

PC를 위한 MAP 망접속기 구현에 관한 연구

°임 용재, 김 덕우, 정 범진, 추 영열, 권 옥현
서울대학교 공과대학 제어계측 공학과

A Study on the Implementation of MAP network interface for PC

Yong Je Lim, Deok Woo Kim, Bum Jin Chung, Yung Yul Chu, Wook Hyun Kwon
Dept. of Control and Instrumentation Engr., Seoul National Univ.

ABSTRACT

The implementation of MAP(Manufacturing Automation Protocol) NIU(network interface unit) is studied and presented in this paper. Hardwares and system softwares for network management, user interface, and medium access control are developed. Also LLC(Logical Link Control) protocol is implemented. Afterwards, The operations of NIU using self-test functions are tested. Compatibility between interfaces according to IEEE 802.4 standard is also tested.

1. 서론

산업의 발달과 함께 공장 자동화가 지속적으로 추구되면서 제조 과정에 많은 자동화 기기들이 도입되었다. 이러한 자동화 기기들이 제조 과정에 도입됨으로써 정보의 수집 및 제어, 모니터링 등에서도 점차 기능을 확대해가고 있다[1]. 기존의 각종 자동화 기기를 역시 근래에 눈부신 발전을 거듭하고 있는 반도체 기술에 힘입어 점차 지능화되고 기능에 있어서도 다양화되고 있다. 이러한 프로그램 가능한 제어 기기들은 제조 공정이나 과정에서 개개의 프로세스가 바뀌어도 프로그램의 변경에 의해, 유연하게 적용할 수 있는 폭이 확대되었다. 앞으로도 이러한 자동화의 추세는 더욱 가속화될 것이 분명하다. 이에 따라 제조 과정에 있어서 각 요소의 자동화와 함께 이 요소들의 전체적인 통합이 주요한 과제로 대두되었다[2]. 통합된 제조 공정(Integrated Manufacturing)에 있어서 전체적인 운영의 효율성은 공장내 각 기기들의 유효한 정보 교환 능력에 의해 좌우된다. 더우기 생산 공정이 변경될 때마다 비용이 추가되게 되고 기기의 설치와 소프트웨어의 변경에 소요되는 인력과 시간의 낭비, 그에 따라 교육에 소비되는 시간등의 소모 역시 무시할 수 없는 요소이다. 이러한 배경에서 GM은 MAP(Manufacturing Automation Protocol) Task Force를 설립하고 공장용 자동화 네트워크로써 MAP을 제안하게 되었다[3]. 이렇게 시작된 MAP Specification 작업은 그간 여러 번의 개정을 거쳐 1987년 5월에 MAP V3.0[4]이 발표된 바 있다. 국내에서도 그동안 MAP에 대하여 연구가 진행되어

왔으나 Mini-MAP에 대한 연구는 부진한 실정이다. 본연구는 공장내의 여러 자동화 기기들을 통합 관리하고, 실시간내에서의 정보 교환을 가능케하는 공장용 네트워크인 Mini-MAP[5][6]을 구현한 망접속기의 구현을 목적으로 한다.

II. Mini-MAP의 구조

Mini-MAP은 실시간 처리를 위하여 제안된 특별한 구조이다. 이는 계층 1,2,7[7] 로만 이루어지고 carrier band를 사용하며 5 Mbps의 전송 속도를 갖는 토큰캐싱버스[8] 구조로 되어있다. 제 2 계층은 응답 시간을 빠르게 하기 위하여 IEEE 802.2 class 3를 선택사항으로 채택하였다. 또한 제 7 계층의 경우는 공장의 자동화 기기를 연결하기 위한 산업용 LAN의 통신 명령 언어를 규정하는 MMS(Manufacturing Message Standard)가 채택되어 현재 표준안을 확정하기 위한 작업중에 있다.

a. 물리적 계층: IEEE 802.4[8]

IEEE 802.4 는 Duo Binary AM-PSK, Phase Coherent FSK, Phase Continuous FSK 등 3가지 변조방식을 채택하고 있으며 MAP 은 이중에서 처음 2 가지 방식을 채택하고 있다. Mini-MAP이 채택한 것은 캐리어밴드 전송방식이다.

b. MAC 계층[8][7][4]

MAC(Medium Access Control) 계층은 실제적으로 네트워크의 구조와 성능을 결정짓는 계층으로 LLC 계층과 밀접한 연관을 지닌다. MAC 계층의 기능은 다음과 같이 요약될 수 있다.

