

자동화 공정내의 워크스테이션 제어기 통신 소프트웨어 개발

도 성 희*, 박 경 진* , 강 무 진***

Communication Software Development for a Workstation Controller in a CIM System

Sung-Hee Do, Gyung-Jin Park, Mu-Jin Kang

Abstract

Computer Integrated Manufacturing(CIM) is an emerging technology which aims at the total automation of a manufacturing system. Despite the extensive research on CIM in this decade, total automation seems to be far from reality for many reasons. In this study, the problems of implementing a CIM system are analyzed and a solution method is suggested. A set of software has been developed and tested for a workstation controller which controls closely integrated equipment in the low level of a flexible manufacturing system. The functions of the programs are communications between machines and computers and control of workstation scheduling. The software group is developed on IBM OS/2 operating system with 'c' language to enhance the portability. The usage of the software is presented to users through the Presentation Manager which is menu handler of OS/2 system. Although the software is developed for limited machine equipment, the scheme can be adopted to various machines due to the generality.

1. 서 론

컴퓨터 통합생산(Computer Integrated Manufacturing, CIM) 기술은 재래적 개념으로는 기계적 자동화(Mechanical Automation) 즉 주로 기계공업에 있어서 자동생산에 대한 자동화의 의미에 원격조작과 자동제어에 의한 공정 자동화(Process Automation) 및 컴퓨터에 의한 경영관리 정보의 자동 처리와 결정을 대상으로하는 사무 자동화(Office Automation)에서의 정보(Information)의 요소를 추가로 활용하여 설계, 가공과 조립, 검사, 저장 및 운반, 그리고 생산관리간의 정보교환을 통합적으로 제어, 응용함으로써 전 생산공정

의 컴퓨터를 통한 자동화를 구현해 총체적 최적생산을 추구하는 생산형태로서 종래의 개념으로서는 양립된 생산성(Productivity)과 유연성(Flexibility)을 동시에 얻고자하는 첨단 생산기술이다. 더욱 더 짧아지는 제품 개발주기(Product Life Cycle)와 유연성이 중요시되는 다품종 소량생산(Many-Item-Small-Quantity) 체제로의 전환이 시급히 요구되는 근래의 생산개발 환경에 적응하기 위해서 종래의 고정된 자동화 기술로는 한계가 있기 때문에 생산의 제반기능을 정보처리의 통합화를 통하여 최적으로 조화시키려는 결과로 얻어지는 기술이 곧 컴퓨터 통합생산 기술이다.

이러한 컴퓨터 통합생산을 추구하는 방법에는 여러가

* : 한양대학원 기계공학과

** : 한양대학교 공학대학 기계공학과

*** : 한국과학기술연구원

지가 있는데 선진국의 예를 보면 국가마다 서로 다른 방법으로 연구 노력을 기울이고 있으며, 많은 시행착오를 거치고 있는 중이다.^{1, 2, 3} 국내에서 추진되고 있는 컴퓨터 통합생산을 위한 연구는 이미 국책사업으로 통합생산 자동화(CIM) 국책 연구개발 사업단이 발족되어 한국과학기술연구원(KIST)의 주관으로 연구를 하고 있다.⁴

현재 컴퓨터 통합생산 기술에 대한 실제 생산공정은 많은 경우 아직 실용화 단계에 있지 못하고 있으며 생산기술을 세계적으로 주도하고 있는 소수의 연구기관 및 기업에서 모델 플랜트를 시험 운영하고 있는 정도에 그치고 있다. 그 이유는 여러가지가 있으나 가장 큰 원인을 살펴보면 대략 두가지 정도로 집약할 수 있다. 첫번째는 컴퓨터 통합생산 기술개발의 추진에 있어서 그 전략을 수립하기가 쉽지 않다는 것이다. 시스템 통합에 대한 종합적 계획(Planning)과 그 계획에 대한 확인(Verification) 과정이 제대로 되어있지 않다면 막대한 시설 및 연구투자를 유용한 결과로 연결시키기 위한 노력에는 한계가 있기 마련이다. 컴퓨터 통합생산의 계획과 아울러 각각의 플랜트(Plant) 특성에 맞는 공장 제어 시스템을 개발하고 도입하는 과정을 성공리에 달성하기 위해서는 수없이 많은 보완과 시행착오를 거쳐야 한다. 한편, 컴퓨터 통합생산을 이룩하기 위한 목표 달성에 있어서 마치 병목현상처럼 집중적으로 논쟁의 대상이 되고 있는 문제점으로는 통신의 어려움이 있다. 공장내에있는 기기들은 대부분 다른 기기와 협력없이 혼자 작동될 수 있도록 만들어져 있거나, 혹은 자기 자신만이 사용하는 제어 방식을 가지고 있으므로 통신이나 제어를 대단히 어렵게 만든다. 이러한 통신의 어려움으로 발생하는 것이 자동화의 고립(Island of Automation) 현상이며, 생산 관리 시스템, 설계 자동화 시스템, 제조 자동화 시스템 등을 유기적으로 통합된 시스템으로 만들려는 컴퓨터 통합생산을 위해서는 시급히 해결해야 할 문제점인 것이다.

본 연구에서는 생산공장의 하부구조, 즉 현장의 기기들과 통제하는 컴퓨터가 맞물려있는 부분에 대한 전략을 수립하여 어느정도 보편화 되어있는 기기들의 공정 제어(Process Control) 및 통신에 관해 연구하고 이를 구현하는 소프트웨어를 제작하고 실험을 거쳐 검증하려 한다. 초기 단계에서는 퍼스널 컴퓨터(Personal Computer, PC)상에서 프로그램의 실험을 하고 이를 바탕으로 다중작업 오퍼레이팅 시스템(Multi-Task

Operating System)으로의 전이를 꾀할 것이다.

2. 컴퓨터 통합생산 환경에서의 통신 방법

2-1 컴퓨터 통합생산에서의 통신

컴퓨터 통합생산에서는 제품의 설계, 가공과 조립, 검사, 저장 및 운반 그리고 생산관리간의 정보 교환을 종합적으로 제어한다. 따라서 정보의 흐름을 어떻게 효율적으로 유지하는가가 주요 관건으로 대두되며 이는 각 시스템간의 통신을 정의하는데 필수적이다. 또한 각 시스템과의 통신을 구현하는데 있어서 프로토콜(Protocol)이라고 불리는 각기기의 통신규약을 사용한다. 프로토콜은 정보를 전달하는 송신기와 수신기가 데이터의 교환을 통제하는 일련의 규칙을 정해 놓고 그 규칙을 함께 지키면서 정보전달의 신뢰도를 높이려는 모든 노력을 의미하며 통신을 위해서는 양자의 동일한 프로토콜의 사용이 필수적이다.

