프로그래머블 콘트롤러

권 욱 현* · 변 대 규**
(서울대 공대 제어계측 공학과)
*교수 **대학원생

1. PC의 개요

널리 알려진 바와 같이 프로그래머블 콘트롤러(이하 PC로 약함)는 60년대말 GM(General Motors)사가 자동차 조립 line에의 적용을 위한 전자제어장치의 여러가지 조건들 (표1)을 발표하고 이러한 조건에 맞는 제어장비를 Gould사 등에서 개발하여 판매하기 시작한 것이 그 효시로 볼 수 있다.

표 1. PC에 관한 10개 조건

① 프로그램 및 그 변경이 용이할 것.
② 보전이 용이할 것.
③ 릴레이식 제어반보다 현장에서의 신뢰성이 높을 것.
④ 릴레이식 제어반보다 소형일 것.
⑤ 중앙 데이터 처리장치로 데이터를 직송할 수 있을 것.
⑥ 릴레이, 반도체방식과 경제적으로 경합될 수 있을 것.
⑦ 입력은 AC115V를 받을 수 있을 것.
⑧ 출력은 AC115V, 2A이상으로 솔레노이드 밸브, 모터 스타트 등을 조작할 수 있을 것.
⑨ 확장이 가능할 것.
⑩ 메모리는 최대 4K Word까지 확장될 것.

70년대초 마이크로 프로세서의 출현과 새로운 전자부품의 개발은 기존의 전자제어장치의 기능을 더욱 다양화, 범용화시켰고 이들은 기존의 릴레이 제어반에 비해 경제성, 신뢰성, 편이성 등 여러가지 면에서 장점을 가지게 되어 자동차, 철강, 화학 등 대형 공장에서 소형 기기에 이르기까지 급격한 수요 증가를 창출하였다. 단순한 시퀀스 제어장치에서 출발하여 연산, 계수기 등 컴퓨터의 일부 기능까지 갖추게 된 이 제어장치를 NEMA (National Electrical Manufacturing Association)는 1978년 정식으로 PC라는 이름을 채택하고 다음과 같이 PC를 정의하게 되었다.

"A Programmable Controller is defined as : A digitally operating electronic apparatus which uses a programmable memory for the internal storage of instructions for implementing specific functions such as logic, sequencing, timing, counting and arithmetic to control, through digital or analog input/output modules, various types of machine or processes. A digital computer which is used to perform the functions of a programmable controller is considered to be within this scope. Excluded are dru and similar mechanical type sequencing controllers."

최근 들어 PC는 더욱 고기능, 다기능화 되고 있으며 이미 중형이상의 PC시스템에서는 Analog 신호 처리 기능이 구비되어 있고 PID제어 기능도 대부분의 PC에서 제공하고 있는 실정이다. 제어 시스템이 분산화, 계층화 됨에 따라 네트웍 기능 보유 또한 PC의 필수적인 기능이 되었다. 종래 중·대형 PC에서 구비된 이

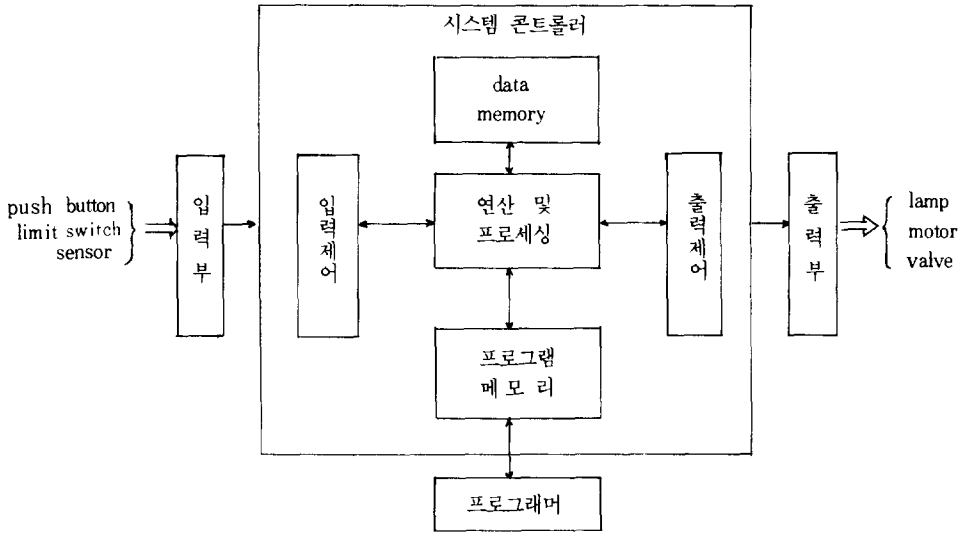


그림 1. PC의 내부 구조

러한 기능들도 이젠 점점 소형 기종에도 구비되고 있다.

본고에서는 일반적인 PC의 내부 구조 및 PC의 분류 방법을 고찰한다. 또 공정 제어 시스템에서의 PC의 역할 및 네트웍 기능에 대해서 알아본다.

2. PC의 구조 및 기능

그림은 간략화된 PC의 구조이다. 그림에서 보듯이 PC는 시스템 콘트롤러, 프로그래머, 그리고 입출력부로 구성된다.

일반적으로 시스템 콘트롤러는 연산 및 프로세싱부, 메모리부, 입출력제어부를 가지는데 전체 시스템의 관장, 입출력 데이타의 처리, 제어 프로그램의 처리 등과 같은 기능을 수행한다. 프로그래머는 제어 프로그램을 작성하여 시스템 콘트롤러로 전달하거나 시스템 콘트롤러 내부 상태의 감시, 시스템 콘트롤러 내부의 제어 프로그램 검색 등과 관련된 기능을 수행한다. 입출력부는 시스템 콘트롤러의 입출력제어부와 실제 공정 간의 입출력 데이타를 교환시키는 부분이다. 이를 위해 입출력부는 외부 신호와 PC내부 신호 사이의신호변환을 위한 회로를 제공하게 된다. 이 외에 PC는 상용전원을 공급받아 PC내부에서 사용하는 전원으로 변환해 주는 전원부 및 이상시와 정전 복구 기능을 위한 백업 모듈 등을 제공한다.

