

산업용 로봇트 주변장치의 분산제어에 관한 연구 (Study on a Distributed Control System for Industrial Robot Peripherals)

김 대 훈* , 변 증 남**
(Kim, Dae-Hoon Bien, Zeungnam)

要 約

교시형 산업용 로봇트의 주변장치들을 산업용 로봇트의 작업동작에 맞춰 제어하기 위한 분산제어 시스템의 설계에 대하여 연구하였다. 설계 제작된 분산제어 시스템은 16 bit machine 의 미니컴퓨터와 수대의 8 bit machine 인 마이크로컴퓨터를 결합시켜 구성하였다. 본 논문에서는 분산제어 시스템의 구조, 시스템운전에 필요한 소프트웨어의 구성 및 분산된 제어모듈간의 통신망에 대하여 약술하였으며 결합회로(Interface) 등을 포함한 실제적인 문제점을 토의하였다.

Abstract

A distributed control system is designed and implemented to control various peripherals of a playback-type industrial robot for synchronized work motion. The distributed control system consists of a 16-bit Mini-computer and several 8bit Micro-computers. In the paper, the structure of the control system, the software requirements for system operation as well as the hierarchical computer network for interfacing module computers and the Mini-computer are briefly described, and practical difficulties including interfacing problem are discussed.

1. 序 論

디지털 컴퓨터의 발달에 따라서 컴퓨터를 이용한 제어시스템은 여러 종류의 산업제작공정들을 제어하는데 사용하여 왔다. 산업제작공정의 규모가 커짐에 따라서, 전체 시스템을 제어하기 위한 소프트웨어 및 하드웨어는 규모에 비례하여 더욱 복잡해지고 있다. 이런 복잡한 대형 시스템을 다루는 제어장치는 설계 및 제작을 위한 코스트도 높을 뿐 아니라, 원하는 스펙(specification)에 맞추어서 제어하기도 어려운 것이 상례이다.

종래에는 대형 규모의 시스템을 제어하기 위하여 큰 기억용량을 갖는 고성능의 대형 컴퓨터를 이용한 중앙식 제어(centralized control) 방법을 기용하였으나, 최근에는 소형의 미니컴퓨터 및 여러 대의 마이크로컴퓨터를 이용한 분산 제어(distributed control)의 개념

을 도입하여 이용하고 있다. 즉, 대형규모의 제어대상물, 기능별로 작은 유닛들로 나누고, 이들 각각의 유닛들은 마이크로컴퓨터에 의하여 분산하여 제어하고, 이들 전체를 미니컴퓨터에 의하여 관리 제어하게 하는 방식이다. 이러한 분산제어의 개념은 화학공정시스템(process control system) 또는 펄프, 석유 제작 공정과 같은 대규모 시스템을 제어하는 경우뿐만이 아니라, 물리적규모는 작더라도 복잡성(complexity)이 스페이스쉽(space ship)의 제어장치등에도 적용될 수 있는 것으로 저렴하고도 신뢰도가 높은 마이크로프로세서등을 포함한 기술의 발전에 의하여 실현 가능케 된 것이다.

본 논문에서는 분산제어방식을 복잡한 시스템의 제어에 실제로 적용할 때의 문제점을 고찰하고 이를 해결하기 위하여 실제의 하드웨어를 이용하여 실험 연구한 내용을 간추려 수록하였다. 이를 위하여 제어대상으로는 한국과학원 전기 및 전자공학과 제어실험실에서 개발한 산업용로봇트, X-Y 테이블 및 교육용 수치제어 공작기계등을 묶어 하나의 대형 시스템으로 선택하였으며, 각각의 제어유닛들은 마이크로컴퓨터

* 正會員, 흥능기계 연구원
(A. D. D.)

** 正會員, 한국과학원 전기 및 전자공학과
(Dept. of Electrical Science, KAIS)

接受日字: 1980年 4月 9日

터에 의하여 제어되도록 하고, 전체는 NOVA-830 미니컴퓨터에 의하여 제어되도록 하는 실험을 설계하였다.

미니컴퓨터는 마이크로컴퓨터에 의해 간섭(inter-rupt)이 걸리는 경우에만 제어루틴(control routine)을 수행하게 설계함으로써, 미니컴퓨터가 평소에 다른 일도 수행하게 설계함으로써, 미니컴퓨터가 평소에 다른 일도 수행할 수 있게 미니컴퓨터의 이용도(availability)를 높게 하였다.

본 논문의 구성은 다음과 같다. 제 2 절에서는 재래식의 중앙제어와 분산제어에 관하여 비교 설명하고 제 3 절에서 본 실험연구를 위하여 구성된 산업용 로봇 및 그 주변장치의 분산제어시스템에 대하여 설명하고 기능 및 동작의 시험 결과를 간단히 토의하였으며, 제 4 절에서 결론을 짓는 것으로 한다.