한편 컴퓨터 통합생산 기술을 추진해 나가기 위한 모델은 시험 운영 되고있는 몇몇 모델 플랜트들이 있으며 이들 모델 플랜트들은 각기 고유의 계획(Planning)과 그 계획에 대한 확인(Verification) 과정을 거쳐 운영되고 있다. 본 연구에서는 이들중 NIST(National Institute of Standards and Technology)에서 추진하고 있는 그림 1과 같은 형태의 모델을 기본 바탕으로 하고 있는데 그 내용은 다음과 같다.⁵

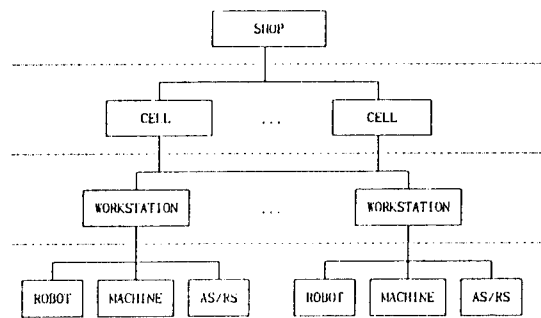


Fig. 1 CIM Hierarchy

공장 레벨(Shop Level)에서는 제품의 수주에 따른 작업주문(Order)에 대한 전반적인 계획을 담당한다. 셀 레벨(Cell Level)에서는 생산에 관한 스케줄(Scheduling)작업을 담당하며 이들 두 부분을 컴퓨터 통합생산에서의 상부구조라 부르기도한다. 워크스테이션

레벨(Workstation Level)에서는 여러 단위기기들의 조합된 작업을 제어하는 역할을 담당한다. 장비 레벨(Equipment Level)에서는 하나의 단위기기로 구성되어 있으며 각 기기들의 작업을 통제하는 역할을 담당한다. 이들 두 부분을 컴퓨터 통합생산에서의 하부구조라 부르기도 한다.

앞에서 정의한 각각의 기능에 따라서 컴퓨터 통합생산에서는 제품의 설계, 가공과 조립, 검사, 저장 및 운반 그리고 생산관리간의 정보를 상부구조에서 하부구조로 혹은 하부구조에서 상부구조로 전달 하며 이들 정보의 적절한 유지와 구성에 따라 생산성과 유연성을 향상시킬 수 있다.

2-2 컴퓨터 통합생산의 하부구조에서의 통신

현재 국내외를 통해 컴퓨터 통합생산을 추구하고자 공장내에 있는 기기들을 컴퓨터를 이용해 통합하고 제어하는 시도가 많이 이루어지고 있다. 현대의 컴퓨터와 하나의 기기(NC/CNC, Robot등)를 연결하여 자료를 전송하고 상태를 감지(Monitoring)하는 일들은 상당히 진전되어 있으며 그에 대한 국부적인 소프트웨어도 어느 정도 개발되어 있다. 즉 NC/CNC 이용시 컴퓨터의 CAD/CAM(Computer Aided Design and Manufacturing) 소프트웨어에서는 생산된 NC/CNC 프로그램을 직접 전송하는 DNC의 개념이나 Robot등에서 Off-Line Programming 기술등이 그것이다. 그러나 각기 사용되는 기기마다 다른 소프트웨어가 사용됨으로 인해 앞에서 언급한 자동화의 고립 현상이 발생하게 되었다. 즉 각 산업체별로 자사에서 정의한 프로토콜에 의해서 자사제품끼리의 통신만 가능하며 따라서 이종기기간의 협력은 거의 불가능하다. 한편 몇대의 Robot나 NC/CNC 등의 기기류를 컴퓨터에서 소프트웨어적으로 통합하여 제어하는 문제는 통합하려는 전략의 부족과 함께 그 통신 방법의 다양성으로 인해 그 중요성과 비교해 볼 때 거의 시도되지 않고 있으며 제한적으로나마 고가의 소프트웨어를 이용해 해결하려는 경향이다. 이를 위해 외국에서는 LAN(Local Area Network)을 통한 제어 즉 MAP(Manufacturing Automation Protocol) 등이 주장되어 연구를 계속하고 있으나, 규모가 대단히 커다란 점 때문에 너무 고가이고 플랜트에 있는 각 기기들을 통제하기 까지는 어려운 점이 많이 도사리고 있다.

컴퓨터 통합생산의 상부구조에서의 통신은 생산에 필

요한 모든 계획등에 대한 정보를 다루고있기 때문에 정보의 특성상 요즘 많이 사용되는 LAN의 기술을 이용한 네트워크(Network)를 구성하여 통신을 한다. 즉 LAN에서 사용하는 몇가지 기술을 필요에 따라 응용하여 네트워크를 구성하며 그 예로는 Ethernet, MAP등이 있다.⁴⁾

하부구조에서의 통신은 상부구조에서 처럼 네트워크를 구성하여 이들 네트워크들을 통한 통신 방법을 사용하지 않고 공장 내부에서 동작되는 기기들과 워크스테이션 역할을 하는 컴퓨터와의 직접적인 연결을 통한 통신을 시도한다. 따라서 네트워크에서처럼 시스템적인 방법이 아닌 각 장비에 맞는 독자적인 방법에 따라 통신을 실현하여야만 하며 이를 구성하는 방법은 일반적으로 두가지가 있다. 하나는 통신에 필요한 독특한 약속이 규정된 프로토콜이 복잡한 경우의 통신 방법이 있고, 다른 하나는 프로토콜이 단순하여 비교적 간단히 통신을 실현하는 방법이 있다.

