

Wet Station 장비를 제어하기 위한 모니터링 시스템의 설계

임 성 략[†] · 한 광 록[†] · 최 용 업^{††}

요 약

본 논문에서는 웨이퍼를 세정하는데 사용되는 Wet Station 장비의 상태를 감지하고, 이를 간접적으로 제어하기 위한 모니터링 시스템의 설계에 관하여 기술한다. 대부분의 기존 모니터링 시스템은 하드웨어 및 소프트웨어에 의존되어 있다. 제한된 모니터링 시스템의 기본 설계 목표는 사용자의 편의성과 시스템의 이식성을 제공하는 것이다. 이러한 요구조건을 만족시키기 위하여, 본 논문에서는 IBM PC 호환 기종의 윈도우즈 NT 환경에서 GUI 기능과, TCP/IP 통신 프로토콜을 기본으로 한 EtherNet 보드를 이용하여 시스템을 설계하였다.

Design of a Monitoring System for Controlling the Wet Station Equipment

Seong-Rak Rim[†] · Kwang-Rok Han[†] · Yong-Yub Choi^{††}

ABSTRACT

This paper describes the design of the monitoring system for monitoring the current status and indirect controlling of the Wet Station Equipment which is used for cleaning the wafer. Most of the conventional monitoring system depend on the special hardware and software. Basic design goal of monitoring system is to provide the convenience for the user and the portability for the system. In order for the system to fulfil its requirements, it was designed using GUI (Graphical User Interface) facility based on the Windows NT environment of IBM PC compatible and EtherNet board based on the TCP/IP protocol.

1. 서 론

Wet Station은 반도체 제조 공정에서 웨이퍼의 세정을 위한 장비 중의 하나로서 주로 PLC(Programmable Logic Controller)에 의해 직접적으로 제어된다. 따라서 장비에서 처리되고 있는 상태를 사용자가 감시하

고 이를 간접 제어하기 위한 모니터링 시스템이 절대적으로 요구된다. 기존의 모니터링 시스템은 장비의 동작 상태를 실시간으로 감지하기 위하여 제조 업체 고유의 전용 하드웨어 및 운영체제를 사용하였으며, 화면 처리는 텍스트 중심으로 구현되었기 때문에 시스템의 범용성과 사용자의 편의성에 어려움이 있다. 예를 들어, 일본 DNS 업체에서 개발된 Wet Station 모니터링 시스템은 DNS가 자체 개발한 통신용 보드[1]와 운영체제(MTM 커널)를 Mitsubishi PC 기종에서 운용하고 있기 때문에 시스템의 하드웨어 및 소프트웨

* 본 연구는 반도체 제조 장비 국산화 연구센터의 지원에 의하여 이루어졌음.

† 종신회원 : 호서대학교 컴퓨터학부 교수

†† 정 회 원 : 호서대학교 컴퓨터학부 교수

논문접수 : 1998년 12월 3일, 심사완료 : 1999년 2월 26일

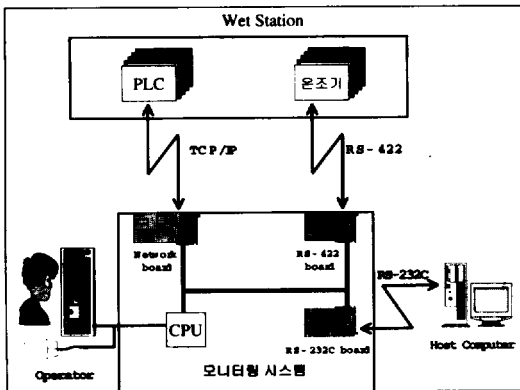
어의 범용성이 전혀 없으며, 국산화를 위한 기술력 확보가 매우 어렵다.

본 논문에서는 이러한 어려움을 해결하기 위하여 IBM PC 호환 기종에서 EtherNet 보드를 이용한 TCP/IP 통신 프로토콜과 윈도우 NT 운영체제의 GUI 기능을 이용함으로써 시스템의 범용성과 사용자의 편의성을 제공할 수 있는 Wet Station 모니터링 시스템을 제시하고자 한다.

2. 시스템 설계

2.1 개발 환경

본 논문에서는 (그림 1)과 같은 개발 환경에 적합한 모니터링 시스템을 설계하고자 한다.