2. 중앙제어(Centralized Control)와 분산제어(Distributed Control)

2.1 중앙제어

종래에 사용되던 컴퓨터 컨트롤시스템은 보통 중앙제어식 방식을 채용하고 있다. 즉, 제어대상인 플랜트의 상태(state)나 출력(output)의 씨그날을 감지(sensing)하면 보통은 이 씨그날을 multiplexing하고 conditioning 하는 장치를 거쳐 한 개의 대형컴퓨터를 사용한 제어실에 중앙식으로 연결하여서 제어하는 방식을 채택하고 있는 것이다.

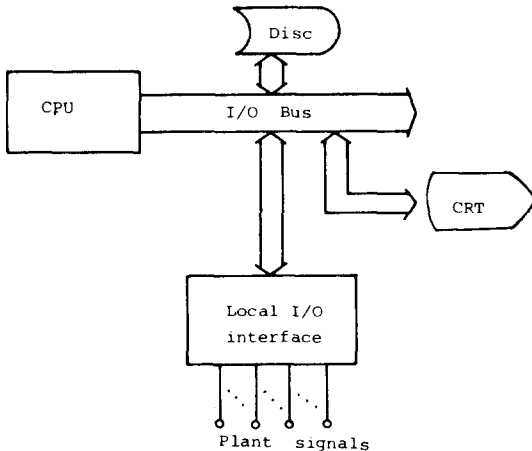


그림 1. 중앙제어 system
Fig.1. Centralized control system.

그림 1은 중앙제어 시스템의 구성을 간단한 다이어그램으로 나타낸 것이다. 중앙제어 시스템은 다음

에 설명하는 분산제어 시스템에 비교하여 다음과 같은 단점이 있다.

- (1) 대형 규모의 시스템을 위한 고신뢰도의 제어장치는 복잡하고, 중앙식 배선 및 잠음제거를 위한 보호장치 등으로 인하여 제어시스템 설계제작비가 높아진다.
- (2) 소프트웨어를 개발하기가 복잡하다.
- (3) 이미 설치한 시스템의 규모를 더 크게 할 때 전체적으로 고쳐야 할 하드웨어 및 소프트웨어의 규모가 방대하다. 즉 융통성(flexibility)이 적다.

2.2 분산제어

제어할 플랜트를 기능별로 나누어서, 제어하는 분량 자체를 분산시킬 수 있다.

분산제어 시스템은, 일반적으로 master 컴퓨터, communication link, slave 컴퓨터를 중심으로 이루어진다. Master 컴퓨터는 slave 컴퓨터보다 high level 컴퓨터로서, 일반적으로 미니컴퓨터가 사용된다.

Master 컴퓨터는 communication link를 통하여 slave 컴퓨터들로부터 정보를 받기도 하고 이에 제어 명령을 발하기도 하면서 전체 시스템을 제어한다.

Communication link는 master 컴퓨터와 slave 컴퓨터를 연결하는 device 이다. Slave 컴퓨터는 기능별로 나누어진 유닛트를 자체 제어 알고리즘에 따라 제어하며, master 컴퓨터와의 정보교환을 통하여 master 컴퓨터에 의해 제어된다. 이러한 구성의 분산제어 방식은 상대적으로 적은 규모의 정보교환 비율 및 I/O를 가질 때 쓰인다. Slave 컴퓨터가 master 컴퓨터에게 data를 보낼 때는:

- (1) Slave 컴퓨터는, master 컴퓨터와 slave 컴퓨터 사이의 시스템 link에 쓰이는 bus의 제어를 요구하며 이를 enable 시킨다.
- (2) Data를 master 컴퓨터에게 보낸다.
- (3) Slave 컴퓨터는 master 컴퓨터로부터의 인식 응답신호를 기다린다.
- (4) 응답신호를 받으면 slave 컴퓨터는 bus의 제어를 disable 시키고, 자신이 하던 일을 계속한다.

Master 컴퓨터가 slave 컴퓨터에게 data를 보낼 때는 slave 컴퓨터에게 data를 보낸 다음, 인식 응답신호를 기다릴 필요는 없다.

단, 여러 개의 data를 보낼 때는 각각의 data에 대한 인식응답신호를 받은 후, 다음 data를 보낸다.

이러한 분산제어 시스템이 중앙제어 시스템에 비해서 우수한 점은 다음과 같다.