자동화 공장 내부에서 동작되는 기기들은 그 제품을 만드는 회사들도 다르고 종류도 다양하다. 따라서 통신에 필요한 약속인 프로토콜 또한 다르며 같은 회사 제품이라도 Robot와 NC/CNC등 기종에 따라서도 그 통신 방법이 다르게 되는 경우가 많다. 이들 기기중에서도 컴퓨터와 통신을 하기위해서는 몇가지 복잡한 과정을 거치도록 설계되어 있는 기기들이 있는데 이러한 경우 이들 기기와의 통신을 위해서는 소프트웨어적으로 그 복잡한 과정을 처리해야 하며 상당히 어려운 통신 소프트웨어가 필요하다. 이처럼 프로토콜이 복잡하게 정의되어있는 경우의 통신 소프트웨어를 프로토콜 변환기(Protocol Converter)라 부른다.

한편, 프로토콜이 단순한 경우의 통신 방법은 컴퓨터와의 통신을 하기 위해서 복잡한 과정이 없이 어느 일정한 조건만 충족되면 통신이 가능하도록 설계되어 있는 기기들과 통신을 하는 방법으로서 이러한 경우의 통신 소프트웨어를 디스패처(Dispatcher)라 명명한다. 프로토콜 변환기(Protocol Converter)와는 비교도 안될 정도로 간단히 통신을 할 수 있으며 기계 가공 공장내의 NC 동작기계군을 분산제어 방식으로 종합관리 할 수 있는 일종의 DNC(Distributed Numerical Control) 개념으로 사용되기도 한다.⁵⁾

컴퓨터 통합생산의 하부구조에서는 공장 내부에서는 동작되는 자동화 기기들의 종류가 많을 뿐더러 제품을 만드는 회사, 제조 시기, 기능들이 천차만별이어서 이

들을 통합한 전체시스템을 구성하기가 어렵기 때문에 컴퓨터와 이들 기기간의 통신의 문제가 최대의 관점이 되고 있다. 따라서 공장 내부에서 작동하는 기기들을 어떻게 정의하고 또 이들 기기들을 연결하는 방법은 어느 방법을 사용하는가에 따라 컴퓨터 통합생산의 성패가 좌우된다고 할 수 있겠다.

3. 워크스테이션의 기능과 통신

3-1. 기능

컴퓨터 통합생산에서 추구하고 있는 기본 모델은 그림 1과 같으며 이 중에서 네트워크를 구성하여 통신을 하는 상부구조에 대한 내용은 여러형태로 많은 연구가 행하여지고 있는 반면에 각종 기기와 직접 통신을 시도하는 하부구조에 대한 연구는 빈약한 형편이라 할 수 있다. 하부구조에서 가장 중요한 역할을 차지하는 부분은 워크스테이션 레벨(Workstation Level)로서 하부구조의 모든 작업은 이 워크스테이션에서 행해지게 된다. 그림 1에서 정의한 워크스테이션은 서너가지의 단위기기들로 구성되어 있으며 이들 단위기기들을 효율적으로 통합하여 제어하는 기능을 맡고 있는데 상부구조의 셀레벨과 연결되어 있어서 상부구조에서 내려오는 생산에 필요한 가공 정보들을 받을 수 있게되어 있으며, 이들 정보를 토대로 제품을 가공하도록 각종 기기들을 제어하게 된다. 하지만 앞에서 언급하였듯이 통신의 문제가 대두되기 때문에 이들을 소프트웨어적으로 처리를 해야 한다.

본 연구에서는 워크스테이션을 정의하는데 있어서 그림 1의 워크스테이션 레벨을 참조로 하여 그림 2와 같이 정의 하였으며 이들 기능들을 효과적으로 연결하여 워크스테이션 제어기 통신 소프트웨어를 개발 하려한다. 현재 Robot나 NC/CNC등과 같은 이종기기간의 소프트웨어적인 통합 및 제어를 위한 노력은 앞서서도 언급하였듯이 극히 미미하여 이를 극복하고자 그림 2와 같은 워크스테이션의 전략의 수립 및 이에 따른 통신 소프트웨어를 개발 한다. 그림 2에서 정의한 워크스테이션의 각각의 기능들을 설명하면 다음과 같다. " " "

- (1) 그림 2의 워크스테이션 영역(Workstation Domain)은 다중작업 오퍼레이팅 시스템(Multi-Task Operating System, 예를 들어 Unix, VMS, OS/2)이며 각 블럭은 하나의 독립된 프로그램으로서 오퍼레이팅 시스템내의 하나의 태

스크(Task)이다. 여기에서 보여진 워크스테이션 영역은 그림 1의 컴퓨터 통합생산의 구조에서 정의한 워크스테이션 레벨중 NC/CNC 부분의 것으로서 검사 워크스테이션(Inspection Workstation) 이라든지 재료 관리 워크스테이션(Material Handling Workstation) 등과같은 워크스테이션으로의 전이를 위한 하나의 모델 역할을 한다.

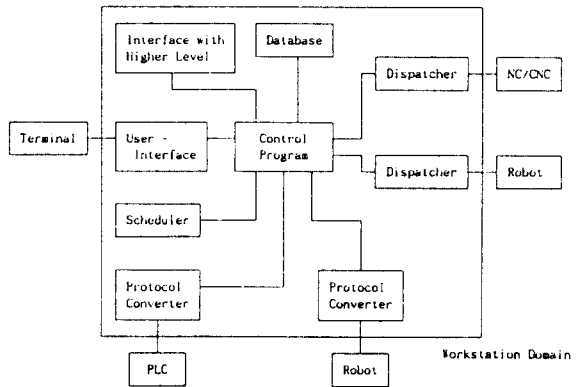


Fig. 2 Workstation Configuration

- (2) 하나의 태스크는 직렬포트(Serial Port, RS232C) 등이나 따로 정의한 조그만 통신 방법을 통하여 공장내의 기기(Machine Tool)와의 통신을 맡는다. 이때 각각의 태스크에 연결되는 케이블(Cable)들은 주어진 태스크에 맞도록 케이블을 사용한다.
- (3) 스케줄 태스크(Scheduler Task)는 워크스테이션 영역의 작업계획(Job Schedule)을 담당한다. 작업계획은 본 워크스테이션 제어기(Workstation Controller)의 주요 태스크로서 워크스테이션 제어기가 실행하는 작업에 대한 일정계획에 맞추어서 필요한 시기에 요구되는 작업을 실행하는 역할을 맡는다. 스케줄 태스크를 위해서는 일정에 필요한 내용을 담은 화일(File)이 데이터 베이스(Database)에 스케줄 데이터 화일의 형태로 존재 해야 한다.
- (4) 각 태스크간의 통신은 오퍼레이팅 시스템에 종속적인 메시지 방법(Message Scheme)을 사용하도록 한다. 본 소프트웨어에서는 OS/2에서 제공하는 프리젠테이션 매니저(Presentation

Manager)라는 윈도우(Window) 제어 프로그램을 사용하였는데 각 태스크간의 통신은 프리젠테이션 매니저에서 제공하는 메뉴 시스템에 따라서 작업을 행하도록 하였다.