(그림 1) 개발 환경

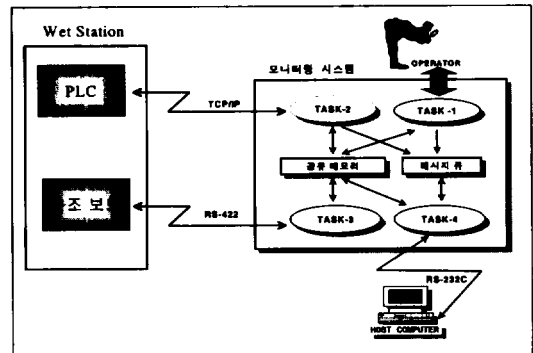
(그림 1)에서 Wet Station 장비는 로봇트 및 주변장치들을 직접 제어하기 위한 다수의 PLC와 Bath의 온도를 제어하기 위한 온조기를 가지고 있다. 따라서 본 논문에서 제시하고자 하는 모니터링 시스템은 다음과 같은 외부 인터페이스 기능이 요구된다.

- ① Wet Station을 직접 제어하는 PLC와 인터페이스
- ② Bath의 온도 제어를 위한 온조기와 인터페이스
- ③ Wet Station의 동작 상태를 감지하고 간접 제어를 위한 사용자 인터페이스
- ④ 근거리 호스트 시스템과 인터페이스

이러한 외부와의 인터페이스를 효과적으로 지원하기 위하여 각각의 인터페이스를 담당하는 모듈을 하나의 독립된 태스크(태스크)로 설계한다. Wet Station을

직접 제어하는 PLC와 인터페이스는 EtherNet 보드를 사용한 TCP/IP 프로토콜을, Bath의 온도 제어를 위한 온조기와 인터페이스는 RS-422 보드를 사용한 RS-422 프로토콜을, 근거리 호스트 시스템과 인터페이스는 COM 포트를 사용한 RS-232C 프로토콜을, 그리고 Wet Station의 동작 상태를 감지하고 간접 제어를 위한 사용자 인터페이스는 윈도우 NT의 GUI 기능을 이용하도록 설계한다. 또한 독립적인 태스크들 간의 통신은 공유 메모리와 메시지 큐를 사용하고, 태스크마다 다수의 스래드를 사용함으로써 고유의 기능을 효율적으로 수행하도록 한다.

결국 본 연구에서 제시하는 모니터링 시스템의 개략적인 구조는 (그림 2)와 같이 4개의 독립된 태스크로 구성된 멀티 태스킹/멀티 스래드 시스템이다.



(그림 2) 모니터링 시스템의 구조

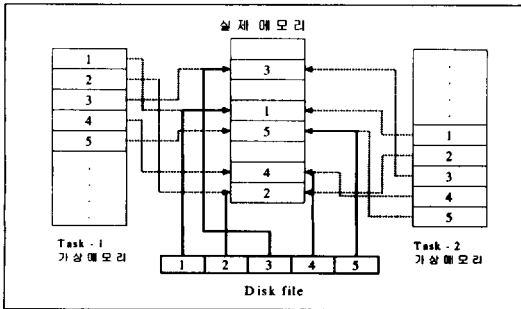
2.2 사용자 인터페이스 모듈(태스크-1)

사용자 인터페이스 모듈은 Wet Station 모니터링 시스템의 초기화 및 화면 제어를 담당하는 메인 태스크로서 다음과 같은 기능을 수행한다.

- ① 시스템 전체에서 사용되는 전역변수 선언 및 초기화
- ② 태스크간 통신을 위한 공유 메모리 확보 및 메시지 큐 생성
- ③ 새로운 태스크 생성
- ④ 공유 메모리에 파라미터 값 설정
- ⑤ Wet Station 설비의 기동 및 동작 상태를 점검하여 화면에 출력
- ⑥ 사용자 명령어를 받아들이기 위한 초기화면 출력 (무한루프)

시스템 전체에서 사용하는 전역변수의 초기 값은 Wet Station에 따라 다르며, 각각에 대한 초기 값과 마지막 실행환경에 대한 데이터를 독립된 파일로 보관하게 된다. 따라서 시스템 기동시 이러한 정보를 파일로부터 읽어와서 전역변수들의 값을 초기화하도록 한다.