- (1) 융통성이 크다. 즉, 시스템의 규모가 커짐에 따

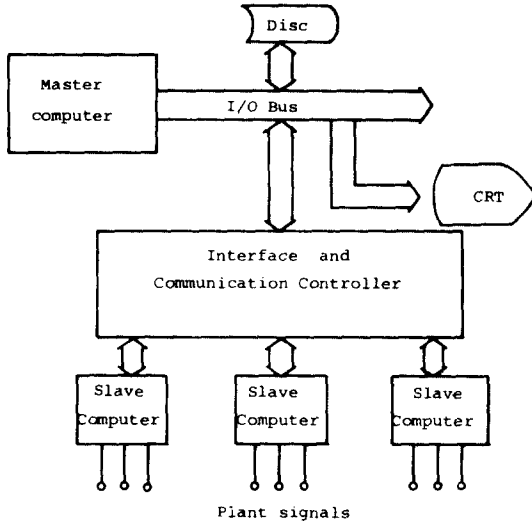


그림 2. 분산제어 system (master / slave type)
Fig.2. Distributed control system.
 (Master/slave type)

라 쉽게 시스템을 확장시킬 수 있다.

(2) 대형시스템을 나누어서 제어하는 것이므로 소프트웨어가 중앙제어 시스템에 비교하면 훨씬 간단해진다.

(3) Physical layout 에서의 코스트가 많이 절감된다.

(4) 각각의 slave 컴퓨터들은 독립적으로도 동작할 수 있으므로, 전체 시스템 중에서 일부가 고장이 난 경우에도 나머지 부분은 계속 동작할 수 있다.

이와 같은 분산제어 시스템은 특히 큰 도시의 교통제어와 같은 초대형 시스템에 쓰이면 유용하다. 즉, 각각의 교차로는 slave 컴퓨터인 마이크로컴퓨터에 의해서 제어하며, 모든 교차로의 교통량의 정보를 master 컴퓨터가 받아들여서, 시내 전체의 교통량을 최적제어하기 위한 정보를 마이크로컴퓨터에게 보내어

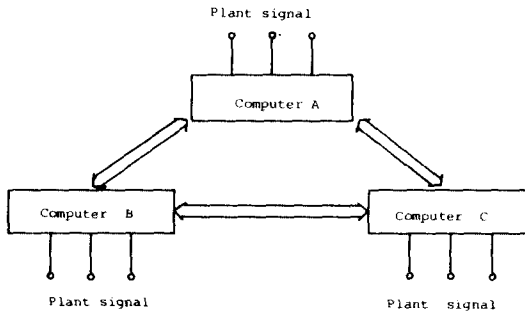


그림 3. Master/master type의 분산제어 시스템
Fig. 3. Distributed control system.
 (Master/master type)

서 제어하는 분산제어 방식은 예컨대 서울시의 교통제어를 위해 사용될 수 있을 것 같다.

분산제어 기술을 실제화하기 위하여 사용되는 또 다른 전형적인 방법으로 master/master 시스템의 구조(organization)를 갖는 형태를 생각할 수 있다.

그림 3에 보인대로 높은 정보교환율과 높은 I/O activity가 예상될 때 쓰이며, 이외에도 공통 기억공간을 이용한 방법들의 변형된 형태가 있으나 자세한 기술은 생략하기로 한다.

3. 산업용로봇 및 주변장치를 위한 분산제어 시스템

3.1 분산제어 시스템

본 논문에서 분산제어 시스템의 실제화(realization)를 위하여 선택한 제어대상은, 산업용로봇 X-Y 테이블 및 수치제어 공작기계이다.

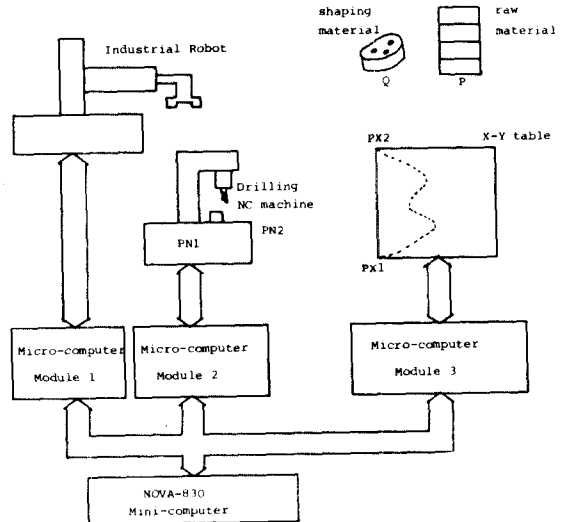
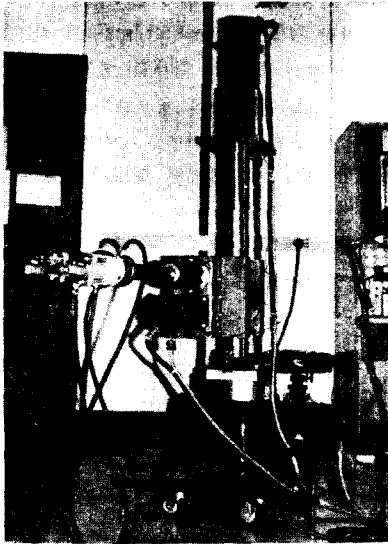