- (5) 제어 프로그램(Control Program)은 전반적인 태스크들의 조정 및 통제를 담당한다. 각각의 태스크들은 부 프로그램(Sub Program)으로서 제어 프로그램에서의 선택에 따라 작업을 시작하도록 되어있다.
- (6) 사용자 인터페이스 태스크(User Interface Task)는 터미널에 메뉴나 그래픽 도구(Graphic Facility)를 통해 각 기기와 일정한 상태를 알려준다. 본 소프트웨어에서는 OS/2에서 제공하는 프리젠테이션 매니저라는 윈도우 제어 프로그램을 이용하여 그래픽처리를 하였다.
- (7) 프로토콜 변환기 태스크(Protocol Converter Task)는 프로토콜이 있는 기기에 대한 통신 및 감시(Monitoring)를 하는데 이에대한 결과는 메뉴로 보여지게 된다. 여기에서는 기기(Machine)에서 요구하는 방식대로 요구신호(Request Signal)와 확인신호(Acknowledge Signal)를 주고 받으며 소프트웨어적으로 통신을 하도록 프로그램을 작성하였다.
- (8) 디스패처 태스크(Dispatcher Task)는 프로토콜이 없이 단지 정보를 보내고 받을 수 있는 기기에 사용되는데 이 경우 종래의 DNC 개념과 비슷하다고 보겠다.
- (9) 데이터 베이스(Database)에는 각각의 기기에 필요한 NC 프로그램 데이터등의 화일이나 작업계획(Job Schedule)등이 저장된다. 여기서 NC 프로그램들은 CAD/CAM 시스템에 의해서 생성된 화일이다.
- (10) 상부구조 인터페이스 태스크(Higher Level Interface Task)는 셀 레벨과의 연결(Interface)을 담당한다. 이들 연결을 통해 상부구조에서 내려오는 정보를 유효적절하게 워크스테이션 제어기에 적용하게 된다.

3.2 통신

컴퓨터 통합생산의 하부구조에서 가장 많이 논의되고 있는 내용은 통신의 어려움이 있는데 워크스테이션 레벨에서의 통신은 두가지로 분류한다. 그림 1에서 보여지

듯이 워크스테이션 레벨은 상부구조인 셀 레벨과 하부구조인 장비 레벨의 사이에 있기때문에 네트워크와도 통신을 해야하고 각종 기기들과도 통신을 해야한다. 따라서 컴퓨터 통합생산 구조중 가장 중요한 위치에 있다고 할 수 있겠다.

상부구조인 셀 레벨과의 통신은 MAP 혹은 Ethernet 등의 LAN 방식을 이용할 수 있는데 네트워크간의 통신이므로 네트워크 시스템에서 정해진 통신 방법을 사용하여 상부구조와의 통신을 시도한다.

하부구조인 장비 레벨과의 통신은 프로토콜 변환기(Protocol Converter)라든가 디스패처(Dispatcher) 등의 통신 소프트웨어를 제작하여 컴퓨터상의 직렬 포트(Serial Port, RS232C)와 기기의 직렬 포트를 연결하는 비동기 통신(Asynchronous Communication)을 시도한다. 직렬로 포트를 이용한 통신은 그 구조가 간단할 뿐만아니라 대다수의 컴퓨터와 기기들에 장착되어있어 이를 이용한 통신이 용이하기 때문에 많은 분야에서 사용되고 있으며 본 연구에서도 이를 이용한 통신을 구현한다.¹²⁾ 한편 통신에 사용되는 신호는 문자(Character)를 사용하며 대표적인 예가 아스키 문자(American Standard Code for Information Interchange, ASCII)로서 이들 아스키 문자를 2진수 형태로 바꾸어서 통신속도, 통신시 에러검사 방법, 문자의 시작위치 등의 정보와 함께 정해진 방법따라 이들 신호들을 직렬로 나열하여 직렬 포트를 통해 통신을 시도한다.

4. 워크스테이션 운용에 관한 프로그램 설계 및 제작

그림 1과 그림 2를 통해 워크스테이션의 정의 및 기능들에 대해서는 이미 설명이 되어 있다. 이러한 워크스테이션을 위한 실질적인 워크스테이션 제어기 통신 소프트웨어 개발을 위해서 본 연구에서는 다음과 같은 방법으로 프로그램 개발을 시도하였다. 우선 퍼스널 컴퓨터에서 가장 많이 사용되는 오페레이팅 시스템인 DOS(Disk Operating System)에서 통신만을 담당하는 모델 프로그램을 제작하여 처음 터미널(Terminal)로 실험을 한 뒤 이를 실제 기기에 적용하여 프로그램의 가능성을 타진하였다.¹³⁾ 다음으로 워크스테이션 제어기의 성격상 단일 작업 오퍼레이팅 시스템(Single Task Operating System)인 DOS에서는 본 소프트웨어가 그 진가를 발휘하기 힘들기 때문에 DOS에서 제작한 모

델 프로그램을 바탕으로하여 다중작업 오퍼레이팅 시스템(Multi-Task Operating System)으로의 전이가 필수적이며 이를 위해 일반 퍼스널 컴퓨터 계열에서 다중작업이 가능하도록 고안된 오퍼레이팅 시스템인 OS/2 (Operating System/2)를 두번째로 선택하여 모델 프로그램을 확장 시켰다.¹⁴ 이들 프로그램을 위해 하드웨어(Hardware)로는 IBM Personal System/55를 사용하였으며, 프로그램 언어로는 비교적 통신을 제어하기 용이하고, 다른 컴퓨터 시스템으로의 전이가 쉬운언어인 'C' 언어를 이용하여 프로그램을 제작하였다. 한편 본 워크스테이션 제어기 소프트웨어를 보다 일반화시키기 위해서는 UNIX 시스템등으로의 전이가 필요하며 현재에는 UNIX 시스템등으로의 전이에 대한 가능성을 준비해놓고 있다.