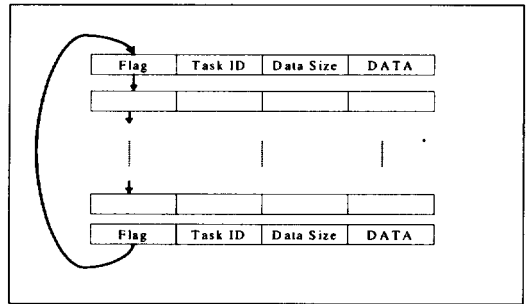
태스크간의 통신을 위한 공유 메모리와 메시지 큐는 그 용도에 따라 다르게 사용된다. 즉, 공유 메모리는 태스크간 신속한 통신을 요구하는 환경에 적용하고, 메시지 큐는 호스트로부터 수신된 메시지처럼 도착된 순서를 유지할 필요가 있을 경우에 사용한다. 윈도우즈 NT에서 제공하는 여러 가지 공유 메모리 기법 중에서 Memory-mapped File 방식을 채택하여 설계한다 [2]. Memory-mapped File 방식은 (그림 3)과 같이 디스크상의 파일의 블록을 태스크의 가상 메모리에 매핑함으로써 디스크상의 파일에 대한 읽기/쓰기가 주기억 장치에서 이루어진다. (그림 3)에서 태스크-1과 태스크-2는 동일한 파일을 각각의 가상 메모리에 매핑 시킴으로써 데이터를 공유하게 된다.



(그림 3) Memory-mapped File 방식에 의한 공유 메모리 설정

공유 메모리의 데이터를 여러 개의 태스크들이 접근할 경우 임계영역(Critical Section) 문제가 예상된다. 이에 대한 대책으로써 MFC(Microsoft Foundation Class)의 C이벤트 객체(object)를 이용하여 각각의 공유 메모리마다 하나의 C이벤트 객체를 생성하도록 한다[3]. 그리고 동기화를 위한 이 C이벤트 객체는 "공유 메모리+이벤트"로 명명한다.

호스트로부터 수신된 메시지를 순차적으로 처리하기 위하여 FIFO(First-In First-Out) 방식의 메시지 큐가 요구된다. 메시지 큐는 공유 메모리 상에 (그림 4)와 같이 N개의 환형 구조로 설계한다.



(그림 4) 메시지 큐의 구조

(그림 4)에서 각 레코드의 Access Flag 필드는 읽기/쓰기를 나타내며, 태스크 ID 필드에 표시된 태스크가 읽을 것인지 아니면 기록한 것인지를 의미한다. 그리고 데이터의 크기가 가변적이기 때문에 데이터의 크기를 표시하기 위한 데이터 Size 필드를 실제 데이터가 들어 있는 데이터 필드와 구분하고, 데이터 필드의 크기는 실제 사용되는 데이터 중 최대 크기로 설정한다.

사용자 인터페이스 모듈은 Wet Station 모니터링 시스템의 메인 태스크로써 다른 인터페이스를 담당하는 태스크들은 생성하여야 한다. 태스크 생성은 MFC API(Application Program Interface)의 CreateProcess() 함수를 이용하도록 한다[3,4].

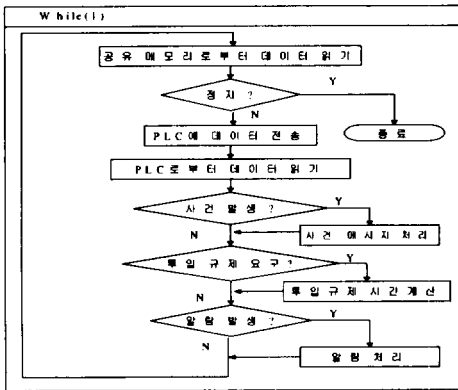
마지막으로 사용자 인터페이스를 위한 화면 제어 기능은 로봇마다 처리 조건을 설정하기 위한 Recipe 데이터 편집, 현재 로봇의 위치, 설비의 상태 출력 등 너무도 다양하다. 따라서 초기 화면 및 세부 화면은 기본적으로 사용자의 요구에 따라 가장 편리한 GUI 방식으로 설계하도록 한다[5].

2.3 PC-PLC 통신 모듈(태스크-2)

Wet Station 장비의 로봇 및 다수의 주변 장치들의 제어는 PLC에 의해 이루어지고 있다. 따라서 장비의 동작 상태를 모니터링하고 간접 제어하기 위하여 PLC와의 통신 모듈이 요구된다. 모니터링 시스템의 PLC 통신 모듈에서 제공해야할 주요 기능과 전체적인 흐름도는 (그림 5)와 같다.

