

## 로봇 데이터가 필드버스의 효율에 미치는 영향에 관한 연구

### A Study on the Effect of the Fieldbus Performance by the Robot Data

\*김주한\*, 김동준\*, 김갑일\*, 김대원\*, 김홍석\*\*, 이호길\*\*

\* 명지대학교 전기전자공학부 (Tel&Fax: 0335-30-6356, Email: kkl@wh.myongji.ac.kr)

\*\* 생산기술연구원 (Tel: 8509-255, Fax: 8509-244, Email: r6169@intech.kaitech.re.kr)

**Abstracts** Recently, the importance of the network using Fieldbus is being increased in an implementation of factory automation system. In spite of expanding the concept and products of Fieldbus, it is not widely used yet. This is partly due to the lack of the standard specification and the relatively low performance with respect to the cost. So, in order to expand the usage of the Fieldbus, not only the standardization effort, but also the increase of performance is needed.

In this paper, the method which increases the performance of Fieldbus by reforming of the sensor and the actuator data is thoroughly estimated and evaluated when the assumptions about the real-time characteristics of a robot workcell as an example of the ICCS(Integrated Communication and Control System) are sustained.

**Keywords:** Fieldbus, Robot, Sensor and Actuator Data, Performance

#### 1. 서론

최근의 자동화 시스템의 특징은 워셀 단위의 모듈화, 디지털화에 따른 정확성, 및 통신망의 도입에 따른 통합성의 양상을 띠고 있다. 자동화시스템의 조직적 운용을 위해서 MAP (Manufacturing Automation Protocol), Mini-MAP 그리고 필드버스의 세가지 종류의 통신망이 개발되었다. 그러나 아직 생산 현장에서 실용성과 경제성이 완전히 검증되지 않은 상태이거나 표준화의 부재 및 인식의 부족에 의해서 사용이 활발하지 않은 상태이다. 그중 필드버스는 설치된 각종 센서, 액추에이터, PLC, 로봇, CNC 등의 자동화 기기에서 생성되는 데이터를 실시간 처리하며, 공장 자동화 통신망에서 최하위에 놓이는 기본 통신망이다[1]. 그러나 MAP 이나 mini-MAP과 같이 표준화가 이루어지지 못하여 여러 나라의 시스템이 각기 다른 사양으로 발표되고 있는 실정이다. 통신망의 표준화를 위해서는 데이터 및 기기의 상호동작성(interoperability)과 상호교환성(interchangeability)을 보장하여야 한다. 한편 필드버스의 특성상 여러센서 및 기기의 많은 데이터를 취급하여야 하므로 통신망의 효율을 높이기 위한 연구도 병행되어야 할 것이다. 이러한 관점에서 본 논문은 로봇 시스템에서의 필드버스 적용시의 로봇 데이터의 분석과 그것을 통한 필드버스 스케줄링에 적합한 데이터의 형태 및 스케줄링에 관하여 다루고자 한다.

이와같이 필드버스의 성능을 향상하기 위한 연구 중에서도 제어시스템과 통신망의 통합(ICC: Integrated Communication and Control System)에 관한 연구를 한 Ray 등[2]은 비동기 TDMA

통신망을 사용하는 제어시스템에 유기되는 시변지연시간(varying delay)의 관점에서 분석 및 설계시의 고려사항에 관해서 언급하였으며, Hong[3]은 ICCS의 제어루프의 지연시간을 스케줄링에 의해서 줄임으로써 통신시스템의 성능을 향상할 수 있다고 하였다. 또한 [4]는 FIP 필드버스를 기반으로하는 제어시스템의 네트워크의 지연시간을 최소화하는 방법을 제시하였다. 그러나 이러한 방법들이 실제의 제어시스템에의 적용을 수행하지는 않았다. 한편 [5]는 IEC 필드버스를 사용하여 오렌지를 따는 로봇시스템의 성능을 분석하여 보았다. 그러나 실제적인 적용이 되어지지 못하고 페트리리넬을 이용한 모델링을 한 수준이다. 본 논문에서는 실제 공장에서의 사용이 가능한 지능형 두 로봇을 이용한 워셀에서의 실제적인 필드버스에서의 데이터의 운용[6,7]에 근거하여 통신성능을 향상하기 위한 로봇시스템의 자료의 형태 및 샘플링에 관해서 연구하고자 한다.