2.1 PC의 하드웨어

1) 시스템 콘트롤러

시스템 콘트롤러는 기본적으로 마이크로프로세서에 의한 컴퓨터 구조를 가지고 있으나 PC의 실시간 동작을 위해 실시간 클럭, 실시간 인터럽트 처리 기능, 시스템의 고장에 대비한 자기 진단 기능 등이 추가된다. 또한 비상 정전 등의 원인에 의하여 전원 공급이 중단되었을 경우, 시스템의 계속 동작과 빠른 복구를 위하여 시스템내의 메모리의 내용을 보존하도록 전원 보호 회로를 구비하여 돌발적인 사고에 대비한다.

-연산 및 프로세싱부

소형 PC인 경우 주로 단일칩 마이크로프로세서 혹은 8비트 마이크로프로세서가 주로 사용된다. 중형 PC인 경우 8비트 마이크로프로세서가 주로 사용되지만 16bit 마이크로프로세서가 사용되는 경우도 있다. 대형 이상의 시스템에서는 주로 16bit 마이크로프로세서 혹은 Bit Slice로 구성된 전용 마이크로프로세서가 사용된다. 대형 PC인 경우 범용 마이크로프로세서의 연산속도가 느리고 기능이 불충분한 경우에는 Bit Silce로 구성된 프로세서의 사용이 유리하지만 최근의 고기능 범용 마이크로프로세서를 이용할 경우엔 개발의 용이성 등을 비롯하여 여러가지 장점을 가지므로 오히려 범용 마이크로프로세서의 사용이 유리하다고 하

겠다.

PC의 규모가 커짐에 따라 연산 및 프로세싱부의 형태도 조금씩 달라지게 된다. 소형인 경우 하나의 프로세서로써 요구되는 기능을 모두 처리할 수 있지만 중형 이상의 시스템에서는 처리해야할 기능이 너무많아져서 하나의 프로세서로선 고속의 처리를 하기가 곤란해진다. 기능의 고속 처리를 위해서 시스템에 따라서는 스퀀스 제어 프로그램 중 AND, OR와 같은 단순 논리 프로그램을 처리하기 위한 특수 하드웨어 혹은 논리 프로세서를 가지기도 한다. 대형 이상의 PC에서는 거의 공통적으로 이러한 특수 하드웨어를 가진다. 더구나 최근엔 PC의 신뢰성 및 프로세싱 속도를 향상시키기 위해 multiple CPU의 사용이 제안되기도 한다[1] 입출력 접점 8192점을 처리할 수 있는 TI565 대형 PC인 경우 1개의 Discrete CPU, 1개의 Arithmetic CPU, 그리고 4개의 I/O processor로 구성된 시스템 콘트롤러를 가진다.

-입출력 제어부

입출력 제어부도 PC 규모의 크기에 따라 형태가 달라진다. 소형 및 중형 PC인 경우는 대부분 범용 마이크로프로세서로 구성된 연산 및 프로세싱부가 그 시스템에 적당한 하드웨어 회로를 가지고 입출력 제어를 수행한다. 하지만 대형 PC에서는 입출력 제어만을 전담하는 I/O scanner를 가지는게 보통이다. 독자적인 CPU를 가지는 I/O scanner는 입출력부로부터 전달된 입력 데이타를 수집하여 연산 및 프로세싱부로 전달하고 연산 및 프로세싱부로부터 전달된 출력 데이타를 입출력부로 전달한다.

이렇게 함으로써 입출력 신호에 관계된 기능을 I/O scanner가 처리할 동안 연산 및 프로세싱부는 제어프로그램 처리 혹은 시스템의 관리에 필요한 작업을 수행하여 전체 시스템의 속도를 향상시킬 수 있게 된다.

또 대형 시스템인 경우 입출력 제어부와 입출력부 사이에 remote 데이타 처리 기능을 제공한다. 이런 경우 입출력제어부는 수km 떨어진 곳에 있는 입출력부와 입출력데이타를 교환할 수 있게 되는데 주로 1메가 bps 정도의 고속 모뎀을 이용하거나 광 케이블을 이용한 광통신이 사용된다. 광 케이블을 이용할 경우 잡음 등에 의한 영향을 줄일 수 있고 고속의 데이타 전송이 가능하므로 이의 이용이 확대될 전망이다.

-메모리부

PC의 메모리부는 주로 다음과 같이 분류될 수 있다.

① 시스템 프로그램용 메모리(ROM)
② 스크래치 패드(RAM)
③ 응용 프로그램용 메모리(RAM)
④ 데이타 메모리(RAM)

이중 ROM에 내장되어 있는 시스템 프로그램이 PC의 시스템 콘트롤러가 수행해야 할 기능에 필요한 모든 프로그램을 가지고 있다. 스크래치 패드는 시스템 프로그램의 수행 중에 필요한 여러가지 데이타를 저장하는 영역이다. 응용 프로그램 메모리에는 사용자가 작성하여 입력한 제어 프로그램이 저장된다. 데이타 메모리는 다시 다음의 네 가지로 나뉘어질 수 있다.

① Input 데이타 메모리
② output 데이타 메모리
③ 내부 제어 릴레이 메모리
④ 레지스타 메모리

Input, output 데이타 메모리는 외부의 입출력 장치의 ON/OFF와 관련된 bit 데이타 정보를 저장하는 영역이다. 내부 제어 릴레이 메모리는 스퀀스 제어 프로그램의 수행 중간에 필요한 ON/OFF정보를 저장하는 영역이다. 제어 릴레이의 ON/OFF 정보는 외부로 보내지는 것이 아니고 프로그램 내부에서만 필요한 정보이다. 레지스타 메모리는 바이트 혹은 워드 형태의 데타를 저장하기 위한 메모리이다. 즉 제어 프로그램 내부의 타이머 혹은 카운터값 그외 산술 연산 등에 필요한 데이타를 저장한다.