- ① 공유 메모리로부터 데이터를 읽어 Wet Station 장비의 PLC에게 전송
- ② Wet Station 장비의 PLC로부터 데이터 읽어 공유 메모리에 쓰기

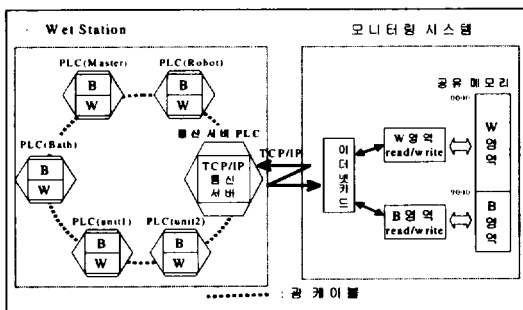
- ③ 이벤트 발생시 이벤트 메시지 생성 및 메시지 큐에 등록
- ④ Alarm 발생시 Alarm 처리
- ⑤ 투입규제 시간 요구시 투입규제 시간 계산



(그림 5) PC-PLC간 통신 모듈의 흐름도

이러한 기능을 지원하기 위하여 EtherNet 보드를 이용한 TCP/IP 프로토콜을 기반으로 PC-PLC간 통신 모듈을 설계한다. Wet Station 장비에는 로봇, Bath, 그리고 주변장치들을 직접적으로 제어하는 여러 개의 PLC가 존재한다. 이들 PLC간의 통신은 “Link Register” 라는 공유 메모리를 통하여 이루어지며, “Master” PLC에 의해 제어된다[1].

PC-PLC간 통신 모듈 설계를 위하여 통신 방식과 메시지 형식을 정의하여야 한다. 일반적으로 PC-PLC간 통신 방식에는 Master/Slave와 Peer-to-Peer 방식이 있다[6]. 전자는 PLC와 PC를 각각 서버와 클라이언트로 운용하며, 후자는 PLC와 PC가 각 서버 혹은 클라이언트로 운용된다. 본 논문에서는 (그림 6)과 같이



(그림 6) PC-PLC간 통신 구성도

Wet Station 장비에 PC와의 통신을 전담하는 TCP/IP 통신 서버용 PLC를 별도로 구성하는 Master/Slave 방식으로 설계한다.

메시지 형식은 PC에서 PLC로 보내는 송신 메시지와 PLC로부터 PC로 수신하는 메시지를 구별하여 (그림 7)과 같이 정의한다.

Sub Header	PC #	Watch dog time	Device #	Device Point	데이터
------------	------	----------------	----------	--------------	-----

Sub Header : PLC 명령어 코드
 PC # : 동작 수행 지역의 PLC 국번(자국 : 00h)
 Watch Dog Time : CPU의 응답시간
 Device # : Device 이름과 오프셋
 Device Point : 데이터의 크기
 데이터 : 실제 메시지 내용

(a) 송신 메시지 형식

Sub Header	Complete Code	데이터
------------	---------------	-----

Sub Header : PLC 명령어 코드 | 0X80H
 Complete Code : 송신 헤더의 오프셋
 데이터 : 실제 메시지 내용

(b) 수신 메시지 형식

(그림 7) 송수신 메시지의 형식

(그림 7)과 같은 송수신 메시지 형식을 이용하여 PC-PLC간 메시지 송수신 과정은 다음과 같이 이루어지도록 한다.

- ① 모니터링 시스템에서 송신 메시지 형식으로 작성하여 PLC로 전송한다.
- ② PLC에서 메시지의 헤더를 분석하여 “Link Register”에 읽기/쓰기를 결정한다.
- ③ 분석된 명령어에 따라 “Link Register”에 읽기/쓰기를 수행한다.
- ④ 응답 메시지를 수신 메시지 형식으로 작성하여 모니터링 시스템으로 전송한다.