2장에서는 대표적인 필드버스 들인 Profibus, FIP, CAN, IEC의 특징들 중에서 매체 접근 방식(media access method)을 비교하여 보고, 3장에서는 로봇 시스템 및 기본자료를 분석하며, 4장에서는 스케줄링 방법 및 효율을 알아보고, 마지막으로 결론을 맺는다.

#### 2. 필드버스 시스템 - 매체접근방식 중심으로

##### 2.1 Profibus

독일 국가 표준안으로 정해진 필드버스로 DIN 19245 에 규정

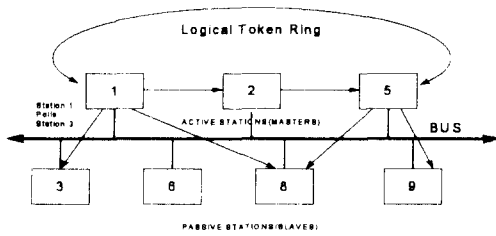


그림 1. Profibus의 매체 접근 방식

Fig. 1. Medium access method of Profibus

되어 있다. 매체 접근 방식은 토큰 버스 방식에 마스터 슬레이브 개념을 더한 혼합형(hybrid) 구조를 갖는다. 그림 1에서 1,2,5는 마스터 스테이션이고 3,6,8,9는 슬레이브 스테이션이다. 마스터 스테이션은 간소화된 IEEE 802.4 프로토콜에 기초한 토큰 메카니즘에 의해 버스에 접근할 수 있는 권한을 가지고 토큰을 가지고 있는 마스터 스테이션은 슬레이브 스테이션을 폴링할 수 있다. 전체 네트워크를 관리하는 관리 스테이션은 없고 각 마스터 스테이션들이 각 스테이션 사이의 토큰 전달, 네트워크 상의 스테이션들의 주소와 상태, 새로운 스테이션의 등록이나 기존 스테이션의 삭제 등을 책임지고 있다. 처음 기동시 LAS(List of Active Station)을 생성하여 현재 통신망 상에서 활성화되어 있는 마스터 스테이션들의 목록을 작성하고, 토큰 보유 시간과 토큰 로테이션 시간에 따라 응용 계층의 서비스가 수행된다[8].

## 2.2 FIP(Factory Instrumentation Protocol)

프랑스의 국가 표준안으로 정해진 필드버스로서 매체접근방식은 중앙 접속 제어 방식(발생자(producer)-분배자(distributor)-소비자(consumer))를 사용하고 거의 브로드캐스트(broadcast)에 의

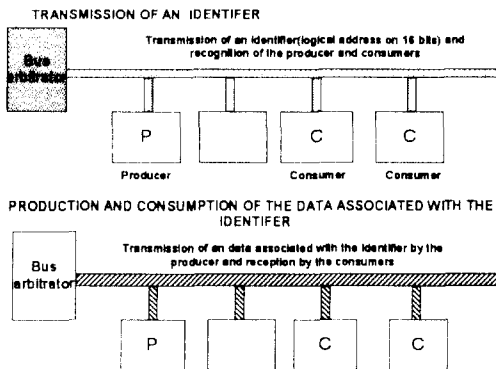


그림 2. FIP의 기본 동작

Fig. 2. Operating principle of the FIP network

존한다. 분배자는 BA(Bus Arbitrator)라 불리는 특별한 프로세서다. 그림과 같이 FIP는 BA와 다양한 스테이션으로 구성된다. BA는 응용 시방으로부터 작성된 모든 변수의 리스트를 가지고 있고 그것에 따라 주기적으로 변수의 이름들을 전송한다. 모든 청취자(subscriber)는 16 bit로 변환된 논리적 주소에 의해 인식되는 각 버퍼 메모리를 가지며, BA는 순차적으로 16 bit의 식별

자(identifier)를 전송하고 그 숫자를 인식한 발생자는 네트워크 상에 대응하는 정보를 보낸다 모든 정보는 네트워크에 연결된 모든 청취자에게 전달되고 이 정보의 인식자를 인식한 하나 또는 몇 소비자에 의해 사용되어 진다[9].

