

# PLC 관련 기술의 동향과 전망

權 旭 鉉\*, 朴 宰 賢\*\*, 張 來 赫\*\*\*  
서울大學校 電氣工學科\*  
仁荷大學校 自動化工學科\*\*  
서울大學校 制御計測新技術研究센터\*\*\*

## I. 서 론

일반적으로 제어 시스템은 센서, 제어기, 구동장치 등으로 구성되며, 제어 신호의 형태에 따라 아날로그 제어, 디지털 제어, 순차제어로 구분된다. 이들 중 순차제어는 '참' 과 '거짓'의 두 가지 상태로 표현되는 입출력 값의 논리적 조합에 의해 제어 동작을 결정하는 제어 방식으로서, 초기에는 순차제어를 구현하기 위해서는 이들 제어 요소 사이에 실제적인 연결을 전기적 배선이나 공유압의 배관 등을 통하여 구현하여왔다. 그러나 산업 구조가 발전함에 따라 생산 제품의 다품종, 소량 생산이 증가되고, 생산 기술의 지속적인 발전에 따른 제어 체계의 변화에 대한 요구가 증대됨에 따라, 기존의 하드와이어 방식의 순차 제어로는 이러한 요구에 부응하지 못함에 따라 새로운 구조의 순차제어기의 필요성이 대두되었다.

1968년 미국의 GM사에서는 자사의 순차제어 시스템을 구성하고 있는 제어요소와 그것들의 결선과정을 소프트웨어적으로 대체할 수 있는 개념을 정립하고 이를 위하여 10개항의 시방서를 제안하였는데 이 시방서의 규격에 따라 1969년 Gould사에서 최초의 프로그램 방식의 순차 제어기를 제작 판매하게 되었고, 1978년 NEMA(National Electrical Manufacturing Association)에서 이를 정식으로 PC(Programmable Controller)라 정의하였으며 현재는 PLC(Programmable Logic Controller)라고 널리 불린다. 이렇게 개발된 프로그램형 순차제어기(PLC)는 기존의 하드와이어 방식의 순차제어 장치를 대체하면서 꾸준히 발전되어 오늘날에는 공장에서 작업의 순서를 지시하거나 단위 기계의 동작을 제어하는 목적으로 사용되며 현대적인 공장 환경에서 가장 널리 사용되는 제어 요소로서의 자리를 확고히 하고 있다. PLC의 사용 예를 살펴보면, 하나의 자동차 조립라인을 제어하기 위하여 수천 개의 입출력 접점을 수 만 단위의 명령으로 제어하는 초대형 PLC로부터 주차장, 광고판 등의 소형 단위 기기를 제어하는 소형 PLC까지 주변 생활 환경에서 손쉽게 응용 예를

찾아볼 수 있다.

이러한 PLC는 처음 개발된 이후 기능적이나 응용 영역에서 많은 변화를 가져왔는데, 초기의 PLC는 단순한 순차제어를 위한 목적으로 개발되어 사용되었으나, 현대의 PLC는 순차제어는 기본이며 고수준의 아날로그 및 디지털제어 능력을 탑재하여 복합적인 제어 목적으로 널리 사용된다. 따라서, 현대의 PLC는 단순한 순차제어기라기보다는 모든 제어요소와, 실시간 제어용 컴퓨터 기술이 복합적으로 연결된 실시간 제어 시스템으로 보는 시각이 타당하다. 이와 같은 관점에서 PLC를 구성하는 요소 기술을 살펴보면, PLC의 핵심 장치인 고성능 마이크로 컴퓨터 관련 기술, 실시간 운영체제 및 시스템 관련 소프트웨어 기술, 입출력 신호처리 기술, 분산 PLC를 위한 네트워크 기술, 제어 언어 및 순차 제어 시스템 해석 및 설계 기술, 고수준의 안정성을 위한 내결함/내고장성에 관한 기술 등이 포함된다.

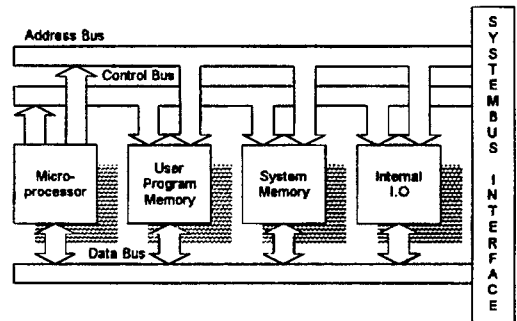
본 글에서는 순차제어의 기본 제어 요소인 PLC를 구성하는 중요 요소 기술을 PLC 개발자의 입장에서 컴퓨터 기술의 관점으로 광범위하게 언급한다. 아울러 최근의 PLC에 관련된 기술 동향, 향후 발전 방향 등을 살펴본다. 본 글을 통하여 PLC의 요소 기술의 동향과 전망에 대한 구체적인 이해 증진에 도움이 되고자 한다.

## II. 마이크로 컴퓨터 관련 기술

### 1. PLC의 중앙 처리부

PLC의 두뇌역할을 하는 중앙 처리부(MPU or CPU)는 프로그래머가 작성한 응용 프로그램을 해석하고 실행하기 위한 장치이다. 그 구성은 그림 1과 같이 마이크로프로세서, 메모리, 내부 입출력 기기, 그리고 외부 입출력 기기와의 연결을 위한 시스템 버스로 이루어져있다. 초기의 PLC가 릴레이 제어반을 대체하는 논리 연산과 입출력 처리 기능이 대부분이었던 것에 비해 현재의 PLC는 아날로그 제어를 비롯한 수치 연산 기능, 보다 편리

한 프로그래밍을 지원하는 디버깅과 테스트 기능, 분산된 시스템 구성을 위한 통신 기능이 부가되어 중앙 처리부는 갈수록 고기능성이 요구되고 있다. 또한, 복잡한 프로그램을 보다 빠른 시간 내에 실행하기 위해 고속의 중앙 처리부가 요구되고 있는데 이와 같은 고기능, 고속의 중앙 처리부를 개발하기 위해서 PLC 제작 회사들은 신제품들에 범용 컴퓨터에서 사용되는 마이크로 컴퓨터 관련 기술들을 도입하고 있다. 마이크로프로세서 개발 기술, 다중 프로세서 구조 기술들이 바로 이와 같은 기술들이다.



〈그림 1〉 PLC CPU 유닛의 구조

### 2. 마이크로프로세서 관련 기술

마이크로프로세서는 중앙 처리부의 핵심이며 특히 PLC의 성능을 결정하는 매우 중요한 장치이다. 초기의 제조업체들은 마이크로프로그래밍 기술을 이용하여 소위 비트 슬라이스라고 불리는 마이크로프로세서를 직접 설계하였다. 이 기술은 프로세서 기능을 위한 내부 요소인 레지스터, 프로그램 카운터, 연산기(ALU) 등을 직접 제어하는 일련의 매우 단순한 명령으로 PLC의 각 명령을 구현하는 것으로 고속의 논리 명령을 실행하는 중앙 처리부의 구현에 적합하다.