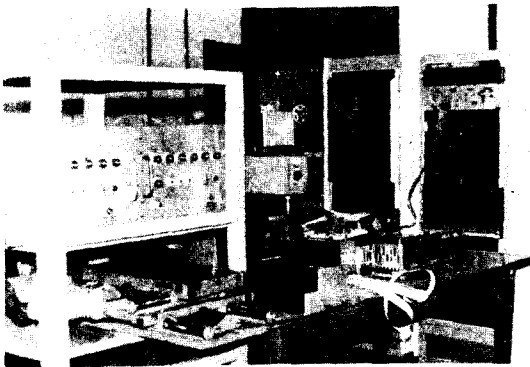
그림 4-A. 산업용 로봇 주변장치의 분산제어 시스템
Fig. 4-A. Distributed control system industrial robot peripherals.

그림 4-A는 전체 시스템의 구조를 블록도로서 보이고 그림 4-B는 사용된 실제 시스템의 실제 배치도로서 산업용로봇, X-Y 테이블 및 NC 공작기계의 제어대상을 포함하여 사용된 미니컴퓨터 및 마이크로컴퓨터 등을 보이고 있다.

설계한 분산제어 시스템을 이용하여, 산업공정에서의 작업을 simulate 하기 위하여 다음과 같은 작업내용을 예로 선택하였다. 즉, 먼저 로봇트는 가공하지 않은 물건을 P점에서 집어서, X-Y 테이블의 Px_1



(a) 산업용로봇



(b) X-Y table, NC machine 등을 포함한 분산 제어 시스템의 실물 배치도

그림 4-B. 분산 제어 시스템의 실물 사진
Fig. 4-B. Photograph of distributed control system under consideration.

점에 놓는다. 그러면 X-Y 테이블은 주어진 경로에 따라 움직이며 물건을 가공한다. 가공이 끝났을 때는 물건이 Px_2 점에 있으며, 로봇트는 물건을 집어서 수치제어 공작기계의 PN_1 점에 놓는다. 동시에 X-Y 테이블은 처음의 위치인 Px_1 으로 돌아온다. 수치제어 공작기계는 주어진 위치에 알맞게 구멍을 뚫어서 가공이 완전히 끝난 물건을 PN_2 점에 놓는다. 수치제어 공작기계가 동작하는 도중에 X-Y 테이블이 처음 위치에 도달하면 산업용로봇트는 가공이 안된 물건을 X-Y 테이블에 갖다 놓을 수도 있다. 로봇트는 가공

이 끝난 물건을* 집어서 Q점에 놓는다. 동시에 수치제어 공작기계는 처음의 위치인 PN_1 으로 돌아온다. 이와 같은 과정을 계속 반복하면서, 미니컴퓨터는 3개의 제어대상의 상태를 받아들여 그때그때에 알맞는 지령(command)을 마이크로컴퓨터에 의해서 제어된다. 또한 마이크로 컴퓨터에서 계산하기에 시간이 걸리거나 할 수 없는 계산은 미니컴퓨터가 계산하도록 하며, 미니컴퓨터는 간섭이 걸릴 때만 제어를 위한 기능을 수행하기 때문에, 다른 프로그램을 수행하면서도 위에서 설명한 분산제어 시스템을 전체적으로 제어할 수 있다.

3.2 Computer Communication system 구조

본 실험연구에서 사용된 NOVA 830 미니컴퓨터와 Z-80 또는 intel 8080 A를 CPU로 하는 마이크로 컴퓨터의 word length가 각각 16bit과 8bit로 서로 맞지 않으므로 마이크로컴퓨터측에서는 두 개의 input/output ports를 사용하여 미니컴퓨터와 정보를 주고 받아야 되며 이를 위한 하드웨어 interface 및 소프트웨어의 구성은 단일 컴퓨터를 사용할 때에 비하여 복잡하여 word format, processor interface 등 고려되어야 할 사항이 많다.