4-1. 사용자 인터페이스와 윈도우 제어 프로그램

워크스테이션 제어기 프로그램은 그림 2에서 정의한 워크스테이션 영역에 보다 쉽게 접근하기 위해서 윈도우 제어 프로그램과 통신 프로그램의 두 부분으로 분류되어 개발 되었다. 즉 그림 2에서 정의한 터미널(Terminal)은 그림 1의 워크스테이션 레벨을 위해 사용되는 컴퓨터의 터미널을 의미하여 이 터미널상에 나타나는 모든 내용은 윈도우 제어프로그램이 담당하고 이들 내용을 실행하는 부분으로 통신 프로그램이 담당하게 된다. 따라서 이들의 기능을 효과적으로 연결하기 위해서는 터미널과 연결된 사용자 인터페이스(User Interface) 태스크와 이 사용자 인터페이스 태스크와 연결된 제어 프로그램(Control Program) 태스크를 적절히 조화 시키는데 있다. 사용자 인터페이스 태스크는 터미널에 메뉴나 그래픽 도구들을 통해 워크스테이션 제어기의 전반적인 상황 또는 각종 기기와 일정의 상태등을 알려준다. 제어 프로그램 태스크는 사용자 인터페이스의 메뉴와 연결되며 전반적인 상황 또는 각종 기기와 일정의 상태등을 알려준다. 제어 프로그램 태스크는 사용자 인터페이스의 메뉴와 연결되며 전반적인 태스크들위 조정 및 통제를 담당하는데 각각의 태스크들은 부 프로그램(Sub-Program)화 되어 있어서 제어 프로그램의 선택에 따라 적절한 작업을 시작하며 이들의 관계는 그림 2의 Control Program 부분과 연결되는 각각의 태스크를 주시하면 된다. 즉 그림 2의 흐름은 터미널 상에 나타나는 사용자 인터페이스의 메뉴로 부터 사용자가 특정한 작업(예를 들면 스케줄, 디스페처, ...)을 선택하면 이들

메뉴 시스템과 연결된 제어 프로그램이 사용자가 원하는 작업을 실행하도록 부 프로그램화 되어있는 프로그램을 연결하여(Call) 적절한 동작을 하게된다.

사용자 인터페이스 태스크를 위해서 본 소프트웨어에서는 OS/2에서 제공하는 프리젠테이션 매니저(Presentation Manager)를 사용한다. 프리젠테이션 매니저는 OS/2에서 그래픽 기능을 담당하도록 고안된 도구(Tool)로서 컴퓨터의 모니터상에 여러개의 화면으로 구성된 윈도우를 그려낼 수 있으며 이들 윈도우에 메뉴 시스템을 적용하여 사용자가 원하는 작업을 실행할 수 있도록 도와준다. 각각의 윈도우를 구성하고 이들 윈도우의 배경에 그림을 그리는 한편 글자 처리를 한다든지, 메뉴를 만든다든지 하는 일련의 그래픽 작업들을 'C'언어로 프로그램하여 실행시킬 수 있으며 실행된 결과는 컴퓨터의 모니터상에 윈도우 형태로 나타나게 된다.¹⁵

4-2. 프로그램의 구조

그림 2에서 정의한 워크스테이션을 바탕으로하여 개발된 실질적인 워크스테이션 제어기 소프트웨어는 세가지 부분으로 구별된다. 첫번째는 스케줄 태스크를 담당하는 스케줄러(Scheduler)모듈이 그것이며 워크스테이션 제어기 소프트웨어의 주된동작을 맡아 실행하게 된다. 두 번째는 프로토콜 변환기 태스크와 디스페처 태스크를 담당하는 선택(Selector)모듈이다. 스케줄러 모듈과는 달리 개별적으로 각종 기기들과의 직접적인 연결을 원하고자할 때 이 선택 모듈을 통해 프로토콜 변환기 프로그램이나 디스페처 프로그램을 실행할 수 있다. 세번째는 상부구조 인터페이스 태스크를 담당하는 상부구조 인터페이스 모듈로서 LAN등의 방법을 이용하여 상부구조의 정보를 받아내는 역할을 맡는다. 그러나 이 모듈은 아직 준비되어있지 않으며 본 워크스테이션 제어기를 본격적으로 가동시키기 시작할 때 상부구조와의 연결을 꾀할 것이다. 이들 각각의 기능을 윈도우 형태에 적용하여 구체적인 모습을 보이면 그림 3과 같이 나타낼 수 있다.

한편 그림 3을 바탕으로하여 각각의 태스크들이 가져야 할 기능들을 살펴보면 다음과 같다. 우선 스케줄 태스크는 스케줄 화일의 선택, 각종 기기와의 연결 상태의 변경, 실행등의 기능들을 보유하고 있다. 선택 모듈중의 프로토콜 변환기 태스크는 프로토콜 변환기 태스크를 위하여 준비된 기기가 제공하는 기능에 따라서

Robot화일선택, 레지스터 조회, 온라인 프로그램 전송, 오프라인 프로그램 전송등의 기능들을 가지고 있다. 선택 모듈중의 디스패처 태스크를 위해서는 NC화일의 선택, 데이터 전송등의 기능을 가지고 있다. 이들의 내용은 표 1에서 설명한다.

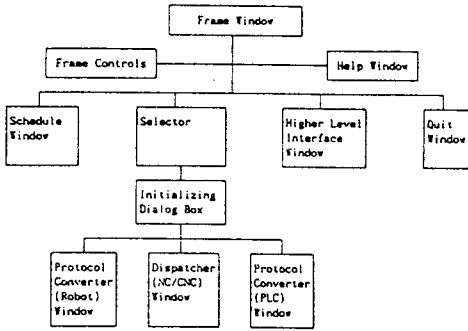


Fig. 3 Window Structure of a Workstation Controller

Table 1. Functions of a Workstation Controller

Task	Function	기능
Schedule	Select Schedule File Change Setting Execute	스케줄 화일의 선택 각종 기기들의 연결상태의 변경 선택된 스케줄 화일의 실행
Select (Protocol Converter : Robot)	Select Robot File Register Inquiry Loading Data Register Online Program Transfer Register Transfer Based on S Code Offline Program Transfer (Robot to Computer) Offline Program Transfer (Computer to Robot)	Robot NC 데이터 화일의 선택 Robot의 레지스터 내용 조사 Robot로 레지스터를 전송 Robot로 NC 프로그램 전송 S Code에 관련된 레지스터 전송 Robot에서 컴퓨터로 NC 데이터 화일을 전송 컴퓨터에서 Robot로 NC 데이터 화일을 전송
Select (Dispatcher : NC/CNC)	Select NC File Send Receive	NC/CNC의 데이터 화일의 선택 NC로 데이터 화일을 전송 컴퓨터로 데이터 화일을 전송
Select (Protocol Converter : PLC)		PLC에서 제공하는 기능에 따라 구성될 예정 (현재 준비중)
Higher Level Interface		LAN등의 연결기능에 따라 구성 (현재 준비중)