반도체 기술이 발전함에 따라 다양한 명령어 세트를 내장한 범용 마이크로프로세서들이 개발되었고 수치 연산 기능이 강화되어 아날로그 제어와 같은 고기능 명령의 구현이 가능하며, 범용 마이크로프로세서가 PLC의 중앙 처리부에 채택되었다. 마이크로프로그램을 개발하는 대신 주어진 마이크로프로세서의 명령어의 조합으로 PLC의 각 명령

을 구현하였지만 범용 마이크로프로세서의 장점인 낮은 가격과 개발 도구의 지원, 고급 언어의 사용 덕분에 짧은 시간 내에 저렴한 PLC를 개발할 수 있게 되었다. 이와 같은 방법은 특히 위와 같은 마이크로프로세서 개발 기술을 가지지 못한 후발 업체들에게는 매우 좋은 방법이었다. 범용 마이크로프로세서는 개인용 컴퓨터의 발전과 함께 급속히 개선되어 PLC의 성능 개선도 쉽게 이를 수 있었다. 그러나 범용 마이크로프로세서는 기본적으로 범용 컴퓨터에서 널리 쓰이는 워드 데이터 연산 및 이동을 위한 것이므로 PLC의 주된 명령인 비트 데이터의 논리 연산 및 이동 명령이 지원되지 않아서 기능의 향상에 비해 속도의 향상에 있어서는 큰 도움을 주지 못한다. 일반적인 컴퓨터에 비해 PLC는 실시간 특성이 증시되며 중앙 처리부가 각 명령을 실행하는 시간을 확정적으로 명시해야 한다. 최신의 범용 마이크로프로세서에서 성능 향상에 매우 중요한 역할을 하는 캐쉬 기술 등의 확률적 성능 향상 기술은 PLC에서는 사용되기 힘들다.

범용 마이크로프로세서의 단점인 논리 명령의 실행 속도를 개선하기 위해 논리 명령만을 처리하는 고속의 논리 처리기가 개발되어 범용 마이크로프로세서와 함께 중앙 처리부를 구성하는 이중 프로세서 구조가 개발되었다. 논리 처리기가 주 프로세서가 되어 명령을 읽고 논리명령이 아닌 경우에는 범용 마이크로프로세서를 구동하여 풀게 하는 방식으로 고속, 고기능의 중앙 처리부를 구현한 것이다. 하지만 두 프로세서간의 전환에는 어느 정도의 시간이 걸린다는 점과 비교 명령 등에서 데이터의 전달이 필요한 점, 그리고 잦은 분기 명령으로 인해 파이프라인 처리의 장애가 많이 일어나는 점등으로 인해 속도 개선의 한계가 있다.

PLC의 중앙 처리부에 쓰이는 마이크로프로세서의 최신 기술은 초기의 기술과 유사하게 PLC의 명령을 수행하는 전용 마이크로프로세서를 설계하는 것이다. 그러나 초기의 PLC에 비해 명령의 수가 훨씬 늘어나서 200 여종이 넘는 명령들을 그대로 마이크로프로세서의 명령으로 만드는 것은 매우 복잡하고 어려운 기술이 요구되는 일이다. 그래

서 RISC의 개념을 도입하여 PLC의 명령과 프로그램에 대한 통계적 분석을 한 후에 빈번하게 쓰이는 명령을 마이크로프로세서의 명령으로 만들고 자주 쓰이지 않는 명령은 여러 개의 명령으로 대체하는 방식이 사용된다. PLC의 명령의 75%가 피 연산자를 메모리에서 읽어야 한다는 통계를 토대로 명령어와 데이터의 버스를 나누어 동시에 액세스가 가능하게 하는 하버드 구조의 마이크로프로세서가 개발되었으며, 파이프라인의 장애를 없애고 효율을 높이기 위해 분기 명령 없이 조건부 수행이 가능한 마이크로프로세서도 설계되고 있다.

### 3. 다중 프로세서 구조 기술

PLC가 널리 쓰임에 따라 그 기능이 확대되고 고속 실행이 요구되어 하나의 마이크로프로세서와 하나의 버스로 모든 기능을 해결하는 초기 구조는 메모리 액세스가 병목 현상을 일으키며 서로 상이한 특성을 가지는 여러 부가적인 태스크의 실행에 부적합하게 되었다. 이를 해결하는 방법으로 중앙 처리부는 응용 프로그램의 실행을 전담하게 하고 부가 기능을 위해서는 전용 프로세서를 사용하는 다중 프로세서 구조가 개발되었다. 입출력 프로세서와 통신 프로세서가 그 대표적인 두 가지 전용 프로세서이다. 입출력 프로세서는 아날로그 입출력 모듈, 축 위치 제어 모듈 및 기타 제어 모듈들에서 주기적 샘플링, 선형화, 스케일링, 제어 연산 등을 실행한다. 통신 프로세서는 MMI를 위한 콘솔 연결, 분산 처리를 위한 네트워크에 쓰이며 정해진 프로토콜에 따라 PLC와 외부 사이의 정보를 주고받는다.

위와 같은 기능별 다중 프로세서화는 이미 대부분의 PLC에서 일반화된 기술이며 고성능 PLC를 위한 최신 기술은 하나의 응용 프로그램을 여러 태스크로 나눈 뒤에 여러 개의 프로세서가 동시에 풀게 하는 형태의 다중 프로세서 구조 기술이다. 하나의 버스를 여러 프로세서가 공유하는 버스 구조가 채택되고 있으며 프로세서간의 정보 교환은 공유 메모리를 통해 이루어진다. 네트워크로 연결된 PLC간의 정보 교환과 마찬가지로 각 프로세서는 입출력 기기를 공유하지 않는 경우가 대부분

이지만 운영 체제 기술 및 시스템 소프트웨어 기술의 발전과 함께 입출력 기기를 공유하고 기타 다른 내부 변수도 공유할 수 있는 다중 프로세서 구조의 중앙 처리부도 연구되고 있다.

#### 4. 앞으로의 전망

PLC는 그 용도와 속성 때문에 빠른 처리 속도 보다 신뢰도와 다기능성이 중요시되는 특수 용도의 컴퓨터 시스템이다. 그러므로 범용 컴퓨터에 비해 마이크로 컴퓨터 관련 기술의 발전과 응용이 느린 편이다. 그러나 고급 기종의 개발과 제품의 차별화 등을 위해 실시간 성능의 향상과 다양한 기능의 보강을 중심으로 마이크로프로세서 개발 기술과 다중 프로세서 구조 기술의 사용이 점차 보편화될 것이다.

### III. PLC의 운영 체제 및 시스템 소프트웨어 기술

운영체제는 컴퓨터내의 연산 프로그램들의 실행과 자원들을 효율적으로 관리하기 위해 존재하는 특수한 기능들의 모임으로 모든 컴퓨터 시스템에서 중요한 역할을 수행한다. PLC에서 필요한 운영체제 역시 PLC 프로그램의 실행과 여러 자원(CPU, 파일, 입출력 장치, 통신장치 등)의 관리 등을 담당한다. PLC의 경우, 입출력 장치 관리 및 통신장치 관리 등 PLC 특유의 역할에 국한된 운영체제가 주를 이루었으나, 컴퓨터의 발달과 더불어 운영체제가 고기능을 가지면서 시분할 시스템(time-sharing system)과 다중 처리 방식(multi-tasking)과 실시간 운영체제(real-time operating system)등이 PLC의 분야에도 적극적으로 적용되는 움직임을 보이고 있다. 특히, 실시간성을 가지는 시스템의 제어를 위한 PLC를 위하여 실시간 운영체제 분야가 부각되고 있다. 실시간 운영체제는 잘 정의된 시간제약을 가져야 한다. 태스크는 정의된 시간 내에 수행이 완료되어야 하며, 그렇지 못할 경우 예상하지 못한 결과가 발생한다. 여기서

는 PLC분야의 운영체제의 특징과 미래지향적인 실시간 기능에 대하여 설명한다.