2) 프로그래머

프로그래머로서는 소형 PC인 경우 LCD를 표시부로 한 handy 프로그래머가 주로 사용된다. 이 때 프로그래머의 기능에 필요한 소프트웨어와 전원은 시스템콘트롤러가 제공하는 경우가 많다. Gould 084 소형 PC인경우는 단일 칩 프로세서를 독자적으로 가진 프로그래머를 사용하고 있다. 이런 경우 프로그래머에 필요한 전원은 시스템 콘트롤러가 제공하지만 프로그래머는 자체의 기능을 처리하기 위한 독자적인 소프트웨어를 가지고 시스템 콘트롤러와는 serial interface로 데이타를 교환한다. 소형인 경우 프로그래머에서 작성된 제어 프로

그램은 그 ROM에 내장되어 시스템 콘트롤러에 정착될 수도 있고 바로 시스템콘트롤러 내부의 응용 프로그램 영역에 저장될 수도 있다. 후자의 경우엔 전원 공급이 중단되면 백업 배터리에서 전원을 공급하여 제어프로그램이 지워지지 않게 되어 있다.

중·대형 PC인 경우 주로 stand alone computer형태의 그래픽 프로그래머를 사용하며 시스템 콘트롤러와는 RS232c로 통신하며 데이타를 교환한다. 이 경우 시스템 콘트롤러와 프로그래머 사이엔 약속된 통신 프로토콜이 사용되며 이는 각 PC마다 다른 형태를 가지므로 다른 PC시스템 사이에는 호환이 불가능하다.

중·대형 PC인 경우에도 현장에서 손쉽게 사용할 수 있도록 하기 위해 handy graphic 프로그래머가 제공되기도 한다. 최근에 들어서는 IBM-PC에서 PC의 프로그래머 기능을 대신할 수 있는 소프트웨어를 개발, 사용하는 경우가 많아졌고 점점 그 사용이 증가할 것으로 보여진다.

3) 입출력부

입출력부는 공정에서의 ON/OFF 신호 혹은 아날로그 신호 등을 시스템 콘트롤러와 교환하는 부분이다.현장에서의 신호가 PC내부의 신호로 바로 사용될 수는 없으므로 입출력부는 신호 변환회로, 잡음제거 회로,현장 신호와 PC내부 신호와의 절연 회로 등을 포함하고 있어야 한다. 이외에 출력부는 출력단의 short로 인한 과전류 방지 회로를 내장해야 한다.

PC에 연결되는 입출력기기는 간단한 push button 스위치에서 NC 장치나 컴퓨터에 이르기까지 다양한 형태를 가진다.

이러한 입출력기기는 기기에 따라서 사용하는 정격, 형식들이 서로 다르므로 PC는 다양한 입출력 모듈을 구비하고 있어야 한다.

표 2,3은 많이 사용되는 입출력기기와 그들의 정격들이다.[2) 3)]

이상에서 언급한 일반적인 입출력장치 이외에도 특수한 작업환경을 위한 Intelligent I/O에 대한 요구가 점점 증가하고 있고 Intelligent I/O의 구비 여부가 제품의 선호도를 결정하는데 중요한 요소가 되고 있다. 다음은 이미 상품화되었거나 상품화되고 있는 Intelligent I/O의

표 2. Discrete 입출력기기와 표준정격

Devices		Standard Rating	
Inputs	Outputs	Input	Output
Selector Switches	Alarms	24 volts AC/DC	12-48volts AC
Push buttons	Control Relays	48 volts AC/DC	120 volts AC
Photoelectric Eyes	Fans	120volts AC/DC	230 volts AC
Limit Switches	Lights	230volts AC/DC	12-48volts DC
Circuit Breakers	Horns	TTL levels	120 volts DC
Proximity Switches	Valves	Non-Voltage	contact (Relay)
Motor Starter Contacts	Motor Starter	Isolated Input	Isolated Output
Relay Contact	Solenoid		TTL level

표 3. Analog입출력기기와 표준정격

Devices		Standard Rating	
Inputs	Outputs	Input	Ouput
Temperature transducer	Analoglvalves & actuators	4−20mA	4−20mA
		0−+1 volts DC	10−50mA
Pressure transducer	Chart Recorders	0−+5 volts DC	0−+5 volts DC
		0−+10volts DC	0−+10volts DC
Load cell transducer	Electric Motor Devices	1−+5 volts DC	±2.5 volts DC
		±5 volts DC	±5 volts DC
Humidity transducer	Analog Meters	±10volts DC	±10volts DC
Flow transducer			
Potentiometers			

일례이다.

- RTD input module, Thermocouple input module
- Network interface module, Fiber optic interface module
- Stepper motor controller, Positioning cotrol module
- Encoder / counter module, High speed counter
- BASIC module, BCD I/O module

이러한 I/O 모듈은 주로 시스템 콘트롤러로부터 다운로드된 동작 모드에 따라 독립적으로 주어진 기능을 수행하게 된다. Intelligent I/O의 도입은 특별히 큰 투자없이 PC의 기능을 한층 확대시킬 걸로 기대된다.[4]

2.2 PC의 소프트웨어

PC의 소프트웨어는 프로그램이 있는 위치에 따라 시스템 콘트롤러 소프트웨어, 프로그래머 소프트웨어, 입출력 소프트웨어로 나뉠 수 있다. 가장 간단한 형태의 PC에서는 이 모든 소프트웨어가 시스템 콘트롤러 한 곳에 있게 된다. PC의 규모가 커지고 기능이 복잡해짐에 따라 각각의 프로그램은 서로 분리되어 할당된 고유의 작업을 수행하게 된다. 프로그램이 서로 분리되어 있는 경우엔 상호간에 즉 시스템 콘트롤러와 프로그래머간 그리고 시스템 콘트롤러와 입출력부 사이에 데이타를 교환하기 위한 통신 프로그램이 있어야 한다. 이러한 통신 프로그램은 정해진 프로토콜에 따라 서로의 데이타를 교환한다.