5. 워크스테이션의 통신기능에 관한 프로그램 설계 및 제작

워크스테이션의 통신 프로그램은 프로그램의 개발 환경에 좌우된다. 본 연구를 위해서 처음에는 DOS에서 통신만을 담당하는 프로그램을 개발 하였는데 DOS상에서의 통신 프로그램은 DOS에서 지원하는 바이어스 (Basic Input/Output System, BIOS)라는 입출력에 관한 함수를 사용하여 프로그램을 작성하였다. BIOS는 DOS라는 오퍼레이팅 시스템이 준비한 기능으로서 모니터 (Monitor)나 키보드 (Keyboard)등의 조작, 프린터

(Printer)의 조작, 보조 입출력 장치 (Auxiliary Device, AUX)의 조작등을 가능하게 하며 DOS에서만 존재한다. 따라서 다른 오퍼레이팅 시스템에서도 DOS의 BIOS와 같은 기능을 하는 입출력 함수가 존재하며, 이들 기능들은 프로그램을 개발하는 환경에 따라 선택적으로 사용된다. 본 워크스테이션 제어기 통신 소프트웨어는 이들 통신을 위해 DOS에서의 BIOS와 같은 기능을 하며, OS/2에서 제공되는 장비입출력 제어함수 (Device I/O Control Function, DevIOctl)를 사용하여 프로그램을 제작하였다."

워크스테이션 제어기 소프트웨어를 실행하면 우선 먼저 그림 4와 같은 화면이 나타나게 된다. 화면의 우측에는 표 1에서 정의한 각각의 모듈을 실행시키기 위한 메뉴가 위치해 있는데 이들 메뉴를 이용하여 스케줄 모듈, 선택 모듈등을 실행시킬 수 있으며 이들 모듈을 실행시키면 새로운 윈도우가 전면으로 부상하면서 원하는 작업을 실행한다.

5-1. 스케줄 태스크

스케줄 태스크 (Scheduler Task)는 워크스테이션 영역의 작업계획을 담당한다. 따라서 워크스테이션 제어기의 핵심 부분이며 워크스테이션 제어기가 실행하는 작업에 대한 일정계획에 맞추어서 필요한 시기에 요구되는 작업을 실행하는 역할을 담당한다. 스케줄 태스크에서는 상부구조에서 전달된 정보나 사용자의 요구에 따라서 정의한 스케줄 화일이 데이터 베이스에 존재해야 하며 스케줄 태스크는 이들 화일을 선택하여 그 내용에 따라 작업을 실행하게 된다. 표 1에서 스케줄 태스크에 대한 구체적인 동작 기능들을 구성하였는데 이를 바탕으로

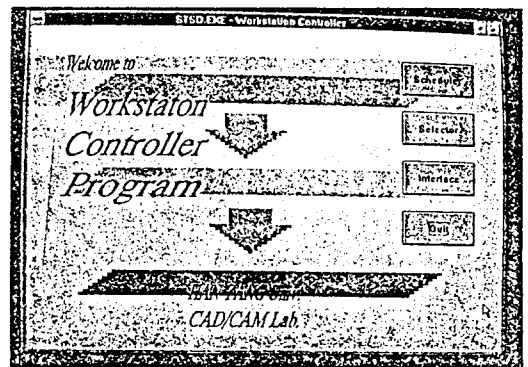


Fig. 4 Initial window of Workstation Controller software

하여 개발된 스케줄 태스크에는 세가지의 메뉴 항목이 있다. 그림 5에 스케줄 태스크의 초기 화면을 나타내었다.

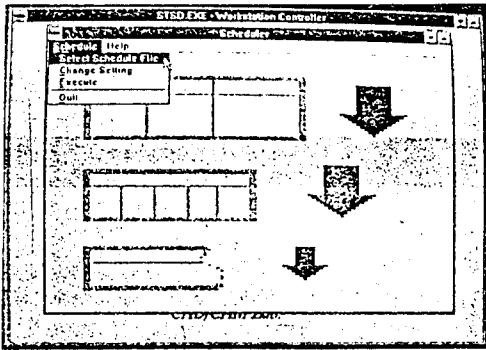


Fig. 5 Initial window of Scheduler module

5-2. 선택 태스크

선택 태스크(Selector Task)에서는 스케줄 태스크와는 달리 개별적으로 각종 기기들과의 직접적인 연결을 원하고자 할 때 실행하는 태스크로서 프로토콜 변환기 프로그램이나 디스페처 프로그램들을 별도로 실행시킬 수 있다. 현재 정의되어 있는 작업은 Cincinnati Milacron의 5축 머시닝 센터(Machining Center)와의 통신을 담당하고 있는 디스페처 프로그램과 FANUC Robot와 통신을 담당하고있는 프로토콜 변환기 프로그램이 준비되어 있다. PLC(Programmable Logic Controller)에 대한 프로그램은 PLC가 준비되어 있지 않기 때문에 현재는 가능성만 제시해 놓고 있는 상태이다. 그림 4의 주화면에서 선택 메뉴를 선택하면 프로토콜 변환기 프로그램을 실행할 것인지 디스페처 프로그램을 실행할 것인지를 선택하는 다이얼로그 박스(Dialog

Box)가 그림6처럼 나타난다. 이 다이얼로그 박스는 실행하려는 태스크, 통신을 시도하려는 직렬포트 그리고 통신에 필요한 프로토콜등을 선택하도록 도와주며 적절한 선택을 하면 제어 프로그램(Control Program)이 지정한 내용에 맞추어서 원하는 태스크를 실행하도록 한다. 각각의 태스크에 대한 설명은 다음과 같다. 태스크

5-2-1. 프로토콜 변환기 태스크

프로토콜 변환기 태스크(Protocol Converter Task)는 공장내에 있는 기기들 중에서 통신을 하기위한 프로토콜이 복잡하게 정의되어있는 기기들과의 통신 및 감시를 위한 태스크이다. 본 연구에서 사용되는 프로토콜 변환기 프로그램은 FANUC Robot AI 모델을 대상으로 작성되었는데 이 기기는 다소 복잡한 통신 프로토콜을 가지고 있다.