#### 1. 태스크 관리기능

PLC 연산 보드에서는 여러 개의 프로그램이 동시에 실행되며 동일한 평선 블럭등을 서로 다른 프로그램들이 인스턴스(instance)의 형태로 사용할 수 있으며 그 구조는 계층적 구조를 가진다. 이와 더불어 래더 언어, IL, SFC등 서로 다른 언어로 표현된 프로그램을 관리할 수 있는 기능이 필요하다. 이러한 기능은 PLC의 표준화 방향에 맞추어 기존의 단순한 태스크의 관리기능에서 벗어나 동적 메모리할당, 태스크간의 통신 기능 등의 운영체제 기술이 적용되어야 함을 의미한다.

#### 2. 태스크의 스케줄링 기능

태스크란 사용자가 SFC나 래더 언어 혹은 PLC의 다른 언어로 작성한 프로그램의 실행 단위를 나타낸다. 멀티태스킹 구조에서 태스크의 스케줄링 방식에는 크게 라운드 로빈(round-robin)방식과 우선 순위 방식이 있다. 라운드 로빈 방식은 모든 태스크에게 동일한 시간 량(time quantum)만큼을 번갈아 할당한다. 따라서, 모든 태스크는 긴급성에 있어서 동일한 우선 순위를 가진다. 이러한 방식은 고유한 스캔 주기를 지정한 태스크의 의미를 상실하는 것이 된다. 즉, 짧은 스캔 주기를 가지는 태스크는 긴 주기를 가지는 태스크보다 긴급히 처리해 주어야 하는데 라운드 로빈 방식은 동일한 시간만큼을 모든 태스크에게 할당하기 때문에 짧은 스캔 주기를 가지는 태스크의 긴급성을 만족시키지 못할 가능성이 있다.

예를 들어, 10개의 태스크가 있는데 타임 쿼텀이 1msec라면 10msec의 실행 시간을 가지는 태스크는 10번의 타임 쿼텀을 필요로 한다. 10번의 타임 쿼텀을 가지려면 10개의 태스크가 존재하므로 100msec의 시간이 필요하게 된다. 따라서, 스캔 주기가 100msec이상의 태스크만이 스케줄이 가능하다는 결론이 된다. 이 방식은 아직도 많은 PLC가 채택한 방식이지만, 실시간성이 긴급한 시스템에는 적합하지 않은 스케줄링 방식이다.

PLC의 특성을 고려하여 실시간 태스크의 스케줄링에 적합한 스케줄링 방식에는 RMA(rate monotonic algorithm)가 있다. RMA방식은 스캔 주기가 짧은 태스크에게 높은 우선 순위를 부여하는 방식이며, 이 방식을 쓰면 라운드 로빈(round robin)의 방식의 단점을 극복할 수 있다. 이 때의 조건은 각 태스크 사이에는 연관 관계가 없으며 선점(preemption)이 가능하여야 한다. RMA의 장점은 또한 긴급한 태스크와 긴급하지 않은 태스크를 구분하여 우선 순위를 설정함으로써 실시간성을 더 효율적으로 구현할 수 있도록 한다는 점이다. 예를 들어 N개의 태스크가 각각  $T_i$ 의 스캔 주기를 가진다고 하자. 여기서 각 태스크의 스캔 주기는  $T_i \leq T_{i+1}$ 이다. 즉 스캔 주기가 짧을수록 우선 순위가 높다. 우선 순위는 1이 가장 높으며 N이 가장 낮다. 이때 태스크의 마감시간은  $T_i$ 이므로  $T_i$ 의 스캔 주기를 가지는 태스크의 실시간 스케줄 가능성은 아래의 식에 의하여 결정된다. 여기서  $C_k$ 는 i번째 태스크의 실행시간이다.

$$\sum_{k=1}^{i-1} \frac{C_k}{T_k} + \frac{C_i}{T_i} \leq i(2^{1/i} - 1), \quad i=1,2,\dots,n$$

일반적으로 태스크의 실행시간은 미리 알아내기 힘든 경우가 많다. 따라서, 여러 태스크가 실행중인 실제 상황에서는 운영체제에 의해 태스크의 데드라인이 만족되는지 항시 점검하여 사용자에게 보고하는 방식을 취한다.

### 3. 입출력 갱신 기능

일반적으로 입출력 갱신은 각 태스크가 종료하는 시점에서 모든 입출력 접점을 갱신한다. 이것은 전체의 실행 시간을 지연시키는 결과를 가져오며 불필요한 작업을 할 가능성이 있다. 입출력 갱신을 어떠한 시점에 할 것인지의 방법에는 아래의 방법이 있을 수 있다.

방법 1 실행코드의 끝 시점에서 입출력 모두 갱신한다(burst update, 집중갱신).

방법 2 실행코드 중간에 필요한 경우 입출력을 갱신한다(continuous update, 연속갱신).

방법 3 실행코드의 시작시점에서 입력을 갱신

하고 끝 시점에서 출력을 갱신한다.

방법 4 실행코드와 관련 없이 일정한 주기(최소 스캔 주기)로 입출력을 갱신한다.

대개의 PLC는 집중 갱신(burst update)으로 한 번의 스캔이 완전히 끝나는 경우에 한꺼번에 처리하게 되며, 사용자가 이를 제어할 수 있도록 특정 명령의 경우 수행과 동시에 갱신될 수 있는 명령을 제공하기도 한다. 또, 초소형의 PLC 경우에는 방법 2의 연속갱신을 하기도 한다. 방법 3은 응답 성능을 개선할 수 있으나 입출력을 구분해야 하므로 태스크의 수가 많아짐에 따라 입출력갱신이 빈번해지며 구현이 어려운 단점이 있다. 방법 4의 입출력 갱신은 별도의 입출력 프로세서가 있는 경우에 적용가능하다.

### 4. 통신 기능

하나의 PLC 연산 보드는 다른 연산보드, 사용자, 외부 호스트와의 인터페이스를 지원하여야 한다. 이러한 기능을 담당하는 태스크는 PLC내의 연산 태스크의 수행시간을 방해하지 않는 방식으로 이루어져야 한다. 즉 외부와의 통신에 대한 이벤트는 태스크의 통신방식인 메일 박스(mail box), 메시지 큐(message queue)등의 원시 기능을 사용하여 구현한다. 운영체제는 이러한 원시 기능을 지원하여야 한다. 최근의 PLC의 운영체제는 이러한 원시기능을 지원하는 추세이다.

### 5. 기타 기능

기타기능으로 PLC의 이상여부를 판단하고 이를 사용자에게 알려주기 위하여 워치 도그 타이머(watch dog timer)기능, 자기진단기능, 데이터 로깅(logging), 이중화 기능, 이상 메시지 출력기능이 있다.

## IV. PLC의 프로그래밍 언어

### 1. 래더 언어

PLC는 하드와이어 릴레이와 타이머를 대체하기

위하여 고안된 것이므로 기존 하드와이어 제어회로를 내부 기호를 사용하여 프로그램 하도록 되어 있다. 기본이 되는 언어는 회로도를 그대로 사용할 수 있는 구조로 된 래더 언어이며, 형식은 회로도의 형식을 따른다. 즉, 기존 하드와이어 로직을 소프트웨어 형태로 입력하는 소프트와이어 방법을 사용한다.