그림 5는 설계된 분산제어 시스템의 컴퓨터 communication 시스템의 구조이다. multiplexing 및 demultiplexing unit는 미니컴퓨터와 여러 개의 마이크로 컴퓨터 사이에 있는 communication bus를 연결시켜 주는 일을 한다. 미니컴퓨터에 있는 digital I/O는 16digital input, 16digital output, one external interrupt 단자로 이루어진 다목적용 interface device이다. 그림 6은 DIO(digital I/O)의 구성을 블록도서 보이고 있다. 마이크로컴퓨터에 있는 interface board는 미니컴퓨터와 마이크로컴퓨터 사이의 interface를 위한 circuit board이다.

그림 7은 interface board의 기능을 표시해 주는 functional block diagram이다. 이 그림에서 알 수 있듯이 마이크로컴퓨터의 input port, output port는 16bit의 미니컴퓨터 data bus와 연결하기 위해서 2개의 port로 이루어졌다. 미니컴퓨터에서 데이터를 보내려면, 먼저 vectored interrupt에 의해서 특정한 마이크로컴퓨터에 간섭이 걸리고 그 다음 2개의 output port를 통하여 데이터가 읽혀 들어가서 주어진 작업을 수행한다. 역으로 마이크로컴퓨터에서 데이터를 보낼 때는 데이터를 output port에 set하고, interrupt signal 회로에서 미니컴퓨터에게 간섭을 걸면, 미니컴퓨터는 데이터를 읽어 들여서 서비스를 수행한다. Control port는 미니컴퓨터와 마이크로컴퓨터

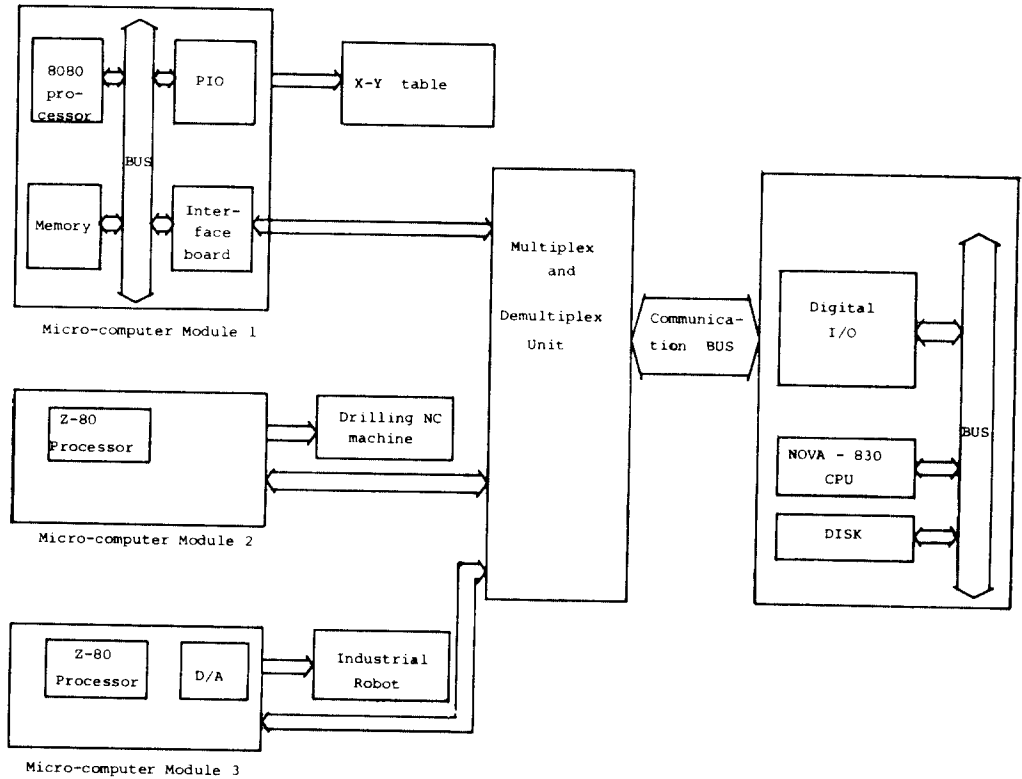


그림 5. 컴퓨터 커뮤니케이션 시스템의 구성도

Fig. 5. Organization of computer communication system.