프로토콜 변환기 프로그램에 사용되는 통신 형태는 여러가지가 있으며 대개 전송하려는 데이터의 앞뒤에 특수 문자들을 첨가시켜 이들을 전송하는 방법을 사용하는데 FANUC Robot와의 통신 방법에도 이를 사용한다. FANUC Robot와의 통신시 여러 검사방법은 수직 여유도 검사로서, 앞에서 설명한 프로토콜과 함께 본 프로토콜 변환기 프로그램에서 사용되는 통신 형태는 요구 확인 방법(Request-Acknowledge Method)이라고 불리운다.

그림 7에 FANUC Robot를 위한 프로토콜 변환기 윈도우의 초기 화면을 나타내었다. 표 1에서도 정의하였듯이 FANUC Robot이 준비하고 있는 컴퓨터 제어 기능(Computer Control Function)에는 6가지의 종류가 있으며 이들의 내용은 그림 7에 메뉴화 되어있어서 이들 메뉴를 선택함으로써 원하는 작업을 행할 수 있다.

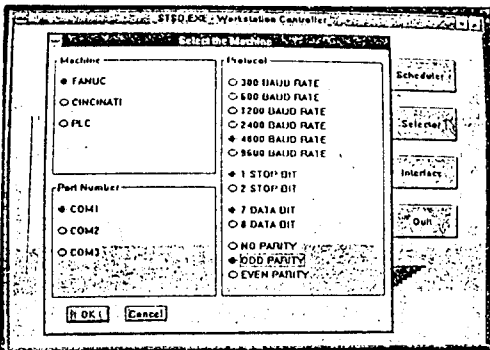


Fig. 6 Initial Dialog box of Selector module

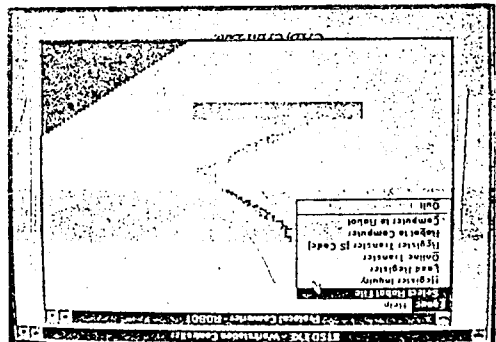


Fig. 7 Initial window of Protocol Converter task

5-2-2. 디스페처 태스크

디스페처 태스크(Dispatcher Task)는 공장내에 있는 기기들중에서 통신을 하기위한 프로토콜이 복잡한 과정이 없이 어느 일정한 조건만 충족되면 통신이 가능하도록 설계되어 있는 기기들과의 통신을 담당하는 태스크로서 종래의 DNC(Distributed Numerical Control) 시스템과 유사하다. 본 연구에서 사용되는 디스페처 프로그램은 Cincinnati Milacron의 5축 머시닝 센터(5 Axis Machining Center)를 대상으로 작성하였다.¹⁴⁾

Cincinnati 머시닝 센터와 컴퓨터와의 통신 프로그램에는 문자 흐름 제어규약인 XON/XOFF를 사용하며 송수신시에 XOFF 신호가 검출되면 XON 신호가 검출될 때까지 대기하다가 XON 신호가 검출되면 다시 송수신을 재개한다. 이 방법은 대형 컴퓨터의 오퍼레이팅 시스템에서 사용하는 시분할 시스템(Time Shearing System)에서도 통신을 가능하게 하는 방법으로서 통신시 필요에 의해 이들 XON/XOFF 제어문자를 적절히 사용하여 통신을 시도하게 된다.

그림 8에 Cincinnati 머시닝 센터를 위한 디스페처 윈도우의 초기 화면을 나타내었다. 표 1에서 정의하였듯이 Cincinnati 머시닝 센터가 준비하고 있는 기능은 두가지로서 하나는 컴퓨터에서 머시닝 센터의 메모리 안으로 NC프로그램 데이터 파일을 전송(Send)하는 기능과 머시닝 센터의 메모리에서 컴퓨터로 NC 프로그램 데이터 파일을 전송(Receive)하는 기능이 있다.

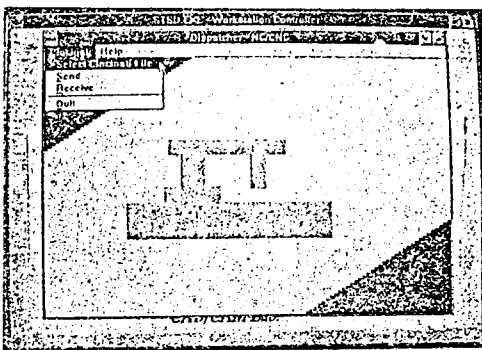


Fig. 8 Initial window of Dispatcher task

6. 개발된 소프트웨어의 응용

현재 개발된 소프트웨어는 공장내부중 실지 기기들과 연결되어 실질적인 자동화를 위한 기초 단계를 충실히

수행하고 있다. 즉, 컴퓨터 통합생산의 상위 레벨에서 계획된 생산정보등을 전달 받아 워크스테이션 제어기에 연결되어 있는 이종기기들을 제어하는 전체 생산구조를 충분히 반영함과 아울러 각각의 기기들의 상황 감지 및 제어에 대한 구체적인 예를 가시화 할 수 있다. 따라서 생산에 필요한 주문으로부터 시작하여 제조 계획 수립, 일정 조정, 생산 그리고 사후관리등의 일련의 과정을 이루는 전체 컴퓨터 통합생산 시스템중 하나의 작은 연결고리 역할을 충분히 수행해 낼 수 있다.