입출력부의 소프트웨어는 주로 시스템에서 요구하는 방식대로 입력 데이타를 수집하고 출력 데이타를 내보내는데 국한된다. 따라서 본고에서는 시스템 콘트롤러와 프로그래머의 소프트웨어의 기능에 대해서만 설명한다.

- 시스템 콘트롤러의 소프트웨어

PC의 크기에 따라 또 제품이 제공하는 기능에 따라 조금씩 차이가 있지만 대형 PC인 경우 다음에 열거한 내용들이 주로 수행이 되고 소형 PC인 경우 이 중 일부의 기능은 수행되지 않는다.

- 시스템의 초기화 작업
- 시스템 관리
- 시퀀스 제어 프로그램 수행
- 입출력 데이터 처리
- PID 프로그램 수행
- 프로그래머와의 통신
- 자기 진단
- Hot 백업
- 네트워크 관련작업
- 특수 입출력 모듈 처리
- 기타

이러한 기능을 수행하기 위한 시스템 콘트롤러의 소프트웨어는 ROM에 내장되어 시스템 콘트롤러의 CPU 보드에 위치한다. PC는 실시간 동작이 요구되는 시스템이므로 이를 동작시키기 위한 소프트웨어도 실시간운영 체제 형태를 갖추어야 한다. 이를 위해서 PC의 시스템 콘트롤러 소프트웨어는 여러가지 종류의 인터럽트를 쉽게 처리할 수 있는 구조가 되어야 할 것이다.[5][6]

시스템 콘트롤러의 동작 형태는 입력을 읽어들이고 제어프로그램을 수행하고 출력을 내보내는 일련의 작업을 반복하는 것으로 간략화시킬 수 있다. 이러한 작업을 한번 수행하는데 걸리는 시간을 일반적으로 스캔타임이라고 한다. 시스템에 따라서 또 수행하는 제어 프로그램의 길이에 따라서 차이가 있으나 대개 20~100ms가 소요된다. 논리 프로그램을 풀기 위한 logic processor, multiple CPU의 도입 등은 scan time의 고속화를 위해서 도입된 기법들이다.

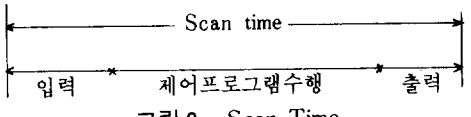


그림 2. Scan Time

처음 전원이 인가되면 시스템 소프트웨어는 시스템의 자기진단을 수행하고 여러가지 하드웨어 소자와 필요한 메모리 영역의 초기화 작업을 수행한 후 스캔을 반복하게 된다.

그러나 PC가 수행해야 할 기능이 많아짐에 따라 스캔의 구조도 좀 더 복잡해진다. 다음의 그림3은 대형 PC의 스캔 구조의 일례이다.[7] 따라서 제어 프로그램의 양이 많을수록, 처리해야할 기능이 많을수록, monitoring 등에 의한 프로그래머와의 통신량이 증가할수록 스캔 타임은 늘어남을 알 수 있다. 몇몇 시스템에서는 사용자가 원하는 스캔 타임을 명시할 수 있도록 하여 그림3에서 중요성이 덜한 뒷부분의 작업은 생략되기도 한다.

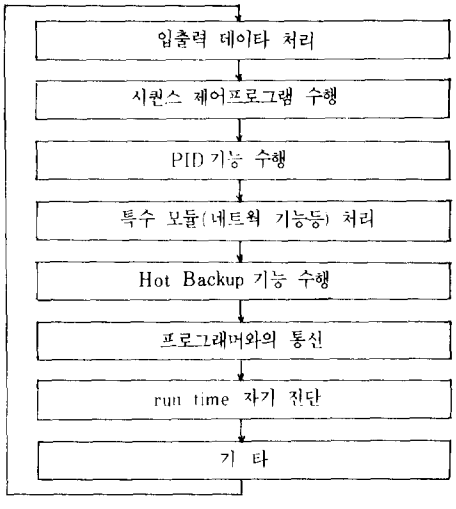


그림 3. 대형 PC의 스캔 구조의 일예

PID제어 기능의 수행 방법은 PC마다 조금씩 차이가 있다. 중·대형 PC인 경우 대부분 제공하고 있고 최근 들어 소형 PC에서 제공하기도 하는 PID 제어기능은 대체적으로 다음의 3가지 형태로 구현된다.

① Intelligent I/O 모듈의 형태로 된 PID 제어모듈
② 래더언어의 블록형태의 명령어를 이용하는 방법
③ PID 제어 기능을 위한 특별한 프로세서를 사용하는 방법

①의 경우 PID 모듈 자체에 A/D, D/A 변환기를 가지고 독자적으로 PID제어 기능을 수행한다. 시스템 콘트롤러와는 P, I, D계수, 샘플링 타임 등의 간단한 정보만이 교환된다.

②의 경우 래더 언어의 수행 중간에 PID 제어가 함께 수행된다. 이 때 필요한 A/D, D/A변환기는 입출력 모듈로서 입출력부에 존재한다.

③의 경우는 래더 언어 등에 의한 시퀀스 제어 프로그램과 PID 제어 프로그램을 수행하는 프로세서가 별도로 있어서 두 가지 작업의 동시에 수행된다. PID 제어에 필요한 A/D, D/A 변환기는 입출력 모듈로서 입출력부에 존재하고 시퀀스 제어 프로그램을 처리하는 master 프로세서와 PID제어 프로세서는 서로 필요한 정보를 교환한다.