래더 언어는 크게 릴레이 로직 명령 군과 데이터 처리 명령 군으로 구성된다. 릴레이 로직 명령 군은 릴레이 로직으로 구성할 수 있는 기본 논리 동작, 즉, 시작(STR), 출력(OUT) 및 기타 기본 논리 동작을 하는 기호로 구성되며, 입출력 접점과 일대일로 대응된다. 데이터 처리 명령 군은 초기에는 타이머, 카운터의 동작을 수행하는 명령으로 구성되었다. 컴퓨터의 발달과 더불어, 데이터 처리 명령 군은 일반 컴퓨터가 담당하는 문자열 처리 및 복잡한 정수 연산은 물론 실수 연산의 기능을 보유하게 되었으며, 이와 더불어 PLC가 담당하는 영역이 점차 넓어져, 단순 논리 동작 외에 데이터 감시 및 아날로그 입출력 제어 등으로 다른 제어 기기의 기능을 일부 포함할 수 있도록 발전하였다.

현재, 중소 규모의 PLC 경우에도 타이머, 카운터, 흐름제어(flow control, JUMP, CALL 등), 데이터 전송, 산술 연산, 프로세스 제어 연산(PID 등), 데이터 통신 등을 포함하는 데이터 처리 명령 군을 가지고 있다. 따라서 PLC에서 사용되는 데이터 형식은 단순히 논리 변수 외에 정수, 실수, 문자, 고정도 정수, 고정도 실수 등을 모두 지원한다.

## 2. 래더 언어의 동작

CPU가 래더 언어를 읽어서 수행하는 것을 스캔(scan)이라 부른다. 래더언어는 릴레이 로직을 나타내는 회로도이므로, 실제 릴레이 로직과 같이 병렬적으로 수행되어야하나, 실제로는 임의의 순서를 정하여 순서적으로 수행할 수밖에 없다. 래더언어를 수행하는 순서는 크게 둘로 나눌 수 있는데, 하나는 열 단위로 해석하는 방법이며(AEG Modicon 등의 경우) 다른 하나는 행 단위로 수행하는 방법이다(Allen Bradley 등의 경우).

래더 언어가 서로 입출력을 교차하여 참조하는 경우에는 실제 릴레이 로직, 행 단위 수행, 열 단위 수행의 결과가 일시적으로 달라질 수 있으며, 사용자가 이를 염두에 두고 프로그램 하여야 한다. 또 실제 릴레이 로직과 다른 점은 항상 왼쪽에서 오른쪽으로만 전류가 흐르는 것과 같은 동작을 한다는 것과, 수직 연결과 층첩 연결 등이 허용되지 않는다는 것이다. 데이터 처리 명령 군은 상자 모양의 기호를 사용하여 표현하며, 출력 코일과 같이 원편에 연결된 릴레이 로직에 의하여 동작 여부가 결정된다.

## 3. IL(Instruction List)

PLC 프로세서는 일반 컴퓨터와 같이 마이크로 프로세서로 구성되어 있는 것이 일반적이므로(경우에 따라서는 비트 슬라이스나 VLSI 등으로 구성하는 것도 있으나, 동작 방법은 같다.) 마이크로프로세서의 기계어와 같은 명령어를 사용할 경우 얻게 되는 장점이 있다. 래더언어로 된 프로그램은 그대로 스캔할 수 없으며 그림의 형태이므로 이를 해석하는데는 적합하지 않으므로 이를 변환하여 처리하는 것이 일반적이다. 래더 언어를 일대일로 변환하여 기술한 언어를 IL이라 하며, 기계어와 같은 구조를 갖는다. 많은 데이터 처리를 하거나 비슷한 동작이 되풀이되는 프로그램을 작성하는 경우에 IL이 래더 언어보다 편리하여 널리 사용된다.

## 4. FBD(Function Block Diagram)

프로세스 제어 등에서는 논리 회로도 외에 제어 알고리즘을 순서도나 블럭도와같이 표현하는 것이 유리하다. 근래 PLC는 프로세스 제어의 목적에도 널리 사용되므로 이를 지원하는 언어가 필요한데, FBD가 이를 지원한다. FBD 역시 수행하는 순서가 상자가 위치하는 곳에 영향을 받으므로, 사용자가 이에 대한 수행 규칙을 염두에 두고 프로그램 하여야 한다.

## 5. SFC(Sequential Function Chart)

병렬 분기 및 판단 분기를 효율적으로 표현하며, 현재 스테이트에 따라 동작을 결정하는 PLC언어

로 SFC가 있다. SFC는 안전 페트리네트를 기초로 하여 고안되었으며, 페트리네트의 성질을 포함하여 기존 래더 언어의 단점을 크게 해결할 수 있다. SFC 만으로 독자적으로 사용되기보다는 래더 언어, IL, FBD, ST 등의 상위에서 사용되는 것이 일반적이며, 경우에 따라서는 SFC 상위에서 계층적으로 사용되기도 한다.

## 6. 고급언어

그림으로 표현하는 언어와 글자로 표현하는 언어에는 각기 장단점이 있다. 글자로 표현하는 IL의 기능을 극대화하기 위하여 일반 컴퓨터와 같이 고급언어를 사용할 수 있는 PLC가 근래에 등장하였다. BASIC, C, PASCAL 등의 고급 프로그램 언어와 비슷한 고유의 언어를 제공하는 PLC에서부터 일반적인 프로그램 언어를 지원하는 PLC도 존재한다. 최근 범용의 내장형 제어기(embedded controller) 하드웨어를 사용하여 소프트웨어로 PLC를 구성하는 제품도 등장하고 있다. 일부는 범용의 PC(개인용 컴퓨터, 특히 IBM 호환을 일컫는다) 하드웨어를 그대로 사용하기도 한다(PC-based PLC). 이러한 제품에서는 고급언어의 중요성이 더욱 강조된다.

## 7. PLC 언어의 표준

래더 언어의 경우에 릴레이 로직 명령 군은 각 제조 업체에 따라서 큰 차이를 보이지 않으나, 데이터 처리 명령군의 경우에는 각 제조 업체별로 수백 종의 고유 명령 군을 가지고 있다. 마찬가지로 IL, FBD 및 고급언어의 경우에도 제조 업체별로 호환성이 없다. 이러한 문제를 해결하기 위하여 최근 제정된 국제 PLC 언어 표준화로는 IEC-1131-3이 있다. IEC-1131-3에서는 PLC에서 지원하는 데이터 형식과 5종의 언어, 래더 언어, IL, FBD, ST(Structural Text, 고급언어), SFC의 표준안을 제시한다. 현재 유럽을 중심으로 표준화가 진행되고 있으며, 국내 업체의 경우에도 이 표준을 따르는 PLC를 개발하고 있으며, 일부 제품이 출하되었다.

## 8. PLC 언어의 실시간성 및 동기 문제

PLC 제어 프로그램을 이산 현상 시스템(DES, discrete event system)의 이론에 의하여 해석해보면, 제어 프로그램이 만족해야할 시간적 제약 조건과 데이터들이 서로 만족해야할 동기(선행관계, precedence relation)가 명확히 존재한다. 현재 PLC의 경우에 시간적 제약 조건과, 동기 관계는 크게 고려되지 않는다. 이상적으로 동작하는 PLC는 시간적 제약 조건을 만족하는 최대 스캔 시간을 유지하며, 데이터간의 동기가 정확히 이루어져야 한다. 현재 PLC의 경우에 이러한 이상적인 최대 스캔시간보다 매우 빠른 스캔 시간을 통하여 데이터간의 동기 문제와 시간적 제약 조건을 해결한다. 최소의 비용으로 안전하고 정확한 제어를 하려면, 정확한 시간적 응답 보장과 데이터 동기 및 사건 구동(정확히 스테이트가 고려되어 설계된 SFC의 경우에는 사건 구동과 비슷한 효율을 가진다.) 등이 고려되는 PLC가 등장하여야 한다.