- 토큰의 관리
- 논리적 링의 구성과 유지
- 오류검출과 복구

c. 데이터링크 계층: IEEE 802.2[8][7]

데이터링크층 IEEE 802.2 LLC type 1 서비스를 사용한다. 그러나 응답에 고속처리가 필요한 경우 선택사항으로 IEEE 802.2 LLC type 3 서비스를 사용할 것을 제안하고 있다. 데이터링크 계층과 응용계층과의 연결은 MAP V3.0에 의해서 정의된다.

d. 응용계층 : MMS[9]

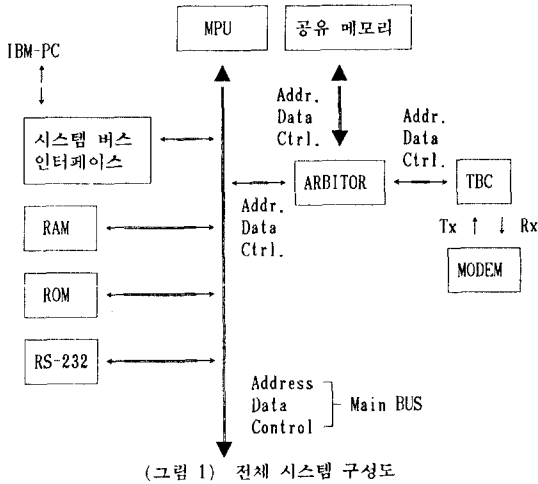
응용계층은 MMS(Manufacturing Automation Protocol)라 불리는 ISO 표준인 ISO/DIS/9506 이 사용 되고 있다.

III. Mini-MAP 망접속기의 구현

(1) 전체의 구조

IBM-PC를 대상 시스템으로 하는 Mini-MAP을 구성하기 위해서는 IBM-PC와 접속이 가능한 인터페이스를 구현 하여야 한다. 이렇게 인터페이스의 형태로 Mini-MAP을 구현하면, Mini-MAP의 계층(Layer)중 시간의 지연이 전체의 성능에 크게 영향을 미치는 MAC와 이에 종속적인 LLC등을 하드웨어에 의해 인터페이스 자체만으로 독립적인 구성이 가능하게 된다. 또한 비교적 사용자와의 인터페이스가 중요시되는 상위계층을 IBM-PC에서 제공할 수 있는 각종 상위언어등을 통하여 손쉽게 구현 한다는 잇점이 있게된다. 인터페이스는 다음과 같이 크게 세부분으로 나누어 진다.

- 프로세서(MPU) 및 메모리부
- 토큰버스 제어기 및 하위 계층 구현부
- 대상시스템 연결부



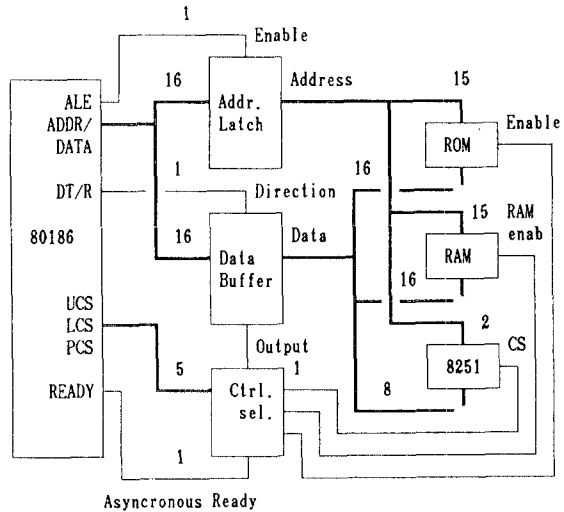
(그림 1) 전체 시스템 구성도

(2) 프로세서(MPU) 및 메모리 부[10][9]