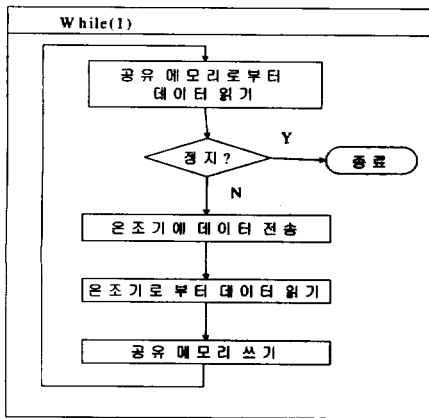
PC-PLC간 통신 모듈은 위와 같은 기본적인 기능 외에도 호스트로부터 장비의 상태를 감지하기 위한 사건 메시지에 대한 응답으로써 SECS 태스크로 하여금 호스트에게 알려줄 수 있도록 사건 메시지를 작성하여 메시지 큐에 전송하여야 한다. 또한 장비로부터 신규 로트의 투입 규제 시간에 대한 요구가 있을 경우 최적

의 투입 규제 시간을 계산하여 알려주어야 한다. 장비가 동작하는 중에 오류가 발생하여 알람이 발생하게 되면 사용자 인터페이스 모듈에 출력할 수 있도록 공유 메모리에 적재하여야 한다. 이러한 일련의 기능은 모두가 Wet Station으로부터 데이터를 수집하여야 하기 때문에 PC-PLC간 통신 모듈에서 제공하도록 한다.

2.4 온도 감지 모듈(태스크-3)

Wet Station 장비 내에는 웨이퍼의 약 60여종의 불순물을 제거하기 위한 여러 개의 화학 조(chemical bath)가 있다. 루트 단위로 투입된 웨이퍼를 각각의 화학조에 들어있는 세정 액에 일정 시간 동안 담금(dip)으로써 웨이퍼에 부착된 불순물들은 용해된다. 이때 각각의조에 들어있는 화학 용액의 온도를 일정하게 유지하기 위하여 온도조(temperature controller)가 있다. 현재 각 조의 온도를 감지하고 온도조와의 인터페이스를 담당하는 태스크가 요구된다. 모니터링 시스템의 온도 감지 모듈에서 기본적으로 제공해야하는 기능과 전체적인 흐름도는 (그림 8)과 같다.

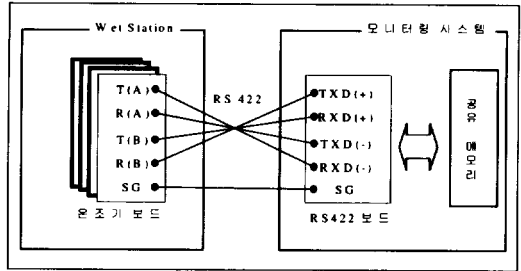
- ① 공유 메모리로부터 데이터를 읽어 Wet Station 장비의 온도조에 전송
- ② Wet Station 장비의 온도조로부터 데이터를 읽어 공유 메모리에 쓰기



(그림 8) 온도 감지 모듈의 흐름도

이러한 기능을 지원하기 위하여 RS-422 보드를 이용한 온도 감지 모듈을 설계한다. Wet Station 장비에는 여러 개의 RS-422 인터페이스 보드가 있기 때문에

1 : N의 통신 채널이 요구된다. 온도조의 RS-422 인터페이스 보드와 모니터링 시스템의 RS-422 보드간의 통신 라인 구성도는 (그림 9)와 같다.



(그림 9) 온도조와 RS-422 보드간의 통신 구성도

(그림 9)에서 온도 감지 태스크는 공유 메모리로부터 데이터를 읽어 온도조로 전송하고, 온도조로부터 데이터를 읽어 공유 메모리에 적재함으로써 다른 태스크와 통신할 수 있다. 메시지 형식은 PC에서 온도조로 송신하는 메시지와 온도조로부터 수신하는 메시지를 구별하여 (그림 10)과 같이 정의한다.

EOT	Station #	Command
-----	-----------	---------

EOT : End of Transmission
 Station # : Station Address
 Command : 처리해야할 명령어

(a) 송신 메시지 형식

EOT	Station #	STX	Command	데이터	ETX	BCC
-----	-----------	-----	---------	-----	-----	-----

STX : Start of Text
 ETX : End of Text
 데이터 : 실제 메시지 내용
 BCC : 오류 검사 코드

(b) 수신 메시지 형식

(그림 10) 송수신 메시지의 형식

(그림 10)과 같은 송수신 메시지 형식을 이용하여 PC-온조기간 메시지 송수신 과정은 다음과 같이 두 가지 경우가 있다.

첫째, 현재 온도조의 데이터를 읽을 경우 :

- ① 모니터링 시스템에서 송신 메시지 형식으로 작성하여 온도조로 전송한다.