## V. 입출력 시스템 관련 Signal Conditioning 기술

### 1. 기본 개념

PLC는 현장에서 기계 장치나 스위치와 같은 외부기와 직접 접속이 되어 사용된다. 이때 PLC의 내부는 +5V의 약전을 사용하고 있기 때문에 입출력 부의 중간 역할은 시스템의 안정에 결정적 요소가 된다. PLC의 입출력부가 갖추어야 할 기본 조건은 아래와 같다.

- 외부기와 전기적 규격이 일치할 것
- 외부기로부터의 노이즈를 막아줄 것
- 외부기와와의 접속이 용이할 것
- 입출력 상태를 감시할 수 있을 것

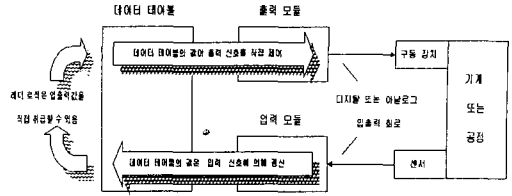
입력 및 출력부에 접속될 수 있는 외부기의 예는 다음의 표와 같다.

### 2. 입출력 시스템의 종류

가. 직접 입출력(direct I/O)

〈표 1〉 입출력 신호의 예

	입력 신호	출력 신호
기계 장치에 부착되는 것	<ul style="list-style-type: none"> <li>리미트 스위치</li> <li>광전 스위치</li> <li>근접 스위치</li> <li>엔코더 등</li> </ul>	<ul style="list-style-type: none"> <li>전자 밸브</li> <li>전자 클러치</li> <li>전자 브레이크 등</li> </ul>
제어반이나 조작반 등에 부착되는 것	<ul style="list-style-type: none"> <li>푸시버튼 스위치</li> <li>셀렉터 스위치</li> <li>써머 릴레이</li> <li>디지털 스위치</li> <li>계측기 등</li> </ul>	<ul style="list-style-type: none"> <li>전자 릴레이</li> <li>표시 등</li> <li>가변속 모터 제어 장치 등</li> </ul>



〈그림 2〉 직접 입출력 모듈

Ω 구리 등

- mV/열전대(TC, thermocouple) 입력 : ± 100mV 또는 B, R, S, E, J, K, T-타입 열전대

직접적인 입출력 기능만을 가지고 있는 것을 직접 입출력 모듈(direct I/O module)이라고 한다. 직접 입출력은 PLC의 데이터 테이블상의 비트나 워드에 직접 대응되는 입출력 모듈의 회로를 가지고 있는 입력 혹은 출력을 의미한다. 이러한 비트나 워드는 해당 입출력 회로의 신호 값을 저장하고 있다. 따라서, 래더의 로직이 입출력 값을 직접 취급할 수 있다. 직접 입출력 모듈은 입출력 회로가 구동부로 혹은 센서로부터 신호를 전달하는 방법에 따라 구분된다.

■ 디지털 입출력 -on과 off의 두 가지 상태를 갖는 스위칭 회로

- 특징-포토 커플러(photo coupler)를 사용하여 외부기기와 내부회로를 전기적으로 절연하여 내 노이즈성을 강하게 한다.
- 전압 범위-24, 120, 220Vac/Vdc 에서 5, 10~60Vdc 의 범위

■ 아날로그 입출력-신호가 특정한 한계값 사이에서 연속적으로 변화하는 회로

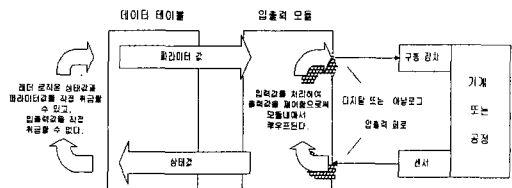
- 특징-아날로그/디지털, 디지털/아날로그 변환을 통해 아날로그 신호와 인터페이스 하는데, 분해능(resolution)은 8~16-비트이다.
- 아날로그 신호의 종류

- 전압 신호 : 0-5Vdc, 1-5Vdc, ± 5Vdc, ± 5Vdc, ± 10Vdc 등
- 전류 신호 : 4-20mA, 0-20mA, ± 20mA 등
- 저항체 온도(RTD, resistance temperature device) 입력 : 100Ω 백금, 120Ω 니켈, 10

나. 지능형 입출력(intelligent I/O)

지능형 입출력 모듈(intelligent I/O module)은 래더 로직의 제어를 위한 데이터 테이블을 거치지 않고 입력값을 온보드(on-board)로 처리하여 출력 값을 제어한다. 따라서 지능형 입출력 모듈은 몇 개의 직접 입출력을 가질 수도 있지만, 모든 입출력을 데이터 테이블과 직접적으로 관련시키지 못 할 수도 있다. 하나의 지능형 입출력 모듈은 디지털 입출력 회로나 아날로그 입출력 회로, 혹은 두 회로 모두를 장비하고 있다. 지능형 입출력 모듈에는 다음과 같은 것들이 있다.

- 선형 위치 결정 모듈(linear positioning module) - 디지털 위치 값과 아날로그 속도 값을 궤환(feedback)받아 처리하여 아날로그/펄스 속도 출력 값을 제어한다.
- PID 모듈-아날로그 공정변수(process variable) 입력을 처리하여 아날로그 제어변수(control variable) 출력 값을 제어한다.
- 기타 특수 모듈 - 고속 카운터 모듈, 위치 검출 모듈, 스테핑 모터 드라이브 모듈, CRT/



〈그림 3〉 지능형 입출력 모듈



LCD 제어 모듈, 음성 출력 모듈, 병렬 인터페이스 모듈 등

### VI. 네트워크 시스템 관련 기술

#### 1. 기본 개념

자동화 공정 내부의 PLC와 PLC, 또는 PLC와 외부 기기 간의 정보를 교환하려는 필요에 의해서 각종 제어기기들은 통신 기능을 갖추게 되었다. 소형 PLC의 경우 필요한 통신 하드웨어 및 소프트웨어가 PLC 본체에 통합되어 있지만, 대형 PLC는 다양한 응용에 알맞은 다수의 통신 모듈을 갖추고 있다. PLC의 통신 포트를 사용하는 경우는 다음과 같다.

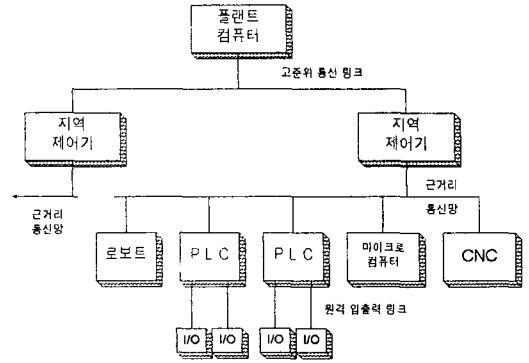
- 프린터나 VDU(visual display unit) 등을 통해 운전 데이터나 알람을 표시하는 경우.
- 컴퓨터 파일이나 기록 매체를 통해 데이터를 로깅 하는 경우: 이 데이터들은 공정의 성능 분석이나 운전 정보로 활용된다.
- 운전 터미널이나 관리 제어기로부터 기존의 PLC 프로그램으로 변수나 파라미터를 전달 하는 경우.
- 현재 상주하는 PLC 프로그램을 변경하는 경우-관리 제어기(PLC 혹은 마이크로컴퓨터)에서 업로딩 또는 다운로드한다.
- 원격 터미널에서 강제로 입출력 접점을 on/off 시키거나, 메모리 값을 변경하는 경우.
- 다수의 PLC와 컴퓨터들을 포함하는 제어 계층 구조 내에 PLC를 연결(link)시키는 경우.