이 부분은 인터페이스 전체의 동작을 통제한다. 이 부분을 통해서 인터페이스가 갖추어야 할 기능중 데이터의 증계와 관리, MAC 기능의 관리와 상위 계층과의 전송등이 이루어 진다. 구성요소는 공유메모리, ROM, RAM, PCI(Programmable Communication Interface) 등이다. 여기서 수행하는 기능은 다음과 같다.

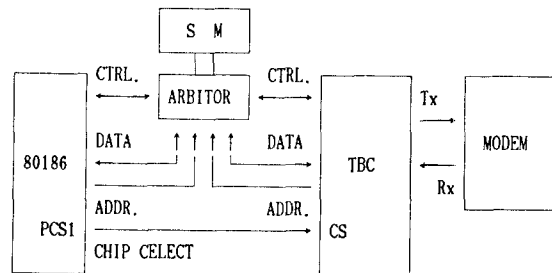
- 인터페이스의 초기화
- 메모리 관리, 송수신 버퍼의 관리
- 송수신 큐(Queue)의 관리
- 토큰버스 제어기(Token Bus Contriller)의 초기화 및 제어
- 링(Ring) 구성후 접속상태의 변화에 따른 관리

MPU로는 인텔(Intel)사의 80186 을 사용하였다. 이 프로세서는 칩 내부에 디코딩(Decoding)회로, DMA 회로, 타이머(Timer)회로 등을 포함하며 외부메모리의 처리속도(Access time)를 맞추기 위한 동기신호 발생회로 등이 내장되어 있으므로 회로 구성시 부품을 적게 사용하는 잇점이 있다. TBC와 공유영역을 제외하고 80186의 독립동작을 위한 기본회로의 구성은 (그림 2)와 같다.



(그림 2) MPU 기본회로

TBC 공유 메모리는 이 영역을 통해 MPU 와 TBC, 궁극적으로 스테이션간의 통신이 이루어지므로 이 영역의 설계 및 관리가 인터페이스 전체 성능에 큰 영향을 미친다. 공유 메모리 영역은 80186과 TBC가 공동으로 사용하지만 전송지연에 의한 오동작을 방지하기 위하여 TBC가 사용 우선권을 갖는다. TBC가 사용 우선권을 갖게 하기 위하여 TBC로 부터 버스 사용요구(BUS REQUEST)가 있을경우 80186의 HOLD 신호를 조작하여 186이 TBC사용 기간동안 버스사용을 포기하도록 한다[11]. 이경우 80186은 HOLD신호가 유효한동안 어떤동작도 할 수 없으므로 (그림 3)과 같이 80186의 비동기 신호를 대신 이용하여 TBC가 공유메모리를 사용하더라도 80186은 동작을 정지하지않고 다른작업의 수행가능 하도록 구성할수있다.



(그림 3) TBC의 접속

전송지연을 미연에 방지하기 위하여 그 지연 시간을 구한다음 위와 같이 구현한 회로의 타당성을 검토한다. 일반적으로 한 스테이션에서 다른 스테이션으로 데이터를 전송할때 전송시간은 다음과 같다[12].

$$\text{전송시간} = T_q + T_a + T_t$$

T_q : 큐잉 지연

T_a : 매체 획득 지연(Media Access Delay)

T_t : 전송지연(Transmission Delay, Propagation Delay 포함)

T_q 는 송신큐의 제일 앞단에 위치할 때 까지의 시간이고 T_a 는 인터페이스가 토큰을 획득하여 전송을 시작하기 직전까지의 시간이며 T_t 는 전송 매체 상에 데이터를 보내기 시작하여 목적 스테이션까지 도착하는데 걸리는 시간이다. (그림 3)과 같은 구조를 통한 데이터의 전송은 비스 경쟁없이 전송지연을 최소화하여 이루어지게 된다. 이상과 같이 설계된 공유 메모리 내부에는 다음과 같은 영역이 위치하게 된다.

