정, 호스트에 의한 제어, 알람 발생 및 처리 등을 구현 하였으며 다음 그림 3.1은 구현된 SECS 프로그램으로써 장비와 호스트 간의 데이터를 송수신하기 위한 최종 통신 상태를 만들기 위해 호스트에서 SECS-II 데이터에 S1F13을 실어 송신하는 것을 보여준다. 장비에서는 S1F13을 수신 한 후 응답 메시지로 S1F14를 송신한다. 이와 동시에 장비에서는 사용자 이벤트를 발생시켜 통신 상태를 Communicating으로 만들고, 반도체 제조 공정 작업의 상태를 나타내는 프로세싱 상태와 제어 상태를 초기화 한다. 그림 3.1은 호스트에서

초기화 설정을 한 결과를 나타낸다. SECS Establish Com 버튼을 눌러 실행시키면 HSMS 헤더에 스트림 값 1과 함수 값 13이 각각 설정되어 송신된다. 장비 측에서는 수신된 메시지의 스트림과 함수를 판별하여 S1F14를 송신한다. 호스트 측의 Data Monitoring창을 살펴보면 이를 확인 할 수 있다. 호스트는 S1F13을 송신한 후 이벤트를 생성하고 이로 인하여 State 상자의 각 상태 값들이 초기화되어 출력되는 것을 볼 수 있다. 또한, Network States 창에 Equipment 메시지가 송신된 것을 알 수 있다.



(a) 호스트에서 초기화한 호스트 측 결과



(b) 호스트에서 초기화한 장비 측 결과

그림 3.1 호스트에서 초기화한 구동 결과

4. 결론

서로 다른 통신 규약을 가진 장비들 간에 통신 규약 통합의 필요성이 대두되었고 이로 인하여 SEMI에서 반도체 장비와 외부 컴퓨터 간의 인터페이스를 위한 SECS를 표준 규약으로 발표하였다. 초기에는 RS-232 기반의 SECS-I를 사용 하였지만 낮은 속도와 케이블 길이의 제약성으로 SECS-I의 대체 통신 규약으로써 HSMS라는 이더넷 기반의 통신규약이 개발되었다. 본 논문은 HSMS를 기반으로 하는 SECS 구동 드라이버를 구현하였다. HSMS가 TCP/IP를 이용한 이더넷 기반이기 때문에 높은 호환성과 여러 종류의 통신 규약을 지원하는 윈도우 소켓을 사용하여 구현하였다. WaitForMultipleObject 함수를 이용하여 쓰레드 간의 동기화를 하였고, 이벤트기법을 WaitForMultipleObject 함수와 함께 사용하여 클라이언트 측과 서버 측의 독립적인 송수신이 가능하게 하였다. 또한 하나의 드라이버에서 서버 측과 클라이언트 측을 선택적으로 사용 가능하게 구현하였다. 향후 안정적이고 빠른 속도를 가지는 입출력 완료 포트라 부르는 고성능의 서버를 이용한 SECS 통신 프로토콜을 구현하여 SECS의 성능을 향상시키고, GEM의 다양한 선택적 사양을 폭넓게 이용할 수 있는 프로토콜을 구현할 계획이다.

후기

본 연구는 산업자원부 지방혁신 사업(RTI04-01-02) 지원으로 수행되었음

참고문헌

- [1] SEMI EQUIPMENT COMMUNICATION STANDARD 1 MESSAGE TRANSFER(SECS-I), SEMI E4-0699, 1999
- [2] SEMI EQUIPMENT COMMUNICATION STANDARD 2 MESSAGE CONTENT(SECS-II), SEMI E5-1102, 2002
- [3] GENERIC MODEL FOR COMMUNICATIONS AND CONTROL OF MANUFACTURING EQUIPMENT, SEMI E30-1000, 2000
- [4] HIGH-SPEED SECS MESSAGE SERVICE(HSMS) GENERIC SERVICES, SEMI E37-0702, 2002
- [5] E. J. Wood, "An object-oriented SECS programming environment," IEEE Trans. on Semiconductor Manufacturing, vol. 6, no. 2, pp.119-127, May. 1993.
- [6] E. J. Wood, "Object-oriented SECS programming system," IEEE/SEMI International Semiconductor Manufacturing Science Symposium, 1992., pp.130-136, June. 1992.
- [7] 김선우, "윈도우 네트워크 프로그래밍," 한빛미디어, August. 2004.
- [8] 윤성우, "TCP/IP 소켓 프로그래밍," FREELEC, March. 2004.
- [9] 박준철, "TCP/IP 소켓 프로그래밍," 사이텍미디어, September. 2004.