#### 2. 기존 PLC 통신망

PLC에서 제공되는 통신망은 계층 구조를 갖고 있다. 이 계층 구조는 그림 4와 같이 3 계층 정도로 구분되는 것이 일반적이다.

가. 원격 입출력 링크(remote I/O link)

전체 계층 구조에서 볼 때 가장 하위에 해당하는 원격 입출력 링크는 PLC 프로세서와 원격 입출력 장치를 연결하는 통신망으로서, 기타 지능형



(그림 4) 통신망의 계층 구조

장치(예: 운전자 인터페이스)의 호스트와도 연결될 수 있다. 이 링크는 PLC 프로세서에 장비되어 있는 스캐너를 통해 접속될 수도 있고, 독립적인 스캐너 모듈을 통해 접속되기도 한다. 후자의 경우를 특별히 컴퓨터 링크(computer link)라고 부르기도 한다. 원격 입출력 링크는 보통 RS232, RS422 또는 RS485 등의 신호 레벨을 사용하며 비동기 전송 방식을 택하고 있다.

나. 근거리 통신망(local area network)

자동화되어 있는 공장이나 공정 전체로 볼 때, 작업성과 생산성을 향상시키기 위해서는 공장 전체적인 통신 기능을 필요로 하게 된다. 이러한 통신 기능은 마이크로 컴퓨터와 로봇, CNC 장비와 PLC 등의 각종 장비를 연결하여 제어하고 프로그램할 수 있게 한다. 대부분의 제조업체에서는 전용의 통신망 시스템을 제공하여 PLC와 자사 제품의 링크 기능을 제공한다. (표 2 참조)

(표 2) 각 PLC 제조업체의 근거리 통신망

PLC 제조 업체	통신망
Allen-Bradley	Data Highway
Gould	Modbus
General Electric	GE Net Factory LAN
Mitsubishi	Melsec-NET
Texas Instruments	TIWAY
SIEMENS	SINEC

다. 고준위 통신 링크(High-level communication link)

통신 기능은 PLC가 특정 공정 혹은 설비의 제어기로서의 역할을 넘어서, 대규모 생산 시설의 다수 스테이션의 제어기 역할을 할 수 있게 해 준다. 즉, 계층적인 제어 구조의 일부가 되어, 다수의 PLC나 CNC 혹은 로봇과 같은 지능형 장비를 관리 감독할 수 있게 되는 것이다. 이와 같은 고준위의 통신 링크를 위해서는 각 업체의 특수한 통신망이 아닌 표준적인 통신망과 연결할 수 있는 기능을 가져야 한다.

### 3. 최근 추세 및 향후 전망

위에서 언급한 바와 같이 현재 현장에 구축되어 있는 대부분의 PLC는 제작사 고유의 프로토콜을 많이 사용하였기 때문에 타 업체의 기기와 데이터를 공유함에 있어서 추가적인 비용과 시간이 소요되는 등 어려움이 많았다. 이는 생산비용과 직결되는 문제이며 개방형 네트워크의 필요성을 대두 시켰다. 개방형 네트워크는 모든 사람에게 통신 규약을 공개하여 누구나 통신망을 쉽게 구축 혹은 제작할 수 있게 한 네트워크이다. 이러한 개방형 네트워크는 국제 표준 기구에서 제안한 7계층 구조인 OSI(open system interconnection) 구조 혹은 그 일부를 가지고 있으며 TCP/IP, MAP, Profibus, FIP 등이 이에 해당된다. 그러나 TCP/IP는 실시간성을 보장하기 어려우며, MAP도 구현 비용과 실시간성 등의 문제로 인해 PLC에 사용하기에는 부적합한 면이 있다. 이러한 단점을 보완하고 주어진 필드에서 필드기기들과 상위의 제어기기간의 효율적인 통신을 위하여 제안된 것이 Field-Bus이다. Field-Bus는 그 응용분야에 따라 FIP, Profibus, IEC/ISA Bus, CAN, BITBUS, HART, INTERBUS-S, LON, MIL STD 1553, SERCOS등 그 종류가 매우 다양하며, 이 중 Profibus와 FIP가 개방형 산업용 네트워크에 적합한 것으로 알려져 있다. Profibus는 Profibus-FMS(fieldbus message specification)와 Profibus-DP(decentralized peripherals)의 2 종류가 표준으로 되어 있으며 공정 제어 응용을 위

한 표준인 Profibus-PA(process automation)가 작업 중에 있다. Profibus-FMS는 일반 산업 응용에 사용하고자 할 때 유용하며, Profibus-DP는 실시간성이 매우 중요하며 빠른 시간 안에 데이터를 전달하고자 할 때 적합하다. 그러나 주기적인 데이터를 처리하기가 용이하지 못하다는 단점을 갖는다. 이에 비해 FIP는 데이터가 주기적으로 교환되어야 하는 경우에 적합하나, 이벤트성 데이터를 처리함에 있어서 비효율적이다.

Field-Bus는 기존 PLC등에서 사용하고 있는 네트워크와 비교할 때 설치 비용 절감, 시스템 확장 및 변형용이, 원격에 의한 유지, 보수 용이 등의 많은 장점을 지닌다. 이러한 장점으로 인해 1980년대 중반부터 유럽, 미국, 일본 등 기술 선진국에서 Field-Bus에 대한 많은 연구가 수행되어 왔다. Profibus와 FIP는 현재 독일과 프랑스에서 각각 국가 표준으로 되어 있으며 최근 PLC를 비롯한 여러 자동화 기기에 적용되기 시작하고 있다. 아직 국제 표준화 작업은 완성되지 않았으나, 여러 측면에서 볼 때 Field-Bus는 차세대 분산 제어 및 자동화 시스템의 구축을 위한 핵심 기술로 부상할 것으로 예상된다. 한편 Field-Bus외에도 PLC용 통신 네트워크의 표준화 작업이 IEC1131-5를 통해 진행되고 있다.

## VII. 내결함 PLC 시스템

내결함 시스템은 시스템의 신뢰성을 높일 수 있다는 장점 때문에 점차 널리 보급되는 추세이다. 내결함 시스템에서는 결함의 빠른 검출, 결함 상태에서의 빠른 회복, 결함 발생시 동작의 지속, 다운타임(downtime)의 최소화 등이 성능지표로 사용되며 이를 효과적으로 구현하기 위한 여러 방법들이 시스템 구조 분야에서 연구되고 있다. 내결함은 기본적으로 기능과 하드웨어 자원의 중복에 의해 구현 가능하다. 즉 결함이 생긴 부분을 제외시켜도 동일한 기능을 하는 중복된 모듈에 의해 결함이 감추어 질 수 있도록 하는 것이다. 중복은 시스템

의 가격 상승과 직접적인 관련이 있으므로 전체 시스템에서 어느 부분을 얼마나 중복하는가를 결정하는 것은 내결함 시스템의 구현에서 중요한 문제가 된다. 내결함 시스템의 대표적인 구조는 다음과 같이 분류될 수 있다.

#### □ 프로세서 이중화

시스템에서 가장 결정적인 부분은 계산을 담당하는 부분(프로세서)이므로 이 부분을 두 개의 독립된 모듈로 구성하는 방식이다. 둘 중의 하나가 동작하는 동안 다른 것은 핫-백업(hot backup) 상태가 되어 동작하고 있는 모듈의 상태를 그대로 유지하며 동작 모듈에서 결함이 검출되면 즉시 백업 모듈로 기능이 이전되는 구조이다. 결함의 검출은 모듈별로 자기 진단 기능에 의해 이루어지므로 프로세서의 동작 정지 등과 같은 결함만이 검출될 뿐 계산상의 오류 등은 검출되지 않는다.