## 2.3 CAN(Controller Area Network)

CAN는 짧은 실시간 제어 메시지를 주고 받기 위해 고안된 통신 버스이다. 이 버스는 자동차 내부의 데이터의 교환같은 소규모 영역의 제어 시스템들을 연결하기 위해 고안되었다. CAN의 프로토콜은 CSMA/AMP (Carrier Sense Multiple Access / Arbitration by Message Priority) 이고 CAN은 여러개의 프로세서가 인터페이스를 통해 연결된 브로드캐스트 버스이다. 데이터는 1-8 byte로 이루어진 메시지가 전송되며 주기적 또는 실시간 등의 요구에 따라 전송된다. 예를 들어 속도계의 데이터 같은 데이터는 1 byte의 메시지로 바뀌어 100ms 마다 브로드캐스트 된다. 데이터들은 11 bit로 표현되는 식별자(identifier)를 부여받는다. 식별자는 두가지 역할을 한다. 즉 메시지를 받았을 때의 필터링과 메시지에 우선권을 부여한다. CAN 버스의 스테이션은 메시지 식별자에 의거하여 메시지를 받을 수 있다. 만약 어떤 특정한 호스트 프로세서가 속도계 데이터를 가져야 한다면 인터페이스 프로세서에서 속도계의 식별자를 가리키다. 요구된 식별자를 갖는 메시지만이 호스트 프로세서에게 주어진다. 따라서 CAN 버스에서 메시지는 목적지를 불리지 않는다[10].

## 2.4 IEC/ISA SP50

IEC/ISA 필드버스는 국제 표준화가 진행 중에 있는 필드버스이다. LAS(Link Active Scheduler)라 불리는 특정 스테이션에 의해 관리되는 중앙 집중 토큰 제어 방식 (Centralized Token-based Protocol)이며 트래픽에 지원되는 세 종류의 토큰은 ET(Execute Transaction), ES(Execute Sequence), CD(Compel Data)이다. ET는 스테이션이 단일 메시지를 전송하도록 하고 단일 주기적 변수의 전송에 적합하다. ES는 같은 프레임 ES에 정해진 타임 윈도우 내에 있는 몇 개의 메시지 전송에 사용되고 스테이션이 주기적, 비주기적 변수를 전송하도록 한다. CD는 LAS나 ES 토큰을 가지고 있는 스테이션에 의해 사용되며 메시지를 전송하기 위해 또 다른 스테이션을 강제하기 위해 사용되고 전형적으로 비주기적 데이터에 적합하나 주기적 데이터에도 적합하다. LAS는 테이블의 길이(macrocycle)에 따라 모든 주기적 전송을 포함하는 스케줄링 테이블에 의해 주기적 데이터의 전송을 관리한다. 링크 상에 연결될 수 있는 스테이션의 종류는 세가지 클래스 즉 폴(poll)에 대해 응답하거나 DLE 간에 연결의 설정, 해제, 및 데이터 교환을 위한 서비스를 제공하는 기본 클래스와 같은 레벨의 LM DLE(Link Master Data Link Entity)간에 LAS 기능을 공유하고, 현재 LAS의 기능을 수행하고 있는 노드의 고장시 LAS의 동작을 대신하며, 링크 내의 로컬 타임은 제공하여 노드 간의 동기화 기능을 수행하는 LM DLE 클래스와 서로 다른 링크를 연결하여, 두개 이상의 로컬 링크간의 데이터 교환을 가능케 하며 내부 타이머를 가지고 있어서 서로 다른 링크간의 동기화를 시켜주는 브리지(Bridge) 클래스로 구분된다[11].

### 3. 로봇 시스템 및 자료 분석

로봇 시스템은 2개의 로봇을 동시에 구동하는 로봇 워크셀에 필드버스를 적용한 것이다. 시스템의 구성은 2개의 로봇, F/T 센서 그리고 PLC 등으로 구성되어 있다. 그 대략적인 개념도는 그림 3과 같으며 여기에서 다루어지는 로봇워크셀의 데이터를 분석하면 다음 표 1과 같다.

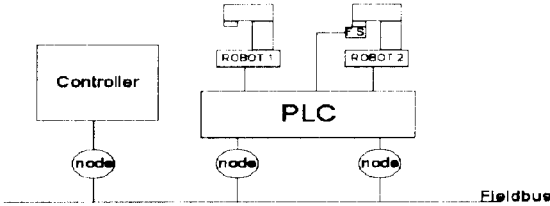


그림 3. 시스템의 구성

Fig 3. Configuration of system

표 1. 로봇 워크셀에서의 데이터 형태

Table 1. Data type in the robot workcell

Device	Actuator	limit S.W.	Emergency Signal	Force/Torque Sensor	Controller
Data generation	resolver encoder	Sensor	Controller	Sensor	controller
Data Type	real-time periodic	real-time Sporadic	real time Sporadic	real time periodic	non-real time data
Raw Data	12 bit	1 bit	?	48 bit	8bit
Modified Data period	2 byte 2 msec	1 byte ?	?	8 byte ?	?