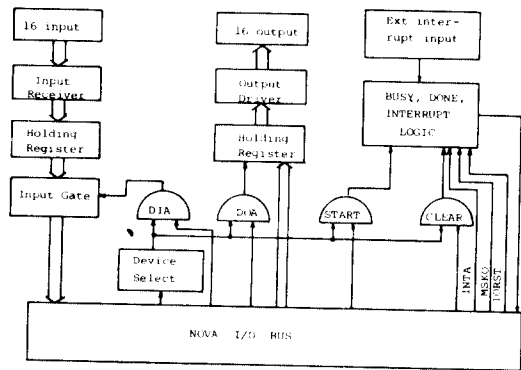


그림 6. DIO의 기능적 블록도

Fig. 6. Functional block diagram of DIO. (Digital I/O)

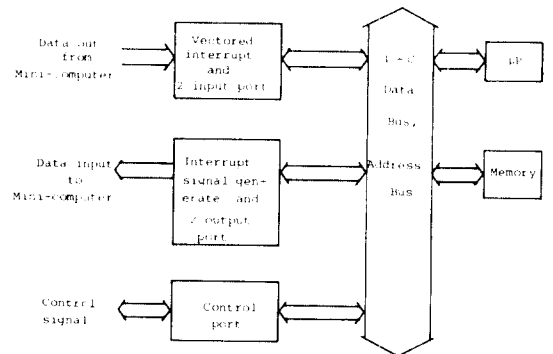


그림 7. Interface board의 기능블록도

Fig. 7. Functional block diagram of interface board.

터 사이의 response signal을 주고 받는 port이다. 본 실험연구를 위하여 이용된 NOVA-830 미니컴퓨터와 Z-80 및 intel 8080 과의 interface 회로 및 communication 소프트웨어는 참고문헌(1)에 자세히 수록되어 있다.

3.3 Software Organization

미니컴퓨터에게 명령이나 data를 보낼 때는 DIO (digital I/O)를 통해서 명령이나 data를 보내면, 마이크로컴퓨터는 vectored interrupt에 의해 명령을 받아들이는 routine, 또는 data를 받아들이는 routi-

ne 을 수행한다. Interrupt routine을 수행한 후는 마이크로컴퓨터는 원래 하던 일을 다시 수행한다.

마이크로컴퓨터에서 미니컴퓨터에게 명령이나 정보를 보내는 과정은 다음과 같다. 16 bit word 중에서 bit 1에서 bit 2는 마이크로컴퓨터의 code로 사용하고, bit 3에서 bit 7은 미니컴퓨터에게 보내는 명령 또는 status를 나타낸다. Bit 8에서 bit 15는 data를 나타낸다.

마이크로컴퓨터에서 2개의 output port를 통해서 16bit word를 미니컴퓨터의 DIO에 보내면 동시에 미니컴퓨터는 간섭에 의해 마이크로컴퓨터가 보낸 word를 DIO input port를 통해서 읽는다. Bit 0에서 bit 2에 의해서 어떤 마이크로컴퓨터가 data를 보냈는가를 알고, bit 3에서 bit 7에 의해 어떤 종류의 명령이나 status인가를 알고, data를 받아들이는 경우는 bit 8에서 bit 15의 data를 읽어들이어서, 그에 알맞는 routine을 수행한 후 미니컴퓨터는 원래의 main program routine으로 돌아가서, 미니컴퓨터는 다른 routine가 사용할 수 있다. 즉, 분산제어 과정에서

에서 미니컴퓨터는 마이크로컴퓨터에 의해 간섭이 걸린 경우에만 routine을 수행함으로써, 미니컴퓨터는 다른 여러 가지 목적으로도 사용될 수 있어서 이용성을 높일 수 있는 것이다.

분산제어 시스템의 과정을 순서적으로 설명하면 다음과 같다. 즉, 마이크로컴퓨터는 X-Y 테이블의 경우, milling path에 대한 data, 수치제어 공작기계의 경우는 drilling point에 대한 data 등의 초기 data를 미니컴퓨터로부터 받아들인 후 작업개시 지령을 기다린다. 이때 미니컴퓨터는 조작자의 지령을 받아 현재의 상태에 알맞는 명령을 마이크로컴퓨터에 내리면, 각각의 마이크로컴퓨터는 주어진 명령에 따라서 control routine을 수행하고, 다음을 기다린다. 그러면 미니컴퓨터는 받아들인 정보에 따라서 다음에 할 일을 결정하고, 마이크로컴퓨터에게 명령이나 정보를 보낸다. 마이크로컴퓨터는 control routine 수행중에 어느 때라도 미니컴퓨터에게 간섭을 걸어서 복잡한 계산을 의뢰할 수 있다.