- 초기화 테이블
- 링 초기화 및 MAC기능에 필요한 변수들의 테이블
- TBC 전용 테이블 (Private Table)
- TBC 가 MAC 기능 수행 중 필요한 변수들의 저장 영역
- FRAME 및 버퍼 DESCRIPTOR
- 송수신 큐

(3) 토큰 버스 제어기 (Token Bus Controller) 부[11][13][14]

이 부분은 기술한 IEEE 802.4 표준안의 MAC 기능을 구현한다. 시간의 지연이 특히 예상되므로 하드웨어에 의한 실현이 요구된다. 주요 기능은 다음과 같다.

- 네트워크로의 데이터 전송 및 수신
- 논리적 링의 구성
- 토큰 관리
- 네트워크에의 참여 및 탈퇴
- 상위 계층 및 모뎀과의 접속
- 오류 관련 기능

이 부분을 구현한 VLSI들이 발표되고 있으며 본 연구에서는 모토롤라의 MC 68824 TBC 칩을 사용하여 설계하였다. TBC 부분은 위로 MPU와 접속되고 아래로 모뎀과 그 이하 (계층 1)과 접속된다.

손실없는 데이터 수신을 위한 FIFO의 크기와 DMA setup 시간에 대해서는 이미 연구된 바 있다[15]. 한 개의 DMA 채널을 가진 경우 다음 의 식이 성립해야 정보의 완전한 수신이 가능하다.

$$B \geq \gamma R \quad \text{-----} \quad (1)$$

$$\gamma + \gamma R / (D-R) \leq L / R \quad \text{-----} \quad (2)$$

γ : DMA setup time

R : 전송 매체의 전송 속도

D : DMA 채널 속도

B : 버퍼 크기

L : 최소 프레임 길이

(1)에서 $\gamma \leq B/R$ 이고 설계된 망접속기의 경우 32바이트의 여유

버퍼 크기를 가지므로 full MAP의 경우

$$32\text{바이트} \times 8\text{비트} \times 0.1\mu\text{sec} = 25.6\mu\text{sec}$$

즉, $25.6\mu\text{sec}$ 의 overflow 지연 시간을 갖는다. Mini MAP의 경우는 $51.2\mu\text{sec}$ 가 된다. 또 (2)식을 정리하면 다음과 같다.

$$\gamma \leq L(D-R)/RD \quad \text{-----} \quad (3)$$

MC 68824는 10MHz의 클럭에 대해 40 Mbps의 속도를 가지므로 full MAP의 경우 $R = 10\text{Mbps}$, Mini MAP의 경우 $R = 5\text{Mbps}$ 에 따라 각각 아래와 같이 된다.

$$\gamma \leq 0.075 L \quad (R = 10\text{Mbps})$$

$$\gamma \leq 0.175 L \quad (R = 5\text{Mbps})$$

Mini MAP의 경우 메시지를 최하로 잡을 경우 16바이트라 하면 $22.4\mu\text{sec}$ 의 setup 여유시간을 주게 되므로 이는 충분한 시간을 제공한다. DMA의 형태는 burst 형태와 제한된 burst mode가 있다. 전자의 경우 한번의 DMA에 FIFO의 데이터를 모두 처리하며 후자는 최대 8 메모리 싸이클 후 버스에 대한 주도권을 풀고 처리할 데이터가 남을 경우 다시 요구한다. TBC는 버스 에러가 발생하지 않은 상태에서 다음의 순서에 의해 버스 사용권을 획득한다.