#### □ 시스템 이중화

프로세서 부분뿐 아니라 I/O와, I/O와 프로세서를 연결하는 버스까지 모두 이중화하여 시스템을 구현하는 방식이다. 따라서 모든 계산 모듈과 메모리 모듈, I/O 모듈은 모두 듀얼 포트를 가지며 이러한 모듈들 자체가 또한 이중화된다. 각 모듈들의 상태는 미리 설정된 체크 포인트에서 검사되어 이상이 검출된 경우 다른 모듈로 기능이 이전된다. 이중화 프로그램이 어려우며 체크 포인트 이외에서 발생한 오류는 검출되지 않는다.

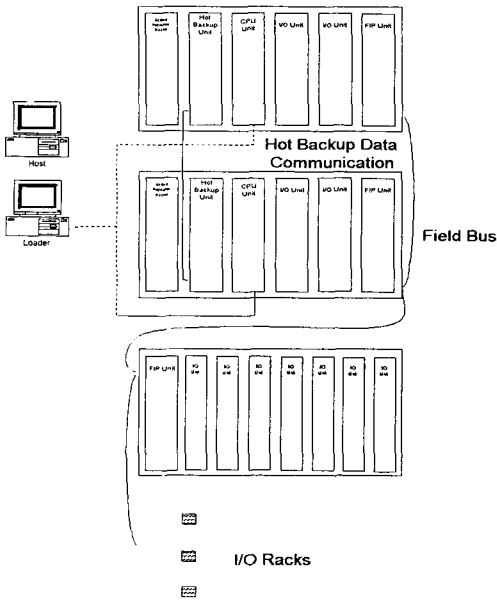
#### □ 투표구조

I/O 혹은 간단한 제어 모듈의 경우에 가장 확실한 결함 은닉성을 제공하는 구조이다. 3개이상의 동일한 기능을 하는 모듈을 동시에 동작시켜 출력되는 결과 중 다수를 차지하는 값을 취하는 것이다. 제어 모듈이 다중화 되는 경우 모든 연산 결과도 검사되지만 시스템 구성에 비용이 많이 든다는 단점이 있다. 그러나 항공기, 원자력 발전소 등 신뢰성이 대단히 중요한 부분에 쓰이는 센서 입력 부분은 모두 이와 같은 구조가 이용된다.

내결함 시스템의 구조에는 이상에서 언급된 구조 외에 4중화된 페어-스페어 구조등 여러 시스템 구조가 있으며 이미 전산기, 대용량 데이터 베이스 시스템 등에 널리 쓰이고 있다. PLC에서도 이러

한 내결함 구조가 채용된 것이 있으며, 곧 대부분의 PLC가 이러한 다중화 구조를 채용할 것이다. 그림 5는 일반적인 내결함 PLC의 구조를 보이고 있다. 현재 상용화되어 있는 대부분의 내결함 PLC는 그림과 같이 프로세서 이중화 구조로 되어 있으며 시스템 이중화 구조도 일부 사용된다. 시스템 이중화 구조에서는 I/O가 이중화되며 프로세서와 I/O를 연결하는 네트워크 또한 이중화된다. 대부분의 PLC는 프로세서와 I/O의 연결에 전용의 네트워크를 사용하지만 현재 서서히 필드 버스와 같은 표준화된 네트워크를 사용하는 경향도 있는데, 표준화된 필드 버스의 경우 버스 이중화의 내용을 담고 있다. 따라서 표준 필드 버스를 사용하는 PLC 시스템은 이중화 구조를 보다 용이하게 구현할 수 있게 된다.

프로세서 이중화 PLC에서는 실행중인 것과 대기중인 것의 상태를 똑같이 하기 위해서 고속의 핫-백업 통신 라인으로 실행중인 것의 상태 데이터를 항상 대기중인 것에 전송한다. 상태 데이터는 I/O 접점 데이터와 내부 접점 데이터 등인데 대용량 PLC의 경우에는 상태 데이터의 크기가 핫-백업 통신 라인으로 전송되기에는 너무 양이 많아서, PLC의 상태가 변하는 스캔 주기 내에 모두 전송되는데 문제가 있다. 이 경우에는 매 스캔 주기마다 두 시스템의 상태를 동기화하지 않고 일정 간격을 두고 데이터를 전송, 동기화 한다. 동기화가 된 후에는 대기 모듈도 입력 데이터를 가지고 실행 모듈의 프로그램을 동시에 수행하므로 두 시스템의 상태는 같아지게 된다. 두 시스템은 동작 중에 끊임없이 자신을 위치 도그 등 다양한 감시 장치에 의해 진단하며, 두 시스템을 연결하는 핫-백업 통신 라인은 시스템 전체에서 가장 신뢰성 있게 설계된다. 이 두 가지가 이중화 PLC 전체의 신뢰성을 결정한다 할 수 있으므로 상용화되어 있는 PLC의 경우 각 PLC 생산 업체들마다 독특한 방식으로 이를 구현하고 있다. 그런데 현재는 이중화 PLC에서 어떠한 오류까지 검출해 낼 수 있는가 보다 오류 상태에서 얼마나 빨리 회복하는가 하는 것이 더 중요한 성능 지표가 되어 있으므로 고쳐져야 할 부분이다.



(그림 5) PLC의 핫-백업 구조

### VIII. 결론 및 향후 동향

1969년에 처음 개발된 PLC는 지난 27년간 공장 자동화 분야에서 큰 영향을 미쳐왔으며 현재에는 응용 기법이 완전히 정착되어 자동화 시스템에서 없어서는 안될 중요한 위치를 차지하고 있다. 이렇게 중요한 제어 기기로서의 PLC는 최근에는 적용 분야의 확대와 마이크로 컴퓨터 및 반도체 기술의 급속한 발전 등에 따라 90년대에 들어와서는 새로운 자리 매김을 하기 위하여 변화되고 있다. PLC는 단순히 하드와이어 릴레이 로직을 대체하는데서 시작하였으나, 마이크로프로세서의 발달에 힘입어 크게 발전하였다. 문자열 처리, 프로세스 제어에 관한 기능의 확대, 각종 안전 장치의 부가, 컴퓨터 시스템에 필적하는 고성능 CPU 유닛, 분산 처리 및 네트워크, 상위 컴퓨터를 통한 연계 제어, 입출력 네트워크를 사용한 원거리 입출력 장치의 사용이 일반화되어, DCS에 해당하는 기능의 대형 PLC가 널리 사용된다. 국제적인 움직임으로서 개방형 PLC 및 표준화 PLC, PC-

based PLC, 소프트웨어 PLC 등의 추세로 발전하고 있으나, 실시간성의 보장, 데이터 동기의 개념 결여 등이 문제점으로 남아있다.

결론으로 최근의 PLC에 관련된 기술 동향과 앞으로의 변화 방향에 대하여 언급한다.