이러한 데이터를 필드버스시스템에 맞추기 위해서는 약간의 변형을 가해서 통신에 편리한 데이터로 바꾸어 주면 통신의 효율을 올릴 수 있을 것이다. 여기서 로봇 당 6축이 있으며 각 축 당 하나의 액추에이터를 갖는 것으로 가정한다. 액추에이터의 정보를 획득하기 위한 센서로는 엔코더(encoder)나 리졸버(resolver)가 쓰이는데 액추에이터 당 12 bit의 데이터를 발생하고 발생된 데이터는 액추에이터의 제어를 위해 주기적으로 샘플링된다. 필드버스에서 액추에이터의 데이터가 어느 축에서 나오는가를 구분하기 위해 적어도 3 bit(6축을 기준)는 있어야 한다. 따라서 액추에이터에서 나오는 데이터에 식별 오버헤드를 붙여 총 15bit에 여유분의 1bit를 포함하여 2 byte로 잡는다. 그리고 제어루프의 샘플링이 2 msec이므로 2msec 마다 데이터가 바뀌게 된다.

그리고 각 축 당 3개의 리미트 스위치(limit switch)를 갖는다. 이것은 각각 오른쪽(right), 왼쪽(left) 끝 및 중간(center)에 있고, 리미트 스위치 당 1 bit 데이터를 갖는다. 이 데이터를 필드버스로 보내기 위해서는 6축이므로 3bit가 붙어야 하고 오른쪽이나 왼쪽이나 중간이나 하는 식별 오버헤드가 2 bit는 붙어야 한다. 그러므로 계산상의 편의를 위해 1 byte로 리미트 스위치에서 나오는 데이터를 잡는다. 그리고 리미트 스위치는 주기적으로 들어오는 데이터가 아니라 산발적으로 발생하는 데이터이다. 또한 위급(Emergency)신호는 로봇의 이상에 의한 오동작에 관한 경보 신호가 추가되며 실시간적 비주기적 데이터이다. 이외에도 로봇에는 많은 데이터들이 생성될 수 있으나 여기서는 기본적으로

로 동작에 필요한 최소한의 데이터의 내용만을 고려해 준다. 한편 백그라운드 데이터(background data)의 생성은 데이터와 프로그램의 덤프, 데이터베이스 생성과 프로그램 다운로드 등과 같은 경우를 생각할 수 있다.

한편 힘/토크 센서는 로봇의 손목에 장착하면 로봇의 손끝에 있는 그리퍼(gripper)에 인가되는 힘을 감지할 수 있는 것으로서 xyz 3축의 힘 및 토크를 측정할 수 있도록 설계되어 있다. 발생하는 데이터는 8 bit x 6 축 즉 48 bit의 데이터를 발생한다. 이 데이터도 또한 실시간 주기적 데이터이다.

PLC(Programmable Logic Controller)는 기기제어나 입출력 포트의 확장을 주목적으로 사용되며 본 논문에서 사용된 PLC는 CPU보드, 4개의 병렬 입출력보드, D/A보드 및 A/D보드를 갖는다. 입력 및 기타 기능을 확장할 경우는 필요한 보드를 확장해 주기만 하면 되며 여기서는 순차 제어를 사용하여 필드버스를 통하여 전달될 각종 신호의 I/O 통로 역할만을 수행한다.

### 4. 스케줄링 방법 및 예시

본 논문에서는 여러 로봇이 하나의 필드버스에 연결되었을 경우를 고려하며 특히 n 개의 컨트롤러 노드와 m 개의 로봇 노드가 있는 필드버스를 생각한다. 단  $n < m$  인 경우이다.

각 로봇의 모든 기본 센서 즉 엔코더나 리졸버는 모든 축(6축)의 데이터가 하나의 패킷을 형성하여 전송되고, 컨트롤러 노드에서 발생되어지는 액추에이터 명령도 모든 축에 대한 명령이 하나의 패킷으로 한 번에 전송된다고 가정한다. 또한 네트워크에서 발생하는 전송 에러는 없다고 가정하고, 컨트롤러에서의 연산시간은 센서나 컨트롤러의 최소 전송시간보다 더 작다고 하며, 샘플링 지연시간도 없다고 가정한다. 이런 가정 하에서 다음과 같은 규칙을 가지고 샘플링을 한다[4].