개발된 소프트웨어는 현재 한국과학기술연구원(KIST)에 있는 이종기기들을 대상으로 실험을 거쳤는데 디스페처 태스크를 위한 프로그램은 Cincinnati Milacron 머시닝 센터와 통신 실험을 하였고, 프로토콜 변환기 태스크를 위한 프로그램은 FANUC Robot와의 통신을 시도하였다.

한편, KIST에서는 컴퓨터 통합생산을 위한 모델 플랜트(Model Plant)의 건립을 추진하고 있는데 전체 플랜트를 통제하기 위한 플랜트 통제 시스템을 개발하고 있으며 본 연구에서 개발된 워크스테이션 제어기 통신 소프트웨어를 플랜트 통제 시스템에 접목시키기 위한 준비도 아울러 이루어 지고 있다.¹⁵⁾

7. 결 론

본 연구에서는 컴퓨터 통합생산 시스템중에서 상부구조와 하부구조를 연결하여 생산에 필요한 정보의 흐름에 따라 실질적인 작업을 실행하기 위한 기초역할을 담당하는 워크스테이션 제어기 통신 소프트웨어를 개발하였다. 컴퓨터 통합생산 시스템의 구조에 적당한 워크스테이션을 정의하였고, 이 정의에 따라서 몇가지의 모듈화된 소프트웨어를 작성하였다. 이들 소프트웨어의 실험에는 현재 한국과학기술연구원(KIST)에 있는 FANUC Robot와 Cincinnati 머시닝 센터와의 직접적인 연결을 통한 실험을 성공적으로 마쳤으며 이들을 통합한 시범적인 워크스테이션 제어기의 검증이 이루어 졌다. 한편 본 소프트웨어에는 공장내에 존재하는 각종 기기들중 복잡한 프로토콜을 가진 기기들을 위한 프로토콜 제어기 프로그램과 프로토콜이 단순한 기기들을 위한 디스페처 프로그램도 아울러 개발되어 있는데 새로운 기기가 첨부될 경우 이에 맞는 통신 프로그램만 제작하면 사용이 가능하도록 되어있다.

본 연구의 결과는 커다란 공장 자동화 시스템의 한

기능을 담당할 수 있다. 즉 워크스테이션 제어기 통신 소프트웨어를 개발함으로써 제조 공장(Manufacturing Shop Floor)의 기기간의 통신 및 제어의 한 예를 제시할 수 있고, 아울러 공장의 워크스테이션 내부의 조직적인 자동화 방법이 가시적으로 보여질 수 있다.

컴퓨터 통합생산 시스템중 실질적인 측면에서 가장 중요한 부분을 차지하는 워크스테이션 제어기를 위한 소프트웨어는 현재까지는 그 가능성을 위한 기초적인 수준에 머물러 있다. 하지만 이를 바탕으로하여 앞으로 확대 되어야 할 부분은 워크스테이션 제어기의 중심 과제인 스케줄 태스크와 상부구조와의 연결을 담당하는 상부구조 인터페이스 태스크등이 좀더 개발되어야 한다. 스케줄 태스크는 워크스테이션 제어기를 위한 데이터 베이스 구축과 아울러 스케줄 작업에 따른 작업 일정등에 대한 보다 구체적이고 전문적인 정의가 필요하고, 이 정의에 따라 적합한 프로그램을 개발하여야 한다.

상부구조 인터페이스 태스크를 위해서는 서로 다른 오퍼레이팅 시스템간의 통신에 대한 고찰과 아울러 이를 제어하기 위한 프로그램을 개발하여야 한다.

[참고 문헌]

1. Simpson J. A., Hoken R. J., Albus J. s., "The Automated Manufacturing Research Facility of the National Bureau of Standards", Journal of Manufacturing Systems, Vol. 1, No. 1, pp 17-31, 1982.
2. IMS 國際共同研究プログラムの概要, 通商産業省, 機械情報産業局, 産業機械課, 1990.
3. ESPRIT 809 Workshop, 1990.
4. '89 생산 기술 및 자동화 심포지움, KIST, 1989.
5. Albus J. S. and Abdol Saleh, "A Multi-Level/Multi-Layer Architecture for Intelligent Shopfloor Control", International Journal of Computer Integrated Manufacturing, Vol. 3, No. 1, pp60-70, 1990.
6. Mc Namara J. E., Local Area Networks and Introduction to the Technology, Digital Press, Burlington, Massachusetts, 1985.
7. Stacy A. H., The MAP Book : An Introduction to Industrial Networking, Industrial Networking Incorporated, Santa Clara, California, 1987.
8. Groover M. P., Zimmers E. W., CAD/CAM Computer-Aided Design and Manufacturing, Prentice Hall Inc., 1984.
9. Park G. J., Lavelle M. R, and Renda R. B., "Software Development for a Workstation Controller in a CIM System", Spring National Design Conference, Chicago, Illinois, March 7-10, 1988.
10. Kowalczyk J., Park G. J, and Renda R, B., "Management Scheme of a Workstation Controller in a Flexible Manufacturing System", Computers in Industry, Vol. 11, pp23-29, 1988.
11. Park G. J., Lavelle M. R, and Renda R, B., "Manufacturing Workstations and Cell Controllers in a CIM Application", Journal of Manufacturing Systems, Vol. 8, No. 2, pp161-165, 1990.
12. RS232C 인터페이스 사용법, 가남사, 1990.
13. 박경진, 도성희, 이원배, "GMP 출연과제 : 금형 공장에서의 이중기기간 통신에 관한 연구", KIST Project 최종보고서, 한양대학교, 1990.
14. 박경진, "금형공장 자동화를 위한 Workstation Controller 소프트웨어 개발에 관한 연구", KIST 국책 Proposal, 한양대학교, 1990.
15. Alan Southerton, Programmer's Guide to Presentation Manager, Addison-Wesley Publishing Company, Inc., 1989.
16. Thuyen Nguyen and Robot Moskal, Advanced Programmer's Guide to OS/2, Brady Books, 1989.
17. FANUC Robot A Model 1 Communications Manual, FANUC Company, 2nd Ed., 1984.
18. Acramatic 950 Serial Data Interface Application Manual, Cincinnati Milacron Marketing Company, 1989.
19. 강무진 外 5名, "컴퓨터를 이용한 통합생산 자동화 기술-플랜트 통제 시스템 개발", KIST 국책 3차년도 연차보고서, 한국과학기술연구원, 1991.