- BR 신호 발생
- BG 응답 여부 확인
- AS와 BGACK의 inactive 기다림
- BGACK active (output)

TBC가 버스를 요구하는 경우는 FIFO와 공유 메모리 사이에서의 데이터 교환과 공유 메모리내 사유 테이블 (Private table)을 사용할 때이다.

IEEE 802.4 표준안은 전송과 수신에 있어 우선 순위를 갖는 4개씩 큐를 규정하고 있다. 이것이 공유 메모리에 구현된다. 데이터 프레임 전송하기 전 프로세서는 다음을 준비해야 한다.

- 프레임 디스크립터 (FD)
- 버퍼 디스크립터 (BD)
- 데이터 버퍼 (DB) → 실제 메시지가 위치
- 프레임 디스크립터와 버퍼 디스크립터를 연계시킴
- 전송 큐 enable 시킴

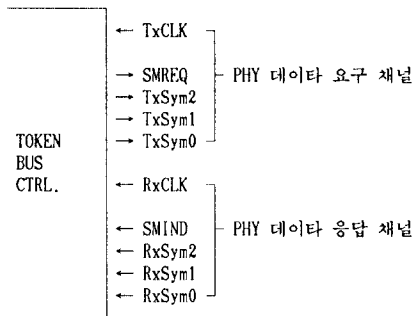
수신 시에도 마찬가지로 작업이 프로세서에 의해 수행 되어야 한다. 데이터가 수신됨에 따라 미리 연계된 순서대로 이 버퍼들이 해당 영역에 할당된다. 이 FD, BD, DB 들이 부족하지 않도록 프로세서는 계속 준비해야 한다.

물리적 계층은 전송 방식 및 전송 매체에 대한 규정이다. 전송 방식과 그 기능은 주로 모뎀(Modem)에 구현된다. 본 망접속기는 IEEE 802.4에서 제시된 3가지 방식 중 위상 응집 FSK(Phase Coherent FSK) 방식의 5 Mbps모뎀을 사용하였다. MAC와의 접속은 TBC의 10개의 신호선을 통해 표준안을 구현한다. (그림 4)는 인터페이스 신호를 보여준다.

모뎀의 동작 형태는 MAC 동작형태와 스테이션 관리형태로 구분되며 채널의 인코딩 형태에따라 구분된다.

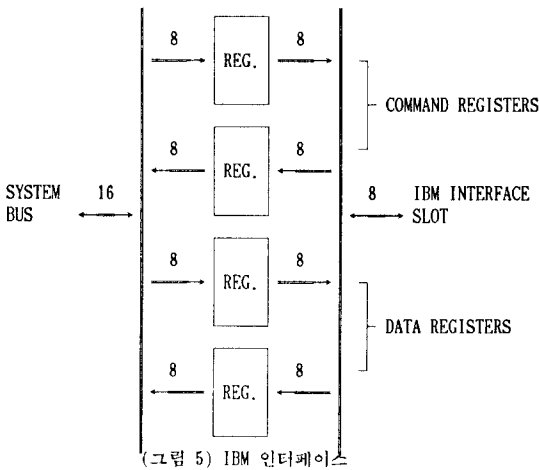
(4) IBM 연결부[16]

(그림 5)는 인터페이스에 구현된 연결부로 폴링과 레지스터에 의한 방법으로 구현되어 있다. 즉, COMMAND 레지스터를 통해



(그림 4) MAC-PHY 접속

서로가 원하는 동작을 지시하거나 자신의 상태를 표시할 수 있으며, 그에 따르는 데이터를 데이터 레지스터를 통해 주고 받게 된다.



(그림 5) IBM 인터페이스

IV. 시스템 소프트웨어

(1) 전체의 구조

시스템 소프트웨어에서 제공하는 기능은 다음과 같다[17].