- ② 온조기에서 메시지의 헤더를 분석한다.
- ③ 분석된 명령어에 따라 해당 "Station"의 데이터 읽기를 수행한다.
- ④ 응답 메시지를 수신 메시지 형식으로 작성하여 모니터링 시스템으로 전송한다.

둘째, 온조기의 데이터를 설정할 경우 :

- ① 모니터링 시스템에서 수신 메시지 형식으로 작성하여 온조기로 전송한다.
- ② 온조기에서 메시지의 헤더를 분석한다.
- ③ 분석된 명령어에 따라 해당 "Station"에 데이터 쓰기를 수행한다.

모니터링 시스템의 온도 감지 모듈은 위와 같은 기본적인 기능 외에도 RS-422 통신 라인을 설정하고, 초기 데이터 값을 공유 메모리 혹은 파일로부터 읽어 올 수 있어야 한다. 또한 현재 온조기의 데이터에 대한 분석과 사용자의 요구 사항을 수행하기 위한 기능을 제공하도록 한다.

2.5 호스트 통신 모듈(태스크-4)

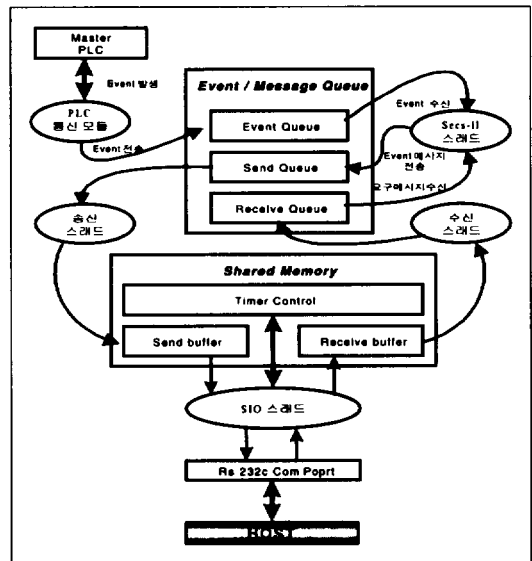
호스트 통신 모듈은 Wet Station 장비의 동작상태를 원격감시하기 위해 태스크이다. 근거리의 호스트 컴퓨터는 RS-232C 통신 채널을 이용하여 모니터링 시스템과 통신하도록 설계한다. 반도체 제조장비를 원격 제어하기 위해 사용되는 일반적인 통신 프로토콜은 SECS(Semiconductor Equipment Communications Standard)이다[7]. SECS 프로토콜은 SECS-I과 SECS-II로 구성되며 이들의 기능과 계층적 구조는 (그림 11)과 같다.

APPLIC.	사용자 응용프로그램
	변환
SECS-2	메시지 형식
	데이터 항목 형식
SECS-1	트랜잭션
	메시지
	블록
	물리계층

(그림 11) SECS 프로토콜의 계층적 구조

본 논문에서는 SECS 프로토콜의 계층적 구조를 기반으로 호스트 통신 모듈을 설계한다. (그림 11)에서 SECS-I은 호스트와의 메시지 송수신을 직접적으로 처리하는 부분이며, SECS-II는 사용자로 하여금 응용 프로그램을 통하여 메시지 송수신을 편리하게 지원해 주기 위하여 메시지의 형식을 정의하는 부분이다. 예를 들어, 호스트로부터 어떤 원격 제어 명령어를 수신할 경우, SECS-I에서는 RS-232C 포트를 통하여 도착된 메시지를 SECS-II로 하여금 가져갈 수 있도록 메시지 수신 큐에 저장한다. SECS-II는 메시지 수신 큐에 저장된 메시지를 분석하여 PLC 통신 모듈로 하여금 PLC에 전달할 수 있도록 공유 메모리에 적재한다. 반면, 이 명령어에 대한 결과를 호스트로 송신할 경우, SECS-II는 메시지를 작성하여 SECS-I으로 하여금 RS-232C 포트로 송신할 수 있도록 메시지 송신 큐에 저장한다.

이러한 기능을 처리하기 위하여 호스트와의 통신 모듈은 (그림 12) 같이 4개의 스래드로 분리하고, 스래드간의 동기화를 플래그를 사용하도록 설계한다.