- 프로그래머 소프트웨어

프로그래머는 제어 프로그램을 작성하여 시스템 콘트롤러로 전달하는 것을 주임무로 하고 있다. PC에서 사용되는 제어 언어는 래더 언어를 위시하여 여러가지의 종류가 있으나 다음의 4가지가 주요한 언어들이다(IEC 안)

- 텍스트 언어 (Textual Language)
  - IL 언어 (Instruction List)
  - ST 언어 (Struetured Text)
- 그래픽 언어 (Graphical Language)
  - LD 언어 (Ladder Diagram)
  - FBD 언어 (Functional Block Diagram)

그림 4는 이들 언어를 사용하여 동일한 제어 프로그램을 작성한 예이다.

	예	언어
(a)	%IX1 %MX3 %QX17 %MX10 C EN OK LT (5)	LD 언어
(b)	%IX1 %MX3 & QX17 FF28 S8. X SR S1 MX 10 EN OK LT	FBD 언어
(c)	QX17=%IX1 AND %MX3 AND S8. X; FF28(S1 : = (C<D)) : %MX10 : =FF28.Q;	ST 언어
(d)	LD S8. X LT D AND %fX1 S1 FF28 AND %MX3 LD FF28 Q ST %QX17 ST %MX10 LD C	IL 언어

그림 4. 제어언어 사용예

위의 4가지 언어 중 LD 언어는 가장 일반적으로 사용되는 제어 언어이다. 초기에는 간단한 논리와 Timer, Counter 등을 처리하는데 국한되었으나 그 후 data 처리, program 제어 등의 복잡한 기능도 처리할 수 있게 확장되었고 제품마다 제공하는 기능과 동작 형태는 조금씩 차이가 있다. IL 언어는 AND, OR, STR등과 같은 몇 개의 간단한 Mnemonic을 가지고 시퀀스 제어프로그램을 표현한다. IL 언어는 제공할 수 있는 기능이 제한되어 있고 주로 소형 PC의 프로그램 언어로써 사

용된다. FBD언어는 특정 기능을 블록으로 표시하고 입출력 제어 조건을 표기하는 방식이다. 최근의 LD 언어는 초기의 LD언어에 FBD언어에서 제공하는 data 변환, parameter 설정, block data 이동 등 고급 기능을 합한 것과 유사한 형태를 가진다. ST 언어는 일반적인 컴퓨터에서 사용하는 고급언어와 유사하며 여러가지 복잡한 기능의 처리가 가능하므로 대형 PC의 언어로 사용이 된다.

프로그래머는 제어 프로그램을 작성하여 시스템 콘트롤러로 전달하는 것 이외에도 시스템 콘트롤러와 통신하면서 다음과 같은 기능을 제공하여야 한다.

- 시스템 configuration 프로그램 : 시스템 콘트롤러의 동작 모드에 관계된 여러가지 초기 데이타값을 설정하게 한다.
- 감시 프로그램 : 시스템 콘트롤러 내부의 상태 및 제어 프로그램의 현상태를 감시할 수 있게 한다. LD언어로 된 제어 프로그램인 경우 Discrete 요소의 ON/OFF 상태 및 register형 요소의 현재값 등을 LD언어 프로그램상에 표시한다.
- 검색(Debugging) 프로그램 : 시스템 콘트롤러 내부에 있는 스퀀스 제어 프로그램을 PC 외부와 직접 입출력 데이타를 교환하지 않고 프로그래머와만 통신하면서 수행시켜 보도록 하여 제어 프로그램 상의 잘못을 미리 점검할 수 있게 한다.
- 프린팅 프로그램 : 제어 프로그램을 line printer로 출력시켜 프린팅할 수 있게 한다.

이 외에 시스템에 따라 제공하는 기능이 달라질 수 있다. PID 기능의 처리가 가능한 PC인 경우 PID 제어 프로그램의 작성에 필요한 기능이 제공되어야 한다.

3. PC의 분류

PC는 입출력 접점수가 수십개 정도인 소형 PC에서 수천개의 입출력 신호를 처리할 수 있는 대형PC에 이르기까지 서로 차이점을 가지는 여러가지 종류가 있다. PC를 분류하는 일반적인 규칙은 없으나 주로 시스템이 처리할 수 있는 입출력 접점수에 의하는 것이 상례인데 그림5와 같이 분류할 수 있다.[3]

그러나 PC의 기능이 다양화, 고도화됨에 따라 단순한 접점수 비교만으로선 시스템의 특성을 정확히 파악하기가 곤란해졌다. 다음에 열거된 항목은 PC의 특성

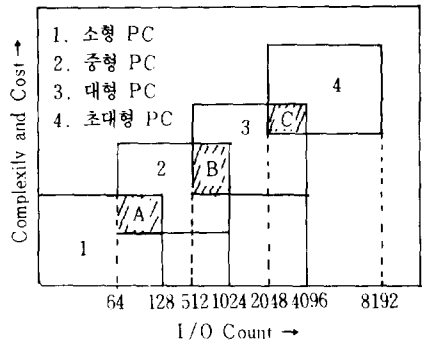


그림5. 입출력 접점수에 따른 PC의 분류

을 비교하기 위한 기준으로 사용될 수 있을 것이다.