- TBC 드라이버 소프트웨어
- IBM 인터페이스 소프트웨어

TBC 드라이버 소프트웨어는 TBC 및 전송과 수신용 버퍼의 초기화 루틴, 링(ring)에 들어간 후에 데이터 프레임(data frame)을 보내는 루틴과 데이터 프레임을 받은 후에 수신용 버퍼를 갱신(update)하는 루틴으로 크게 분류할 수 있다. 반면, IBM 인터페이스 소프트웨어는 호스트로부터의 데이터를 두 개의 레지스터를 통해서 받아들여서 송신용 버퍼에 저장하는 작업과 수신된 데이터를 IBM에 보내는 작업으로 나누어진다[18][19].

(2) TBC 드라이버[11]

TBC는 전술한 바와 같이 IEEE 802.4, 토큰 패싱 버스 프로토콜을 하드웨어적으로 구현한 마이크로 프로그래밍된 VLSI 칩이다. 따라서 TBC는 ISO의 OSI 모델중 MAC sublayer를 담당하게 되는데, 실제로 이러한 기능을 TBC로 하여금 수행토록하기 위해서는 TBC 및 TBC와 serial interface되는 모뎀의 초기화

작업이 필요하게 된다.

TBC는 두 가지 동작 모드가 있는데 CPU의 slave로 동작하는 모드와 CPU로부터 버스를 획득하여 버스 master로서 동작하는 모드가 있다. slave 모드를 지원하기 위해서 TBC에서는 다음과 같은 레지스터를 두고 있다.

- Command register(CR)
호스트 프로세서가 TBC에 명령을 내릴 때에 사용된다.
- Data register(DR)
Initialization table 번지 지정.
- Interrupt Vector register(IV)
호스트에 인터럽트 서비스 루틴을 지정하는 데에 사용된다.
- Semaphore register(SR)
호스트 프로세서가 TBC에 어떤 명령을 내렸을 때 이를 받아들였음을 호스트에 알려주기 위해서 사용된다.

TBC의 초기화 작업을 수행하기 위해서는 먼저 TBC를 reset시킨다. reset이 된 후에는 initialization table을 준비해야 한다. initialization table은 TBC와 공유하는 메모리 영역에 존재하는 256 바이트의 메모리 영역이다. 이 table은 호스트가 TBC 동작시에 필요한 초기 변수들과 포인터들을 TBC에 넘겨주는 데에 사용된다. 또한 이 table은 TBC에 의해서 status, static 카운터, command return information(parameters)들을 호스트에 넘겨줄 때도 사용된다. 이 table외에도 private table이 있는데 이는 TBC가 MAC를 구현하는 데 필요한 값을 수시로 저장하는 데에 사용되는 영역으로 initialization table과 마찬가지로 호스트와의 공유 메모리 영역에 존재한다. initialization table이 준비된 후에는 다음과 같은 순서로 TBC의 초기화 작업을 진행한다[11][14].

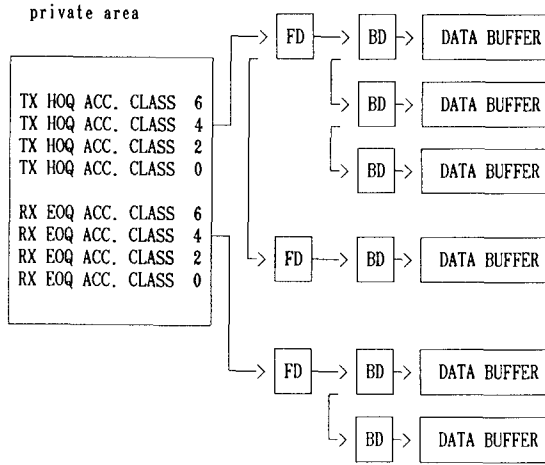
- 인터럽트 벡터 레지스터 초기화
- LOAD INITIALIZATION TABLE FC 명령(D1) -> CR
- SR 레지스터의 값이 'FF'가 될 때까지 기다린다.
- initialization table pointer -> 데이터 레지스터

- INITIALIZE 명령(C7) -> CR
- SET MODE 1 명령(22) -> CR
- SET MODE 2 명령(48) -> CR
- SET MODE 3 명령(63) -> CR
- 모뎀 초기화
- IDLE 명령(F0) -> CR

TBC는 주로 호스트와의 공유 메모리를 통하여 호스트와 통신을 하게 된다. 이 공유 메모리는 아래와 같이 크게 3 부분으로 나누어진다.