(그림 12) 호스트 통신 모듈의 구조

(그림 12)에서 호스트와 모니터링 시스템의 통신 방식은 Master/Slave 관계로서 1 : N으로 연결된다. 따라서 모니터링 시스템은 고유의 식별자를 가지며, 호스트는 이 식별자를 통하여 구별할 수 있다.

SECS-II 스래드는 SECS 프로토콜 계층에서 SECS-II 부분에 해당되는 것이다. 주된 기능은 호스트로부터 수신된 메시지 수신 큐의 내용을 분석하여 Wet Station 설비를 제어할 수 있도록 공유 메모리를 통하여 PLC 통신 모듈에 전달하고, 그 결과를 호스트에게 송신하기 위한 메시지를 작성하여 메시지 송신 큐에 저장하는 기능을 수행한다. 또한 Wet Station에서 어떤 사건이 발생하면 이에 대한 분석과 결과를 호스트로 전달한다.

송신 큐에 저장된 메시지와 사건 큐에 저장된 메시지 블록의 형식을 (그림 13)과 같이 구별하여 설계하였다.

이벤트 ID	이벤트 Value	이벤트 No
--------	-----------	--------

이벤트 ID : 사건 식별자
 이벤트 Value : 사건의 현재 내용(값)
 이벤트 No : 사건 번호

(a) 사건 메시지 형식

Equipment ID	Stream	Function	Block #	System Byte	데이터
--------------	--------	----------	---------	-------------	-----

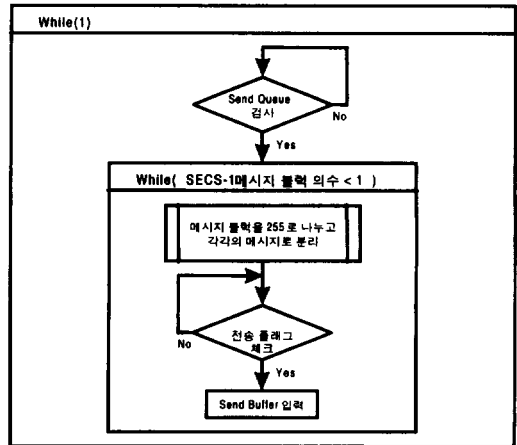
Equipment ID : 장비 식별자
 Stream : 메시지의 Stream 번호
 Function : 메시지의 Function 번호
 Block # : 메시지의 블록 번호
 System Byte : 현재까지 전송된 메시지의 개수
 데이터 : 실제 메시지 데이터

(b) SECS-II 메시지 형식

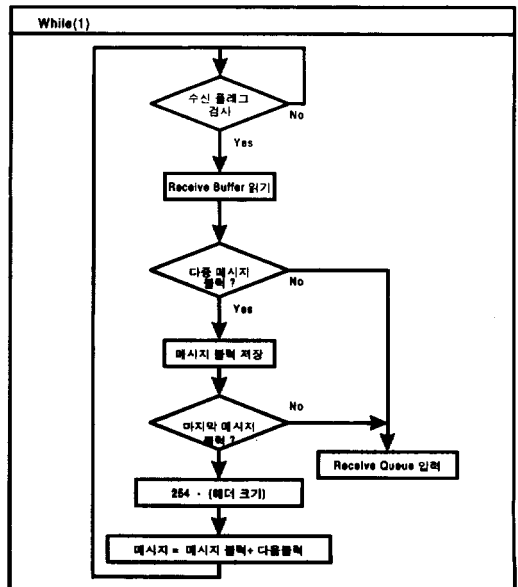
(그림 13) SECS 모듈에서 사용하는 메시지 형식

송신 스래드는 기본적으로 SECS-II에 의해 송신 큐에 저장된 메시지를 가져와서 이를 SECS-I 프로토콜에 맞도록 재구성하여 SIO 스래드로 하여금 송신할 수 있도록 한다. SECS-II의 메시지 블록 형식은 SECS-I의 형식과 다르다. SECS-II의 메시지 블록은 257 바이트이며, 하나의 메시지는 최대 32,767 블록으로 정의되어 있다. 결국, 하나의 메시지는 최대 7,995,148 바이트까지 허용된다. 따라서 송신 스래드에서는 여러 개의 블록으로 구성된 하나의 메시지를 분리하여 각각의 블록마다 헤더를 첨부하여 SIO 스래드에게 전달한다. 이때 전송 플래그를 설정함으로써 SIO 스래드와 동기화 한다.