1. 네트워크의 공허 샘플링이나 데이터 손실이 없도록 샘플링한다.
2. 각 루프의 샘플링 주기는 미리 정해진 최대한계지연시간을 넘어서는 안된다.
3. 할당되어야 할 노드 중에서 최대지연시간이 가장 적은 것부터 할당된다.
4. 센서 노드는 루프에 있는 컨트롤러 노드에 앞서 미디어를 이용한다.

여기서 다음과 같이 변수들을 정의한다.

$T_0$ : 네트워크 오버헤드

$D_{CON}$ : 컨트롤러의 데이터  $T_{CON}$ : 컨트롤러의 데이터전송시간

$D_{RESOL}$ : 리졸버 데이터  $T_{RESOL}$ : 리졸버의 데이터전송시간

$D_{FT}$ : 힘센서의 데이터  $T_{FT}$ : 힘센서의 데이터전송시간

$D_{lim}$ : 리미트스위치데이터  $T_{lim}$ : 리미트스위치 데이터전송시간

$D_{emerg}$ : 위급 신호의 데이터  $T_{emerg}$ : 위급신호의 데이터전송시간

본 논문에서는 문제를 단순화 하기 위해서 그림 3에서와 같이 컨트롤러 노드와 각각 하나씩의 노드를 갖는 2 대의 로봇이 통신 버스에 연결이 된다고 가정한다. 각 로봇은 6 축으로 구성되었고, 각 축마다 하나의 액추에이터와 3 개의 리미트 스위치를 갖는다고 한다. 그리고 한 대의 로봇에만 F/T 센서가 있으며 각 프레임마다의 네트워크 오버헤드를 10 byte라고 하고 전송 속도

를 500 Kbps라 한다.

로봇 1에서 발생하는 데이터 중 리졸버로부터의  $D_{RESOL}$ 는 12 byte( 2 byte x 6축)가 발생되고, 패킷화하면  $D_{RESOL} + T_O = 22$  byte가 되며,  $T_{RESOL}$ 는  $352 \mu sec(22 \times 8 \text{ bit} \times 2 \mu sec)$ 가 된다. 그리고 리미트 스위치로부터의  $D_{lim}$ 은 1 byte가 발생되고, 패킷화하면  $D_{lim} + T_O$ 는 11 byte가 되며,  $T_{lim}$ 은  $176 \mu sec(11 \times 8 \text{ bit} \times 2 \mu sec)$ 가 된다. F/T 센서로부터의  $D_{FT}$ 는 6 byte가 발생되고, 패킷화하면  $D_{FT} + T_O$ 는 16 byte가 되며,  $T_{FT}$ 는  $256 \mu sec(16 \times 8 \text{ bit} \times 2 \mu sec)$ 가 된다. 그리고 위급 신호로부터의  $D_{emerg}$ 는 2 byte가 발생되고, 패킷화하면  $D_{emerg} + T_O$ 는 12 byte가 되며,  $T_{emerg}$ 는  $192 \mu sec(12 \times 8 \text{ bit} \times 2 \mu sec)$ 가 된다. 그리고 로봇 2는 로봇 1에서 F/T 센서를 제외한 것과 같고 컨트롤러 노드에서  $D_{CON}$  12 byte(2 byte x 6축)이고 패킷화하면  $D_{CON} + T_O$ 는 22 byte가 되며  $T_{CON}$ 는  $352 \mu sec$ 가 된다. 이것을 정리하면 다음 표 2와 같다.

표 2. 로봇 데이터 량의 분석  
Table 2. Analysis of robot data

Data	원시 데이터량	패킷의 길이	프레임 전송시간
Robot Resolver	12 byte	22 byte	352 $\mu sec$
limit S/W	1 byte	11 byte	176 $\mu sec$
F/T Sensor	6 byte	16 byte	256 $\mu sec$
Emergency	2 byte	12 byte	192 $\mu sec$
Controller Node	12 byte	22 byte	352 $\mu sec$
합 계	33 byte	83 byte	1328 $\mu sec$

제어 루프의 기본 샘플링 주기는 제어 루프 중 최대 허용 지연 시간이 가장 적은 샘플링 주기를 리졸버의 제어 주기 2 msec로 한다. 먼저 2 msec안에 주기적 데이터 즉  $D_{RESOL}$ ,  $D_{CON}$ 와  $D_{FT}$ 가 주기적으로 샘플링되고 다음 시간에 실시간 데이터인  $D_{lim}$ 와  $D_{emerg}$ 가 샘플링되며, 나머지 시간은 비주기적 비실시간 즉 데이터 베이스와 프로그램 다운로드 같은 일을 위해 예비해 둔다