- initialization table
- private table
- buffer structure

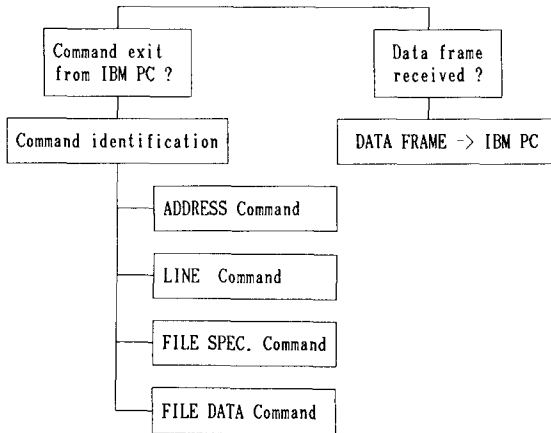
이 버퍼 구조는 다시 다음과 같이 3 부분으로 되어 있어서 서로 링크(link) 구조를 가지고서 연결되어 있다.(그림 참조)



(그림 6) 버퍼 구조

(3) IBM 인터페이스 소프트웨어

호스트 프로세서는 TBC의 초기화 작업후에는 IBM PC와의 통신을 통해서 수신된 데이터를 IBM PC에 전달하거나 IBM PC로부터의 데이터를 송신하는 일을 담당한다.



(그림 7) IBM 인터페이스 드라이버의 전체구조

(3) LLC(Logical Link Control)의 구현[8]

앞에서도 언급된바와 같이 Mini-MAP에는 type 1 형태가 구현되어야한다. 본 연구에서는 C언어를 이용하여 다음의 서비스를 구현하였다.

- UI(Unnumbered Information) command
- XID(Exchange Identification) command/response
- TEST command/response

V. 자기진단 및 망 운영진단

제작된 시스템의 테스트는 크게 둘로 나뉘어 수행되었다. 첫째는 자기진단 테스트로써 인터페이스의 자체 동작을 검사하는 과정이다. 둘째는 망을 구성하고 그 망의 운영상태를 검사하는 과정이다.

(1) 자기 진단 테스트[11][3]

MC68824는 자체 시스템의 동작 여부를 테스트 하기위한 명령들을 갖고있다. 이를 통해서 TBC의 기능, 모델의 기능, 그리고 공유메모리와의 데이터 교환을 테스트 하였다.

- 호스트 접속 테스트
- 쌍방향 루프 백 테스트
- 전송 테스트
- 수신 테스트

(2) 모뎀 테스트[11][3]

모뎀이 IEEE 802.4 규격에 맞는 기능을 수행하는지는 다음의 테스트를 거쳐 확인하였다. 이때, 모뎀은 스테이션 관리상태이고 MAC는 OFFLINE 상태이다.

- Immediate Test

모뎀과의 사이에 데이터의 교환은 없고 단지 Indication 채널을 통한 응답만으로 테스트한다.

- 모뎀과의 데이터 교환 테스트

모뎀이 지능을 가지고 있을 경우 호스트 프로세서와 모뎀은 TBC를 거쳐 제어정보 등의 정보 교환이 가능하다.

(3) 망 운영 진단

- 제작된 망 접속기 간의 망 구성

앞에서 제시된 자기진단 과정을 통해 각 접속기의 진단을 마친후, 각 접속기들을 망으로 묶고 시스템 소프트웨어를 올려 제대로 동작되는가를 확인 하였다.