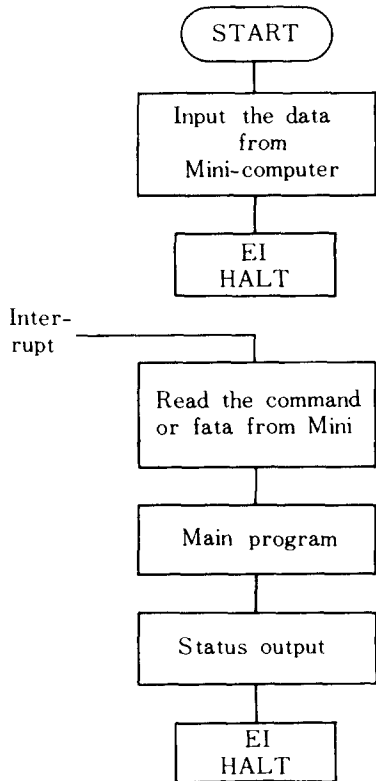


그림 8. 마이크로컴퓨터를 이용한 제어장치의 소프트웨어 구성도

Fig. 8. Software organization of microcomputer-based controller.

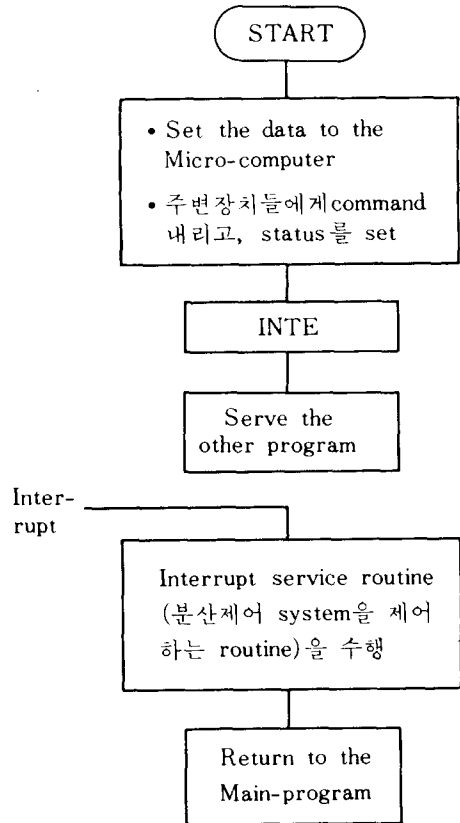


그림 9. 미니컴퓨터를 이용한 제어장치의 소프트웨어 구성도

Fig. 9. Software organization of minicomputer-based controller.

이와 같은 제어과정의 구성도를 그림 8 및 그림 9에 요약하여 도시하였다. 즉 그림 8은 NC 공작기계, Drilling machine 및 산업용로봇트의 제어에 쓰이는 마이크로컴퓨터를 사용한 제어장치의 제어 소프트웨어 구성도로서 각 모듈은 initialization routine, 각각의 제어 목적에 맞는 service routine의 소프트웨어를 포함하고 있다. 그림 9는 분산제어 시스템의 high level controller로서 NOVA 830 minicomputer를 사용한 제어장치의 소프트웨어 구성도이며, command control, status setting, data setting 및 마이크로컴퓨터 사이의 communication 제어를 하는 기능을 하며, 역시 initialization routine, status determining routine, incoming status service routine 등을 포함하고 있다. 이들 제어장치의 상세한 flow chart 및 제어하드웨어와 program listing은 참고문헌(1) 및 (2)에 수록되어 있다.

3.4 분산제어 시스템의 동작기능시험 및 토의

본 연구를 통하여 이상의 3-1 절 이하에서 기술한 분산제어 시스템을 구성하여 시스템의 동작기능시험을 행하였다. Interface 하드웨어 및 제어소프트웨어는 수차례의 시행착오를 거쳐 개선되었다. 특히,

(i) 미니컴퓨터와 마이크로컴퓨터 제어장치와의 거리(physical distance)가 수 미터 이상이 되므로 잡음으로 인한 오동작(malfunction)이 때때로 야기되었는 바, 이 문제는 컴퓨터들을 연결하는 communication bus에 있는 ground line들 사이에 data line을 삽입하여 해결하였고,

(ii) 타이밍을 위하여 요구되는 시간지연(time delay)은 전형적인 RC 회로를 이용하여 구성하였으나 이들이 연결되는 회로의 입력임피던스가 큰 경우에는 one-shot 회로를 이용하였고,

(iii) 미니컴퓨터의 출력신호를 마이크로컴퓨터가 읽어들이도록 하기 위하여 signal latching을 위한 회로 구성등이 별도로 고려되었으며,

(iv) Interface 중 여러 점에서 signal level의 matching을 위한 작업, 즉 pull-up resistor의 사용등이 필요함을 인지하였다.

본 연구를 위하여 개발된 제어알고리즘을 위주로는 소프트웨어 및 interface 하드웨어의 상세한 회로는 참고문헌(1)에 수록되었는 바 이들로 미니컴퓨터를 다른 사용자가 임의의 프로그램을 수행토록 하면서 산업용로봇트 주변장치를 분산제어하는 과정이 수행됨을 관찰하였다.