#### 가. 고속화

PLC의 응용 범위가 확대됨에 따라 필연적으로 수반되는 현상은 기존의 순차제어 기능 이외의 다양한 제어 기능의 수행이다. 이를 위하여 최근의 PLC는 PID 등과 같은 고수준의 연산 기능이 다수 포함되어 있으며 이들 고수준 연산 명령의 사용 빈도가 갈수록 높아져 가는 실정이다. 이와 같은 연산 명령의 사용 빈도가 증가됨에 따라 PLC의 수행 속도가 현저하게 증가되어 응답 속도가 늦어진다. 기존의 PLC의 응답 지연 시간은 20 - 30msec가 일반적이었으나 최근에는 2 - 3msec이내의 고속 응답을 요구하고 있으며 이는 단위 명령당 처리 시간을 200nsec 이내로 감소시킬 것을 요구하고 있다. 이와 같은 고속의 처리 속도를 만족시키기 위하여 세 가지 접근 방법이 고려된다. 첫째는 고속의 순차제어에 적합한 전용 프로세서의 개발이다. 둘째는 다중 프로세서를 이용하여 속도를 증가시키는 방법으로 병렬 프로세서 구조 또는 분산 처리 구조를 이용할 수 있다. 세번째 방법은 처리 방법의 변화로서 기존의 일괄 입력 - 처리 - 일괄 출력의 스캔 처리 방식을 탈피하여 현 상태를 고려한 처리 방식의 도입이다. 이 경우, 선택적인 수행이 가능하므로 수행속도의 증가를 기대할 수 있다.

#### 나. 시스템 엔지니어링적 설계환경

기존의 PLC가 단순한 순차제어에 주된 목적을 두고 있었기 때문에 프로그램 방식을 비롯한 제어 시스템 설계 방식이 단순 제어 프로그램 언어로 설계되어 왔다. 그러나 아날로그 제어 등과 같은 다양한 기능이 추가됨에 따라 계장제어와 같은 전체 제어 시스템을 설계할 수 있는 시스템 엔지니어링 설계 프로그램이 필요하게 된다. 이와 같은 설계환경은 FA-CAD와 FA-CAE와 같은 형태로 출현하며 기존의 RLL 명령이외에도 SFC 및 계장 제어 명령을 포함하는 설계환경이 제시된다.

#### 다. 개방형 PLC 구조

기존의 PLC는 모든 회사가 자체 기술에 국한된 기술을 바탕으로 독자적인 모델을 개발하여 왔으나, 최근에는 개인용 컴퓨터의 발전에 따라, 개인용 컴퓨터와 범용 입출력 모듈을 이용한 PC-based PLC 및 산업 표준안인 VMEbus 등을 이용한 개방형 PLC의 개발이 활발하다. 이와 같은 개방형 하드웨어와 앞서 설명한 시스템 소프트웨어 개발환경이 결합된 범용 개방형 PLC의 개발이 새로운 PLC 개발 방향으로 추구된다.

본 글의 작성에 조언을 주신 서울대학교 전기공학부 제어정보시스템연구소의 구경훈, 김명준, 이재영, 정승권, 조영철에게 감사드린다.

#### 참 고 문 헌

- [1] G. S. Rho, K. Koo, N. Chang, J. Park, Y. Kim, W. H. Kwon, "Implementation of a RISC Microprocessor for Programmable Logic Controllers," *Microprocessors and Microsystems*, vol. 19, no. 10, pp. 599-608, December 1995.
- [2] G. Michel, *Programmable Logic Controllers - Architecture and Applications*, John Wiley & Sons, 1990.
- [3] W. Stallings, "Reduced Instruction Set Computer Architecture," *Proceedings of the IEEE*, vol. 76, no. 1, pp. 58-75, January 1988.
- [4] Y. Shimokawa, T. Matsushita, H. Furno, and Y. Shimanuki, "A High Performance VLSI Chip of Programmable Controller and its Language for Instrumentation and Electrical Control," *Proceedings of '91 IECON*, pp. 884-889, 1991.
- [5] Daniel P. Siewiorek, "Architecture of Fault-Tolerant Computers," *IEEE Computer*, 17(8), pp. 9-18, 1984.
- [6] Omri Serlin, "Fault-Tolerant Systems in Commercial Applications," *IEEE Computer*, 17(8), pp. 19-30, 1984.
- [7] Daniel P. Siewiorek, "Fault Tolerance in Commercial Computers," *IEEE Computer*, 23(7), pp. 26-38, 1990.
- [8] PLC-5 Backup Communication Module : User Manual, ALLEN-BRADLEY, 1992.
- [9] SIMATIC S5 : S5-135U, S5-155U and S5-155H Programmable Controllers Catalog ST 54.1, SIEMENS, 1994.
- [10] Colin D. Simpson, *Programmable logic controllers*, Englewood Cliff, NJ 07632 : Prentice Hall, 1994.
- [11] I. G. Wornak, *Programmable controllers operation and application*, Englewood Cliff, NJ 07632 : Prentice Hall, 1988.
- [12] K. Venkatesh, M. Zhou, and R. J. Caudill, "Comparing ladder logic diagram and Petri nets for sequence controller design through a discrete manufacturing system", *IEEE Trans. Industrial Electronics*, vol. 1, pp. 611-619, Dec. 1994.
- [13] R. David and H. Alla, *Petri nets and Grafcet*, Englewood Cliff, NJ 07632 : Prentice Hall, 1992.
- [14] International Standard IEC 1131-3, *Programmable Controllers- Part 3 : Programming Languages*, 1993-03.
- [15] 특집 PLC기술 개발동향, 월간 제어계측 1995년 8월, 1995. 8
- [16] 프로그래머블 로직 컨트롤러 GOLDSEC-M series, LG산전주식회사, 36-EMPM-02/94, 1996. 1
- [17] Allen-Bradley Automation Systems, *Rockwell Automation Allen-Bradley Reliance Electric, B111*, 1994.
- [18] Series 90-30 Programmable Controller Installation Manual, GE Fanuc Automation, GFK-0356D, 1992. 12.

[19] International Electrotechnical Commission,  
 Programmable Controllers - Part 2:  
 Equipment requirements and tests, IEC  
 Publication 1131-2, 1992.

[20] 박홍성, 권옥현, “산업용 네트워크와 그 응용”, 제어 자동화 시스템 공학회지 제2권 제4호, pp4-18, 1996. 7

저자 소개

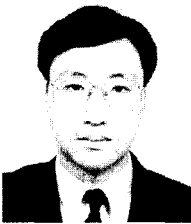


權 旭 鉉

1962年~1966年 서울대 전기공학과 공학사  
 1979年~1971年 " 공학석사  
 1972年~1975年 미국 브라운 대학교 공학박사  
 1977年~ 현재 서울대학교 공과대학 전기공학부 교수

1992年~1996年 서울대학교 제어계측신기술 연구센터 소장  
 현재 서울대학교 공과대학 전기공학부 학부장

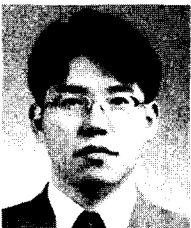
관심분야: 제어용 실시간 컴퓨터 구조, 디지털 시스템, 이산현상 시스템 해석, 공장 자동화



朴 宰 賢

1982年~1986年 서울대 제어계측공학과 공학사  
 1986年~1988年 " 공학석사  
 1989年~1994年 " 공학박사  
 1994年~1995年 미국 미시간 주립대 EECS 박사후 과정  
 1995年~ 현재 인하대학교 자동차 공학과 조교수

관심분야: 실시간 제어용 컴퓨터 구조, 실시간 통신망, 이산현상 시스템 해석



張 來 赫

1985年~1989年 서울대 제어계측공학과 공학사  
 1989年~1992年 " 공학석사  
 1992年~1996年 " 공학박사  
 1996年~ 현재 서울대 제어계측기술연구소 연구원

관심분야: 제어용 실시간 컴퓨터 구조, 디지털 시스템, 내장형 시스템, 이산현상 시스템 해석, 공장 자동화