수신 스래드는 송신 스래드와 반대되는 기능으로써 수신 플래그가 설정되어 있을 경우, 수신 버퍼로부터 메시지 블록을 가져와 SECS-II의 메시지 블록 형식에 맞도록 변환하여 메시지 수신 큐에 저장하고 수신 플래그를 해제한다.



(a) 송신 스래드



(b) 수신 스래드

(그림 14) 송수신 스래드 흐름도

SIO 스래드는 실시간으로 처리되어야 한다. 전송 플래그가 설정되어 있으면 송신 버퍼의 내용에 체크섬

바이트를 첨가하여 SIO 포트로 전송하고 전송 플래그를 해제한다. 호스트로부터 메시지를 수신할 경우, 체크섬 바이트를 확인한 후 수신 버퍼에 저장하고 수신 플래그를 설정한다.

3. 결 론

반도체 제조 과정에서 사용되는 Wet Station 장비의 동작 상태를 감지하고, 이를 간접 제어하기 위한 Wet Station 모니터링 시스템을 제시하였다. 제시한 시스템은 외부 인터페이스의 요구사항을 모듈화하여 4개의 태스크로 구성하고, 각 태스크의 기능과 구조를 설계하였다.

시스템 설계의 기본 목표는 하드웨어 및 소프트웨어 범용성과 사용자의 편의성을 제공하는 것이다. 이를 위하여 IBM PC 호환 기종의 윈도우즈 NT 운영체제에서 GUI 기능을 이용한 사용자의 편의성과, Ethernet 보드를 이용한 PC-PLC간 인터페이스, RS-422 보드를 이용한 PC-온조기간 인터페이스, 그리고 SIO 포트를 이용한 RS-232C 인터페이스를 사용함으로써 시스템의 범용성을 제공할 수 있도록 설계하였다.

본 논문에서 제시한 모니터링 시스템은 윈도우즈 NT 환경에서 Visual C++를 사용하여 각각의 태스크를 구현함으로써 Wet Station 장비에 적용하였다.

참 고 문 헌

- [1] Mitsubishi Electric Co., "Programmable Controller Melsec-A Manual", 1996.
- [2] Silberschatz, Galvin, "Operating System Concepts," Addison Wesley, 1994.
- [3] Richard Simson, "Win32 Programming API Bible," Waite Group Press, 1996.
- [4] Kelvin J. Goodman, "Building Windows 95 Applications", M&T Books, 1997.
- [5] 황미자, 한광록, "Wet Station 모니터링을 위한 제어 모듈의 설계 및 구현", 한국정보처리학회 논문지, 5권7호, pp.159-163.
- [6] 이상중, 임성락, "윈도우즈 NT에서 TCP/IP를 이

용한 PC-PLC간 통신", 한국정보처리학회 '97 춘계학술발표 논문집, 4권1호, pp.821-826.

- [7] Min System, "SECS Communication Handbook," GW Associate, 1989.



임 성 락

e-mail : srrim@dogsuri.hoseo.ac.kr
 1979년 서강대학교 전자공학(학사)
 1983년 서울대학교 컴퓨터공학(석사)
 1992년 서울대학교 컴퓨터공학(박사)

1983년~1990년 금성 반도체(주) 연구소
 1993년~현재 호서대학교 컴퓨터학부 부교수
 관심분야 : 운영체제, 실시간처리 시스템, 분산처리 시스템



한 광 록

e-mail : krhan@dogsuri.hoseo.ac.kr
 1984년 2월 인하대학교 전자공학과 졸업
 1986년 2월 인하대학교 대학원 전자공학과(공학석사)
 1989년 8월 인하대학교 대학원 전자공학과(공학박사)

1991년 3월~1999년 현재 호서대학교 컴퓨터공학부 교수
 관심분야 : 자연어처리, 정보검색, HCI, 멀티미디어 저작시스템 등

최 용 엽

e-mail : yychoi@dogsuri.hoseo.ac.kr
 1979년 명지대학교 전자공학과 졸업(공학사)
 1990년 명지대학교 대학원 전자공학과(공학석사)
 1990년 명지대학교 대학원 전자공학과(공학박사)

1987년~현재 호서대학교 컴퓨터공학부 교수
 관심분야 : 퍼지이론, 멀티미디어 시스템, 인공지능