표 3. 스케줄링 테이블  
Table 3. Scheduling Table

기본 주기 #1		기본 주기 #2		기본 주기 #3	
주기내 누적시간 ( $\mu s$ )	할당 데이터	주기내 누적시간 ( $\mu s$ )	할당 데이터	주기내 누적시간 ( $\mu s$ )	할당 데이터
352	Robot1 Resolver	352	Robot 1 Resolver	352	Robot 1 Resolver
702	Controllier	702	Controllier	702	Controllier
1054	Robot2 Resolver	1054	Robot2 Resolver	1054	Robot2 Resolver
1406	Controllier	1406	Controllier	1406	Controllier
1611	F/T	1598	Emergency	1598	Emergency
1803	Emergency	1774	Limit	1774	Limit
1979	Limit				
2000	Database Program download	2000	Database Program download	2000	Database Program download

표 3은 이상의 결과를 이용하여 스케줄링한 결과이며 표 3의 결과로부터 알 수 있듯이 로봇의 리졸버와 각 위급신호나 리미트 신호는 2 msec 마다 한 번씩 샘플링되고, F/T 센서는 6ms

마다 한 번씩 샘플링된다. 데이터의 형태를 정리하면 액츄에이터는 데이터(12 bit)에서 축오버헤드를 붙여 2 byte로 변형되었고, 리미트 스위치는 데이터(1 bit)에서 축과 위치인식 오버헤드가 붙어 1 byte로 변형되었다. 그리고 만약 같은 노드에 같은 기기가 여러 대 존재할 경우에는 각각의 데이터에 식별 오버헤드를 붙여서 변형시켜야 한다.

## 5. 결론

본 논문에서는 필드버스를 이용한 로봇웍셀의 효율적인 이용을 위해서 로봇 데이터를 분석하고 통신망에서의 실시간 운용을 위해서 제어시스템에서 발생하는 데이터의 샘플링 시간과 통신선로상에서의 샘플링 시간을 고려하여 필드버스가 어느 정도의 데이터 운용이 가능한지를 분석하여 보았다. 추후과제로서 본 논문의 결과를 실제 시스템에 적용하는 실험을 통하여 그 효율성을 입증함이 필요하다. 이러한 연구는 앞으로 많은 문제점을 해결하여야 하며 어느 통신망에서나 효율적으로 운용될 수 있는 데이터의 형태를 지닌 제어시스템의 구성이 필요하다. 앞으로 통신 속도도 점차 빨라지겠지만 제어시스템의 데이터의 형태도 규격화를 통하여 통신의 효율을 높이는 한편 제품의 상호 동작성 및 상호호환성을 유지하는 것도 의미 있는 일이라 여겨진다.

## 참고문헌

- [1] J.D.Decotignie, P.Pleinevaux, "A survey on industrial communication networks," *Annales des telecommuications*, Vol.48, No. 9-10, pp. 435-448, 1993.
- [2] A Ray, Y.Haveli, "Integrated communication and control systems: Part I & II," *ASME J. of DSMC*, vol. 110, pp. 367-381, Dec. 1988.
- [3] S.H.Hong, "Scheduling algorithm of data sampling times in the integrated communication and control system," *IEEE Trans. on control systems technology*, Vol.3, No.2, pp. 225-230, June 1995.
- [4] Yong Ho Kim, Wook Hyun Kwon, Hong Seong Park, "Stability and a scheduling method for network-based-control systems," *IECON 96*, pp. 934-939, Aug. 1996.
- [5] S.Cavalieri, A.Di.Stefano, O.Mirabella, "Meeting time requirements in robotics by a fieldbus communication system," *IECON 93*, Hawaii, pp. 1915-1920, Nov. 1993.
- [6] 김동준, 김갑일, 장혁수, "Field-bus를 이용한 Intelligent Robot Workcell의 구성에 관한 연구," *대한전기학회논문지*, 제44권, pp. 662-670, May 1995.
- [7] 박세웅, 김동준, 김갑일, 김대원, 김홍석, 이호길, "필드버스를 이용한 로봇트 워크셀의 구현과 성능 자료이용", *한국자동제어학술회의 논문집*, 게재예정, Oct. 1996.
- [8] DIN 19 245 Profibus Standard Part 1: 1991
- [9] FIP TECHNICAL DESCRIPTION, pp 1-38
- [10] INTERNATIONAL STANDARD ISO 11898 :1993
- [11] Date Link Protocol Specification, ISA/SP50-360L(draft standard)