① 입출력부문
- 총 입출력 접점수
- 입출력 모듈의 종류
  - Discrete 입출력 :
    5V/12V/24V/48V DC 입출력
    110V/220V AC 입출력
    TTL level 입출력
  - Analog 입출력
    4~20mA 입출력
    0~5V, 0~10V, 1~5V, ±5V, ±10V 입출력
- 리모트 입출력 기능(중형 이상)
- Intelligent 입출력 기능(중형 이상)

② 프로그램 언어
- 래더 혹은 블록 언어 기능
  - 명령어
    Timer/Counter/Shift Register (TCS)
    연산 명령어 종류
    데이타 처리 명령어 종류
    Drum Timers or Sequencer
    Subroutine 명령어(중형 이상)
    Jump 명령어(중형 이상)
    Matrix Function 처리 명령어(중형 이상)
    ⋮
- 고급 언어 기능(중형 이상)

③ 스캔 타임

④ PID 제어 기능(중형 이상)
- PID 모듈 형태
- system software PID 형태

⑤ 메모리

- 최대 기억 용량
- 최대 step 수
- Error detection/correction
- 메모리 종류
- 백업 기능
- 사용자 configuration 여부(중형 이상)

⑥ RS 232/422 접속 기능

⑦ 자기 진단 기능

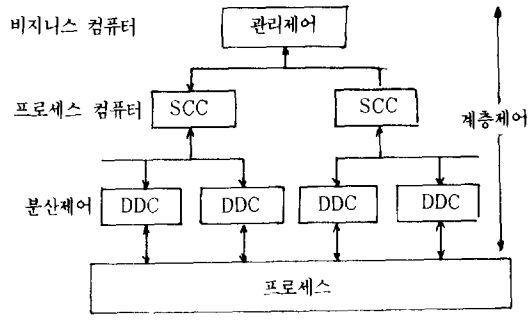
⑧ 프로그래머
- 형태 : -LCD type (소형)
  -Graphic Programmer (중형 이상)
  : Digital Cassette Tape
  Floppy Disk
  -IBM PC 사용 가능 여부(중형 이상)
- 기능 : -제어 프로그램작성
  -시스템 감시
  -디버깅
  -자기 진단
  -ROM 프로그래밍

⑨ LAN 기능(중형 이상)
- 방식-MAP 채택 여부
- 통신 속도
- 통신 거리
- 최대 접속 node 수

⑩ Hot Backup 기능(중형 이상)
- Auto update to backup
- User programming 필요
- Voting system approach

⑪ power supply options

이상의 항목에서 입출력 접점수가 대략 128점까지인 소형 PC의 경우는 ON/OFF 제어를 위한 기능을 위주로 간단한 연산기능, TCS 기능 등을 가지며 아날로그 신호를 처리할 수 있는 소형 PC로 늘어가고 있다. 중형 이상의 시스템인 경우 제공하는 기능을 명확히 구분하는 것이 무리이지만 일반적으로 시스템이 커지는 것에 비례하여 아날로그 신호 처리 기능이 확대되고 좀 더 고급 연산 기능이 추가되며 PID 기능을 제공하게 된다. 프로그래머로서는 주로 그래픽 프로그래머가 사용되며 네트워크 기능이 대부분 가능해진다. 그 외에 Intelligent 입출력을 제공하는 PC가 많으며 시스템의 신뢰성을 높이기 위해 Hot Backup 기능을 제공하는 등



DDC :Direct Digital control
SCC : Supervisory Computer Control

그림 6. 디지탈 계장 제어 시스템의 구성도

의 Fault-tolerant 기능이 시스템의 확장에 비례하여 강화된다.

4. PC의 네트웍 기능과 공정 제어 시스템

그림 6은 계층화와 분산화로 특징지어지는 최근의 디지탈 공정제어 시스템의 구성도이다. 여기서 각 계층은 다음과 같은 기능을 분담한다.

1) 비지니스 컴퓨터 :
- 공정간 조정
- 실적 수집
- 작업 지시, 작업 진행 관리
- 2개 이상의 공정 간의 조정 (예 : 원료 집중 관리)
- 공정 내의 초대형 모델 관리 (예 : 환경 확산 모델)

2) 프로세스 컴퓨터 :
- 대상 공정 설비의 제어 (모델링, 설정치 계산)
- 대상 공정 설비의 고유 데이타 처리(조업 데이타의 실시간 처리, 실적 수집)
- 대상 공정 간의 조정 및 감시

3) DDC :
- PC, 단일 루프 제어기, 로보트, NC 머쉰 등
- 단순 모델의 제어
- Data Hiahway에서 프로세스 컴퓨터 혹은 다른 제어기들과 정보 교환

그림 6에서와 같이 분산화, 계층화된 공정 제어 시스템에 네트웍 기능은 필수적임을 알 수 있다. 네트웍

표 4. 대표적인 PC의 data highway 특성비교

Manufacturer	Highway Name	Max. Nodes	Max. Length(ft.) Max. Baud Rate	Baud Rate (max)	Access Method	Comments
Allen-Bradley	Data Highway	64	10,000	56K	Token	
GTE Sylvania	Control Net	254	5,000	1 M	Token	X.35 Compatible Gateway
General Electric	GE net	999	15,000	5 M	Collision Detection	IEEE 802.3 Compatible
Gould-Modicon	Modbus	247	15,000	19.2K	Master-Slave	User 2 Twisted Pairs. Shielded
Gould-Modicon	Modway	250	15,000	1.544M	Token	
Industrial Solid State Controls	Copnet	254	32,000	115.2K	Master-Slave	Includes interfaces to Allen-Bradley. Gould-Modicon and Texas instruments PC's. HDLC Portocol.
Measurex	Data-Freeway	63	10,000	1 M	Collision Detection	
Rdiance Electric	R-Net	255	12 000	800K	Token	ASCII/X3.28/HDLC Gateway. Uses HDLC Framing
Square D	SY/Net	200	2,000	500K	Timed Token	Worst case access = 500msec with 50 PC's.
Texas Instruments	TIWAY I	254	10,000	115.2K	Master-Slave	HDLC Protocol
Texas Instruments	TIWAY II	2*	32,000	5 M	Token	Broadband IEEE 802.4 compatible
Westinghouse	WDPF	254	18,000	2 M	Token	100msec Fixed Access Time. 10,000 Points/Sec Throughput
Westinghouse	Westnet	50	10,000	1 M	Master-Salve	Gateway Interface to Westnet
Comments			Broadband networks cover unlimited distances using CATV repeaters.	Lower baud rates allow greater lengths		

기능은 PC와 PC간, PC와 다른 콘트롤러간, PC와 상위 컴퓨터 간에서 생산정보나 운전 제어 데이타의 송수신을 비롯하여 프로그램의 업, 다운 로드를 간편하게 하고 정보의 신속한 수집, 정리 판단을 기능케 한다. 다음의 표 4는 대표적인 PC 업체로부터의 data-highway 기능을 비교하여 나타낸 것이다.[3]

구체적으로 PC에서의 네트웍 기능이 어떤 식으로 이루어지는지를 보기 위해 HDLC protocol을 사용하고 있는 표 4의 TIWAY I에 대해서 간단히 살펴본다. 그림 7은 TIWAY I의 네트웍 구성도이다.