본 연구에서 다른 분산제어 시스템과 하나의 고정

능 미니컴퓨터를 사용하는 중앙제어시스템에 대한 정성적(qualitative)인 차이점은 2-1 절에서 지적된 것에 첨가하여 좀 더 구체적으로 다음과 같이 이해될 수 있을 것이다. 즉

(i) 일반적으로 시정수(time-constant)가 작은 써보기구를 컴퓨터를 이용한 디지털방식으로 제어하는 경우 real-time control의 중요한 설계요건인 샘플링주기(sampling period)의 선택에 따라 시스템의 상대적 안정도(relative stability)가 크게 좌우되는 바 중앙식 제어시스템은 multiplexing에 의해 선택의 자유도가 제한받는데 비해 분산식제어 시스템은 자유롭다.

(ii) 중앙제어 시스템에서는 미니컴퓨터가 시스템제어만을 위해 사용되어 performance/cost비가 낮는데 비해, 분산식제어 시스템에서는 제어기능외에 중요한 다른 데이터 처리기능을 동시에 행할 수 있음으로써 고가인 미니컴퓨터의 이용도가 좋다.

(iii) Interface 하드웨어나 제어알고리즘의 실현(implementation)에 있어서 다양한 signal conditioning, noise 처리 등 분산제어 시스템에서 경우에 따라 더 고려되어야 할 사항이 많다는 단점이 있다.

이상에서 검토된 사항외에, 이들 두 방식은 신뢰도, 코스트, 확장성 등에 대하여 공장환경에서 계수적으로 비교되는 것이 바람직하며, 특히 분산제어 시스템의 경우 16-bit의 마이크로컴퓨터(Z-8000, M6 8000 등)의 등장으로 새로운 형태의 분산제어 시스템이 제안될 수 있을 것이다.

4. 결 론

대형 규모의 산업제작 공정이나 다른 정교한 구조를 갖는 복잡다단한 시스템을 제어하는데 있어서는, 재래식의 중앙제어(centralized control)와 비교하여 분산제어(decentralized or distributed control) 방식을 사용함이 코스트면, 신뢰도, 확장성 및 복잡성 등의 관점에서 유리함이 여러 가지 응용논문들을 통하여 입증되고 있다. 본 논문에서는 산업용로봇트와 주변장치로서 교육용 수치제어 공작기계, X-Y 테이블등을 포함하는 정교한 구조의 전체 시스템을 분산제어하기 위하여 구성한 하드웨어 및 소프트웨어의 구조에 대해 간단히 설명하였다. 실제의 하드웨어와 본 논문에서 묘사된 일체의 제어시스템 및 제어과정은 제어 소프트웨어를 사용하여 실험관찰한 결과를 수록하였다. 미니컴퓨터는 마이크로컴퓨터가 쓸 작업을 알고리즘을 program file로 저장할 수 있고 마이크로컴퓨터는 필요할 때 수시로 찾아 사용할 수 있으므로 최

적제어, 적응제어를 행할 수도 있다.

현재는 한 개의 미니컴퓨터에 3개의 마이크로컴퓨터가 연결되어 있으나, 16bit word 중에서 bit ϕ 에서 bit 2가 마이크로컴퓨터의 code로 사용하기 때문에 8개까지 연결될 수 있는 융통성이 있다.

마이크로컴퓨터의 assembler language 가 보다 기능이 우수한 미니컴퓨터의 assembler, basic, fortran language 등과 연결될 수 있으므로, 보다 큰 기능을 할 수 있는 시스템을 개발할 수 있을 것이다

참 고 문 헌

1. Daehoon Kim, "Implementation of a Distributed Control System for Industrial Robot Peripherals" M. S. Thesis, KAIS, 1980.
2. Seungpil Kim, " μp based Control System for Playback type Industrial Robot", M. S. Thesis, KAIS, 1980.
3. Robert M. Amunson, "A Hierarchical Network for Data Acquisition and Control", IEEE Trans. on IECI, Vol. 23, No. 1, Feb. 1976.
4. Wladyslaw Findeisen, "On line Hierarchical Control for Steady-state systems", IEEE Trans on Automatic Control. Vol. 23, No. 2, April, 1978.
5. Satish Chandra, "A Communication Network for Distributed Data Acquisition and Control in Industrial Plants", IEEE Trans. on IECI, Vol. 25, No. 3, Aug. 1978.
6. Abraham Hirsch, "Design Constraints for a UART-Based Mini-computer Communication Interface", Computer Design, June, 1977.