- 기존의 MAP 망 접속기와의 망 구성[20][21]

기존의 시스템으로는 Concord사의 MAP 인터페이스를 이용하였다. 모뎀을 통하여 토큰이 회전되고 있음을 확인하였고, 기존의 시스템에서 제작된 망접속기의 주소로 전달된 데이터들이 오류없이 받아들여지고 있음을 또한 확인하였다.

VI. 결론

본 연구에서는 Mini-MAP의 규격에따르는 MAP 망접속기의 하드웨어와 LLC를 구현하고 망 운영을 위한 소프트웨어를 작성하였다. 이를 토대로 제작된 망 접속기간의 망 구성여부를 시험하였으며 사용자간의 데이터전달을 확인하였다. 또한 기존의 MAP 망 접속기와의 망 구성여부 또한 시험하였다. 망접속기 내의 물리적 계층 컨트롤러와 망 접속기 운영 프로세서간의 접속 형태등 망 접속기의 구조 설정을 위한 연구가 부가적으로 진행되었으며, 앞으로 MMS를 비롯한 상위 계층의 프로토콜 구현 및 망 접속기의 성능해석을 위한 연구가 더 진행되어야 할것이다.

VII. 참고문헌

[1] 추 영열, MAP을 위한 망접속기의 구현에 관한 연구,1988, 서울대학교 대학원 제어계측 공학과 석사 학위 논문

- [2] Mary Gallagher, Low Cost Networking for Island of Automation, Ctrl. Engr. Oct. 1986.
- [3] 최 업, 공장 자동화용 네트워크의 기술 동향, 한국 자동제어 학술회의 논문집, Vol.1 pp.376-381, 1987. 10.
- [4] MAP Task Force, MAP V3.0 General Motors, 1987. 5.
- [5] Maris Graube, The Carrierband Network and Mini-MAP:Low Cost Solutions, Control Engr., 1986. 10.
- [6] T.J. Balph, Carrierband Get Nod for Industrial Networks, Electronic Products, Aug. 1985.
- [7] Knowles, Larmouth and Knightson, Standard for Open Systems Interconnection, BSP, 1987.
- [8] IEEE Standard 802.2/802.4, Mar. 19, 1985
- [9] W. Stallings, Computer Communications: Architectures, Protocols, and Standard.
- [10] James W. Coffron, Programming the 8086/8088, SYBEX, 1983.
- [11] Motorola, MC68824 Token Bus Controller User's Manual, 1987.
- [12] C.F. Summer and A.C. Weaver, Performance of IEEE 802.4 - Basis of MAP, IECON'86, IEEE.
- [13] Ivan L. Erickson, VLSI for a Carrierband Node, IECON'86, IEEE.
- [14] R.A. Dirvin and A.R. Miller, The MC 68824 Token Bus Controller, IEEE MICRO, Jun. 1986.
- [15] J.W. Wong and Werner Bux, Analytic Modeling of an Adapter to Local Area Networks, IEEE Tr. on Comm. Vol. Com-32, Oct. 1984.
- [16] IBM, Technical Reference PC-XT, 1984.
- [17] G. Morris and R. Carler, MAP Network Management and Administration, Ctrl. Engr. Oct. 1986.
- [18] 김 덕우, 실시간 응용시 MAP의 성능 해석에 관한 연구, 1987, 서울대학교 대학원 제어계측 공학과 석사 학위 논문.
- [19] 정 범진, 스테이션에서의 시간 지연을 고려한 Full MAP의 성능 해석에 관한 연구, 1988, 서울대학교 대학원 제어계측 공학과 석사 학위 논문.
- [20] Concord Comm., S-1200 PC Bus Controller Programmer's Reference Manual, 1987. 10.
- [21] Concord Comm., S-1200 Carrierband Modem Product Description, 1987.