그림 7에서 NIM (Network Interface Module)은 Intelligent 입출력 모듈의 한 종류로 입출력부에 존재한

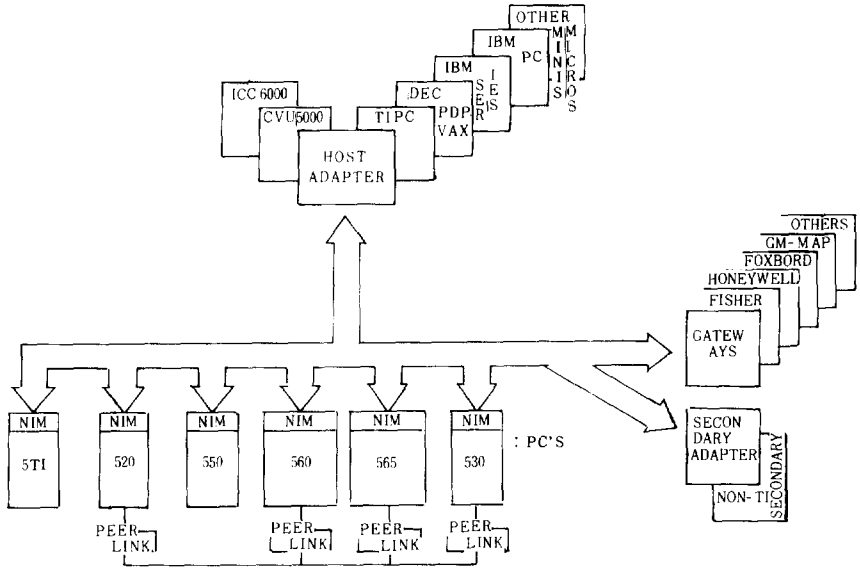


그림7. TIWAY I 네트웍 구성도

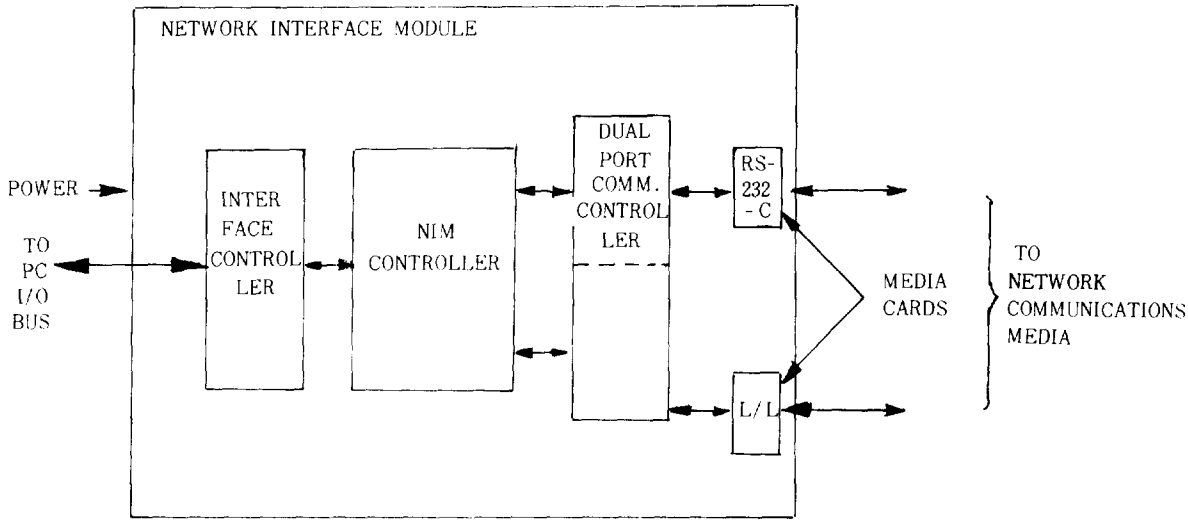


그림8. NIM의 내부 구조

다. NIM의 내부 구조는 그림8과 같다. Host Adapter는 상위 컴퓨터와 RS-232C로 연결되고 TIWAY I 의 master controller로서 TIWAY I을 관리한다. 그림9는 Host Adapter의 내부구조이다.

Host에서 특정한 번지를 가진 PC에 명령을 내릴 때 일어나는 동작순서는 다음과 같다.

먼저 Host가 필요한 명령을 Host Adapter로 보내면 Host Adapter는 정해진 전송 프로토콜을 만들어 media로 보낸다. 지정된 번지를 가진 NIM은 이 전송 프로토콜에 맞춰 전달된 데이타 중에서 명령에 관계된 데이타만을 PC로 전달한다. 그러면 PC는 주어진 명령을 수행한다. 명령이 PC내부의 데이타를 원하는 것

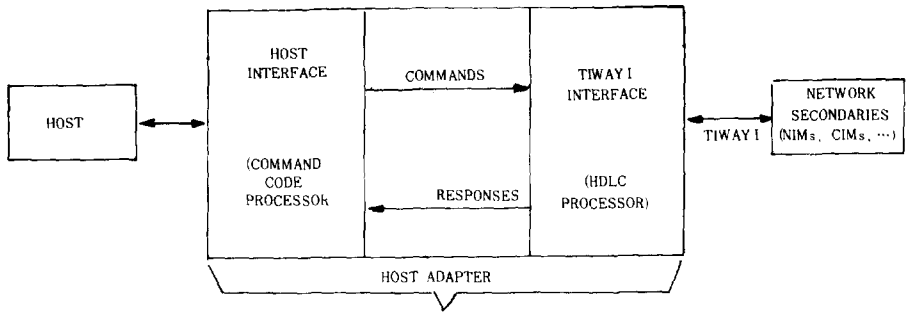


그림 9. HOST ADAPTER의 내부 구조

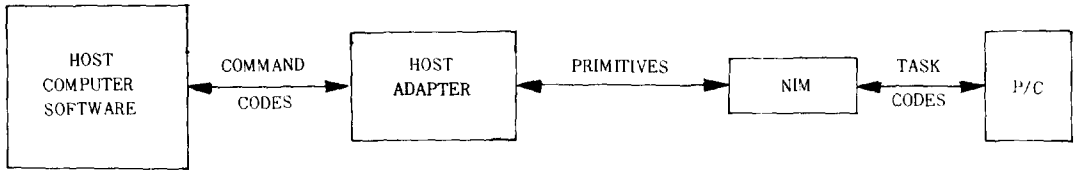


그림 10. 네트웍 통신 Level

일 경우엔 그 데이타를 정해진 프로토콜에 맞추어 다시 Host로 보내준다. 그림 10은 네트웍 통신 level을 표시한다.

표 4에서 보듯이 PC에서 사용하는 네트웍은 업체마다 서로 다른 형태를 가지고 있다. 그러나 최근 들어 공장형 네트웍에 대한 표준화 작업이 활발히 전개되고 있고 그 중 GM에서 제안한 MAP (Manufacturing Automation Protocol)으로의 표준화가 가장 유력시되고 있다. 유수의 PC업체들도 자사의 기존 네트웍을 포기하고 이미 MAP의 채택을 발표하고 있는 실정이므로 장래의 PC에는 MAP에의 접속 가능성 여부가 중요한 요소가 될 것으로 보인다.

5. PC의 전망

컴퓨터가 양호한 제어환경에서는 비교적 쉽게 기존의 PC가 수행하던 기능을 대체가능함에 따라 PC도 컴퓨터가 가질 수 없는 특성의 소유가 필수적이 되었다. 사용자들은 좀 더 강인한 시스템을 요구하고 있고 fault tolerant capability는 PC가 갖추어야 할 중요한 특성이 되었다. PC는 이를 위해 Real-time backup 능력, 자기진단기능 등을 향상시키고 있고 잡음의 제거에 노력을 기울이고 있다. redundancy 보장과 voting system의 도입[6] 등도 fault tolerant capability를 위한 노력들이다.

PC가 FA시스템의 핵심적인 요소가 됨에 따라 네트웍 기능의 보유가 요구되고 중·대형 PC는 대부분이 기능을 보유하고 있다. 하지만 표준화되지 않은 네트웍 기능은 PC의 수요 확산을 저해하는 요소가 되었다. 최근 GM사에서 제안한 MAP (manufacturing automation protocol)은 공장 자동화를 위한 표준 네트웍으로서 점점 더 많은 호응을 얻고 있으며 이미 많은 굴지의 PC 메이커에서 고유의 네트웍을 포기하고 MAP의 채택을 발표하고 있으므로 MAP이 표준 네트웍이 될 것이 확실시 되고 있다. 따라서 MAP 기능의 보유는 제어시스템의 정보 집중화 및 관리의 효율화 등과 연관해서 필수적이라 할 것이다.

이 외에 좀 더 손쉽게 사용할 수 있는 제어언어의 개발, 고속 동작을 위한 다중 프로세서 기법의 도입 및 특수 하드웨어 개발, 다양한 Intelligent I/O개발, Graphic 기능의 도입에 따른 man-machine interface의 강화 등이 현재 PC의 고도화를 위해 진행 되고 있는 작

업들이다.

이러한 작업을 통하여 PC는 사용자가 요구하는 복잡다양한 요구를 만족시키며 기능확대에 따른 새로운 수요의 창출과 함께 현대의 복잡한 제어 환경에서 유용하게 사용될 수 있는 필수적인 기기로 성장할 것이다.

참 고 문 헌

1) H. F. Felder and G. A. Tendulkar, "Multiprocessing boosts PC performance,"I&CS-The Industrial & Process Control Magazine, Jan. 1985.

2) "Programmable Controller", 월간 자동화 기술, 1987년 7월~1988년 1월

3) 김정호, 하정현, 채영도, 조삼현, "공정 제어용 프로그램식 제어기(programmable Controbler)에 관한 고찰, "전자공학회지, vol. 13, no. 2, 1986년 4월

4) M. E. Wylie, "Intelligent I/O : New Smarts for PC's", The Industrial & Process Control Magazine, Jan. 1985.

5) 권 욱현, 김 종일, 최 한홍, 김 덕우, 추 영열, 정 범진, 이 기원, 홍 진우 "대형 프로그래머블 콘트롤러의 개발 : Part I, Hardware", 1987 한국 자동제어 학술회의 논문집, vol. 1, pp. 407-412, 1987년 10월

6) 권 욱현, 박 홍성, 변 대규, 최 한홍, 김 덕우 "대형 프로그래머블 콘트롤러의 개발 : Part II, Software", 1987 한국자동제어 학술회의 논문집. vol. 1, pp. 413-418, 1987년 10월

7) TI Model 560 and 565 product/sales training, Rev 1.0, Feb. 1985

8) G. J. Blickley, "New Developments in Pro - grammable Controllers and Peripherals", Control Eng., Jan. 1987